



TECHNICAL UNIVERSITY OF KOŠICE
Faculty of Electrical Engineering and Informatics



Electrical Engineering and Informatics XI

Proceedings of
the Faculty of Electrical Engineering and Informatics
of the Technical University of Košice

ISBN 978-80-553-3585-8

Electrical Engineering and Informatics XI
Proceedings of the Faculty of Electrical Engineering and Informatics
of the Technical University of Košice

Published by: Faculty of Electrical Engineering and Informatics
Technical University of Košice
Letná 9, 04200 Košice, Slovak Republic

Date of publication: August 2020 Language: English, Slovak
Printing: 50 pieces CD Pages: 520

Editorial board chairman: prof. Ing. Alena Pietriková, CSc.

Proceedings reviewers: prof. Ing. Dušan Kocur, CSc.
prof. Ing. Dušan Levický, CSc.
prof. Ing. Stanislav Marchevský, CSc.
prof. Ing. Ján Paralič PhD.
prof. Ing. Daniela Perduková, PhD.
prof. Ing. Alena Pietriková, CSc.
prof. Ing. Peter Sinčák CSc.
prof. Ing. Ján Šaliga, CSc.
Dr.h.c. prof. RNDr. Ing. Ján Turán, DrSc.
prof. Ing. Iveta Zolotová CSc.
assoc. prof. Ing. Lubomír Beňa, PhD.
assoc. prof. Ing. Marek Bundzel PhD.
assoc. prof. Ing. Peter Butka PhD.
assoc. prof. Ing. Lubomír Doboš, PhD.
assoc. prof. Dr. Ing. Bystrík Dolník
assoc. prof. Ing. Miloš Drutarovský, PhD.
assoc. prof. Ing. Jaroslav Džmura, PhD.
assoc. prof. Ing. Mária Gamcová, PhD.
assoc. prof. Ing. Ján Gamec, PhD.
assoc. prof. Ing. Ján Jadlovský, CSc.
assoc. prof. Ing. Anna Jadlovská, PhD.
assoc. prof. Ing. Ján Kaňuch, PhD.
assoc. prof. Ing. Irida Kolcunová, PhD.
assoc. prof. Ing. Milan Lacko, PhD.
assoc. prof. Ing. Marián Mach CSc.
assoc. prof. Ing. Kristína Machová PhD
assoc. prof. Ing. Dušan Medveď, PhD.
assoc. prof. Ing. Ján Papaž, PhD.
assoc. prof. Ing. Jaroslav Petráš, PhD.
assoc. prof. Dr. Ing. Ján Vaščák
assoc. prof. Ing. Jaroslava Žilková, PhD.

Editors: prof. Ing. Alena Pietriková, CSc.
Ing. Emília Pietriková, PhD.

Contents

Ján Bačík, Peter Tkáč <i>Vývoj univerzálnej mobilnej robotickej platformy</i>	7
Jozef Frivaldský, Marián Mach <i>Porovnanie výberových stratégií horolezeckého algoritmu v rámci domény TSP</i>	10
Róbert Baláž, Marián Mach <i>Plánovanie trajektórie pohybu mobilného robota pomocou SMT</i>	16
Alžbeta Sklenáriková, Branislav Madoš, Anton Baláž <i>FPGA Hardvérový Akcelerátor Minimalizácie Booleovskej Funkcie</i>	21
Jozef Bučko, Matúš Pleva <i>Využitie fúzie modalít pri biometrickej autentifikácii užívateľa</i>	27
Jakub Oravec, Ľuboš Ovseník, Ján Turán <i>Vylepšenie kvantizácie pre obrazové šifrovacie algoritmy</i>	32
Peter Girovský, Ján Kaňuch <i>DC menič pre napájanie univerzálneho motora</i>	37
Martin Čontofalský, Dávid Cymbalák <i>Komplexný systém pre rezerváciu videokonferenčných miestností na báze QR kódov</i>	42
Jakub Urbanský, Daniel Pál, Vladimír Krištof <i>Short-circuit currents and their calculation</i>	47
Maksym Oliinyk, Jaroslav Džmura <i>Machine Learning in the Energy Sector</i>	52
Kristián Goliaš, Ján Genči, Michal Vrábel <i>Rozpoznávanie vzorov atmosférických spršok v dátach EUSO experimentov</i>	58
Marek Skyba, Anna Birková, Ján Genči <i>Softvérový pH normalizátor</i>	64
Ivan Bernát, Pavol Bobík, Ján Genči <i>Automatizácia výpočtov sekundárnych častíc kozmického žiarenia v softvéri Corsika</i>	70
Martin Vaško, Pavol Bobík, Ján Genči <i>Refaktorizácia kódov pre výpočet trajektórii častíc kozmického žiarenia v magnetosfére Zeme</i>	76
Dominik Osvald, Ján Genči <i>Prepracovanie Inter-Process Communication zadania z predmetu Operačné systémy</i>	82
Veronika Rúšová, Matúš Pleva <i>Rozšírenie code-switching databázy pre účely automatického rozpoznávania bilingválnej reči</i>	87
Erik Klembara, Marián Mach <i>Robotický operačný systém pre vytváranie hier jedného hráča</i>	93
Richard Nezník, Ján Papaj <i>Výber bezpečných prenosových uzlov pre hybridnú MANET-DTN siet</i>	98

Ján Papaj, Gabriela Vašková <i>Teória hier a smerovanie v hybridnej MANET-DTN sieti</i>	104
Iveta Gladišová, Ján Mihalík <i>Nilssonov-Heydenov algoritmus aktívnych obrysov</i>	109
Zuzana Gadušová, Dušan Medveď <i>Dopad havárií jadrových elektrární na vývoj jadrovej energetiky na Slovensku a vo svete</i>	115
Jana Handriková, Peter Šulaj <i>Ensuring cyber security in the Internet of Things environment</i>	120
Patrik Kancír, Dušan Medveď <i>Návrh alternatívneho tepelného zdroja pre kláštor Krásny Brod</i>	126
Maroš Baumgartner, Renát Haluška, Peter Šulaj, Jana Handriková <i>Návrh ad-hoc Siete pre Prenos Videotokov z Nepilotovaných Lietajúcich Prostriedkov</i>	132
Ján Mihalík <i>Štandardný videokodek H.265/HEVC</i>	138
Miroslav Tomčo, Dušan Medveď <i>Výpočet energetickej bilancie horského hotela v Poráčskej doline s využitím obnoviteľných zdrojov energie</i>	142
Kinga Nagyová, Slavomír Šimoňák <i>Application of Educational Computer Games in Algorithms and Data Structures Courses</i>	148
Matúš Kokinda, Zdeněk Havlice, Ján Trembulák, Iveta Kindernayová <i>Applying Automated Testing Tools for Desktop Applications with Graphical Interface</i>	154
František Ferenčík, Zdeněk Havlice <i>Requirements Metamodeling, Modeling, Verification and Validation</i>	160
Peter Havran, Roman Cimbala <i>Test tepelne namáhaného progresívneho kvapalného dielektrika v časovej doméne</i>	167
Peter Havran, Roman Cimbala <i>Dielektrická spektroskopia online tepelne namáhaného kvapalného dielektrika</i>	173
Peter Ličko, Marián Mach <i>Hrací agent na báze prehľadávania stromov</i>	179
Martin Chlebovec, Miloš Drutarovský <i>Kryptografické zabezpečenie a bezpečná aktualizácia firmvéru v IoT aplikáciách na báze ESP32</i> .	184
Dávid Karabaš, Ondrej Kainz, Miroslav Michalko, Rastislav Petija, František Jakab <i>Enhanced Smart Monitoring of Water Consumption</i>	190
Monika Kužmová, Slavomír Šimoňák <i>Use of non-traditional teaching methods in the subject Data Structures and Algorithms</i>	196
Kristína Machová, Lucia Vargová <i>Strojové Učenie v Identifikácii Klamlivých Správ v Online Diskusiách</i>	202
Teodor Kiseľ, Dušan Medveď <i>Analýza správania sa prosumerov v elektrickej sieti</i>	207

Vladimír Kohan, Zsolt Čonka, Michal Kolcun <i>Vývoj HVDC systémov</i>	213
Vladimír Kohan, Michal Kolcun <i>Vplyv dynamických javov na stabilitu elektrizačnej sústavy s pripojenými OZE v elektrizačnej sústave</i>	220
Martin Halko, Dušan Medveď <i>Návrh výpočtovej metodiky nasadenia inteligentných zariadení v budovách</i>	229
Jozef Biroš, Marián Mach <i>Porovnanie konvergencie včelami inšpirovaných algoritmov rodín ABC, BSO a BA</i>	235
Anton Záborský, Ľuboš Ovseník, Ján Turán, Tomáš Huszaník <i>Návrh a realizácia experimentálneho modelu PON a FTTx sietí</i>	241
Tomáš Huszaník, Ľuboš Ovseník, Ján Turán, Anton Záborský, Norbert Zdravecký <i>Porovnanie charakteristík optických vláknových zosilňovačov vo vysokokapacitnom DWDM systéme</i>	248
Samuel Grobár, Stanislav Marchevský, Peter Šulaj, Renát Haluška <i>Spracovanie a prenos videotokov z videosenzorov z dronov</i>	256
Tomáš Juščík, Ján Vaščák <i>Ovládanie robota v prostredí virtuálnej reality</i>	262
Ján Vaščák, Richard Rusňák <i>Vizuálna interakcia s prostredím inteligentného priestoru</i>	268
Kristína Machová, Peter Smolnický <i>Vplyv ironických príspevkov na presnosť analýzy sentimentu</i>	274
Ján Mihalík, Iveta Gladišová <i>Deskripcia regionálneho tvaru pomocou Zernikeho momentov</i>	278
Daniela Perduková, Pavol Fedor <i>Fuzzy approach to Dynamic Systems Control</i>	283
Kristína Machová, Michal Janok <i>Detekcia kritických stavov pomocou analýzy sentimentu</i>	291
Renát Haluška, Peter Šulaj, Ľuboš Ovseník <i>Example of the robotic receptionist</i>	296
Renát Haluška, Maroš Lapčák, Ľuboš Ovseník <i>Example of Hard Switching for Hybrid FSO/RF System</i>	299
Jozef Humeník <i>Bezpečnosť a riadenie inteligentných sietí v elektroenergetike</i>	302
Ján Kaňuch, Peter Girovský <i>Diskový krokový motor s dvojitém rotorom s permanentnými magnetmi</i>	307
Radka Kačmáreková, František Babič <i>Úspora Nákladov na Prepravu Prostredníctvom Optimalizácie Trasy</i>	316
Patrik Tužák, František Babič <i>Párovanie Duplicitných a Takmer Identických Obrázkov Pomocou Percepčného Hashovania</i>	320

Peter Daňko, František Babič <i>Vývoj Inovativného Pluginu pre Redakčný Systém</i>	324
Branislav Vangor, Dušan Medved <i>Porovnanie zariadení pre akumuláciu elektrickej energie pre domácnosť</i>	328
Ján Vaščák <i>Simulované prihrávanie lopty v robotickom futbale</i>	334
Jana Jakobová, František Babič, Ľudmila Pusztová <i>Podpora Kupónového Marketingu prostredníctvom Vhodných Analytických Metód</i>	339
Diana Klimonová, František Babič, Ľudmila Pusztová <i>Riešenie Problému Odchodovosti Zákazníkov pomocou Dátovej Analytiky</i>	344
Kristián Hai Le Thanh, Viera Maslej Krešňáková, Peter Butka <i>Metódy hlbokého učenia pre klasifikáciu rádiových galaxií</i>	350
Peter Kulbaga, Viera Maslej Krešňáková, Peter Butka <i>Anotácia a detekcia lineárnych objektov v prehliadke oblohy</i>	355
Martin Harman, Viera Maslej Krešňáková, Peter Butka <i>Analýza dát v doméne astronomických pozorovaní</i>	360
Kristián Maťašovský, Viera Maslej Krešňáková, Peter Butka <i>Interaktívna vizualizácia dát zo satelitných pozorovaní</i>	365
Lukáš Humeník, Viera Maslej Krešňáková, Peter Butka <i>Metódy hlbokého učenia pre klasifikáciu zákrytových premenných hviezd</i>	369
Zuzana Dávidová, Viera Maslej Krešňáková, Martin Sarnovský <i>Metódy hlbokého učenia pre detekciu a klasifikáciu antisociálneho správania v online priestore</i> .	373
Nikola Štieberová, Ľubomír Beňa, Anastázia Margitová, Martin Kanálik <i>Výpočet parametrov týkajúcich sa dynamickej ampacity vonkajších vedení</i>	379
Juraj Biľanský, Milan Lacko <i>Návrh koncepcie nabíjacej a vybíjacej časti testera li-ion batériových článkov</i>	385
Michaela Tarhaničová, Marián Mach <i>Robotický operačný systém a jeho použitie v evolučnej robotike</i>	390
Vladimír Jurenka, Dušan Medved <i>Možnosti akumulácie tepelnej energie</i>	396
Peter Girovský <i>Porovnanie vplyvu defuzzifikačnej metódy na priebeh regulácie synchronného motora</i>	402
Michal Solanik, Pavol Bobík, Ján Genči <i>Paralelization of two-dimensional models of cosmic rays distribution in heliosphere</i>	407
Maksym Oliinyk, Jaroslav Džmura <i>The Impact of Electric Vehicles on Voltage Symmetry</i>	413
Ján Vaščák <i>Simulovanie synchronizácie činností robotov</i>	419
Daniel Pál, Ľubomír Beňa, Jakub Urbanský <i>Vplyv pripojenia nového zdroja energie na vybrané ukazovatele v elektrickej sieti</i>	423

Samuel Biroš, Matúš Sulír <i>Analysis of Questions and Answers on UX.StackExchange</i>	428
Natália Bavoľárová, Matúš Sulír <i>Brief Analysis of Game Development Stack Exchange</i>	434
Ján Vaščák <i>Návrh systému pre testovanie simulovania lokalizácie mobilných robotov</i>	440
Kamil Adamišín, Michal Budiš, Lukáš Hruška, Jakub Karľa <i>Deep reinforcement learning agents playing nine men's morris</i>	445
Tomáš Balluch, Štefan Korečko, Marián Hudák, Branislav Sobota <i>Smartfón ako periférne zariadenie pre zdieľané virtuálno-realistné prostredie LIRKIS G-CVE</i> ...	450
Kristína Matiková, Štefan Korečko, Marián Hudák, Branislav Sobota <i>Použitie MS HoloLens v zdieľanom virtuálno-realistnom prostredí LIRKIS G-CVE</i>	456
Michal Ivan, Štefan Korečko, Marián Hudák, Branislav Sobota <i>Rozšírenie možností komunikácie a správy klientov zdieľaného virtuálno-realistného prostredia LIRKIS G-CVE</i>	462
Kristína Machová, Matej Vasilko <i>Detekcia trolla typu oponent</i>	468
Martin Sarnovský, Michal Kolárik <i>Metódy adaptívneho učenia na dynamických dátach</i>	474
Martin Sarnovský, Júlia Jacková <i>Webová aplikácia pre analýzu sekvencií DNA v prostredí R</i>	479
Jakub Grib, Branislav Sobota, Štefan Korečko <i>Riadená plošina pre virtuálnu jaskyňu LIRKIS CAVE</i>	484
René Fedorčák, Branislav Madoš, Anton Baláž <i>Tvorba a vizualizácia troj-dimenzionálnych máp prostredia</i>	491
Radko Varchola, Anton Baláž, Branislav Madoš <i>Automatizovaný odpočet meračov</i>	497
Tomáš Marczi, Anton Baláž, Branislav Madoš <i>Forensic analysis of mobile instant messenger applications</i>	505
Jana Handriková <i>Aspekty Kybernetickej bezpečnosti v procese vzdelávania počas krízy COVID-19</i>	511
Author's Index	518

Vývoj univerzálnej mobilnej robotickej platformy

¹Ján BAČÍK, ²Peter TKÁČ

¹Katedra elektrotechniky a mechatroniky, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

^{1,2}Photoneo s.r.o., Slovenská republika

¹jan.bacik.2@tuke.sk, ²tkac@photoneo.com

Abstrakt—V tomto príspevku je opísaný vývoj univerzálnej mobilnej robotickej platformy určenej pre ďalší výskum a vývoj navigačných a riadiacich algoritmov z oblasti mobilnej robotiky. Vyvinutá platforma predstavuje robustný podvozok schopný prekonať nerovnosti terénu a stúpanie do 5°. Bezpečnosť platformy je zabezpečená systémom bezpečnostných laserových skenerov, bezpečnostných enkodérov a bezpečnostného programovateľného modulu.

KLúčové slová— mobilná platforma, robotika, AGV systémy

I. ÚVOD

S mobilnými robotickými systémami sa v súčasnosti človek v bežnom živote stretáva čoraz častejšie. Tieto robotické systémy si našli svoje uplatnenie nie len v priemysle ale aj v bežných domácnostiach v podobe automatizovaných vysávačov a kosačiek [1]. Taktiež sa tieto domáce roboty stali obľúbenou platformou rôznych výskumných inštitúcií zaoberajúcich sa vývojom mobilných robotických systémov [2]. Výhodou týchto platforiem je ich univerzálnosť a možnosť doplnenia systému o ďalšie senzory a výpočtové jednotky. V oblasti priemyslu si mobilné robotické systémy našli uplatnenie ako logistické ťahače vo veľkých výrobných halách, kde je ich úlohou autonómny transport materiálu z miesta na miesto [3].

Významným faktorom pri vývoji robotických platforiem je dôraz na bezpečnosť ich používania. Základným ochranným komponentom, pri väčšine týchto platforiem, je bezpečnostný skener umiestnený na prednej strane platformy. Tento bezpečnostný skener generujú ochranné pole pred robotom na základe informácií o jeho rýchlosti. Keďže sa v dnešnej dobe stále viac dbá na bezpečnosť, je toto riešenie síce dostačujúce [4], ale často je dopĺňané o ďalšie bezpečnostné komponenty akými sú 3D kamery a ultrazvukové snímače.

V článku sme zamerali pozornosť na vývoj univerzálnej mobilnej robotickej platformy (Obr. 1), ktorá by umožňovala ďalší vývoj algoritmov z oblasti mobilnej robotiky. Dôraz počas vývoja bol kladený na jej mechanickú robustnosť, prechodnosť terénu a v prvom rade bezpečnosť. Bezpečnosť platformy bola realizovaná inovatívnym použitím a umiestnením dvojice bezpečnostných laserových skenerov, ktoré sú doplnené o malé 3D kamery a ultrazvukové snímače. Článok opisuje stručný popis základnej mechanickej konštrukcie a jej elektrických komponentov, spolu s riešením bezpečnostného ochranného poľa okolo platformy.



Obr. 1 Mobilná robotická platforma

II. POPIS PLATFORMY

Robotická platforma je koncipovaná ako zváraná hliníková konštrukcia. Systém uloženia kolies umožňuje vyrovnávať nerovnosti terénu a dovoľuje tak platforme prekonať stúpanie do veľkosti 5° . Zváraná konštrukcia zároveň slúži na uchytenie držiakov elektrických komponentov a vane určenej pre umiestnenie dvoch olovených akumulátorov. Nosná konštrukcia a vnútorné komponenty platformy sú chránené sústavou vonkajších krycích plechov. Zároveň samotná konštrukcia a krycie plechy slúžia ako skriňa rozvádzača. Z toho dôvodu je vrchný odnímateľný horný krycí kryt zabezpečený zámkami ako ochrana pred otvorením neoprávnenou osobou.

A. *Systém uloženia kolies*

Samotný systém uloženia kolies pozostáva z hlavného ramena vahadla, zadného podporného kolesa, ložiskového domčeka, ktorý slúži na prepojenie a uchytenie hnacieho motora s hnacím kolesom, lôžka pre uchytenie bezpečnostného enkodéru a sústavy dvoch ozubených kolies pre prenos krútiaceho pohybu z hlavného hnacieho kolesa na os bezpečnostného enkodéru.

Nakoľko sa jedna tretina tiaže celého robota spolu s tiažou nákladu prenáša na hnaciu sústavu (motor - hnacie koleso) ako radiálne zaťaženie na os otáčania, bolo potrebné navrhnuť uchytenie motora a hnacieho kolesa tak, aby toto uchytenie bolo schopné dané zaťaženie zniesť. Z toho dôvodu je prepojenie motora a hnacieho kolesa realizované pomocou uloženia v ložiskovom domčeku. Táto konštrukcia umožňuje preniesť väčšie zaťaženie bez rizika poškodenia hnacej sústavy.

B. *Nosný rám a bezpečnosť*

Nosný rám robota tvorí zváraná konštrukcia z hliníkových nosníkov, ku ktorej sú upevnené všetky mechanické a elektrické komponenty robota - zadné oporné koleso, systém uloženia kolies, laserové skenery, ochranné kryty a pomocné plechové držiaky na upevňovanie elektrokomponentov. Konštrukcia mobilnej robotickkej platformy je dimenzovaná na priame zaťaženie nákladom o hmotnosti 100kg.

Všetky kolesá sú zo strán chránené hliníkovými krytmi. Tie nechránia len kolesá, ale robota ako celok. Odkryté ostávajú len nevyhnutné časti ako antény, snímacia časť skenerov, ovládacie tlačidlá a prepínače, nabíjacie plošky, ultrazvukové senzory a konektory periférií. Vozidlo je tiež chránené voči prístupu nepovoleným osobám k jeho elektrickým častiam zámkami umiestnenými v rohoch horného krycieho plechu.

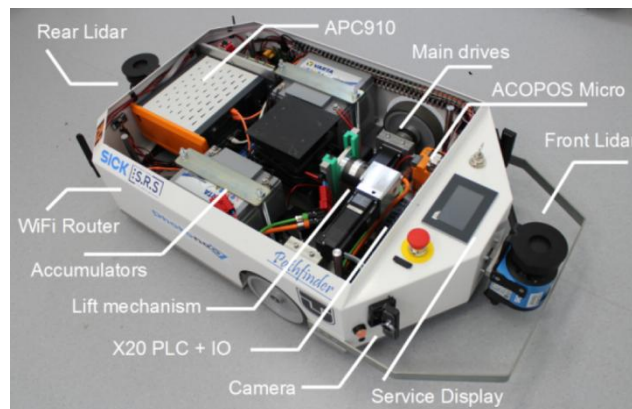
Ochrana ľudí je zabezpečená dvojicou bezpečnostných laserových skenerov, ktoré sú umiestnené v protihľých rohoch robota. Pre dosiahnutie ochrany v celom okolí robota musela byť konštrukcia prispôbena tak, aby žiadna časť konštrukcie nezasahovala do aktívneho meracieho poľa jednotlivých laserových skenerov. To sa podarilo dosiahnuť zúžením krycích plechov v časti, ktorá zodpovedá rovine aktívneho merania bezpečnostných laserových skenerov. Tým sa podarilo dosiahnuť 360° súvislé ochranné bezpečnostné pole bez vzniku mŕtvych uhlov.

C. *Elektrické komponenty*

Riadenie motorov a spracovanie signálov zo snímačov ma na starosti PLC, konkrétne od spoločnosti B&R, ktorého súčasťou sú aj bezpečnostné programovateľné karty. Tieto karty vyhodnocujú signály z bezpečnostných skenerov a bezpečnostných enkodérov. V prípade narušenia ochranného poľa alebo zisteniu rozdielu rýchlosti medzi bezpečnostnými a motorovými enkodéromi odpájajú riadiacu časť meniča motorov. Platforma disponuje výkonným priemyselným počítačom, ktorý je určený pre vývoj navigačných algoritmov robota. Priemyselný počítač má predinštalovaný operačný systém Ubuntu 18.04 a otvorenú robotickú platformu ROS. Súčasťou predinštalovaných balíkov je aj ROS Ethernet Powerlink [5], ktorý umožňuje rýchlu a robustnú komunikáciu medzi inými balíčkami z frameworku ROS a priemyselnými komponentmi, konkrétne medzi PLC a motorovým meničom. Vnútorná

eternetová sieť, ktorým ústredným členom je bezdrôtový smerovač, umožňuje komunikovať a pridávať zariadenia, ktoré podporujú TCP/IP komunikáciu. Bezdrôtový smerovač zároveň vytvára lokálnu bezdrôtovú sieť. Bezpečnosť platformy je rozšírená o dvojicu 3D kamier Astra Mini umiestnených v prednej a zadnej časti konštrukcie. Výstupom z kamier je množina bodov, v ktorej je možné pomocou vhodných algoritmov, detegovať prekážku a dať povel na zastavenie. Pre ochranu pred prekážkami zo skla, sa rovnako v prednej aj zadnej časti konštrukcie nachádzajú ultrazvukové snímače, ktorých signál je spracovaný analógovou kartou v PLC.

Napájanie platformy zabezpečuje dvojica dvanásť voltových trakčných olovených akumulátorov zapojených do série. Elektrickú ochranu zabezpečuje sústava DC ističov a napäťové monitorovacie relé, ktoré chráni batériu pred úplným vybitím.



Obr. 2 Prepojenie komponentov mobilnej robotickkej platformy.

III. ZÁVER

Mobilná robotická platforma, popísaná v príspevku, predstavuje univerzálnu platformu pre výskum a vývoj navigačných a riadiacich algoritmov z oblasti mobilnej robotiky. Platforma disponuje diferenciálne riadeným podvozkom uchyteným na systéme vahadiel, ktoré jej umožňujú prejazd aj zložitejším terénom. Samotná bezpečnosť je zabezpečená dvojicou bezpečnostných skenerov umiestnených v rohoch platformy tak, že spolu vytvárajú okolo mobilného podvozku 360°stupňové ochranné pole. Základná bezpečnosť je navyše rozšírená o systém 3D kamier a ultrazvukov. Výpočtový výkon, ktorý sa nachádza vo vnútri platformy je dostatočný pre vývoj a testovanie výpočtovo náročných algoritmov.

POĎAKOVANIE

Táto práca vznikla s podporou grantu VEGA 1/0187/18.

LITERATÚRA

- [1] J. Forlizzi, C. DiSalvo, Service robots in the domestic environment: a study of the roomba vacuum in the home, proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction, New York, 2006.
- [2] M. Dekan, L. Chovanec, A. Babinec, A. Vitko, New modules for the iRobot Create platform, Procedia Engineering, 2012.
- [3] Ceit Logistic Trucks: available on web: <http://www.ceittechnovation.eu/index.php/sk/agv-system>
- [4] STN P CLC/TS 62046, Bezpečnosť strojových zariadení. Aplikácia ochranných zariadení na detekciu prítomnosti osôb, Úrad pre normalizáciu, metrológiu a skúšobníctvo Slovenskej republiky, 2009.
- [5] J. Bačík, F. Ďurovský, M. Biroš, K. Kyslan, D. Perdukova, „S Padmanaban, Pathfinder–development of automated guided vehicle for hospital logistics, IEEE Access IEEE, 2017.

Porovnanie výberových stratégií horolezeckého algoritmu v rámci domény TSP

¹Jozef FRIVALDSKÝ, ²Marián MACH

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹jozef.frivaldsky@student.tuke.sk, ²marian.mach@tuke.sk

Abstrakt — Cieľom článku je problematika využitia prehľadávacích algoritmov pre riešenie úloh kombinatorickej optimalizácie. Pozornosť je zameraná na porovnanie výberových stratégií, používaných v rámci horolezeckého algoritmu. Článok sa zaoberá štyrmi stratégiami – najväčším zlepšením, prvým náhodným zlepšením a najmenším zlepšením. Kľúčovými faktormi vzájomného porovnania sú časové nároky týchto stratégií a kvalita dosahovaných riešení. V rámci experimentovania bolo realizovaných niekoľko experimentov s týmito stratégiami v rámci domény TSP na vybranom testovacom probléme v kombinácii s tromi rôznymi transformačnými operátormi pre generovanie nových návrhov riešení.

Kľúčové slová — horolezecký algoritmus, transformačný operátor, TSP, výberová stratégia

I. ÚVOD

Potreba riešiť úlohu z oblasti kombinatorickej optimalizácie je pomerne častou v každodennej praxi. Preto je dôležité sa vracať k metódam riešenia takýchto úloh a hľadať odporúčania, kedy použitie ktorej metódy je vhodné a kedy nie.

Jednou z populárnych skupín metód sú prehľadávacie algoritmy. Tieto vychádzajú z existencie priestoru prehľadávania, tvoreného veľkým počtom návrhov na hľadané riešenie (bodov tohto priestoru), z ktorých každý by mohol byť v nejakej miere považovaný za možné riešenie – niektorý by bol prijateľný viac a iný zase menej. Prijateľnosť jednotlivých návrhov je daná optimalizačnou funkciou. Prehľadávacie algoritmy prehľadávajú tento priestor a snažia sa nájsť taký návrh, ktorý je najprijateľnejší. Jedna populárna skupina takýchto algoritmov je reprezentovaná algoritmami lokálneho prehľadávania [1], ktoré sa vzdávajú úplnosti (a tým aj garancie nájdenia optimálneho návrhu) výmenou za prijateľné časové nároky, umožňujúce ich použitie v praktických podmienkach.

II. HOROLEZECKÝ ALGORITMUS A VÝBEROVÉ STRATÉGIE

Horolezecký algoritmus (známy aj pod menom iteračné zlepšovanie) je základným príkladom lokálneho prehľadávacieho algoritmu [2]. Začína v určitom bode priestoru prehľadávania a následne sekvenčne prechádza ďalšími bodmi tohto priestoru takým spôsobom, že hodnota optimalizačnej funkcie pre každý novo navštívený bod je vyššia ako hodnota predchádzajúceho bodu. Pamätá si iba aktuálny (posledne navštívený) bod, predchádzajúce body zabúda. Ak už nie je schopný nájsť bod priestoru s vyššou hodnotou optimalizačnej funkcie, tak svoju činnosť končí a za nájdené riešenie považuje aktuálny bod. Formálnejšie je toto možné vyjadriť pseudokódom

```
x := init( S )
while not null( x )
    v := x
    x := step( v , S )
return v
```

kde S je priestor prehľadávania, v je aktuálny bod, procedúra *init* náhodne vyberá jeden bod z priestoru S , *null* testuje neexistenciu a *step* vyberá nasledujúci bod.

Ťažiskom algoritmu je výber nasledujúceho bodu pomocou *step*. Tento výber sa nedeje z celého priestoru S ale iba z jeho časti – aby nový bod bol „podobný“ aktuálnemu bodu, tak sa

vyberá iba z lokálneho okolia $N(v)$ ($N \subseteq S \times S$) aktuálneho bodu v . Prakticky toto okolie je dané ako množina všetkých tých bodov priestoru S , na ktoré je možné aktuálny bod transformovať pomocou použitého transformačného operátora. Transformačné operátory sú doménovo špecifické. Pri použití rôznych transformačných operátorov vznikajú rôzne lokálne okolia.

A. Výberové stratégie

Procedúra *step* vyberá jeden bod z okolia aktuálneho bodu tak, aby novo vybraný bod bol z hľadiska optimalizačnej funkcie g ($g: S \rightarrow \mathfrak{R}$) lepší ako aktuálny bod. Vyberá sa teda iba z množiny $\{x \in N(v) \mid g(x) > g(v)\}$. Ak žiadny takýto bod v aktuálnom okolí neexistuje (a teda aktuálny bod je v danom okolí globálnym extrémom), *step* nevyberie nič a algoritmus končí svoju činnosť. Samotná voľba je realizovaná na základe nejakej stratégie (niekedy označovanej ako pivočné pravidlo). Najznámejšími stratégiami sú tieto [3]:

- Najväčšie zlepšenie – hľadá taký bod okolia, ktorý má najvyššiu hodnotu $g(x)$, v prípade viacerých takýchto bodov sa z nich vyberá náhodne
- Prvé zlepšenie – body okolia sa skúmajú v určenom poradí až pokiaľ sa nenájde bod lepší ako aktuálny
- Prvé náhodné zlepšenie – podobne ako predchádzajúca stratégia až na to, že poradie bodov nie je vopred určené ale je náhodné
- Najmenšie zlepšenie – hľadá taký bod okolia, ktorý má najnižšiu hodnotu $g(x)$ avšak väčšiu ako $g(v)$, v prípade viacerých takýchto bodov sa z nich vyberá náhodne

Z hľadiska zlepšovania hodnôt optimalizačnej funkcie stratégia najväčšieho zlepšenia sa snaží postupovať čo najväčšími krokmi s cieľom čo najskôr dospieť k hľadanému riešeniu. Naopak stratégia najmenšieho zlepšenia postupuje čo najmenšími krokmi s cieľom vyhnúť sa rýchlemu sklznutiu a uviaznutiu v lokálnom extréme aj za cenu predĺženia hľadania. Obe ostatné stratégie reprezentujú strednú cestu.

Z hľadiska počtu kontrolovaných bodov okolia stratégie najväčšieho aj najmenšieho zlepšenia sú extrémne – vždy musia najprv skontrolovať všetky body okolia, aby mohli prijať konečné rozhodnutie. Stratégie prvého (náhodného) zlepšenia typicky kontrolujú menšie množstvo bodov okolia, avšak ich počet nie je možné predpovedať – závisí na tom, koľko bodov okolia je lepších ako aktuálny bod a ako sú v okolí rozmiestnené.

Každé hľadanie pomocou horolezeckého algoritmu končí uviaznutím v lokálnom extréme, ktorý môže ale nemusí byť extrémom globálnym. Keďže sa kvalita tohto lokálneho extrému nedá vopred určiť, voľba stratégie generuje otázky:

- Najdlhšie hľadanie možno očakávať pri stratégii najmenšieho zlepšenia. Vyváži kvalita výsledného riešenia dlhšiu dobu hľadania?
- Odrazí sa na kvalite riešenia fakt, že stratégie prvého zlepšenia prehládajú iba (malú) časť okolia aktuálneho bodu?
- Je porovnateľná doba hľadania stratégiou najväčšieho zlepšenia (celé okolie) a hľadania stratégiami prvého zlepšovania (iba časť okolia ale viac iterácií algoritmu kvôli pomalšiemu zlepšovaniu hodnôt optimalizačnej funkcie)?
- Je rozdiel medzi pevne stanoveným a náhodným poradím bodov okolia pri stratégiách prvého zlepšenia?

III. VYBRANÉ EXPERIMENTY S VÝBEROVÝMI STRATÉGIAMI

A. Testovacia doména

Pre experimentovanie s rôznymi výberovými stratégiami bol v úlohe testovacej domény vybraný problém obchodného cestujúceho (TSP – Traveling Salesman Problem) motivovaný úlohou obchodného zástupcu navštíviť všetkých svojich zákazníkov v rôznych mestách a vrátiť sa späť prejdením čo najkratšej cesty. Jedná sa o známy problém z oblasti kombinatorickej optimalizácie, patriaci do NP-hard zložitostnej triedy [4].

Pre samotné testovanie bol použitý problém „Berlin52“ z TSPLIB (knihnica inštancií rôznych typov TSP) [5]. Jedná sa o symetrický (nezáleží na smere prechodu medzi miestami) a Euklidovský (vzdialenosť miest počítaná ako Euklidovská vzdialenosť) problém prechodu $n=52$ miestami. Hľadaným riešením je teda cyklická cesta cez všetkých n miest, pričom optimalizačnou funkciou je dĺžka cesty. Priestor prehládavania S bude pozostávať z

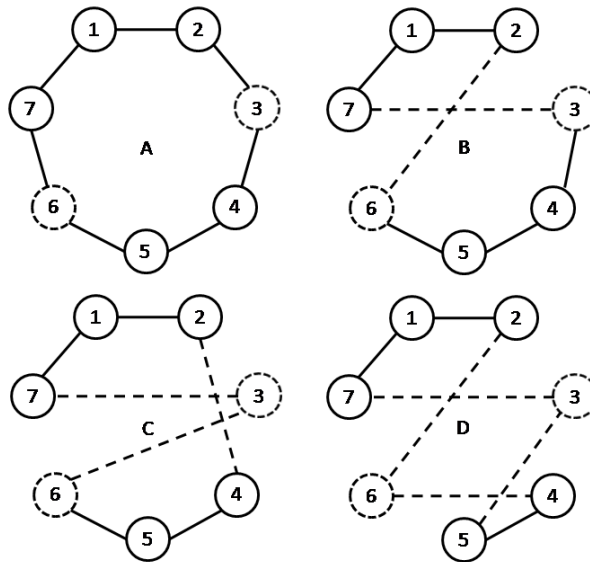
$$\frac{n!}{2n} = \frac{52!}{2 \cdot 52} = 7.755593766 \times 10^{65} \quad (1)$$

bodov, pričom každý z nich bude reprezentovať jednu kompletnú cyklickú cestu. Procedúra *init* štartuje hľadanie najkratšej cesty vygenerovaním náhodnej cesty cez všetkých 52 miest

a procedúra *step* používa transformačný operátor, ktorý transformuje cestu cez všetkých 52 miest na inú (aj keď podobnú) cestu, prechádzajúcu rovnakými miestami (avšak v čiastočne zmenenom poradí).

B. Transformačné operátory

Pretože rôzne transformačné operátory definujú okolie aktuálneho bodu rôznym spôsobom, tak pre minimalizovanie vplyvu konkrétneho operátora na porovnanie výberových stratégií boli použité viaceré operátory – operátory REVERSE, INSERT a SWAP [6]. Ich činnosť je ilustrovaná na obr. 1.



Obr. 1 Ilustrácia pôsobenia operátorov – A) pôvodná cesta, B) až D) nová cesta vytvorená z pôvodnej použitím jedného z operátorov - REVERSE (B), INSERT (C) a SWAP (D)

Všetky tri operátory vyžadujú výber dvoch miest (na obrázku bolo najprv vybrané miesto 3 a potom miesto 6). Ďalšia činnosť operátorov sa už líši:

- REVERSE – v úseku definovanom vybranými dvomi miestami (vrátane týchto dvoch miest) sa poradie miest obráti (obr. 1 - B). Dôsledkom je, že v ceste sa dva prechody medzi miestami zamenia za dva nové prechody.
- INSERT – prvé vybrané miesto sa zo svojho umiestnenia v ceste presunie tesne za druhé vybrané miesto (obr. 1 - C). Dôsledkom je, že v ceste sa tri prechody medzi miestami zamenia za tri nové prechody.
- SWAP – vybrané miesta si navzájom vymenia svoju polohu (obr. 1 - D). Dôsledkom je, že v ceste sa štyri prechody medzi miestami zamenia za štyri nové prechody.

Masívnosť zmien použitím jednotlivých operátorov je rôzna – najmenej zmeny má za následok operátor REVERSE zatiaľ čo cestu najmasívnejším spôsobom mení operátor SWAP.

C. Metodika testovania

Pri testovaní bola využitá softvérová implementácia z [7]. Z dôvodu stochastickej povahy horolezeckého algoritmu (či už pri inicializácii hľadania alebo pri výbere nového bodu z okolia aktuálneho jedinca) bolo pre každú kombináciu výberovej stratégie a transformačného operátora použité opakované hľadanie. Pri stratégii najmenšieho zlepšenia (z dôvodu dlhého hľadania) bol použitý znížený počet 200 opakovaní, pri ostatných výberových stratégiách bolo možné použiť 1000 opakovaní.

Pre meranie časovej náročnosti nebol použitý priamo čas potrebný na zavŕšenie procesu hľadania riešenia, pretože ten by silne závisel ako na charakteristikách hardvéru použitého pre testovanie tak aj na kvalite softvérovej implementácie horolezeckého algoritmu a použitých stratégií a operátorov. Namiesto toho boli použité dva zástupné parametre – počet iterácií algoritmu potrebných pre to aby sa algoritmus z iníciačného stavu dopracoval k uviaznutiu v lokálnom extrémne a celkový počet kontrolovaných (a teda aj generovaných ciest). Vychádzali sme z toho, že zložitosť použitých operátorov je viac menej rovnaká (všetky tri generujú rovnako veľké okolia, keďže všetky rovnako vychádzajú z výberu dvoch rôznych miest v aktuálnej ceste).

Pre meranie kvality nájdených riešení (ciest) bola priamo použitá dĺžka týchto ciest, pričom pre určenie dĺžky jednotlivých prechodov boli použité súradnice miest definované v testovacom probléme – horolezecký algoritmus bol použitý v minimalizačnom tvare. Pretože cieľom je posúdiť vplyv výberových stratégií a nie vplyv transformačných operátorov, tak dĺžky ciest boli normalizované do relatívneho tvaru – dĺžky ciest nájdených nejakým operátorom boli predelené najkratšou cestou nájdenou daným operátorom.

D. Porovnanie výberových stratégií

Výsledky časových charakteristík sú uvedené v tab. 1. Tabuľka obsahuje dvojice hodnôt, kde prvá hodnota udáva počet iterácií algoritmu a druhá celkový počet generovaných ciest.

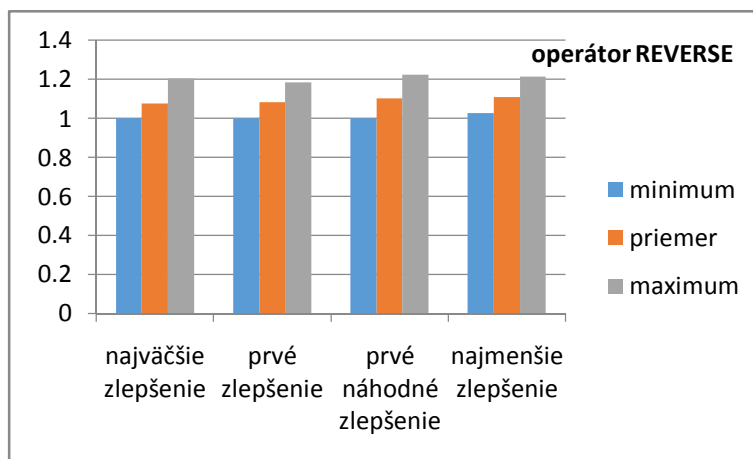
Tabuľka 1

Časová náročnosť hľadania riešenia rôznymi operátormi pri použití rôznych stratégií. Prvá hodnota v dvojiciach reprezentuje počet realizovaných iterácií algoritmu, druhá hodnota je počet generovaných ciest.

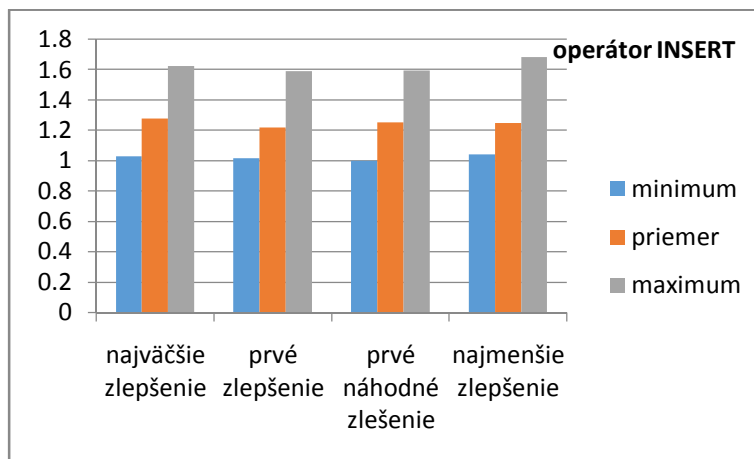
Operátor		Najväčšie zlepšenie	Prvé zlepšenie	Prvé náhodné zlepšenie	Najmenšie zlepšenie
REVERSE	Min	38 / 50389	121 / 17630	105 / 3594	4380 / 5807881
	Priemer	47.6 / 63151.7	173.8 / 47467.4	142.3 / 7038.4	8317.3 / 11028760.7
	Max	62 / 82213	262 / 98182	187 / 12894	11676 / 15482377
INSERT	Min	31 / 41107	160 / 16993	85 / 4209	3338 / 4325413
	Priemer	49.5 / 65691.1	244.5 / 47498.6	128.9 / 8241.9	4753.8 / 6303639.3
	Max	74 / 98125	381 / 115329	175 / 18088	6152 / 8157553
SWAP	Min	32 / 42433	122 / 16257	86 / 4127	4645 / 6159271
	Priemer	48.9 / 64969.7	183.5 / 52808.1	132.5 / 9462.4	7476.2 / 9913501.9
	Max	78 / 103429	255 / 110608	188 / 22444	10516 / 13944217

Ak keď nárast počtu iterácií aj generovaných ciest pri najmenšom zlepšení bol očakávaný, prekvapením je veľkosť tohto nárastu. Trojrádový nárast činí túto stratégiu prakticky nevhodnou – iba ak by to dokázala kompenzovať nárastom kvality poskytovaných riešení. Stratégia najväčšieho zlepšenia nedokázala prehládávanie celého okolia kompenzovať menším počtom iterácií. Aj keď pri nej počet iterácií bol niekoľkokrát menší ako pri stratégiách prvého zlepšenia, tie boli v konečnom dôsledku rýchlejšie – o tom svedčí celkový počet generovaných ciest (najmä v minimálnych hodnotách, v priemerných a vyšších hodnotách sa rozdiel medzi najväčším a prvým zlepšením začína stierať). Prekvapením je rozdiel medzi použitím pevného poradia prehládávania okolia a náhodného poradia pri stratégiách prvého zlepšenia. Použitie náhodného poradia dokázalo zmenšiť počty generovaných ciest o jeden rád. Celkovo možno konštatovať, že z časového hľadiska je jednoznačným víťazom stratégia prvého náhodného zlepšenia.

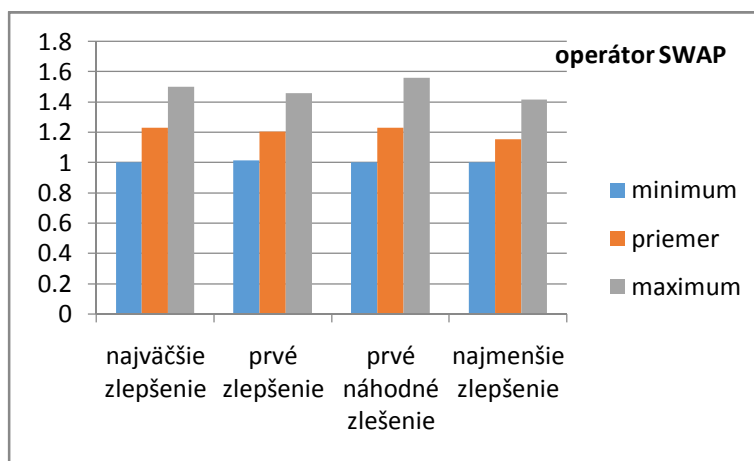
Dosiahnutá kvalita riešení je ilustrovaná grafmi na obr. 2, obr. 3 a obr. 4.



Obr. 2 Relatívne dĺžky ciest generovaných operátormi pri použití operátora REVERSE



Obr. 3 Relatívne dĺžky ciest generovaných operátormi pri použití operátora INSERT



Obr. 4 Relatívne dĺžky ciest generovaných operátormi pri použití operátora SWAP

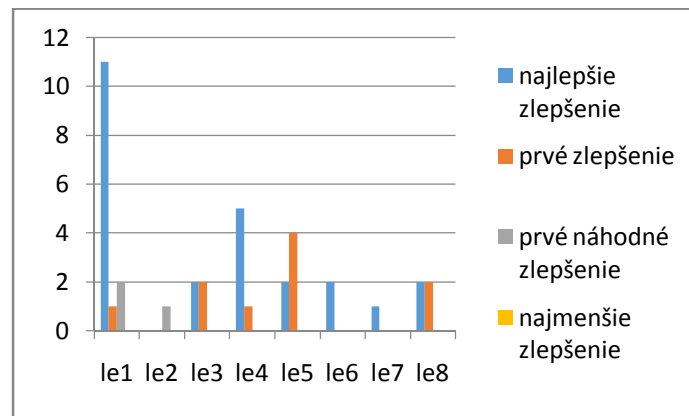
Na týchto obrázkoch boli použité operátory uvažované osobitne, aby vplyv každého z nich bol ohraničený iba na jeden graf - ak nejaká závislosť bude v každom z grafov, budeme ju považovať za všeobecne platnú (nezávislú na použitom operátore).

Stratégia najmenšieho zlepšenia neprináša nijaké podstatnejšie zlepšenie - v kategórii najkratšej cesty dokonca patrí k tým horším, takže vzhľadom na jej dĺžku hľadania nepatrí k tým, ktoré by sme doporučovali.

Najväčší rozdiel medzi ostatnými tromi stratégiami je v oblasti najhoršieho riešenia, v oblasti najlepšieho a priemerného riešenia sú porovnateľné, pričom pri jednotlivých operátoroch si vymieňajú miesta - pri operátore REVERSE je najlepším prvé zlepšenie a najhorším prvé náhodné zlepšenie, pri operátore INSERT je najhorším najväčšie zlepšenie a pri operátore SWAP je zase najhorším prvé náhodné zlepšenie.

Pre jemnejší pohľad obr. 5 zobrazuje osem najlepších lokálnych extrémov, nájdených pomocou operátora REVERSE (keďže rôzne operátory môžu mať rôzne lokálne extrémny, vybrali sme ten operátor, ktorý pre testovací príklad dosiahol najlepšie výsledky).

Ani jedna zo stratégií nenašla pri svojom opakovanom hľadaní všetkých osem extrémov. Stratégia najmenšieho zlepšenia potvrdila svoju nevhodnosť - nenašla ani jeden z nich. Na druhej strane všetky ostatné tri stratégie našli najlepší extrém - aj keď niektorá častejšie ako iné. Z hľadiska nachádzania tých najlepších riešení najúspešnejšou stratégiou bolo najlepšie zlepšenie (25 nájdení), druhou stratégiou v poradí bolo prvé zlepšenie (10 nájdení) a treťou zase stratégia prvého náhodného zlepšenia (3 nájdenia).



Obr. 5 Histogram najlepších lokálnych extrémov nájdených pomocou operátora REVERSE

IV. ZÁVER

Na základe vykonaných experimentov je možné konštatovať, že medzi výberovými stratégiami existujú rozdiely a je preto pri ich použití potrebné sa zamyslieť nad tým, čo je cieľom zamýšľaného hľadania. Ak je dôraz kladený na dobu hľadania, tak je možné jednoznačne odporučiť stratégiu prvého náhodného hľadania. Ak je dôraz na čo najväčšiu kvalitu riešenia bez ohľadu na časové hľadisko, potom sa vhodným odporúčaním javí stratégia najlepšieho zlepšenia. V prípade nutnosti čo najrýchlejšieho nájdenia čo najlepšieho riešenia to je stratégia prvého zlepšenia (ktorá je kompromisná – výsledok nebude k dispozícii ani tak rýchlo ako by mohol byť a asi nebude ani tak kvalitný ako by mohol byť).

POĎAKOVANIE

Táto publikácia vznikla vďaka podpore Agentúry pre podporu výskumu a vývoja v rámci projektu APVV-15-0731 2016-2020 „Cloud Based Human Robot Interaction“.

LITERATÚRA

- [1] H.H. Hoos a T. Stutzle, “Stochastic Local Search Algorithms: An Overview” in *Handbook of Computational intelligence*, Kacprzyk-Pedricz (Eds), Springer-Verlag, Berlin-Heidelberg, 2015, 1085-1105.
- [2] Z. Michalewicz a D.B.Fogel, *How to Solve It: Modern Heuristics*, 2nd ed. Springer-Verlag, Berlin-Heidelberg, 2010.
- [3] H.H. Hoos a T. Stutzle, *Stochastic Local Search: Foundations and Applications*. Morgan Kaufmann, 2004.
- [4] W.J.Cook, *In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation*. Princeton University Press, 2011.
- [5] TSPLIB - library of sample instances for the TSP. Dostupné: <http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html> (7.5.2020)
- [6] L. Sengupta, R. Mariescu-Istodor a F. Pasi, “Which Local Search Operator Works Best for the Open-loop TSP?”, *Applied Sciences*, roč. 9, 2019, 24 strán.
- [7] J. Frivaldský, *Riešenie úlohy typu Problém obchodného cestujúceho pomocou prehľadovacích algoritmov*, Bakalárska práca, Fakulta elektrotechniky a informatiky Technická univerzita v Košiciach, 2020.

Plánovanie trajektórie pohybu mobilného robota pomocou SMT

¹Róbert BALÁŽ, ²Marián MACH

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹robert.balaz@student.tuke.sk, ²marian.mach@tuke.sk

Abstrakt — Plánovanie trajektórie pohybu je jednou z typických úloh, riešených v oblasti mobilnej robotiky. Cieľom článku je poukázať na možnosti prístupu založeného na využití logického popisu vlastností hľadanej trajektórie, pri ktorom sa samotný proces hľadania trajektórie nahrádza snahou o dokázanie splniteľnosti jej popisu prostredníctvom logického dokazovania. Pozornosť je venovaná SMT logike kombinujúcej výrokovú počet s modulovými teóriami, pričom v našom prípade bola použitá modulová teória lineárnej aritmetiky. Navrhnutý postup bol implementovaný a prakticky otestovaný na pohybe laboratórnych mobilných robotov.

Kľúčové slová — modulová teória, plánovanie trajektórie, SMT, splniteľnosť

I. ÚVOD

Pohyb mobilného robota v prostredí je typicky plánovaný na základe existencie mapy prostredia, ktorá umožňuje odlíšiť voľný priestor, umožňujúci pohyb robota, od priestoru obsadeného prekážkami, bránicami robotovi v pohybe. V princípe existujú dva základné prístupy [1]. Pri prvom sa vytvára graf možných pozícií robota a bezpečných ciest medzi týmito pozíciami. Trajektória pohybu je hľadaná niektorou z metód pre hľadanie cesty v grafe medzi dvomi uzlami. V druhom prístupe je trajektória hľadaná priamo vo voľnom priestore bez vytvárania pomocnej grafovej štruktúry.

Do tejto druhej kategórie patrí aj prístup využívajúci možnosti logiky. Tento prístup transformuje úlohu nájdenia trajektórie vo voľnom priestore na úlohu splniteľnosti logických formúl, reprezentujúcich hľadanú trajektóriu. Bola použitá logika splniteľných modulových teórií SMT (Satisfiability Modulo Theories) [2], ktorá rozširuje výrokovú logiku o možnosť jej kombinácie s rôznymi teóriami. V našom prípade sme použili teóriu reálnych čísel (keďže súradnice v priestore sú reprezentované reálnymi číslami) a modulovú teóriu lineárnej aritmetiky (pre reprezentáciu vzájomných vzťahov medzi hodnotami súradníc hľadanej trajektórie).

II. GENEROVANIE TRAJEKTÓRIE POHYBU

A. Definovanie vlastností trajektórie

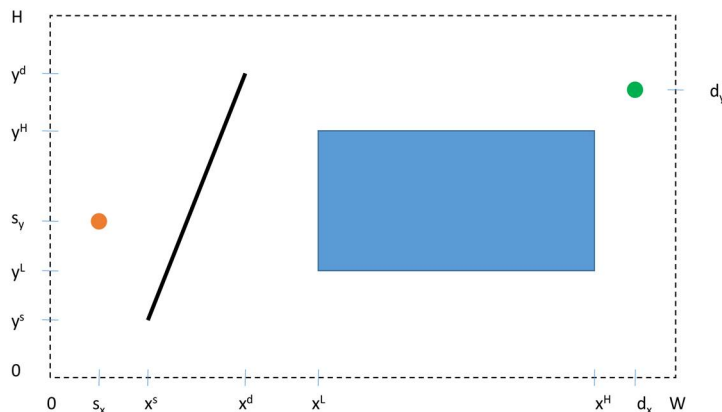
Predpokladáme existenciu mapy prostredia (kvôli pohybu po rovnej podložke uvažujeme dvojrozmernú mapu), v ktorej sa dá odlíšiť voľný priestor, použiteľný pre pohyb robota, od priestoru obsadeného prekážkami. Toto prostredie je ohraničené (uvažovaným priestorom, zorným poľom robota, dostupnou mapou priestoru, ap.), má stanovenú šírku W a dĺžku H . Je známa aj počiatočná pozícia robota (s_x, s_y) a cieľová pozícia (d_x, d_y) do ktorej sa má robot dostať [3][4].

Trajektória, ktorú je potrebné naplánovať, bude pozostávať zo sekvencie lineárnych segmentov. Každý segment je reprezentovaný dvomi dvojicami reálnych premenných (x_i^s, y_i^s) a (x_i^d, y_i^d) , kde i je označenie segmentu, s označuje štartovací bod segmentu a d označuje koncový bod segmentu. Prvá dvojica premenných teda predstavuje súradnice počiatočného bodu a druhá dvojica zase súradnice koncového bodu i -teho segmentu.

Prekážky môžu byť reprezentované ako súbor obdĺžnikových oblastí so stranami rovnobežnými so stranami skúmaného priestoru. Každý takýto obdĺžnik je vlastne kartézsky súčin intervalov $\langle x_j^l, x_j^h \rangle \times \langle y_j^l, y_j^h \rangle$, kde j je označenie j -tej obdĺžnikovej oblasti.

Situácia so štartovacím a cieľovým bodom, jedným segmentom navrhovanej trajektórie a

jednou takouto oblasťou s prekážkou je ilustrovaná na obr. 1.



Obr. 1 Ilustrácia popisu segmentu trajektórie a prekážky v prostredí

Aby trajektória bola validná, tak pre každý segment musí platiť niekoľko podmienok.

Ohraničenie prostredia – segment musí celý ležať vo vnútri uvažovaného priestoru (stačí ak v ňom ležia jeho krajné body)

$$0 \leq x_i^s \leq W \quad (1)$$

$$0 \leq y_i^s \leq H \quad (2)$$

$$0 \leq x_i^d \leq W \quad (3)$$

$$0 \leq y_i^d \leq H \quad (4)$$

Spájanie segmentov – segmenty musia na seba nadväzovať, tam kde končí prechádzajúci segment začína nasledujúci segment.

$$x_i^s = x_{i-1}^d \quad (5)$$

$$y_i^s = y_{i-1}^d \quad (6)$$

Počiatok a koniec trajektórie – prvý segment musí začínať v počiatocnom bode trajektórie, posledný segment musí končiť v koncovom bode trajektórie.

$$x_1^s = s_x \quad (7)$$

$$y_1^s = s_y \quad (8)$$

$$x_N^d = d_x \quad (9)$$

$$y_N^d = d_y \quad (10)$$

Vyhýbanie sa prekážkam – segment nesmie pretínať prekážku. Postačuje, ak celý segment leží vľavo, vpravo, pod alebo nad prekážkou.

$$(x_j^L - x_i^s \geq 0 \wedge x_j^L - x_i^d \geq 0) \vee \quad (11)$$

$$(x_i^s - x_j^H \geq 0 \wedge x_i^d - x_j^H \geq 0) \vee \quad (12)$$

$$(y_j^L - y_i^s \geq 0 \wedge y_j^L - y_i^d \geq 0) \vee \quad (13)$$

$$(y_i^s - y_j^H \geq 0 \wedge y_i^d - y_j^H \geq 0) \quad (14)$$

Uvedené ohraničenia sú nutnými ohraničeniami. Okrem týchto ohraničení je možné definovať aj ďalšie, napríklad presne daný smer segmentu (trebárs rovnobežný s niektorou z osí) prípadne vzťah medzi susednými segmentmi (trebárs kolmé na seba alebo zabránenie rovnobežnosti segmentov).

B. Vytváranie trajektórie

Pri vytváraní trajektórie zo štartovacieho bodu do bodu cieľového je potrebné najprv stanoviť počet segmentov, z ktorých má trajektória pozostávať. Následne je potrebné vygenerovať príslušné podmienky pre všetky segmenty a výsledný súbor podmienok poskytnúť SMT solveru (budeme používať tento termín kvôli jeho častému použitiu v rámci komunity a nie napr. riešič či riešiaci modul), ktorý je schopný kombinovať výrokovú logiku s modulovou teóriou reálnej lineárnej aritmetiky. Výsledkom činnosti solvera je buď potvrdenie existencie riešenia (a

poskytnutie hodnôt koncových bodov jednotlivých segmentov) alebo dokázanie neexistencie riešenia pre daný počet segmentov.

Problémom môže byť určenie počtu segmentov. Pri príliš nízkom počte riešenie nemusí existovať, pri príliš vysokom počte robíme úlohu zbytočne zložitou (s počtom segmentov rastie počet premenných a s ním aj počet generovaných podmienok). Často sa v tomto prípade používa iteračný prístup, keď sa začína s malým počtom segmentov a postupne sa ich počet zvyšuje až kým nie je nájdené riešenie (a zistený minimálny počet segmentov pre existenciu riešenia). Formálnejšie je toto možné vyjadriť pseudokódom

```
x := 0
do
  x := x + 1
  podm := generuj_podmienky( x )
  traj := SMT_solver( podm )
while null( traj )
```

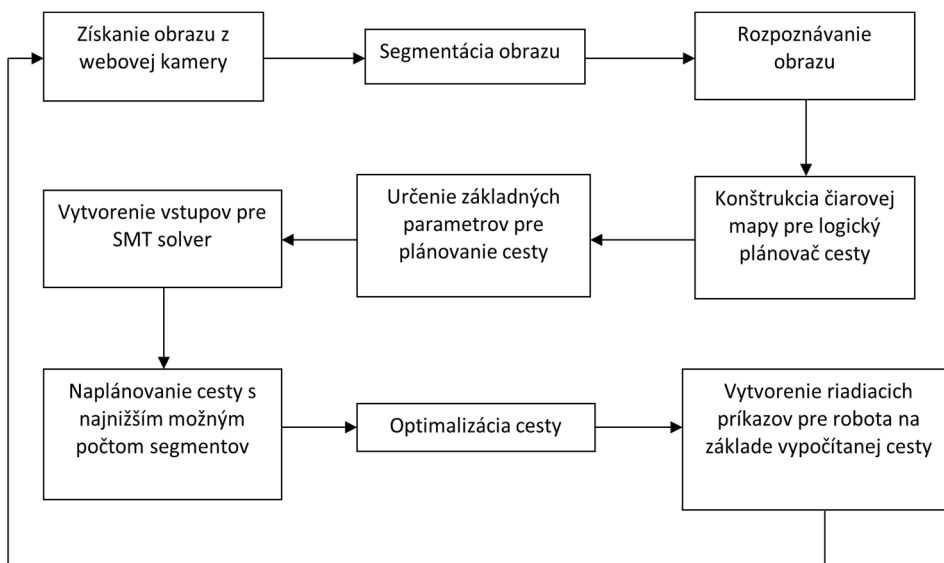
kde x je počet segmentov, $podm$ je súbor generovaných podmienok v tvare logického výrazu a $traj$ je generovaná trajektória.

Ak je potrebné hľadať v nejakom zmysle optimálnu trajektóriu (napríklad čo najkratšiu), tak po nájdení vhodného počtu segmentov je možné pre nájdenú trajektóriu určiť hodnotu optimalizačnej funkcie a pridať ďalšiu podmienku pre hľadanie takej trajektórie s daným počtom segmentov, ktorá má lepšiu hodnotu optimalizačnej funkcie. A zase iteračným postupom je možné nachádzať lepšie a lepšie trajektórie.

III. OVERENIE METÓDY PLÁNOVANIA TRAJEKTÓRIE

A. Štruktúra experimentu

Popísaná metóda generovania trajektórie pohybu bola experimentálne overená [5]. Experiment bol zameraný nielen na samotné generovanie trajektórie ale širšie na pohyb laboratórneho robota v prostredí. Pri experimente bolo použité zostavenie experimentu podľa diagramu na obr. 2.



Obr. 2 Bloková schéma implementácie

V tomto diagrame iba štyri bloky súvisia priamo s generovaním trajektórie pohybu. Sú to:

- určenie základných parametrov pre plánovanie cesty,
- vytvorenie vstupov pre SMT solver,
- naplánovanie cesty s najnižším možným počtom segmentov,
- optimalizácia cesty.

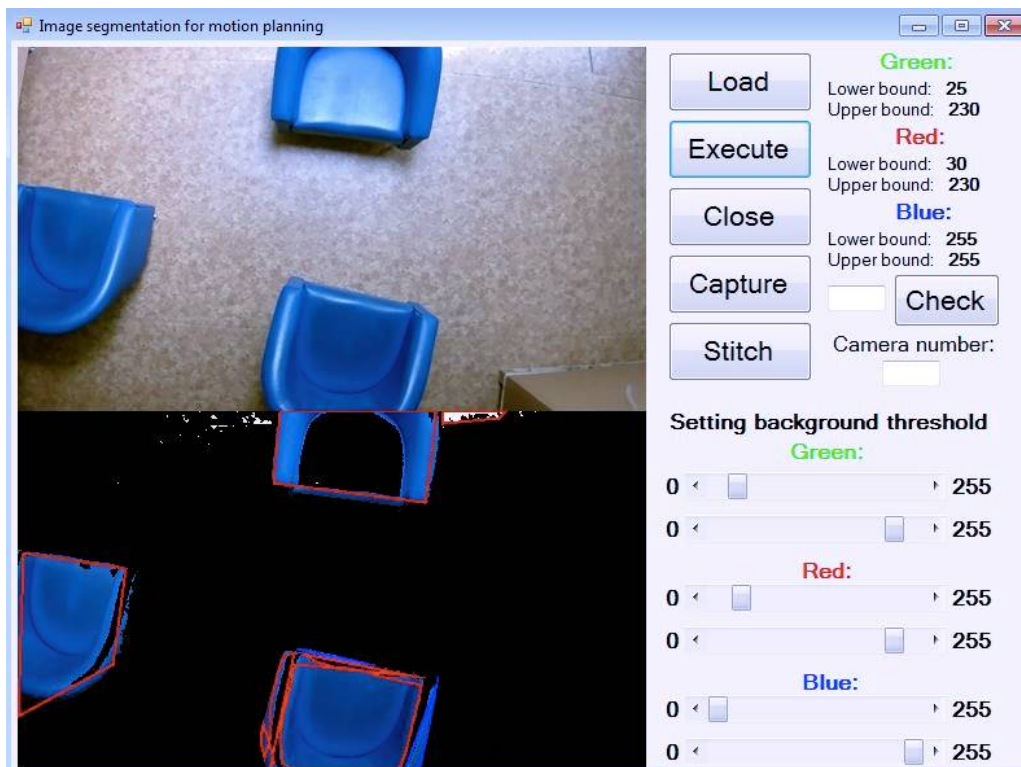
Funkčnosť týchto blokov realizuje vyššie popísanú metódu. Ostatné bloky je možné rozdeliť do dvoch skupín:

- vytvorenie aktuálnej mapy priestoru pre zistenie segmentovania okolitého priestoru na voľný priestor a priestor zabratý prekážkami.
- realizácia pohybu pozdĺž plánovanej trajektórie.

Mapa bola vytváraná prostredníctvom webových kamier, umiestnených v stropnej pozícii v priestore, v ktorom sa robot mal pohybovať. Kvôli obmedzeniu perspektívy bol u každej z kamier využitý iba priamy pohľad nadol, bez realizácie zmien natočenia kamier a stupňa priblíženia.

Kamera poskytla obraz priestoru, ktorý mohol byť ďalej spracovaný. Toto spracovanie pozostávalo zo segmentácie obrazu pre separáciu objektov od pozadia (v našom prípade pozadím bola podlaha). Samotná segmentácia bola realizovaná ako operácia prahovania voči limitom farieb v RGB farebnom modeli. Následným krokom bolo rozpoznanie tvaru objektov, ktoré boli vlastne prekážkami v plánovanom pohybe robota. Pre tento účel bol použitý algoritmus Houghovej transformácie. Nepravidelný tvar objektov bol aproximovaný polygonálnym tvarom. Kvôli možnému šumu za objekt bol považovaný iba zhluk minimálne desiatich pixlov.

Pre segmentáciu a rozpoznávanie bola využitá AForge.NET knižnica a EmguCV knižnica bola použitá pre získavanie obrazových rámcov z web kamier. Príklad daného spracovania možno vidieť na obr. 3.



Obr. 3 Spracovanie obrazu prostredia

Po úspešnom vygenerovaní trajektórie pohybu bola záverečne vygenerovaná sekvencia príkazov pre použitého robota, ktorá mu umožňovala pohyb pozdĺž trajektórie. Boli generované dva typy príkazov – priamy pohyb do určitej zadanej vzdialenosti a rotácia robota o zadaný uhol. Toto malo za následok nespojitý pohyb, spôsobený tým, že generovaná trajektória má tvar lomenej čiary. V experimente nebolo použité dodatočné vyhladzovanie trajektórie a náhrada spojenia dvoch segmentov v jednom bode pod nejakým uhlom kruhovým oblúkom. Príkazy boli generované pre robotov Nao a Qbo.

B. Generovanie trajektórie

Experiment bol vykonaný pri použití niekoľkých rôznych konfigurácií prekážok – vo všetkých prípadoch počiatočný a cieľový bod bol určený tak, aby neexistovala priama línia, spájajúca tieto dva body, ktorá by nekolidovala so žiadnou prekážkou. Ukážka vygenerovanej trajektórie je na obr. 4, kde štartovacím bodom bol ľavý koniec trajektórie a cieľovým zase jej pravý koniec. Ako vidno, trajektória sa úspešne dokázala vyhnúť všetkým prekážkam.

Metóda bola úspešná ako v prípade „riedkych“ prekážok, keď bol dostatok voľného priestoru pre pohyb robota, tak aj v prípade „hustých“ prekážok, keď oblasti, do ktorých bolo potrebné umiestniť zlomové body trajektórie boli značne obmedzené (prípady úzkych prechodov medzi prekážkami).



Obr. 4 Naplánovanie trajektórie pohybu v reálnom prostredí

Predstavu o časových nárokoch prezentovanej metódy podáva tab. 1, pričom berie do úvahy fakt, že faktormi, ktoré vieme ovplyvniť sú požadovaný počet segmentov trajektórie a počet prekážok v prostredí (nevieme ovplyvniť rozloženie možných riešení v priestore prehľadávanom SMT solverom, ktoré má značný vplyv na rýchlosť činnosti solvera v prípade, že solver vie potvrdiť existenciu riešenia).

Tabuľka 1
Časové nároky generovania trajektórie

Počet segmentov	Počet prekážok	Čas [ms]	Počet prekážok	Čas [ms]	Počet prekážok	Čas [ms]
20	8	45	60	277	660	1995
40	8	74	60	992	660	4958
60	8	84	60	1859	660	6794

IV. ZÁVER

V článku bola prezentovaná metóda generovania trajektórie pohybu pomocou využitia SMT logiky, použiteľná v spojitosti s ľubovoľným typom mapy, ktorý umožňuje segmentovať priestor do dvoch kategórií – voľný priestor a priestor obsadený prekážkami. Jedná sa o úplnú metódu, ktorá nájde riešenie v prípade, že riešenie existuje a naopak ak ho nenájde, to znamená že riešenie neexistuje. Vykonané experimenty potvrdili životaschopnosť metódy a to ako z hľadiska nachádzaných trajektórií, tak aj z hľadiska časových nárokov – metóda je použiteľná, ak je k dispozícii časové kvantum do sto milisekúnd v prípade typických jednoduchších prostredí alebo rádovo sekundy v prípade veľmi zložitých prostredí. V budúcnosti by bolo vhodné získané výsledky porovnať s inými používanými metódami v danej oblasti.

POĎAKOVANIE

Táto publikácia vznikla vďaka podpore Agentúry pre podporu výskumu a vývoja v rámci projektu APVV-15-0731 2016-2020 „Cloud Based Human Robot Interaction“.

LITERATÚRA

- [1] R. Siegwart a I.R. Nourbakhsh a D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd ed, MIT Press, 2011, 472 strán.
- [2] L. DeMoura a N.S. Björner, “Satisfiability Modulo Theories: Introduction and Applications”, in *Communications of the ACM*, roč. 54, 2011, č. 9, 69-77.
- [3] Y. Shoukry et al, “Scalable Motion Planning Using Lazy SMT-based Solving”, in *IEEE Conference on Decision and Control*, Las Vegas, NV, 2016, 6683-6688, DOI: 10.1109/CDC.2016.7799298.
- [4] W. Hung et al, “Motion Planning with Satisfiability Modulo Theories”, in *Proc. of the IEEE Int. Conference on Robotics and Automation*, Hong Kong, 2014, 113-118.
- [5] R. Baláž, *Aplikácia logického prístupu pre plánovanie pohybu*, Bakalárska práca, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, 2017, 74 strán.

FPGA Hardvérový Akcelerátor

Minimalizácie Booleovskej Funkcie

¹Alžbeta SKLENÁRIKOVÁ, ²Branislav MADOSĎ, ³Anton BALÁŽ

¹Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹alzbeta.sklenarikova@student.tuke.sk, ²branislav.mados@tuke.sk

Abstract – Článok sa zaoberá problematikou systematickej minimalizácie Booleovských funkcií, konkrétne jej druhým krokom, ktorým je riešenie problému pokrytia Booleovskej funkcie. V článku navrhnuté riešenie umožňuje, s využitím hardvérového akcelerátora založeného na využití Field Programmable Gate Array a popísaného prostredníctvom jazyka VHDL, identifikáciu esenciálnych prostých implikantov a identifikáciu pokrytia jednotkových bodov Booleovskej funkcie prostými implikantmi.

Keywords – Booleovská funkcia, minimalizácia, Karnaughova mapa, Quine-McCluskey, kombinačný obvod, implikant, prostý implikant, hardvérový akcelerátor, VHDL, FPGA

I. ÚVOD

Tak ako sa počítače vyvíjali od obdobia ich nulej generácie, až po súčasnú piatu generáciu, sa neustále zdokonaľoval základný prvok, na ktorom bol založený ich hardvér. Prvá generácia počítačov bola založená na elektrónkach, ktoré boli rozmerné, prehrievali sa, mali vysokú spotrebu energie a boli málo spoľahlivé. Ďalšia generácia priniesla počítače založené na tranzistoroch, ktoré boli podstatne kompaktnějšíe, mali menšiu spotrebu energie a boli oveľa spoľahlivejšie. S nástupom ďalšej generácie sa objavili integrované obvody, ktoré sa skladali z viacerých tranzistorov integrovaných na jednom kremíkovom plátku. To odštartovalo revolúciu v oblasti konštruovania počítačov, keď sa počet tranzistorov integrovateľných do integrovaného obvodu zvyšoval obrovským tempom. V počiatkoch bola integrácia na úrovni niekoľkých tranzistorov, v súčasnosti sa ich počet na jednom kremíkovom plátku počíta na miliardy. Návrhy logických obvodov sa v súlade s týmto trendom stávajú čoraz komplikovanejšími.

Paralelne s týmto vývojom sa objavila otázka, či sú návrhy jednotlivých logických obvodov optimálne, teda či sa neskladajú z vyššieho počtu prvkov ako je potrebné. Hlavne v úvodných fázach, keď integrované obvody doslova bojovali o každý tranzistor, ktorý do nich bolo možné integrovať, bola táto otázka mimoriadne dôležitá a tak sa vytvorili postupy, ako nielen popisovať logické obvody, ale ako ich aj optimalizovať, resp. minimalizovať. Vznikli algoritmy umožňujúce uistiť sa, že obvod je zložený skutočne z minimálneho množstva prvkov, ako sú algoritmy systematickej minimalizácie kombinačných logických obvodov, medzi ktoré patrí minimalizácia v Karnaughových mapách alebo Quine-McCluskeyho metóda. So stúpajúcim počtom premenných a teda so stúpajúcim počtom riadkov pravdivostnej tabuľky je však systematická minimalizácia čoraz náročnejšia, pretože je náročná na priestor, teda na veľkosť pamäte, ako aj čas, teda na množstvo výpočtov, ktoré je potrebné urobiť. Táto náročnosť stúpa s počtom premenných tak rýchlo, že pri stovkách vstupných premenných je použitie systematickej minimalizácie nepraktické. Vyvinuté preto boli algoritmy druhej generácie, ktoré predstavujú heuristickú minimalizáciu. Tá dokáže v podstatne kratšom čase a/alebo s použitím menšieho priestoru zabezpečiť riešenie, ktoré nemusí byť optimálne, ale ktoré sa k optimálnemu riešeniu blíži natoľko, že ho možno akceptovať.

Cieľom tejto práce je navrhnúť FPGA hardvérový akcelerátor pre realizáciu druhého kroku systematickej minimalizácie Booleovskej funkcie a umožniť tak identifikáciu jednotkových bodov pokrytých konkrétnym prostým implikantom a identifikáciu esenciálnych prostých implikantov pre konkrétne jednotkové body Booleovskej funkcie.

II. SÚVISIACE PRÁCE

Allan Marquand priniesol v roku 1881 Marquandove diagramy, ktoré umožňovali zjednodušenie grafickej reprezentácie Vennových diagramov pre väčší počet premenných [1]. V roku 1952 Edward Westbrook Veitch vyvinul metódu minimalizácie Booleovských funkcií [2] a príslušné

diagramy, ktoré sa tiež zvyknú nazývať Marquand-Veitch diagrams. Túto metódu neskôr zdokonalil Maurice Karnaugh v roku 1953 a dnes je známa ako Karnaughove mapy (KM or K-map) [3][4].

Karnaughove mapy, ktoré sú niekedy označované aj ako Karnaugh-Veitch maps (KV maps), slúžia nielen pre účely grafického zápisu Booleovských funkcií ale najmä pre ich minimalizáciu. Využívajú ľudskú schopnosť vyhľadávať vzory v grafickej reprezentácii Booleovskej funkcie vo forme diagramu, namiesto programovej realizácie minimalizácie Booleovskej funkcie s využitím výpočtov prostredníctvom počítača. Obmedzením tejto metódy je možnosť vizuálneho vyhľadávania vzorov a následnej simplifikácie v Karnaughových mapách s len veľmi malým počtom vstupných premenných, pričom ich hraničný počet je uvádzaný na úrovni 5 až 6 premenných. Ďalšou nevýhodou je ľudský faktor, ktorý do procesu môže vnášať chyby.

Quine-McCluskeyho metóda, označovaná aj ako Q-M, je tabulková metóda systematickej minimalizácie Booleovských funkcií, ktorá je z hľadiska dosahovaných výsledkov analogická s Karnaughovými mapami. Bola vyvinutá Willardom Quineom a Edwardom McCluskeym v roku 1952 [5][6]. Je to dvojkroková metóda, keď v prvom kroku sú generované prosté implikanty Booleovskej funkcie a v druhom kroku je realizované riešenie problému pokrytia Booleovskej funkcie pomocou týchto prostých implikantov. Výhodou tejto metódy oproti Karnaughovým mapám je, že sa nespolieha na schopnosť človeka vyhľadávať vzory, ale zavádza ľahko programovo implementovateľný algoritmus, ktorý umožňuje spracovávať Booleovské funkcie s výrazne vyšším počtom premenných oproti Karnaughovým mapám, rádovo v desiatkach prípadne stovkách.

Heuristický minimalizátor MINI bol predstavený Hongom et al. v roku 1974 [7]. De facto štandardom pri heuristickej minimalizácii sa stal minimalizátor s názvom Espresso, ktorý bol vyvinutý R. K. Brayton et al. s cieľom umožniť minimalizáciu logických obvodov s použitím heuristických metód [8]. Espresso-MV ako derivát minimalizátora Espresso bol vyvinutý v roku 1986 Richardom L. Rudellom a bol popísaný v [9]. Zdrojový kód minimalizátora Espresso je k dispozícii v [10]. Neskoršie vylepšenia algoritmu Espresso známe ako Espresso-Exact a Espresso Signature sú bližšie popísané v [11].

Hlavička a Fišer vyvinuli minimalizačný nástroj s názvom BOOM – BOOlean Minimizer, ktorý je založený na novom poňatí generovania implikantov metódou zhora nadol, na rozdiel od iných metód, ktoré generujú implikanty metódou zdola nahor. Výhodou je aj redukcia počtu prostých implikantov. Pomocou tohto nástroja je možné minimalizovať Booleovské funkcie s veľkým počtom premenných (až do tisícok), keď ďalšie algoritmy nedokážu dať požadovaný výsledok v rozumnom čase [12][13][14][15][16]. FC-Min minimalizačný nástroj bol predstavený v [17] Fišerom a Kubátovou a neskôr bol skombinovaný s BOOM algoritmom do algoritmu BOOM-II [18][19].

III. VLASTNÝ NÁVRH HARDVÉROVÉHO AKCELERÁTORA

Vlastný návrh riešenia pre realizáciu hľadania pokrytia jednotkových bodov Booleovskej funkcie a esenciálnych prostých implikantov, teda implikantov, ktoré ako jediné pokrývajú konkrétny stĺpec mriežky prostých implikantov, zahŕňa návrh:

- **PE** - *Procesného Elementu*, pre hodnotenie pokrytia jednotkového bodu prostým implikantom.
- **RIM** - *Registra IMplikantu*, pre ukladanie kódov prostých implikantov.
- **RIE** - *Registra Indikátora Esenciálnosti* pre indikovanie esenciálnosti prostého implikantu pre jednotkový bod.

A. *Procesný element PE*

Základom pre realizáciu celého obvodu je procesný element (PE), ktorý je určený na zistenie, či konkrétny jednotkový bod Booleovskej funkcie je pokrytý konkrétnym prostým implikantom. Procesný element má dva vstupy a jeden výstup.

Na vstup je privádzaný:

- Kód konkrétneho jednotkového bodu, ktorého pokrytie je testované, vo forme n -bitového vstupného vektora $A(n-1:0)$, kde n je parameter obvodu a je zhodné s počtom premenných Booleovskej funkcie, označených ako $x_z : z \in \langle n-1:0 \rangle$, zapísaných do vektora $A(n-1:0) = x_{(n-1)}x_{(n-2)}x_{(n-3)} \dots x_2x_1x_0$. Ak je príslušná premenná v popise daného jednotkového bodu negovaná, je reprezentovaná bitom 0 a ak je priama, je reprezentovaná bitom 1.

- Kód konkrétneho prostého implikantu, kde je testované, či daný prostý implikant pokrýva konkrétny jednotkový bod, vo forme m -bitového vstupného vektora $B(m-1:0)$, kde $m = 2n$, označených ako $x_z : z \in \langle m-1:0 \rangle$ zapísaných ako vektor $x_{(m-1)}x_{(m-2)}x_{(m-3)} \dots x_2x_1x_0$. Premenná, ktorá má vo vektore A pozíciu p , je vo vektore B reprezentovaná dvojicou bitov s pozíciou $2p$ a $2p+1$. Ak sa príslušná premenná nezúčastňuje na popise príslušného prostého

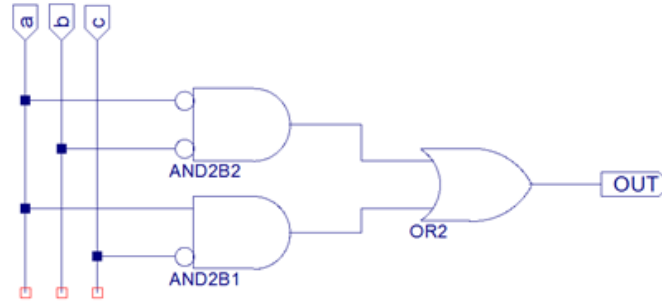


Fig. 1 PE umožňujúci zhodnotenie, že daný prostý implikant pokrýva daný jednotkový bod pre jednu premennú.

implikantu, je uvedená ako binárny kód 00. Ak je súčasťou popisu prostého implikantu ako negovaná, je reprezentovaná binárnym kódom 01 a napokon, ak je súčasťou prostého implikantu ako priama, je uvedená ako binárny kód 10. Bit s pozíciou $2p$ je označovaný v tejto práci ako c , bit s pozíciou $2p + 1$ je označovaný ako b .

Výstup procesného elementu:

- Výstupom procesného elementu s označením *OUT* je jediný bit, ktorý kóduje, či jednotkový bod pôsobiaci na vstupe obvodu prostredníctvom vektora bitov *A* je pokrytý prostým implikantom, pôsobiacim na vstupe procesného elementu prostredníctvom vektora bitov *B*. Ak je pokrytý, výstup *OUT* je nastavený na hodnotu 1, ak nie je pokrytý, výstup je nastavený na hodnotu 0.

Aby bolo možné tvrdiť, že daný jednotkový bod je pokrytý príslušným prostým implikantom, musí platiť o každej premennej Booleovskej funkcie popisujúcej príslušný prostý implikant, jedna z nasledujúcich možností:

- Premenná nie je pri popise príslušného prostého implikantu použitá. Potom môže byť táto premenná v popise jednotkového bodu tak negovaná, ako aj priama.
- Premenná v popise príslušného prostého implikantu je negovaná, a súčasne je príslušná premenná negovaná aj v popise jednotkového bodu Booleovskej funkcie.
- Premenná v popise príslušného prostého implikantu je priama, a súčasne je príslušná premenná priama aj v popise jednotkového bodu Booleovskej funkcie.

Ak premenná *a* predstavuje kód premennej v príslušnom binárnom vektore popisujúcom jednotkový bod a premenné *b*, *c* predstavujú dvojbítový kód príslušnej premennej v binárnom vektore popisujúcom príslušný prostý implikant, potom kombinačný obvod zobrazený na Fig. 1 umožňuje zhodnotiť, či je splnená podmienka, že daný prostý implikant pokrýva daný jednotkový bod (na výstupe je hodnota 1), alebo nepokrýva (na výstupe je hodnota 0).

Na Fig. 2 je potom schematické znázornenie kombinačného obvodu PE, ktorý umožňuje zhodnotiť, či jednotkový bod, ktorého premenné sú reprezentované bitmi a_2, a_1, a_0 , je pokrytý prostým implikantom, ktorý je kódovaný pomocou bitov b_2, b_1, b_0 a c_2, c_1, c_0 .

V FPGA čípe je implementovaných x takýchto procesných elementov s označením $PE_z : z \in \langle 0; x - 1 \rangle$, pričom vykonávajú svoju funkciu paralelne. Ich počet je ovplyvnený celkovou kapacitou príslušného FPGA obvodu ako aj maximálnym počtom premenných Booleovskej funkcie spracovateľnej pomocou tohoto akcelarátoru.

B. Register Implikantu RIM

Výstupné záchytné registre (RIM), ktorých počet je zhodný s počtom procesných elementov *PE*, sú určené pre uloženie kódu prostého implikantu. Veľkosť daného záchytného registra je *m* bitov, označených ako $(m - 1 : 0)$. Ak je bod pokrytý aspoň jedným prostým implikantom, bude v príslušnom výstupnom záchytnom registri uložený kód takéhoto prostého implikantu. Ak je bod pokrytý viacerými prostými implikantmi, v záchytnom registri bude uložený kód posledného spracovávaného prostého implikantu, ktorý pokrýva daný jednotkový bod.

Vstup *CLR* umožňuje nulovať záchytný register, teda nastaviť každý preklápací obvod typu *D*, z ktorého sa skladá, na hodnotu 0. Vstup *CLK* predstavuje vstup taktovacieho hodinového signálu. Vstup *PEIN*, na ktorý je napojený výstup *OUT* z príslušného *PE*, nastavený na hodnotu 1 indikuje, že daný jednotkový bod je pokrytý príslušným prostým implikantom, a teda že jeho binárny kód je potrebné zapísať do *RIM*. Preto je tento signál privedený na vstup *CE* každého

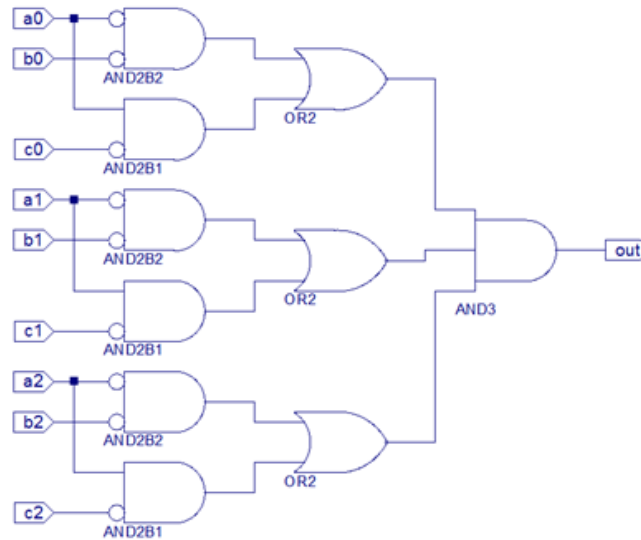


Fig. 2 Kombinačný obvod PE pre spracovávanie Booleovských funkcií s maximálne tromi premennými.

preklápacieho obvodu typu D . Na príslušné vstupy D jednotlivých preklápacích obvodov sú potom privedené príslušné bity binárneho kódu prostého implikantu.

Za predpokladu, že na vstup $PEIN$ je privedený signál s hodnotou 1, pri nábežnej hrane signálu CLK dôjde k zapísaniu jednotlivých bitov vstupného vektora predstavujúceho prostý implikant do jednotlivých preklápacích obvodov a následne sa tento binárny kód prostého implikantu objaví na výstupe $IMP(m-1:0)$.

C. Register indikátora esenciálnosti RIE

Súčasťou návrhu je aj 2-bitový register indikátora esenciálnosti (RIE), ktorý pre každý PE nesie informáciu o počte nájdených prostých implikantov pokrývajúcich daný jednotkový bod. Register obsahuje 2 bity a môže teda niesť štyri rôzne hodnoty s nasledujúcim významom:

- Hodnota **00** – dosiaľ nebol nájdený žiadny prostý implikant, ktorý by pokrýval daný jednotkový bod.
- Hodnota **01** – dosiaľ bol nájdený jediný prostý implikant, ktorý pokrýva daný jednotkový bod, a preto je tento prostý implikant esenciálny.
- Hodnota **10** – nedovolená hodnota – túto hodnotu register nenadobudne.
- Hodnota **11** – dosiaľ boli nájdené minimálne dva prosté implikanty pokrývajúce daný jednotkový bod, a preto žiadny z nich nie je možné pokladať za esenciálny prostý implikant pre daný jednotkový bod.

Modul registra disponuje vstupom CLK , ktorým je privádzaný taktovací signál. Vstup CLR nastavený na hodnotu 1 umožňuje vynulovať register, teda nastaviť obidva jeho preklápacie obvody typu D na hodnotu 0. Po vynulovaní registra, ak sa na vstupe $PEIN$ tohto obvodu objaví na jeden takt hodnota 1, nastaví sa register na hodnotu 01. To indikuje, že bol nájdený esenciálny prostý implikant pre daný jednotkový bod. Ak je na vstup $PEIN$ druhýkrát privedená hodnota 1 na jeden takt, indikuje to ďalší prostý implikant nájdený pre daný jednotkový bod a register sa nastaví na hodnotu 11. Výstup registra má označenie (EI). Schematické znázornenie obvodového riešenia tohto registra možno vidieť na Fig. 3.

IV. RESULTS

Z vyššie uvedených modulov bol zostavený prototyp hardvérového akceleračora, ktorý umožňuje paralelne testovať pokrytie 4 jednotkových bodov Booleovskej funkcie, pretože obsahuje 4 procesné elementy PE , označené ako PE_0 až PE_3 , do ktorých sú jednotlivé jednotkové body privedené ako vstupné vektory A_0 až A_3 , pričom každý z nich predstavuje 4-bitový vektor ($n = 4$) $A_0(3:0)$ až $A_3(3:0)$. Vstup B je využitý pre sekvenčné zavádzanie prostých implikantov minimalizovanej Booleovskej funkcie, tvoriacich jej skrátenú normálnu formu. Vstup B predstavuje 8-bitový vektor $B(7:0)$. Každý z PE elementov má na vstupe vektor B a svoj príslušný vektor $A_x : x \in \langle 0; 3 \rangle$. Výstup príslušného PE je vo forme 1b OUT , ktorý

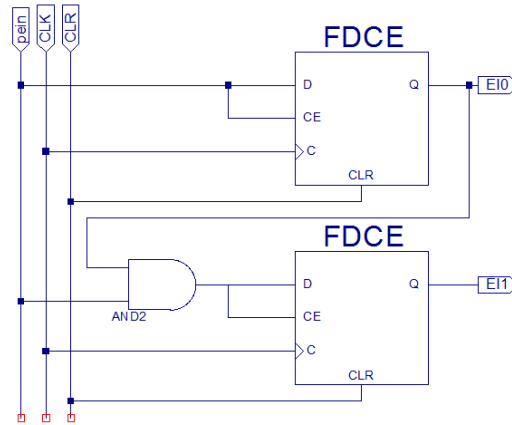


Fig. 3 Schéma 2-bitového registra RIE pre indikovanie existencie esenciálneho prostého implikantu.

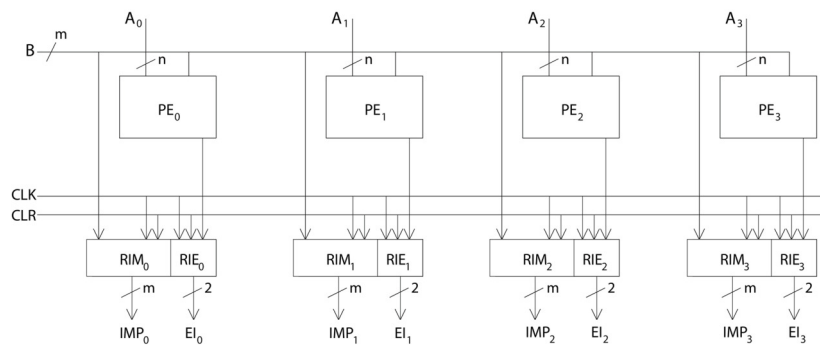


Fig. 4 Schematické znázornenie hardvérového akcelarátoru minimalizácie Booleovskej funkcie.

indikuje, či príslušný implikant pôsobiaci na vstupe B pokrýva jednotkový bod Booleovskej funkcie A . Tento výstup sa stáva vstupom príslušného 2b registra RIE , ktorých je v akcelarátoru implementovaných takisto 4, s označením RIE_0 až RIE_3 . Súčasne je vstup B privedený na vstupy príslušných registrov RIM_0 až RIM_3 . Register RIM je určený na zachytenie príslušného implikantu, ktorý pokrýva jednotkový bod a register RIE je určený na indikovanie, či bol jednotkový bod vôbec pokrytý, či bol pokrytý raz, alebo viackrát. Príslušný výstup z registra RIM je označený ako IMP , a výstup z RIE je označený ako EI . Schematické znázornenie hardvérového akcelarátoru popísaného v tejto kapitole je na Fig. 4.

Vstup CLK je vstupom hodinového signálu, určeného na synchronizáciu práce preklápacích obvodov registrov RIM a RIE . Vstup CLR umožňuje nulovanie preklápacích obvodov RIM a RIE , teda nastavenie ich hodnoty na 0. Procesné elementy PE sú realizované ako kombinačné logické obvody. Pred začiatkom vykonávania výpočtov je potrebné nastaviť signál CLR na hodnotu 1, aby došlo k nastaveniu všetkých preklápacích obvodov na hodnotu 0. Inak obvod nebude správne fungovať.

Všetky štyri procesné elementy, ako aj všetky štyri dvojice registrov pracujú paralelne, čo je zdrojom akcelerácie výpočtu oproti sekvenčnému vykonávaniu algoritmu prostredníctvom CPU klasického počítačového systému. Zdrojom akcelerácie výpočtov je aj samotné vykonávanie funkcie procesného elementu, v ktorom paralelne prebieha porovnávanie jednotlivých premenných z prostého implikantu a jednotkového bodu Booleovskej funkcie, ktoré takisto musí prebiehať v klasickom počítači v CPU pomocou vykonávania sekvencie inštrukcií.

V. CONCLUSION

Výhodou dosiahnutého riešenia je jeho kompaktnosť, keď sa skladá z minimálneho počtu modulov, konkrétne je to procesný element, ktorý je jednoduchým kombinačným obvodom, realizovaným prostredníctvom hradieľ typu AND a OR. Pre modul s n premennými je použitých $2n$ hradieľ typu AND s dvoma vstupmi a n hradieľ typu OR s dvoma vstupmi. Pre realizáciu výstupného registra implikantu a registra esenciálnosti implikantu bolo pre implikant s n premennými potrebných $2n+2$ preklápacích obvodov typu D a jedno dvojjstupové hradlo typu AND. Takto je vytvorený rez pre jeden jednotkový bod Booleovskej funkcie a takýchto modulov je možné implementovať v FPGA relatívne veľké množstvo.

Výhodou z hľadiska rýchlosti realizácie funkcionality navrhnutého obvodu je to, že jednotlivé jednotkové body sú, vzhľadom k danému prostému implikantu posudzované paralelne, čo by v prípade klasického CPU nebolo možné v takom rozsahu, v akom je to v tomto hardvérovom akcelerátore, ktorý umožňuje potenciálne paralelne posudzovať desiatky až stovky takýchto jednotkových bodov paralelne. Paralelizácia vykonávania operácií je realizovaná aj v rámci procesného elementu, kde je paralelne posudzovaný celý vektor tvoriaci množinu vstupných premenných Booleovskej funkcie. V prípade klasického počítača by bolo potrebné posudzovať tieto premenné pre daný implikant a jednotkový bod paralelne.

Ďalšou výhodou navrhnutého akcelerátora je, že počet jednotkových bodov, ktoré je možné spracovávať paralelne je síce obmedzený jeho hardvérovou konštrukciou, avšak počet prostých implikantov, ktoré možno vložiť na vstup B hardvérového akcelerátora s cieľom zhodnotiť, či je daný jednotkový bod pokrytý, a či existuje preň esenciálny prostý implikant, je neobmedzený.

Nevýhodou dosiahnutého riešenia je, že vzhľadom na zvolené kódovanie nie je dĺžka vstupného vektora A pre jednotkový bod zhodná s dĺžkou vstupného vektora B pre prostý implikant, čo potenciálne skomplikuje rozvoj tohto hardvérového akcelerátora v budúcnosti, keď sa budú realizovať ďalšie časti druhého kroku minimalizácie Booleovskej funkcie, konkrétne riešenia problému pokrytia riadku, resp. stĺpca mriežky prostých implikantov.

ACKNOWLEDGMENT

This research was supported by the Slovak Research and Development Agency, project number APVV-18-0214 and by KEGA Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under Grant No. 003TUKE-4/2017 Implementation of Modern Methods and Education Forms in the Area of Security of Information and Communication Technologies towards Requirements of Labor Market.

LITERATÚRA

- [1] A. Marquand, "XXXIII: On Logical Diagrams for n terms", *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 5. 12 (75), pp. 266–270. doi:10.1080/14786448108627104.
- [2] E.W. Veitch, "A Chart Method for Simplifying Truth Functions", *Proceedings of the 1952 ACM Annual Meeting (Pittsburgh, Pennsylvania, USA)*. New York, USA: Association for Computing Machinery (ACM), pp. 127–133. doi:10.1145/609784.609801.
- [3] M. Karnaugh, "The Map Method for Synthesis of Combinational Logic Circuits", *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, 72 (5), 1953, pp. 593–599. doi:10.1109/TCE.1953.6371932.
- [4] N.B. Dodge, "Simplifying Logic Circuits with Karnaugh Maps", *The University of Texas, Dallas, Erik Jonsson School of Engineering and Computer Science*.
- [5] M.V. Quine, "The Problem of Simplifying Truth Functions", *Amer. Math. Monthly*, Vol. 59, 1952, No. 8, pp. 521–531.
- [6] E.J. McCluskey, "Minimization of Boolean functions", *The Bell System Technical Journal*, 35, No. 5, Nov. 1956, pp. 1417–1444.
- [7] S.J. Hong, R.G. Cain and D.L. Ostapko, "MINI: A heuristic approach for logic minimization", *IBM Journal of Res. and Dev.*, Sept. 1974, pp. 443–458.
- [8] R.K. Brayton, G.D. Hachtel, C.T. McMullen and A.L. Sangiovanni-Vincentelli, "Logic Minimization Algorithms for VLSI Synthesis" (9th printing 2000, 1st ed.). Kluwer Academic Publishers. ISBN 0-89838-164-9.
- [9] R.L. Rudell, "Multiple-Valued Logic Minimization for PLA Synthesis", *Memorandum No. UCB/ERL M86-65*. Berkeley.
- [10] Espresso source code, University of California, Berkeley, <https://ptolemy.berkeley.edu/projects/embedded/pubs/downloads/espresso/>
- [11] P. McGeer, J.V. Sanghavi, R.K. Brayton and A.L. Sangiovanni-Vincentelli, "ESPRESSO-SIGNATURE: A new exact minimizer for logic functions", *Proceedings of DAC'93.*, 1996, pp. 432–440.
- [12] P. Fišer and J. Hlavička, BOOM, "A Heuristic Boolean Minimizer", *Computing and Informatics*. Vol. 22, pp 19–51, 25 Jun 2003, ISSN: 2585-8807.
- [13] P. Fišer and J. Hlavička, "Efficient Minimization Method for Incompletely Defined Boolean Functions", *Proceedings of the 4th International Workshop on Boolean Problems, University of Mining and Technology, Freiberg, Germany*, IWSBP 4, September 21–22, 2000, pp. 91–98, ISBN: 3-86012-124-3.
- [14] J. Hlavička and P. Fišer, A Heuristic Method of Two-Level Logic Synthesis. *Proceedings of The 5th World Multiconference on Systemics, Cybernetics and Informatics ISAS-SCI'2001, Orlando, Florida (USA), July, 22. –25., 2001, Vol. XII, pp. 283–288, ISBN 980-07-7541-2.*
- [15] P. Fišer and J. Hlavička, On the Use of Mutations in Boolean Minimization. *Proceedings of the Euromicro Symposium on Digital Systems Design, Warsaw (Poland) September, 4.–6. 2001, pp. 300–307.*
- [16] J. Hlavička and P. Fišer, BOOM — a Heuristic Boolean Minimizer. *Proceedings of the 2001 IEEE/ACM International Conference on Computer-Aided Design, ICCAD 2001, San Jose, CA, USA, November 4-8, 2001. IEEE Computer Society 2001, pp. 439–442, ISBN 0-7803-7249-2.*
- [17] P. Fišer and H. Kubátová, "Boolean Minimizer FC-Min: Coverage Finding Process", *Proc. 30th Euromicro Symposium on Digital Systems Design (DSD'04), Rennes (France), 31.8. - 3.9.04, pp. 152–159.*
- [18] P. Fišer and H. Kubátová, Two-Level Boolean Minimizer BOOM-II, *Proc. 6th International Workshop on Boolean Problems (IWSBP'04), Freiberg, Germany, 23.-24.9.2004, pp. 221–228.*
- [19] P. Fišer and H. Kubátová, Flexible Two-Level Boolean Minimizer BOOM II and Its Applications, *Proc. 9th Euromicro Conference on Digital Systems Design (DSD'06), Cavtat, (Croatia), 30.8. - 1.9.2006, pp. 369–376.*

Využitie fúzie modalít pri biometrickej autentifikácii užívateľa

Jozef BUČKO, Matúš PLEVA

Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita Košice, Slovenská Republika

jozef.bucko@student.tuke.sk, matus.pleva@tuke.sk

Abstrakt – Tento dokument pojednáva o biometrických technológiách využívaných na identifikáciu človeka na základe ľudskej rohovky a odtlačku prsta. Venuje sa určovaniu prahovej hodnoty, ktorá oddeľuje skupiny genuine (pravé) a impostor (falošné). Získané poznatky sú následne demonštrované aj na fúzii biometrických modalít, pri simulácii multimodálneho biometrického systému.

Kľúčové slová — biometria, EER, fúzia, multimodalita, odtlačok prsta, rohovka

I. ÚVOD

Biometria je oblasť vedy a inžinierstva zaoberajúca sa identifikáciou osôb. Táto vedná disciplína je založená na rôznych atribútoch, ktoré môžu byť fyzické, chemické alebo behaviorálne [1]. Spája aspekty biológie, štatistiky, forenznej vedy, ľudskeho správania, dizajnu, súkromia a bezpečnosti. Biometria je v dnešnom svete čoraz viac relevantná a potrebná, pretože narastá dopyt a potreba rôznych systémov na zabezpečenie prístupu a identity, ktoré si zakladajú na presnom determinovaní identity osoby. Hlavným zámerom systémov na zabezpečenie prístupu a identity je zamedziť podvodníkom prístup k zabezpečeným zdrojom a súborom.

II. BIOMETRIA A BIOMETRICKÉ METÓDY

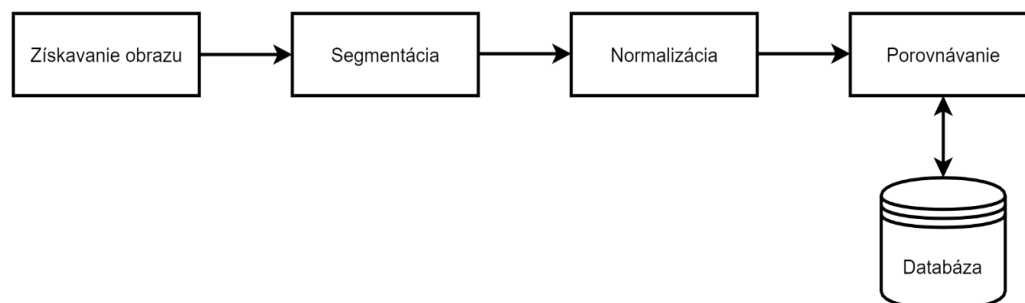
Biometria je vedná disciplína zaoberajúca sa identifikáciu individuálnych osôb na základe rôznych biometrických modalít. Biometrické modalitty delíme na tri kategórie.

- Fyziologické (ľudská rohovka, odtlačok prsta, tvárová biometria atď.)
- Behaviorálne (biometria hlasu, podpis/rukopis, štýl chôdze)
- Chemické (DNA)

Biometria ponúka množstvo výhod, oproti iným bezpečnostným technológiám ako napríklad heslo. Napríklad Negatívne rozpoznanie [1] je proces, ktorým systém určuje, že konkrétna individuálna osoba je v systéme už registrovaná, aj keď to daný človek popiera. Takýmto postupom sa dotyčný môže pokúsiť získať rôzne výhody registráciou sa pod rôznymi, účelovo vytvorenými prihlasovacími údajmi. Biometrické metódy môžu takýmto situáciám zabrániť.

III. ZÁKLADNÝ POSTUP SPRACOVANIA MODALÍT

V posledných rokoch neustále narastal dopyt po nových biometrických technikách ale tradičný postup spracovania ostal takmer nezmenený [2]. Základný postup používaný v systémoch na rozpoznávanie dúhovky sa skladá zo štyroch hlavných fáz [3]. Postup spracovania je zobrazený na obr. 1.



Obr. 1. Postup spracovania

A. Akvizícia

Pri získavaní obrazu (akvizícii) veľmi záleží na kvalite a rozlíšení získaného obrazu, ktorý budeme ďalej spracovávať a porovnávať. Počet zhodujúcich sa prvkov sa vo výsledku znižuje spolu s kvalitou a podmienkami pri získavaní obrazu. Kvalita obrazu nie je závislá len na kvalite akvizičného zariadenia (snímač, senzor, kamera), ale aj na okolitých podmienkach a zaobchádzaní so zariadením. Toto je obzvlášť dôležité pri získavaní snímok rohovky oka, pretože do procesu vstupuje mnoho oklúzných vplyvov ako napríklad odraz svetla od snímaného povrchu, rozmazanie obrazu, obranné mechanizmy oka ako napríklad rozťahovanie zrenice či žmurkanie. Najmä kvôli obranným mechanizmom oka sa komerčné akvizičné zariadenia na snímanie rohovky vybavujú podsvietením s vlnovou dĺžkou vyžarovaného svetla v intervale 700-900 nm. Oko na takéto svetlo nereaguje nijakými nežiadúcimi pohybmi a toto svetlo sa neodráža od povrchu oka ale prenikne až na povrch rohovky, kde odhalí skryté textúry, voľným okom ťažko badateľné [4].

B. Segmentácia

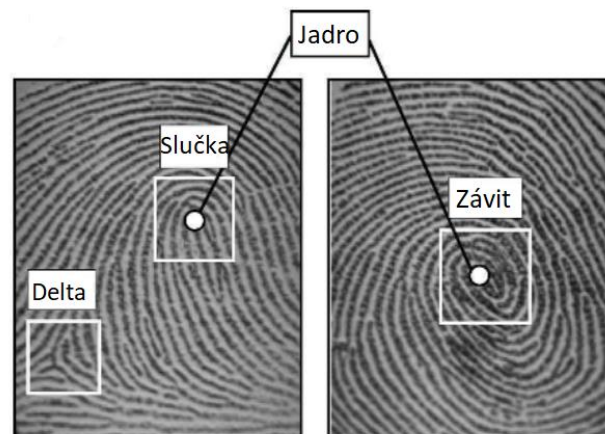
Úlohou predspracovania obrazu (segmentácie) je presne a robustne lokalizovať prvky biometrickej modalít a odstrániť šumové prvky, hlavne pri rohovke, pretože pri snímaní sa spolu s rohovkou na obraze vyskytujú aj šumové elementy ako napríklad mihalnice, či očné viečka.

1) Rohovka

Podľa Daugmanovej metódy sa po určení oblasti medzi zrenicou a vonkajším limbickým ohraničením získajú binárne prvky [3]. Pixely sú identifikované podľa uhla a vzdialenosti od zrenice k vonkajšiemu ohraničeniu. Takto vieme tolerovať odlišnosti zapríčinené rozširovaním zrenice v dôsledku svetelných podmienok.

2) Odtlačok prsta

Na snímke odtlačku (obr. 2) prsta sú ryhy svetlé a vystupujúce časti kože sú tmavé. Tieto krivky sú často vedené paralelne, niekedy sa rozdejoja, inokedy zaniknú. Pri analýze v globálnej mierke, vzory odtlačku odhalia jednu alebo viacero oblastí kde ryhy nevyzerajú tak, ako by sa dalo predpokladať. Tieto oblasti nazývame *singularity* a môžu byť klasifikované do troch kategórií: *slučka*, *delta*, *závit*. Zakončenia (*minuatie*) sú ďalšie dôležité prvky používané pri rozpoznávaní odtlačkov prstov. Sú to miesta kde dve ryhy kolidujú navzájom a vytvárajú jednu alebo sa ryha úplne stráca [1].



Obr. 2. Singularity v odtlačku prsta

C. Normalizácia

Hlavnou úlohou normalizácie je zmapovať textúru obrazu oka I o veľkosti $m \times n$ do Dugmenovho homogénneho modelu [1]. Teda vykreslenie mapy R , ktorá priradí každej usporiadanej dvojici (θ, r) uhla θ a radiálnej vzdialenosti r prislúchajúcu pôvodnú polohu $R(\theta, r)$ v rámci I , bez ohľadu na rozšírenie zrenice:

$$R : [0, 2\pi) \times [0, 1] \rightarrow [0, m] \times [0, n].$$

Z týchto dvoch predprocesných úkonov „segmentácie“ a „normalizácie“ je vo výskumnej oblasti viac pozornosti venovanej hlavne segmentácii, tvorbe špecifických postupov zameraných na detekciu šumových elementov a na správne určenie a extrahovanie prvkov

D. Porovnanie

Obrazy biometrických modalít sa po spracovaní kódujú. Po kódovaní väčšinou vznikne matica, v ktorej sú vyjadrené extrahované prvky. Zakódovaný obraz rohovky sa nazýva *iris kód*, dáta,

extrahované z odtlačkov prsta reprezentujúce lokáciu (poloha) a smer zakončení, sa kódujú do *matice zakončení*.

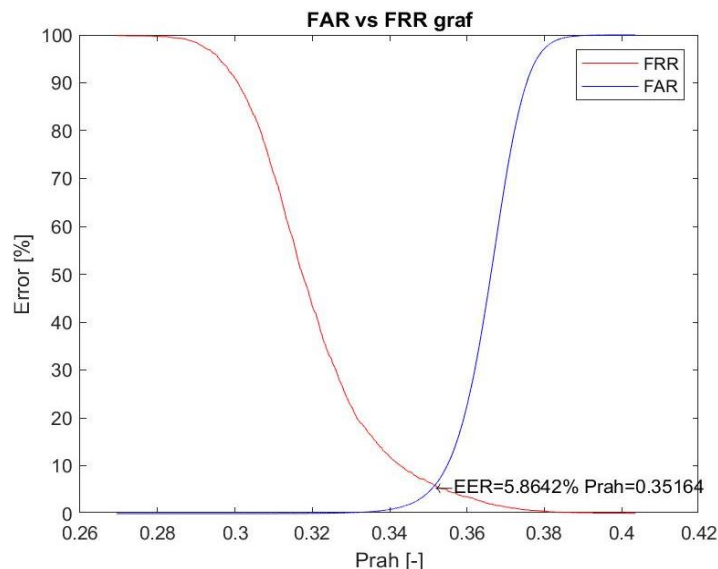
Ak chceme porovnať dve vzorky rovnakej biometrickej modality, porovnáваме dané matice. Na porovnanie sa používajú rôzne metódy, ktorých výstupom je nejaké skóre. Takéto skóre môže byť v ľubovoľnom rozsahu či význame, tento rozsah a aj jeho význam však musí byť pri oboch porovnávaných obrazoch jednej modality rovnaký. Ako príklad môžeme uviesť *Hammingovu vzdialenosť* (HV) dvoch iris kódov, čo je vzdialenosť dvoch binárnych slov rovnakej dĺžky [5]. Čím sú si dva porovnávané obrazy podobnejšie, tým je ich Hammingova vzdialenosť nižšia. Opakom Hammingovej vzdialenosti je *stupeň podobnosti* (SP). Rozsah týchto parametrov sa môže pohybovať v intervale $\langle 0;1 \rangle$.

IV. EQUAL ERROR RATE

Dve matice môžeme priradiť jednej osobe iba v prípade, že ich HV je nižšia ako nejaký konkrétne stanovený prah nazývaný prahová hodnota. V teoretickej rovine, HV známej osoby (*genuine*) by vždy mala byť nižšia ako HV cudzej osoby (*impostor*). Ak by toto tvrdenie bolo pravdivé, existovala by jedna prahová hodnota, ktorá by oddeľovala tieto dve skupiny ľudí.

Kvôli niekoľkým dôvodom však tento predpoklad nie je správny v reálnych podmienkach. V niektorých prípadoch, matice *impostorov* môžu vygenerovať HV nižšiu, ako je HV niektorých matíc *genuinov*. Z tohto dôvodu, keďže sa vyberá nejaká prahová hodnota na klasifikáciu, nie je možné vylúčiť výskyt chýb pri klasifikácii.

Ak je vybraná prahová hodnota HV tak nízka, že skutočne žiadny impostor nebude pod túto hranicu spadať, teda žiaden iris kód nebude falošne akceptovaný (FAR – False acceptance rate). Na druhej strane iris kódy *genuinov*, ktorých HV je vyššia ako je prahová hodnota budú falošne odmietnuté (FRR – False rejection rate). Naopak je možné vybrať prahovú hodnotu HV tak vysokú, že žiadny z *genuinov* nebude falošne zamietnutý ale pri takomto scenári, príliš vysoký počet *impostorov* je falošne akceptovaných. Pri výbere prahovej hodnoty niekde medzi týmito dvomi hodnotami sa vyskytnú aj falošné akceptovania aj falošné zamietnutia [6].



Obr. 3 – FAR vs FRR graf

V. MULTIMODÁLNA FÚZIA BIOMETRICKÝCH MODALÍT

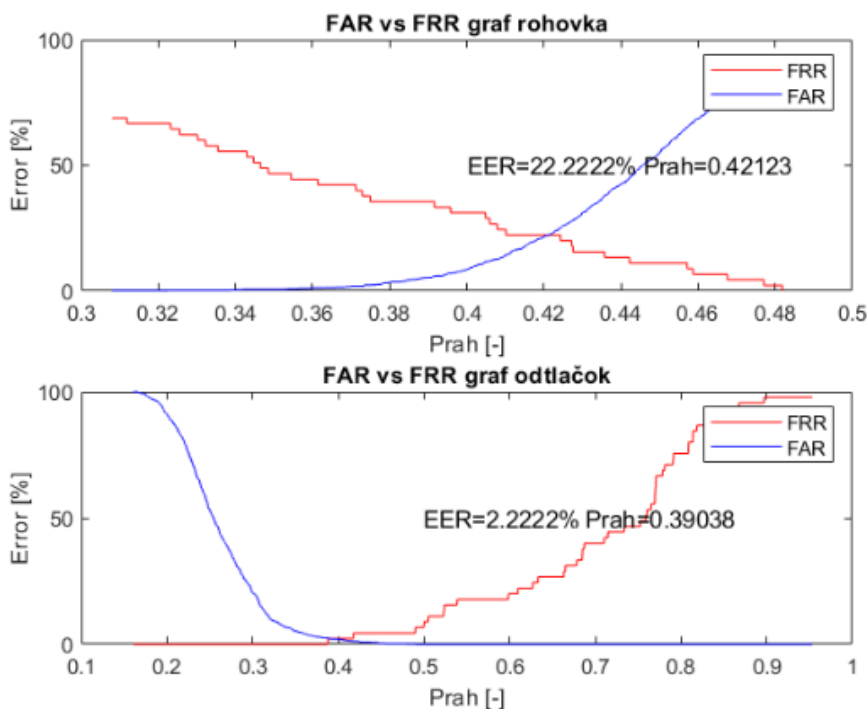
Systémy používajúce *multimodálnu fúziu biometrických modalít* sú systémy, ktoré na rozdiel od unimodálnych používajú na identifikáciu viacero biometrických modalít naraz [7] [8]. Na demonštráciu takého systému, ktorý funguje na princípe fúzie identifikácie na základe odtlačku prsta a obrazu rohovky, sme využili databázu pätnástich subjektov, ktorá pre každý subjekt obsahuje tri snímky rohovky jedného oka a tri snímky odtlačku jedného prsta. Snímky od jednotlivých užívateľov sme navzájom porovnali s prislúchajúcimi snímkami od daného užívateľa. Ďalej sme v prostredí Matlab vytvorili skript, ktorý porovná každú snímku, od všetkých používateľov so snímkami od všetkých ostatných používateľov. Teda odtlačky prsta s odtlačkami prsta a snímky rohovky so snímkami rohovky. Získali sme dáta, ktoré reprezentujú dosiahnuté skóre pri porovnávaní daných vzoriek. Pri odtlačkoch prsta je to stupeň podobnosti a pri rohovke je to Hammingova vzdialenosť, ktorá je vo všeobecnosti nepriamoúmerná stupňu podobnosti.

- HV $\in \langle 0;1 \rangle$ pričom čím vyššia je výsledná hodnota, tým sú si vzorky rozdielnejšie

- $SP \in <0;1>$ pričom čím vyššia je výsledná hodnota, tým sú si vzorky podobnejšie

Získali sme 4 vektorové matice. Tieto matice sme následne použili na vykreslenie grafov FAR a FRR, čo nám v zapätí umožňuje určiť hodnotu EER a aj prahovú hodnotu, ktorá najlepšie oddeľuje skupinu *impostorov* od skupiny *genuine*. Vykreslené grafy môžete vidieť na Obr. 4. Zo získaných výsledkov je vidieť že odtlačok prsta je v našich podmienkach o 20% lepší identifikátor ako rohovka. Tento poznatok nám pomôže pri určovaní váhy jednotlivých modalít pri fúzii.

Na simuláciu systému multimodálnej biometrie využijeme *fúzijský stupeň podobnosti S* [9]. Tento parameter získame ako vážený aritmetický priemer HV a SP podľa vzorca, pričom parameter získaný spracovaním odtlačkov prstov (SP) bude mať váhu o 20% vyššiu, ako parameter získaný spracovaním obrazov ľudskej rohovky.

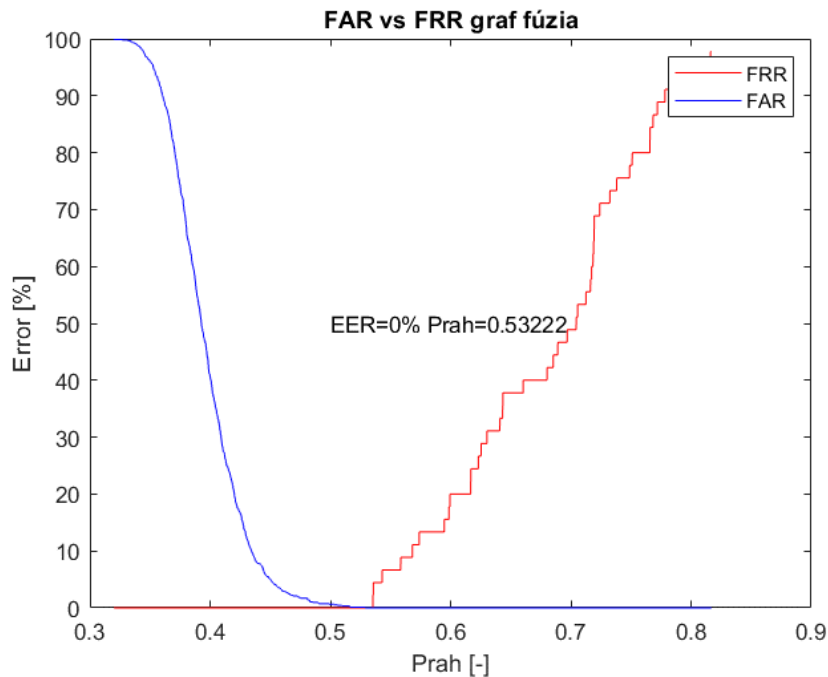


Obr. 4 – EER pre odtlačok prsta a rohovku

$$S = \frac{1 - \text{Hammingova vzdialenosť} + 1,2 * \text{Stupeň podobnosti}}{2,2}$$

Pri vyhodnocovaní výsledkov databázy 15 ľudí sme zistili že odtlačok prsta je vo všeobecnosti kvalitnejšou biometrickou modalitou než rohovka, za použitia spracovacích techník, ktoré používajú naše nástroje. Vo výsledku to znamená že pri fúzii má odtlačok prsta 1,2 násobnú váhu oproti rohovke. Taktiež je dôležité uvedomiť si, že výstup získaný z dvoch rôznych algoritmov párovania je nehomogénny. Tento problém môže byť vyriešený normalizáciou vstupných dát

Získané matice reprezentujúce fúzijský stupeň podobnosti *S* použijeme na vykreslenie kriviek FAR a FRR. Po zistení parametra EER pri fúzii dát, vidíme že tento parameter klesol na nulu. To znamená, že pri zvolení prahovej hodnoty 0,53222 žiaden z *impostorov* nebude falošne prijatý systémom a rovnako ani žiaden z *genuine* užívateľov nebude systémom falošne odmietnutý. Toto tvrdenie platí za použitia nami použitých algoritmov na spracovanie dát a databázy pätnástich účastníkov.



Obr. 5 – EER pre fúziu

VI. ZÁVER

Analyzovali sme problematiku biometrických technológií využívajúce biometrické modality ľudskej rohovky a odtlačku prsta. Preštudovaním množstva literatúry a publikácií sme sa oboznámili s rôznymi postupmi na spracovanie týchto biometrických modalít ako aj s rôznymi prístupmi na štatistickú analýzu vyhodnocovania kvality týchto postupov ako napríklad, určenie bodu rovnakej miery chybovosti či simulácia multimodálneho biometrického systému, ktorý využíva fúziu dvoch biometrických modalít. V budúcnosti chceme otestovať algoritmy hlbokých neurónových sietí v oblasti biometriky [10].

POĎAKOVANIE

Táto práca bola podporená Agentúrou na podporu výskumu a vývoja vďaka projektu APVV-15-0731 „Multimodálna interakcia človek-robot s využitím cloudových prostriedkov“, Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva vedy výskumu a športu Slovenskej republiky vďaka projektu číslo KEGA 009TUKE-4/2019 s názvom „Inovácia obsahu a príprava učebných textov pre predmet Biometrické systémy bezpečnosti“, Vedeckou grantovou agentúrou realizáciou výskumného projektu VEGA 1/0753/20 „Robustné rečové technológie metódami hlbokého učenia“ financovaného z prostriedkov Ministerstva školstva, vedy, výskumu a športu SR a nakoniec aj cez Operačný program Výskum a inovácie, financovaného z Európskeho fondu regionálneho rozvoja v rámci projektu „Univerzitný vedecký park TECHNICOM pre inovačné aplikácie s podporou znalostných technológií – II. fáza“ Kód ITMS: 313011D232.

REFERENCES

- [1] Handbook of Biometrics, A. K. Jain, P. Flynn, A. A. Ross, ISBN-13: [978-0-387-71040-2](https://doi.org/10.1002/9781119816444)
- [2] Bowyer, K., Hollingsworth, K., Flynn, P.: Image understanding for iris biometrics: A survey. *Comput. Vis. Image Understand.* 110(2), 281–307 (2007). doi: [10.1016/j.cviu.2007.08.005](https://doi.org/10.1016/j.cviu.2007.08.005)
- [3] C. Rathgeb, A. Uhl, P. Wild, *Iris Biometrics: From Segmentation to Template Security*, ISBN [978-1-4614-5570-7](https://doi.org/10.1007/978-1-4614-5570-7)
- [4] Kollias, Nikiforos. (1995). The Spectroscopy of Human Melanin Pigmentation. [10.1016/0738-081X\(95\)00075-Q](https://doi.org/10.1016/0738-081X(95)00075-Q)
- [5] Robinson, Derek J. S. (2003). *An Introduction to Abstract Algebra*. Walter de Gruyter. pp. 255–257. ISBN 978-3-11-019816-4. <https://doi.org/10.1515/9783110198164>
- [6] Singh, Nilu & Khan, Prof. Raees & Pandey, Rajshree. (2014). Equal Error Rate and Audio Digitization and Sampling Rate for Speaker Recognition System. *Advanced Science Letters*. 20. [10.1166/asl.2014.5458](https://doi.org/10.1166/asl.2014.5458).
- [7] Singh, M., Singh, R. and Ross, A., 2019. A comprehensive overview of biometric fusion. *Information Fusion*, 52, pp.187-205. <https://doi.org/10.1016/j.inffus.2018.12.003>
- [8] Pleva, M., Bours, P., Ondáš, S. et al. Improving static audio keystroke analysis by score fusion of acoustic and timing data. *Multimed Tools Appl* 76, 25749–25766 (2017). <https://doi.org/10.1007/s11042-017-4571-7>
- [9] J. Fierrez-Aguilar, Y. Chen, J. Ortega-Garcia, A. K. Jain, Incorporating image quality in multi-algorithm fingerprint verification, in: *International Conference on Biometrics*, 2005, pp. 213–220. https://doi.org/10.1007/11608288_29
- [10] Messerschmidt, M. and Pleva, M., 2019, November. Biometric Systems Utilizing Neural Networks in the Authentication for E-learning Platforms. In *2019 17th International Conference on Emerging eLearning Technologies and Applications (ICETA)* (pp. 518-523). IEEE. DOI: [10.1109/ICETA48886.2019.9040132](https://doi.org/10.1109/ICETA48886.2019.9040132)

Vylepšenie kvantizácie pre obrazové šifrovacie algoritmy

Jakub ORAVEC, Luboš OVSENÍK, Ján TURÁN

Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach

{jakub.oravec, lubos.ovsenik, jan.turan}@tuke.sk

Abstrakt – Tento príspevok sa bude zaoberať vylepšením kvantizácie v obrazových šifrovacích algoritmoch, ktoré využívajú logistické zobrazenie. Toto zobrazenie je možné považovať za jedno z najpoužívanejších v oblasti obrazových šifrovacích algoritmov, aj keď má niekoľko známych nedostatkov. Niektoré z nich už boli vyriešené a tieto riešenia boli následne publikované, ale aj v týchto prípadoch je potrebné uvažovať o výpočtovej náročnosti jednotlivých riešení. Tento príspevok sa bližšie zaoberá problematikou nerovnomerného rozloženia hodnôt iterátov a existenciou závislostí medzi hodnotami susedných iterátov. Miernou zmenou predpisu kvantizácie sa podarilo dosiahnuť potlačenie uvedenej dvojice nevýhod logistického zobrazenia, čo je ilustrované aj pomocou histogramov rozloženia hodnôt iterátov a Poincarého diagramov. Príspevok poskytuje aj stručný náhľad na výpočtovú náročnosť riešenia.

Príčné slová – histogram, kvantizácia, logistické zobrazenie, obrazové šifrovacie algoritmy, Poincarého diagram

I. ÚVOD

Prvé z obrazových šifrovacích algoritmov boli predstavené koncom 90. rokov minulého storočia. Spopularizované boli najmä po uvedení riešenia od J. Fridrich [1] v roku 1998. Zatiaľ čo v období ich uvedenia mnohí výskumníci uvažovali o týchto algoritmoch ako o alternatívnom riešení ku použitiu tzv. konvenčných šifrovacích algoritmov (ako napr. Advanced Encryption Standard, AES), neskôr sa objavilo viacero príspevkov, ktoré poukazovali na mnohé nedostatky obrazových šifrovacích algoritmov [2], [3].

Veľká časť nedostatkov súvisí s jednoduchosťou použitých matematických nástrojov, resp. s ich nevhodnou implementáciou [4]. Tieto nástroje, slúžiace napr. na generovanie pseudo-náhodných postupností (PNP) boli pritom vyberané tak, aby obrazové šifrovacie algoritmy dosahovali v oblasti šifrovania špecifického druhu dát (intenzít obrazových prvkov) výsledky porovnateľné s konvenčnými šifrovacími algoritmami. Pre dosiahnutie tohto cieľa bolo potom potrebné navrhnuť a aplikovať viacero opráv, pričom vo väčšine prípadov takéto opravy negatívne vplyvajú na výpočtovú náročnosť celého riešenia [5].

V tomto príspevku sa budeme zaoberať problematikou kvantizácie prvkov, ktoré sú výsledkom jedného z najčastejšie používaných matematických nástrojov v oblasti obrazových šifrovacích algoritmov. Keďže vo väčšine prípadov je kvantizáciu potrebné vykonať, malá zmena jej predpisu by nemala znamenať veľký nárast výpočtovej náročnosti. Na druhej strane je možné uvažovať o výhodách, akými sú rovnomerné rozloženie hodnôt týchto prvkov a potlačenie závislostí medzi nasledujúcimi prvkami.

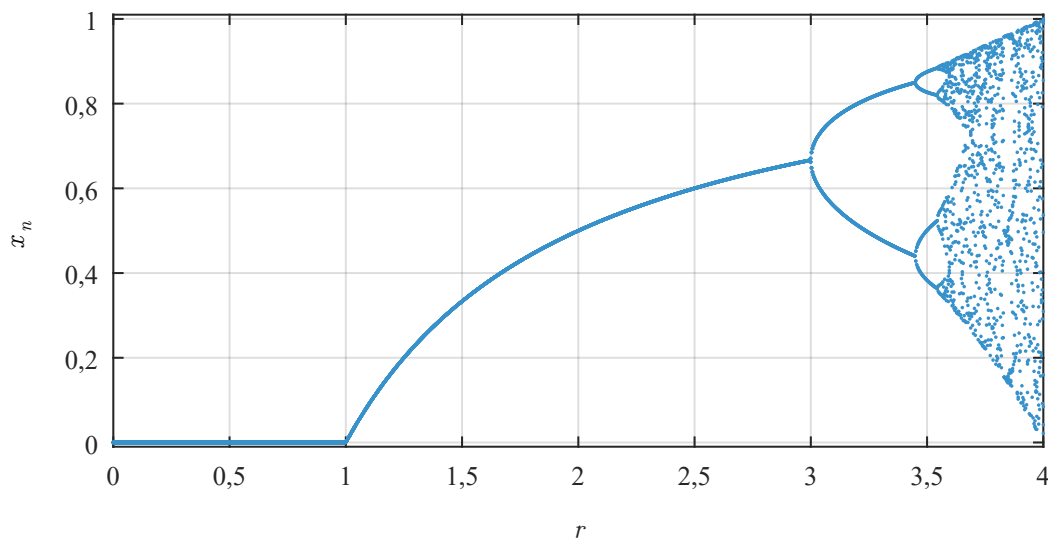
II. LOGISTICKÉ ZOBRAZENIE

Logistické zobrazenie (LZ) patrí k jednorozmerným chaotickým zobrazeniam s jedným parametrom. Vlastnosti takýchto zobrazení boli študované od 60. rokov minulého storočia, konkrétne prvá výrazná práca týkajúca sa LZ bola predstavená v roku 1976 R. Mayom [6]. Chaotickosť tohto zobrazenia spočíva najmä v správaní hodnôt získaných iterátov, ktoré je pre určité intervaly parametra predpokladateľné, no pre iné intervaly je značne zložitá. Predpis LZ je možné matematicky vyjadriť ako (1):

$$x_{n+1} = r \cdot x_n(1 - x_n), \quad (1)$$

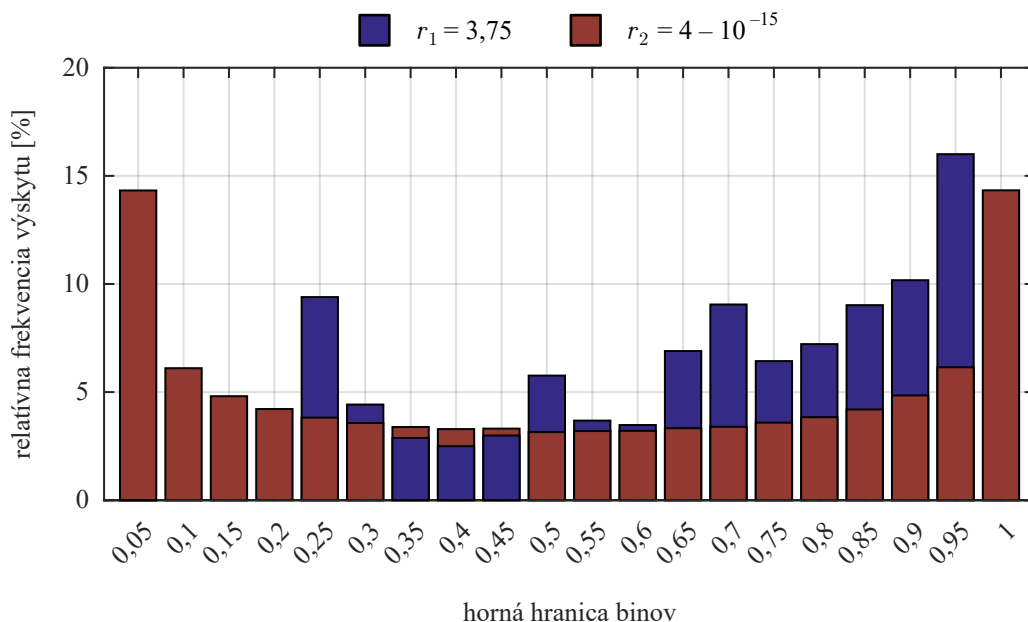
kde $x_{n+1} \in (0; 1)$ je hodnota nasledujúceho iterátu, $r \in (0; 4)$ je parameter zobrazenia a $x_n \in (0; 1)$ je hodnota aktuálneho iterátu. Pre generovanie x_1 je potrebná tzv. počiatočná hodnota $x_0 \in (0; 1)$. Z rovnice (1) je zjavné, že LZ je iteratívna funkcia, keďže pri nemennej hodnote parametra r platí $x_{n+1} = f(x_n)$.

Závislosť správania hodnôt generovaných iterátov na hodnote parametra r je možné ilustrovať pomocou tzv. bifurkačného diagramu. Príklad takéhoto diagramu je znázornený na Obr. 1.



Obr. 1 Bifurkačný diagram logistického zobrazenia

Bifurkačný diagram poukazuje mimo iného aj na skutočnosť, že ak sa hodnota parametra r neblíži ku hodnote 4, hodnoty generovaných iterátov nezaplnia celý interval $x_n \in (0; 1)$. Rozloženie hodnôt iterátov je možné demonštrovať aj histogramom. Histogram je stĺpcový graf, v ktorom každý stĺpec, nazývaný aj bin (z angl. bin v zmysle hlasovacej urny) vyjadruje početnosť hodnôt v určitom intervale. Histogram s hodnotami celkovo 10^6 iterátov x_n , ktoré sú zaradené do jedného z 20 rovnomerne rozložených binov pre $r_1 = 3,75$ a $r_2 = 4 - 10^{-15}$ je ilustrovaný na Obr. 2.



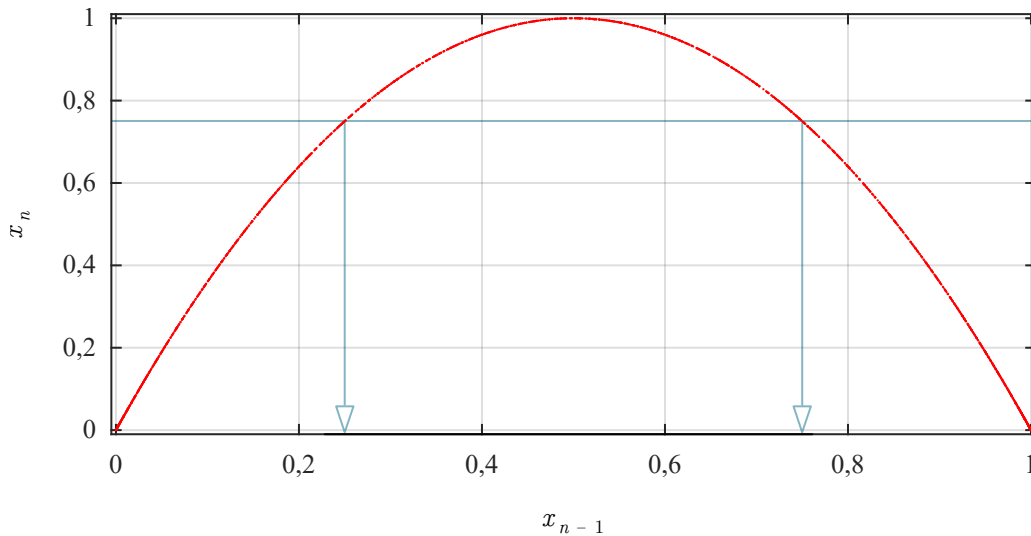
Obr. 2 Histogram iterátov logistického zobrazenia

Z histogramu pre hodnotu parametra r_1 (modré stĺpčeky) môžeme vidieť, že prvé štyri a zároveň posledný bin histogramu sú prázdne, teda nevygenerovalo sa žiadne $x_n \in (0; 0,2) \cup (0,95; 1)$. Pre hodnotu parametra r_2 (bordové stĺpčeky) sú síce všetky biny histogramu zaplnené, avšak stále je zjavné nerovnomerné rozloženie hodnôt iterátov do jednotlivých binov histogramu.

Okrem nerovnomerného rozloženia hodnôt iterátov sa v tomto príspevku budeme venovať aj závislosti dvojíc nasledujúcich iterátov. Ako už bolo spomenuté, pri nemennej hodnote parametra r je predpis LZ príkladom iteratívnej funkcie. Ak použijeme substitúciu $x_{n+1} \rightarrow x_i$ a $x_n \rightarrow x_{i-1}$, úpravou predpisu (1) dostávame kvadratickú rovnicu, ktorej koreňmi sú $x_{i-1,1}$ a $x_{i-1,2}$ (2):

$$x_{i-1} = \frac{1}{2} \left(1 \mp \sqrt{1 - 4 \cdot \frac{x_i}{r}} \right). \quad (2)$$

Na skutočnosť, že pri známej hodnote parametra r je možné z každej hodnoty aktuálneho iterátu x_n získať práve dva korene x_{n-1} poukazuje aj Poincarého diagram. Tento diagram má hodnoty x_n na vertikálnej osi a hodnoty x_{n-1} na horizontálnej osi. Príklad Poincarého diagramu s 2000 párami hodnôt iterátov a hodnoty x_{n-1} získané pre x_n pri $r = 4 - 10^{-15}$ ukazuje Obr. 3. Hodnoty predošlých iterátov sa určujú ako x -ové súradnice priesečníkov úsečky $y = x_n$ a paraboly, ktorá je daná kvadratickou rovnicou získanou z predpisu (1).



Obr. 3 Poincarého diagram a spôsob určenia hodnôt predošlých iterátov

Dve prezentované vlastnosti LZ patria ku jeho nedostatkom, ktoré môžu vytvárať pochybnosti o bezpečnosti riešení, ktoré používajú LZ. Rozloženie hodnôt iterátov aj ich závislosti je však možné potlačiť vylepšenou variantou kvantizácie, ktorá posúva desatinné miesta iterátov doľava. Kvantizácia je navyše potrebná vo väčšine obrazových šifrovacích algoritmov, pretože tieto pracujú buď s prvkami PNP z množiny $\{0; 1; \dots; 255\}$ alebo z množiny $\{0; 1\}$. Mierna zmena kvantizačného predpisu teda nemusí významne ovplyvniť trvanie tejto operácie.

III. NAVRHNUTÉ RIEŠENIE

Pre kvantizáciu iterátov LZ je možné použiť predpis (3):

$$x'_n = \lfloor (max_{val} + 1) \cdot x_n \rfloor, \quad (3)$$

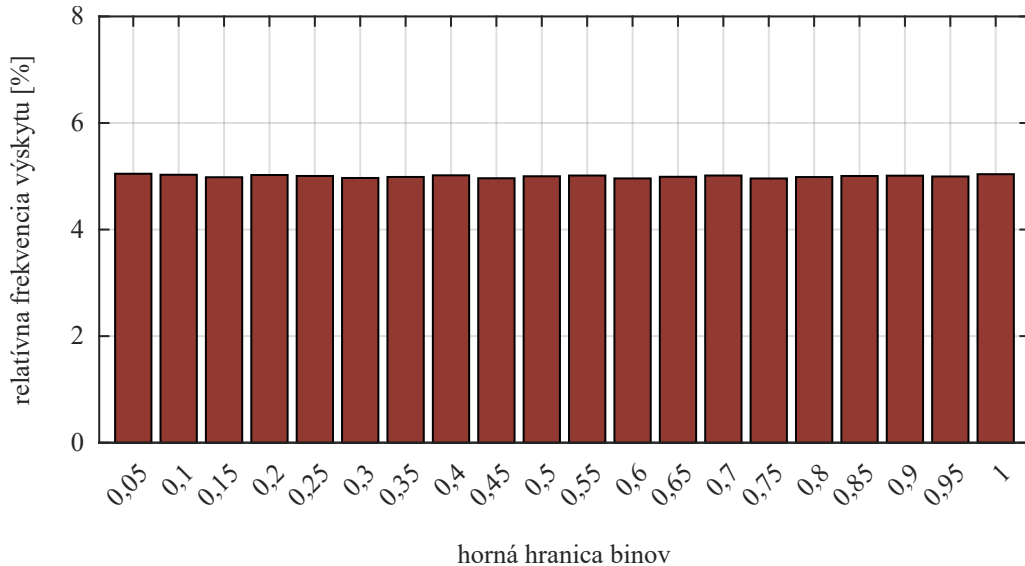
kde x'_n je postupnosť okvantovaných iterátov, max_{val} je maximálna možná hodnota okvantovaného iterátu a x_n je postupnosť iterátov pred kvantizáciou.

Miernou zmenou predpisu (3) získame navrhované riešenie, ktoré spočíva v posune štyroch desatinných miest pred desatinnú čiarku a odstánenia celočíselnej časti desatinného čísla (4):

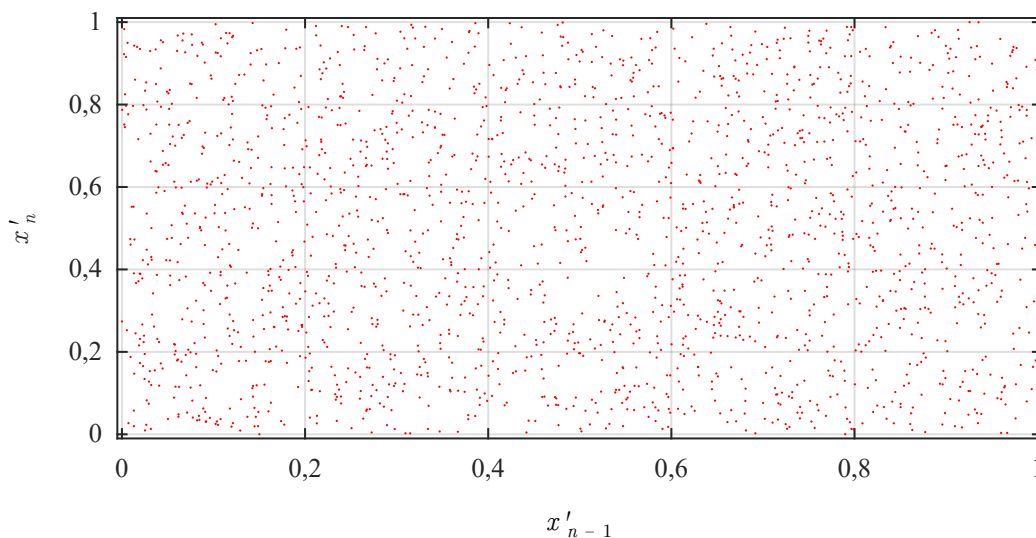
$$x'_n = \lfloor (max_{val} + 1) \cdot (10^4 \cdot x_n \pmod{1}) \rfloor. \quad (4)$$

Vplyv posunu o štyri desatinné miesta je ilustrovaný na Obr. 4 a Obr. 5. Histogram bol zostavený z rovnakého počtu iterátov ako pri Obr. 2 (10^6 iterátov) s použitím hodnôt $r = 4 - 10^{-15}$ a $x_0 = 0,5$.

Na Obr. 4 je možné vidieť ako posun o štyri desatinné miesta zmenil rozloženie hodnôt iterátov x'_n . Toto rozloženie je blízke rovnomernému (cca 5 % relatívneho výskytu pre každý z 20 binov histogramu). Experimentálne bolo zistené, že pri posune o menší počet desatinných miest ostávajú stále viditeľné špičky na okrajoch histogramu. Pri posune o väčší počet desatinných miest narastá



Obr. 4 Histogram iterátov logistického zobrazenia po vylepšenej kvantizácii



Obr. 5 Poincarého diagram po vylepšenej kvantizácii

negatívny vplyv dynamickej degradácie chaosu [4], ktorá je zapríčinená reprezentovaním hodnôt iterátov x'_n dátovým typom s konečnou presnosťou (väčšinou double precision) [7].

Poincarého diagram na Obr. 5 ukazuje, že vďaka vyrovnanjšiemu rozloženiu bodov v diagrame nie je možné jednoducho určiť hodnotu predchádzajúceho iterátu x'_{n-1} z hodnoty aktuálneho iterátu x'_n . Je to zapríčinené rôznym počtom bodov na úsečke $y = x'_n$ a aj skutočnosťou, že po získaní koreňov x'_{n-1} sa jedná o hodnoty po kvantizácii. Takéto hodnoty už neobsahujú prvé štyri desiatinné miesta, ktoré je potrebné kvôli irreverzibilnej povahe kvantizácie odhadnúť.

Miernou nevýhodou navrhnutého riešenia je zvýšenie výpočtovej náročnosti. Na ilustráciu miery zvýšenia bol vykonaný jednoduchý experiment vo výpočtovom prostredí MATLAB R2015a na PC s 2,5 GHz procesorom a 12 GB operačnej pamäte. V tomto prostredí sa vypočítala postupnosť 10^6 iterátov LZ s $r = 4 - 10^{-15}$ a $x_0 = 0,5$. Kvantizácia hodnôt týchto iterátov predpisom (3) trvala približne 4,4 ms, zatiaľ čo ich kvantizácia predpisom (4) trvala približne 48,1 ms. Uvedené hodnoty sú aritmetickými priermi súboru 100 opakovaných meraní. Z experimentu vyplýva, že použitie vylepšenej kvantizácie spôsobí viac ako desaťnásobný nárast výpočtovej náročnosti. Vzhľadom na potlačenie nedostatkov LZ sa však aj tento nárast javí ako akceptovateľný.

IV. ZÁVER

Tento príspevok sa zaoberal predstavením vylepšenia kvantizácie, ktoré pri použití v oblasti obrazových šifrovacích algoritmov pomáha potlačiť niektoré nedostatky logistického zobrazenia.

Analyzovanými nedostatkami boli nerovnomerné rozloženie hodnôt iterátov a existencia závislosti medzi hodnotami dvoch susedných iterátov. Vďaka pomerne jednoduchej úprave bežne používaného predpisu kvantizácie iterátov bolo dosiahnuté približne rovnomerné rozloženie hodnôt iterátov ako aj nemožnosť získania koreňov predošlého iterátu z hodnoty aktuálneho okvantovaného iterátu. Tieto vlastnosti boli dosiahnuté za cenu mierneho zvýšenia výpočtovej náročnosti predpisu kvantizácie. Nárast sa môže zdať byť vysoký, avšak pri porovnaní s technikami, ktoré využívajú iné obrazové šifrovacie algoritmy je akceptovateľný. Detailnejšia analýza ďalších techník z ostatných obrazových šifrovacích algoritmov sa javí ako pomerne rozsiahla téma a je možné považovať ju za námet ďalších príspevkov.

POĎAKOVANIE

Táto práca bola podporovaná nasledujúcimi výskumnými projektami: APVV-17-0208 – Odolné mobilné siete na doručovanie obsahu a VEGA 1/0584/20 – UWB senzorové systémy monitorovania osôb pracujúce v reálnych podmienkach.

LITERATÚRA

- [1] J. Fridrich, „Symmetric Ciphers Based on Two-Dimensional Chaotic Maps,“ *International Journal of Bifurcation and Chaos*, 1998, roč. 8, č. 6, s. 1259–1284. ISSN: 0218-1274. DOI: 10.1142/S021812749800098X.
- [2] D. Arroyo, G. Alvarez, V. Fernandez, „On the Inadequacy of the Logistic Map for Cryptographic Applications,“ *Proceedings of 10th Spanish Meeting on Cryptology and Information Security*. Salamanca (Spain), 2008, s. 1–6.
- [3] M. Preishuber, T. Hütter, S. Katzenbeisser, A. Uhl, „Depreciating Motivation and Empirical Security Analysis of Chaos-based Image and Video Encryption,“ *IEEE Transactions on Information Forensics and Security*, 2018, roč. 13, č. 9, s. 2137–2150. ISSN: 1556-6013. DOI: 10.1109/TIFS.2018.2812080.
- [4] J. Oravec, J. Turán, L. Ovseník, T. Huszaník, „A Chaotic Image Encryption Algorithm Robust against the Phase Space Reconstruction Attacks,“ *Acta Polytechnica Hungarica*, 2019, roč. 16, č. 3, s. 37–57. ISSN: 1785-8860. DOI: 10.12700/aph.16.3.2019.3.3.
- [5] G. Alvarez, J. M. Amigó, D. Arroyo, S. Li, *Lessons Learnt from the Cryptanalysis of Chaos-Based Ciphers*. Springer Verlag, Berlin-Heidelberg, 2011, s. 257–295. ISBN: 978-36-4220-541-5. DOI: : 10.1007/978-3-642-20542-2_8.
- [6] R. May, „Simple Mathematical Models with Very Complicated Dynamics,“ *Nature*, 1976, roč. 261, č. 5560, s. 459–467. ISSN: 0028-0836. DOI: 10.1038/261459a0.
- [7] „IEEE 754-2019: IEEE Standard for Floating-point Arithmetic.“ DOI: 10.1109/IEEESTD.2019.8766229.

DC menič pre napájanie univerzálneho motora

¹Peter Girovský, ²Ján Kaňuch

^{1,2} Katedra elektrotechniky a mechatroniky, Fakulta elektrotechniky a informatiky, Technická univerzita, Letná 9, 04200 Košice, Slovenská Republika

¹peter.girovsky@tuke.sk, ²jan.kanuch@tuke.sk

Abstrakt—Tento príspevok sa zaoberá návrhom jednosmerného meniča pre napájanie univerzálneho motora. Univerzálne motory nachádzajú široké uplatnenie v rôznych aplikáciách a môžu byť prevádzkované so striedavým alebo jednosmerným napätím. Veľkou výhodou univerzálnych motorov je najmä vysoký rozběhový moment, veľmi kompaktná konštrukcia a vysoká prevádzková rýchlosť. Nevýhodou univerzálnych motorov je najmä ich komutátor a elektromagnetické rušenie, ktoré vyžaruje a ktoré môže ovplyvňovať okolité spotrebiče. Nakoľko z predchádzajúcich meraní bolo zistené, že univerzálny motor dosahuje lepšie prevádzkové vlastnosti pri napájaní jednosmerným napätím, je v tomto príspevku prezentovaný návrh jednosmerného meniča pre napájanie tohto univerzálneho motora.

Kľúčové slová—univerzálny motor, napájanie, menič, meranie

I. ÚVOD

Univerzálne motory patria k najviac používaným elektrickým strojom v domácich elektrospotrebičoch a v dielenskom ručnom náradí vďaka ich výborným regulačným vlastnostiam. Mnohostranné využitie univerzálnych motorov je dané aj skutočnosťou, že tieto motory môžu byť napájané nielen striedavým, ale aj jednosmerným napätím.



Obr. 1 Univerzálny motor

V oboch prípadoch je rýchlosť motora riadená veľkosťou napájacieho napätia. Univerzálny motor je známy aj pod názvami ako striedavý sériový motor alebo tiež ako striedavý komutátorový motor. Tento motor sa skladá zo statora s dvoma koncentrovanými budiacimi vinutiami zapojenými do série, rotora s vinutím, komutátora a dvoch uhlíkov. Vinutia statora a rotora sú zapojené v sérii k zdroju elektrickej energie. Sériové spojenie vinutia statora a rotora je vytvorené trecími kontaktmi komutátora na rotore a dvoch uhlíkov.

Na rozdiel od jednosmerného motora s cudzím budením, univerzálny motor napájaný konštantným napätím nemá stabilnú rýchlosť. Pri konštantnom napätí (AC alebo DC) motor beží tak rýchlo, ako to dovoľí jeho zaťaženie a krútiaci moment motora klesá s rýchlosťou. Avšak univerzálne motory majú veľmi vysoký pomer krútiaceho momentu k hmotnosti zo všetkých typov elektromotorov.

Hlavné vlastnosti, ktoré sú typické pre univerzálne motory sú nasledovné:

- vysoká rýchlosť, vysoká účinnosť, relatívne nízka životnosť (neodporúča sa nepretržitá a aj dlhá doba prevádzky),
- medzi komutátorom a uhlíkmi je iskrenie, vytvára sa ozón a generuje EMI,

- prevádzka je buď na jednosmerné alebo jednofázové striedavé napätie, pričom prevádzkový výkon univerzálneho motora je 50 až 1000 W a jeho rýchlosť je 10000 - 20000 otáčok za minútu.

Univerzálny motor sa využíva prevažne v aplikáciách, ktoré vyžadujú veľký rozbehový moment a malú zmenu momentu za chodu, vysokú rýchlosť a krátky čas prevádzky s dlhšou prestávkou. Niektoré z typických aplikácií sú nasledovné:

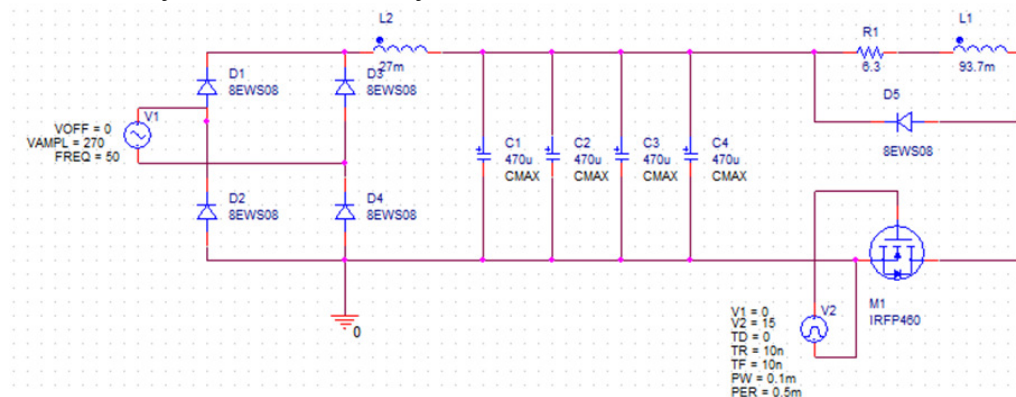
- domáce a kancelárske využitie: práčka, mixér, odšťavovač, ventilátor, brúska, kuchynský robot, vysávač,
- ručné elektrické náradie, otváranie garážovej brány, vodné čerpadlo, kompresor, stolové obrábacie stroje,
- osobná starostlivosť (masážne strojčeky, sušič vlasov),
- kancelárske potreby: skartovač, strúhadlo a pod. [1, 2, 3, 4]

II. NÁVRH MENIČA PRE UNIVERZÁLNY MOTOR

Nakoľko má univerzálny motor lepšie prevádzkové vlastnosti pri napájaní jednosmerným napätím [5], budeme v tejto časti príspevku prezentovať návrh jednosmerného znižovacieho meniča pre napájanie univerzálneho motora. Pre návrh bol použitý univerzálny motor z práčky Indesit s nasledujúcimi parametrami:

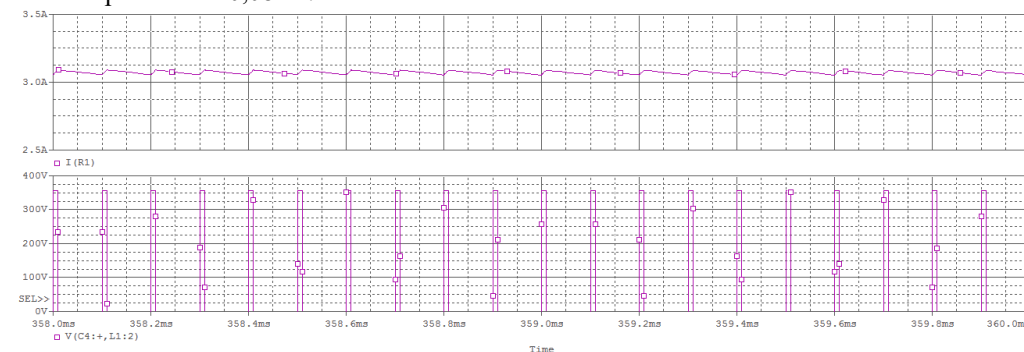
$$U_N=220-240 \text{ V}, I_N=3 \text{ A}, P_N=300 \text{ W}$$

Znižovací menič pre svoju funkčnosť potrebuje jednosmerné napätie na vstupe (najlepšie ideálne vyhladené), keďže sa mení stredná hodnota napätia a nie charakter napätia. Najjednoduchším riešením pre napájanie je použiť neriadený usmerňovač na vstupe, čo predstavuje aj najlepšiu voľbu z hľadiska spätného vplyvu na sieť. Pri neriadenom usmerňovači vnika najmenej vyšších harmonických. Pre obmedzenie nabíjacieho prúdu kondenzátorov bola za usmerňovač vložená tlmivka a kondenzátory v obvode slúžia na vyhladenie napätia. Samotný znižovací menič sa skladá z MOSFET-u a nulovej diódy. Simulačná schéma navrhovaného znižovacieho jednosmerného meniča je zobrazená na obr. 2.

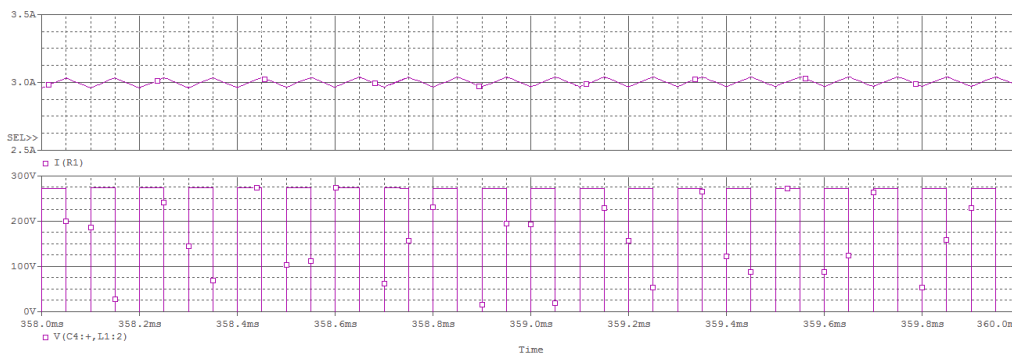


Obr. 2 Simulačná schéma znižovacieho meniča

Pre simuláciu bola zvolená spínacia frekvencia 10 kHz. Výsledné priebehy zo simulácie sú zobrazené na obr. 3 a 4. Ako vidno z priebehov pri záťažnom prúde 3 A a 10 %-nej striede bolo zvlnenie prúdu len $\pm 0,08 \text{ A}$.



Obr. 3 Priebeh napätia a prúdu záťaže pri striede $z=0,1$

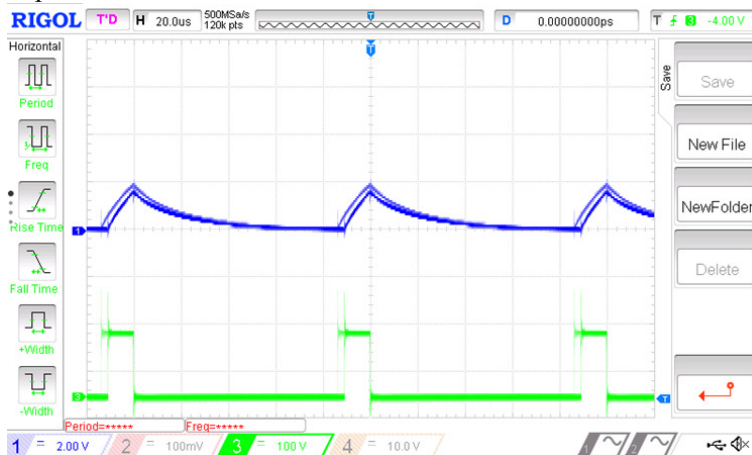


Obr. 4 Priebeh napätia a prúdu záťaže pri striede $z=0,5$

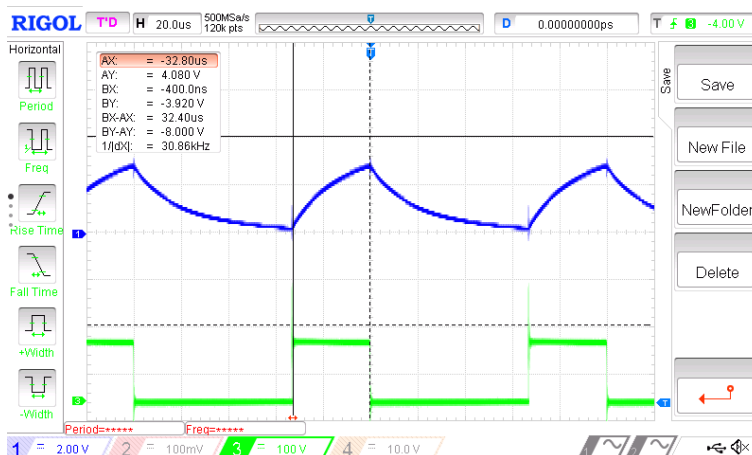
III. PRAKTICKÁ VERIFIKÁCIA JEDNOSMERNÉHO ZNIŽOVACIEHO MENIČA

Navrhnutý a simulačne overený jednosmerný znižovací menič bol následne aj prakticky overený. Samotný jednosmerný menič bol zapojený podľa simulačnej schémy zobrazenej na obr.2. Na generovanie impulzov spínacej frekvencie 10 kHz bolo použité Arduino. IRFP460 potrebuje na zopnutie napätie U_{GS} minimálne 10 V. Nakoľko má Arduino na výstupe iba 5 V a pre zopnutie použitého MOSFET-u je potrebných 10 V, boli použité budič IR2113 a kvôli ochrane Arduina bol použitý optočlen HCPL2232. Výstupná strieda sa riadila pomocou zapojeného potenciometra. Priebehy jednotlivých prúdov a napätí pri testovaní jednosmerného meniča na RL záťaži sú zobrazené na nasledujúcich obrázkoch.

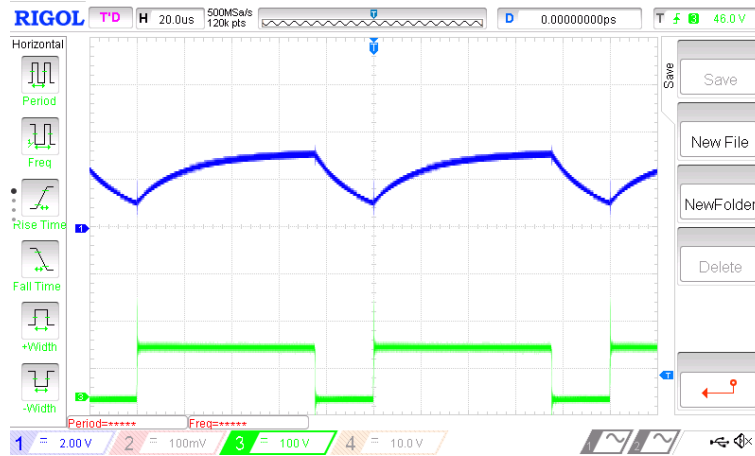
Podľa priebehov získaných z osciloskopu už pri 32% striede bol priebeh prúdu na hranici prerušovaných prúdov.



Obr. 5 Priebeh napätia na záťaži (zelený) a prúdu na záťaži (modrý) na pri striede $z=0,1$



Obr. 6 Priebeh napätia na záťaži (zelený) a prúdu na záťaži (modrý) na pri striede $z=0,32$

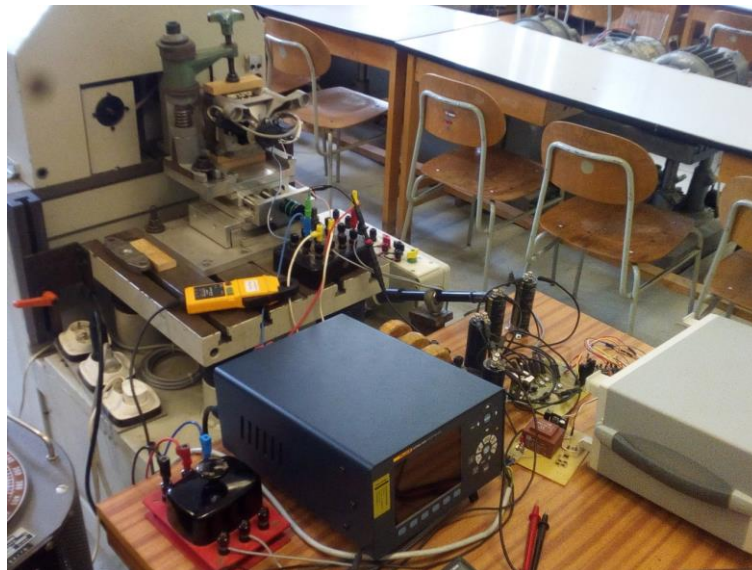


Obr. 7 Priebeh napätia na záťaži (zelený) a prúdu na záťaži (modrý) na pri striede $z=0,74$

Následne bol menič overený aj priamo na univerzálnom motore. Hodnoty dosiahnuté pri meraní na univerzálnom motore napájanom jednosmerným znižovacím meničom sú uvedené v tabuľke 1.

Tabuľka 1
 Namerané hodnoty pri napájaní univerzálneho motora jednosmerným znižovacím meničom

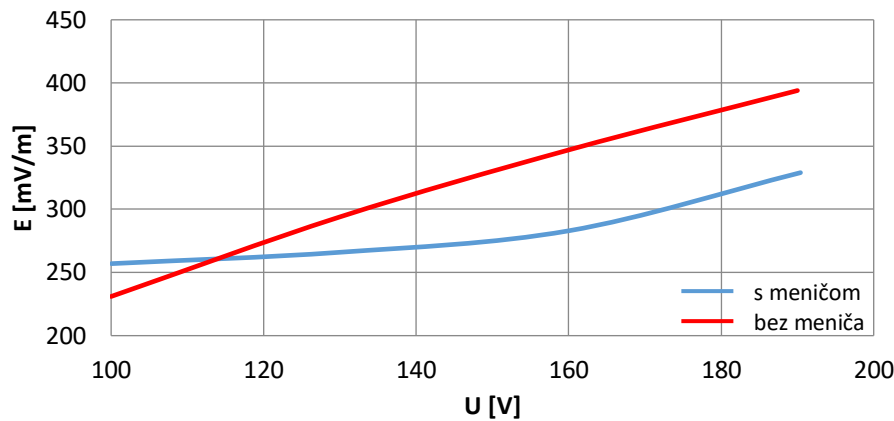
U [V]	I [A]	Pp [W]	n [ot/min]	E [mV/m]	P [W]	η [%]
100,1	3,1	202,83	2450	257	53,8	26,52
130,0	3,0	356,08	4920	266	109,7	30,80
160,0	3,0	480,87	7120	283	154,3	32,08
190,4	3,0	573,66	8650	329	187,3	32,65



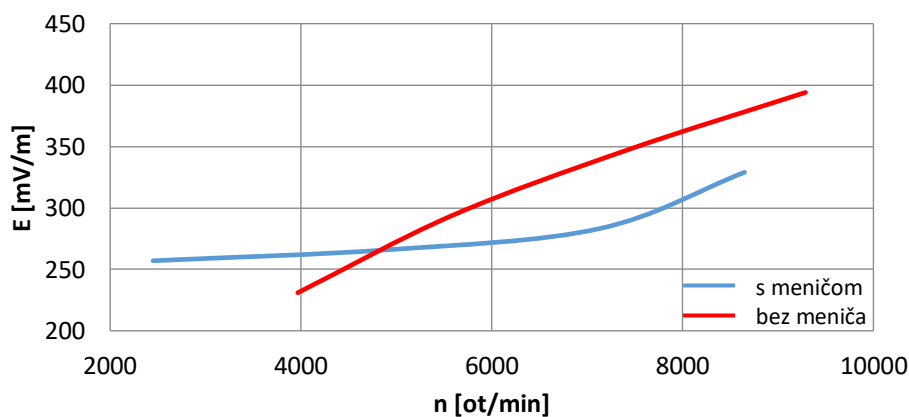
Obr. 8 Meracie pracovisko pre univerzálny motor

Ako možno vidno z priebehov na obr. 9 a 10, použitím navrhnutého jednosmerného znižovacieho meniča sa podarilo dosiahnuť nižšie elektromagnetické rušenie pri rovnakej účinnosti motora.

Červenou je zobrazený priebeh elektromagnetického rušenia pri napájaní univerzálneho motora bez meniča a modrou je zobrazený priebeh elektromagnetického rušenia pri napájaní z jednosmerného znižovacieho meniča.



Obr. 9 Porovnanie elektromagnetického rušenia pri napájaní univerzálneho motora s jednosmerným meničom a bez neho pri zvyšovaní napätia



Obr. 10 Porovnanie elektromagnetického rušenia pri napájaní univerzálneho motora s jednosmerným meničom a bez neho pri rastúcich otáčkach

IV. ZÁVER

V tomto článku sme chceli prezentovať návrh jednosmerného znižovacieho meniča pre napájanie univerzálneho motora. Návrh jednosmerného znižovacieho meniča bol robený pre univerzálny motor, ktorý bol pôvodne použitý v pračke značky Indesit a po simulačnom overení bol verifikovaný aj reálny jednosmerný znižovací menič, čím sa overila jeho funkčnosť a vlastnosti. Z výsledkov merania na tomto jednosmernom znižovacom meniči vyplýva, že pri jeho použití pre napájanie univerzálneho motora vykazuje motor nižšie hodnoty elektromagnetického rušenia, ako pri napájaní bez tohto meniča.

POĎAKOVANIE

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-15-0750.

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-16-0206.

Táto práca vznikla s podporou grantu VEGA 1/0187/18.

POUŽITÁ LITERATÚRA

- [1] J. Kaňuch, P. Vinyi: „DC drive for universal motor“, Maszyny elektryczne : Zeszyty problemowe. No. 84 pp. 7-11, 2009.
- [2] J. Timko, J. Žilková, J. Dudrik, P. Girovský: Elektrické akčné členy a pohony. Technická univerzita v Košiciach, 2009. ISBN 978-80-553-0168-6
- [3] V. Hrabovcová, P. Rafajdus, M. Franko: Measuring and modeling of the electrical machines; University of Žilina press, 2004, Slovakia.
- [4] E. Klug, M. Duč-Anci, B. Bachratý: „Contribution to the design of an AC. universal motor“. XII. International Symposium on Electric Machinery in Prague, ISEM 2004.
- [5] P. Girovský, J. Kaňuch, Z. Gombos: „Analýza vplyvu napájania univerzálneho motora“, Elektrotechnické listy [Online], ročník. III, číslo 3, 2018, dostupné: http://elektrotechnickeлисты.eu/casopis/rocnik_III/clanky/EL_08_2018.pdf

Komplexný systém pre rezerváciu videokonferenčných miestností na báze QR kódov

¹Martin ČONTOFALSKÝ, ²Dávid CYMBALÁK

¹Katedra počítačov a informatiky, Fakulta Elektrotechniky a Informatiky, Technická univerzita v Košiciach, Slovenská republika

¹martin.contof@tuke.sk, ²david.cymbalak@cni.sk

Abstrakt – Táto práca sa zaoberá návrhom a implementáciou rezervačného systému v prostredí rozsiahlych video-konferenčných infraštruktúr. Navrhnutý systémový model je založený na analýze súčasných rezervačných systémov a doplnený o funkcionality viacúrovňovej rezervácie QR kódmi a prepojením na zobrazenie vstavaného živého vysielania. Systém bol experimentálne implementovaný v simulovanom prostredí, kde bola simulovaná komplexná komunikácia medzi TMS systémom a rezervačným portálom v multiplatformovom prostredí.

Kľúčové slová – Angular, Cisco TMS, Ionic, Hybridný vývoj, QR kód, Soap

I. ÚVOD

V súčasnej situácii sa zaznamenáva zvýšený nárast využitia video-konferenčných technológií na globálnej, ale aj národnej úrovni. Vznikajú rozsiahle video-konferenčné infraštruktúry, ktoré je žiaduce efektívne spravovať, prehľadne plánovať vznikajúce video-konferenčné spojenia a jednoduchým spôsobom doručiť účastníkom informáciu o pripojení sa do stretnutí. Jednou z takýchto infraštruktúr je NTI tvoriaca východiskový bod pre vznik tejto práce.

Úlohou je teda nadviazať na rezervačný systém NTI, vypracovať návrh a implementáciu alternatívneho multiplatformového rezervačného systému pre manažment video-konferencií s väzbou na zobrazenie živého vysielania a rezerváciu prístupu s pomocou fyzických prvkov nachádzajúcich sa v aproximácii ku konferencii.

II. REZERVAČNÉ SYSTÉMY VO VIDEO-KONFERENČNOM PROSTREDÍ

Termínom rezervačný systém pre video-konferencie sa popisuje riešenie prepájajúce softvér, používateľov a fyzické miesta. Používatelia ho využívajú na spojenie sa s inými používateľmi alebo na rezervovanie si miesta v nejakom časovom rámci. Prepojenie a komunikácia sú obvykle súčasťou softvérového riešenia. Tie sú zväčša poskytované prostredníctvom zvukového a obrazového prenosu v reálnom čase, no môžu byť zabezpečené aj zorganizovaním osobného stretnutia používateľov. Je na riešení aby verifikovalo časové rámce, aby nedochádzalo ku časovým kolíziám zo strany používateľov alebo miestností. Má poskytovať informácie o aktuálnych stretnutiach, rezerváciách a nech je možné ich plánovať. Stretnutia môžu byť rôzneho typu a to napríklad verejné alebo súkromné.

Preto rezervačné systémy aplikujú aj identifikáciu používateľov aby zamedzili prístup neoprávneným osobám, ktoré ho nemajú mať a naopak. Systémy obsahujúce citlivé osobné informácie o používateľoch, musia byť pripravené čeliť hrozbám s tým spätými. Napríklad krádež identity používateľa alebo osobných údajov. Zmena rezervácii nepovolenými spôsobmi a používateľmi.

Keďže ich úlohou je aj samotné sprostredkovanie informácii pre používateľa, tak do úvahy pri ich tvorbe je nutné komponovať techniky dizajnu pre grafické užívateľské rozhranie. Aby riešenie bolo intuitívne, malo rýchly a jednoduchý systém prístupu k požadovaným informáciám a akciám.

A. Východiskové riešenia, rezervačné systémy

Súčasný systém na rezerváciu využívaný v Centrum vedecko-technických informácií (CVTI) SR poskytuje vizualizáciu pomocou kalendára. Kalendár má 3 formy a to mesačný, týždenný a detailný rozvrh jedného dňa. Pričom je možné zobrazovať kategórie ako všetky, iba verejné a taktiež tie, ktoré vytvoril iba daný prihlásený používateľ. Obrazovka je rozdelená na 2 časti pričom kalendár zaberá ľavú stranu. Na pravej strane je možné po zvolení dňa alebo konferencie vidieť detailný popis danej jednotky. Jednotka môže byť celý deň alebo samotná konferencia. Pre funkcie editovania vlastných privátnych stretnutí je nutné sa prihlásiť. Pri vybraní konferencie sa zobrazia jej detaily ak má daná konferencia byť vysielaná tak sa zobrazí aj video s jej živým prenosom.

B. Využitie QR kódov pre uľahčenie prístupu

Základom rezervačných systémov je fyzické miesto, ako kancelária, stretávacia miestnosť, hala a mnohé ďalšie. Pre zvýšenie efektivity, použiteľnosti a zjednodušenie prístupu sú zaradené do rezervačných systémov aj fyzické objekty. V prípade ak sa objekt využíva na samotné rezervovanie konferenčných miestností mal by o nich poskytovať potrebné informácie a mal by byť na dostupnom mieste. Medzi vhodné technológie, ktoré všetky tieto požiadavky spĺňajú a sú používané patria napríklad QR kódy a NFC tagy. Menej vhodnou technológiou je bluetooth beacon a to hlavne pre vyššiu cenu, nutnosť meniť batérie a menšiu presnosť odkazovania na špecifickú miestnosť. Sekundárnou výhodou objektov reprezentujúcich fyzické lokácie je nielen rezervácia, ale aj možnosť zdieľania konania samotnej verejnej konferencie.

Quick Response (QR) kód dvojrozmerná značka vytvorená spoločnosťou Denso, ktorá bola schválená pod ISO s označením IEC18004 [1]. Bola zamýšľaná iba pre označovanie riadenia produkcie v automobilovom priemysle. Tento štandard sa rozšíril a používa sa hlavne pre niekoľko vylepšených vlastností ako napríklad zvýšená kapacita oproti bežným čiarovým kódom a to až sto násobne vyššia hustota zápisu dát. Maximálna kapacita je 7 089 číselných znakov pričom má lepšiu výkonnosť pri dekódovaní na rovnaký počet zakódovaných dát [1]. Na čítanie QR kódu je potrebná obyčajná kamera akú obsahuje väčšina moderných mobilných telefónov. Je možné ho používať zadarmo na akúkoľvek aplikáciu keďže patent bol vydaný do verejnej domény.

Vlastnosti separujúce ho od ostatných spôsobov záznamu dát na dvojrozmerné značky sú napríklad možnosť čítania z každého smeru, čiže vodorovné natočenie je zanedbateľné. Odolnosť voči deformácii, keďže vzory sú usporiadané do pravidelných intervalov a vypočítava sa mapovanie pre každú bunku a to je napravené. Teda ďalšou vlastnosťou je oprava chýbajúcich a nečitateľných dát. QR kód môže byť definovaný viacerými symbolmi. Možnosť jednoduchého šifrovania na základe použitia špeciálnej prevodovej tabuľky.

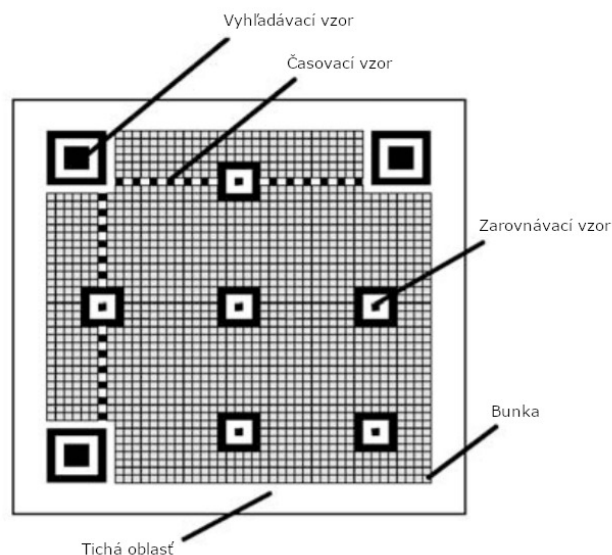


Fig. 1 Vizualizácia štruktúry QR kódu [1]

QR kód je možno rozdeliť do piatich hlavných častí a to bunka, tichá oblasť, vyhľadávací, časovací a zarovňavací vzor. Na obrázku 1 je zobrazené ich rozmiestnenie. Vyhľadávací vzor pozostáva z 3 znakov na okraji QR kódu a slúži hlavne k určeniu pozície, veľkosti a natočenia.

Zarovnávací vzor sa skladá z bodov obkolesených čiernym okrajom a používa sa k oprave deformácií. Časovací vzor sa skladá zo striedavo usporiadaných čierny a prázdnych bodov. Taktiež sa používa k náprave skreslených dátových buniek. Tichá zóna je jednoducho oblasť potrebná pre lepšie čítanie.

III. HYBRIDNÝ VÝVOJ

Pre vývoj rezervačného systému pracujúceho v rovnakom čase na rôznych prostrediach je nutné dbať práve na kompatibilitu a zhodnosť poskytovanej funkcionality. To zahŕňa napríklad prístup k zdrojom zariadenia potrebným ku vykonávaniu svojej činnosti. Preto treba vybrať technológiu, ktorú majú všetky tieto prostredia zhodnú. Vo väčšine prípadov je možné použiť webové technológie.

Porovnanie	Natívne	Hybridné
Výkon	Vysoký. Priami prístup k platformovej funkcionalite.	Nižší. Nemá/nemusí mať priami prístup k funkcionalite platformy.
Znovupoužitie kódu	Obmedzené. Pre každú platformu sa rieši separátne.	Neobmedzené. Pre platformy podporujúce hybridný developoment sa používa jeden kód.
Konzistencia používateľského prostredia	Rozdielne. Originálny výzor používateľského rozhrania platformy.	Rovnaké. Vyzerá rovnako na všetkých platformách.
Cena	Vyššia. Pri tvorbe systému pre každú platformu.	Nižšia. Jedno riešenie pokrýva všetky platformy.

Table 1
Porovnanie natívneho a hybridného prístupu k vývoju systému [2].

A. Programovací rámec IONIC

”Je postavený na Angular frameworku. Celkovým základným kameňom je programovací jazyk JavaScript s používaním typov, inak nazývaný TypeScript. Angular bol vytvorený na umožnenie tvorby reaktívnych jedno-stranových aplikácií. To znamená, že aplikácia sa spúšťa z jedného HTML súboru, ktorý beží v samotnom prehliadači. Táto stránka je menená za behu, pomocou DOM (dokument objektový model)” [3]. Toto je omnoho rýchlejší spôsob prehľadávania obsahu ako za každým žiadať server o novú HTML stránku. Tým je aj šetrená šírka-pásma a záťaž servera.

Framework určený k vývoju front-end aplikácií. Na vývoj nepoužíva natívny prístup, ale hybridný, čiže kód je nutné napísať iba raz a následne pracuje na akejkoľvek platforme podporujúcej webové prehliadanie. Tým pádom má výhodu rýchlosti vývoja aplikácie, ktorá dosahuje až dvojnásobnú alebo trojnásobnú rýchlosť oproti natívnemu vývoju zameranému na všetky platformy. Hybridné aplikácie sa odlišujú od natívnych hlavne spôsobom využitia technológií pri ich tvorbe. Sú použité najmä webové technológie a jazyky ako JavaScript, HTML, a CSS narozdiel od proprietárnych alebo špecializovaných jazykov použitých najmä v iOS a Android platformách. Samotné aplikácie sú spustené ako celo-obrazové prehliadače nazývané tiež aj Webview. Poskytuje konzistentný výzor aplikácie aj keď používateľ zmení jeho aktuálnu platformu. Na prístup k natívnym funkciám ako napríklad kamera alebo NFC sa používajú pluginy. Jednoducho sa spravuje a udržiava. V neposlednom rade poskytuje jednoduchý prechod na budúce zariadenia, keďže webové prehliadanie je štandard tejto doby. Nevýhodami sú hlavne závislosti na pluginoch a určitá nadstavba nad hardvérovými prvkami, ktorá znižuje výkon napríklad v aplikáciách používajúcich 3D vykresľovanie. Ostatné porovnania sú zobrazené v tabuľke 1.

IV. SOAP KOMUNIKÁCIA A CISCO TMS

Cisco TMS je centralizovaný systém, ktorý je určený na monitorovanie a manažovanie videokonferenčnej siete. Má podobu štrukturovaného rozhrania umožňujúceho ovládanie miestnych, ale aj vzdialených video systémov. Súčasťou je plánovací systém pre tieto video siete. Automatizuje konfiguráciu pre základné videokonferenčné siete. Je ale možné zladiť toto správanie, tak aby zodpovedalo požiadavkám pre vlastné riešenia.

Cisco ponúka svoj produkt TMS aj ako skúšobnú verziu, pri ktorej používateľ má prístup len k obmedzenému počtu funkcionalít. Žiaľ API TMS neposkytuje väčšinu svojich funkcií v prípade skúšobnej verzie, to znamená, že odpovedou na väčšinu typov dotazov je chybová odpoveď obsahujúca chybový kód -2147219503 znamenajúci správa konferencie bez potrebnej licencie. No na druhú stranu je možné extrahovať WSDL súbor obsahujúci všetky tvary požiadaviek a odpovedí na SOAP komunikáciu. Tento WSDL súbor je ideálny pre vytvorenie servisu, ktorý bude predstierať, že je TMS API. V rámci tohto servisu je možné načúvať na požiadavky. Je možné čítať samotné atribúty a na ich základe sa rozhodnúť aké budú odoslané odpovede.

Za týmto účelom bol využitý práve systém SoapUI poskytujúci práve možnosť vytvorenia servisu na predstieranie pomocou WSDL súboru. Prvým krokom je extrakcia WSDL jednoduchým sa odkázaním na reálny koncový-bod TMS API BookingService. Potom pomocou extrahovaného súboru je možné založiť projekt v rámci SoapUI. Následne je daný projekt použitý prostredníctvom ponuky projektu k vytvoreniu servisu na predstieranie. Takýto servis obsahuje iba prázdnu štruktúru požiadaviek a možných prázdnych odpovedí.

Preto je nutné vložiť rozhodovacie postupy na základe vstupov a prvotného nastavenia. Pre funkčnosť rezervačnej aplikácie je nutné takto upraviť aspoň základné funkcionality ako uloženie, načítanie a vymazanie konferencie. Keďže zložité funkcie ako napríklad "obdržanie konferencie pomocou id, ktorá má najskorší prebiehajúci alebo nevykonávaný čas" by si vyžadovali vytvorenie samotného systému ako celku. No v rámci základných funkcií je treba dodržiavať dokumentáciu TMS API aby pri reálnom nasadení tohto servisu nevznikali početné nezrovnalosti.

V. MODELOVANIE A RIEŠENIE PRE REZERVAČNÝ SYSTÉM

Prvým krokom pri vytváraní akýchkoľvek systémov odhliadnuc od špecifického riešenia je navrhnutie modelov. Modelovanie systémov slúži k vytvoreniu koncepčných návrhov, na ktorých základe sa konštruuje reálne riešenie. Existuje mnoho spôsobov modelovania ako funkčné, architektonické alebo modelovanie biznis procesov. Ako spôsob modelovania sa dá považovať aj vytváranie prototypov. Benefitom je spätná väzba od používateľa či zadávateľa a na jej základe jednoduchá úprava v pred-vývojovej fáze. Prototypy a počiatkové modely sa počas vývoja upravujú až kým sa nedospeje k koncovému modelu a finálnemu riešeniu. Teda ich prostredníctvom je dodržiavaný správny smer vývoja.

Aplikácia je založená na Ionic štruktúre. Čiže ako aj v prípade iných projektov rámca Angular je rozdelená do samostatných komponentov, modulov a servisov. Tieto časti sú vytvorené so snahou byť čo najviac nezávislé od ostatných častí tvoriacich aplikáciu a to z dôvodu uľahčenia modifikovania a budúcej údržby. Komponenty určené k zobrazovaniu využívajú šablóny v jazyku HTML spolu s SCSS štýlami. Objekty bez týchto šablón sú buď servismi alebo poskytovateľmi. Prostredníctvom UML relačného diagramu 2 sú reprezentované vzťahy jednotlivých komponentov. Zároveň je zobrazená hranica aplikácie voči externým entitám.

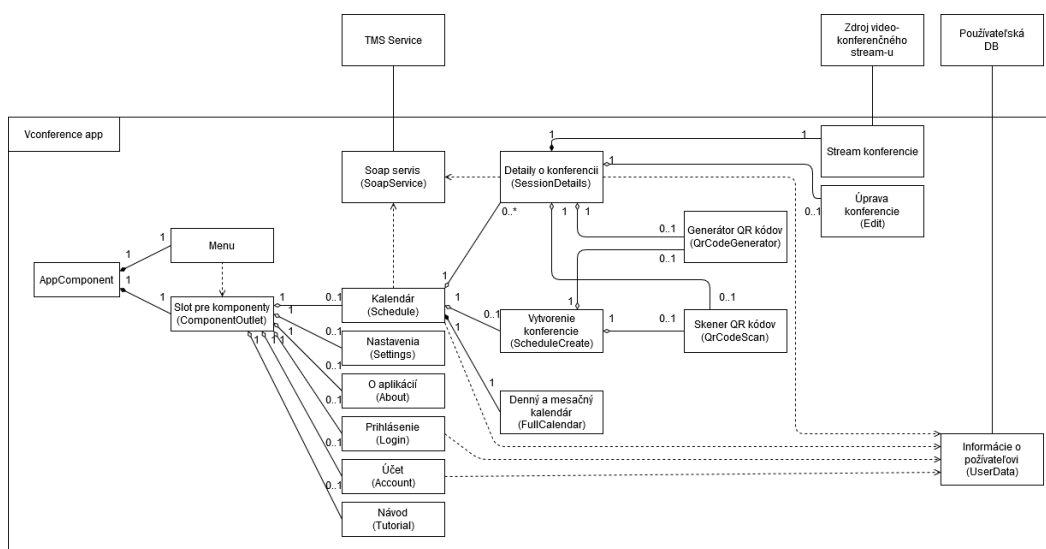


Fig. 2 UML relačný diagram.

Okrem toho, že táto aplikácia je postavená na Angular rámci tak využíva rozšírenie Ionic. Toto rozšírenie poskytuje aj možnosť využitia Cordova rámca. Keďže jednou z hlavných úloh bolo vytvorenie hybridnej aplikácie a práve Cordova poskytuje túto funkcionality. Špecifickejšie umožňuje využitie HTML, CSS a JavaScript pre natívne aplikácie v rôznych mobilných platformách. Pričom vykresľuje webovú aplikáciu v rámci natívnych WebView zobrazení.

VI. ZÁVER

Vďaka tejto práci vznikol alternatívny systém pre rezervovanie video-konferenčných miestností s využitím fyzických rozšírení. Vďaka hybridnému riešeniu je dizajn používateľského rozhrania jednotný a zároveň dynamický na základe možností platformy používateľa. Fyzické prvky sú reprezentované prostredníctvom QR kódov uľahčujúcich vyhľadanie registrovanej konferencie alebo uľahčenie vytvárania konferencie.

Používateľské rozhranie je rozdelené na dve časti, ktorými sú menu a bloková časť zobrazujúca obsah podľa výberu používateľa. Hlavným zobrazením v bloku je kalendár obsahujúci konferencie. Vďaka Soap servisu sú kalendár a ostatné komponenty dynamicky upravované podľa back-end API. Pri zobrazení detailov jednotlivých konferencií je možné sledovať živý video-prenos vo vstavanom prehrávači. Požiadavky o autorizáciu a práva na zobrazovanie sú smerované na proprietárny Node.js servis pripojený k databáze používateľov.

Kvôli licenčným obmedzeniam Cisco TMS bol vytvorený Soap servis simulujúci jeho základné správanie. Použitý bol WSDL súbor extrahovaný z TMS a teda tvary požiadaviek a odpovedí v simulovanom servise sú zhodné s reálnym riešením. Taktiež bol vytvorený databázový servis pre správu používateľov, ktorý načúva na POST a GET požiadavky. Tie sú pred preposlaním na vstavanú MySQL databázu upravené do vhodného tvaru.

Obmedzením k využitiu tohoto systému v reálnych podmienkach je nutnosť otestovania komunikácie s nesimulovaným Cisco TMS. Do budúcnosti by bolo vhodné rozšíriť tento systém, tak aby sa dalo využívať netriviálne metódy reálneho TMS API.

REFERENCES

- [1] T. J. Soon, "Qr code," *Synthesis Journal*, vol. 2008, pp. 59–78, 2008.
- [2] S. Bosnic, I. Papp, and S. Novak, "The development of hybrid mobile applications with apache cordova," in *2016 24th Telecommunications Forum (TELFOR)*. IEEE, Nov. 2016. [Online]. Available: <https://doi.org/10.1109/telfor.2016.7818919>
- [3] M. Schwarz Müller, *Angular 8 - The Complete Guide (2019+ Edition)*, Udemy, 11 2019.

Short-circuit currents and their calculation

Jakub Urbanský¹, Daniel Pál², Vladimír Krištof³

^{1,2} Department of Electric Power Engineering, FEI TU of Košice, Slovak Republic,
Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Slovak Republic

³ Gridman s.r.o., Košice, Slovak Republic

¹jakub.urbansky@tuke.sk, ²daniel.pal@tuke.sk, ³vladimir.kristof@gridman.sk

Abstract — There are a number of reasons why it is important to analyze short-circuit conditions in electrical networks. One of the most important aspect is the safety and reliability of operation. The Most interesting are the minimum and maximum values of short circuit currents. Many of software tools are used to calculate these values. This article deals with basic description of short-circuit currents, reasons leading to their calculation, comparing and verifying the results of short-circuit currents calculation in the various software tools.

Keywords — Short-circuit current, short-circuit calculation, deviation.

I. INTRODUCTION

The short circuit is one of the major incidents affecting electrical systems. The consequences are often serious, if not dramatic. It is necessary to know the short-circuit conditions in operated power system. It is important for a safe and reliable control and operation of the power system. Short-circuit currents at individual points of the power grid must be known because of several reasons, such as:

- design of electrical equipment from point of view of thermal and dynamic effects of short-circuit currents,
- control of circuit breakers switching capability,
- design of grounding systems and the associated determination of allowable touch-voltage, step-voltage and transferred voltage (for example in low-voltage grid during earth-fault in high-voltage grid),
- design and operation of protection devices,
- stability control of parallel operated synchronous machines in power grid,
- control of voltage ratios during short-circuit and during the start-up of large asynchronous motors,
- calculation of induced voltage in telecommunication lines caused by high-voltage or extra high-voltage networks,
- control of propagation and impact of upper harmonics in power system,
- assessment of overvoltage occurrence during line-to-ground faults.

II. DESCRIPTION OF THE SHORT-CIRCUIT CURRENT

The short-circuit is defined as accidental or intentional conductive path between two or more conductive parts (e.g. three-phase short circuit) forcing the electric potential differences between these conductive parts to be equal or close to zero [1].

Course of short-circuit current is considerably complicated and depends on many circumstances such as R/X ratio of circuit, moment of short-circuit occurrence, angle between voltage and current before short-circuit and so on.

Possible course of short-circuit current is shown on Fig.1 and consists of two parts:

- symmetrical short-circuit currents (its amplitude is during the short-circuit constant – blue course),
- dc component of short-circuit currents (its amplitude is exponentially decreasing in time during short-circuit currents – red course).

Sum of both parts results in asymmetrical short-circuit current flowing in point of fault (yellow course).

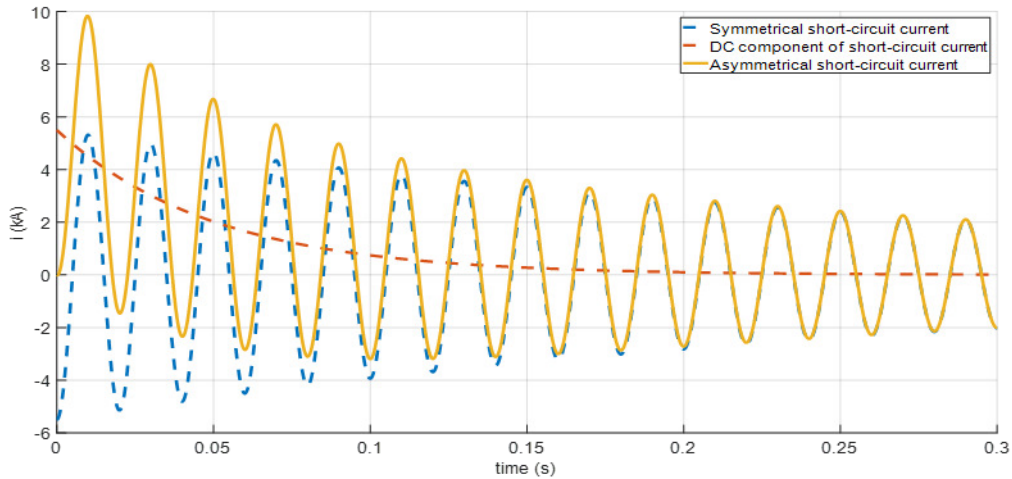


Fig. 1 Short-circuit current flowing in fault point (F)

From the course of short-circuit currents were derived characteristic values of short-circuit currents which serve as basic criteria for short-circuit current assessment. These are:

- initial symmetrical short-circuit current (I_k''): it is RMS value of the alternating current (AC) symmetrical component of a prospective (available) short-circuit current applicable at the instant of short circuit if the impedance remains at zero-time value (see Fig.2),
- peak short-circuit current (i_p): is maximum possible instantaneous value of the prospective short-circuit current (see Fig.2)).
- equivalent thermal short-circuit current (I_{th}): the RMS value of a current having the same thermal effects as the actual short-circuit current (see Fig.3)

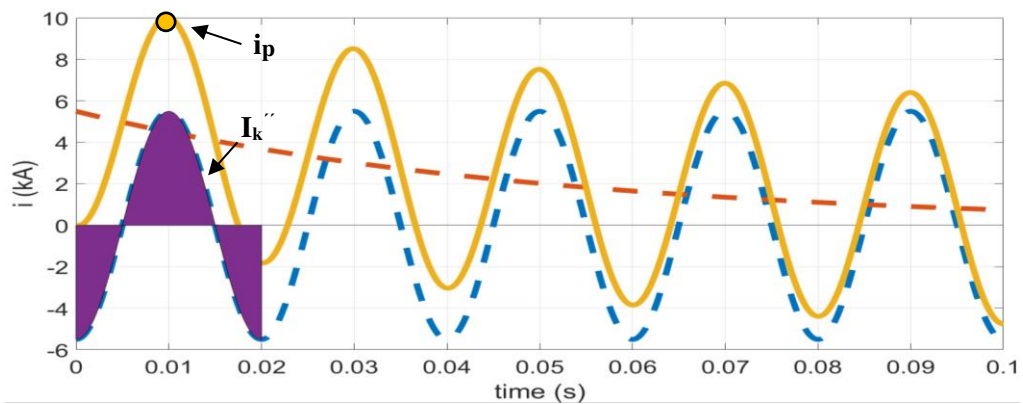


Fig. 2 Characteristic values of short-circuit current

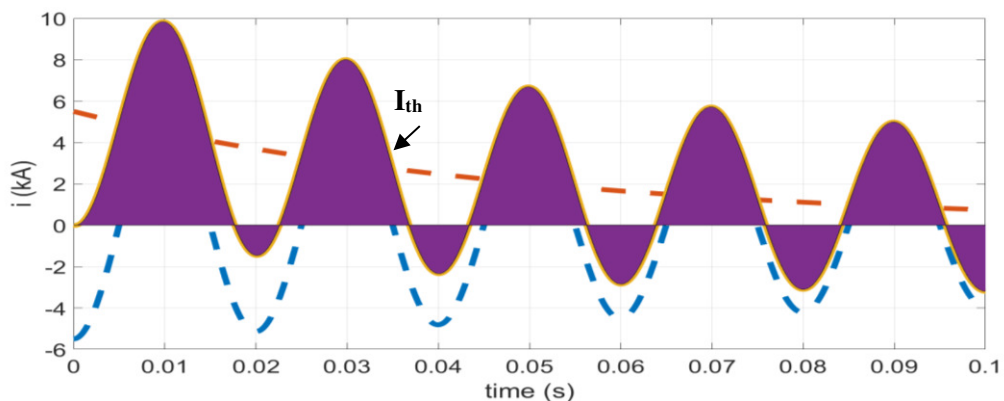


Fig. 3 Equivalent thermal short circuit current

III. CALCULATION OF SHORT-CIRCUIT CURRENTS

In the past, various numerical and numerical-graphic procedures were used to calculate short-circuit currents. From the group of numerical-graphic methods is possible to mention: calculation according to short-circuit curves and using nomograms. Short-circuit curves can be used if the task of finding short-circuit conditions is limited to finding the short-circuit current at the short-circuit location. This method has gained popularity with its simplicity and relative accuracy. The method consisted in the application of special curves which give the value of the alternating component of the short-circuit current as a function of the so-called computational reactance of the diagram for any moment of the short-circuit event. Short-circuit curves were processed in ČSN 38 04 11 standard. The nomogram method is a simple graphical method that consists of subtracting the necessary parameters of the electrical system elements (impedances) from the graphs that were pre-printed on the sheets, separately for each voltage level. The disadvantage of nomograms is their limited use only for beam networks and the fact that accuracy is directly dependent on the accuracy of reading from the graphs drawn.

From the numerical methods it is possible to distinguish: the method of superposition and the method using the equivalent voltage source at the short-circuit point. The superposition method gives a short-circuit current only in relation to one assumed network load condition. Therefore, it does not necessarily lead to the maximum short-circuit current. To overcome this problem and to find the case with the worst load conditions that lead to the maximum short-circuit current at the short-circuit point, a method for changing operating conditions - an equivalent voltage source method - has been developed. The principle of this method is to introduce an equivalent voltage source at the short-circuit point. This short-circuit voltage source is the only active system voltage [2].

Brief characteristics of the methods used:

- ohm method, the disadvantage of this method is its cumbersome if the system under investigation contains several voltage levels [1],
- per united method, is no better in terms of manual calculations than the previous method, since it involves a number of relationships and bonds associated with reference values, which can often cause calculation errors [1],
- the superposition method, this method is very often used but requires knowledge of steady state ratios before the short circuit occurs, which reduces the slag's applicability for general and rapid calculations [3],
- Extended Active Two-Port Method [4].
- The MVA method: this method is quite simple, fast, accurate and the procedure is easy to remember [5].

The solved power system (transmission and distribution system) is usually very robust and complex system so the manual calculation would be very time-consuming and computationally intensive. Therefore, for this reason a lot of software tools were developed.

It is necessary to say that fault calculation is not simple for a number of reasons:

- there are many different types of fault in three phase systems,
- the impedance characteristics of all electrical items in the system must be known,
- the fault impedance itself may be non-zero and difficult to estimate,
- there may be substantial fault current contribution from rotating machines etc.,
- the initial cycles of fault current may be asymmetric with substantial DC offset,
- the earth impedance in earth faults can be difficult to estimate accurately,
- DC system faults also include inductance effects in fault current growth.

Each of software tool has its own calculation accuracy given by calculation method, modeling of power system elements and so on. European short-circuit calculation standard IEC 60909 allows only a permissible deviation of the results (maximum $\pm 5\%$).

The aim of this paper is to compare the results of calculating short-circuit current across the selected programs used in practice and their comparison with the manual calculation of a simple network from literature [2]. There were selected following tools for short-circuit calculation:

- **GLF** (Graphical Load Flow): is simple and overview program. GLF uses Fortescue method of symmetrical components to short-circuit calculation. It is intended to solution mainly high voltage, very high voltage, ultra-high voltage networks. It can be used for steady state calculation, analysis of voltage conditions, analysis of short-circuit conditions checking the reliability of network operations by the criterion (n-1).
- **Pas Daisy**: is package of programs used in preparation of operations, planning for further development, design, evaluation and operation of networks. It is characterized

- by enhanced supply of calculation method.
- **PSLF** (Positive Sequence Load Flow): is a suite of programs for the analysis of (mainly) transmission system. It allows the calculation of the steady state, as well as transitional phenomena (short circuit, earth-faults or dynamic stability).
- **NEPLAN**: is user-friendly program, serving on the planning and calculation of electric, gas or water supply networks.
- **MODES**: is very complex network simulator with very precisely developed calculation methods and extensive library of dynamic models of power system elements. It can be used for wide range of tasks across power system (load flow, stability of power system, motor-start up, operational planning, protection system settings and so on). MODES contains module ZKRATY, which serve for short-circuit currents calculation.
- **DigSilent PowerFactory**: is a power system analysis software application for use in analyzing generation, transmission, distribution and industrial power systems.

A. Example of manual calculation

Considering the following case – see Fig 4 (taken from [2]). Power plant supplies the power system and 3-phase fault occurs in point F. The parameters of the system are following:
 Generator: $S_{rG} = 400 \text{ MVA}$, $U_{rG} = 21 \text{ kV}$, $\cos\varphi = 0.8$, $x''_d = 0,25$, $p_G = 0,05$
 Transformer: $S_{rT} = 400 \text{ MVA}$, $U_{THV} / U_{TLV} = 230 / 21 \text{ kV}$, $u_k = 15\%$
 System: $U_{qmin} = 230 \text{ kV}$, $U_{nq} = 220 \text{ kV}$, $c = 1,1$

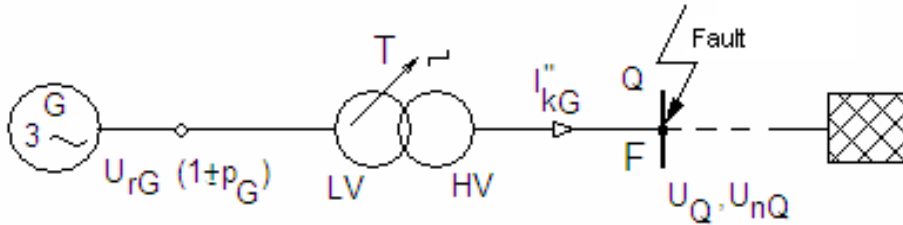


Fig. 4 Short-circuit contribution from power plant to point of fault (F).

The corrective factors according to STN IEC 60909 are considered in calculation [1].

For the short-circuit impedance calculation in electrical power plant block with tap-changer the following relationship is used:

$$\mathbf{Z}_S = K_S (t^2 \cdot \mathbf{Z}_G + \mathbf{Z}_{THV}) \quad (1)$$

Where K_S is corrective factor:

$$K_S = \frac{U_{nQ}^2}{U_{rG}^2} \cdot \frac{U_{TLV}^2}{U_{THV}^2} \cdot \frac{c_{max}}{1 + (x''_d - x_T) \sin \varphi_{rG}} \quad (2)$$

$$K_S = \frac{220 \cdot 230}{21^2} \cdot \frac{21^2}{230} \cdot \frac{1,1}{1 + (0,25 - 0,15) \sin 36,87^\circ}$$

Short-circuit impedance of generator:

$$\mathbf{Z}_G = x''_d \frac{U_{rG}^2}{S_{rG}} = 0,25 \frac{21^2}{400} = 0,276 \Omega \quad (3)$$

Short-circuit impedance of transformer:

$$\mathbf{Z}_{THV} = u_k \frac{U_{THV}^2}{S_{rT}} = 0,15 \frac{230^2}{400} = 19,836 \Omega \quad (4)$$

Short-circuit impedance of considered power system (in point F) is:

$$\begin{aligned} \mathbf{Z}_S &= K_S (t^2 \cdot \mathbf{Z}_G + \mathbf{Z}_{THV}) = \\ &= 0,9926 \left(\left(\frac{230}{21} \right)^2 \cdot 0,276 + 19,836 \right) = 52,552 \Omega \end{aligned}$$

The maximum initial symmetrical three-phase short-circuit current:

$$I_{kG} = \frac{c.U_n}{\sqrt{3}.Z_s} = \frac{1,1.220}{\sqrt{3}.52,552} = 2,659kA \quad (5)$$

B. Comparison of the results

Following example was modeled in selected software tools and results of short-circuit calculation are listed in Table 1. As the reference value the short-circuit current value calculated manually was chosen.

Deviation for each case was calculated by the formula below:

$$D = \frac{I_{kB} - I_{kA}}{I_{kA}} \cdot 100(\%) \quad (6)$$

Table 1
 Comparison of results of short-circuit calculations in various software tools

	I''_{k3} [kA]	Deviation[%]
Manual calculation	2,66	-
GLF	2,65	-0,33
Pass Daisy	2,79	4,9
PSLF	2,71	2,03
NEPLAN	2,72	2,4
MODES	2,69	1,1
DigSilent	2,68	0,8
Powerfactory		

IV. CONCLUSION

Calculation and comparison of the results was focused on maximum three phase bold short-circuit current value because in most cases that value takes into account for dimensioning equipment. A deviation with value to (+/-) 2% can be neglected. The maximum possible deviation allowed by standard is (+/-) 5%. Deviation are mainly due to the following facts, that not all of the software packages consider corrective factors, which were considered in manual calculations (according to STN IEC 60909). STN IEC 60909 standard is conservative, which means that it sets strict conditions for calculating short-circuit current through the correction factors. With these corrective factors is achieved better dimensioning and also setting up of protection relays. That is very important for safety and reliable operation of power system.

ACKNOWLEDGMENT

This work was supported by the Scientific Grant Agency of the Ministry of Education of Slovak Republic and the Slovak Academy of Sciences by the projects VEGA No. 1/0372/18.

REFERENCES

- [1] STN IEC 60909-0: Short-circuit currents in three-phase a.c. systems. Part 0: Calculation of currents. November 2016
- [2] MEŠTER, M., – Výpočet skratových prúdov v trojfázových striedavých sústavách. ABB-elektro, s.r.o., 2005. ISBN 80-89057-10-1
- [3] KAKILLI, A.: System Analysis with the MVA Method for Symmetrical Three-Phase Faults. TEM Journal, vol2 no1, 2013.
- [4] MÁŠLO, K. a kol.: Řízení a stabilita elektrizační soustavy. ČEPS a Asociace energetických manažerů. Praha 2013. ISBN 978-80-260-4461-1
- [5] JAFFARI, H.: Fault Current Calculation. Power System Review: Module 6. Northeast Public Power Association, 2013.
- [6] KRÍŠTOF, V., – Výpočet skratových pomerov podľa STN IEC 60909. Diplomová práca. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2009.
- [7] MITOLO, Massimo: SHORT-CIRCUIT CALCULATION METHODS, 10/2004 Dostupné na internete: <http://ecmweb.com/mag/electric_shortcircuit_calculation_methods/>.

Machine Learning in the Energy Sector

¹Maksym Oliinyk, ²Jaroslav Džmura

¹Department of Electric Power Engineering Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

²Department of Electric Power Engineering Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

¹maksym.oliinyk@tuke.sk, ²jaroslav.dzmura@tuke.sk

Abstract — the so-called artificial intelligence (AI) is increasingly mentioned in the context of the technological transformation taking place at present. According to many experts, the use of artificial intelligence algorithms will affect almost all areas of the economy. In this regard, its impact on the energy sector seems interesting.

Keywords — machine learning, smart grids, renewable energy, forecasting

I. MACHINE LEARNING IN THE ENERGY SECTOR

An analysis of the publicly available information shows that the main areas of application of artificial intelligence in the energy sector at present are [1]:

- Forecasting tasks (meteorological information, equipment operating status, changes in consumption, etc.);
- Optimization tasks (operating modes of power system components, consumption, network configuration, etc.);
- Management tasks (artificial lighting, renewable energy and batteries, asset efficiency, etc.);
- tasks of communication (energy companies with consumers);
- Tasks of developing services and services (in terms of customer satisfaction with the range of services provided by companies, participation of enterprises in the work of energy markets, addressing quality assurance issues).

The expansion of the use of artificial intelligence tools in the energy sector will inevitably occur along with such unfolding processes as [2]:

- Energy transformation due to the increased use of local renewable energy sources, as well as the intellectualization of production, transmission and consumption of energy (smart technology);
- Digital transformation, due to the growing needs of monitoring and data analysis (big data) and the introduction of new technologies (for example, blockchain, "digital substation", unmanned devices for monitoring objects, etc.);
- The unification and mutual influence of various sectors of the energy and transport sectors (for example, Power-to-X technology). In light of the above, there is no doubt that the position of energy, as one of the most interesting areas of application of artificial intelligence methods will be strengthened.

Examples of the use of artificial intelligence algorithms in forecasting problems are enough today [3]. The dependence of renewable energy production on weather conditions has significantly increased the need for accurate forecasting. In the US state of Colorado, energy provider Xcel uses artificial intelligence algorithms to process information received from the National Center for Atmospheric Research (including satellite observation data in wind farm areas). This allows the company to generate detailed reports and optimize the operation of wind farms. IBM, together with the US Department of Energy (US Department of Energy), implements the SunShot initiative, in which a self-learning program can reliably predict the generation of renewable sources (solar, wind and hydro) [4]. The algorithm uses a large amount of historical data along with real-time weather monitoring information. According to analysts, the large-scale use of

artificial intelligence algorithms to improve the operation of U.S. wind parks would theoretically allow them to increase their production in 2017 by 12 billion kWh and increase the share of wind energy in the total balance (6.3% last year). You can find examples of intellectualization tasks. PowerScout received two grants from the US Department of Energy for the development of cost-cutting programs for network companies and consumers (smart home), taking into account the integration of renewable sources [5]. Programs also use artificial intelligence algorithms. London Company Green Running Ltd. Develops a machine-learning-based Verv application designed to optimize the energy consumption of homes. The application works on computers, tablets and smartphones. The German company Schleswig-Holstein Netz AG, which operates electrical networks in the federal state of Schleswig-Holstein, reported an interesting application of artificial intelligence methods. Here, a self-learning network is used to identify the locations of the alleged damage. As the initial data, information is used on the life of the components of the electric networks and the repairs carried out, as well as information on loads and weather conditions. In addition, the American company AirFusion, which uses unmanned aerial vehicles to monitor the status of high-voltage power lines and wind power plants, uses software with artificial intelligence algorithms to process monitoring results. A neural network helps to better solve the problem of pattern recognition, for which, during the training process, thousands of images of damaged wind turbines are loaded into the program (including the consequences of lightning strikes, delamination, coating erosion, etc.) [6][7].

II. REGRESSION MODELS

Regression analysis is a method of modeling measured data and studying their properties. The data consists of pairs of values of the dependent variable (response variable) and the independent variable (explanatory variable). The regression model is a function of an independent variable and parameters with a random variable added. Model parameters are adjusted so that the model best approximates the data. The criterion for the quality of the approximation (objective function) is usually the standard error: the sum of the squares of the difference between the values of the model and the dependent variable for all values of the independent variable as an argument. Regression analysis is used to predict, analyze time series, test hypotheses, and identify hidden relationships in data [8].

The practice of regression analysis suggests that the linear regression equation often adequately expresses the relationship between the indicators even when in fact they turn out to be more complex. This is explained by the fact that, within the limits of the studied quantities, the most complex dependences can be approximately linear. In general form, the linear regression equation has the form [8]:

$$y = a_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots + b_k \cdot x_k + \varepsilon_i \quad (1)$$

Where y - is a productive sign, the investigated variable; x - designation of the factor (independent variable); i is the total number of factors; and a_0 is a constant (free) term of the equation; b - regression coefficient with a factor.

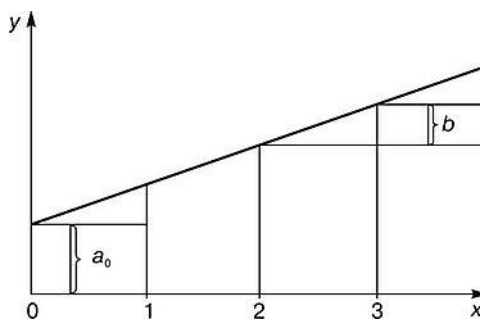


Fig. 1 Graph of a simple pairwise linear regression of $y = a_0 + b \cdot x$: the segment b shows the increment of y with increasing x by one

Phenomena are determined, as a rule, by a large number of cumulatively acting factors. In this regard, the problem often arises of studying the dependence of one variable y on several explanatory variables $x_1, x_2 \dots x$. This problem is solved using multiple regression analysis. The construction of the multiple regression equation begins with the solution of the question about the specification of the model, including the selection of factors and the choice of the type of regression equation. Factors included in multiple regression must meet the following requirements: they must be quantifiable (qualitative factors must be quantified); between factors

there should not be a close correlation, and even less a functional dependence, i.e. there should be no multicollinearity. The inclusion of multicollinear factors in the model can lead to the following consequences: it is difficult to interpret the parameters of multiple regression as characteristics of the action of factors in a “pure form”, since the factors are interconnected; linear regression parameters lose their economic meaning; parameter estimates are unreliable, have large standard errors and vary with the volume of observations [8].

Evaluation of the quality of the regression equation is carried out based on a set of criteria that verify the adequacy of the model to actual conditions and the statistical reliability of the regression. One of the most effective estimates of the adequacy of the model is the coefficient of determination of R^2 . The true coefficient of determination of the model of the dependence of the random variable y on the factors x is determined as follows [9]:

$$R^2 = 1 - \frac{\sigma^2}{\sigma_y^2} \quad (2)$$

Where σ_y^2 is the variance of the random variable y , σ^2 — conditional (in terms of x factors) variance of the dependent variable (variance of model error).

R^2 describes the proportion of variation in the dependent variable due to regression or variability of the explanatory variables. The closer R^2 is to unity, the better the constructed regression model describes the relationship between the explanatory and the dependent variable. In the case of $R^2 = 1$, the studied relationship can be interpreted as functional (rather than statistical), which requires additional qualitative and quantitative information and changes in the research process [9].

III. PREDICTION OF SOLAR AND WIND ENERGY PRODUCTION USING MACHINE LEARNING

To demonstrate the operation of the regression models, three predicted variables were selected:

- The amount of energy that is generated by solar power plants in Germany;
- The amount of energy that is generated by wind power plants in Germany;
- The price of electricity in Germany.

Quite often in a single European energy system, critical situations arise related to the excessive production of electricity by renewable sources in Germany. This leads to violations of planned electricity supplies, cross-border lines overload, and price hikes in the electricity market. One possible solution to this problem is to more accurately predict the amount of energy generated by renewable energy sources. To build forecasting models, the Python programming language and the Sklearn machine-learning library were used [10]. This library allows you to quickly build regression models, as well as conduct a quick analysis of the results. To predict the used three regression models: linear, quadratic and cubic. The following data were used as input [11]:

- 1) Wind speed, km/h;
- 2) Solar radiation, W/m^2 ;
- 3) Direct solar radiation, W/m^2 ;
- 4) Air temperature, $^{\circ}C$;
- 5) Installed power of solar power plants, MW;
- 6) Installed power of wind power plants, MW;
- 7) Electricity price, $\text{€}/MWh$;
- 8) Planned load power, MW.

For the first models and the simplicity of the graphical representation, only one attribute was used to train the model. With an increase in the number of features, the accuracy of model predictions increases, however, it is no longer possible to graphically present a model with six features. To predict the amount of energy in wind power plants, the sign of wind speed was used, and to predict the amount of energy in solar power plants, the sign of direct solar radiation was used. For models in which many features were used, only graphs of the overlay of the predicted results on the actual data are presented.

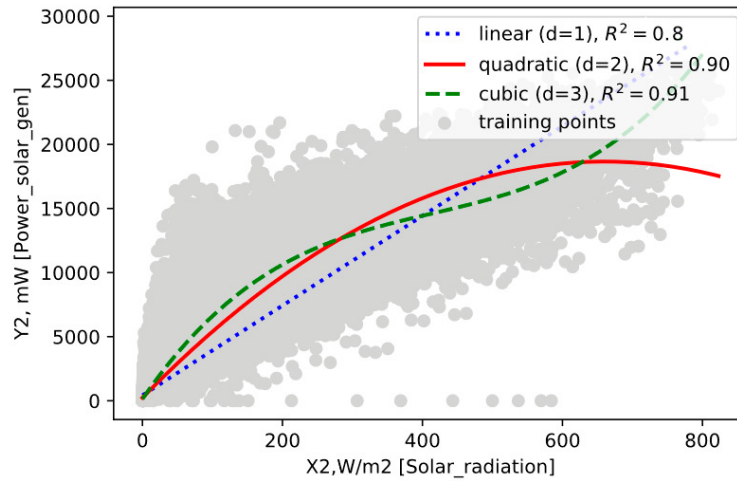


Fig. 2 Schedule of regression lines for generating energy from solar power plants.

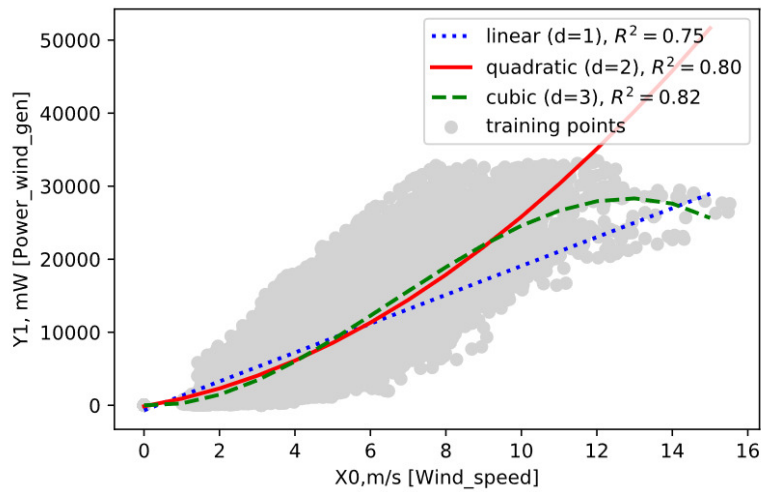


Fig. 3 Schedule of regression lines for generating energy from wind farms.

As can be seen in graphs 2 and 3 with the use of quadratic and cubic regression, the accuracy of predictions increases. However, a further increase in the degree of the polynomial does not give a significant increase, but it increases the operating time of the algorithm. To further increase the accuracy of forecasts, it is necessary to add additional features. In the future, quadratic regression for 6 features is used. The accuracy can be estimated by table 1, as well as on graphs 4, 5, 6.

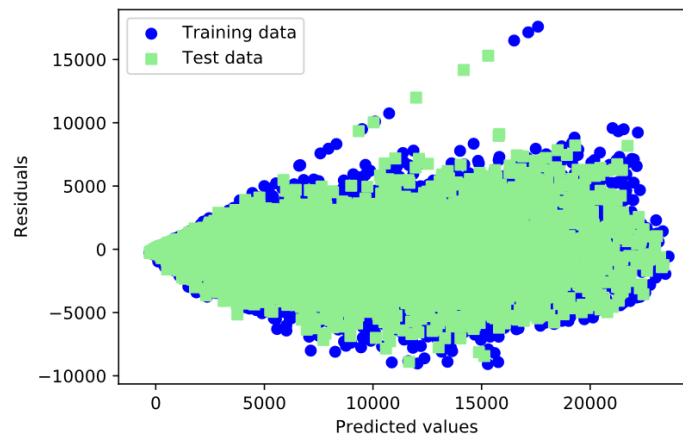


Fig. 4 Model predictive accuracy graph for solar power

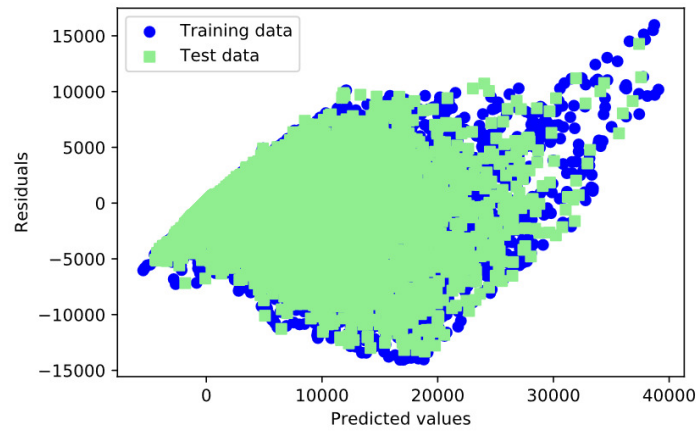


Fig. 5 Model predictive accuracy graph for wind power

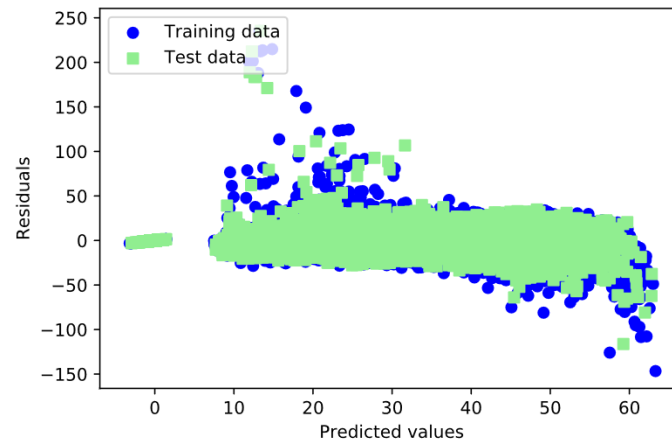


Fig. 6 Model predictive accuracy graph for electricity prices

As can be seen in the fig. 4, 5, 6, the accuracy of the forecasts is quite high. On the graphs, you can see that the worst of the entire model is able to predict values with a strong deviation. To predict such values, a large sample is needed to train the model, as well as additional features, such as wind direction, data on local hurricanes (storms), and repairs on power lines / substations / power plants. You can see the data on the indicator (coefficient of determination of R^2) R^2 in table 1. The closer the values of this indicator are to 1, the higher the accuracy of the model predictions.

Table 1
 Coefficient of Determination of R^2

Model	R^2
1 Feature predictions solar power (linear)	0.8
1 Feature predictions solar power (quadratic)	0.9
1 Feature predictions solar power (cubic)	0.91
1 Feature predictions wind power (linear)	0.75
1 Feature predictions wind power (quadratic)	0.80
1 Feature predictions wind power (cubic)	0.82
6 Features predictions wind power	0.87
6 Features predictions solar power	0.94
6 Features predictions price electricity	0.79

IV. CONCLUSION

Accuracy of the model predictions is quite high, especially in relation to the energy generated by solar power plants; this is primarily due to the constancy of solar radiation. To improve the forecasting of the amount of energy in wind power plants, it is necessary to add such a feature as the direction of the wind, this parameter plays a rather important role in the possibility of energy production at this type of power plant. The accuracy of forecasting electricity prices is not the highest. This is mainly because not only Germany but also a number of neighboring countries

have access to the exchange (where the price is set), to improve the accuracy of forecasts, we need data on energy consumption, as well as energy generated on renewable energy in these countries.

With the increase in the number of renewable energy sources, as well as the number of smart meters, the relevance of machine learning in the electric power industry will only increase.

ACKNOWLEDGMENT

This work was supported by the Scientific Grant Agency of the Ministry of Education of Slovak Republic and the Slovak Academy of Sciences by the projects VEGA No. 1/0372/18

REFERENCES

- [1] A. Mogilenko, "Application of artificial intelligence algorithms in the global energy industry", in *Energy and Industry of Russia*, 2018, no 7, p. 23.
- [2] A. Losev, "Artificial Intelligence in Nuclear Power", in *Atomic Expert*, 2018, no 3-4, pp. 35-43.
- [3] E. Kamchatova, A. Vasilieva, N. Lyasnikov, M. Dudin, N. Vysotskaya, "Energy saving management in urban economy and industry" in *International Journal of Civil Engineering and Technology*, 2018, vol. 9, no 6, pp. 1423-1429.
- [4] American Wind Energy Association, "U.S. wind industry annual market report year ending 2012," Amer. Wind Energy Assoc., Washington, DC, USA, 2013 [Online]. Available: [http://awea.files.cmsplus.com/images/AWEA_USWindIndustryAnnualMarketReport2012_ExecutiveSummary\(2\).pdf](http://awea.files.cmsplus.com/images/AWEA_USWindIndustryAnnualMarketReport2012_ExecutiveSummary(2).pdf).
- [5] "How does the domestic business connect to digitalization projects in the power industry", 24 RosInfo, (Unpublished work style)," unpublished
- [6] R. Jursa and K. Rohrig, "Short-term wind power forecasting using evolutionary algorithms for the automated specification of artificial intelligence models", in *Int. J. Forecast.*, vol. 24, pp. 694–709, 2008.
- [7] M. Monfared, H. Rastegar, and H.M. Kojabadi, "A new strategy for wind speed forecasting using artificial intelligent methods", *Renew. Energy*, vol. 34, pp. 845–848, 2009
- [8] D. A. Freedman, *Statistical Models: Theory and Practice*, Cambridge University Press. p. 26.
- [9] Warne, T. Russell (2011). "Beyond multiple regression: Using commonality analysis to better understand R2 results". *Gifted Child Quarterly*. 55 (4): 313–318. doi:10.1177/0016986211422217
- [10] Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, *JMLR* 12, pp. 2825-2830, 2011.
- [11] Open Power System Data. [Online]. Available: <https://data.open-power-system-data.org>

Rozpoznávanie vzorov atmosférických spršok v dátach EUSO experimentov

¹Kristián GOLIAŠ, ²Ján GENČI, ³Michal VRÁBEL

^{1,2,3}Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹kristian.golias@student.tuke.sk, ²jan.genci@tuke.sk, ³michal.vrabel@tuke.sk

Abstrakt – Tento článok sa venuje spracovaním dát z EUSO misií, špecificky dátam z EUSO-SPB1 misie. Pri skúmaní detekcie rôznych druhov javov sme pracovali s výstupmi skrytej vrstvy nami navrhnutých autoenkóderov alebo upraveného DCEC autoenkódera, ktoré spracovali dáta z EUSO-SPB1 misie. Výstupné dáta zo skrytej vrstvy sme ďalej spracovali pomocou t-SNE a UMAP algoritmov určených na dimenzionálnu redukciu. Vyhodnocovanie výsledkov prebiehalo prezeraním vizualizácií zobrazujúcich výstupy týchto algoritmov, kde sme sa sústredili na zoskupovanie záznamov s dráhou častice v jednom snímku. Podľa nášho názoru vizualizácie dimenzionálnej redukcie a klastrovania ukázali, že dáta z netrénovaných modelov autoenkóderov nedokázali jednoznačne zoskupovať záznamy s dráhou častice v jednom snímku.

Kľúčové slová – EUSO-SPB1, dimenzionálna redukcia, klastrovanie, autoenkóder

I. ÚVOD

Let EUSO-SPB1 misie bol naplánovaný na 24. apríla 2017 z Wanaka základni v Novom Zélande a let bol naplánovaný na 100 dní [1]. Hlavnými úlohami EUSO-SPB1 misie boli prvé pozorovanie atmosférických spršok pri pohľade zo suborbitálneho priestoru, meranie pozadia ultra vysokého svetla v noci cez oceán a mraky a vyhľadanie iných javov, ktoré sa prejavujú rýchlymi pulzmi v UV spektre [1]. Počas letu však nastali technické problémy a misia bola ukončená po 12 dňoch [1]. Väčšinu časť dát z letu sa však podarilo zachrániť a môžu tak byť vyhodnocované a analyzované výskumnými tímami.

V dátach aktuálne neboli nájdené žiadne udalosti so sprškou a to navzdory použitiu rôznych prístupov. Pomocou simulácii detektora a spršok v nástroji EUSO OffLine, odhaduje, že v dátach sa nachádza približne 1.2 ± 0.4 udalosti obsahujúcich spršku [2]. Ak by však let misie EUSO-SPB1 trval ako bolo naplánované, tak by bola vysoká pravdepodobnosť odhalenia javov obsahujúcich spršku. Tieto dáta však poslúžia ako príprava na budúcu misiu EUSO-SPB2, ktorá by sa mala začať v roku 2022 [2].

Detektor slúžiaci na pozorovanie atmosférických spršok v misii EUSO-SPB1 pozostáva z ohniskovej plochy o veľkosti jedného PDM, čo predstavuje 48×48 pixelov. PDM je hlavnou súčasťou detektora, ta sa však rozdeľuje ešte na menšie bloky. Tieto menšie bloky sa nazývajú EC a veľkosť takéhoto jedného bloku je 16×16 pixelov. EC však ešte nieje najmenší typ bloku v ohniskovej ploche. Tým je MAPMT a veľkosť jedného MAPMT je 8×8 pixelov, čo znamená, že v jednom bloku EC sa nachádzajú 4 bloky MAPMT. Teda celkový počet pixelov ohniskovej plochy pre EUSO-SPB1 detektor je 2304.

Dáta zozbierané z detektora počas EUSO-SPB1 misie sú uložené v súboroch nazývaných *ACQUISITION*. *ACQUISITION* súbory obsahujú pakety, ktoré dokázali aktivovať spúšťač systému počas pozorovania v časovom rozpätí 120 alebo 40 sekúnd [3]. Štruktúra jedného paketu pozostáva z 128 nespracovaných obrázkov, kde jeden obrázok má rozmer 48×48 pixelov [4].

II. SÚVISIACA PRÁCA

Nájdenním reprezentácie dát na vysokej úrovni sa zaoberal experiment zaoberajúcimi sa anti-neutrínami produkovaných v Daya Bay a Ling Ao jadrových elektrárni. Na získavanie údajov sa použil detektor o veľkosti 192 PMT [5]. Naučenie nových reprezentácií prebiehalo použitím neurónových sietí s dozorom a bez dozoru. Pri dozorovom učení sa pracovalo s konvolučnou neurónovou sieťou a pri bez dozorovom učení sa pracovalo s konvolučným autoenkóderom [6].

Na hodnotenie reprezentácií bol použitý algoritmus strojového učenia pre vizualizáciu t-SNE [6]. Úlohou tohto algoritmu je redukcia n-rozmerných údajov do 2 alebo 3 dimenzií. Tento algoritmus taktiež zabezpečuje, aby body pri sebe blízko v n-rozmerovom priestore boli blízko pri sebe taktiež aj v nižších rozmerov [7].

Výsledky experimentu ukázali, že konvolučné autoenkóbery môžu úspešne rozpoznávať vzory v dátach [6]. Dáta skúmané v tomto experimente sú podobné dátam z misie EUSO-SPB1, pretože v oboch prípadoch sa jedná v podstate o 2D obrázky. Keďže v experimente boli schopní úspešne rozpoznávať vzory v dátach bolo pre nás veľkou motiváciou použitie podobného postupu ako v tomto experimente, to znamená použitie pre dimenzionálnu redukciu algoritmus t-SNE a konvolučný autoenkóder na rozpoznávanie vzorov.

O možnosti využitia konvolučných neurónových sietí pre EUSO-SPB1 misiu sa zaoberala diplomová práca Petra Szakácsa [8]. Na natrénovanie modelov boli použité dáta, ktoré boli generované dvomi rozličnými spôsobmi, kde modely boli najprv natrénované osobitne a potom spolu.

Dátové sady použité na tréning modelov pozostávajú z troch zložiek. Počet záznamov v týchto troch zložkách je vždy rovnaký, pretože pozícia určujeme o aký záznam sa jedná. Prvou zložkou sú samotné dáta, ktoré boli vytvorené zo snímkov paketu alebo simulácii. Tieto dáta pozostávajú z projekcií XY, GTUX a GTUY. Projekcia XY je projekcia maximálnych hodnôt snímkov paketu v čase [8]. Projekcia GTUX je projekcia maximálnych hodnôt pixlov pozdĺž osi Y [8]. Projekcia GTUY je projekcia maximálnych hodnôt pixlov pozdĺž osi X [8]. Druhou zložkou sú očakávané výstupy neurónovej siete. Tretou zložkou sú metadáta, ktoré obsahujú dodatočné informácie o záznamoch ako sú napríklad cesta k súboru obsahujúcich paket v ktorom sa záznam nachádza, index paketu v danom súbore, index prvého a posledného použitého snímku z paketu.

Výsledky natrénovaných modelov vyhodnotili v dáta z EUSO-SPB1 misie viac než 200 položiek dát ako potencionálne zaujímavých [8]. V našej práci sme využili podobné dátové sady na tréning našich autoenkóderov. Z dátových sád nás však budú hlavne zaujímať zložky dáta a metadáta. V práci pracujeme iba s projekciou XY zo zložky dáta.

III. NAŠE RIEŠENIE

Na vývoj našich autoenkóderov sme použili knižnicu Keras, presnejšie TensorFlow vysoko úrovňovú API Keras. Jedným z hlavných dôvodov použitia tejto knižnice bolo, možnosť rýchlej implementácie autoenkódera a umožnenie tak rýchleho experimentovania s autoenkóderom.

Na spracovanie výstupných dát zo skrytej vrstvy autoenkóderov sme pre t-SNE algoritmus použili implementáciu z knižnice t-SNE-CUDA. Pred realizáciou našej práce, dostupná implementácia t-SNE pre spracovanie dát z EUSO-SPB1 misie bola Multicore-TSNE a t-SNE z knižnice scikit-learn. Knižnica t-SNE-CUDA je oproti aktuálnym knižniciam niekoľkonásobne rýchlejšia [9].

A. Konvolučný autoenkóder (*Convolutional autoencoder*)

Neurónové siete, kde modely sú tréňované aby ich cieľový výstup bol čo najviac podobný ako vstup, sa nazývajú autoenkóbery [10]. Úlohou autoenkóderov je naučiť sa, ako komprimovať a rekonštruovať príklady z cvičných dát. Autoenkóbery pozostávajú z enkóderovej a dekóderovej časti. Úlohou enkódera je transformovať vstup do vektora na výstupe a úlohou dekódera je rekonštrukcia výstupu dekódera na pôvodný vstup. Ak architektúra autoenkódera pozostáva z konvolučných vrstiev, tak sa jedná o konvolučný autoenkóder. Úlohou konvulčnej vrstvy je extrahovanie vlastností zo vstupného obrázka a zachovanie vzťahu medzi pixelmi pomocou učenia sa obrazových prvkov pomocou malých štvorcov zo vstupných údajov.

Na vývoj nášho konvulčného autoenkódera bola využitá TensorFlow vysoko úrovňová API Keras. Základnou myšlienkou pri návrhu nášho konvulčného autoenkódera bola, časť neurónovej siete slúžiaca ako enkóder obsahovala 2 až 3 konvulčné vrstvy a výstup z enkóder časti obsahoval približne okolo 100 neurónov. Architektúru nášho konvulčného autoenkódera môžeme vidieť v tejto tabuľke 1. Ako sa dá z tabuľky vyčítať pri enkóder časti sme použili 3 konvulčné vrstvy nasledovaných max-pooling vrstvou a výstup z enkóder časti sa skladá z 144 neurónov. Znak ? v obrázku predstavuje počet záznamov, ktoré sú poslané na vstupe.

B. Hustý autoenkóder (*Dense autoencoder*)

Ak architektúra autoenkódera pozostáva z hustých vrstiev, tak sa jedná o hustý autoenkóder. Hustá vrstva je vrstva pri ktorej je každý vstupný neurón pripojený k výstupnému neurónu. Taktiež aj na vývoj nášho hustého autoenkódera bola využitá TensorFlow vysoko úrovňová API Keras. Základnou myšlienkou pri návrhu nášho hustého autoenkódera bola, aby časť neurónovej siete slúžiaca ako enkóder obsahovala 2 husté vrstvy a výstup z enkóder časti obsahoval približne okolo

vrstva	typ	vstup	vystup	aktivácia
1	InputLayer	(?, 48,48)	(?, 48,48)	-
2	Reshape	(?, 48,48)	(?, 48,48,1)	-
3	Conv2D	(?, 48,48,1)	(?, 48,48,32)	relu
4	MaxPooling2D	(?, 48,48,32)	(?, 24,24,32)	-
5	Conv2D	(?, 24,24,32)	(?, 24,24,64)	relu
6	MaxPooling2D	(?, 24,24,64)	(?, 12,12,64)	-
7	Conv2D	(?, 12,12,64)	(?, 12,12,1)	relu
8	Flatten	(?, 12,12,1)	(?, 144)	-
9	Reshape	(?, 144)	(?, 12,12,1)	-
10	Conv2DTranspose	(?, 12,12,1)	(?, 12,12,1)	relu
11	UpSampling2D	(?, 12,12,1)	(?, 24,24,1)	-
12	Conv2DTranspose	(?, 24,24,1)	(?, 24,24,64)	relu
13	UpSampling2D	(?, 24,24,64)	(?, 48,48,64)	-
14	Conv2DTranspose	(?, 48,48,64)	(?, 48,48,32)	relu
15	Conv2DTranspose	(?, 48,48,32)	(?, 48,48,1)	relu
16	Reshape	(?, 48,48,1)	(?, 48,48)	-

Tabuľka 1: Architektúra nami navrhnutého konvolučného autoenkódera.

100 neurónov. Teda myšlienka bola podobná ako pri konvolučnom autoenkódere, ale namiesto konvolučných vrstiev sme použili hustú vrstvu. Architektúru nášho hustého autoenkódera môžeme vidieť v tejto tabuľke 2. Ako sa dá z tabuľky vyčítať pri enkóder časti sme použili 2 husté vrstvy a výstup z enkóder časti sa skladá z 128 neurónov. Znak ? v obrázku predstavuje počet záznamov, ktoré sú poslané na vstupe.

vrstva	typ	vstup	vystup	aktivácia
1	InputLayer	(?, 48,48)	(?, 48,48)	-
2	Flatten	(?, 48,48)	(?, 2304)	-
3	Dense	(?, 2304)	(?, 256)	relu
4	Dense	(?, 256)	(?, 128)	relu
5	Dense	(?.128)	(?, 256)	relu
6	Dense	(?, 256)	(?, 2304)	relu
7	Reshape	(?, 2304)	(?, 48,48)	-

Tabuľka 2: Architektúra nami navrhnutého hustého autoenkódera.

C. DCEC autoenkóder

DCEC autoenkóder je konvolučný autoenkóder zložený z konvolučných a hustých vrstiev a klaster vrstvy, spojenými s vloženou vrstvou [11]. Dôvod zvolenia tohto autoenkódera je kvôli vyskúšaní sofistikovanejšej metódy pre identifikáciu klastrov v záznamoch pomocou autoenkódera. Aby sme však boli schopní použiť DCEC autoenkóder na našich dátových sádach, bolo potrebné najprv upraviť implementáciu DCEC autoenkódera. Upráva sa hlavne týkala funkcionality aby autoenkóder dokázal spracovať na vstupe naše dátové sady. Architektúru DCEC autoenkódera môžeme vidieť v tejto tabuľke 3. Znak ? v obrázku predstavuje počet záznamov, ktoré sú poslané na vstupe.

vrstva	typ	vstup	vystup	aktivácia
1	InputLayer	(?, 48,48,1)	(?, 48,48,1)	-
2	Conv2D	(?, 48,48,1)	(?, 24,24,32)	relu
3	Conv2D	(?, 24,24,32)	(?, 12,12,64)	relu
4	Conv2D	(?, 12,12,64)	(?, 6,6,128)	relu
5	Flatten	(?, 6,6,128)	(?, 4608)	-
6	Dense	(?, 4608)	(?, 10)	-
7a	Dense	(?, 10)	(?, 4608)	-
8a	Reshape	(?, 4608)	(?, 6,6,128)	-
9a	Conv2DTranspose	(?, 6,6,128)	(?, 12,12,64)	relu
10a	Conv2DTranspose	(?, 12,12,64)	(?, 24,24,32)	relu
11a	Conv2DTranspose	(?, 24,24,32)	(?, 48,48,1)	relu
7b	ClusteringLayer	(?, 10)	(?, 10)	-

Tabuľka 3: Architektúra DCEC autoenkódera.

D. Spracovanie výstupov z výstupnej časti enkódera

Výstupy z výstupnej časti enkódera sme ďalej spracovali pomocou algoritmov t-SNE-CUDA a UMAP určených na dimenzionálnu redukciu. Výstupnú enkóder časť v konvolučnom autoenkódere predstavuje ôsma vrstva z tabuľky 1, v hustom autoenkódere sa jedná o štvrtú vrstvu z tabuľky 2 a v DCEC autoenkódere sa zasa jedná o vrstvu 6 z tabuľky 3. Výstupy algoritmov t-SNE-CUDA a UMAP sú hodnoty X a Y pozícií záznamov v 2-D priestore.

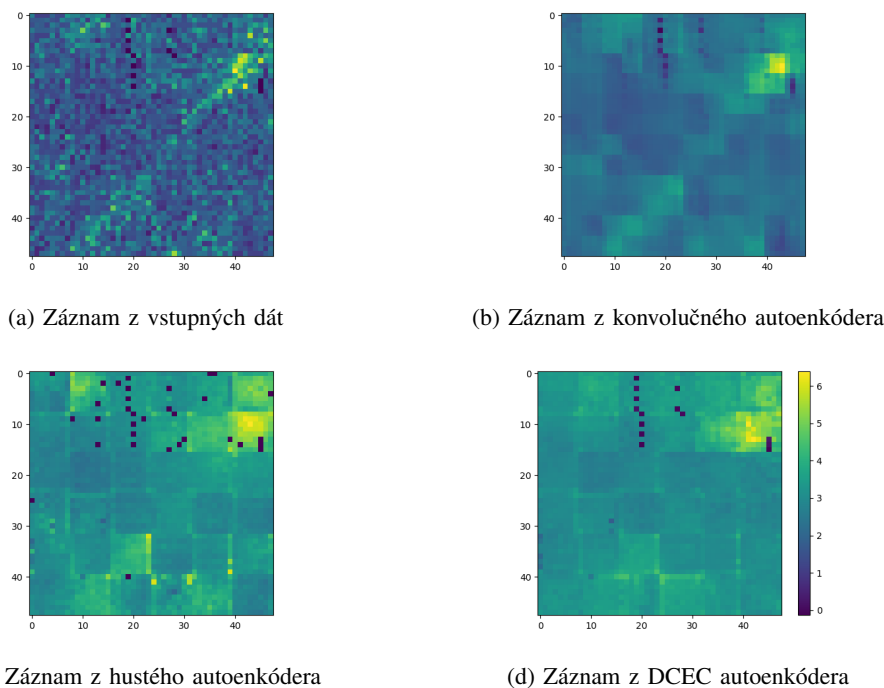
IV. TRÉNING MODELOV

Dátová sada na tréningovanie modelov pozostávala z viac ako 200 000 záznamov z letu EUSO-SPB1. Na týchto záznamov boli tréňované všetky modely konvolučných, hustých a DCEC autoenkóderov. Pri tréningu sme použili optimalizáciu Adam a MeanSquaredError pre meranie hodnoty funkcií straty a metriky. Každý model autoenkóderov bol tréňovaný od nuly, kde dĺžka každého tréningu pozostávala z 500 epochou. Jedna epocha je, keď sa spracuje celá dátová sada raz.

V. HODNOTENIE

Hodnotenie autoenkóderov prebiehalo porovnávaním výslednej hodnoty funkcie straty a porovnaním vizualizácii výstupov jednotlivých výstupov dekóder časti s vstupnými dátami. Z porovnaní výsledných hodnôt funkcie straty najlepšie obstal model nášho konvolučného autoenkódera, nasledovaní DCEC autoenkóderom a najhorší výsledok sme dosiahli s hustým autoenkóderom. Podobne sme obstáli aj pri vizualizáciách jednotlivých výstupov dekóder časti s vstupnými dátami. Výsledky však dopadli podobne ako pri výsledných hodnotách funkcie straty.

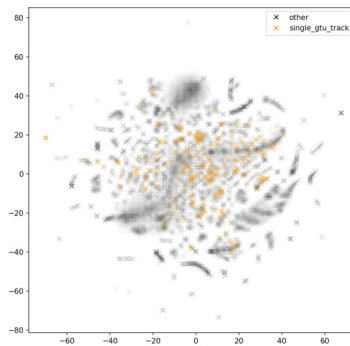
Príklad vizualizácie jedného záznamu môžeme vidieť na obrázku 1, kde sú zvýraznené záznamy označené klasifikačnou triedou "single_gtu_track". Klasifikačná trieda "single_gtu_track" predstavuje jav dráhy častice v jednom snímku, pri ktorom nemôžeme pozorovať pohyb častice medzi snímkami. Očakávaným dôvodom vzniku tohto javu je zasiahnutie častice kozmického žiarenia nízkej energie do samotného detektora.



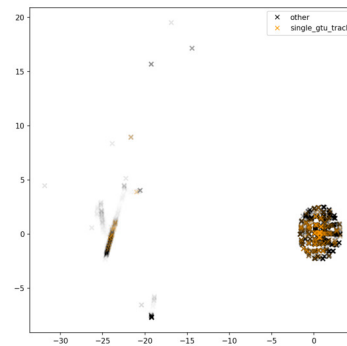
Obr. 1: Hodnoty záznamu pre vstupné dáta a výstupné dáta konvolučného, hustého autoenkódera a DCEC autoenkódera.

Ďalším krokom bola vizualizácia výstupov skrytej vrstvy z konvolučného autoenkódera spracovaných algoritmi dimenzionálnej redukcie t-SNE-CUDA a UMAP. Vo vizualizáciách sme zisťovali, či sa záznamy označené klasifikačnou triedou "single_gtu_track" zoskupujú. Dôvod vybratia tejto klasifikačnej triedy je ten, že sa zo všetkých klasifikačných tried najviac podobá javu so sprškou. Výsledné vizualizácie však ukázali, že výstupy z dimenzionálnych redukcii nedokázali jednoznačne zoskupovať záznamy označené ako klasifikačná trieda "single_gtu_track". Výsledok vizualizácia na surových dátach môžeme vidieť na obrázku 2.

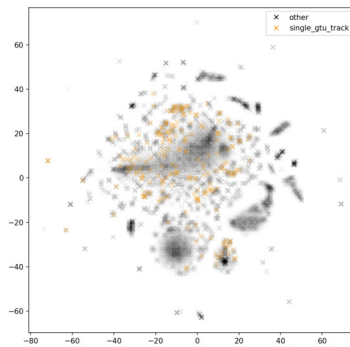
Keďže modely konvolučného autoenkódera tréňovaných na surových dátach nedokázali jednoznačne zoskupovať záznamy označené ako klasifikačná trieda "single_gtu_track", rozhodli sme sa dátovú sadu, ktorú sme použili na tréning autoenkóderov filtrovať pomocou BM3D algoritmu, lokálnej a globálnej Otsu metódy, Yen metódy a nakoniec non-local means algoritmu. Výsledky avšak ani po filtrácii neboli to čo by sme očakávali, to je zoskupovanie záznamov označených ako klasifikačná trieda "single_gtu_track". Výsledok vizualizácii na dátach výstupov z BM3D algoritmu, lokálnej a globálnej Otsu metódy, Yen metódy a non-local means algoritmu môžeme vidieť na obrázku 2.



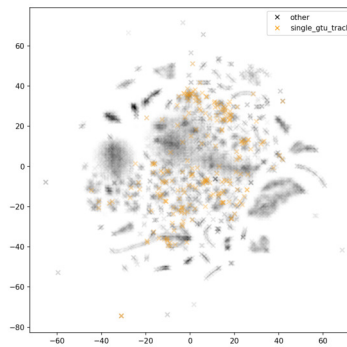
(a) "Surové" dáta



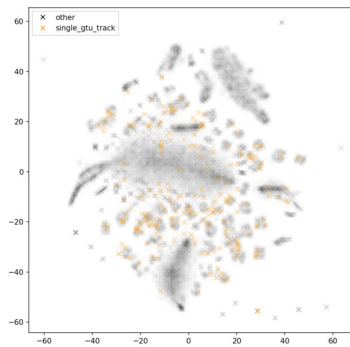
(b) BM3D dáta



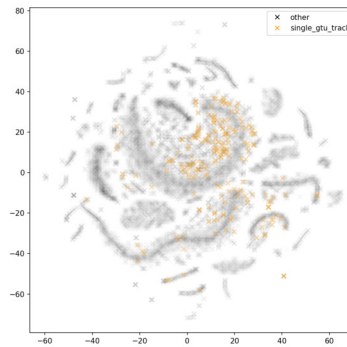
(c) Lokálne Otsu dáta



(d) Globálne Otsu dáta



(e) Yen dáta

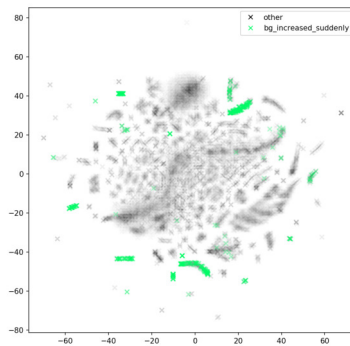


(f) Non-local means dáta

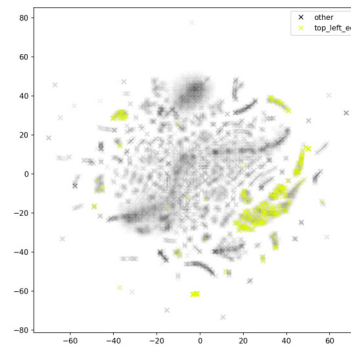
Obr. 2: Vizualizácie výstupov t-SNE-CUDA algoritmu zvyrazňujúce manuálnu klasifikačnú triedu `single_gtu_track` pre jednotlivé modely nášho konvolučného autoenkódera.

Keďže vo vytvorených vizualizáciách boli viditeľné určité klastre, preto sme prezerali o aké klasifikačné triedy sa jedná. Po prezretí vizualizácii sme zistili, že sa jedná o klasifikačné triedy "bg_increased_suddenly" a "top_left_ec". Klastrovanie týchto klasifikačných tried môžeme vidieť na obrázku 3. Klasifikačná trieda "bg_increased_suddenly" predstavuje jav, kedy sa hodnoty pozadia náhle zvýšia a klasifikačná trieda "top_left_ec" predstavuje jav, kedy sú hodnoty v hornom ľavom EC vyššie ako pozadie.

Prekvapivú vizualizáciu ukázal výstup z modelu konvolučného autoenkódera naučeného na vyfiltrovaných dátach z BM3D algoritmu. Pretože ako môžeme vidieť väčšina záznamov sa nachádza v jednom klastry. Pri prvotnom prezretí obrázku sme si mysleli, že došlo k chybe pri konfigurácii t-SNE-CUDA algoritmu avšak podobného výsledku sme dosiahli aj pri vizualizácii výstupov z algoritmu UMAP. Ak výstupy z výstupu enkóder časti sú správne a pri natrénovaní modelu nedošlo k žiadnej chybe, tak sa jedná o zaujímavú vizualizáciu. Avšak problém pri tejto vizualizácii je, že väčšina záznamov sú v jednom klastry a v menších klastroch sa nenachádzajú iba záznamy označené jednou klasifikačnou triedou.



(a) Klasifikačná trieda “bg_increased_suddenly”



(b) Klasifikačná trieda “top_left_ec”

Obr. 3: Vizualizácie výstupu t-SNE-CUDA algoritmu zvyrazňujúce manuálne klasifikačné triedy, pri ktorých sa záznamy jednoznačne klastrovali.

VI. ZÁVER

Práca sa zaujíma skúmaním potenciálu možnosti detekcie rôznych druhov javov v dátach z EUSO-SPB1 misie. Na detekciu rôznych druhov javov sme použili nami navrhnuté modely konvolučného a hustého autoenkódera a DCEC autoenkóder. Avšak ani jeden z modelov nedokázal jednoznačne zoskupovať záznamy s dráhou častice v jednom snímku. Podobné výsledky sme dostali aj po filtrácii “surových” tréningových dátových sadách pomocou BM3D algoritmu, lokálnej a globálnej Otsu metódy, Yen metódy a nakoniec non-local means algoritmu.

Preto v budúcich prácach zaoberajúcich sa možnosťou detekcie rôznych druhov javov v dátach JEM-EUSO misií odporúčame implementáciu sofistikovanejšieho autoenkódera, ktorý by mohol ponúknuť zaujímavejšie výsledky. Ďalším možným vylepšením by bolo analyzovanie hustoty bodov v klastroch a nie len ich vizualizácia.

Skripty a metódy použité v našej práci môžu byť použité aj na budúcich dátach pripravovanej misie EUSO-SPB2, ktorá by sa mala konať v roku 2022. Práca taktiež môže slúžiť ako základ pre ďalšie práce zaoberajúcou sa podobnou tematikou pre JEM-EUSO misie.

LITERATÚRA

- [1] L. Wiencke and A. O. and, “EUSO-SPB1 mission and science,” in *Proceedings of 35th International Cosmic Ray Conference — PoS(ICRC2017)*. Sissa Medialab, Oct. 2017. [Online]. Available: <https://doi.org/10.22323/1.301.1097>
- [2] K. Shinozaki, M. Bertainia, F. Bisconti, F. Fenu, S. Ferrarese, S. Monte, A. Anzalone, A. Bruno, S. Briz, A. Diaz, J. Eser, L. Wiencke, A. Olinto, and M. V. for JEM-EUSO Collaboration, “Estimation of the exposure for the air shower detection mode of eso-spb1,” 2019.
- [3] M. Vrábel, “Pattern recognition in the jem-euso experiment,” Ph.D. dissertation, Technická univerzita v Košiciach, 4 2020.
- [4] C. Fornaro, F. S. Cafagna, G. Osteria, V. Scotti, F. Perfetto, and L. Conti, “The onboard software of the EUSO-SPB pathfinder experiment,” *Software: Practice and Experience*, vol. 49, no. 3, pp. 524–539, Oct. 2018. [Online]. Available: <https://doi.org/10.1002/spe.2655>
- [5] F. P. An, Q. An, J. Z. Bai *et al.*, “Improved measurement of electron antineutrino disappearance at daya bay,” *Chinese Physics C*, vol. 37, no. 1, p. 011001, Jan. 2013. [Online]. Available: <https://doi.org/10.1088/1674-1137/37/1/011001>
- [6] E. Racah, S. Ko, P. Sadowski, W. Bhimji, C. Tull, S.-Y. Oh, P. Baldi, and Prabhat, “Revealing fundamental physics from the daya bay neutrino experiment using deep neural networks,” in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, Dec. 2016. [Online]. Available: <https://doi.org/10.1109/icmla.2016.0160>
- [7] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
- [8] P. Szakács, “Aplikácia neurónových sietí pre detekciu atmosférických spršok v dátach z experimentu eso-spb,” Master’s thesis, Technická univerzita v Košiciach, 4 2019.
- [9] D. M. Chan, R. Rao, F. Huang, and J. F. Canny, “Gpu accelerated t-distributed stochastic neighbor embedding,” *Journal of Parallel and Distributed Computing*, vol. 131, pp. 1–13, 2019. [Online]. Available: <https://doi.org/10.1016/j.jpdc.2019.04.008>
- [10] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, “Greedy layer-wise training of deep networks,” in *Advances in neural information processing systems*, 2007, pp. 153–160.
- [11] X. Guo, X. Liu, E. Zhu, and J. Yin, “Deep clustering with convolutional autoencoders,” in *Neural Information Processing*, D. Liu, S. Xie, Y. Li, D. Zhao, and E.-S. M. El-Alfy, Eds. Cham: Springer International Publishing, 2017, pp. 373–382.

Softvérový pH normalizátor

¹Marek Skyba, ²Anna Birková, ³Ján Genčí

¹ Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

² Ústav lekárskej a klinickej biochémie, Lekárska fakulta, Univerzita Pavla Jozefa Šafárika v Košiciach, Slovenská republika

³ Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹marek.skyba@student.tuke.sk, ²anna.birkova@upjs.sk, ³jan.genci@tuke.sk

Abstrakt — Fluorescenčná analýza moču je jednou z novších a menej preskúmaných metód diagnostiky pacientov. Cieľom tejto práce je vytvorenie pH normalizátora, ktorý by normalizoval hodnoty fluorescence fluorescenčnej analýzy s istým pH na iné pH zadané používateľom, a zjednodušil by tak spracovanie dát. V prvej časti práce je rozobratá problematika fluorescence, moču, pH a korelácie. Poznatky z týchto oblastí boli využité pri nájdení závislosti medzi pH a fluorescenciou. V druhej časti práce je opísaný postup pri hľadaní tejto závislosti, ktorý pozostával z vizualizácie dát rôznymi spôsobmi a ich analýzou. Závislosť od pH sa ukázala rozdielna pre moč dospelého a detského človeka. Po nájdení vzťahu je opísaná implementácia softvérového pH normalizátora, ktorý normalizuje dáta podľa pH samostatne pre dva typy močov. Dáta na analýzu boli poskytnuté Ústavom lekárskej fakulty chémie, biochémie a klinickej biochémie a LABMED a.s. LF UPJŠ v Košiciach.

Kľúčové slová — fluorescencia, pH, softvérový normalizátor

I. ÚVOD

Diagnostika zdravotného stavu pacientov v medicíne nie je, ani napriek výrazným pokrokom v nedávnej minulosti, triviálnou záležitosťou. Často býva časovo a finančne náročná, niektoré diagnostické metódy môžu byť pre pacientov veľmi nepríjemné. Aj kríza v podobe aktuálnej pandémie koronavírusu Covid-19 ukazuje potrebu rýchlych a lacných metód na diagnostiku chorôb, či iných anomálií v ľudskom tele.

Vedci sa aj z týchto dôvodov stále snažia prichádzať s novými metódami diagnostiky stavu pacientov a zlepšovať tie existujúce. Jednou z novších a menej známych metód je fluorescenčná analýza, ktorá využíva princíp fluorescence. Fluorescencia je jav, pri ktorom látka po vystavení svetelnému žiareniu vyžaruje svetlo. Fluorescenčná analýza sleduje práve toto vyžiarené svetlo, ktoré poskytuje rôzne informácie o analyzovanej látke.

V medicíne je fluorescenčná analýza využívaná napríklad pri skúmaní krvi, buniek, či moču. Moč sa na fluorescenčnú analýzu hodí z toho dôvodu, že obsahuje látky schopné fluorescence. Okrem toho je ho možné jednoducho získať od pacienta, bez zdĺhavých či nepríjemných procedúr. Pri analýze moču fluorescenčnou metódou je možné napríklad odlišiť zdravých a chorých pacientov na základe vzoriek.

Moč je látkou, ktorá obsahuje veľmi veľa substancií a každá z nich môže mať istý vplyv na hodnoty fluorescence. pH moču veľmi varíruje a závisí od množstva faktorov, a zatiaľ nebolo zistené, aká je súvislosť medzi pH moču a jeho fluorescenciou.

Práve fluorescenčnej analýze moču sa venuje táto práca. Motiváciou pre túto prácu bolo zjednodušenie práce expertov pri analýze moču touto metódou. Naša práca mala nájsť odpoveď zatiaľ nezodpovedanú otázku, a to aká je súvislosť medzi pH moču a jeho fluorescenciou.

Naším cieľom bolo zistiť, či existuje súvislosť medzi pH moču a jeho fluorescenciou a ak áno, vytvoriť softvérový pH normalizátor - program, ktorý znormalizuje namerané hodnoty fluorescence s konkrétnym pH moču na hodnoty, ktoré by nastali, ak by pH moču malo inú hodnotu. Takýto program by expertom pomohol zrýchliť a zefektívniť analýzu tým, že by nebola potrebná manuálna zmena pH moču.

II. AKTUÁLNY STAV PROBLEMATIKY

Existuje niekoľko prác, ktoré sa venujú podobnej problematike: fluorescencii, pH, moču a vzťahu medzi týmito veličinami. Prácu s úplne rovnakým zameraním sme nenašli.

Najbližšie k nášmu problému má práca Leinera a kolektívu [1], ktorí v práci zameranej na celkovú fluorescenciu ľudského moču skúmali aj efekty pH. Celkovú fluorescenciu ľudského moču merali pri rôznych excitačných a emisných vlnových dĺžkach a zistili, že aj napriek zložitosti moču je možné pozorovať 3 až 5 fluorescenčných maxim. Skúmaním vplyvu pH prišli k záveru, že s najväčšou pravdepodobnosťou pozorované maximá nie sú výsledkom jediného fluorescenčného metabolitu moču, ale skôr niekoľkých druhov, ktoré majú podobné spektrálne vlastnosti a sú prítomné v porovnateľných koncentráciách.

Lichardusová a kol. [2] skúmali niektoré konkrétne fluofóry moču spolu s tým, ako ich fluorescenciu ovplyvňujú rôzne faktory, okrem iných aj pH. V práci prišli k záveru, že pre metabolity tryptofán a tyrozín pH nemá vplyv na fluorescenčné maximá a môže byť zanedbaný. To však platí len pre konkrétne fluofóry, nie celý moč.

Dubayová a kol. [3] sa venovali koncentrácii moču a koncentračným matriciam fluorescencie. Rozlišovali moč podľa hustoty (obsah vody) na normálny, koncentrovaný a riedky. Skúmali a znázornili to, aký vplyv má hustota moču na jeho koncentračnú maticiu. To značí, že hustota moču je teda jednou z veličín, ktorú je potrebné pri snahe o normalizáciu fluorescencie moču zväžiť.

III. NÁVRH A IMPLEMENTÁCIA

Pre splnenie hlavného cieľa, ktorým bolo vytvorenie softvérového normalizátora, sme najprv potrebovali zistiť, či existuje korelácia medzi pH a fluorescenciou moču, a ak áno, aká je a ako je možné ju vyjadriť. K dispozícii sme mali vzorky štyroch močov (každý s piatimi rôznymi pH), čo nebol dostatočný počet na to, aby sme mohli použiť metódy strojového učenia. Preto sme túto metódu vylúčili. Rozhodli sme sa skúmať a vizualizovať dáta a takýmto spôsobom prísť na to, aký vzťah medzi spomenutými veličinami existuje.

A. Hľadanie závislosti medzi pH a fluorescenciou

Prvým krokom v hľadaní závislosti bola vizualizácia fluorescencie pri rôznych pH hodnotách jednej vzorky moču a každej vlnovej dĺžky bez akéhokoľvek zásahu do dát, či ich filtrovania. Vytvorili sme grafy pre každý moč a jeho riedenie, zobrazujúce hodnoty fluorescencií pri každom meraní na danej vlnovej dĺžke. Keďže krokovanie bolo každých 0,5 nanometrov vlnovej dĺžky a generovali sme grafy pre hodnoty od 250 do 548,5, pre každé riedenie bolo vygenerovaných 598 grafov. Analýzou týchto grafov sme zistili, že pri niektorých bola pozorovateľná istá závislosť, pri iných vôbec nie. Jednoznačný záver teda nebolo možné z týchto grafov vyvodit'.

Analýzou dát a konzultáciou s odborníkmi sme zistili, že veľké množstvo dát zobrazených v grafoch opísaných v predošlej kapitole bolo zbytočných a zobrazovali iba chybu či odchýlku pri meraní stroja. Preto sme sa v ďalšom kroku rozhodli:

- Ignorovať intenzitu fluorescencie do hodnoty 50, ktoré nemajú význam a môžu sa považovať len za namerané pozadie vzorky,
- nezobrazovať všetkých 13 meraní, ale len prvých 10, ktoré sa považujú za dôležité,
- pre lepšiu čitateľnosť grafov a ich ľahšie porovnanie os x normalizovať a os y pre každý graf naškálovať na rozpätie od 0 do 1000.

Aj grafov tohto typu bolo pre každé riedenie a hodnotu pH vygenerovaných 598. Ich analýzou sme zistili, že tento graf už vykazuje väčšiu informačnú hodnotu v porovnaní s nefiltrovaným a signalizuje aj to, že istý vzťah medzi hodnotou fluorescencie a pH existuje. Tieto grafy nám poskytli dôležitý poznatok v tom, že nie pri každej hodnote vlnovej dĺžky a riedenia boli krivky grafov podobné. Líšili sa, čo značí, že závislosť od pH sa s hodnotou vlnovej dĺžky a riedením mení.

Po zväžení zistenia, že sa závislosť od pH mení pre rôzne vlnové dĺžky, sme sa rozhodli pre pokus o vyjadrenie matematického vzťahu pre konkrétne vlnové dĺžky močov a nie celý graf. Z predošlého grafu totiž vyplývalo, že do vzťahu medzi pH a fluorescenciou zasahovalo viac premenných, o ktorých sme nevedeli a nemali sme ich zaznamenané. Jedným z týchto faktorov je hustota moču. V tabuľke č. 1 je možné vidieť hustoty jednotlivých vzoriek moču. Vzorka 1 nemá presnú hodnotu, pretože ju merací prístroj nedokázal zaznamenať, iba to, že bola väčšia ako 1030 g/dm³. Preto sme vzorke pre naše potreby priradili hodnotu 1035 g/dm³.

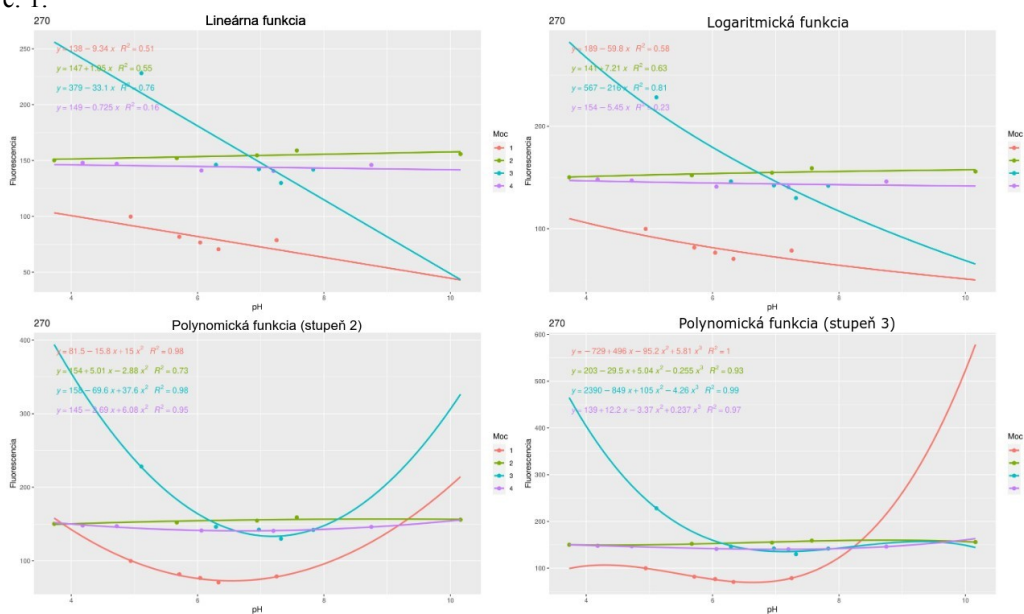
Keďže cieľom bolo vytvorenie systému, ktorý by normalizoval hodnoty fluorescencie koncentračných matric, ďalším krokom v hľadaní závislosti bola vizualizácia koncentračných matric, ktoré zahŕňujú všetky riedenia konkrétnej vzorky moču s určitým pH v jednom grafe. Z koncentračnej matrice nás konkrétne zaujímali maximálne hodnoty fluorescencie pri každej vlnovej dĺžke a ich porovnanie pri rôznych hodnotách pH. Preto sme vytvorili ďalšie tri grafy,

ktoré ukazujú krivky týchto maximálnych hodnôt, pre každý zo štyroch poskytnutých vzoriek moču. Výsledkom boli štyri grafy. Dôležité je spomenúť, že v moči pri vlnových dĺžkach od 310 do 340 hodnota fluorescencií z istého dôvodu (pacient, ktorý poskytol moč, mohol brať lieky alebo mať v tele iné látky spôsobujúce silnú fluorescenciu) presiahla detekčný limit stroja (1000) a zaznamenala len hodnoty blízke limitu. Z ďalšej analýzy sme preto tento interval vzorky moču 4 vylúčili, keďže by mohol skresliť výsledky. Analýzou tohto typu grafov sme zistili, že sa závislosť od pH mení pre rôzne vlnové dĺžky, a preto sme sa rozhodli pre pokus o vyjadrenie vzťahu matematicky pre jednotlivé vlnové dĺžky močov.

Tabuľka 1
 Atribúty analyzovaných vzoriek moču

Moč	Typ moču	Hustota (v g/dm ³)	Definovaná konštanta
Vzorka 1	detský	väčšia ako 1030 (1035)	1,35
Vzorka 2	dospelý	1005	1,02
Vzorka 3	detský	1030	1,3
Vzorka 4	dospelý	1020	2,2

Z hodnôt pre každú vzorku sme vyjadrili vzorec a krivku závislosti rôznymi funkciami, pričom x označuje hodnotu pH a y hodnotu fluorescencie. Takisto sme vyjadrili koeficient determinácie (R^2), ktorý označuje mieru kvality regresného modelu vyjadreného vzorcom. Skúšali sme vizualizovať štyri funkcie: lineárnu, logaritmickú a polynomicke so stupňom polynómu 2 a 3. Príklady týchto grafov pri hodnote 270 nanometrov sú ilustrované na obrázku č. 1.



Obr. 1 Grafy rôznych funkcií pri vlnovej dĺžke 270 nanometrov

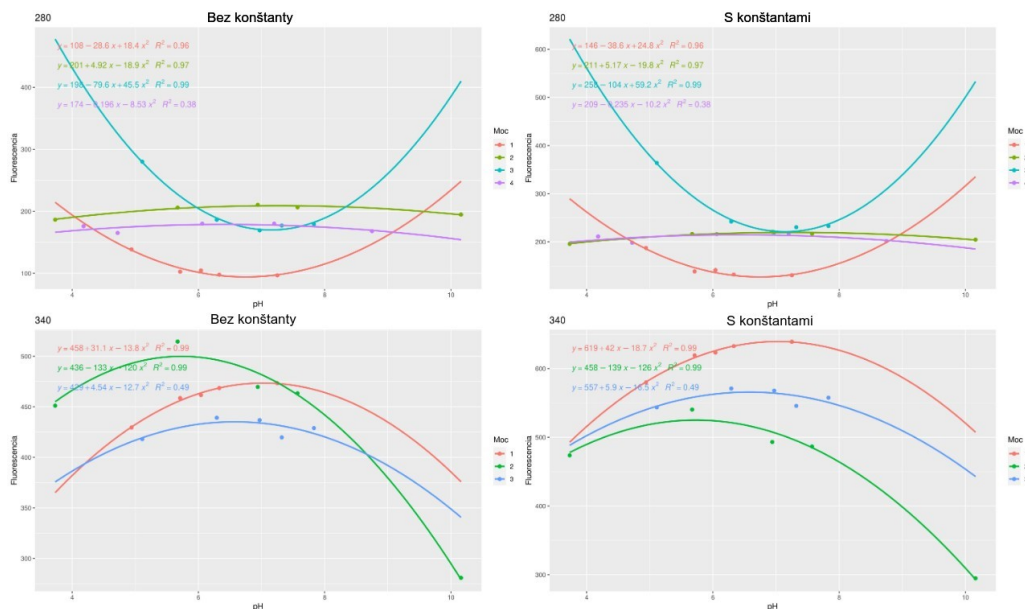
Ako najvhodnejšia funkcia s najväčšou presnosťou sa ukázala polynomická funkcia so stupňom polynómu 2, teda kvadratická. Funkcia so stupňom 3 síce mala vo väčšine prípadov lepší koeficient determinácie a jej krivka najlepšie obsiahla namerané hodnoty, vývoj jej krivky ale mal pri neznámych hodnotách často rapidne klesanie alebo stúpanie, ktoré aktuálne známe hodnoty fluorescencií pri pH nijako neindikovali.

Keďže sme mali k dispozícii aj údaje o hustote analyzovaných močov, rozhodli sme sa ich využiť a pozrieť sa na to, či ich zohľadnenie ovplyvní krivky a ako. Preto sme si určili konštanty, ktoré je možné vidieť v treťom stĺpci tabuľky č. 1. Týmito konštantami sme vynásobili namerané fluorescencie a opäť znázornili vzťah polynomickeou funkciou stupňa 2. Ako možno pozorovať z obr. č. 2, použitie konštant hustoty pri niektorých hodnotách vlnovej dĺžky spôsobilo väčšiu podobnosť kriviek niektorých močov. Na obrázku 2 je to ilustrované pri vlnovej dĺžke 280 nm vzorky 2 a 4, pri vlnovej dĺžke 340 nm sú to vzorky 1, 2 a 3.

Skúmaním grafov tohto typu sme prišli na to, že fluorescencia moču môže so zmenou pH nadobúdať pri každej vlnovej dĺžke dve varianty vývoja hodnôt. V intervaloch vlnovej dĺžky od 250 do 315 a od 375 do 400 nanometrov mali podobné krivky vzorka 1 so vzorkou 3 a vzorka 2 so vzorkou 4. Z pohľadu na tabuľku č. 1 vidíme, že vzorky 1 a 3 sú od dieťaťa, zatiaľ

čo 2 a 4 sú človeka dospelého. To je dôležité zistenie, ktoré naznačuje, že fluorescencia sa v závislosti od pH mení aj podľa toho, aký typ moču to je.

V intervaloch vlnovej dĺžky od 315 do 375 a od 400 do 545 nm sú zväčša podobné krivky troch močov a jeden sa líšil. Krivku ovplyvnila aj jedna hodnota, ktorá sa vymykala tým ostatným. Mohlo to byť aj z dôvodu chybovosti meracieho stroja. Preto sme pre tieto vlnové dĺžky spriemerovali všetky štyri funkcie a predpokladali sme, že v týchto rozpätiach majú obidva typy (moč detského aj dospelého jedinca) rovnaký vývoj kriviek.



Obr. 2 Porovnanie grafov funkcií bez a s použitím konštant hustoty pri vlnových dĺžkach 280 a 340

Normalizáciu funkcií do jednej sme vykonali jednoduchým spriemerovaním podobných funkcií pre každý desiaty krok vlnovej dĺžky (pri krokovani 0,5 teda od 255, 260, 265 až do 545 nm). V rozpätiach od 250 do 315 a od 375 do 400 nm sme spriemerovali samostatne vzorky detí a dospelých, vo zvyšku sme urobili priemer funkcií všetkých močov. Takýmto spôsobom sme získali funkcie, ktoré sme využili pri implementácii softvérového normalizátora.

B. Softvérový pH normalizátor

Vyššie opísaným postupom sme získali dva typy funkcií (pre dospelý alebo detský moč) pre 59 vlnových dĺžok. Tieto funkcie využíva pH normalizátor pri normalizácii konkrétnej hodnoty fluorescence. Postup pri normalizácii hodnoty pre konkrétnu vlnovú dĺžku je nasledovný:

- načítanie hodnoty fluorescence originálneho merania pri konkrétnej vlnovej dĺžke,
- načítanie funkcie pre analyzovaný typ moču a konkrétnu vlnovú dĺžku,
- získanie rozdielu medzi nameranou hodnotou fluorescence a hodnotou, ktorá sa vypočíta pri dosadení pH merania do funkcie vlnovej dĺžky,
- rozdiel získaný v predošlom kroku označuje posun, o ktorý je potrebné funkciu posunúť vertikálne (na osi y),
- získanie upravenej funkcie vlnovej dĺžky pomocou vypočítaného posunu,
- dosadenie cieľového pH do upravenej funkcie získanej v predošlom kroku a vypočítanie normalizovanej hodnoty fluorescence pre cieľové pH.

Pri ostatných vlnových dĺžkach (pre ktoré nie sú vyjadrené funkcie) aplikácia dopočíta hodnotu interpoláciou.

IV. VÝSLEDKY A VYHODNOTENIE

Vytvoreniu softvérového pH normalizátora predchádzalo nájdenie vzťahu medzi fluorescenciou a pH. Analýzou dát sme zistili, že závislosť fluorescence od pH sa pri rôznych vlnových dĺžkach líši. Z tohto dôvodu sme vzťah medzi fluorescenciou a pH skúmali pri konkrétnych vlnových dĺžkach. Vzťah sme skúšali vyjadriť lineárnou funkciou, funkciou polynomicou 2. stupňa (kvadratickou), polynomicou 3. stupňa a logaritmickou funkciou. Zistili sme, že daný vzťah najlepšie vyjadruje kvadratická funkcia. Skúmaním kriviek konkrétnych vzoriek moču sme prišli k poznatku, že pri vlnových dĺžkach od 250 do 315 nm a od 375 do 400 nm sa závislosť medzi pH a fluorescenciou líši pre moč dospelých a detí. Vo zvyšných vlnových dĺžkach je závislosť podobná pre obidva typy moču. Aby teda bola možná

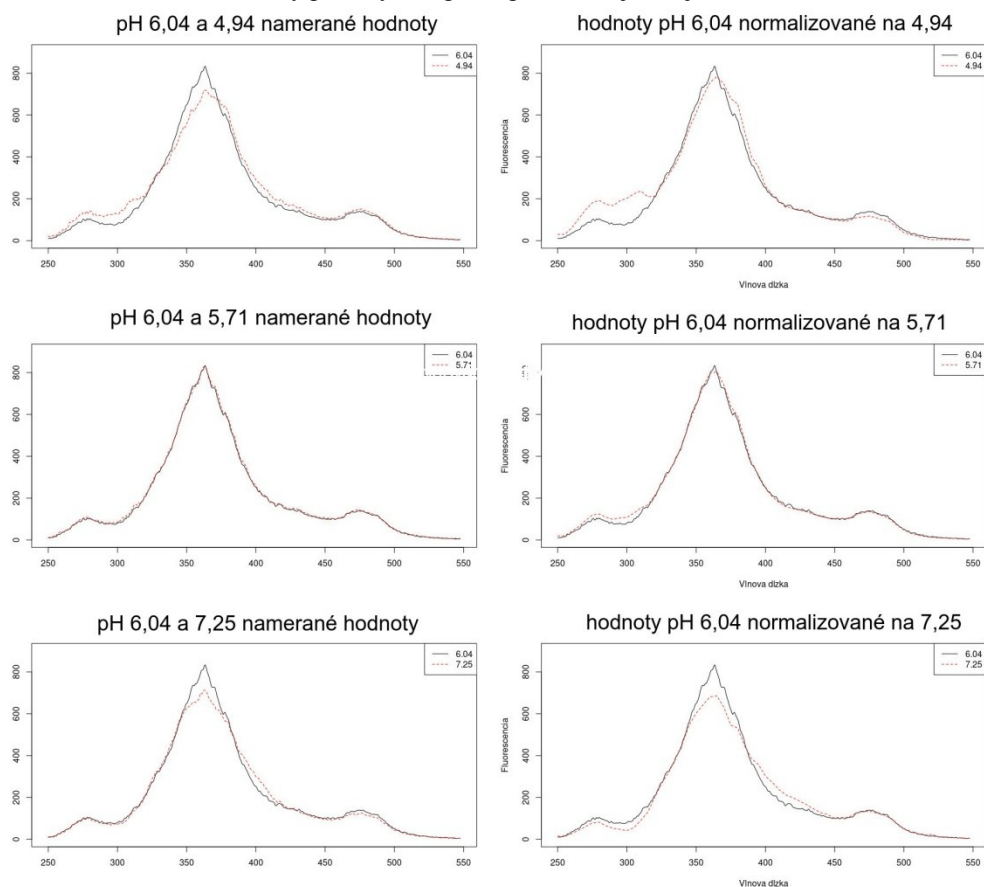
normalizácia, pre každú piatu vlnovú dĺžku sme vyjadrili dve funkcie normalizácie, pre detský a dospelý moč samostatne (v intervaloch, kde boli krivky všetkých vzoriek moču podobné, to bola len jedna funkcia).

Výsledná aplikácia pH normalizátora, dostáva ako používateľský vstup typ moču a na základe toho vypočíta normalizované hodnoty pre vlnové dĺžky s vyjadrenými funkciami vzťahu. Pri ostatných vlnových dĺžkach, ktoré vzťah nemajú vyjadrený, je hodnota dopočítaná lineárnou interpoláciou.

Výstup po spracovaní aplikáciou pH normalizátora je vygenerovaný skript v jazyku R, ktorý po spustení uloží normalizované hodnoty vo formáte CSV (ako možnosť pre ďalšiu analýzu či vizualizáciu) a zobrazí graf normalizovaných hodnôt pri cieľovom pH v porovnaní s pH originálneho merania.

Testovanie softvérového pH normalizátora prebiehalo porovnávaním normalizovaných hodnôt a reálne nameraných hodnôt pri konkrétnych pH. Ukázalo sa, že pH normalizátor normalizoval hodnoty uspokojivo pri hodnotách pH od 4 do 8, teda zvyčajných hodnotách pH moču, pričom to platilo pre husté aj riedke vzorky moču. Normalizované krivky majú veľmi podobný priebeh ako skutočne namerané krivky. Niektoré príklady porovnania skutočných a normalizovaných kriviek sú zobrazené na obrázku č. 3.

Pri extrémnych hodnotách pH (menších ako 4 a väčších ako 10) vytvorený softvérový pH normalizátor použiteľný nie je. Na obrázku č. 4 je možné vidieť porovnania skutočných a normalizovaných kriviek pri extrémne nízkej a extrémne vysokej hodnote pH. Tieto krivky sa v niektorých intervaloch vlnovej dĺžky vôbec nepodobajú. Nepresnosť a nepoužiteľnosť pri týchto hodnotách pH ale nie je problém, keďže takéto hodnoty moču zvyčajne nenadobúda a o normalizáciu hodnôt na hodnoty pri takýchto pH experti nemajú záujem.



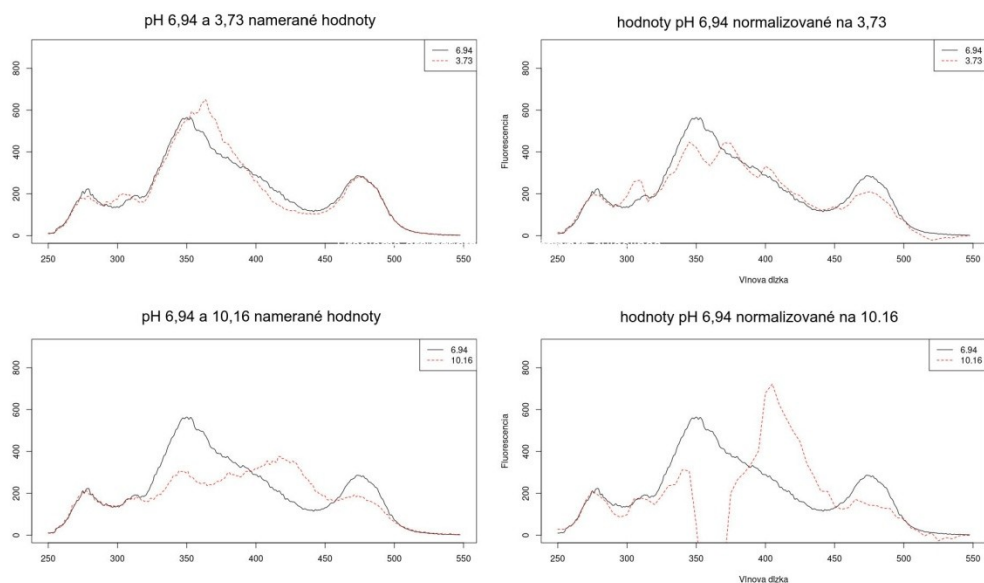
Obr. 3 Porovnanie reálnych a normalizovaných kriviek pri niektorých pH v rozmedzí od 4 do 8

Je potrebné poznamenať, že k dispozícii na analýzu boli 4 vzorky moču a pri interpretácii výsledkov treba zohľadniť, že:

- vzorky boli veľmi variabilné - napríklad vzorka 2 bola extrémne riedka, vzorky 1 a 3 boli od detí, vzorky 2 a 4 od dospelých ľudí,
- vzorky močov neboli štandardizované - zvyčajne sa pri fluorescenčnej analýze používa ranný moč, pre tento experiment však boli použité aj vzorky moču odobrané náhodne, ,

- merací stroj má určitú chybovosť, ktorú nie je možné ovplyvniť.

Vyššie spomenuté faktory mohli mať vplyv na vyjadrené funkcie vzťahov a je možné, že ak by sa použil štandardizovaný moč, presnosť vyjadrených funkcií vzťahov, a tým aj normalizátora by bola vyššia. Vyššiu presnosť normalizácie by pomohol dosiahnuť aj vyšší počet analyzovaných vzoriek, keďže by regresná analýza mohla použiť viac hodnôt a byť presnejšia.



Obr. 4 Porovnanie reálnych a normalizovaných kriviek pri niektorých extrémnych hodnotách pH

V. ZÁVER

Hlavným cieľom práce bolo vytvorenie softvérového pH normalizátora, čo sa nám podarilo splniť. K tomuto cieľu sme sa dopracovali postupnými krokmi. Rôznymi spôsobmi vizualizácie dát fluorescenčných meraní a ich analýzou sme najprv hľadali závislosť medzi pH a intenzitou fluorescencie. Skúmaním vzťahu pri konkrétnych vlnových dĺžkach sme zistili, že závislosť medzi pH a intenzitou fluorescencie nie je vždy rovnaká, ale mení sa, a pri rôznych vlnových dĺžkach je iná. Pri vybraných vlnových dĺžkach sme vzťah vyjadrili lineárnou funkciou, funkciou polynomicou 2. stupňa (kvadratickou), polynomicou 3. stupňa a logaritmickou funkciou. Zistili sme, že daný vzťah najlepšie vyjadruje kvadratická funkcia.

Skúmaním funkcií sme zistili, že v istých intervaloch sú podobné funkcie pre všetky vzorky močových, v niektorých sa vyskytujú dva typy podobných funkcií, v závislosti od toho, či je moč od dospelého človeka alebo dieťaťa. Konkrétne boli dve samostatné funkcie podľa typu moču vyjadrené v intervaloch vlnovej dĺžky od 250 do 315 nm a od 375 do 400.

Implementovaný softvérový pH normalizátor vykonával dva typy normalizácie v závislosti od parametra, ktorým bol typ moču (detský alebo dospelý). Normalizácia meraní z jedného pH na vybrané pH sa vykonávala použitím normalizačných funkcií vyjadrených pre konkrétne vlnové dĺžky, pre ostatné hodnoty vlnovej dĺžky sa dáta dopočítali lineárnou interpoláciou.

Testovaním normalizátora sme zistili, že normalizátor pracuje uspokojivo pre zvyčajné hodnoty pH v rozmedzí od 4 do 8. Pri extrémnych hodnotách mimo tohto rozmedzia normalizátor nepracuje správne, čo ale nevadí, keďže takéto hodnoty moču v skutočnosti nenadobúda.

Aj napriek tomu, že vytvorený softvérový pH normalizátor dosahuje uspokojivú normalizáciu meraní, stále existujú možnosti ako ho vylepšiť. V budúcnosti je možné zvýšiť presnosť normalizačných funkcií a tým aj celkovej presnosti normalizátora získaním viac vzoriek moču na analýzu.

ZOZNAM LITERATÚRY

- [1] M.J.P. Leiner, M.I.R. Hubmann, O. S. Wolfbeis, "The total fluorescence of human urine" in *Analytica Chimica Acta* 198 (1987), s. 13–23. issn: 0003-2670
- [2] L. Lichardusová a kol. "The factors influencing direct spectral fluorimetry of some urine metabolites." in *Prague medical report* 111 4 (2010),s. 272–8.
- [3] K. Dubayová a kol., "A Novel Way to Monitor Urine Concentration: Fluorescent Concentration Matrices" in *Journal of clinical and diagnostic research: JCDR* 9 (mar. 2015).

Automatizácia výpočtov sekundárnych častíc kozmického žiarenia v softvéri Corsika

¹Ivan BERNÁT, ²Pavol BOBÍK, ³Ján GENČI

¹Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

²Ústav experimentálnej fyziky, Slovenská akadémia vied, Slovenská republika

³Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹ivan.bernat@student.tuke.sk, ²bobik@saske.sk, ³jan.genci@tuke.sk

Abstrakt – Táto práca sa zaoberá vytvorením aplikácie AutoSimVis na automatizáciu tvorby simulácií a vizualizácií spŕšok sekundárneho kozmického žiarenia. Na vytváranie simulácií bol použitý balík CORSIKA [1]. Aplikácia ponúka používateľom možnosť zadať žiadosti na simuláciu spŕšok prostredníctvom webového formulára. Po dokončení simulácie sa získané dáta roztriedia a následne sa vytvoria viaceré vizualizácie zo získaných dát, ktoré budú dostupné na stránke pre používateľa na prezeranie a stiahnutie.

Kľúčové slová – kozmické žiarenie, simulačný nástroj CORSIKA, vizualizácie, Python

I. ÚVOD

Spŕšky sekundárneho kozmického žiarenia sú tvorené primárnym kozmickým žiarením s vysokými energiami, ktoré sa skladá z vysoko energetických častíc, primárne protónov a alfa častí letiacich vesmírom. Ak takéto primárne častice kozmického žiarenia narazia do zemskej atmosféry, interagujú s jadrami atómov molekúl atmosféry a vznikajú takzvané časticové spŕšky. Tie sa následne šíria atmosférou a dopadajú na zemský povrch [2]. Pôvod primárnych častíc kozmického žiarenia tvoriacich spŕšky dopadajúce na povrch Zeme zostáva zatiaľ neznámy. Domnievame sa však, že ich zdrojmi by mohli byť aktívne jadrá galaxií.

Analyzovať dáta takýchto spŕšok vyžaduje podrobné modelovanie vzniku spŕšok, od zrážky primárnych častíc s atmosférou až po dopad častíc spŕšky na zemský povrch. To môžeme dosiahnuť pomocou Monte Carlo simulácií, zameraných na častice vysokých energií a elektromagnetické interakcie. Preto vznikli rôzne simulačné programy, ktoré simulujú vznik a vývoj extenzívnych časticových spŕšok v atmosfére, a ich následný dopad na zemský povrch. Jedným z takýchto programov je aj balík CORSIKA.

II. CORSIKA

Práca s balíkom CORSIKA, vykonanie simulácií spŕšok a spracovanie výstupných dát zo simulácií je komplexný proces vyžadujúci znalosť samotného balíka ako aj dostatočne výkonný hardvér. Z toho dôvodu vznikla aplikácia *AutoSimVis*, ktorej úlohou je automatizovať proces spúšťania balíka CORSIKA, spracovania výstupného binárneho súboru zo simulácie a následná vizualizácia týchto dát do podoby prospešnej pre používateľov. Samotná aplikácia je súčasťou COR modelu a je prístupná na stránke <http://cor.crmodels.org/>.

A. Príprava balíka CORSIKA

Na spustenie samotného balíka je nutné disponovať dátovým súborom *DAT00000X*, ktorý je nutné vygenerovať procesom simulácie. Na vykonanie simulácie je najprv potrebné skompilovať balík CORSIKA, použitím programu *coconut*, ktorý je súčasťou samotného balíka.

Pri jeho spustení program ponúkne nastavenia parametrov pre simuláciu, akými sú výber 32 alebo 64 bitového módu, typ hadrónového modelu - *DPMJET*, *EPOS*, *NEXUS*, výber nízko energetického modelu - *GHEISHA*, *FLUKA* [3], detektorovú geometriu, kde sa vykoná výber medzi horizontálnym plochým poľom detektorov, neplochou detektorovou geometriou a vertikálnou detektorovou geometriou [4].

Po vygenerovaní súboru *DAT00000X* je možné spustiť samotnú simuláciu zavolaním balíka príkazom *"corsika75600Linux_DPM <input"*, kde *input* predstavuje názov vstupného súboru, *input*.

Každý hadrónový model má vlastný súbor vo formáte *all-inputs-meno_modelu*. Teda dostupné budú všetky kombinácie modelov, ktoré balík ponúka, bez nutnosti dodatočnej manipulácie s konfiguráciou samotného balíka. V nich je možné nastaviť vstupné parametre ako množstvo simulovaných spŕšok, typ primárnych častíc a podobne.

B. STANDARD & COMPACT formát

Balík CORSIKA po ukončení simulačného procesu dodá na výstupe binárny súbor obsahujúci všetky informácie z vytvorenej simulácie spŕšky, akými sú energie častíc, typy častíc, smer príchodu častíc či informácie o Čerenkovom žiarení. Takýto binárny súbor nie je priamo čitateľný a preto je nutné ho konvertovať do čitateľného formátu.

CORSIKA môže ale vygenerovať výstupný súbor v dvoch formátoch, *STANDARD* a *COMPACT*. V prípade *STANDARD* formátu je možné použiť program *corsikaread.f* balíka CORSIKA, ktorého výstupom je čitateľný textový súbor *fort.7*. V ňom sú dáta zapísané ako jeden súvislý blok, ktorý rozdelíme, pomocou tabuliek 7 až 13 v oficiálnej dokumentácii balíka CORSIKA [1], do blokov o dĺžke 273 záznamov.

Prvým súborom bude *Run header sub-block*, kde sa nachádzajú všeobecné informácie o danom behu, tj. číslo behu, dátum simulácie a dodatočné dáta. Ďalším je súbor *Event header sub-block* obsahujúci dáta o jednotlivých *event-och*.

Hodnoty jednotlivých častíc sú rozdelené do príslušných súborov *0_gamma*, *0_eplus*, *0_emin*, *0_muplus*, *0_mumin*, *0_muplusAI* a *0_muminAI*, kde druh častice je určený podľa ich identifikačných čísel. Taktiež sa vytvoria súbory *blocks_X*, kde *X* predstavuje poradové číslo bloku, obsahujúce dáta o jednotlivých blokoch 1.

Časticový sub-block: (až po 39 častíc, 7 hodnôt pre každú)	
Poradie hodnoty	Obsah hodnoty (ako reálne čísla R*4)
$7 * (n - 1) + 1$	opis častice zakódovaný ako: id častice * 1000 + hadrónová generácia * 10 + počet observačných úrovní
$7 * (n - 1) + 2$	px, hybnosť v x-ovom smere v GeV/c
$7 * (n - 1) + 3$	py, hybnosť v y-ovom smere v GeV/c
$7 * (n - 1) + 4$	pz, hybnosť v z-ovom smere v GeV/c
$7 * (n - 1) + 5$	pozícia x-ovej súradnice v cm
$7 * (n - 1) + 6$	pozícia y-ovej súradnice v cm
$7 * (n - 1) + 7$	čas t od prvej interakcie/vstupu do atmosféry v n sekundách
	pre n = 1 : : : 39 ak nie je posledný blok naplnený, pridajú sa nuly po koniec

Tabuľka 1
 Štruktúra časticového bloku. [5]

V prípade *COMPACT* formátu, balík neponúka žiadnu možnosť konverzie binárneho súboru do čitateľnej podoby, ako tomu je pri *STANDARD* formáte, keďže jeho dáta sú zapísané v skrátenej forme.

Prvou zmenou vo formáte je veľkosť jednotlivých dátových blokov, ktorá nie je pevná, ako tomu bolo pri normálnom formáte. Zmenu podstúpila aj štruktúra súboru, ktorá kompletne vynechané koncové bloky. Treťou zmenou sú hlavičky súboru, kde iba prvá má všetkých 273 záznamov a je označená sekvenciou znakov *EVTH*. Hlavičky ostatných blokov sú skrátene, začínajú sekvenciou *EVHW* a obsahujú iba prvých 12 parametrov, ktoré sú opísané v tabuľke 2, prevzatej z oficiálnej dokumentácie balíka CORSIKA [1].

Tieto dátové bloky majú rôzne dĺžky, kde prvým záznamom je 4-bajtový *integer* definujúci dĺžku jeho dátového bloku. Maximálna dĺžka jedného dátového bloku je $273 + 1$ štvorbajtových slov. Koncové nuly sú v každom dátovom bloku kompletne vynechané.

C. Konverzia COMPACT formátu

Na obrázku 1 môžeme vidieť konverziu *COMPACT* formátu do čitateľnej formy, ktorú dostaneme použitím programu napísaného v jazyku C, ktorý bol vytvorený počas tvorby tejto práce. Program otvorí vstupný binárny súbor a vyčíta z neho všetky dáta, ktoré usporiada po štyroch bajtoch. Tieto štvorice sú následne konvertované do dátových typov *float*, *integer* a *hexadecimal*, ktoré program zapíše v podobe stĺpcov do výstupného súboru.

Hlavička udalostového sub-bloku:	
Poradie hodnoty	Obsah hodnoty (ako reálne čísla R*4)
1	'EVHW'
2	číslo udalosti
3	identifikačné číslo častice
4	celková energia v GeV
5	štartovacia nadmorská výška v g/cm ²
6	číslo prvého cieľa
7	Z súradnica (výška) prvého styku v cm
8	hybnosť px v smere x súradnice v GeV/c
9	hybnosť py v smere y súradnice v GeV/c
10	hybnosť pz v smere z súradnice v GeV/c
11	zenitový uhol θ v radiánoch
12	azimutový uhol ϕ v radiánoch

Tabuľka 2
 Štruktúra hlavičkového bloku pri použití Compact formátu. [5]

```

16 |          0.0000 |          0 | 0 0 0 0 | .....
17 |         -2.6700 | -1070931640 | 48 e1 2a c0 | H4*A
18 |        10000.0000 | 1176256512 | 0 40 1c 46 | @*F
19 |        10000.0000 | 1176256512 | 0 40 1c 46 | @*F
20 |          0.0000 |          0 | 0 0 0 0 | .....
21 |          0.0000 |          0 | 0 0 0 0 | .....
22 |         120.0000 | 1123024896 | 0 0 f0 42 | ..0B
23 |         120.0000 | 1123024896 | 0 0 f0 42 | ..0B
24 |         120.0000 | 1123024896 | 0 0 f0 42 | ..0B
25 |         120.0000 | 1123024896 | 0 0 f0 42 | ..0B
26 |    637131520.0000 | 1310189420 | 6c e7 17 4e | lc*N
27 |     6000000.0000 | 1225948160 | 0 7c 12 49 | *+I
28 |    20000000.0000 | 1240736768 | 0 24 f4 49 | *S0I
29 |    -1365.2404 | -995448911 | b1 a7 aa c4 | ±S*A
30 |          0.0000 |          0 | 0 0 0 0 | .....
31 |          0.0458 | 1027317513 | 9 9f 3b 3d | ;
    
```

Obr. 1 Konvertované dáta COMPACT súboru v čitateľnej podobe.

Tieto stĺpce obsahujú zľava poradové čísla, hodnoty v dátovom type *float*, hodnoty v dátovom type *integer*, ASCII hodnoty jednotlivých znakov a binárnu formu dát.

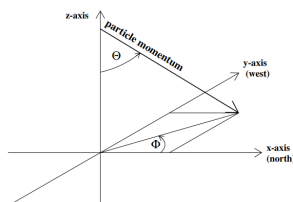
Rovnako ako tomu je pri spracovaní binárneho súboru STANDARD formátu, takto spracovaný výstupný súbor je následne využiteľný na triedenie dát do hlavičkových a *block_X* súborov, ako aj súborov jednotlivých typov častíc.

III. VIZUALIZÁCIA DÁT SIMULÁCIE SPŔŠOK

Vytvorené vizualizácie sa sústredili viac na zobrazenie údajov, ktoré sú primárne zamerané na odborníkov, napríklad fyzikov v danej sfére výskumu. Vytvorené boli tri nové typy vizualizácií použitím jazyka Python a jeho knižníc Matplotlib verzie 1.5.0, ktorá sa stará o samotnú vizualizáciu dát, a NumPy verzie 1.11.2, zodpovednej za zjednodušenie manipulácie s dátami.

A. Vizualizácie distribúcie uhlov častíc prichádzajúcich na zemský povrch

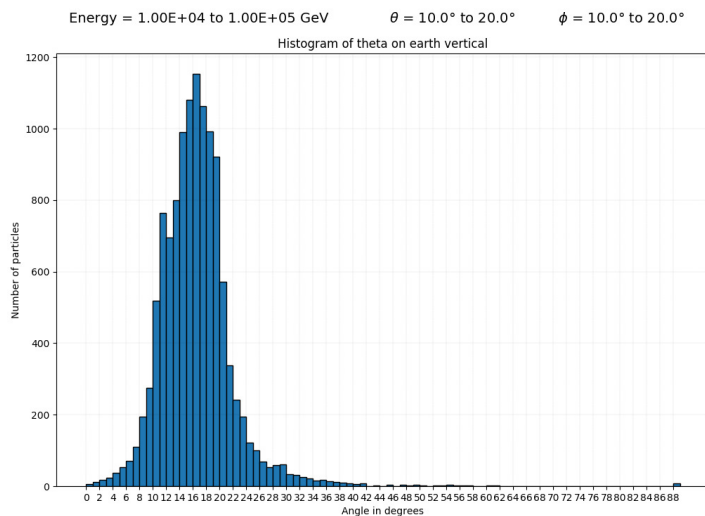
Prvou vizualizáciou sú histogramy distribúcie uhlov častíc prichádzajúcich na zemský povrch. Pri vytvorení časticovej spŕšky sa častice tejto spŕšky rozptýlia a dopadajú na povrch Zeme. Keď sa pozrieme na obrázok 2, tak si môžeme všimnúť, že uhol θ sa nachádza medzi hybnosťou prichádzajúcej častice a Z-ovou osou, ktorá predstavuje smer nahor, tj. nahor od zemského povrchu do atmosféry. Tento uhol θ predstavuje uhol dopadajúcich častíc od osi Z súradnej sústavy, ktorý je vypočítaný vzorcom 1.



Obr. 2 Koordinačný systém použitý v balíku CORSIKA.[6]

$$\theta_{earth} = \arctan \sqrt{\frac{p_x^2 + p_y^2}{p_z}} \quad (1)$$

Na obrázku 3 môžeme vidieť príklad už spomenutej vizualizácie, ktorá ukazuje distribúciu uhlov. Vizualizácia znázorňuje uhly dopadu jednotlivých častíc daného typu na zemský povrch vo forme histogramu.



Obr. 3 Histogram uhlov príchodu častíc na povrch Zeme pre záporné mióny.

V ňom X-ová os predstavuje jednotlivé uhly, pod ktorými častice dopadli, a Y-ová os predstavuje počet častíc prichádzajúcich z daných uhlov. Tento obrázok ilustruje ako vyzerá dopad častíc záporných miónov. Pri vstupnom zenitovom uhle θ v rozmedzí 10 až 20 stupňov, a vstupnom azimutovom uhle ϕ , taktiež v rozmedzí 10 až 20 stupňov, môžeme vidieť, že majoritná časť častíc dopadá hlavne v tomto rozmedzí. Niektoré sa môžu počas ich letu cez atmosféru mierne odchyliť od ich pôvodného smeru a dopadanú s malými odchyľkami. Tieto častice sa nachádzajú tesne pred 10 stupňami alebo za 20 stupňami.

V ojedinelých prípadoch môžu častice dopadať aj v úplne odľahlých stupňoch, ako môžeme vidieť na prípadoch pri 88. stupni. Keďže táto vizualizácia bola vytvorená pri nízkych energiách, pre rozsah energií $1.E04$ až $1.E05$ GeV, s 10 simulovanými spŕškami, tak celkový počet častíc miónov je približne iba 12 tisíc. Tie sú primárne rozložené v rozmedzí vstupných 10 až 20 stupňov, takže jednotlivé stupne dosahujú v tomto histograme početnosť iba približne 1200 častíc.

B. Vizualizácie grafov spektier intenzít vstupných energií

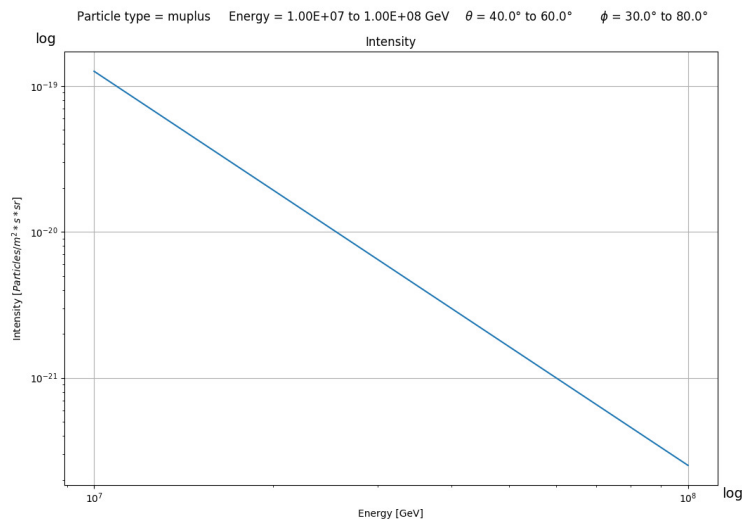
Ďalším grafom je znázornenie spektier intenzít primárnych častíc. Do našej atmosféry vchádzajú častice kozmického žiarenia z rôznymi energiami. Ich početnosť je pre rôzne energie opisovaná ich spektrom. Tvar tohto spektra je určený dvoma hlavnými vplyvmi. Vplyvom magnetosféry Zeme a vplyvom heliosféry. Na hranici heliosféry je spektrum kozmického žiarenia izotropné a v čase nemenné. Pri prechode častíc heliosférou je modulované, čo znamená, že len určitá časť kozmického žiarenia nachádzajúca sa na hranici heliosféry sa aj dostane do vnútornej heliosféry a postupne k orbite Zeme. Modulácia je rôzna na rozličných energiách, kde sa navyše mení aj so slnečným cyklom. Následne kozmické žiarenie vstupuje do magnetosféry, ktorá taktiež ovplyvní jeho spektrum.

Názorná ukážka tohto vstupného spektra je vizualizácia 4, v ktorej je zobrazená vizualizácia spektier intenzít kozmického žiarenia s rozsahom energii od $1.E07$ až $1.E08$ GeV. Simulovaných bolo 10 spŕšok so zenitovým uhlom θ nachádzajúcim sa v rozmedzí od 40 po 60 stupňov, a azimutovom uhle ϕ v rozmedzí od 30 až po 80 stupňov. Na X-ovej osi znázorňujeme tieto vstupné energie, zatiaľ čo na Y-ovej osi sú zobrazené vypočítané intenzity prislúchajúce k daným energiám.

Jednotlivé intenzity sú vypočítavané pomocou vzorca 2.

$$intensity = E^{slope} \times c \quad (2)$$

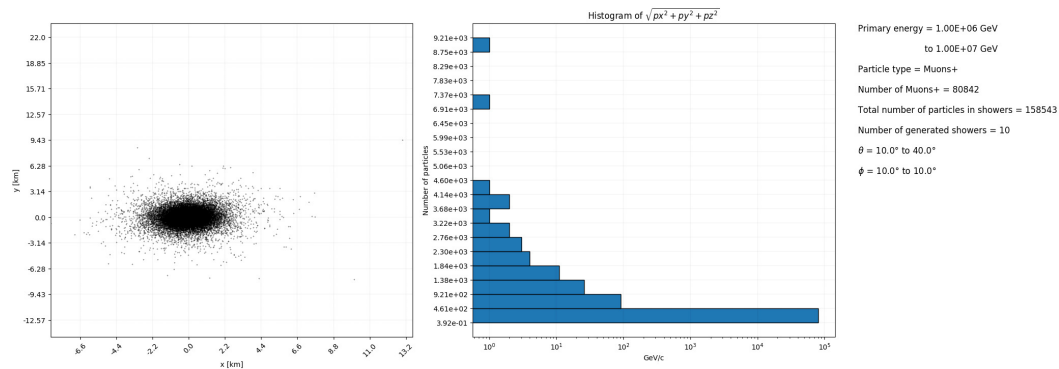
kde E sú energie primárneho kozmického žiarenia, E_{slope} je mocninový sklon spektra, ktorý je pri tejto vizualizácii $-2,7$, a c je konštanta, ktorá je nastaviteľná podľa potrieb, ale v tomto prípade bola nastavená na hodnotu 1.



Obr. 4 Spektrum intenzít energií pre kladné mióny pri sklone spektra -2,7.

C. Vizualizácie grafov zobrazujúcich prehľad dôležitých informácií

Posledný typ vizualizácií je celková, ukázková vizualizácia obsahujúca dva typy variantov vizualizácií, tj. vizualizáciu elíps podľa hodnôt x a y , a histogram hybnosti pre kladne nabité mióny. Obe vizualizácie sú zoradené v jednej línii, ktorá je ukončená zoznamom dodatočných dát pochádzajúcich alebo získaných zo samotného procesu simulácie a vizualizácie dát. Oproti ostatným vizualizáciám, tento variant taktiež obsahuje celkový počet častíc daného typu, tj. miónov, spolu s celkovým simulovaným počtom častíc všetkých typov. Vďaka tomu má používateľ rýchlu predstavu o veľkosti zastúpenia daného typu častíc z celkového počtu. Ukázkový výstup tretieho variantu je možné vidieť na 5.



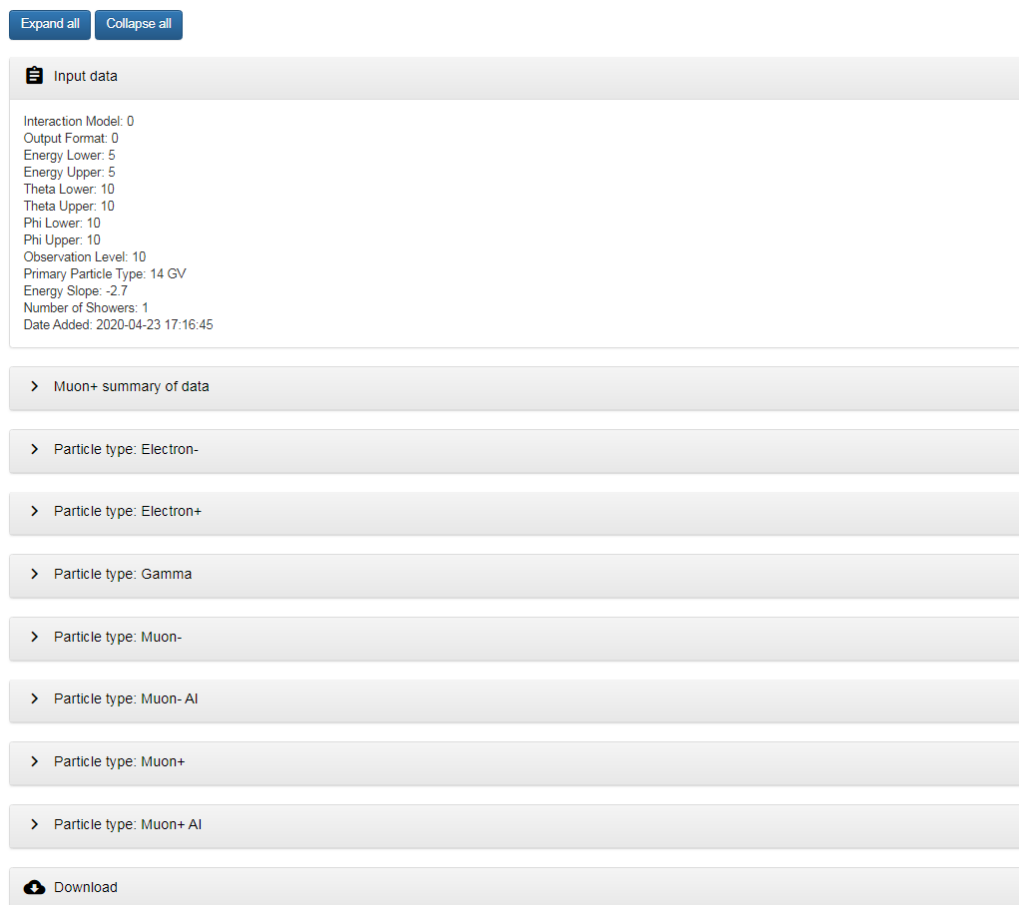
Obr. 5 Celková vizualizácia obsahujúca dodatočné dáta pre kladné mióny.

IV. AUTOMATIZÁCIA PROSTREDNÍCTVOM WEBOVEJ STRÁNKY

AutoSimVis aplikácia má byť jednoducho použiteľná pre akéhokoľvek používateľa, bez ohľadu na jeho fyzikálne alebo informatické znalosti. Preto je nasadená na serveri a používateľ má prístup ku formuláru na zadávanie nových simulácií, a vizualizácií prostredníctvom webového rozhrania. V webovom rozhraní stránky <http://cor.crmodels.org/> môže používateľ zadať rôzne vstupné parametre ich žiadosť na simuláciu a tá bude následne realizovaná. Dĺžka výpočtu závisí od rôznych premenných, kde niektoré z nich môže používateľ ovplyvniť, ale niektoré sú už dopredu preddefinované a používateľ ich nemôže zmeniť.

Po zadaní vstupných parametrov a ich odoslaní na server, sa na serveri spustí simulácia použitím automatizačného skriptu, ktorý je využívaný už existujúcim výpočtovým modelom COR webovej stránky. Ten ho zaradí do poradovníka a keď sa dostane žiadosť na rad, tak bude žiadosť spracovaná, tj. vykoná sa proces simulácia a následná vizualizácia výstupných dát. Po ich vypracovaní budú dostupné na webovej stránke v časti *CorsikaCals*. V nej všetci používatelia

budú mať prístup ku výstupu z vykonania danej žiadosti cez ktorú sa dostanú na stránku žiadosti, kde budú mať zobrazené jej výsledky. Stránka žiadosti obsahuje parametre simulácie, všetky výstupné vizualizácie, ktoré si používatelia môžu prezrieť, ako aj možnosť stiahnutia lokálnej kópie výstupu žiadosti vo formáte *zip*. Takýto komprimovaný súbor obsahuje roztriedené dáta, ktoré vznikli počas simulácie, a všetky výstupné vizualizácie vo formáte *png*. Príklad vykonanej žiadosti môžeme vidieť na obrázku 6.



Obr. 6 Detail vykonanej žiadosti

V. ZÁVER

Cieľom práce bolo ponúknuť používateľom spôsob jednoduchšej a efektívnej možnosti tvorby simulácií, a vizualizácií spršok sekundárneho kozmického žiarenia. To bolo dosiahnuté použitím balíka CORSIKA na vytváranie simulácií, webového formulára na zadanie žiadostí o simulácie a umiestnením aplikácie AutoSimVis na server, kde sa aplikácia postará o spúšťanie simulácií, spracovanie výstupných dát a ich zobrazenie v podobe viacerých vizualizácií znázorňujúcich rôzne fyzikálne parametre.

LITERATÚRA

- [1] I. o. T. Institute for Nuclear Physics Karlsruhe. Corsika: an air shower simulation program. [Online]. Available: <https://www.ikp.kit.edu/corsika/>
- [2] F. Blaschke, "Analýza korelovaných spršok kozmického záření," Master's thesis, Slezská univerzita v Opavě, 2009.
- [3] P. A. Observatory. (2005) Fluka: Montecarlo simulation package. [Online]. Available: <http://www.fluka.org/fluka.php>
- [4] J. Vasilko, "Využitie houghovej transformácie pri rozpoznávaní spršok tvorených časticami ultravysokých energií," Master's thesis, Technická univerzita v Košiciach, 4 2015.
- [5] I. o. T. Institute for Nuclear Physics Karlsruhe. (2019) Corsika user guide. [Online]. Available: <https://web.ikp.kit.edu/corsika/usersguide/usersguide.pdf>
- [6] ——. (2004) Corsika guide. [Online]. Available: https://web.ikp.kit.edu/corsika/physics_description/corsika_phys.pdf

Refaktorizácia kódov pre výpočet trajektórii častíc kozmického žiarenia v magnetosfére Zeme

¹Martin VAŠKO, ²Pavol BOBÍK, ³Ján GENČI

¹Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

²Ústav experimentálnej fyziky, Slovenská akadémia vied, Slovenská republika

³Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹martin.vasko.5@student.tuke.sk, ²bobik@saske.sk, ³jan.genci@tuke.sk

Abstrakt – Práca pojednáva o refaktorizácii programu Ústavu experimentálnej fyziky Slovenskej akadémie vied (ÚEF SAV), zodpovedného za simuláciu trajektórií častíc kozmického žiarenia v magnetosfére Zeme, z programovacieho jazyka Fortran do jazyka C. Pre overenie korektnosti výstupov boli vykonané testy trasovania trajektórií v uniformnom magnetickom poli a metódou *backward-forward tracing*. Taktiež bolo implementované paralelné vykonávanie rozhraním Open Multi-Processing (OpenMP) pre zvýšenie efektivity programu.

Kľúčové slová – kozmické žiarenie, magnetosféra, refaktorizácia, Fortran, C, OpenMP

I. ÚVOD

Hlavným predmetom tejto práce sú vysoko-energeticky nabité častice letiace vesmírom relativistickými rýchlosťami. Tieto častice sú označované názvom *kozmicke žiarenie*, medzi ktoré patria viaceré typy častíc, ale prevažne je kozmické žiarenie tvorené protónmi a alfa časticami [1].

ÚEF SAV dlhodobo vyvíja interný softvér simulujúci trajektórie častíc kozmického žiarenia [2] [3], prichádzajúce izotropne do prostredia magnetosféry Zeme z medziplanetárneho priestoru. Častice sú po vstupe do magnetosféry ovplyvňované geomagnetickým poľom, ktoré môže ovplyvniť ich trajektórie. Kozmické žiarenia s energiami nad 100 GeV nebude ovplyvnené, ale častice s nižšími energiami budú mať ich trajektórie ovplyvnené tým viac, čím je ich energia nižšia [4]. Dôsledkom je zmena trajektórie, zachytenie v magnetosfére alebo ich dopad na povrch Zeme.

Tento softvér využíva modely interného geomagnetického poľa, International Geomagnetic Reference Field (IGRF), a externého geomagnetického poľa, Tsyganenko 04, pre simulovanie vplyvov magnetosféry Zeme. Na simulovanie trajektórií je použitý fyzikálny model zahrňujúci časť pre výpočet pohybu nabitých častíc v magnetickom poli, použitím Lorentzovej rovnice.

II. ANALÝZA LOGIKY FORTRAN ZDROJOVÉHO KÓDU

Existuje viacero verzií programu, ale na refaktorizáciu bola použitá verzia, ktorej základ bol dokončený v roku 2012, ako vychádza z názvu súbora, *T04_jul2012_arg_divInfil.f*.

A. Vstupný súbor - *infil*

Pre spustenie programu je nutné disponovať súborom *infil*, obsahujúcim vstupné hodnoty vo špecifickom formáte usporiadania. Tieto hodnoty budú pri spustení programu vyčítané zo súbora a poslúžia ako parametre simulácie. Príklad *infil* súbora môžeme vidieť vo výpise 1.

```
1 0.1000 -1. 2.8000
2 1.00 64.13 -21.93
3 64.13 -21.93
4 2013 2 16 47 12 00 00
5 100 1 1 0.10
6 3 1.52 6.80 1.10
7 0.20 0.27 0.15 0.13 0.24 0.27
8 -1.00
```

Kód 1 Ukážkový obsah súbora *infil*.

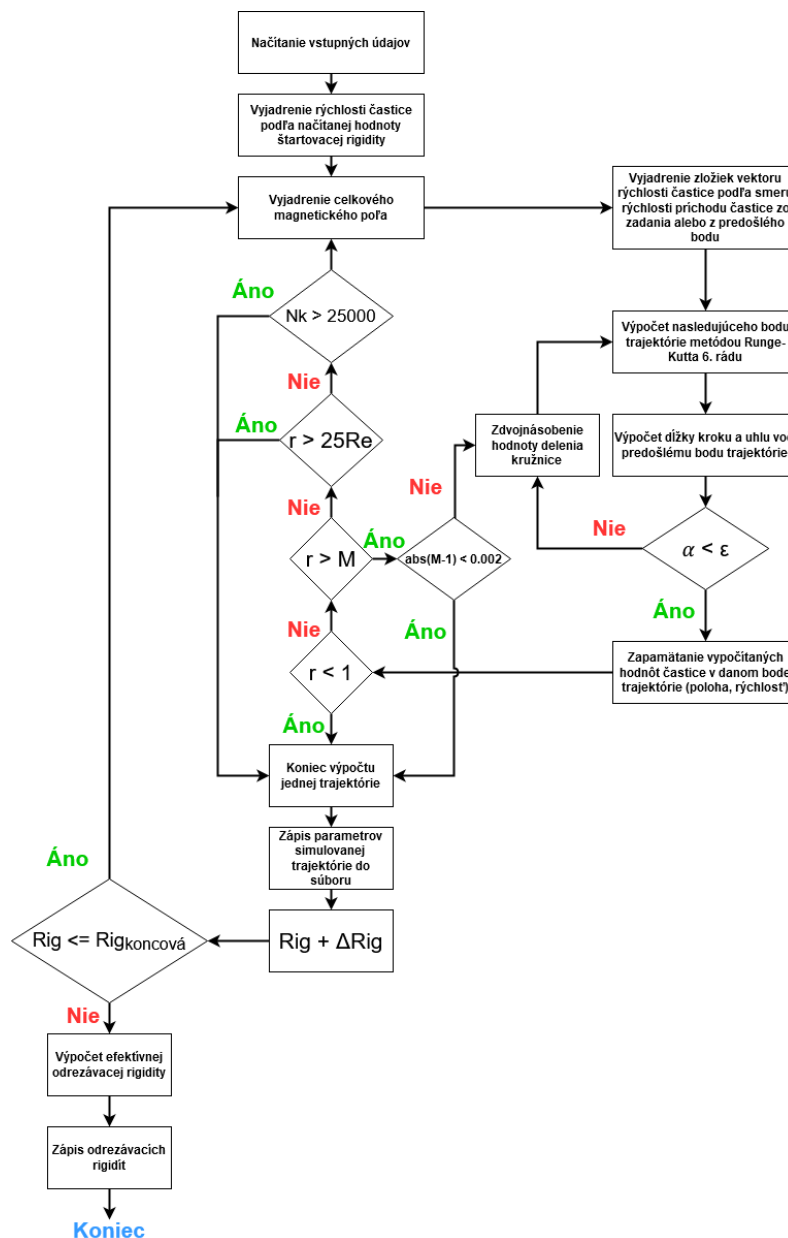
Na vysvetlenie jednotlivých hodnôt použijeme nasledujúci list, ktorý opíše ich význam:

- 1) riadok - štartovacia rigidita, hodnota určujúca druh častice, -1 pre protón, koncová rigidita
- 2) riadok - rádus v jednotkách R_e , geografická šírka a dĺžka počiatku trajektórie
- 3) riadok - dĺžka a šírka pre smer príchodu častice v geografickej sústave
- 4) riadok - rok, mesiac, deň, deň v roku, hodiny, minúty a sekundy
- 5) riadok - prvé čísla predstavujú parametre delenia trajektórie a posledné opisuje krok rigidity
- 6) riadok - hodnota Dst indexu v nanoTesla, hodnota $pdyn$ dynamického tlaku Slniečného vetra v nanoPascaloch, a intenzita y-ovej, a z-ovej zložky medziplanetárneho poľa
- 7) riadok - vstupné parametre W1 až W6 modelu externého geomagnetického poľa
- 8) riadok - -1.00 označuje koniec *infil* súbora

B. Vývojový diagram

Okrem vstupno/výstupných operácií a vykonania prípravných funkcií pre výpočet, je vykonávanie programu sústredné do základného cyklu programu, v ktorom sa iteratívnym spôsobom kalkulujú parametre častice s danou hodnotou rigidity, prichádzajúcej z určitého smeru. Po dosiahnutí medzných hodnôt sa cyklus preruší a výsledné hodnoty parametrov častice sa vpišu do výstupného súbora. Hodnota rigidity častice sa zvýši o hodnotu kroku a celý cyklus pokračuje pokiaľ hodnota rigidity nepresiahne hornú hranicu špecifikovanú vo vstupnom súbore.

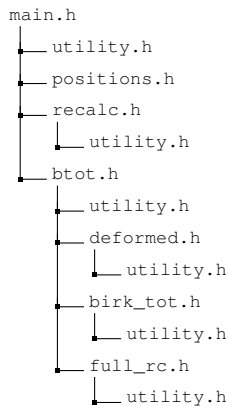
Vývojový diagram softvéru podrobnejšie opisuje tento cyklus na obrázku 1.



Obr. 1 Vývojový diagram zobrazujúci logiku softvéru.

III. ZMENA ŠTRUKTÚRY ZDROJOVÝCH KÓDOV

Zdrojový kód pôvodnej verzie v jazyku Fortran je takmer 5000 riadkový súbor, obsahujúci množstvo zastaraných funkcií, premenných a neplatných komentárov, ktoré sťažujú jeho rozširovanie a optimalizáciu. Z toho dôvodu bol kód logicky rozložený do viacerých zdrojových súborov, obsahujúcich funkcie súvisiace iba s danou oblasťou simulácie. Toto rozloženie môžeme vidieť na obrázku 2, zobrazujúcom štruktúru hlavičkových súborov.



Obr. 2 Štruktúra zdrojových kódov refaktorovanej verzie softvéru.

IV. IMPLEMENTÁCIA PARALELIZMU PROSTREDNÍTVOM OPENMP

Čas vykonania je významným prvkom pri vedeckých simuláciách, keď že čím viac presnejších dát je možné získať v čo najkratšom čase, tým efektívnejší pokrok nastáva v danej sfére výskumu. Preto je jedným z chcených vylepšení úprava programu do podoby, ktorá by podporovala zavedenie škálovateľného paralelizmu, čím by sa výkon softvéru značne zvýšil [5].

Pôvodná verzia programu obsahovala mnohé *goto* príkazy, ktoré bolo nutné odstrániť, keďže ich prítomnosť zvyšovala komplexnosť, zhoršovala čitateľnosť a sťažovala zavedenie paralelizmu. Po ich odstránení a ukončení samotnej refaktorizácie, bolo implementované rozhranie OpenMP, obsahujúce sústavu knižníc a procedúr na zavedenie súbežného vykonávania [6].

Vytvorenie procesov a začiatok paralelného vykonávania kalkulácií nastáva počas volania *#pragma*, ktorá slúži na dodatočné špecifikovanie žiadostí od kompilačného programu. Samotné volanie pre OpenMP je:

```
#pragma omp parallel for ordered schedule(dynamic, 1) reduction(+:nza) if (run_in_parallel)
```

Jeho význam si môžeme rozložiť nasledovne:

- *omp parallel for* - špecifikácia, že *for* cyklus pod volaním sa ma vykonať paralelne
- *ordered* - spôsobí sekvenčné vykonanie cyklu pre zachovanie poradia výstupných hodnôt
- *schedule(dynamic, 1)* - plánovač náhodne priradí jednu hodnotu danému vláknu
- *reduction(+:nza)* - každý proces vytvorí skrytú premennú a po ukončení cyklu sa získané hodnoty sčítajú, a uložia do už existujúcej premennej *nza*
- *if (run_in_parallel)* - ak podmienka nie je rovná *true*, tak cyklus nebude vykonaný paralelne

Implementáciou OpenMP bola dosiahnutá extrakcia vykonávania softvéru do paralelnej podoby, čo je dôležitý krok pre oddialenia sa od sekvenčného vykonávania a umožnenie paralelného vykonávania na viac-jadrových procesoroch [7].

V. TESTOVANIE KOREKTNOSTI VÝSTUPOV

Na overenie korektnosti refaktorizovanej verzie bolo nutné vykonať testy na overenie zhodnosti a fyzikálnej korektnosti výstupov.

A. Testovanie trasovaním trajektórie častice v uniformnom magnetickom poli

Prvé testovanie pozostávalo z trasovania trajektórie častice v uniformnom magnetickom poli po kruhovej trajektórii, čo umožnilo overenie vzniku odchýlky pri počte krokov $1e4$, $1e5$, $1e6$ a $1e7$.

To bolo dosiahnuté nastavením premenných *bx*, *by* a *bz*, predstavujúcich intenzity zložiek geomagnetického poľa pre polohu častice v priestore, na hodnoty 0, 50000 a 0 nanoTesla. Tento prepis hodnôt bol vykonaný za každou inštanciu volaní funkcií *btot()* a *geogsm()*. Tým bol

dosiahnutý stav, v ktorom sa častica pohybovala po presne definovaných súradniciach v rovine XZ, vytvárajúcich trajektóriu kružnice, ktorej veľkosť nebola väčšia ako veľkosť magnetosféry.

Touto zmenou sa dosiahlo neohraničené simulovanie pohybu častice v priestore, čo umožnilo overiť predpoklad prítomnosti odchýlky v prípade nekorektnej refaktorizácie, ktorá by viedla ku vychýleniu častice z kruhovej trajektórie v uniformnom magnetickom poli.

Okrem samotnej modifikácie, bol pre potreby testovania produkovaný dodatočný výstupný súbor, obsahujúci súradnice častice x a z , ktoré boli použité pre výpočet rádiusu kruhovej trajektórie častice pre kroky simulácie. Tieto vypočítané hodnoty rádiusu boli následne porovnávané, s cieľom objaviť na ktorom kroku a vo výstupe ktorej verzie softvéru sa začala vyskytovať odchýlka.

Výsledkom bolo, že aj v najvyššom teste s $1e7$ krokmi, sa ani v jednej z verzií softvéru nepreukázala žiadna odchýlka. Teda je možné konštatovať, že korektnosť výpočtu a zhodnosť výstupov bola dosiahnutá, tj. nedokázal sa predpoklad prítomnosti kumulatívnej chyby vznikajúcej s rastúcim počtom krokov výpočtu.

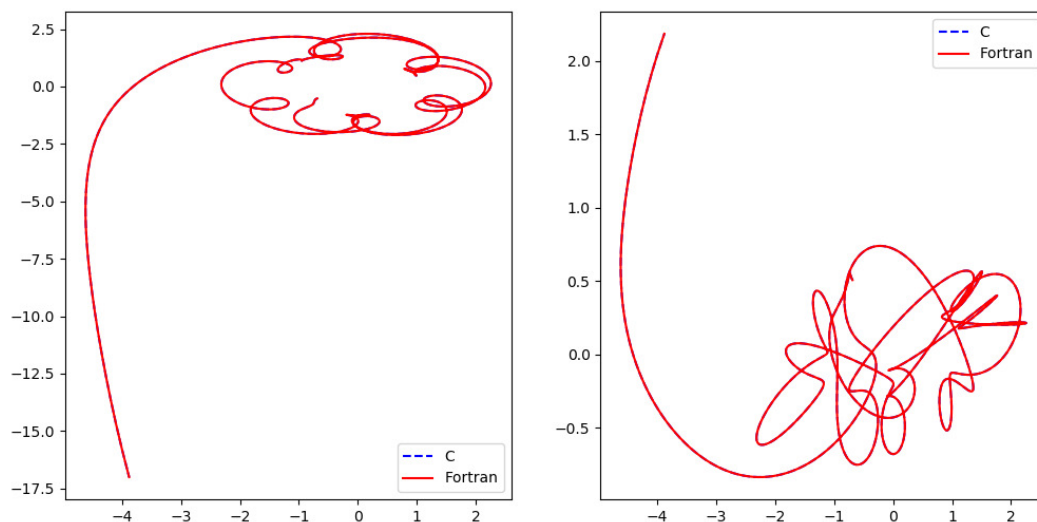
B. Testovanie trasovaním častice v reverznom smere príchodu do magnetosféry

Druhé testovanie pozostávalo z trasovania reverzného pohybu častice a porovnávania zhodnosti súradníc trajektórií oboch verzií. Toto testovanie pozostávalo z dvoch fáz.

V prvej fáze bola na trasovanie použitá metóda *backward tracing*, pričom bod a smer pod ktorým antičastica vyšla z magnetosféry, je bodom, a smerom, pod ktorým tam častica vošla. Z tohto bodu sa v druhej fáze trasuje nová trajektória v opačnom smere pre časticu s rovnakou energiou, ako mala antičastica smerom do magnetosféry. Tento postup vedie k pohybu častice po rovnakej trajektórii, ako bola trajektória získaná technikou *backward tracing*. Odchýlka je preverením chyby výpočtu.

Štandardne sa simulácie vykonávajú pre časticu antiprotón, ale pre tento test bola simulácia vykonaná pre protón. Taktiež bola zmenená štartovacia poloha na získanú polohu na magnetopauze, a uhly príchodu častice boli otočené, tj. k ich dĺžke bolo pripočítaných 180° a šírkovému uhlu bolo zmenené znamienko na opačné.

Výstupmi z tohto testu boli súbory s x , y a z súradnicami trajektórie častice. Tieto dáta boli následne vizualizované prostredníctvom Python skriptu využívajúceho knižnicu Matplotlib. Výstupné vizualizácie nepreukázali prítomnosť odchýlky trajektórií, tj. neprítomnosť diskrepancií vo výstupných hodnotách oboch verzií softvéru. Príklad jednej z takýchto vizualizácií môžeme vidieť na obrázku 3, kde osi x a y sú v jednotkách zemských polomerov.



Obr. 3 Vizualizácia porovnania trajektórie v XY (ľavý panel) a XZ (pravý panel) perspektívach.

VI. TESTOVANIE VÝKONU PROGRAMU

Celkovo bolo vykonaných 18 testov, pre tri rôzne pozície na Zemi, pre tri verzie programu, s hodnotami delenia kružnice rovnou 100 a 1000. Týmto testovaním bol pokrytý dostatočne veľký okruh potenciálnych výpočtov na preukázanie výkonnosti jednotlivých verzií.

Samotné testovanie bolo vykonané na výpočtovom systéme ÚEF SAV disponujúcom štyrmi osem jadrovými procesormi, *AMD Opteron Processor 6134*, dokopy ponúkajúci 32 jadri procesorov.

A. Testovanie výkonnosti pre vysokošírkové súradnice

Ako pozícia na vysokých geografických šírkach boli zvolené súradnice so šírkou 64.13° a dĺžkou -21.93° . V testoch bolo vykonaných 10 behov Fortran a C sekvenčnej verzie programu, a 100 behov C paralelnej verzie. Výstupné časy boli spracované a vložené do nasledujúcich tabuliek.

Programovací jazyk	Implementácia	Výkonnosť	
		Priemer v sekundách	Zrýchlenie
Fortran	Sekvenčná	411.5289	–
C	Sekvenčná	356.2319	1.15x
C	Paralelná	14.57016	28.24x

Tabuľka 1
 Tabuľka výstupov 1. testu s hodnotou násobenia kružníc 100.

Sekvenčná verzia v jazyku C prekonala Fortran verziu o viac ako 15% a paralelné vykonanie dosiahlo zrýchlenie o hodnote 28.24-násobku. Po vykonaní týchto testov, bola zvýšená hodnota delenia kružnice zo 100 na 1000 a testy boli vykonané znova. Výsledky je možné vidieť v tabuľke 2.

Programovací jazyk	Implementácia	Výkonnosť	
		Priemer v sekundách	Zrýchlenie
Fortran	Sekvenčná	599.9277	–
C	Sekvenčná	515.1019	1.16x
C	Paralelná	17.84122	32.62x

Tabuľka 2
 Tabuľka výstupov 1. testu s hodnotou násobenia kružníc 1000.

Zvýšením hodnoty násobenia kružnice sa zvýšila komplexnosť výpočtu z dôvodu precíznejšieho trasovania častice. Z toho dôvodu v tabuľke pozorujeme nárast priemerných hodnôt vykonania. C sekvenčná verzia znova dosiahla viac ako 15% zrýchlenie a zrýchlenie paralelnej verzie vyskočilo na hodnotu 32.62-násobku.

B. Testovanie výkonnosti pre strednošírkové súradnice

Pre druhý test boli trajektórie častíc počítané pre geografické súradnice šírky 44.13° a dĺžky -21.93° , reprezentujúce pozíciu na stredných geografických šírkach. Touto zmenou bola zmenená komplexnosť výpočtov na demonštráciu simulácií s dlhšími kalkulačnými časmi.

Programovací jazyk	Implementácia	Výkonnosť	
		Priemer v sekundách	Zrýchlenie
Fortran	Sekvenčná	2286.8937	–
C	Sekvenčná	1964.7387	1.16x
C	Paralelná	96.05335	23.80x

Tabuľka 3
 Tabuľka výstupov 2. testu s hodnotou násobenia kružníc 100.

Nastal očakávaný nárast priemernej doby vykonania o viac ako päťnásobok, celkové zrýchlenie C sekvenčnej verzie ostalo rovnaké, v rozmedzí 15 – 16%, a paralelná verzia dosiahla 23.80-násobné zrýchlenie oproti Fortran verzii programu. Výsledky testov pre hodnotu delenia kružnice rovnej 1000 sú v tabuľke 4.

Programovací jazyk	Implementácia	Výkonnosť	
		Priemer v sekundách	Zrýchlenie
Fortran	Sekvenčná	1094.3637	–
C	Sekvenčná	937.556	1.17x
C	Paralelná	36.68593	29.80x

Tabuľka 4
 Tabuľka výstupov 2. testu s hodnotou násobenia kružníc 1000.

Vo výsledkoch pozorujeme 17% zrýchlenie pre sekvenčnú C verziu a takmer 30-násobné zrýchlenie paralelnej verzie. Taktiež nastal pokles časov vykonania o takmer 53% pre sekvenčné testy, a viac ako 61% pre paralelné testovanie. Tento jav je ľahko vysvetliteľný tým, že pre dané súradnice môže zvýšenie hodnoty delenia kroku zapríčiniť, že simulované trajektórie častíc

dopadnú na Zem a preto sa výpočet pre danú trajektóriu častice ukončí skôr. Takáto udalosť nemusí ale nastať pri použití nižšej precíznosti, tj. hodnoty delenia kružnice.

C. Testovanie výkonnosti pre pozíciu v blízkosti rovníka

Posledný test použil súradnice blízko geografického rovníka, tj. šírku 4.13° , a -21.93° .

Programovací jazyk	Implementácia	Výkonnosť	
		Priemer v sekundách	Zrýchlenie
Fortran	Sekvenčná	484.9339	—
C	Sekvenčná	413.6715	1.17x
C	Paralelná	13.35475	36.31x

Tabuľka 5
Tabuľka výstupov 3. testu s hodnotou násobenia kružníc 100.

Výsledky taktiež poukazujú na viac ako 15% zrýchlenie C sekvenčnej verzie a paralelná verzia programu dosiahla najvyššie zrýchlenie oproti Fortran verzii, o hodnote 36.31-násobného zrýchlenia. Posledná tabuľka, 6, obsahuje výstupné hodnoty z tretieho testovania pre hodnotu delenia kružnice rovnej 1000.

Programovací jazyk	Implementácia	Výkon	
		Priemer v sekundách	Zrýchlenie
Fortran	Sekvenčná	153.5858	—
C	Sekvenčná	130.7991	1.17x
C	Paralelná	4.24394	36.19x

Tabuľka 6
Tabuľka výstupov 3. testu s hodnotou násobenia kružníc 1000.

Znova pozorujeme rovnaké, 17%, zrýchlenia oproti Fortran verzii programu, a paralelná verzia sa preukázala o viac ako 36-násobne rýchlejšia.

VII. ZÁVER

RefaktORIZÁCIA pôvodného programu slúžiaceho na simulovanie trajektórií častíc kozmického žiarenia v zemskej magnetosfére dosiahla všetky požadované ciele, ktoré sa od nej očakávali. Čitateľnosť a prehľadnosť zdrojového kódu bola zvýšená vytvorením samostatných hlavičkových súborov pre jednotlivé časti kódu, ako aj zjednodušením jednotlivých funkcií, a odstránením nepotrebných častí v zdrojovom kóde. RefaktORIZÁCIU bolo dosiahnuté viac ako 15% zrýchlenie, oproti pôvodnej verzii, a s implementáciou paralelného vykonávania, prostredníctvom OpenMP, bolo dosiahnuté mnohonásobné urýchlenie vykonania program, závislého od počtu voľných jadier procesora.

LITERATÚRA

- [1] P. Bhattacharjee and G. Sigl, "Origin and propagation of extremely high-energy cosmic rays," *Physics Reports*, vol. 327, no. 3-4, pp. 109–247, 2000.
- [2] K. Kudela and P. Bobik, "Long-Term Variations of Geomagnetic Rigidity Cutoffs," vol. 224, no. 1-2, pp. 423–431, Oct. 2004.
- [3] K. Kudela, R. Bučík, and P. Bobík, "On transmissivity of low energy cosmic rays in disturbed magnetosphere," *Advances in Space Research*, vol. 42, no. 7, pp. 1300–1306, Oct. 2008.
- [4] F. Donato, D. Maurin, P. Salati, A. Barrau, G. Boudoul, and R. Taillet, "Antiprotons from spallations of cosmic rays on interstellar matter," *The Astrophysical Journal*, vol. 563, no. 1, p. 172, 2001.
- [5] V. Saravanan, M. Radhakrishnan, A. Basavesh, and D. Kothari, "A comparative study on performance benefits of multi-core cpus using openmp," *International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 1, p. 272, 2012.
- [6] L. Dagum and R. Menon, "Openmp: an industry standard api for shared-memory programming," *IEEE computational science and engineering*, vol. 5, no. 1, pp. 46–55, 1998.
- [7] B. Mustafa and W. Ahmed, "Parallel algorithm performance analysis using openmp for multicore machines," *International Journal of Advanced Computer Technology (IJACT)*, vol. 4, no. 5, 2015.

Prepracovanie Inter-Process Communication zadania z predmetu Operačné systémy

Dominik OSVALD, Ján Genči

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická
univerzita v Košiciach, Slovenská republika

dominik.osvald@student.tuke.sk, genci@tuke.sk

Abstrakt — IPC zadanie je súčasťou predmetu Operačné systémy už od počiatku tohto predmetu. Súčasná forma zadania ktorá sa dodnes využíva, bola nasadená do výučby v roku 2006. Tento článok stručne popisuje prepracovanie tohto zadania a jeho následnú integráciu do systému na odovzdávanie zadaní. Výsledkom práce je plne automatizované generovanie variantov, ich automatické priradovanie a následná kontrola zadaní. Prepracované zadanie je potenciálne schopne nasadenia do procesu výučby.

Kľúčové slová — Medziprocesova komunikácia, automatizované vyhodnocovanie, operačné systémy

I. ÚVOD

Na Technickej univerzite v Košiciach sa vyučuje predmet Operačné systémy. Ten ma za úlohu oboznámiť študentov informatiky hlavne prácou s jadrom (systémovými volaniami), so súbormi, regulárnymi výrazmi, linuxovým príkazovým riadkom a v poslednom rade s medziprocesovou komunikáciou

Zadanie Inter-process communication (IPC) je v osnovách predmetu už od roku 2006. Počas tohto celého času dané zadanie neprešlo žiadnymi úpravami a je v osnovách stále v pôvodnom znení. Študenti majú za úlohu vypracovať niekoľko procesov (programov), ktoré medzi sebou posúvajú určitý druh správy. Pre riešenie zadania sa hlavne kladie dôraz na vedomosti, ktoré by mali študenti nadobudnúť behom cvičení. Jedná sa hlavne o oblasti medziprocesovej komunikácie (rúry, zdieľaná pamäť, sockety). Samotné programy majú vlastne za úlohu presunúť obsah súborov cez všetky študentove a aj kontrolne procesy a následne vygenerovať výstupný súbor s daným obsahom. Každý tento proces obsahuje iný druh medziprocesovej komunikácie, tú využíva aby komunikoval s nasledujúcim procesom, kvôli odoslaniu danej správy.

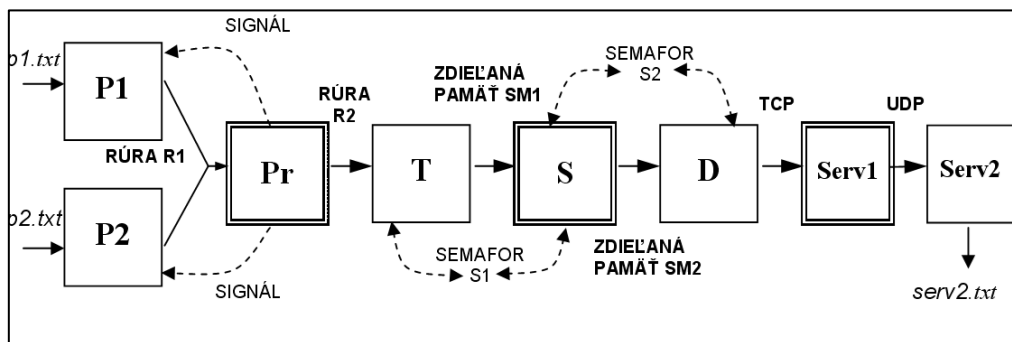
Na samotnú kontrolu a odovzdávanie tohto IPC zadania slúži systém na odovzdávanie zadaní v predmete Operačné systémy (OS). Ten bol vytvorený v roku 2005 Marekom Urbančíkom [1]. Jedná sa o webovú aplikáciu. Pôvodne tento systém slúžil na testovanie zadaní zo súborových systémov FAT12/16/32. O rok neskôr bol tento systém rozšírený v jadre kontroly o nový modul, ktorý poskytoval možnosť kontroly zadania IPC. Tento modul bol pridaný Ondrejom Krasnayom [2]. Samotný systém spolu s už zmienaným zadaním je využívaný dodnes. Za celý čas chodu, systém prešiel viacerými zmenami, pričom poslednú veľkú zmenu vykonal Peter Piatničko [3]. Upravil zdrojové kódy, ktoré sa starajú o kontrolu originality a podobnosti študentských zadaní. Do systému taktiež pridal paralelný algoritmus na kontrolu zadaní, keďže sa dovtedy priebeh testovania vykonával sekvenčne. Posledné úpravy systému boli ale zaznamenané Pavlom Dudríkom [4]. Študenti počas výučby OS odovzdávajú archív zdrojových kódov vypracovaných v jazyku C alebo C++, ktoré systém spúšťa v chránenom prostredí. Prínosom takéhoto automatizovaného riešenia je odbúranie časovo náročnej práce manuálneho porovnania jednotlivých zadaní pedagógom. Pre každé zadanie v systéme existuje kontrolný skript, ktorý overuje funkcionality zadania.

II. FORMA IPC ZADANIA

A. Terajší stav

Zadanie Unix1[3] je momentálne jediným zadáním, na predmete OS, ktoré sa zaoberá medziprocesovou komunikáciou v systémoch unix. Toto zadanie je implementované do systému na odovzdávanie zadaní z predmetu OS. Cieľom tohto zadania je preveriť študentove schopnosti v oblasti medziprocesovej komunikácie a jej komunikačných metód. Zadanie priamo obsahuje: rúry, signály, zdieľanú pamäť, semaforey a protokol TCP aj UDP. Ide vcelku o náročnejšie zadanie, ktoré vyžaduje od študenta zvládnutie všetkých komunikačných technológií, ktoré sa vyučujú na cvičeniach predmetu Operačné systémy.

Študent má za úlohu vypracovať procesy na obr 1.(tie ktoré nie sú zvýraznené) a hlavný program, ktorý má všetky tieto procesy riadiť. V prvých dvoch procesoch z programovej schémy má za úlohu po prijatí signálu (od zvýrazneného kontrolného procesu) čítať riadky z predom vytvorených súborov a následne tieto slová postupne odosielať. Odosielanie prebieha pomocou rúry. Je nutné zdôrazniť že tieto signály, ktoré prijímajú, fungujú ako synchronizačná technika, aby oba procesy sa nesnažili zapisovať do jednej rúry naraz. Študent ďalej vytvára proces, v ktorom má za úlohu prijať (prečítať) slová z rúry. Potom odoslať (zapísať) slová do zdieľanej pamäte (ktorú zdieľa s ďalším kontrolným procesom). Ako synchronizačná metóda pre zdieľanú pamäť má využiť semafor. Ten využíva na to aby vedel kedy kontrolný proces smie údaje čítať. Študent ďalej vytvára proces v ktorom má za úlohu prijať (prečítať) slová zo zdieľanej pamäte a takisto zachovať synchronizáciu pomocou semafora. Potom v rámci procesu odosiela slová pomocou TCP socketu. Napojí sa na TCP server (ktorý vytvoril posledný kontrolný proces) a následne naň odosiela prijaté slová. Vo svojom poslednom procese má už iba za úlohu vytvoriť UDP server. A postupne prijaté slová zapisovať do novo vytvoreného súboru. Každé slovo na nový riadok. Celý tento chod je zobrazený na obr. 1.

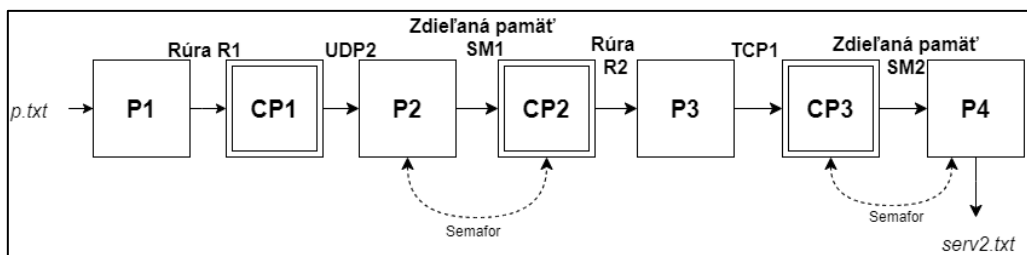


Obr. 1 Programová schéma terajšieho IPC zadania.

Študent ešte vytvára hlavný program (ten nie je súčasťou programovej schémy), v ktorom vyhradzuje všetky systémové prostriedky pre medziprocesovú komunikáciu a následne všetky podprogramy spúšťa aj spolu s dodanými prostriedkami formou argumentov.

B. Nová forma

Nová forma zadania vychádza z tej predošlej. Študent má stále za úlohu vytvoriť jednotlivé podprogramy, ktoré obsahujú jednotlivé druhy medziprocesovej komunikácie. Takisto hlavný program, ktorý riady chod všetkých ostatných procesov. Rozdiel ale nastáva vo vytvorení rôznych variantov zadania. Hlavný princípom týchto variantov je aby každý študent v systéme mal pridelenú inú prácu. Varianty svojím spôsobom berú staršie zadanie a prehadzujú jednotlivé medziprocesové prostriedky tak aby ani jeden variant zadania nebol úplne rovnaký a týmto spôsobom aby každý študent robil aspoň o kúsok iné zadanie. Takže študentovi budú dodané kontrolne procesy a samotný študent bude pracovať na podprogramoch, ktoré ale budú mať rôzne kombinácie medziprocesových prostriedkov. Schéma ako vyzerá 1 variant tohto zadania je zobrazená na obr. 2.



Obr. 2 Programová schéma jedného z variantov zadania.

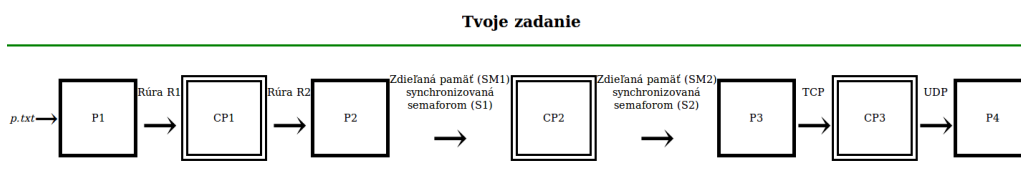
III. AUTOMATICKÉ PRIRAĐOVANIE ZADANÍ

Bolo absolútne nevyhnutné aby došlo k pozmeneniu systému, aby sa vyriešil problém automatického priraďovania rôznych variantov zadania študentom. Keďže pomocou terajšieho stavu systému by bolo nutné ručne vytvárať každý variant zadania a následne ho ručne priraďovať každému študentovi.

A. Vytvorenie variantov zadania

Varianty sú formované pomocou skriptu, ktorý nám vytvorí všetky variácie 6. triedy z množiny 4 prvkov (rúra, zdieľaná pamäť, TCP, UDP) s možnosťou opakovania a následne všetky nadbytočne vytvorené prvky vyfiltruje. Vstupná množina bola určená prirodzenými číslami (1-4), ktoré boli substituované pôvodnými prvkami. Výsledok bol uložený do textového súboru. Každý variant sa nachádza na osobitnom riadku. Následne sa filtrovaním zabezpečilo, že každý prvok množiny sa mohol nachádzať na párnej a nepárnej pozícii výstupného variantu maximálne raz. Ak sa napríklad na párných pozíciách nachádzajú prvky 1,2,3 a na nepárnych prvky 2,3,4, tak výsledný reťazec vyzerá nasledovne: 213243. Pred filtrovaním bolo vytvorených spolu 4096 variantov. Po filtrovaní ostalo vo výslednom súbore 576 variantov.

Keďže samotný systém nepodporuje viaceré varianty zadania a tým pádom ani ich nezobrazuje, bolo nutné všetky vytvorené varianty jednak uložiť a jednak konvertovať aby sa dali zrozumiteľne čítať. Všetky varianty z textového súboru sa pred priraďovaním ukladajú do systémovej databázy. Proces číta zo súboru, kde sa nachádzajú varianty zadania. Súbor číta po riadkoch. Každý riadok predstavuje vo vytvorenej tabuľke ID variantu. V ďalšom stĺpci sa ukladá už obsah riadku. No a následne prebieha generovanie html kódu z pred-pripraveného vzoru do ktorého sa už iba vkladajú namiesto čísel názvy jednotlivých medziprocesových komunikácií. Celý tento vygenerovaný kód sa uloží do posledného stĺpca tabuľky. Tento kód predstavuje grafické zobrazenie zadania. Príklad takto zobrazenia takto vygenerovaného kódu sa nachádza na obr. 3.



Obr. 3 Grafické zobrazenie vygenerovaného variantu.

B. Priraďovanie variantov

Pri procese priraďovania zadania študentom, zistíme predtým ako odošleme údaje do databázy, údaj o poslednom pridelenom IPC zadani a jeho dodatočnom ID (ID variantu zadania). Pre tieto účely bolo nutné vytvorenie dodatočného stĺpca v tabuľke s priradenými zadaniami. Ak doposiaľ nebolo priradené žiadne zadanie alebo bolo predchádzajúce zadanie posledným variantom (v tabuľke s variantmi), tak začne ID od čísla 1. Pri každom ďalšom prípade už iba zvýši hodnotu o 1.

Študentovi je následne zobrazovaný jemu pridelený variant formou html kódu, ktorý je vyselektovaný na základe ID variantu.

IV. TESTOVANIE

A. Príprava testovania

Akonáhle študent cez systém odovzdá svoje zadanie, systém odovzdá celý archív už

konkrétnemu kontrolnému skriptu. Ten riadi celý proces kontroly. Ten má na starosti ako celú prípravu rôznych testovacích súborov, kontrolu študentových procesov, tak ich spúšťanie a následne aj vyhodnotenie a zobrazenie výsledkov. Je nutné poznamenať, že v tomto konkrétnom prípade to funguje naozaj podobne.

Po prijatí študentovho zadania sa vytvorí pracovná zložka s jeho menom. Zadanie sa tu rozbalí. Prebehne kontrola nepovoleným príkazov a následná kompilácia študentových programov. Pripraví sa vstupné testovacie súbory. Pripraví sa kontrolné procesy. Tie sa pozmenia z predom pripraveného zdrojového kódu na konkrétny variant zadania. To prebieha pozmeňovaním premennej vnútri kódu ktorá obsahuje variant zadania pre potreby korektného testovania. Následne sa tieto procesy skompilujú a dodajú sa k študentovým programom. V poslednom kroku dochádza k spusteniu študentovho zadania v chránenom prostredí.

B. Priebeh testovania

Program podľa poradia kontrolného procesu prečíta 2 príslušné znaky, ktoré sa nachádzajú v premennej, ktorá bola dodaná. Potom si kontrolný proces deklaruje všetky potrebné premenné. Je nutné zdôrazniť, že daný proces obsahuje všetky varianty zadania, preto aj pri deklarácii premenných je nutné si vyhradiť prostriedky pre všetky možné varianty. V ďalšej časti dochádza k overovaniu argumentov s ktorými sa proces spúšťal. Tie by mali zodpovedať vstupnej a výstupnej medziprocesovej komunikácii. Potom sa už iba odošle signál rodičovskému procesu, že je všetko pripravené na prenos údajov. Ďalej dochádza k prijímaniu a následnému odosielaniu údajov. Desať údajov prijme a odošle. Medzi týmito dvoma časťami dochádza ešte ku overovaniu a generovaniu nového reťazca, ktorý sa neskôr odosiela ďalej. Týmto spôsobom sa zaisťuje integrita prenášaných údajov.

Tá spočíva v myšlienke pripravenia zakódovaných údajov a následne kontroly a modifikovania týchto prenášaných údajov medzi jednotlivými procesmi. Týmto spôsobom každý kontrolný proces vie, aký presný údaj má prijať a tým pádom ho vie aj skontrolovať. Každý z kontrolných procesov modifikuje prenášaný údaj a odošle ho ďalej. Ak prijatý údaj nie je platný, teda bol určitým spôsobom počas prechodu procesov modifikovaný (či už úmyselne alebo nie), proces, ktorý tento údaj aktuálne spracováva, vypíše chybu s ňou spojenú a následne ukončí svoje vykonávanie. Takto sa zistí platnosť údajov zo vstupného procesu a následný prechod všetkými procesmi komunikácie. Na vstupe sa pripraví jeden textový súbor, ktorý obsahuje šifrované údaje. Behom chodu procesov sa kontrolujú tieto šifrované údaje a následne sa vygenerujú nové, pre ďalší kontrolný proces v poradí.

C. Vyhodnotenie

Na úplnom konci, po skončení študentovho zadania sa vyhodnotí výstupný súbor na základe prvého vygenerovaného textového súboru a ich nasledujúcich modifikácií počas chodu jednotlivých kontrolných procesov. Spôsoby generovania vstupného súboru, overovania jednotlivými procesmi a finálnym vyhodnotením boli vytvorené v predchádzajúcej implementácii testovania zadania[2]. Skript ktorý pripravoval všetky potrebné veci následne vyhodnotí správnosť riešenia. Uloží výsledky do systému a celý pracovný adresár zmaže. Všetky výsledne súbory generované kontrolnými procesmi sú zobrazené študentovi. Hlavne kvôli prípadným chybám v študentovom zadaní a ich následnej jednoduchšej korektúry.

V. VYTVORENIE KONTROLNÝCH PROCESOV PRE LOKÁLNU REALIZÁCIU ZADANÍ

Samotné zadanie sa kúsok líši od ostatných, keďže študenti musia pri realizovaní pracovať s dodanými procesmi. Aby sa predišlo zbytočnému zaťažovaniu systému kontroly, bolo umožnené študentom ladenie i na iných platformách založených na systéme Unix. Takto budú mať študenti väčšiu voľnosť pri tvorbe svojho riešenia, keďže budú môcť celú túto prácu vykonávať doma na svojich počítačoch bez potreby neustáleho odosielanie zadania do systému. V prvom kroku bolo nutné pozmeniť súčasne kontrolne procesy aby neobsahovali overovanie pomocou "značkovania". Došlo teda k odstráneniu tejto časti. Pri týchto procesoch sa prijatý reťazec jednoducho pošle ďalej, na výstupnú medziprocesovú komunikáciu.

Na začiatku skript jednoducho nakopíruje pozmenené procesy do zložky s dočasnými súbormi. Zložky s dočasnými súbormi (tzv. temp) sú vstavanou funkciou systému. V tejto zložke bola vytvorená nová zložka, ktorej názov bol špecifikovaný ako priezvisko, meno študenta a ID variantu (napr. Priezvisko_Meno_1). Následne prebehlo pozmenenie zdrojového kódu na konkrétny variant zadania. To prebehlo pozmenením premennej vnútri kódu z ktorej sa variant kontroluje. Po tomto vlastne už nebolo nutné žiadne ďalšie pozmeňovanie kontrolných procesov. Celý tento pozmenený proces ďalej beží ďalej ako klasický kontrolný proces. Ďalej

prebehne kompilácia. Aby mohli študenti využívať programy na domácu prácu a testovanie zadania, boli tieto programy skompilované staticky. Tieto staticky skompilované programy prekopírujú knižnice využívané v kontrolných procesoch. Programy sa ukázali ako vhodné pre testovanie na rôznych verziách systému Linux. Doteraz nebol zaznamenaný ani jeden prípad nekompatibility týchto programov s verziou systému Unix, kde by nebolo možné programy spustiť. Menšou nevýhodou staticky skompilovaných programov, je podstatne väčšia veľkosť. V poslednom kroku už dôjde iba k vytvoreniu archívu s vykompilovanými programami. Všetky ostatné súbory sú následne zmazané.

Všetko toto sa vykonáva akonáhle sa študent pokúsi stiahnuť procesy zo systému na odovzdávanie zadani. Po úspešnom stiahnutí programov vo forme archívu sa tento archív následne zo systému zmaže.

VI. ZÁVER

Inter-process communication zadanie má svoje fixné miesto v procese výučby predmetu Operačné systémy. Je dlhodobo využívané na pochopenie problematiky medziprocesovej komunikácie. Samotný systém, ktorý toto zadanie kontroluje má takisto svoje pevné miesto v tomto predmete keďže pomáha vyučujúcim predmetu pri kontrole a samotnom hodnotení študentov. Hlavnou podstatou zmeny bolo vytvorenie viacerých variant zadania, ktoré by si rôznym spôsobom obmieňali IPC prostriedky a prispôsobiť na to testovanie v systéme. Po analýze systému sme, ale zistili, že samotný systém nie je na niečo také prispôbený. Bez jeho pozmenenia by bolo nutné vytvárať všetky varianty ručne, následne vytvárať ručne popisy ku každému a v poslednom rade ručne priradovať študentom. Za týmto účelom bol samotný systém pozmenený. Systém pri hromadnom pridelovaní zadania automaticky vygeneruje varianty zadani (ak žiadne neexistujú), ktoré následne priraduje študentom.

Samotný proces kontroly v podstate zmenený nebol. Funguje na tej báze, že každý kontrolný proces si pozmení získane údaje. Spôsob overovania a pozmeňovania týchto údajov je taký aký bol doteraz. Kontrolne procesy boli prispôbené tak aby obsahovali všetky časti kontroly. Tie boli rozdelené jednoduchými podmienkami, ktoré kontrolujú o aký variant sa jedná následne spustia danú časť kódu. Dodanie tohto variantu sme zaistili dynamickým pozmeňovaním zdrojových súborov, ešte pred začatím samotného procesu kontroly. V rámci vlastnej iniciatívy bol navrhnutý aj implementovaný generátor procesov. Aby sa zamedzilo väčšiemu náporu na systém, bude možné si dané kontrolne procesy stiahnuť, na lokálne testovanie. Tie systém automaticky generuje a následne umožní študentom ich stiahnutie. Celý systém spolu s novým zadaním je potencionálne schopný nasadenia do procesu výučby. Samotné zadanie IPC vďaka svojej rozsiahlosti poskytuje široké možnosti návrhu rozšírenia o ďalšie druhy medziprocesovej komunikácie. Systém na odovzdávanie zadani z predmetu OS je už totiž na to prispôbený.

REFERENCES

- [1] URBANČÍK, Marek. Testovanie a úpravy systému na odovzdávanie systému z OS. Košice, 2005. Diplomová práca. Technická univerzita v Košiciach.
- [2] KRASNAY, Ondrej. Systém pre správu, automatizáciu odovzdávania a kontroly zadani. Košice, 2006. Diplomová práca. Technická univerzita v Košiciach.
- [3] PIATNIČKO, Peter. Systém pre správu, automatizáciu odovzdávania a kontroly zadani. Košice, 2015. Diplomová práca. Technická univerzita v Košiciach.
- [4] DUDRÍK, Pavol. Porovnanie antiplagiátorských algoritmov a prístupov v systéme na odovzdávanie zadani z OS. Košice, 2018. Diplomová práca. Technická univerzita v Košiciach.

Rozšírenie code-switching databázy pre účely automatického rozpoznávania bilingválnej reči

Veronika RÚŠOVÁ, Matúš PLEVA

Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita Košice, Slovenská Republika

veronika.rusova@student.tuke.sk, matus.pleva@tuke.sk

Abstrakt – Hlavnou témou tejto práce je vylepšenie automatického rozpoznávania bilingválnej reči so zameraním na slovenčinu a angličtinu. Základnými pojmami sú code switching, code mixing, automatické rozpoznávanie reči a metódy spracovania a rozpoznávania reči, ktoré úzko súvisia so skúmanou problematikou. Dôležitou súčasťou práce je praktická časť, v ktorej sme sa venovali v prvej fáze hľadaniu dvojjazyčných slovensko-anglických nahrávok a ich spracovaniu pomocou systému SARRA. Výsledkom tejto fázy bola databáza nahrávok a prepisov. V druhej fáze praktickej časti sme riešili štatistické vyhodnotenie prepisov a automatizáciu exportu výsledkov do tabuľkových súborov.

Kľúčové slová — bilingvizmus, dvojjazyčnosť, multilingvizmus, viacjazyčnosť, prepínanie medzi jazykmi, miešanie jazykov, automatické rozpoznávanie reči

I. ÚVOD

V súčasnosti sme svedkami rozsiahlej expanzie moderných komunikačných prostriedkov, ktoré sú založené na najmodernejších technológiách. Používanie dvoch i viacerých jazykov v jednej komunikácii, či dokonca v rámci formulácie viet jedného človeka je dnes už úplne bežnou súčasťou nášho každodenného života. Stretávame sa s týmto fenoménom v škole, pri sledovaní televízie, pri využívaní internetu, pri práci, športovaní, ale aj na koncertoch obľúbených spevákov a kapiel.

Katedra elektrotechniky a multimediálnych telekomunikácií (KEMT) Fakulty elektrotechniky a informatiky Technickej univerzity Košice spolupracuje na projekte multilingválneho automatického rozpoznávania reči. Tento projekt má označenie APVV SK-TW-2017-0005. Jeho súčasťou je aj dvojjazyčný slovensko-anglický model. Pre účely vývoja tohto modelu boli vytvorené základy slovensko-anglickej bilingválnej databázy. Táto databáza je v súčasnosti anotovaná o ďalšie bilingválne dáta na zvýšenie presnosti a zároveň na testovanie rozpoznávacieho bilingválneho systému. Pred touto prácou boli v databáze anotované nahrávky v trvaní 121,9 minút, obsahujúce viac ako 9 500 slov a 23% podiel anglických slov [1]. Doterajšie výsledky bilingválneho DNN (Deep learning Neural Network) LVCSR (Large Vocabulary Continuous Speech Recognition) s využitím modulu LID (Language Identification) a uvádzané v mierke WER (Word Error Rate) pre evaluačnú množinu dát slovenského jazyka sú rovné 16,13% a anglického jazyka 8,97% [1]. Tieto hodnoty vyjadrujú percentuálnu úspešnosť rozpoznávania slov ako podiel správne rozpoznávaných slov ku počtu všetkých slov testovanej databázy.

Vykonalí sme prieskum medzi voľne dostupnými audiovizuálnymi dátami. Následne sme vykonali anotáciu nahrávok, s ohľadom na dodatočné korekcie a jednotné označovanie anglických slov v texte, s použitím nástrojov pracoviska KEMT (systém SARRA). Spracované prepisy nahrávok sme štatisticky vyhodnotili. Cieľom vyhodnotenia bolo zistiť početnosť anglických a slovenských slov, ako aj celkovú dĺžku spracovaných nahrávok.

II. CODE SWITCHING A CODE MIXING

Oba javy code switching (prepínanie kódov) aj code mixing (miešanie kódov) zahŕňajú vytváranie hybridných slov alebo prepínanie medzi dvoma alebo viacerými jazykmi v rámci komunikácie [2]. Niektorí odborníci považujú tieto výrazy za vzájomne zameniteľné, najmä tí,

ktorí študujú morfológiu, syntax a iné formálne aspekty jazyka. Iné oblasti, napríklad odbory lingvistiky, komunikácie alebo teórie vzdelávania, majú svoje vlastné extrémne špecifické definície miešania kódu [3].

A. Code switching

Code switching nastáva, keď rečník strieda v konverzácii dva alebo viac jazykov, dialektov, či variantov jazyka. Tento jav sa vyskytuje väčšinou v dvojazyčných komunitách, ktoré používajú určité stratégie na zefektívnenie a zmysluplnosť komunikácie. Code switching sa týka použitia dvoch jazykov vo vete alebo v rozhovore. Ide o prirodzený jav vyskytujúci sa veľakrát medzi viacjazyčnými jedincami, ktorí majú spoločných dva alebo viac jazykov [3].

Existuje niekoľko možných dôvodov prečo ľudia prepínajú z jedného jazyka do druhého.

- Splnenie potreby - rečník, ktorý sa nevie vyjadriť v jednej reči, môže prejsť na inú, aby nahradil svoj nedostatok. Následne sa stáva, že rečník bude chvíľu hovoriť v inom jazyku [3].
- Vyjadrenie solidarity - k prechodu na inú reč mnohokrát dochádza keď jednotlivец chce vyjadriť solidaritu s danou sociálnou skupinou. Vzťah sa vytvorí medzi rečníkom a poslucháčom, keď poslucháč odpovie podobnou zmenou reči [3].
- Vylúčenie ostatných - code switching tiež používajú ľudia na vylúčenie ostatných z konverzácie tým, že použijú jazyk, ktorým ostatní nerozumejú [3].

B. Code mixing

Pri code mixing sa používajú len určité slová a frázy z iného jazyka namiesto tých, ktoré sú v pôvodnom jazyku v rámci jednej vety [4]. Rečník môže meniť jazyky, ak nevie správny preklad konkrétneho slova do iného jazyka. V skutočnosti po určitom čase sa mnoho nahradených slov používa natoľko často, že sa stanú súčasťou jazyka [3].

C. Rozdiely medzi pojmami code mixing a code switching

Code switching je niečo, čo hovoriaci robia úmyselne, pretože sa chcú vyjadriť osobným štýlom alebo chuťou, na druhej strane, code mixing je niečo, čo by rečníci mohli robiť neúmyselne jednoducho preto, že nepoznajú správne slovo alebo frázu [3]. Kým code switching je opísaný ako prepínanie medzi jazykmi, založené na zmenách v rečovej situácii, kde sa téma alebo členovia konverzácie menia, code mixing nezávisí od situačných zmien v konverzácii [4].

III. ANOTÁCIA NAHRÁVOK S BILINGVÁLNOU REČOU

V lingvistike anotácia často znamená doplnenie pôvodného textu slovným rozborom každého jeho slova. Súbory takto anotovaných textov tvoria korpusy a sú dôležitou pomôckou pre počítačovú lingvistiku pri automatickom rozpoznávaní reči, pri tvorbe strojových prekladov a podobne. V týchto záznamoch vystupovalo jeden alebo viac rečníkov nachádzajúcich sa v rozličných prostrediach. Podarilo sa nám nájsť nahrávky z rôznych oblastí, ako napríklad móda, moderné technológie, šport, politika, hudba, jazykové kurzy, technické prednášky a z mnohých iných sfér života. Tieto nahrávky boli vo formáte *.wav*.

A. Sarra

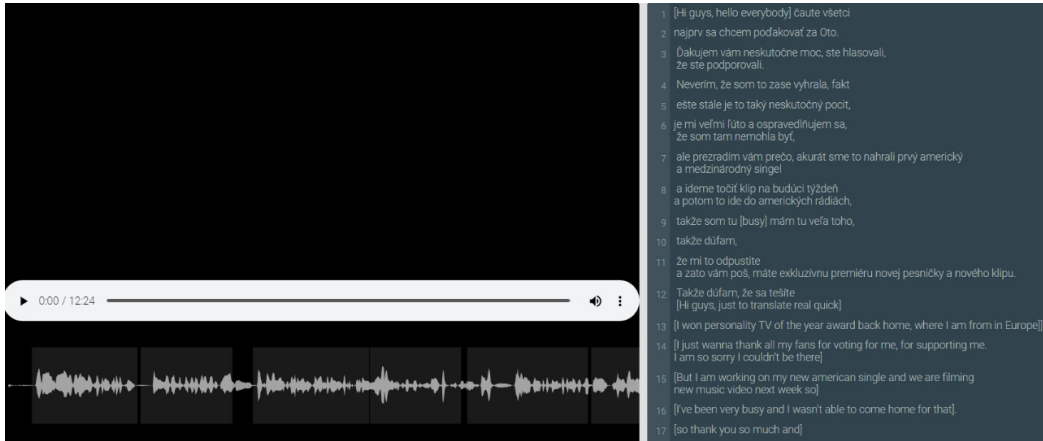
Na anotovanie záznamov s bilingválnou rečou sme použili systém Sarra¹ [5] [6]. Tento systém rozpoznal a vygeneroval prepis slovenskej reči nachádzajúcej sa v danej nahrávke [7]. Následne sme vykonali nasledovné úpravy a korekcie textu:

- anglické slová alebo skupiny anglických slov, stojacich v texte pri sebe, sme vložili do hranatých zátvoriek
- citoslovčia sme odlišili pomocou znaku „%”
- nekorektné vyslovené slová sme vložili do dvojitého okrúhlych zátvoriek
- skratky vyslovené anglicky sme označili pomocou „^^”
- skratky vyslovené po slovensky sme označili znakom „~”

¹<https://marhula.fe.i.tuke.sk/sarra/>

Dôvodom na vyššie uvedené úpravy a korekcie textu bola aj skutočnosť, aby sa do výsledného štatistického vyhodnotenia nezapočítavali citoslovčia a nekorektné vyslovené slová, ktoré nepomáhajú splňať vytýčené ciele.

Výstupom boli textové súbory s príponou *.vtt*. Dátový súbor uložený vo formáte Web Video Text Tracks (WebVTT) obsahuje informácie o webovom videu, ako sú titulky, popisy, kapitoly a metadáta.



Obr. 1 Ukážka systému Sarra

WEBVTT

00:00:00.760 --> 00:00:03.520

[hey guys, hello everybody] ahojte čaute všetci

00:00:03.600 --> 00:00:06.000

najprv sa chcem poďakovať za Oto

00:00:06.550 --> 00:00:09.610

ďakujem vám neskutočne moc, že ste hlasovali, že ste podporovali

00:00:09.600 --> 00:00:11.980

neverím, že som to zase vyhrala, fakt ako

00:00:12.140 --> 00:00:15.300

ešte stále je to je to také neskutočný pocit

00:00:15.380 --> 00:00:18.920

%ehm je mi veľmi ľúto a ospravedlňujem sa, že som tam nemohla byť

00:00:19.360 --> 00:00:24.280

ale prezradím vám prečo, akurát sme donahráli prvý americký a medzinárodný singel

00:00:24.310 --> 00:00:27.360

a ideme točiť klip na budúci týždeň a potom to ide do amerických rádiách

00:00:27.360 --> 00:00:30.600

takže %ehm som tu [busy] mám tu veľa toho,

00:00:30.640 --> 00:00:31.740

takže dúfam, že

00:00:31.780 --> 00:00:34.200

mi to odpustíte %ehm a zato vám ((poš))

Obr. 2 Príklad výstupného textového súboru vo formáte *.vtt*

IV. ŠTATISTICKÉ VYHODNOTENIE ANOTOVANÝCH NAHRÁVOK

Na účely štatistického vyhodnotenia spracovaných nahrávok a prepisov dialógov sme naprogramovali skript spustiteľný pomocou Windows PowerShellu od Microsoftu, ktorý je súčasťou operačných systémov Windows. Nie je preto potrebná dodatočná inštalácia ani investícia na nákup iného softvéru okrem operačného systému Windows. Okrem toho má charakter open source.

A. Vstupy do skriptu

Základnými vstupnými prvkami sú nahrávky vo formáte *.wav*, a prepisy nahrávok vo formáte *.vtt*. Tieto súbory môžu byť umiestnené v samostatných priečiňkoch počítača, ale nie je to nevyhnutná podmienka správnej funkčnosti skriptu. Je to skôr vhodnejšie kvôli prehľadnosti, najmä pri väčšom počte nahrávok a prepisov. Názvy prepisov musia byť rovnaké ako názvy príslušných nahrávok.

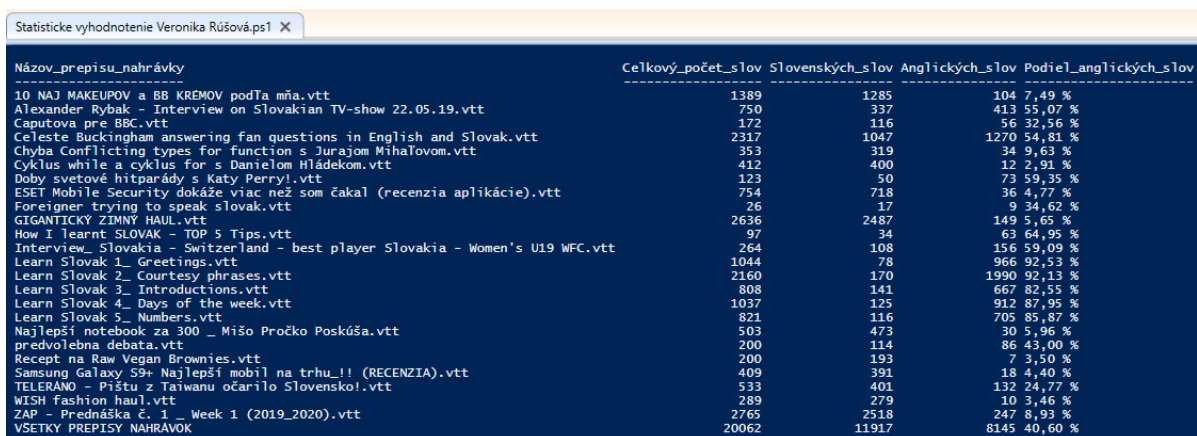
Posledným vstupom skriptu je zadanie umiestnenia nahrávok a prepisov v danom počítači. Nie sú pri tom kladené žiadne obmedzenia. Iné formáty vstupných súborov by bolo možné spracovať a vyhodnotiť predmetným skriptom po jeho drobnej modifikácii.

B. Výstupy zo skriptu

Do skriptu sme zakomponovali tri druhy výstupov ako štatistických vyhodnotení spracovaných nahrávok:

1. Tabuľku v prostredí Windows PowerShell, ktorá obsahuje nasledovné údaje:
 - a. Názov prepisu nahrávky
 - b. Celkový počet slov
 - c. Celkový počet slovenských slov
 - d. Celkový počet anglických slov
 - e. Podiel anglických slov v percentuálnom vyjadrení
 - f. Dĺžku trvania nahrávky podľa časového vyjadrenia konca posledného dialógu prepisu danej nahrávky
 - g. Dĺžku trvania nahrávky získanú z nahrávky formátu *.wav*
 - h. Sumárne údaje za všetky spracované prepisy nahrávok

Údaje uvedené v bodoch *a.* až *g.* predstavujú stĺpce a údaje v bode *h.* predstavujú posledný riadok tabuľky.



Názov_prepisu_nahrávky	Celkový_počet_slov	Slovenských_slov	Anglických_slov	Podiel_anglických_slov
10 NAJ MAKEUPOV a BB KREMOV podTa mňa.vtt	1389	1285	104	7,49 %
Alexander Rybak - Interview on Slovakian TV-show 22.05.19.vtt	750	337	413	55,07 %
Caputova pre BBC.vtt	172	116	56	32,56 %
Celeste Buckingham answering fan questions in English and Slovak.vtt	2317	1047	1270	54,81 %
Chyba Conflicting types for function s Jurajom Mihalovom.vtt	353	319	34	9,63 %
Cyklus while a cyklus for s Danielom Hládekom.vtt	412	400	12	2,91 %
Doby svetové hitparády s Katy Perry!.vtt	123	50	73	59,35 %
ESET Mobile Security dokáže viac než som čakal (recenzia aplikácie).vtt	754	718	36	4,77 %
Foreigner trying to speak slovak.vtt	26	17	9	34,62 %
GIGANTICKÝ ZIMNÝ HAUL.vtt	2636	2487	149	5,65 %
How I learnt SLOVAK - TOP 5 Tips.vtt	97	34	63	64,95 %
Interview_ Slovakia - Switzerland - best player Slovakia - Women's U19 WFC.vtt	264	108	156	59,09 %
Learn Slovak 1_ Greetings.vtt	1044	78	966	92,53 %
Learn Slovak 2_ Courtesy phrases.vtt	2160	170	1990	92,13 %
Learn Slovak 3_ Introductions.vtt	808	141	667	82,55 %
Learn Slovak 4_ Days of the week.vtt	1037	125	912	87,95 %
Learn Slovak 5_ Numbers.vtt	821	116	705	85,87 %
Najlepší notebook za 300 _ Mišo Pročko Poskúša.vtt	503	473	30	5,96 %
predvolebná debata.vtt	200	114	86	43,00 %
Recept na Raw Vegan Brownies.vtt	200	193	7	3,50 %
Samsung Galaxy S9+ Najlepší mobil na trhu!! (RECENZIA).vtt	409	391	18	4,40 %
TELERANO - Pištu z Taiwanu očarilo Slovensko!.vtt	533	401	132	24,77 %
WISH fashion haul.vtt	289	279	10	3,46 %
ZAP - Prednáška č. 1 _ Week 1 (2019_2020).vtt	2765	2518	247	8,93 %
VSETKY PREPISY NAHRÁVOK	20062	11917	8145	40,60 %

Obr. 1 Ukážka časti výstupnej tabuľky v prostredí Windows PowerShell (z dôvodov čitateľnosti zobrazenia boli vynechané stĺpce vyjadrujúce dĺžku trvania videí)

2. Export údajov z tabuľky Windows PowerShellu do súboru *csv* formátu, pričom oddeľovačom je čiarka. Výsledný súbor s príponou *.csv* sa uloží do rovnakého priečinka, v ktorom sú prepisy nahrávok.

3. Súbor *csv* formátu je exportovaný do excelovského súboru s príponou *.xlsx* a uloží sa do rovnakého priečinka, v ktorom sú prepisy nahrávok a *.csv* súbor.

C. Výhody skriptu

Skript umožňuje spracovanie a vyhodnotenie ľubovoľného počtu získaných nahrávok a spracovaných prepisov. Výstupy sú naprogramované rovnako univerzálne bez obmedzení nielen počtu, ale aj tvorby názvov nahrávok, ktoré musia byť rovnaké ako názvy prepisov. Aj vo výslednej excelovskej forme vyhodnotenia sa jednotlivé grafické úpravy prispôbujú rozsahu vstupných informácií, nielen čo do počtu, ale aj do textovej dĺžky a spôsobu tvorby názvov.

Tab. 1 Zhodnotenie výsledkov štatistického vyhodnotenia prepisov nahrávok

Názov prepisu nahrávky	Celkový počet slov	Počet slovenských slov	Počet anglických slov	Podiel anglických slov na celkovom počte slov v %	Dĺžka trvania nahrávky podľa času konca posledného dialógu	Dĺžka trvania nahrávky vo formáte .wav
Prepis 1	1 389	1 285	104	7,49%	00:07:49.200	0:07:51
Prepis 2	750	337	413	55,07%	00:04:44.840	0:04:45
Prepis 3	172	116	56	32,56%	00:01:12.830	0:01:12
Prepis 4	2 317	1 047	1 270	54,81%	00:12:23.560	0:12:24
Prepis 5	353	319	34	9,63%	00:02:32.140	0:02:33
Prepis 6	412	400	12	2,91%	00:03:30.950	0:03:31
Prepis 7	123	50	73	59,35%	00:00:39.570	0:00:40
Prepis 8	754	718	36	4,77%	00:04:25.760	0:04:26
Prepis 9	26	17	9	34,62%	00:00:21.900	0:00:23
Prepis 10	2 636	2 487	149	5,65%	00:15:34.460	0:15:34
Prepis 11	97	34	63	64,95%	00:00:44.940	0:00:44
Prepis 12	264	108	156	59,09%	00:01:51.600	0:01:52
Prepis 13	1 044	78	966	92,53%	00:05:46.780	0:05:54
Prepis 14	2 160	170	1 990	92,13%	00:11:38.350	0:11:46
Prepis 15	808	141	667	82,55%	00:05:17.720	0:05:25
Prepis 16	1 037	125	912	87,95%	00:06:54.160	0:07:01
Prepis 17	821	116	705	85,87%	00:06:16.440	0:06:23
Prepis 18	503	473	30	5,96%	00:02:49.940	0:02:50
Prepis 19	200	114	86	43,00%	00:01:05.620	0:01:06
Prepis 20	200	193	7	3,50%	00:01:22.510	0:01:22
Prepis 21	409	391	18	4,40%	00:02:26.060	0:02:26
Prepis 22	533	401	132	24,77%	00:03:35.900	0:03:36
Prepis 23	289	279	10	3,46%	00:02:01.700	0:02:01
Prepis 24	2 765	2 518	247	8,93%	00:21:12.840	0:21:13
VŠETKY PREPISY	20 062	11 917	8 145	40,60%	02:06:19.770	2:06:58

Názvy prepisov v prvom stĺpci tabuľky číslo 1 sú uvedené v zjednodušenej forme kvôli svojej dĺžke a pre zachovanie prehľadnosti tabuľky. Sú to rovnaké názvy ako v prvom stĺpci tabuľky na obrázku číslo 4.

V. ZÁVER

Anotovali sme 24 nahrávok s celkovou dĺžkou trvania 2 hodiny 6 minút a 58 sekúnd. Všetky nahrávky sme spracovali celé. Rozdiel medzi výsledkami v posledných dvoch stĺpcoch v Tab. 1

je v tom, že po skončení posledného dialógu v nahrávkach väčšinou nasleduje záverečná znelka, potlesk alebo iná forma ukončenia bez hovorenej reči.

Celkový počet slov v spracovaných nahrávkach je 20 062, z čoho 11 917 je slovenských a 8 145 anglických. Z toho vyplýva, že podiel anglických slov je 40,60% za celú spracovanú databázu nahrávok v tejto práci.

Okrem objemu anotovaných dát aj výsledok pomeru anglických slov vo výške 40,6% hovorí o tom, že táto práca by mohla prispieť k rozšíreniu a skvalitneniu existujúcej slovensko-anglickej bilingválnej databázy. A tým prispieť k zvýšeniu WER bilingválneho DNN LVCSR s využitím modulu LID, čo by znamenalo zvýšenie percentuálnej úspešnosti rozpoznávania slov.

Užitočným výsledkom tejto práce je aj naprogramovaný skript, ktorý umožnil automatické štatistické vyhodnotenie anotovaných nahrávok a je využiteľný aj pre ľubovoľnú množinu ďalších anotácií pre databázu.

POĎAKOVANIE

Táto práca bola podporená Agentúrou na podporu výskumu a vývoja vďaka projektu APVV-15-0731 „Multimodálna interakcia človek-robot s využitím cloudových prostriedkov“, Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva vedy výskumu a športu Slovenskej republiky vďaka projektu číslo KEGA 009TUKE-4/2019 s názvom „Inovácia obsahu a príprava učebných textov pre predmet Biometrické systémy bezpečnosti“, Vedeckou grantovou agentúrou realizáciou výskumného projektu VEGA 1/0753/20 „Robustné rečové technológie metódami hlbokého učenia“ financovaného z prostriedkov Ministerstva školstva, vedy, výskumu a športu SR a nakoniec aj cez Operačný program Výskum a inovácie, financovaného z Európskeho fondu regionálneho rozvoja v rámci projektu „Univerzitný vedecký park TECHNICOM pre inovačné aplikácie s podporou znalostných technológií – II. fáza“ Kód ITMS: 313011D232.

REFERENCIE

- [1] Pleva, Matus, Yuan-Fu Liao, Daniel Hladek, Jan Stas, Martin Lojka, Jozef Juhar, and Stanislav Ondas. "Automatic Recognition of Slovak-English Bilingual Speech." In *ITAT - 19th Conference Information Technologies - Applications and Theory*, pp. 176-179. 2019. [cit.2020-03-28]. Dostupné na internete: <<http://ceur-ws.org/Vol-2473/paper28.pdf>>.
- [2] Khullar, P.: *Difference between Code Mixing and Code Switching*[online].2018 [cit.2020-03-28]. Dostupné na internete: <<http://languagelinguistics.com/2018/02/06/difference-code-mixing-code-switching/>>.
- [3] Seçkin, E.: *Code Switching: Definition, Types, and Examples* Hacettepe University, 2019[cit.2020-03-28]. Dostupné na internete: <https://owlcation.com/humanities/Code-Switching-Definition-Types-and-Examples-of-Code-Switching>
- [4] Basnight-Brown, D.M. and Altarriba, J., 2007. *Code-switching and code-mixing in bilinguals: Cognitive, developmental, and empirical approaches*. [online]. Department of Psychology University at Albany, State University of New York, 2007 [cit.2020-03-28]. Dostupné na internete: <https://www.researchgate.net/publication/287788250_Code-switching_and_code-mixing_in_bilinguals_Cognitive_developmental_and_empirical_approaches>.
- [5] S. Gereg, P. Vizslay, J. Staš and M. Lojka, "Semi-automatic processing and annotation of meeting audio recordings," *2019 17th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, Starý Smokovec, Slovakia, 2019, pp. 204-209, doi: [10.1109/ICETA48886.2019.9040119](https://doi.org/10.1109/ICETA48886.2019.9040119).
- [6] Lojka M., Vizslay P., Staš J., Hládek D., Juhár J. (2019) *Slovak Broadcast News Speech Recognition and Transcription System*. In: Barolli L., Kryvinska N., Enokido T., Takizawa M. (eds) *Advances in Network-Based Information Systems*. NBiS 2018. Lecture Notes on Data Engineering and Communications Technologies, vol 22. Springer, Cham, pp 385-394. https://doi.org/10.1007/978-3-319-98530-5_32
- [7] Pleva, M., Ondas, S., Hládek, D., Juhar, J., Staš, J. and Liao, Y.F., *Building of children speech corpus for improving automatic subtitling services*. ROCLING 2019, New Taipei City, ACLCLP, pp. 325-333. <http://www.aclclp.org.tw/rocling/2019/O19-1030.pdf>

Robotický operačný systém pre vytváranie hier jedného hráča

¹Erik KLEMBARA, ²Marián MACH

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹erik.klembara@student.tuke.sk, ²marian.mach@tuke.sk

Abstrakt — Cieľom článku je objasniť problematiku vytvárania hier jedného hráča v prostredí Robotického operačného systému. Pozornosť je zameraná na vytvorenie virtuálneho prostredia s uzlami pre ovládanie herných prvkov s implementáciou mechaník logickej hry Sokoban. Článok sa zaoberá možnosťou využitia Sokoban riešiaceho modulu ako jadra plánovacieho systému a spôsobom prípravy vstupných dát, definujúcich mapu hry, pre tento modul vo forme textovej reprezentácie. V rámci experimentovania boli vytvorené skripty pre generovanie konfiguračných súborov a implementáciu herných mechaník pre dané virtuálne prostredie robotickej simulácie na základe špecifikácií užívateľa. Experimenty boli zamerané na overenie funkčnosti riešenia a vhodnosti prostredia Robotického informačného systému pre účely vytvárania hier.

Kľúčové slová — Gazebo, prehľadávanie do hĺbky, Robotický operačný systém, Sokoban

I. ÚVOD

Robotický operačný systém (ROS) nie je v skutočnosti klasický operačný systém, ale skôr open-source meta-operačný systém, ktorý poskytuje hardvérovú abstrakciu [1][2]. Použitie ROSu nie je limitované len na robotiku, keďže väčšina nástrojov ROSu je kompatibilných s viacerými periférnymi zariadeniami a tak ROS môže byť použitý pre riešenie rôznych úloh, napríklad aj pre vytvorenie hry Sokoban [3]. V tejto hre má hráč za úlohu dotlačiť jednotlivé krabice do vyznačených cieľových pozícií ich postupným tlačením z iníciačných pozícií. Môže sa pohybovať v jednom zo štyroch možných smerov, pričom však musí rešpektovať prekážky (steny a tie krabice, ktoré sú blokované stenami alebo inými krabicami).

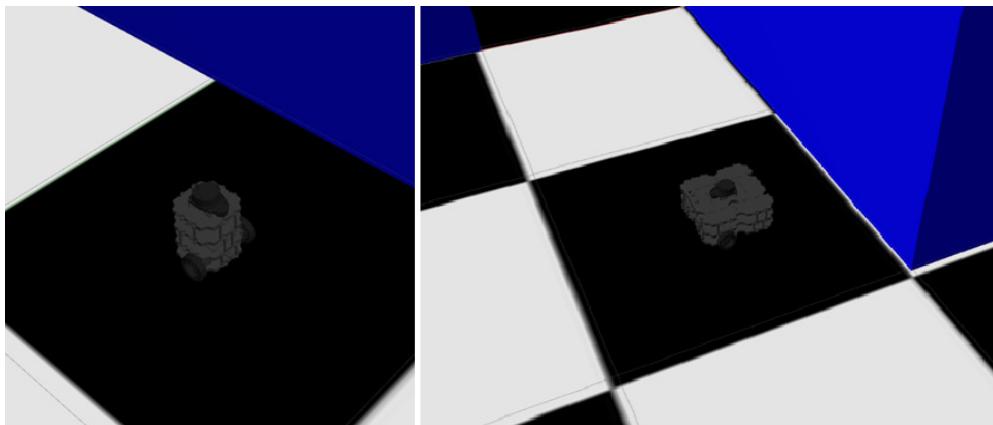
Využitie systému ROS v implementácii hier sľubuje jednoduchší prenos týchto hier do robotického sveta, čo na jednej strane umožňuje doplniť tieto hry atraktívnym vizuálnym rozhraním a na druhej strane rozširuje doménu v ktorej je možné testovať vlastnosti „robotických“ algoritmov aj v inom než typickom prostredí.

Pri implementácii hry Sokoban do prostredia ROSu je možné použiť viacero rôznych prístupov. Jedným z nich je použitie simulačných prostriedkov ako je Gazebo na vizualizáciu samotnej hry a balíkov vopred pripravených robotov, na reprezentovanie herných prvkov.

II. GENEROVANIE VIRTUÁLNEHO PROSTREDIA

Pred tým než bolo možné začať experimentovať a implementovať herné mechaniky hry Sokoban, bolo nutné určiť reprezentáciu jednotlivých prvkov hry. Hráč a krabice boli reprezentované robotmi – presnejšie dvomi verziami robota TurtleBot3, ktoré môžeme vidieť na obr. 1. Steny boli reprezentované modrými kockami, ktoré sme ukladali na jednotlivé políčka podľa špecifikácie mapy. Cieľové destinácie boli reprezentované zelenými valcami, ktoré boli dostatočne nízke na to aby akýkoľvek robot mohol na nich bezproblémovo zastaviť.

Špecifikácie umiestnenia stien, cieľových pozícií a pozícií robotov boli definované v zdrojovom súbore, ktorý využíval ROS na generovanie simulácie sveta. Všetky špecifikácie jednotlivých objektov mali určené vlastnosti relatívne k ich pozícií v tomto textovom súbore, ktorý predstavoval zdroj pre skript na generovanie prostredia vo formáte, ktorý požaduje simulátor Gazebo. Formát zdrojového textového súboru môžeme vidieť na obr. 2.



Obr. 1 Vľavo robot hráča (TurtleBot3 Burger), vpravo robot krabice (TurtleBot3 Waffle)

```
#####  
#####P.#  
#.....#  
#...CC.G#  
#.#...###  
##G...###  
#####
```

Obr. 2 Formát zdrojového súboru pre generovanie sveta

Robotov reprezentujúcich hráča a krabice bolo potrebné ovládať pomocou ROS uzlov a teda to neboli statické objekty. Ich vytváranie do simulovaného prostredia sa definovalo v spúšťacom súbore, kde na vytvorenie ich inštancií a vloženie do sveta na požadovanú pozíciu boli použité príslušné balíky, ktoré sú už predpripravené v ROSe.

Skript po vykonaní všetkých operácií vygeneroval dva súbory na základe špecifikácií zdrojového textového súboru, pričom použil základné princípy objektovo orientovaného programovania vo forme definície tried pre každý herný prvok a pri generovaní týchto súborov prispôbil všetky pozície herných prvkov, korešpondujúc súradnicovému systému Gazeba.

III. POHYB ROBOTA A KRABÍC

Pohyb robota hráča zabezpečoval uzol, ktorý obsahoval metódy pre základný pohyb robota vpred a otočenie doprava alebo doľava. Na základe týchto metód boli vytvorené metódy vyššej abstrakcie, ktoré zabezpečovali vykonanie kroku. Krok spočíval v prekonaní určitej vzdialenosti, ktorá odpovedala vzdialenosti medzi stredmi okolitých herných políčok v určitom smere. Pred vykonaním kroku ovládací uzol zistil aktuálny smer natočenia robota, vykonal kontrolu kolízie a zavolať metódy pre otáčanie doprava alebo doľava podľa parametru aktuálnej hodnoty natočenia. Kontrola kolízie spočívala v zistení prítomnosti robota krabice na políčku, ktoré predstavovalo cieľovú destináciu nasledujúceho kroku hráča. Ak sa na políčku nachádzala krabica, uzol hráča publikoval správu na tému, ktorú odoberal uzol zodpovedný za ovládanie krabíc. Uzol, ktorý správu obdržal, ju spracoval a inicializoval vykonanie príslušného kroku pre danú krabicu tak, aby výsledná pozícia krabice bola vzdialená o jedno políčko v smere nasledujúceho kroku robota.

Pohyb robotov krabíc zabezpečoval druhý uzol, ktorý sa staral o všetky krabice v rámci hry. Metódy pre pohyb, otáčanie a vykonanie kroku boli takmer totožné s metódami uzla hráča. Hlavný rozdiel bol v tom, že metódy uzla krabíc boli volané s parametrom typu, ktorý predstavoval triedu krabice. To umožnilo opätovnú použiteľnosť metód pre každý objekt typu krabice. Uzol krabíc udržiaval každú inštanciu objektu krabice v zozname. Každý objekt mal vlastnosti ako aktuálna pozícia v osi x , y a hodnotu natočenia. Taktiež mali vlastnú metódu,

ktorá odoberala správu odometrie pre daný aktuálny objekt. Pre pohyb publikovali uhlovú a lineárnu rýchlosť na príslušnú tému, ktorá bola priradená danému robotovi reprezentujúcemu konkrétnu krabicu.

IV. VYKONÁVANIE PREDPRIPRAVENÉHO PLÁNU

Po implementácii pohybu hráča, krabíc, interakcie medzi nimi a generovania prostredia nasledovala implementácia Sokoban riešiacieho modulu. Modul generoval riešenia v podobe krokov, ktoré následne vykonával robot hráča. Generovanie pozostávalo z prečítania zdrojového textového súboru mapy, prehľadávania priestora stavov, nájdenia riešenia a následného zápisu tohto riešenia formou krokov do textového súboru, ktorý neskôr použil uzol hráča pre automatické hranie hry. Bežnou metódou riešenia takýchto úloh je prehľadávanie stavového priestoru, kde stavovým priestorom sa rozumie množina všetkých stavov, do ktorých je možné sa dostať z počiatočného stavu postupnosťou zostavenou z možných ťahov. Toto prehľadávanie je možné implementovať viacerými spôsobmi. My sme použili metódu prehľadávania do hĺbky. Pri tejto metodike prehľadávania sa expanduje koreň s najväčšou hĺbkou [4].

Uzol hráča bol rozšírený o metódu automatického vykonania predpripraveného plánu pre aktuálnu mapu. Uzol prečítal všetky riadky súboru, ktorý vygeneroval Sokoban riešiaci modul, pričom jednotlivé slová rozdelil podľa prázdných znakov a slová pridal do zoznamu. Každé slovo predstavovalo jeden krok pre hráča. Metóda iterovala zoznamom týchto krokov a volala zodpovedajúce metódy pre jednotlivé kroky.

V. EXPERIMENTY

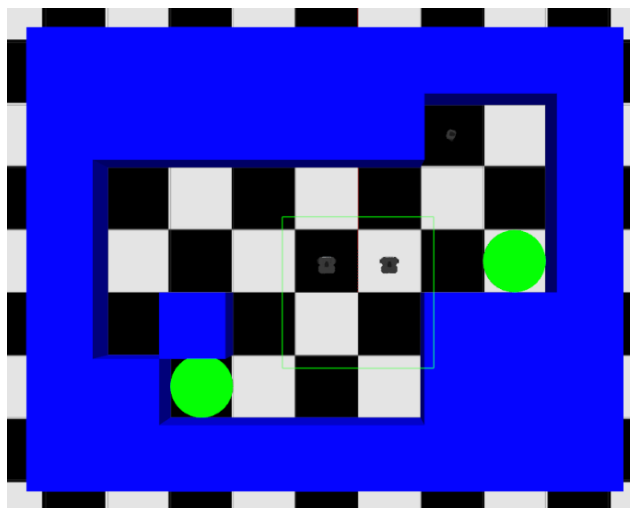
Pre účely experimentov [5] boli vytvorené dve mapy s rôznym rozmiestneným herných prvkov. Cieľom experimentov bolo otestovať funkcionality, presnosť a úspešnosť implementácie jednotlivých mechanizmov hry.

Pre prvý experiment sme si zvolili dve krabice a dve cieľové destinácie, pričom hráč začínal na hornej hranici sveta. Textový súbor mapy môžeme vidieť na obr. 3.

```
#####  
#####P.#  
#.....#  
#...CC.G#  
#.#...###  
##G...###  
#####
```

Obr. 3 Súbor mapy prvého experimentu

Po vygenerovaní sveta zo súboru mapy bol k dispozícii simulovaný svet zobrazený na obr. 4. Generovanie sveta bolo úspešné. Objekty a roboti sa nachádzali na požadovaných pozíciách.



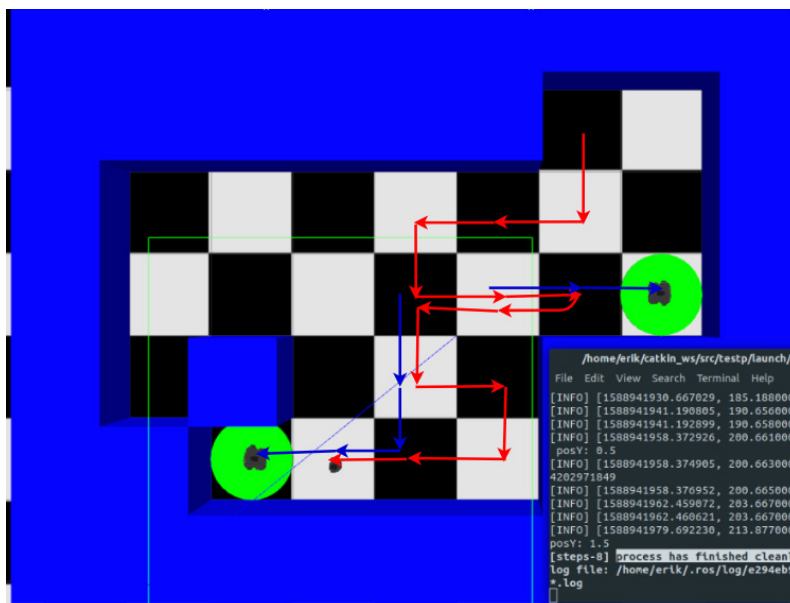
Obr. 4 Vygenerovaný svet prvého experimentu

Pre túto mapu prvého experimentu algoritmus hľadania riešenia vygeneroval sériu krokov v počte 13, ktoré sú ilustrované na obr. 5.

```
<SOLUTION FOUND!>
()
('Number of steps needed: ', 13)
()
['down', 'left', 'left', 'down', 'right', 'right', 'left', 'left', 'down', 'right', 'down', 'left', 'left']
```

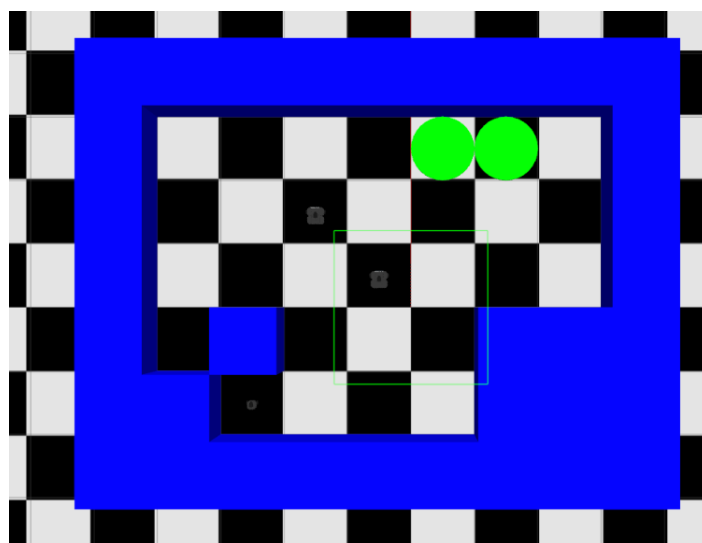
Obr. 5 Vygenerované riešenie prvého experimentu

Vykonanie všetkých krokov a konečný stav hracích prvkov je zobrazený na obr. 6.



Obr. 6 Vyriešený prvý experiment (červené šípky – kroky hráča, modré šípky – kroky krabic)

Robot hráča bez problémov vykonal všetky vygenerované kroky a splnil prvú hraciu mapu. Na druhý experiment bola vytvorená zmenená mapa opäť s dvoma krabicami a dvoma cieľovými destináciami. Kvôli testovaniu pohybu robota hráča bol tento tentoraz umiestnený v dolnej oblasti. Vygenerovaný simulovaný svet je zobrazený na obr. 7.



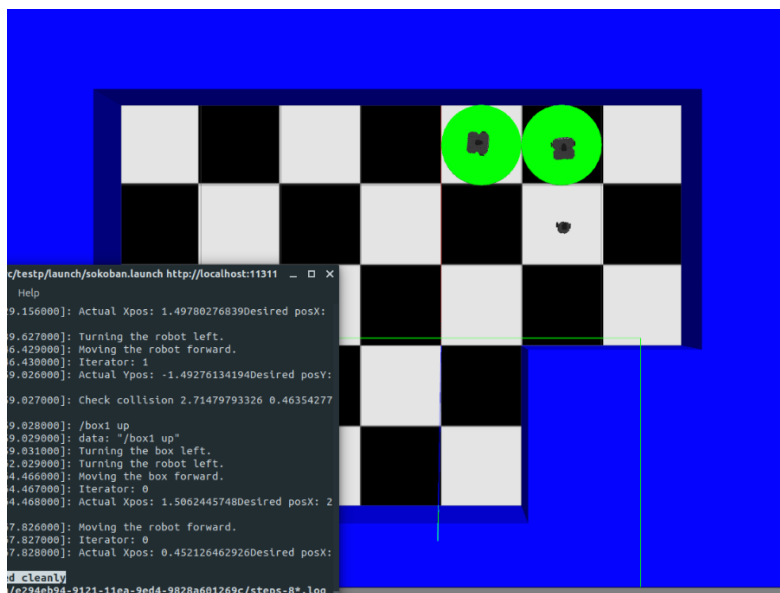
Obr. 7 Vygenerovaný svet druhého experimentu

Riešenie vygenerované Sokoban modulom pre druhý experiment, pozostávajúce z o niečo viac krokov, je zobrazené na obr. 8.

```
<SOLUTION FOUND!>
()
('Number of steps needed: ', 17)
()
['right', 'right', 'up', 'up', 'left', 'up', 'right', 'left', 'left', 'up', 'right', 'right', 'down', 'right', 'down', 'right', 'up']
```

Obr. 8 Vygenerované riešenie druhého experimentu

Robot hráča aj teraz bez problémov vykonal všetky vygenerované kroky a splnil druhú hraciu mapu, ktorej výsledná podoba je zobrazená na obr. 9. Krabice reagovali na správy pre posun a robot hráča dokázal postupne vykonávať jednotlivé predpripravené kroky.



Obr. 9 Vyriešený druhý experiment

VI. ZÁVER

Na základe realizovaných experimentov je možné konštatovať, že využitie Robotického operačného systému pri vytváraní hier jedného hráča je možné, avšak takýto prístup vyžaduje veľmi dobre systémové špecifikácie pracovnej stanice, kvôli svojej výpočtovej náročnosti. Počas riešenia sme sa však stretli s niekoľkými problémami, sťažujúcimi samotnú realizáciu. Najvypuklejším prípadom bolo presné smerovanie pohybu robota, ktoré bolo často odlišné od očakávaného – aj keď odchýlka bola menšia, integrovaní pohybu robota vo výsledku narastala do neprijateľnej miery a musela byť kompenzovaná korekčnými pohybmi.

Výhodou je veľmi realistické fyzikálne jadro Gazeba a možnosť škálovania procesov v ROSe. Nevýhodou je previazanosť simulačnej presnosti a rýchlosti vykonávania simulácie s výkonom pracovnej stanice, na ktorej je simulácia vykonávaná. Vytváranie hier jedného hráča v prostredí Robotického operačného systému si vyžaduje značnú mieru optimalizácie.

LITERATÚRA

- [1] Morgan Quigley, Brian Gerkey & William D. Smart: Programming Robots with ROS, O'Reilly Media, Inc, USA, 2015.
- [2] Anil Mahtani, Luis Sánchez, Enrique Fernández, Aaron Martinez : Effective Robotics Programming with ROS Third Edition, Packt Publishing, 2016.
- [3] Sokoban: The beautiful world of remodels. Available: <https://sokoban-jd.blogspot.com>
- [4] Russel S., Norvig P.: Artificial Intelligence: A Modern Approach, Third Edition, Pearson, 2009.
- [5] Erik Klembara: Použitie Robotického operačného systému pre vytváranie hier jedného hráča. Diplomová práca, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, 2020, 62 strán.

Výber bezpečných prenosových uzlov pre hybridnú MANET-DTN sieť

¹Richard NEZNIK, ²Jan PAPAĽ

^{1,2}Katedra Elektroniky a Multimediálnych Telekomunikácií,
Fakulta Elektrotechniky a Informatiky Technická Univerzita Košice,
Slovenská Republika

¹richard.neznik@student.tuke.sk, ²jan.papaj@tuke.sk

Abstract – Tento článok sa venuje problematike výberu bezpečných prenosových uzlov prostredníctvom teórie hier. Výber bezpečných uzlov využíva kooperatívnu a nekooperatívnu hru. Popisuje porovnanie výhod a nevýhod týchto hier pri tvorbe bezpečnej cesty na prenos dát zo zdrojového do cieľového zariadenia pri použití rovnakej topológie. Využitie tohto spôsobu prenosu je možné nielen v MANET sieťach, ale aj v iných bezdrôtových sieťach, zameraných na bezpečnosť.

Keywords – 5G siete, bezpečnosť, spolupráca, MANET, Teória hier.

I. ÚVOD

Postupom času sa začal využívať internet častejšie a prenos dát bol rýchlejší. Keď si uvedomíme, aké máme dnes možnosti, tak je to priam neuveriteľné. Hocičo si môžeme behom chvíle vyhľadať na internete cez naše smartfóny, tablety atď. Každým rokom sa technológie v oblasti IT neustále posúvajú dopredu. Žiadnou výnimkou nie je ani oblasť počítačových sietí. Stále viac a viac zariadení sa pripája na internet a v budúcnosti bude táto technológia využívaná všade, kde to bude možné. Bezpečnosť je veľmi dôležitým faktorom a mali by sme na ňu klást vysoký dôraz. O to viac, keď sa bavíme o bezdrôtovom prenose dát cez internet, tam to platí dvojnásobne. Ak chceme, aby veci fungovali tak ako majú, musia byť zabezpečené proti útokom zvonku. Práve 5G siete predstavujú budúcnosť vo svete internetu. Tieto siete budú využívané primárne na zabezpečenie plynulého chodu Internetu vecí (Internet of Things – IoT).

II. BEZPEČNOSŤ PRE VIACPRESKOKOVÉ MOBILNÉ SIEŤE

Viacpreskokové mobilné siete nie sú orientované na bezpečnosť, preto je potrebné neustále napredovať a vyvíjať nové technológie a spôsoby na ochranu týchto sietí. Mobilné viacpreskokové siete (Mobile Ad-hoc Network – MANET) predstavuje výzvu pre riešenia v oblasti bezpečnosti. V poslednom desaťročí sa výskumníci snažili vyvinúť nové riešenia, ktoré by sa dali uplatniť pre siete MANET. Veľké množstvo existujúcich zabezpečení bolo neefektívnych a nedali sa aplikovať pre mnohé nasadené prostredia MANET. Pretože mnohé smerovacie protokoly neberú do úvahy bezpečnosť, niektoré výskumy sa zameriavajú na vývoj bezpečného smerovania protokolov alebo zavedeniu bezpečnostných rozšírení existujúcich smerovacích protokolov. Práve tieto smerovacie protokoly boli navrhnuté na boj proti tzv. sebeckým aktivitám (selfish activities) tým, že prinútili tieto uzly medzi sebou spolupracovať [1], [2]. Spôsoby implementácie zabezpečenia sa líšia. Čím je väčšia vzdialenosť medzi uzlami, tým je nebezpečenstvo vzniku útokov väčšie. Na druhej strane čím je vzdialenosť kratšia, tak tým sa logicky toto riziko zasa znižuje. Pri krátkom kontakte susedov dokážu títo susedia medzi sebou prenášať niektoré tajné informácie (napríklad tajné kľúče). To by zvýšilo úroveň bezpečnosti, pretože fyzické komunikačné linky sú bezpečnejšie ako bezdrôtové linky.

Používaním bezdrôtovej komunikácie je spôsobené to, že sieť je náchylná na útoky ako aktívne rušenie a odposluch. Na vykonanie útokov nie je potrebný prístup k sieti. Keďže bezdrôtové siete majú zvyčajne nižšie šírky pásma, čo využívajú útočníci, ktorí bránia komunikácií medzi danými uzlami. Útočník môže generovať umelú prevádzku na danej frekvencii kanála a tým docieľi využívanie plnej kapacity prenosového kanála. To vedie k neschopnosti komunikácie medzi zariadeniami, teda nie je dostupná kapacita na prenos dát

medzi komunikujúcimi zariadeniami. Uzly v MANET sieťach môžu opustiť sieť a následne sa znova pripojiť k sieti, tak isto sa môžu nezávisle presúvať. Nie je vôbec jednoduché rozlíšiť normálne správanie siete od správania škodlivého. Uzly, ktoré majú nedostatočnú fyzickú ochranu môžu byť často vystavované riziku. MANET siete nemajú obrannú líniu. To znamená, že útoky môžu prísť zo všetkých smerov. Hranica, ktorá delí sieť na vnútornú a vonkajšiu nie je jasne definovaná. Z tohto hľadiska teda nie je jasne určené miesto, kde by bolo jednoduché sledovať systémy a mechanizmy kontroly prístupov. Ďalšou zraniteľnou oblasťou je obmedzenie zdrojov [2].

Bezpečnostné ciele v týchto sieťach nie sú až tak odlišné oproti iným sieťam. Nakoľko MANET využíva bezdrôtovú komunikáciu, bude princíp zabezpečenia podobný ako pri Wi-Fi sieťach. Keďže MANET siete nevyužívajú router ako nadradený prvok, je potrebné, aby sa toto zabezpečenie vykonávalo na každom jednom zariadení samostatne. *Autentifikácia* je proces, na overenie tvrdení o totožnosti daného zdroja informácií. Bez overenia by tak mohol ľahko preniknúť neautorizovaný uzol, ktorý by začal používať dostupné zdroje bez siete. *Dôveryhodnosť* dovoľuje získať prístup k informáciám, obsiahnutých v prenášaných dátach iba oprávneným osobám alebo systémom, ktoré sú chránené. Slúži hlavne na to, aby sa predišlo úniku dôverných informácií. *Integrita* slúži na ochranu, aby informácie neboli upravené, poprípade poškodzované cudzími používateľmi, ktorí nemajú oprávnenie na ich použitie. *Dostupnosť* je schopnosť siete zabezpečovať služby, ktoré sa od nej očakávajú. *Neodvolateľnosť* je využívaná pre to, aby nebolo možné zamietnuť dané akcie. Medzi najväčšie hrozby patria jednoznačne útoky formou zahltenia (Denial of Service – DoS). Napríklad vo vojenskom prostredí je tento typ útoku veľmi nebezpečný [2].

III. TEÓRIA HIER

Teória hier je odvetvím v aplikovanej matematike. Poskytuje formálny rámec so súborom matematických nástrojov na štúdium zložitých interakcií medzi hráčmi, ktorí si vyberajú racionálne stratégie. Používa modely na skúmanie interakcií daných hier. Skúma očakávané a skutočné správanie jednotlivcov v konkrétnych hrách. Teória hier je matematický model, pomocou ktorého sa snažíme dosiahnuť taký kompromis, aby bolo čo najmenej ľuďom uškodené. V dosť veľkej miere sa práve toto pravidlo používa aj v informatike.

A. Kooperatívne a nekooperatívne hry

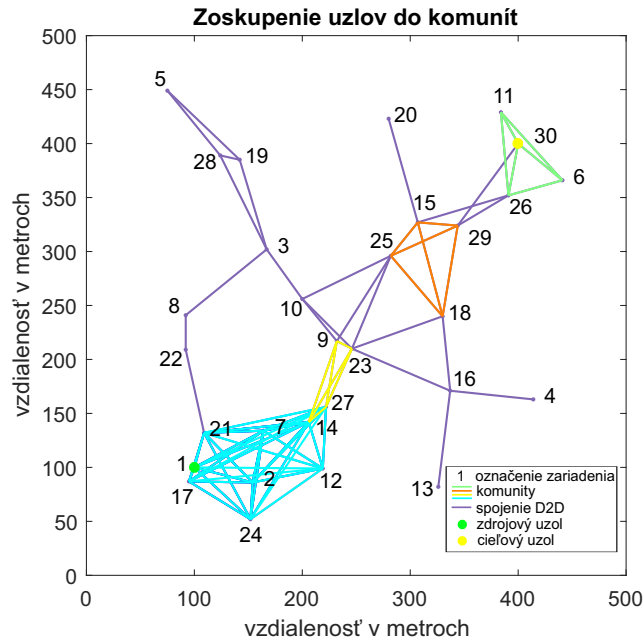
Kooperatívna hra je taká hra, pri ktorej sú hráči schopní urobiť, poprípade vytvoriť medzi sebou dohodu, ktorú nazývame koalícia. „V tomto type hry sú hráči vo všeobecnosti v úplnej vzájomnej komunikácii s ostatnými hráčmi a majú mechanizmy, ktoré zabezpečia vykonávanie všetkých uzavretých dohôd“ [3]. Spoločným cieľom tejto hry je nájsť taký výsledok, ktorý bude kolektívne optimalizovať výsledky pre všetkých zúčastnených hráčov.

Nekooperatívna hra je taká hra, pri ktorej hráči nie sú schopní vytvoriť takúto koalíciu. „Hra je nekooperatívna, ak záväzky (dohody, prísluby alebo hrozby nie sú vymožitelné)“ [4]. To znamená, že každý hráč sa správa nezávisle voči ostatným hráčom. Inými slovami sa každý hráč zameriava na svoje vlastné záujmy. Pri nekooperatívnych hrách sú všetky stratégie pre hráčov preddefinované dopredu. „Nekooperatívna hra sa označuje ako statická hra, kde hráči robia ťahy iba raz v rovnaký čas, alebo dynamická hra, kde títo hráči vyberajú svoje ťahy počas hry“ [4]. V týchto hrách sa teda hráči nemôžu dohodnúť na vytvorení koalície, ale môžu sa dohodnúť na výsledku hry. Samozrejme za predpokladu, že nevyvíjajú obmedzujúce dohody. Pri nekooperatívnych hrách poznáme 2 formy. Prvou z nich je klasická hra. Druhou z nich je rozšírená hra[5].

IV. NÁVRH METÓDY PRE VÝBER BEZPEČNÉHO UZLA VYUŽITÍM TEÓRIE HIER

Pod pojmom bezpečný uzol rozumieme uzol, ktorý je považovaný za dôveryhodný a využije sa na prenos dát z hľadiska bezpečnosti. Cieľom takéhoto prenosu je docieľiť to, aby sa dáta alebo správa dostali zo zdrojového do cieľového uzla. Neberieme však do úvahy počet krokov, čas, či iné veličiny, ktoré by nás zaujímali pri iných typoch prenosov. Bezpečný prenos neznamená to, že prenos musí byť vytvorený za každú cenu. Pri tomto výbere sa riadim výhradne bezpečnostnými parametrami. Bezpečný výber uzlov využíva dva typy hier, ktorými sú kooperatívna a nekooperatívna hra.

Nekooperatívna hra (n-person game) využíva na výber bezpečného uzla dva parametre. Prvým parametrom je počet stratégií pre každého jedného hráča. Druhým parametrom je kombinácia histórie kontaktu a pomeru prijatých a odoslaných paketov. Výsledkom tejto



Obr. 1 Hráči a ich možnosti výberu spolupráce k ostatným hráčom.

Tabuľka 1
 Parametre simulácie

Simulačná plocha [m]	500 x 500
Rádiový dosah [m]	100
Uzly [počet]	30

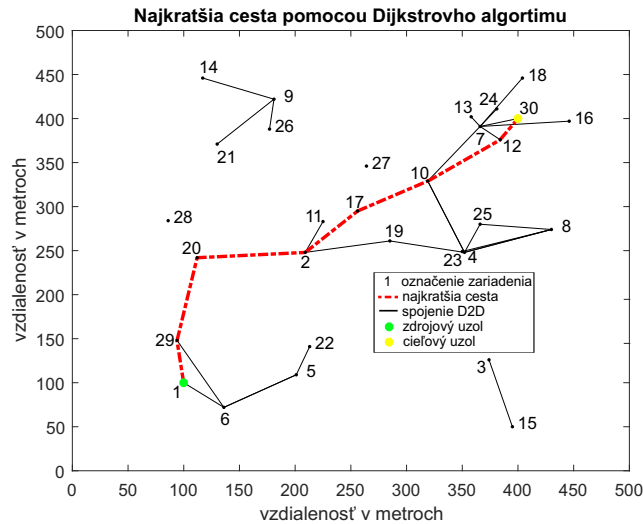
hry je zisk, počet iterácií a chybovosť. Počet hráčov je v tejto hre určený počtom susedov pre každý jeden uzol. Nasledujúcim parametrom, ktorý prichádza do procesu výberu bezpečného uzla, je dôveryhodnosť jednotlivých uzlov.

Pri kooperatívnej hre (zero-sum game) sa ako vstupný parameter na výber bezpečného uzla použije pomer prijatých a odoslaných paketov. Výstupom z tejto hry je zmiešaná stratégia. Rovnaký proces sa vykoná s parametrom história kontaktu. Výslednou hodnotou z kooperatívnej hry je zmiešaná stratégia získaná z dvoch parametrov. Pri výbere bezpečného uzla sa proces zameriava na také uzly, ktoré majú najlepšiu hodnotu stratégie pre dané spojenie. Počet susedov zariadenia znamená počet hráčov, ktorí sa podieľajú pri procese výberu bezpečného uzla. Výber bezpečného uzla funguje na podobnom princípe ako pri nekooperatívnej hre, no je tam pár zmien. Zmenou je vyhodnocovanie vhodných uzlov na prenos využitím jednotlivých komunit. Komunity sú tvorené pomocou algoritmu k-clique. Na Obr. 1 sú zobrazené jednotlivé komunity.

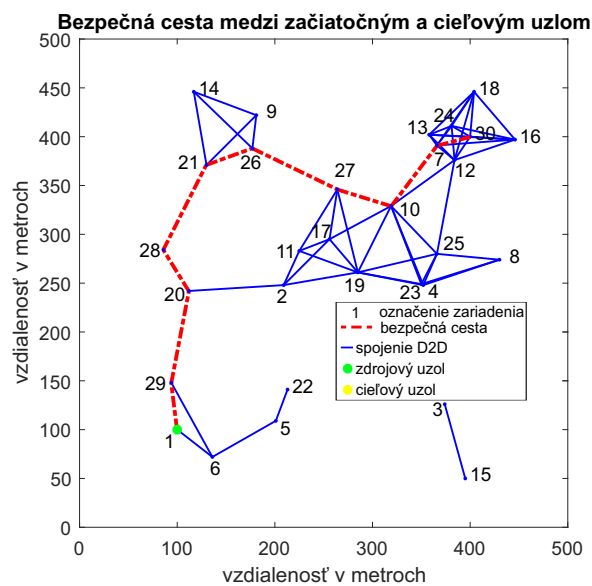
V. SIMULÁCIA A VÝSLEDKY

Simulácia na overenie navrhovaných metód prebieha pomocou statickej topológie, ktorá je vygenerovaná na skúmanej ploche náhodným spôsobom. Pre overenie vhodnosti použitia hier pre bezpečný výber som si vo svojej práci vybral simulačné prostredie MATLAB. Zdrojový uzol v simuláciách predstavuje vždy uzol s číslom 1. Jeho pozícia je fixne zadaná. Cieľový uzol je uzol s označením 30, pričom ide o uzol, ktorý je získaný z parametra *počet_uzlov*. Jeho poloha je tak isto fixne zadaná.

V oboch hrách využívam pre tvorenie komunit k-clique algoritmus. V skratke funguje tento algoritmus tak, že na vstupe sa zadávajú dva parametre. Prvým parametrom je veľkosť grafu, teda počet spojení na vytvorenie komunity uzlov označený ako *community_links* a druhým vstupom je matica *spojenie_uzlov*. Na vytvorenie komunity využívam minimálne 4 uzly, také, ktoré sú v spojení s každým uzlom danej komunity. Teda ak máme komunitu o veľkosti $k = 4$, tak každý sused je v spojení s 3 zvyšnými uzlami tejto komunity. Komunitu tak budú tvoriť zoskupenia uzlov, ktoré sú vytvorené z minimálne 4 a viac uzlov. Na Obr. 1 môžeme vidieť farebne odlišné rôzne komunity. Tieto komunity simulujú hybridnú MANET-DTN sieť. Výber bezpečných uzlov funguje na princípe, že pri nekooperatívnej hre je vždy zvolený taký uzol, ktorý má najlepší zisk a zároveň jeho hodnota dôveryhodnosti nie je

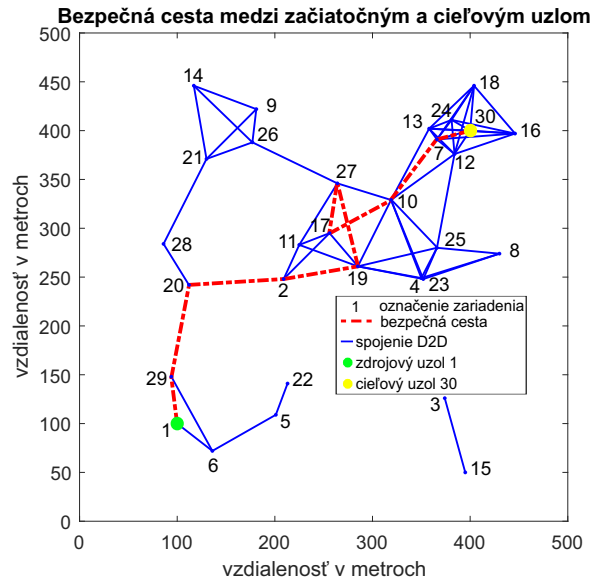


Obr. 2 Najkratšia cesta vytvorená pomocou Dijkstrovhho algoritmu pri rovnakej topológii.

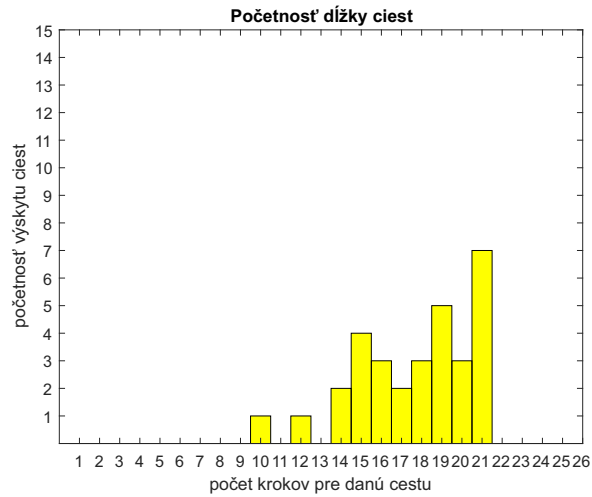


Obr. 3 Najkratšia cesta vytvorená pomocou navrhnutého algoritmu nekooperatívnej hry pri rovnakej topológii.

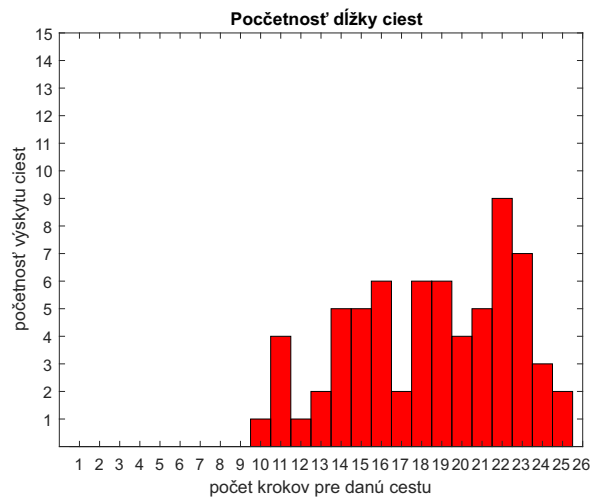
záporná (teda je 0 alebo 1), tak takýto uzol sa zvolí za bezpečný. Pokiaľ by nastala situácia, že niektorá dvojica susedov má rovnaký prínos, ale ich hodnota dôveryhodnosti je 0 a 1, tak sa vyberie ten, ktorého hodnota je 1. Uzol, ktorý už bol raz použitý, sa označí a zníži sa jeho hodnota zisku. Pri ďalšom použití takéhoto uzla sa jeho prínos zníži ešte viac a viackrát už takýto uzol nebude možné použiť na prenos dát. Takýmto spôsobom prejdeme všetky uzly a následne sa vytvorí bezpečná cesta na prenos zo zdrojového do koncového uzla. Keď nastane situácia, že žiaden zo susedov nie je dôveryhodný na prenos, tak sa cesta nevytvorí. Pri kooperatívnej hre je princíp výberu podobný, akurát sa tu ešte vyhodnocujú informácie, ktoré uzly zozbierajú od susedov. Prípadne informácie o príslušnosti jednotlivých uzlov v komunitách. Pre oba navrhované algoritmy som si zvolil na porovnanie ich najkratšie cesty. Cesty porovnávam s Dijkstrovým algoritmom pre najkratšiu cestu. Nájdenie najkratšej cesty pomocou Dijkstra algoritmu je zobrazený na Obr. 2. Najkratšia cesta pomocou algoritmu pre nekooperatívnu hru zobrazuje Obr. 3. Obr. 4 znázorňuje nájdenie najkratšej cesty pre kooperatívnu hru. Nekooperatívna hra nevyužíva niektoré informácie dostupné od svojich susedov a preto je počet nájdených ciest nižší. Oproti tomu kooperatívna hra okrem stratégie hry, využíva aj tieto informácie a tým zvyšuje možnosť nájdenia bezpečnej cesty. Podľa obrázkov môžeme vidieť, že rozdiel medzi týmito dvoma hrami je takmer dvojnásobný. Pri nekooperatívnej hre našiel algoritmus 31 bezpečných ciest, no pri kooperatívnej hre našiel algoritmus 68 bezpečných ciest. Tieto početnosti sú zobrazené na Obr. 5 a Obr. 6.



Obr. 4 Najkratšia cesta vytvorená pomocou navrhnutého algoritmu kooperatívnej hry pri rovnakej topológii.



Obr. 5 Početnosť nájdených a nenájdených ciest pre nekooperatívnu hru s rovnakou topológiou.



Obr. 6 Početnosť nájdených a nenájdených ciest pre kooperatívnu hru s rovnakou topológiou.

VI. ZÁVER

Bezpečný výber prenosových uzlov pre kooperatívnu aj nekooperatívnu hru jasne ukázal, aké sú rozdiely pri oboch navrhovaných algoritmoch. Kým pri nekooperatívnej hre sa tento výber riadil bezprostredne len parametrami ako dôveryhodnosť uzlov, história kontaktu, či pomer prijatých a odoslaných paketov, tak pri kooperatívnej hre bol tento výber rozšírený o informácie získané z komunit. Pri simuláciách sa síce ukázalo, že kooperatívna hra dokáže nájsť oveľa väčší počet ciest, no zaujímavým faktorom bolo aj to, že tieto cesty boli oveľa dlhšie ako pri hre nekooperatívnej. Tento faktor síce nepovažujem za dôležitý z hľadiska bezpečného výberu. Pri dlhodobom simulovaní oboch algoritmov som zistil, že oba navrhované algoritmy sú zhruba podobne vhodné na výber bezpečnej cesty pri použití rovnakej topológie.

POĎAKOVANIE

Táto práca vznikla v spolupráci Ministerstva školstva Slovenskej republiky pod výskumom VEGA 1/0492/18, KEGA 046TUKE-4/2018 a APVV-17-0208.

LITERATÚRA

- [1] MANSHAEI, Mohammad Hossein, et al. Game theory meets network security and privacy. *ACM Computing Surveys (CSUR)*, 2013, 45.3: 1-39.
- [2] PATHAN, Al-Sakib Khan (ed.). *Security of self-organizing networks: MANET, WSN, WMN, VANET*. CRC press, 2016, ISBN: 9781439819203.
- [3] GILLMAN, Richard Alan; HOUSMAN, David. *Game Theory: A Modeling Approach*. CRC Press, 2019, ISBN: 9781482248098.
- [4] GRØNBÆK, Lone, et al. *Game Theory and Fisheries Management*. Springer Books, 2020.
- [5] BENSLAMA, M. – BOUCENNA, M. L. – BATATIA, H., *Ad Hoc Networks Telecommunications and Game Theory*, Wiley, 2015, ISBN: 20512481.
- [6] HAO, Fei, et al. *K-Clique Community Detection in Social Networks Based on Formal Concept Analysis*. *IEEE Systems Journal*, 2015, 11.1: 250-259.

Teória hier a smerovanie v hybridnej MANET-DTN sieti

¹Ján PAPAJ, ²Gabriela VAŠKOVÁ

¹ Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

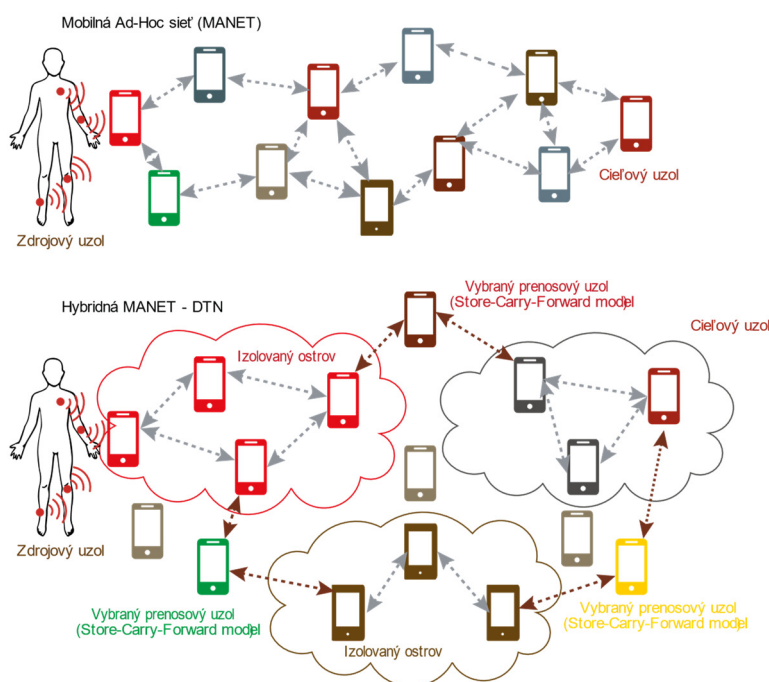
¹jan.papaj@tuke.sk

Abstrakt — Rozvoj moderných aplikácií a služieb si vyžaduje nové typy mobilných viacpreskokových sietí. Význam viacpreskokových sietí narastá s možnosťou ich využitia nielen ako sietí na prenos dát, ale hlavne ako sietí, ktoré je možné využívať v prípade krízových situácií či živelných katastrof. Tento článok popisuje základné využitie teórie hier v procese smerovania v hybridných MANET- DTN sieti, kedy existujúce smerovacie protokoly nie sú schopné nájsť existujúcu komunikačnú cestu medzi jednotlivými uzlami. Článok zároveň poskytuje popis základných princípov teórie hier, ktoré sa aplikujú v procese smerovania. Hlavná výhoda využívania daných algoritmov spočíva vo zvýšení efektivity procesu smerovania.

Keywords — hybridná MANET -DTN, teória hier, smerovanie

I. ÚVOD

Nielen nové 5G siete, ale aj krízové situácie (prírodné katastrofy či nešťastia) poskytujú nové možnosti použitia nových typov mobilných sietí, ktoré nebudú využívať žiadnu pevnú infraštruktúru. Pri záchranárskych operáciách potrebujú záchranárske skupiny komunikovať a vymieňať údaje za účelom vytvorenia si a udržania aktuálneho obrazu o danej situácii. Pevné komunikačné siete nemusia byť k dispozícii, a to buď preto, že boli zničené, alebo sú preťažené v dôsledku zvýšeného záujmu prenášať informácie hlavne pri prírodných alebo živelných katastrof (Obr. 1) [1][2].



Obr. 1 Základná idea MANET a hybridnej MANET-DTN siete [1].

Za správne a efektívne smerovanie v MANET (Mobile Ad-Hoc Network) je zodpovedný smerovací protokol, ktorý riadi a zabezpečuje doručenie dát od zdrojového uzla až k cieľovému

uzlu. Protokol musí zabezpečiť správne doručenie dát od zdroja k cieľu a tiež musí byť schopný riešiť problémy, ktoré môžu vzniknúť. Pri sieťach s nestabilnou infraštruktúrou a s neustále sa meniacou topológiou je zložitosť smerovania ešte vyššia [3x0].

MANET je bezdrôtová decentralizovaná a samo konfigurovaná sieť. Sieť tvoria uzly, ktoré pre prenos dát využívajú ako prenosové bezdrôtové médium. Infraštruktúra siete nie je dopredu určená a vzniká za behu za určitým cieľom. V sieti neexistuje žiaden nariadený prvok, teda uzol, ktorý riadi ostatné uzly. V tejto sieti sú si všetky uzly rovnocenné. Každý uzol v sieti je koncovým uzlom, avšak môže zastávať aj funkciu smerovača a posilať dáta ďalej uzlom, ktoré sa priamo nepodieľajú na komunikácii, pretože nie sú v dosahu smerovania [2]. Všetky zariadenia v sieti by mali zachovávať svoju samostatnosť a tiež chrániť svoje zdroje. Kedykoľvek je možné opustiť túto sieť [2].

Smerovanie v sieti tolerujúcej oneskorenie (Delay Tolerant Network) sa zaoberá schopnosťou prenášať dáta od zdrojového uzla k cieľovému uzlu. Schopnosť prenášať dáta musia mať všetky siete, aby dokázali plniť svoju funkciu. Charakteristickým prvkom sietí tolerujúcich oneskorenie je častokrát chýbajúce spojenie medzi uzlami. Aby boli siete schopné ľahko prenášať dáta potrebujú informácie o budúcich kontaktoch, teda o uzloch s ktorými budú v kontakte. Tieto kontakty môžu byť prerušované alebo oportunistické. Ďalej potrebujú informácie o mobilite uzlov, respektíve, ktoré uzly sú mobilné. Je niekoľko možností mobility uzlov. Nemusia byť žiadne uzly mobilné, alebo môžu byť iba niektoré uzly mobilné, alebo môžu byť takmer všetky uzly mobilné [2],[3].

Smerovanie je hlavným bodom výskumu týchto sietí. Výskum sa týka zistenie trás medzi uzlami a následného udržiavania týchto zistených trás. Dôležité je nájsť najvhodnejšiu trasu z pomedzi dostupných trás. Keďže v týchto sieťach je voľný pohyb uzlov, je potrebné trasy neustále prepočítavať, keďže najvhodnejšia trasa sa môže zmeniť [2].

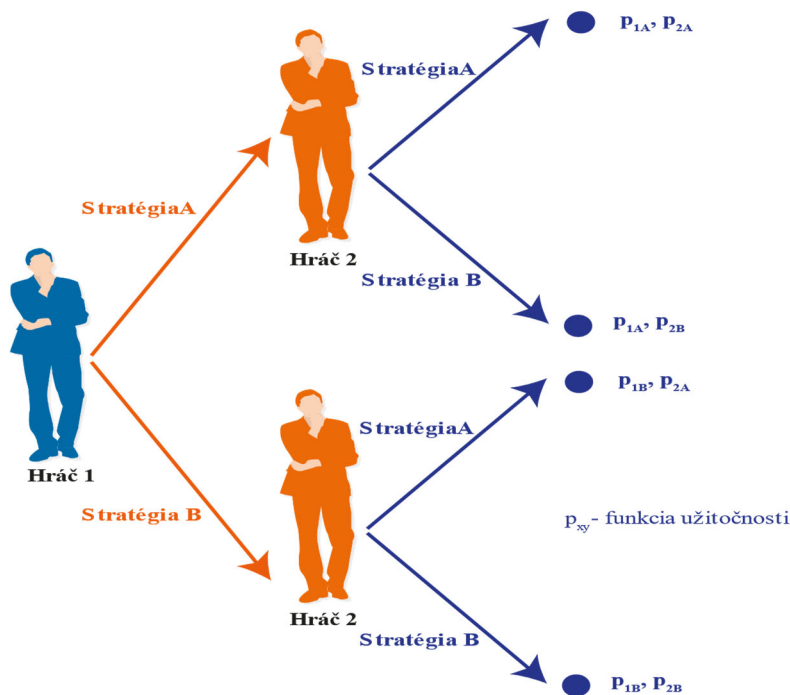
Hlavné charakteristiky týchto sietí sú:

- *Dynamická topológia* – V sieti sa môžu uzly samovoľne pohybovať rôznymi smermi a rôznymi rýchlosťami. Z toho dôvodu sa topológia mobilnej siete v rôznych časoch mení. Sieť sa dynamicky prispôbuje doprave.
- *Prevádzka s obmedzenou energiou* – V sieti uzly berú energiu z napájania, ktoré má obmedzené batérie. Dôležité je preto brať do úvahy spotrebu energie siete. Preto je nevyhnutné optimalizovať úsporu energie.
- *Obmedzená šírka pásma* – Nižšiu kapacitu ako siete s typickou topológiou majú bezdrôtové siete. Do úvahy sa berie aj priepustnosť bezdrôtovej komunikácie s viacnásobným prístupom, šum, interferencie a podobne.
- *Bezpečnostné hrozby* – Siete, ktoré sú bezdrôtové majú obvykle vyššiu náchylnosť ohrozovať fyzickú bezpečnosť, na rozdiel od typických pevných káblových sietí. Existuje možnosť odpočúvania a útokov, čo by sa malo dôsledne zväziť.

II. TEÓRIA HIER PRE HYBRIDNÚ MANET-DTN

Teória hier (TH) je oblasť aplikovanej matematiky, ktorá sa zaoberá, akým spôsobom sa zúčastnené strany, nazývaní hráči, rozhodujú. Hráči sa rozhodujú vzájomne nezávisle, čo spôsobuje, že všetci zúčastnení hráči berú do úvahy rozhodnutia a stratégie iných hráčov pri rozhodovaní, akú stratégiu si sami zvolia. Výsledok hry popisuje rozhodnutia hráčov, ktorí sa môžu rozhodovať podobne, protichodne alebo zmiešane. Z čoho môžu vzniknúť rôzne závery hry [3]. TH mohla byť a zároveň aj bola použitá na analýzu spoločenských hier. Jej použitie je však rozmanitejšie. Teóriu hier vyvinul matematik amerického pôvodu, ktorý bol narodený v Nemecku. Jeho cieľom bolo využiť ju v ekonómii na riešenie problémov. Nakoniec sa aplikovala na rozsiahle množstvo situácií, kedy rozhodnutie hráčov ovplyvňuje výsledok. Jej využitie je napríklad pri určovaní, aké politické strany sa pravdepodobne vytvoria alebo za akú cenu sa môže výrobok predať vzhľadom na konkurenciu [4]. Dôležitým prvkom TH je voľba vhodnej stratégie a funkcie užitočnosti. V teórii hier sa za stratégiu považuje predpis, ktorým je určená jedna z možností, ako sa hráč bude pri hre správať, teda ako sa bude rozhodovať. Správanie hráčov a výber ich stratégie je zobrazený na obr. 2 [5].

Každý hráč má svoju množinu stratégií, z ktorých si vyberie tú, podľa ktorej bude postupovať. Stratégia nie je prvoplánový nápad, ktorý napadne hráčovi niekedy počas hry. Je to vopred zvolený plán, ktorý hráč dodržiava počas niektorých situácií v hre [5].



Obr. 2 Dvaja hráči a ich jednotlivé stratégie.

Funkcia užitočnosti (Utility function), je funkcia, pomocou ktorej je možné vyjadriť akékoľvek správanie hráčov. Pozostáva zo spotreby energie. Možno pomocou nej vyjadriť preferenciu hráča [9]. Funkcia užitočnosti hráča závisí od stratégie hráča, ale tiež aj od stratégie iných hráčov. Funkciu užitočnosti ovplyvňuje rozhodnosť hráča, ako aj rozhodnosť iných hráčov [6]. Hráč si v každej časti hry zvolí cestu, kade odošle správu. Ak sa aktualizuje cesta, zvýši sa pravdepodobnosť v budúcnosti ísť touto cestou. Zvýšením užitočnosti cesty sa zníži cena [6]. Vo chvíli, keď si hráči vyberú svoju stratégiu je o výsledku hry rozhodnuté. Práve vďaka funkcii užitočnosti. U hier, v ktorých hrajú dvaja hráči je funkcia užitočnosti vyjadrená maticou, v ktorej je zapísaný výsledok pre každú dvojicu stratégií [7].

Sedlový bod sa používa na riešenie jednodimenzionálnych hier. Pri riešení sa hľadá stabilné riešenie. To znamená riešenie, v ktorom sa ani jednému hráčovi neoplatí utiecť od stratégie. Má to byť také riešenie, v ktorom ak prvý hráč zmení stratégiu a druhý hráč v hre ostane pri svojej stratégii, potom si prvý hráč pohorší. Platí to rovnako aj s druhým hráčom [7].

Stratégia hráča A je maximinimová, pretože hráč A predpokladá, že B hráč zvolí stratégiu, pri ktorej je minimálna šanca na jeho prehru. Vyberie si z minimálnych výhier v riadkoch maxima. Stratégia, v ktorej je toto maximum najväčšie je pre hráča A optimálna. Táto stratégia sa nazýva aj dolná cena hry [8]. Stratégia hráča B je minimaxová, ak hráč B zistí, koľkokrát môže pri danej stratégii prehrať, čo sú vlastne maximá v stĺpcoch. Z týchto najväčších prehier vyberie minimum. Táto stratégia sa nazýva aj horná cena hry [8]. Ak v matici existuje prvok, ktorý je najmenší vo svojom riadku a najväčší vo svojom stĺpci sa tento prvok nazýva sedlový bod [8]. Teória hier sa môže použiť na výber uzlov, ktoré sa nachádzajú v hybridných MANET – DTN sieťach.

A. Teória hier v procese smerovania pre hybridnú MANET-DTN

Cieľom použitia teórie hier pri smerovaní je riešiť nájdenie optimálnej cesty. Smerovanie a ďalšie problémy boli vyriešené ako formulácie spoločenskej hry. Nekooperatívne smerovacie hry riešia problém cesty, kde cesta je uzavretá vnútri siete od zdroja k cieľu. Snažia sa o najlepšie výhody pre seba a súťažia s inými zdrojmi. Štúdia teoretického modelu hry pre smerovanie bola uvedená Rosenthalom, v roku 1973, kde uvažuje o n-hráčoch, ktorí sú v konkurenčnom prostredí. Každý hráč chce niečo od zdroja k cieľu, pričom jeho snahou je minimalizovať cenu na prepravu. V tomto modeli sa použila Nashova rovnováha. Existuje niekoľko hlavných kapitol [8]:

- Nashova rovnováha
- Shapleyho hodnota
- Jadro

Aby bolo možné vyhodnotiť smerovanie, boli navrhnuté niektoré parametre na zhodnotenie kvality smerovania. Týmito parametrami sú [3],[9]:

- *Celková prijatá smerovacia prevádzka [bit/s]* - Množstvo smerovacej prevádzky prijatej v bitoch v celej sieti. Pokrýva všetky náklady protokolu na ahoj správy, smerovacie pakety, pakety odpovedí, smerovanie chýb a paketov údržby, smerovanie aktualizácií a pod'akovania. Smerovanie prevádzky má vysoký vplyv na priepustnosť siete. Vo väčšine prípadov platí, že čím je vyššia smerovacia prevádzka, tým je nižšia priepustnosť pre konkrétny prenos dát.
- *Pomer dosiahnuteľnosti* – Definovaný ako pomer možných prípustných trás a všetkých možných trás medzi niektorými/všetkými rôznymi zdrojmi do niektorých/všetkých možných cieľov. Znázorňuje percento úspešných pokusov o zistenie smerovania, ktorým sa podaril zistiť aspoň jednu trasu do žiadaného cieľa.
- *Oneskorenie zistenia trasy [s]* - Prezentuje časové oneskorenie potrebné na nájdenie trasy do konkrétnych cieľových uzlov. Počíta sa to od okamihu, kedy zdrojový uzol vyšle požiadavku na trasu, aby našiel trasu do zvoleného cieľa, až do času, keď sa príjme odpoveď s vhodnou trasou do cieľa.
- *Oneskorenie siete [s]* – Predstavuje celkový čas oneskorenia prenosu dát medzi zdrojovým a cieľovým uzlom
- *Celkový opakovaný prenos [pakety]* – Celkový počet prenesených paketov medzi zdrojovým a cieľovým uzlom.

Vo viacpreskokových mobilných sieťach je ich nevyhnutnou súčasťou vyrovnanie pôsobenia uzlov. Uzly niekedy pôsobia sebecky, aby si zachovali energetické zdroje. Sebecké správanie môže spôsobiť zníženie výkonu siete, a niekedy dokonca aj znížiť celú sieť. Siete nemajú typickú infraštruktúru, ich momentálna infraštruktúra môže ovplyvniť vysielačový výkon, prekážka vo vonkajšom prostredí, pohyblivosť, alebo aj správanie, ktoré môže brániť spolupráci. Kvôli ich infraštruktúre sa údaje vymieňajú s účelom udržania komunikácie uzlov. Uzly fungujú ako smerovače pre vzájomnú výmenu informácií. Na podporu prístupu spolupráce medzi jednotlivými uzlami sa využívajú nasledovné metódy [10]:

- *Úverové* – V úverových prístupoch uzly využívajú virtuálnu menu, pomocou ktorej platia za používanie služieb potrebných na prenos. V tomto prípade uzly potrebujú dostatočný kredit na poskytnutie služby iným uzlom.
- *Založené na reputácii* – V prístupoch založených na reputácii uzly kontrolujú kooperatívne správanie iných uzlov. Je to potrebné na zistenie, koľko služieb je potrebných a koľko služieb im bude poskytnutých. Uzly spolupracujú len s uzlami, ktoré majú dôveryhodnú povest'. Uzly so zlou povest'ou sú zanedbané.

Uzol sa v hre rozhodne, či spolupracovať bude alebo nebude. Uzly sa rozhodujú podľa vyhodnotenia potenciálnych prínosov a nákladov, ktoré si spolupráca vyžaduje. Vplyv má aj očakávaná stratégia iných uzlov, ktoré sa nachádzajú v konkrétnej sieti. Pri rozhodovaní pomáha uzlom funkcia užitočnosti. Funkcia obsahuje faktory, ktoré môžu pomôcť pri rozhodovaní. Funkcia užitočnosti vyjadruje ciele uzla, keď si uzol čaká na reakciu iných hráčov na jeho akciu. V nekooperatívnych hrách konajú uzly nezávisle. Naopak, v kooperatívnych hrách sa hráči môžu dohodnúť na vytvorení koalície. Tiež sa môžu dohodnúť na tom, že budú rozhodovať na základe stratégie spolupráce. Pri kooperatívnej hre sa môžu aj všetky uzly spojiť do jednej veľkej koalície, čo ale nie je najlepšou voľbou pre účastníkov hry. Koalície z hľadiska stability delia na [8],[9]:

- *Párová stabilita* – Každý uzol v párovej stabilite je lepší, keď je v tejto stabilite, akoby bol sám.
- *Koaličná stabilita* – Koaličná stabilita predstavuje väčšiu stabilitu. Neexistuje skupina uzlov v tejto stabilite, ktoré by boli lepšie, keby sa oddelili a vytvorili inú koalíciu.

Tieto siete tvoria uzly, ktoré nie sú nevyhnutne spojené, spájajú sa podľa potreby. To znamená, že sa môžu samovoľne spájať, ale aj rozpájať. Z toho vyplýva, že nemajú pevne danú topológiu. Infraštruktúra siete sa môže dynamicky meniť, keďže uzly sa dokážu samovoľne pohybovať všetkými smermi. Každý uzol dokáže samostatne fungovať ako smerovač. Uzol ako smerovač môže prijímať a odosielať pakety, teda informácie do iných uzlov. Z dôvodu, že sa uzly dokážu samovoľne hýbať a kapacita energie je obmedzená sú mobilné siete náchylnejšie na časté zmeny topológie siete. Sieť je schopná udržiavať komunikáciu napriek týmto zmenám [10]. Teóriu hier

je možné klasifikovať podľa viacerých kritérií [8],[9],[10]:

- *Podľa počtu hráčov* – hry, kde je počet hráčov 2 až n , pričom $n > 2$.
- *Podľa súčtu výplat hráčov* – hry, kde je súčet konštantný alebo nekonštantný; špeciálny prípad hry s konštantným súčtom je hra s nulovým súčtom.
- *Podľa veľkosti priestoru pre stratégiu* – hry, ktorú sú konečné a nekonečné.
- *Podľa informácie* – hry, v ktorých hráči sú informovaní o výsledkoch protihráčov sú hry s úplnou informáciou, a hry, v ktorých hráči nemajú informáciu o protihráčoch sú hry s neúplnou informáciou.
- *Podľa záujmu hráčov* – hry antagonistické a hry neantagonistické, ktoré sa ďalej rozdeľujú na hry kooperatívne a nekooperatívne.
- *Podľa informácií o dôsledkoch voľby* – hry deterministické, v ktorých ak zvolíme stratégiu vieme presne, koľko dostaneme a hry stochastické, v ktorých do hry vstupuje náhoda, čo znamená, že výplata hráčov má pravdepodobnostné rozdelenie.
- *Podľa racionality hráčov* – hry hrané podľa minimaxu a hry, ktoré sú hrané proti prírode.

ZÁVER

Cieľom tohto článku je poskytnúť základný pohľad na problematiku teórie hier viacpreskokových mobilných sieťach. Teória hier je dynamicky rozvíjajúca sa veda, ktorá poskytuje široké možnosti využitia nielen v ekonomických vedách, ale aj v mobilných komunikáciách, špeciálne v hybridných MANET-DTN sieťach. Efektívny výber vhodných prenosových uzlov do značnej miery zvýši efektívnosť jednotlivých smerovacích protokolov. Samotná činnosť teórie hier umožňuje nielen efektívny výber uzlov, ale aj umožní zvýšiť bezpečnosť samotného výberu jednotlivých uzlov, čím sa eliminuje činnosť zlozmyselných uzlov. V nasledovnej vedeckej činnosti sa zameriame na implementáciu kooperatívnych a nekooperatívnych hier do procesu smerovania a prenosu samotných dát.

POĎAKOVANIE

Táto publikácia vznikla vďaka podpore Agentúry na podporu výskumu a vývoja v rámci projektu č. APVV-17-0208 - Odolné mobilné siete na doručovanie obsahu a vedeckých grantových agentúr v rámci projektu VEGA 1/0492/18 - Odolná dátová komunikácia pre krízové situácie využívajúca viacpreskokové mobilné siete.

REFERENCES

- [1] J. Papaj, L. Dobos, A. Cizmar, A., "Hybrid MANET-DTN and a New Algorithm for Relay Nodes Selection. Wireless Personal Communication, Vol. 96, pp. 5145-5166, 2017.
- [2] J. Papaj, L. Dobos, "Cooperation between Trust and Routing Mechanisms for Relay Node Selection in Hybrid MANET-DTN", Mobile Information Systems. Vol. 2016, pp. 1-18, 2016.
- [3] H. Wang, H. Wang, J. Tan, H. Lv and M. Zhu, "A Delay Tolerant Network Routing Policy Based on Optimized Control Information Generation Method," in IEEE Access, vol. 6, pp. 51791-51803, 2018, doi: 10.1109/ACCESS.2018.2869380.
- [4] G. Jinarajadasa, L. Rupasinghe and I. Murray, "A Reinforcement Learning Approach to Enhance the Trust Level of MANETs," 2018 National Information Technology Conference (NITC), Colombo, 2018, pp. 1-7, doi: 10.1109/NITC.2018.8550072.
- [5] K. Khaitiyakun, T. Sanguankotchakorn, A. Tunpan, "Data dissemination on MANET using content delivery network (CDN) technique", The International Conference on Information Networking 2014 (ICOIN2014), Phuket, 2014, pp. 502-506.
- [6] S. Ergün, T. Aydoğan, S. Zeynep Alparslan Gök: An Overview of Game Theory and its Applications in Network Routing. 2014. 15 s.
- [7] A. Naja, M. Boulmalf, M. Essaïdi, "A game theoretical based rebroadcasting protocol for content dissemination in VANETs," 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 2019, pp. 591-596, doi: 10.1109/IWCMC.2019.8766594.
- [8] Q. Wang, Y. Zhu and X. Wang, "Incomplete Information based Collaborative Computing in Emergency Communication Networks," in IEEE Communications Letters, doi: 10.1109/LCOMM.2020.2995501.
- [9] S. Chu, Z. Fang, S. Song, Z. Zhang, C. Gao and C. Xu, "Efficient Multi-Channel Computation Offloading for Mobile Edge Computing: A Game-Theoretic Approach," in IEEE Transactions on Cloud Computing, doi: 10.1109/TCC.2020.2994145.
- [10] A. Tomar, P. Chanak, "A Game Theory based Fault Tolerance Routing Scheme for Wireless Sensor Networks," 2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCECS), Bhopal, India, 2020, pp. 1-5, doi: 10.1109/SCECS48394.2020.132.

Nilssonov-Heydenov algoritmus aktívnych obrysov

Iveta GLADIŠOVÁ, Ján MIHALÍK

Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

iveta.gladisova@tuke.sk, jan.mihalik@tuke.sk

Abstrakt — Článok pojednáva o algoritme podľa Nilssona a Heydena, ktorý je určený pre pohyb aktívnych obrysov pri segmentácii obrazov. Stručne sú opísané aktívne modely a Level set metóda, nakoľko skúmaný algoritmus je efektívnou aproximáciou Level set metódy. Ďalej je predstavená implementácia navrhnutého Nilssonovho-Heydenovho algoritmu do programového prostredia MATLAB a zhodnotenie dosiahnutých výsledkov simulačným testovaním tohto algoritmu.

Kľúčové slová — segmentácia obrazu, aktívne modely obrysov, Nilssonov-Heydenov algoritmus

I. ÚVOD

Pre mnoho metód spracovania obrazov je v dnešnej dobe kľúčovým prvkom segmentácia obrazu. Segmentačné metódy a algoritmy sú aplikované napríklad pri rozpoznávaní a klasifikácii objektov v obraze. Segmentácia obrazu predstavuje proces, ktorý delí vstupný obraz na súvislé oblasti vyhovujúce danému kritériu [2]. Tieto časti obrazu, oblasti, sú potom predmetom záujmu v ďalšom spracovaní obrazu [3]. Inými slovami, princípom segmentácie je oddelenie obrazových bodov uvažovaných objektov, ktoré sa nachádzajú v popredí obrazu od obrazových bodov prislúchajúcich do pozadia obrazu. Výstupom segmentácie by mal byť súbor oblastí, ktoré zodpovedajú objektom vo vstupnom obraze. Ak nájdené oblasti presne zodpovedajú objektom vo vstupnom obraze, potom sa jedná o tzv. kompletnú segmentáciu. Ak oblasti presne neodpovedajú objektom, tak takúto segmentáciu nazývame čiastočnou. Kompletná segmentácia všeobecne využíva vyššiu úroveň spracovania, ktorá je založená na znalostiach riešeného problému. Čiastočná segmentácia je založená na princípe homogenity obrazových vlastností (napríklad jas, farba) vo vnútri segmentu [4].

Segmentácia obrazu siaha do ďalekej histórie, konkrétne do rokov 1970 [5]. V priebehu desaťročí bolo publikovaných veľa rozmanitých segmentačných metód. Boli to algoritmy predstavené od jednoduchých až po veľmi zložité. Metódy zahŕňali všeobecné postupy až po špecializované algoritmy, ktoré boli navrhnuté a vyvíjané pre určité funkcie v spracovaní obrazu [6]. Segmentačné metódy s ktorými sa môžeme stretnúť patria do rôznych skupín, ako napríklad: segmentácia prahovaním, hybridné metódy segmentácie, znalostné metódy (tzv. knowledge-based), metódy orientované na oblasti (tzv. region-based) a segmentácia pomocou detekcie hrán (tzv. edge-based). K segmentácii pomocou detekcie hrán zaraďujeme takzvané aktívne obrysy a Level set metódu [7], [8]. Práve z týchto dvoch metód vychádza rýchly algoritmus Nilssona a Heydena (A fast algorithm for level set-like active contours) [1], ktorým sa zaoberá tento článok.

II. AKTÍVNE MODEL Y OBRYSOV

Hlavnou úlohou aktívnych modelov je deformácia krivky (obrysu), ktorá je umiestnená v rovine, v prípade dvojrozmerných obrazov, alebo je to deformácia povrchu v priestore pri trojrozmerných obrazoch [3], [7]. Optimálny proces deformácie krivky je následkom vhodne nastaveného (vybraného) energetického funkcionálu. V bližšom opise je pohyb krivky/povrchu spôsobený vplyvom vonkajších a vnútorných síl v čase, kde krivka sa zastaví v mieste obrazu s najmenšou energiou. Energetický funkcionál teda obsahuje dve zložky: energiu vnútorných síl a energiu vonkajších síl. Vnútorné sily sú podmienené len aktuálnym stavom geometrického modelu a udržiavajú ho vcelku. Vonkajšie sily sú závislé na čase a pozícii modelu. Spôsob reprezentácie pohybu krivky alebo povrchu (všeobecne rozhrania) rozdeľuje aktívne modely na 2 skupiny, a to explicitné – nazývané ako parametrické a implicitné – známe aj ako geometrické [1], [2].

Pri parametrických aktívnych modeloch sa uvažuje počiatočná uzavretá krivka v dvojrozmernom priestore a pohyb tejto krivky rýchlosťou F v smere jej normálového vektora generuje množinu kriviek. Rýchlosť F predstavuje skalárne hodnoty v každom bode krivky. Ak zvolíme jej vhodnú diskretizáciu, získame súbor rovnomerne rozmiestnených bodov krivky obrysu. Pre vyjadrenie množiny bodov po presune za čas Δt existujú vzťahy [9], ktoré sú s využitím aproximácie parciálnych derivácií aplikované na hodnoty množiny bodov krivky obrysu. Potom pomocou výpočtov takýchto aproximácií získavame nové hodnoty bodov posunutej krivky obrysu. Výpočtový postup parametrických aktívnych modelov prináša niektoré nevýhody. Jednou z nich je, že pri výpočte rovnice pre pohyb krivky prudko stúpa zložitosť implementácie. Z aplikácie parametrických modelov môžu vyplývať aj chyby spojené s tzv. reparametrizáciou. Pri aproximácii parciálnych derivácií sa pre výpočet derivácie v určitom bode využívajú aj údaje o pozícii jeho okolitých bodov. Problém nastáva pri výpočte bodov umiestnených (navzorkovaných) v konvexných a konkávných oblastiach krivky. Nové body sú rozmiestnené v týchto častiach krivky nerovnomerne, a preto je potrebná reparametrizácia krivky. V opačnom prípade, ak by nedošlo k prevzorkovaniu, začali by narastať chyby pri výpočte parciálnych derivácií nesprávne umiestnených (navzorkovaných) bodov [1].

Implicitné aktívne modely pre reprezentáciu krivky obrysu využívajú implicitnú funkciu ϕ . Podľa literatúry z [1], [7] je krivka vyjadrená ako nulová vrstevnica implicitnej funkcie, ktorá je navzorkovaná v napevno zadanej súmernej sieti bodov v dvojrozmernom priestore. Implicitná funkcia označovaná ako $\phi(x, y, t)$ predstavuje v tomto prípade funkciu dvoch (priestorových) premenných x, y a časovej premennej t . Krivka sa pohybuje v smere normálového vektora s rýchlosťou F v rámci vývinu celej implicitnej funkcie ϕ . Po ukončení pohybu a dosiahnutí času Δt je nájdená nová nulová vrstevnica. Zmena funkcie ϕ je riešená pomocou parciálnej diferenciálnej rovnice [1], [9]. Pri zmene topológie nie sú potrebné dodatočné výpočty pre opravu krivky. Medzi výhody implicitných aktívnych modelov je zaradený aj test príslušnosti. Je to postup, pomocou ktorého je možné určiť polohu daného bodu krivky \mathbf{p} v súvislosti s rozhraním. Výsledkom testu je informácia o tom, či sa bod \mathbf{p} nachádza vo vnútri alebo zvonku uzavretého pohybujúceho sa rozhrania a je určená znamienkom, ktoré prislúcha hodnote funkcie $\phi(\mathbf{p}, t)$. Ak je hodnota funkcie $\phi(\mathbf{p}, t) < 0$, poloha bodu \mathbf{p} je vo vnútri hranice (obrysu) a ak nadobúda opačné, kladné hodnoty, bod \mathbf{p} sa vyskytuje zvonku hranice (obrysu) [1], [9].

Level Set metódu vynašli v roku 1988 Sethian a Osher [10] a bližšie ju opísali napríklad v literatúre [7], [9]. Level Set je typ metódy segmentácie obrazov, ktorá predstavuje výpočtovú realizáciu implicitných aktívnych modelov. Teda rozdiel od aktívnych obrysov je ten, že tvar krivky nie je meraný priamo, ale prostredníctvom level set funkcie. V tomto type segmentácie je krivka reprezentovaná tzv. nulovou hladinou (nazývanou aj vrstevnicou, z anglického názvu *zero level set*), teda rezom v rovine xy určitej viacrozmernej funkcie. Takúto funkciu nazývame *level set function* [7]. Level set funkcia priradzuje každému bodu so súradnicami $[x, y]$ výšku z nad alebo a pod touto nulovou hladinou. V literatúre [10] Mulder, Osher a Sethian ukázali, že ako začiatočnú funkciu je vyhovujúce použiť tzv. *signed distance function* ϕ . V tomto prípade, inicializačný stav hranice (obrysu) je $C(0)$, ktorá zároveň predstavuje nulovú hladinu (vrstevnicu) implicitne zadanej funkcie ϕ . Potom platí:

$$C(0) = \{\mathbf{p} | \phi(\mathbf{p}, 0) = 0\} \quad (1)$$

Pre riešenie v čase je funkcia pozmenená do tvaru $\phi(\mathbf{p}, t)$. Ďalej pre výpočet je potrebné odvodiť vhodný tvar diferenciálnej rovnice. Ak je dané pohybujúce sa rozhranie v čase, tak nech funkcia $x(t)$ predstavuje dráhu jedného bodu \mathbf{p} na tomto rozhraní, potom $x(0)$ je bod definovaný na rozhraní $C(0)$ a z toho vyplýva [9]:

$$\phi(x(t), t) = 0 \quad (2)$$

Keď na rovnicu aplikujeme pravidlo pre deriváciu zloženej funkcie, získame:

$$\phi_t(x(t), t) + \nabla\phi(x(t), t) \cdot x_t(t) = 0 \quad (3)$$

Ďalej pre rýchlosť pôsobiacu na obrys v smere normálového vektora \vec{N} platí vzťah

$$F = x_t(t) \cdot \vec{N}, \text{ kde normálový vektor } \vec{N} = \frac{\nabla\phi}{|\nabla\phi|}, \text{ a teda:}$$

$$x_t(t) = F \cdot \frac{|\nabla\phi|}{\nabla\phi} \quad (4)$$

Nasledovné úpravy rovnice (3) tak sú:

$$\phi_t + \nabla\phi \cdot F \cdot \frac{|\nabla\phi|}{\nabla\phi} = 0 \quad (5)$$

$$\phi_t + F \cdot |\nabla\phi| = 0 \quad (6)$$

V predchádzajúcich rovniciach je zavedená pôsobiaca rýchlosť F , pre ktorú teraz priblížime vyjadrenie pomocou jej troch zložiek. Vzťah pre rýchlosť je daný [9], [3] nasledovne

$$F = F_0 + F_{curv} + F_{ext}, \quad (7)$$

pričom prvá zložka F_0 predstavuje konštantnú rýchlosť obrysu (rozhrania) v smere normálového vektora. Druhá zložka je závislá na lokálnej krivosti a jej vzťah je $F_{curv} = -\varepsilon \cdot \kappa$. Ovplyvňuje pohyb krivky, ktorý závisí od tvaru implicitne zadanej funkcie ϕ . Poslednou zložkou je

$F_{ext} = \beta [\vec{V}(x, y, z) \cdot \vec{N}]$, ktorá udáva rýchlosť vo vnútornom vektorovom poli. Hodnota rýchlosti F_{ext} je výsledkom skalárneho súčinu \vec{N} a vektora $\vec{V}(x, y, t)$, ktorého smer a veľkosť je závislá od času a pozície. V zložkách F_{curv} a F_{ext} reprezentujú ε a β mieru vplyvu na lokálnu krivosť a vplyv vnútorného vektorového poľa na pohyb rozhrania (obrysu).

Pripomeňme, že F_0 a F_{ext} vyplývajú z pôsobenia vonkajších síl, kde F_0 definuje rýchlosť nafukovacej sily a F_{ext} je rýchlosť obrazovej sily. F_{curv} potom určuje pôsobenie vnútorných síl na rozhranie [3], [7].

Vyjadrenie výslednej rozšírenej rovnice pre level set metóde pomocou zložiek rýchlosti je tak

$$\phi_t + F_0 \cdot |\nabla\phi| - \varepsilon \cdot \kappa \cdot |\nabla\phi| + \beta \cdot \vec{V} \cdot |\nabla\phi| = 0 \quad (8)$$

Podľa vzťahu (8) môžeme pozorovať, že pre určenie rýchlostnej funkcie je potrebné vyjadriť vzťahy pre zložku rýchlosti F_0 a vektorové pole \vec{V} . Potom je možné odvodiť vzťah pre celkovú rýchlostnú funkciu v jednom bode označenú ako $F(\mathbf{p})$.

III. PRINCÍP SEGMENTAČNÉHO NILSSONOVHO-HEYDENOVHO ALGORITMU

Pohybujúce sa rozhranie tu nie je vyjadrené pomocou numerického riešenia parciálnej diferenciálnej rovnice, ako to bolo spomínané vo všeobecnej metóde aktívnych obrysov, ale hranica obrysu sa počas jednej iterácie presunie z určitého bodu do susedného bodu práve o jednu dĺžku vzdialenosti. Podstatou algoritmu sú dve hlavné myšlienky. Prvou je, že pohyb rozhrania pozostáva z presne určenej postupnosti pohybu jeho jednotlivých bodov, ktorým sú pridelované určité hodnoty. Druhá myšlienka hovorí o geometrických vlastnostiach rozhrania, ktorými sú krivosť a normálový vektor. Tie sú účinne aproximované a nie je nutné, aby bola implicitná funkcia ϕ vo forme *signed distance function*, ktorá je požadovaná pri numerickom výpočte spomínaných metód (Level Set) [9].

Počas výpočtu algoritmu sa stretávame s dvoma typmi premenných spojených s časom. Sú to čas odchodu $T_d(\mathbf{p})$ a čas príchodu $T_a(\mathbf{p})$. V jednotlivých iteráciách je vypočítavaný čas odchodu bodu \mathbf{p} z rozhrania podľa nasledujúceho vzťahu: [1]

$$T_d(\mathbf{p}) = T_a(\mathbf{p}) + \frac{1}{\max\{F(\mathbf{p}), \gamma\}} \quad (9)$$

Čas odchodu pozostáva z dvoch zložiek, z času príchodu pohybujúceho sa rozhrania do bodu \mathbf{p} , označený ako $T_a(\mathbf{p})$, a z doby $\frac{1}{\max\{F(\mathbf{p}), \gamma\}}$, za ktorú hranica zotrúva v bode \mathbf{p} . Operácia maxima v menovateľovi zlomku v rovnici (9) zabezpečuje, aby nedošlo deleniu nulou. γ predstavuje nemennú konštantu s malým číslom (v našom testovaní algoritmu bola táto hodnota rovná 10^{-9}). Algoritmus prebieha tak, že opätovne vyberá z rozhrania ten bod \mathbf{p} , ktorý má najmenší čas odchodu T_d a prenáša rozhranie z \mathbf{p} do jeho susedných bodov (pracujeme so 4-susedstvom). Sú to susedné body bodu \mathbf{p} ležiace buď vo vnútri alebo zvonku nulovej vrstevnice funkcie ϕ . Hranica je posúvaná podľa hodnoty $F(\mathbf{p})$, konkrétne podľa znamienka.

Vo vyššie uvedenom texte sme spomínali, že geometrické vlastnosti rozhrania sú účinne aproximované a nepožadujú zložité vyjadrenie implicitnej funkcie. Algoritmus preto pracuje so zjednodušeným tvarom implicitnej funkcie, ktorá je označená ako $\tilde{\phi}$ a jej zápis je nasledovný: [1]

$$\tilde{\phi} = \begin{cases} -1 & \mathbf{p} \in S^- \\ 0 & \mathbf{p} \in S \\ 1 & \mathbf{p} \in S^+ \end{cases} \quad (11)$$

Body patriace do množiny S^- predstavujú množinu bodov patriacich do vnútra rozhrania a hodnota funkcie je na pozícií týchto bodov nastavená na -1 . S je množina bodov \mathbf{p} , ktoré ležia na rozhraní, tu prislúcha implicitnej funkcii hodnota 0 . Poslednou množinou je S^+ a je to množina bodov popisujúcich vonkajšiu stranu pohybujúceho sa rozhrania. V týchto bodoch (\mathbf{p} patriace S^+) je zjednodušenej implicitnej funkcii pridelená hodnota 1 .

Ak je určitý bod vzdialený od hranice o viac než jednu pozíciu v mriežke bodov, potom zjednodušená implicitná funkcia $\tilde{\phi}$ neobsahuje informáciu, ktorá by hovorila o takejto vzdialenosti. V algoritme Nilssona Heydena je krivosť hranice aproximovaná nasledujúcim vzťahom: [1]

$$\tilde{\kappa}(\mathbf{p}) = \frac{h(\mathbf{p})}{N} - \frac{1}{2} \quad (12)$$

kde $h(\mathbf{p})$ reprezentuje počet bodov z množiny S^- , ktoré sa nachádzajú v kruhovom okolí bodu \mathbf{p} s polomerom R a N je počet bodov, ktoré tvoria kruhové okolie.

Pretože v tomto algoritme krivosť ovplyvňuje znamienko pre jednu zo zložiek rýchlosti, je potrebné zmeniť vo vzťahu (7) znamienko rýchlosti F_{curv} . Výsledný vzťah potom bude [1]:

$$F = F_0 - F_{curv} + F_{ext} \quad (13)$$

Výpočet gradientu ∇ v normálovom vektore \vec{N} je nahradený výpočtom centrálnych diferencií.

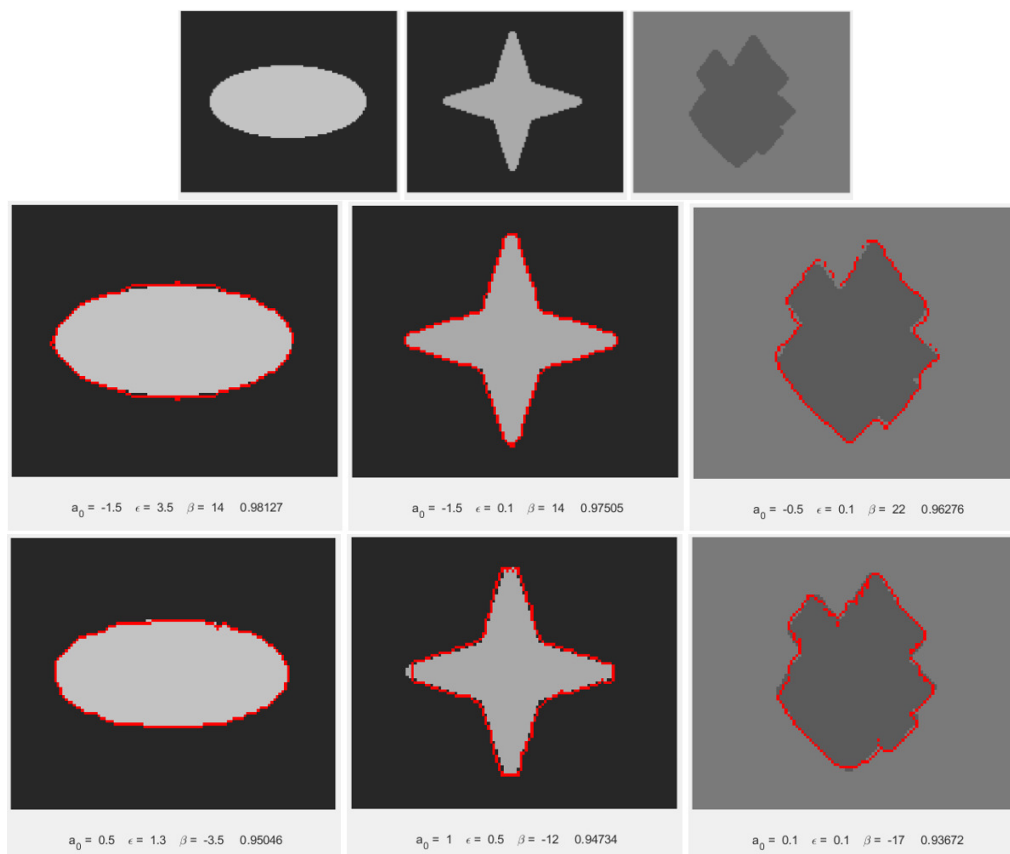
Priebeh Nilssonovho-Heydenovho algoritmu je pre následnú implementáciu v programovom prostredí Matlab možné sformulovať nasledovne:

1. Inicializácia zjednodušenej implicitnej funkcie $\tilde{\phi}$ podľa zadaného počiatočného rozhrania od používateľa. Na pozíciách všetkých bodov \mathbf{p} patriacich do množiny S sú nastavené hodnoty $T_a(\mathbf{p}) = 0$ a následne sú vypočítané hodnoty $F(\mathbf{p})$ a $T_d(\mathbf{p})$.
2. Ďalším krokom je hľadanie pozície bodu \mathbf{p} , u ktorého bola vypočítaná najmenšia hodnota T_d . Tento bod sa odstráni z množiny S a podľa vypočítanej hodnoty $F(\mathbf{p})$ na tejto pozícii sa vykoná jedna z dvoch podúloh:
 - a. Hodnota $F(\mathbf{p}) > 0$. Nastaví sa hodnota zjednodušenej implicitnej funkcie tohto bodu na $\tilde{\phi}(\mathbf{p}) = -1$ a bod \mathbf{p} bude pridaný do množiny S^- , všetky susedné body z množiny S^+ (body s hodnotou 1) okolo tohto bodu \mathbf{p} sa presunú do množiny S (novo-pridané body do množiny S označíme ako množinu P), ich hodnota $\tilde{\phi}$ sa zmení na 0 a ich čas príchodu T_d sa nastaví na najmenšiu nájdenú hodnotu $T_d(\mathbf{p})$. Ďalej je nutné meniť hodnoty krivosti pripočítaním hodnoty $\frac{1}{N}$ k vzťahu (12) v bodoch množiny S (vynímajúc body z množiny P), ktoré sa nachádzajú v kruhovom okolí bodu \mathbf{p} . Teda pre body v kruhovom okolí bodu \mathbf{p} v bodoch rozhrania S , ktoré sa nemenili, sa k hodnotám krivosti pripočíta hodnota $\frac{1}{N}$. Výpočet krivosti v množine P sa uskutočňuje v nasledujúcom kroku 3.
 - b. Hodnota $F(\mathbf{p}) \leq 0$. Potom sa zmení hodnota zjednodušenej implicitnej funkcie v tomto bode \mathbf{p} na $\tilde{\phi}(\mathbf{p}) = 1$ a bod \mathbf{p} sa pridá do S^+ a všetky jeho okolité (susedné) body s hodnotami $\tilde{\phi} = -1$ sa premiestnia do množiny S (aj v tomto prípade ich označíme ako množinu P), nastaví sa ich hodnota funkcie $\tilde{\phi}$ na 0 a čas príchodu T_d na čas podľa bodu \mathbf{p} , teda na hodnotu $T_d(\mathbf{p})$. Na záver tohto kroku je potrebné meniť hodnotu krivosti, a to vo všetkých bodoch množiny S , ktoré ležia v kruhovom okolí jednotlivých bodov množiny P . V tomto prípade sa hodnota krivosti upravuje odčítaním hodnoty $\frac{1}{N}$.
3. V novo pridaných bodoch rozhrania S (body z množiny P) sa krivosť prepočíta nanovo. Hodnoty F a T_d sa vypočítajú v bodoch, ktoré boli pridané do množiny S a bola zmenená ich hodnota funkcie na $\tilde{\phi} = 0$ a taktiež vo všetkých bodoch, u ktorých bola prepočítavaná hodnota krivosti $\tilde{\kappa}$.
4. Tento posledný krok predstavuje ukončenie algoritmu a to v prípade, že množina S bude prázdna (alebo sa nastaví iné zastavovacie kritérium, ako napríklad maximálny počet iterácií). V opačnom prípade sa algoritmus cyklicky opakuje od 2. kroku.

IV. DOSIAHNUTÉ VÝSLEDKY A ICH ZHODNOTENIE

Navrhnutý program na implementáciu Nilssonovho-Heydenovho algoritmu aktívneho obrysu objektov v programovom prostredí MATLAB spracováva rôzne vstupné šedé obrazy. Pre dosiahnutie najlepších výsledkov segmentácie je potrebná čo najpresnejšia voľba vhodných vstupných parametrov vplývajúcich na výpočet rýchlosti posunu jedného bodu $F(\mathbf{p})$, a teda aj na posun obrysu. Zároveň sme sa venovali testovaniu pohybu obrysu so začiatočnou polohou v rôznych pozíciách vstupného obrazu (zvonku a zvnútra objektu). Ďalej sme sa zamerali na výpočet percentuálnej zhodnosti nájdeného segmentovaného objektu s pôvodným objektom v originálnom obraze. Následne sme s výsledkami zhodnosti porovnávali aj výsledky po pohybe obrysu, ktorý sme aplikovali na obrys po segmentácii. Riešili sme aj automatické vyhľadávanie správnych vstupných parametrov do algoritmu aktívnych obrysov pre vybrané vstupné obrazy, a to: a_0 (vplyv na nafukovanie/vyfukovanie obrysu), ϵ (vplyv na vyhladenie obrysu), β (vplyv pritiažnutia obrysu k hrane), ktoré boli použité pri výpočte rýchlosti $F(\mathbf{p})$ všetkých bodov množiny S .

Pre automatické testovanie nájdenia vstupných parametrov Nilssonovho-Heydenovho algoritmu sme použili testovacie obrazy zobrazené na Obr.1 v hornom rade. Pod nimi sú v strednom rade zobrazené dosiahnuté výsledky s nájdenými vstupnými parametrami (pod konkrétnym obrazom a výslednou hodnotou úspešnosti segmentácie (numerická hodnota celkom vpravo pod každým z obrazov) spolu s „pritiažnutým“ výsledným obrysom (červená farba), keď začiatočný inicializačný obrys bolo nastavený blízko vnútorného okraja obrazu (t.j. zvonka). V spodnom rade Obr.1 sú dosiahnuté výsledky parametrov a výsledné obrysy, keď začiatočný inicializačný obrys bol nastavený vo vnútri (strede) obrazu.



Obr.1 Testovacie obrázky (horný rad), nájdené vstupné parametre a výsledné obrisy pritiahnuté k objektom z vonkajšieho (stredný rad) a z vnútorného (spodný rad) inicializačného obrisy.

Na Obr.1 v strednom rade vpravo môžeme pozorovať pokles úspešnosti segmentácie s narastajúcou zložitou tvaru objektu (zľava doprava, a to z hodnoty 0,98127 na 0,9627). Obrys sa zmršťoval a prítahoval k hranám objektu od obvodových okrajov obrazu. Spodný rad znázorňuje výsledky aktívneho obrisy s inicializačným rozhraním umiestneným do stredu obrazu, t.j. obris v priebehu segmentácie narastal zo stredu obrazu smerom k najbližším vnútorným hranám objektu. Počet iterácií bol pre obraz zľava 3500, obraz v strede a naľavo 4200.

Na nasledujúcom Obr.2 sú zobrazené výsledné obrisy objektov reálneho obrazu veľkosti 300 x 246 obrazových prvkov, a to červenou farbou a zelenou farbou inicializačné aktívne rozhranie umiestnené popri hranách obrazu. V popise obrazu sú uvedené nájdené veľkosti vstupných parametrov a_0 , ϵ , β . Segmentácia bola nastavená pre zmršťovanie inicializačného aktívneho obrisy podľa záporného parametra a_0 .



Obr.6 Výsledky segmentácie pre parametre: $a_0 = -4$, $\epsilon = 3,0145$; $\beta = 67,7$.

Tab. 1 Tabuľka presnosti segmentačného algoritmu s prislúchajúcimi vstupnými parametrami.

Parametre β	Parametre ϵ					
	0,1	0,4	1,3	3,5	5	7,5
0,1	0,7453	0,7453	0,7453	0,7453	0,7453	0,7453
0,5	0,7453	0,7453	0,7453	0,7453	0,7453	0,7453
1,5	0,7617	0,7617	0,7605	0,7485	0,7453	0,7453
3,5	0,7882	0,7868	0,7655	0,7612	0,7601	0,7601
14	0,9771	0,9795	0,9749	0,9813	0,9753	0,9712
22	0,9686	0,9672	0,9668	0,9631	0,9557	0,9557

Tabuľka 1 obsahuje výsledky automatického vyhľadania parametrov vstupného obrazu s veľkosťou 120 x 100 (znázornený na Obr.1 v ľavom hornom rohu) podľa presnosti segmentačného algoritmu nájdenia obrysu objektu. Test bol nastavený na segmentáciu s pohybom obrysu od hrán obrazu smerom dovnútra k hranám objektu, pričom sme uvažovali parameter $a_0 = -1,5$. Zvýraznený stĺpec a riadok ukazuje na výsledné hodnoty parametrov ε a β pri 98,13% presnosti nájdeného obrysu v porovnaní s originálnym obrysom objektu.

V. ZÁVER

Článok bol zameraný na opis princípu a implementácie Nilssonovho - Heydenovho algoritmu aktívnych obrysov. Pojem aktívneho obrysu je spájaný s implicitnými aktívnymi modelmi, ktoré reprezentujú pohyb rozhrania (určitého inicializačného obrysu) v obraze. Sú to geometrické modely, ktoré môžu byť transformované a výsledkom je deformácia obrysu. Deformácia nastáva na základe pôsobenia vonkajších a vnútorných síl, a preto sa inicializačný obrys v obraze začne pohybovať. Výpočet pohybu obrysu je potom riešený pomocou rýchlostnej funkcie podľa Level set metódy. Skúmaný algoritmus efektívne aproximuje Level set metódu a rozhranie (obrys) sa pohybuje v obraze po jeho bodoch bez nutnosti riešenia parciálnych diferenciálnych rovníc. Jednotlivé zložky vo výpočte potom predstavujú pôsobiace sily a ich vplyv na pohyb obrysu môže byť korigovaný pomocou vstupných parametrov. Zamerali sme sa na pozorovanie pohybu obrysu pri voľbe rôznych vstupných parametrov. Význam rýchlosti rozhrania F_0 a rýchlosti vnútorného vektorového poľa F_{ext} spočíva v tom, že majú vplyv na inicializáciu pohybu (rozhybanie) a na zastavenie obrysu v správnom mieste obrazu. Je to miesto, kde nastáva zvýraznenie hrany pomocou výpočtov gradientu.

V simulačnom testovaní segmentačného Nilssonovho - Heydenovho algoritmu aktívnych obrysov v MATLAbe sme zisťovali, aké vstupné parametre boli pre segmentáciu najvhodnejšie. Pohyb obrysu sme pozorovali z rôznych polôh inicializačného obrysu v obraze. Bola to pozícia v strede obrazu alebo objektu, keď obrys postupne narastal; ďalej pozícia na hranách (po obvode) obrazu, kedy sa kontúra zmršťovala, a tak obkolesila aj viacero objektov naraz. Skúmali sme aj inicializáciu obrysu vo vnútri objektu a zároveň z vonkajšej strany objektu. Takáto segmentácia ale nebola výhodná, pretože bolo potrebných oveľa viac iterácií a taktiež vzniklo nadbytočné množstvo informácií o polohe obrysu. Pre lepšie testovanie segmentácie sme k algoritmu implementovali parameter, ktorý percentuálne hodnotil správne oddelenie popredia (hľadaného objektu) od pozadia. Na základe tohto výpočtu sme mohli vykonať automatické testy v programe, a tak efektívnejšie nájsť vhodné vstupné parametre pre segmentáciu. Z pozorovaných výsledkov vyplynulo, že obrys sa opakovane zastavil 1 až 2 obrazové prvky od hrany objektu. Z tohto dôvodu bola navrhnutá funkcia, ktorá posúvala obrys v smere jej pohybu ešte o 1 obrazový prvok bližšie k objektu a percento úspešnosti segmentácie sa zlepšilo. Simuláciu sme otestovali aj reálnych obrazoch, kde bola potvrdená funkčnosť navrhnutého algoritmu. Zo simulácií Nilssonovho-Heydenovho algoritmu aktívnych obrysov vyplynulo, že výsledky segmentácie sú závislé od všetkých troch vstupných parametrov. Vstupná hodnota každého parametra je teda podmienená od zvyšných dvoch parametrov, kde a_0 a β majú väčší vplyv na segmentáciu ako ε .

POĎAKOVANIE

Poďakovanie patrí Ing. Lucii Stašovej, ktorá sa podieľala na riešení úloh z tejto problematiky.

LITERATÚRA

- [1] Nilsson B., Heyden A.: A fast algorithm for level set-like active contours. Pattern Recognition Letters 24, 2003, pp.1331-1337. DOI: 10.1016/S0167-8655(02)00374-4
- [2] Tomori Z., Nikorovič M.: Počítačové videnie v praxi. Košice. Ústav experimentálnej fyziky SAV, 2016. Online: https://home.saske.sk/~tomori/Downloads/Poc_videnie/PV_2016.pdf
- [3] Gladišová, I. - Mihalik, J.: Aktívne obrysové modely. In: Electrical Engineering and Informatics 7, Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, 2016, pp. 31-36.
- [4] Pratt W. K.: Digital Image Processing. California. Los Altos. John Wiley&Sons, 2001.
- [5] Faugeras O.: Image segmentation, a historical and mathematical perspective. Online: <http://miccai.irisa.fr/Program/description/miccai04-slides-faugeras/talk.pdf>
- [6] Španěl M., Beran V.: Obrazové segmentační techniky: Přehled existujících metod. Vysoké učení technické v Brně. Fakulta informačních technologií. Ústav počítačové grafiky a multimédií, 2006.
- [7] Osher S., Fedkiw R.: Level Set Methods and Dynamic Implicit Surfaces. New York. Springer-Verlag, 2003.
- [8] Gladišová, I. - Mihalik, J.: Level Set metóda segmentácie obrazov. In: Electrical Engineering and Informatics 8, Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, FEI TU, 2017, p. 598-602.
- [9] Sethian J. A.: Level Set Methods and Fast Marching Methods. Berkeley. University of California, 1999.
- [10] Mulder W., Osher S., Sethian J. A.: Computing interface motion in compressible gas dynamics. Journal of Computational Physics, 100, 1992, pp.209-228.

Dopad havárií jadrových elektrární na vývoj jadrovej energetiky na Slovensku a vo svete

¹Zuzana GADUŠOVÁ, ²Dušan MEDVEĎ

^{1,2}Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky Technickej univerzity v Košiciach, Slovenská republika

¹zuzana.gadusova.2@student.tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — Článok pojednáva o jednotlivých haváriách na jadrových zariadeniach a ich dopade na budúcnosť jadrovej energetiky. Tento príspevok popisuje aktuálny stav jadrových elektrární na Slovensku. Taktiež rieši uzatváranie JE v jednotlivých krajinách a použitie jadra v boji proti klimatickým zmenám.

Kľúčové slová — jadrová elektrárňa, jadrová energia, havária, vývoj jadrovej energetiky

I. SÚČASNÝ STAV JADROVEJ ENERGETIKY NA SLOVENSKU

K dnešnému dňu sa na Slovensku nachádzajú dve jadrové elektrárne. Sú nimi JE Mochovce a JE Jaslovské Bohunice.

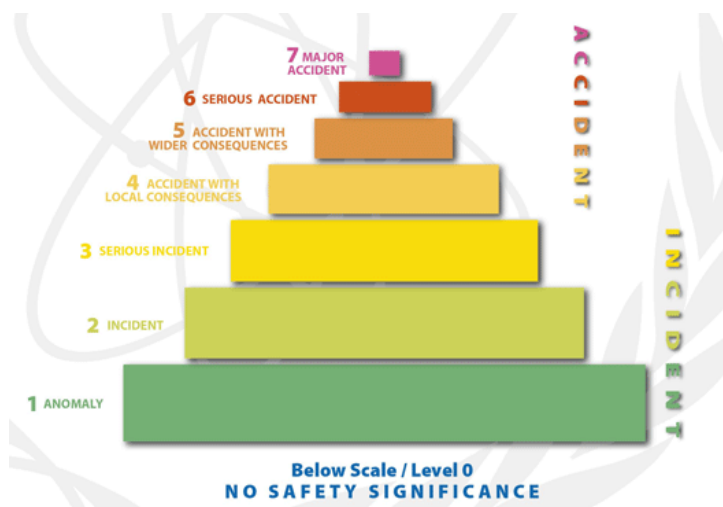
Jadrové elektrárne Mochovce sa nachádzajú na juhu Slovenska. Disponujú štyrmi blokmi s tlakovodnými reaktormi typu VVER-440/V-213, každý s výkonom 470 MW. Tretí a štvrtý blok by mali byť uvedené do prevádzky koncom roka 2021 [1].

V elektrárňach Jaslovské Bohunice sa elektrická energia vyrába v dvoch blokoch s tlakovodnými reaktormi typu VVER-440/V-213, ktoré boli k energetickej sieti postupne pripojené v rokoch 1984 a 1985. Inštalovaný výkon reaktorov je 2×505 MW. Ako chladivo a moderátor je použitá demineralizovaná voda [2].

II. HAVÁRIE NA JADROVÝCH ZARIADENIACH

A. Stupnica INES

Jednotlivú závažnosť jadrových havárií popisuje medzinárodná stupnica hodnotenia závažnosti jadrových a radiačných udalostí (INES). Má osem stupňov závažnosti (0÷7) [3].



Obr. 1 Stupnica INES

B. Havárie jadrových zariadení

Three Mile Island (28. marec 1979)

Príčina: Zlyhanie odľahčovacieho ventilu v kombinácii s chybou operátora pri interpretácii podmienok po odstavení havarijného chladenia reaktora viedli k strate chladiva v reaktore, k prehriatu aktívnej zóny a k čiastkovému taveniu paliva.

Následky: Ohraničené úniky rádioaktivity do vonkajšieho prostredia. Trvalá strata zariadenia, prerušenie výstavby a licencovania ďalších blokov, faktor prispievajúci k zrušeniu objednávok na mnoho ďalších nových JE.

Hlavné výstupy: Dôkladné zmeny v príprave operátorov, v predpisoch pre riadenie havárií, v zohľadnení ľudského faktora, v projekte blokových dozorní, v systéme kontroly a riadenia, vo vykonávaní jadrového dozoru a v havarijnom plánovaní.

Černobyl (26. apríl 1986)

Príčiny: Chyby projektu aktívnej zóny reaktorov RBMK (vrátane kladného dutinového koeficienta), chyby operátorov pri vykonávaní skúšky v oblasti mimo projektovej bázy s blokovanými bezpečnostnými systémami, chýbajúci kontajntment, nedostatočná havarijná odozva.

Následky: Usmrtenie pracovníkov obsluhy v dôsledku radiačného ožiarenia pri požiari, zničenie zariadenia a jeho uzatvorenie do sarkofágu, neriadené úniky a kontaminácia vonkajšieho prostredia, ktorá zasiahla väčšinu Európy a Bieloruska, evakuácia a presídlenie viac ako 336 000 obyvateľov z kontaminovaných oblastí.

Hlavné výstupy: Projektové zmeny reaktorov RBMK (Kanálový varný reaktor s uránovo-grafitovým moderátorom), pre všetky projekty jadrových zariadení, vrátane projektových princípov pre zariadenia na odstavenie reaktora a kontajntment. Ustanovenie nezávislých dozorných orgánov nad jadrovou bezpečnosťou v niektorých krajinách.

Fukushima Daiichi (začiatok 11. marec 2011)

Príčiny (blok č.1 až č.3): Podmorské zemetrasenie s magnitúdou 9 vybudilo cunami, ktoré zaplavilo elektrárňu a malo za následok úplnú stratu elektrického napájania (blackout), nedostatočná ochrana proti cunami v projektovej báze.

Následky (blok č.1 až č.3): Tavenie aktívnej zóny na 3 blokoch, výbuchy vodíka, poškodenie tlakovej nádoby reaktora a primárneho kontajntmenta, veľký neriadený únik rádioaktivity do vonkajšieho prostredia, havarijná evakuácia asi 110 000 obyvateľov, zničené zariadenia.

Hlavné výstupy (blok č.1 až č.3): Mnohonásobné prehodnocovanie v priemyselnom odvetví, vládami a dozornými orgánmi, zvyšovanie bezpečnosti v širokom zábere v celosvetovom meradle, ktoré sa prejavuje v návrhoch na nové základy jadrovej bezpečnosti.

Príčiny (zvyšné bloky): Podmorské zemetrasenie s magnitúdou 9, vybudilo cunami, ktoré viedlo k strate napájania z vonkajšej siete a s rozdielnou úrovňou zaplavili a poškodili zariadenia na 5. a 6. bloku JE Fukushima Daiichi, na 1. až 4. bloku JE Fukushima Daiichi, a na 2. bloku JE Tokai a na blokoch č.1 až č.3 JE Oginava.

Následky (zvyšné bloky): Všetkých 10 blokov JE zvládli udalosť, ale s čiastkovou stratou havarijných kapacít a bezpečnostných rezerv, úplná obnova bola sťažená porušením infraštruktúry v regióne.

Hlavné výstupy (zvyšné bloky): Rozsiahle prehodnocovanie, ako pri blokoch č.1 až č.3. Ďalšia príležitosť využiť poučenie z havárie. Všetky JE v Japonsku sa nachádzajú v studenom odstavenom stave v dôsledku nevyriešeného rozhodovania vlády týkajúceho sa povolenia na spustenie [4].

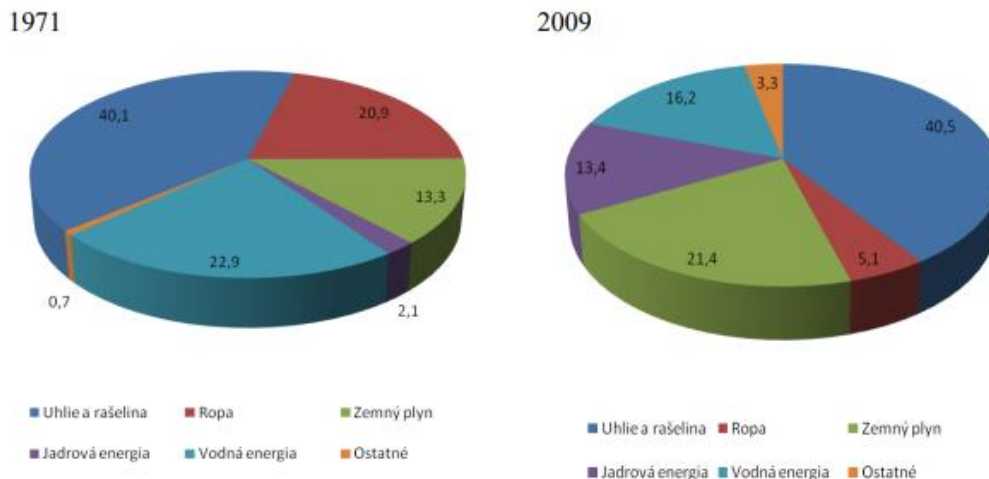
III. UZATVÁRANIE JADROVÝCH ZARIADENÍ

Zaujímavým príkladom pre uzatváranie jadrových zariadení je Nemecko. Vláda Angely Merkelovej, ktorá predtým presadzovala jadrovú energiu, rozhodla sa v dôsledku havárie Fukushima Daiichi v roku 2011 okamžite uzavrieť osem najstarších JE v Nemecku a zatvoriť ďalších 9 do roku 2022 [4]. Aká je však cena za tento krok? Emisie a úmrtia. Prvá závažná skutočnosť bola, že spomínané zrušené jadrové elektrárne boli nahradené hlavne elektrárnami uhoľnými. Prírodné, ekologickým riešením sa to nazvať nedá. Ročne sa tak počet vypustených emisií oxidu uhličitého zvýšil asi o 5 %. Väčšie spaľovanie uhlia zároveň viedlo k väčšiemu lokálnemu znečisteniu ovzdušia, čo má podiel aj na väčšej úmrtnosti ľudí. Celková cena následku zrušenia jadrových elektrární je tiež alarmujúca (cca 10,9 miliardy eur) ročne. Ide o ďaleko vyššiu sumu, akú predtým krajina utrácala na udržiavanie jadrových elektrární. Tieto zistenia jednoznačne hovoria v prospech JE, berúc do úvahy aj potenciálne nebezpečie

súvisiace s nehodou, alebo skladovaním jadrového odpadu [5].

IV. PRODUKCIA ENERGIE VO SVETE

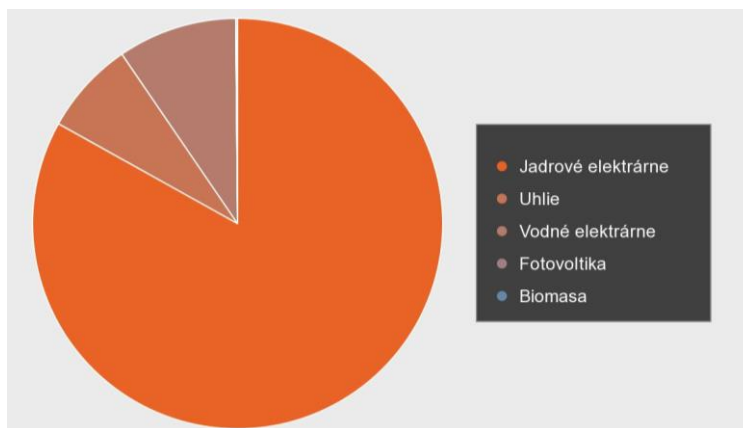
V súčasnosti sa viac ako 85 % celosvetovo spotrebovanej energie získava z fosílnych palív. Tento podiel je skoro konštantný od sedemdesiatych rokov minulého storočia. Vysoké investičné náklady jadrových a obnoviteľných zdrojov spôsobili, že návratnosť investícií do týchto zdrojov je buď neistá, alebo príliš dlhá, aby mohla byť atraktívna pre súkromný sektor [6].



Obr. 2 Podiel energetických zdrojov na výrobe elektrickej energie vo svete za rok 1971 a rok 2009

V. PRODUKCIA ENERGIE NA SLOVENSKU

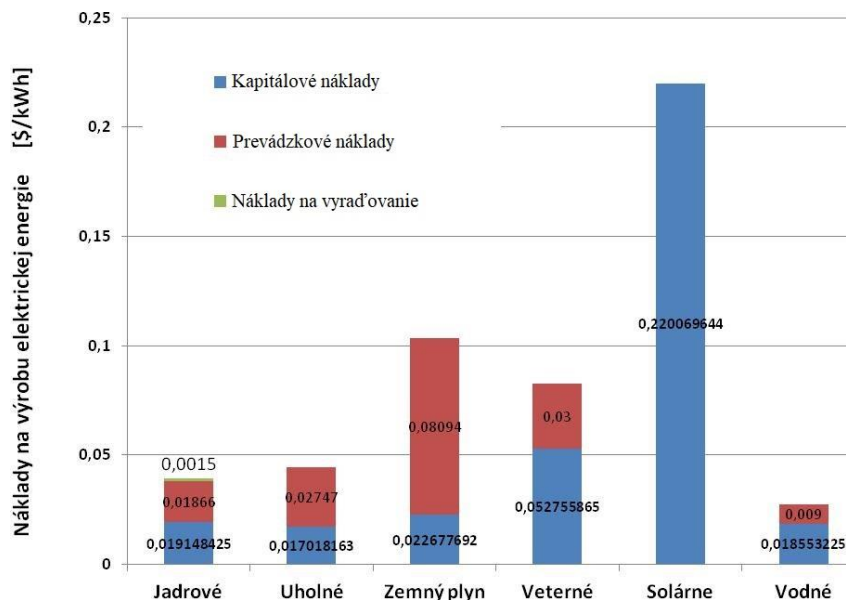
Slovenské elektrárne v roku 2019 celkovo vyrobili 18 865 GWh elektriny. Čistá dodávka elektriny Slovenských elektrární dosiahla 17 097 GWh. Jadrové elektrárne sa podieľali na výrobe elektrickej energie množstvom 83,1 %, uhoľné elektrárne 7,4 % a vodné elektrárne 9,4 % [7].



Obr. 3 Podiel energetických zdrojov na výrobe elektrickej energie na Slovensku za rok 2019

VI. NÁKLADY NA VÝROBU ELEKTRICKEJ ENERGIE

Obnoviteľné zdroje energie ponúkajú istý potenciál na výrobu elektrickej energie. Je pravda, že využitie ich potenciálu v súčasnosti nedosahuje maximálnu hodnotu a sú predpoklady na ďalšie zvýšenie využitia do budúcnosti, ale v dôsledku ich značnej závislosti od zemepisnej polohy a od daných poveternostných podmienok, je pri súčasných technológiách ťažko predstaviteľné, aby dokázali pokryť viac ako 90 % dopytu po elektrickej energii, ako to predpokladá jeden z uvedených scenárov Európskej komisie. Výhodou jadrových elektrární oproti obnoviteľným zdrojom, ktoré sú charakterizované prerušovaným a menej stabilným priebehom dodávky elektrickej energie, je ich dlhodobá stabilná prevádzka v základnom pásme diagramu zaťaženia. Jadrová energia je jediný energetický zdroj neprodukujúci CO₂, ktorý pri súčasných technológiách dokáže byť nasadený vo veľkom rozsahu a poskytuje značný priestor pre ďalší rozvoj do budúcnosti. Jedným z najdôležitejších faktov pri výbere energetických zdrojov je cena výroby elektrickej energie pre daný zdroj [8].



Obr. 4 Porovnanie energetických zdrojov na základe ceny výroby elektrickej energie

VII. POUŽITIE JADRA V BOJI PROTI KLIMATICKÝM ZMENÁM

A. Jadrová energia = Čistá energia?

Jadrová energetika je v podstate bez emisií. Ako príklad, produkcia CO₂ prepočítaná na 1 kWh produkovanej energie je asi 50-krát nižšia, než pri spaľovaní uhlia. Tradične je zelený nesúhlas s jadrovou energetikou zakorenený predovšetkým v strachu z radiácie uvoľnenej pri havárii reaktora, alebo z rádioaktívneho odpadu z úložísk. Dnes viaceré krajiny budujú hĺbkové podzemné úložiská, kde môžu bezpečne uskladniť vysoko-rádioaktívny odpad.

B. Budúcnosť jadrovej energie

Elektrina z jadra nikdy lacná nebola a dnes zápasí o konkurencieschopnosť čisto na základe ekonomických ukazovateľov v porovnaní s elektrinou vyrobenou spaľovaním zemného plynu. Avšak príbeh vyznieva úplne inak, ak vezmeme do úvahy aj emisie skleníkových plynov, ako náklad sám o sebe. V priemere za dobu životnosti elektrárne má spaľovanie uhlia za následok 979 ton oxidu uhličitého (na 1 GWh) uvoľneného do atmosféry. Z plynu sa uvoľní 550 ton. Hodnota pre elektrinu z jadra je 32 ton [9].

C. Obnoviteľné zdroje alebo jadro

Na základe analýzy skúseností mnohých krajín vyplýva, že to, čo môže trvať celé storočie s obnoviteľnými zdrojmi, by mohlo byť vykonané za 20 rokov s jadrovou energiou. Jadrová energia nevypúšťa do ovzdušia uhlík, je vysoko koncentrovaná, čo minimalizuje vplyv na životné prostredie z ťažby odpadu a funguje 24 hodín denne, bez potreby batérií. Najdôležitejšie je, že sa môže rýchlo vystupňovať. Jadro tiež vytvára oveľa menej odpadu ako iné zdroje energie, vrátane obnoviteľných zdrojov energie. Vyhorené palivo z jadrovej výroby elektrickej energie spotrebovanej za života priemerným Američanom by sa vmestilo do plechovky od koly. Jadrová energia nemusí byť ani príliš drahá. Existujú jadrové elektrárne v USA, ktoré vytvárajú jednu pätinu národnej elektrickej energie, produkujú menej nákladnú energiu ako uhlie alebo plyn. V Južnej Kórei elektrina z jadrovej energie stojí menej ako 4 centy/kWh, čo je lacnejšie ako elektrina z akéhokoľvek iného zdroja [10].

VIII. ZÁVER

Tento príspevok sa zaoberal dopadom havárií jadrových elektrární na vývoj jadrovej energetiky na Slovensku a vo svete. Z príspevku vyplýva, že aj napriek skutočnosti vážnosti jednotlivých havarijných udalostí, je jadrová energia pilierom pri energetickom mixe v jednotlivých krajinách a nedá sa 100 %-ne nahradiť obnoviteľnými zdrojmi energií.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18.

LITERATÚRA

- [1] SEAS, „*AE Mochovce*“ [online] [cit. 01-05-2020] Dostupné na internete: < <https://www.seas.sk/ae-mochovce> >
- [2] SEAS, „*AE Jaslovské Bohunice*“ [online] [cit. 01-05-2020], Dostupné na internete: < <https://www.seas.sk/ae-bohunice> >
- [3] INES, „*Mezinárodní stupnice hodnocení závažnosti jaderných a radiačních událostí*“ [online] [cit. 08-05-2020], Dostupné na: < https://www.sujb.cz/fileadmin/sujb/docs/dokumenty/publikace/INES-2008_cz_preklad.pdf >
- [4] VUJE, a.s., „*Významné havárie na jadrové energetických zariadeniach*“
- [5] SNUS, „*Nemecké energetické spoločnosti varujú, že odstavenie jadra povedie k chýbajúcej energii*“ [Online] [cit. 11-05-2020] Dostupné na internete: < <https://www.nuclear.sk/nemecke-energeticke-spolocnosti-varuju-ze-odstavenie-jadra-povedie-k-chybajucej-energii/> >
- [6] SNUS, „*Cena zrušenia jadrových elektrární v Nemecku? Miliardy ročne, zvýšenie emisií aj počtu úmrtí, hovorí ŠTÚDIA*“ [Online], Dostupné na internete: < <https://denniks.sk/21824/cena-zrusenia-jadrovych-elektrarni-v-nemecku-miliardy-ročne-zvysenie-emisii-aj-poctu-umrti-hovori-studia/> >
- [7] Haščík, J., „*Prečo vyrábať elektrinu v jadrovej elektrárni?*“ [Online] [cit. 08-05-2020] Dostupné na internete: < http://www.dukonline.sk/tmp/asset_cache/link/0000018326/4.%20predn%C3%A1%C5%A1ka.pdf >
- [8] SEAS, „*Atómová elektrárň*“ [Online] [cit. 01-05-2020] Dostupné na internete: < <https://www.seas.sk/atomova-jadrova-elektraren> >
- [9] SNUS, „*Hans Blix: Chcete zastaviť klimatickú zmenu? Tak je čas vrátiť sa k starej láske: k jadrovej energii*“ [Online] [cit. 11-05-2020] Dostupné na internete < <https://www.nuclear.sk/hans-blix-chcete-zastavit-klimaticku-zmenu-tak-je-cas-vratit-sa-k-starej-laske-k-jadrovej-energii/?fbclid=IwAR3SUWuHMGloQ5akL4xRmmiiUAiOivbiil5O7nuB9szl8vEMjD1jcRx8InE> >
- [10] SNUS, „*Boj proti klimatickým zmenám potrebuje 100-200 jadrových elektrární za rok*“ [Online] [cit. 11-05-2020] Dostupné na internete: < https://www.nuclear.sk/boj-proti-klimatickym-zmenam-potrebuje-100-200-jadrovych-elektrarni-za-rok/?fbclid=IwAR0IMxbmG8x4BiIAfjVlnOlrJhajoVHq1eFAqvH1IYOgP_6YaTg36ikU-dY >

Ensuring cyber security in the Internet of Things environment

¹Jana Handriková, ²Peter Šul'aj

¹ Department of Technical Science and Informatics, Faculty of Security Engineering, University of Žilina, Slovak Republic

² Department of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹jana.handrikova@fbi.uniza.sk, ²peter.sulaj@tuke.sk

Abstract — The spread of information technology to all areas of life is associated with a growing need to ensure that these systems are protected from destruction, misuse or failure. Information technologies are currently an integral part of industrial control systems, including systems ensuring the correct and error-free operation of critical infrastructure equipment. It is not necessary for cyber security to become an important part of the requirements set for a system before it is deployed. In the Internet of Things environment, these are distributed systems, which are characterized by heterogeneity of the computing environment. It is necessary to examine and evaluate the cyber security of all components of such systems - cyber security of terminal equipment, transmission networks and processing sites – servers. The car plate recognition system presented in the article uses several security mechanisms of cyber security.

Keywords — BOINC, car plate recognition system, cyber security, internet of things

I. INTRODUCTION

The spread of information technology to all areas of life is accompanied by a growing need to ensure its availability, to enable their use by the general professional and lay public, but it is also associated with a growing need to ensure protection of these systems from destruction, misuse or malfunction. Information technologies are currently an integral part of industrial control systems, including systems ensuring the correct and error-free operation of critical infrastructure facilities. Related to these facts are efforts to find procedures that will ensure the identification and detection of threats to these systems, enable the protection of the organization's assets and use the knowledge gained from the analysis of the course of early detected and undetected attacks on these systems.

II. CYBER SECURITY

A. Definition

Cyber security is defined in the Concept of Cyber Security of the Slovak Republic for the years 2015 - 2020 [9] as follows: "Cyber security is a system whose task is to provide the information environments of the socio-economic structure of the state with a secure, protected and, to an appropriate extent, open cyberspace, i.e. j. a guarantee of the security of the electronic information, communication and control systems located in that area, the data stored, processed and transmitted in those systems, as well as the services provided by those systems. "

However, ensuring the cyber security of an organization is not just a technical issue. When defining cyber security, it is necessary to start from the model of cyber security management, which defines six main areas critical for determining the processes of cyber security in the organization. These areas are (processed according to [3]):

1. legal framework in which the organization operates,
2. good governance (practices established in the organization),
3. risk management,
4. security culture in the organization,
5. technology management,
6. incident management.

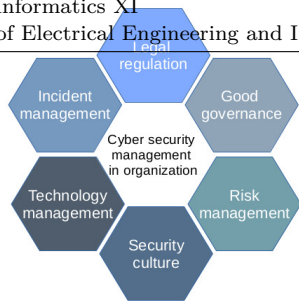


Fig. 1 Cyber security management model [3].

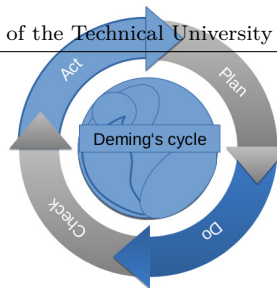


Fig. 2 Deming's cycle

It should be emphasized that the process of ensuring the cyber security of an organization is a cyclical and never-ending process, as illustrated by Deming's model. He points out that the phases of planning, implementation of planned activities, control of their implementation and verification of their impact on the managed process together with the last phase of process improvement in the organization are a cyclically repetitive process.

III. INTERNET OF THINGS

There is no uniform definition of the Internet of Things. In general, the Internet of Things can be characterized as a system whose goal is to process signals from the surrounding physical environment into digital form for the purpose of their subsequent processing and analysis. Based on them, new information about the scanned environment can be obtained and a corresponding, adequate response can be elicited. These systems consist of a large number of diverse external devices that communicate with each other through (not only) computer networks. The development of information technologies brings not only an improvement in performance parameters and an increase in the number of sensor types, but also a reduction in the cost of calculations. Therefore, it is possible to move calculations from the central point (s) to these end devices.

In [5], three main shortcomings in the Internet of Things environment are highlighted in terms of cyber security:

- large number of diverse endpoints,
- inconsistent protocols,
- physical security concerns.

At present, there are no mechanisms in place to manage the consistent security of endpoints for such a large-scale system. In addition, the diversity of standards within the Internet of Things brings the responsibility of each individual actor in the technology chain for security. The security of terminal equipment, whether ensuring the legitimacy of their use through access control, or ensuring the security of communications and transmitted data intended for further analysis, is becoming one of the important areas of cyber security. In addition, these devices can also be used as sensors in the field of physical security of persons and property.

As the Internet of Things can be characterized as a decentralized system of cooperating intelligent objects, distributed computing systems are playing an increasingly important role in this environment, which also perform their tasks using methods, procedures, artificial intelligence algorithms, such as machine learning.

The use of proven distributed systems, such as volunteering systems such as BOINC, can also help to increase cyber security in the inhomogeneous Internet of Things environment.

IV. BOINC SYSTEM

A. Basic characteristics of the BOINC

The BOINC (Berkeley Open Infrastructure for Network Computing) software platform can be used for volunteer computing and computer grids. The system is designed to support applications and with high resource requirements for calculations, storage, or both. The main requirement for the application is the ability to divide it into a large number of mutually independent tasks. This, among other things, allows resources to be shared between autonomous projects.

The basic features of the BOINC system are [7]:

1. Project autonomy.
2. Flexibility of volunteers.
3. Flexible application structure.
4. Security.
5. BOINC code availability.
6. Server performance and scalability.

7. Support big data solutions.
8. Many usable platforms.
9. Open, extensible platform.
10. Support of the volunteer community.

B. Cyber security of the BOINC system

BOINC is a freely distributable system, so its misuse cannot be ruled out. It can be the creation of fraudulent projects that appear to be projects focused on a research area of benefit to humanity, or imitations of established projects that mimic the original websites of established projects but are a source of malware. BOINC provides a list of verified running projects on its website. If the project is not in the given list, it is recommended to verify the project domain, contact the organization that is the author of the project, verify the references on the BOINC project website.

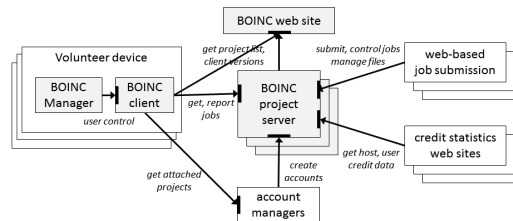


Fig. 3 BOINC system diagram [10]

- The website [6] describes some types of security problems and built-in options for their elimination, which are related to volunteer calculations, i. j. calculations in a heterogeneous environment that requires the use of distributed computing. These issues are:
 Falsifying results. Attackers return incorrect results.
- Credit forgery (ratings). Attackers return results and declare higher CPU usage time than they actually were.
- Distribution of malicious executable files. Attackers infiltrate the BOINC server and, by modifying the database and files, try to distribute their own executable files (such as virus programs) that are disguised as BOINC applications.
- Data server capacity exceeded. Attackers repeatedly send large files to BOINC data servers, filling their disks and degrading them.
- Theft of a participant's account information by an attack on a server. Attackers break into a BOINC server and steal email addresses and other account information.
- Theft of a participant's account information by a network attack. Attackers use BOINC protocols to steal account information.
- Theft of project files. Attackers steal input and / or output files.
- Intentional misuse of host computers by projects. The project intentionally distributes an application that abuses host participants, e.g. by stealing sensitive information stored in files.
- Unintentional misuse of host computers by projects. The project releases an Application that inadvertently abuses the host, e.g. deleting files or causing failures.

Mechanisms that can detect or reduce the risk of falsifying results include the ability to use replication, where the task is performed on n hosts, and rules, that determine the conditions under which the result is considered valid, are set out. It is possible to use homogeneous and adaptive replication of tasks, fuzzy logic can be used to determine the limits of tolerance of the validity of the result.

In an effort to prevent the spread of malware, BOINC uses a public signature procedure based on cryptography using a public key to authenticate executables and the BOINC project server. Each project has a pair of keys to sign the code, which consists of a public key and a private key. The public key is sent to the client after the project is added. The private key is used to create signatures for programs. The BOINC client will only run programs with valid signatures. Projects are instructed to keep the private key only on a computer that is never connected to the network and that is physically secure. In the interests of the cyber security of their systems, project participants should ask project administrators for a description of the use of this mechanism. If this mechanism is used incorrectly for publicly available projects, the BOINC project can be used to distribute malware.

BOINC provides an optional certificate mechanism to prevent attacks on data servers. Each output file is assigned a maximum size. Each project has a pair of authentication keys to upload. The public key is stored on the project's data servers. Descriptions of the resulting files are sent to the client with a digital signature, which is sent to the data server when the file is uploaded. The data server verifies the file description and ensures that the amount of data transferred does not exceed the maximum size.

In an effort to prevent the theft of information about the accounts of project participants by attacking the BOINC project server, the necessary application of standard security procedures is emphasized - the use of a firewall and the shutdown of all unused network services. Access to the server should only be using encrypted protocols such as SSH. Servers should be subject to regular security audits.

BOINC does not provide any mechanisms to reduce the risk of obtaining project participant account information by attacking a computer network - attackers who break into network traffic can obtain a user's account key and use it to obtain a user's e-mail address or change user preferences.

Theft of project files is also a real threat, as the input and output files used by BOINC applications are not encrypted. Applications can do this themselves, but it has little effect because the data is in text format in memory, where it is easy to access using a debugger.

Another security mechanism that protects against intentional misuse of hosts by project participants is the application execution department - applications run under a specially created account. If file and directory permissions are set correctly, applications cannot access files outside the BOINC directory. An even stronger mechanism is the use of virtual machines.

In the event of unintentional misuse of hosts by project participants, BOINC tries to avoid some problems: for example, it detects when applications are using too much disk space, memory, or CPU time and interrupts the execution of those applications. Projects can minimize the likelihood of causing problems by pre-testing applications. Projects should thoroughly test their applications on all platforms and with all input scenarios before upgrading them to production status.

V. PROPOSED SYSTEM

A. Requirements for the functionality

The aim was to design a distributed system using the BOINC system, which will solve the task of physical security of the area by recording and automatically recognizing the registration numbers of vehicles passing through checkpoints. The OpenAlpr software package was used for the recognition of registration numbers, which was modified for use in the BOINC system, and the program was parallelized during the processing of individual video frames.

The requirements for the functionality of the system were set as follows:

1. The device for the initial processing of a video signal must be capable of processing, in addition to storing the recorded recording, its processing, i. j. abstraction of higher level information (in this particular implemented example of ECV recognition). This presupposes a relatively large computing power and a favorable price-performance ratio.
2. Equipment for the initial processing of the video signal should be commonly available.
3. The electricity consumption of the video processing equipment for the initial processing of the video signal should be as low as possible, due to the possibility of relocating the recording equipment and minimizing its operation and maintenance costs (battery and photovoltaic power supply).
4. There is no need for high capacity available to store data - the prerequisite is to send the extracted information for further processing at regular intervals.
5. A system for the post-processing of data extracted from video should be able to process a large number of independent tasks.
6. A system for the post-processing of data extracted from video should be able to manage a large data repository.
7. A system designed for the subsequent processing of data extracted from video should be able to handle even computationally intensive requirements.
8. Good availability of the system for subsequent processing of data extracted from video.

Based on the requirements set out in this way, the Raspberry Pi was chosen as the terminal device for recording and extracting data from this video. The Raspberry PI is a single-chip computer integrated on a small printed circuit board, developed by the British Raspberry Pi Foundation. It is designed primarily for single-purpose applications such as control, regulation, etc., mainly in the field of research, teaching and education.

B. Security mechanisms used in system development

Due to the separation of the development environment, a part of the system for extracting properties was developed for concentrating, storing and managing the extracted properties on a virtual machine, specifically VirtualBox from Oracle was used.

Oracle VM VirtualBox allows you to create and run multiple virtual machines, running on different operating systems, on the same computer at the same time. Oracle VM VirtualBox is available as Open Source or precompiled binaries for Windows, Linux, Mac OS X, and Solaris.

The http protocol is used to communicate between the client and the server. If you need increased protection of the data transferred between the server and the client, it is possible to use the https protocol. The aim of the system design was not to achieve maximum operational safety of the system, but to develop and test a system for recognizing vehicle registration numbers and subsequent processing of this data.

One of the cyber security mechanisms provided by BOINC is a system used to authenticate executables and a BOINC project server using a digital signature procedure based on public key cryptography.

It follows from the BOINC system architecture that this project, like all projects running in the BOINC system environment, has its own directory created on the client (participating) computer. All files related to this project - application, input and output files are stored in the project directory (specifically cplate).

Setting the parameters of the client program (vehicle registration program) allows you to determine the maximum system resources that will be available to solve this task. The BOINC client settings allow great variability for defining the times in which the task is to be solved. If multiple clients are assigned to one account, it is possible to control them by changing the settings in the user account.

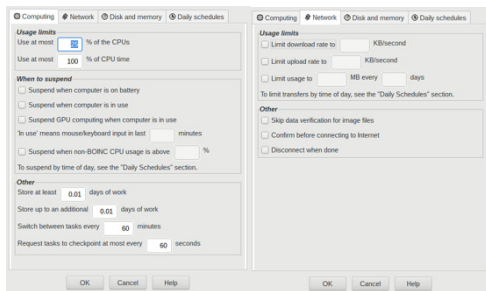


Fig. 4 Display the settings of the BOINC client, Computing and Network tab parameters

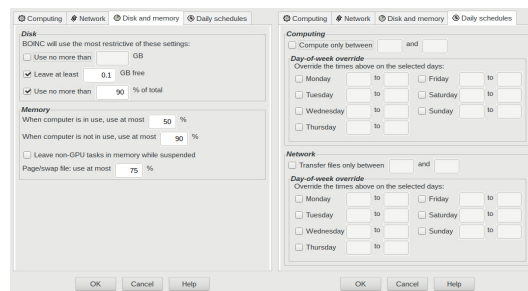


Fig. 5 Display the settings of the BOINC client parameters, the Disk and memory tab and the Daily schedules tab

An integral part of the security solution of any digital environment is security management, which also includes the ability to monitor the behavior of the system. The processing of individual tasks can be monitored in the graphical interface of the BOINC client. The overall status of the project can be monitored on the project website.

The application was tested using a printed image with the vehicle registration number. The method of testing the application was chosen with regard to the applicable legislation in the field of personal data protection and imaging of public spaces by a camera system.

In this case, the system validator was defined very simply - each result was considered correct. If necessary, it is possible to incorporate more complex functionality into the validator - for example, homogeneous or adaptive replication of tasks, fuzzy logic can be used to determine the limits of tolerance of the validity of the result.

Because the BOINC server was installed within the Intranet, no mechanisms were built in to protect the system from computer network attacks. The entire hardware and software infrastructure was implemented within the computer network of the university, which is protected by the use of a firewall.

C. Results

The achieved results showed that it is possible to implement the BOINC system as a basic communication tool between Internet of Things devices and system servers intended for the subsequent processing of data extracted from video recording. The possibility of deploying server applications on PCs with possible distribution of the load of individual parts of the BOINC system on multiple PCs, the use of Raspberry Pi as a terminal device, the price of which is around 50 Euros, allows full research, development and testing of such applications in signal processing for

integrated security systems. In the case of real deployment, in addition to the mentioned security measures, it would be necessary to ensure physical protection, especially of terminal equipment designed to capture the parameters of the surrounding world, with emphasis on their resistance to weather conditions and deliberate destruction. The critical point of cyber protection of such systems is to ensure the protection of the database with stored patterns of verified, resp. identified objects.

VI. CONCLUSION

During the implementation of the system for the recognition of vehicle registration numbers, some of the built-in security mechanisms were used, which are part of the cyber security of the BOINC system, primarily intended for the implementation of volunteer projects. It is necessary to point out that the BOINC system also provides other mechanisms of cyber security, which can be added to the created system, and thus increase its security in a heterogeneous environment of the Internet of Things. As the BOINC system is primarily intended for distributed computing that takes place in the environment of systems communicating with each other, whether via a fixed computer network, mobile wireless connection or other means, the security of such systems cannot be considered as an isolated system.

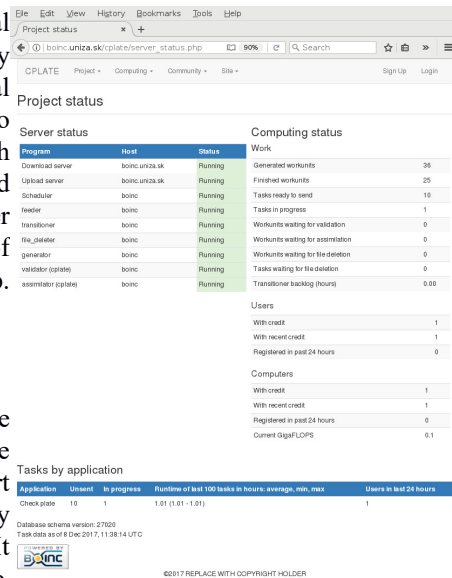


Fig. 7 Overall status of the project

REFERENCES

- [1] Şeker, Ensar. (2019). Use of Artificial Intelligence Techniques / Applications in Cyber Defense. Available on https://www.researchgate.net/publication/333477899_Use_of_Artificial_Intelligence_Techniques_Applications_in_Cyber_Defense 30.01.2020
- [2] Vähäkainu, Petri & Lehto, Martti. (2019). Artificial intelligence in the cyber security environment Artificial intelligence in the cyber security environment. Available on https://www.researchgate.net/publication/338223306_Artificial_intelligence_in_the_cyber_security_environment_Artificial_intelligence_in_the_cyber_security_environment/citation/download, cited 30.01.2020
- [3] Limba, Tadas & Plėta, Tomas & Agafonov, Konstantin & Damkus, Martynas. (2017). Cyber security management model for critical infrastructure. ENTREPRENEURSHIP AND SUSTAINABILITY ISSUES. 4. 559-573. 10.9770/jesi.2017.4.4(12). Available on https://www.researchgate.net/publication/317715298_Cyber_security_management_model_for_critical_infrastructure/citation/download, cited 30.01.2020
- [4] B. A. Khalaf, S. A. Mostafa, A. Mustapha, M. A. Mohammed and W. M. Abdulllah, "Comprehensive Review of Artificial Intelligence and Statistical Approaches in Distributed Denial of Service Attack and Defense Methods," in *IEEE Access*, vol. 7, pp. 51691-51713, 2019. doi: 10.1109/ACCESS.2019.2908998 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8692706&isnumber=8600701> 30.01.2020 unpublished.
- [5] M. S. Jalali, J. P. Kaiser, M. Siegel and S. Madnick, "The Internet of Things Promises New Benefits and Risks: A Systematic Analysis of Adoption Dynamics of IoT Products," in *IEEE Security & Privacy*, vol. 17, no. 2, pp. 39-48, March-April 2019. doi: 10.1109/MSEC.2018.2888780, <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8677360&isnumber=8677281>, cited 30.01.2020
- [6] <https://boinc.berkeley.edu/trac/wiki/SecurityIssues>, cited 30.01.2020
- [7] <https://boinc.berkeley.edu>, cited 30.01.20207
- [8] <https://www.csirt.gov.sk/bezpecnostna-studovna/organizacna-bezpecnost/zavadzanie-informacnej-bezpecnosti-do-organizacie-876.html>, cited 30.01.2020
- [9] <https://www.nbu.gov.sk/wp-content/uploads/kyberneticka-bezpecnost/Koncepcia-kybernetickej-bezpecnosti-SR-na-roky-2015-2020-A4.pdf>
- [10] David P. Anderson: BOINC: A Platform for Volunteer Computing, Available on <https://arxiv.org/pdf/1903.01699.pdf>, cited 24.04.2020.
- [11] P. Šul'aj, R. Haluška, L. Ovseník, S. Marchevský, "Examples of Real-Time UAV Data Processing with Cloud Computing", In: 23rd Conference of Open Innovations Association FRUCT., Helsinki (Finland): FRUCT Oy, 2018, p. 543-548, ISBN 978-952-68653-6-2, ISSN 2305-7254
- [12] P. Šul'aj, R. Haluška, L. Ovseník, S. Marchevský, S. Marchevský, P. Pulli, V. Kramar: "UAV Management System for the Smart City" In: DISA 2018: IEEE World Symposium on Digital Intelligence for Systems and Machines: proceedings, Danver (USA): Institute of Electrical and Electronics Engineers, p. 119-124 [online]. ISBN 978-1-5386-5101-8

Návrh alternatívneho tepelného zdroja pre kláštor Krásny Brod

¹Patrik KANCÍR, ²Dušan MEDVEĎ

^{1,2} Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky Technickej univerzity v Košiciach, Slovenská republika

¹patrik.kancir@student.tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — Príspevok pojednáva o návrhu vykurovacieho systému pre Monastýr Krásny Brod pri jej aktuálnom stave. V budove sú aktuálne implementované niektoré druhy energetických zdrojov, ku ktorým je požiadavka pridať ďalší energetický zdroj v podobe tepelného čerpadla vzduch-voda. Tento príspevok popisuje aktuálny stav energetických zdrojov v budove, opis budovy a jej parametre, ktoré sú potrebné pre výpočet tepelných strát budovy. Na ich základe sa stanovuje tepelný zdroj k súčasným podmienkam. Výsledkom analýzy energetických potrieb je výber konkrétneho druhu a typu tepelného zdroja.

Kľúčové slová — tepelné čerpadlo, tepelné straty budovy, vykurovanie

I. SÚČASNÝ STAV BUDOVY MONASTÝR KRÁSNY BROD

Budova Monastýru Krásny Brod bola postavená v roku 2002. Nachádza sa v obci Krásny Brod, neďaleko okresného mesta Medzilaborce na východnom Slovensku. V nej sa nachádza suterén a dve nadzemné podlažia. Na prvom nadzemnom podlaží sa nachádzajú ubytovacie izby, spoločenská miestnosť, jedáleň a kuchyňa. Na druhom poschodí sa nachádza veľká učebňa, 20 ubytovacích izieb, skladové miestnosti a malá kaplnka. V treťom poschodí je umiestnená knižnica.

V suteréne sa nachádza kotolňa a priestory využívané ako sklad. V súčasnom systéme vykurovania sa ako zdroj tepla využíva kondenzačný plynový kotol značky Junkers s menovitým tepelným výkonom 25 kW. Druhým zdrojom je kotol na tuhé palivo značky Defro s menovitým tepelným výkonom 45 kW. Teplá úžitková voda v 300 l nádrži sa ohrieva prostredníctvom štyroch solárnych panelov. V prípade potreby sa voda ohrieva plynovým kotlom.

Vedenie vykurovania je rozdelené do týchto štyroch vetiev:

1. prízemie
2. poschodie
3. chodby
4. chrám – vzdialený 70 m od budovy monastýru (podlahové kúrenie).

Vykurovací systém obsahuje jednu expanznú 200 l nádrž, ktorá slúži ako ochrana systému pri zmenách tlaku vody.



Obr. 1 Pohľad na budovu Monastýru Krásny Brod

II. ZDROJE TEPELNEJ ENERGIE

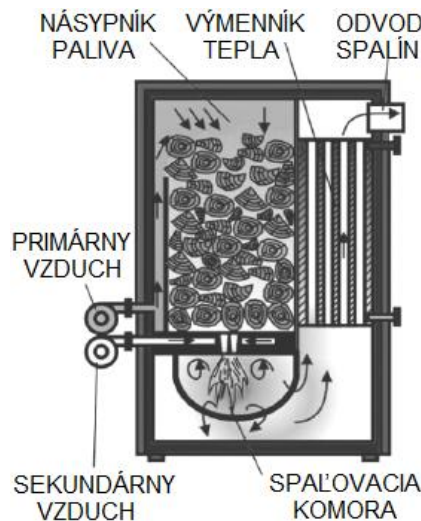
Energetika vykurovania sa v posledných rokoch mení. Do popredia sa uprednostňujú alternatívne spôsoby vykurovania, ako napríklad: solárne vykurovanie, fotovoltaické systémy, tepelné čerpadlá, solárne kolektory. Hlavnými dôvodmi tejto zmeny je snaha zníženia nákladov na vykurovanie a vyčerpateľnosť klasických zdrojov.

A. Kotel na tuhé palivo

Biomasa, ako palivo, je najstarším zdrojom energie na svete. Pod pojmom biomasa však nerozumieme len kusové drevo, drevný odpad, ako je kôra, štiepka, piliny, slama či tiež suché časti rastlín na účel spaľovania, akými sú topoľ, osika, šľaveľ, konope a pod.

Kotel na tuhé palivo je charakteristický manuálnym nakladaním paliva do kotla cez horné alebo čelné nakladacie dvierka. Pri týchto typoch kotlov je potrebné venovať pozornosť bezpečnej prevádzky kotla a to najmä v prípade výpadku elektrickej energie, kedy by generovaný, neodoberaný tepelný výkon, mohol spôsobiť neprimerané zvýšenie teploty a tlaku vo vykurovacom systéme. Aby sa predišlo takémuto scenáru, používa sa tzv. bezpečnostná batéria s prívodom chladiacej (studenej) vody do bezpečnostného výmenníka. Bezpečnostný ventil studenej vody sa otvára pri teplote 95 °C. Súčasne pri týchto typoch kotlov nie je možné regulovať ich tepelný výkon a proces spaľovania dodávkou paliva, ale množstvom privádzaného primárneho a sekundárneho spaľovacieho vzduchu [1].

Pre svoju vysokú účinnosť a dostupnosť paliva je tento typ tepelného zdroja pomerne značne žiadaný aj v súčasnosti. Najviac sa používa v miestach bez dostupnosti zemného plynu. Ale pri porovnaní s novšími zdrojmi tepla je biomasa ekonomicky neefektívna, nakoľko cena paliva, či už v podobe kusového dreva alebo vo forme brikiet v posledných rokoch rastie. Kúrenie drevom má najmenej komfortnú obsluhu a reguláciu. Príkladanie dreva do kotla nie je automatizované, treba ho zabezpečiť ručne a to spravidla 3-krát za deň. Je potreba vhodných skladovacích priestorov, pretože kotly na drevo sú citlivé na kvalitu a vlhkosť dreva. Spaľovanie dreva je spojené s vysokou produkciou emisií a to najmä vysoko-škodlivých tuhých znečisťujúcich častíc a CO₂.



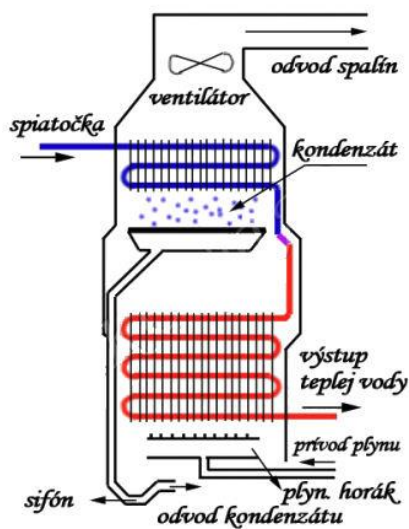
Obr. 2 Kotel na tuhé palivo s ručnou obsluhou [1]

B. Plynový kondenzačný kotel

Historický je známa aj plynofikácia Slovenska, ktorá začala v 70-tych rokoch minulého storočia. S postupnou plynofikáciou a dostupnosťou plynu sa začali implementovať kotly na plyn.

Plynový kondenzačný kotel predstavuje špičku vo vykurovacej technike. Je orientovaný na čo najnižšiu spotrebu paliva a dosiahnutie maximálneho výkonu. Pri jeho prevádzke sa využíva energia z paliva (okrem citelného tepla aj latentné teplo viazané v spalinách), ktoré je ochladzované v kondenzačnom výmenníku vratnou vykurovacou vodou s teplotou (20 ÷ 50 °C) nižšou ako je teplota rosného bodu spalín [1].

Ide o častú a obľúbenú alternatívou vykurovania, nakoľko toto vykurovanie je veľmi pohodlné a nenáročné na ovládanie. Je to ekologické zariadenie s minimálnou produkciou emisií. Má čistú a komfortnú prevádzku. Jeho účinnosť prípravy tepla je až 97 %. Nie je potrebný skladový priestor na drevo či palety. Je však potrebná pravidelná prehliadka kotla. Pri vzniku poruchy kotla vzniká riziko úniku zemného plynu.



Obr. 3 Kondenzačný plynový kotol [2]

III. ZDROJE TEPELNEJ ENERGIE

Vymeniť vykurovacie zariadenie je všeobecne veľmi dobré rozhodnutie. Vyššia účinnosť nových zariadení podstatne znižuje energetickú spotrebu vykurovacieho systému. Výmena zariadenia vedie k nižším prevádzkovým nákladom a má menší vplyv na životné prostredie.

Ochrana životného prostredia sa stáva v mnohých oblastiach nášho bežného života úplnou samozrejmosťou. Využitie alternatívnych zdrojov energií, elektromobilov, úsporných zdrojov svetla, nízkoenergetických domácich spotrebičov, to sú len najzákladnejšie príklady. Zamýšľať sa treba aj nad príspevkom k ochrane životného prostredia, vzhľadom k spôsobu vykurovania.

K ekologickým zdrojom vykurovania sa bezpochyby zaraďujú aj tepelné čerpadlá. Vykazujú výborné stupne účinnosti, preto sú vhodné aj pre nepretržitú prevádzku a zároveň poskytujú maximálny komfort [3].

Tepelné čerpadlo je nezávislé od distribučnej siete tepla a distribúcií palív. Neovplyvňuje ho dostupnosť tepla z teplárne, plynu v plynovode či zásoby uhlia predajcov fosílnych palív. Nie je pri ňom nutné mať priestory na skladovanie zásob dreva alebo uhlia. Zdroj tepla je vždy pod kontrolou a na dosah. Tepelné čerpadlá využívajú energiu napríklad zo vzduchu či geotermálneho vrtu. Ich využitie nijako nepoškodzuje krajinu a nemá zásadný negatívny vplyv na životné prostredie.

Na naše podnebné podmienky má tepelné čerpadlo pri stálej údržbe dlhú životnosť. Napomáha k tomu neustály vývoj, ktorý z neho urobil spoľahlivejší a výkonnejší zdroj tepla.

Tepelné čerpadlá majú aj iné vlastnosti, napríklad vedú aj chlaďať, preto sa v lete môže využiť aj ako klimatizácia. Dajú sa použiť na ohrev vody vo vonkajšom bazéne, môžu sa nastaviť a komunikovať s fotovoltaickým zariadením [4].

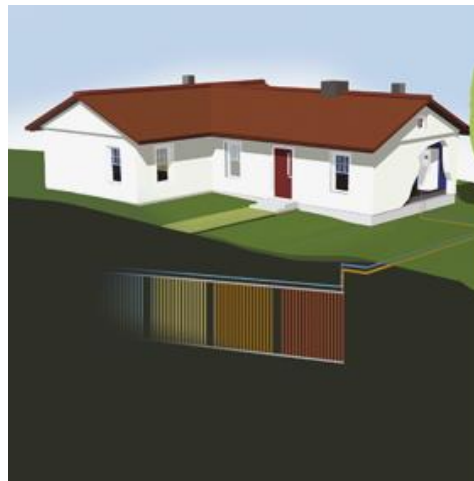
A. Typy tepelných čerpadiel



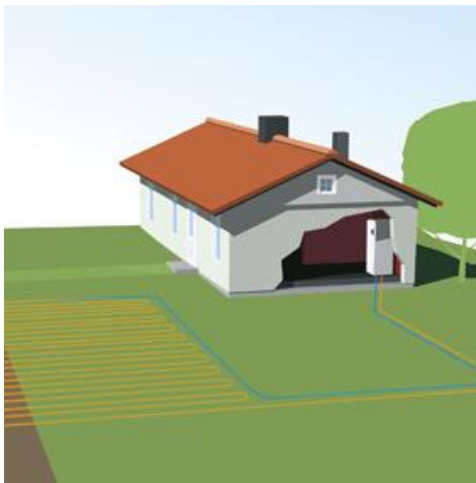
Obr. 4 Tepelné čerpadlo typu vzduch – vzduch [5]



Obr. 5 Tepelné čerpadlo typu voda – vzduch [5]



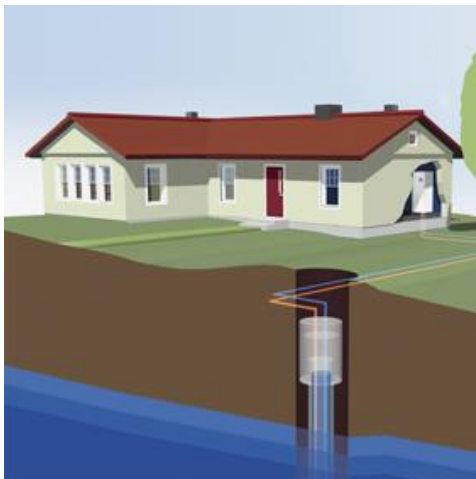
Obr. 6 Tepelné čerpadlo typu vetrací vzduch – voda [5]



Obr. 7 Tepelné čerpadlo typu zem – voda [5]



Obr. 8 Tepelné čerpadlo typu zem – voda (plocha) [5]



Obr. 9 Tepelné čerpadlo typu zem – vodná plocha [5]



Obr. 10 Tepelné čerpadlo typu zem – voda (studňa) [5]

B. Energetická efektívnosť tepelných čerpadiel

Energetická efektívnosť systémov, pracujúcich na báze termodynamického chladiaceho obehu (chladiace a klimatizačné zariadenia, tepelné čerpadlá), je hodnotená tzv. výkonovým číslom COP. Skratka pochádza z anglického **C**oefficient **O**f **P**erformance (koeficient výkonnosti), čo je vlastne výkonové číslo tepelného čerpadla. Je to bezrozmerné číslo, ktoré hovorí o účinnosti tepelného čerpadla. Je to pomer medzi vyrobeným teplom a spotrebovanou energiou potrebnou na činnosť čerpadla.

Tepelné čerpadlo potrebuje pri svojej prevádzke pohonnú energiu, no tepelnej energie vyrobí niekoľkonásobne viac, ako je spotreba na pohon. Je zrejmé, že čím väčšiu hodnotu COP systém dosahuje, tým vyrobí viac užitočnej tepelnej energie na jednotku dodávanej pohonnej energie

a je teda energeticky efektívnejší [6].

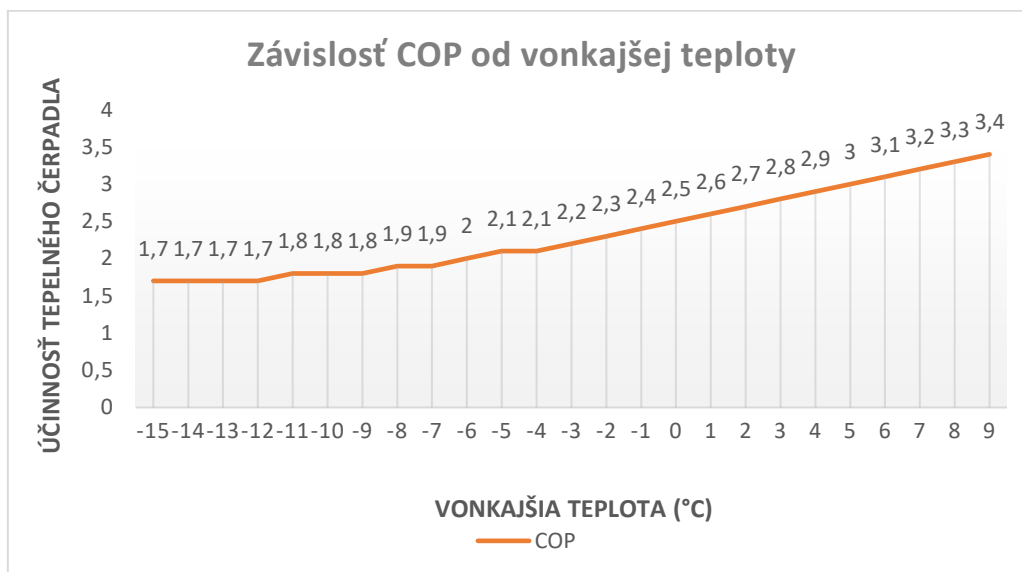
Koeficient výkonnosti vieme vyjadriť zo vzťahu: $COP = \frac{Q_t}{P_t}$

kde: Q_t – tepelný výkon zariadenia [W]

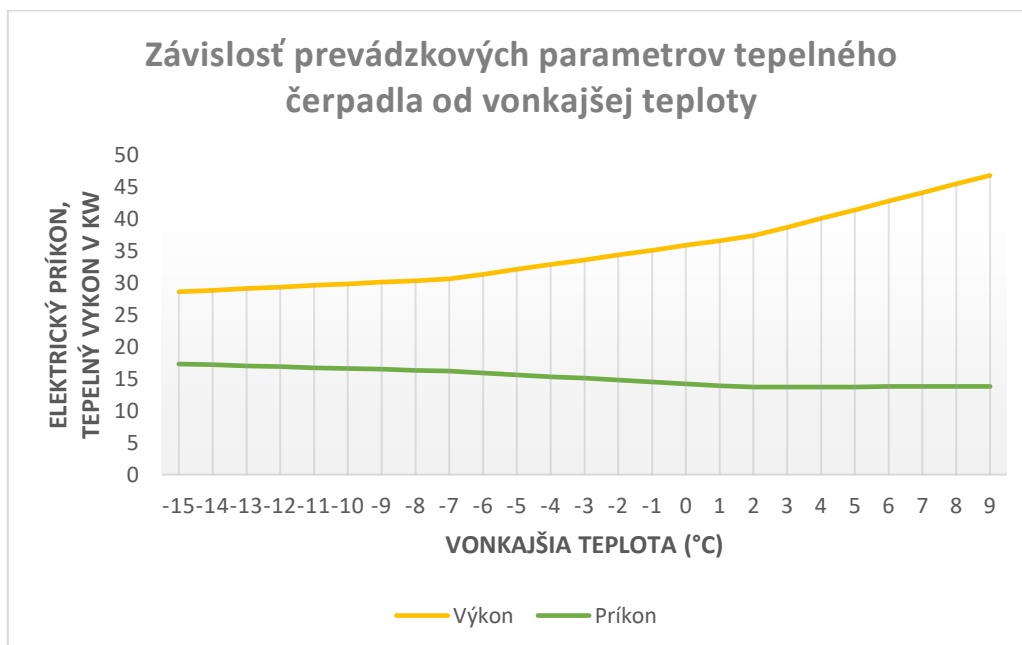
P_t – dodávaná energia (príkion) [W]

Rôzne typy tepelných čerpadiel majú rôzne vykurovacie faktory. Veľmi záleží od zdroja, z ktorého čerpadlá získavajú energiu. Napríklad tepelné čerpadlo typu zem–voda má tepelný vykurovací faktor vysoký počas celého roka. Aj pri mrazoch dosahuje hodnotu až 4,8.

Vykurovací faktor tepelného čerpadla typu vzduch–voda behom roku značne kolíše, podľa vonkajšej teploty.



Obr. 11 Závislosť COP od vonkajšej teploty



Obr. 12 Závislosť prevádzkových parametrov tepelného čerpadla od vonkajšej teploty

IV. ZÁVER

Tento príspevok sa zaoberal pozitívnym prínosom zmeny fosílného tepelného zdroja monastýru Krásny Brod alternatívnym zdrojom tepla, ktorým je tepelné čerpadlo. Vysoké ceny fosílnych palív a rýchly technologický pokrok odhaľujú aj v relatívne nových budovách rezervy v možnostiach zvýšenia úspornosti pri vykurovaní. V príspevku boli opísané klady a zápory jestvujúcich tepelných zdrojov v objekte. Tepelné čerpadlo sa môže použiť ako samostatný

zdroj vykurovania, ale len v prípade nízkoenergetických domov. V extrémnych zimách môžu pomáhať tepelným čerpadlám typu vzduch–voda aj iné zdroje tepla dosiahnuť požadovaný tepelný výkon potrebný na vykurovanie. Najčastejšie je to vstavané elektrické dokurovanie, alebo kotol na zemný plyn. Takýto systém je tzv. bivalentný. Preto je vhodné, ak sa k súčasnému plynovému kotlu doplní tepelné čerpadlo, ktoré zabezpečí temperovanie, prípadne vykurovanie pri bežnej prevádzke.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18 a tiež v rámci operačného programu Výskum a vývoj pre projekt: *Centrum výskumu účinnosti integrácie kombinovaných systémov obnoviteľných zdrojov energií*, s kódom ITMS: 26220220064, spolufinancovaný zo zdrojov Európskeho fondu regionálneho rozvoja.



Podporujeme výskumné aktivity na Slovensku/Projekt je spolufinancovaný zo zdrojov EÚ.

LITERATÚRA

- [1] Košičanová, D., Kováč, M., Knížová, K. „*Vykurovacie systémy*“. Košice: TU SvF, 2010. 87s. ISBN 80-8076-000-4.
- [2] EKOVIDAL, s.r.o., „*Kondenzačné plynové kotly*“ [online] [cit. 01-05-2020] Dostupné na: < <https://dodavatelia.123dopyt.sk/601619-ekovital-s-r-o/ponuka/370977-kondenzacne-plynove-kotly> >
- [3] Klimax heat pump, „*Tepelná čerpadla vzduch-voda*“ [online]. Dostupné na: < https://klimatherm.sk/wp-content/uploads/2017/07/HOKKAIDO_Tepelne_cerpadla_vzduch_voda.pdf >
- [4] Lepší deň, „*7 dôvodov, prečo si v roku 2019 vybrať ako zdroj tepla tepelné čerpadlá*“ [online] [cit. 08-05-2020] Dostupné na: < <https://lepsiden.sk/7-dovodov-preco-si-vybrat-tepelne-cerpadla/> >
- [5] IVT s.r.o., „*Typy tepelných čerpadiel*“ [online] [cit. 11-05-2020] Dostupné na: < <https://www.ivt.sk/sk/typy-tepelnych-cerpadiel> >
- [6] Immergas Europe s.r.o., „*Čo potrebujete vedieť pred kúpou tepelného čerpadla?*“ [online] [cit. 11-05-2020] Dostupné na: < <https://mojdom.zoznam.sk/cl/10055/1824759/Co-potrebujete-vediet-pred-kupou-tepelneho-cerpadla> >
- [7] Kancír, P., „*Analýza energetického zabezpečenia pre kláštor Krásny Brod*“, Bakalárska práca. Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2020.
- [8] Kancír, P., Medveď, D., „*Analýza energetického zabezpečenia pre kláštor Krásny Brod*“, In: Electrical Engineering and Informatics X: proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice. Košice: FEI TU, 2019 S. 81-84. ISBN 978-80-553-3342-7.

Návrh ad-hoc Sieťe pre Prenos Videotokov z Nepilotovaných Lietajúcich Prostriedkov

¹Maroš BAUMGARTNER, ²Renát HALUŠKA, ³Peter ŠULAJ,
⁴Jana HANDRIKOVÁ

Katedra elektroniky a multimediálnych telekomunikácií
Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach
Park Komenského 13, 042 00 Košice, Slovenská republika
Katedra informatiky a technických vied, Fakulta bezpečnostného inžinierstva,
Žilinská univerzita v Žiline, Slovenská univerzita

¹maros.baumgartner@student.tuke.sk, ²renat.haluska@tuke.sk, ³peter.sulaj@tuke.sk,
⁴jana.handrikova@fbi.uniza.sk

Abstrakt — V článku autori prezentujú výsledky porovnania simulácií ad-hoc MANET sietí v simulátore NS3 s použitím proaktívnych a reaktívnych smerovacích protokolov. Medzi základné parametre simulácie patrí smerovací protokol a počet uzlov v sieti. Pri simulácii sa predpokladala malá rýchlosť zmeny vzájomnej polohy jednotlivých uzlov. Simulácia bola realizovaná pre statický typ siete. Predpokladom je, že lietajúce nepilotované prostriedky zaujmú statickú polohu v priestore a monitorujú tento priestor ako skupina monitorovacích prvkov. Hlavným parametrom pre hodnotenie vhodnosti použitia konkrétneho smerovacieho protokolu bol zvolený parameter priepustnosť siete, ktorá môže byť limitujúcim faktorom pre použitie takéhoto typu siete. Pri prenose videotoku je často požadovaný prenos dát v reálnom čase. Nedostatočná kvalita videotoku môže výrazne obmedziť až znemožniť následnú analýzu videotoku.

Kľúčové slová — MANET, DSDV, DSR, AODV, OLSR, smerovacie protokoly, smerovanie, NS3

I. ÚVOD

V súčasnosti sú kľúčovými požiadavkami na bezdrôtové siete ich mobilita a schopnosť dynamickej zmeny v závislosti od vplyvov okolia. Wifi siete tieto požiadavky čiastočne spĺňajú, avšak ich mobilita a dynamické vlastnosti sú výrazne obmedzené. Preto sa čoraz častejšie využívajú tzv. MANET siete (Mobile Ad-Hoc Network). Ide o siete, ktoré sú zložené z mobilných uzlov a nie sú riadené centrálnym bodom, teda nemajú pevnú štruktúru. To im zaručuje neobmedzené možnosti mobility, dynamického rozvoja a tiež odolnosť voči prerušeniu komunikačných kanálov pri výpadku konkrétnych uzlov v sieti. Svoje využitie nachádzajú vo Wifi sieťach alebo sieťach štvrtej generácie, ale ich skutočný potenciál je využitý najmä v 5G sieťach.

Článok je rozdelený na teoretickú a praktickú časť. Teoretická časť je venovaná popisu MANET siete a popisu sieťového simulátora NS3, v ktorom boli realizované samotné simulácie. Zaoberá sa i teóriou smerovania a rozdelením smerovacích protokolov.

V praktickej časti autori uvádzajú dosiahnuté výsledky simulácií s AODV, DSR, DSDV a OLSR smerovacími protokolmi a venujú sa analýze získaných výsledkov.

II. MANET

A. MANET siete vo všeobecnosti

MANET sieť (Mobile Ad Hoc Network) je autonómny systém mobilných uzlov. Ide o bezdrôtovú sieť, ktorá pozostáva z mobilných platforiem, ktoré sa označujú ako uzly. Tie môžu byť umiestnené staticky na jednom mieste, alebo sa môžu pohybovať po ľubovoľnej trajektórii. Najčastejšie ide o zariadenia, ktoré sú umiestnené na alebo v lietadlách, lodiach, automobiloch, ľudoch, alebo ako v našom prípade to môžu byť bezpilotné prostriedky (drony). Pre tieto siete je typické, že nepotrebujú infraštruktúru alebo centralizovanú administratívu. Úlohy týchto zariadení (rozbočovače, smerovače) preberajú jednotlivé uzly v sieti. Jednotlivé uzly siete sa nesprávajú iba ako hostia, ale aj ako smerovače. Tieto uzly v MANET sieťach disponujú

bezdřtřovými prijímačmi a vysielacími, používajú antény, ktoré môžu byť smerové (point – to – point), všesmerové (broadcast) alebo ich kombináciou. V určitom časovom okamihu v závislosti od úrovne vysielacieho výkonu, kanálového rušenia (ko), od pozície uzlov a ich vzorov pokrytia prijímača a vysieláča existujú 3 druhy prepojenia a to: náhodné bezdřtřové prepojenie, viacskokový graf alebo prepojenie siete typu „Ad-hoc”. Topológia typu „Ad-hoc” sa mení v čase, keď sa uzly pohybujú v priestore alebo upravujú svoje parametre. [1][2][3]

B. Smerovanie

V prípade nadrozmerých topológií je bežnou praxou, že medzi dvoma uzlami existuje viacero možných ciest na komunikáciu, tento jav sa nazýva redundancia. V prípade potreby (napríklad problém s uzlami, zlyhanie cesty a podobne) sú pakety posielané cez niektorú zo záložných trás. Okrem toho smerovanie v MANET sieťach musí riešiť úlohu, ktorá je spojená s mobilitou uzlov. Uzly sa môžu náhodne pohybovať v priestore a čase, čím sa optimálne cesty, ktoré boli vytvorené, menia. Ideálna cesta, ktorá bola využívaná, nemusí o niekoľko sekúnd neskôr pracovať správne.

Dynamické smerovacie protokoly je možné rozdeliť do dvoch skupín:

- Protokoly zamerané na pripojenie – pred odosielaním paketov je cesta známa.
- Protokoly bez pripojenia – cesta je získaná až po odoslaní paketov.

Podľa tohto rozdelenia sa delia jednotlivé MANET smerovacie protokoly na 3 základné skupiny:

- Proaktívne (tabuľkové smerovanie)
- Reaktívne (smerovanie podľa potreby)
- Hybridné (kombinácia reaktívneho a proaktívneho smerovania). [4][5]

C. Smerovacie protokoly

Pri proaktívnom smerovaní v MANET sieťach každý uzol aktívne získava informácie o topológii a možných cestách k zvolenému cieľu. Tieto informácie sa získavajú pred odosielaním smerovacích tabuliek. Na celej komunikačnej trase jednotlivé uzly disponujú informáciami o počte skokov, ktoré sú potrebné na dosiahnutie cieľového uzla. V každom uzle je generované poradové číslo, ktoré predstavuje počet skokov, ktorý je potrebný na dosiahnutie cieľového uzla. Tento typ proaktívnych protokolov je vhodný najmä pre menšie siete, pri ktorých je potrebná neustála aktualizácia smerovacích tabuliek. Z toho vyplýva jedna z hlavných nevýhod proaktívneho smerovania: jednotlivé uzly v sebe udržiavajú informácie o trasách k uzlom, ktoré nie sú zahrnuté do vysielania v danom čase. Na druhej strane, jednou z výhod proaktívnych smerovacích protokolov je, že uzly disponujú aktuálnymi informáciami. Pri požiadavke na komunikáciu preto nevznikne žiadne oneskorenie, ktoré býva zapríčinené vypočítavaním a hľadaním ideálnej trasy. Medzi najvyužívanejšie proaktívne protokoly patrí DSDV, FSR, WRP a OLSR.

Medzi najviac využívané protokoly v mobilných ad-hoc sieťach patria protokoly reaktívneho smerovania, pretože vo veľkej miere reprezentujú ich skutočnú povahu. Na rozdiel od stálych aktualizácií smerovacích tabuliek a smerovacích informácií, ktoré sú vykonávané proaktívnymi protokolmi, reaktívne protokoly vykonávajú aktualizácie týchto informácií až v okamihu, kedy je zaznamenaná požiadavka na komunikáciu, čo má pozitívny vplyv na zaťaženie siete. Hlavne z tohto dôvodu sú tieto typy protokolov využívané v typoch sietí s veľkou mobilitou jednotlivých uzlov. Hľadanie optimálnej cesty je založené na princípe vysielania Route Request Packet (RREQ) z uzla, ktorý inicializuje komunikáciu. Tieto pakety sú broadcastovo vysielané k zdrojovým uzlom, prijímajú ich všetky mobilné uzly, ktoré sa nachádzajú v dosahu, a tie tento paket vysielajú ďalej, až kým nedosiahne požadovaný cieľový uzol. Tento cieľový uzol po obdržaní RREQ paketu vysielá späť Route Reply Packet (RREP), ktorý potvrdí toto spojenie. Medzi tento typ protokolov patria napríklad DSR, AODV a ADV protokoly.

Hybridné smerovacie protokoly sú kombináciou reaktívnych a proaktívnych protokolov, ktoré sú spomenuté vyššie v rámci tejto kapitoly. Kombináciou týchto protokolov je možné výrazne znížiť režijné náklady a latenciu, ktorá vzniká pri objavovaní trás. Medzi hybridné protokoly patria napríklad ZRP – Zone Routing Protocol a TORA – Temporally Ordered Routing Algorithm. [4][5][6][7]

III. SIEŤOVÝ SIMULÁTOR NS3

Sieťový simulátor NS3 je rovnako ako jeho staršia veria NS2 riadený diskretnými udalosťami. NS3 má však novú a prepracovanú vnútornú štruktúru, čo zamedzuje spätnej kompatibilitě s NS2. Napriek tomu, že oba simulátory NS3 aj NS2 sú napísané v programovacom jazyku C++,

simulátor NS3 nepodporuje aplikačné rozhranie svojho predchodcu. Je určený najmä na výskumné a vzdelávacie účely v oblasti sieťových technológií už od roku 2006, kedy bol prvýkrát uvedený ako open source projekt najmä pre výskumných pracovníkov, aby mohli prispievať a zdieľať svoj softvér. Bol vyvinutý tak, aby poskytoval otvorenú a rozšíriteľnú platformu sieťovej simulácie pre výskum a vzdelávanie v oblasti sietí.

NS3 poskytuje modely pre simuláciu fungovania a testovanie výkonnosti dátových sietí, a tak používateľom poskytuje simulačný mechanizmus na vykonávanie experimentálnych simulácií. Jedným z hlavných dôvodov využívania simulátora NS3 je jednoduchšie vytváranie projektov a štúdií, v porovnaní so zložitou ich vytváraním v profesionálnych systémoch. Preto je vhodný najmä na reprodukovanie prostredí s dynamickými parametrami a na pochopenie fungovania sietí. Najčastejšie je však využívaný na simulovanie mobilných ad-hoc sietí najmä kvôli širokým možnostiam konfigurácie.

NS3 je navrhnutý ako súbor knižníc, ktoré je možné kombinovať s inými externými softvérovými knižnicami. Niektoré simulačné platformy poskytujú používateľom jednotné integrované grafické používateľské rozhranie, v ktorom sa vykonávajú všetky úlohy. Simulátor NS3 je v tomto ohľade modulárnejší.

Na to, aby používateľ vedel využiť plný potenciál NS3 simulátora, je potrebné implementovať a využívať externé nástroje na animáciu a nástroje na analýzu, spracovanie a vizualizáciu dosiahnutých výsledkov. Medzi takéto nástroje patria napríklad NetAnim, GNUPlot a podobne, ktoré boli využívané v navrhnutých simuláciách.

NS3 rovnako ako NS2 neponúka grafické užívateľské rozhranie, teda užívatelia musia ovládať a konfigurovať simulácie z príkazového riadka a priamym zásahom do zdrojového kódu, ktorý je napísaný v C++ a / alebo objektovo orientovanom jazyku Python.

Z pohľadu kompatibility je NS3 možné využívať v systémoch Linux alebo MacOS, hoci existuje podpora pre systémy BSD a tiež pre systémy Windows, ktoré dokážu zostaviť linuxový kód, pomocou softvérov Windows Subsystem alebo Cygwin. [8]

IV. DOSIAHNUTÉ VÝSLEDKY

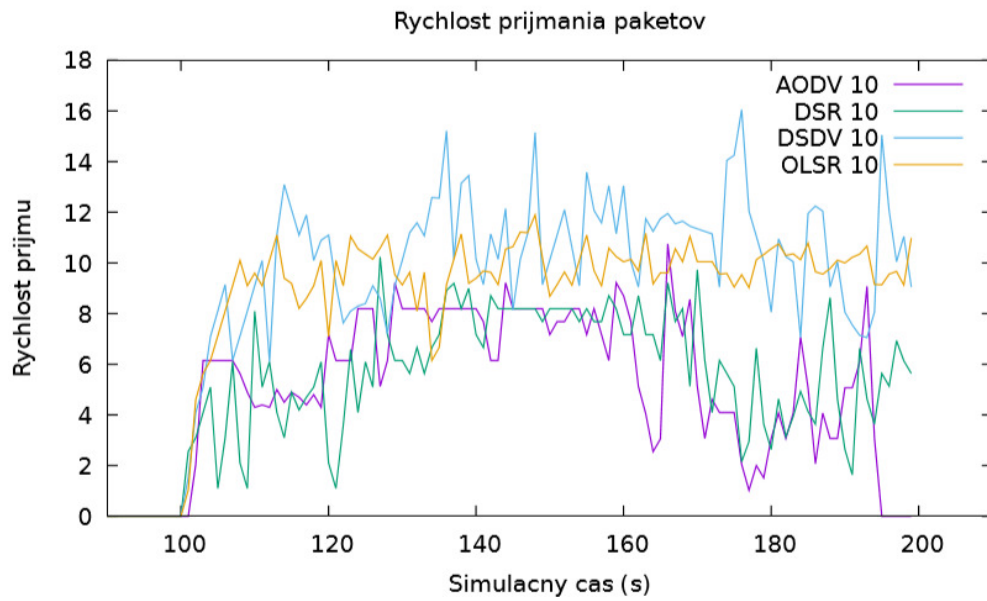
Na základe simulovaných sietí a parametrov smerovacích protokolov je možné konštatovať, že v každom scenári každého smerovacieho protokolu vplývala zmena počtu uzlov najmä na zväčšovanie spotreby energie a oneskorenia prijímaných paketov.

V rámci simulácií boli zahrnuté smerovacie protokoly zo skupiny reaktívnych a proaktívnych protokolov. Táto skutočnosť nám umožnila porovnať tieto dve skupiny protokolov pri rôznych počtoch uzlov v sieti. Vďaka tomu sme mohli pozorovať, že v menej obsadených sieťach s 10 a 20 uzlami boli vhodnejšie, stabilnejšie a lepšie pracovali proaktívne smerovacie protokoly, ktoré vykazovali lepšiu spotrebu energie a menšiu hodnotu oneskorenia.

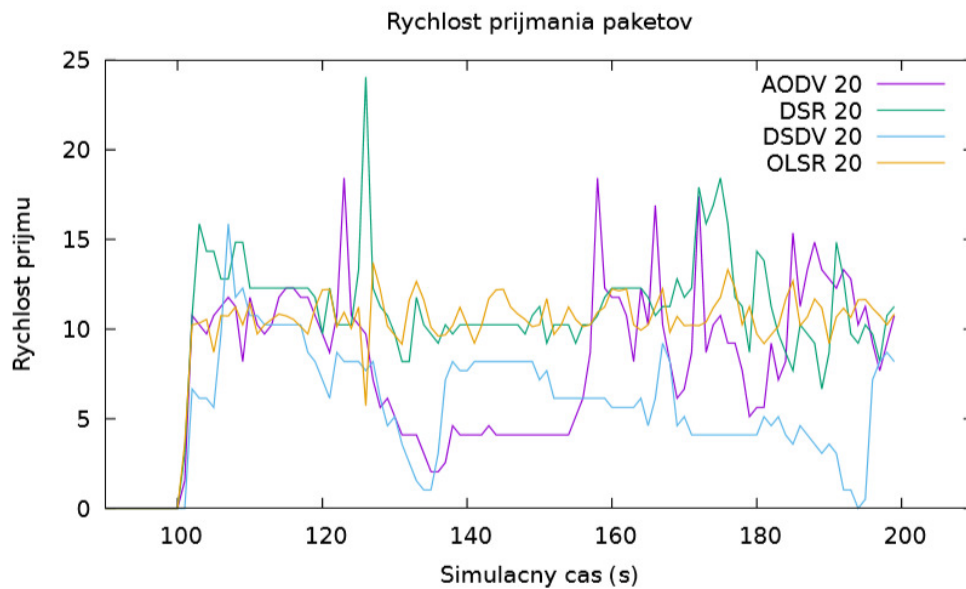
Avšak pri každom navýšení počtu uzlov sa parametre správania proaktívnych protokolov začali zhoršovať. Ak porovnáme správanie sa všetkých protokolov v sieťach s 10, 20 a 50 uzlami, reaktívne protokoly majú tendenciu aj pri minimálnom navýšení počtu uzlov zlepšovať svoje parametre. Z toho vyplýva, že pre menšie siete sú výhodnejšie proaktívne protokoly a pre rozvinutejšie siete sú vhodnejšie reaktívne protokoly, ktoré nevykazujú takú stabilitu v malých sieťach ako protokoly DSDV a OLSR.

Vzhľadom na hardvérové možnosti zariadenia, na ktorom boli tieto scenáre simulované, sa ich parametre odvíjali od tejto skutočnosti. Vyššie spomínaný počet mobilných uzlov bol simulovaný v priestore o rozmeroch 1500x300 metrov s dĺžkou každej simulácie 60 sekúnd a prenosovou rýchlosťou dátových paketov 2 Mbps.

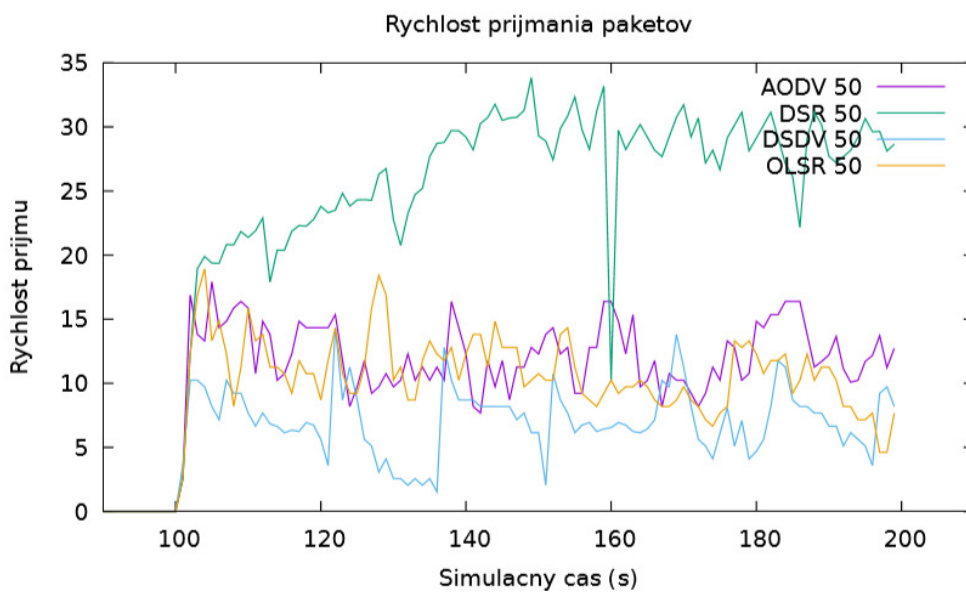
Na nasledujúcich obrázkoch sú graficky znázornené porovnania a rozdiely v počte prijatých paketov, rýchlosti prijímania a priepustnosti siete jednotlivito pre siete s 10, 20 a 50 uzlami pre každý smerovací protokol.



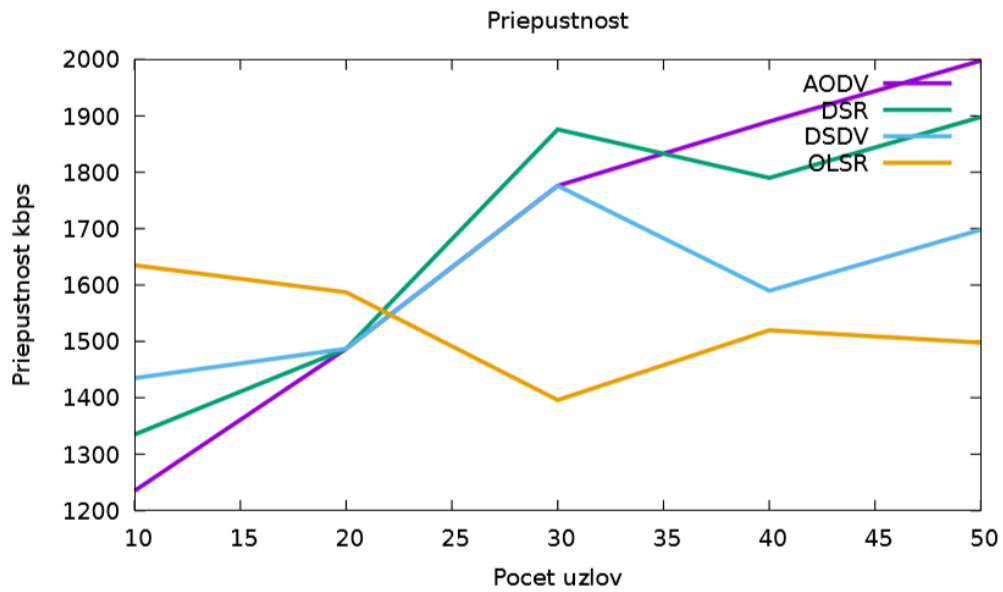
Obr. 1 Rychlost' prijmania paketov smerovacich protokolov s 10 uzlami



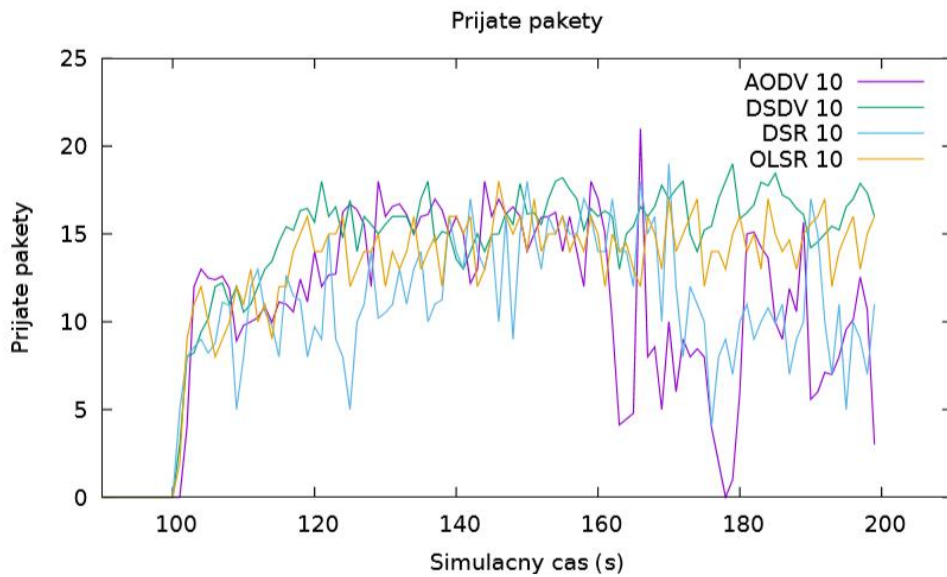
Obr. 2 Rychlost' prijmania paketov smerovacich protokolov s 20 uzlami



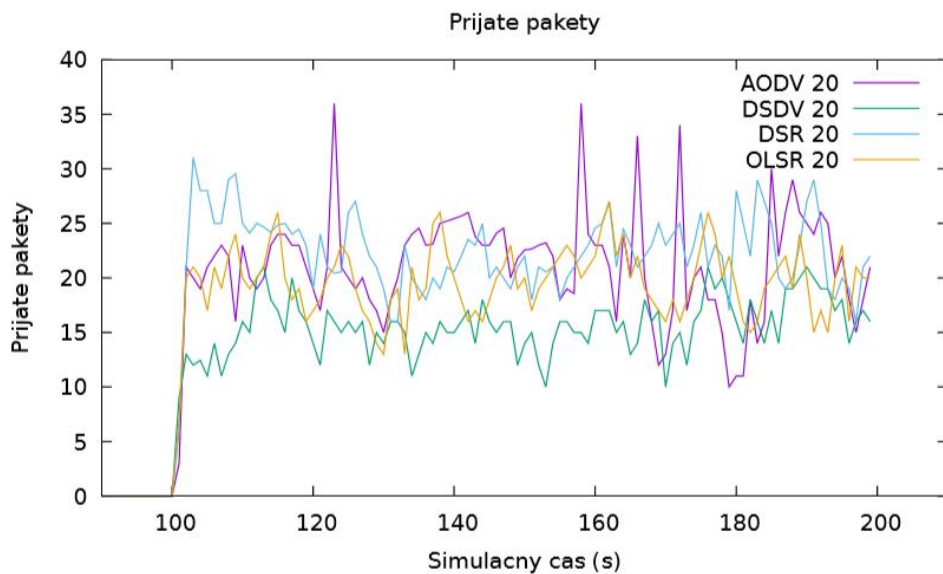
Obr. 3 Rychlost' prijmania paketov smerovacich protokolov s 50 uzlami



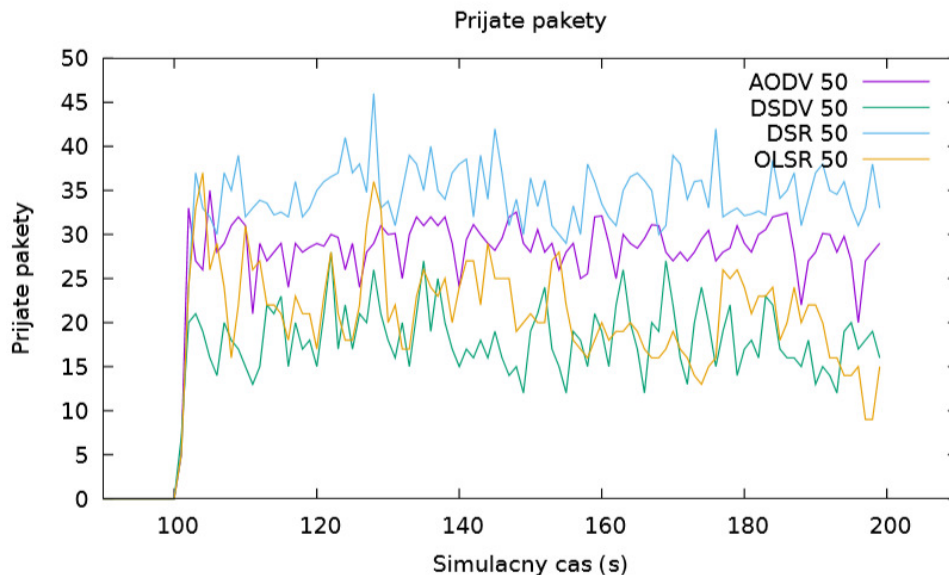
Obr. 4 Porovnanie priepustnosti siete so simulovanými smerovacími protokolmi



Obr. 5 Porovnanie prijatých paketov v sieti s 10 uzlami



Obr. 6 Porovnanie prijatých paketov v sieti s 20 uzlami



Obr. 7 Porovnanie prijatých paketov v sieti s 50 uzlami

Na základe vykonaných simulácií a dostupnej teórie nie sú najlepším riešením pre MANET siete iba reaktívne alebo proaktívne protokoly, ale ich kombinácia v hybridných protokoloch. Nakoľko sú MANET siete dynamické, sú pre nich hybridné smerovacie protokoly vhodnejšie. Keďže reaktívne protokoly pracujú s väčšou efektívnosťou, ak nie je požiadavka na komunikáciu častá, a proaktívne protokoly pracujú lepšie, ak je potreba zahájiť komunikáciu častejšia, je vhodné využívať ich kombináciu s cieľom dosiahnuť čo najväčšiu efektívnosť a úsporu energie.

V. ZÁVER

Na základe definovaných parametrov pre 3 rôzne počty uzlov v MANET sieti (10, 20, 50 uzlov) a použitia štyroch smerovacích protokolov v sieťach s týmito počtami uzlov, sme vytvorili 12 simulačných scenárov. Pre následnú simuláciu prevádzky siete sme použili simulátor NS3, ktorý sa javí ako vhodný a efektívny nástroj pre simuláciu takéhoto typu sietí.

Na základe predpokladu statickej konfigurácie siete, dochádza k relatívne malým požiadavkám na zmenu počtu prvkov siete – uzlov, prípadne zmenu zdrojového a cieľového uzla pre prenos signálu, javia sa ako najvhodnejší AODV protokol, ktorý má najväčšiu priepustnosť siete.

POĎAKOVANIE

Túto prácu podporila Agentúra pre kultúru a vzdelávanie (KEGA) Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky v rámci projektu č. 062TUKE-4/2017, projektu č. 023TUKE-4/2017 a Slovenská agentúra pre výskum a vývoj na základe zmluvy č. „APVV-17-0208 - Odolné mobilné siete na doručovanie obsahu“.

LITERATÚRA

- [1] CORSON S. MACKER J. University of Maryland. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, 1999.
- [2] ILYAS, M. Ad hoc wireless networks. Vyd. 1. New York: CRC Press, 2003, ISBN 0-8493-1332-5.
- [3] ROY, Radhika Ranjan. Handbook of mobile ad hoc networks for mobility models. New York: Springer, c2011, s. 3-5. ISBN 978-1-4419-6048-1.
- [4] LOMNICKÝ M. Vysoké učení technické v Brně. Absolventská práca. Směrování a směrovací protokoly: Technologie sítí WAN (CCNA4), 2007 8 s. Dostupné na internete : <http://netacad.fit.vutbr.cz/texty/ccna-moduly-ccna2-6.pdf>.
- [5] VADKERTI G. Vysoké učení technické v Brně. Bakalárska práca. Vytvoření experimentální sítě v simulátoru NS-3, 2013.
- [6] HOŠEK P. Vysoké učení technické v Brně. Diplomová práca. Směrovací protokol OLSR pro MANET sítě v simulačním prostředí OPNET Modeler, 2011.
- [7] REXA D. Vysoké učení technické v Brně. Bakalárska práca. Uspořádání ad hoc sítí a různé typy útoků, 2015.
- [8] Network Simulator 3 documentation. Introduction. About NS3. Dostupné na internete: <https://www.nsnam.org/docs/tutorial/html/introduction.html>.

Štandardný videokodek H.265/HEVC

Ján MIHALÍK

Department of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

jan.mihalik@tuke.sk

Abstract — V súčasnosti najnovším štandardným videokodekom je H.265/HEVC (High Efficiency Video Coding) alebo MPEG-4 H, ktorý nasleduje po štandarde H.264 (AVC) a je tiež spoločným produktom obidvoch medzinárodných štandardizačných orgnizácií ITU-T a ISO/IEC. Tento najnovší štandard H.265/HEVC pokrýva celý rozsah aplikácií jeho predchodcu H.264 (AVC). Tieto aplikácie, ale umožňuje so zvýšenou kvalitou a rozlíšením, pričom rozširuje aj použitie paralelných architektúr spracovania videosignálov.

Kľúčové slová — videokodek, algoritmy, videosekvencia, štandardy, H.264 (AVC), H.265/HEVC

I. INTRODUCTION

Štandardné videokodeky boli ustanovené najmä na základe ich vývoja na poli organizácií ITU-T a ISO/IEC. Najprv z organizácie ITU-T sa získali štandardy H.261 a H.263 a z ISO/IEC to boli štandardy MPEG-1 ako aj MPEG-4 Visual. Potom tieto organizácie spoločne vyprodukovali štandardy H.262/MPEG-2 Video a H.264/MPEG-4 Advanced Video Coding (AVC). Práve posledné dva video štandardy zo spoločnej produkcie boli aplikované v širokom rozsahu výroby rôznych druhov video zariadení. Tento kontinuálny vývoj video štandardov postupne zvyšoval kompresiu údajov, umožňoval vysoké rozlíšenie (HD-high definition) a vylepšoval ich ďalšie charakteristiky, ako napr. robustnosť dátových strát. Pritom zohľadňoval úroveň výpočtových prostriedkov (digitálnych technológií) v čase vývoja každého z týchto štandardov pre ich praktické použitie v jednotlivých video zariadeniach.

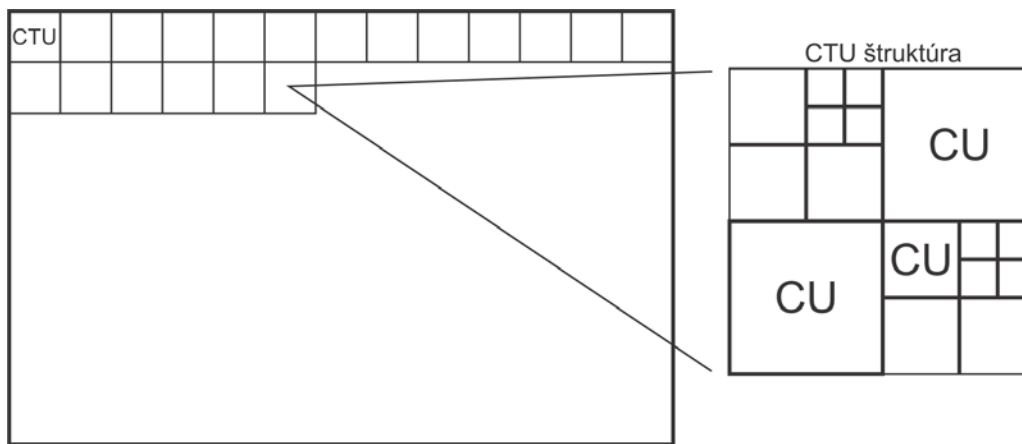
Predsa však s narastajúcou rozmanitosťou video služieb a tiež popularitou HD (2k x 1k) videa s prechodom až na formáty s ultra HD (UHD), t.j. s ešte vyšším rozlíšením (napr. 4k x 2k alebo 8k x 4k) vznikla potreba ďalej zvýšiť účinnosť kódovania štandardu H.264 (AVC). Táto potreba je ešte výraznejšia, keď je žiaduce s takto vysokým rozlíšením kódovať stereo alebo multi snímkové videosignály napr. pre digitálnu 3RTV (3-rozmerná TV). Toto zvyšovanie účinnosti kódovania vyplýva aj z mobilných video aplikácií s narastajúcimi požiadavkami na kvalitu a rozlíšenie videosignálov so súčasným zdokonaľovaním technológií mobilných sietí.

V súčasnosti najnovším štandardným videokodekom je H.265/HEVC (High Efficiency Video Coding) [1], ktorý nasleduje po štandarde H.264 (AVC) Tento najnovší štandard H.265/HEVC [2] pokrýva celý rozsah aplikácií jeho predchodcu H.264 (AVC). Tieto aplikácie, ale umožňuje so zvýšenou kvalitou a rozlíšením, pričom rozširuje aj použitie paralelných architektúr spracovania videosignálov. Svojou zvýšenou účinnosťou kódovania oproti svojmu predchodcovi umožňuje zníženie bitovej rýchlosti až o 50 % pri zachovaní rovnakej kvality kódovaného videosignálu.

Realizácia štandardného videokodeka H.265/HEVC [3], tak ako jeho všetkých predchodcov, je založená na klasickom blokovom hybridnom kódovaní videosignálov [4]. Potom jeho základné algoritmy sú: medzismímková predikcia využívajúca časové štatistické závislosti, vnútrosmímková predikcia pre odstraňovanie priestorových štatistických závislostí vstupného videosignálu a tiež ortogonálna transformácia pre ďalšie potláčanie priestorových štatistických závislostí, ale z predikčného chybového signálu. Nakoniec sú to aj algoritmy kvantovania a entropického kódovania symbolov. Neexistuje žiadny samostatný algoritmus v H.265/HEVC, ktorý by prioritne významne zvýšil jeho účinnosť kódovania oproti jeho predchodcom, ale je to multiplikatívny výsledok menších účinkov od mnoho jeho algoritmov. Ďalej uvedieme len najvýznamnejšie vylepšenia (modifikácie) algoritmov štandardného videokodeka H.265/HEVC [5] oproti obdobným algoritmom jeho posledného predchodcu H.264 (AVC) [6], [7].

II. ORGANIZÁCIA, REPREZENTÁCIA A DELENIE VSTUPNEJ VIDEOSEKVENČIE

Rovnako ako vo všetkých predchádzajúcich štandardných videokodekoch vstupná videosekvencia je reprezentovaná vo farebnom priestore s jasovou zložkou Y a farebnými rozdielovými zložkami $U=B-Y$ a $V=R-Y$. Pritom si zachováva aj typický videoformát 4:2:0, kedy farebné zložky majú dva krát menej riadkov aj stĺpcov ako jasová zložka. Pritom jednotlivé obrazové prvky (op) týchto zložiek sú digitálne vyjadrené s 8 bitmi. Rozdiel je iba v rozdelení jej snímok (polsnímkov) na menšie rektangulárne oblasti (jednotky) s možnosťou ich nezávislého kódovania. V predchádzajúcich štandardných videokodekoch takýmito jednotkami boli makrobloky s veľkosťou jasovej zložky 16×16 op a dvoch korešpondujúcich farebných zložiek 8×8 op. V štandardnom videokodeku H.265/HEVC veľkosť týchto jednotiek môže byť aj dva alebo tri krát väčšia. Spolu možno vytvoriť jednotky s veľkosťou jasovej zložky 16×16 , 32×32 a 64×64 op a stále polovičnými veľkosťami im prislúchajúcich farebných zložiek. Pritom tieto jednotky budú mať svoju stromovú štruktúru odvodenú z charakteristík oblastí snímky v ktorých sa nachádzajú. Potom táto je popísaná definovaným kvadrantovým stromom [8] ako to vidno z Obr.1.

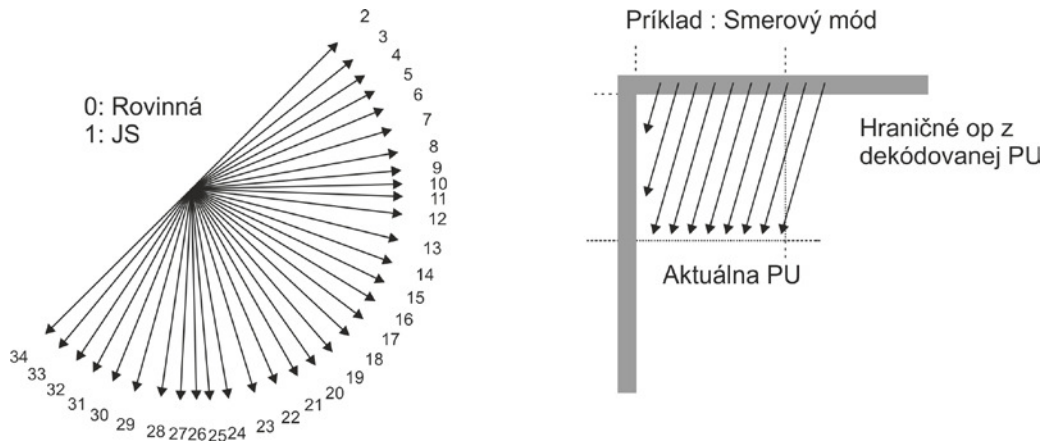


Obr. 1 Delenie snímky na jednotky so stromovou štruktúrou.

Po rozdelení snímky na jednotky CTU (Coding Tree Units) sa tieto vo všeobecnosti pomocou prislúchajúcich kvadrantových stromov môžu ďalej deliť na menšie CU (Coding Units) a to súbežne pre ich jasové aj farebné zložky. Zoskupovaním určitého počtu po sebe nasledujúcich CTU možno vytvárať skupiny, čo umožňuje ľahkú resynchronizáciu v prípade straty dát, ale aj kontrolu prípustného maximálneho počtu bitov pre paketové videokomunikácie [9]. Počty CTU v jednotlivých skupinách sa značne menia a závisia od aktivity vizuálnej scény v mieste danej skupiny. Okrem toho možno vytvárať samostatne obsahovo nezávisle dekódovateľné rektangulárne štruktúry týchto CTU za účelom ich paralelného spracovania v procese ich kódovania a dekódovania.

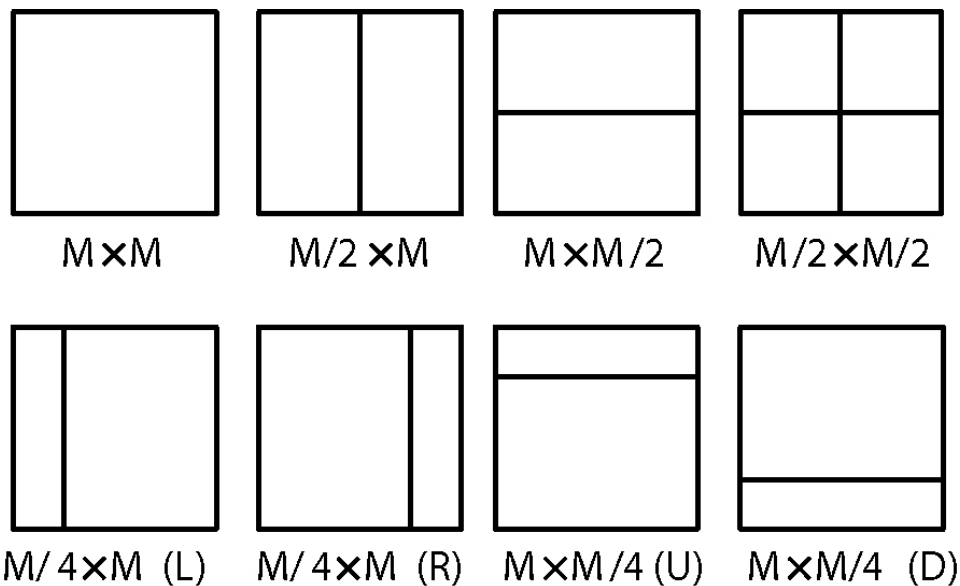
III. VNÚTROSŇÍMKOVÁ A MEDZISŇÍMKOVÁ PREDIKCIA

V procese kódovania jednotlivých CTU možno použiť rôzne transformačné a predikčné módy [10] a to vo všeobecnosti aj s rôznymi stromovými štruktúrami. Pritom veľkosť predikčných blokov sa môže meniť od 4×4 do 64×64 op, ale transformačných blokov len od 4×4 do 32×32 op. Za účelom dosiahnutia vysokej kompresie dát pri zachovaní dostatočnej presnosti dekódovaných snímok bude efektívnejšie kódovať hladké oblasti snímok s veľkými blokmi a oblasti s detailmi s malými blokmi. Predikčný mód CU je signalizovaný s vnútrosnímkovou alebo medzisosnímkovou predikciou. Štandardný videokodek H.265/HEVC zachováva jednosmernú aj rovinnú vnútrosnímkovú predikciu, ale rozširuje počet možných smerov tejto predikcie až na 33 ako to vidno z Obr.2. Pre medzisosnímkovú predikciu umožňuje CU deliť aj na asymetrické predikčné jednotky (PU) s nerovnakou šírkou alebo výškou ako to vidno na Obr.3 a týmto presnejšie estimovať jej pohyb.



Obr. 2 Módy vnútrostránkovej predikcie a príklad smerového módu 29.

Podobne ako v štandarde H.264 (AVC) deblokujúci filter v spätnej väzbe H.265/HEVC umožňuje zlepšiť medzisnímkovú predikciu a týmto aj potlačiť blokový efekt a zároveň dosiahnuť vyššiu kompresiu dát. Zatiaľ čo tento filter v H.264 (AVC) sa aplikuje na bloky 4x4, tak v H.265/HEVC na bloky 8x8 a to najmä na op umiestnené na ich hranách (okrajoch) tak pre jasové ako aj farebné bloky. Významným rozdielom je zapojenie SAO (Sample Adaptive Offset) filtra na výstup deblokujúceho filtra v H.265/HEVC, ktorý sa aplikuje na všetky op v bloku napr. v závislosti od gradientných podmienok. Jeho výsledkom je lepšia rekonštrukcia amplitúdy op pomocou vyhľadávacej tabuľky popísanej niekoľkými dodatočnými parametrami určenými na základe analýzy histogramu na vysielacej strane.



Obr. 3 Spôsoby delenia CU na predikčné jednotky s rovnakou, ale aj nerovnakou šírkou alebo výškou.

IV. TRANSFORMÁCIA, KVANTOVANIE, A ENTROPICKÉ KÓDOVANIE

Tak ako v predchádzajúcich štandardných videokodekoch, tak aj v H.265/HEVC sa predikčné chybové snímky kódujú pomocou vnútrostránkoveho transformačného kódovania [11]. Vzhľadom na stromovú štruktúru CTU s rôznou veľkosťou blokov tento štandardný videokodek používa dvojrozmernú transformáciu s možnými veľkosťami blokov 4x4, 8x8, 16x16 a 32x32 op. Jadro tejto transformácie je tvorené celočíselnými báзовými funkciami odvodené ako celočíselná aproximácia obdobných báзовých funkcií DCT pre rovnaké veľkosti blokov s blízkym zachovaním vlastnosti ich ortogonalitu. Pre vnútrostránkove transformačné kódovanie chybových jasových snímkov tento štandardný videokodek umožňuje po vnútrostránkovej predikcii použiť aj jadro 4x4 op odvodené analogickým spôsobom z báзовých sínusových funkcií.

V tomto štandardnom videokodeku sa spektrálne koeficienty kvantujú pomocou lineárneho kvantovania s indexovaním [12], pričom počet kvantizačných parametrov zostáva rovnaký ako v H.264 (AVC). Tieto parametre sú ale usporiadané do rôznych matic v závislosti od veľkosti blokov. Pritom prípustné rozmery týchto matic sú 4x4 a 8x8 op vzhľadom aj na pamäťové požiadavky ich záznamu. Pre väčšie rozmery blokov, t.j. 16x16 a 32x32 op umožňuje tento štandard získať im zodpovedajúce matice kvantizačných parametrov rovnakého rozmeru zo zadanej matice 8x8 spôsobom rozšírenia jej hodnôt na oblasť 2x2 alebo 4x4 op.

Entropické kódovanie spektrálnych koeficientov v H.265/HEVC sa vykonáva iba pomocou stavového adaptívneho binárneho aritmetického kódovania [13] podobne ako v H.264 (AVC). Jeho modifikácia, ale umožňuje paralelné spracovanie a pri menšom počte možných stavov umožňuje dosiahnuť väčšiu kompresiu dát pri zmenšených požiadavkách na ich pamäť. Pritom entropické kódovanie parametrov videosekvencie sa vykonáva s nemennou alebo premenlivou dĺžkou kódových slov [14].

V. PROFILY, ÚROVNE A MOŽNÉ APLIKÁCIE

Význam profilov a úrovní zostáva rovnaký ako v predchádzajúcich štandardných videokodekoch. Profilom sa definuje súbor nástrojov, resp. algoritmov, ktoré môžu byť použité pre generovanie vyhovujúcich bitových tokov. Naopak úrovne predstavujú obmedzenia na kľúčové parametre ako je bitová rýchlosť, raster (veľkosť) obrazu, kompresný pomer, kapacita vyrovnávacej pamäte, atď. Minimalizácia počtu profilov poskytuje maximálnu interoperabilitu medzi zariadeniami pre rôzne služby ako je TV vysielanie, Internet video streamovanie a tiež mobilné video služby, čo umožňuje ich spoločné použitie pre všetky z nich. Pre prvú verziu H.265/HEVC sú definované tri profily: hlavný, hlavný 10 a hlavný statického obrazu. Hlavný profil podporuje CIF videoformat 4:2:0 s dĺžkou 8 bitov jeho digitálnych op najmä pre mobilné videotelefony a HDTV terminály. Potom hlavný 10 profil rozširuje dĺžku kódových slov digitálnych op na 10 bitov a je obzvlášť vhodný pre UHD TV terminály, umožňujúci vysokú kompresiu videosekvencií so širokým dynamickým rozsahom bez pásmových artefaktov, ktoré sa môžu niekedy objavovať pre dĺžku 8 bitov. Posledný profil je zavedený pre statické obrazy a je to vlastne subset hlavného profilu pre videosekvencie. V tejto verzii tohto štandardu je definovaných 13 úrovní s obmedzením videoformátov od 176x144 op (Quarter CIF) do 7680x4320 op (8kx4k) a bitových rýchlostí od 64 kbit/s do 240 Mbit/s.

V súčasnosti rozlišujeme už päť verzií [15] štandardu H.265/HEVC, ktoré boli postupne vytvorené za účelom rozširovania jeho funkcií. V súvislosti s tým pripúšťa videoformáty so zväčšenou dĺžkou kódových slov digitálnych op ako aj rozlíšením (rastrom) farebných zložiek. Tiež umožňuje škálovanie videosekvencií v prostredí heterogénnych sietí a pri používaní videoterminálov s rôznym výpočtovým výkonom. Boli doplnené aj algoritmy kódovania stereo a multi snímkových videosignálov včítane hĺbkových máp priestorových súradníc op pre digitálnu 3RTV. Zaviedli sa aj profily algoritmov obsahového kódovania vizuálnych scén a ďalšieho zväčšovania rozsahu škálovania ako aj vysokej bitovej priepustnosti.

LITERATÚRA

- [1] Ericsson Review, April 2013, Next Generation Video Compression.
- [2] ITU-T Recommendation H.265 v1. High Efficiency Video Coding. Geneva, 2013.
- [3] G. J. Sullivan – J. R. Ohm – W. J. Han – T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. On Circuits and Systems for Video Technology*, vol. 22, no. 12, 2012, pp. 1649- 1668.
- [4] J. Mihalík, "Adaptívne hybridné kódovanie obrazu," *Elektrotech. časopis*, roč. 44, č. 3, 1993, s. 85-89.
- [5] I. U. Khan – S. Kumar – S. H. Saeed – N. R. Kidwai, "Next Generation Video Coding Systems (H.265/HEVC) and Its Comparison from H.264/AVC Video Codec," *Int. Journal of Engineering Technology, Management and Applied Sciences*, vol.3, issue 4, 2015, pp.42-45.
- [6] J. Mihalík, "Štandardný videokodek H.264/AVC," *Slaboproudý obzor*, roč. 67, č. 4, 2011, s. 4-10.
- [7] T. Wiegand - G. J. Sullivan - G. Bjontegaard – A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, 2003, pp.1-19.
- [8] K. R. Rao – J. J. Hwang – D. N. Kim, *High Efficiency Video Coding and Other Emerging Standards*. River Publishers, 2017.
- [9] J. Mihalík, "Videokomunikácie na Internete," *Slaboproudý obzor*, roč. 72, č. 4, 2016, s.9-13.
- [10] J. Mihalík, *Kódovanie obrazu vo videokomunikáciach*. Mercury-Smekal, Košice, 2001.
- [11] J. Mihalík, "Adaptívne transformačné kódovanie obrazu," *Slaboproudý obzor*, roč. 52, č. 11-12, 1991, s. 253-257.
- [12] J. Mihalík - R. Štefanišin, "Algoritmy kvantovania štandardných videokodekov," *Acta Electrotechnica et Informatica*, vol. 4, no. 1, 2004, s. 47-54.
- [13] I. Gladišová – J. Mihalík – J. Zavacký, "Bezstratová kompresia obrazu pomocou stavového aritmetického kódovania jeho bitových rovín," *Slaboproudý obzor*, roč. 62, č. 1-2, 2006, s. 18-21.
- [14] J. Mihalík - R. Štefanišin, "Entropické kódovanie v štandardných videokodekoch," *Acta Electrotechnica et informatica*, vol. 5, no. 2, 2005, s. 43-50.
- [15] ITU-T Recommendation H.265 v5. High Efficiency Video Coding. Geneva, 2018.

Výpočet energetickej bilancie horského hotela v Poráčskej doline s využitím obnoviteľných zdrojov energie

¹Miroslav TOMČO, ²Dušan MEDVEĎ

¹ Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky Technickej univerzity v Košiciach, Slovenská republika

¹miroslav.tomco@tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — Tento článok sa zaoberá návrhom modelu spolupráce fotovoltaickej elektrárne, solárnych kolektorov a elektrokotla. Úlohou je odstrániť peletkový kotol, ktorý pracuje v letných mesiacoch v neekonomickom a nehospodárnom režime a navrhnúť alternatívny zdroj tak, aby bola pokrytá spotreba teplej úžitkovej vody a spotreba vody pre vykurovanie. Navrhnutý zdroj by mal využívať zelenú energiu, teda tento zdroj má patriť k obnoviteľným zdrojom energie.

Kľúčové slová — elektrokotol, fotovoltaická elektráreň, solárne kolektory, teplá úžitková voda

I. ÚVOD

V súčasnosti sa vo svete spotrebáva obrovské množstvo energie. Tento vývoj odštartoval zhruba od počiatku 20. storočia a úzko súvisí so spotrebným spôsobom života v priemysle rozvinutých krajinách. Zásoby tradičných zdrojov energie sú obmedzené a odhaduje sa, že budú stačiť prinajmenšom na ďalších 100 rokov za predpokladu, že spotreba bude pokračovať v súčasnom trende [1].

Viac ako 100 rokov vykazuje spotreba energie vo svete viac-menej stabilný rastový trend, priemerne 3 % ročne. Odhaduje sa, že tento trend bude pokračovať. Prvým dôvodom je úsilie rozvojových krajín, ako je napríklad Čína, dosiahnuť rovnakú úroveň ako v najrozvinutejších krajinách. Druhým dôvodom je pokračujúci rast svetovej populácie. Dnes, viac ako dve miliardy ľudí musí žiť bez elektriny a dá sa predpokladať, že väčšina z nich bude skôr alebo neskôr chcieť využívať prednosti elektriny [1].

Nepriaznivá realita našej súčasnosti súvisiaca s dnes už zjavnými klimatickými zmenami, nás vyzýva na vytváranie predpokladov pre trvalo udržateľné riešenia, ktoré sú na celom svete impulzom k hľadaniu a nachádzaniu možností využívania alternatívnych zdrojov energií, obnoviteľných zdrojov energie (OZE). Jednou z možností hromadenia nevyhnutnej potreby energie, ktorá je výsledkom dlhodobého výskumu s dnes už reálnou praxou, je aplikácia technológií, ktoré využívajú energiu Slnka – pre nás, nevyčerpatel'ny zdroj tepla a svetla. Množstvo dopadajúcej energie získanej zo Slnka je približne 200-násobne väčšie ako je súčasná spotreba energetických zdrojov. Z energetického hľadiska ide teda o mimoriadne zaujímavú možnosť získavania energie. Solárna energia sa v posledných rokoch začína využívať vo väčšej miere a to hlavne aplikovaním fotovoltaických systémov [2].

Fotovoltaika je technológia, ktorá dnes zažíva vo svete neobvyklý rozmach a do budúcnosti patrí k tým najperspektívnejším obnoviteľným zdrojom energie. Fotovoltaické systémy využívajú denné svetlo k priamej výrobe elektriny. Najväčšia prednosť fotovoltaiky je jej univerzálne použitie. Fotovoltaické systémy je možné používať v širokom rade výkonov, od zlomkov wattov až po megawattové elektrárne, prakticky kdekoľvek na zemskom povrchu, i vo vesmíre. Podobne ako moderné telekomunikačné systémy dovoľujú spojenie aj s najzapadnutejším koncom Zeme, bez použitia drôtov, tak aj fotovoltaika dáva podobnú možnosť jej využitia kdekoľvek.

II. DEFINOVANIE PROBLÉMU

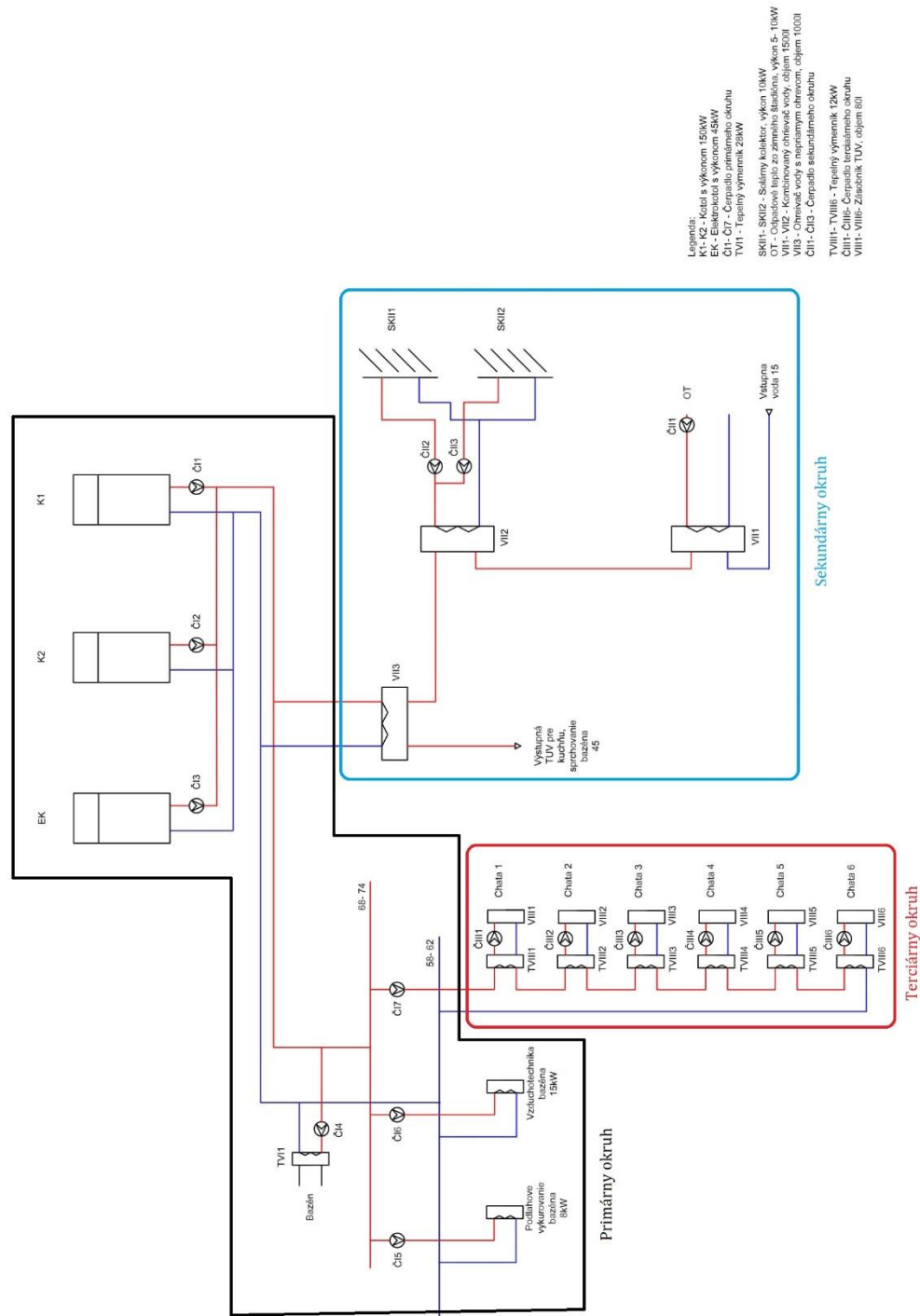
Peletkový kotol s menovitým výkonom 150 kW pracuje počas letných mesiacov v neekonomickom režime, výkon kotla sa vtedy pohybuje okolo 40 kW. Spúšťa sa v krátkych časových intervaloch, čo má nepriaznivý vplyv na prevádzku kotla. Vytvára sa struska, výmenník sa zanáša sadzami. Cieľom je odstavenie peletkového kotla v letnej prevádzke a jeho nahradenie

vhodným paralelným zdrojom k peletkovému kotlu do vykurovacieho okruhu. Výsledkom bude návrh výkonu fotovoltaickej elektrárne tak, aby sa zabezpečil ohrev prostredníctvom tepelných vykurovacích telies (elektrických špirál) v ohrievačoch vody pre:

- ohrev teplej úžitkovej vody (TÚV),
- ohrev vody vo wellness,
- ohrev TÚV v chatkách,
- ohrev vzduchotechniky bazéna a
- ohrev podlahového vykurovania v miestnosti bazéna.

III. AKTUÁLNY STAV

Na obr. 1 je znázornená principiálna schéma zapojenia, v ktorej môžeme vidieť jednotlivé zariadenia. Tieto zariadenia sa využívajú na ohrev, zásobovanie a spotrebu vody pre vykurovanie a zabezpečenie teplej úžitkovej vody.



Obr. 1 Principiálna schéma aktuálneho zapojenia ohrevu vody [3]

Ohrev vody pre vykurovanie a zabezpečenie teplej úžitkovej vody sa zabezpečuje prostredníctvom:

- peletkového kotla (inštalovaný výkon 150 kW)
- solárnych kolektorov (inštalovaný výkon 2×10 kW)
- odpadového tepla zo zimného štadióna (inštalovaný výkon $5 \div 10$ kW)

Teplá úžitková voda vzniká ohrievaním pitnej vody v ohrievačoch vody. Teplá úžitková voda sa v našom systéme využíva pre:

- kuchyňu,
- wellness,
- chatky a
- penzióny.

Vykurovanie je riešené cez rozdeľovač a zabezpečuje sa pre:

- podlahové vykurovanie bazéna a
- vzduchotechniku bazéna.

Vykurovanie sa tiež zabezpečuje pre ohrev vody v bazéne, avšak tepelný výmenník je napojený na okruh vykurovania pred rozdeľovačom.

IV. SPOTREBA TÚV A VODY PRE VYKUROVANIE

Aby sa zabránilo poddimenzovaniu, prípadne predimenzovaniu zdroja na prípravu teplej úžitkovej vody a vody na vykurovanie, treba pri navrhovaní poznať diagram, resp. hodnoty spotrebovanej TÚV a vody pre vykurovanie.

A. Spotreba TÚV v reštaurácii

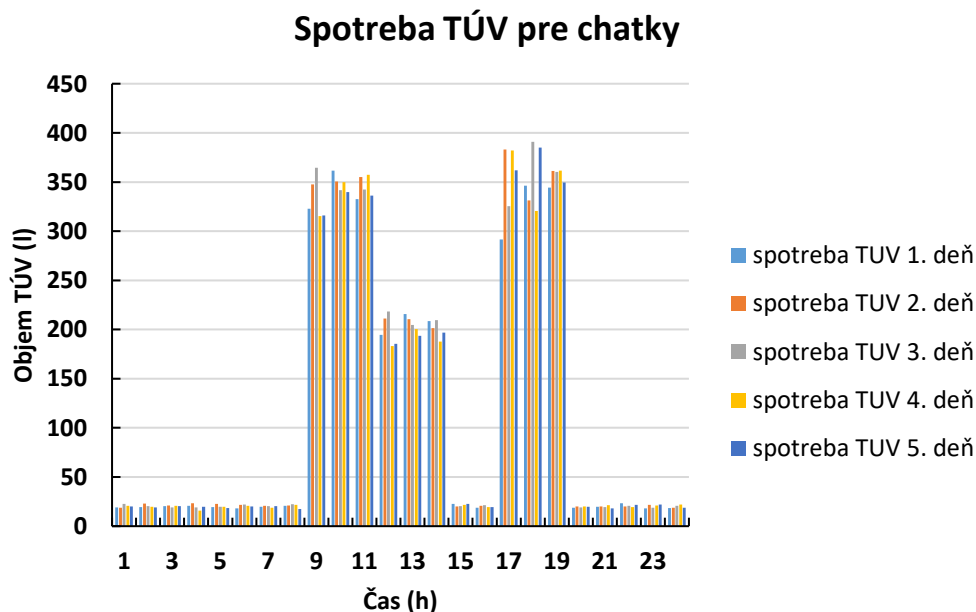
Objem spotrebovanej TÚV predpokladáme na 30 litrov na osobu. Je nutné zabezpečiť, aby bola TÚV k dispozícii v čase, keď je reštaurácia otvorená. Pri výpočte spotrebovanej TÚV uvažujeme, že denne reštauráciu navštívi 300 ľudí.

B. Spotreba TÚV pre wellness

TÚV sa vo wellness využíva pre sprchy bazéna. Predpokladáme, že wellness denne využíva 120 ľudí, pričom spotreba na jednu osobu je 20 litrov.

C. Spotreba TÚV pre chatky

V areáli horského hotela Poráč Park sa nachádza šesť chatiek. Kapacita jednej chatky je desať osôb. Uvažujeme, že spotreba na jednu osobu je 50 litrov za deň. Tepelné výmenníky sa nachádzajú v terciárnom okruhu, kde sa TÚV ohrieva z primárneho okruhu prostredníctvom tepelných výmenníkov. Spotrebovaný objem TÚV pri piatich rôznych dňoch môžeme vidieť na obr. 2.



Obr. 2 Objem spotrebovanej TUV pre chatky [3]

D. Spotreba tepelnej energie pre ohrev vody v bazéne

Teplá voda sa v bazéne ohrieva pomocou tepelného výmenníka, ktorého menovitý výkon je 24 kW. Voda v bazéne nepotrebuje akumuláciu tepla, teda voda v bazéne sa ohrieva priamo cez deň. Voda sa v bazéne ohrieva 10 hodín denne, a to v čase od 8:00 h do 18:00 h. V ostatných časoch sa voda len dohrieva s 10 % výkonom z menovitého výkonu.

E. Spotreba tepelnej energie pre podlahové vykurovanie

Podlahové vykurovanie je realizované plastovými rúrkami (hadmi), ktoré sú zaliate v betónovom potere. Pri výpočte spotreby tepelnej energie pre podlahové vykurovanie uvažujeme s menovitým výkonom 8 kW, v čase podobnom, ako pri spotrebe tepelnej energie pre ohrev vody v bazéne.

F. Spotreba tepelnej energie pre vzduchotechniku

Vzduchotechnika je určená na dohrievanie vzduchu na požadovanú hodnotu obehovým vzduchom a odvlhčovaním. Menovitý výkon vzduchotechniky je 15 kW a využíva sa v čase podobnom, ako pri spotrebe tepelnej energie pre ohrev vody v bazéne.

V. VÝROBA TÚV A VODY PRE VYKUROVANIE

Okrem tepelnej energie, ktorá sa bude získavať z FVE prostredníctvom priamych ohrievačov vody, tepelná energia sa bude získavať aj zo:

- solárnych kolektorov – inštalovaný výkon 2×10 kW a
- odpadového tepla zo zimného štadióna – inštalovaný výkon 5÷10 kW.

VI. NÁVRH A VÝPOČET ENERGETICKEJ BILANCIE

Výpočet bol realizovaný v programe Matlab v minútových intervaloch spotrebovanej a vyrobenej teplej úžitkovej vody a vody pre vykurovanie. Okrem programu Matlab sme pre konečný návrh inštalovaného výkonu FVE použili aj program PVSol, ktorý zohľadňuje slnečné podmienky a tak simuluje možnú vyrobenú elektrickú energiu pre konkrétne miesto.

Pre spoľahlivejší výsledok sme v programe Matlab výpočet realizovali počas piatich dní a vybrali sme najvyšší výkon, ktorý je nutné dodať z FVE. V nasledujúcej tab. 1 môžeme vidieť potrebný výkon FVE počas piatich dní. Maximálne hodnoty výkonov sa pohybujú v rozmedzí od 39,322 kW do 41,366 kW.

Tabuľka 1
 Vypočítaný výkon FVE [3]

Čas	Výkon FVE (kW)				
	1. deň	2. deň	3. deň	4. deň	5. deň
1:00	1,142	0,820	1,429	0,807	1,154
2:00	1,531	0,920	0,792	1,355	1,289
3:00	1,176	1,239	0,668	0,981	0,765
4:00	1,392	1,178	1,233	0,990	1,097
5:00	1,329	0,943	0,827	1,114	0,932
6:00	1,027	0,708	1,113	1,064	1,255
7:00	1,180	1,358	1,280	0,788	0,701
8:00	36,035	41,221	39,499	39,572	41,366
9:00	39,322	38,752	41,109	34,999	40,385
10:00	36,991	40,709	39,456	38,511	39,513
11:00	37,836	38,655	37,127	37,609	36,430
12:00	34,568	38,083	36,046	37,793	33,538
13:00	33,505	34,172	36,576	35,479	36,404
14:00	33,295	36,799	36,531	34,543	35,179
15:00	32,542	32,039	34,128	31,608	31,179
16:00	32,137	32,483	33,378	32,420	31,757
17:00	39,061	37,693	37,096	38,467	37,690
18:00	1,247	0,897	1,041	0,882	1,239
19:00	1,142	1,260	1,237	1,145	0,814
20:00	1,330	1,126	0,785	1,207	1,461
21:00	1,149	1,287	1,149	1,045	1,329
22:00	0,734	0,879	0,881	1,419	1,300
23:00	0,843	0,890	0,882	1,488	0,953
0:00	0,825	1,142	1,396	1,337	1,190

Najvyšší výkon, ktorý je potrebný pre zachovanie energetickej bilancie, dosahoval hodnotu 41,366 kW. Na tento výkon by mala byť navrhnutá fotovoltaická elektrárň. V programe PVSol sme realizovali niekoľko rôznych simulácií pri rôznom inštalovanom výkone FVE. Z týchto simulácií sme navrhli výkon FVE tak, aby boli investičné náklady a náklady na prevádzku čo najmenšie tak, aby bola zachovaná spoľahlivosť prevádzky a energetická bilancia. Výsledný inštalovaný výkon FVE je 52,92 kWp, a je znázornený na obr. 3.



Location:	Poprad
Climate Data Record:	Poprad (1986-2005)
PV Output:	52,92 kWp
Gross/Active PV Surface Area:	365,99 / 333,17 m ²

Obr. 3 Inštalovaný výkon FVE [3]

Pri nedostatočnej výrobe elektrickej energie z FVE a teda pri nedostatočnom ohreve vody pre vykurovanie a TÚV, bude sa využívať na dohrievanie elektrokotol. Elektrokotol je zapojený v sérii za ohrievačom vody s priamym ohrevom. Inštalovaný výkon elektrokotla je 45 kW. Hodnoty výkonu fotovoltaickej elektrárne a elektrokotla pre jednotlivé hodinové intervaly a rôzne slnečné podmienky, sú uvedené v nasledujúcej tab. 2.

Tabuľka 2
Spolupráca FVE a EK [3]

Čas	Spolupráca fotovoltaickej elektrárne a elektrokotla					
	Slnečný deň			Zamračený deň		
	FVE (kW)	EK (kW)	Prebytok FVE (kW)	FVE (kW)	EK (kW)	Prebytok FVE (kW)
1:00	0,00	0,818	0,000	0,00	0,791	0
2:00	0,00	0,986	0,000	0,00	1,157	0
3:00	0,00	1,048	0,000	0,00	1,010	0
4:00	0,20	0,763	0,000	0,00	1,045	0
5:00	0,70	0,905	0,000	0,00	1,115	0
6:00	6,70	0,411	0,000	0,65	0,000	0,025
7:00	18,60	3,984	0,000	1,61	9,145	0
8:00	32,20	18,563	0,000	2,16	36,561	0
9:00	40,60	5,461	0,000	3,02	40,431	0
10:00	48,40	0,000	3,131	3,24	34,903	0
11:00	51,80	0,000	16,937	3,52	31,758	0
12:00	52,70	0,000	24,883	3,19	32,072	0
13:00	48,80	0,000	16,111	2,87	30,533	0
14:00	42,70	0,000	12,934	2,30	31,730	0
15:00	33,10	0,000	0,076	1,77	30,363	0
16:00	21,70	9,664	0,000	1,77	29,688	0
17:00	9,80	25,255	0,000	1,17	37,484	0
18:00	1,20	10,101	0,000	0,61	12,572	0
19:00	0,30	7,052	0,000	0,05	5,506	0
20:00	0,00	1,323	0,000	0,00	1,091	0
21:00	0,00	1,228	0,000	0,00	1,071	0
22:00	0,00	0,886	0,000	0,00	1,068	0
23:00	0,00	1,231	0,000	0,00	1,414	0
0:00	0,00	1,053	0,000	0,00	1,139	0

VII. ZÁVER

Tento príspevok sa zaoberal výpočtom energetickej bilancie horského hotela v Poráčskej doline s využitím obnoviteľných zdrojov energie. Ako obnoviteľný zdroj sa využila energia zo Slnka – fotovoltaická elektrárňa. Tepelná energia, ktorá je potrebná pre ohrev TÚV a ohrev vody pre vykurovanie, ohrieva sa v ohrievačoch vody s priamym ohrevom, kde sa využíva elektrické vykurovacie teleso, ktoré je napájané práve z tejto fotovoltaickej elektrárne.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18.

LITERATÚRA

- [1] Inforse, *International Network for Sustainable Energy*, „Prečo potrebujeme obnoviteľné zdroje energie?“ [online]. [cit. 2020-05-19]. Dostupné na internete: < <http://www.inforse.org/europe/fae/OEZ/preco/preco.html> >
- [2] Oravcová, E., „Fotovoltaika, technológia „budúcnosti“ aplikovaná v tvorbe študentov architektúry“ [online]. [cit. 19-05-2020] Dostupné na internete: < <https://www.siea.sk/wp-content/uploads/files/medzinarodne/projekty/pure/Fotovoltaika.pdf> >
- [3] Matuška, T., „Typy solárnych kolektorů“ [Online] Nejnavštěvovanější odborný portal pro stavebnictví a technická zařízení budov. 2020. [cit. 17.03.2020] Dostupné na internete: < <https://oze.tzb-info.cz/solarni-kolektory/154-typy-solarnich-kolektoru> >
- [4] Jelínek, V., „Návrh úsporného ohřevu teplé vody“. Praha: 2007, GAS s.r.o [cit. 25.03.2020] ISBN 978-80-7329-137-3.
- [5] Fáber, A., Iliaš, I., et. al., „Slnko k službám: možnosti využitia slnečnej energie“, Bratislava: 2010. Energetické centrum Bratislava. ISBN 798-80-969646-1-1.
- [6] Baláš, M., „Kotle a výměníky tepla“, Brno: 2009, Akademické nakladatelství CERM. Učební texty vysokých škol. ISBN 978-80-214-3955-9.
- [7] Stehlík, P., Kohoutek, J., Nemčanský, J., „Tepelné pochody: Výpočet výměníku tepla“, Brno: 1991, Vysoké učení technické. Bakalárska práca.
- [8] Tomčo, M., „Výpočet energetickej bilancie horského hotela v Poráčskej doline s využitím obnoviteľných zdrojov energie“. Diplomová práca. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2020.

Application of Educational Computer Games in Algorithms and Data Structures Courses

¹Kinga NAGYOVÁ, ²Slavomír ŠIMONÁK

¹Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

²Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

¹kinga.nagyova.2@student.tuke.sk, ²slavomir.simonak@tuke.sk

Abstract – In this paper we analyze Oguz Aks and Gagnes model and the importance of educational games. We created a game, IT Adventure, based on Gagnes model, which can be helpful in teaching the subject Algorithms and Data Structures. After description of the implementation of the game, follows the evaluation of the game itself, which took place in the form of a questionnaire. The positive results of the evaluation show that learning through games is a very attractive and useful way of gaining new knowledge.

Keywords – algorithm, data structures, educational games, Gagne based model, gamification, model of educational game design

I. INTRODUCTION

Algorithms and data structures are an important part of a well-structured software or a computer network. Mastering them is a very important task that can cause problems for students. Students need to learn new, abstract concepts and understanding of them cannot be left to fantasy alone. There are many visualization and simulation methods to facilitate the acquisition of algorithms and data structures. Computer games that try to pass on new knowledge [1] are included as well.

Games have been a part of human life since childhood and are known to have a developing effect. Through games, children learn the basic rules of social behavior, develop their memory, imagination, and body as well [2]. During the game, the player can reach the Flow state, which was determined by Mihály Csíkszentmihályi. This state expresses the level of happiness of the individual, when he concentrates maximally on the given task and its performance, forgetting the passage of time and physical needs [3]. In pedagogy there is a growing space for learning through play and the pursuit of a state of flow.

Gamification elements incorporated into education have a positive effect on raising students' attention [4]. Teaching with computer games is becoming more popular and can make it quite attractive for students to acquire new knowledge. Oguz Ak has created a model that defines how to design a computer game for educational purposes (Fig. 1). The model consists of 3 parts: **Input, Process, Outcome**. The input part contains the game characteristics and the curriculum. The process section defines the learning process aimed at achieving the Flow state. The outcome part evaluates the performance in the game and tests the knowledge acquired by the students. The game should serve as a sufficient motivation for the player to gain additional knowledge as well as continuously improve himself [5].

The visualization of algorithms and data structures during a computer game can be: **Algorithm Text**, which contains a description of the operating principle of the given algorithm or data structure. **Algorithm Flowchart** that statically represents the operating principle of the algorithm, data structure. **Game Demonstarion**, which dynamically depicts the principle of operation of the algorithm / data structure. The **Game** itself, which simulates the operation of the algorithm / data structure by requiring interactions [6].

We have created a game called **IT Adventure**, which helps to learn the algorithms and data structures learned in the Algorithms and Data Structures course in a playful way. This is an edutainment game designed for fun and educational purposes with game-reduced curriculum. The aim of the game is to get the students attention, better involve the students in the course of the lessons, and help them to learn and grasp and practice the new knowledge in a playful way.

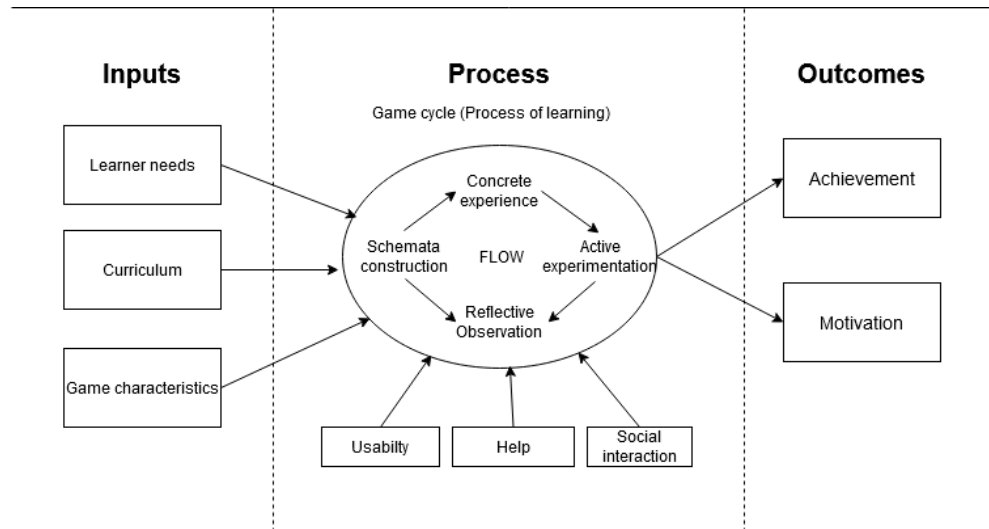


Fig. 1 Model of educational game design [5]

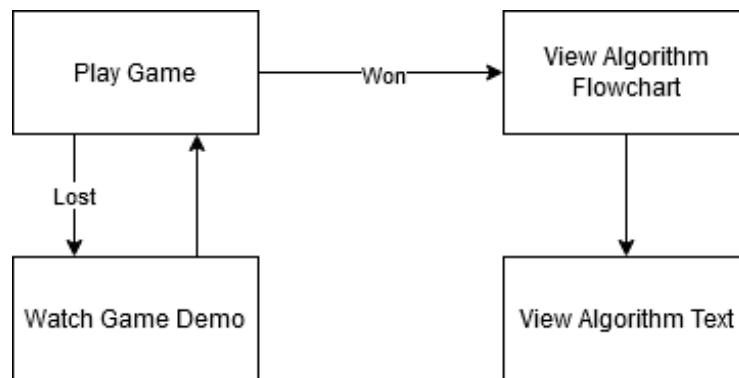


Fig. 2 Gagne based model [6]

II. IT ADVENTURE

Oguz Aks model served as the basis for designing the game, meaning the game should be fun, pass enough new knowledge, and motivate individual development. Using the 4 algorithm visualization methods mentioned above, we had to create the game with the help of Gagnes Nine Events of Instruction. The game works based on the Gagne model as follows: The player starts playing the game in which he has to solve tasks. If the player is unable to complete the task, they can ask for help or watch a demo version of the game. The player can now complete the tasks. After each completed task, the player is given an algorithm descriptive text or diagram of the particular algorithm, which he can master after analysis. A players performance can be measured by completing tasks: how many times he has won and how many times he has lost (Fig. 2) [6].

A. Scenario of the game

The professor gave his student a task to prepare from the scripts. The student can collect these scripts only by solving certain tasks in the school premises. A student can prepare for the final exam and finish the school year with honors only if he / she successfully solves the tasks and collects all the scripts. The player plays with the character of the student and after successfully completing the tasks he/she gets the necessary scripts.

This game was inspired by an educational game created by students at the Mumbai University that introduces DFS (Depth First Search) and BFS (Breadth First Search) algorithms to players in a fun way [7].

Tasks are divided into different tracks according to topics. The game contains a total of 6 levels, each level working on a different topic. In each level or track the player has 3 or 4 tasks to solve. The included topics are: **Tree traversal, Binary Search Tree, Sorting Algorithms, Stack, Queue, Linked List.**

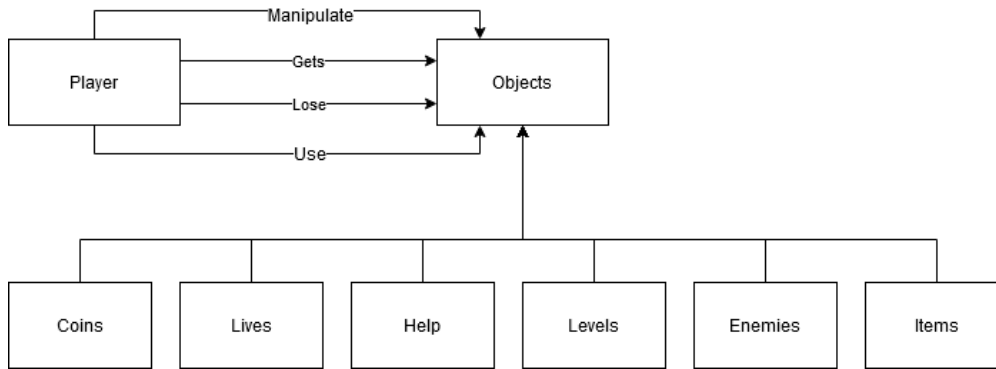


Fig. 3 Conceptual model of the game

B. Conceptual model of the game

At the heart of the game is the player who must complete the tasks within each level. The player encounters enemies within the tracks and encounters obstacles, ensuring a classic gaming experience for the player. To complete the tasks, the player must work with different items: either to move the given item from one place to another, or to jump on the top of items in the correct order. Before each task, the player finds a brief instruction on what to complete in that task. After each task, there is a gate that only opens if the player has successfully completed that task. This prevents the player from running through the level without completing the tasks, and collecting the necessary script. The player starts the game with 5 lives and 50 coins. The player collects coins during the game for which he can purchase help. The player gets a new life for every 100 coins. The player can gain and lose lives during the levels. Because the game is based on the Gagne model, the player is given a description of the algorithm associated with the task after each correctly solved task. If the player gets stuck solving the task, he can claim this description sooner, but it will cost the player 50 coins. If the player has successfully completed all the tasks on the level, he obtains the script at the end of the level. This script indicates successful completion of the given track. The player can replay the already completed level anytime, only the already acquired script will not appear at the end of the track. Conceptual model of the game is shown on Fig. 3.

C. World and the levels of the game

After starting the game, the player enters the **World**, which depicts a computer lab. This "room" has 6 buttons and a huge screen (Fig. 4). The player can activate each level by jumping on each button. In this case, information of the activated level is displayed on the screen. The player can enter each level by pressing the "Enter" key.

Level Tree Traversal: this level introduces the player to the tree traversals. Specifically with level-order, inorder, preorder and postorder traversal. In this course, the player must traverse the binary tree according to the correct principle. The player must step on the items of the binary tree in the correct order.

Level Binary Search Tree: in this level, the player must build a binary tree from the specified items. Here the same task is repeated, only on a different level of difficulty. The player must grab the items that symbolize the array and place them in the correct positions on the binary tree (Fig. 5).

Level Sort: this level introduces the player 3 sorting algorithm, namely the *bubble sort*, *selection sort* and *insertion sort*. In this level, the player must jump on the items according to the rules of the algorithm and sort them.

Level Stack and Queue: these 2 levels contain 3 identical tasks, only depending on the level, the task must be solved according to the operating principle of the stack or queue. With the tasks, the player practices the *Push / Enqueue*, *Pop / Dequeue* principles, and all elements of a filled stack / queue must be transferred to an empty stack / queue.

Level Linked List: This level helps to practice the linked list. The player becomes familiar with the *Insert*, *Delete*, and *Cut* operations by moving the arrows, which symbolize the pointers, to the correct location in the correct order [8].

Implementation

To create the game, we used the **Unity3D game engine**, which supports the C# programming language. The popularity of Unity lies in the fact that it has a very user-friendly editor and can be

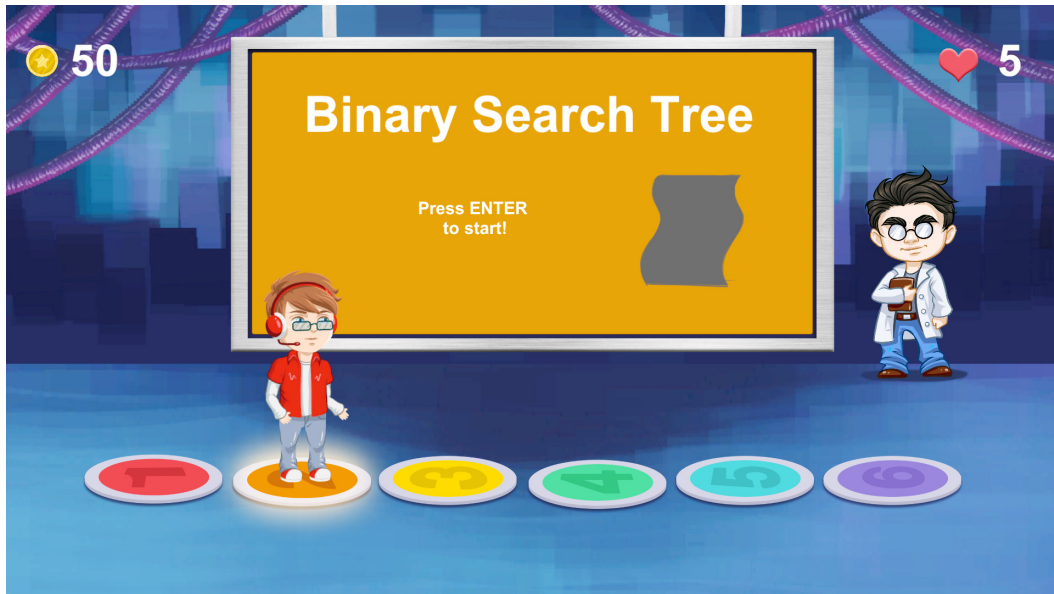


Fig. 4 World of the game



Fig. 5 First task of Binary Search Tree level

used to create games on several platforms at the same time. IT Adventure is currently available on Windows and Linux platforms.

Another specialty of Unity is that it works with **scenes** and **camera views**. The scenes allow us to separate the individual tracks from each other, i.e. each task appears in a separate scene between which we can provide a passage. The camera view determines what the player sees from that scene as the game runs.

The Unity 3 package allows you to animate the characters used in the game. The 3 packages are: *Animation 2D*, *2D IK*, *2D PSD Importer*. Thanks to these packages, we were able to animate the movement of the players avatar ourselves. Thus, there is no need to draw individual phase images separately.

The game consists of different **objects**, the components of which are the **scripts (classes)** themselves, that control the operation of the objects. The component of a players avatar is the *PlayerScript*, which controls the players movement, loss of life, and just launches the appropriate animation. Player data is stored in the *PlayerData* class. Player data includes, for example, the number of lives, the number of coins earned, and boolean-type variables named after the levels. If a player has succeeded in completing a particular level, obtained the script at the end of the level, then the *PlayerData* bool type variable for that level is given a true value. Two methods

are defined here: *Save()* and *Load()*. The *Save()* method saves the *PlayerData* data to a file under the operating system, while *Load()* reads the data saved in this file. The *PlayerData* class is responsible for maintaining the player's data, and it is not assigned to any object as a component.

Other objects in the game, such as *Checkpoint*, *Holes*, *Finish*, *Help*, *Gate*, *Lives* and *Coins* objects, are controlled by scripts with same name.

Tasks that require moving game elements are controlled by the *Objective* script. The *RegisterEvent()* method in this script checks whether the task has been completed or not. Upon successful completion of the task, it opens the gate for the task, displays a description of the algorithm, and sets the next task of the player. Movable items are controlled by *GameItem* script. Each item can only be placed in the place assigned for it or in the place from which the player took it. The positions where the player can place or pick items are controlled by the *ItemSlot* script.

The Tree Traversal track is controlled by the *LevelOrderManager* script. The items that a player must touch in the correct order are controlled by the *CollisionDetection* script. When a player touches an item, *CollisionDetection* class assigns that item as a parameter to the *LevelOrderManager* script. The *OnTouch()* method compares the currently affected item with a list called *Elements[progress]*, where we pre-saved the correct order. If the item in question is in the correct order in the list, the color of the item changes to green and the progress variable of type int increases. This comparison lasts until the progress variable reaches the number of items in the *Element* list. When this is done, the task is completed successfully, then the gate opens and help is displayed. The *Sort* level works on a similar principle, only here the main task is controlled by *SortOrderManager* script and the contact of the items is examined by *SortCollisionDetection*. In these tasks, the player only needs to make a comparison according to the principle of the algorithm, the exchange of items is provided by the *SortOrderManager* itself.

During the tasks, the player receives feedback on the progress of the game with sound effects and texts. For example, if a player makes a mistake while solving a task, the player is returned to the last activated *Checkpoint*, a text called *Wrong step!* is displayed, and the sound effect assigned to it is played [8].

Questionnaire

After the game was finished, 25 people set out to try it out and share their opinions. All 25 applicants were students of the Technical University of Košice, who have already successfully completed the course *Algorithms and Data Structures*. This way, they can comment on whether this game would have proved useful in their studies.

Testing took the form of a questionnaire that respondents completed after testing the game. The questionnaire included 15 and one bonus question. Since the game is based on Gagne's Nine Events of Instruction, in the first set of questions we wanted to get an answer to how well the game managed to meet the points formulated by Gagne. The game catches the attention of 84% of respondents. Also, learning in this form is considered very attractive. More than half of the respondents, i.e. 15 people, would be inspired by this game to participate more actively in the lessons as well as to improve themselves.

In the second part of the questionnaire we wanted to get an answer to how effective the application of the Gagne model in a computer game is, i.e. how effective it is to display the text describing the given algorithm after completing the task. Respondents rated the Gagne model itself positively, but the game received a mixed rating when they tried out the help function of the game. This can motivate us to further develop the game and expand the description of the help section and algorithms.

In the last questions of the questionnaire, we asked about the game itself. The majority of respondents find the game fun and useful in terms of learning and practicing algorithms. We asked respondents which visualization of topics was most useful to them. Most of the votes went to binary search trees, which also includes tree traversals.

In the last question, the bonus question, respondents were able to share their views on the game. Respondents consider the game a nice idea, but would recommend it mainly to the younger age group (teenagers). Ideas and suggestions were received to make the user interface better and more convenient, and players would be happy to be able to influence the control of the player's avatar [8].

III. CONCLUSION

In this paper, we analyzed Oguz Aks game design model and Gagne's model, which served as inspiration for creating our own game, *IT Adventure*. This game can be used as part of the *Algorithms and Data Structures* course. The essence of the game is to introduce and

practice curriculum learned in the course in a fun way. Of course, the game is no substitute for programming itself, but it can be a great help in mastering the theory. The evaluation of the game took the form of a questionnaire. The responses to the questionnaire show that learning in the form of a computer games is very attractive to students. Gaming elements built into lessons, or learning with a computer game itself, increase students class activity and can serve as a motivation for self-improvement. The results of the questionnaire show that it is worthwhile to make additional educational games based on the Gagne model. As well as additional developments and new tasks, IT Adventure can be a useful part of education.

ACKNOWLEDGMENT

The first author of the paper is very grateful for the great support of the family, as this work could not have been realized without it. Special thanks to my sister, Rita Nagyová, who designed and realized the visual world of the game.

REFERENCES

- [1] T. Susi, M. Johannesson, and P. Backlund, "Serious games: An overview," 2007.
- [2] C. E. Schaefer and S. E. Reid, *Game Play: Therapeutic Use of Childhood Games*, 2nd ed. John Wiley & Sons, 2004.
- [3] M. Csíkszentmihályi, *Az áramlat. A tökéletes élmény pszichológiája*. Akadémiai Kiadó, 2018.
- [4] R. Pinter, S. M. Čisar, Z. Balogh, and H. Manojlović, "Enhancing higher education student class attendance through gamification," *Acta Polytechnica Hungarica*, vol. 17, pp. 13–33, Jan. 2020.
- [5] A. Oguz, "A game scale to evaluate educational computer games," *Procedia - Social and Behavioral Sciences*, vol. 46, pp. 2477–2481, Dec. 2012.
- [6] S. Sahar and J. Chen, "Simplifying algorithm learning using serious games," *WCCCE 2009 - Proceedings of the 14th Western Canadian Conference on Computing Education*, pp. 34–41, May 2009.
- [7] N. Ansari, N. Parikh, A. Narvekar, M. Phatapekar, and A. Yadav, "GamEd - learning data structure algorithm using computer game," *IOSR Journal of Computer Science (IOSR-JCE)*, pp. 62–67, 2014.
- [8] K. Nagyová, "Possibilities of application of educational games in algorithms and data structures courses," Master's thesis, Technická univerzita v Košiciach, May 2020, (in Slovak).

Applying Automated Testing Tools for Desktop Applications with Graphical Interface

Matúš KOKINDA,¹ Zdeněk HAVLICE,
²Ján TREMBULÁK,³ Iveta KINDERNAYOVÁ

¹Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic

²R-SYS s.r.o., Bielerkevska 29, 040 22 Košice, Slovak Republic

³R-SYS s.r.o., Zgútha Vrbického 4291/20, 031 01 Liptovský Mikuláš, Slovak Republic

matus.kokinda@student.tuke.sk, ¹zdenek.havlice@tuke.sk,
²jan.trembulak@r-sys.sk, ³iveta.kindernayova@r-sys.sk,

Abstract — The topic of this article is the test automation issue of desktop applications with graphical user interface. The research was based on the analysis of the current testing process for R-SYS Ltd. and evaluation of manual test automation options through free automation tools. Due to the complexity of the developed systems, a proposal of methodology for automatic testing was developed, which differs from standard procedures by the application of automation tools using computer vision. The methodology was experimentally verified and evaluated in the process of C2D application testing, which is part of the VERA-NG passive tracking system. The result of the solution is a defined method of automatic testing that can be generally applied to desktop applications with a graphical user interface. By applying the methodology in practice, it is possible to effectively approach testing automation, independently of the source code of the system under test and without increasing project costs.

Keywords — GUI, automated regression testing, desktop applications, testing tools, methodology

I. INTRODUCTION

An essential part of software development is its thorough testing, which aims to verify in detail the functionality, performance, compatibility and security of the proposed solutions. Testing generally increases user confidence in the quality of systems, including a wide range of software products with different user interfaces. The current trend in software application development is focused on intuitive control of the system, through a graphical user interface. The GUI complexity and variety of technologies used to display it can pose a potential time and technical risk in automation testing. The problem is especially noticeable in the automatic control of GUI desktop applications. Currently available testing tools are highly dependent on the GUI technology used, or do not support all GUI components used in the system under test.

The aim of the research was to propose a suitable methodology of automatic testing that could be generally applied to desktop applications with GUI. In theoretical part, the problem of automation testing was analyzed, especially from perspective of regression testing and existing techniques of automatic GUI control. There are three categories of software regression errors [7]: local errors, remote errors, uncovered errors. The above mentioned errors are mainly caused by changes in the tested software. Change is a key concept in regression testing, and the reasons for these changes can generally be divided into four categories [8]: implementing new and modifying existing functionality, integration, bug fixes.

Subsequently, the current state of testing for R-SYS Ltd. was evaluated and the existing market tools for desktop application testing were analyzed. The practical part of the research contains a design of the methodology and its application into the automation testing process of C2D application [2], which is used to display the air situation from radar sensors, over the map base. The result is the methodology of automatic testing of R-SYS software products.

II. ANALYSIS

R-SYS Ltd. is engaged in the development and production of innovative IT solutions for aerospace, defense and security industry. Product line consists of SW solutions in various domains such as ATM, Surveillance Data Processing, Aeronautical Information Management, Digital Audio, Video & Data Recording Systems, and Geographical Information Systems (GIS) [2]. Company R-SYS Ltd. in the development and maintenance of desktop applications with a graphical user interface is mainly using the manual type of integration, system and acceptance tests execution. Various simulation tools and mechanisms are used to test these critical systems. In the process of analyzing currently available software products for testing GUI native applications, several commercial and open source testing tools were validated in order to evaluate their effectiveness, usability and possibilities of mutual integration. The standard testing of the tools was in the environment of the Windows 10 operating system, with the C2D application installed. The individual tools were experimentally integrated into the testing process, based on which it was necessary to verify the following properties:

- automatic launch specific tested application,
- automatic start and execution of supporting simulation tools and scripts,
- basic test case automatic execution and identification of known bugs,
- automatic reports generation.

Based on the knowledge gained from the analysis of existing testing tools on the market was assessed, that there is currently no testing tool available that would meet the complex project testing requirements. Therefore, an alternative solution was proposed using open source available automated tools that interact with each other to create a comprehensive test system, independent of access to GUI and source code of the system under test.

III. DESIGN OF AUTOMATIC TEST SYSTEM

The proposed solution includes three main tools that will enable automation of the manual testing process at various levels through mutual integration. These tools include: automatic GUI control tool (SikuliX) [3], tool for managing, running and evaluating tests (JMeter) [5], test data simulation tool (AIS, UFE Simulator) [2].

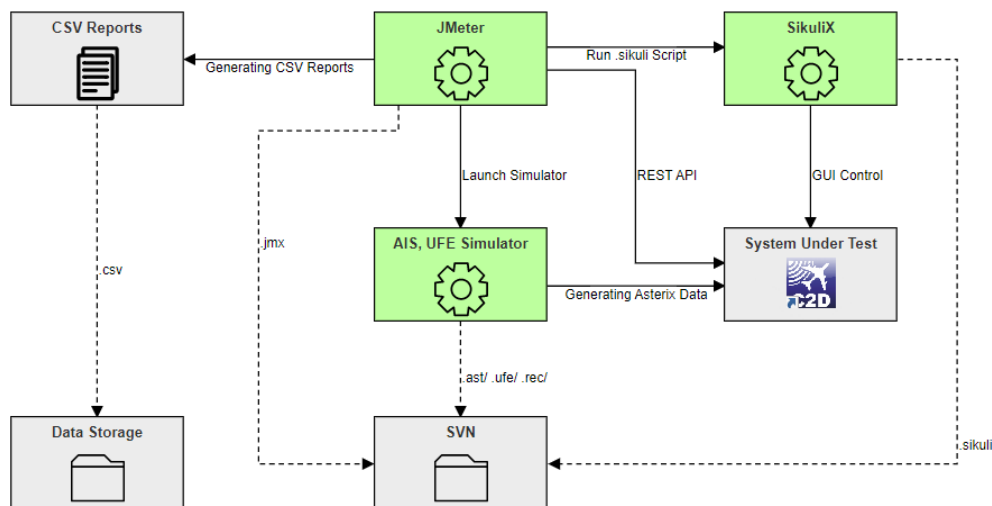


Figure 1 Topology of integrated automation tools and mechanisms

Based on the requirements analysis, it is advisable to focus on free software. However, it is also necessary to choose a software solution that can be implemented on all the mentioned platforms. The system of automatic testing must be generally applicable to desktop applications with a graphical interface in OS Windows 10 and must allow to perform all possible actions over GUI objects SUT (System Under Test). It must also be integratable with existing testing tools and mechanisms in the company and must support basic work with the file system, automatically run internal simulation tools (scripts) and third-party applications. After testing, it is required to automatically evaluate the test results and generate a report in CSV format.

The automatic testing system is based on topology in Figure 1. The main components are three selected automated tools marked in green. By their mutual interaction, processes are executed in parallel with a time sequence, according to the prepared test plan. The source code

and sample test data of these tools are versioned via TortoiseSVN [9] subversion system. Generated test reports in CSV format are stored separately in the test data storage. The data storage structure corresponds to the current version numbering of the tested application.

If an error is detected during the automatic test execution, the report will be supplemented by a separate directory, which will contain the current screenshot at the time and the expected object on the screen (Figure 2).

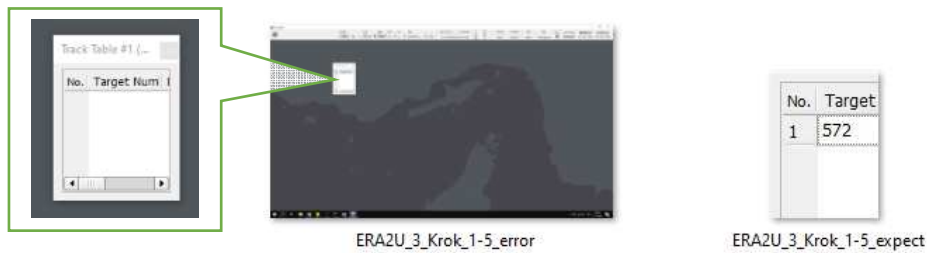


Figure 2 Generating screenshots

In this way, faster errors identification is ensured. The automatic testing system applies the technique of GUI automatic control based on computer vision. The GUI object in the user interface of the system under test is localized based on a graphical representation. This functionality is provided by open source SikuliX automation tool [3]. The principle based on optical object recognition was designed for the following reasons:

- unsuccessful connection to the GUI by analyzed testing tools,
- testing of applications requiring GUI object manipulation over the map base,
- testing applications in various operating systems,
- usability for native and web applications testing,
- graphical verification possibilities of GUI objects on the screen.

Managing automated tests, running tests, generating reports, and executing operating system commands is provided by the JMeter tool [5] from the Apache Software Foundation. The JMeter tool is also connected with internal simulation tools, basic on which it is possible to display current data for specific types of tests in SUT.

To implement the solution correctly, it is necessary to use several software tools. These are described in the following sections.

A. SikuliX

SikuliX [3] is an open source tool that automates anything displayed on the screen of computer running Windows, Mac or some Linux/Unix. It uses image recognition powered by OpenCV to identify GUI components [1]. This is handy in cases when there is no easy access to a GUI's internals or the source code of the desktop application or web page.

Tool allows users to take a screenshot of a GUI element (such as a toolbar button, icon, or dialog box) and use the screenshot instead of elements name as shown in Figure 3. SikuliX also provides a visual scripting API for automating GUI interactions, using screenshot patterns to direct mouse and keyboard events.

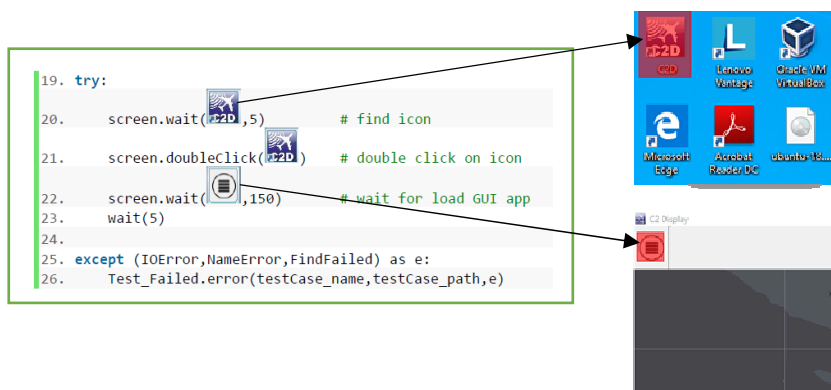


Figure 3 Visual scripting and image recognition

A SikuliX script (.sikuli) is stored in a directory that consists of a Python source file (.py), and all the image files (.png) used by the source file. All images used in a SikuliX script are simply a path to the .png file in the .sikuli bundle.

This tool does not contain components for creating test plans, evaluating tests and generating reports. These functions are provided by the JMeter automation tool.

B. JMeter

The Apache JMeter [5] application is open source software designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but now the tool contains components for testing FTP (File Transport Protocol), databases, emails, etc.

JMeter works through a test plan designed by the user. The test plan is a list of processes that are performed during testing. It can contain logic controllers, samplers, timers, and other configuration elements. All these elements belong to a specific hierarchy and therefore their placement in the test plan has an impact on the automated test execution. In the test plan, each automatic SikuliX script is assigned exactly one OS Process Sampler component, which allows its execution. When the process is started, the simulation is executed automatically using the SikuliX tool. After completing the simulation, JMeter evaluates the test result based on the logs that were created during the SikuliX script execution. The test results are summarized in the View Results Tree component, where they are automatically generated into the final report in CSV format.

The topology represented in Figure 1 shows the direct interaction of the JMeter tool with the tested C2D application, via the REST API. This functionality is available based on the existing internal debugging interface of the C2D application, which expands the possibilities of data simulation and GUI control.

IV. METHODOLOGY

Based on the knowledge and experience with the used automation tools, a methodological procedure was created, which describes the application of the VSBT (Visual Script Based Testing) methodology in the software life cycle. The basic steps of the methodology are incorporated into the process model, which is shown in Figure 4. The model consists of two parts. The left part of the diagram represents the basic testing process. In the right part, the basic phases of the methodology are defined below [6]:

- analysis of risks and objectives in the application of automatic testing,
- test environment preparation,
- transforming manual test cases into test scripts,
- generating reports,
- maintenance of automatic tests in post-project phases.

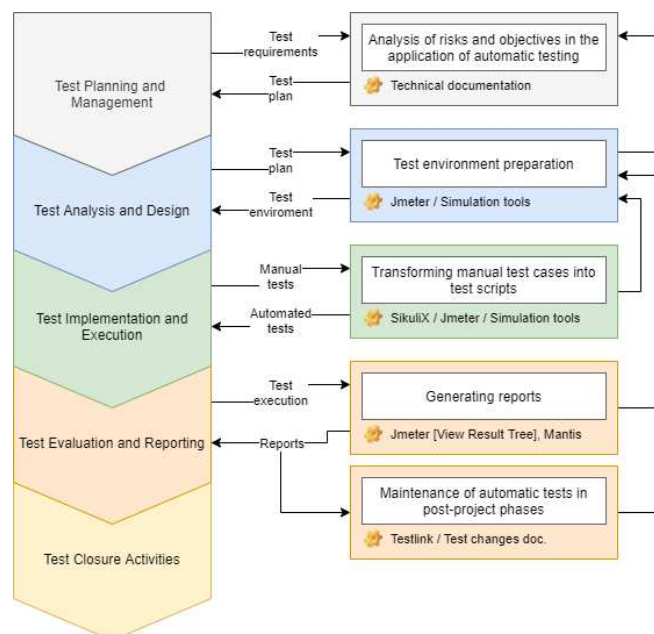


Figure 4 Process model of VSBT methodology

More detailed information on the individual steps and activities of the methodology were defined in the methodological manual. This manual describes exemplary steps of using the VSBT methodology in software projects.

A. Experimental verification

The VSBT methodology was experimentally verified in the process of C2D application testing. The C2 Display (C2D) application is used to present UFE, ASTERIX and AIS information over the map base [2]. The software is designed for VERA-NG passive surveillance system operators [4], allowing users to change display parameters based on the attributes of observed objects using a GUI.

A set of release regression tests was automated during testing. These tests verify the basic functionality across the entire C2D application in simple steps. So far, the tests have been executed manually, in regular iterations before sending the setup to production. During test automation, it was necessary to verify:

- correct GUI object identification and localization on the screen,
- correct screenshots generation if an error is detected,
- correct report generation,
- possibilities of the test system interaction with the user,
- connection of test system with internal simulation tools.

The C2D application contains standard GUI control objects such as buttons, radio buttons, check and text boxes, menu lists, trees, bookmarks, etc. All objects were easily identified and correctly located during the automated testing. For better maintenance of SikuliX test scripts, it was necessary to implement a set of auxiliary functions for frequently repeating steps of test cases.

For better control over the automated test, it is useful to apply functionality that allows SikuliX scripts to interact with the user during execution. If the test fails, user will be prompted for the way to handle the situation. A dialog box appears containing a GUI object that could not be identified on the screen and over which an action should have been executed [6]. Then the user has the following options:

- *retry* – try again to find the image (GUI object),
- *skip* – continue the script execution,
- *abort* – end the script.

If the user selects the *abort* option, the current screenshot and the expected GUI object are generated in a separate directory as shown in Figure 2. At the end of the test, a test report in CSV format was generated, which was used to analyze several errors.

V. EVALUATION

In the process of experimental application of the VSBT methodology in the C2D project, some limitations were identified, and several advantages of the automatic testing system were verified.

Advantages:

- based on open source automation tools,
- independent of SUT source code,
- usable for desktop and web applications testing,
- supported in Windows, Mac, Linux,
- modularity – the test system can be modified depending on the project requirements,
- supports computer vision GUI control and image verification.

Limitation:

- during testing, the test system actions keep the interaction over the computer (it is not possible to perform several activities on the computer in parallel),
- GUI object recognition depends on the screen resolution (it is recommended to create automatic tests with a screen resolution of 1920 × 1080 and more).

The VSBT methodology has also been successfully validated in web applications. Anyway, for web applications, it is recommended to keep the existing GUI control method based on HTML objects. For desktop applications, it is recommended to unify the process of test automation using the VSBT methodology and to replace the existing partial solutions in the company [6].

Automated testing using VSBT methodology costs less effort than manual test cases execution.

The manual execution of the release test case takes an average of 6-8 hours. By applying automation using the VSBT methodology, the average release test execution time was reduced to 2-2.5 hours. This means significant cost savings over time.

VI. CONCLUSION

This article dealt with the test automation issue of desktop applications with a graphical user interface. Based on knowledge and experience with open source automation tools, it was possible to design an integrated automation test system. The possibilities of using the test system in software projects were defined in the VSBT methodology.

The VSBT automated testing methodology differs from standard testing procedures by the application of automation tools using computer vision. These tools are different in their approach to GUI control issue than the tools using the source code of the system under test. By applying the methodology, it is possible to effectively approach testing automation, independently of the source code of the system under test and without increasing project costs

ACKNOWLEDGMENT

I would like to thank R-SYS Ltd. and its employees for the opportunity to conduct research in the field of automated testing.



REFERENCES

- [1] SikuliX, How Sikuli Works - Sikuli X 1.0 documentation, 2012, [cit. 2020-04-25], Available online: <http://doc.sikulix.org/devs/system-design.html>.
- [2] R-SYS Ltd, C2 Display – User manual (Internal project documentation of the C2D project), 2018, [cit. 2020-04-25].
- [3] SikuliX, Automate what you see on a computer monitor, 2020, [cit. 2020-04-25], Available online: <http://www.sikulix.com/>
- [4] ERA a.s., VERA-NG - passive tracking and identification system, 2020, [cit. 2020-04-25], Available online: <https://www.era.aero/cs/military-security/vera-ng>.
- [5] APACHE JMeter, The Apache JMeter™ open source software, 2020, [cit. 2020-04-25], Available online: <https://jmeter.apache.org/>.
- [6] KOKINDA, M., Applying automated testing tools for desktop applications with graphical interface, 2020, [cit. 2020-04-25]. Master's thesis. Technical University of Kosice, Faculty of electrical engineering and Informatics, Dept. of Computer and Informatics. Thesis supervisor Zdeněk Havlice.
- [7] BUREŠ Miroslav, RENDA Miroslav; DOLEŽEL Michal. Efektivní testování softwaru: klíčové otázky pro efektivitu testovacího procesu. Grada Publishing as, 2016. ISBN: 978-80-247-5594-6.
- [8] BESSONOVA Tatiana. Regression Testing Strategies: an Overview [online]. 2018 [cit. 2019-5-4]. Available online: <https://www.infoq.com/articles/regression-testing-strategies>.
- [9] THE TORTOISESVN TEAM. An Apache™ Subversion (SVN)® client [online]. TortoiseSVN 2019. [cit. 2019-11-30]. online: <<https://tortoisesvn.net/>>.

Requirements Metamodeling, Modeling, Verification and Validation

¹František FERENČÍK, ²Zdeněk HAVLICE

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹frantisek.ferencik@student.tuke.sk, ²zdenek.havlice@tuke.sk

Abstract — The article points out the possibility of facilitating and streamlining processes during the development of a software system, starting with the design, through its implementation to maintenance, including verification and validation of requirements. The article focuses on a specific category of software, which is the area of web applications based on the MVC architecture. In the first stage, a method was developed together with a methodology for modeling critical parts of such systems. This method uses the principles of DSM domain-specific modeling. In the second stage, the CASE tool MVCStudio was developed for this method and was built on the WebGME generic modeling platform. For this purpose, a DSML language of domain-specific modeling was created and implemented in the WebGME environment using metamodeling. In the third stage, several extensions of the CASE tool were implemented to support automation and more efficient development processes.

Keywords — metamodeling, modeling, verification and validation, WebGME, MVC architecture

I. INTRODUCTION

One-dimensional textual representation is often less comprehensible and not as effective as the multidimensional graphical representation used by modeling. The basis of modeling is also the use of abstraction, which in the history has proven to be the key to increasing the efficiency of software project development [1].

Currently, software development using modeling has several problems. With strict methodologies, many so-called one-time "throw away" models are created, which are not used later due to their complexity or are outdated when compared to a live project. Streamlining can be achieved through more efficient use of models, which will be usable multiple times in the software lifecycle and will allow faster detection of problematic parts, ensure better system explanation or generate source code.

In projects using agile approaches, the situation is often such that activities that do not lead directly to the results or partial delivery of the product are postponed. Over time, requests to expand, improve, or repair arise, and the problem may be in the absence of documentation for important parts of the system. To address these issues, a tool that would provide a better view of the system and reveal the context that needs to be explored at length in the source code would be appropriate. This system may include a mechanism that allows the system to be changed and adapted. In addition to increasing productivity, it could also benefit from more efficient learning and minimizing system maintenance and modification costs [2].

Currently, UML is one of the most widely used modeling languages. However, by striving to be a flexible tool for everyone and for everything, it is not always able to solve problems in an effective way and also its versatility does not allow to increase abstraction beyond a certain limit [1]. Developers always strive to improve productivity by increasing abstraction, visualization, and automation. Domain-specific modeling DSM considers these concepts as its basic pillars. The result of using DSM is a higher level of abstraction than UML, because it directly uses DSML language tailored to the application domain. Such a language is very strong but is severely limited to use only for the specific category of systems.

The article points out the possibility of creating an individualized DSM method of modeling specific systems implemented in the CASE tool, which will optimize development processes and solve application domain problems in an efficient way [2].

II. WEB APPLICATION MODELING METHOD

A. Critical parts

In order to create a method for modeling a specific category of systems, it is necessary to identify its critical parts, and these parts must be supported by models. Based on the principles of the MVC design pattern as well as experience with the development of such web applications, we have identified, that critical parts are **data layer**, **GUI graphical user interface**, **control of data flow** and **application state management**.

B. Creating a method

Data layer modeling - It is necessary to model data entities persisted in the database and the relationships between them. The inspiration for the model diagram is the physical ERD entity-relational diagram. The model diagram contains the following elements:

- **Entity** - database table (user, product, order, etc.);
- **Entity attribute** - table column; characterizes the type of information;
- **Index** - a structure that improves the speed of operations
- **Primary key** - uniquely identifies the record in the table;
- **Foreign key** - primary key reference; identifies relationships between entities;
- **Relation** - defines the connection of entities;
- **Cardinality** - number of entity occurrences in this relationship;
- **Relational handling restrictions** - enlargement compared to the ERD; RI protection.

According to the literature [3], the classical ERD diagram has been extended by defining the rules for ensuring RI reference integrity. These rules can be defined for **U-update** and **D-delete** operations. RI security methods are **restrictive**, **null setting** and **cascading**. Sample *ModelDiagram* containing 3 data entities with defined attributes and relationships is displayed In Fig. 1. It is a simple scheme for linking orders and products, which ensures that one order can contain multiple products and at the same time one product can be assigned to multiple orders.



Fig. 1: Sample ModelDiagram of the database layer

Modeling of control - The user switches between different screens on which he performs certain actions. We perceive the set of these screens and transitions as states and transitions between them. The inspiration for the *ControllerDiagram* is the STD diagram of state transitions. The *ControllerDiagram* contains the following components:

- **Action** – state analogy in STD. Processing the URL of the request triggers a specific action, and thus specific state is entered within the application when loading the URL.
- **Start and end terminators** – initial or final state of the application.
- **Controller** – combines several actions, such as displaying the product in the e-shop.
- **Transition** – oriented state connection. It is possible to add a transition invocation condition, for example, after authentication redirection to the list of orders. If the transition does not lead to a specific action but only to the controller, then the action marked as an index is automatically considered to be the destination of the transition.

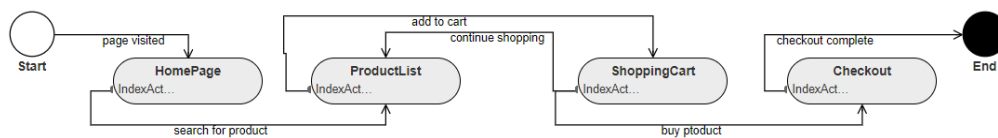


Fig. 2: Sample ControllerDiagram - state model of a simple e-commerce

State model of a simple e-commerce is in Fig. 2. After coming to the website, the user is on the home page, and after using the search, he gets to the list of products, where he can add the product to the cart. He can return to the product list or continue to process the order. After equipping it, he finishes working in the application.

Modeling of system modules - *ControllerDiagram* diagrams can group controllers within larger logical units - modules. Connection of application modules is in Fig. 3. The *Front* module contains the primary e-commerce business logic services and the *Admin* module contains order fulfillment services.

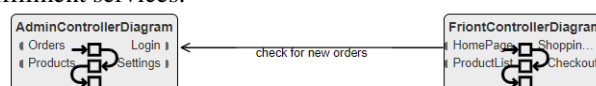


Fig. 3: Composition of modules modeled by ControllerDiagram

User interface modeling - This *ViewDiagram* is not just about the GUI. After processing the request, it is necessary to return data that can be in various formats. The most used is the HTML code that the browser interprets into the GUI. There are other data transfer formats that are used to communicate with other programs. It is, for example, the JSON format. The file can also be the answer to the request. The components of the *ViewDiagram* are:

- **Format** – response type definition - HTML, JSON, File and others.
- **HTML element** – base unit containing information or enabling user input to be recorded. For example, title, form, text input, link, etc.
- **JsonObject** – used for JSON format. It contains data in a structured text form of the key-value type, such as information on the quantity of the product in stock.
- **FileStream** – the answer is a file in a specific format, such as a PDF file.

Model of the product screen based on *ViewDiagram* is in Fig. 4.. The page contains the name, description, product price, available quantity in stock and a form for adding to the basket containing another model of the form containing field for entering the quantity and a button for adding to the basket.



Fig. 4: *ViewDiagram* – product page GUI screen

Data flow modeling - In order for the origin of the processed data to be clear and where the results of operations are stored or displayed, it is necessary to model this process in the *DataFlowDiagram*. It contains following components:

- **Process** – any process that changes data and produces output based on inputs, such as the process of creating an order with sending notification emails.
- **Input / output** – process input and output data, respectively; for example, email for notification.
- **External entity** - third party system, for example warehouse management server.
- **Data store** – files or database entities that persist information.
- **Flow** – direction of information flow. It contains a description characterizing the transmitted information.

The *DataFlowDiagram* (Fig. 5) contains a process that retrieves from the product EAN database and API access data. Subsequently, with this data, the API calls the vendor, which returns the number of pieces in stock. This number is finally assigned to the corresponding element of the GUI template.

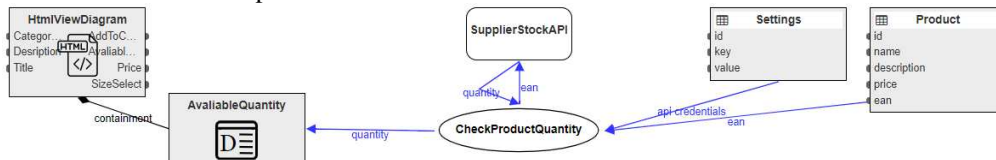


Fig. 5: Sample of *DataFlowDiagram*

C. Development of methodology

There is no general and universal procedure for designing a system and compiling its individual models. However, we recommend finding inspiration in our approach and adapting the individual steps to specific needs according to the current situation:

1. Based on the defined system requirements, it is necessary to design and model the data layer. It is not a problem if certain errors and inconsistencies appear in the original models, because these models can be modified at any time later. It will also help to detect design errors in a timely manner. Thus, even if the structure of the data is not yet completely clear at the beginning, it is necessary to model at least what is known and other elements will be continuously supplemented during the design and compilation of other models.
2. The main system functions are also hidden in the system requirements definitions. These functions need to be identified and divided into interrelated groups. It is a good practice to focus these groups on larger functional units and create modules from them (e.g. administration, user profile, front module, etc.).
3. In this step, for each module, we re-examine the functions of the system and again try to divide them into interrelated smaller groups. We perceive these groups as the states of the application in which the user is currently located (e.g. home screen, product list, shopping cart, etc.) Within these states, certain bases can also be defined, which we

refer to as actions. They are important in supporting the partial functions provided by the given controller (e.g. checking the condition of the product in stock).

4. In the next step, we need to design and model the GUI and possibly other types of views (HTML template, JSON string or file). Each type of view contains its specific components, from which it is necessary to compile this view according to the defined requirements. For example, when modeling a GUI template, components such as a form, various input fields, links, buttons, and many other elements are available.
5. In this last step, we have modeled and have made available everything necessary to create a data flow diagram. It is advisable to have the previous diagrams already created, because it uses them to a large extent. Therefore, we recommend modeling this type of diagram completely at the end. The data stream can connect the data layer with the view layer through processes in the controllers.

III. IMPLEMENTATION OF THE METHOD TO THE CASE TOOL MVCSTUDIO

The chapter deals with the implementation of DSML in the CASE tool MVCStudio [2] on WebGME platform [4] [5].

A. WebGME

WebGME is a generic modeling environment developed at Vanderbilt University. This tool allows creating individualized DSML language. For this purpose, WebGME provides a meta-model based on a UML class diagram. By configuring this meta-model, a new DSML can be defined and WebGME is automatically configured to support it [4][5][6].

B. Basic meta-model

This meta-model is shown in Fig. 6. It contains the FCO object - the first class object forming the root of the inheritance tree, next the *MVCStudioLanguage* object, which combines all objects of the modeling language in the composition for the sake of clarity, and finally, the *Documentation* object, which allows creating documentation for any artifact, which provides a connection to the *FCO* node with 0: N cardinality. The *Requirements* object enables recording text requests just like the *Documentation*. The last object, the *Package* object, makes it possible to better structure the project using packages.

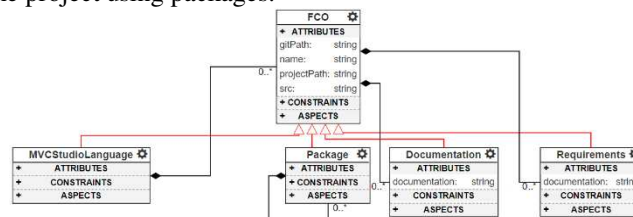


Fig. 6: Basic meta-model

C. Meta-model of the ControllerDiagram

The meta-model for the *ControllerDiagram* in Fig. 7 allows modeling of application states. It contains a *ControllerDiagram* object representing the diagram itself. It is linked to *Package*, which means that a package can contain several of these diagrams. The objects contained in this diagram, which are *Controller*, *Action*, *Start*, *End*, are connected to it. These inherit from the *StateBase* object that is linked to the *Transition* link object. The connection inherits all components, which allows them to be interconnected. There is a link to other diagrams, which enables creating multiple *ViewDiagram* and *DataFlowDiagram* objects in an *Action* object.

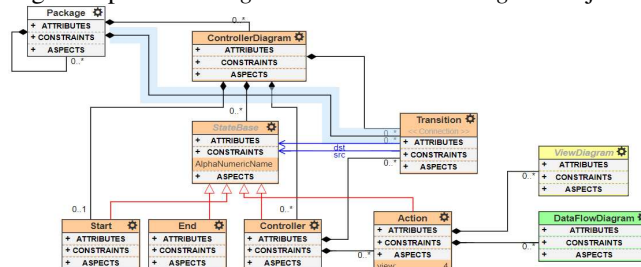


Fig. 7: Meta-model of the ControllerDiagram

D. Meta-model of the ViewDiagram

The meta-model *ViewDiagram* in Fig. 8 enables modeling the view and thus the GUI. It contains an abstract *ViewDiagram* object that combines all the components of the diagram. It inherits all kinds of *View* diagrams from it, such as *FileViewDiagram* which contains *FileStream*, *JsonViewDiagram* containing *JsonObject*, and finally *HtmlViewDiagram* containing *Element* objects. These are further specified for specific HTML components.

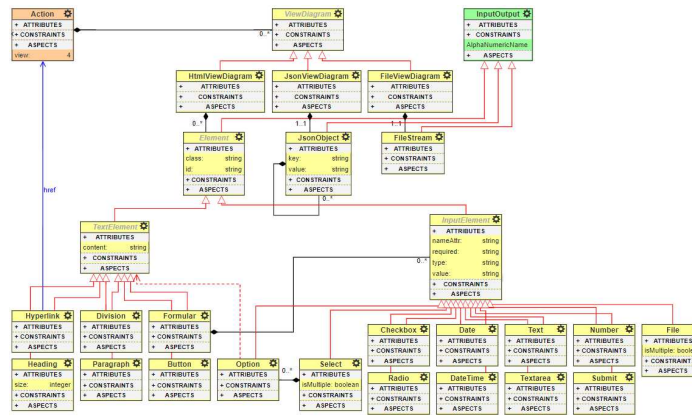


Fig. 8: Meta-model of the ViewDiagram

A *Hyperlink* object has a link to *Action*, which makes it possible to create hyperlinks between the link and the action (*Action* from the *ControllerDiagram*). There is an *Input / Output* object from the *DataFlowDiagram* metamodel from which certain objects are inherited. This allows linking *DataFlowDiagram* to *ViewDiagram*.

E. Metamodel DataFlowDiagram

The *DataFlowDiagram* metamodel in Fig. 9 models the application data flow. It contains components such as *Process*, *Input / Output*, *ExternalEntity*, *DataStore*. These can be linked in the model thanks to the *DataFlow* object, which has pointers linked to them. As for the links, the *DataFlowDiagram* is linked to the *Action*, which enables linking the *DataFlowDiagram* to the *ControllerDiagram*. Furthermore, the *Entity* and *Attribute* objects inherit from the *DataStore* object, which allows the *DataFlowDiagram* to be linked to the *ModelDiagram*. Finally, there are the yellow objects from the *ViewDiagram* inheriting from the *Input / Output* object, which ensures the connection of the *DataFlowDiagram* with the *ViewDiagram*.

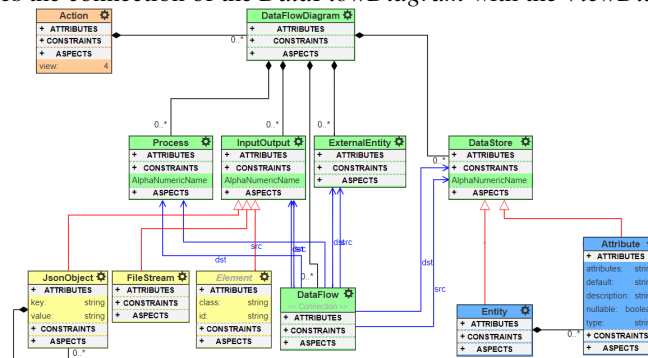


Fig. 9: Meta-model of the DataFlowDiagram

F. Meta-model of the ModelDiagram

The *ModelDiagram* meta-model in Fig. 10 represents the data layer of the application. It contains a *ModelDiagram* object linked to a *Package* object, so a project package can contain this diagram. It contains *Entity* and *Attribute* objects. The *Index* object is linked to the *Attribute*, which enables defining indexes for a specific database column. The *Attribute* object is linked to the *Relation* object, which enables creating links in the form of foreign keys. There is also a *DataStore* object from the *DataFlowDiagram*, which inherits Entities and Attribute from it, which allows *DataFlowDiagram* to link processes with database entities as well as its attributes.

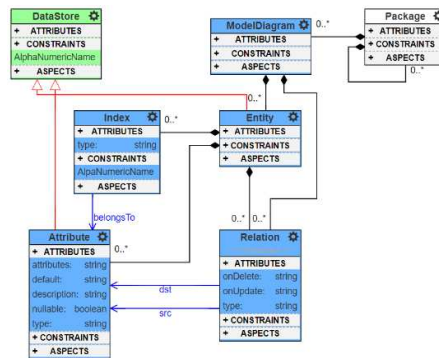


Fig. 10: Meta-model of the ModelDiagram

IV. MVCSTUDIO AS AN EXTENSION OF CASE TOOL

Add-ons have been implemented in the CASE tool to improve, facilitate or automate the work of developers on the proposed system, some of which are described below.

Connecting the model to the software – Individual artifacts in CASE system models can be represented in the target system by specific source codes. The purpose of this extension is to quickly and easily get from the artifact to its implementation in the form of source code of the target system. This extension is of particular benefit to system developers in that it saves them time in finding an implementation of this model in source code, which can usually be very large. As you can see in Fig. 11, models contain buttons that, when clicked, open a representation of the artifact in the source code in *gitlab* or the developer IDE.

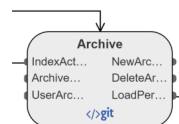


Fig. 11: Archive controller with buttons for one click source code access

To implement IDE integration, individualized protocol *editor://* was created which transmits the identification data of the source code from the web browser to the guest operating system. This protocol looks like this: *editor://open/?file=APP_ROOTapp/Template.php&line=1*

When the button is pressed, this protocol is invoked, the guest operating system intercepts it and moves it to a predefined program for further processing. Based on the user configuration, this program selects the preferred development environment and runs it together with a command to open the source code based on the identifiers contained in the received protocol.

Generating models from text analysis of requirements – allows, based on word-class analysis, to generate an initial design of model artifacts from a list of text requirements.

As can be seen in Fig. 12, the Requirements object contains a change request in text form. Specifically, it is a requirement that the user be able to keep his Xcontest [7] external service identifiers. There are requirements after running the *ArtifactGenerator* plugin analyzed and new objects of this model are created based on certain rules. In this case, these nodes are placed below the requirements. This is *xcontest_id*, *model*, *xcontest_login* and *slug*. The developer will now decide what needs to be done next with these objects. Object model in this makes no sense in the context, so the developer will remove it.

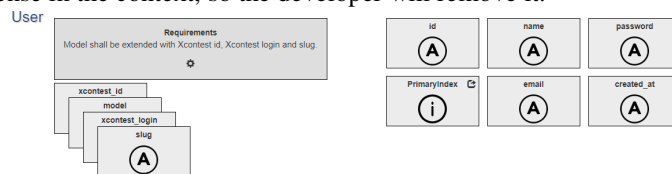


Fig. 12: Generated attributes for User entity from text requirements

Generating source code from models – allows generating, based on the created models, the source code of the skeleton of critical parts of the system. For this purpose, another 2 extensions of the WebGME environment were implemented. **ArtifactExporter** - analyzes all system models and exports important information not only for the purpose of generating source code in the form of a JSON string. Upon successful execution, JSON file is generated in the result report. This extension can also be initialized by another extension, taking over its output and working with it further. **SourceCodeGenerator** - responsible for the generation of source code. Before running it, it is possible to set the source code generation method and turn on the generation of database migration files. Once launched, the *ArtifactExporter* extension is initialized from which it takes a schema of the designed artifacts and generates controllers and

models from it, or database migration files, if required. Source codes will be placed on the basis of defined namespaces within a specific model in the development version of the software system project, which is defined in the *ROOT* node of the project model.

V. CONCLUSION

This article dealt with the possibility of facilitating and streamlining engineering work throughout the entire life cycle of a software system from its initial design through implementation to maintenance, including verification and validation of requirements for this system with use of models. It was proposed methodology for modeling critical parts of systems with MVC architecture. This methodology involves domain specific modeling language (DSML) and CASE tool developed with use of the generic WebGME modeling platform. DSML uses diagrams for data layer modeling, modeling of control switching between states of the system, modeling of system modules, user interface modeling and data flow modeling. The methodology and CASE tool were used for changes in GUI of real application igc2pdf [8] intended to generate flight documentation.

REFERENCES

- [1] STEVEN, K., Improving Developer Productivity With Domain-Specific Modeling Languages, July 2005, [cit: 2020-04-10], available on the Internet: <https://www.developerdotstar.com/mag/articles/domain_modeling_language.html>
- [2] FERENČÍK, F., Requirements Metamodeling, Modeling, Verification and Validation, diploma thesis, Technical University of Košice 2020.
- [3] HAVLICE, Z., Modeling and prototyping in the design of information systems, 1998, ISBN 80-88786-95-9.
- [4] MAROTI, M., KECSKES, T., KERESKENYI, R., VOLGYESI, P., LEDECZI, A., Online Collaborative Environment for Designing Complex Computational Systems, ISIS, Vanderbilt University, Nashville, TN, USA., 2014, [cited: 2019-03-22], available on the Internet: <https://webgme.org/WebGMEWhitePaper.pdf>
- [5] ISIS/VANDERBILT UNIVERSITY. WebGME Documentation, Release 1.0.0, Marec 2019, [cit, 2019-03-19], available on the Internet: <https://media.readthedocs.org/pdf/webgme/latest/webgme.pdf>.
- [6] JOHN P. VAN GIGCH, System Design Modeling and Metamodeling, Springer Science & Business Media, 2013, ISBN: 1489906762.
- [7] World XContest. [cited: 2020-05-20], available on the Internet: <https://www.xcontest.org/world/en/>
- [8] FERENČÍK F., Processing of flight records in IGC format into text documents. Bachelor thesis. [Online] Technical University of Košice, 25.05.2018. [Cit. 2019-12-03]. Available on the Internet: <https://opac.crzp.sk/?fn=detailBiblioForm&sid=5301BA0C1F12FE31B5404AA915FD>

Test tepelne namáhaného progresívneho kvapalného dielektrika v časovej doméne

¹Peter Havran, ²Roman Cimbala

¹ Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

² Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹peter.havran@tuke.sk, ²roman.cimbala@tuke.sk

Abstrakt — Nosným faktorom tejto publikácie je vyšetrovanie prúdových charakteristík teplom degradovaného progresívneho kvapalného dielektrika pomocou impedančnej spektroskopie. Publikácia obsahuje teoretickú a praktickú časť. Teoretická časť sa skladá z problematiky impedančnej spektroskopie v časovej oblasti. Praktická časť rozoberá popis experimentálneho testu, jeho analýzu a grafické výstupy nabíjacieho prúdu na čase.

Kľúčové slová — časová oblasť, impedančná spektroskopia, nabíjací prúd, teplota

I. ÚVOD

K dispozícii je niekoľko rôznych metód, ako je analýza oleja alebo rôzne elektrické merania na vykonanie hodnotenia stavu olejových transformátorov. Skúmanie dielektrických parametrov olejovej izolácie je dôležitou súčasťou existujúcich metód. Meranie dielektrickej odozvy sa môže uskutočňovať v časovej alebo frekvenčnej oblasti. Experimentálne výsledky a modelovanie dielektrickej odozvy v časovej oblasti pre diagnostiku transformátora známe z literatúry znázorňujú vysokú citlivosť výsledkov na zmenu základných materiálových vlastností, ako je vodivosť. Ako je známe, vlhkosť a kontaminácia majú tendenciu zvyšovať vodivosť tuhého a tekutého dielektrika v transformátore. Použitie metód merania v časovej oblasti vykazuje výhody najmä pri posudzovaní stavu zložitých konštrukcií transformátorov. Výsledky získané z polarizačného prúdu umožňujú odhadnúť niekoľko parametrov, ako je izolačný odpor, časové konštanty a vodivosť dielektrických komponentov použitých v izolácii [1].

Meraný a skúmaný bol počas tohto testu progresívny izolačný olej Shell DIALA S4 ZX-1, vyrábaný na báze technológie GTL, pomocou nedeštruktívnej metódy impedančnej resp. dielektrickej spektroskopie.

Táto nedeštruktívna metóda obsadzuje v oblasti diagnostiky elektroizolačných materiálov významnú pozíciu medzi viacerými novodobými metódami. Táto významná pozícia súvisí s meraním malého signálu elektrickej odozvy (dipólové otáčanie) predmetného materiálu a následnú analýzu odozvy pre získanie informácií o stave izolačného systému [2].

II. IMPEDANČNÁ SPEKTROSKOPIA V ČASOVEJ OBLASTI

Významom impedančnej spektroskopie je schopnosť odlíšiť dielektrické a elektrické vlastnosti jednotlivých príspevkov skúmaných materiálov. Touto metódou je možné skúmať dielektrické relaxačné procesy a rôzne deje prebiehajúce v dielektriku, preto pojem impedančná spektroskopia možno bezproblémovo nahradiť pojmom dielektrická spektroskopia [3].

Meranie polarizačných a depolarizačných prúdov (PDC) po aplikácii kroku jednosmerného napätia je jedným zo spôsobov skúmania pomalých polarizačných procesov v časovej oblasti. Na izolačný systém sa aplikuje jednosmerné nabíjacie napätie $U(t)$ s nasledujúcimi charakteristikami:

$$U(t) = \begin{cases} 0 & t \leq 0 \\ U_c & 0 \leq t \leq t_c \\ 0 & t \geq t_c \end{cases}$$

Na vyhodnotenie medzivrstvovej polarizácie a stavu vzorky je potrebný dlhý čas nabíjania (napr. 10000 s). Počas nabíjania sa meria polarizačný (alebo absorpčný) prúd $I_{pol}(t)$ cez testovaný objekt. $I_{pol}(t)$ možno vyjadriť ako:

$$I_{pol}(t) = C_0 \cdot U_c \left[\frac{\sigma_0}{\epsilon_0} + \epsilon_\infty \cdot \delta(t) + f(t) \right] \quad (1)$$

kde C_0 je geometrická kapacita, U_c je krokové napätie (nabíjacie napätie), σ_0 je jednosmemá vodivosť dielektrického materiálu, $\epsilon_0 = 8,852 \cdot 10^{-12} \text{ F} \cdot \text{m}^{-1}$ je permitivita vákua, ϵ_∞ predstavuje vysokú frekvenčnú zložku permitivity, $\delta(t)$ je funkcia delta vyplývajúca z privedeného krokového napätia pri $t = 0$ a $f(t)$ je funkcia odozvy dielektrického materiálu [4].

Po polarizačnej perióde sa skúšobná vzorka skratuje opätovným privedením napätia na $t = t_c$, čo umožňuje meranie depolarizačného (alebo resorpčného) prúdu $I_{depol}(t)$ v opačnom smere. $I_{depol}(t)$ má nulovú jednosmernú zložku vodivosti [4].

Podľa princípu superpozície sa náhle zníženie napätia U_c na nulu môže považovať za použitie kroku záporného napätia v čase $t = t_c$. Zanedbaním funkcie delta v (1) dostaneme pre $t > t_c$:

$$I_{depol}(t) = -C_0 \cdot U_c [f(t - t_c) - f(t)] \quad (2)$$

Obr. 1 zobrazuje meracie obvody PDC s typickou zmenou I_{pol} a I_{depol} po aplikácii a odstránení krokového nabíjacieho napätia U_c . Z meracích prúdov PDC je možné odhadnúť jednosmemú vodivosť σ_0 . Ak je testovaný objekt nabitý dostatočne dlho, potom $f(t_c) = 0$ a funkcia dielektrickej odozvy $f(t)$ bude úmerná depolarizačnému prúdu. Prepísanie (1) ako:

$$f(t_1) = \frac{I_{pol}(t_1)}{C_0 \cdot U_c} - \frac{\sigma_0}{\epsilon_0} \quad (3)$$

a (2) ako:

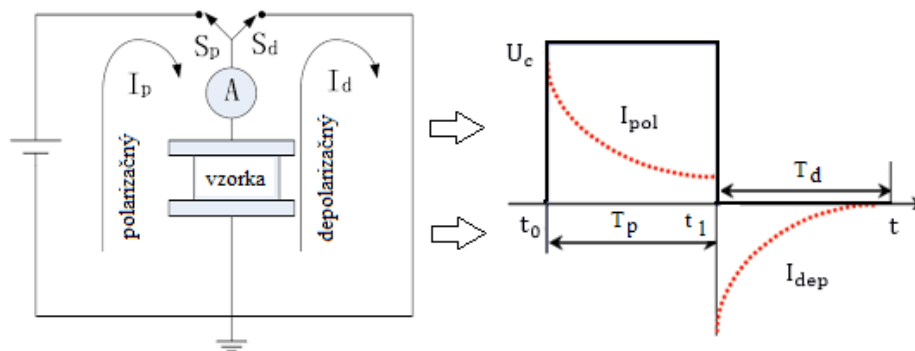
$$f(t - t_c) \approx \frac{-I_{depol}(t - t_c)}{C_0 \cdot U_c} \quad (4)$$

dostaneme:

$$\sigma_0 \approx \frac{\epsilon_0}{C_0 \cdot U_c} (I_{pol}(t_1) + I_{depol}(t - t_c)) \quad (5)$$

pre všetky t a t_1 také, že t_1 (počas nabíjania) = $t - t_c$ (počas vybíjania).

Aj bez priameho merania vodivosti na vzorke oleja sa dá vodivosť oleja získať použitím (5). Rovnakým spôsobom sa dá vodivosť papiera odhadnúť z dlhodobých hodnôt polarizačných a depolarizačných prúdov [4].



Obr. 1 Prúdová odozva na aplikovaný napätový impulz [4]

III. EXPERIMENTÁLNY TEST

Cieľom tohto testu bolo sledovanie a porovnanie dielektrických vlastností izolačného oleja prostredníctvom merania nabíjacích prúdov charakterizujúcich elektrofyzikálnu štruktúru materiálu vplyvom urýchleného tepelného starnutia. Test pozostával zo siedmich meraní s nasledovnou dobou tepelného starnutia izolačného oleja:

- 1. meranie - po 250 hodinách starnutia,
- 2. meranie - po 430 hodinách starnutia,
- 3. meranie - po 600 hodinách starnutia,
- 4. meranie - po 814 hodinách starnutia,
- 5. meranie - po 1080 hodinách starnutia,
- 6. meranie - po 1580 hodinách starnutia,
- 7. meranie - po 1830 hodinách starnutia.

Všetky merania boli uskutočňované v časovej oblasti impedančnej spektroskopie, pri ktorých boli skúmané a porovnávané priebehy nabíjacích prúdov I_{nab} pri rôznych dobách tepelného

stamutia aplikovaním jednosmerného napätia na vzorku pomocou IRC (Isothermal Relaxation Current) analýzy.

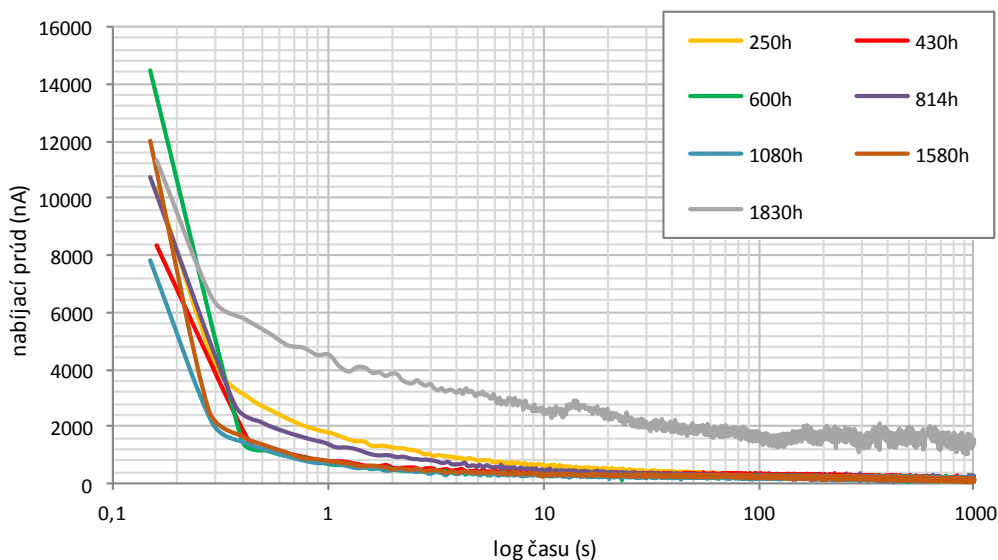
Test bol realizovaný na novej nekontaminovanej vzorke izolačného oleja Shell DIALA S4 ZX-1. Tento progresívny elektroizolačný olej predstavuje budúcnosť pre výkonové transformátory a svojim návrhom ponúka predĺženú životnosť s dôrazom na nulový obsah síry pomocou výrobných technológií GTL, ktorá premieňa zemný plyn na kvapalné uhľovodíky čoho výsledkom je olej.

K realizácii urýchleného tepelného starnutia izolačného oleja bola použitá univerzálna rúra UF55^{plus}. Nádobka so vzorkou izolačného oleja bola vložená do rúry, kde prebiehalo starnutie pri nastavenej teplote 90 °C. Ako je popísané vyššie, pred každým zo siedmich meraní bol olej počas určitej doby (250 h, 430 h, 600 h, 814 h, 1080 h, 1580 h a 1830 h) namáhaný teplotou 90 °C. Meranie pred začiatkom tepelného starnutia (0 h) nebolo realizované z dôvodu vysokej vlhkosti oleja, keďže nie je známe, kde bol olej skladovaný. Pred každým meraním vzorka izolačného oleja chladla približne jeden deň, aby bolo možné uskutočniť meranie. Merania boli uskutočňované pomocou meracieho prístroja KEITHLEY 6517B a riadené prostredníctvom softvéru Agilent VEE Pro, ktorý zaznamenával namerané hodnoty nabíjacieho prúdu.

IV. ANALÝZA EXPERIMENTÁLNEHO TESTU

Analýza nameraných údajov v časovej oblasti zahŕňa správanie sa nabíjacích prúdov I_{nab} v časovom pásme 1000 s meracieho prístroja KEITHLEY 6517B a porovnanie prúdových charakteristík pri rôznej dobe tepelného starnutia (250 h, 430 h, 600 h, 814 h, 1080 h, 1580 h a 1830 h).

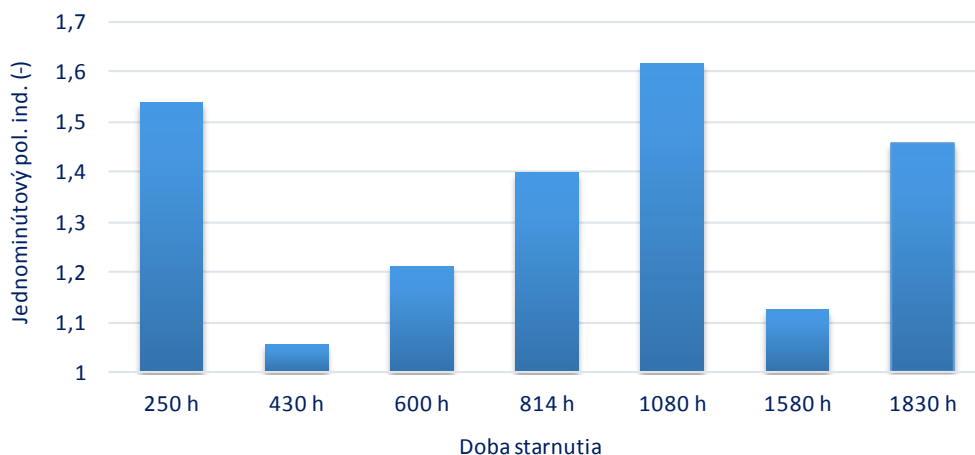
Ako je známe z teoretických poznatkov, priebeh nabíjacieho prúdu klesá exponenciálne s časom nabíjania. Nie je tomu inak ani v nameraných charakteristikách nabíjacieho prúdu, kedy priebehy prúdov exponenciálne klesali v čase pri každom meraní po určitej dobe tepelnej degradácie. Na Obr. 2 je graficky znázornené porovnanie nabíjacích prúdov pri rôznej dobe tepelného starnutia. Ako už je známe, priebeh nabíjacieho prúdu je zložený z kapacitného, absorpčného a vodivostného prúdu. Z dynamického správania sa nabíjacieho prúdu v čase 0,15 s (kapacitný prúd), nie je možné toto správanie bližšie charakterizovať, keďže ide o skokovitú zmenu hodnôt nabíjacích prúdov pri rôznej dobe tepelného starnutia.



Obr. 2 Porovnanie hodnôt I_{nab} (nA) pri rôznej dobe tepelného starnutia

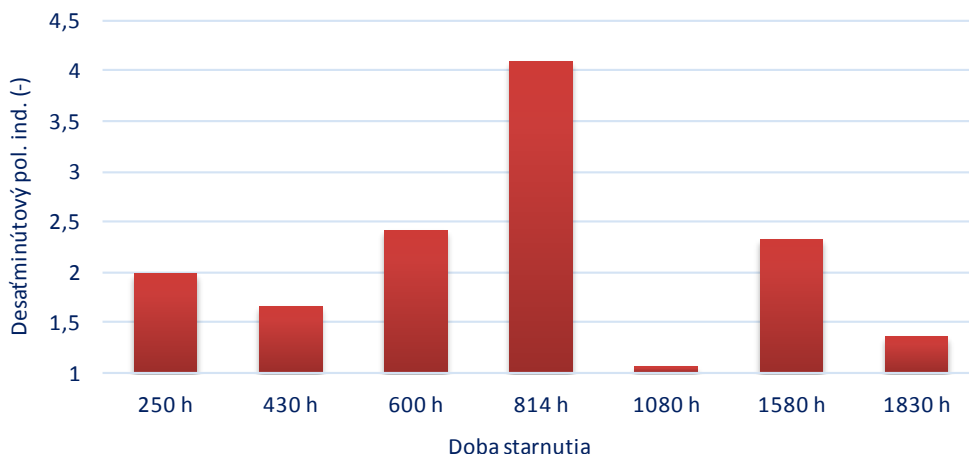
Na lepšiu charakterizáciu správania sa absorpčného prúdu boli vypočítané polarizačné indexy jednominútové a desaťminútové pre všetky doby tepelného starnutia, ako je to popísané v Tab. 1. Pri získaní hodnôt jednominútového polarizačného indexu bola hodnota prúdu v čase 15 s podelená hodnotou prúdu v čase 60 s. Podobne pri získaní hodnôt desaťminútového polarizačného indexu bola hodnota prúdu v čase 60 s podelená hodnotou prúdu v čase 600 s. Grafické porovnanie jednominútového a desaťminútového polarizačného indexu pri rôznej dobe tepelného starnutia je znázornené na Obr. 3 a Obr. 4.

1-minútový polarizačný index



Obr. 3 Jednominútový polarizačný index pri rôznej dobe tepelného starnutia

10-minútový polarizačný index



Obr. 4 Desaťminútový polarizačný index pri rôznej dobe tepelného starnutia

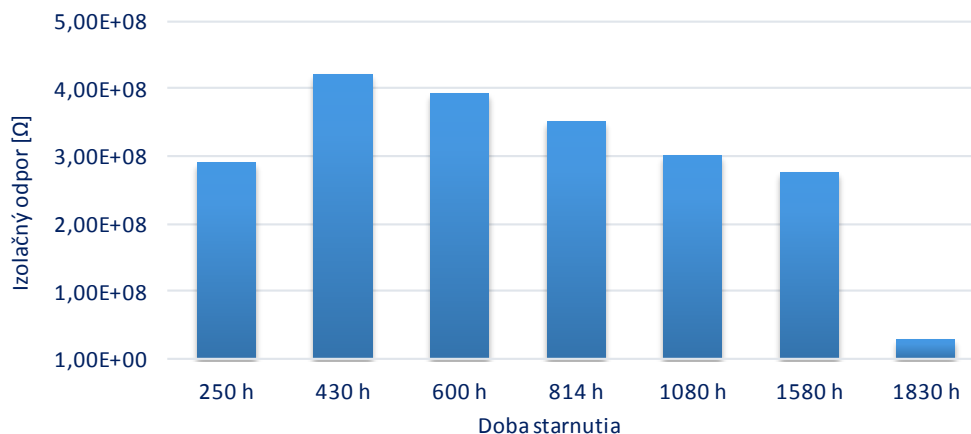
Z poznatkov z praxe je hodnota polarizačného indexu izolačného systému nových transformátorov pred uvedením do prevádzky minimálne 1,3 a obvykle je to v okolí hodnôt 1,7 až 1,8. Pri jednominútovom polarizačnom indexe boli hodnoty väčšie ako 1,3 pri dobe starnutia 250 h, 814 h, 1080 h a 1830 h. Pri desaťminútovom polarizačnom indexe boli všetky hodnoty väčšie ako 1,3 okrem hodnoty pri tepelnom starnutí 1080 h. Na základe desaťminútového polarizačného indexu je možné konštatovať, že napriek tepelnej degradácii izolačného oleja 1830 h vyhovuje izolačný olej prevádzkovým požiadavkám z tohto hľadiska.

Z odčítaných hodnôt nabíjajúcich prúdov v čase 1000 s (vodivostné prúdy) bolo zistené, že zvyšovaním tepelnej degradácie izolačného oleja sa zvyšovali aj hodnoty vodivostných prúdov tak, ako je to známe z teoretických poznatkov. Hodnota vodivostného prúdu pri dobe tepelného starnutia 250 h je trochu vyššia v porovnaní so starnutím 430 h, 600 h a 814 h. Môže to byť spôsobené vyššou vlhkosťou oleja na začiatku tepelného starnutia. Pri dobe tepelného starnutia 430 h bol zaznamenaný pokles hodnoty oproti dobe 250 h, kedy vlhkosť bola eliminovaná zvýšením doby tepelného starnutia v rúre. Hodnoty vodivostných prúdov s následným prepočtom na hodnoty izolačného odporu pri rôznej dobe tepelného starnutia sú porovnané v Tab. 1.

Tabuľka 1
 Namerané hodnoty I_v a vypočítané hodnoty p_1 , p_{10} a R_{iz}

Doba starnutia	Polarizačné indexy		Vodivostné prúdy I_v (nA)	R_{iz} (G Ω)
250 h	1 min	1,538	138	0,2899
	10 min	1,978		
430 h	1 min	1,055	95	0,4211
	10 min	1,654		
600 h	1 min	1,210	102	0,3922
	10 min	2,405		
814 h	1 min	1,399	114	0,3509
	10 min	4,086		
1080 h	1 min	1,616	133	0,3008
	10 min	1,054		
1580 h	1 min	1,124	145	0,2759
	10 min	2,320		
1830 h	1 min	1,457	1410	0,0284
	10 min	1,353		

Izolačný odpor v čase 1000 s



Obr. 5 Izolačný odpor v čase 1000 s pri rôznej dobe tepelného starnutia

Obr. 5 znázorňuje prepočet hodnôt vodivostných prúdov na hodnoty izolačného odporu v čase 1000 s, kde je jasne badateľné znižovanie izolačného odporu vplyvom rastúceho tepelného namáhania.

V. ZÁVER

Namerané hodnoty nabíjajúcich prúdov priniesli zhodu s teoretickými poznatkami o tom, že priebeh nabíjacieho prúdu klesá exponenciálne s časom. Takéto exponenciálne klesanie bolo zaznamenané pri každom meraní (po určitej dobe tepelného starnutia). Z dynamického správania sa nabíjacieho prúdu v čase 0,15 s (kapacitný prúd) v dôsledku nameraných hodnôt nebolo možné toto správanie bližšie charakterizovať, keďže išlo o skokovitú zmenu hodnôt. Na základe polarizačných indexov získaných z absorpčného prúdu je možné konštatovať, že napriek tepelnej degradácii izolačného oleja 1830 h vyhovuje izolačný olej prevádzkovým požiadavkám z tohto hľadiska. Pri odčítaní hodnôt nabíjajúcich prúdov v čase 1000 s (vodivostné prúdy) bolo zistené, že zvyšovaním tepelnej degradácie izolačného oleja sa zvyšovali aj hodnoty vodivostných prúdov (resp. znižovali hodnoty izolačného odporu), čím tento fakt korešponduje s teoretickými poznatkami.

POĎAKOVANIE

Túto prácu podporili štrukturálne fondy EÚ v rámci projektov ITMS 313011T565 a Ministerstvo školstva pre VEGA 1/0340/18, 2/0141/16 a APVV 15-0438, 17-0372, 18-0160.

LITERATÚRA

- [1] O. Kuzmin, D. Sautter, S. Thummalacharla, "Experience with transformer diagnostics based on dielectric response measurements in time domain", PRZEGLAD ELEKTROTECHNICZNY, ISSN 0033-2097, R. 94 NR 10/2018.
- [2] Y. Feldman, "The Physics Of Dielectrics 83887. Lecture 1", Department of Applied Physics, The Hebrew University of Jerusalem, Online: <http://aph.huji.ac.il/courses/2008_9/83887/index.html>.
- [3] S. W. Zaengl, "Dielectric spectroscopy in time and frequency domain for HV power equipment", Zurich: Swiss federal institute of technology, 2001.
- [4] A. Bouaicha, I. Fofana, M. Farzaneh et al., "Dielectric Spectroscopy Techniques as Quality Control Tool: A Feasibility Study", IEEE Electrical Insulation Magazine, Vol. 25, No. 1, January 2009.

Dielektrická spektroskopia online tepelne namáhaného kvapalného dielektrika

¹Peter Havran, ²Roman Cimbala

¹ Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

² Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹peter.havran@tuke.sk, ²roman.cimbala@tuke.sk

Abstrakt — Tento článok rieši porovnanie dielektrických parametrov dvoch teplotných testov s rôznou dobou zohriatia kvapalného dielektrika využitím metódy dielektrickej spektroskopie. Teoretická časť opisuje metódu dielektrickej spektroskopie. Praktická časť sa zameriava na opis experimentu a analýzu nameraných údajov.

Kľúčové slová — čas, dielektrická spektroskopia, frekvencia, teplota

I. ÚVOD

Izolácia elektrického zariadenia je počas svojej prevádzkovej životnosti vystavená rôznym napätiam (ako je elektrické, mechanické, tepelné). V priebehu času tieto napätia znižujú dielektrickú pevnosť a ďalšie požadované izolačné vlastnosti. Vlhkosť, ktorá sa vytvára ako vedľajší produkt starnutia, ďalej urýchľuje degradáciu izolačného systému. Preto je potrebné študovať dielektrické vlastnosti izolácie, aby sa zabezpečila bezporuchová prevádzka elektrických zariadení. V nedávnej dobe sa medzi nástrojmi na hodnotenie stavu izolačného systému stali populárne merania a analýzy dielektrických spektroskopických údajov v časovej a frekvenčnej oblasti. V časovej oblasti sa vykonáva meranie polarizačného a depolarizačného prúdu (PDC) alebo IRC (Isothermal Relaxation Current) analýza a meranie zotaveného napätia (RV). Zatiaľ čo vo frekvenčnej oblasti rôzne parametre, ako je komplexná kapacita, stratový činiteľ $\tan \delta$, sa komplexná permitivita počíta ako funkcia frekvencie. Meranie vo frekvenčnej oblasti (FDS) je uprednostňované pred ostatnými dvoma metódami, pretože je spoľahlivejšie a efektívnejšie [1].

II. DIELEKTRICKÁ SPEKTROSKOPIA

A. Dielektrická spektroskopia vo frekvenčnej oblasti

Analytický prechod z časovej na frekvenčnú oblasť sa môže uskutočniť pomocou Laplaceovej alebo Fourierovej transformácie. Predpokladá sa ideálna kroková odozva pre celkovú prúdovú hustotu funkcie dielektrickej odozvy $f(t)$, keď do úvahy aj procesy okamžitej polarizácie:

$$j(t) = \sigma_0 E(t) + \epsilon_0 \frac{dE(t)}{dt} + \epsilon_0 \frac{d}{dt} \int_0^t f(t-\tau) E(\tau) d\tau \quad (1)$$

kde platí

$$j(t) \Rightarrow j(p); E(t) \Rightarrow E(p); E'(t) \Rightarrow pE(p); f(t) \Rightarrow F(p);$$

a vzhľadom na konvolúciu poslednej podmienky v tejto rovnici dostaneme, pričom p je Laplaceova operácia:

$$j(p) = \sigma_0 E(p) + \epsilon_0 pE(p) + \epsilon_0 pF(p)E(p) \quad (2)$$

Pretože p je komplexná frekvencia $i\omega$, môžeme rovnicu zredukovať na

$$j(\omega) = E(\omega) [\sigma_0 + i\omega\epsilon_0 (1 + F(\omega))] \quad (3)$$

Takže $F(\omega)$ je Fourierova transformácia funkcie dielektrickej odozvy $f(t)$ alebo komplexnej susceptibility:

$$\chi(\omega) = F(\omega) = \chi'(\omega) - i\chi''(\omega) = \int_0^{\infty} f(t) \exp(-i\omega t) dt \quad (4)$$

Poznamenávame, že frekvenčná stupnica je teraz $0 \leq \omega \leq \infty$. Kombinácia (3) a (4) ukazuje celkovú prúdovú hustotu:

$$j(\omega) = \{\sigma_0 + \varepsilon_0 \omega \chi''(\omega) + i\omega \varepsilon_0 [1 + \chi'(\omega)]\} E(\omega) \quad (5)$$

Hlavná časť tohto prúdu má svoj pôvod v komplexnej elektrickej indukčii $D(\omega)$, ktorý je úmerný komplexnej dielektrickej permitivite $\mathcal{E}(\omega)$, so vzťahom:

$$D(\omega) = \varepsilon_0 \mathcal{E}(\omega) E(\omega) \quad (6)$$

kde:

$$\mathcal{E}(\omega) = \varepsilon'(\omega) - i\varepsilon''(\omega) = (1 + \chi'(\omega)) - i\chi''(\omega) \quad (7)$$

Skutočné merania tejto dielektrickej odozvy vo frekvenčnej oblasti je ťažké vykonať, ak sa frekvenčný rozsah stane veľmi veľkým. Zvyčajne a prinajmenšom v elektroenergetike sa vykonáva iba jedno meranie „C - tan δ “, t. j. pri výkonovej frekvencii. Sofistikované laboratórne prístroje však dokážu pokryť aj frekvenciu mnohých desiatok. Všimnite si, že podľa (5) takéto prístroje nemôžu rozlišovať medzi súčasným prínosom „čistej“ jednosmernej vodivosti σ_0 a dielektrickou stratou $\chi''(\omega)$. To znamená, že nameraná relatívna dielektrická permitivita $\widetilde{\varepsilon}_r(\omega)$ sa líši od relatívnej permitivity $\mathcal{E}(\omega)$ definovanej v (6) a (7). Potom sa zmeria relatívna dielektrická permitivita $\widetilde{\varepsilon}_r(\omega)$ z tohto vzťahu:

$$j(\omega) = i\omega \varepsilon_0 \widetilde{\varepsilon}_r(\omega) E(\omega) \quad (8)$$

Teda:

$$\widetilde{\varepsilon}_r(\omega) = \varepsilon'_r(\omega) - i\varepsilon''_r(\omega) = \varepsilon'(\omega) - i[\varepsilon''(\omega) + \sigma_0/\varepsilon_0\omega] = 1 + \chi'(\omega) - i[\chi''(\omega) + \sigma_0/\varepsilon_0\omega] \quad (9)$$

a stratový činiteľ $\tan \delta(\omega)$,

$$\tan \delta(\omega) = \frac{\varepsilon''_r(\omega)}{\varepsilon'_r(\omega)} = \frac{\varepsilon''(\omega) + \sigma_0/\varepsilon_0\omega}{\varepsilon'(\omega)} \quad (10)$$

Reálna časť (9) predstavuje kapacitu testovaného objektu, zatiaľ čo imaginárna časť predstavuje straty. Obidve veličiny závisia od frekvencie. Táto skutočnosť sa často neuznáva, ak sa meranie „C - tan δ “ vykonáva iba pri jednej frekvencii. Keďže účinky stamutia zmenia tieto veličiny v úplne odlišných a špecifických frekvenčných rozsahoch, nové diagnostické nástroje budú tento účinok monitorovať a zisťovať [2][3].

Nakoniec treba poznamenať, že všetky dielektrické veličiny sú viac-menej závislé od teploty. Pri každom porovnávaní alebo meraní týchto veličín sa to musí zohľadniť [3].

B. Dielektrická spektroskopia v časovej oblasti

Princípom metódy dielektrickej spektroskopie je nepriame pozorovanie polarizácie polarizovaných alebo polarizovateľných materiálov prostredníctvom určitej odozvy, ktorá bola spôsobená zmenou aplikovaného vonkajšieho elektrického poľa. Časová oblasť je zameraná na sledovanie polarizačného (nabíjacieho) a depolarizačného (vybíjacieho) prúdu prechádzajúceho dielektrikom, keď sa aplikuje jednosmerné elektrické pole, inými slovami, časový priebeh dielektrického prúdu sa zaznamenáva pri zmene krokového napätia [4].

Veľkosť dielektrickej polarizácie počas nabíjania (alebo vybíjania) a jej časová závislosť je opísaná funkciou odozvy ϕ_t :

$$P(t) = \varepsilon_0(\varepsilon_{\infty} - 1)E + \varepsilon_0(\varepsilon_s - \varepsilon_{\infty})E\phi(t) \quad (11)$$

kde $\varepsilon_0 = 8,854 \cdot 10^{-12} \text{ F}\cdot\text{m}^{-1}$ je permitivita vákua, ε_{∞} predstavuje optickú permitivitu (relatívna permitivita pri frekvencii $f(\varepsilon_{\infty}) \rightarrow 10^{14} \text{ Hz}$), ε_s určuje statickú permitivitu, E predstavuje intenzitu elektrického poľa a t je čas. Nasledujúci text opisuje vzťah medzi časovou závislosťou nabíjajúcich a vybíjajúcich prúdov a časovou závislosťou polarizácie. Nabíjacie a vybíjacie prúdy v dielektriku majú v priebehu času monotónne rastúce a klesajúce krivky. Tieto udalosti sú opísané v prípade vákuového kondenzátora exponenciálnymi funkciami času. Pomocou funkcie poklesu $\varphi(t)$ časovej závislosti vybitia (depolarizácie) prúdu v dielektriku resp. jeho súčasná hustota j_{depol} je opísaná:

$$j_{depol} = \varepsilon_0(\varepsilon_s - \varepsilon_{\infty})E\varphi(t) \quad (12)$$

Nabíjací (polarizačný) prúd, resp. jeho prúdová hustota, je opísaný súčtom dvoch zložiek:

$$j_{pol}(t) = \frac{dP}{dt} + \gamma E \quad (13)$$

kde γE je zložka prúdovej hustoty spôsobenej vodivosťou v dielektriku (kde γ je špecifická elektrická vodivosť dielektrika po aplikácii jednosmerného elektrického poľa a následnom rozptyle dielektrického náboja). Prúdová hustota vybíjacieho (depolarizačného) prúdu neobsahuje vodivú zložku prúdu, a preto sa rovná:

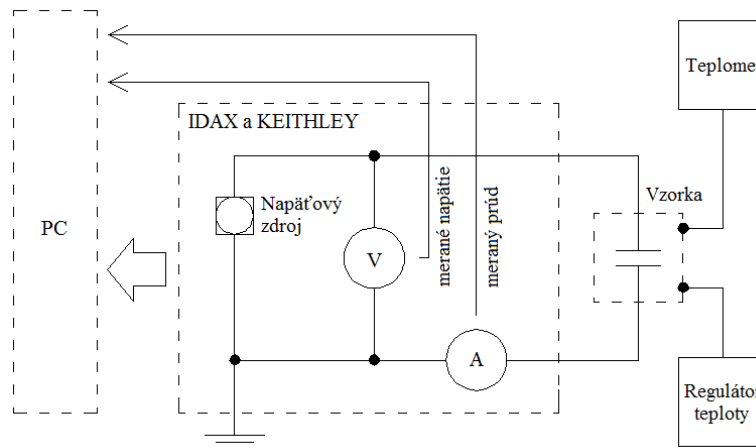
$$j_{depol}(t) = \frac{dP}{dt} \quad (14)$$

Na základe štúdia dielektrických spektier je možné identifikovať dielektrické relaxačné mechanizmy, ktoré sú spojené s mechanizmami polarizácie a úplnou vnútornou štruktúrou dielektrika. Prítomnosť rôznych typov elektrických dipólov, iónov alebo elektrónov schopných vykonávať rôzne pohyby v dôsledku vplyvu vonkajšieho elektrického poľa je v týchto materiáloch rozpoznateľná, čo vedie k väčšiemu počtu relaxačných mechanizmov [4].

III. EXPERIMENT

Hlavnou motiváciou tohto experimentu bolo skúmanie a porovnanie dielektrických vlastností transformátorového oleja MOGUL prostredníctvom merania diagnostických parametrov charakterizujúcich elektro fyzikálnu štruktúru materiálu vplyvom zvyšovania teploty na kvapalné dielektrikum. Experiment porovnával dva online testy. 1. test charakterizuje meranie vo frekvenčnej a v časovej oblasti pri zvyšovaní teploty oleja od 20 °C do 90 °C s krokom 10 °C a následnom znižovaní teploty od 90 °C do 20 °C s krokom 10 °C. To znamená, že 1. test obsahoval 16 meraní vo frekvenčnej oblasti a 16 meraní v časovej oblasti. 2. test sa zameriaval na meranie vo frekvenčnej a v časovej oblasti pri znižovaní teploty oleja od 90 °C do 20 °C s krokom 10 °C. Čiže 2. test obsahoval 8 meraní vo frekvenčnej oblasti a 8 meraní v časovej oblasti. Pri oboch testoch bol vo frekvenčnej oblasti použitý merací prístroj IDAX 300, ktorý generoval na vzorku izolačného oleja striedavé napätie 100 V vo frekvenčnom rozsahu od 1 mHz do 3 kHz, kde boli odmerané hodnoty stratového činiteľa $tg \delta$. V časovej oblasti bol pri oboch testoch použitý merací prístroj KEITHLEY 6517B, ktorý meral hodnoty nabíjajúcich prúdov I_{nab} vygenerovaných jednosmerným napätím 100 V po dobu nabíjania 1000 s. Rozdiel medzi 1. a 2. testom je v čase dosiahnutia teploty z 20 °C na 90 °C. Pri 1. teste išlo o postupné zvyšovanie teploty na rozdiel od 2. testu, kde bola teplota 90 °C dosiahnutá priamo. Toto porovnanie hralo kľúčovú úlohu pri vyšetrovaní správania sa vlhkosti v izolačnom oleji.

Na Obr. 1 je znázomená schéma zapojenia meracieho pracoviska s použitými prístrojmi.



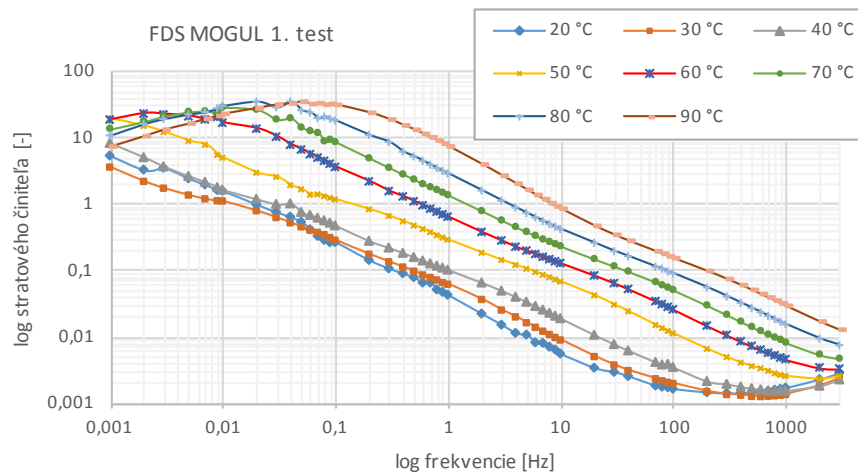
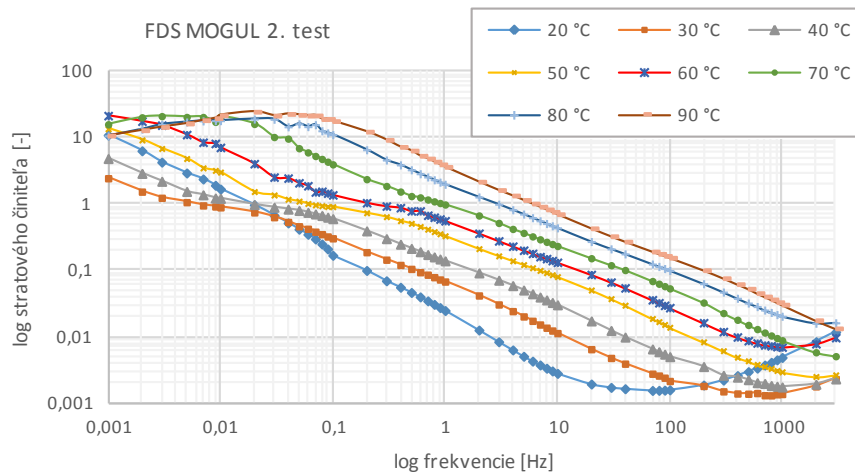
Obr. 1 Schéma zapojenia meracieho pracoviska

IV. ANALÝZA NAMERANÝCH ÚDAJOV

V tejto časti sú popísané namerané výsledky vo frekvenčnej a v časovej oblasti a porovnanie oboch testov.

A. Analýza vo frekvenčnej oblasti

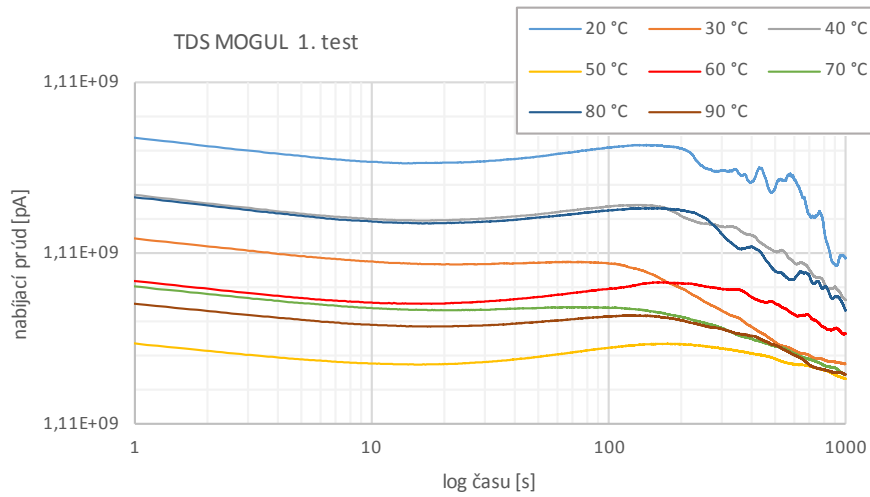
Z nameraných údajov vo frekvenčnej oblasti boli pomocou programu Excel vynesené grafické závislosti stratového činiteľa $tg \delta$ v závislosti na frekvencii pri znižovaní teploty z 90 °C na 20 °C a porovnané obidva testy s rôznou dobou zohriatia oleja na 90 °C.

Obr. 2 Závislosť $tg\delta = f(f)$ pri poklese teploty z 90 °C na 20 °C - 1. testObr. 3 Závislosť $tg\delta = f(f)$ pri poklese teploty z 90 °C na 20 °C - 2. test

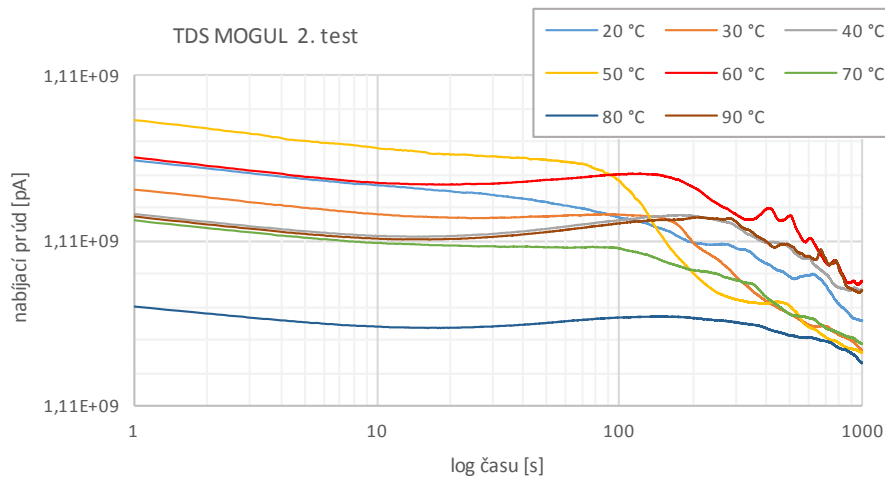
Z grafických závislostí na Obr. 2 a 3 je možné si všimnúť, že hodnoty stratového činiteľa klesajú vo frekvenčnom pásme od 0,1 Hz do 100 Hz so znižujúcou sa teplotou pri oboch testoch, čo korešponduje s teoretickými poznatkami. Pri porovnaní oboch testov bolo zistené, že v oblasti nízkych frekvencií (kedy prevláda jednosmerná vodivosť) dosiahli hodnoty stratového činiteľa vyššie čísla pri 1. teste než pri 2. teste vo všetkých teplotách okrem 20 °C. Pri teplote 20 °C boli hodnoty stratového činiteľa vyššie pri 2. teste.

B. Analýza v časovej oblasti

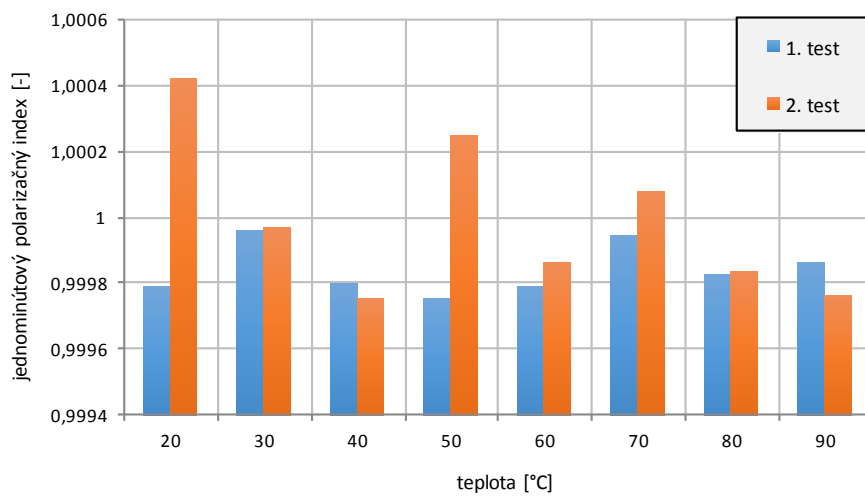
Z nameraných údajov v časovej oblasti boli pomocou programu Excel vynesené grafické závislosti nabíjacieho prúdu I_{nab} v závislosti na čase pri znižovaní teploty z 90 °C na 20 °C a porovnané obidva testy s rôznou dobou zohriatia oleja na 90 °C. Na Obr. 4 a 5 sú znázornené hodnoty nabíjajúcich prúdov v závislosti na čase pre 1. a 2. test. Na lepšie porovnanie 1. a 2. testu boli hodnoty nabíjajúcich prúdov prepočítané na jednominútové polarizačné indexy p_1 , kde boli podelené hodnoty prúdu v čase 15 s hodnotami prúdu v čase 60 s. Grafické znázornenie jednominútových polarizačných indexov v závislosti na teplote pre obidva testy je na Obr. 6.



Obr. 4 Závislosť $I_{nab} = f(t)$ pri poklese teploty z 90 °C na 20 °C - 1. test



Obr. 5 Závislosť $I_{nab} = f(t)$ pri poklese teploty z 90 °C na 20 °C - 2. test



Obr. 6 Porovnanie testov pomocou jednonímútových polarizačných indexov pri rôznej teplote

Z Obr. 6 možno vyčítať, že polarizačné indexy pri 2. teste dosahovali vyššie hodnoty ako pri 1. teste pri všetkých teplotách okrem teploty 40 °C a 90 °C. To znamená, že 2. test poukazoval na väčšiu elimináciu vlhkosti v oleji ako 1. test. Čiže z danej skutočnosti je možné usúdiť, že s rýchlejšim časom zohriatia oleja MOGUL na 90 °C dochádza pri postupnom chladnutí oleja

k nižšiemu obsahu vlhkosti v izolačnom oleji ako s pomalším časom zohriatia. Z Obr. 6 je taktiež možné si všimnúť, že jednominútové polarizačné indexy sa u oboch testoch menia pri každej teplote a predstavujú určitý proces kmitania. Tento jav by mohol byť v budúcnosti predmetom ďalšieho skúmania.

V. ZÁVER

Cieľom tohto príspevku bolo porovnať jednotlivé časové testy zohrievania oleja na 90 °C s dôrazom na vlhkosť v oleji vo frekvenčnej oblasti a v časovej oblasti dielektrickej spektroskopie. Vo frekvenčnej oblasti bolo zistené, že v oblasti nízkych frekvencií (kedy prevláda jednosmerná vodivosť) dosiahol stratový činiteľ vyššie hodnoty pri 1. teste než pri 2. teste. Časová oblasť poukázala nato, že 2. test dosiahol väčšiu elimináciu vlhkosti v oleji ako 1. test, prostredníctvom vyšších hodnôt jednominútových polarizačných indexov. To znamená, že týmto príspevkom bol zistený pozitívny vplyv rýchlejšieho zohriatia oleja (90 °C) na vlhkosť v izolačnom oleji ako pri pomalšom zohrievaní a to prostredníctvom stratového činiteľa vo frekvenčnej oblasti aj polarizačného indexu v časovej oblasti dielektrickej spektroskopie.

POĎAKOVANIE

Túto prácu podporili štrukturálne fondy EÚ v rámci projektov ITMS 313011T565 a Ministerstvo školstva pre VEGA 1/0340/18, 2/0141/16 a APVV 15-0438, 17-0372, 18-0160.

LITERATÚRA

- [1] R. Cimbala, P. Semančík, "Tepelná degradácia kvapalných izolačných materiálov", Technická univerzita v Košiciach, Letná 9, Košice. 2016, 147 p., ISBN 978-80-553-2545.
- [2] H. Zhao, J. Tian, D. Zhang, X. Liu, "Research on Dielectric Properties about Different Curing Degree of Epoxy Glass Fiber", *Condition Monitoring and Diagnostic*, 2018.
- [3] R. Farber, C. M. Franck, A. Nasef, "Online Dielectric Response Analysis Under Mixed-Frequency Medium-Voltage Stress", *21th International Symposium on High Voltage Engineering*, 2020.
- [4] Ch. Suo, Y. Sun, Z. Li, Y. Han, "Application of L1 Trend Filtering Technology on the Current Time Domain Spectroscopy of Dielectrics", DOI: 10.3390/electronics8091046, 2019.

Hrací agent na báze prehľadávania stromov

¹Peter LIČKO, ²Marián MACH

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹peter.licko@student.tuke.sk, ²marian.mach@tuke.sk

Abstrakt — Cieľom článku je problematika tvorby agentov schopných hrania sekvenčných ťahových hier dvoch hráčov. Pozornosť je venovaná prístupu založenému na prehľadávaní stromov hier, nevyžadujúcemu fázu učenia. Článok sa zaoberá návrhom agenta využívajúceho algoritmus Monte-Carlo Tree Search a niekoľkými jeho modifikáciami. Navrhnutý a implementovaný agent bol experimentálne overený na hre Connect-4. Ťažisko vykonaných experimentov spočívalo v zistení výkonnosti tohto agenta a jeho časových požiadaviek v závislosti na nastavení niektorých parametrov.

Príučové slová — Connect-4, Monte-Carlo Tree Search, nekooperatívna hra, strom prehľadávania

I. ÚVOD

Pri tvorbe agentov pre hranie sekvenčných ťahových hier je populárnym prístup založený na prehľadávaní stromov hier. V hre jednotliví hráči striedavo vykonávajú svoje ťahy, čím sa situácia hry neustále mení až do okamihu keď nastane situácia, znamenajúca ukončenie hry. Pri vytváraní stromov hier jednotlivé situácie sú reprezentované stavmi, ktoré tvoria uzly stromu. Koreňom stromu je buď počiatočná situácia, z ktorej hra začína, alebo aktuálna situácia v rozohrenej hre, ktorá má pokračovať. Ak v nejakej situácii je možné vykonať niekoľko rozličných ťahov, tak v strome uzol má toľko nasledovníkov, koľko je týchto možných ťahov, pričom každý nasledovník reprezentuje situáciu po vykonaní jedného z ťahov. Sekvencia ťahov je potom ekvivalentná ceste v strome z koreňového uzla. Situácia, znamenajúca ukončenie hry, je v strome reprezentovaná koncovým uzlom.

Tradičný algoritmus minimax [1] prehľadáva strom hry uniformne do zadanej hĺbky. Ak sa mu podarí rozvinúť strom úplne, tak poskytuje presné výsledky a umožňuje vybrať optimálny ťah. To sa však podarí iba pri veľmi jednoduchých hrách. MCTS (Monte-Carlo Tree Search) dynamicky prioritizuje uzly stromu, ktoré sú v danom čase najslubnejšie a na základe toho robí odhad skutočnej hodnoty uzlov na vyššej úrovni, potrebný pre konečný výber ťahu [2].

II. ALGORITMUS MCTS

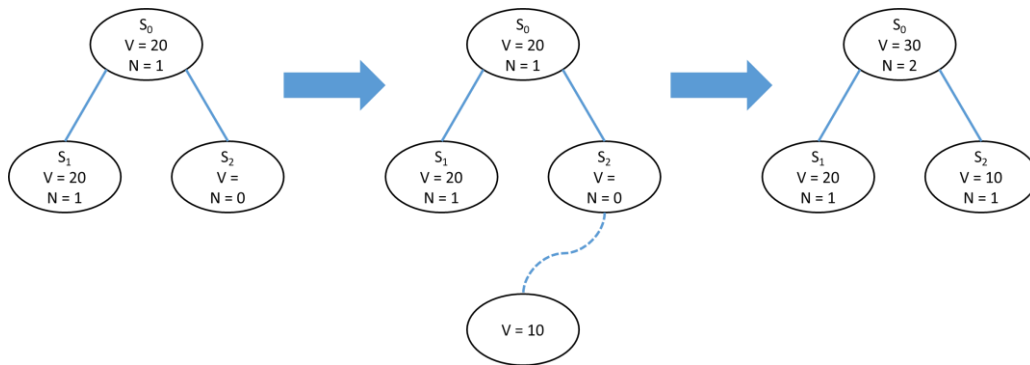
A. Základná podoba algoritmu

Základná podoba algoritmu [3] pri každej voľbe ťahu buduje strom hry odznova, pričom začína stromom iba s jedným uzlom, ktorý reprezentuje aktuálny stav hry. Tento strom je v ďalšom rozvíjaný o ďalšie stavy, dosiahnuteľné z aktuálneho stavu sekvenciou ťahov, pričom aktuálny stav ostáva koreňom stromu. Samotný algoritmus pozostáva zo štyroch fáz:

- Selekcia – postupne vyberá toho potomka aktuálneho uzla, ktorý maximalizuje hodnotu použitého kritéria. Skončí po dosiahnutí listového uzla.
- Expanzia – ak listový uzol stromu ešte nie je koncovým uzlom hry, tak sa expanduje o svojich potomkov a z nich sa vyberie jeden.
- Simulácia – zo stavu, reprezentovanom vybratým listovým uzlom, sa vykoná simulácia priebehu hry až do koncového stavu. Ak aktuálny uzol je už koncovým uzlom hry, tak sa simulácia nevykonáva (lebo výsledok hry je už známy).
- Spätná propagácia – od listového uzla, v ktorom bola vykonaná simulácia, sa informácia propaguje naspäť až ku koreňovému uzlu stromu.

Činnosť algoritmu je ilustrovaná na obr. 1. Každý uzol stromu obsahuje svoj identifikátor, hodnotu V a počet N , koľkokrát už uzol bol algoritmom uvažovaný ako aktuálny uzol. Strom

hry na obrázku momentálne obsahuje tri uzly, pričom algoritmus v predchádzajúcich iteráciách navštívil doteraz iba dva z nich (S_0 a S_1).



Obr. 1 Ilustrácia činnosti algoritmu MCTS – stav stromu hry zobrazený po selekcii (vľavo), po simulácii (v strede) a po spätnej propagácii (vpravo)

Novú iteráciu algoritmus začína v koreňovom uzle S_0 . V rámci selekcie vyberá jedného z potomkov – nech to je uzol S_2 . Keďže je to listový uzol, selekcia končí. Vzhľadom na to, že uzol ešte doteraz nebol navštívený, expanzia sa nerobí (robila by sa, ak by namiesto uzla S_2 bol vybraný S_1). Keďže uzol nie je koncovým a hra z daného stavu môže pokračovať ďalej, tak sa vykoná simulácia tejto hry a zistí sa jej výsledok. Tento výsledok sa propaguje naspäť a následkom propagácie sa zmení informácia v uzloch S_2 a S_0 – inkrementuje sa počet navštívení a upraví sa hodnoty týchto uzlov.

V prípade výberu uzla pri expanzii sa vyberá náhodne. Pri selekcii sa ako kritérium pre výber uzlov často používa UCT (Upper Confidence Bound) [4] v tvare

$$UCB_i = H_i + C \sqrt{\frac{\ln N_p}{N_i}} \quad (1)$$

kde H_i je priemerná hodnota všetkých potomkov uzla i (odvodená z V_i a počtu potomkov) a N_p je počet navštívení rodiča uzla i . Hodnota C sa označuje ako exploračný faktor a udáva v akom pomere budú uvažované obe zložky. Prvá zo zložiek preferuje uzol s najlepšou hodnotou, zatiaľ čo druhá zložka preferuje málo navštevovaný uzol.

Po skončení iterovania sa zvyčajne vyberá ten ťah, ktorý z koreňového uzla vedie na najnavštevovanejší uzol – aby bol uzol často navštevovaný, musí mať najlepšiu hodnotu UCB a teda musí mať aj vysokú hodnotu V . Takýto uzol sa uprednostňuje pred uzlom so síce vysokou hodnotou ale malým počtom navštívení, pretože sa berie do úvahy, že hodnota uzla je iba odhadnutá a nie je presná.

B. Modifikácia základného algoritmu

Základná podoba algoritmu pripúšťa rôzne modifikácie [5]. My sme sa zamerali na dve. Počas hľadania vhodného ťahu má algoritmus k dispozícii iba obmedzený čas na to, aby vygeneroval strom hry a odhadol hodnoty uzlov. Pritom na začiatku hry by potreboval času viac (pretože pred hráčmi je viac ťahov, ktoré je potrebné vykonať do konca hry) ako v neskorších fázach hry (do konca hry ostáva vykonať menej ťahov). Pre zohľadnenie štádia hry bol použitý počet ťahov a časový limit na jeden ťah bol určený vzťahom

$$T_i = T_0 \left(\frac{n_T - i}{n_T} \right)^\alpha \quad (2)$$

kde i reprezentuje aktuálny ťah, n_T je maximálny počet ťahov, T_0 je najdlhší povolený čas na jeden ťah a α je útlmový parameter, určujúci spôsob skracovania času s narastajúcou dĺžkou odohranej hry. Čím je útlm vyšší, tým je čas skracovaný rýchlejšie.

Druhou modifikáciou je viacnásobná simulácia – ak je listový uzol expandovaný o svojich potomkov, tak potom simulácia sa nevykoná iba pre jedného z nich ale pre všetkých. Pre časovú optimalizáciu:

- Simulácie z vybraných listových uzlov do konca hry sa vykonávajú paralelne.
- V rodičovskom uzle vybraných listových uzlov sa vykoná agregácia informácie zo všetkých simulácií.
- Agregovaná informácia je spätne propagovaná stromom hry smerom zdola nahor až k jeho koreňu.

To znamená, že po skončení iterácie v strome hry nie je žiadny listový uzol, ktorý by ešte

nebol navštevovaný – a teda v ďalšej iterácii algoritmu po fáze selekcie vždy bude nasledovať fáza expanzie.

III. VYBRANÉ EXPERIMENTY

A. Testovacia hra

Pre testovanie hracieho agenta bola použitá hra Connect-4 známa aj pod menom „štyri v rade“. Je to hra s hracou doskou obsahujúcou 6x7 políčok, ktorá je umiestnená vo vertikálnej polohe. Hra je znázornená na obr. 2.



Obr. 2 Hracia doska hry Connect-4

Hru hrajú dvaja hráči, striedajúci sa vo svojich ťahoch. Hra je hraná podľa týchto štyroch pravidiel:

- Hráč na ťahu vkladá žetón svojej farby do jedného zo stĺpcov, pričom tento žetón padá nadol na najnižšiu voľnú pozíciu v danom stĺpci.
- Žetón môže byť vložený iba do stĺpca, ktorý ešte nie je úplne zaplnený vloženými žetónmi.
- Hráč, ktorý svojím ťahom uzavrel líniu štyroch žetónov svojej farby (vodorovne, zvisle alebo diagonálne) sa stáva víťazom.
- Hra končí remízou v prípade, že všetky stĺpce sú zaplnené ale žiadny hráč neuzavrel líniu štyroch svojich žetónov.

Jedná sa o sekvenčnú ťahovú hru s perfektnou informáciou a nulovým súčtom. Hra má stavový priestor obsahujúci 4.5×10^{12} rôznych stavov. Jej faktor vetvenia je na začiatku 7 avšak v priebehu hry sa znižuje (podľa toho ako sa zaplňajú jednotlivé stĺpce) [6]. Pretože niektoré stavy sú dosiahnuteľné viacerými sekvenciami ťahov, úplný strom prehľadávania má viac uzlov než stavový priestor.

B. Počiatočné nastavenie

Výsledok hry sa určuje až po dosiahnutí konečného stavu. Hra môže skončiť výhrou hráča, remízou alebo prehrou hráča, pričom výhra jedného hráča znamená prehru hráča druhého. Z hľadiska algoritmu po dosiahnutí niektorého koncového uzla, reprezentujúceho dosiahnutý konečný stav hry, nastáva spätná propagácia. Táto propagácia informácie je realizovaná všetkými uzlami, ktoré ležia na ceste od koreňa stromu k tomu listu, z ktorého bola vykonávaná simulácia. Toto zahŕňa propagáciu:

- Počtu návštev – hodnota N každého uzla na danej ceste je inkrementovaná
- Hodnôt uzla – hodnota V každého uzla na danej ceste je zvýšená o 1 (ak hráč, ktorého ťah je očakávaný v koreňovom uzle, vyhral), 0 (ak daný hráč prehral) alebo 0.5 (v prípade remízy).

Keďže rozmer hracej dosky je pevne daný, je zrejmé že hra môže trvať maximálne 42 ťahov (v prípade, že všetky polia dosky budú zaplnené žetónmi).

Experimenty boli realizované na počítači s Ryzen 7 3800X (8-core, 4.1 GHz).

C. Porovnanie výkonnosti

Prvým experimentom [7] bolo porovnanie výkonnosti hracieho agenta založeného na MCTS

voči agentovi, ktorý svoje kroky vyberal náhodne z aktuálne dostupných ťahov s rovnakou pravdepodobnosťou. Obaja agenti zohrali v každom nastavení 240 hier, pričom sa uvažoval ako prípad, keď začínal náhodný agent, tak aj prípad, keď začínal agent MCTS. Skúmalo sa, aký vplyv má množstvo dostupného času T_0 na výkon agenta MCTS (útlm α bol nulový). Dosiiahnuté výsledky sú v tab. 1.

Tabuľka 1
 Porovnanie MCTS s náhodným výberom ťahu

Časový limit [s]	RANDOM vs MCTS	MCTS vs RANDOM
0.005	62 : 0 : 178	199 : 0 : 41
0.01	53 : 0 : 187	211 : 0 : 29
0.02	54 : 0 : 186	214 : 0 : 26
0.05	23 : 0 : 217	233 : 0 : 7
0.1	4 : 2 : 234	237 : 0 : 3
0.2	1 : 0 : 239	240 : 0 : 0
0.4	0 : 0 : 240	240 : 0 : 0

Z výsledkov jasne vidieť, že hra nie je symetrická, ale výhodu má začínajúci hráč. MCTS agent mal v každom nastavení prevahu nad náhodným agentom, ktorá stúpala so stúpajúcim časovým limitom.

Teda $T_0=0.4$ postačovalo na náhodného agenta (čo je však vlastne slabý protivník). Preto bol MCTS agent postavený voči svojej kópii. Rozdiel bol v povolenom časovom limite. Výsledkom súboja 120 hier medzi MCTS($T_0=0.4$) a MCTS($T_0=2.0$) = bolo skóre 23 : 93. Teda algoritmus dokázal ťažiť z ďalšieho zvýšenia časového limitu.

Ďalším experimentom [7] bolo skúmanie vplyvu útlmového parametra. Predpokladalo sa, že pridanie útlmu spôsobí zhoršenie výkonu agenta, otázkou bolo aké veľké zhoršenie. V rámci experimentu súťažili medzi sebou dve kópie MCTS agenta, obe s nastavením $T_0=2.0$. Jedna kópia mala zapnutý útlm, druhá používala útlm nulový. Dosiiahnuté výsledky tohto experimentu sú v tab. 2.

Tabuľka 2
 Porovnanie výkonu MCTS s útlmom a bez útlmu

Útlm	MCTS(α) vs MCTS	MCTS vs MCTS(α)
1	27 : 2 : 31	34 : 1 : 25
1.5	25 : 1 : 34	34 : 3 : 23
2	23 : 3 : 34	37 : 5 : 18
2.5	17 : 4 : 39	38 : 2 : 20
3	14 : 3 : 43	40 : 5 : 15
3.5	20 : 1 : 39	40 : 0 : 20
4	8 : 2 : 50	46 : 0 : 14

So zvyšujúcim sa útlmom výkon agenta degradoval podľa očakávania. Použitie útlmu malo vo všetkých prípadoch negatívny vplyv – avšak pri malých hodnotách útlmu pokles vo výkonnosti nebol príliš závažný.

Tabuľka 3
 Časové porovnanie MCTS s útlmom a bez útlmu

T_0 [s]	MCTS($\alpha=1$) [s]	MCTS($\alpha=0$) [s]
2	201	370
1.5	154	280
1	105	190
0.5	54	90

Otázkou je, či tento pokles výkonnosti je vhodne kompenzovaný znížením spotrebovaného času na výber ťahu. Tab. 3 zobrazuje súboj dvoch MCTS agentov na sérii 60 hier, pričom jeden z nich bol bez útlmu ($\alpha=0$) zatiaľ čo druhý používal útlm ($\alpha=1$).

Z experimentu je zrejmé, že zapnutie útlmu dokáže znížiť časové nároky pri hre až o polovicu, čo sa zdá dostatočnou kompenzáciou za pokles výkonnosti podľa tab. 2.

IV. ZÁVER

Na základe vykonaných experimentov je možné konštatovať, že algoritmus MCTS je vhodným prostriedkom na vytváranie hracích agentov v tých prípadoch, keď strom hry je príliš veľký a nie je ho možné úplne rozvinúť. Uplatnenie nájde v prípadoch, keď má k dispozícii čas na výber ťahu v rozmedzí stoviek milisekúnd až sekúnd. Pri zmenšení tohto času sa už jeho použitie javí ako nevhodné, naopak pri zväčšení časového limitu je zlepšovanie jeho výkonnosti otáznou. Výhodou algoritmu je to, že nevyžaduje fázu učenia sa hrať danú hru, ale môže byť nasadený bez tejto fázy. V budúcnosti by bolo vhodné ho porovnať s hracími agentmi, založenými na učiacej fáze, využívajúce napríklad učenie s posilňovaním či tréning neurónových sietí.

POĎAKOVANIE

Táto publikácia vznikla vďaka podpore Agentúry pre podporu výskumu a vývoja v rámci projektu APVV-15-0731 2016-2020 „Cloud Based Human Robot Interaction“.

LITERATÚRA

- [1] Russel S., Norvig P.: Artificial Intelligence: A Modern Approach, Third Edition, Pearson, 2009.
- [2] Browne C.B., Powley E., Whitehouse D., Lucas S.M., Cowling P.I.: A Survey of Monte Carlo Tree Search Methods. IEEE Trans. on Computational Intelligence and AI in Games, 4 (1), 2012, 1-43.
- [3] Chaslot G., Bakkes S., Szita I., Spronck P.: Monte-Carlo Tree Search: A New Framework for Game AI. Proc. of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conference, Stanford, CA, 2008, 216-217.
- [4] Kocsis L., Szepesvári C.: Bandit based Monte-Carlo Planning. Proc. of the 17th European Conference on Machine Learning, Berlin, Springer, 2006, 282-293.
- [5] Baier H.J.S.: Monte-Carlo Tree Search Enhancements for One-Player and Two-Player Domains. Dissertation, Maastricht University, 2015.
- [6] S. Bagheri, M. Thill, P. Koch, W. Konen: Online Adaptable Learning Rates for the Game Connect-4. IEEE Trans. on Computational Intelligence and AI in Games, 8 (1), 2016, 33-42.
- [7] P. Ličko: Antagonistické hry a metódy umelej inteligencie, Diplomová práca, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, 2020, 68 strán.

Kryptografické zabezpečenie a bezpečná aktualizácia firmvéru v IoT aplikáciách na báze ESP32

¹Martin CHLEBOVEC, ²Miloš DRUTAROVSKÝ

¹Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky
a informatiky, Technická univerzita v Košiciach, Slovenská republika

²Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky
a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹martin.chlebovec.2@student.tuke.sk, ²milos.drutarovsky@tuke.sk

Abstrakt — Článok poukazuje na možnosti vzdialenej aktualizácie firmvéru na platforme ESP32 prostredníctvom nástroja (frameworku) ESP-IDF. Dôraz je kladený na možnosti kryptograficky zabezpečenej aktualizácie v podobe overenia aktualizácie vloženým a neskôr overeným digitálnym podpisom na strane ESP32 klienta, ktorý aktualizáciu prevzal. Je opísané aj využitie funkcionality zabezpečeného štartu systému (secure boot) pre zvýšenie bezpečnosti bootovacieho procesu počas inicializácie systému.

Kľúčové slová — digitálny podpis, firmvér, OTA aktualizácia, Secure Boot

I. ÚVOD

Espressif Systems je čínskou firmou, ktorá vznikla v roku 2008. Je producentom, vývojárom IoT čipov s označením ESP. Do roku 2014 firma vyrábala WiFi čipy pre smartfóny a tablety. Čipy zabezpečovali WiFi konektivitu týchto zariadení. Následne sa firma zviditeľnila predovšetkým s vývojom a vytvorením prvého IoT WiFi čipu ESP8266, ktorý okrem WiFi časti obsahoval aj flash pamäť, ktorá umožňovala spúšťanie užívateľských programov, pričom existuje podpora viacerých programovacích jazykov, v ktorých je možné vyvíjať firmvér (ESP-IDF, Wiring, Lua, Node.js, MicroPython, Mongoose OS).

Výrazným spôsobom sa zaslúžila o rozvoj IoT vo svete. V roku 2016 firma Espressif Systems predstavila platformu ESP32 [1], ku ktorej aj v súčasnosti vyvíja framework ESP-IDF (Espressif – IoT Development Framework).

Framework je určený pre vývojárov a spoločne s operačným systémom reálneho času RTOS bežiacim na ESP32 je možné na tejto IoT platforme vytvárať komplexnejšie programy, využívať procesy (tasks), ktoré môžu nezávisle na sebe pracovať na dvojjadrovom procesore Xtensa s definovanou prioritou.

Framework obsahuje príklady pre rôzne implementácie konkrétnych aplikácií, zberníc a je doplnený o komponenty – vývojárske nástroje, ktoré umožňujú konfigurovať dodatočné funkcionality mimo bežiaceho programu.

II. AKTUALIZÁCIA FIRMVÉRU NA PLATFORME ESP32 POD FRAMEWORKOM ESP-IDF

A. Uloženie firmvéru v pamäti flash

Platforma ESP32 sa radí k IoT platformám, ktoré podporujú vzdialenú aktualizáciu firmvéru distribuovanú serverom, jedná sa o tzv. vzdialenú (remote) aktualizáciu zo vzdialeného centralizovaného bodu. Proces aktualizácie firmvéru sa nazýva aj OTA (Over The Air) [2]. Procesory na báze ESP32 obsahujú pamäť flash, ktorú je možné prispôsobiť podľa potreby pre akúkoľvek aplikáciu.

Pamäť flash [3] je rozdelená na systém partícií, ktoré tvoria logický celok. Partície a ich veľkosti sú zapísané v tabuľke partícií, čo je v podstate .csv súbor – Partition Table. Okrem samotných programových partícií (v Tab. 1) sú dostupné aj partície, ktoré obsahujú konfiguráciu WiFi adaptéra – MAC adresu fyzického rozhrania a iné.

Štandardná OTA implementácia obsahuje rozdelenie partícií súvisiace s firmvérom zobrazené v Tab.1.

Partícia	Posun (Offset)	Veľkosť	Popis
FACTORY	0x10000	1MB	Hlavná aplikácia
OTA_0	0x110000	1MB	Aktualizovaný firmvér 1
OTA_1	0x210000	1MB	Aktualizovaný firmvér 2
OTA_DATA	0xd000	2kB	Príznak pre bootovanie z partície

Tab. 1 - Programové partície FACTORY + 2x OTA

Z partícií FACTORY, OTA_0, OTA_1 je možné systém inicializovať (bootovať). Celkovo tak môže WiFi systém na báze ESP32 obsahovať až tri samostatne spustiteľné programy v pamäti zároveň. Do partície FACTORY je zapísaná základná aplikácia používateľom, napríklad prostredníctvom USB kábla. Z menovanej partície sa za normálnych okolností prioritne inicializuje v prípade, že nie je boot príznak pozmenený v partícii OTA_DATA. Pri aktualizácii sa partícia FACTORY neprepisuje, funguje ako tzv. „fail-safe“ riešenie v prípade nemožnosti bootovať z inej partície.

Pôvodný program je tak stále dostupný. Každá z OTA partícií umožňuje uložiť spustiteľný (binárny) program a je z nich možné tiež bootovať. OTA partície slúžia na zápis a uchovávanie aktualizovaného programu. Na jednej OTA partícii tak môže byť aktuálne bežiaci firmvér, pričom druhá OTA partícia je pripravená pre uloženie novej verzie firmvéru v prípade jeho dostupnosti.

Partícia OTA_DATA zodpovedá za to, ktorá partícia bude bootovaná prostredníctvom zavádzača (bootloadera). Údaje z tejto partície sa využívajú pri spustení ESP32. Framework ESP-IDF vo verzii 4.0 ponúka OTA metódu vzdialenej aktualizácie firmvéru, ktorá je založená na princípe klienta-server. Klient na báze ESP32 sa pripojí na vzdialený server, z ktorého prevezme aktualizáciu, ktorá je tvorená binárnym spustiteľným súborom firmvéru. Firmvér sa následne uloží do jednej z dvoch dostupných OTA partícií.

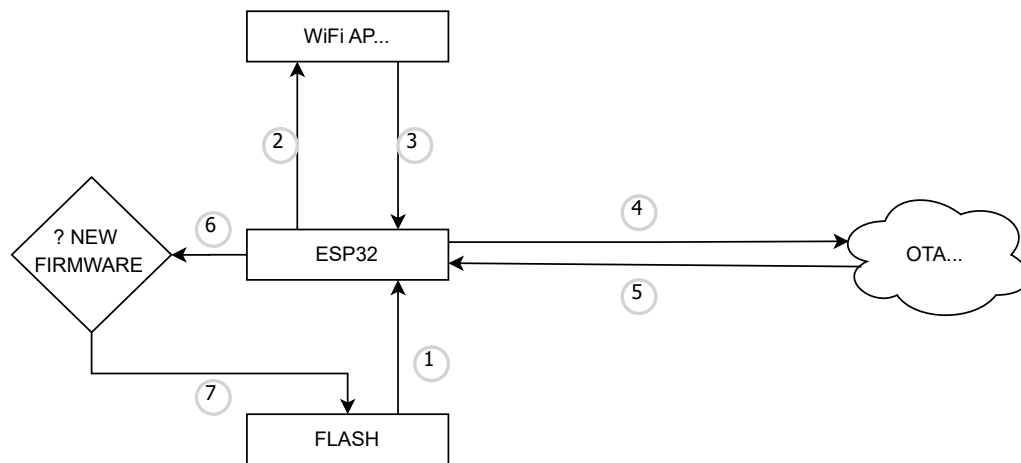
Softvérová implementácia príkladu Native OTA [4] umožňuje pripojenie na vzdialený server prostredníctvom zabezpečeného HTTPS kanálu s využitím certifikátu certifikačnej autority pre realizáciu zabezpečeného spojenia. Okrem zabezpečeného HTTPS spojenia obsahuje príklad Native OTA aj implementáciu pre kontrolu verzie stiahnutého firmvéru, čo rozhoduje o jeho prednostnom bootovaní. Verzia firmvéru je nastavená v textovom súbore version.txt obsiahnutom v príklade a aplikuje sa v procese kompilácie pri vytváraní aplikácie.

B. Princíp procesu aktualizácie

Na blokovej schéme (Obr. 1) je znázornený spôsob vykonania aktualizácie. Systém Čip ESP32 bootuje firmvér z pamäte flash (1). Na základe príznaku z partície OTA_DATA je možné bootovať z troch dostupných programovateľných partícií. Následne ESP32 získa konektivitu do WiFi siete (2) autentizáciou heslom na konkrétne SSID siete štandardu WPA/WPA2 PSK. Prístupový bod prideli čipu IP adresu (3) verzie 4 a link-local IP adresu verzie 6.

Čip ESP32 realizuje šifrované HTTPS spojenie (4) s využitím Root CA certifikátu (koreňového certifikátu certifikačnej autority) na vzdialený server, ktorý distribuuje OTA aktualizáciu. Odpoveďou servera (5) je binárny súbor firmvéru, ktorý ESP32 stiahne a uloží do dostupnej OTA partície. Čip ESP32 porovná verziu bežiaceho firmvéru s verziou stiahnutého (6) a v prípade, že nie sú zhodné, vykoná sa prepísanie obsahu OTA_DATA partície (7) pre príznak bootovania firmvéru z partície, do ktorej bol nový firmvér zapísaný. Softvérovo sa ESP32 rešartuje a bootuje z novej partície. V prípade, že sú obe verzie firmvérom zhodné, prepis OTA_DATA partície, ani rešart sa nevykoná.

Programová implementácia vykoná stiahnutie a porovnanie firmvéru jednorázovo po štarte čipu ESP32.



Obr. 1 Princíp realizácie OTA aktualizácie firmvéru v ESP32.

III. KRYPTOGRAFICKÉ ZABEZPEČENIE AKTUALIZÁCIE V ESP-IDF

A. Metóda digitálneho podpisu

Framework ESP-IDF (v4.0) je vybavený jednoduchým príkazovým riadkom – konzolou [5], pomocou ktorej je možné ovládať všetky funkcionality frameworku. Ten ponúka vývojárske nástroje, ktoré je možné konfigurovať pomocou konfiguračného menu [6], ktoré obsahuje každá implementácia v ESP-IDF. Menu obsahuje predpripravené skripty napríklad pre zabezpečenie firmvéru, ktoré je možné výberom v konfiguračných nastaveniach zahrnúť do výsledného skompilovaného firmvéru.

Využitie digitálneho podpisu je jednou z podporovaných možností [7]. Digitálny podpis umožňuje garantovať integritu firmvéru, ktorý je distribuovaný zo vzdialeného servera. Tým vieme zaručiť, že distribuovaný firmvér nebol pozmenený (ani zámerným útokom) a je bezpečný pre spustenie. Nakoľko však generovaný firmvér nie je podpísaný počas procesu kompilácie, je nutné ho podpísať dodatočne pred fyzickým nahrať program do platformy ESP32.

Digitálny podpis je založený na asymetrickej kryptografii na báze eliptických kriviek (Elliptic Curve Cryptography) s využitím verejného a súkromného kľúča. Implementácia digitálneho podpisu vyžaduje generovanie dvojice kľúčov. Súkromný kľúč je možné generovať pomocou Python skriptu `espsecure.py` [8], ktorý je súčasťou ESP-IDF. Ako alternatívu je možné využiť aj kryptografický nástroj OpenSSL [9]. Súkromný kľúč je generovaný pre eliptickú krivku NIST256p (ekvivalent `prime256v1` v nástroji OpenSSL).

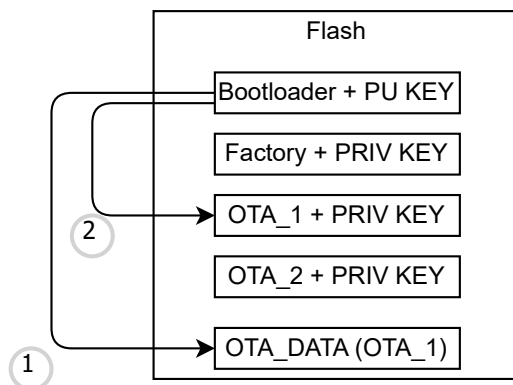
Veľkosť generovaného súkromného kľúča je 256 bitov a je generovaný v štandardom PEM formáte. Tento kľúč je neskôr využitý pre podpísanie firmvéru manuálne vydavateľom firmvéru. Nástrojom `espsecure.py` je možné zo súkromného kľúča extrahovať verejný kľúč. Tento verejný kľúč slúži na overenie digitálneho podpisu, ktorým je podpísaný nový firmvér. Veľkosť verejného kľúča je tiež 256 bitov, pričom je v binárnom formáte. Verejný kľúč je vložený do zavádzača (bootloadera) v procese kompilácie.

Po vykonaní kompilácie je nutné dodatočne podpísať firmvér súkromným kľúčom. Pre podpísanie firmvéru je možné opäť použiť Python skript `espsecure.py`. Po vykonaní tohto kroku je možné program nahrať do čipu ESP32 napríklad fyzicky prostredníctvom USB-UART rozhrania z konzoly ESP-IDF. Nový firmvér, aktualizácia pre platformu ESP32, je umiestnený na distribučný server a musí byť taktiež podpísaný rovnakým súkromným kľúčom aj pri budúcich aktualizáciách.

B. Princíp funkčnosti overenia digitálneho podpisu

Overenie digitálneho podpisu je možné realizovať dvoma spôsobmi: s využitím aplikácie (ON UPDATE) alebo pomocou bootloadera [10]. U oboch spôsobov sa overenie realizuje pomocou verejného kľúča. Pri možnosti ON UPDATE je overenie digitálneho podpisu realizované softvérovou po stiahnutí a uložení aktualizovaného firmvéru do OTA partície. Digitálny podpis je overený a v prípade neplatného/neexistujúceho digitálneho podpisu sa zmena v partícii `OTA_DATA` nevykoná, rovnako tak ani reštart čipu ESP32. Táto funkcionality je užitočná najmä proti útokom MITM (Man In The Middle) typu podvrhnutia firmvéru. Integrita firmvéru by v tomto prípade nebola overená a čip ESP32 by nebootoval nový firmvér, ktorý by bol potenciálne podvrhnutý. Možnosť overenia digitálneho podpisu bootloaderom (Obr.2) sa realizuje pri procese bootovania. ESP32 bootuje z partície, ktorá je predvolená (1)

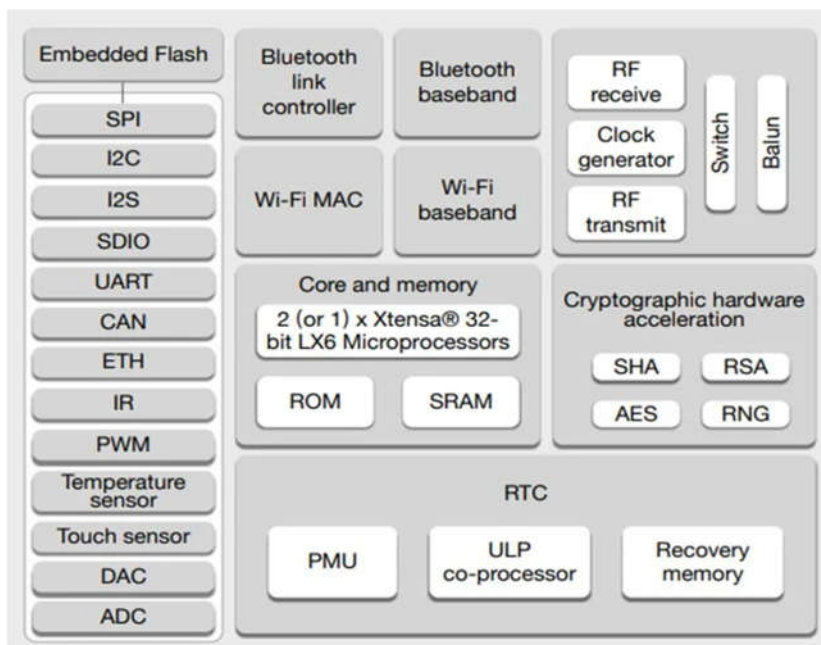
v OTA_DATA partícii (vzorovo OTA_1). Bootloader overuje, či je firmvér na tejto partícii podpísaný, overuje digitálny podpis firmvéru svojim verejným kľúčom (2), ktorý bol do neho vložený v procese kompilácie. V prípade, že je digitálny podpis overený, naboootuje sa daný firmvér.



Obr. 2 Princíp overenia digitálneho podpisu bootloaerom.

C. Zabezpečený štart systému (Secure Boot)

Secure Boot je hardvérová funkcionlita pre zvýšenie bezpečnosti bootovacieho procesu. Využíva digitálny podpis na overenie firmvéru pri inicializácii systému a obsahuje samostatné overenie bootloaera hardvérovým dopočítaním odtlačku (kryptografický kontrolný súčet – hašovací kód), ktorý závisí predovšetkým od šifrovacieho kľúča a obsahu bootloaera [11]. Čip ESP32 na to využíva aj niekoľko integrovaných hardvérových koprocesorov (Obr. 3), ktoré slúžia pre kryptografické operácie a akceleráciu kryptografických výpočtov, napríklad pre realizáciu zabezpečeného spojenia pomocou symetrickej šifry AES (Advanced Encryption Standard), alebo spomínaného výpočtu odtlačku. Medzi dostupnými koprocesormi sa nachádza aj generátor náhodných čísel RNG (Random Number Generator) a nízko-prikonový ULP (Ultra Low Power) koprocesor vhodný pre aplikácie prevádzkované na batériu pri spánku hlavného procesora v ESP32 čipe.



Obr. 3 Bloková schéma čipu ESP32.

Šifrovací kľúč je jednorázovo zapísaný do eFuse (jednorázovo programovateľnej pamäte). Pamäť BLK2 má veľkosť 256 bitov a je určená práve pre šifrovací kľúč secure bootu. Túto metódu je možné implementovať jednorázovým procesom, pričom jej funkcia je trvalá, bez možnosti jej vypnutia. Zapnutie tejto funkcionality je možné v konfiguračnom menu frameworku ESP-IDF. Implementácia secure bootu vyžaduje generovanie šifrovacieho kľúča. Tento kľúč je

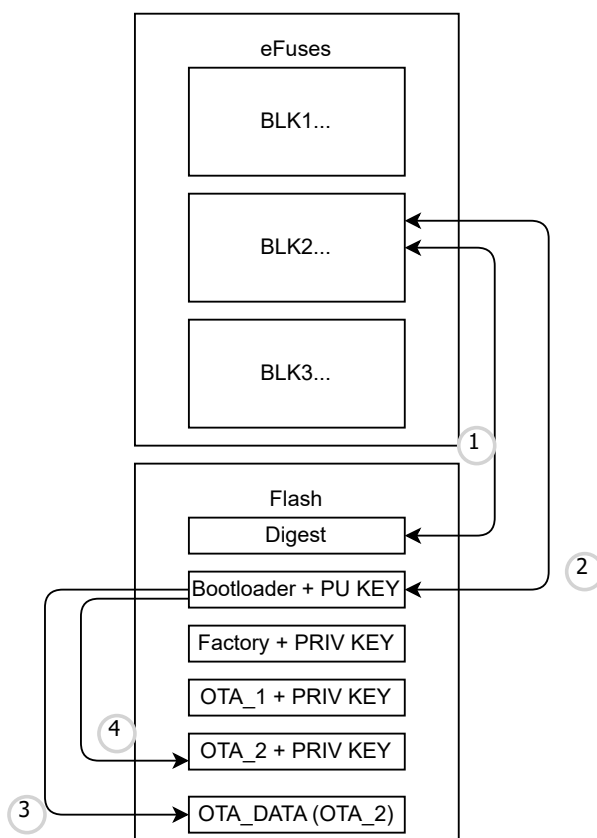
možné generovať Python skriptom `espsecure.py`. Vygenerovaný kľúč má dĺžku 256 bitov a je exportovaný v binárnom formáte, pričom je generovaný pre eliptickú krivku NIST256p.

Šifrovací kľúč je jednorázovo zapísaný do eFuse BLK2 nástrojom `espefuse.py`, ktorý slúži na prácu s dostupnými eFuses. Zápisom do eFuse BLK2 sa automaticky zapne secure boot. Kľúč zapísaný v eFuse BLK2 nie je možné softvérovo prečítať, modifikovať a ani prepisovať. Šifrovací kľúč je ďalej použitý pre vygenerovanie odtlačku. Odtlačok je tvorený inicializačným vektorom, obsahom bootloadera a zo šifrovacieho kľúča secure bootu. Pre generovanie odtlačku je možné použiť nástroj `espsecure.py`. Vygenerovaný odtlačok je zapísaný do pamäte flash s posunom 0x0.

D. Princíp bootovania pri Secure Boote (Obr.4)

Po privedení napájania sa hardvérovo vypočíta odtlačok (1), ktorý je tvorený obsahom bootloadera a šifrovacím kľúčom secure bootu v eFuse BLK2. Výsledný odtlačok je porovnaný s odtlačkom nachádzajúcim sa v pamäti flash (s posunom 0x0). V prípade, že je odtlačok odlišný, alebo chýbajúci, secure boot zablokuje bootloader a nie je možné bootovať firmvér zo žiadnej partície. Ak je odtlačok overený, spustí sa bootloader (2).

Bootloader získava z partície OTA_DATA (3) bootovací príznak (vzorovo OTA_2) a bootuje firmvér z tejto partície. Pred bootovaním je overený digitálny podpis firmvéru na tejto partícii (4). V prípade, že je overený, bootloader bootuje z tejto partície. V opačnom prípade bootloader bootuje z iných dostupných partícií, na ktorých taktiež overuje digitálny podpis pred bootovaním.



Obr. 4 Princíp funkčnosti zabezpečeného štartu (Secure Boot).

IV. ZÁVER

Opísané princípy kryptografickej ochrany a zabezpečenej aktualizácie firmvéru tvoria základ zabezpečeného senzorového uzla, ktorý je vyvíjaný v rámci Diplomovej práce – Bezpečná aktualizácia firmvéru v senzorovej sieti. Práca kombinuje využitie vzdialenej aktualizácie firmvéru spoločne s meteorologickým senzorom BME280, pričom komunikuje so vzdialeným webovým rozhraním pomocou HTTPS protokolu a poskytuje snímané dáta na vizualizáciu a archiváciu. Dôraz je kladený predovšetkým na bezpečnosť, možnosť vzdialenej aktualizácie a využitie možností zabezpečenia firmvéru, ktoré sú dostupné vo frameworku ESP-IDF.

V prípade, že do hardvéru vzdialeného uzla na báze ESP32 niekto nahrá iný neautorizovaný firmvér (napríklad fyzicky prostredníctvom USB-UART rozhrania), pozmení bootloader,

prípadne zmaže obsah pamäte flash, bootovanie nového firmvéru sa nevykoná, keďže odtlačok bude prepísaný, prípadne bude neplatný pri nahratí iného bootloadera. Šifrovací kľúč z eFuse BLK2 je chránený a nie je možné ho prečítať, modifikovať, vynulovať. Útočník tak nevie vytvoriť náhradný odtlačok, ktorý by mu umožnil spustiť neautorizovaný firmvér. Ďalší vývoj je smerovaný do využitia ULP koprocesora s cieľom vytvoriť kryptograficky zabezpečenú aktualizáciu firmvéru v nízkoprikonovom senzorovom uzle s WiFi konektivitou na báze platformy ESP32.

POĎAKOVANIE

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-18-0373 a Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. 1/0584/20.

REFERENCIE

- [1] ESP32 [online] Wikipedia The Free Encyclopedia [cit 2020-04-01]. Dostupné z: <https://en.wikipedia.org/wiki/ESP32>.
- [2] Over-the-air programming [online] Wikipedia The Free Encyclopedia [cit 2020-02-24]. Dostupné z: https://en.wikipedia.org/wiki/Over-the-air_programming.
- [3] Partition Tables - ESP32 [online] ESP-IDF Programming Guide latest documentation [cit 2020-05-17]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-guides/partition-tables.html>.
- [4] Native OTA example [online] Github [cit 2020-04-30]. Dostupné z: https://github.com/espressif/esp-idf/tree/master/examples/system/ota/native_ota_example.
- [5] Get started - ESP32 [online] ESP-IDF Programming Guide latest documentation [cit 2020-05-17]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/>.
- [6] Project Configuration - ESP32 [online] ESP-IDF Programming Guide latest documentation [cit 2020-05-17]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/kconfig.html>.
- [7] Secure Boot - ESP32 (Remote Signing of images) [online] ESP-IDF Programming Guide latest documentation [cit 2020-05-17]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/security/secure-boot-v1.html#remote-signing-of-images>.
- [8] esptool/espsecure.py [online] Github [cit 2020-05-06]. Dostupné z: <https://github.com/espressif/esptool/blob/master/espsecure.py>.
- [9] Dokumentácia [online] OpenSSL [cit 2020-05-17]. Dostupné z: <https://www.openssl.org/docs/>.
- [10] Secure Boot - ESP32 (Signed App Verification Without Hardware Secure Boot) [online] ESP-IDF Programming Guide latest documentation [cit 2020-05-17]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/security/secure-boot-v1.html#signed-app-verification-without-hardware-secure-boot>.
- [11] Secure Boot - ESP32 (Signed App Verification Without Hardware Secure Boot) [online] ESP-IDF Programming Guide latest documentation [cit 2020-05-17]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/security/secure-boot-v1.html#secure-boot-process-overview>.

Enhanced Smart Monitoring of Water Consumption

¹Dávid KARABAŠ, ²Ondrej KAINZ, ³Miroslav MICHALKO, ⁴Rastislav PETIJA,
⁵František JAKAB

^{1,2,3,4,5}Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹david.karabas@student.tuke.sk, ²ondrej.kainz@tuke.sk, ³miroslav.michalko@tuke.sk,
⁴rastislav.petija@tuke.sk, ⁵frantisek.jakab@tuke.sk

Abstract—The aim of this research is to develop experimental software solution for water consumption measurement. First, we analyzed conditions and approaches for recognizing of numerical values from the image. Then, we determined the suitable recognition method and suggested implementation of a device, which has been chosen as a result of the analysis. The experimental implementation includes deployment of the new device and software developed for recognition of numerical values from image. The proposed solution is based on prior research, in comparison with this previous solution, the new device is from a financial point of view more advantageous. This is due to its proportions, which enables more efficient handling. The device has also incomparably lower energy consumption.

Keywords— consumption, intelligent, measuring, micropython, smart, water

I. INTRODUCTION

Water is a limited resource which is essential for agriculture, industry and for the existence of life on Earth, including people. Many people do not realize that getting enough water during the day is basic necessity of life. We are also unaware of the rarity of water and we are wasting that resource uselessly. This problem is slightly related to poor water distribution, inefficient use and inadequate water management. Therefore, the efficient use and monitoring of water is a potential limitation of the water management system in your household or office. Every living organism on Earth needs water for its existence. More than 60% of human body is made up of water. We use clean water for drinking, growing crops, producing food, running factories, swimming, surfing, fishing and sailing. Here we can see that water is vitally important to all aspects of our lives. Nowadays, the amount of water loss in distribution systems is literally indescribable. Water loss causes socio-economic costs to the environment and could have a serious impact on urban infrastructure. Correction and localization of water leaks should be the priority in the management of drinking water systems. Household water leaks are an essential part of the ecosystem that needs to be dealt with. In households, water can leak in small amounts, such as through loose seals or small cracks in pipes. It can also leak in larger quantities, for example in the event of appliance failure or large cracks in the pipes. Such a thing like forgetting to stop the water and leaving the house may happen to all of us. These leaks can affect the functioning of the household. Leakage control would be possible by constantly inspecting pipes and appliances in our house, but it would be time consuming. To detect a water leak, it is necessary to detect the amount of water flowing in the time horizon. Water companies in Slovakia provide solutions that may be appropriate for us, but these are financially demanding and require interventions in the water installation. We will continue in the prior research [1], where such system has been already developed. To monitor the water flow, we will therefore use a solution for image detection and subsequent sending of the results of the read values to the database, where these data will be further processed and evaluated.

II. ANALYSIS OF SOLUTION FOR SMART WATER CONSUMPTION

Research presented in this paper is based on prior research focused on the smart water consumption.

A. Backend

The backend of an existing application is created in PHP Laravel. Laravel is an open source PHP web framework. The application is mainly used for communication between the NoSQL database and the frontend application. The main task of the server part of the application is to

mediate communication through the application programming interface API. Backend application provide API endpoints for communication.

B. Frontend

The frontend application is created in the Angular JavaScript framework. It is used for the user to log in to the application, to display water consumption, to display non-standard situations and to observe water consumption in various parts of the day. The application obtains data from the server application through an API.

C. Hardware solution

Following the research [2], which this research is based on, the Raspberry Pi device was used as a hardware solution. The Raspberry Pi is a low-cost, credit card sized computer that plugs into a computer monitor or TV and uses a standard keyboard and mouse. It is a small device that allows people of all ages to explore computer work and learn to program in languages like Scratch and Python. It can do everything one could expect from a desktop computer, from browsing the Internet, playing high-definition video to creating tables, word processing and programming.

D. ESP32 development boards

ESP32 development boards with a camera are very popular. There are several models with different features such as support of microSD cards, microphone, screen etc. These boards allow developer to create projects with images, video streaming, face recognition and detection and other artificial intelligence applications. The comparison of camera boards is shown in the Tab. 1.

Table 1
 Comparison of camera boards

Boards	ESP32-cam	ESP-EYE	M5-CAMERA
MicroSD card	✓	x	x
Microphone	x	✓	✓
Button	x	✓	x
Battery	x	x	x
GPIOs	10	x	x

E. OpenMV Cam H7

OpenMV Cam H7 is a small board with low power consumption, which is described in the paper [3]. The board makes it easy to implement applications using real-world machine vision. The board is programmed using Python instead of C / C ++.

F. Hardware comparison

Three boards were analyzed: Raspberry Pi, EPS32, OpenMV. All of these boards are appropriate for use in our research. Comparison is based mainly on the functionality, which those boards provide and their price. In the table below you can see a simple comparison, which is helpful in choosing the right board. In Tab. 2 can be seen the comparison of the boards. As an outcome of the comparison, we chose the EPS32-cam board as the most suitable candidate, which has all the necessary features. Based on the comparison, that board is the smallest and can be easily placed even in the cramped conditions. The price of this board is much lower than of the others, which is really important factor when choosing the right board.

Table 2
 Comparison of development boards

Boards	Raspberry Pi 3B	ESP32-cam	OpenMV Cam H7
WiFi	✓	✓	✓
Size	85.60 × 56.5 mm	27*40.5 mm	45*36*30 mm
Camera	x	✓	✓
Flash	x	✓	✓

III. ANALYSIS OF APPROACHES FOR DIGITS RECOGNITION AND THEIR IDENTIFICATION

The water meter that will be used for data collection has a dial that uses the Arabic form of numbers. Those values will have to be identified from the image. Preparatory to identification numbers by using various techniques, the image must be prepared to be sufficiently legible for subsequent identification of the value. In the paper [4], the authors describe the process of identifying characters from a car brand and subsequently recognizing them.

Image processing consists of the basic steps:

- image representation and format,
- filtering,
- improvement and renewal,
- edge segmentation and detection.

A. *Representation of the image and format*

The data we collect or generate is mostly raw data that is not suitable for use in applications for many reasons. Therefore, at first it is necessary to analyze them, perform the necessary pre-processing and afterwards to use them. The image processing described in [5] states that each image is represented by a set of pixels, which means matrix of pixel values. For grayscale images, pixel values range from 0 to 255 and represent the intensity of that pixel. An image with dimensions of 20 x 20, is represented by a matrix of 20x20 (a total of 400-pixel values). There are several image formats, each using different techniques and being specific to different applications. The most common image file formats are: JPEG, JPG, JPE, GIF, PNG, BMP. These formats are also used when printing, scanning and using the Internet.

B. *Filtering*

Image filtering is used to remove disturbances and to lighten contrast. Filters can be classified as linear or non-linear. Linear filters are also known as convolutional filters because they can be represented by matrix multiplication. The assumption of linearity is not fulfilled for nonlinear filters.

C. *Improvement and recovery*

The aim of the improvement is to highlight specific features for subsequent analysis or image displaying. The improvement process is usually performed in interactive way. Renewal is a process that attempts to reconstruct or restore an image that has been degraded by using unknown phenomena.

D. *Edge segmentation and detection*

Segmentation is the process of classifying pixels, whereby an image is divided into subgroups by assigning individual pixels into classes. Edge detection is an image processing technique for finding object boundaries in images. The basic methods for edge detection are Canny and Sobel.

E. *Machine learning*

The machine learning, described in [6], and deep learning play an important role in computer technology and artificial intelligence. With the use of deep learning and machine learning, human effort can be cut down in recognition, learning, prediction and many other areas. In machine learning, a sample of data is needed to train the model, which will be used to recognize other values. The set of numbers has to be prepared manually or freely available set should be downloaded. At this time, there are many publicly available databases that could be suitable for our solution. The paper [7] presents the MNIST database and the Chars74K database. To test the correctness of machine learning, the division of one data set into a training file and a test file is used. It is necessary to create a model that can correctly generalize new data. In machine learning, it is necessary to verify the trained model based on test data.

F. *Optical recognition of signs*

Optical recognition of signs OCR, described in [8], is a process that allows a system to identify expressions or written texts into user verbal communication without human intervention. Techniques of identification or classification of optical signs OCR and magnetic character recognition techniques MCR are generally used to recognize models or alphabets. In the research [2] in which we continue, a comparison was made between several systems for recognizing values. The comparison was between Tesseract, GOCR, MathLab, OCRopus systems. Tesseract was chosen as the most suitable one. Tesseract-ocr Tesseract was developed from 1985, by the Hewlett-Packard (HP) Labs company, until 1995. When recognizing coloured text, it is recommended that you remove colours from the image before recognizing it. Tesseract is written in C and C ++. The application is available on Windows, GNU / Linux and MacOS. Tesseract can recognize 29 different languages and the system can be extended to other languages.

G. *Comparison of value recognition approaches.*

In the previous subchapters, several approaches to value recognition were mentioned. In our case, it is not about recognizing complex ever-changing values. It is a matter of recognizing

constantly recurring values, so a machine learning approach is suitable. That approach will help us to determine the value from a given image with high accuracy.

IV. DETECTION AND RECOGNITION OF NUMERICAL VALUES FROM THE IMAGE

Based on the performed analysis, it is necessary to create a design and then carry out the experimental implementation.

A. Suggestion of hardware and algorithm for value recognition

In the previous section, several available solutions, that are suitable for our project, were analyzed. The ESP32-cam device was chosen as the best hardware solution. To install this device, we must create firmware that will contain all the functionality that is needed to operate the camera module. Urequest package has to be installed on the device, which is required for functionality and is not in the firmware by default. The firmware has to be uploaded to the device. Initially, the device will be connected to the Wi-Fi network using the credentials stored on the device. After the connection, a photo with sufficient quality will be taken and then this photo will be sent to the server, where it will be further processed. To send an image to the server, at first it is important to log in to the server using the /api/login endpoint. If it fails to log in or send a photo, the process will be repeated two more times; if it still cannot send or log in after these attempts, the device will continue.

B. Suggestion of the algorithm for recognizing of numerical values out of the image

The image that we obtain from the server will need to be identified using a script and a numerical value obtained from it. Part of image pre-processing is very important when designing value recognition. When preparing the image, we will use the OpenCV library, with the help of which we will obtain all the necessary digits from the image separately in the correct order. These numbers are then recognized by the KNN algorithm using a trained model. To recognize numerical values, we will create a data set that will help us in testing and with which the model will be trained. The data set will contain a minimum of 10 different sample patterns for each number. Finally, it will be necessary to incorporate the new solution into the existing solution on the server. Where we use this solution as a function that gets the path to the image at the input and returns the value it read from the image. The read value will then be stored in the database on the server.

C. Hardware installation and image capture

The Raspberry Pi device was chosen as a hardware solution for taking a photograph of the water meter, that device was replaced by the ESP32-cam device after a thorough analysis of the hardware solution. We opted for MicroPython, which is an effective implementation of the Python 3 programming language. MicroPython has similar libraries as Python 3. In our case, it was necessary to modify an existing Raspberry Pi solution that uses Python 3 since the ESP32-cam supports MicroPython. To use MicroPython, it was necessary to create firmware that will support the control of the camera, as this firmware is not the base of the provided firmware for various devices from MicroPython. We created our own firmware that supported the usage of the camera and set the necessary configuration for the camera. The resolution 1024x768 was selected and we set the image quality to the highest.

When working with the device, we used esptool.py. It is an open source tool for communicating with the ROM bootloader in Espressif ESP8266 and ESP32 chips. With it, we can upload new firmware and new files to the device, with which we can program the device. We can also connect to the device using an interactive command line, which allows us to work with the device in real time. We used uPyCraft, a MicroPython development environment, to program the device. The uPyCraft application is publicly available. This application allows us to easily upload new changes to the code on the device or run pieces of code on the device using the user interface.

The functionality for sending an image to the server was added. We have also created functionality that processes the image and prepares the data to send the image using the urequests library. The ESP32-cam has the very bright built-in flash we need. The flash is internally connected to the GPIO 4 pin. We will use the built-in flash when taking a photo of the water meter. To connect the ESP32-cam to a computer, we need a connector that allows us to connect. We will use FT232RL FTDI USB 3.3V 5.5V to TTL Serial Adapter Module.

D. Communication with backend

After the photo is received by the server, the photo is saved in a user-specific directory. The server runs the cron software utility, which is a task planner. The task defines a task that runs every n minutes. This task is used for identification of new photos. If a new photo is identified, a function is started, which has the task of loading given photo, processing it and identifying

numerical values on it. After identification, the values are stored in a database and then it is possible to view the values in a frontend application that communicates with the database.

E. Preparation of the image

Python language and `cv2` library (OpenCV) were used for image preparation. We carried out the preparation in several steps. It is important to place the device so that the photo is as good as possible. Since there are several types of water meters, we focused on one specific type. To get started, we used a function to change the color of the image on the base image and changed the image from colored one to grayscale. We used a function from the `cv2` library. We also applied smoothing to the image. Smoothing is a simple and frequently used image processing operation. There are many reasons for smoothing. In our case, we focused on smoothing to reduce image inaccuracies. After applying the smoothing, we detected the edges in the image using the Canny function from the OpenCV library. This function has three parameters that need to be set. The first one is the image on which we want to detect the edges, the other two are the minimum and maximum threshold. The values 105 and 170, which were the most suitable for implementation and testing, were used for the minimum and maximum threshold. Next, the contours in the figure were searched using the `findContours` function and the main contour of the number column was identified by setting the length and width of the contour. Based on this outline, the image was then cropped and assigned to its own variable for further work with it.

This image section was then further processed. In similar steps as in the first phase, a filter was applied to it to identify edges with other parameters. After the edges were identified, the outlines were searched for in the picture so that we could easily separate each number separately. After finding the contours, the contours of the given characters in the figure were identified and they were cut into separate figures. We applied an adaptive thresholding method to the cut-out numerical values. This method returns an image adjusted to the thresholds. The image was further modified by swapping white and black pixels.

F. Content recognition

We then needed to read the value from the prepared image. The way we chose to do this is machine learning.

To train the model, it was necessary to create our own dataset. We created the dataset using an image editing application. The numerical values we needed to recognize are as follows: 0123456789. Therefore, for each of these values, we created n variants of images, in which case each image looked different. We decided to use 14x21 pixels as the image size of each number, this size was optimal with respect to the necessary digits. The number name was saved in the form 0.1.png, which means the value of the number in the image. This nomenclature of numbers further serves for their further processing.

The `KNeighborsClassifier`, which achieved the best results in testing, was used for training. A prepared data set was used for training, where the image data were adjusted for better recognition before training. The `numpy` library was used to read the data, where the data was read as an array and assigned to the `data_X` variable. Each image was loaded as a two-dimensional array, where each row contained n pixels. The fields were converted to one-dimensional fields before being assigned to the variable, where each subsequent row of pixels of one row was assigned to the pixels of the previous row, unless the previous row was the first row.

The images modified in this way are then assigned to the field of all images and this field is assigned to the above-mentioned variable `data_X`. It was also necessary to add the values of the given images to the image data. These values were assigned to a separate field and then assigned to the `data_Y` variable. The values of these images were parsed from the image name we created in the previous chapter. This helped us to easily identify the correct values. Using the prepared data, we trained the model and saved this model in the SAV format, which is used to save the trained model.

The K-nearest neighbors algorithm was used for recognition. The pre-trained model had to be uploaded to the server. This model was loaded and subsequently the function was run over the modified image, which we mentioned in the chapter above. As an output, we received a single-digit sheet that represents the number that the trained model selected as the best match from the samples. This recognized sample was then read and sent to the database with the correct time data, which was parsed from the file name of the photo.

V. EVALUATION AND TESTING THE APPLICATION

The newly designed device was placed on a water meter. That new solution is smaller, much easier to install and does not take up so much space. Hardware testing was successful when the device managed to connect to the Wi-Fi network. The device then communicated with the server,

took a photo and sent it back to the server. Compared to the first design, the current solution cannot produce such high-quality photographs as the first device, but the quality is sufficient to recognize the values. The current solution consumes much less energy than the first hardware design. The device was tested for a shorter time, where we were able to create images and send them to the server. The processed data from the created images were then stored in the database and displayed on the dashboard.

The images were recognized on the server and the values were written to the database, which we could also see in the frontend application. The recognition of numerical values from three images was tested in the production, in Tab. 3 you can see the test results and the numbers that were read. Several samples were tested locally, with a success rate of 80%, and approximately 40 images were tested. The numbers that failed the subtractor were numbers that were in between values. From these values we received incorrect values.

Table 3
Test results

Expected value	Recognized value	State
00143906	00143906	Success
00143907	00143907	Success
00143908	00143908	Success

VI. CONCLUSION

These days, it is extremely important to innovate data processing approaches, especially in areas such as water or gas deduction. There is at least one such facility in each household, thus this research can be very beneficial for society and can save a lot of time and money. In this research, we focused on measuring water consumption using the ESP32-cam device, where the values from the image were recognized and written to the database and displayed in the frontend application. While working on intelligent measurement of water consumption, we dealt with several issues. Prior to the actual design and implementation, it was necessary to analyze the current state of the solution for measuring water consumption, which was created by previous solvers. In this step, we came to several deficiencies. The hardware analysis was successful because we were able to find a better option that was more beneficial in terms of price, size, consumption and complexity. ESP32-cam was chosen instead of Raspberry Pi. Raspberry Pi functionality has been rewritten from Python to MicroPython and transferred to the new device. We came to the conclusion that the functionality for reading the numerical value from the image was not functional and needed to be replaced. An analysis was performed to recognize numerical values from the image, where the most suitable method was selected. We performed the recognition of values using machine learning, where the model was trained on a data set prepared by us. The solution design was implemented and tested in a real environment. The functionality of the application was successful during testing. The newly designed device worked properly and sent the image to the server. This image was correctly recognized, the values processed and stored in the database. The system is ready for use and possible innovative solutions have been proposed that could improve quality.

ACKNOWLEDGMENT

The research work described in that paper was supported by the project MIDIH “Manufacturing IoT Digital Innovation Hubs for Industry 4.0”, H2020-767498.

REFERENCES

- [1] O. Kainz, E. Karpel, R. Petija, M. Michalko, F. Jakab. „Detection of non-standard situation in smart water metering”. In: IEEE 15th International Scientific Conference on Informatics. 2019.
- [2] M. Dujava. “Meranie spotreby vody založené na spracovaní obrazu”. Dipl. pr. Slovensko: Technická univerzita v Košiciach, 2018.
- [3] I. Abdelkader. Low Power Deep Learning on the OpenMV Cam powered by the Cortex-M7 Processor. <https://openmv.io/blogs/news/deeplearning-on-a-cortex-m7-camera-3ma-deep-learning>. Accessed: July 2018.
- [4] K.M. Sajjad. “Automatic license plate recognition using python and opencv”. In: Department of Computer Science and Engineering MES College of Engineering, 2010.
- [5] I. Pitas. Digital image processing algorithms and applications. John Wiley & Sons, 2000.
- [6] R. Kraft. “A machine learning approach to improve precision for navigational queries in a Web information retrieval system”, 2002.
- [7] H. Kusetogullari et al. “ARDIS: a Swedish historical handwritten digit dataset”. In: Neural Computing and Applications, pp. 1–14, 2019.
- [8] F. Ribemont. “Research Document Number Plate Recognition”. Research Document, 2011.

Use of non-traditional teaching methods in the subject Data Structures and Algorithms

¹*Monika KUŽMOVÁ*, ²*Slavomír ŠIMONÁK*

¹Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹monika.kuzmova@student.tuke.sk, ²slavomir.simonak@tuke.sk

Abstract – In this paper we analyze non-traditional methods in education of data structures and algorithms. Based on the analysis, a web application is designed and implemented, created using the Electron.js framework, which includes selected topics from the subject Data Structures and Algorithms. The application itself is divided into three main parts - visualization, practicing and questions. Selected topics are first described and visualized, followed by a practical test of the behavior of individual algorithms and verification of the acquired knowledge through a series of questions. We carry out the evaluation using a questionnaire, which is filled in by users after testing the application. The results suggest that we have created an educational material that will make the teaching of the subject more efficient and attractive.

Keywords – algorithm, data structures, JavaScript, visualization, web application

I. INTRODUCTION

Algorithms and data structures are an important part of teaching computer science at universities around the world. The Technical University of Košice, Faculty of Electrical Engineering and Informatics offers teaching the subject of data structures and algorithms in the second year of bachelor's studies in the field of Informatics. Understanding the data structures and the large number of algorithms that students encounter in the first years of study is a basic step in further education in the study program Informatics. Due to the high level of abstraction and the difficulty of mediating the dynamic aspects of algorithmic processes using only texts and images, several interactive exercises, visualizations, simulations and even games have been developed over the years to support the teaching of data structures and algorithms. These non-traditional forms of teaching provide a higher level of student engagement with the support of practical testing of the operation of algorithms and data structures. The motivation for creating this work was to create an educational material for the next years of students, which will increase students' interest in the subject, allow them to interact directly and facilitate their understanding of data structures and algorithms. The application was originally introduced in the thesis [1].

II. UNCONVENTIONAL METHODS IN EDUCATION

Information technologies have become a part of our lives and are part of the educational process. In recent years, there has been an increase in interest in unconventional forms of teaching in schools, and new technologies and interactive aids are being used to make teaching more attractive. Unconventional teaching methods include the use of interactive tools such as interactive whiteboards and information technologies in the form of various programs, applications, visualizations and games that increase students' interest in the teaching process.

A. Visualizations of algorithms and data structures

Visualization is a technique of creating images, diagrams or animations. Visualization in the form of pictures or animations is used in teaching in primary schools. Nowadays, a typical area of application of visualization is computer graphics [2]. Visualization helps us to easily simulate functionality, processes and even individual steps of process. The easiest way to visualize algorithms is to use flowcharts. The flowchart shows the individual steps of the algorithm using

various geometric shapes, such as rectangles and rhombuses, these shapes are interconnected by arrows. Over the years, many tools, programs and websites have been created to visualize the behavior of algorithms.

VizAlgo is a program for visualization of algorithms, which was developed as a support tool for teaching the subject Data Structures and Algorithms [3]. The main goal of this program is to support practical forms of education and increase students' attention and interest. The program is created using the Java programming language, so it can be run on any device that supports Java. The user interface of the program is relatively simple and intuitive. The environment consists of the following panels: input panel, algorithm pseudocode, algorithm visualization, information and control panel. The user selects an algorithm in the input panel and enters parameters if it is necessary. The user controls the visualization using the control panel.

Algomaster is a visualization platform developed using the .NET framework and the C# programming language [4]. The user chooses an algorithm in the menu. On the left is the pseudo-code and on the right is the visualization itself. The Start, Step Back, Play and Step Forward buttons are used to control the visualization. Compared to VizAlgo, Algomaster also allows step back.

OpenDSA is a project to support the teaching of data structures and algorithms, which includes graphical primitives, data structures, pseudocode display, visualization and simulation of algorithms [5]. Simulations allow you to set the execution speed as well as step-by-step animation control using the buttons. The goal of OpenDSA is to include the teaching activities that engage and motivate students, which will facilitate the teaching of the subject. The OpenDSA project provides visualization of dynamic processes and feedback, so users receive a response to whether the exercise was performed correctly with the ability to show the correct solution and a hint that describes the issue. Thus, students are more engaged and learn through practice. OpenDSA consists of a series of modules, which correspond to chapters in regular textbooks or lectures in this area [6]. OpenDSA is an open source project, and thus these learning materials are freely available and can be used in online courses or can also be used as part of content in learning management systems. OpenDSA can be used standalone, but it also supports data storage, where students' progress is recorded after logging in and test results are stored.

JSAV - The JavaScript Algorithm Visualization Library is an open source library to support the development of algorithm visualizations written in JavaScript with HTML5 support. JSAV contains a special set of functions designed to support exercises aimed at algorithms visualizing [7]. This library includes active learning techniques where the visual context can be easily integrated into online sites. JSAV has combined the best of existing algorithm visualization systems. JSAV supports various types of visualizations, from simple static images of data structures to complex animations. These animations show the steps of the algorithms' behavior. The user can control the animation by moving one step forward or backward. JSAV also supports the use of simple questions as well as multiple-choice questions. JSAV consists of three types of objects: data structures, graphical primitives, and code. Supported data structures are array, linked list, tree, binary tree, and graphs. JSAV also supports visualization of pseudocode and variables.

B. Gamification

In the 21st century, playing computer games is a part of spending free time, whether children or adults. Visualization is an integral part of computer games. When playing computer games, positive effects on visual attention, coordination, motor skills and sensory abilities were demonstrated [8]. In connection with these findings, the idea arose to use computer games in the educational process. These games are called educational games. Educational games have great potential in supporting learning. As a result of the high interest in this form of education, many educational games have emerged simulating the learning process in areas such as biology, geography, computer science, mathematics, health care and languages. Educational games vary from simple puzzles to complex online games. Active learning and educational games are very widespread in the field of algorithms and data structures.

The Stack Game [9] is one of the games developed for students to understand and practice data structures and algorithms. The game focuses on three aspects of teaching data structures: conceptualization, application, and implementation. The first part of the game helps students understand the principles of the stack by arranging the colored blocks in the correct order. The

second part of the game consists of three levels and its goal is to solve problems such as converting arithmetic expressions from infix to postfix form. The third part of the game focuses on the implementation of the stack, in this part it is necessary to properly arrange the code or complete the missing parts of the code. The game also includes a tutorial, which is available while playing the game.

Heapsort digital game [10] was created to support the teaching of sorting algorithms in a fun way. The game simulates heapsort algorithm steps, so the player must arrange the field values as a heapsort sorting algorithm would do. The player has a start button, a value exchange button that exchanges values between father and child, and a delete root button, which is used to exchange the first and last element of the field. The game also exists in the board game version.

III. DESIGN AND IMPLEMENTATION OF THE SOLUTION

The created web application combines visualizations, exercises to practice and verification of the acquired knowledge. Although the web application will primarily serve students of the subject Data Structures and Algorithms, as only a web browser is required for the operation of the web page, access to the materials will not be restricted in any way. The web application focuses on selected topics that are part of the subject data structures and algorithms:

- binary tree traversal,
- ternary tree traversal,
- binary search tree,
- AVL tree,
- 2-3 tree,
- sorting algorithms.

We divided the application into three main parts - visualizations, exercises and questions. The task of visualizations is to show to the user the dynamic aspects, and to show the behavior of either the binary tree traversal, the balancing of the AVL tree, the addition of nodes to the tree or the process of executing sorting algorithms. Part of the visualizations is the pseudocode display and accompanying texts, which approach the topic and describe the performed steps of visualizations. The second part of the web application is used to practice the acquired knowledge from the visualization part. The aim of this section is to elucidate behavior of algorithms to the user and consolidate the acquired knowledge. In this part, the user directly manipulates individual visualizations, adds algorithm steps to the prepared fields and partly creates its own visualizations. This section includes various tasks to practice the behavior of algorithms and binary tree traversal. Question part is used to verify the acquired knowledge. After answering questions, user score will be displayed.

To implement the web application, we used the framework for creating web applications Electron.js and a JavaScript library to support visualizations - JSAV. Since the JSAV library was created to support the teaching of data structures, in the application we will use data structures, methods for manipulating and modifying the appearance of these structures, functions for working with code, and functions for creating exercises that this library offers.

A. Visualizations

In the visualization part, we focus on displaying the individual steps of the algorithms. Visualization of the binary tree traversal displays step by step strategies inorder, preorder, postorder, and level-order. To visualize the binary tree, we used the data structure of the JSAV library - binarytree. The ternary tree traversal visualization shows the ternary tree traversal using the preorder and postorder strategies. Both visualizations contain a description of the topic, a visualization and the pseudocode of the given strategy. The AVL tree visualization shows step by step insertion of nodes to the tree and removing nodes from the tree, followed by self-balancing of the tree if necessary. Visualization of the 2-3 tree shows step by step insertion of nodes and removing nodes from the 2-3 tree using the method of assigning elements to the leaves of the tree as a dictionary. We used the tree data structure to create 2-3 trees. The visualization includes a description of the topic and pseudocode of the insert and delete operations and search and addson procedures.

Visualizations of sorting algorithms show sorting algorithms - bubble sort, selection sort, insertion sort, quick sort, merge sort and heap sort. Visualizations of sorting algorithms are displayed using an array. We used JSAV library data structure - array to create the array. The array contains elements of different values. The algorithm then sort this array. The bubble sort, insertion sort, and selection sort visualizations allow the user to enter eight numbers, which are then sorted. The visualization contains buttons that allow user to move the visualization forward and backward, return to the beginning of the visualization, and the ability to go to the end of the visualization.

B. Exercising

Binary tree traversal exercises are focused on practicing inorder, preorder and postorder strategies. The user's task is to simulate the order of visitation of individual nodes by a given binary tree traversal strategy by clicking on nodes. The exercise is created using the JSAV library function `exercise(modelSolution, reset, options)`. Function `modelSolution` contains the correct exercise solution. If necessary, the user can use the option to view the correct solution to the task by clicking the Model Answer button and use the Reset button to reset the exercise to its initial state. If the user performs an incorrect step, a window will appear on the screen notifying that the step was incorrect, and the number of points lost will increase by one. The exercise includes a counter of points gained, the number of total points and the number of lost points.

Exercises focused on bubble sort, insertion sort and selection sort were created using the JSAV library function for creating exercises. These exercises contain a randomly generated array of elements. The user's task is to sort the array by simulating the steps that would be performed by a given algorithm. In the exercises focused on bubble sort and insertion sort, the user simulates the steps of the algorithm by clicking on an element in the array. This click is handled by the `clickHandler(index)` function, which exchanges the position of the clicked element with the next element. In the exercise focused on selection sort, the user clicks on the element, the element is highlighted, then clicks on the element with which he wants to exchange it. This exchange is also handled by the `clickHandler(index)` function, which ensures the exchange of elements. The exercises include buttons reset and a model answer. The model answer button displays the correct procedure for executing the algorithm in the form of steps that the user should perform. If the user performs an incorrect step, notification is displayed on the screen. The exercises include a display of the number of points gained, the total number of points, the number of points that the user can still gain and the number of lost points.

The merge sorting exercise contains an array of eight unordered elements. Subsequently, we simulated the steps of the merge sorting algorithm and displayed temporary arrays on the screen, which were created according to the algorithm by gradually dividing the main array into arrays of size one. The student's task in this exercise is to gradually simulate the steps of merge these arrays. The student has prepared fields in which elements in proper order will be inserted in a manner that the algorithm itself would do it. As you can see in figure 1, the student response section is displayed similar to the merge sorting visualization. The supplementary part for the user is thus divided into six temporary fields. To verify the correctness of the input numbers in the individual lines, the user can use the buttons *Prvý riadok*, *Druhý riadok* and the button *Skontrolovať*, which verifies the correctness of all entered numbers.

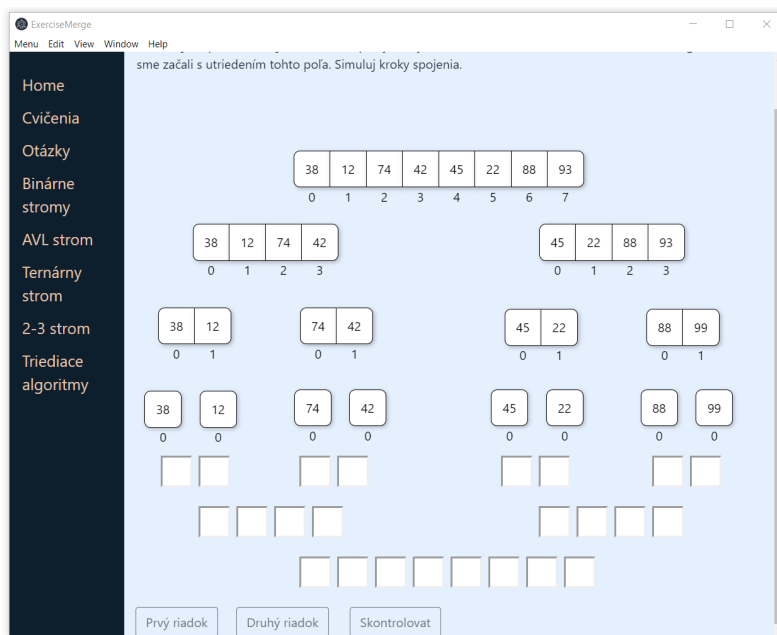


Fig. 1 Exercises focused on merge sort.

The goal of the exercise focused on code completion is to allow users to construct their own visualizations. The screen shows a description of the algorithm and part of the pseudocode.

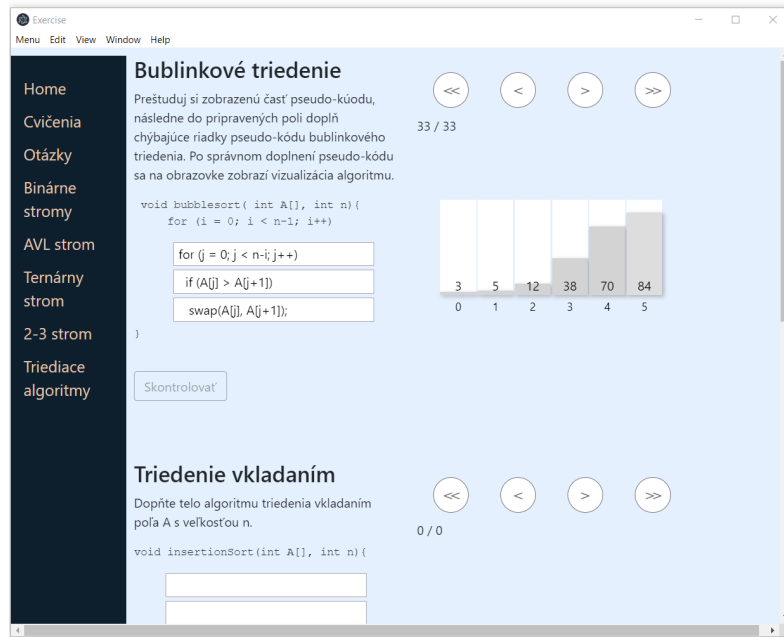


Fig. 2 Exercises focused on code completion.

Furthermore, there are fields for user input on the screen, into which the user input the missing parts of the pseudocode. Visually, the page is divided into two halves. On the left is a description of the algorithm and the fields for user input. On the right side, after filling in the missing text and clicking on the check button, a visualization will be displayed (figure 2). Exercise consists of three parts, the first part of the exercises aimed at complementing of bubble sort pseudocode, the second part focuses on complementing of insertion sort pseudocode and the third part focus on complementing of selection sort algorithm pseudocode. The user enter the missing pseudocode lines to the prepared input fields. After completion all the fields, the user uses the check button. If the user does not enter anything or if input does not correspond to the expected code, a warning will be displayed on the screen stating that the entered code is incorrect. As it is our effort to allow the user to visualize entered code, the code checking functions check the occurrence of the expected parts of the code in the user's input. If the user input corresponds to the expected input, the user input is subsequently modified to correspond with the JavaScript code of the algorithm. The modified code is then visualized and displayed to the user.

C. Questions

Part of the questions includes questions focused on the topics of binary tree traversal, ternary tree, AVL tree, 2-3 tree and the sorting algorithms that were covered in the previous sections. In this section, there are questions to choose from. Furthermore, there are visualizations, which task is to help the user answer correctly the questions to which these visualizations are related. There are also examples of pseudo-code. All questions are scored and at the end after pressing the check button, the points are calculated, and the obtained score is displayed. In addition to listing the score, the correct answers are then highlighted on the screen.

IV. EVALUATION

To find out the benefits of our application and the opinions of users, we created a questionnaire. The questionnaire contained 12 questions and one sub-question, which followed the answer to the previous question, these questions were answered by a total of 44 respondents aged 18 to 25 years. The questions were focused on the functionality of the application, on the usefulness of the application in teaching data structures and algorithms, but also on the overall opinion of users on the unconventional method of teaching and the opinions of users on the application.

91% of respondents stated that they were interested in this type of teaching material. 34% of respondents would choose this form of teaching if they had a choice and 11% of respondents said they would choose a non-traditional teaching method that would include our application. 48% of respondents stated that they would choose teaching in the form of an educational game. The answers to this question show that up to 93% of respondents would choose a non-traditional form of teaching if they could choose.

70.5% of respondents said that visualizations of binary tree traversal strategies helped them better understand how they behave, and 68% of respondents said that visualizations of sorting algorithms helped them better understand how they behave. These values confirm the importance of visualization and confirm our opinion that visualization is a fundamental form of engagement.

75% of respondents said that the exercises for the operation of algorithms helped them and that they were successful in them. 64% of respondents said that the pseudocode completion exercise helped them better understand the workings of sorting algorithms. 60% of respondents when asked what interested them most about the application stated exercises. 77% of respondents agreed with the statement - these exercises helped me to better understand the topics and only 9% of respondents did not agree with this statement. From these obtained values it follows that practicing the operation of individual algorithms, and thus part of our application focused on practice, is an important part of the learning process. We found that the respondents positively evaluated the exercises in which they could react, make changes and construct visualizations. In these exercises, we included several levels of engagement - viewing, responding, changing and constructing.

V. CONCLUSION

From the results of the questionnaire, we concluded that the use of our web application in the process of teaching data structures and algorithms would increase the attractiveness of the subject, and thus increase students interest in the subject, which would ultimately lead to improved student performance in this subject.

As the created application focuses on selected topics from the subject Data Structures and Algorithms, our goal in the future would be to include additional topics that are part of this subject. In the future, we would also like to add login to the application, which would simplify testing students from the available topics. And since currently the application interface is available in Slovak only, we would consider the English version too.

REFERENCES

- [1] M. Kužmová, *Application of Unconventional Methods in Education of Algorithms and Data Structures*. Masters thesis, Faculty of Electrical Engineering and Informatics, Technical University of Košice, 2020 (in Slovak).
- [2] T. A. Duchowski, Andrew T.; Davis, "Teaching algorithms and data structures through graphics," *Proc. of Eurographics.*, 2007.
- [3] S. Šimoňák, "Algorithm visualization using the vizalgo platform," *Acta Electrotechnica et Informatica*, vol. 13, no. 2, p. 54, 2013.
- [4] S. Šimoňák and M. Benej, "Visualizing algorithms and data structures using the algomaster platform," *Journal of Information, Control and Management Systems*, vol. 12, no. 2, pp. 189–201, 2014.
- [5] C. A. Shaffer, V. Karavirta, A. Korhonen, and T. L. Naps, "OpenDSA: beginning a community active-ebook project," in *Proceedings of the 11th Koli Calling International Conference on Computing Education Research*. ACM, 2011, pp. 112–117.
- [6] V. Karavirta and C. Shaffer. (2019) How to use this system. [Online]. Available: <https://opensa-server.cs.vt.edu/ODSA/Books/CS3/html/Intro.html>
- [7] V. Karavirta and C. A. Shaffer, "JSAV: the javascript algorithm visualization library," in *Proceedings of the 18th ACM conference on Innovation and technology in computer science education*, 2013.
- [8] U. Ritterfeld, M. Cody, and P. Vorderer, *Serious games: Mechanisms and effects*. Routledge, 2009.
- [9] D. Dicheva and A. Hodge, "Active learning through game play in a data structures course," in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 2018, pp. 834–839.
- [10] P. E. Battistella, C. G. Von Wangenheim, A. Von Wangenheim, and J. E. Martina, "Design and large-scale evaluation of educational games for teaching sorting algorithms." *Informatics in Education*, vol. 16, no. 2, pp. 141–164, 2017.

Strojové Učenie v Identifikácii Klamlivých Správ v Online Diskusiách

¹Kristína MACHOVÁ, ²Lucia VARGOVÁ

^{1,2} Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹kristina.machova@tuke.sk, ²lucia.vargova99@gmail.com

Abstract — Práca je zameraná na problematiku detekcie zvolených typov anti-sociálneho správania v online komunitách. Sústreďuje sa na detekciu falošných správ a spamu, ktorý má znaky poplašných správ. Boli použité metódy strojového učenia, konkrétne Naive Bayes, metóda podporných vektorov a náhodné lesy rozhodovacích stromov. Pomocou týchto metód boli generované modely na detekciu falošných správ a hoaxov. Získané modely boli testované použitím obvyklých mier efektívnosti klasifikácie.

Keywords — online médiá, antisociálne správanie, falošné správy, hoax, metódy strojového učenia

I. UVOD

Online médiá ovplyvňujú miliardy ľudí pomocou komunikačných kanálov. Predstavujú obrovský zdroj informácií a taktiež umožňujú používateľom komunikovať s inými používateľmi a tak si vytvárať priateľstvá, rozvíjať poznatky v rôznych oblastiach, zdieľať fotografie, videá, články a myšlienky. Aj keď sú sociálne médiá veľkým prínosom v dnešnej technologickej dobe, sú ohrozené antisociálnym správaním vo forme Trollingu, falošných správ, spamov či hoaxov, ktoré predstavujú každodennú hrozbu.

Sociálne médiá slúžia hlavne na združovanie a rozvíjanie spoločenských kontaktov. Výhodou je, že konverzačný obsah môže tvoriť, upravovať alebo komentovať každý zaregistrovaný používateľ. Taktiež poskytujú priestor pre kreativitu a nápady samotných používateľov, ktoré sa následne ľahko šíria medzi ostatnými používateľmi. Najdôležitejšou vlastnosťou kanálov sociálnych médií je povzbudenie používateľov k interakcii a vytváraniu svojho vlastného obsahu[1]. Sociálne alebo takzvané online médiá môžu mať rôznu formu ako sociálne siete, blogy, wiki stránky, chatovacie miestnosti, Social Bookmarking a podobne [2].

Aj keď online médiá prinášajú do nášho života veľa pozitív, sú tu aj znepokojujúce negatíva, ako antisociálne chovanie. Za asociálne správanie, sa považuje správanie nezodpovedajúce spoločenským pravidlám či normám spoločnosti, v ktorej sa jednotlivec alebo skupina pohybuje. Vznik, výskyt takéhoto správania je podmienený pôsobením veľkého množstva vplyvov: psychologické, politické, sociálne, kultúrne a taktiež osobnostné faktory. Asociálne správanie sa prejavuje vo forme nesprávnych informácií ako sú fámy, falošné či poplašné správy, falošné hodnotenia. Tie, sú zámerne vytvorené na maskovanie pravdy. Ich výskyt má tendenciu sa zvyšovať okolo populárnych tém. Asociálne správanie sa môže vyskytovať v rôznych formách ako napríklad trolling, spamming, hating, flaming, griefing, falošné správy a podobne. Rozsah spomenutých foriem antisociálneho chovania sa zväčšuje vďaka narastajúcemu záujmu využívania webových technológií. V minulom roku bolo zaznamenaných 8-10% falošných účtov na sociálnych sieťach, približne 16% falošných recenzií a 40% používateľov bolo obťažovaných inými používateľmi. Asociálne správanie je podporované anonymitou problémových používateľov.

II. TYPY ANTISOCIÁLNEHO CHOVANIA

A. Trolling a Falošné Správy

Trolling predstavuje obťažujúce chovanie v diskusiách a na chatoch, ktoré môže mať formu nevhodných príspevkov alebo neopodstatnenej kritiky. Za trolla sa považuje ten používateľ, ktorý je obvykle anonymný a zámerne kritizuje a vyvoláva hádky a svojím spôsobom otravuje ostatných členov v diskusií. Hlavným cieľom takéhoto používateľa je vyprovokovať ostatných používateľov k emotívnym odozvám, prípadne inak narušiť diskusiu. Dôvod takéhoto asociálne

chovania je ťažko definovať, a však často krát je dôvodom fádnosť, frustrácia z ich osobného a spoločenského života. Na základe spôsobu vyjadrovania vieme definovať rozdielne typy trollov. Najčastejším typom trolla je užívateľ spôsobujúci chaos a horlivé diskusie. Pridávané príspevky sú zamerané na vzbudenie určitej ostrej reakcie, čaká v úzadí čo jeho príspevok medzi ostatnými užívateľmi vyvolá. Je apatický voči dôsledku svojho konania, či vyvolaná diskusia je konštruktívna alebo deštruktívna. Ďalším typom je používateľ, ktorý vďaka tomu, že internet poskytuje istý stupeň anonymity odkrýva svoju temnú stránku, v podobe rasistických, náboženských, homofónnych útokov. Existujú používatelia, ktorých zámerom je vyvolať výmenu názorov na úkor zábavy. S narastajúcim stupňom horlivosti diskusie, sa zväčšuje potešenie z dosiahnutých reakcií iných účastníkov zapojených do debaty. Posledným typom sú trolly, platení politickými organizáciami. Ich cieľom je obťažovať ostatných diskutujúcich dehonestovaním ich politických názorov na kandidátov alebo manipulovať ich názory.

Falošné správy predstavujú vymyslené, nevyžiadané správy alebo pseudo-správy. Tieto správy sú považované za vynález ľži vytvorenej z ničoho, ktoré sa javia ako skutočné správy. Nepravdivé informácie pozostávajú z úmyselnej dezinformácie, sú účelovo vytvorené na maskovanie pravdy tak, aby širokú verejnosť uviedli do omylu. Dezinformácia je šírená rôznymi spôsobmi ako sú napríklad aj tradičné médiá, medzi ktoré patrí tlač a vysielanie, no najrýchlejšie sa správy tohto charakteru šíria na sociálnych médiách. Falošné správy môžu mať rôzne podoby, ako napríklad:

- Falošné správy z politiky, ktorých relevantnosť sa v spoločnosti rapídne zvýšila. Média majú schopnosť prilákať viac voličov. Dopady takýchto falošných správ na spoločnosť môžu byť vážne. Môžu výrazne ovplyvniť voličov pri hlasovaní v politických voľbách.
- Veľkolepé falošné správy, ktoré sú zdieľané len preto, že sú senzačné. Príkladom bol aj titulok o automobilovej nehode údajne spôsobenej hraním známej mobilnej hry PokémonGo. V tomto prípade autori použili skutočnú fotografiu, ale dali jej názov, ktorý nemá nič spoločné s realitou.
- Falošné správy, ktoré sa snažia niečo predať vyskytujúce sa na platformách elektronického obchodu. Buď sú to recenzie uvedené v krátkom časovom období s vysokým hodnotením alebo tak zvané „zázračné“ správy zaplavujúce internet o produktoch so zázračnými účinkami, napríklad prípravky na chudnutie.
- Falošné správy propagujúce rasistické myšlienky. Tieto publikácie prispievajú k udržaniu strachu z iných rás alebo etník. Namiesto diskusie či dialógu slúži tento druh správ na rozduchovanie neznášanlivosti.

B. *Spamming a Poplašné Správy*

Za *spam* sa v súčasnosti považuje hromadná a nevyžiadaná správa cez mobil, sociálne siete alebo rôzne fóra. Často nám neumožňuje odhlásenie z odberu cez odkaz, ani žiadosťou o zrušení zasielania takýchto správ. Správy, ktoré požadujú osobné údaje patria medzi najnebezpečnejšie spamy. Medzi nebezpečné typy nevyžiadaných správ patria:

- Spoofing – správy, v ktorých má odosielateľ falošnú identitu. Príkladom sú aj e-mailly, v ktorých sa ako odosielateľ objaví vaša emailová adresa.
- Phishing – činnosť podvodných používateľov, so snahou vylákať heslá k bankovému účtu alebo heslá k e-mailovým adresám. Pováčšine sa phishing vykonáva na webových stránkach, ktoré sú len kópiou originálnej stránky. Vo forme emailov používateľom oznamujú zmenu účtu, prípadne nutnú aktualizáciu.
- Scamming - taktiež zvané aj ako „Nigérijské e-mailly“. Najčastejším obsahom stránok sú lukratívne ponuky podnikania zo zahraničných krajín. Používateľ je nútený zadať svoje osobné údaje na založenie účtu, kde by sa mali peniaze uložiť aby používateľ mohol podnikat' v inej krajine. Obete takýchto podvodov sú vylákané na zaplatenie administratívnych poplatkov.
- Hoax – poplašná správa varujúca pred neexistujúcim nebezpečenstvom. Text správy obsahuje popis nebezpečenstva, jeho ničivé účinky. Autor sa snaží používateľov presvedčiť o dôveryhodnosti, preto používa „overené“ zdroje..

Pojem *hoax* vieme preložiť ako poplašná správa, ktorá je šírená prostredníctvom Internetu. Napriek svojej nezmyselnosti vyzýva na to, aby sa preposielala ďalším používateľom. Jej jednoduchosť a prakticky nulová cena napomáha veľkému rozmachu tejto formy dezinformácie. K hoaxom môžeme priradiť varovné správy, pokusy o rozšírenie nejakej klamlivej informácie ukrývajúcej reálne skutočnosti a pod. Vo všeobecnosti ide o zbytočné správy, ktoré vo väčšine prípadov len zdržujú používateľa. Hoax vzniká v snahe sa zviditeľniť alebo spustiť lavínu. Ďalším dôvodom vzniku môže byť nesprávne porozumenie niektorých skutočností či snaha odhaliť

problém tam, kde nikdy nebol. Napríklad, ak niekto chce niekomu uškodiť, tak spustí hoax namierený voči dôvere daného miesta, značke alebo jednotlivca. Spúšťačom môže byť veľakrát udalosť v médiách, zlyhávajúci systém, nedostatočná ochrana prevádzkovateľa určitej služby, ale aj neistota používateľov daného miesta. Nato aby hoax dokázal prežiť a šíriť sa ďalej obsahuje vetu v zmysle „pošli to ďalej svojím priateľom a varuj ich“. Tomu napomáhajú rôzne webstránky aj sociálne siete. Cieľom je zakryť pravdu, prekryť fakty alebo vytvoriť situáciu, v ktorej sa všetky tvrdenia začnú stavať rovnako pravdepodobnými. V dnešnej dobe hoax tvorí mocný nástroj pre politické hry, pre ovplyvňovanie verejnej mienky a názorov ľudí [3]. Typický text poplašnej správy obsahuje vo väčšine prípadov tieto skutočnosti:

- Apel na dôležitosť správy sa snaží presvedčiť, že ide o šokujúce informácie, nové nebezpečenstvo.
- Autor poplašnej správy sa usiluje zaujať postoj dôveryhodného zdroja, prípadne presvedčiť, že to varovanie prišlo k nemu od dôveryhodného zdroja.
- Autor presvedča že ide o informácie, o ktorých médiá mlčia a nesmie sa ich o nich hovoriť. No on ich objavil a vyzýva k ich zdieľaniu.

III. USED METHODS OF MACHINE LEARNING

A. Naive Bayes Classifier

Metóda Naive Bayes predstavuje jednoduchý pravdepodobnostný klasifikátor, ktorý je často s úspechom používaný hlavne na spracovanie textu. Je založený na tzv. Bayesovskej vete. Je „naive“, pretože sa predpokladá, že atribúty (slová), ktoré do modelu vstupujú, sú na sebe nezávislé, čo v reálnych dátach takmer nikdy nie je splnené.. Multinomiálny Naive je vhodný vtedy ak v trénovacej množine sa vyskytujú iba diskkrétne atribúty (slová). Gausov Naive Bayes realizuje rozšírenie numerických hodnôt atribútov pomocou Gaussovho rozdelenia, na základe odhadu priemernej hodnoty a štandardnej odchýlky tréningových dát. Pravdepodobnosť nových hodnôt sa potom vypočíta pomocou Gaussovej funkcie hustoty pravdepodobnosti.

B. Metóda Podporných Vektorov

Metóda podporných vektorov (SVM Support Vector Machine) je určená najmä na klasifikáciu, ale je možné ju použiť aj na riešenie regresnej úlohy. Patrí medzi najpopulárnejšie algoritmy strojového učenia. Dosahuje vysokú presnosť, môže dobre fungovať aj keď dáta nie sú lineárne separovateľné. Princípom je rozdelenie tréningových dát na dve oblasti prislúchajúce triedam v množine príkladov. SVM generuje tzv. nad-rovinu (deliacu hranicu), ktorá separuje od seba príklady rôznych tried. Pre optimálnu deliacu hranicu platí, že musí byť umiestnená v čo najväčšom odstupe od krajných bodov reprezentovaných podpornými vektormi. To znamená, že nad-rovinu musí tvoriť okolo seba čo najširšie rozpätie. Stred tohto rozpätia musí byť rovnako vzdialený od oboch okrajov rozpätia.. SVM možno rozdeliť na lineárnu a nelineárnu. Lineárna SVM sa používa vtedy, keď sú dáta lineárne separovateľné. Nelineárna SVM je variant, využívaný pri nelineárnych dátach. V tomto prípade sa uplatňujú Kernelove funkcie, ktoré lineárne neseparovateľné dáta rozširujú o ďalšiu dimenziu. V novom viacdimenzionálnom priestore už budú dáta lineárne separovateľné [4].

C. Náhodné Lesy

Algoritmus náhodných lesov rozhodovacích stromov (RF Random Forest) sa používa hlavne pri klasifikačných problémoch ale je možné ho použiť aj na riešenie regresných úloh, kde potom pôjde o les regresných rozhodovacích stromov. Funguje na princípe generovania veľkého počtu náhodných stromov a následnej sumarizácie výsledkov každého z nich a to vo forme hlasovania ak ide o klasifikačné stromy alebo spriemerňovania ak ide o regresné stromy. Výhodou je, že náhodným výberom atribútov pri generovaní stromov sa tieto stávajú na sebe nezávislé, preto dosahuje táto metóda lepšie výsledky a taktiež je možný paralelný výpočet[5].

Algoritmus zabezpečuje, že stromy sú navzájom dekorelované. Na to aby sa zaistila rozmanitosť modelov sú využívané dva prístupy:

1. Rozhodovacie stromy sú citlivé na údaje, na ktorých sa model učí. Aj malé zmeny v dátovom súbore môžu mať za následok výrazne odlišnú štruktúru stromov. Náhodný les preto využíva proces *Bagging*. Ten, umožňuje každému jednotlivému stromu náhodne vzorkovať dáta. Hlavným dôvodom jeho využitia je zlepšenie stability a presnosti algoritmu.
2. V rozhodovacom strome, v čase rozdeľovania uzlov, zvažujeme každý možný atribút a vyberieme ten, ktorý má najvyšší informačný zisk. V náhodnom lese sa testovací atribút vyberá z náhodne zvolenej podmnožiny atribútov, čím sa zabráni tomu aby nejaký silný

atribút nebol vybraný v každom strome a teda všetky stromy by si boli podobné. Vďaka tomu v modeli nastáva nízka korelácia a väčšia diverzifikácia medzi jednotlivými rozhodovacími stromami [5].

IV. TRÉNOVANIE MODELOV NA DETEKCIU FALOŠNÝCH SPRÁV

A. Dátové Množiny a ich Predspracovanie

Pre falošné recenzie bola zvolená dátová množina Amazon, ktorá obsahovala recenzie zákazníkov jednotlivých produktov. K dispozícii boli informácie o produktoch: popis, informácie o kategóriách, cena, značka a vlastnosti. Dáta boli zaznamenané od mája 1996 do júla 2014, takže databázu tvorilo 1 689 188 hodnotení, od 192 403 zákazníkov o 63 011 produktoch. Dátová množina Amazon je dostupná na stránke <http://jmcauley.ucsd.edu/data/amazon/>.

Na detekciu hoaxov sme použili dátovú množinu, ktorá bola zostavená zo spamových údajov z webovej stránky Kaggle a archívu UCI. Dátová množina, spamham, obsahovala 5573 záznamov s 2 atribútmi. Atribúty sú označené ako „Category“ a „Message“. Každá správa bola klasifikovaná ako ham – nie spam alebo ako spam. Zistený podiel hodnôt v cieľovom atribúte je nerovnomerný, a to 86% záznamov ku 14%.

V rámci predspracovania údajov boli vykonané nasledovné úkony: odstránenie interpunkčných znamienok, extrahovanie koreňa slov, lematizácia, odstránenie stop slov a medzier, tokenizácia, frázové modelovanie, vytvorenie slovníka, vektorizácia textov (bag words, TF-IDF a Word2vec).

B. Modelovanie

Na tréningových množinách boli aplikované tri algoritmy strojového učenia a to Naive Bayes, SVM a náhodné lesy. Modely boli testované na 20% testovacích dát. Zvyšných 80 % bolo použitých na tréningovanie. Modely boli vyhodnotené pomocou nasledovných ukazovateľov binárnej klasifikácie: spoľahlivosť (Accuracy), presnosť (Precision), návratnosť (Recall) a F1 skóre (micro, macro). Výsledky testov sú v Tabuľke 1 a Tabuľke 2.

Tabuľka 1

Výsledky testov modelov náhodný les (RF), metóda podporných vektorov (SVM) a Gausov naivný Bayes (GNB) na detekciu falošných recenzií

Model	Accuracy	Precision	Recall	F1 score
RF	0,658	0,703	0,912	0,658
SVM	0,696	0,714	0,965	0,696
GNB	0,557	0,704	0,667	0,557

Tabuľka 2

Výsledky testov modelov náhodný les (RF), metóda podporných vektorov (SVM) a Gausov naivný Bayes (GNB) na detekciu e-mailových hoaxov

Model	Accuracy	Precision	Recall	F1 score
RF	0,972	0,970	0,998	0,942
SVM	0,985	0,983	0,999	0,969
GNB	0,957	0,953	0,999	0,957

Ako je vidieť v Tabuľke 1 a Tabuľke 2, Najvhodnejším modelom na detekciu falošných správ aj hoaxov je model naučený metódou podporných vektorov SVM.

V. ZÁVER

Článok bol zameraný na nájdenie vhodného modelu prečo najefektívnejšiu detekciu asociálneho správania. Na riešenie úlohy boli použité dve dátové množiny. Jedna obsahovala falošné recenzie webového obchodu Amazon, druhá dátová množina obsahovala e - mailové hoaxy. Obe dátové množiny boli predspracované. Vo fáze modelovania boli vstupné dáta sme rozdelené na tréningovú množinu (80%) a testovaciu množinu (20%). Boli vygenerované tri modely učení pomocou nasledovných algoritmov strojového učenia: Gausov Naive Bayes, náhodné lesy (RF) a metóda podporných vektorov (SVM). Na overenie výsledkov boli použité ukazovatele binárnej klasifikácie: spoľahlivosť (Accuracy), presnosť (Precision), návratnosť (Recall) a F1 skóre.

Z dosiahnutých výsledkov vyplýva, že najvhodnejším modelom na detekciu antisociálnych recenzií je model naučený metódou podporných vektorov SVM

POĎAKOVANIE

Práca prezentovaná v tomto článku bola podporená Slovenskou agentúrou na podporu vedy a výskumu v rámci projektu APVV-017-0267 “Automatizované rozpoznávanie antisociálneho správania v online komunitách”.

LITERATÚRA

- [1] D. Nations, What Is Social Media?. [Online], <https://www.lifewire.com/what-is-social-media-explaining-the-big-trend-3486616>.
- [2] M. Rouse, Social media. [Online] Techtarger, 2019, <https://whatis.techtarger.com/definition/social-media>.
- [3] Čo je to HOAX?. [Online], 202, <https://hoax.sk/co-je-to-hoax/>.
- [4] K. MACHOVÁ, New Trends in Machine Learning. Statistic approach. Edition of Scientific Writings of Faculty of Electrical Engineering and Informatics TU Košice, Košice, 2016, 1-96, ISBN 978-80-553-2602-3.
- [5] N.S. Chuhan, Randon Forest – A powerful Ensemble Learning algorithm. [Online], <https://towardsdatascience.com/random-forest-a-powerful-ensemble-learning-algorithm-2bf132ba639d>.

Analýza správania sa prosumerov v elektrickej sieti

¹Teodor KISEL, ²Dušan MEDVEĎ

^{1,2} Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky Technickej univerzite v Košiciach, Slovenská republika

¹teodor.kisel@student.tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — Tento príspevok sa zaoberá analýzou správania sa prosumerov, resp. on-grid elektrickej siete, ktorá je pripojená do väčšej synchronnej elektrickej siete, a off-grid systému. Na to boli vytvorené dva modely v programe Simulink (Matlab), kde v jednom prebieha simulácia činnosti on-grid systému elektrickej siete a v druhom prebieha simulácia činnosti off-grid systému elektrickej siete. V rámci oboch systémov hodnotíme v danom simulovanom období správanie sa modelu z pohľadu prechodných stavov a kvalitatívnych ukazovateľov elektrickej energie pri rôznych konfiguráciách zapojenia.

Kľúčové slová — prosumer, on-grid elektrická sieť, off-grid elektrická sieť, Simulink, fotovoltaická elektrárňa, veterná elektrárňa

I. ÚVOD

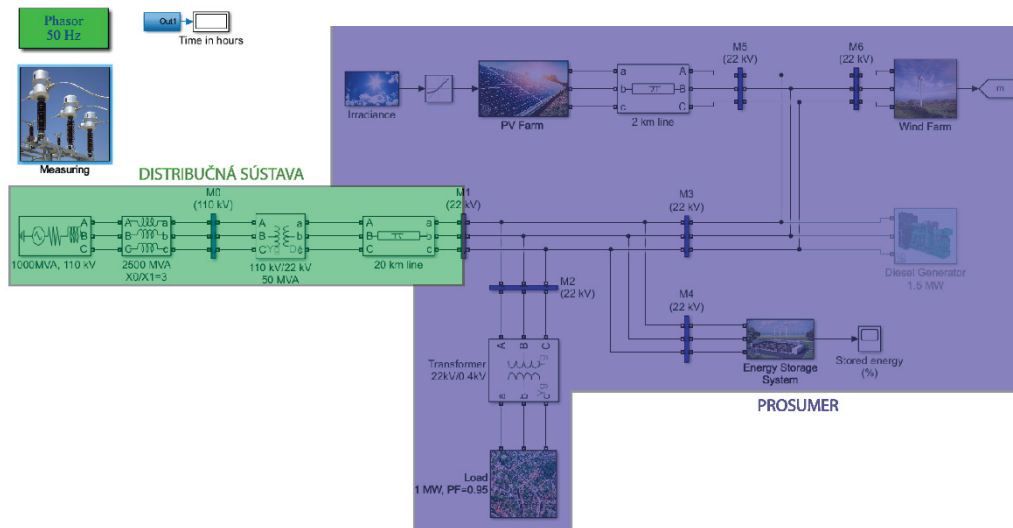
V rámci tohto článku je prezentovaná analýza správania sa prosumerov, resp. elektrickej siete, ktorá je pripojená do väčšej synchronnej elektrickej siete a off-grid elektrickej siete. Jedná sa o analýzu prechodných stavov a kvalitatívnych ukazovateľov v rámci sledovaného simulačného obdobia pri rôznych konfiguráciách zapojenia fotovoltaickej elektrárne, veternej elektrárne, záťaže, diesel-generátora a ESS (Energy Storage System), ktorý môže reprezentovať niekoľko možných spôsobov uloženia energie, ako sú napríklad zotrvačníky, stlačený vzduch, superkondenzátory, batériový systém.

Samotná simulácia jednotlivých prípadových simulácií prebiehala v prostredí programu Simulink (Matlab). Na riešenie prechodných stavov elektrického obvodu bol využívaný tzv. „phasor“ mód, vďaka ktorému bolo možné uskutočniť simuláciu v rozsahu niekoľkých desiatok hodín a ďalej pracovať so získanými údajmi. Medzi najdôležitejšie sledované parametre patrili tok činných výkonov a prúdov, napätie a frekvencia. Na zefektívnenie práce s nameranými dielčiami výsledkami v podobe grafov a hodnôt elektrických veličín bol nami využívaný zdrojový kód v Matlab-e.

II. POPIS JEDNOTLIVÝCH BLOKOV SIMULAČNÉHO MODELU

V prípade on-grid systému (obr. 1) model pozostáva zo zapojenia distribučnej sústavy (DS), ktorú tvorí 110 kV napäťový zdroj, znižujúci transformátor 110 kV/22 kV, 20 km dlhé vedenie a prosumer. Uzol s označením M1 tvorí pripojovací bod prosumera k DS. Prosumer je tvorený fotovoltaickou elektrárnou (FVE) s meracím bodom v uzle M5, veternou elektrárnou (VE) s meracím bodom v uzle M6, zásobníkom elektrickej energie (ESS) s meracím bodom v uzle M4, záťažou s meracím bodom v uzle M2.

Model sa vyznačuje komplexným poskytovaním dát merania amplitúd dôležitých parametrov v reálnom čase, ako je napríklad tok prúdov a činných výkonov, sledovanie napätia v jednotlivých uzloch, sledovanie frekvencie, sledovanie stavu SOC ESS, ponúka komparáciu činných výkonov jednotlivých výrobných a nevýrobných jednotiek, sledovanie charakteristických veličín veternej elektrárne, ako napr. natočenie lopatiek turbíny, rýchlosť rotora v pomerných jednotkách, rýchlosť vetra na základe charakteristiky vetra.



Obr. 1 Model zapojenia prosuamera k distribučnej sústave

V prípade off-grid systému, systém bol tvorený rovnakými prvkami ako je tvorený prosuumer na obr. 1.

A. Blok veternej elektrárne

Samotný subsystem veternej elektrárne (VE) pozostáva z troch logických častí: z *ochrannej časti*, ktorá je súčasťou bloku s označením Wind Turbine Protections, z *bloku elektrárne*, kde sa spracujú vstupné signály (charakteristika vetra) a generuje sa výstupný signál (štandardný blok VE, ktorý ponúka knižnica Simscape) a z *výpočtovej časti*, na základe ktorej dostávame výsledky vektorov s nameranými signálmi (blok Data acquisition).

Ochrany veternej turbíny sú nastavené s automatickým resetom po určitej preddefinovanej dobe. V prípade, ak došlo k prekročeniu medzných hodnôt (výstupom TripStatus bude logická 1), ochrana odpojí veternú turbínu zo siete, následne po určitej nastavenej dobe dôjde k resetu ochrany s možným opätovným pripojením veternej turbíny, v prípade, ak je poruchový stav odstránený.

VE je navrhnutá s inštalovaným výkonom 400 kW. Elektrárne začína vyrábať elektrickú energiu od rýchlosti 5 m/s a pri rýchlosti 9 m/s dosahuje maximálny výkon. Nad touto rýchlosťou dochádza k aktívnej regulácii „Pitch“, t.j. využíva natáčanie celého listu rotora podľa okamžitej rýchlosti vetra tak, aby bol celkový nábeh prúdu v danom okamihu optimálny **Chyba! Nenašiel sa žiaden zdroj odkazov..**

Výstupné napätie generátora VE je 575 V, ktoré je ďalej transformované na napäťovú úroveň 22 kV. Samotná VE je vzdialená od miesta záťaže 2 km dlhým vedením.

Výstupom z VE je merací blok s označením m_wt1, ktorého výstupom sú sledované charakteristické veličiny ako činný a jalový výkon v pomerných jednotkách, natočenie lopatiek a rotorová rýchlosť turbíny. Hodnoty z týchto veličín sú ďalej spracované v bloku Data acquisition, kde dochádza ku konverzii z pomerných jednotiek na jednotky sústavy SI.

B. Blok fotovoltaickej elektrárne

Model FVE nie je súčasťou knižnice Simulink. Jedná sa o pomerne zjednodušený model, kde sú vynechané konštrukčné prvky v podobe výkonového meniča. FVE je tvorená prúdovými zdrojmi, ktoré predstavujú zapojenie solárnych panelov, transformátorom 0,4 kV/22 kV, logickými blokmi, kde prebieha výpočet veľkosti injektovaného prúdu jednotlivými prúdovými zdrojmi a výstupnými meracími blokmi. FVE je od miesta záťaže, respektíve distribučnej siete, vzdialená km dlhým vedením.

Samotný model funguje na princípe prúdového zdroja, ktorý injektuje prúd v závislosti od vstupnej charakteristiky slnečného žiarenia a od nastavenej hodnoty účinnosti FVE a užitočnej plochy solárnych panelov.

C. Blok ESS (Energy Storage System)

Samotný subsystem ESS (Energy Storage System), resp. BESS (Battery Energy Storage System) je tvorený 4 blokmi: blok riadenia (ESS Control), výpočet stavu SOC, prúdovými zdrojmi, ktoré simulujú činnosť ESS (rovnaký princíp činnosti, ako v prípade FVE bloku) a transformátorom 0,4 kV/22 kV.

ESS, resp. BESS sa nabíja z DS v čase zopnutia nízkej tarify, čo v danom simulačnom modeli

predstavuje nastavený čas od 24. do 6. hodiny. Samotné nabíjanie je dovolené v prípade, ak kapacita ESS je menšia ako menovitá kapacita ESS.

Do riadiacej časti bloku ESS (ESS Control) vstupuje niekoľko premenných: združené napätia a prúdy v uzle M1 – výpočet činného výkonu, maximálny dovolený odoberaný činný výkon z DS – konštanta a maximálny nabíjací výkon – konštanta.

V riadiacej časti dochádza k porovnaniu parametrov okamžitej nameranej hodnoty činného výkonu s maximálnym dovoleným odoberaným činným výkonom. V prípade, ak dôjde k prekročeniu maximálneho dovoleného odoberaného činného výkonu z DS prosumerom, tak výstupom z bloku regulátora je rozdiel hodnôt medzi okamžitou hodnotou činného výkonu a maximálnou dovolenou hodnotou odoberaného činného výkonu, teda dochádza k vybíjaniu ESS. Táto činnosť riadiaceho člena platí iba v intervale medzi pravidelnými nabíjacími cyklami, t.j. v čase od 6. hodiny do 24. hodiny. Na riadenie logiky nabíjania slúži blok s označením Charging Logic. Výstupom z tohto bloku je v pravidelnom vyhradenom čase nabíjania logická 1, ktorá sa porovnáva so stavom SOC.

V prípade, ak je úroveň SOC nižšia ako nastavený parameter, tak v preddefinovanom intervale dôjde k nabíjaniu ESS.

Samozrejme, aby bolo možné bezchybne odsimulovať správanie sa ESS, je nutné v reálnom čase zaznamenávať stav SOC, na čo nám slúži blok Stored Energy Calculation.

Samotný ESS slúži na pokrytie zaťaženia v prípade, ak dôjde k prekročeniu maximálneho odoberaného výkonu z DS.

D. Blok záťaže

Blok záťaže pozostáva z troch častí: z dynamickej záťaže, ktorá predstavuje zapojenie niekoľkých stoviek domácností a príľahlej priemyselnej časti, statickej záťaže, ktorá predstavuje pripojenie technologických zariadení (napr. niekoľkých elektrických kotlov na výrobu TÚV) v troch stupňoch (3×50 kW), v prípade prebytku činného výkonu na strane prosumera a asynchrónneho motora.

Pri dynamickej záťaže môžeme nastaviť nominálny výkon v MW, účinník, v závislosti od ktorého meníme veľkosť odoberaného jalového výkonu záťažou a charakteristiku zaťaženia. Meranie elektrických veličín prebieha v uzle M2 (22 kV), teda už so zohľadnením strát transformátorom.

V prípade dynamickej záťaže ide podobne ako v prípade FVE bloku alebo ESS bloku o zapojenie prúdových zdrojov, ktoré pracujú na rovnakom princípe (identické zapojenie), avšak riadenie prúdových zdrojov je podmienené charakteristikou zaťaženia a do obvodu dodávajú záporný striedavý prúd, čiže sa správajú ako spotrebič.

E. Blok merania

Blok merania, označený ako Measuring, je najdôležitejším komponentom modelu. Do bloku merania vstupuje množstvo elektrických i neelektrických veličín v podobe ich amplitúd. Meranie elektrických veličín prebieha trojfázovo, ale z dôvodu zapojenia súmerných trojfázových zdrojov napätia, resp. prúdu a súmernej trojfázovej záťaže, vidíme v jednotlivých grafoch iba amplitúdu jednej fázy, nakoľko sa vzájomne prekrývajú. Samotný priebeh meraných veličín je možné sledovať v reálnom čase počas simulácie prostredníctvom bloku Scope. Vybrané dôležité veličiny sú automaticky ukladané do súborov formátu .mat, ktoré využívame pri vykresľovaní grafických priebehov pomocou zdrojového kódu v Matlab-e.

F. Blok diesel-generátora

Model bloku diesel-generátora pozostáva zo štandardných blokov, ktoré sú súčasťou knižnice Simulink, konkrétne zo synchronného generátora s vyjadrenými pólmi, budiaceho systému, bloku, ktorý kontroluje synchronne otáčky (PID regulátor) a transformátora. Pomocou rotorovej rýchlosti diesel-generátora môžeme v celom modeli zaznamenávať frekvenciu v sieti.

Vstupmi bloku synchronného generátora sú mechanický vstup, konkrétne mechanický výkon P_m , a vstup pre budiace napätie s označením V_f . Samotný blok diesel generátora môže pracovať v dvoch prevádzkových režimoch a to v generátorovej a motorovej prevádzke. V prípade zápornej hodnoty mechanického výkonu z PID regulátora (Diesel Engine Governor), pracuje diesel-generátor v motorickej prevádzke a naopak, v prípade kladnej hodnoty na mechanickom vstupe pracuje v generátorovom režime.

Výstupom synchronného generátora je trojfázové napätie, ktoré je vyvedené do bloku transformátora a merací výstup, pomocou ktorého zaznamenávame veľkosť statorového napätia ako spätnú väzbu pre blok budiaceho systému a rotorovú rýchlosť pre PID regulátor, ktorý udržiava synchronne otáčky.

III. SIMULÁCIA MODELU OFF-GRID A ON-GRID SYSTÉMU

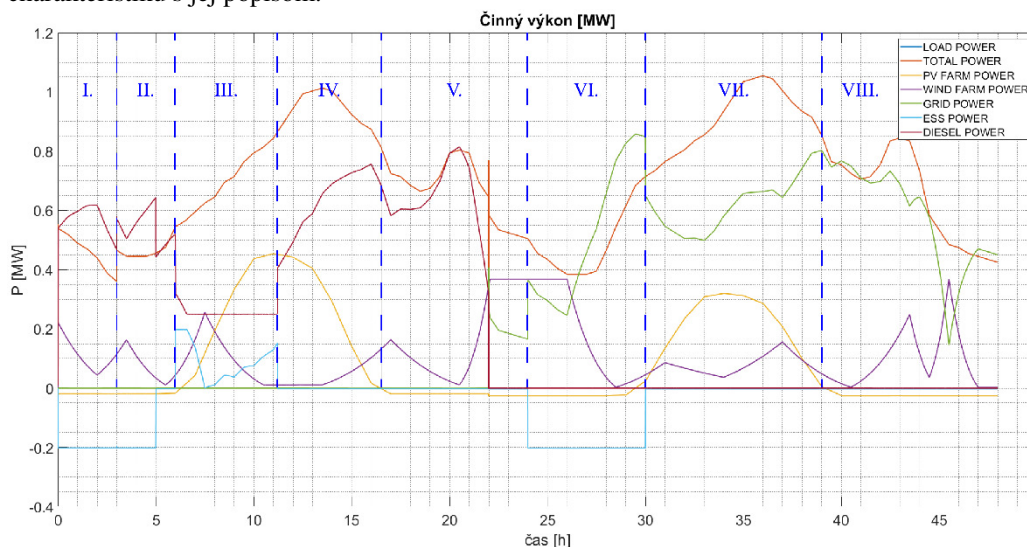
A. On-grid systém

On-grid systém bol odsimulovaný na modeli pri rôznych konfiguráciách zapojenia, kde bolo analyzované správanie sa prosumera z pohľadu prechodných stavov a kvalitatívnych ukazovateľov.

V danej simulácii bolo cieľom odsimulovať činnosť on-grid systému elektrickej siete, ktorá je prepojená v uzle M1 s DS podľa obr. 1. V prípade prebytku výkonu na strane prosumera je výkon dodávaný do DS za účelom dodávky, resp. predaja elektrickej energie a naopak, v prípade nedostatku výkonu, na pokrytie zaťaženia, je potrebný výkon odoberaný z DS. Výstupmi simulácie boli priebehy činných a jalových výkonov, priebehy napätia a prúdu, frekvenčná charakteristika, výpočet spotrebovanej/dodanej elektrickej energie, v prípade VE aj natočenie lopatiek, rotorová rýchlosť turbíny a rýchlosť vetra.

B. Off-grid systém

V danej simulácii bolo cieľom ukázať činnosť off-grid systému (model totožný s obr. 1 bez DS), kde predpokladáme, že v 22. hodine došlo k poruche na diesel-generátore. Chýbajúci výkon v sieti bol zabezpečený pripojením off-grid systému k DS. Nižšie, na obr. 2, môžeme vidieť priebeh činných výkonov s následným popisom prechodných stavov a na obr. 3 frekvenčnú charakteristiku s jej popisom.



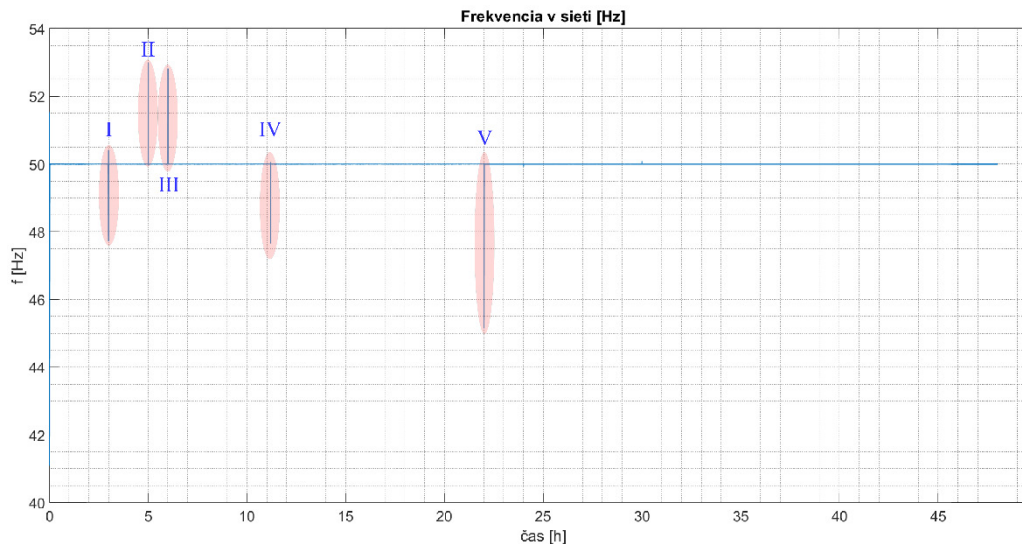
Obr. 2 Priebeh činných výkonov pre danú simuláciu

Na začiatku I. intervalu prebieha nabíjanie ESS (počiatočná úroveň SOC = 40 %) v čase nízkej tarify. Na pokrytie zaťaženia sa podieľa hlavne diesel-generátor spolu s veternou turbínou. V II. intervale v 3. hodine došlo k pripojeniu záťaže v podobe asynchrónneho motora s výkonom 100 kW a v 5. hodine k ukončeniu nabíjania ESS (SOC = 95 %). V III. intervale začína svoju činnosť ESS, čím znižuje podiel dodávaného výkonu diesel-generátora až do 11. hodiny. V danom intervale taktiež začína do siete dodávať výkon aj VE. V IV. intervale došlo k odpojeniu ESS z dôvodu vyčerpania menovitej kapacity (úroveň SOC = 10 %), regulácia činného výkonu v sieti je zabezpečená prostredníctvom diesel-generátora. V V. intervale ukončila svoju činnosť aj FVE, na pokrytie zaťaženia slúži iba výkon z VE a diesel-generátora. V 22. hodine došlo k poruche na diesel-generátore, v dôsledku čoho muselo dôjsť k jeho odpojeniu. Chýbajúci výkon v off-grid sieti nemohol byť nahradený žiadnym dostupným zdrojom, ktorý je súčasťou tohto systému. Preto došlo k pripojeniu off-grid siete k DS, čím sa zabezpečila dodávka výkonu bez prerušenia. V uzloch off-grid siete to spôsobilo nárast napätia z hodnoty 21,7 kV na hodnotu 23,5 kV. VI. interval je charakteristický nabíjaním ESS v čase od polnoci do 6. hodiny 2. dňa, kde došlo k jeho odpojeniu. V VII. intervale začína dodávať výkon FVE. Špičkový výkon, približne 0,3 MW, dosiahne v 34. hodine, avšak oproti 1. dňu ide o pokles výroby, čo je dané vstupnou charakteristikou slnečného žiarenia. V poslednom VIII. intervale FVE prestáva dodávať do siete výkon, na pokrytie zaťaženia sa podieľa predovšetkým výkon dodávaný z DS. Za celý sledovaný úsek bolo najviac elektrickej energie dodanej z DS v celkovom množstve 14,32 MWh, následne diesel-generátorom, 11,6 MWh. Celkovo bolo spotrebovanej elektrickej energie v objeme 33,36 MWh.

Tabuľka 1
 Spotrebovaná, resp. dodávaná elektrická energia pre danú simuláciu

W [MWh]	LOAD	TOTAL	ESS	GRID	PV FARM	WIND FARM	DIESEL
W_{IN}	33,36	0,00	-2,30	0,00	-0,61	0,00	0,00
W_{OUT}	0,00	33,36	0,46	14,32	4,66	5,24	11,60

kde: W_{IN} je elektrická energia spotrebovaná záťažou, resp. vo forme strát.
 W_{OUT} je elektrická energia dodávaná výrobnými jednotkami.
 TOTAL je elektrická energia vyrobená všetkými výrobnými jednotkami.



Obr. 3 Frekvenčná charakteristika v sieti

Popis frekvenčnej charakteristiky v sieti:

I. – pokles frekvencie pod nominálnu hodnotu f_n na úroveň 47,75 Hz je spôsobený pripojením asynchrónneho motora s nominálnym výkonom $P_n = 100$ kW, k vyregulovaniu siete došlo zvýšením dodávky výkonu diesel-generátorom.

II. – zvýšenie frekvencie nad nominálnu hodnotu f_n na úroveň 53 Hz je spôsobené odpojením záťaže, resp. ukončenie nabíjania ESS.

III. – zvýšenie frekvencie nad nominálnu hodnotu f_n na hodnotu 52,8 Hz opäť spôsobené činnosťou ESS, ale už nie ako spotrebič, ale dodávkou činného výkonu.

IV. – k poklesu frekvencie na úroveň 47,65 Hz došlo odpojením ESS, nakoľko hladina SOC ESS dosiahla svoje minimum (SOC = 10 %).

V. – pokles frekvencie na hodnotu 45,15 Hz spôsobený poruchou diesel-generátora (odpojenie v čase 22 hod.), vyregulovanie siete zabezpečené pripojením off-grid systému k distribučnej sústave.

V rámci simulačného obdobia bolo z OZE vyrobenej 27,85 % elektrickej energie, prevažná časť elektrickej energie bola dodaná z DS a diesel generátorom.

IV. ZÁVER

Predkladaný článok popisuje činnosť modelu off-grid a on-grid systému elektrickej siete vytvoreného v programe Simulink (Matlab). Na analýzu správania sa prosumerov bol vytvorený model, ktorý pozostával zo zapojenia veternej elektrárne, fotovoltaickej elektrárne, systému na uskladnenie elektrickej energie (ESS), záťaže (dynamickej, statickej) a distribučnej sústavy.

Pre off-grid systém bol vytvorený samostatný model, ktorý pozostával z rovnakých komponentov ako zapojenie on-grid systému doplnený o diesel generátor, ako hlavný regulačný člen činného výkonu a frekvencie v danom systéme. V rámci danej simulácie sme taktiež otestovali reakciu systému na poruchu na diesel-generátore s následným prifázovaním off-grid systému elektrickej siete k distribučnej sústave, aby bolo možné porovnať dané systémy v rámci jednej simulácie.

V rámci modelov je možné analyzovať off-grid, resp. on-grid elektrické siete pri dlhšom časovom rozmedzí, možnosť pridať nové zdroje akumulácie, napr. prečerpávaciu vodnú elektrárňu na podklade ESS, prispôbiť model vlastným potrebám.

Z pohľadu hodnotenia kvality elektrickej energie podľa STN EN 50160 (33 0121) pre NN a VN môžeme zhodnotiť, že v celej prepojenej elektrickej sieti pri pripájaní, resp. odpájaní záťaže alebo výrobnjej jednotky, nedošlo k žiadnym nebezpečným javom a z pohľadu najdôležitejších kvalitatívnych ukazovateľov, t.j. napätia a frekvencie, boli dodržané v dovolených medziach. Z frekvenčných charakteristík môžeme posúdiť, že v prípade systému off-grid siete pri spínaní záťaže v podobe asynchrónneho motora, či ESS pri nabíjaní, resp. vybíjaní, došlo k oveľa výraznejším odchýlkam frekvencie, čo nie je v prípade takéhoto systému nič výnimočné. V rámci on-grid systému pri všetkých prípadových simuláciách boli odchýlky frekvencie pri regulácii činného výkonu rádovo v desatinách a neprekročili $\pm 0,5$ Hz ani v jednom simulovanom prípade.

Pri modelovaní jednotlivých prípadových simulácií sme narazili na množstvo problémov, napr. v prípade modelu veternej elektrárne, ktorý výrazne spomaľuje danú simuláciu a taktiež predlžuje čas pri výpočtoch v Matlab-e. Z tohto dôvodu bolo nutné použiť zidealizované vstupné charakteristiky slnečného žiarenia, rýchlosti vetra a charakteristiku zaťaženia. V prípade simulácie bez VE je simulačný čas rádovo v sekundách, s použitím do 1 hodiny. Dané modely off-grid a on-grid elektrickej siete je možné aplikovať pri reálnych modelových situáciách, lenže je nutné počítať s výrazne dlhším simulačným časom.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18.

LITERATÚRA

- [1] Kudelas, D., Rybár, R., Cehlár, M., „*Energia vetra*“. Košice: Edičné stredisko Fakulty FBERG, TU v Košiciach, 2009. 216 s. ISBN: 978-80-553-0169-3.
- [2] Hpower, „*Solárne panely*“ [online]. [cit. 2020-04-22]. Dostupné z: < https://www.hpower.sk/solarne_panely/ >
- [3] Quachning, V., „*Obnoviteľné zdroje energií*“, Praha: Grada Publishing, a.s., 2010. 296 s. ISBN 978-80-247-3250-3.
- [4] Sapi, „*Fotovoltika na Slovensku*“ [online]. 2019 [cit. 2020-03-30]. Dostupné z: < <https://www.sapi.sk/fotovoltika-na-slovensku> >
- [5] IRENA, „*Solar Energy Data*“ [online]. [cit. 2020-04-21]. Dostupné z: < <https://www.irena.org/solar> >
- [6] Kudelas, D., Rybár, R., Cehlár, M., „*Energia vetra*“, Košice: Edičné stredisko Fakulty FBERG, TU v Košiciach, 2009. 216 s. ISBN: 978-80-553-0169-3.
- [7] STN EN 50160: 2011, „*Charakteristiky napätia elektrickej energie dodávanej z verejnej elektrickej siete*“, Bratislava: Slovenský ústav technickej normalizácie, 2010.
- [8] Kiseľ, T., „*Analýza správania sa prosumerov v elektrickej sieti*“. Diplomová práca. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2020.

Vývoj HVDC systémov

¹Michal Kolcun, ²Zsolt ČONNKA, ³Vladimír KOHAN

¹ Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita Košice, Slovenská republika

² Katedra elektroenergetiky, FEI TU Košice, Slovenská republika

¹michal.kolcun@tuke.sk, ²zsolt.conka@tuke.sk, ³vladimir.kohan.2@student.tuke.sk

Abstrakt — Tento príspevok sa zaoberá problematikou HVDC systémov. Úvodné kapitoly nám približujú výhody, nevýhody a ekonomické aspekty HVDC systémov. Ďalšie kapitoly sú zamerané na princípy konverzných (koncových) terminálov a ich následnými zapojeniami so vzdušným alebo káblovým vedením v znázornených schémach. Nasledujúca kapitola obsahuje stručný popis porúch a je úvodom ku kapitole ochrán. Dôležitou súčasťou sú filtre a ich zapojenia do sústavy, pre správny a spoľahlivý tok výkonu, ktorým sa zaoberá jedna zo záverečných kapitol. Posledná kapitola je venovaná moderným trendom, ktoré sa rýchlym tempom dostávajú do popredia a stávajú sa neodmysliteľnou technológiou pre prenos a rozvod elektrickej energie.

Kľúčové slová — HVDC, jednosmerný prenos výkonu, tok výkonov, stabilita prenosu, elektrizačná sústava

I. ÚVOD

Výkonové polovodičové spínače sú kľúčovými komponentmi vo výkonových konverteroch pre HVDC a FACTS systémy. Majú pomerne zložitú polovodičovú štruktúru a môžu fungovať tak, že tok elektrickej energie môže byť presne riadený. V dnešnej dobe sú najpoužívanějšími výkonovými zariadeniami v tejto oblasti GTO tyristory, IGCT a IGBT tranzistory [1].

V oblasti HVDC technológií stále dochádza k významným vývojom a zdokonaľovaniu. Systémy VSC-HVDC našli využitie v odbere výkonu z pobrežných veterných elektrární, napájania výkonu morských plošín, dodávky regulovateľného výkonu na zlepšenie bezpečnosti, integráciu rozptýlenej alebo decentralizovanej výroby. Spoľahlivosť sa v nedávnej dobe výrazne zlepšila a LCC na báze HVDC je stále ekonomická pri ultra vysokých napätiach. Súčasne trendy sú vývoj ± 800 kV DC na dlhú vzdialenosť prenášaného výkonu v menovitých hodnotách 5000-6000 MW. Hoci HVDC prenos je považovaný za pomerne dosť zreľú technológiu, je vcelku úžasné s koľkými novými aspektmi a projektmi počíta [2], [3].

Cieľom predkladaného príspevku je vniest' vyšší stupeň znalosti o základných modeloch a prvkoch rýchlo rozvíjajúcich sa zariadení HVDC systémov, ktoré sú v súčasnej dobe stále viac a viac žiadané s rastúcim vývojom a výhodami ich použitia.

II. PREHĽAD PROBLEMATIKY A POPIS VYUŽITIA HVDC SYSTÉMOV

Národný priemyselný rast si vyžaduje stále zvyšovanie spotreby elektrickej energie. To viedlo k zvýšeniu výrobných a prenosových zariadení na uspokojenie narastajúceho dopytu. V rozvojových krajinách sa dopyt zdvojnásobuje každých sedem rokov čo si vyžaduje značné investície a rozvoj v elektroenergetickom odvetví [2].

Tento stále zvyšujúci sa dopyt po výkone nie je vždy možné uskutočniť prechodom na vyššiu napätovú hladinu. Problémy striedavého prenosu, zvlášť pri diaľkových prenosoch viedli ku vývoju jednosmerných prenosových systémov. Avšak ako výroba a využitie elektrickej energie stále zostáva v popredí striedavý prúd [2].

Moderné HVDC systémy sú zreľou a stále viac rozvíjajúcou sa technológiou, ktoré hrajú dôležitú úlohu v dvoch hlavných častiach a to pri prenose a vzájomnom prepojení dvoch alebo viacerých odlišných systémov, napríklad back to back systémy (BtB). Kombináciou vysokej spoľahlivosti s dlhodobou životnosťou sú často inštalované od hlavného reťazca v elektrizačnej sústave. Hlavným prvkom HVDC systému je výkonový striedač a usmerňovač, ktorý slúži ako spojka medzi striedavým systémom [3].

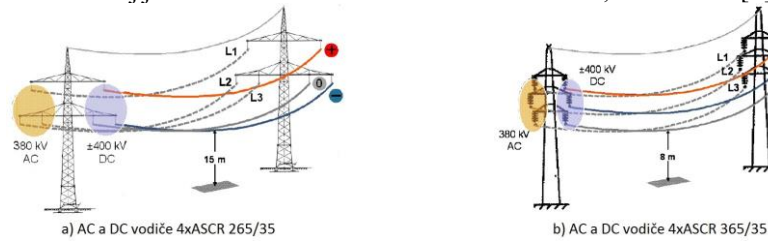
A. Výhody a nevýhody použitia HVDC technológií

Výhody DC prenosových sústav

Použitie klasických HVDC systémov na prenos veľkých výkonov cez veľké vzdialenosti je omnoho viac výhodnejšie a efektívnejšie, pretože celkové náklady pre takýto prenosový systém

sú menšie a straty výrazne nižšie ako pri striedavých prenosoch. Podstatnou výhodou použitia jednosmerného prepájania je, že neexistuje žiadne obmedzenie v stabilite pri veľkosti výkonu alebo vzdialenosti sústavy. Jednosmerné vedenie si nevyžaduje viac ako dva vodiče v porovnaní so striedavým vedením, ktoré používa tri [3].

Ak striedavé vedenie dosiahne obmedzenie stanovené stabilitou systému, alebo obmedzenie jeho tepelnou kapacitou, nie je možné pridať ďalšie paralelné vedenie, ale prenos sa môže uskutočniť pomocou HVDC vedenia. Prenosová kapacita jednosmerného vedenia dosahuje až trojnásobok AC kapacity na prenos zmenou stožiarovej nosnej konfigurácie. Je možné prevádzkovať striedavé aj jednosmerné vedenia na rovnakom stožiaru, viď. Obr. 1 [3], [6].



a) AC a DC vodiče 4xASCR 265/35

b) AC a DC vodiče 4xASCR 365/35

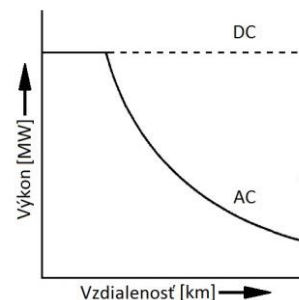
Obr. 1 a) Trojuholníkový dvojité poťah vedenia s AC a DC systémom [6]

b) Vertikálny dvojité poťah vedenia s AC a DC systémom [9]

HVDC prenos má niekoľko pozitívnych funkcií, ktoré v sebe AC prenosy nemajú zahrnuté. Jedná sa predovšetkým o rýchlu ovládateľnosť výkonov v DC linkách prostredníctvom ovládania konvertorov (menič/striedač) [2], [3].

Navyššie DC prenos prekonáva niektoré z problémov AC prenosu, ako napríklad:

- *hranica stability* - prenos elektrickej energie v striedavých linkách je závislý na rozdiel uhlá fázora napätia na začiatku a konci vedenia. Schopnosť prenosového systému AC a DC sústavách od vzdialenosti je znázornená na Obr. 2. Kde jednosmerná sústava takmer nie je ovplyvnená vzdialenosťou prenosu. V tomto prípade sú HVDC sústavy obmedzené len prúdovou zaťažiteľnosťou (oteplenie prúdom).



Obr. 2 Schopnosť prenosu výkonu voči dĺžke vedenia [2]

Nevýhody HVDC prenosových sústav

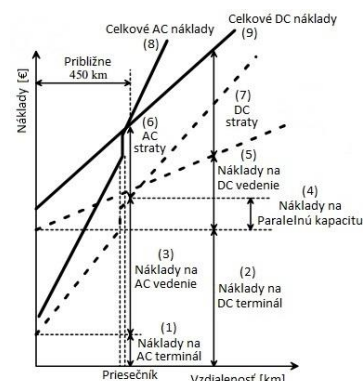
Rozsah použitia jednosmerného prenosu je obmedzená nasledujúcimi faktormi:

- náročnosť na prerušenie skratových jednosmerných prúdov, čo má za následok vysoké náklady na jednosmerné vypínače,
 - neschopnosť použitia transformátora na zmenu napäťových hladín,
 - vysoké náklady na konverzné zariadenia,
 - vytváranie vyšších harmonických frekvencií, ktoré vyžadujú ladenie pomocou jednotlivých filtrov na AC aj DC strane,
- komplikované ovládanie [2].

B. Ekonomické aspekty, náklady a charakteristika HVDC systémov

Cena prevádzkových vedení v sebe obsahuje investičné a prevádzkové náklady. Investície zahŕňajú náklady na vymedzenie priestoru RoW, prenosové stožiare, vodiče, izolátory a terminálové zariadenia. Prevádzkové náklady v sebe zahŕňajú náklady vo výkonových stratách v prenosu [2]. Vplyv koróny na jednosmerné vodiče býva menej významný ako pri striedavých a to tiež vedie k voľbe prevádzkovej (ekonomickej) veľkosti vodičov v DC prenosoch. Ďalšie okolnosti, ktoré ovplyvňujú prenosové náklady sú náklady na kompenzáciu a koncové terminálové zariadenia. Jednosmerné vedenia si nevyžadujú kompenzáciu, ale náklady na koncové terminály sa zvyšujú v dôsledkom

prítomnosti konvertorových zariadení a filtrov [2].



Obr. 3 Prenosová vzdialenosť a investičné náklady pre AC a DC prenosové vedenia [3]

Zlomová vzdialenosť môže kolísat približne (500-800) km pri vzdušných vedeniach a 40 km káblových vedeniach v závislosti za jednotku na nákladoch vedenia. [2]

Súhrn nákladov na HVDC v porovnaní s HVAC systémami

Pre veľké vzdialenosti je z efektívneho a cenového hľadiska jednosmerný prenos v porovnaní so striedavým omnoho viac výhodnejší (nad 500-800 km dĺžky vedenia). V prípade podmorských káblov, ako je možné vidieť na Obr. 3 je priesečník čiar, ktoré určujú vzdialenosť pomerne kratšia a to približne 40 km dĺžky káblového vedenia [3].

III. PRINCÍP ZÁKLADNÝCH PRVKOV A ZARIADENÍ HVDC SYSTÉMU

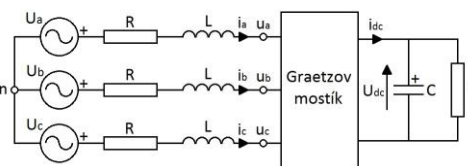
A. Funkcia tyristorov a tranzistorov využívaných v HVDC technológiách

HVDC konvertory sú zostavené z kaskád, ktoré majú tú vlastnosť, že sú vodivé v priepustom smere a blokované v závernom smere. Kaskády sa skladajú z mnoho sériovo a paralelne pospájaných spínačových článkov (buniek). Zatiaľ čo tyristorové kaskády sú využívané v LCC (CSC), IGBT tranzistorové kaskády sú používané vo VSC systémoch [2].

B. Zapojenie LCC a VSC konfigurácií

VSC

VSC používa konštantný jednosmerný napäťový zdroj realizovaný veľkým kondenzátorom na vstupe. Je riadený PWM moduláciou, ktorá sa používa aj vo FACTS aplikáciách. Konvertor môže obrátiť polaritu napätia pomocou rôznych režimov a tok prúdu je reverzibilný [1].



Obr. 4 Schéma VSC [2]

LCC (CSC) a CCC

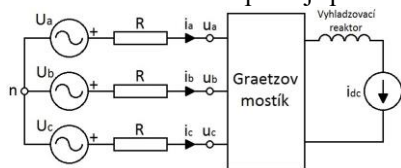
Používa konštantný jednosmerný prúdový zdroj realizovaný veľkou indukčnosťou na vstupe. Typickým príkladom je 6-pulzný, tyristorovo riadený plný mostík vo veľkej miere používaný v LCC HVDC. Aby bolo možné zmeniť smer toku výkonu, konvertor je riadený takým spôsobom, že je obrátená polarita napätia na výstupe [1].

Dôsledkom rozptylovej indukčnosti transformátora a sieťovej impedancie sa prúd v kaskáde nemôže razom zmeniť a tak komutácia z jednej kaskády do ďalšej nemôže byť okamžitá [2].

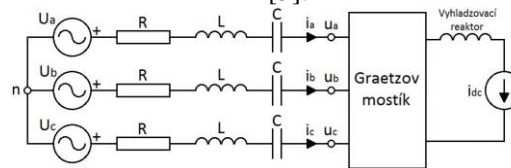
Konfigurácia kapacitného komutovaného konvertora (CCC) je znázornená na

Obr. 6. Sú v ňom vložené sériové kondenzátory medzi konvertorový transformátor s kaskádami na pomoc komutácie, obzvlášť pri prevádzke striedača [2].

V tomto systéme je umiestnený sériový kondenzátor medzi kaskádu a transformátor. Alternatívny systém je známy ako CSCC HVDC, tento variant môže využiť TCSC techniku. Táto alternatívna technika pracuje podobným spôsobom ako CCC HVDC [3].



Obr. 5 Schéma LCC (CSC) [2]



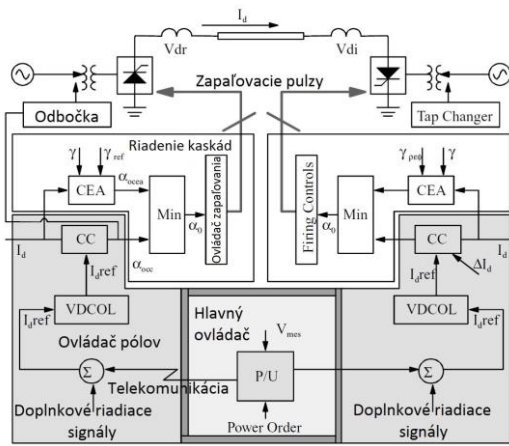
Obr. 6 Schéma CCC [3]

C. Konverzia a ovládanie HVDC systémov

Obr. 7 znázorňuje základný riadiaci diagram HVDC systému. HVDC systém

môže byť rozdelený do niekoľkých úrovní. Hlavné riadenie určuje charakter výkonu alebo charakter frekvencie a vypočítava charakter prúdu pre oba póly. Charakter prúdu, ktorý bol prijatý z hlavného ovládača je modifikovaný riadiacimi funkciami a obmedzeniami v ovládači pólov. Riadenie kaskádových skupín pozostáva z ovládača konvertora a ovládača kaskádového zapnutia. Konvertorový ovládač zahŕňa prúdový regulátor. Ovládač

kaskádového zapalovania rozdeľuje zapalovací signál do všetkých tyristorov [3].



Obr. 7 Základný diagram riadenia HVDC systému [3]
 VSC a LCC

VSC je založený na nútených komutačných zariadeniach ako je IGBT alebo GTO, ktoré umožňujú prevádzku konvertora vo všetkých štyroch kvadrantoch P-Q rovinách. Pretože komutácia môže byť dosiahnutá rýchlo a nezávisle na systéme striedavého napätia, je možný celkom iný typ prevádzky v porovnaní s LCC konvertorom [3].

LCC závisí na systéme striedavého napätia pre jeho správnu prevádzku. LCC pracuje na oneskorení účinníka, pretože otváranie konvertora musí byť oneskorené vzhľadom k prechádzanému napätiu pre ovládania DC napätia [3].

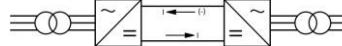
D. Schémy konfigurácií rôznych typov HDC liniek

Monopolárne HVDC vedenia

Monopolárne HVDC sa skladá z jedného vodiča, zvyčajne zápornej (-) polarít, ktorý používa zem alebo more ako spätný vodič. Používa sa hlavne na prenos výkonu pomocou káblov. Kovová spätná väzba je uprednostňovaná namiesto uzemnenia, kde odpor pôdy je príliš vysoký alebo podzemné/podmorské kovové komponenty môžu spôsobovať rušenie [2], [3].

Najbežnejšie prevádzkové konfigurácie:

Symetrické (súmerné) monopolárne vedenie (linka):



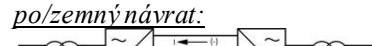
Obr. 8 Schéma symetrického monopolárneho vedenia [4]

Asymetrické (nesúmerné) monopolárne vedenie, kovový návrat:



Obr. 9 Schéma asymetrického monopolárneho vedenia s kovovým návratom [4]

Asymetrické (nesúmerné) monopolárne vedenie, po/zemný návrat:



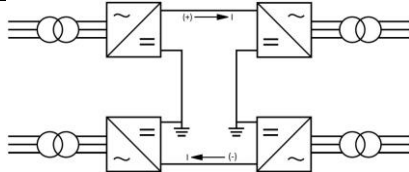
Obr. 10 Schéma asymetrického monopolárneho vedenia so zemným návratom [4]

Bipolárne HVDC vedenia

Bipolárne HVDC vedenie sa skladá z dvoch vodičov a z dvoch pólov (+ a -) a z uzemneným neutrálnym bodom. Každý terminál má dve sady meničov s identickými menovitými hodnotami, sériovo spojenými na jednosmernej strane. Uzol medzi týmito konvertormi je uzemnený buď na jednom alebo na oboch koncoch. Pri prevádzke v ustálenom stave oba póly pracujú pri rovnakých prúdoch a tak je zemný prúd nulový. Ak jeden z pólu zlyhá potom druhý pól môže sám prenášať polovičný výkon so zemou použitou ako spätnú väzbu, náhradu za nefunkčný pól. V bipóle je hodnota prenosového výkonu zvýšená o dvojnásobok v porovnaní s prípadom monopolárnej linky. Vytvára menej harmonických v porovnaní s monopolárnym prípadom. Spätný tok výkonu môže byť riadený konvertovaním polarít oboch pólov [2], [3].

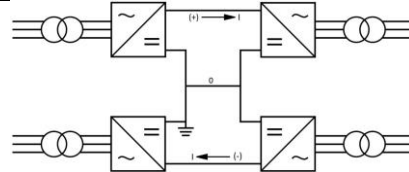
Najbežnejšie prevádzkové konfigurácie:

Bipolárne uzemnené vedenie (s uzemnenými elektródami):



Obr. 11 Schéma uzemneného bipolárneho vedenia (s uzemnenými elektródami na oboch stranách) [4]

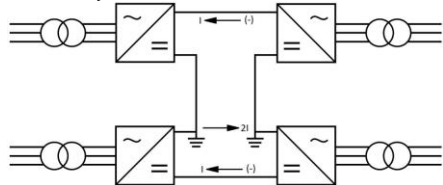
Bipolárne vedenie s kovovým neutrálnym (zemným) vodičom:



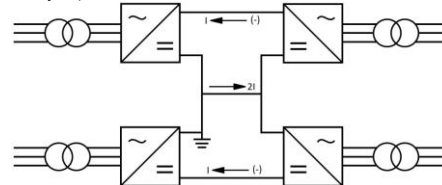
Obr. 12 Schéma bipolárneho vedenia s kovovým neutrálnym (uzemneným) vodičom [4]

Homopolárne HVDC vedenia

Homoplárne vedenie má dva alebo viac vodičov, pričom všetky majú rovnakú polaritu (obvykle zápornú), a sú vždy prevádzkované so zemou alebo kovovou spätočnou cestou [2].
Homoplárne uzemnené vedenie (s uzemnenými elektródami): Homoplárne vedenie s kovovým neutrálnym (zemným) vodičom:



Obr. 13 Schéma homoplárneho vedenia (s uzemnenými elektródami na oboch stranách) [2]



Obr. 14 Schéma homoplárneho vedenia s kovovým neutrálnym (uzemneným) vodičom [2]

ZHRNUTIE

Pretože prevádzka jednosmerného vedenia bez návratu v zemi je vhodnejšia, bipolárne vedenie je najčastejšie používané. Homoplárne vedenie má výhodu v znížení nákladov na izoláciu, ale nevýhodu zemného návratu, ktorá prevažuje jeho výhodu. Účinky koróny v jednosmerných vedeniach sú podstatne menšie pri zápornej (-) polarite vodiča. Monopolárna prevádzka sa používa a slúži ako prvý stupeň k vývoju bipolárneho vedenia, keďže investície konvertorov môžu byť odložené, až do nárastu zaťaženia, ktoré si vyžaduje bipolárnu prevádzku na zdvojnásobenie kapacity monopolárnej linky [2].

HVDC Back-To-Back systémy

V tomto type systému sú konverzné prvky, usmerňovač a striedač umiestnené v tej istej stanici. Vo všeobecnosti platí, že sa používa na poskytovanie asynchrónneho vzájomného prepojenia dvoch striedavých systémov (50Hz/60Hz). Amplitúda napätia je všeobecne malá, okolo 150 kV (500 kV) na optimalizáciu nákladov na kaskády [3].

HVDC Multi - Terminálové systémy (MTDC):

Týka sa to HVDC systémov, ktoré sa skladajú z troch alebo viacerých terminálových staníc. Jeho architektúra je zložitejšia v porovnaní so systémom dvoch terminálov. Vyžaduje si značnú zložitosť na uľahčenie komunikácie a riadenia medzi jednotlivými stanicami. Avšak, tento systém je považovaný za relatívne novú technológiu a má rozsiahly potenciál pre veterné využitie v budúcnosti. Pričom sú dva typy MTDC liniek - paralelný a sériový typ [3].

MTDC systém má viac ako dve konvertorové stanice, pričom niektoré sú prevádzkované ako usmerňovacie a iné ako striedacie terminály [2].

IV. PORUCHY, SKRATY A OCHRANA VOČI NEŽIADUCIM VPLYVOM

A. Poruchy na strane DC linky

Poruchy musia byť detekované a systém musí byť chránený pomocou spínačov a riadiacich činností tak, že narušenie v prenose je minimalizované.

Okrem narušenia normálnej prevádzky rôzne poruchy, ktoré sa môžu vyskytnúť v systémoch HVDC spôsobujú namáhanie zariadení v dôsledku nadprúdov a prepätí. V konvertorových staniciach sú najkritickejšími zariadeniami kaskády, ktoré musia byť chránené pred poškodením spôsobeným stúpaním teploty v PN prechode spínačov, čo zapríčiňuje nadmerné straty výkonu v zariadení. Významnú rolu pri ochrane zariadení hrá riadenie [2].

Poruchy konvertora

Existujú tri základné typy porúch, ktoré sa môžu nastať v konvertoroch:

1. Poruchy spôsobené v dôsledku nefunkčnosti kaskád a regulátorov:
2. Porucha komutácie v striedačoch.
3. Skrat v konvertorovej stanici.

B. Ochrana na strane DC linky

Ochrany na jednosmernej strane sú špeciálne navrhnuté pre HVDC prenos a tvorí ich:

- Skratová ochrana
- DC nadprúdová ochrana
- Ochrana pri komutačných poruchách
- Ochrana voči vyšším DC harmonickým kmitom
- Ochrana proti napätovému namáhaniu
- Ochrana proti nadmernému oneskoreniu uhla
- DC ochrana proti zemnému spojeniu
- DC ochrana proti prepätiu
- Ochrana proti minimálnemu DC napätiu
- Ochrana DC vedenia [3]

C. Typy využívaných filtrov v AC a DC sieťach pre HVDC systémy

HVDC konvertory generujú ako striedavé tak aj jednosmerné harmonické frekvencie, ktoré sú jednotlivo vnášané do striedavého systému a jednosmerného vedenia jednotlivo [2].

AC filtre sa stále používajú na odfiltrovanie harmonických frekvencií striedavých prúdov, ktoré sú kritické. Jednosmerná vyhladzovacia tlmivka spolu s jednosmernými filtermi vykonávajú funkciu filtrovania harmonických v jednosmerných vedeniach. Zatiaľ čo väčšina filtrov použitých v konvertorových staniách sú pasívne filtre používajúce tlmivky a kondenzátory, aktívne filtre na báze VSC boli zavedené od roku 1993 v spolupráci s pasívnymi filtermi [2].

Pasívne a aktívne DC filtre

Ladené aj tlmené filtre sú používané pre jednosmerné filtre. Back To Back (BtB) HVDC linky a tie, ktoré používajú káblové systémy nemajú jednosmerné filtre, pretože indukovaný šum nie je problémom [2].

Aktívne filtre boli považované pre jednosmerné filtrovanie, aby spĺňali prísne podmienky z energetických podnikoch pri obmedzovaní telefónneho rušenia. Použitie len pasívnych filtrov môže podstatne zvýšiť náklady a preto sa používajú v sérii s aktívnymi filtermi [2].

V. MODERNÉ TRENDY A DOPYT V NÁVRHOCH NA VYUŽITIE HVDC SYSTÉMOV

Budúce výkonové polovodičové zariadenia

Návrh HVDC je priestorovo náročný obzvlášť pre pobrežné použitie. Pôdorys konvertorov môže byť zmenšený použitím konceptu hybridných výkonových zariadení. S touto koncepciou môže byť viac zariadení hybridno-integrovaných na rovnakej polovodičovej štruktúre. Príkladom je spätne priepustný IGBT vyvinutý firmou ABB, kde FWD a IGBT sú integrované do takzvaných BI-ódu IGBT za účelom dosiahnutia vyššej výkonovej schopnosti pre rovnaký pôdorys výkonového modulu [1].

Výkonové zariadenia na báze karbidu kremíka (SiC). U SiC bolo dokázané, že má lepšiu širokopásmovú medzeru v polovodičovom materiáli ako Si, pretože dosahuje 10 krát vyššie blokovanie napätia a významné spínacie straty [1].

ABB 1 100 kV HVDC systém

Firma ABB vyvinula kompletne riešenie pre 1 100 kV HVDC prenosový systém. Napätie 1 100 kV zvyšuje prenosový výkon na 10 GW na viac ako 3 000 km. Kde spoločnosť ABB úspešne vyvinula všetky významné zariadenia potrebné pre toto nasadenie [5].

ZÁVER

Použitie výkonovej elektroniky postupne odstránilo technologické bariéry, ktorým čelila AC technológia. Vývoj HVDC systémov je dôležitý pre vytváranie už projektovanej siete SuperGrid, ktorá umožní ľudskej spoločnosti integrovať v širšej škále veterné a solárne elektrárne, čo zabezpečí príjem čistejšej a stabilnejšej energie [1].

So začiatkom novej energetickej éry a potreby vybudovania inteligentnejšej siete SmartGrid sa očakáva, že HVDC bude rásť ďaleko za svoju tradičnú pozíciu ako doplnok ku striedavému prenosu. HVDC je teraz metódou vhodne volenou pre podmorský prenos a vzájomné prepojenie asynchrónnych striedavých sietí, ktoré poskytujú efektívny, stabilný prenos a schopnosť riadenia výkonu. HVDC je taktiež technológiou vhodnou pre prenos výkonu na veľmi dlhé vzdialenosti s nízkymi elektrickými stratami. Dôvody pre voľbu HVDC namiesto striedavého prenosu energie, sú v konkrétnych prípadoch často početné a komplexné. V mnohých prípadoch sú HVDC linky opodstatnené na základe kombinácie technických, ekonomických a environmentálnych výhod [8].

Danú problematiku je možné ďalej rozšíriť aj pre prax kde by týmto spôsobom by bolo možné odľahčiť určité prenosové vedenia jednotlivých štátov, ktoré sú vplyvom liberalizácie trhu s elektrinou preťažované a to niekedy až po hranicu ich maximálnych zaťažení a následného tzv. stavu BlackOut. Vhodným projektom s využitím HVDC linky, HVDC spojky alebo FACTS systémov je možné stabilne riadiť tok výkonu a tým zabezpečiť spoľahlivé dodávky elektrickej energie spotrebiteľom s možnosťou vyvarovania sa preťažovaní vedenia.

POĎAKOVANIE

Túto prácu podporila Vedecká grantová agentúra Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky a Slovenskej akadémie vied grantom VEGA č. 1/0372/18. Na tomto mieste by som sa rád poďakoval môjmu školiteľovi, Ing. Zsoltovi Čonkovi, PhD., za jeho cenné rady, odbornú pomoc, usmerňovanie pri písaní tohto článku a užitočné rady, ktoré mi bol ochotný zveriť v korektnom prístupe.

LITERATÚRA

- [1] Mircea Eremia, Chen-Ching Liu, Abdel-Aty Edris. Advanced Solutions in Power Systems; HVDC, FACTS, and Artificial Intelligence, First Edition. IEEE Press, © 2016 by The Institute of Electrical and Electronics Engineers, Inc. Published 2016 by John Wiley & Sons, Inc. ISBN: 978-1-119-03569-5
- [2] K. R. Padiyar. HVDC Power Transmission System (2nd Edition). India Institute of Science, Bangalore India. © 2011 by New Academic Science Limited. ISBN: 978-1-906574-77-2, Replika Press Pvt. Ltd.
- [3] Chan-Ki Kim, Vijay K. Sood, Gil-Soo Jang, Seong-Joo Lim and Seok-Jin Lee. HVDC TRANSMISSION, Power Conversion Applications in Power System. Korea Electric Power Corporation, © June 2009 John Wiley & Sons (Asia) Pte Ltd, ISBN: 978-0-470-82295-1, Wiley-IEEE Press.
- [4] ABB: HVDC Light, [Online], It's time to connect, [cit. 2016-10-11]. Dostupné na internete: <<https://library.e.abb.com/public/2742b98db321b5bfc1257b26003e7835/Pow0038%20R7%20LR.pdf>>
- [5] ABB develops complete system solution for 1 100 kV HVDC power transmission, [Online], [cit. 2017-03-02]. Dostupné na internete: <<http://www.abb.com/cawp/seitp202/d83ce03a99b85dd7c1257d400041aab5.aspx>>
- [6] ENTSO-E, Technologies for Transmission System, [Online], [cit. 2017-03-27]. Dostupné na internete: <<http://tyndp.entsoe.eu/insight-reports/technology/>>
- [7] Jean-Paul Burnet; Basics of Accelerator Science and Technology at CERN. [Online] Magnet power supplies.- CAS, Budapest, 2016/10/08, [cit. 2017-02-06]. Dostupné na internete: <<https://cas.web.cern.ch/cas/Hungary2016/Lectures/BudapestLectures/Burnet.pdf>>
- [8] ABB: Why HVDC ? [Online], [cit. 2017-05-10]. Dostupné na internete: <<http://new.abb.com/systems/hvdc/why-hvdc>>

Vplyv dynamických javov na stabilitu elektrizačnej sústavy s pripojenými OZE v elektrizačnej sústave

¹Michal KOLCUN, ²Vladimír KOHAN

¹ Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita Košice, Slovenská republika

² Katedra elektroenergetiky, FEI TU Košice, Slovenská republika

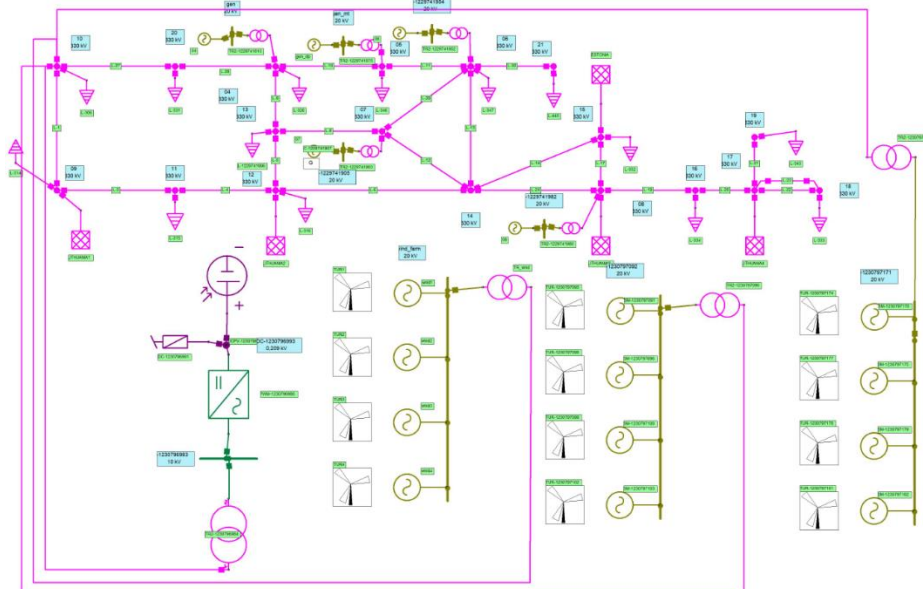
¹michal.kolcun@tuke.sk, ²vladimir.kohan.2@student.tuke.sk

Abstrakt — Práca je zameraná na problematiku spolupráce obnoviteľných zdrojov energie s konvenčnými zdrojmi z hľadiska prenosu elektrickej energie a vplyvov na dynamickú a statickú stabilitu. Jadro práce skúma a vyhodnocuje vplyv obnoviteľných zdrojov energie na elektrizačnú sústavu. Za týmto účelom bolo navrhnutých viacero modelov sústavy a následne vytvorené v programe NEPLAN pričom každej schéme bol nastavený trojfázový skrat (kvôli sledovaniu reakcie OZE) na pozorovanie dynamických dejov. Do prvého boli pripojené solárne zdroje a po nastavení skratu boli pozorované reakcie konvertorov na prudkú zmenu parametrov siete a ich schopnosť vrátiť sieť do ustáleného stavu. Podobným spôsobom bola zostrojená ďalšia schéma siete, do ktorej však bola pripojená veterná elektrárňa. Aj v tomto prípade sa dôraz kládol na reakciu PWM regulátora na vzniknutý skrat. Následne bol vytvorený model reálnej prenosovej sústavy Lotyšska do ktorej boli implementované ako solárne tak aj veterné zdroje. Do danej schémy boli implementované a odsimulované štyri odlišné scenáre, ktoré môžu nastať počas bežnej prevádzky. Špecifické udalosti, ktoré vznikajú počas prevádzky takýchto systémov boli taktiež spomenuté a vyhodnotené.

Kľúčové slová — Obnoviteľné zdroje energie, Šírkoivo pulzná modulácia, Výkonová elektronika, NEPLAN, Veterná energia, Slnčná energia, Skrat

I. ÚVOD

Za účelom skúmania dynamických dejov v reálnych podmienkach bol vytvorený model prenosovej sústavy Lotyšska. Údaje potrebné k vytvoreniu modelu boli poskytnuté Technickou univerzitou v Rige. Na Obr. 1 sa nachádza náhradná schéma spomenutej prenosovej sústavy, do ktorej boli pripojené obnoviteľné zdroje energie. Model veternej turbíny v programe NEPLAN predstavuje zapojenie synchronného generátora s regulátorom používaným pri veterných turbínach. Model bloku fotovoltaickej elektrárne je v prostredí programu NEPLAN zložený z viacerých komponentov. Ide o fotovoltaický panel, RC filter a PWM regulátor [1], [2].

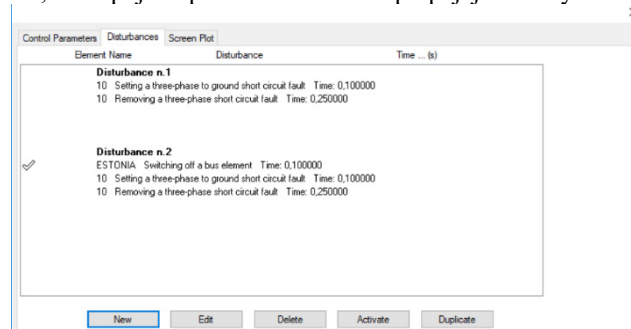


Obr. 1 Schéma prenosovej sústavy Lotyšska s pripojenými FV a WIND zdrojmi

Do siete bola implementovaná solárna elektrárň s výkonom 100 MW a taktiež tri veterné farmy (2x 21 MW, 1x 16 MW). Následne boli vytvorené štyri situácie, ktoré môžu nastať počas bežnej prevádzky.:

- Dostatok slnečného žiarenia a vetra
- Dostatok slnečného žiarenia, ale nedostatok vetra
- Dostatok vetra, ale nedostatok slnečného žiarenia
- Nedostatok slnečného žiarenia ani vetra

V každom variante bol nastavený trojfázový skrat v čase 0,1 s a odstránený bol v čase 0,25 s. Taktiež bolo v čase 0,1 s odpojené prenosové vedenie prepájajúce Lotyšsko s Estónskom.

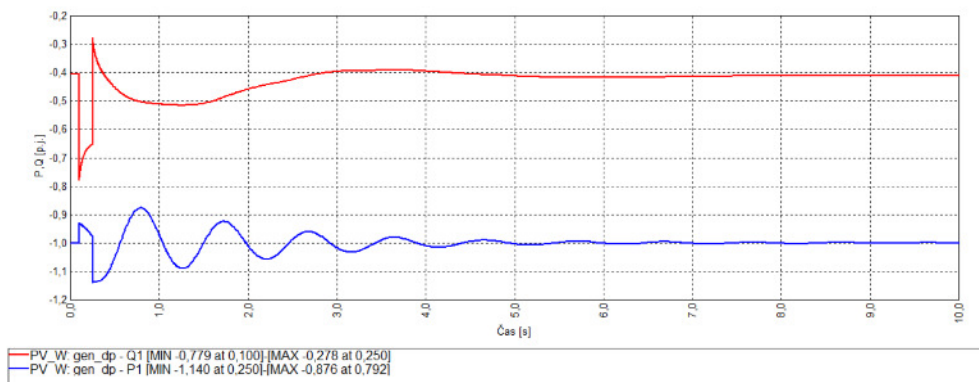


Obr. 2 Nastavenie skratu v programe NEPLAN

II. VARIANT A)

Ideálna situácia nastáva v prípade, že k dispozícii je dostatok výkonu ako zo slnečnej tak aj z veternej farmy. Sieť je dostatočne „tvrdá“ na to, aby sa po odstránení skratu vyrovnalo kolísanie a dostala sa do ustáleného stavu.

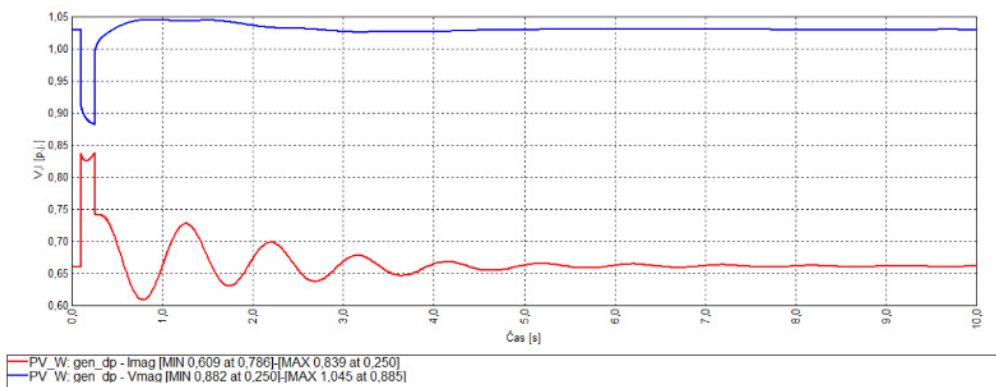
Priebeh činného a jalového výkonu generátora



Obr. 3 Priebeh činného (modrá) a jalového (červená) výkonu generátora

Z Obr. 3 je zrejmé, že po prvotnom rozkývaní sa činný výkon postupne ustálil asi v čase 5 s. Generátor pomocou regulátora budiča ustabilizoval napätie a teda jalový výkon. Hodnota jalového výkonu v momente skratu prudko klesla a až po odstránení poruchy narástla. Po následnom poklese sa taktiež stabilizovala v čase 3 s.

Priebeh prúdu a napätia generátora

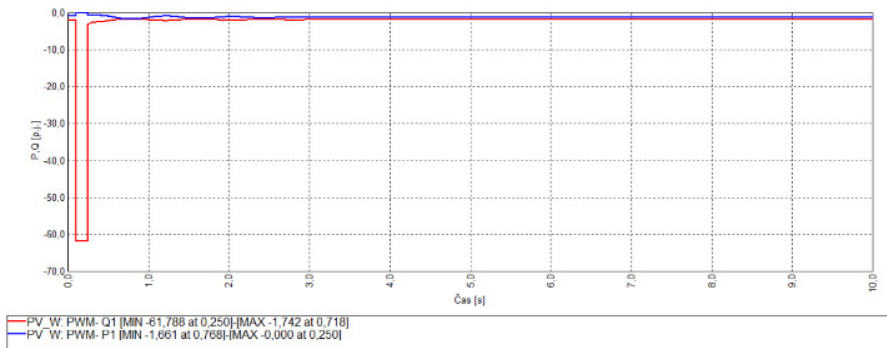


Obr. 4 Priebeh prúdu (červená) a napätia (modrá) generátora

Konštantný prúd v momente skratu strmo narastie, zatiaľ čo napätie má opačný charakter. Tým pádom je potrebné čo najrýchlejšie odstránenie akéhokoľvek typu skratu, pretože so zvyšujúcim

sa časom trvania skratu dochádza k väčšiemu kyvu generátorov. Po prekročení určitého času je možné, že generátor alebo skupina generátorov sa nedokážu ustáliť a úplne vypadnú zo synchronizmu, čo môže mať za následok úplný rozpad sústavy, alebo jej časti. Tento čas sa nazýva kritický čas (určuje čas, dokedy je potrebné eliminovať skrat) a určuje maximálnu dobu, dokedy musia zareagovať ochrany a zabezpečiť tým schopnosť generátorov ustáliť sa. V tomto prípade sa generátor ustálil a najväčšie kyvy vznikali do doby 5 s od odstránenia skratu [3].

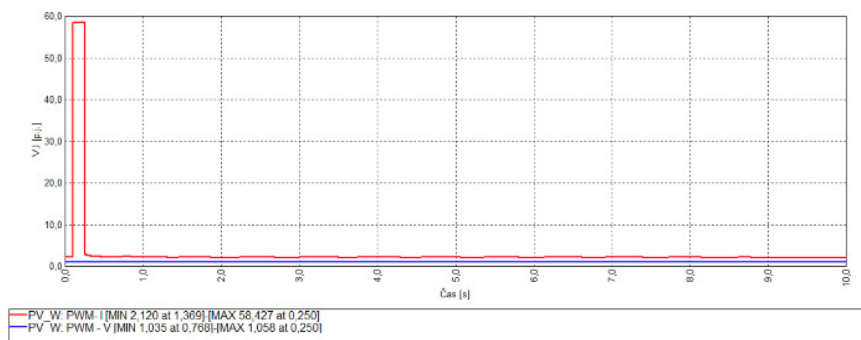
Priebieh činného a jalového výkonu PWM konvertora



Obr. 5 Priebieh činného (modrá) a jalového (červená) výkonu PWM konvertora

Zaujímavá situácia nastala pri pohľade na Obr. 5, kde je viditeľné, že priebeh činného a jalového výkonu nikdy neprekročil kladné hodnoty. Ak by boli zaznamenané kladné hodnoty, znamenalo by to, že elektráreň odoberá. Konvertor je schopný meniť reverzibilitu, avšak v prípade PWM je to blokované.

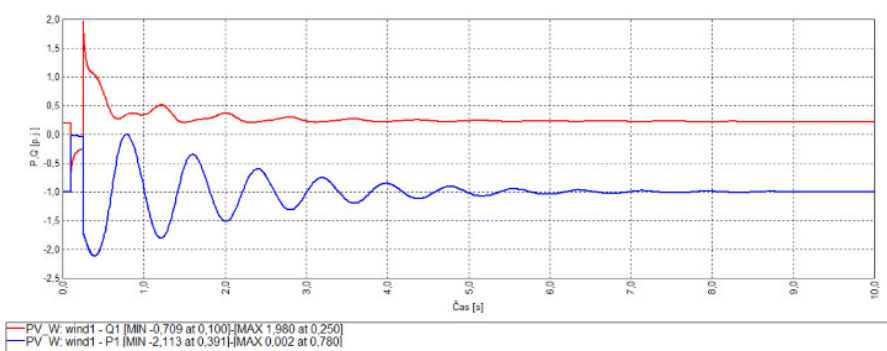
Priebieh prúdu a napätia PWM konvertora



Obr. 6 Priebieh prúdu (červená) a napätia (modrá) PWM konvertora

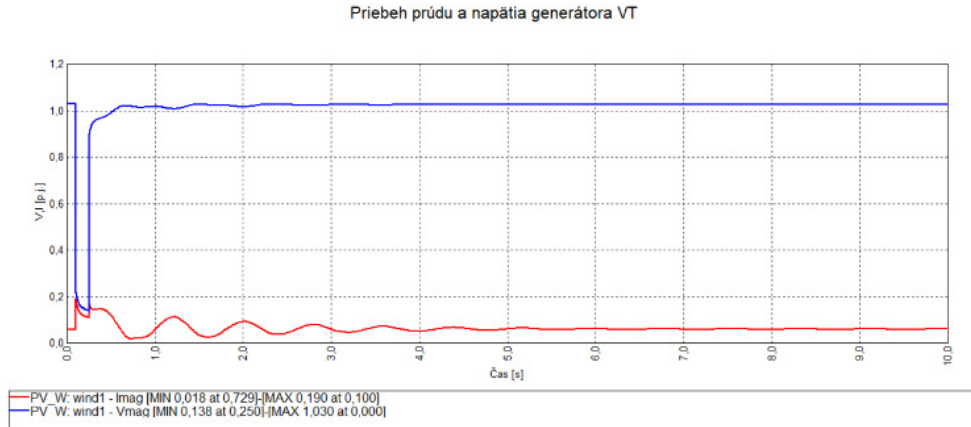
Na Obr. 5 je znázornený priebeh činného a jalového výkonu PWM konvertora, ktorý dodával výkon z PV elektrárne do siete a teda konvertoval jednosmernú zložku napätia na striedavú zložku. Keďže konvertory sú netočivé zariadenia, ktoré sa skladajú z polovodičových súčiastok (tranzistory, tyristory) predpokladáme menší prechodný dej vzhľadom na to, že tento systém obsahuje len statické (netočivé) prvky. Tieto typy konvertorov sú voči skratom chránené tým, že sa automaticky odpoja od siete, aby bola zabezpečená ochrana drahých a citlivých polovodičových prvkov. V tomto prípade (Obr. 6) by bol PWM konvertor odpojený od siete, kvôli vysokému skratovému prúdu. Skratový prúd by mal fatálne následky na kaskády v konvertore. Tým, že sa jedna len o počítačovú simuláciu a pozorovanie dynamických dejov, bolo možné ponechať PWM konvertor bez odpojenia a sledovať ako sa vysporiada so vzniknutou situáciou. Regulátor PWM konvertora je schopný vďaka vyspelej technológii spínacích výkonových polovodičových prvkov s extrémne rýchlou odozvou vplývať na rôzne zmeny v pripojenej sieti [4]. [6].

Priebieh činného a jalového výkonu generátora VT



Obr. 7 Priebieh činného (modrá) a jalového (červená) výkonu generátora VT

Tým, že veterné turbíny obsahujú točivé generátory, je možné pozorovať viditeľné oscilácie pri poruchách. Vplyvom nastavenej poruchy v programe NEPLAN sa regulátor turbíny po odstránení skratu snažil čo najrýchlejšie ustabilizovať výkon. Z Obr. 7 je možné vidieť, že generátor nevypadol zo synchronizmu na pôvodnom výkone, aký bol dodávaný pred skratom [5].



Obr. 8 Priebeh prúdu (červená) a napätia (modrá) generátora VT

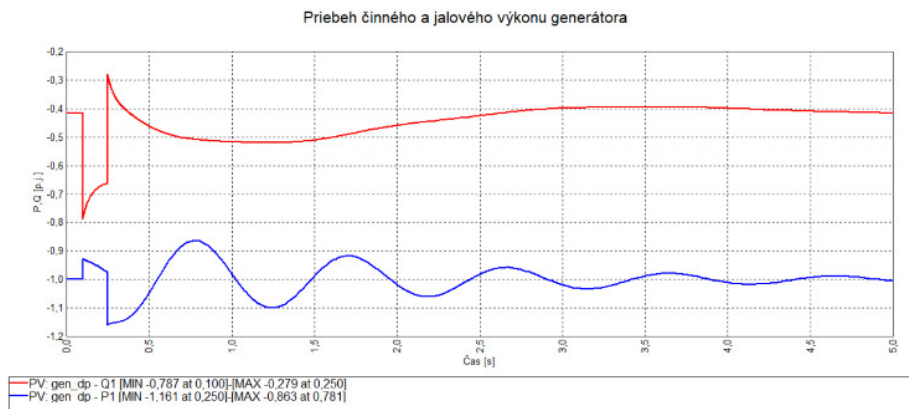
Ako je známe, pri skrate dochádza k prudkému poklesu napätia a naopak nárastu prúdu tak aj v tomto prípade tomu nebolo inak.

Tab. 1 Tabuľka nameraných hodnôt

Generátor								
	P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX	-55,188	0,792	-13,065	0,25	20,9	0,885	1,507	0,25
MIN	-71,82	0,25	-36,611	0,1	17,64	0,25	1,094	0,786
Generátor VT								
	P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX	0,042	0,78	44,925	0,25	20,6	0	1,935	0,1
MIN	-44,373	0,39	-16,087	0,1	2,76	0,25	0,183	0,729
PWM konvertor								
	P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX	0	0,25	-138,14	0,718	11,083	0,25	297,83	0,25
MIN	-195,41	0,768	-4899,83	0,25	10,842	0,768	10,81	1,369

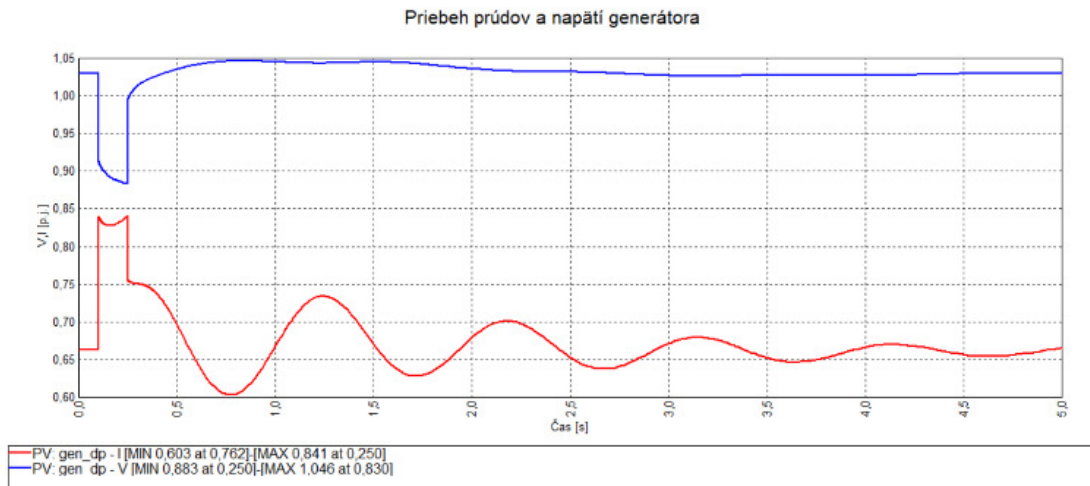
III. VARIANT B)

V tomto prípade simulovaný priebeh poukazuje na stav siete, kde po vyvolaní a následnom odstránení trojfázového skratu je do siete pripojený len solárny obnoviteľný zdroj energie.



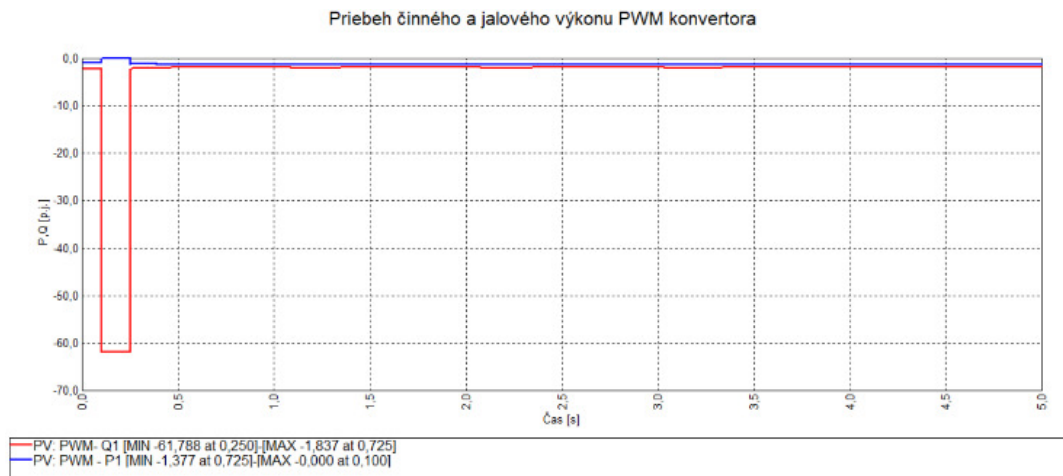
Obr. 9 Priebeh činného (modrá) a jalového (červená) výkonu generátora

V čase 0,1 s, teda po vyvolaní skratu hodnota činného výkonu jemne narastie a po následnom odstránení už výraznejšie klesá, čo ma za následok rozkolísanie hodnôt činného výkonu generátora, ktoré nie je plne ustálené ani v čase 5 s. Naopak, hodnota jalového výkonu v momente skratu prudko klesla. Preto zareagoval regulátor budiča, ktorý stabilizoval hodnotu napätia. Ako je možné vidieť na Obr. 10, jalový výkon bol teda v závislosti od napätia ustálený približne v čase 3 s rovnako ako napätie.



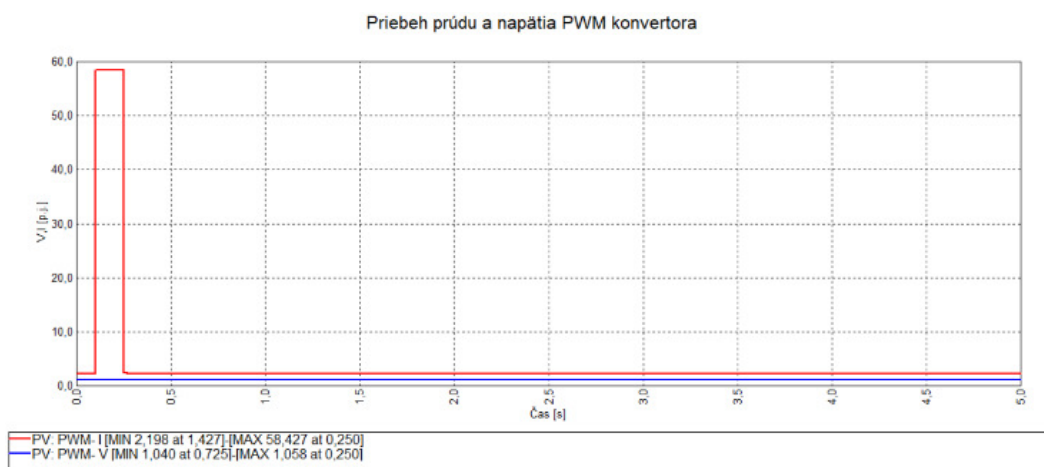
Obr. 10 Priebeh prúdov (červená) a napätí (modrá) generátora

Prúd generátora dosiahol najvyššiu hodnotu v čase odstránenia skratu 2,34 kA. Následne klesol až na hodnotu 1,68 kA a ani do času 5 s sa nebol schopný úplne ustáliť. Systém PWM dokáže pri veterných farmách minimalizovať dynamické javy a kývania generátorov [7], [9].



Obr. 11 Priebeh činného (modrá) a jalového (červená) výkonu PWM konvertora

Zdroje používajúce technológiu PWM majú vplyvom rýchleho spínania a teda rýchlejšie reakcie omnoho menej viditeľné kyvy v porovnaní s točivými strojmi. Je to spôsobené aj tým, že nemajú žiadnu zotrvačnosť. Na druhej strane točivé stroje (napríklad synchronné generátory) majú pri prechodných dejoch pomerne veľkú zotrvačnosť, čo vedie k výrazne vyšším kývaniu výkonu.



Obr. 12 Priebeh prúdu (červená) a napätia (modrá) PWM konvertora

Vysoký skratový prúd, zaznamenaný na Obr. 12 by spôsobil v reálnej prevádzke odpojenie PWM konvertora od siete. Regulátor PWM konvertora opäť veľmi rýchlo zareagoval a ustálil prúd na pôvodnú hodnotu. Situácie vyobrazené na Obr. 11 a Obr. 12 nemôžu nastať v skutočnej prevádzke, pretože by mali zničujúce účinky pre konvertor. Cieľom bolo podotknúť na schopnosť konvertora rýchlo reagovať na zmeny v sústave a to, že vplyvom rýchlych spínacích zariadení dokáže eliminovať zmeny v sieti [8], [10].

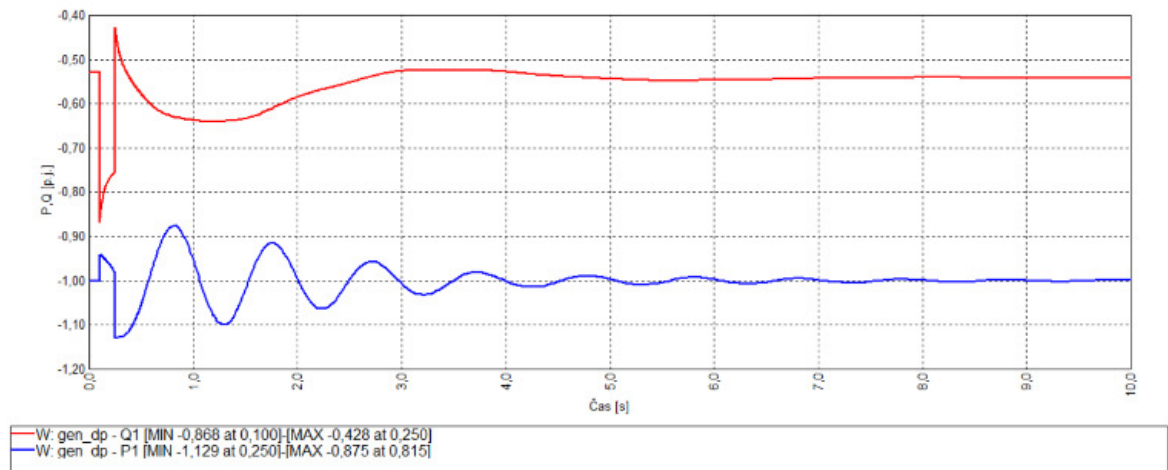
Tab. 2 Tabuľka nameraných hodnôt

		Generátor							
		P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX		-54,369	0,781	-13,041	0,25	20,92	0,83	2,344	0,25
MIN		-73,143	0,25	-36,787	0,1	17,66	0,25	1,681	0,762
		PWM konvertor							
		P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX		0	0,1	-142,28	0,725	10,555	0,25	281,535	0,25
MIN		153	0,725	-4785,7	0,25	10,375	0,725	10,591	1,427

IV. VARIANT C)

V nasledujúcej situácii boli okolnosti nastavené tak, že solárne elektrárne boli odpojené a ako OZE slúžili len veterné farmy.

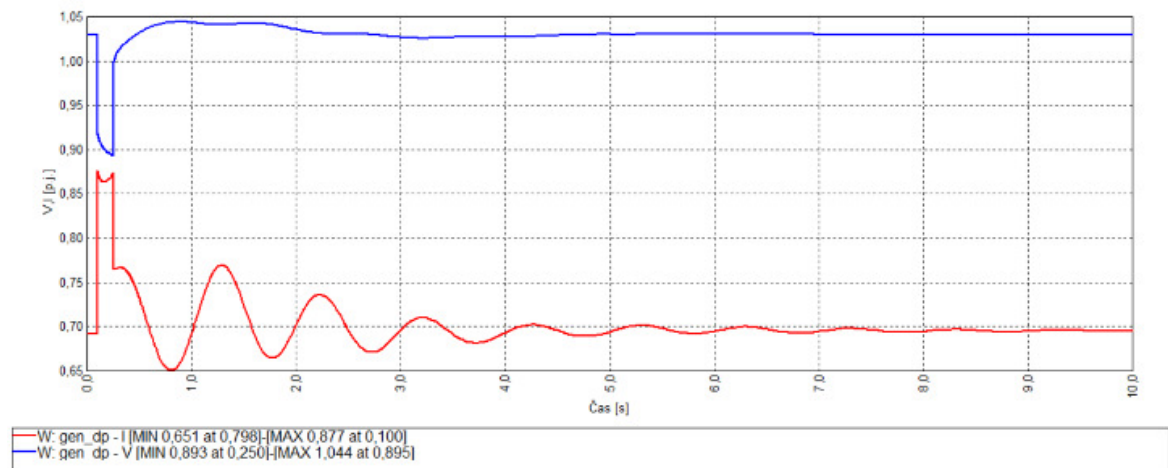
Priebeh činného a jalového výkonu generátora



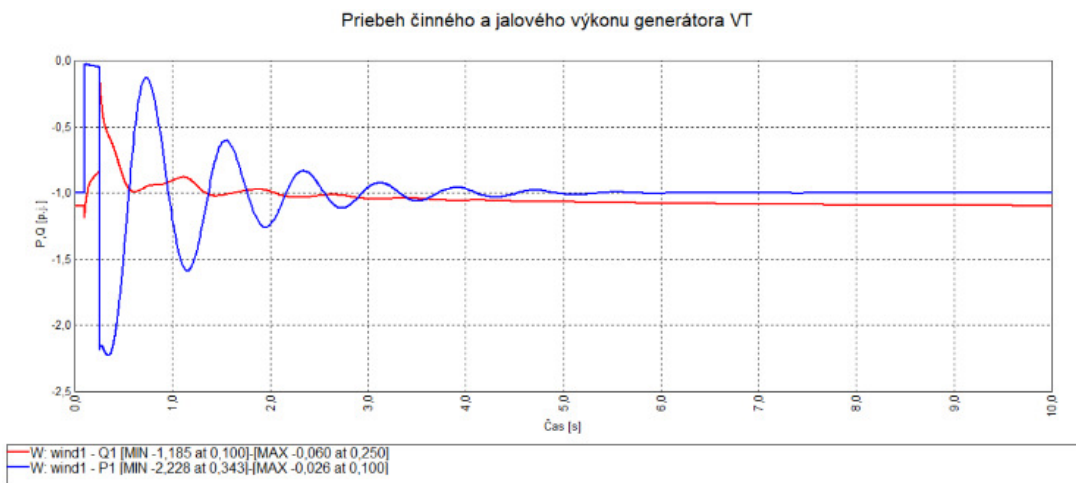
Obr. 13 Priebeh činného (modrá) a jalového (červená) výkonu generátora

V momente skratu hodnota konštantného napätia klesla, zatiaľ čo hodnota prúdu kriticky vzrástla. Rozkývavie sa generátora bolo ustálené v čase 8 s. Maximálna hodnota prúdu bola v momente skratu 2,36 kA a daná hodnota sa výrazne znížila až po odstránení skratu. Čas, potrebný na odstránenie poruchy je pre sústavu kľúčový vzhľadom na nepriaznivé účinky dlhotrvajúceho skratu, ako napríklad možnosť úplného výpadku zo synchronizmu. Z Obr. 13 a Obr. 14 je zrejmé, že generátor sa správa veľmi podobne ako v predchádzajúcom variante. Hlavné zmeny sú viditeľné pri PWM konvertore resp. generátore VT.

Priebeh prúdu a napätia generátora

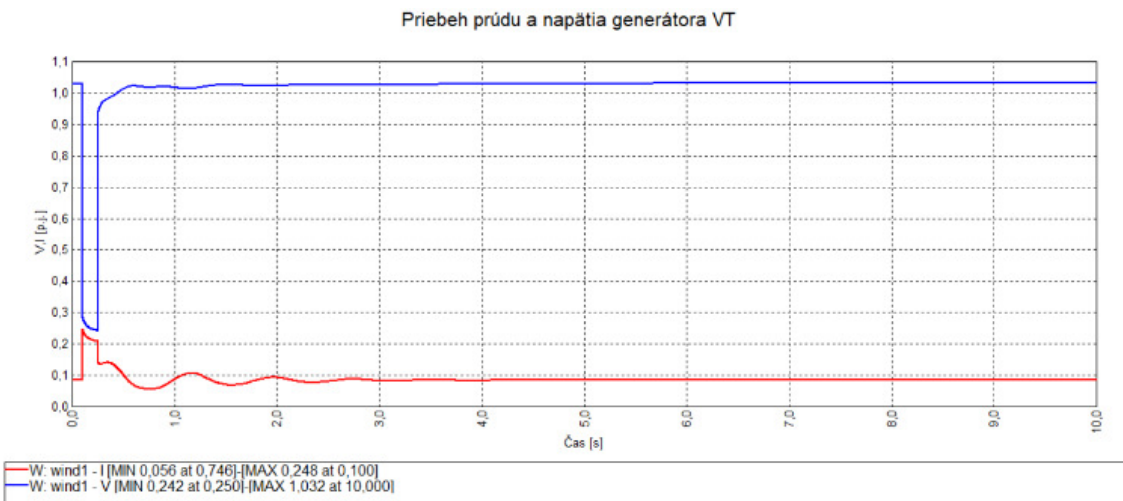


Obr. 14 Priebeh prúdu (červená) a napätia (modrá) generátora



Obr. 15 Priebek činného (modrá) a jalového (červená) výkonu generátora VT

Zaujímavý pohľad sa naskytá na priebeh činného a jalového výkonu generátora VT v kontraste s Obr. 11, kde je vykreslený priebeh činného a jalového výkonu PWM konvertora. Z Obr. 15 sa dá jednoznačne potvrdiť, že generátory nepoužívajúce technológiu PWM čelia v prípade poruchy alebo skratu značne vyššiemu kývaniu výkonu. V čase 0,1 s, teda v momente skratu generátor dodáva iba 0,546 MW, pričom po odstránení skratu to je takmer 47 MW. Hodnota dodávaného výkonu sa ustáli približne v čase 5 s na pôvodných 21 MW.



Obr. 16 Priebek prúdu (červená) a napätia (modrá) generátora VT

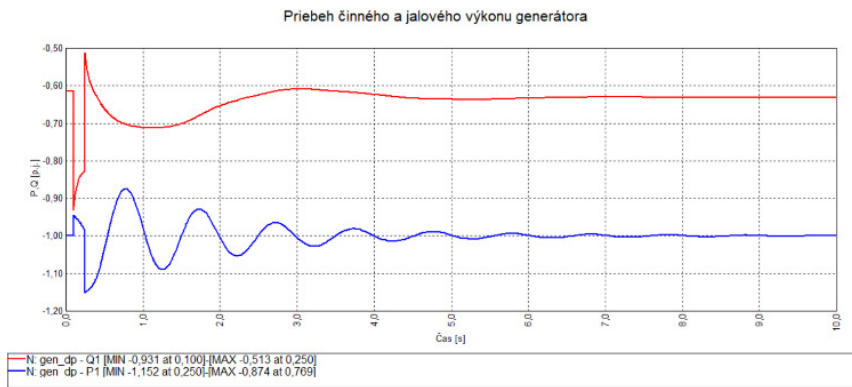
Diametrálne odlišné správanie sa synchronného generátora v protiklade s technológiou PWM je viditeľné aj na Obr. 16, kde je znázornený priebeh prúdu a napätia generátora VT. V čase skratu došlo k výraznému poklesu napätia a to až na 4,84 kV z pôvodných 20,6 kV. Naopak, hodnota skratového prúdu dosiahla trojnásobok konštantnej hodnoty a to až 1,802 kA, pričom hodnota konštantného prúdu pred vznikom skratu bola len 0,654 kA.

Tab. 3 Tabuľka nameraných hodnôt

	Generátor							
	P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX	-55,125	0,815	-17,089	0,25	20,88	0,895	2,361	0,1
MIN	-71,127	0,25	-34,657	0,1	17,86	0,25	1,753	0,798
	Generátor VT							
	P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX	-0,546	0,1	-0,556	0,25	20,64	10	1,802	0,1
MIN	-46,788	0,343	-10,989	0,1	4,84	0,25	0,407	0,746

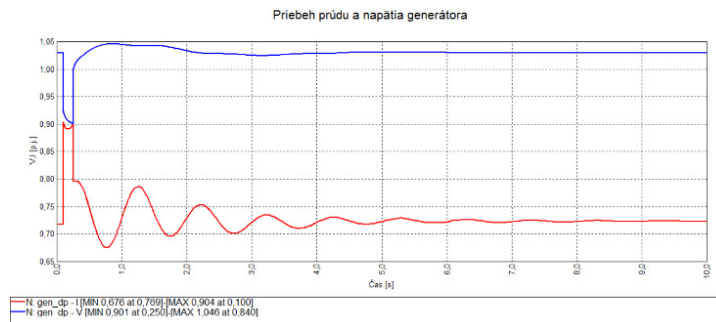
V. VARIANT D)

Model prenosovej sústavy Lotyšska bol zostrojený na základe reálnych údajov. Pripojenie OZE však bolo vytvorené len pre potreby tejto diplomovej práce. V nasledujúcom prípade je znázornená prenosová sústava Lotyšska, bez činnosti obnoviteľných zdrojov energie. Teda vypočítané hodnoty po nastavení poruchy odzrkadľujú reálne správanie sa sústavy.



Obr. 17 Priebeh činného (modrá) a jalového (červená) výkonu generátora

Hodnota jalového výkonu v momente skratu výrazne klesla a regulátoru budiča chvíľu trvalo, kým sa mu podarilo stabilizovať jalový výkon v závislosti od napätia. Hodnota činného výkonu sa ustálila na pôvodných dodávaných 63 MW až takmer po 10 s od momentu skratu.



Obr. 18 Priebeh prúdu (červená) a napätia (modrá) generátora

Konštantný prúd dosahoval pred skratom 1,87 kA a konštantné napätie malo úroveň 20,6 kV. V momente skratu dosiahol prúd svoje maximum 2,35 kA. Napätie sa vyšvihlo do svojho maxima 20,92 kV po odstránení poruchy.

Tab. 4 Tabuľka nameraných hodnôt

	Generátor							
	P [MW]	t [s]	Q [MVar]	t [s]	U [kV]	t [s]	I [kA]	t [s]
MAX	-55,062	0,769	-18,471	0,25	20,92	0,84	2,35	0,1
MIN	-72,576	0,25	-33,521	0,1	18,02	0,25	1,76	0,769

VI.

VII. VYHODNOTENIE A ODPORÚČANIA PRE PRAX (ZÁVER)

V tejto podkapitole sú zhrnuté a stručne opísané výsledky spracovania všetkých variant zapojenia veternej a solárnej elektrárne do elektrizačnej sústavy.

Tab. 5 Porovnanie nameraných hodnôt na generátore vo všetkých variantoch

	Generátor					
	A			B		
	KONŠT	MIN	MAX	KONŠT	MIN	MAX
P [MW]	-63	-71,82	-55,18	-63	-73,14	-54,37
Q [MVar]	-19,27	-36,61	-13,07	-19,49	-36,79	-13,04
U [kV]	20,6	17,64	20,9	20,6	17,66	20,92
I [kA]	1,85	1,09	1,51	1,84	1,68	2,34
	C			D		
	KONŠT	MIN	MAX	KONŠT	MIN	MAX
P [MW]	-63	-71,13	-55,13	-63	-72,58	-55,06
Q [MVar]	-21,16	-34,66	-17,09	-21,96	-33,52	-18,47
U [kV]	1,86	17,86	20,88	20,6	18,02	20,92
I [kA]	20,6	1,75	2,36	1,87	1,76	2,35

V tomto prípade sú pozorované najnižšie výkyvy činného výkonu pri variante C, kedy je do sústavy zapojená len veterná farma. Na druhej strane, variant B má najvyšší kvv činného výkonu, pričom do sústavy bola zahrnutá len solárna elektrárň.

Tab. 6 Porovnanie nameraných hodnôt PWM konvertora vo všetkých variantoch

	PWM Konvertor					
	A			B		
	KONŠT	MIN	MAX	KONŠT	MIN	MAX
P [MW]	-100	195,41	0	-100	153	0
Q [MVAr]	166,49	4899,83	138,14	170,39	4785,71	142,28
U [kV]	10,48	10,84	11,08	10,48	10,38	10,55
I [kA]	10,70	10,81	297,83	10,89	10,59	281,54

Tab. 7 Porovnanie nameraných hodnôt na generátore VT vo všetkých variantoch

	Generátor VT					
	A			C		
	KONŠT	MIN	MAX	KONŠT	MIN	MAX
P [MW]	21	44,37	0,042	21	46,79	0,54
Q [MVar]	-4,31	16,09	44,93	10,2	10,99	0,56
U [kV]	20,6	2,76	20,6	20,6	4,84	20,64
I [kA]	0,6	0,18	1,94	0,65	0,41	1,8

Výsledkom spracovania údajov môžeme tvrdiť, že činný výkon je v konverteroch dostatočne rýchlo stabilizovaný s veľmi malými osciláciami a skorým ustálením na požadovanú hodnotu. V prípade jalového výkonu, na ktorý reaguje konvertor veľmi citlivo (v prípade skratu) a snaží sa do sústavy dodať viac jalového výkonu vplyvom prudkého poklesu napätia. Týmto spôsobom sa v konverteroch vytvárajú oscilácie jalového výkonu a napätia, pričom tieto kyvy môžu predstavovať hrozbu a zničiť krehké polovodičové prechody v zariadeniach. Oscilácie sú omnoho miernejšie ako pri klasických zapojeniach synchronných generátorov cez blokovaný transformátor.

Pre praktické využitie to znamená, že veterné a solárne elektrárne prepojené cez konvertorové meniče nepredstavujú hrozbu alebo vyššie nebezpečenstvo pre dynamickú stabilitu sústavy. Práve naopak, toto riadenie môže zlepšiť a zmierniť oscilácie a kývania generátorov zapojených v ich blízkosti. Najnevyhnutnejšou potrebou týchto konvertorov je ich chránenie. Vzhľadom na to, že systémy obsahujú citlivé polovodičové zariadenia, je potrebné čo najrýchlejšie vypínanie ich od siete, kde vznikla porucha, ktorá by mohla predstavovať nebezpečenstvo pre tieto prvky [1], [11].

POĎAKOVANIE

Túto prácu podporila Vedecká grantová agentúra Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky a Slovenskej akadémie vied grantom VEGA č. 1/0372/18. Vďaka taktiež patrí aj pani profesorkám z Lotyšského hlavného mesta Rígy z univerzity RTU, ktoré nás sprevádzali, podporovali a boli nápomocné aj pri poskytovaní informácií o parametroch Lotyšskej sústavy. Všetko to bolo možné aj vďaka programu Erasmus+.

LITERATÚRA

- [1] ZICMANE, I. – BERZINA, K. – SOBOLEVSKIS, A. – KOVALENKO, S. 2017. Latvian EPS Stability's Analysis Taking into Account New Development Strategy until 2025. [online]. Riga: 1-6 p. [cit.25.4.2019] Dostupné na internete: https://www.researchgate.net/publication/321401639_Latvian_electrical_power_system_stability's_analysis_taking_into_account_new_development_strategy_until_2025.
- [2] DOI 10.1109/SIELMEN.2017.8123372.
- [3] EREMIA, M. LIU, CH. CH., EDRIS, A. A. 2016. Advanced Solutions in Power Systems; HVDC, FACTS, and Artificial Intelligence, 1. ed. New Jersey: Wiley-IEEE Press, 2016.
- [4] 271-717 p. ISBN 978-1-119-03569-5.
- [5] GESC. Vplyv rozšírenia obnoviteľných zdrojov energie na reguláciu elektrizačnej sústavy SR, analýza problémov a nedostatkov. [online] 05/2012 [cit. 25.4.2019]. Dostupné na internete: <<http://www.gesc.sk/wp-content/uploads/2014/09/vplyv-rozsirenia-oze-na-regulaciu-elektrizacnej-suustavy-SR-analyza-problemov-a-ndostatkov.pdf>>
- [6] Kosterec, M.; J. Kurimsky, M. Fol'ta, Z. Conka, S. Bucko, „Investigation of effects of non-ionizing electromagnetic fields interacting with biological systems“; In: 8th International Scientific Symposium on Electrical Power Engineering (ELEKTROENERGETIKA 2015) Pages: 564-567, 2015
- [7] M. Kolcun, Z. Conka, L. Bena, D. Medved, „Improvement of Transmission Capacity by FACTS devices in Central East Europe power system“ In: IFAC PAPERSONLINE, Vol. 49, Issue:27 Pages: 376-381, DOI: 10.1016/j.ifacol.2016.10.756, ISSN: 2405-8963
- [8] Z. Conka, M. Kolcun, G. Morva, „Utilizing of Phase Shift Transformer for increasing of Total Transfer Capacity“ In: Acta Polytechnica Hungarica, Vol. 13, Issue:5, Pages: 27-37, 2016, DOI: 10.12700/APH.13.5.2016.5.2, ISSN: 1785-8860
- [9] P. Hocko, M. Novak, M. Kolcun, Z. Conka, „Influence of photovoltaic power plants on the power system to the prediction for year 2020“ In: 14th International conference on environment and electrical engineering (EEEIC), Krakow, Poland, May 10-12, 2014, DOI: 10.1109/EEEIC.2014.6835848
- [10] M. Kolcun, M. Komatka, A. Gawlak, Z. Conka, „Benchmarking the reliability of medium-voltage lines“ In: Journal of electrical engineering, Vol. 68, Issue: 3, Pages: 212-215, 2017, DOI: 10.1515/jee-2017-0031, ISSN: 1335-3632
- [11] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

Návrh výpočtovej metodiky nasadenia inteligentných zariadení v budovách

¹Martin HALKO, ²Dušan MEDVEĎ

^{1,2} Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky Technickej univerzity v Košiciach, Slovenská republika

¹martin.halko@student.tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — Tento článok sa zaoberá súčasným stavom a výhodami inteligentných zariadení, popisom vhodných metodík pre nastavovanie smart zariadení, návrhom výpočtovej metodiky, ktorej úlohou je zefektívniť využitie inteligentných zariadení vďaka ich optimálnemu nastaveniu.

Kľúčové slová — inteligentné zariadenia, inteligentné systémy, inteligentná domácnosť, inteligentné vykurovanie, optimalizácia, optimalizačné algoritmy

I. ÚVOD

Inteligentné systémy a optimalizačné algoritmy sú témou súčasnosti a budúcnosti, keďže sa deň čo deň vyvíjajú a zdokonaľujú. Inteligentné systémy je v dnešnej dobe možné využiť takmer všade. Sú to systémy, ktorých úlohou je riadiť, regulovať, monitorovať a komunikovať. Užívateľovi prinášajú množstvo výhod a pocit pohodlia. Optimalizačné algoritmy sa stali taktiež súčasťou každodenného života, no obrovské množstvo ľudí ich vôbec nepozná aj napriek tomu, že sú tak často využívané. Je to možno tým, že najčastejšie sa využívajú v oblasti techniky a matematiky a to ľuďmi s vysokoškolským vzdelaním pre výpočty najvhodnejších kombinácií či pomerov. Ich výsledky sa ale týkajú takmer nás všetkých.

Využitie inteligentných systémov a optimalizačných algoritmov súčasne, poskytuje možnosť efektívneho riadenia systému, ktoré môže viesť k nemalým úsporám nákladov spojených s prevádzkou budov. Návravnosť investícií je však závislá od veľkosti úspor, ktoré môžeme týmto spôsobom dosiahnuť.

II. SÚČASNÝ STAV A VÝHODY INTELIGENTNÝCH ZARIADENÍ

Inteligentné zariadenia so sebou prinášajú obrovské množstvo výhod a komfortu pri monitorovaní či riadení našich domácností, hotelov, spoločností, obchodných centier a mnohých iných budov, či priestorov. Najväčšou výhodou je znižovanie nákladov, ktoré sú potrebné na prevádzku budovy. Ďalšou z veľkých výhod je napríklad prehľad o spotrebe energií na osvetlenie, vykurovanie, chladenie, prevádzku jednotlivých elektrických zariadení a taktiež pohodlné ovládanie ktorejkoľvek časti domácnosti pomocou jedného zariadenia a to napríklad pomocou smartfónu. S výhodami a komfortom prichádza aj zodpovednosť a preto musia výrobcovia zabezpečiť bezpečnosť svojich systémov tak, aby zabránili ľahkému zneužitiu zariadení a dát získaných pomocou týchto systémov, ku ktorým má mať prístup iba majiteľ alebo prevádzkovateľ objektu.

V súčasnosti sa inteligentné zariadenia využívajú takmer v každej oblasti priemyslu alebo domácnosti. Najčastejšie sa s nimi môžeme stretnúť pri výrobe, hoteloch, obchodných centrách, distribúcii elektrickej energie a za posledné roky sa objavuje čoraz väčšie množstvo domácností, ktoré tieto inteligentné systémy využívajú. Základné inteligentné systémy sú:

- *Inteligentná elektroinštalácia* (osvetlenie; zásuvky; tienenie; video vrátnik; zabezpečovací systém)
- *Inteligentná vzduchotechnika* (klimatizácia; vykurovanie; rekuperácia vzduchu a vetracie systémy)
- *Inteligentné roboty*
- *Inteligentné meracie zariadenia*

III. ROZDELENIE VHODNÝCH METODÍK A TEORETICKÝ POPIS VYBRANEJ METODIKY

Tak ako existuje veľké množstvo optimalizačných, minimalizačných a maximalizačných úloh, existuje aj mnoho metodík, pomocou ktorých sa dajú dané úlohy riešiť. Pre zvolenie správnej metódy je potrebné najprv správne určiť problém, ktorý je potrebné riešiť a jeho parametre. Výber metodiky závisí najmä od druhu problému. Najčastejšie ide o optimalizačné problémy výrobných procesov ako napríklad optimálne zostavenie výrobných plánov pre výrobu produktov, maximalizácia využitia materiálu na rezanie, minimalizácia nákladov na výrobu či prevádzku zariadení alebo minimalizácia nákladov na prevádzku budov.

Pokiaľ už máme problém definovaný a poznáme všetky podrobnosti potrebné k riešeniu daného problému, môžeme prejsť k výberu metódy riešenia. Metódy riešenia je možné rozdeliť do viacerých skupín:

- *Evolučné algoritmy* (Samo-organizujúci sa migračný algoritmus (SOMA); Diferenciálna evolúcia)
- *Expertné systémy* (Diagnostické; Plánovacie; Hybridné)
- *Lineárne programovanie* (Grafické riešenie; Bázické riešenie; Simplexová metóda)

A. SOMA: Samo-organizujúci sa migračný algoritmus

SOMA je algoritmus, ktorý existuje a používa sa od roku 1999. Jeho činnosť je založená na geometrických princípoch. SOMA algoritmus pracuje s populáciami podobne ako genetické algoritmy a výsledok po jednom evolučnom cykle je totožný s genetickými algoritmi či diferenciálnou evolúciou. To znamená, že ho je možné zaradiť medzi evolučné algoritmy napriek faktu, že behom jeho chodu nie sú vytváraní noví potomkovia, ako pri ostatných evolučných algoritmoch. Pokiaľ by sme hľadali biologickú analógiu, tak by sa dal tento algoritmus prirovnať skôr k háremovej tvorbe potomkov v stáde, ako ku klasickému výberu rodičov z populácie. Presnejšie zaradenie je medzi memetické algoritmy [1].

Pôvodná myšlienka, ktorá viedla k jeho vytvoreniu, spočíva v napodobnení správania sa skupiny inteligentných jedincov, ktorí kooperujú pri riešení spoločného problému, ako je napríklad hľadanie potravy. SOMA od svojej základnej verzie pokročila vďaka niekoľkým významným zmenám až do dnešnej podoby, kedy sa svojou robustnosťou, v zmysle nájdenia globálneho extrému, vyrovná algoritmom ako je diferenciálna evolúcia [1].

Tento algoritmus, ktorý pracuje rovnako ako ostatné evolučné algoritmy s populáciou jedincov, bol vyvinutý na princípoch, ktoré je možné odpozorovať v prírode a ktorými sa v sociálno-biologickom prostredí riadia inteligentní jedinci, ktorý kooperujú na riešení spoločného problému. Na rozdiel od ostatných evolučných algoritmov neprebíha v ňom tvorba nových jedincov filozofiou kríženia rodičov, ale je založený na kooperatívnom prehladávaní priestoru možných riešení daného problému. Vzhľadom na to, že vlastné jadro SOMA nekopíruje už zmienené evolučné princípy, ale riadi sa princípmi vychádzajúcimi zo spolupráce inteligentných jedincov migrujúcich v priestore možných riešení tak, ako ich biologickí partneri po krajine, bol pre evolučný cyklus známy ako "Generácia" zvolený názov "Migračné kolo". Príklady takéhoto správania je možné nájsť v reálnom svete. Ide napríklad o mravce, včely, vlky a podobne [1].

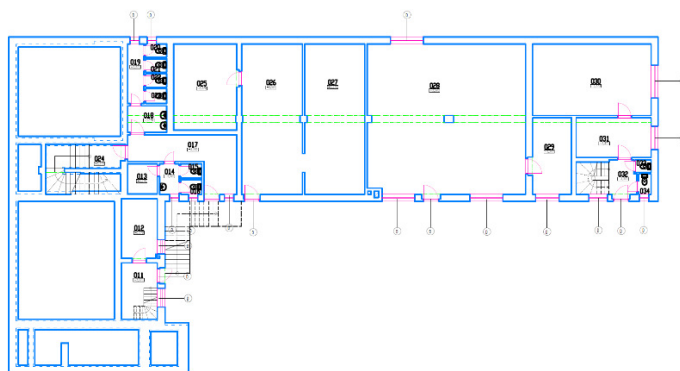
Vzhľadom na to, že hlavná myšlienka algoritmu SOMA nie je založená na princípe evolúcie ako takej, ale je založená na už zmienenom princípe "svorky", nie je klasifikovaný ako algoritmus evolučný, ale takzvaný memetický. V prípade, že pre výpočet nových pozícií sú použité evolučné princípy na úrovni genómu, sú tieto algoritmy nazývané ako genetické. Vlastnosť samoorganizácie u SOMA algoritmu plynie z faktu, že jedinci sa ovplyvňujú navzájom behom hľadania lepšieho riešenia, čo častokrát vedie k tomu, že v priestore možných riešení vznikajú skupiny jedincov, ktoré sa rozpadajú či spájajú, putujú cez prehladávaný priestor. Inými slovami si skupina jedincov sama organizuje vzájomný pohyb jedincov [1].

IV. ZVOLENÁ METODIKA A POPIS OBJEKTU

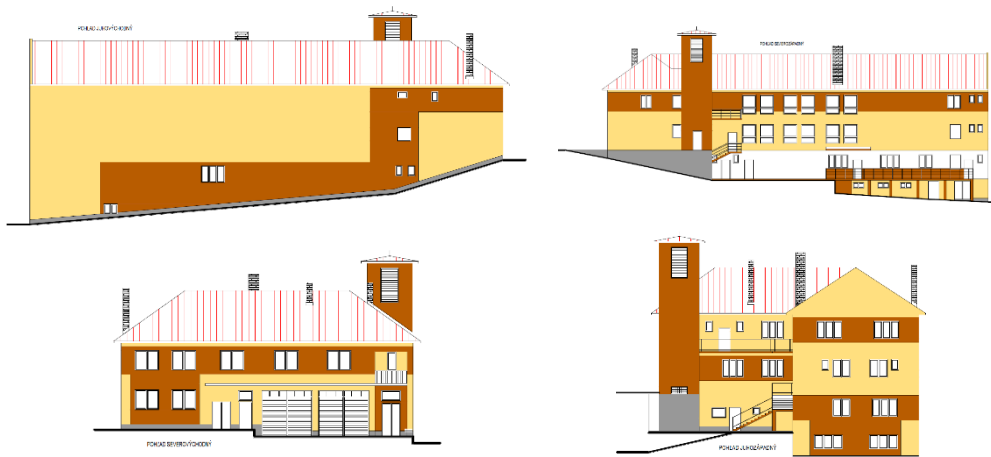
V našom prípade sa budeme venovať výpočtu optimálnej hodnoty vnútornej teploty pre budovu obecného úradu pomocou SOMA algoritmu, ktorá bude následne nastavená v regulačnom systéme. Najprv budeme potrebovať vstupné údaje o budove. Taktiež je potrebné brať ohľad na požiadavky spojené s takzvanou „Tepelnou pohodou“, ktoré sú uvedené vo vyhláske 99/2016 Z.z.; 9 227/2019 Z.z.; 355/2007 Z.z. Taktiež potrebujeme vytvoriť účelovú funkciu, pomocou ktorej budeme vypočítavať potrebnú energiu na ročné vykurovanie a zároveň minimálne ročné náklady na vykurovanie.

Prečo práve SOMA algoritmus? SOMA algoritmus sme vybrali práve vďaka jeho základnej vlastnosti a to, že dokáže prepočítať veľké množstvo kombinácií za pomerne krátky čas.

Zvolenou budovou je obecný úrad v obci Veľká Franková.



Obr. 1 Pôdorys budovy – 1. podlažie

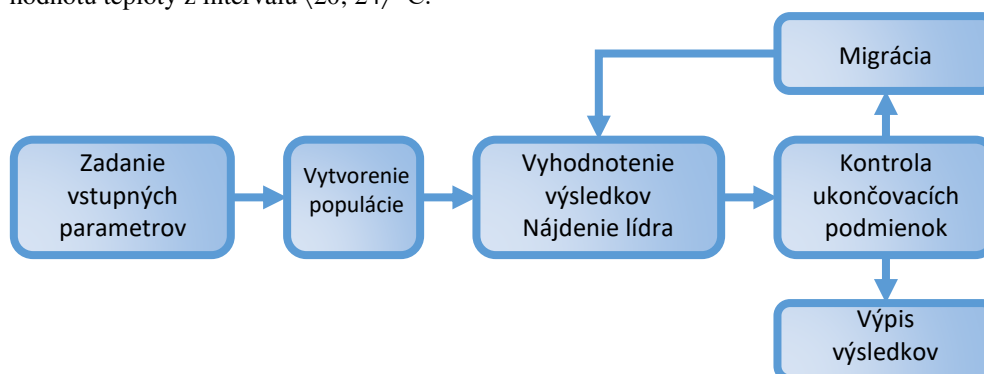


Obr. 2 Nákres budovy

Tabuľka 1
 Tabuľka hodnôt tepelných strát

Tepelné straty objektu					
Konštrukcia	Plocha (m ²)	Straty (kWh/m ² /rok)	Vykurovacie obdobie	Straty objektu (kW)	Kotol (kW)
Zvislé steny	1259,502	74,22	231 dní	31,938	39,923
Strop	460,733				
Podlaha	460,733				
Okná	135,684				
Dvere	58,71				
Spolu	2375,362				

Podľa spomínanej legislatívy sa nachádzame v triede činnosti 1 podskupine „a“, pre ktorú platí, že minimálna prípustná teplota je 20 °C a maximálna prípustná teplota je 26 °C. Minimálna optimálna teplota je 20 °C a maximálna optimálna teplota je 24 °C, čo pre nás znamená, že pre nastavenie teploty na regulačnom systéme a minimalizáciu nákladov na vykurovanie hľadáme hodnotu teploty z intervalu (20; 24) °C.



Obr. 3 Zjednodušená bloková schéma princípu SOMA algoritmu

V. VYTVORENIE ÚČELOVEJ FUNKCIE

Účelová funkcia je v tomto prípade cenovou funkciou, keďže pomocou nej sú počítané ročné náklady na vykurovanie budovy. Pomocou SOMA algoritmu sú najprv náhodne generované hodnoty teploty „ t_{is} “, ktorénásledne nastavíme na termostate. Postupným porovnávaním výsledných nákladov jedinci putujú k tej hodnote teploty, pri ktorej sú náklady na vykurovanie najnižšie. Následne jedinci migrujú a sú porovnávaní navzájom medzi sebou dovtedy, kým nie je splnená aspoň jedna ukončovacia podmienka. Výsledná hodnota funkcie po ukončení všetkých migrácií je najnižšou možnou hodnotou ročných nákladov na vykurovanie a hodnota *the_best_value* predstavuje hodnotu teploty t_{is} , pri ktorej sú náklady minimálne.

$$f_{\text{cost}}(t_{is}) = \frac{e_i \cdot e_t \cdot e_d}{\eta_o \cdot \eta_r} \cdot 24 \cdot dW \cdot \frac{d \cdot (t_{is} - t_{es})}{t_{is} - t_e} \cdot c \quad [\text{€/rok}]$$

t_i [°C]	hodnota vnútornej teploty	e_i [-]	tepelné straty infiltráciou a postupom
t_{es} [°C]	priemerná vonkajšia teplota počas vykurovacieho obdobia	e_t [-]	zníženie teploty mimo pracovnej doby
t_e [°C]	minimálna vonkajšia teplota počas vykurovacieho obdobia	e_d [-]	doba vykurovania počas týždňa
d [-]	počet dní vykurovacieho obdobia	η_o [-]	účinnosť regulácie
dW [kW]	tepelné straty objektu	η_r [-]	účinnosť rozvodu vykurovania
		c [€/kWh]	cena plynu

VI. INICIALIZÁCIA PARAMETROV

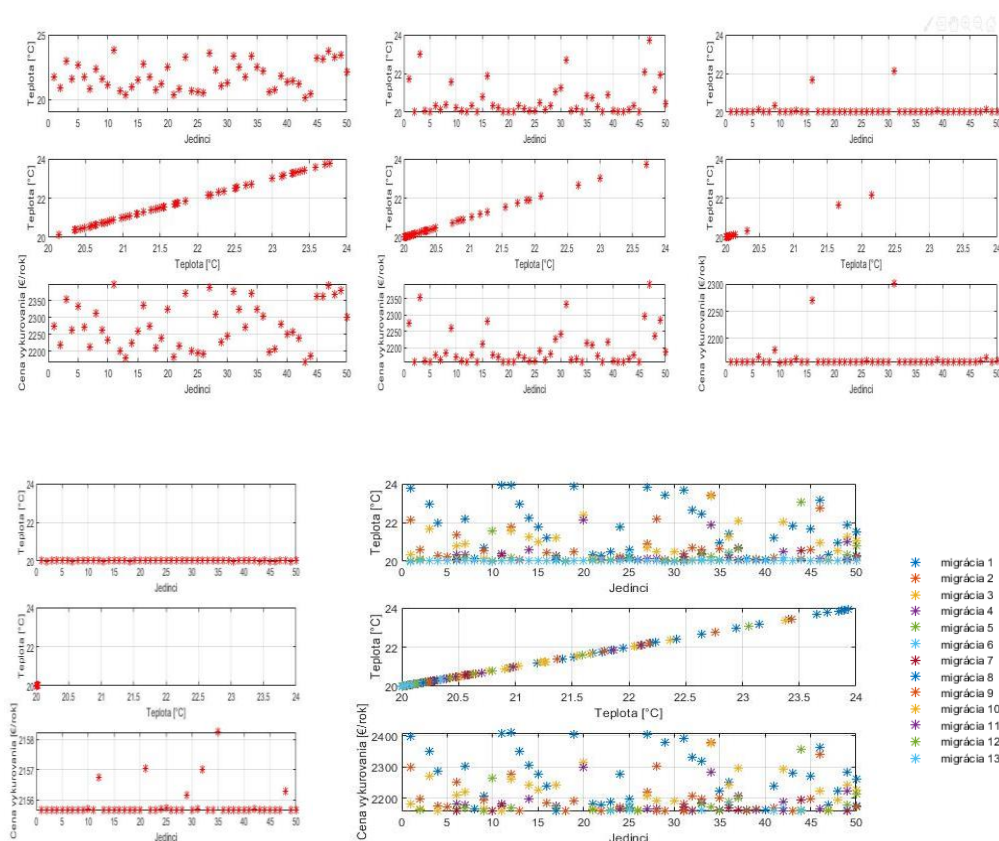
- dimension = 1 Počet dimenzií volíme 1, optimalizujeme jednu premennú.
- Step = 0.11 Krok volíme 0.11 pre dôkladné prehľadane hyperplochy.
- PRT = 0.1 PRT parameter volíme 0.1.
- PopSize = 50 Počet jedincov, volíme (štandardne) 50.
- PathLength = 1.8 Dĺžku trasy volíme 1.8, experimentálne zvolená tak, aby bol dobre viditeľný vývoj jedincov a ich putovanie.
- Migrations_Max = 100 Maximálny počet migrácií algoritmu (ukončovacia podmienka).
- FEs_Max = 10000-dimension Maximálny počet vyhodnotení algoritmu (ukončovacia podmienka).
- $d = 231$ Počet dní vykurovacieho obdobia závislý od lokality.
- $t_{es} = 3.8$ Priemerná vonkajšia teplota počas vykurovacieho obdobia [°C].
- $t_e = -15$ Minimálna vonkajšia teplota počas vykurovacieho obdobia [°C].
- $dW = 31.938$ Tepelné straty objektu [kW].
- $e_i = 0.85$ Tepelné straty infiltráciou a postupom – volí sa 0.8 až 0.9. Keďže tento parameter nepoznáme volíme priemer 0.85.
- $e_t = 0.8$ Zníženie teploty mimo pracovnej doby – volí sa 0.8 až 1. Teplotu znižujeme maximálne efektívne a volíme 0.8.
- $e_d = 0.8$ Počet pracovných dní v týždni $7/7 = 1$, $6/7 = 0.9$, $5/7 = 0.8$. Pracovná doba je pondelok až piatok, takže volíme 0.8.
- $n_o = 1$ Možnosti regulácie – volí sa 0.9 (tuhé palivo) až 1 (plyn). Používame plynový kotol, tak volíme 1.
- $n_r = 0.96$ Účinnosť rozvodu vykurovania – volí sa 0.95 až 0.98. Keďže ide o staršiu budovu a staršie rozvody volíme 0.96.
- $c = 0.0469$ Cena plynu [€/kWh].
- VarMin = 20 Minimálnu hodnotu vnútornej teploty volíme 20 °C.
- VarMax = 24 Maximálnu hodnotu vnútornej teploty volíme 24 °C.

VII. VYPOČÍTANÉ HODNOTY

Tabuľka 2

Výsledné hodnoty pre stav bez znižovania teploty a so znižovaním teploty mimo pracovnej doby a cez víkendy

	bez znižovania teploty	so znižovaním teploty
Počet migrácií	12	12
Doba výpočtu [s]	0,0888	0,0536
Vnútorná teplota [°C]	20	20
Znížená vnútorná teplota [°C]	–	16
Ročná spotreba plynu [m ³ /rok]	6803,909	4354,502
Ročná spotreba energie [MWh/rok]	71,817	45,963
Cena [€/rok]	3368,20	2155,65



Obr. 4 Prehľad migrácií

VIII. POROVNANIE ROČNÝCH NÁKLADOV NA VYKUROVANIE

Tabuľka 3

Porovnanie spotreby energie a ročných nákladov na vykurovanie s možnými riešeniami

Kalendárny rok Možné riešenia	Ročná spotreba energie [MWh/rok]	Cena [€/rok]
2017	141,749	9 569,13
2018	138,286	9 335,36
2019	132,751	8 961,72
2020 (Po zateplení bez regulácie)	79,796	3 368,2
2020 (Po zateplení s reguláciou)	45,963	2 155,65

IX. VÝPOČET INVESTÍCIE

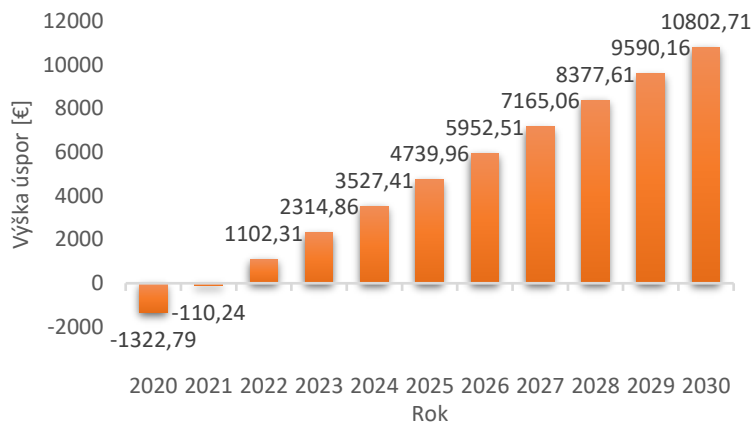
Motorický ventil je bezdrôtový (napájanie 2 AA batérie), komunikuje bezdrôtovo s centrálnou jednotkou pomocou rádio-frekvenčného signálu ZigBee 2,4 GHz, takže nie sú potrebné stavebné zásahy či realizácia novej kabeláže a v prípade potreby sa dá signál zosilniť s minimálnymi stavebnými úpravami alebo aj bez nich, ak sa použijú bezdrôtové zosilňovače signálu. Počet motorických ventilov vychádza z počtu regulovaných vykurovacích telies.

Tabuľka 4
 Investičné náklady

Zariadenie	Jednotková cena [€]	Počet kusov [ks]	Suma [€]
Vitocontrol	240	1	240
Motorický ventil	53,38	43	2295,34
Spolu			2535,34

X. NÁVRATNOSŤ A ÚSPORY

Pre zriadenie inteligentného systému regulácie je potrebná investícia vo výške 2 535,34€. Na tomto grafe si môžete všimnúť, že návratnosť investície je pomerne rýchla a to už na prelome druhého a tretieho roku používania tohto systému. Ak by bol systém inštalovaný v roku 2020, tak do roku 2030 by mohol zabezpečiť úsporu nákladov na vykurovanie až do výšky 10 800€.



Obr. 5 Graf návratnosti v jednotlivých rokoch prevádzky

XI. ZÁVER

Možností využitia inteligentných systémov a optimalizačných algoritmov je viacero. Tak, ako to bolo prakticky znázornené pri vykurovaní, je možné takúto kombináciu využiť nielen pri vzduchotechnike, ale aj v iných oblastiach, ako je výroba, zdravotníctvo, potravinárstvo, logistika a podobne. V každom odvetví ako inteligentné systémy, tak aj optimalizačné algoritmy, prinášajú obrovské množstvo výhod ako napríklad urýchlenie výrobných procesov, minimalizáciu nákladov, minimalizáciu vzniku odpadového materiálu, presnejšiu a rýchlejšiu (efektívnejšiu) diagnostiku technických zariadení, ale aj v zdravotníctve – diagnostiku pacientov, optimalizovanie logistiky, maximalizáciu ziskov a podobne.

Plynové kotly sú jednou z dobrých volieb na vykurovanie, vďaka dobrej dostupnosti plynu a možnostiam regulácie. V tomto príspevku bolo preukázané, že spolupráca inteligentných zariadení a optimalizačných algoritmov, je veľmi účinná. Úlohou optimalizačnej časti tejto spolupráce je vyhodnotiť, aká konfigurácia systému je najvhodnejšia vzhľadom na stanovené kritériá a úlohou inteligentných zariadení je následné riadenie, a regulácia systému tak, aby bola výsledná konfigurácia udržiavaná na vypočítaných (optimálnych) hodnotách.

Ďalším faktorom je fakt, že využitie niektorých smart zariadení nie vždy prináša úsporu energií alebo rýchlu návratnosť investície, ale aj napriek tomu sa používajú a to preto, že poskytujú určitý komfort. Sú to napríklad zariadenia, ktoré v podstate slúžia iba na zapínanie a vypínanie zariadení na základe hlasových povelov (okrem tejto funkcie žiadnu inú nemajú). Preto, ak chceme ušetriť a aby sa nám investícia vrátila, je potrebné vedieť, čo chceme, čo potrebujeme, respektíve aký inteligentný systém a optimalizačný (resp. minimalizačný / maximalizačný) algoritmus je pre riešenie nášho problému vhodný. V tomto príspevku bol použitý SOMA algoritmus, pre jeho jednoduchosť a rýchlosť výpočtu, pomocou ktorého bolo vypočítané nastavenie inteligentného regulačného systému vykurovania na základe určitých kritérií (pracovná doba) pre minimalizovanie nákladov na vykurovanie.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18 a tiež v rámci operačného programu Výskum a vývoj pre projekt: *Centrum výskumu účinnosti integrácie kombinovaných systémov obnoviteľných zdrojov energií*, s kódom ITMS: 26220220064, spolufinancovaný zo zdrojov Európskeho fondu regionálneho rozvoja.



Podporujeme výskumné aktivity na Slovensku/Projekt je spolufinancovaný zo zdrojov EÚ.

LITERATÚRA

- [1] Zelinka, I., „*Umělá inteligence v problémech globální optimalizace*“, Praha: BEN 2002. ISBN 80-7300-069-5.
- [2] Haľko, M., „*Návrh výpočtovej metodiky nasadenia inteligentných zariadení v budovách*“, Diplomová práca. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2020.

Porovnanie konvergenzie včelami inšpirovaných algoritmov rodín ABC, BSO a BA

¹Jozef BIROŠ, ²Marián MACH

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹jozef.biros.2@student.tuke.sk, ²marian.mach@tuke.sk

Abstrakt — Cieľom článku je problematika využitia včelami inšpirovaných algoritmov pre riešenie úloh numerickej optimalizácie. Pozornosť je zameraná na tri rodiny včelích algoritmov, ktorých jednotiacim prvkom je spoločná inšpirácia správaním sa včiel pri hľadaní a vyťažovaní zdrojov potravy v prostredí. Článok sa zaoberá tromi algoritmi reprezentujúcimi rodiny algoritmov ABC, BA a BSO majúce veľa spoločných prvkov avšak lišiacimi sa najmä prístupom k skupine aktuálnych riešení a generovaním nových kandidátov riešení počas zlepšovacej časti algoritmu. V rámci experimentovania bolo realizovaných niekoľko experimentov s týmito algoritmi na sade testovacích funkcií, zameraných na konvergenciu algoritmov ku globálnemu minimu.

Kľúčové slová — ABC algoritmus, BA algoritmus, BSO algoritmus, numerická optimalizácia

I. ÚVOD

Významná trieda prístupov k riešeniu úloh numerickej optimalizácie je reprezentovaná prehľadávacími algoritmi, ktoré prehľadávajú tzv. reálnu mnohorozmernú kocku – mnohorozmerný priestor reálnych čísel, kde každá súradnica tohto priestoru je reálnou osou reprezentujúcou možnú hodnotu niektorého z parametrov optimalizačnej funkcie (počet parametrov funkcie určuje rozmernosť kocky). Každý bod tohto priestoru definuje hodnotu každého parametra a takým spôsobom reprezentuje kandidáta na riešenie problému.

Dôležitou triedou prehľadávacích algoritmov sú algoritmy, ktoré našli svoj zdroj inšpirácie vo výtvoroch prírody. Medzi týmito algoritmi významnú úlohu hrajú tie, ktoré sa inšpirovali pozorovaniami sveta hmyzu. V tejto kategórii patrí popredné miesto včelám – včelie algoritmy simulujú správanie sa včiel v rozličných situáciách. Je možné si vybrať algoritmus inšpirujúci sa sociálnou organizáciou roja, organizáciou úľa, procesom párenia sa, vyhľadávaním a zbieraním potravy a podobne, pričom na výber je včela medonosná, čmeliak alebo osa [1].

Pozornosť tohto článku je zameraná na tri rodiny algoritmov, inšpirovaných zbieraním potravy včelami – ABC (Artificial Bee Colony), BSO (Bee Swarm Optimization) a BA (Bees Algorithm), pričom pre každú z nich bol uvažovaný typický reprezentant danej rodiny.

II. ALGORITMY RODÍN ABC, BA A BSO

Rôzne včelie algoritmy sa navzájom líšia, avšak majú aj veľa prvkov spoločných. Najprv uvedieme vlastnosti spoločné všetkým trom rodinám a až v ďalšom sa budeme venovať vzájomným odlišnostiam.

A. Spoločné vlastnosti uvažovaných rodín

Každý algoritmus týchto rodín je populačným algoritmom, ktorý v každej fáze v D rozmernom priestore S všetkých možných riešení identifikuje podmnožinu všetkých riešení – N bodov v tvare numerických vektorov $X_i = \{x_{i,1}, \dots, x_{i,D}\}$, $i = 1, \dots, N$.

Na začiatku hľadania sú tieto riešenia prvotne inicializované. Táto inicializácia sa deje pomocou náhodného výberu hodnôt zložiek všetkých riešení z definičného intervalu hodnôt s uniformnou hustotou pravdepodobnosti postupným generovaním zložiek

$$x_{i,j} = x_{min,j} + \chi(x_{max,j} - x_{min,j}) \quad (1)$$

kde $i=1, \dots, N$, $j=1, \dots, D$, χ je náhodným číslom z intervalu $\langle 0, 1 \rangle$ a samotná zložka je generovaná z intervalu $\langle x_{min}, x_{max} \rangle$.

Po ukončení inicializačnej časti nasleduje iteračná časť algoritmu s $iter_{max}$ iteráciami, keď dochádza k náhrade aktuálnych riešení novými riešeniami. Táto náhrada môže mať dvojaký charakter:

- zlepšovanie jednotlivých riešení, kde pod zlepšením riešenia X_i sa rozumie náhrada tohto riešenia iným riešením V_i , ktoré je odvodené od pôvodného riešenia X_i ,
- opustenie jednotlivých riešení, kde pod opustením riešenia X_i sa rozumie náhrada tohto riešenia iným riešením V_i , ktoré bolo generované náhodným spôsobom.

Náhrada zlepšením sa realizuje iba v prípade, ak platí $f(X_i) > f(V_i)$, teda ak dochádza k nájdeniu nového riešenia s menšou hodnotou optimalizovanej funkcie f než bola u pôvodného riešenia (prípád minimalizácie danej funkcie). Ak nedochádza k zlepšeniu, tak sa náhrada nerealizuje a pôvodné riešenie X_i prežíva.

Náhrada opustením sa tiež nevykonáva vždy ale sa deje iba pri splnení určitej podmienky, pričom rôzne algoritmy môžu používať rôzne podmienky.

Spôsob delenia jednotlivých iterácií algoritmu na fázy, delenia množiny aktuálnych riešení na skupiny, generovania nového riešenia V_i , počet zlepšovaných riešení a počet opúšťaných riešení sú rôzne pre rôzne algoritmy daných rodín algoritmov.

B. Rodina ABC

Algoritmus ABC (Artificial Bee Colony) [2] bol prvým algoritmom danej rodiny algoritmov. Množina aktuálnych riešení nie je delená na skupiny, v každej fáze iteračného cyklu sa pracuje so všetkými riešeniami.

Každá iterácia tohto algoritmu pozostáva z troch fáz:

1. urobí sa pokus o zlepšenie každého aktuálneho riešenia (teda spolu N pokusov, jeden pokus na každé aktuálne riešenie),
2. urobí sa spolu ďalších N pokusov o zlepšenie riešení, pričom však riešenia sú volené náhodne s preferenciou tých riešení, ktoré reprezentujú menšiu hodnotu optimalizovanej funkcie (teda nejaké konkrétne riešenie nemusí byť vôbec uvažované alebo naopak môže byť zapojené viackrát),
3. riešenie, ktoré prežíva viac pokusov o zlepšenie bez jeho náhrady (strážené premennou *limit*) lepším riešením môže byť opustené a nahradené iným riešením.

Nové riešenie V_i (v prvej aj druhej zlepšovacej fáze iterácie) bolo odvodené od pôvodného riešenia X_i zmenou hodnoty jednej zložky

$$v_{i,j} = x_{i,j} + \rho(x_{i,j} - x_{m,j}) \quad (2)$$

kde j je náhodne zvolený index zložky z množiny $\{1, \dots, D\}$, X_m je náhodne zvolené riešenie (rôzne od X_i) a ρ je náhodným číslom z intervalu $\langle -1, 1 \rangle$. Nové riešenie sa tak generuje na základe porovnania aktuálneho riešenia s iným, náhodne zvoleným riešením, pričom zmena nastáva iba v jednej, náhodne vybranej zložke.

V tretej fáze iterácie sa nahrádza maximálne jedno (najdlhšie prežívajúce z tých, ktorých prežívanie už presiahlo limitnú hodnotu) riešenie novo inicializovaným riešením, ktoré je novo generované podľa (1).

C. Rodina BA

Algoritmus BA (Bees Algorithm) [3] bol prvým algoritmom danej rodiny algoritmov. Množina aktuálnych riešení je na začiatku každej iterácie zotriedená podľa kvality riešení a rozdelená na tri skupiny – elitná skupina (N_e riešení), neelitná skupina (N_n riešení) a zvyšok ($N - N_e - N_n$ riešení). V každej fáze iteračného cyklu sa pracuje iba s jednou skupinou riešení.

Každá iterácia tohto algoritmu pozostáva z troch fáz:

1. urobí sa n_e pokusov o zlepšenie každého aktuálneho riešenia z elitnej skupiny (teda spolu $N_e * n_e$ pokusov),
2. urobí sa n_n pokusov o zlepšenie každého aktuálneho riešenia z neelitnej skupiny (teda spolu $N_n * n_n$ pokusov),
3. všetky riešenia zo zvyšnej skupiny sú opustené – každé z nich je nahradené novo vygenerovaným riešením podľa (1).

Viac pokusov o zlepšenie riešenia znamená snahu o jeho postupné zlepšovanie – riešenie sa nemusí zmeniť vôbec (teda k zlepšovaniu nedošlo) alebo môže byť zlepšené raz alebo aj viackrát za sebou. Typicky platí nerovnosť $n_e > n_n$ a teda riešeniam z elitnej skupiny je venovaná väčšia pozornosť než tým, ktoré sú zaradené v neelitnej skupine.

Nové riešenie V_i (v prvej aj druhej zlepšovacej fáze iterácie) bolo odvodené od pôvodného riešenia X_i zmenou hodnoty jednej zložky

$$v_{i,j} = x_{i,j} + \rho * r \quad (3)$$

kde j je náhodne zvolený index zložky z množiny $\{1, \dots, D\}$, r je maximálne povolená veľkosť zmeny a ρ je náhodným číslom z intervalu $\langle -1, 1 \rangle$. Nové riešenie sa tak generuje na základe lokálneho prehľadávania oblasti s polomerom r okolo aktuálneho riešenia, pričom zmena nastáva iba v jednej, náhodne vybranej zložke.

Na konci každej iterácie je hodnota polomera r zmenená podľa vzťahu

$$r = r * r_{damp} \quad (4)$$

kde r_{damp} je tlmiaci faktor (menší ako 1). S pribúdajúcimi iteráciami sa teda preskúmava čoraz menšie okolie aktuálnych riešení.

D. Rodina BSO

Algoritmus BSO (Bee Swarm Optimization) [4] bol prvým algoritmom danej rodiny algoritmov. Množina aktuálnych riešení je na začiatku každej iterácie algoritmu zotriedená podľa kvality riešení a rozdelená na tri skupiny – elitná skupina $((N - N_z)/2$ riešení), neelitná skupina $((N - N_z)/2$ riešení) a zvyšok (N_z) riešení. Elitná skupina je teda rovnako veľká ako neelitná skupina. V každej fáze iteračného cyklu sa pracuje iba s jednou skupinou riešení.

Každá iterácia tohto algoritmu pozostáva z troch fáz:

1. urobí sa pokus o zlepšenie každého aktuálneho riešenia z elitnej skupiny (teda spolu $(N - N_z)/2$ pokusov),
2. urobí sa pokus o zlepšenie každého aktuálneho riešenia z neelitnej skupiny (teda spolu $(N - N_z)/2$ pokusov),
3. všetky riešenia zo zvyšnej skupiny sú opustené – každé z nich je nahradené novo vygenerovaným riešením.

Nové riešenie V_i v prvej zlepšovacej fáze iterácie bolo odvodené od pôvodného riešenia X_i zmenou hodnoty každej zložky j

$$v_{i,j} = x_{i,j} + \rho_l \omega_l (x_{l,j} - x_{i,j}) + \rho_g \omega_g (x_{g,j} - x_{i,j}) \quad (5)$$

kde X_l je lokálne najlepšie riešenie (najlepšie riešenie z tých riešení, ktoré boli predkami aktuálneho riešenia X_i), X_g je globálne najlepšie riešenie (najlepšie riešenie z doteraz nájdených), ω_l a ω_g sú váhy nastavujúce dôležitosť lokálne a globálne najlepšieho riešenia a ρ_l a ρ_g sú náhodnými číslami z intervalu $\langle 0, 1 \rangle$. Nové riešenie sa tak generuje na základe približovania aktuálneho riešenia k lokálne a globálne najlepšiemu riešeniu.

Nové riešenie V_i v druhej zlepšovacej fáze iterácie bolo odvodené od pôvodného riešenia X_i zmenou hodnoty každej zložky j

$$v_{i,j} = x_{i,j} + \rho_e \omega_g (x_{e,j} - x_{i,j}) \quad (6)$$

kde X_e je náhodne vybrané riešenie z elitnej skupiny (s preferovaním lepších riešení oproti horším) a ρ_e je náhodné číslo z intervalu $\langle 0, 1 \rangle$. Nové riešenie sa tak generuje na základe približovania aktuálneho riešenia z neelitnej skupiny k niektorému aktuálnemu riešeniu z elitnej skupiny.

Nové riešenie V_i v tretej fáze iterácie, ktoré nahradí opustené riešenie X_i , je generované zmenou každej zložky pôvodného riešenia X_i

$$v_{i,j} = x_{i,j} + \rho \left[\tau_{max} - \frac{iter}{iter_{max}} (\tau_{max} - \tau_{min}) \right] x_{i,j} \quad (7)$$

kde ρ je náhodné číslo z intervalu $\langle -1, 1 \rangle$ a parametre τ_{max} a τ_{min} umožňujú dynamickú zmenu veľkosti maximálnej zmeny v závislosti na aktuálnej iterácii $iter$.

III. VYBRANÉ EXPERIMENTY S KONVERGENCIU

A. Testovacie funkcie

Pre experimentovanie s algoritmi boli použité štyri testovacie funkcie, ktoré sú často využívané pre testovanie optimalizačných algoritmov [6]. Výber bol rovnaký ako v [5] aby výsledky boli porovnateľné. Boli zvolené tieto funkcie:

Ackleyho funkcia je definovaná vzťahom

$$f_1(X) = (-20) \exp \left[(-0.2) \sqrt{\frac{1}{D} \sum_{j=1}^D x_j^2} \right] - \exp \left[\frac{1}{D} \sum_{j=1}^D \cos(2\pi x_j) \right] + 20 + \exp(1) \quad (8)$$

Globálny extrém funkcie je $f_1(0, \dots, 0) = 0$. Funkcia je charakterizovaná málo sa zvažujúcim tvarom, čo je kompenzované strmým poklesom v okolí počiatku. Superponovaná periodická zložka vytvára veľké množstvo lokálnych extrémov.

Rastriginova funkcia je definovaná vzťahom

$$f_2(X) = 10D + \sum_{j=1}^D [x_j^2 - 10 \cos(2\pi x_j)] \quad (9)$$

Globálny extrém je $f_2(0, \dots, 0) = 0$. Funkcia má konvexný tvar s mnohými lokálnymi minimami, ktoré sú pravidelne distribuované a sú tým väčšie, čím sú vzdialenejšie od globálneho minima.

Rosenbrockova funkcia je definovaná vzťahom

$$f_3(X) = \sum_{j=1}^{D-1} [100(x_{j+1} - x_j^2)^2 + (x_j - 1)^2] \quad (10)$$

Globálny extrém je $f_3(1, \dots, 1) = 0$. Funkcia je unimodálna, extrém leží v úzkom parabolickom údolí, ktoré je jednoduché nájsť – avšak konvergencia k minimu v tomto údolí je obtiažna kvôli plochosti dna údolia.

Salomonova funkcia je definovaná vzťahom

$$f_4(X) = 1 - \cos\left(2\pi \sqrt{\sum_{j=1}^D x_j^2}\right) + 0.1 \sqrt{\sum_{j=1}^D x_j^2} \quad (11)$$

Globálny extrém je $f_4(0, \dots, 0) = 0$. Funkcia je tiež vysoko multimodálna so symetrickým tvarom.

B. Počiatočné nastavenie parametrov

Implementované algoritmy vyžadujú nastavenie množiny parametrov na vhodné hodnoty. Avšak nejaké konkrétne nastavenie, vhodné pre nejakú testovaciu funkciu, môže byť menej vhodným v prípade funkcie inej. Z tohto dôvodu sme rovnako ako v [5] zvolili osobitné nastavovanie pre každú testovaciu funkciu. Samotný postup nastavenia aj teraz pozostával z dvoch po sebe idúcich fáz:

- voľba počiatočných hodnôt,
- iteračné zlepšovanie hodnôt.

V prvej fáze bol pre každý algoritmus definovaný tzv. počiatočný stav – stav, v ktorom boli použité určité preddefinované hodnoty (doporučené rôznymi literárnymi zdrojmi). Tieto hodnoty sú v nasledujúcej tabuľke

Tabuľka 1
 Preddefinované hodnoty parametrov

	N	iter _{max}	D			
Všetky algoritmy	30	3000	10			
ABC	Limit					
	100					
BA	N _e	N _n	n _e	n _n	r	r _{damp}
	5	10	30	15	1.0	0.99
BSO	N _z	ω _l	ω _g	τ _{min}	τ _{max}	
	6	0.5	0.5	0.2	2.0	

V druhej fáze bol vždy menený iba jeden parameter zatiaľ čo hodnoty ostatných boli pevné, pričom po nastavení jedného parametra bol nastavovaný druhý atď. Hodnoty niektorých parametrov sa ukázali byť použiteľné pri všetkých funkciách (napríklad *limit=300*), iné parametre pre rôzne funkcie vykazovali rôzne hodnoty ako svoju najlepšiu voľbu.

C. Porovnanie výkonnosti algoritmov

Z dôvodu stochastickej povahy algoritmov bolo pre každú funkciu a každý algoritmus vykonaných 30 experimentov a výsledky boli spriemerené. Dosiahnuté výsledky [7] sú zhrnuté v tabuľke tab. 2.

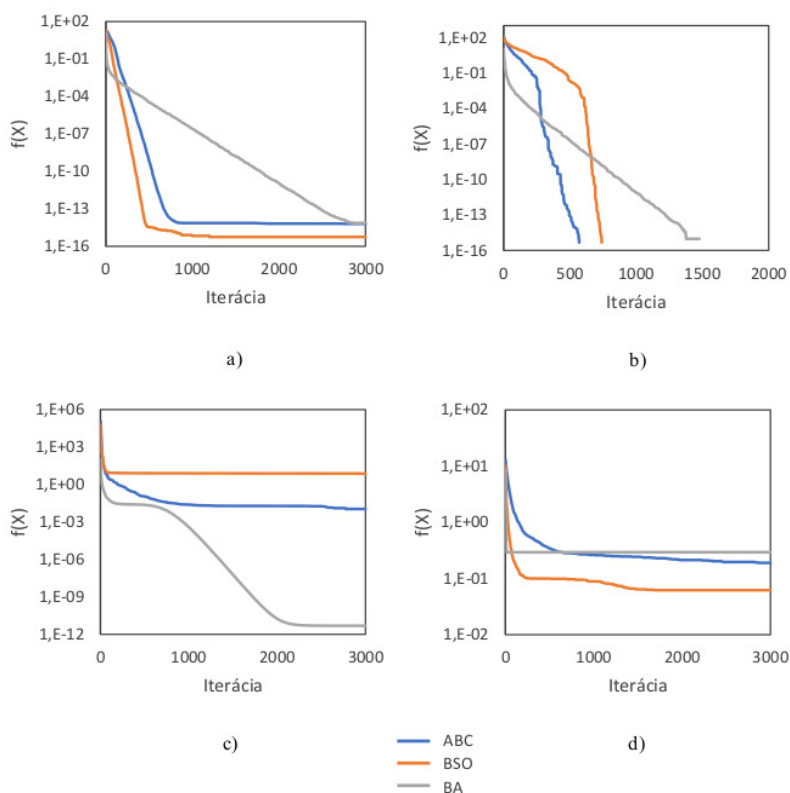
Každý z algoritmov po uplynutí 3000 iterácií dospel k rovnakému výsledku pri Rastriginovej funkcii, pri ostatných funkciách sa dosiahnuté výsledky líšili – a žiadny z algoritmov pre ne nenašiel presnú polohu globálneho minima. Algoritmus ABC zaostal za ostatnými konkurentmi, čo je trochu prekvapujúce, pretože z hľadiska využívania vzájomných vzťahov medzi aktuálnymi riešeniami je strednou voľbou – modifikuje aktuálne riešenie jeho približovaním

k inému riešeniu, zatiaľ čo konkurenti aktuálne riešenie alebo približujú súčasne k dvom riešeniam alebo takéto približovanie nevyužívajú vôbec.

Tabuľka 2
 Priemerné najlepšie riešenia

	ABC	BA	BSO
Ackley	6.365279E-15	7.312669E-15	5.625130E-16
Rastrigin	0.000000E+00	0.000000E+00	0.000000E+00
Rosenbrock	1.112347E-02	4.920303E-12	7.417538E+00
Salomon	1.898739E-01	2.932067E-01	6.193712E-02

Grafické zobrazenie konvergenčnej výkonnosti algoritmov na každej z testovaných funkcií je zobrazené na nasledujúcom obrázku.



Obr. 1 Konvergenca algoritmov ABC, BA a BSO na funkciách: Ackley (a), Rastrigin (b), Rosenbrock (c) a Salomon.(d)

Na grafoch je možné rozpoznať niekoľko rôznych konvergenčných chovaní. Najvýraznejšie sú tieto:

- Dosiahnutie niektorého lokálneho minima a uviaznutie v ňom (priebeh končí vodorovným úsekom) – ABC v prípade a), BSO v prípade c) a BA v prípade d).
- Neustále prechádzanie z jedného riešenia na druhé bez uviaznutia v lokálnom extréme (priebeh neustále pokračuje v miernom sklone) – ABC v prípadoch c) a d).
- Uviaznutie v lokálnom minime na určitý čas avšak s následným únikom (priebeh strieda vodorovné úseky s klesajúcimi) – BA v prípade c) a BSO v prípade d).

Dôležité je to, že žiadny z algoritmov nie je fixovaný iba na jeden konvergenčný vzor, čo znamená že algoritmy dokážu vykazovať rôzne chovanie podľa charakteru optimalizačnej funkcie.

V zásade sa nedá povedať, že niektorý z algoritmov je jednoznačne lepší ako ostatné. ABC bol síce podľa tab. 2 horší, avšak to že v niektorých prípadoch neviazol znamená, že pri vyššom povolenom počte iterácii by bol schopný dosiahnuť kvalitnejšie riešenie.

D. Rozšírené porovnanie výkonnosti

Z hľadiska detailnejšieho porovnania algoritmov bolo (rovnako ako v [5]) realizované porovnanie na rozšírenej množine testovacích funkcií. Okrem vyššie spomínaných štyroch funkcií boli použité tieto dodatočné testovacie funkcie: *sférická* funkcia, *Griewankova* funkcia, *Levyho* funkcia, *Schweffelova* funkcia, *Quingova* funkcia a *Zakharova* funkcia [6]. Celkom bolo teda použitých desať testovacích funkcií.

Dosiahnuté výsledky sú v nasledujúcej tabuľke, ktorá udáva relatívnu výkonnosť testovaných algoritmov navzájom.

Tabuľka 3
Relatívne porovnanie testovaných algoritmov

	ABC	BSO	BA
ABC	-	6	5
BSO	3	-	3
BA	3	6	-

Tabuľka hovorí, že napríklad algoritmus ABC bol na šiestich funkciách lepší ako algoritmus BSO a naopak BSO bol lepší ako ABC na troch testovacích funkciách (a teda na jednej funkcii ich výkonnosť bola porovnateľná).

Z výsledkov je vidieť zlepšenie výkonnosti algoritmu ABC a jeho dotiahnutie sa na úroveň ostatných dvoch algoritmov (iba na jednej funkcii bol horší od oboch, na ostatných funkciách bol buď najlepší alebo druhý najlepší). Vo vzájomnom porovnaní medzi algoritmi BA a BSO vychádza prvý z nich ako vhodnejší na použitie. Netreba však zabúdať, že vďaka štruktúre algoritmov napriek rovnakej hodnote $iter_{max}$ algoritmus BA v skutočnosti generoval o rád viac riešení než BSO, čím mu bolo umožnené rozsiahlejšie prehľadávanie priestoru možných riešení.

IV. ZÁVER

Na základe vykonaných experimentov je zrejma mierna prevaha algoritmu ABC nad algoritmi BA a BSO. Netreba však zabúdať na to, že táto prevaha nie je univerzálna a nie je problém nájsť také funkcie pri optimalizácii, pri ktorých jeho výkonnosť je naopak horšia. Experimenty naznačili, že každý z testovaných algoritmov má dôvod na existenciu a je preto vhodné všetky zaradiť do výbavy potrebnej pre riešenie úloh numerickej optimalizácie.

POĎAKOVANIE

Táto publikácia vznikla vďaka podpore v rámci výskumu podporovanom Národným projektovým grantom pre výskum a vývoj 1/0773/16 2016-2019 „Metódy umelej inteligencie pre inteligentnú robotiku založené na cloudovom počítaní“.

LITERATÚRA

- [1] D. Karaboga et al.: A comprehensive survey: artificial bee colony (ABC) algorithm and applications, *Artificial Intelligence Review*, roč. 42, 2014, 21-57.
- [2] D. Karaboga a B. Basturk: On the performance of artificial bee colony (ABC) algorithm, *Applied Soft Computing*, roč. 8, 2008, 687-697.
- [3] D.T. Pham et al.: The Bees Algorithm – A Novel Tool for Complex Optimisation Problems. *Intelligent Production Machines and Systems*, 2nd I*PROMS Virtual Int. Conference, Elsevier, 2006, 454-459.
- [4] R. Akbari, A. Mohammadi a K. Ziarati: A powerful bee swarm optimization algorithm. IEEE 13th Int. Multitopic Conference, Islamabad, 2009, 1-6.
- [5] J. Biroš a M. Mach: Prírodné inšpirované algoritmy rodiny ABC a porovnanie ich konvergencie. Proc. of the *Electrical Engineering and Informatics 10*, Technická univerzita v Košiciach, 2019, 496-500.
- [6] M. Jamil and X.S.Yang: A Literature Survey of Benchmark Functions for Global Optimization Problems, *Int. Journal of Mathematical Modelling and Numerical Optimization*, roč. 4, 2013, č. 2, 150-194.
- [7] J. Biroš: Porovnanie prírodné inšpirovaných optimalizačných metód, Diplomová práca, Fakulta elektrotechniky a informatiky Technická univerzita v Košiciach, 2019, 83 strán.

Návrh a realizácia experimentálneho modelu PON a FTTx sietí

¹Anton ZÁBORSKÝ, ²Luboš OVSENÍK, ³Ján TURÁN, ⁴Tomáš HUSZANÍK

^{1, 2, 3, 4} Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹anton.zaborsky@student.tuke.sk, ²lubos.ovsenik@tuke.sk, ³jan.turan@tuke.sk,
⁴tomas.huszanik@tuke.sk

Abstrakt — Prístupové siete FTTx predstavujú súbor riešení pripojenia koncových zariadení zákazníkov do siete. Toto sa dá doceliť rôznymi prístupmi, kedy sa môžu na základe možností a požiadaviek použiť hybridné siete, kombinujúce optické vlákna s medenými káblami. V súčasnosti však v prístupových sieťach FTTx čoraz viac prevládajú architektúry, ktoré používajú čisto optické riešenia sietí typu PON (Passive Optical Network). Vďaka neprítomnosti aktívnych prvkov nevyžadujú žiadne externé napájanie elektrickou energiou, vďaka čomu sú schopné poskytovať najkvalitnejšie služby a pokrývať tak náročné požiadavky firemných aj súkromných zákazníkov. Tento príspevok sa venuje návrhu a realizácii pasívnych optických sietí PON a FTTx, ktoré pozostávajú z modelu PON TUKE a optického rozvádzača. Postupne je popísaný návrh takejto siete, simulácia v programe OptSim a jej meranie metódou OTDR.

Kľúčové slová — FTTx, OTDR, PON

I. ÚVOD

Použitie optických vlákien v súčasnosti dominuje pri realizáciách dátových sietí vďaka svojim pozitívnym vlastnostiam, ktoré v mnohých kľúčových parametroch prevyšujú zastarané metalické vedenia. Okrem veľkej šírky pásma a spoľahlivosti sú odolné proti elektromagnetickému žiareniu a taktiež veľmi bezpečné vďaka prenosu, pri ktorom je veľmi náročné realizovať odpočúvanie. Všetky tieto výhody, ktoré so sebou optické vlákna prinášajú sa využívali v minulosti najmä v chrbtových sieťach, ale v súčasnosti sú čisto optické riešenia veľmi rozšírené aj v prístupových sieťach v blízkosti koncových zákazníkov.

Už pri samotnom návrhu optickej siete sa vykladá úsilie na minimalizovanie prenosových strát. Preto je vhodné, aby fyzickému návrhu predchádzala simulácia, ktorá je schopná pred zahájením samotnej výstavby predpovedať nežiaduce vplyvy jednotlivých prvkov na optický prenos. Dobře pripravený návrh so simuláciou môže zamedziť vzniku problémov v budúcnosti, ktoré by sa inak objavili až pri fyzickej realizácii.

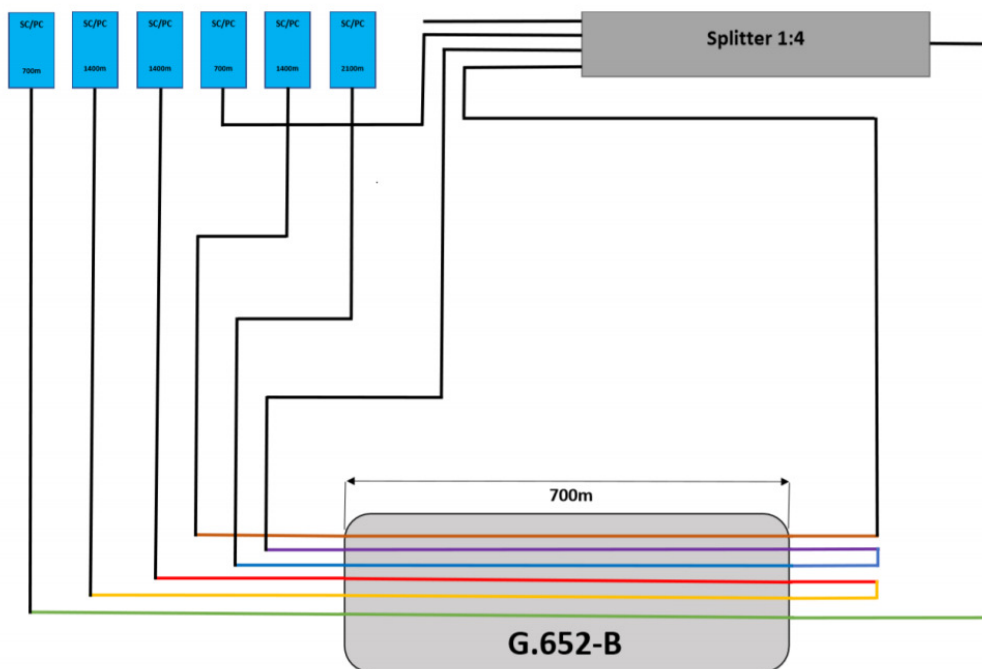
Po výstavbe siete je potrebná pravidelná kontrola prenosových vlastností, či na danej trase nevznikli chyby, vo forme prerušení alebo ohybov. Medzi najpoužívanejšie metódy merania optických trás patrí metóda optickej časovej reflektometrie OTDR (Optical Time Domain Reflectometer), vďaka ktorej dokážeme z jedného konca vlákna odmerať celú trasu na základe analýzy spätne rozptýleného optického žiarenia a následne vykresliť graf udalostí pozdĺž optickej trasy [1-3].

II. NÁVRH EXPERIMENTÁLNEJ SIETE PON A FTTX

Zrealizovaná experimentálna sieť pozostáva z optického rozvádzača a modelu PON TUKE. Oba systémy používajú kábel typu G.652-B s dĺžkami 700 metrov a 200 metrov. Kábel tvorí 6 jednovidových vlákien, ktoré sú chránené v mikrotrubičkách vyplnených géloom s následnou vrstvou spevňujúcich aramidových vlákien a vonkajšieho obalu z odolnej gumy. Kábel dosahuje na vlnovej dĺžke 1550 nm maximálny vložný útlm 0.35 dB. Celý systém experimentálnej siete je typu PON, a teda neobsahuje žiadne prvky s aktívnym napájaním elektrickou energiou. Na prepojenie systémov bol použitý pasívny optický rozbočovač s deliacim pomerom 1:4, ktorý dokáže prepojiť 4 linky optického rozvádzača s modelom PON TUKE [4-6].

A. Optický rozvádzač

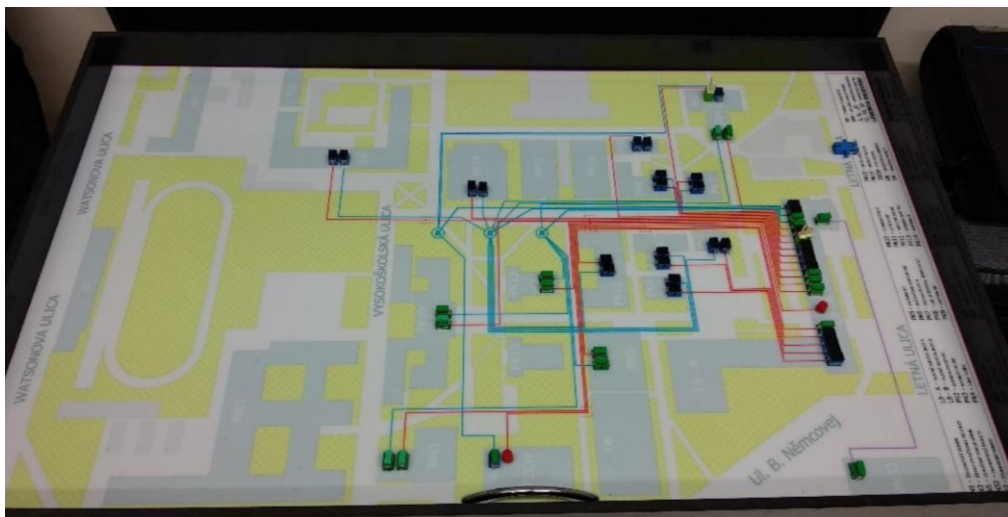
Optický rozvádzač obsahuje spolu 6 liniek, ktoré sú vytvorené pomocou kábla typu G.652-B s dĺžkou 700 m. Jednotlivé linky sú ukončené konektormi typu SC/PC. 2 linky sú priame s dĺžkou 1400m a 4 linky používajú pasívny optický rozbočovač, pričom dosahujú dĺžky 700, 1400 a 2100m. Schéma jednotlivých liniek je zobrazená na Obr. 1 [2].



Obr. 1 Schéma liniek optického rozvádzača

B. Model PON TUKE

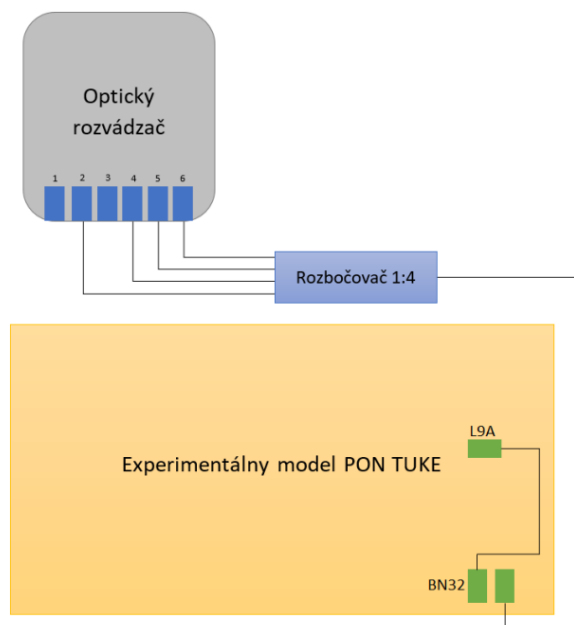
Model PON TUKE simuluje prístupovú sieť s mierkou približne 1:1 v areáli Technickej univerzity v Košiciach. Na vytvorenie požadovaných dĺžok je použitý optický kábel G.652-B s dĺžkou 206 m. Sieť obsahuje 17 priamych liniek a 3 pasívne optické rozbočovače pomocou ktorých je vytvorených 18 liniek [2]. Ukončovacie konektory sú typu SC/PC, SC/APC, E2000 a FC. Model PON TUKE je zobrazený na Obr. 2.



Obr. 2 Model PON TUKE

C. Realizácia experimentálnej siete PON a FTTX

Prepojenie optického rozvádzača a modelu PON TUKE je realizované na trase BN32. Pre prepojenie je použitý pasívny rozbočovač s deliacim pomerom 1:4, ktorý obsahuje 4 konektory SC/PC, ktoré sa pripoja na optický rozvádzač a 1 konektor SC/APC, ktorý sa inštaluje do budovy BN32 modelu PON TUKE. Schéma celého systému experimentálnej siete je zobrazená na Obr. 3.



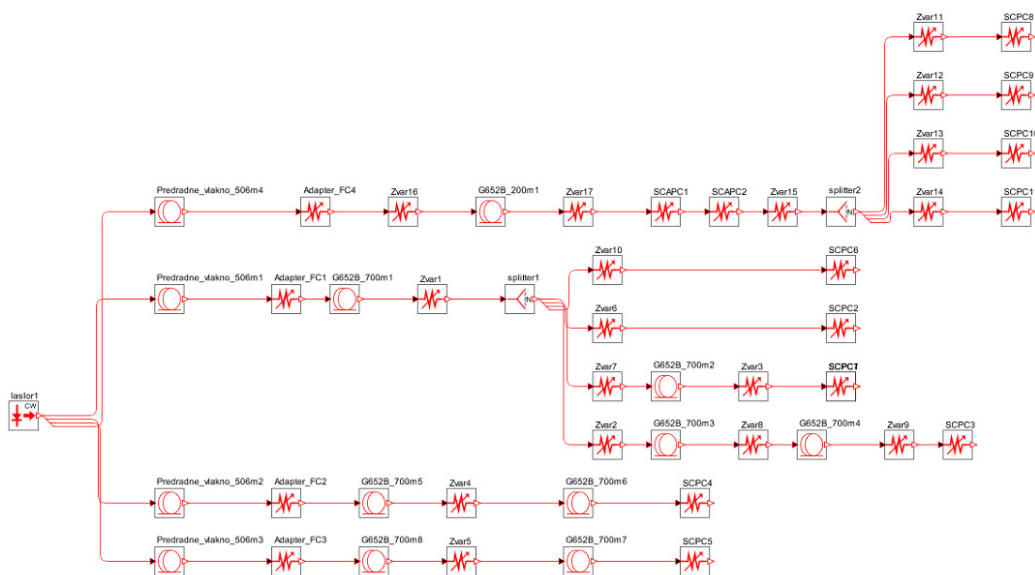
Obr. 3 Schéma systému experimentálnej siete

III. SIMULAČNÝ PROGRAM OPTSIM

Simulácia jednotlivých liniek bola vytvorená v programe OptSim od firmy RSoft, pomocou ktorého je možné navrhnuť ľubovoľnú optickú sieť, ktorú je následne možné odsimulovať. Návrh siete funguje na blokovom princípe, kde každý blok predstavuje určitý komponent siete alebo útlmový prvok, ako napríklad konektor, úsek optického vlákna alebo zvar. Každému bloku sa priradia požadované parametre a ich vzájomným prepojením sa potom vyskladá celá trasa. Program obsahuje knižnicu, v ktorej sú integrované všetky najpoužívanejšie zariadenia a komponenty siete, ktorých parametre je možné dodatočne upravovať. Okrem blokového režimu je možné pracovať v režime vzoriek. Toto je najvýhodnejšie pri kooperácii s Matlabom, keďže je možné každú vzorku prepojiť s jeho zdrojovým kódom a ďalej s ňou pracovať. Ako zdroj optického signálu je v našom prípade použitý Lorentzov laser. Na koniec simulovanej linky je potrebné pripojiť blok optickej sondy, ktorá po simulácii vyhodnotí spektrum a optický výkon danej linky. Pre všetky bloky je na základe informácií od výrobcu potrebné zadať parametre vloženého útlmu [7-9].

A. Blokovaná schéma simulovaných liniek

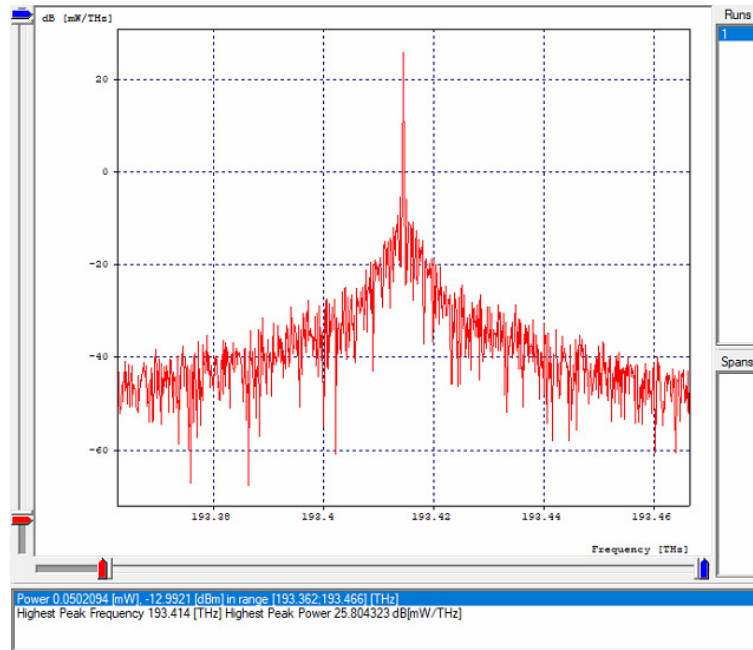
V programe OptSim bola vytvorená schéma experimentálnej siete, ktorá bola následne odsimulovaná na vlnových dĺžkach 1310 nm a 1550 nm. Schéma je zobrazená na Obr. 4.



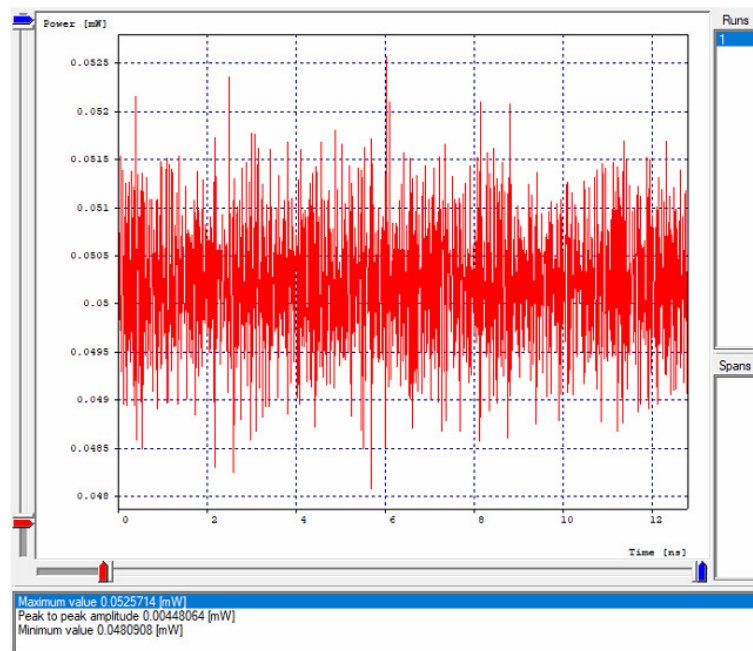
Obr. 4 Schéma experimentálnej siete v programe OptSim

B. Výsledky simulácie

Simulácia optického spektra 2600 m dlhej trasy na optickom rozvádzači s vlnovou dĺžkou 1550 nm je zobrazená na Obr.5. Celkový útlm trasy je 12.9921 dB s výstupným výkonom na úrovni 0.0502094 mW vo frekvenčnej oblasti medzi 193.362 a 193.466 THz. Najvyšší špičkový výkon je na úrovni 25.804323 dB pri frekvencii 193.414 THz. Na Obr.6 je zobrazený optický výkon trasy na vlnovej dĺžke 1550 nm s minimálnou hodnotou na úrovni 0.0480908 mW, maximálnou na úrovni 0.0525714 mW a hodnotou špička – špička 0.00448064 mW.



Obr. 5 Spektrum trasy optického rozvádzača so splitterom pri vlnovej dĺžke 1550 nm



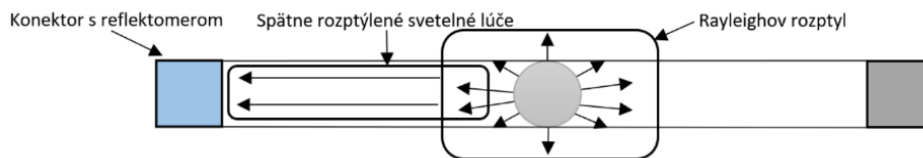
Obr. 6 Optický výkon trasy optického rozvádzača so splitterom pri vlnovej dĺžke 1550 nm

IV. MERANIE METÓDOU OTDR

Metóda optickej časovej reflektometrie OTDR patrí medzi najrozšírenejšie meracie metódy v praxi. Cely princíp je založený na analýze spätne rozptýleného svetla, ktoré sa na základe Rayleighovho rozptylu všesmerovo rozptyľuje na nehomogenitách v materiáli vlákna. Pri prudkých zmenách hustoty v materiáli zasa dochádza k Fresnelovmu odrazu. V praxi sa táto metóda používa v optickom časovom reflektometri, ktorý sa napojí ku konektoru optickej trasy. Meranie sa realizuje vyslaním intenzívnych svetelných impulzov s trvaním niekoľko desiatok až stoviek nanosekúnd, ktorých spätne odrazená časť je následne analyzovaná. Prístroj je potom schopný z rozdielu výkonov signálu na vstupe a výstupe, a časovej variácie spätne rozptýleného svetelného lúča odhadnúť zmenu výkonu a jeho odchýlku. Potom takto dokážeme odmerať celkovú dĺžku trasy, lokalizáciu a útlm jednotlivých udalosti ako konektory zvary alebo ohyby. V konštrukcii reflektometra je zakomponovaná lavínová fotodióda APD (Avalanche Photo Diode), ktorá slúži na detekciu spätne odrazených svetelných lúčov. Signál sa následne spriemeruje v integrátore a dôjde k vyhodnoteniu hodnôt jednotlivých útlmov [1][10].

A. Rayleighov rozptyl

Tento jav vzniká na malých nehomogenitách v optickom vlákne. Takéto malé nehomogenity vznikajú hlavne pri výrobe vlákna, z dôvodu rýchleho tuhnutia po vybratí z pece, kedy vo vlákne ostanú priestory s rozdielnou hustotou. Toto spôsobuje zmeny indexu lomu, následkom čoho sa svetlo všesmerovou rozptyľuje, čím zväčší straty spôsobené vyžiarovaním. Tento jav sa využíva pri meraní vlákien optickou časovou reflektometriou, kde sa analyzuje svetlo vyžiarené v spätnom smere, ktoré tvorí zhruba 4% všesmerovo vyžiareného svetla [1][10]. Svetlo, vyžiarené vplyvom Rayleighovho rozptylu je závislé na veľkosti vlnovej dĺžky, pričom princíp odrazu s využitím spätne rozptýleného svetla je zobrazený na Obr. 7.



Obr. 7 Princíp Rayleighovho odrazu s využitím spätne rozptýleného svetla

B. Fresnelov odraz

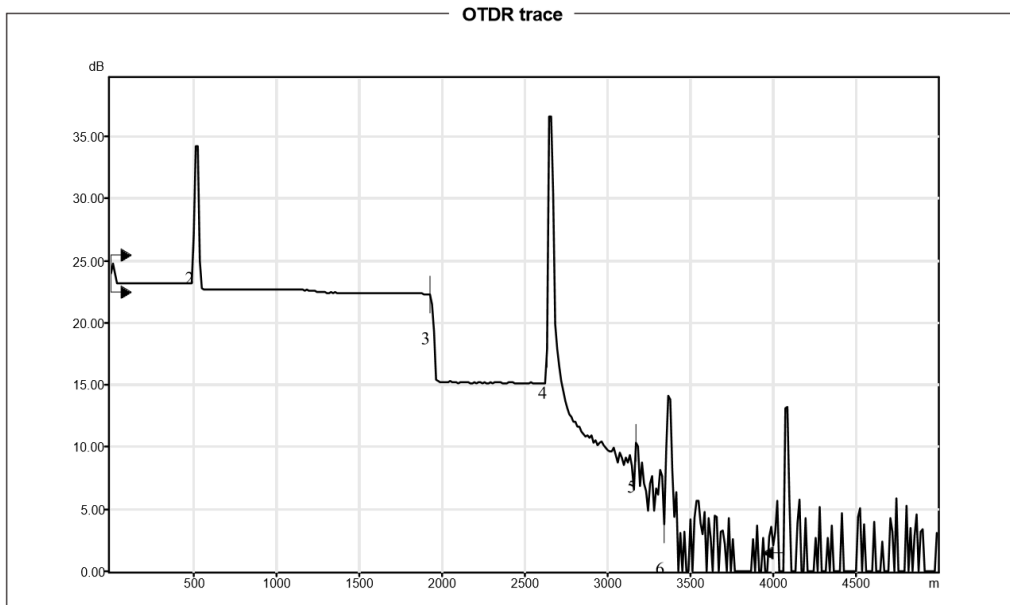
Vzniká pri náhlych, veľkých zmenách hustoty na meranej optickej trase, kedy sa intenzívny svetelný lúč odrazí opačným smerom späť k zdroju žiarenia. Takáto zmena hustôt nastáva pri rôznych makronehomogenitách. V praxi ide najmä o ukončenia vlákna konektormi alebo prerušením vlákna, čo je najčastejšie spôsobené prasknutím zvaru alebo zlomením vlákna. Intenzita svetla v spätne odrazenom lúči závisí od viacerých faktorov, najviac však od veľkosti zmeny indexu lomu a uhlom pri šírení svetla v prostrediach s rozdielnymi hustotami [1][10].

C. Mŕtva zóna

Mŕtva zóna popisuje úsek vlákna na ktorom reflektometer nie je schopný spoľahlivo analyzovať spätne rozptýlene svetlo, kvôli silnému odrazu svetla s vysokou intenzitou. Dôjde teda k oslepeniu reflektometra. Takto silný odraz vznikne na základe Fresnelovho odrazu na makronehomogenite pri vysokých rozdieloch hustoty materiálu. Reflektometer sa dostane do stavu nasýtenia a po dobu jeho regenerácie do schopného stavu nebude možné odhaliť udalosti na úseku vlákna, ktoré zodpovedá tejto dobe regenerácie. Mŕtva zóna vzniká na začiatku každého merania, pričom jej dĺžku ovplyvňujú najmä útlmy nehomogenít a šírka vysielačích impulzov. Okrem toho záleží na kvalite detektora v optickom reflektometri, ktorý sa potom dokáže rýchlejšie zotaviť zo stavu nasýtenia. Pre najefektívnejšie potlačenie mŕtvej zóny na začiatku merania je však najvhodnejšie použiť predradné vlákno, ktorého dĺžka pohltí mŕtvu zónu. Pri meraní dlhších tras bude potrebné merať pri vyšších hodnotách doby trvania impulzov, čo bude mať za následok vznik dlhšej mŕtvej zóny. Preto treba v takomto prípade použiť predradné vlákno s väčšou dĺžkou [1][10].

D. Výsledky merania OTDR

Obr. 8 znázorňuje meranie 2643 m dlhej trasy s pasívnym optickým rozbočovačom na vlnovej dĺžke 1550 nm a Tab.1 obsahuje výsledky tohto merania na vlnovej dĺžke 1550 nm.



Obr. 8 Meranie trasy optického rozvádzača s pasívnym rozbočovačom na vlnovej dĺžke 1550 nm

Tab. 1 Výsledky OTDR merania trasy s pasívnym rozbočovačom na vlnovej dĺžke 1310 nm

Číslo merania	Lokácia [km]	Javy na trase	Strata [dB]	Odraz [dB]	Útlmy [dB/km]	Celkové straty [dB]
1	0.0000	Zahájenie merania	-	-57.2	-	0.000
		Predradné vlákno	0.099		0.196	0.099
2	0.5064		0.412	-35.2		0.511
		Optické vlákno (0.14288 km)	0.499		0.350	1.011
3	1.9353		7.086			8.097
		Optické vlákno (0.7082 km)	0.120		0.170	8.217
4	2.6435		4.551	-14.4		12.768

E. Porovnanie výsledkov simulácie a meranie OTDR

V Tab. 2 sú zobrazené výsledky meraní OTDR a simulácií jednotlivých liniek na vlnových dĺžkach 1310 nm a 1550 nm. Je možné pozorovať rozdiely pri meraní liniek v ktorých je zakomponovaný pasívny optický rozbočovač s vysokým vloženým útlmom 7.4 dB. Tento pasívny prvok spôsobuje najväčší pokles výkonu, preto dosahujú priame trasy menší utlm, ktorý je spôsobený iba zvarmi, drobnými nehomogenitami a ohybmi. Tieto udalosti dosahujú pri správnej realizácii nízke hodnoty útlmu do 1 dB. Taktiež je vidieť rozdiely pri meraní na vlnovej dĺžke 1310 nm, pri ktorej sa najviac prejavujú drobné nehomogenity s malou veľkosťou. Pri vlnovej dĺžke 1550 nm sa zasa viac prejavujú rôzne makroohyby vlákna.

Tab. 1 Výsledné hodnoty meraní OTDR a simulácií v programe OptSim

Konektor	Dĺžka trasy [km]	Výsledky simulácie [dB] 1310 nm	Výsledky simulácie [dB] 1550 nm	Odmerané hodnoty [dB] 1310 nm	Odmerané hodnoty [dB] 1550 nm
K2	1.9316	9.735	1.247	9.387	1.187
K3	1.9319	1.403	1.984	1.393	1.808
K4	1.2218	9.023	12.321	8.699	11.927
K5	1.9324	17.531	9.125	17.356	8.212
K6	2.6436	17.778	12.992	16.468	12.768
BN32	0.7115	0.686	0.655	0.635	0.505

V. ZÁVER

V tomto príspevku dostal priestor popis návrhu a realizácie experimentálnej siete, ktorá používa opticky rozvádzač a model PON TUKE. Oba systémy sú prepojené pomocou pasívneho optického rozbočovača s deliacim pomerom 1:4. Jednotlivé trasy návrhu boli odsimulované v programe OptSim na vlnových dĺžkach 1310 nm a 1550 nm. Príspevok tiež obsahuje meranie OTDR, kde je popísaný princíp metódy OTDR, Rayleighov rozptyl, Fresnelov odraz a Mŕtva zóna. Ďalej sú prezentované výsledky merania OTDR na vlnových dĺžkach 1310 nm a 1550 nm. Výstupom je porovnanie odsimulovaných a nameraných hodnôt, kde môžeme sledovať zmeny. Výsledky simulácií dosahujú väčšie hodnoty útlmov kvôli počítaniu s najhorším možným scenárom, kedy sú hodnoty vložených útlmov nastavené na maximálne možné, ktoré udáva výrobca. V skutočnosti, je útlm jednotlivých trás menší.

POĎAKOVANIE

Táto práca bola podporovaná nasledujúcimi výskumnými projektami: APVV-17-0208 – Odolné mobilné siete na doručovanie obsahu a VEGA 1/0584/20 - UWB senzorové systémy monitorovania osôb pracujúce v reálnych podmienkach.

LITERATÚRA

- [1] R. Ramaswami, K.N. Sivarajan, G.H.Sasaki, "Optical Network", A practical perspective third edition, ISBN: 978-0-12374092-2, 2010, 845 pp.
- [2] T. Ivaniga, P. Ivaniga, L. Ovseník, J. Turán, "Deployment of the PON with an Optical Fibre G-652.B," IEEE 14th International Scientific Conference on Informatics, 2017, pp. 289-292.
- [3] Rsoft Design Group Inc, "Manual of optsim Models Reference Volume 1 Sample Mode" 2009, 543p.
- [4] G.P.Agrawal, "Nonlinear Fiber Optics (Fifth Edition)", The Institute of Optics University of Rochester, New York, 2013, ISBN: 978-0-12397-023-7, 621 pp.
- [5] T. Huszaník, J. Turán, L. Ovseník: "Utilization of 10 Gbps DWDM System with Duobinary Modulation into Passive Optical Network", Journal of Communications Software and Systems, Vol. 14, no. 4 (2018), p. 367-375, ISSN 1845-6421.
- [6] Spolitis, V. Bobrovs, P. Gavars, and G. Ivanovs, "Comparison of passive chromatic dispersion compensation techniques for long reach dense WDM-PON system," Elektron. ir Elektrotechnika, vol. 122, no. 6, pp. 65-70, 2012.
- [7] T. Huszaník, J. Turán, L. Ovseník: "Simulation of Downlink of 10G-PON FTTH in the city of Košice", Carpathian Journal of Electronic and Computer Engineering. Vol. 11, no. 1 (2018), p. 33-39. - ISSN 1844-9689.
- [8] Rsoft Design Group Inc: "Manual of optsim Models Reference Volume 1 Sample Mode" 2009, 543p.
- [9] T. Ivaniga, L. Ovseník: "Experimentálne overenie vplyvu SPM na DWDM systém v programovom prostredí OptSim", Posterus, vol. 8 no. 7, pp. 1-10, 2015.
- [10] J. Turán, L. Ovseník, T. Ivaniga, L. Kuchár: "Experimenty s fyzikálnym modelom PON optických prenosových sietí", Electrical Engineering and Informatics 5, Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice., Košice , pp. 120-125, 2017.

Porovnanie charakteristík optických vláknových zosilňovačov vo vysokokapacitnom DWDM systéme

¹Tomáš HUSZANÍK, ²Luboš OVSEŇÍK, ³Ján TURÁN, ⁴Anton ZÁBORSKÝ,
⁵Norbert ZDRAVECKÝ

^{1, 2, 3, 4, 5} Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹tomas.huszanik@tuke.sk, ²lubos.ovsenik@tuke.sk, ³jan.turan@tuke.sk,
⁴anton.zaborsky@student.tuke.sk, ⁵norbert.zdravecky@student.tuke.sk

Abstrakt — V tomto článku je skúmaný vysokokapacitný 16 kanálový DWDM systém s použitím vláknového zosilňovača dopovaného erbiom (EDFA), Ramanovho optického zosilňovača (ROA) a polovodičového optického zosilňovača (SOA). Každý typ zosilňovača sa analyzoval individuálne a výkon sa porovnával na základe preneseného výkonu. Výkonnosť systému DWDM sa následne vyhodnotila pomocou parametrov BER a Q-faktor pre jednotlivé vlnové dĺžky alokované v optickom pásme C (1530 nm - 1565 nm).

Kľúčové slová — DWDM, EDFA, ROA, SOA.

I. ÚVOD

Ak pri prenose optickým komunikačným systémom STM-64 vytvoríme 100 spektrálnych kanálov pomocou WDM, dostaneme až kaacitu až 1 Tb/s na prenosový kanál. Pri prenose na veľké vzdialenosti - nad 100 km - je potrebné zosilniť signál a teda použiť optické zosilňovače na zosilnenie signálu. V tomto prípade sa používajú zosilňovače, ktoré umožňujú priame zosilnenie optického signálu. Prenos prostredníctvom optických komunikačných systémov patrí medzi najstabilnejšie, ale tieto systémy majú aj obmedzenia, napríklad lineárne a nelineárne javy. Nelineárne javy v optických komunikačných systémoch zohrávajú dôležitú úlohu vo výskume v oblasti vysokorýchlostných a vysokokapacitných optických prenosových systémov. Na jednej strane, akonáhle sa tieto nelineárne javy experimentálne potvrdia, bolo stanovené, že môžu ovplyvniť určité parametre optických vlákien. Spočiatku boli tieto javy irelevantné, ale príchod WDM a nasadenie optických zosilňovačov ich štúdium urýchlilo, pretože došlo k skresleniu signálu prenášaného takýmto systémom. V tomto článku uvádzame porovnávaciu analýzu troch najpoužívanejších optických zosilňovačov. Stručné porovnanie týchto zosilňovačov je v Tab.1.

Tab. 1
Parametre optických zosilňovačov

Parameter	Typ zosilňovača		
	EDFA	Raman	SOA
Zisk [dB]	>40	>25	>30
Vlnová dĺžka [nm]	1530-1560	1280-1650	1280-1650
Šírka pásma [3 dB]	30-60	Závislá na pumpe	60
Max. saturácia [dBm]	22	0.75 x pump	18
Citlivosť na polarizáciu	Nie	Nie	Áno
Výkon pumpy	25 dBm	>30 dBm	<400 mA
Integrácia	Rack	Bulk	Compact
Cenové náklady	Stredné	Vysoké	Nízke

Ramanov rozptyl je nelineárny efekt vlákna používaný na zosilnenie optického signálu v OV. K hromadnému nasadeniu ROA došlo na konci dvadsiateho storočia a dnes sa s ním stretáme takmer na všetkých diaľkových trasách. V zásade je to len laserový zdroj, ktorý je pripojený k optickej ceste a využíva rozptyl na častiach materiálu na zosilnenie optického signálu. Pri použití ROA nie sú hodnoty zisku také vysoké ako hodnoty EDFA. Úroveň signálu sa môže zvýšiť o 15 dB - 20 dB. EDFA a ROA je možné navzájom kombinovať. Hlavnou výhodou je, že ROA môže pracovať v širokom spektre vlnových dĺžok, takže má schopnosť prispôsobiť spektrum zisku pomocou niekoľkých čerpadiel s rôznymi vlnovými dĺžkami. Aktívnym ramanovým médiom je optické vlákno využívajúce kryštál, planárny optický svetelný vodič alebo bunka s plynným alebo tekutým roztokom [7-10].

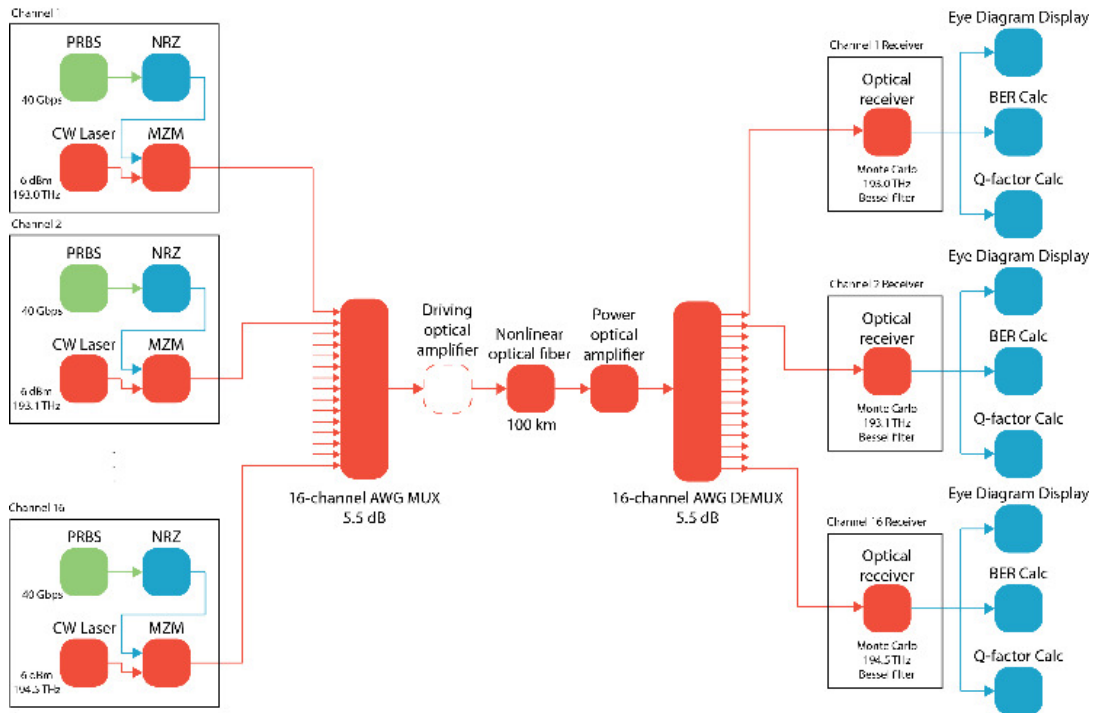
Polovodičový zosilňovač (SOA) sa líši od EDFA a ROA v aktívnej oblasti, v ktorej sa vytvára zisk. V prípade EDFA alebo ROA sa zisk generuje priamo v optickom vlákne, ale v SOA sa zisk generuje priamo v štruktúre vodiča. Princíp dodávky energie sa tiež líši, pretože laserové čerpadlo sa používa pre EDFA a ROA. V SOA je energia napájaná elektrickým poľom. Princíp zosilnenia svetla je založený na rekombinácii elektrónov a dier pri prechode p-n. SOA sa vyrábajú ako čipy, ktoré sú umiestnené v utesenom puzdre schopnom udržať konštantnú teplotu. Na reguláciu teploty je možné nastaviť vhodnú pracovnú vlnovú dĺžku, aby sa dosiahol maximálny zisk. Konštrukcie SOA sú svojim dizajnom podobné polovodičovému laserom s tým rozdielom, že ich lasery obsahujú odrazový povrch rezonátora. Pri konštrukcii SOA je potrebné vybrať materiál s dobrou kvantovou účinnosťou. Kvantová účinnosť sa udáva ako pomer maximálneho počtu generovaných fotónov k počtu excitovaných nosných poplatkov. Vhodné prvky zahŕňajú: arzén (As), gálium (Ga), hliník (Al), indium (In) a fosfor (P). Použitie materiály sú potom zliatiny týchto prvkov, napr. : GaAs, AlGaAs, InGaAs, InGaAsP, InAlGaAs a InP. K saturácii zisku dochádza pri silnom optickom vstupnom signáli v dôsledku vyčerpania voľných nosičov náboja v aktívnej oblasti. Zisk potom klesá so zvyšujúcim sa výkonom. Hodnota saturačného výkonu je daná znížením zisku 3dB z maximálnej hodnoty zisku. Pre ľahšiu integráciu sa SOA používajú ako predzosilňovače [11]. Hoci sa SOA mohli použiť na amplifikáciu viacerých kanálov súčasne, aby našli uplatnenie v WDM, trpia zásadným problémom spojeným s ich rýchlou reakciou. V praxi ovplyvňujú nelineárne javy, ako napríklad FWM (Four Wave Mixing) alebo XPM (Cross Phase Modulation) [12]. Tieto a ďalšie parametre, ako napríklad vysoké straty pripojenia, polarizácia alebo citlivosť, ich robia nepoužiteľnými, keď je pripojený IN-Line. SOA našli svoje uplatnenie ako rýchle prepínače smerovania vo vlne v systémoch WDM. Používajú sa aj ako lacnejšia alternatíva v metropolitných sieťach [12 - 15].

II. SIMULAČNÝ MODEL

Simulačný model 16-kanálového systému DWDM s rôznymi typmi optických zosilňovačov je znázornený na Obr. 1. Celé systémy pozostávajú z troch hlavných častí: vysielačovej časti, optickej distribučnej časti a prijímacej časti. Ilustrácia modelu je farebne označená: zelené čiary predstavujú logický signál; modré čiary predstavujú elektrický signál a červené čiary predstavujú signál v optickej doméne. Vysielače navrhovaného systému DWDM pozostávajú z PRBS (Pseudo Random Binary Source), ktorý generuje logický signál s dátovou rýchlosťou 40 Gbps. Tento logický signál potom prechádza do elektrického impulzného zosilňovača. Druh elektrickej modulácie je NRZ (Non-Return to Zero). Elektrický signál je modulovaný pomocou optického Mach-Zehnderovho modulátora s CW (Continuous Wave) laserom. Frekvencie / vlnové dĺžky laserov sa nastavujú podľa ITU-T G. 694.1. Frekvenčný rozsah je 193 THz až 194,5 THz a rozstup kanálov je 100 GHz. Výkon lasera je 6 dBm. Všetkých 16 kanálov je multiplexovaných pomocou multiplexora AWG (Arrayed Waveguide Grating) so stratou vkladania 5,5 dB (na vstup).

Ďalší blok je budiaci optický zosilňovač. Na tomto mieste sú umiestnené hlavné tri typy optických zosilňovačov - EDFA, SOA, ROA. Amplifikovaný optický signál sa potom vysiela cez 100 km nelineárneho optického vlákna (NOF).

Výkonný optický zosilňovač sa používa na zosilnenie prijímaného signálu, aby sa zabezpečila úspešná detekcia. Prijatý optický signál sa konvertuje na elektrický signál a neskôr sa analyzuje z hľadiska BER (bitová chybovosť) a Q (kvalita).



Obr. 1 16 kanálový DWDM systém

Najdôležitejším parametrom je bitová chybovosť (BER). Ďalším parametrom je Q-faktor. Q-faktor je funkciou OSNR a predstavuje toleranciu systému v dB. BER a Q-faktor sú vo veľmi úzkom vzťahu:

$$BER = \frac{1}{2} \operatorname{erfc} \left(\frac{Q}{\sqrt{2}} \right) \approx \frac{1}{\sqrt{2\pi}Q} \exp \left(-\frac{Q^2}{2} \right), \quad (1)$$

Všeobecný vzťah pre Q-faktor je:

$$Q = \frac{I_1 - I_0}{\sigma_1 + \sigma_0}, \quad (2)$$

kde I_1 a σ_1 sú stredné hodnoty a výstup rozptylu pomocou gaussovského impulzu „1“ a I_0 a σ_0 sú stredné hodnoty a výstup rozptylu pomocou gaussovského impulzu „0“.

Výkon 16 kanálového DWDM sa analyzuje s tromi typmi optických zosilňovačov: EDFA, ROA a SOA.

A. DWDM systém s EDFA

Parametre EDFA použité v simulácii sú v Tab. 2.

Tab. 2
 Parametre EDFA

Parameter	Value
Rádus jadra EDF	2.2 μm
Meta-stabilná životnosť Erbium	10 ms
Numerická apertúra	0.24
Strata pri 1550 nm	0.1 dB/m
Strata pri 980 nm	0.15 dB/m
Dĺžka EDF	10-40 m
Vlnová dĺžka pumpy	980 & 1480 nm

V prvom experimente sa dĺžka vlákna dopovaného erbiom (EDF) zmenila medzi 10 až 40 m. Z experimentálnych výsledkov vidíme, že optická energia sa postupne znižuje so zvyšujúcou sa dĺžkou vlákna dopovaného erbiom. Tento zaujímavý jav je spôsobený dĺžkou vlákien dopovaných erbiom (EDF) vo vnútri EDFA. Presné úrovne energie po optickej distribučnej ceste po 50 km a 100 km sú uvedené v Tab. 3.

Tab. 3
Výkon prijatého signálu po 50 km a 100 km

Pumpa	980 nm	1480 nm	980 nm	1480 nm
EDF [m]	Výkon po 50 km [dBm]		Výkon po 100 km [dBm]	
10	13.386	11.748	6.872	7.182
20	14.958	17.341	7.137	8.560
30	7.935	17.269	7.484	8.921
40	6.743	11.305	6.835	12.645

V ďalšom nastavení sa pumpa EDFA strieda, aby sa pozorovala závislosť amplifikácie od vlnovej dĺžky pumpy. Bežne sa používajú dve vlnové dĺžky pumpy 980 nm a 1480 nm. Výsledky tejto simulácie sú v Tab. 4. Čerpanie je opačné.

Tab. 4
Hodnoty BER 16 kanálového DWDM systému s EDFA

EDF [m]	10	20	30	40
Kanál	980 nm pumpa			
	BER			
1	3.874x 10 ⁻¹³	2.830x 10 ⁻⁶	2.275x 10 ⁻²	2.275x 10 ⁻²
7	2.122x 10 ⁻²⁴	1.800x 10 ⁻¹⁶	2.416x 10 ⁻³	1.689x 10 ⁻³
12	1.132x 10 ⁻²⁶	6.051x 10 ⁻²⁰	2.195x 10 ⁻⁷	2.738x 10 ⁻³
16	6.848x 10 ⁻³³	1.350x 10 ⁻²¹	8.564x 10 ⁻¹³	5.611x 10 ⁻⁹
Kanál	1480 nm pumpa			
	BER			
1	7.014x 10 ⁻¹⁵	4.017x 10 ⁻¹⁰	2.180x 10 ⁻²	2.275x 10 ⁻²

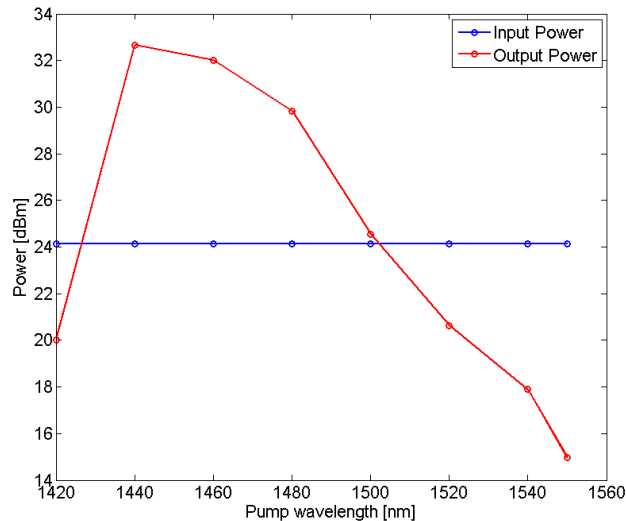
B. DWDM systém s ROA

V tomto scenári je nasadený optický zosilňovač Raman a hodnotí sa výkon 16 kanálového DWDM. Vlastnosti Ramanovho optického zosilňovača sú uvedené v Tab. 5. Poloha zdrojového čerpadla hrá dôležitú úlohu v Ramanovom zosilňovači. Poloha zdrojového čerpadla pri 1440 nm vytvára najväčší rozdiel medzi výkonom vstupného výkonu pre vlákno a výstupom energie z neho. Uvedený rozdiel medzi vstupným a výstupným výkonom je 14 dBm v polohe zdrojového čerpadla 1440 nm. Zisk výkonu sa súčasne znižuje s vlnovou dĺžkou presahujúcou 1440 nm.

Tab. 5
Parametre ROA

Parameter	Hodnota
Dĺžka segmentu	20/30/40/50/60/70/80/90 /100 km
Útlm	0.2 dB/km
Vrchol Ramanovho zisku	1e-013
Vlnová dĺžka pumpy	1440 nm
Teplota	300 K
Polarizačný faktor	2
Rayleighov spätný rozptyl	5e-005 1/km

Závislosť medzi vlnovou dĺžkou pumpy a výkonom ROA je na Obr. 2.

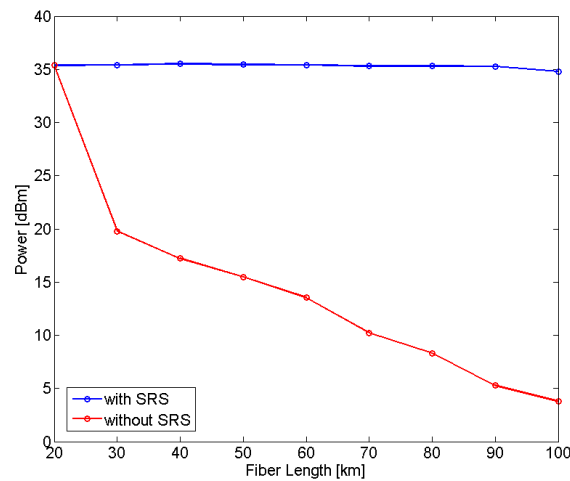


Obr.2 ROA – vlnová dĺžka pumpy vs výkon

BER je medzi 1×10^{-11} a 1×10^{-25} . Najväčšia chyba je na prvom kanáli. K vyšším frekvenciám sa miera chybovosti znižuje, čo je spôsobené vďaka SRS, pretože kanály na vyšších frekvenciách majú vyšší výkon ako kanály na nižších frekvenciách.

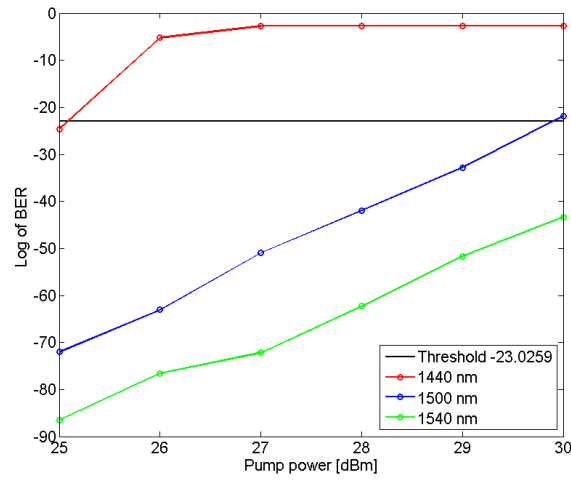
Ďalšia simulácia je zameraná na získanie vzťahu medzi výstupným výkonom a dĺžkou optického vlákna s a bez uvažovania nelineárneho efektu nazývaného SRS (Stimulated Raman Scattering). Vstupný výkon bol nastavený na konštantnú hodnotu (500 mW pri 1440 nm).

Bez SRS sa výstupný výkon optického signálu po prekročení 30 km optického vlákna rýchlo znižuje, ako vidno na Obr. 3. Pri uvažovaní SRS zostáva úroveň výkonu nad 35 dBm (s veľmi miernym poklesom) do 100 km optického vlákna vlákno.



Obr.3 ROA – dĺžka vlákna vs výkon

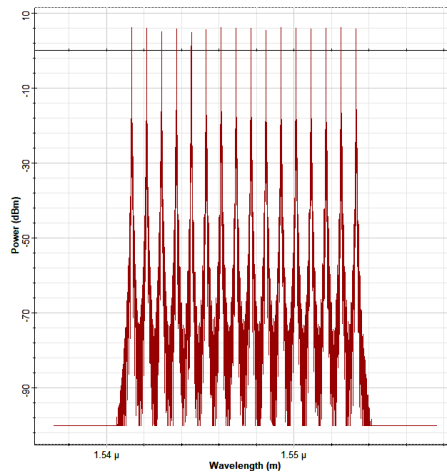
Nakoniec sa skúma zvyšujúci sa výkon pumpy. Vyberáme tri najčastejšie používané vlnové dĺžky pumpy - 1440 nm, 1500 nm a 1540 nm. Rozsah výkonu čerpadla je 25 - 30 dBm. Táto závislosť je znázornená na Obr. 4



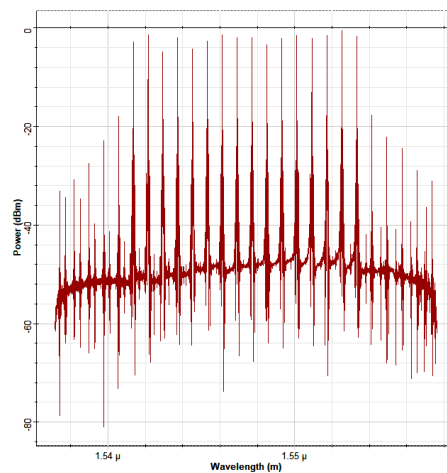
Obr.4 ROA – výkon pumpy vs BER

C. DWDM systém s SOA

Parametre SOA použité v simulačnom modeli sú v Tab. 6. Vstupné a výstupné optické spektrá po 100 km optického vlákna sú znázornené na Obr. 5 a 6.



Obr. 5 Spektrum 16 kanálového DWDM systému na vstupe



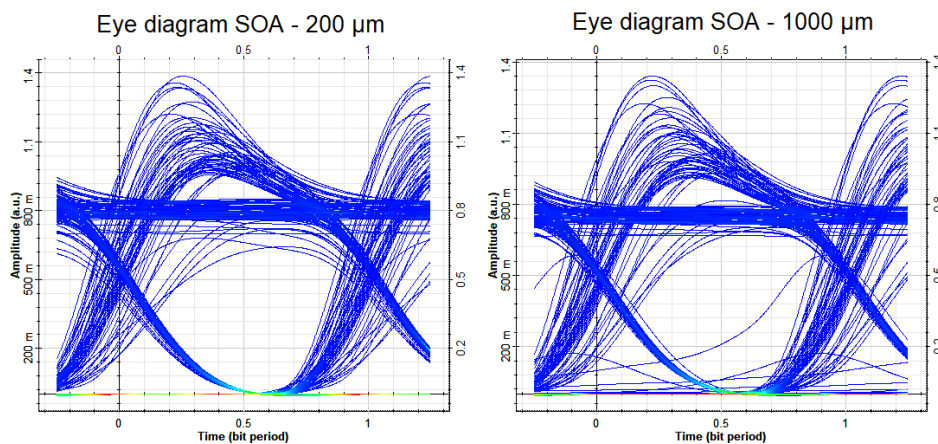
Obr. 8 Spektrum 16 kanálového DWDM systému so SOA

Tab. 6
 Parametre SOA

Parameter	Hodnota
Bias	100 mA

Dĺžka SOA	200/500/1000 μm
Šírka aktívnej vrstvy	3 μm
Hrúbka aktívnej vrstvy	0.8 μm
Spontánna životnosť nosnej	0.3 ns
Vstupné straty	3 dB
Saturácia	40 mW
Napájanie	0.15 A

Obr. 7 zobrazuje prijaté diagramy oka pre dve rôzne dĺžky SOA pri frekvencii 193,5 THz. Vypočítaná BER pre kanál 193,5 THz, berúc do úvahy 200 μm SOA, je $4,98 \times 10^{-25}$, pre 500 μm SOA je $8,25 \times 10^{-23}$ a $1,32 \times 10^{-20}$ pre 1000 μm dlhá SOA.



Obr.7 SOA – prijaté diagramy oka

III. ZÁVER

Skúmali sme vplyv EDFA, ROA a SOA na kvalitu prenosu v 16-kanálovom DWDM systéme s prenosovou rýchlosťou 40 Gbps na kanál. Vyhodnocujú sa rôzne scenáre a parametre pre každý typ optického zosilňovača. Výber správneho typu optického zosilňovača môže mať zásadný vplyv na výkon celého systému DWDM.

POĎAKOVANIE

Táto práca bola podporovaná nasledujúcimi výskumnými projektami: APVV-17-0208 – Odolné mobilné siete na doručovanie obsahu a VEGA 1/0584/20 - UWB senzorové systémy monitorovania osôb pracujúce v reálnych podmienkach.

LITERATÚRA

- [1] P. Cho, Y. Achiam, G. Levy-Yurista, M. Margalit, Y. Gross, and J. Khurgin, "Investigation of SOA Nonlinearities on the Amplification of DWDM Channels with Spectral Efficiency Up to 2.5 b/s/Hz," *Photonics Technology Letters, IEEE*, vol. 16, no. 4, 2004. doi: 16. 918 - 920. 10.1109/LPT.2004.823687.
- [2] A. Rani, M. Singh, "Impact of Different Modulation Data Formats on DWDM System Using SOA with Narrow-Channel Spacing," *Journal of Optical Communications*, vol. 40, no. 4, p. 1-5, 2017. doi: 10.1515/joc-2017-0093.
- [3] J. Ruzbarsky, J. Turan, and L. Ovsenik, "Stimulated Brillouin scattering in DWDM all optical communication systems," in *Proceedings of the in 26th International Conference Radioelektronika (RADIOELEKTRONIKA '16)*, pp. 395–398, April 2016.
- [4] ITU-T, "Spectral grids for WDM applications: DWDM frequency grid," Recommendation G.694.1, 2012.
- [5] T. Huszanik, J. Turán, and L. Ovsenik, "Mitigation of Fiber Nonlinear Effects in 1.28 Tbps DQPSK Modulated DWDM System," *Electronics ETF*, vol. 23, no. 1, pp. 1-8, 2019. doi: 23. 3. 10.7251/ELS1923003H.
- [6] T. Huszanik, J. Turán and L. Ovsenik, "On Mitigation of Four-Wave Mixing in High Capacity Ultra-DWDM System," *2019 20th International Carpathian Control Conference (ICCC)*, Krakow-Wieliczka, Poland, 2019, pp. 1-4. doi: 10.1109/CarpathianCC.2019.8766021.
- [7] T. Ivaniga, J. Turán and L. Ovsenik, "Verification of the SPM Impact in DWDM System Using AWG Multiplexer/Demultiplexer," *Electrotechnica et Informatica*, vol. 17, no. 1, pp.17–22, 2017. doi:10.15546/aeei-2017-0003.
- [8] J. M. SIMMONS, "Optical Network Design and Planning," 529 p., 2017. ISSN: 1935-3847, doi: 10.1007/978-3-319-05227-4.

- [9] L. G. Kazovsky, W. T. Shaw, D. Gutierrez, N. Cheng, S. W. Wong, "Next-Generation Optical Access Networks," *Journal of Lightwave Technology*, vol. 25, no. 11, 2017. doi: 10.1109/JLT.2007.907748.
- [10] M. Wasfi, "Optical Fiber Amplifiers-Review," *International Journal of Communication Networks and Information Security*, 2009.
- [11] M. ul zaman, S. Iqbal and Isha, "Performance analysis of EDFA amplifier for DWDM system," 2014 International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, 2014, pp. 840-843. doi: 10.1109/IndiaCom.2014.6828081.
- [12] S. Semmalar, Poonkuzhali and P. Devi, "Optimized gain EDFA of different lengths with an influence of pump power," 2011 International Conference on Electronics, Communication and Computing Technologies, Pauls Nagar, 2011, pp. 90-95. doi: 10.1109/ICECCT.2011.6077076.
- [13] M. M. J. Martini, C. E. S. Castellani, M. J. Pontes, M. R. N. Ribeiro and H. J. Kalinowski, "Performance comparison for Raman+EDFA and EDFA+Raman hybrid amplifiers using recycled multiple pump lasers for WDM systems," 2015 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC), Porto de Galinhas, 2015, pp. 1-5. doi: 10.1109/IMOC.2015.7369103.
- [14] Z. Li, G. Li, "Linear and nonlinear crosstalk suppression for DWDM RZ-DPSK transmitter using a saturated SOA as power booster amplifier," in *IEEE Photonics Technology Letters*, vol. 18, no. 7, pp. 811-813, April 1, 2006. doi: 10.1109/LPT.2006.871843.
- [15] N. Thakral and L. Kumar, "Performance evaluation of different hybrid optical amplifiers for DWDM system," 2016 IEEE Annual India Conference (INDICON), Bangalore, 2016, pp. 1-4. doi: 10.1109/INDICON.2016.7838905.

Spracovanie a prenos videotokov z videosenzorov z dronov

¹Samuel Grobár, ²Stanislav Marchevský, ³Peter Šulaj, ⁴Renát Haluška

Katedra elektroniky a multimediálnych telekomunikácií
Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach
Park Komenského 13, 042 00 Košice, Slovenská republika

¹samuel.grobar@student.tuke.sk, ²stanislav.marchevsky, ³peter.sulaj@tuke.sk,
⁴renat.haluska@tuke.sk

Abstract — Úlohou tohto článku je priblížiť sa k riešeniu problému spracovania viacnásobného (panoramatického) videa pomocou bezpilotných lietadiel. Je implementovaný proces spojenia dvoch videí a proces detekcie a klasifikácie objektov pomocou programovacieho jazyka Python, knižníc OpenCV a TensorFlow. Celý proces je implementovaný na minipočítači Raspberry Pi. Vstupné videá sú prijímané z dvoch kamier, modulu kamery Raspberry a webovej kamery USB. Celý proces je rozdelený na dve časti, a to šitie videa, po ktorom nasleduje detekcia a klasifikácia objektov v obraze.

Keywords — videocoding, panorama, videostitching, object detection, OpenCV, Raspberry

I. ÚVOD

Problematika spracovania a prenosu viacpohľadového videa v reálnom čase je stále aktuálnym trendom pri vývoji 3D aplikácií. Hlavnou úlohou je vytvoriť vysokokvalitný panoramatický obraz z niekoľkých kamier, ktoré sú zamerané na tú istú scénu. O to väčšou motiváciou a výzvou je využívať doterajšie poznatky pre realizáciu takéhoto panoramatického videa za pomoci bezpilotných prostriedkov. Cieľom tejto práce je navrhnúť a implementovať vhodné metódy pre realizáciu takéhoto panoramatického videa v reálnom čase spolu s detekciou a klasifikáciou objektov.

II. DRONY A VIDEOSTITCHING

Dron je bezpilotné lietadlo, niekedy nazývané aj UAV (z angl. Unmanned Aerial Vehicle) alebo aj dron (z angl. drone) je lietadlo bez posádky, ktoré môže byť riadené na diaľku, alebo lietať samostatne pomocou predprogramovaných letových plánov alebo pomocou zložitejších dynamických autonómnych systémov. Bepilotné lietadlá sa používajú často v armáde k prieskumom ale aj k útočným letom. Používajú sa taktiež v mnohých civilných úkonoch, napr. k haseniu požiarov, policajným sledovaniam alebo prieskumu terénu. Prvé bezpilotné lietadlo bolo z roku 1913 od profesora Archibalda Montgomeryho Lowa a nazývalo sa „Aerial target“ (vzdušný cieľ). Nasledovalo mnoho ďalších lietadiel riadených na diaľku, vrátane Hewitt-Sperry Automatic Airplane, ktorý bol vyrobený počas prvej svetovej vojny v USA. Bepilotné lietadlá sa využívali a dodnes sa využívajú aj pri fotografovaní alebo nahrávaní videa zo vzduchu. S miniaturizáciou všetkých technológií v 80. a 90. rokoch 20. storočia záujem o bezpilotné lietadlá stále rástol. Prvá generácia bola najmä určená k prieskumným letom, ale niektoré z nich boli vybavené zbraňami, ako napr. MQ-1 Predator, ktorý obsahoval strely vzduch-zem. Bepilotné lietadlá rozdeľujeme do nasledujúcich funkčných kategórií (aj keď začínajú postupne prevažovať multifunkčné lietadlá):

- **Ciele a návnady**- poskytujú pozemné a vzdušné ciele stimulujúce nepriateľské lietadlá
- **Bojové**- schopné útoku
- **Logistické**- UAV špeciálne navrhnuté pre logistické účely
- **Výskumné a vývojové**- používané k ďalšiemu vývoju technológie UAV
- **Civilné a komerčné**- UAV špeciálne navrhnuté pre civilné alebo komerčné aplikácie

III. PROBLEMATIKA VYUŽITIA DRONOV PRI VIDEOSTITCHINGU

Šitiu obrazu je v literatúre venovaná veľká pozornosť. Väčšina dostupných algoritmov šitia je zameraná na zlepšenie kvality bez ohľadu na dobu trvania šitia. V poslednej dobe sa algoritmy prešívania obrazov použili pri postupnom spájaní obrazov na báze UAV. [1] Princípom je, že UAV v podstate letí ponad cieľovú oblasť pozdĺž vopred naplánovanej trasy a nepretržite prijíma snímka obraz, ktorý sa prenáša na pozemnú stanicu a postupne sa dané obrazy prešívajú, aby mohol byť vytvorený panoramatický obraz cieľovej oblasti. Daniel Wischounig-Strucl vylepšil spôsob prenášania dát, najprv sa vysielajú metadáta a obrazy s nízkym rozlíšením, obrazy s vysokým rozlíšením sa prenášajú až v neskoršej fáze po pristátí. [2] Existujú tiež komerčné systémy pre šitie obrazu založené na UAV. Spoločnosť NAVSYS uvádza, že jej systém dokáže dosiahnuť rýchlosť šitia 1 obrazu za sekundu. [3] Hlavným cieľom je vytvoriť cieľový obraz $I_{res,n}$. Cieľová oblasť dostane skupinu n jednotlivých snímok $\{I_i\}$. Celkový obraz je možné iteratívne vytvoriť takto:

$$I_{res,0} = O, \quad I_{res,i} = Merge(I_{res,i-1}, T_i(I_i)) \quad (1)$$

Kde O je matica prázdneho pozadia, T je transformačná funkcia a funkcia $Merge$ kombinuje transformovaný obraz do výsledného obrazu. Toto tzv. „mozaikovanie“ možno opísať ako problém s optimalizáciou, pri ktorom musíme nájsť T_i tak, aby maximalizoval funkciu kvality $\lambda(I_{res,i})$. Táto funkcia na základe použitého systému vyvažuje vizuálny vzhľad a presnosť georeferencie. Zatiaľ čo v niektorých aplikáciách je dôležitejšie mať vizuálne na pohľad atraktívnejší obraz, iné aplikácie môžu vyžadovať presné georeferencie v prehľadovom obraze. Najčastejšie sa používa funkcia kvality, ktorá je kombináciou korelácie medzi prekrývajúcimi sa obrazmi a relatívnymi vzdialenosťami v generovanom obraze v porovnaní s reálnou pozemnou vzdialenosťou. [4][5]V nasledujúcom texte sú zhrnuté najdôležitejšie úlohy, ktoré je potrebné vyriešiť ak chceme snímať obraz pomocou UAV malej mierky:

A. Nízka nadmorská výška

Pri získavaní obrazu z nízkej nadmorskej výšky už nie je predpoklad rovinatej plochy pravdivý. Predmety, ako sú budovy, stromy alebo autá, spôsobujú v obrazoch vysoké skreslenie perspektívy. Bez spoločnej základnej roviny si porovnávanie prekrývajúcich sa obrazov vyžaduje informácie o hĺbke. Transformácie obrazu využívajúce korešpondenciu bodov v rôznych výškach môžu viesť k závažným chybám pri porovnávaní jednotlivých obrazov.

B. Vplyv vetra

Vďaka svojej nízkej hmotnosti sú bezpilotné lietadlá náchylné na vplyvy vetra, ktoré si vyžadujú vysokú dynamiku riadenia, aby sa dosiahlo stabilné letové správanie. Dokonca aj vtedy, keď je poloha kamery aktívne kompenzovaná, nie je možné zabezpečiť dokonalý pohľad.

C. Nepresné informácie o polohe a orientácii

Pomocné senzory bezpilotných lietadiel ako sú GPS, IMU a výškomer sa používajú na určenie jeho polohy a orientácie. Takéto pomocné senzory pri malých bezpilotných lietadlách majú iba obmedzenú presnosť, ktorá nie je porovnateľná s väčšími typmi lietadiel. V dôsledku toho sa nie je možné stopercentne spoliehať na presné a spoľahlivé údaje o polohe, orientácii a nadmorskej výške bezpilotného lietadla. Preto sa táto nepresnosť musí brať do úvahy pri procese mozaikovania resp. spájania.

D. Obmedzenia zdrojov

V tejto časti ide hlavne o zdroje ako sú napríklad výpočtový výkon a pamäť, ktoré nám dokáže pozemná stanica resp. bezpilotné lietadlo poskytnúť. V prípade využitia pri veľkých katastrofách zvyčajne nie je možné mať k dispozícii obrovskú výpočtovú infraštruktúru. Základná stanica sa zvyčajne skladá z prenosných počítačov a štandardných stolových počítačov. To znamená, že rýchlosť spracovania údajov vo veľkej miere bude závisieť aj od hardvéru, ktorým budú dáta spracovávané.

E. Spresnenie výsledkov

Jednotlivé obrazy sa snímajú z viacerých bezpilotných lietadiel v ľubovoľnom poradí. Je potrebný postupný prístup, aby sa používateľovi poskytli dostupné údaje čo najskôr. Čím viac vstupných obrazov, tým je presnejší výsledný obraz. To tiež znamená že, každý pohľad môže vyžadovať úpravu polohy už spracovaných obrazov, tak aby sa zlepšila celková kvalita výsledného panoramatického obrazu.

IV. SPÁJANIE OBRAZOV

Ako bolo uvedené v predošlej časti je potrebné nájsť vhodné transformácie T_i pre každý obraz I_i aby sa vyriešil problém pri spájaní. Na výpočet týchto transformácií existujú dva základné postupy:

Metadátovej prístup využíva informácie pomocného senzora na zistenie polohy a orientácie kamery, ktorá sa potom využíva na výpočet transformácií. V takomto prípade sa predpokladá, že každý zachytený obraz má poskytnuté údaje z pomocného senzora (GPS, IMU, výška a čas). Prístup založený iba na vstupnom obraze sa využíva len na výpočet transformácií. V tejto časti budú uvedené základné typy prístupov, neskôr bude opísaný hybridný prístup, ktorý zlepšuje zarovnanie na základe metadát s porovnávaním na základe vstupných obrazov. Prezentované prístupy sa líšia v požiadavkách na zdroje a v dosiahnutých výsledkoch. [6]

A. Zarovnanie podľa polohy

Jedným z najzákladnejších a najjednoduchších spôsobov je zarovnanie obrazov na základe polohy kamery. Na zarovnanie obrazu sa koordináty kamery mapujú na zodpovedajúce súradnice pixelov v generovanom výslednom obraze. Definícia pôvodu obrazu pozorovanej cieľovej oblasti je uvedená ako $o_{world} = (lat, lon, alt)^T$ vo svetových súradniciach sú všetky súradnice obrazu prepojené v tangenciálnej rovine (LTP) aproximáciou k zemskému modelu WGS84. Vzhľadom na polohu kamery sa vypočíta oblasť, na ktorú sa obraz vzťahuje vo svetových súradniciach pričom sa zohľadňujú aj vnútorné parametre kamery. Relatívne svetové súradnice priamo súvisia s pixelovými súradnicami v generovanom obraze. V takomto prípade sa predpokladajú primerane presné informácie o polohe.

B. Zarovnanie podľa polohy a orientácie

Pokročilejšie riešenie spočíva v rozšírení nastavenia polohy vyrovnaním odchýlky orientácie kamery (t.j. naklonenia, stúpania, uhlov natočenia). Funkcia umiestnenia jednotlivých obrazov na vytvorenie výsledného obrazu je rovnaká ako v rovnici (1). No využíva perspektívnu transformáciu T s ôsmimi stupňami tzv. „voľnosti“. Ak je predpokladaný najvyšší možný bod vzletu, tak sa transformácia T redukuje len na transformáciu podobnosti.

C. Zarovnanie podľa obrazu

Zarovnanie podľa obrazu možno rozdeliť do pixelových metód a metód založených na vlastnostiach. Cieľom je nájsť transformáciu T_i a následne polohu každého nového obrazu, ktorý maximalizuje funkciu kvality:

$$\lambda[Merge(I_{res,i-1}, T_i(I_i))] \quad (2)$$

Prístupy založené na pixeloch sú výpočtovo náročnejšie, pretože funkcia kvality sa počíta zo všetkých pixelov v prekrývajúcich sa častiach dvoch obrazov. Prístupy založené na vlastnostiach sa snažia znížiť výpočtovú náročnosť tak, že najprv extrahujú charakteristické body a potom sa hľadá zhoda s bodmi v prekrývajúcich sa častiach. V závislosti od zvoleného stupňa voľnosti sa výsledná transformácia využíva buď transformáciu podobnosti alebo transformáciu perspektívy. Výhodou tohto prístupu je to, že vygenerovaný výsledný obraz je vizuálne detailnejší. Naopak hlavnou nevýhodou je že vyhľadávací priestor rastie s počtom obrazov, ktoré sa majú spájať a jednotlivé obrazy sa môžu zdeformovať.

D. Hybridný prístup

Hybridný prístup je kombináciou predtým spomenutých prístupov. Jeho cieľom je najskôr umiestniť nové obrazy na základe informácií o polohe a orientácii kamery na už vygenerovaný obraz. V ďalšom kroku sa použijú obrazové metódy na opravu nepresných informácií o polohe a orientácii a zároveň na zlepšenie vizuálneho vzhľadu. Keďže už poznáme približnú polohu jednotlivých obrazov z polohy kamery, môžeme výrazne znížiť priestor na vyhľadávanie. Tak je transformácia T_i z rovnice (4.2) rozdelená na dve transformácie, zatiaľ čo $T_{i,pos}$ predstavuje transformáciu založenú na polohe, $T_{i,img}$ je transformáciou založenou na obraze, ktorá zároveň optimalizuje zarovnanie obrazu. Tieto dve transformácie maximalizujú funkciu kvality:

$$\lambda\left(Merge\left(I_{res,i-1}, T_{i,img} \circ T_{i,pos}(I_i)\right)\right), \quad (3)$$
$$pos \in \{(u, v) | u \in [x_{min}, x_{max}], v \in [y_{min}, y_{max}]\}$$

Obmedzuje vyhľadávací priestor na zníženú množinu možných pozícií na základe očakávanej nepresnosti informácií o polohe a orientácii. Takýmto prístupom je možné vygenerovať výstupný obraz bez výrazných perspektívnych skreslení a zároveň si zachovať

relatívne vzdialenosti a georeferencie vo vygenerovanom obraze. Okrem tohto sa týmto spôsobom zamedzí nerovnostiam pri spájaní obrazov.[6]

V. HÁRDVÉROVÉ A SOFTVÉROVÉ PROSTRIEDKY

Táto časť popisuje hardvérové a softvérové riešenie projektu. V jednotlivých častiach je stručne opísaný použitý hardvér a softvér.

A. Použitý hardvér

Čo sa týka hardvérovej časti táto práca je realizovaná pomocou minipočítača Raspberry Pi ku ktorému boli pripojené dve kamery a to kamerový modul od spoločnosti Raspberry a USB kamera Microsoft LifeCAM-HD5000.

B. Použitý softvér

1) OS Raspbian

V rámci softvérovej časti bol zvolený operačný systém Raspbian konkrétne jeho verzia Buster. Patrí medzi distribúcie Linuxu, ktoré sú založené na Debiane, je oficiálne poskytovaný nadáciou Raspberry Pi Foundation, primárne je určený pre Raspberry Pi ale je možné ho použiť aj na osobných počítačoch.

2) Programovací jazyk Python

Pre realizáciu hlavnej časti projektu bol vybraný programovací jazyk Python. Programovací jazyk Python je objektovo orientovaný, interpretovaný, dynamický a multiplatformový softvér s jednoduchou a čitateľnou syntaxou, zameraný na jednoduché pochopenie, použitie a na vysokú produktivitu programovania.

3) Knižnice OpenCV a TensorFlow

Knižnice OpenCV sú najlepšou voľbou pre prácu s prenosom obrazu. Je to open source sada knižníc pre prácu s počítačovým videním v reálnom čase. Je možné s ňou vyvíjať aplikácie pre rozpoznávanie gest, tvári, objektov a pod. Je teda hlavne využívaná v oblasti robotiky a pre implementáciu aplikácií z oblasti experimentálnych rozhraní pre styk človeka s počítačom. TensorFlow je open-source softvérová knižnica pre strojové učenie, V tejto práci je použitá verzia Lite, ktorá primárne určená pre mobilné zariadenia. Využíva databázu pre detekciu objektov, ktorá umožňuje detekciu osemdesiatich objektov.

VI. REALIZÁCIA PRÁCE

V tejto kapitole rozoberieme samotnú realizáciu práce, inštaláciu operačného systému a jednotlivých knižníc a výstupy z navrhnutého algoritmu.

A. Inštalácia OS Raspbian

Ako prvé bolo potrebné nainštalovať operačný systém vhodný pre naše Raspberry. Zvolili sme operačný systém Raspbian. Pre úspešnú inštaláciu budeme potrebovať SD kartu s veľkosťou minimálne 4GB. Prvým krokom bolo naformátovanie SD karty, na ktoré bol použitý program SD Card Formatter.

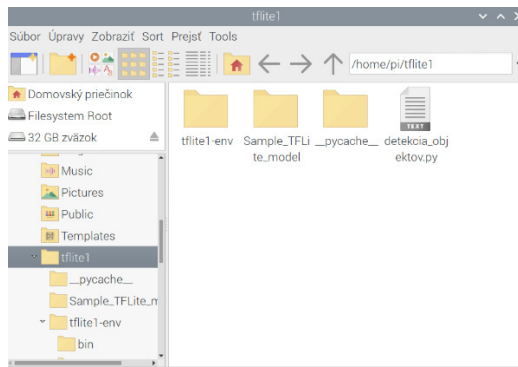
Po sformátovaní SD karty bolo potrebné stiahnuť obraz distribúcie Raspbianu, ktorý je dostupný zdarma na oficiálnej stránke poskytovateľa. V našom prípade sa jednalo konkrétne o verziu Raspbian Buster.

Následne po stiahnutí distribúcie je potrebné tento obraz nahráť na sformátovanú SD kartu. Nato bol použitý program Etcher. Bolo potrebné zvoliť stiahnutý obraz distribúcie a vybrať médium na, ktoré médium sa má obraz nahráť následne po stlačení voľby „Flash“ sa obraz nahral na SD kartu.

B. Štruktúra projektu

Táto časť popisuje štruktúru projektu a jednotlivé výstupy navrhnutého algoritmu, ktorý sa skladá z týchto častí:

- tvorba panoramatického obrazu
- detekcia objektov
- klasifikácia objektov

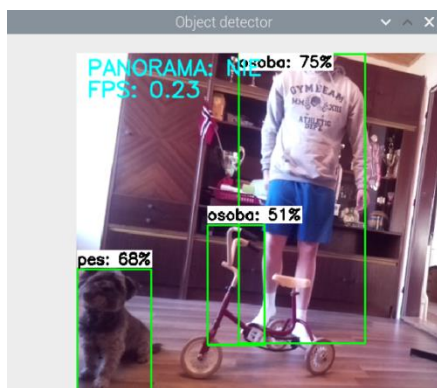


Obr. 1 Štruktúra projektu

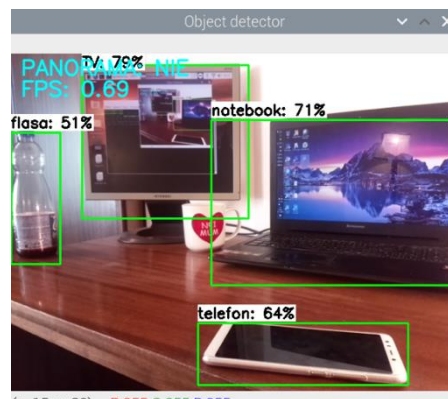
V zložke *tflite1* sa nachádza niekoľko súborov potrebných pre správne fungovanie projektu. V podadresári *tflite1-env* sa nachádzajú všetky využívané knižnice jazyka Python. V tomto podpriechodku je umiestnené virtuálne prostredie jazyka Python, ktorého inštalácia bola popísaná v predchádzajúcej podkapitole. Podpriechodok *Sample_TFLite_model* obsahuje databázu pre klasifikáciu identifikovaných objektov. Do zložky *_pycache_*, ktorá je automaticky vygenerovaná po spustení projektu sa ukladá skompilovaný kód vo forme súborov s príponami *.py*, *.pyc*, *.pyo*. Súbor *detekcia_objektov.py* hlavný skript spustiteľný v jazyku Python.

C. Výsledky

V tejto časti sú na nasledujúcich obrázkoch zobrazené výstupy z jednotlivých experimentov. V dvoch prípadoch sa jedná o výstup, v ktorom sa nepodarilo nájsť zhodu v obraze medzi dvoma kamerami a v dvoch prípadoch sa jedná o vytvorený panoramatický obraz. Proces detekcie a klasifikácie prebieha takým spôsobom že na interpretáciu týchto výsledkov môžeme vidieť tzv. „skóre“ každého s detegovaných objektov, ktoré sa pohybuje v rozmedzí od 0-100% a naznačuje pravdepodobnosť, že bol daný objekt správne detegovaný. V tomto prípade je určená hranica, ktorá nezobrazuje výsledky detekcie pod skóre 50%. Pre každý zistený objekt sa vráti model, ktorý je reprezentovaný štvoricou čísel predstavujúci ohraničujúci obdĺžnik, ktorý by mal obklopať jeho plochu. Pre poskytovaný model sú čísla zoradene takto: top, left, bottom a right. Horná hodnota alebo „top“ zobrazuje vzdialenosť horného okraja obdĺžnika od vrchu obrázka v pixeloch. Ľavá hodnota alebo „left“ reprezentuje vzdialenosť ľavého okraja zľava od vstupného obrazu. Takto obdobne sú riešené aj časti „bottom“ a „right“ kde sa berie vzdialenosť od spodnej časti vstupného obrazu resp. z pravej časti vstupného obrazu.

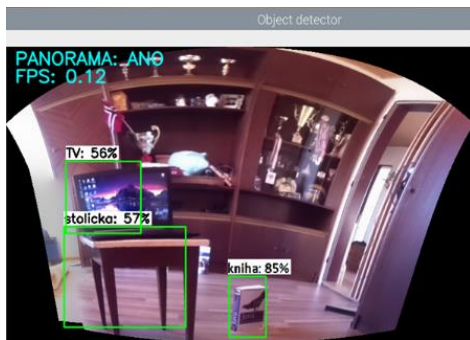


Obr. 3 Vyobrazenie bez spojenia obrazu so zlou klasifikáciou objektov

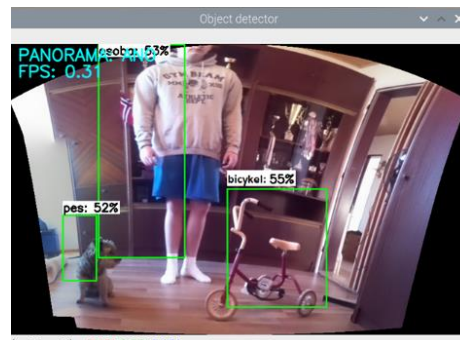


Obr. 2 Vyobrazenie bez spájania obrazov s úspešnou klasifikáciou objektov

Na Obr.5 môžeme vidieť vyobrazenie bez spojenia obrazov, pri ktorom bol jeden z objektov zle klasifikovaný a to konkrétne časť bicykla, ktorej algoritmus pridelil 51% podobnosť človeku. Obr. 6 zobrazuje podstatne presnejšiu klasifikáciu objektov pri vyobrazení bez spojenia obrazov, avšak jeden z predmetov, konkrétne pohár nie je zdetegovaný. Jednou z možných príčin mohlo byť prekrytie väčším objektom, ktorý sa nachádzal za ním.



Obr. 4 Panoramatický obraz
s chybou pri klasifikácii



Obr. 5 Panoramatický obraz
s úspešnou klasifikáciou objektov

Na Obr.7 môžeme vidieť panoramatický obraz, avšak so zlou klasifikáciou objektu, pri ktorej bol objekt notebook klasifikovaný ako televízia resp. monitor.

Na Obr.8 je vyobrazený panoramatický obraz s úspešnou klasifikáciou. Detekcia miesta, kde sa dané objekty mali nachádzať nie je úplne presná.

Ako je možné vidieť v niektorých prípadoch nebol panoramatický pohľad realizovaný. Možných dôvodov môže byť viacero a to zlé nasmerovanie kamier alebo nedostatočné osvetlenie ale v mnohých prípadoch to môže byť aj kvôli štruktúre pozadia. Taktiež je z daných výsledkov vidieť, že detekcia resp. klasifikácia objektov nemusí byť v každom prípade presná. Je možné ich vylepšiť pomocou tzv. tréningu modelov.

ZÁVER

V tejto publikácii boli opísané rôzne riešenia problematiky spájania videa v reálnom čase z bezpilotných prostriedkov. Pri realizácii vytváraní algoritmu boli využité najnovšie poznatky, ktoré sa využívajú pri spracovaní panoramatického obrazu, detekcii a klasifikácii objektov. Táto téma ponúka veľmi zaujímavú výzvu do budúcnosti. Čo sa týka využitia poznatkov a výsledkov, ktoré boli dosiahnuté v tejto práci a ich použitím pri bezpilotných lietadlách, takýto obdobný systém by mohol byť využívaný pri katastrofických udalostiach, pri hľadaní osôb, identifikácii rôznych predmetov a podobne.

LITERATÚRA

- [1] S. Yahyanejad, D. Wischounig-Strucl, M. Quaritsch, and B. Rinner. Incremental mosaicking of images from autonomous, small-scale UAVs. In Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS '10), Aug. 2010.
- [2] D. Wischounig-Strucl, M. Quartisch, and B. Rinner. Prioritized data transmission in airborne camera networks for wide area surveillance and image mosaicking. In Proceedings of CVPRW '11, Jun. 2011
- [3] A. Brown, C. Gilbert, H. Holland, and Y. Lu. Near real-time dissemination of geo-referenced imagery by an enterprise server. In Proceedings of the GeoTec Event, Jun. 2006.
- [4] G. B. Ladd, A. Nagchaudhuri, M. Mitra, T. J. Earl, and G. L. Bland. Rectification, georeferencing, and mosaicking of images acquired with remotely operated aerial platforms. In Proceedings of ASPRS 2006 Annual Conference, page 10 pp., Reno, NV, USA, May 2006.
- [5] L. Lou, F.-M. Zhang, C. Xu, F. Li, and M.-G. Xue. Automatic registration of aerial image series using geometric invariance. In Proceedings of IEEE International Conference on Automation and Logistics, pages 1198–1203, 2008.
- [6] D. Wischounig-Strucl, S Yahyanejad, M. Quaritsch, B. Rinner. Incremental Mosaicking of Images from Autonomous, Small-Scale UAVs, August 2010
- [7] [Online]: <https://coral.ai/products/accelerator/>

Ovládanie robota v prostredí virtuálnej reality

¹Tomáš JUŠČÍK, ²Ján VAŠČÁK

¹Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská Republika

²Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská Republika

¹tomas.juscik@student.tuke.sk, ²jan.vascak@tuke.sk

Abstrakt – Cieľom tejto práce je predstaviť problematiku diaľkového ovládania mobilného robota s využitím virtuálnej reality. V rámci riešenia tejto problematiky sa práca zaoberá tvorbou virtuálneho prostredia, dynamickým vkladáním objektov do takéhoto prostredia, jeho modelovaním ako aj návrhom algoritmu na navigáciu robota. V tejto práci sú preto teoreticky popísané pojmy ako virtuálna realita, rozšírená realita ako aj možnosti, akými je možné túto úlohu zhotoviť. Napokon sa práca venuje samotnej realizácii návrhu a simulácii vyvinutého riešenia. Následne sú popísané vykonané experimenty a ich vyhodnotenie.

Kľúčové slová – navigácia, riadenie, robot, virtuálna realita

I. ÚVOD

V súčasnosti je robot bežnou technológiou, ktorou disponuje skoro každá domácnosť ako aj firmy, či sa už jedná o menšie alebo väčšie podniky. Medzi najväčšiu dilemu vo svete robotiky patrí to, ako správne navigovať robota v priestore.

Cieľom tejto práce bolo namodelovať virtuálny model robota, vložiť ho do virtuálneho prostredia a následne zostrojiť softvér na jeho navigáciu v tomto virtuálnom prostredí ako aj vkladanie objektov, ktorým sa má vyhnúť.

II. VIRTUÁLNA REALITA

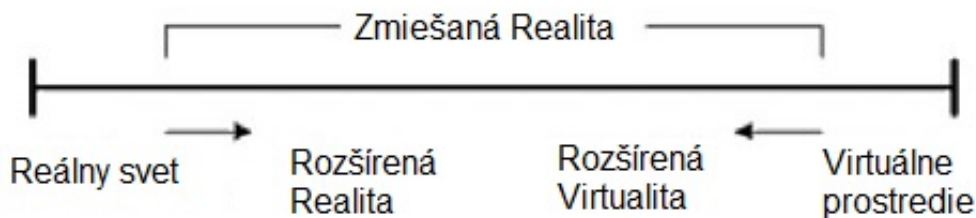
Pojem virtuálna realita (VR) bol definovaný mnohými rôznymi spôsobmi (napr. [1]), čo zapríčiniť, že momentálne VR môže znamenať rôzne veci v rôznych kontextoch. VR môže mať rozsah od jednoduchého prostredia, ktoré je prezentované na stolnom počítači, až po úplne pohlcujúce multisenzorické prostredie, obsahujúce hardvér, akým môže byť napríklad špeciálny odev alebo špeciálne okuliare na hlavu [2], [3].

Vo všetkých svojich prejavoch sa VR pokladá za spôsob, ako simulovať, alebo replikovať prostredie a následne ponúknuť užívateľovi pocit, že sa v danom prostredí nachádza, preberá kontrolu a osobne interaguje s týmto prostredím pomocou svojho vlastného tela.

Okrem simulácie trojrozmerného prostredia majú všetky formy VR spoločný vstup ako aj kontrolu počítača. Je všeobecne známe, že pravá podstata VR spočíva v jej počítačom generovaných 3D svetoch. Účastníkov ponorí rozhranie VR do 3D syntetizovaného prostredia, ktoré je vytvorené jedným alebo viacerými počítačmi, a tak im umožňuje konať v reálnom čase v tomto prostredí pomocou jedného alebo viacerých ovládacích zariadení a zapojením jedného alebo viacerých fyzických zmyslov. Výsledkom virtuálnej reality je stimulácia zmyslov účastníka, ktorá dodáva živý dojem, že je ponorený do syntetického prostredia, s ktorým účastník interaguje.

Na základe niekoľkých koročných skúsenosti je možné tvrdiť, že VR predovšetkým nachádza svoje využitie ako inštruktážna technológia, napr. lekárske alebo zubné vzdelávanie. Inžinierske vzdelávanie do oblasti podporujúcich účinnosť VR ako inštruktážneho nástroja si taktiež nachádza cestu [2].

V poslednej dobe je badať tendenciu, že sa výskum zameriava na tzv. pohlcujúcu virtuálnu realitu viac než na počítačovú virtuálnu realitu. Zatiaľ čo virtuálna realita má výhody ako inštruktážna technológia, výskumníci poukázali aj na jej obmedzenia. Jedným dôležitým obmedzením je vysoká požadovaná úroveň zručnosti a nákladov potrebných na vývoj a implementáciu VR, najmä ak sa jedná o pohlcujúcu virtuálnu realitu.



Obr. 1 Milgramova reálno-virtuálna kontinua. [3].

Na rozvoj pohlcujúcej VR sú potrebné veľmi vysoké odborné znalosti z oblasti programovania a grafiky ako aj veľmi drahý hardvér a softvér na ich efektívne používanie. Aj keď nová technológia VR pre stolné počítače dramaticky znížila požiadavky na zručnosti a na náklady pre potrebný hardvér, stále je potrebné investovať značné financie ako aj čas.

Ďalšia sada obmedzení VR vyplýva z povahy vybavenia, ktoré je nutnosťou. Dlhodobým problémom s pohlcujúcou VR sú obavy o zdravie a bezpečnosť z dlhodobého hľadiska používania, napr. bolesti hlavy, nevoľnosti, narušenia rovnováhy a ďalších fyzických problémov spôsobenými používaním týchto systémov. Aj keď tieto problémy zo súčasného výskumu VR do značnej miery zmizli, keďže vybavenie medzitým zlepšilo a zdá sa, že v nových stolových počítačových systémoch došlo k vylepšeniu, o dlhodobých fyzických alebo psychologických účinkoch používania VR sa vie stále len málo.

III. ROZŠÍRENÁ REALITA

Rozšírenú realitu môžeme definovať ako priamy alebo nepriamy pohľad na fyzické prostredie skutočného sveta v reálnom čase, ktoré bolo vylepšené alebo rozšírené o informácie generované virtuálnym počítačom [3].

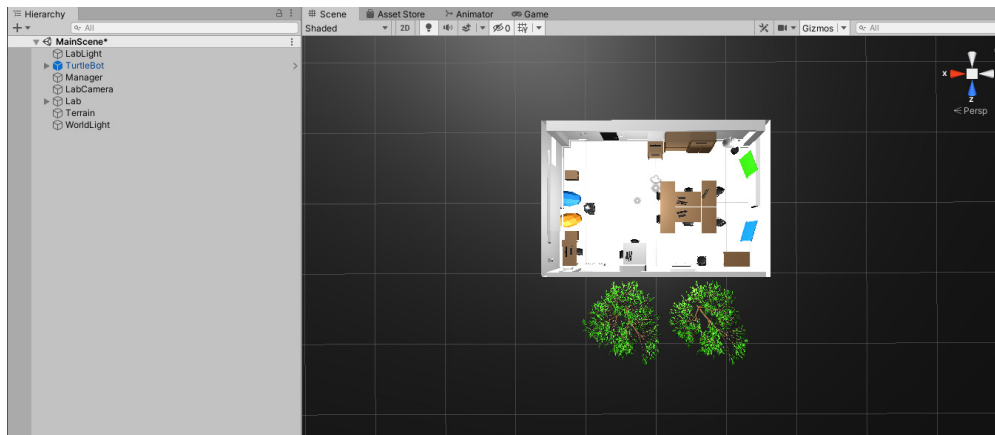
Rozšírená realita je interaktívna a registrovaná v 3D prostredí a kombinuje skutočné ako aj virtuálne objekty. Tzv. Milgramova reálno-virtuálna kontinua je definovaná ako kontinuum, ktorého rozpätie medzi reálnym prostredím a virtuálnym prostredím zahŕňa rozšírenú (augmented) realitu (AR) a rozšírenú virtualitu (AV), kde AR smeruje bližšie k reálnemu svetu a AV smeruje bližšie k virtuálnemu prostrediu, ako je možno vidieť na Obr. 1.

Hlavným cieľom rozšírenej reality je zjednodušiť život používateľa a tým, že mu virtuálne informácie prináša nielen do jeho bezprostredného okolia, v ktorom sa nachádza, ale aj do akéhokoľvek nepriameho pohľadu na prostredie v skutočnom svete, ako je napríklad tok živého videa. Rozšírená realita zvyšuje vnímanie a interakciu používateľa so skutočným svetom. Kým technológia VR úplne ponorí používateľa do syntetického sveta bez toho, aby videl skutočný svet, technológia AR zvyšuje pocit reality tým, že vkladá do prostredia virtuálne objekty a naráža na skutočný svet v reálnom čase.

AR nie je možné obmedziť len na konkrétny typ zobrazovacích technológií, ako je napríklad HMD, alebo iba na zrakové zmysly. Rozšírená realita môže byť potenciálne aplikovaná aj na všetky ostatné zmysly ako napr. pach, dotyk alebo sluch. Rozšírenú realitu možno použiť aj ako rozšírenie alebo nahradenie, ak používateľ prišiel o nejaký zmysel, zmyslovou substitúciou, ako je napríklad rozšírenie zraku pre nevidiacich alebo slabo vidiacich používateľov pomocou zvukových podnetov alebo rozšírenie sluchu pre nepočujúcich používateľov pomocou vizuálnych podnetov.

Je možný aj prístup, ktorý upravuje AR pre aplikácie, ktoré vyžadujú odstránenie skutočných objektov z prostredia. V tomto prípade sa AR nahrádza pomenovaním ako je *sprostredkovaná realita* alebo *znížená realita*, kde sa odstránené objekty nahradzujú virtuálnymi objektmi [4]. Odstránenie objektov zo skutočného sveta reálne zodpovedá pokrytiu objektu virtuálnymi informáciami, ktoré zodpovedajú pozadiu, aby sa v užívateľovi vzbudil dojem, že sa tam objekt nenachádza. Virtuálne objekty pridané do reálneho prostredia ukazujú užívateľovi informácie, ktoré užívateľ nemôže priamo zistiť pomocou svojich zmyslov.

Informácie odovzdané pomocou virtuálnych objektov môžu používateľovi pomôcť pri vykonávaní každodenných úloh, akými sú napríklad navádzanie pilotov v lietadle, kde sa zobrazujú digitálne informácie prostredníctvom náhlavnej súpravy. Informácie taktiež môžu slúžiť na sledovanie, ako napríklad u aplikáciách *Wikitude* alebo u inej mobilnej aplikácii využívajúcej rozšírenú realitu. Existuje mnoho ďalších tried aplikácií využívajúcich AR, ako napríklad lekárska vizualizácia, zábava, reklama, údržba a opravy, anotácie alebo navigácia robotov [3].



Obr. 2 Pohľad na *MainScene* s obsahujúcimi komponentami.

IV. SOFTVÉROVÁ A SIMULAČNÁ REALIZÁCIA RIADIACEHO SYSTÉMU ROBOTA

Na úvod je potrebné zdefinovať, v akom hernom engine, čiže programovacím prostredím a programovacím jazyku, sa bude realizovať vývoj ovládania robota vo virtuálnom prostredí. Ako herný engine, ako bolo spomenuté pri návrhu, sa použije *Unity3D*. Ako programovací jazyk pre programovanie algoritmov bude slúžiť programovací jazyk *C#* a programovacie prostredie pre tento jazyk bude zastrešovať *Visual Studio 2019 Community*, ktorý je veľmi kompatibilný s týmto programovacím jazykom ako aj s herným jadrom *Unity3D*. celý systém môžeme rozdeliť na tzv. front end a back end.

Front end aplikácie slúžia na vizualizáciu činnosti práce, ktorou je ovládanie robota vo virtuálnom prostredí. Vo front end-e sú použité prvky, ktoré obsahuje samotné jadro *Unity3D*, akými sú napríklad kamera, osvetlenie, rôzne 3D objekty ako napr. kocka alebo rovina. Na front end-e sa vizualizuje scéna, v ktorej sa nachádza samotný model laboratória, robota (v našom prípade *TurtleBot*), taktiež jeho pohyb k zadanej destinácii, ukazovateľ kurzora, cieľová destinácia označená užívateľom, alebo vložené objekty.

Front end obsahuje predovšetkým komponent *MainScene*, ktorý ponúka sklad pre všetky ostatné komponenty, ktoré majú byť použité v danom náhľade aplikácie. Tento typ komponentu ako taký samotný nemá žiadne vlastnosti, s ktorými by sa dalo pracovať. *MainScene* je znázornená na Obr. 2 a obsahuje tieto ďalšie komponenty ako napr. *LabLight*, *TurtleBot*, *Manager*, *WorldCamera*, *Lab*, *Terrain* a *WorldLight*.

Back end obsahuje niekoľko základných komponentov, ktorých úloha spočíva hlavne v spracovaní údajov, vrátane vygenerovania riadiacich signálov pre robota. Okrem toho plní aj niektoré podporné úlohy pre samotný front end. Na Obr. 3 je zobrazený základný vzťahový diagram medzi týmito komponentami a ďalej si stručne popíšeme niektoré z nich.

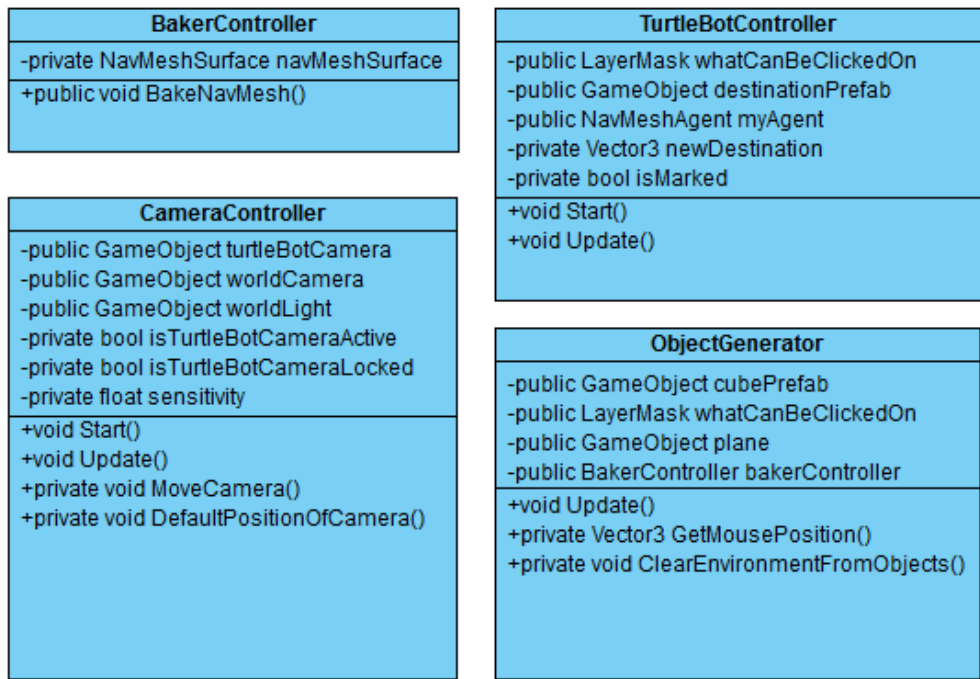
Komponent *BakerController* slúži na úpečenie "povrchu", čiže sa zistí, ktoré miesta na povrchu sú priechodné robotom a ktoré nie. Táto funkcia, ako je možné vidieť na Obr. 3, obsahuje jednu súkromnú globálnu premennú *navMeshSurface*, ktorá predstavuje front end-ový komponent *Plane* a verejnú metódu *BakeNavMesh()*, ktorá zahŕňa kód pre tento úkon. *CameraController* má za úlohu zaisťovať funkcie pre všetky kamery, ktoré sa nachádzajú na scéne. Najviac funkcionality zabezpečuje pre kameru, ktorá je nasadená na mobilnom robotovi. *ObjectGenerator* umožňuje vkladať objekty do virtuálneho prostredia, v ktorom sa nachádza robot. Napokon komponent *TurtleBotController* predstavuje samotnú navigáciu a riadenie pohybu robota *TurtleBot* [5], ktorý využíva funkcie ponúkané knižnicou *Unity.AI*.

V. EXPERIMENTY A ICH VYHODNOTENIE

Za účelom overenia funkčnosti navrhnutého systému sa vykonávali dva typy experimentov, a to experimenty s tieňmi a experimenty s vkladanými objektami.

Experiment s tieňmi sa zameriava na vplyv tieňov na plynulosť obrazu ovládania robota vo virtuálnom prostredí. Najprv sa vykoná ovládanie robota s vypnutými tieňmi, následne sa vykonajú experimenty ovládania robota so zapnutými tieňmi.

Tabuľka 1 znázorňuje údaje získané pri vykonávaní jednotlivých experimentov s rôznymi parametrami. Prvé dva experimenty boli vykonávané najprv bez vkladaných objektov, kde sa len vyplí a zaplí tieň. Je vidno, že zapnutie tieňov má veľký vplyv na plynulosť obrazu. Následné ďalšie štyri pokusy sa vykonávali už aj s vloženými objektami. Najprv sa vložilo 10 kusov objektov, rôzne rozmiestnených vo virtuálnom prostredí a následne sa zistovala hodnota dosiahnuteľnosti



Obr. 3 Vzt'ahový diagram komponent ovládania robota vo virtuálnom prostredí.

maximálneho počtu vytvorených snímkov za sekundu (frames per second – FPS) pri zapnutých a vypnutých tieňoch.

Tabuľka 1
 Tabuľka maximálneho počtu vytvorených snímkov za sekundu pri experimente s tieňmi.

Číslo experimentu	Tiene	Vložené objekty	FPS
1.	Vypnuté	Nie	29
2.	Zapnuté	Nie	18
3.	Vypnuté	Áno(10 ks)	29
4.	Zapnuté	Áno(10 ks)	18
5.	Vypnuté	Áno(20 ks)	29
6.	Zapnuté	Áno(20 ks)	18

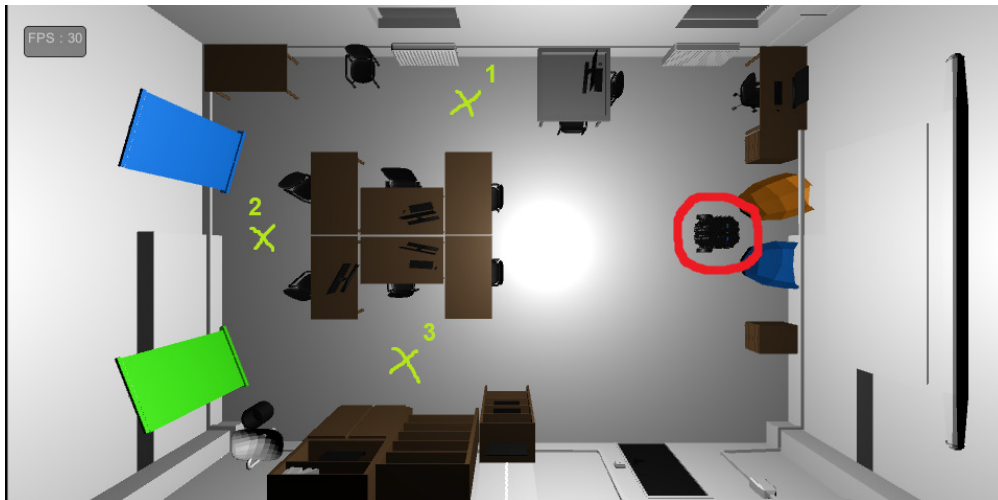
Tento experiment ukázal, že obraz je plynulejší pri vypnutých tieňoch ako pri zapnutých. Počet vložených objektov nie je podstatný pri zvyšovaní alebo znižovaní hodnoty FPS.

Experiment s vkladacími objektami sa venuje dobe presunu robota zo štartovného bodu k predpísaným destináciám vo virtuálnom prostredí, ktoré sú znázornené krížikmi na Obr. 4. Pri každej destinácii sa robot ocitne v situácii, keď má voľnú cestu bez prekážok a v situácii, keď mu budú do cesty vkladané objekty.

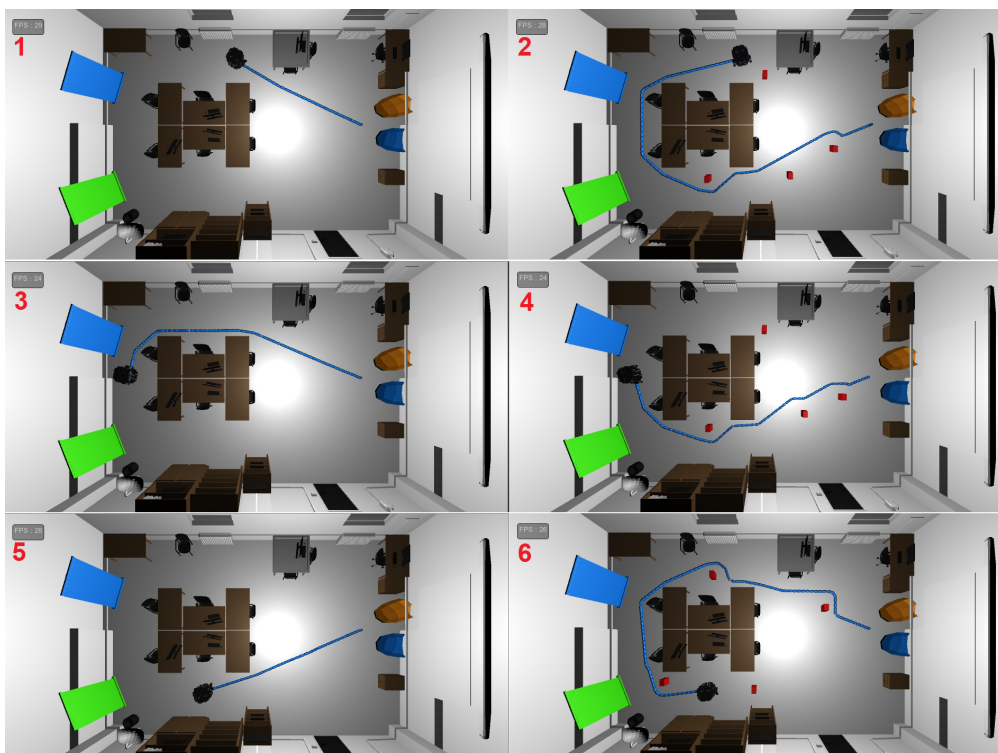
Na Obr. 5 sú znázornené vykonané experimenty. Vľavo v hornom rohu sú čísla experimentov. Ako je možné vidieť v Tabuľke 2, najkratší čas robot dosiahol pri presune bez prekážok na pozíciu 1. V situácii, keď sa pred robota vkladali virtuálne objekty, robot prišiel v najhoršom čase pri presune na pozíciu 1. Najnižší rozdiel času pri presune bez prekážok ako aj s prekážkami bol pri pozícii 2., kde je rozdiel iba necelá sekunda.

Tabuľka 2
 Tabuľka získaných údajov pri experimente s vkladacím objektom.

Číslo experimentu	Číslo destinácie	Počet prekážok	Čas
1.	1	0	5.19s
2.	1	4	16.31s
3.	2	0	9.81s
4.	2	4	10.76s
5.	3	0	6.09s
6.	3	4	15.17s



Obr. 4 Znáznorné použité destinácie vo virtuálnom prostredí.



Obr. 5 Znáznorné experimenty vo virtuálnom prostredí.

VI. ZÁVER

Cieľom tejto práce bolo namodelovať virtuálneho robota, virtuálny priestor, vyvinúť ovládanie tohto robota vo virtuálnom prostredí, ako aj vyvinúť funkciu na vkladanie objektov do virtuálneho prostredia.

Výsledky experimentov potvrdzujú úspešnosť tohto návrhu. Táto práca má potenciál na rozšírenie do budúcnosti, keďže mobilné roboty sú čím viac používané v priemyselnej výrobe a je potrebné, aby sa robot vedel navigovať aj v prípade, že pri navigácii narazí na nečakanú prekážku. V tejto práci je implementovaná knižnica *RosSharp*, ktorá do budúcnosti ponúka možnosť napojiť virtuálny model robota na reálny robot pomocou pripojenia na jeho IP adresu a riešiť také úlohy, ako je napr. mapovanie prostredia [6].

POĎAKOVANIE

Táto práca je finančne podporená Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky prostredníctvom projektu KEGA č. 033

TUKE-4/2018 – Smart Industry/Architektúry inteligentných informačných a kybernetických systémov.

LITERATÚRA

- [1] A. Alrazzah, F. Lamya, and L. Fattouh, “Environments and system types of virtual reality technology in STEM: A survey,” *International Journal of Advanced Computer Science and Applications*, vol. 8, pp. 77–89, 06 2017.
- [2] L. J. Ausburn and F. B. Ausburn, “Desktop Virtual Reality: A Powerful New Technology for Teaching and Research in Industrial Teacher Education,” *Journal of Industrial Teacher Education*, vol. 41, no. 4, pp. 1–16, 2004.
- [3] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic, “Augmented reality technologies, systems and applications,” *Multimedia Tools Applications*, vol. 51, no. 1, pp. 341—377, Jan. 2011.
- [4] M. R. Marnier, R. T. Smith, J. A. Walsh, and B. H. Thomas, “Spatial user interfaces for large-scale projector-based augmented reality,” *IEEE Computer Graphics and Applications*, vol. 34, no. 6, pp. 74–82, 2014.
- [5] A. d. H. B. M. Tavares, S. P. Madruga, A. Brito, and T. P. Nascimento, “Non-linear model predictive formation control applied to non-holonomic autonomous mobile robots,” in *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, 2017, pp. 1–6.
- [6] H. Aagela, M. Al-Nesf, and V. Holmes, “An asus_xtion_probased indoor mapping using a raspberry pi with turtlebot robot,” in *2017 23rd International Conference on Automation and Computing (ICAC)*, 2017, pp. 1–5.

Vizuálna interakcia s prostredím inteligentného priestoru

¹Ján VAŠČÁK, ²Richard RUSŇÁK

¹ Katedra Kybernetiky a Umelej Inteligencie, Fakulta Elektrotechniky a Informatiky, Technická Univerzita Košice, Slovenská Republika

² Katedra Kybernetiky a Umelej Inteligencie, Fakulta Elektrotechniky a Informatiky, Technická Univerzita Košice, Slovenská republika

¹jan.vascak@tuke.sk, ²richard.rusnak@student.tuke.sk

Abstrakt — Táto práca prináša prehľad problematiky vizuálnej interakcie s prostredím inteligentného priestoru. Taktiež obsahuje prehľad technológií využívaných pri tejto interakcii, ktoré zvyšujú úroveň interakcie medzi používateľom a systémom. Súčasťou týchto technológií je samotná vizualizácia, hlasové ovládanie a samostatnosť systému. Pri realizácii tohto systému sme vytvorili webovú aplikáciu zahŕňajúcu spomenuté technológie. Výsledkom je aplikácia voľne sprístupnená pomocou internetového pripojenia.

Kľúčové slová — cloudové systémy, hlasoví asistenti, inteligentný priestor, Raspberry Pi, vizualizácia

I. ÚVOD

V dnešnej dobe je stále čo raz bežnejšie ovládanie zariadení okolo človeka pomocou softvéru navrhnutému špeciálne pre ich kontrolu. Či už ide o bežné domáce spotrebiče alebo mohutnejšie systémy pre väčšiu množinu zariadení. Cieľom tejto práce je ukázať na bežné softvérové a hardvérové prostriedky, s ktorými je pomerne jednoducho možné zrealizovať vizuálnu interakciu s prostredím inteligentného priestoru, ktorú predstavuje jednak 3D model a jednak riadiaci modul, umožňujúci buď manuálne riadenie, alebo automatické riadenie s využitím expertného systému. Za týmto účelom je uvedený jednak stručný prehľad najmodernejších prostriedkov používaných pre tento typ úloh a jednak vlastný návrh architektúry riešenia riadenia procesov v školskom laboratóriu. V ďalšom ukážeme fungovanie aplikácie, spôsoby vizualizácie a ovládania. Napokon poukážeme na budúce možné vylepšenia a rozšírenia.

II. PROSTRIEDKY INTERAKCIE S PROSTREDÍM INTELIGENTNÉHO PRIESTORU

V tejto časti uvedieme stručný prehľad základných komponentov, ktoré sa využívajú pri návrhu systémov na vzdialenú interakciu človeka s prostredím ako sú prostriedky vizualizácie prostredia, strojového učenia, hlasového ovládania, či cloudových služieb.

Hlavným cieľom vizualizácie v systémoch je zjednodušenie predstavy hlavných častí systému. Z počiatku existovala len 2D vizualizácia, ktorá mala veľký úspech a z tohto dôvodu počas vývoja výpočtovej techniky prišla na rad 3D vizualizácia s novými možnosťami [1].

Skúsenosti počas vývoja softvérov pomocou vizualizácie zaznamenávajú lepšie porozumenie systému s nižšími nákladmi vývoja. Podstatu vizualizácie predstavuje vytvorenie obrazu daného systému pomocou vizualizovaných objektov. Často sú tieto objekty vizualizované za behu určitých procesov v zložitejšom systéme a taktiež sú zahrnuté 3D objekty obsahujúce animácie, čo zvyšuje atraktivnosť vizualizačného systému a zapamätateľnosť ľudského vizuálneho systému. Medzi najmodernejšie 3D modelovacie nástroje patria ArchiCAD, Tekla a Blender [1].

V tejto práci sa najviac venujeme vizualizáciám inteligentných priestorov. Cieľom bolo vytvorenie 3D modelu predstavujúceho inteligentný priestor schopný interakcie s používateľom. Nevyhnutnou súčasťou pri tvorbe takého modelu je optimalizácia kvôli 3D animáciám prehrávaným v reálnom čase. Základnou vlastnosťou vizualizačných systémov je udržanie vysokej frekvencie snímok za sekundu. Priemernú hodnotu predstavuje 15Hz, pričom ak dôjde k nižšej frekvencii, klesá záujem interakcie používateľa so systémom. Vo vykresľovanom priestore existuje ohraničenie množstva 3D objektov, ktorý je možné optimalizovať pomocou rôznych techník [1]:

- Optimalizácia – využíva dostupný hardvér a pomocou GPU vykonáva požiadavky najvyššej kvality. V situácii, kedy výkon GPU nepostačuje, systém využíva CPU. Optimalizovať vizualizáciu je možné pomocou vopred uložených objektov do pamäte GPU, tým pádom nie je potrebný prenos údajov pre jednotlivé snímky. Ďalšie spôsoby optimalizácie sú spojenie geometrických údajov objektov nachádzajúcich sa vedľa seba, redukcia kvality materiálov objektov a vykresľovanie rovnakých objektov spolu.
- Úroveň detailov – je úmerné znižovanie kvality objektov v závislosti ku vzdialenosti a aktuálnemu pohľadu. To znamená, že pri zobrazovaní vzdialenejších objektov nižšou kvalitou nedochádza k nižšiemu vizuálnemu dojmu používateľa.
- Viditeľnosť objektov – predstavuje zobrazenie objektov nachádzajúcich sa vo viditeľnom priestore kamery. Objekty nachádzajúce sa mimo tohto pohľadu nie je potrebné vykresľovať, z tohto dôvodu sa výkon viac využíva pre viditeľné objekty. Tento princíp je častou súčasťou pri vizualizovaní rozsiahlych priestorov.

Moderné technológie sú často súčasťou inteligentných priestorov, napríklad budov, laboratórií alebo bežných domácností, od ktorých sa vyžaduje určitá autonómnosť. Pre tieto účely sú preto často používané algoritmy strojového učenia a umelej inteligencie [2].

Jedným z prostriedkov využívaných na tento účel sú expertné systémy, ktoré sú založené na pravidlách, blízkych ľudskej reprezentácii znalostí. Príklad pravidla v spojení s inteligentným priestorom vyzerá nasledovne [2]:

AK teplota ≤ 20 **POTOM** zapni radiátor **INAK** vypni radiátor

Súbor takýchto pravidiel sa nazýva „systém založený na pravidlách“. Základ predstavuje súbor pravidiel, kde jednotlivé pravidlo predstavuje podmienku obsahujúcu takzvanú predpokladovú (ľavú) a záverovú (pravú) stranu kontextového pravidla. Tieto pravidlá sú definované expertom, ktorý dokáže správne reagovať na špecifické situácie. Ľavá strana pravidla sa skladá z niekoľkých podmienok, ktoré sa porovnávajú s ich faktami, pričom všetky podmienky musia byť splnené. Pri uplatnení viacerých pravidiel, sa ich závery zlučujú a miera vplyvu daného pravidla na celkovom výstupe závisí od jej tzv. sily, čo je miera kompatibility jej predpokladu s reálnymi nameranými vstupmi [2].

Hlasové ovládanie vo veľkej miere zvyšuje úroveň interakcie používateľa s inteligentným priestorom. Hlasová interakcia so systémami je jeden z významných cieľov umelej inteligencie. Taktiež je táto oblasť jedna z najrýchlejšie rozvíjajúcich sa. Príkladom týchto systémov sú hlasoví asistenti Cortana od Microsoftu, Siri od Applu, Alexa od Amazonu alebo Assistant od Googlu. Ich hlavnou úlohou je rýchle a efektívne riešenia rôznych úloh prostredníctvom hovoreného prejavu. V dnešnej dobe tieto systémy nie sú obmedzované typom otázok, ale dokážu reagovať na široké množstvo požiadaviek. Spôsob návrhu záleží od aplikácie a jej zložitosti [3].

V budúcnosti sa predpokladá, že hlasoví asistenti budú bežnou súčasťou telefónov, inteligentných okuliarov, kuchynských spotrebičov, televízorov, hracích konzol a ďalších podobných zariadení. Tieto systémy tvoria dôležitú súčasť interakcie človeka s inteligentným priestorom. Hlavným dôvodom je nepotrebnosť manuálnej interakcie, čo vedie ku zvýšeniu komfortu používateľa [3].

Samotný návrh takéhoto systému závisí od zložitosti aplikácie. Najkomplexnejší návrh predstavuje tzv. multi-modálny dialógový systém, ktorý ponúka spracovanie viacerých kombinácií používateľských vstupov ako reč, obrázky alebo gestá [3].

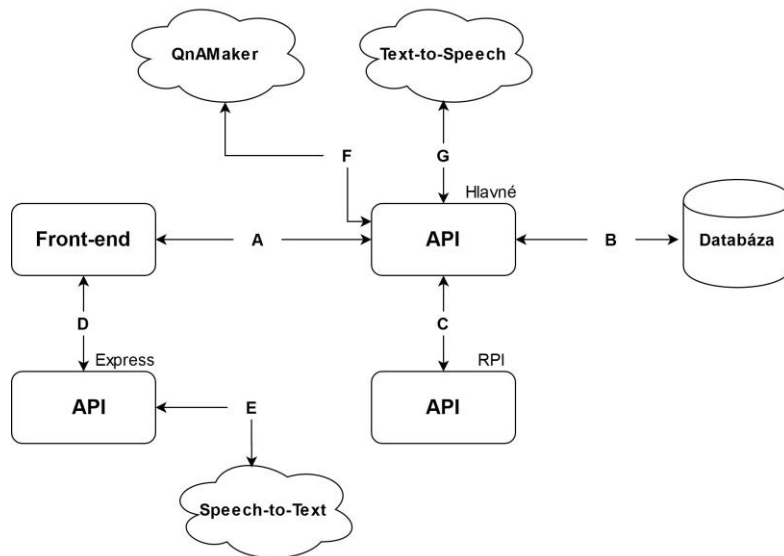
Cloudové služby sú častou súčasťou hlavne hlasových asistentov. Medzi najznámejšie služby patria:

- LUIS – služba od spoločnosti Microsoft, ktorá dokáže zistiť význam vety pomocou technológií strojového učenia a následne spracuje získané kľúčové informácie [4].
- QaA Maker – taktiež služba od Microsoftu, ktorá sa využíva ak báza znalostí obsahujúca statické informácie, ak postačuje vždy rovnaký typ odpovede [5].
- Speech-to-Text – služba spoločnosti Google, ktorá dokáže efektívne transformovať hovorenú vetu do textu [7].
- Text-to-Speech – taktiež služba Googlu, ktorá transformuje text do podoby ľudskej reči [6].

III. ARCHITEKTÚRA VIZUÁLNEJ INTERAKCIE S INTELIGENTNÝM PROSTREDIEM

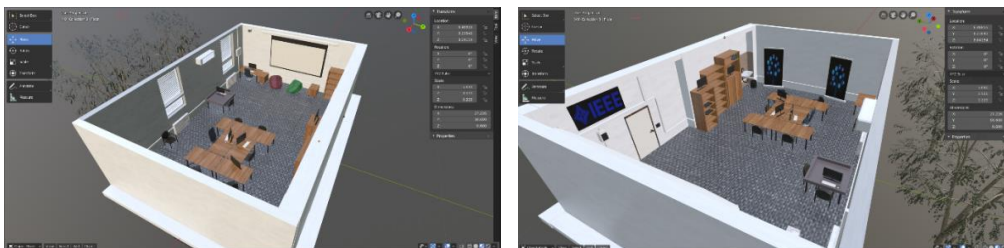
Cieľom bolo vytvoriť aplikáciu jednoducho dostupnú pre viacerých používateľov. Z tohto dôvodu sme sa rozhodli pre vývoj webovej aplikácie pozostávajúcej z front-endu a dvoch aplikačných programovacích rozhraní (API). Prvé API je spustené na zariadení Raspberry Pi 3 Model B, ku ktorému je pripojené hardvérové rozhranie GrovePi so senzormi AllThingsTalk. API naprogramované v knižnici Express slúži pre účely hlasového ovládania, ktoré bude

podrobnejšie popísané neskôr. Komunikácia medzi komponentami prebieha pomocou REST architektúry. Ako databázu sme zvolili PostgreSQL, ktorá je taktiež spustená na Raspberry Pi. Na Obr. 1 sa nachádza celková architektúra navrhnutého riešenia.



Obr. 1 Architektúra aplikácie pre vzdialenú interakciu s inteligentným prostredím.

Prvotným cieľom bolo vytvorenie 3D modelu priestoru. Pre realizáciu tohto problému sme zvolili softvér Blender 4.0. Vizualizovali sme laboratórium Internetu vecí, katedry kybernetiky a umelej inteligencie, Technickej Univerzity v Košiciach, viď Obr. 2.



Obr. 2 Dva vizualizačné pohľady na použitý inteligentný priestor.

Na front-ende sme použili JavaScript knižnicu React, ktorá dokáže importovať 3D model vytvorený softvérom Blender pomocou knižnice ThreeJS. Model sme importovali pomocou GLTF Loadera, ktorý efektívne vykresľuje model aj s jeho materiálmi. Aplikácia taktiež simuluje nepriaznivé údaje zo senzorov zaznamenávajúcich plyn a oheň. V prípade zaznamenania týchto hodnôt sa zapnú vizualizácie, viď Obr. 4. Medzi ďalšie vizualizované zariadenia patrí klimatizácia v podobe zmeny farby 3D objektu, natočenie roliet okien a streamovanie kamery v reálnom čase. Systém taktiež zaznamenával pohyb v priestore, teplotu a vlhkosť.

Ako vidieť na Obr. 1, vytvorili sme 2 API. Prvé, spustené na Raspberry Pi, je naprogramované pomocou jazyka Python a knižnice Flask. Hlavnou úlohou tohto API je poskytovanie reálnych dát senzorov klientskej aplikácii. K modulu GrovePi boli pripojené tieto senzory:

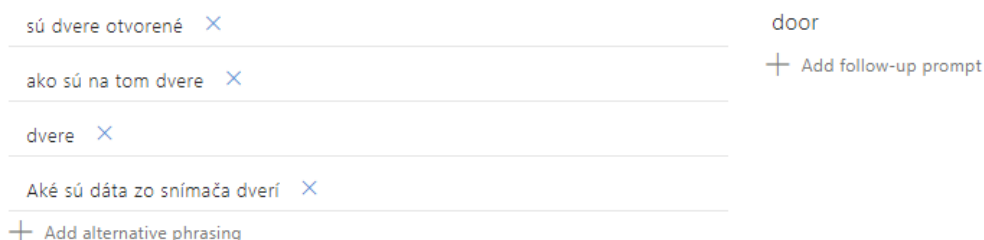
- DHT11 – senzor teploty a vlhkosti
- PIR – senzor zaznamenávajúci pohyb
- Magnetický spínač – senzor simuluje otváranie dverí
- MQ3 – senzor zaznamenávania plynu
- KY-026 – senzor zaznamenávania ohňa
- Prepínač – simuluje zapnutie klimatizácie
- Svetelný senzor – ponúka lepšiu vizualizáciu osvetlenia v priestore
- USB Web kamera – ponúka možnosť streamovania v reálnom čase
- Rotor – simuluje natočenie roliet okien
- 3 LED diódy – simuluje zapnuté svetlo v priestore

- Bzučiak – simuluje alarm v prípade zaznamenania ohňa alebo plynu

Toto API je napojené na cloudovú službu QnAMaker pre potrebu hlasového ovládania. Taktiež ponúka autonómne riadenie priestoru. Druhé API, naprogramované v jazyku JavaScript pomocou knižnice Express, bolo vytvorené z dôvodu využívania cloudových služieb Text-to-Speech a Speech-to-Text. Knižnica Flask nedokázala správne spracovať zvukové dáta posielať z klientskej aplikácie a zároveň tieto služby nedokázali prijímať požiadavky z dôvodu spustenia API pod balíkom Evelnet. Express tieto služby využíval bezproblémovo.

Hlasové ovládanie poskytuje interakciu pomocou hlasu používateľa. Pomocou tohto hlasového asistenta je možné hlasom zistiť stav zariadenia alebo prepnúť ho do iného stavu. Vzhľadom na zložitosť vývoja takého asistenta sme sa rozhodli využiť cloudové služby.

Aplikácia pozostáva z panelu pre využitie tejto služby. V panely sú dve tlačidlá a na spodnej časti sa ukazuje text na základe hlasu v reálnom čase. Po stlačení tlačidla „Nahrávať“ sa spustí nahrávanie hlasu, ktoré sa posiela na službu Speech-to-Text. Po stlačení tlačidla „Stop“ sa nahrávanie zastaví a celá veta sa posiela na službu QnAMaker, ktorá ju vyhodnotí a pošle naspäť príslušnú odpoveď. Na Obr. 3 je znázornený jeden typ odpovede. Na ľavej strane sú možnosti otázok ohľadom senzora dverí. Odpoveď prichádza v type „door“. Táto odpoveď sa následne posiela front-end aplikácii, ktorá vygeneruje odpoveď vo forme celej vety. Táto celá veta následne vstupuje do služby Text-to-Speech, pomocou ktorej nastane hlasová odpoveď systému.



Obr. 3 Príklad činnosti služby QnAMaker.

Tento systém bol naprogramovaný pomocou jazyka Python a knižnice Expert slúžiacej pre vývoj expertných systémov a sa spúšťa na zariadení Raspebrry Pi. Vytvorili sme 11 pravidiel, ktoré definujú správanie systému na základe zaznamenaných dát zo senzorov. V Tabuľka 1

sú dve z jedenástich pravidiel expertného systému. V hlavičke sú vstupy do tohto systému, na základe ktorých sa vyvodí akcia. Tento systém je spustený ak používateľ zvolí zapnutie inteligentného režimu v aplikácii.

Tabuľka 1
Príklad pravidiel expertného systému.

Čas	Svetlá	Pohyb	Plyn	Oheň	Okná	Bzučiak	Klíma	Teplota	AKCIA
>16	×	✓							Zapnutie svetla
					×		✓		Zatvorenie okien

Po vytvorení celkovej webovej aplikácii bolo cieľom ju sprístupniť viacerým používateľom. Front-end aplikáciu sme spustili na cloudovom serveri Heroku. Rovnakým spôsobom sme riešili nasadenie API slúžiace pre hlasové ovládanie. Vzdialený prístup pre API spustenom na Raspberry Pi sme riešili pomocou externej IP adresy a presmerovania portov na lokálnej sieti.

IV. EXPERIMENTY

Samotné experimenty spočívali v otestovaní funkčnosti a rýchlosti odozvy systému na zmeny na zmeny stavov prostredia a príkazy od vzdialeného užívateľa. Boli zostavené modelové situácie. V ďalšom uvádzame na ilustráciu jednu z nich.

Na Obr. 4 je vo vizualizačnom okne znázornený prvotný stav prostredia, konkrétne otvorenie obidvoch okien a zapnutie svetla. Pri zadaní príkazu sa zmeny v modely prejavujú v reálnom čase pomocou socketovej komunikácie.

Cieľom tohto experimentu je vrátiť pôvodný stav pomocou 3D modelu. Po kliknutí na obe okná a vypínač svetla sa stav dostáva do pôvodného, čo znamená úspešné dokončenie experimentu. Danú akciu signalizujú aj kontrolné LED diódy, ktoré sú pripojené na Raspberry Pi a aj týmto spôsobom dostáva užívateľ informáciu o vykonaní akcie.

Podobne môžeme otestovať aj bezpečnostný systém pozostávajúci zo simulácie ohňa a plynu

v 3D modely. Ak je napr. zaznamenaný oheň v miestnosti, spustí sa aj vizualizácia v 3D modely, vid' Obr. 5a. Podobne sa vizualizuje správanie aj ostatných senzorov a zariadení. Napr. vo vizualizácii sa zaznamená pohyb motorčekov, ktoré na príkaz zatvoria okná (Obr. 5b), alebo zapnutie klimatizácie spôsobí menu farby jej ikony v 3D modeli.



Obr. 2 vizualizácia vstupov



a)



b)

Obr. 5 Ukážky zobrazovania vykonania akcií na základe povelu užívateľa (ručne alebo expertný systém) vo vizualizačnom 3D modeli: a) – animácia ohňa; b) – animácia zatvárania okna.

Na Obr. 6 je zobrazený panel s hlasovým ovládaním, na ktorom je zaznamenaná veta: „Zavri okno jedna.“ Na základe tohto povelu sa v modeli zavrie okno bližšie ku stene s plátňom. Podobným spôsobom je možné sa odkázať na ostatné zariadenia a senzory.

V základnom ovládacom okne je možné systém prepnúť do tzv. inteligentného režimu (tlačidlo „Smart mode“), v ktorom sa spúšťa expertný systém a vykonávajú sa požadované akcie na základe zadefinovaných pravidiel, vid' Tab. 1.



Obr. 6 Panel hlasového ovládania (okno vpravo dole).

ZÁVER

Vytvorený systém pre vizuálnu interakciu s inteligentným priestorom funguje spoľahlivo aj pri ovládaní viacerých používateľov v rovnakom čase. Obmedzením je neprítomnosť bezpečnostného ovládania na API spustenom na Raspberry Pi. Riešením by bolo prepojenie

Raspberry so serverom, ktorý poskytuje tento certifikát. Ako vylepšenie aplikácie by bol kvalitnejší nahrávací systém hlasu používateľa a presnejšia metóda pre rozpoznanie významu vety. Expertný systém taktiež funguje spoľahlivo na aktuálny stav senzorov v priestore. Ďalšie vylepšenie by bolo použitie lepšej kamery pre potreby streamovania v reálnom čase.

POĎAKOVANIE

Táto práca je finančne podporená Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky prostredníctvom projektu KEGA č. 033TUKE-4/2018 - Smart Industry/Architektúry inteligentných informačných a kybernetických systémov.

LITERATÚRA

- [1] M. Johansson, M. Roupé and P. Bosh-Sijsema, „Real-Time Visualisation of Building Information Model (BIM)“, Chalmers University of technology, Sweden, 2015, pp. 1-25
- [2] I. uddin, „A Rule-based framework for Developing Context-Aware Systems for Smart Spaces“, Thesis submitted the University of Nottingham, 2019
- [3] V. Kepuska and G. Bohoute, „Next-generation of Virtual Personal Assistent (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home)“, Electrical & Computer Engineering Department, Florida Institute of technology, Melbourne, FL, USA, 2015, pp. 99-103
- [4] Microsoft Azure, „What is LUIS?“, 2020, dostupné na internete: <<https://docs.microsoft.com/en-us/azure/cognitive-services/luis/what-is-luis>>
- [5] Microsoft Azure, „From data to bot in minutes“, 2020, dostupné na internete: <<https://www.qnamaker.ai/>>
- [6] Google Cloud, „Cloud Text-To-Speech“, 2020, dostupné na internete: <<https://cloud.google.com/speech-to-text>>
- [7] Google Cloud, „Speech-To-Text“, 2020, dostupné na internete: <<https://cloud.google.com/speech-to-text>>

Vplyv ironických príspevkov na presnosť analýzy sentimentu

¹Kristína MACHOVÁ, ²Peter SMOLNICKÝ

^{1,2} Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹kristina.machova@tuke.sk, ²peter.smolnicky@student.tuke.sk

Abstract — Človek dnešnej doby je vystavený obrovskému množstvu informácii, ktoré sa snaží efektívne využiť pri svojich rozhodnutiach. Potrebné informácie môže získať zo skúseností ostatných ľudí vo svojom blízkom okolí alebo prostredníctvom iných informačných zdrojov. Skúsenosti a názory môžeme vyhľadávať v súčasnosti napríklad cez Internet, kde nájdeme veľké množstvo informácií. Tieto informácie môžu byť vo forme objektívnych a subjektívnych viet. Objektívne vety obsahujú iba fakty. Subjektívne vety obsahujú vyjadrenia, v ktorých ľudia opisujú svoje skúsenosti alebo pocity vzhľadom na určitý objekt alebo jeho vlastnosti. Jedná sa o vyjadrenia osobného postoja autora textu. Manuálne vyhodnotenie obrovského množstva informácií je extrémne náročné, jedná sa o zložitý proces a predstavuje problém aj pre ľudí. Preto sa na túto úlohu používajú nástroje analýzy sentimentu. Cieľom je pomôcť pochopiť text a emóciu pisateľa textu. Analýza sentimentu, tiež známa ako dolovanie názorov, je mladá výskumná oblasť, ktorá sa v poslednom čase progresívne vyvíja. Čelí viacerým problémom a výzvam. Medzi hlavné problémy, ktoré sa riešia pri analýze sentimentu patrí aj rozpoznávanie ironie a vplyv ironických príspevkov na analýzu sentimentu. Správna interpretácia ironických poznámok má významný vplyv na presnosť vyhodnotenia subjektívneho textu pomocou nástrojov analýzy sentimentu. V tejto práci sa používa na dolovanie názorov vopred vytvorený nástroj na analýzu sentimentu. Skúma sa vplyv ironických príspevkov pri rôznom počte príspevkov v danej dátovej množine. Vyhodnotia sa jednotlivé dátové množiny s rôznym počtom ironických príspevkov a následne sa porovnávajú výsledky. Zisťuje sa primárne to, či ironické príspevky skresľujú reálne výsledky analýzy sentimentu a keď áno, tak v akej miere.

Keywords — analýza sentimentu, dolovanie názorov, spracovanie prirodzeného jazyka, polarita názoru, ironické príspevky, vplyv ironie na presnosť analýzy

I. ÚVOD

Online médiá Vzhľadom na rastúci objem dostupných textových informácií existuje veľký dopyt po technikách spracovania prirodzeného jazyka (NLP - Natural Language Processing), ktoré dokážu automaticky spracovať texty, podporovať vyhľadávanie informácií a komunikáciu prostredníctvom Internetu. Techniky spracovania prirodzeného jazyka riešia nejednoznačné jazykové štruktúry, ktoré ľudia používajú v každodennom prejave. Častým fenoménom v komunikácii človeka, ktorý sa vyskytuje v hovorenom aj písomnom prejave a ktoré je potrebné brať do úvahy, resp. zvážiť sú napríklad slangy, gramatické chyby, dialekty, a takisto obrazový jazyk. Obrazový jazyk ako je ironia, sarkazmus, metafora, atď. predstavujú vážnu výzvu pre spracovanie prirodzeného jazyka. Zistiť z textu, čo si ľudia myslia, pomáha spoločnostiam, politickým stranám a iným verejným subjektom pri strategických rozhodnutiach. Keď sa ľudia zapájajú do neformálnej diskusie, takmer určite používajú ironiu na to, aby vyjadrili niečo iné, čo je uvedené vo vete. Z uvedeného dôvodu veľa úsilia sa v poslednom čase venovalo odhaľovaniu a riešeniu fenoménov obrazového jazyka v texte. V tejto súvislosti sa táto práca zameriava na hodnotenie toho, ako špecifické oblasti obrazového jazyka, ironia, resp. sarkazmus ovplyvňujú nástroje analýzy sentimentu. Článok uvádza niektoré prístupy venované detekcii ironie a následnému vplyvu ironických príspevkov na presnosť analýzy sentimentu, kde sa skúma tento vplyv pri rôznom počte príspevkov v dátových množinách a porovnáva výsledky.

II. ANALÝZA SENTIMENTU

A. Dolovanie názorov

Existujú dva základné prístupy k dolovaniu názorov z textu a to:

- prístup založený na strojovom učení

- slovníkový prístup.

Strojové učenie disponuje algoritmami, pomocou ktorých na základe poskytnutých údajov sú počítače schopné sa učiť. Jedná sa o konanie počítača bez toho, aby bol presne naprogramovaný. Program vďaka týmto algoritmom sa efektívne prispôsobuje zmenám bez potreby editácie kódu programu. O významnom využívaní strojového učenia sa dá hovoriť iba v posledných rokoch, keď jeho využiteľnosť sa týka značného počtu odvetví. Dôvodom je prienik informačných technológií do obrovského množstva odborov. Vďaka tomu vznikajú obrovské množstvá dát, ktoré môžu byť využité v strojovom učení. Pomocou algoritmov strojového učenia je počítač naučený znalostiam z veľkého množstva tréningových dát, ktoré následne dokáže aplikovať na neznáme prípady a tak predikovať buď triedu (klasifikačná úloha) alebo presnú hodnotu (regresná úloha). Hlavnou výhodou počítačov v tomto je takmer neobmedzené množstvo tréningových dát a takisto ich neporovnateľne rýchlejšie a presnejšie spracovanie počítačom v porovnaní s človekom.

Slovníkový prístup používa slovník emocionálnych slov na určenie polaritu príspevku. Slovníkový prístup vyhľadáva slová vyjadrujúce sentiment a ich sumarizáciou určuje celkovú polaritu názoru prezentovaného v texte [1]. „Keď text obsahuje viac pozitívnych slov ako negatívnych slov, vyjadruje pozitívny názor, keď viac negatívnych slov ako pozitívnych, vyjadruje negatívny názor.“ Hodnotenie sentimentu iba podľa počtu pozitívnych a negatívnych slov by bolo nepresné, pretože kontextový význam slov môže zmeniť pozitívny význam v negatívny a opačne. Pri analýze sa preto neposudzujú iba samotné slová, ale aj ich frázy. Napríklad veta obsahujúca pozitívne slovo „vynikajúce“ môže vyjadrovať negatívny sentiment: „Malo to byť vynikajúce.“ Pri analýze názorov pomocou slovníkového prístupu je zásadné spracovať aj intenzifikáciu slova slovom (spracovanie fráz) a negácie [2].

Úlohou analýzy názorov nie je vyťažovanie faktov, ale zisťovanie subjektívneho kontextu textov a postoja pisateľa textu ku skúmanej téme. V súčasnosti existuje množstvo otvorených problémov ako je rozpoznanie irónie, presná identifikácia negácie a intenzifikácie, identifikácia sentimentu kontextu a pod., ktoré robia vyhodnotenie sentimentu textu veľmi zložitou záležitosťou. Analýza názorov z textov alebo viac špecifickejšie z obsahu sociálneho webu je rýchlo rastúca oblasť výskumu. V súčasnosti sa tento výskum rozšíril mimo počítačovej vedy na manažérske vedy a spoločenské vedy z dôvodu jej dôležitosti pre podniky a spoločnosť ako celku. Rastúci význam analýzy názorov sa zhoduje s rastom sociálnych médií, ako sú recenzie, diskusie na sociálnych sieťach, hodnotenia produktov a služieb na stránkach dodávateľov a ďalších platformách umožňujúcich vyjadrenie názoru. Údaje na sociálnych stránkach obsahujú neštruktúrované údaje vyslané rôznymi typmi ľudí. Na základe týchto príspevkov môžeme analyzovať názory o konkrétnom produkte, službe, udalosti alebo inej záležitosti. To hrá dôležitú úlohu pri umožnení podnikom aktívne pracovať na zlepšovaní obchodnej stratégie a získať hĺbkový pohľad na spätnú väzbu kupujúceho o ich produkte. V súčasnom svete sú recenzie a názory, ktoré máme k dispozícii, kľúčovým faktorom, ktorý formuje naše perspektívy a ovplyvňuje úspech značky, služby alebo produktu.

B. Irónia a jej Formy

Irónia je efektívny, ale náročný spôsob komunikácie, ktorý umožňuje rečníkovi vyjadriť názory bohaté na sentiment s humorom. Je to akýkoľvek jemný, zastretý výsmech či posmech. Z pohľadu literatúry iróniou sú zvolené slová, ktoré sa úmyselne používajú na označenie iného významu ako doslovného. Irónia je sofistikovaná forma prejavu, v ktorej autori napíšu opak toho, čo si myslia. Irónia je premenlivým jazykovým fenoménom, chýbajú jej explicitné signály a nedá sa presne definovať. Z Gibbsovej analýzy vyplýva, že v bežnej komunikácii medzi priateľmi existuje viacero foriem irónie [3]. Tieto formy sa rozlišujú aj tým ako na nich ich príjemcovia reagujú. Za základné delenie irónie považujeme Attardovo delenie na iróniu negatívnu a pozitívnu [4].

- Negatívna irónia označovaná aj ako klasická, obyčajne vyjadruje kritiku.
- Pozitívna irónia ako je žartovanie, podpichovanie s cieľom rozhnevať adresáta, tým, že hovoríme niečo negatívne, ale s pozitívnym zámerom, je menej častá a menej účinná. Jej používanie je obvykle medzi blízkymi priateľmi. Vedecká verejnosť sa obyčajne sústreďuje na skúmanie negatívnej irónie.

Ďalším deliacim kritériom irónie je delenie na základe formy a účinku, pričom pre účely tejto práce použijeme Gibbsovo rozlíšenie [3] na päť druhov irónie:

- Žartovnosť – sa dá jednoducho nazvať ako pochvala kritikou. Tento druh irónie je dôkazom vzájomnej blízkosti. Hovorca pozitívnu iróniu použije v prípade, že riziko nepochopenia je minimálne a prípadné nedorozumenie je možné vysvetliť.

- Rečnicke otázky – sú typu „Musíš ukladať to prádlo na zem?“ Hovorca sa formálne pýta, ale nečaká odpoveď. Rečnická otázka má len upozorniť na nejakú nepríjemnosť.
- Hyperbola – obsahuje detailné vyjadrenie, zveličuje popisovanú skutočnosť, jednoducho povedaná robí „z komára somára“.
- Zľahčovanie – zámerne znižuje mieru negatívneho dopadu.
- Sarkazmus – je definovaný ako irónia, ktorá je značne podpichovačná až zraňujúca v porovnaní s bežnou rečou. Môžeme ju charakterizovať ako agresívnu formu irónie s jasným zámerom a cieľom. Sarkazmus je osobitná forma irónie, ktorou osoba vysielala implicitné informácie, zvyčajne opak toho, čo hovorí. Môžeme povedať, že pokiaľ je vyjadrenie ironické a zároveň až cynické tak hovoríme o sarkazme.

Z pohľadu tejto bakalárskej práce sa sarkazmus chápe ako najbežnejšia forma irónie, vychádzajúc z predpokladu, že bežní ľudia oba termíny používajú synonymne.

III. VPLYV IRONICKÝCH PRÍSPEVKOV NA PRESNOSŤ ANALÝZY

Metodológia skúmania vplyvu ironických príspevkov na presnosť analýzy sentimentu je založená na sloníkovej metóde analýzy sentimentu, ktorá je aplikovaná na dáta s rôznym percentuálnym zastúpením ironických príspevkov.

A. Popis dátových množín

Dáta určené na analýzu sentimentu sa dajú získať dvomi spôsobmi. Jeden spôsob je vymyslenie, vytvorenie množiny príspevkov vrátane ironických, ich zaznamenanie a následná analýza sentimentu. Druhý spôsob, ktorý bol použitý aj v tejto práci, je nájdenie a predspracovanie dostupnej množiny príspevkov vrátane ironických. Nájdenie množiny dát na základe nami stanovených kritérií bolo prácne a časovo náročné. Po analýze niekoľkých vytvorených dátových množín bola nájdená jedna, ktorá obsahovala dostatočný počet ironických príspevkov a vyhovovala našim požiadavkám. Zvolená dátová množina obsahovala anglické príspevky (formát .xlsx). Vyskytovalo sa v nej približne 2500 príspevkov z ktorých 500 bolo ironických. Dataset obsahoval ironické príspevky označené číslom 1 a neironické číslom -1. V Exceli použitím filtra boli rozdelené ironické a neironické príspevky za účelom ľahšej práce a manipulácie s nimi. Z neironických príspevkov bolo vybraných 300 príspevkov, 150 s negatívnou a 150 s pozitívnou polaritou. Boli vybrané také príspevky, pri ktorých sa s istotou dala určiť polarita, tzn. príspevky jasné, s jednoznačným obsahom bez dvojzmyselných slov a fráz. Takto vytvorená dátová množina bola nazvaná dataset 100-0, čo znamená, že obsahuje 100% neironických a 0% ironických príspevkov. Ďalšie množiny dát boli vytvorené pomocou pridávania ironických príspevkov ku datasetu 100-0. Tak vznikli dataset 90-10, dataset 80-20 a dataset 70-30.

B. Nástroj na analýzu sentimentu

Na analýzu sentimentu existuje dnes množstvo efektívnych nástrojov. Bol využitý online nástroj nazvaný SmartReader. K uvedenému nástroju bol prístup cez Google na stránke <https://smartreader.paralldots.com/>. Je to platený nástroj na analýzu sentimentu využívaný veľkými a uznávanými firmami ako napríklad Dolby. Pri registrácii nám bolo automaticky pridelených 4000 bodov. To znamená 4000 príspevkov na analýzu za nulové poplatky. Dataset určený na analýzu bol načítaný vo formáte .xlsx (Excel). V excelovskom súbore musel byť vytvorený stĺpec Content, v ktorom museli byť zadané všetky analyzované príspevky za účelom ich rozoznania zvoleným nástrojom analýzy sentimentu. Po prevedenej analýze výsledky boli obdržané na zadanú e-mailovú adresu pri registrovaní. Do načítaného súboru nástroj automaticky pridal stĺpec nazývaný sentiment, ktorý obsahoval hodnoty reprezentujúce mieru negative, positive alebo neutral sentimentu.

C. Výsledky testov a ich analýza

Na výpočet výsledkov ukazovateľov miery efektívnosti analýzy sentimentu bol použitý nástroj nachádzajúci sa na internetovej stránke <http://onlineconfusionmatrix.com/>. Uvedený nástroj pracuje s hodnotami správne pozitívny (True Positive TP), správne negatívny (True Negative TN), chybné pozitívny (Fail Positive FP) a chybné negatívny (Fail Negative FN). Uvedené hodnoty boli vnesené do kontingenčnej tabuľky a z nej boli odvodené hodnoty nasledovných mier efektívnosti TPR=Návratnosť, PPV=Presnosť, F1 score a ACC=Správnosť (Accuracy). Konkrétne výsledky testov podľa jednotlivých datasetov sú zobrazené v Tabuľke 1.

Tabuľka 1
Výsledky testov analýzy sentimentu pri rôznej miere ironických príspevkov v tréningových dátach

Dataset	Návratnosť	Presnosť	F1 skóre	Správnosť
100-0	0,852	0,847	0,850	0,850
90-10	0,843	0,796	0,819	0,827
80-20	0,833	0,743	0,786	0,803
70-30	0,827	0,720	0,770	0,795

Ako je vidieť v Tabuľke 1, hodnoty F1 sa pridávaním ironických príspevkov postupne znižovali približne o 0.03 v rámci jednotlivých datasetov, s výnimkou posledného datasetu, kde sa táto hodnota oproti predchádzajúcemu datasetu znížila iba o 0.0154. Hodnoty Správnosti (ACC) sa znižovali približne o 0.02. Z uvedeného vyplýva, že zvyšovanie podielu ironických príspevkov v datasete zhoršuje výsledky sledovaných ukazovateľov, t.j. znižuje presnosť výsledkov analýzy sentimentu.

IV. ZÁVER

Článok sa zameriaval na zistenie vplyvu ironických príspevkov na presnosť analýzy sentimentu. Dosažené výsledky sú vcelku priaznivé, keďže Správnosť (ACC) a F1 score sú nad hodnotou 0.75. Napriek tomu, našou úlohou nebolo dosiahnuť čo najlepšie výsledky efektívnosti analýzy sentimentu ale zistenie ako sa miera efektívnosti líši v rámci jednotlivých datasetov s rôznym percentuálnym zastúpením ironických príspevkov. Výsledky testov nám jednoznačne poukazujú na to, že tieto odchýlky nie sú zanedbateľné. S vyššou mierou ironických príspevkov, miera efektívnosti analýzy sentimentu klesá.

To znamená, že v analýze sentimentu je potrebné venovať mimoriadnu pozornosť ironickým príspevkom, ich vplyv na presnosť analýzy sentimentu je významný. Eventuálnym ignorovaním ironie by došlo k značnému skresľovaniu výsledkov miery efektívnosti analýzy sentimentu. V zmysle záverov tejto práce všetky aktivity vynaložené na zdokonaľovanie nástrojov na detekciu ironických príspevkov majú svoje opodstatnenie. Pokusy o automatickú detekciu ironických príspevkov a riešenie výziev v tejto oblasti majú byť predmetom budúcich výskumov v oblasti analýzy sentimentu. Budúci výskum sa bude zameriavať taktiež na identifikovanie sarkazmu cez hashtagy. Kvôli tomu bol vytvorený hashtag tokenizátor pre GATE, takže sentiment a sarkazmus nachádzajúci sa v hashtagoch sa dá ľahšie odhaliť [5]. Taktiež bolo vytvorených mnoho pravidiel [6], ktoré nám umožňujú zvýšiť presnosť analýzy sentimentu v prípade výskytu sarkazmu.

POĎAKOVANIE

Práca prezentovaná v tomto článku bola podporená Slovenskou agentúrou na podporu vedy a výskumu v rámci projektu APVV-017-0267 "Automatizované rozpoznávanie antisociálneho správania v online komunitách".

LITERATÚRA

- [1] B. Donkor, On social sentiment and sentiment analysis [online], 2016, Dostupné na internete: <<http://brnrd.me/social-sentiment-sentiment-analysis>>.
- [2] M. Míkula, K. Machová, Spracovanie negácie pre klasifikáciu názorov v slovenskom jazyku. In: Data a znalosti 2015, 1-2.10.2015, Praha, VŠB – TU Ostrava, 2015, 41-45, ISBN 978-80-248-3824-3
- [3] R.W. Gibbs, Irony in talk among friends. *Metaphor and Symbol* 15, 2000, 5-27.
- [4] S. Attardo, Irony as relevant inappropriateness. In: *Journal of Pragmatics* 32,2000, 793-826.
- [5] H. Cunningham et al., GATE: an Architecture for Development of Robust HLT Applications. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, 7–12 July 2002, ACL '02, Stroudsburg, PA, USA. Association for Computational Linguistics, 2002, 168–175.
- [6] D.G. Maynard, M.A. Greenwood, Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. In: LREC 2014 Proceedings. Language Resources and Evaluation Conference (LREC), 26-31 May 2014, Reykjavik, Iceland [online], 2014, Dostupné na internete: <http://eprints.whiterose.ac.uk/130763/>. ISBN 978-2-9517408-8-4

Deskripcia regionálneho tvaru pomocou Zernikeho momentov

Ján MIHALÍK, Iveta GLADIŠOVÁ

Katedra elektroniky a multimediálnych telekomunikácií, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

jan.mihalik@tuke.sk, iveta.gladisova@tuke.sk

Abstrakt — Článok pojednáva o Zernikeho momentoch, ktoré majú matematické vlastnosti, vďaka ktorým sú ideálnymi obrazovými funkciami, ktoré sa dajú použiť ako deskriptory tvarov pri problémoch s ich klasifikáciou. Majú invariantné rotačné vlastnosti a za určitých podmienok by mohli byť považované aj za mierkovo a translačne invariantné. Pri ich použití je však potrebné uvažovať mnoho faktorov. V článku sú opísané výsledky použitia Zernikeho momentov ako deskriptorov regionálneho tvaru, ktoré boli dosiahnuté ich implementáciou v programovom prostredí Matlab. V niektorých prípadoch sa pri ich použití ukázala sto percentná presnosť.

Keľúčové slová — Zernikeho momenty, deskriptory regionálneho tvaru, rotačná invariantnosť

I. ÚVOD

Deskriptory tvaru sú užitočným nástrojom používaným v širokej oblasti spracovania obrazu, ako napríklad porovnávanie objektov, klasifikácia, rozpoznávanie a identifikácia [1], [2], [3]. Existuje niekoľko generických deskriptorov tvaru, ktoré sú schopné poskytnúť vysoko dimenzionálny znakový vektor, ktorý presne opisuje špecifické tvary. Niektoré deskriptory opisujú len jednotlivé vlastnosti, ktoré sú prítomné v rôznych tvaroch, ako je kruhovitost', elipticita, pravouhlost', priamočiarost', zložitost', zakrivenie, symetria, atď. [4], [5].

Všeobecne existujú dva typy deskriptorov tvaru:

- Deskriptory obrysového tvaru – využívajú informácie len z obrysov tvaru.
- Deskriptory regionálneho tvaru – využívajú informácie z celej oblasti tvaru.

Dalo by sa povedať, že v minulosti bola väčšia pozornosť venovaná práve deskriptorom regionálneho tvaru. Tieto deskriptory sú robustnejšie s ohľadom na šum. Ďalším dôvodom, prečo je väčšina metód založených na oblasti tvaru namiesto obrysu tvaru je to, že sú výpočtovo jednoduchšie. Napríklad pre výpočet oblasti tvaru stačí spočítať počet obrazových prvkov v tvare, zatiaľ čo výpočet obrysu tvaru nie je priama úloha. V závislosti od konkrétnej situácie a podmienok je potrebné použiť rôzne metódy. Ďalším príkladom by mohli byť geometrické momenty [6], [7]. Tie sa dajú relatívne ľahko vypočítať zo zodpovedajúceho objektového obrazu, zatiaľ čo výpočet obrysu zahŕňa výpočet integrálov dráhy, ktoré sa nedajú jednoducho odhadnúť z diskretných údajov, ktoré sa používajú hlavne pri spracovaní obrazu a počítačovom videní. Na druhej strane, metódy založené na obrysoch tvaru sú vhodnejšie pre vysoko presné úlohy, ako je napríklad identifikácia osôb.

II. DESKRIPTOR NA BÁZE ZERNIKEHO MOMENTOV

V súčasnosti sú Zernikeho polynómy prispôbené na spracovanie obrazu pri aplikáciách rozpoznávania tvarov [8], [9]. Ortogonálne vlastnosti Zernikeho momentov sú vhodnejšie pre takéto aplikácie, pretože na rozdiel od geometrických momentov ich invarianty môžu byť vypočítané nezávisle na ľubovoľných vysokých radoch bez nutnosti počítania invariantov nižších radov. Taktiež ich ortogonálne vlastnosti umožňujú vyhodnotiť, akého vysokého rádu je vhodné počítať momenty na získanie spoľahlivého deskriptora pre zvolenú databázu tvarov [10], [11].

M.R. Teague [12] ako prvý poukázal na to, že bázová funkcia geometrických momentov $x^p y^q$ ich robí ťažko použiteľnými pre aplikácie spracovania obrazu z dôvodu ich informačnej redundancie. Navrhol preto použiť ortogonálne bázové funkcie, a to Zernikeho polynómy, pretože majú aj vhodnú vlastnosť - invariantnosť voči rotácii. Zernikeho polynómy sú definované ako funkcie $V_{n,m}$ [8]

$$V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{n,m}(\rho) e^{jm\theta} \quad (1)$$

Polynóm je rozdelený na 2 časti, na reálnu časť $R_{n,m}(\rho)$ a komplexnú časť $e^{(jm\theta)}$. V tejto rovnici je n celé kladné číslo alebo 0, m je celé číslo také, že $|m| \leq n$ a $(n - |m|)$ je párne číslo, ρ je dĺžka vektora od začiatku po súradnice (x, y) a θ je azimutálny uhol. Radiálny polynóm $R_{n,m}(\rho)$ je definovaný ako [8]

$$R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2}-s\right)! \left(\frac{n-|m|}{2}-s\right)!} \rho^{n-2s} \quad (2)$$

kde n je rád radiálneho polynómu, m reprezentuje opakovanie azimutálneho uhla.

Potrebuje vypočítavať len kladné Zernikeho momenty, lebo platí, že $R_{n,-m}(\rho) = R_{n,m}(\rho)$, pretože vždy berieme absolútnu hodnotu m v rov.(2). Zernikeho polynómy tvoria ortogonálnu množinu, t.j. skalárny súčin medzi dvomi polynómami tejto množiny je rovný 0, pokiaľ nie je násobený sebou samým. Takže pre polynóm $p(x)$ stupňa k platí (d_i sú váhové konštanty)

$$p(x) = d_0 p_0(x) + d_1 p_1(x) + \dots + d_k p_k(x) \quad (3)$$

pre skalárny súčin $\langle \cdot, \cdot \rangle$ s ohľadom na polynóm p_i dostaneme

$$\langle p, p_i \rangle = d_0 \langle p_0, p_i \rangle + d_1 \langle p_1, p_i \rangle + \dots + d_k \langle p_k, p_i \rangle = 0 + 0 + \dots + d_i \langle p_i, p_i \rangle + 0 + 0 \quad (4)$$

a keďže $\langle p_i, p_i \rangle = 0$ pre všetky $i \neq j$, potom môžeme získať riešenie pre d_i z hľadiska p a p_i

$$d_i = \frac{\langle p, p_i \rangle}{\langle p_i, p_i \rangle} \quad (5)$$

Pre Zernikeho polynómy je toto ortogonálne správanie sa vyjadrené nasledovne [8]

$$\int \int_{x^2+y^2 \leq 1} [V_{nm}(x, y)]^* V_{pq}(x, y) dx dy = \frac{\pi}{n+1} \delta_{np} \delta_{mq} \quad (6)$$

kde δ_{ab} je Kroneckerova delta, ktorá sa rovná 1 ak $a = b$, inak sa rovná 0. Zernikeho momenty sú projekciou obrazovej funkcie do týchto ortogonálnych básových funkcií. Zernikeho momenty A_{nm} rádu n s opakovaním m pre digitálny obraz môžeme definovať ako

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) V_{nm}^*(\rho, \theta), \quad x^2 + y^2 \leq 1 \quad (7)$$

Pri výpočte Zernikeho momentov sa stred obrazu považuje za začiatok súradného systému a súradnice obrazových prvkov sú mapované v rozsahu jednotkovej kružnice, t.j. $x^2 + y^2 \leq 1$. Obrazové prvky mimo jednotkového kruhu sa pri výpočte nepoužívajú. Tým, že Zernikeho momenty sú ortogonálne, môžu sa ľahko použiť pri rozpoznávaní tvarov v obraze, avšak ortogonalita nie je ich najvýznamnejšia vlastnosť. Akýkoľvek ortogonálny polynóm by sa mohol použiť ako básová funkcia, z ktorej sa dajú ľahko vykonať rekonštrukcie. Čo robí Zernikeho momenty vhodnými pre úlohu klasifikácie tvarov je to, že sú rotačne invariantné.

III. IMPLEMENTÁCIA DESKRIPTOROV TVARU POMOCOU ZERNIKEHO MOMENTOV V MATLABE

Implementáciu deskriptorov regionálneho tvaru pomocou Zernikeho momentov v programovom prostredí Matlab opíšeme nasledujúcimi podstatnými časťami skriptov navrhnutého programu:

Výpočet radiálnych polynómov:

```
rad = zeros(size(r));
for s = 0:(n-abs(m))/2
    c = (-1)^s * factorial(n-s) / (factorial(s) * factorial((n+abs(m))/2-s) * ...
        factorial((n-abs(m))/2-s));
    rad = rad + c * r.^(n-2*s);
end
```

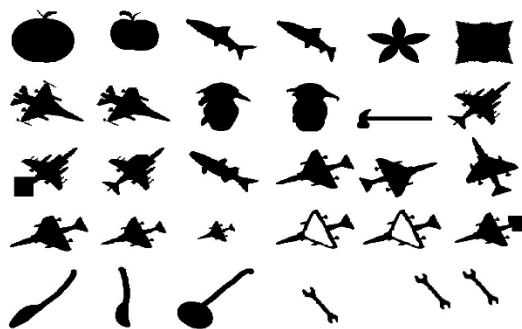
Výpočet Zernikeho básových funkcií:

```
N = size(p,1);
x = 1:N; y = x;
[X,Y] = meshgrid(x,y);
R = sqrt((2.*X-N-1).^2 + (2.*Y-N-1).^2) / N;
Theta = atan2((N-1-2.*Y+2), (2.*X-N+1-2));
R = (R <= 1) .* R;
Rad = radialpoly(R,n,m);
```

Výpočet Zernikeho momentov:

```
Product = p(x,y) .* Rad .* exp(-1i*m*Theta);
V = sum(Product(:));
cnt = nnz(R)+1;
Z = (n+1)*V/cnt;
```

Databáza tvarov objektov, ktoré boli použité pri testovacích simuláciách, je zobrazená na obr.1.



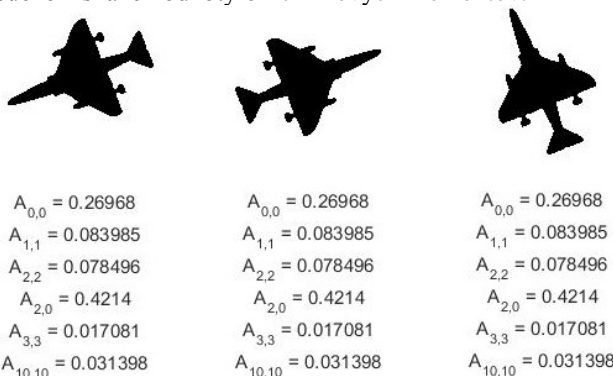
Obr.1 Databáza tvarov

Najprv sme vypočítali hodnoty prvých 17 Zernikeho momentov a jeden moment 10.rádu pre tvar lietadlo (obr.2 vľavo), ich hodnoty sú uvedené v Tab.1. Pri testovaní sme používali prvých 5 Zernikeho momentov $A_{0,0}$, $A_{1,1}$, $A_{2,2}$, $A_{2,0}$, $A_{3,3}$ a aj 10.rádu $A_{10,10}$.

Tab. 1 Tabuľka 18 Zernikeho momentov pre tvar lietadlo.

Zernikeho rád	Hodnota momentu	Zernikeho rád	Hodnota momentu	Zernikeho rád	Hodnota momentu
$A_{0,0}$	0,26968	$A_{4,4}$	0,08822	$A_{6,6}$	0,09086
$A_{1,1}$	0,83985	$A_{4,2}$	0,00706	$A_{6,4}$	0,06046
$A_{2,2}$	0,07850	$A_{4,0}$	0,30642	$A_{6,2}$	0,05093
$A_{2,0}$	0,42140	$A_{5,5}$	0,05979	$A_{6,0}$	0,13412
$A_{3,3}$	0,01708	$A_{5,3}$	0,13288	$A_{7,7}$	0,04257
$A_{3,1}$	0,03140	$A_{5,1}$	0,08279	$A_{10,10}$	0,03140

Potom sme otestovali *invariantnosť deskriptora voči rotácii* tým, že sme porovnali vstupný tvar (lietadlo vľavo na obr.2) s jeho rotovanými tvarmi (na obr. 2 v strede a vpravo). Pod každým z obrázkov sú uvedené získané hodnoty 6 Zernikovych momentov.



Obr.2 Testovanie Zernikovych momentov voči rotačnej invariantnosti tvaru.

Z dosiahnutých výsledkov vidíme, že deskriptory (Zernikove momenty) sú invariantné voči rotácii, čo potvrdilo teoretický predpoklad z odbornej literatúry [8]. Túto vlastnosť deskriptora vieme jednoducho dokázať pomocou nasledujúceho vzťahu [13]

$$A'_{nm} = A_{nm} e^{-j m \alpha} \tag{8}$$

kde α je uhol, o ktorý sme otočili pôvodný tvar. Keďže rotácia tvaru modifikuje len fázu Zernikeho momentu, absolútna hodnota Zernikeho momentu zostáva invariantná.

V ďalšom kroku sme otestovali *invariantnosť deskriptorov voči translácii tvaru*. Dosiahnuté výsledky hodnôt 6 Zernikeho momentov sú uvedené na obr.3. Z tohto testovania je vidieť že deskriptor nie je invariantný voči translácii, aj malý posun tvaru ovplyvní veľkosť deskriptora. Vidíme však, že nulový moment sa nemení, a to z dôvodu že tento moment opisuje len plochu daného tvaru.

Nasledovalo otestovanie *invariantnosti voči zmene mierky (škály) tvaru*. Z obr.4 je možné vidieť že deskriptor nie je invariantný voči tejto zmene a veľkosť deskriptorov klesala so zmenšujúcim sa tvarom.



$A_{0,0} = 0.050615$	$A_{0,0} = 0.050615$	$A_{0,0} = 0.050615$
$A_{1,1} = 0.027637$	$A_{1,1} = 0.047756$	$A_{1,1} = 0.03466$
$A_{2,2} = 0.011821$	$A_{2,2} = 0.011359$	$A_{2,2} = 0.031529$
$A_{2,0} = 0.081777$	$A_{2,0} = 0.036826$	$A_{2,0} = 0.082092$
$A_{3,3} = 0.021169$	$A_{3,3} = 0.02095$	$A_{3,3} = 0.031415$
$A_{10,10} = 0.00083123$	$A_{10,10} = 0.0021101$	$A_{10,10} = 0.033491$

Obr.3 Testovanie Zernikovych momentov na zmeny v translácii tvaru.



$A_{0,0} = 0.26968$	$A_{0,0} = 0.18087$	$A_{0,0} = 0.068719$
$A_{1,1} = 0.083985$	$A_{1,1} = 0.031394$	$A_{1,1} = 0.017362$
$A_{2,2} = 0.078496$	$A_{2,2} = 0.033125$	$A_{2,2} = 3.1635e-05$
$A_{2,0} = 0.4214$	$A_{2,0} = 0.37574$	$A_{2,0} = 0.1774$
$A_{3,3} = 0.017081$	$A_{3,3} = 0.0021255$	$A_{3,3} = 0.0014714$
$A_{10,10} = 0.031398$	$A_{10,10} = 0.0032728$	$A_{10,10} = 1.9159e-05$

Obr.4 Testovanie Zernikovych momentov na zmeny mierky tvaru.

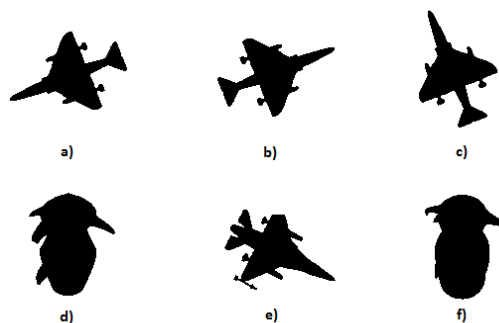
Ďalej sme otestovali *vplyv zmien vo vnútri tvaru* pri zachovanej veľkosti tvaru. Z hodnôt na obr.5 je možné pozorovať výhodu tohto deskriptora, a to, že je citlivý na zmeny tvaru, či už na obryse alebo vo vnútri tvaru.



$A_{0,0} = 0.26968$	$A_{0,0} = 0.16761$	$A_{0,0} = 0.12643$
$A_{1,1} = 0.083985$	$A_{1,1} = 0.061196$	$A_{1,1} = 0.043757$
$A_{2,2} = 0.078496$	$A_{2,2} = 0.087853$	$A_{2,2} = 0.038207$
$A_{2,0} = 0.4214$	$A_{2,0} = 0.1676$	$A_{2,0} = 0.14471$
$A_{3,3} = 0.017081$	$A_{3,3} = 0.026901$	$A_{3,3} = 0.0022753$
$A_{10,10} = 0.031398$	$A_{10,10} = 0.031376$	$A_{10,10} = 0.010568$

Obr.5 Testovanie Zernikovych momentov na zmeny vo vnútri tvaru.

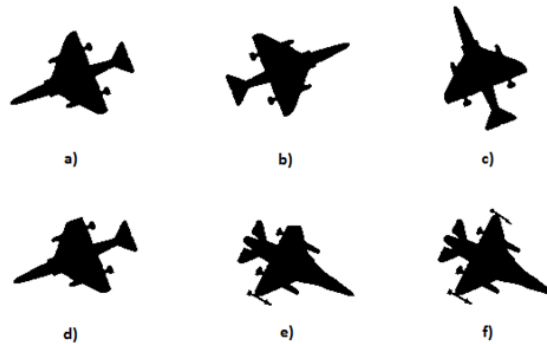
Ako posledný test sme previedli *vyhľadávanie podobných tvarov* z databázy. Buď sme hľadali len na základe jedného vybraného Zernikového momentu, v našom prípade 3. rádu (obr.6), alebo pomocou prvých piatich Zernikeho momentov (obr.7). Vstupným tvarom, pomocou ktorého sa hľadali podobné tvary, bol tvar na nasledujúcom obrázku v časti a).



Obr.6 Nájdené podobné tvary pomocou jedného Zernikeho momentu 3.rádu.

Pri vyhľadávaní podobných tvarov len pomocou jedného Zernikeho momentu 3. rádu vidíme na obr.6 nájdené rovnaké, avšak otočené tvary, to je vďaka invariantnosti deskriptora voči rotácii. Ďalšie nájdené tvary sa už na pôvodný (obr.6a) nepodobajú a vzhľadom na to môžeme zhodnotiť vyhľadávanie podobných tvarov len pomocou jedného Zernikeho momentu ako nie veľmi presné.

Hľadanie podobných tvarov pomocou prvých piatich Zernikeho momentov prináša už lepšie výsledky (obr.7). Všetkých 5 nájdených podobných tvarov je takmer zhodných s porovnávaným tvarom, čím môžeme toto vyhľadávanie vyhodnotiť ako veľmi presné. Použitie Zernikeho momentov na vyhľadávanie podobných tvarov z databázy je tak vhodné iba v prípade, ak sa použije viacej momentov naraz.



Obr.7 Nájdené podobné tvary pomocou prvých piatich Zernikeho momentov.

IV. ZÁVER

Článok bol zameraný na použitie Zernikeho momentov ako deskriptorov regionálneho tvaru a ich implementácie do programového prostredia Matlab. Opísali sme jednotlivé deskripty, princíp fungovania, použité metódy a vlastnosti týchto deskriptorov.

Druhá časť článku bola venovaná implementácii vybraných deskriptorov do prostredia Matlab, v ktorom sme overili vlastnosti, výhody a nevýhody momentových deskriptorov. Z dosiahnutých výsledkov môžeme potvrdiť invariantné rotačné vlastnosti implementovaných deskriptorov a citlivosti na zmeny tvaru. Deskriptor na báze geometrických momentov je z hľadiska širšej invariantnosti určite lepšou voľbou, no disponuje nižšou citlivosťou na malé zmeny tvaru. Deskriptor využívajúci Zernikeho momenty ponúka vlastnosť iba rotačnej invariantnosti, ale už pri malých zmenách tvaru dosahuje lepšie výsledky ako deskripcia pomocou geometrických momentov. Pri hľadaní podobných tvarov z databázy deskriptor na báze Zernikeho momentov preukázal, že je vhodnou voľbou aj pri takýchto aplikáciách, avšak je potrebné porovnávať tvary pomocou viacerých momentov.

LITERATÚRA

- [1] Amanatiadis, A. - Kaburlasos, V.G. - Gasteratos, A. - Papadakis, S.E.: Evaluation of Shape Descriptors for Shape-Based Image Retrieval. IET Image Processing, 5 (5), pp. 493-499, 2011.
- [2] M. Bober, "MPEG-7 visual shape descriptors," IEEE Trans. Circuits and Systems for Video Technology, vol. 11, no. 6, pp. 716-719, 2001.
- [3] Mihalík, J.: Číslkové spracovanie obrazov. LČSOV FEI TU Košice, 2018.
- [4] Kayaert, G. - Wagemans, J. - Vogels, R.: Encoding of Complexity, Shape, and Curvature by Macaque Inferotemporal Neurons. Frontiers in Systems Neuroscience, 04, July, 2011. On line: <https://www.frontiersin.org/articles/10.3389/fnsys.2011.00051/full>
- [5] Mokhtarian, F. - Bober, M.: Curvature scale space representation: theory, applications, and MPEG-7 standardization. Kluwer Academic Pub, 2003.
- [6] Latecki, L.J. - Lakämper, R. - Eckhardt, U.: Shape Descriptors for Non-rigid Shapes with a Single Closed Contour. Proc. of IEEE Conference. on Computer Vision and Pattern Recognition, pp. 424-429, 2000.
- [7] Flusser J. - Suk T. - Zitova B.: Moments and Moment Invariants in Pattern Recognition. Wiley and Sons Ltd., 2009.
- [8] Vorobyov M.: Shape Classification Using Zernike Moments. Technical report. iCamp- University of California Irvine, 2011. Dostupné na internete: <https://pdfs.semanticscholar.org/711b/e25acba94fbf913d2b734d312c27d5fe5226.pdf>
- [9] Tahmasbi A. - Saki, F. - Shokouhi, S.B.: Classification of Benign and Malignant Masses Based on Zernike Moments Computers in Biology and Medicine, vol. 41, no. 8, pp. 726-735, 2011.
- [10] Ruiz, A. - Ujaldón, M.: Exploiting Kepler Capabilities on Zernike Moments, Annals of Multicore and GPU Programming, 1 (1), pp. 27-37, 2014.
- [11] Tahmasbi A. - Saki, F. - Shokouhi, S.B.: An Effective Breast Mass Diagnosis System using Zernike Moments. Proc. of IEEE 17th Iranian Conference on Biomedical Engineering (ICBME'2010), pp.1-4, 2010.
- [12] Teague M. R.: Image analysis via the general theory of moments. Optical Society of America, 70(8), pp.920 - 930, 1979.
- [13] Khotanzad, A. - Hong, Y.H.: Invariant Image Recognition by Zernikeho Moments. IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 12, no.5, pp.489-497, 1990.

Fuzzy approach to Dynamic Systems Control

¹Daniela PERDUKOVÁ, ¹Pavol Fedor

¹ Department of Electrical Engineering and Mechatronics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

² Department of Electrical Engineering and Mechatronics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹daniela.perdukova@tuke.sk, ²pavol.fedor@tuke.sk

Abstract — The paper deals with a simple universal methodology of investigation and fuzzy description of the basic properties of dynamic systems and the possibilities of its application in their control. Basic properties of proposed control structures are presented in model examples. A brief description of its application in the temperature control in an injection moulding machine is provided.

Keywords —dynamic system, fuzzy model, fuzzy system, moulding machine, temperature control

I. INTRODUCTION

In technical practice there are often requirements for control of systems whose parameters, and often also internal structure, are not known in detail or are difficult to identify. In spite of this fact these systems (control of injection temperature in power plant boiler units, control of boiler unit output, control of temperature in different injection moulding machine zones, etc.) are being successfully controlled, most often by skilled and experienced operators, although a precise characteristic of the control is not known and it can in fact differ in dependence on the particular person operating the machine. This is most often an area suitable for the application of fuzzy techniques in the modeling and control of dynamic systems, which are not based on analytical knowledge of the system being investigated, but require knowledge of its quality characteristics [1]-[3]. These can essentially be obtained in two ways – from experts or from a database of suitably measured data of the investigated object [4]-[6]. The latter of the two (if applicable) is much more convenient, because it does not require any experts and it can be automated. However, in the course of its application several serious problems may be encountered – the consistency of the measured data database, insufficient coverage of the whole working /operating range of the investigated equipment, requirement for a large number of rules for achieving sufficient quality of the fuzzy system, difficulties in defining the dynamic part of the fuzzy system, etc [7]-[9]. This paper presents a method of modeling the basic quality characteristics of common dynamic thermal systems and its application in their control.

II. METHODOLOGY OF FINDING DYNAMIC THERMAL SYSTEM QUALITY CHARACTERISTICS AND BUILDING ITS FUZZY MODEL

Consider a system with a single output and p inputs in the form

$$y = f(\mathbf{x}, \mathbf{u}) \quad (1)$$

Consider further that it is a stable system, i.e. that to each constant input vector \mathbf{u} and initial state \mathbf{x}_0 there relates a stabilized state of output y . In case of a change in the input vector \mathbf{u} from one constant value \mathbf{u}_1 to another constant value \mathbf{u}_2 (or, in non-linear systems, also taking into account the initial state of vector \mathbf{x}_0) the output value of the system will be stabilized following a transition characteristic which can essentially have an aperiodic or oscillating damped character as demonstrated in Fig.1.

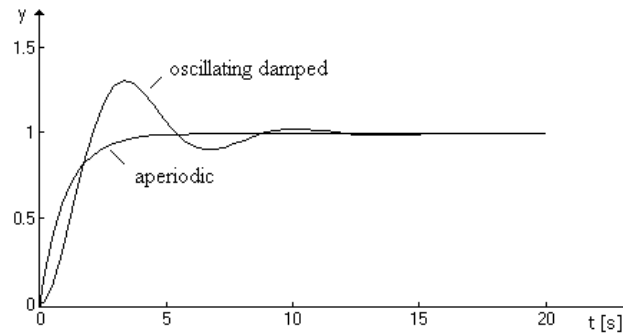


Fig.1 Typical responses of the dynamic system to step of input value

The quality of these responses is expressed in just a few parameters, such as response time, overshoot and oscillating frequency of the characteristic. Based on these simple assumptions we can model the dynamic system by a fuzzy system in accordance with Fig. 2. The dynamic part of the DS will be formed by a simple n -th order linear system (by transfer in the form of a rational fractional function) with variable parameters K . We will choose this subsystem in a way that it will approximate the responses of the system being modeled as closely as possible. The FS subsystem will be a static fuzzy system which will set actual parameter values of the dynamic part K' on basis of inputs u or states x .

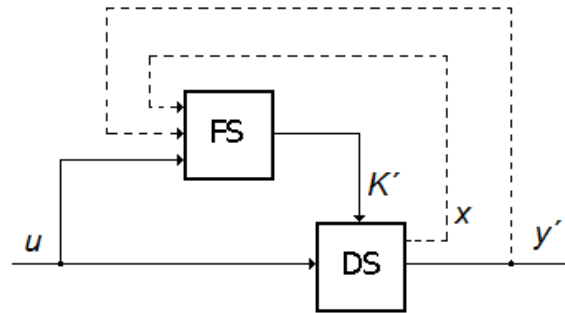


Fig.2 Fuzzy model of dynamic system

A. *Setting up a fuzzy model of a simple thermal system*

A simple thermal system with a single input and single output can most often be described by a first order inertia member with transfer function

$$F(s) = \frac{K_1}{1+sK_2} \tag{2}$$

where K_1 is the amplification of the system and K_2 is the time constant of the system. Let us assume that parameters K_1, K_2 are the following functions of input u :

$$\begin{aligned} K_1 &= f_1(u) = 2 * u^2 + 2 \\ K_2 &= f_2(u) = 0.7 * u + 2.7 \end{aligned} \tag{3}$$

The responses of such system to various steps of input u are shown in Fig. 3.

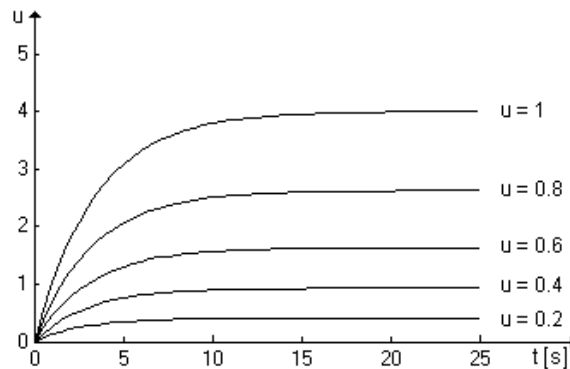


Fig.3 System responses to various steps of input u

For each of these responses it is possible to determine the concrete values of parameters K_1 , K_2 and hence set up a table with a database of rules that describe the static part FS of the system's fuzzy model, representing, in fact, function $k=f(u)$.

Table 1
 Database for FS Rules Development

u	K_1	K_2
0,0	2	2,7
0,2	2,08	2,84
0,4	2,32	2,98
0,6	2,72	3,12
0,8	3,28	3,26
1,0	4	3,4

Characteristic curves for fuzzy estimate of vector K elements on basis of Tab. 1 are shown in Fig. 4.

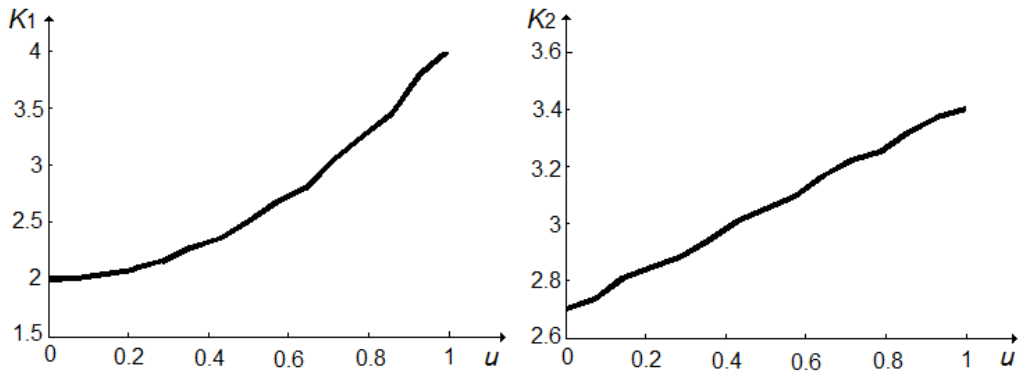


Fig.4 Characteristic curves for fuzzy estimate of system parameters according to Fig. 3

The structure of the fuzzy model for the simple thermal system under consideration is then shown in Fig. 5; Fig. 6 shows a comparison of its responses to various values of input u .

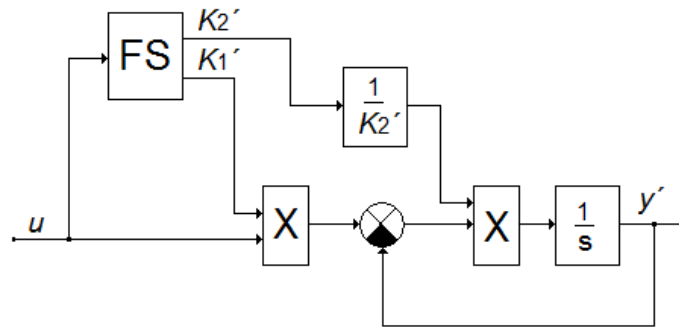


Fig.5 Fuzzy model structure of simple thermal system

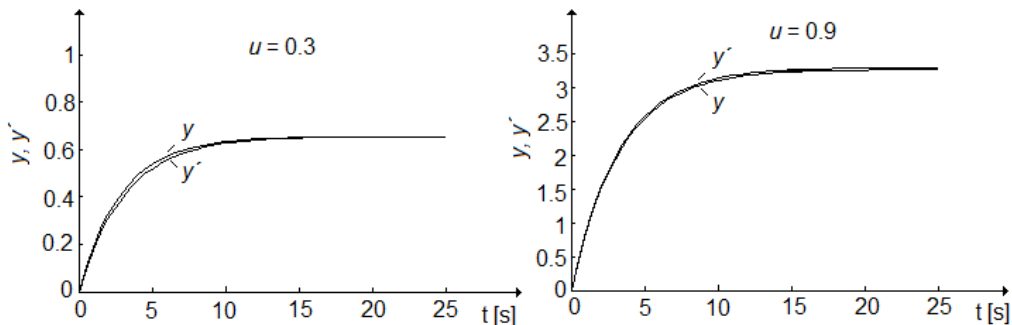


Fig.6 Comparison of responses of thermal system and of its respective fuzzy model

B. Simple thermal system controller design

Let us assume the desired transfer function of enclosed control circuit for the thermal system in Section A above has the form

$$F_u(s) = \frac{1}{1+sT_z} \tag{4}$$

where T_z is the time constant determining the control dynamics. The controller is of the PI type with transfer function in the form

$$F_R(s) = K_p + \frac{K_I}{s} \tag{5}$$

where K_p is the amplification of the proportional part and K_I the amplification of the integrating part of the controller. Hence it is possible to derive relationships that describe the dependence of its parameters K_p and K_I on the actually estimated thermal system parameters in the form:

$$K_I' = \frac{1}{T_z K_1'} \tag{6}$$

$$K_p' = K_1 K_2' \tag{6}$$

where K_1' and K_2' are fuzzy estimated actual parameters of the controlled thermal system. Fig. 7 shows a block diagram of the enclosed control circuit and Fig. 8 shows the responses of the system for selected desired values w and $T_z=0.5$.

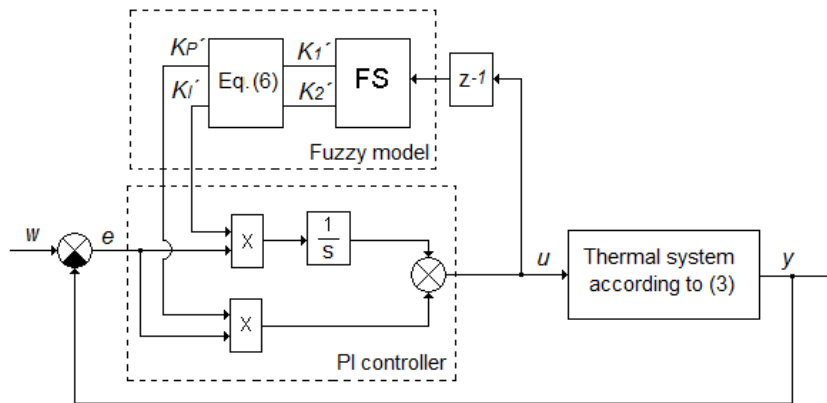


Fig.7 Block diagram showing enclosed control circuit of thermal system

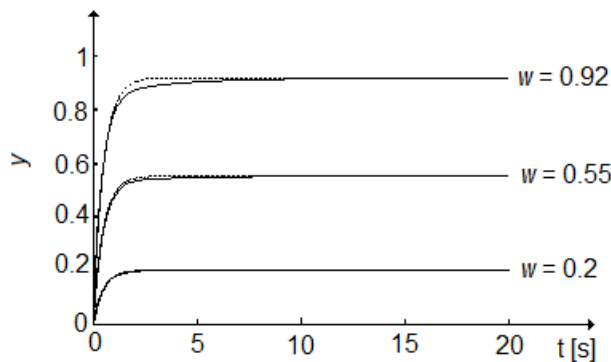


Fig.8 Temperature control dynamics for various desired values of w

Note 1: For purpose of simplification, in the given examples we consider all values of the control circuit normalized into the range $< -1,1 >$, e.g. for example $y=T/T_N$ etc., where T indicates temperature and T_N its nominal value.

Note 2: For comparison, the broken lines in Fig. 8 show the ideal courses of output temperature.

C. Fuzzy modeling and control of a multidimensional thermal system

Let us further assume a thermal system consisting of two heating zones, each of them having its dedicated heating coil and dedicated temperature measurement, with the two zones strongly influencing one another. For purpose of simulation, from the viewpoint of the 1st zone the structure of the system with two inputs and two outputs can be simply illustrated as shown in Fig. 9.

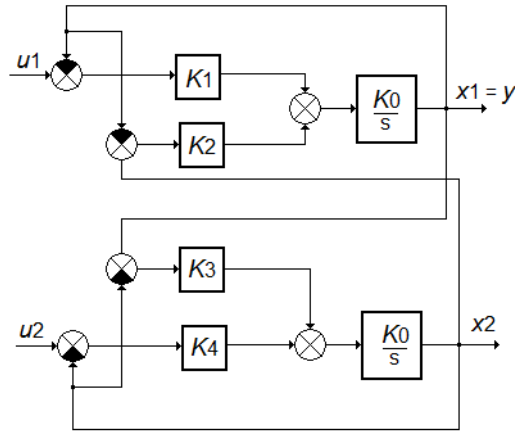


Fig.9 Structure of a two-zone thermal system

Let us further assume that for selected pairs of inputs of vector \underline{u} the following temperature courses have been measured for the 1st zone (Fig. 10):

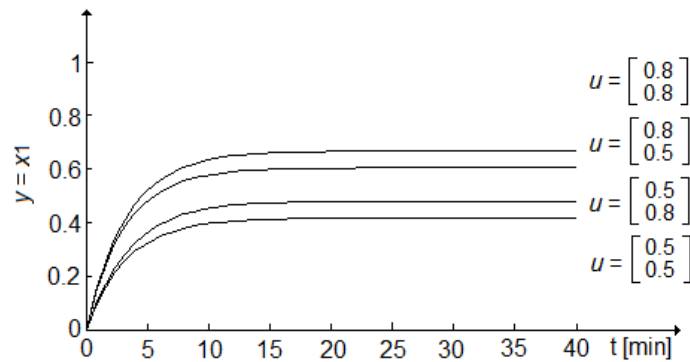


Fig.10 Responses of the dynamic system to various values of input u

In terms of dynamics, a system with the above responses can be approximated by a first order system with transfer function according to relationship (2).

The values of parameters K , T of the dynamic part of the DS for the above responses are shown in Tab. 2 below.

Table 2
 Database for FS Rules Development

u_1	u_2	K	T
0,5	0,5	0,416	3,33
0,5	0,80	0,479	3,59
0,8	0,5	0,603	3,13
0,8	0,8	0,666	3,33

The rules for the FS part of the fuzzy model result directly from this table. A graphic presentation of the fuzzy model areas is given in Fig. 11. The response of this model to step $u=[0.65, 0.65]$ is presented in Fig. 12.

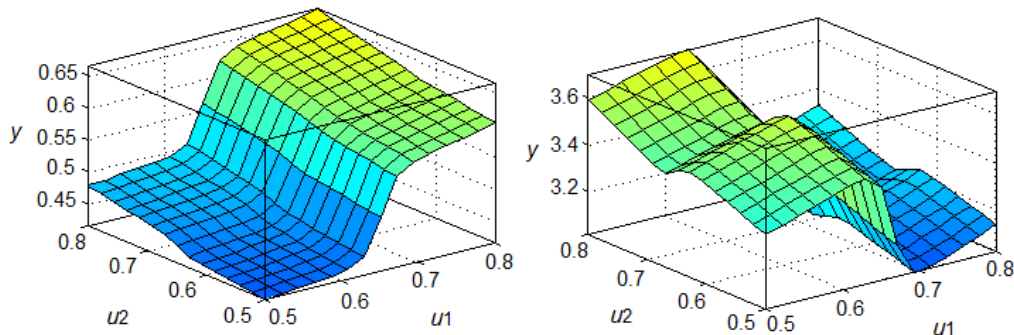
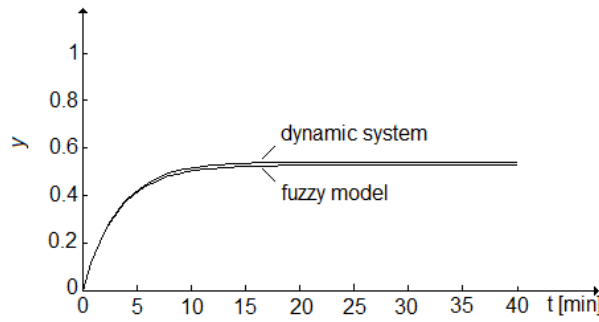


Fig.11 Characteristic areas of the fuzzy model for a two-zone thermal system



Obr.12 Comparison of responses of the dynamic system and its fuzzy model

The 1st zone temperature PI controller will be designed similarly as in case of the simple system in Example 2. The resulting control dynamics, assuming that $u_2 = 0.5$ (constant value of the second input), is shown in Fig.13. Exactly the same procedure may be applied for the second, or in general for any further zone.

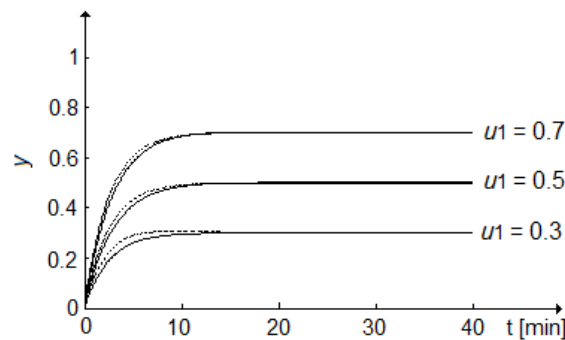


Fig.13 Control dynamics of the 1st zone in a multizone thermal system

III. VERIFICATION OF RESULTS ON A PHYSICAL MODEL OF AN INJECTION MOULDING MACHINE THERMAL SYSTEM

In injection moulding machines, the heating of the material is executed gradually in several heating zones, and one of the decisive quality parameters influencing the final product is the exact temperature of the material at the moment of its injection into the mold (the required temperature tolerance is usually less than 1°C). At the Department of Electrical Engineering and Mechatronics in Technical University of Košice a physical model of a three-zone thermal system was built, as illustrated in Fig. 14, and it was connected to the control automat PP41 made by B&R.

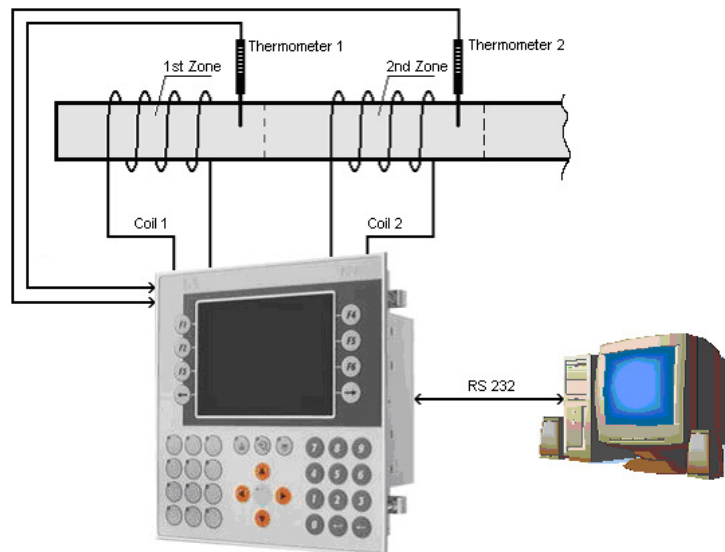


Fig.14 Principle of a physical model of an injection moulding machine thermal system

The model was used for measuring responses of heating and cooling of the individual zones at all combinations of relative input values on the range $\langle 0, 0.1, 0.2, 0.3, 0.4 \rangle$, i.e. altogether

16 measurements for a two-zone system, from which values of K_1 , K_2 for the database of fuzzy estimate of the dynamics of the individual zones at various temperatures of heating coils were determined. From this database a Sugeno type fuzzy model of each thermal zone was built. These fuzzy models estimated and set the PI controller parameters for each zone, exactly as shown in Fig. 7. Both the fuzzy models and the PI controllers were implemented in the B&R control automat.

The results were first verified in the control of one zone. Figure 15 shows the temperature course at desired heating time setting of 8 minutes and heating temperature 50°C. Step change of the desired zone temperature from one operating point to the next is illustrated in Fig. 16.

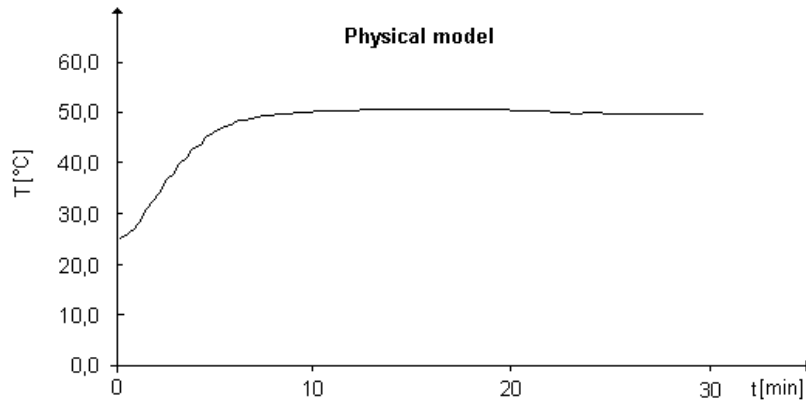


Fig.15 Heating of one zone during desired time of 8 min and to desired temperature of 50°C

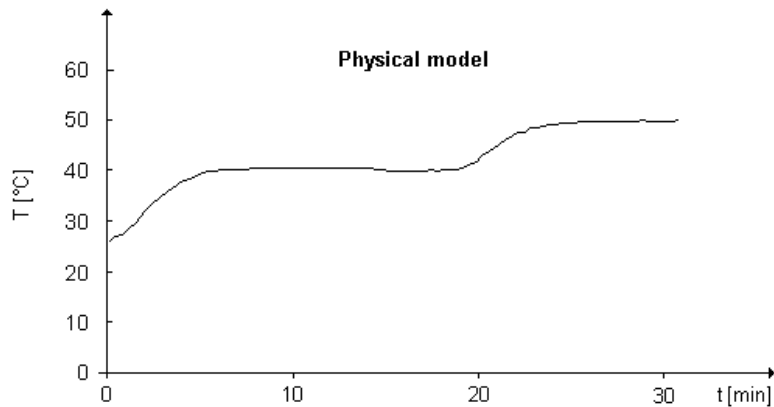


Fig.16 Transition from operating point 40 °C to operating point 50 °C

Figure 17 shows the heating up of a two-zone thermal system over desired time of 4 min to zone temperatures $T_{1z}=35$ °C and $T_{2z}=40$ °C respectively. The heating up and change of operating point of the 1st zone are shown in Fig. 18.

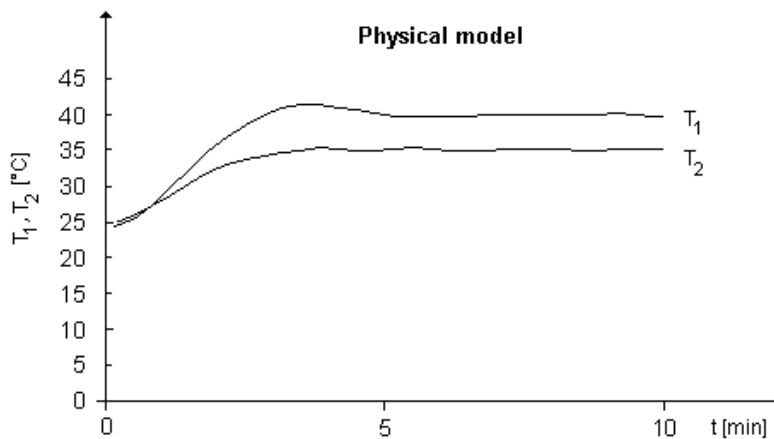


Fig.17 Heating up of two-zone system

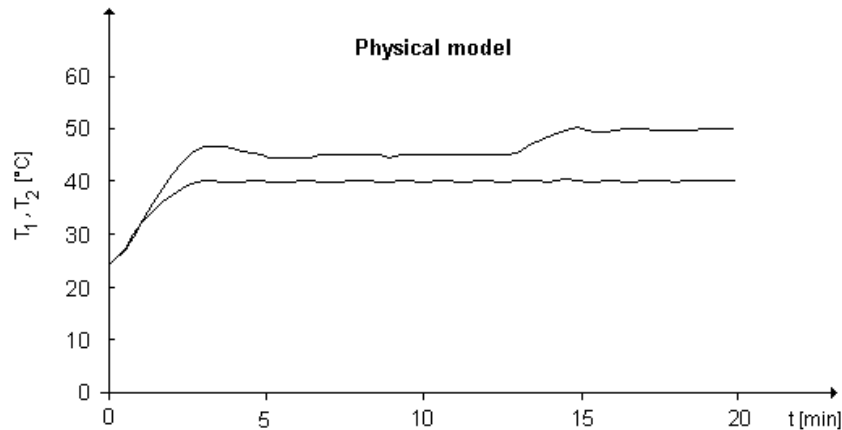


Fig.18 Heating up and change of operating point of two-zone system

IV. CONCLUSION

The paper presents a possible procedure of building a relatively universal dynamic system fuzzy model having a firmly defined DS dynamic part and covering the entire working area of the DS, which does not require the complicated procedure of determining rules for the static fuzzy part of the FS model. This approximation enables the design of simple robust control structures suitable for systems with inaccurately defined parameters and possibly also structure.

The quality of the proposed control structures was verified through simulation in examples, by measurements carried out on a physical model of a multi-zone thermal system, and finally by practical implementation in a completely new control system of an injection moulding machine (type CS manufactured by CASSPOS) built on basis of control automat by B&R. This control practically meets the demands regarding the quality of thermal control on the range of 1 °C and substantially contributes to the increase of quality of the final products.

ACKNOWLEDGMENT

The authors wish to thank the project VEGA 1/0187/18 for its support.

REFERENCES

- [1] P. Fedor, D. Perduková, "Model Based Fuzzy Control Applied to a Real Nonlinear Mechanical System," *Iranian Journal of Science and Technology, Transactions of Mechanical Engineering*, Volume 40, Number 2, 2016, pp. 113-124. DOI 10.1007/S40997-016-0005-9.
- [2] Y. Bai, Q. Zhuang, Z.S. Roth, "Fuzzy logic control to suppress noises and coupling effects in a laser tracking system," *IEEE Transactions on Fuzzy Systems*, Vol. 13, No. 1, 2005, pp.113-121.
- [3] S.A. Mousavi, M. Hashemipour, M. Sadeghi, J.S. Petrofsky, M.A. Prowse, "A fuzzy logic control system for the rotary dental instruments," *Iranian Journal of Science & Technology, Transaction B: Engineering*, Vol. 34, No. B5, 2010, pp 539-551.
- [4] G. Feng, "A survey on analysis nad design of model-based fuzzy control systems," *IEEE Transactions on Fuzzy Systems*, Vol. 14, No. 5, 2006, pp. 676-697.
- [5] J. Amaral da Silva, G. Luiz de Oliveira Serra, :A new methodology for fuzzy model based robust control for nonlinear systems,: in: *Proceeding of IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, IEEE, 2012, pp. 1-8. DOI: 10.1109/FUZZ-IEEE.2012.6251329.
- [6] X. Liu, Z. Xioung, L. Chen and Z. Zhu, "A new Takagi-Sugeno fuzzy approach of process modeling and fault detection," in: *35th Chinese Control Conference – CCC*, IEEE, 2016, 7126–7130. DOI: 10.1109/ChiCC.2016.7554483.
- [7] W. Tang, Z. Rui, H. Jiang and H. Gao, "Coagulation control fuzzy modeling based on cluster analysis and optimization technique," in: *35th Chinese Control Conference (CCC)*, IEEE, 2016, 2154-2159. DOI: 10.1109/ChiCC.2016.7553687.
- [8] L. Yan, Y.M. Ma, „Modeling fuzzy information in fuzzy extended entity-relationship model and fuzzy relational databases,” *Journal of Intelligent & Fuzzy Systems* 27(4) 2014, 1881-1896. DOI: 10.3233/IFS-141154.
- [9] Ch. Wu, J. Wang, H. Li and H. Liang, "Fuzzy-model-based control for nonlinear networked systems with random packet losses," in: *5th International Conference on Information Science and Technology (ICIST)*, IEEE, 2015, 455-460. DOI: 10.1109/ICIST.2015.7289015.

Detekcia kritických stavov pomocou analýzy sentimentu

¹Kristína MACHOVÁ, ²Michal JANOK

^{1,2} Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹kristina.machova@tuke.sk, ²michal.janok@student.tuke.sk

Abstrakt — Duševné poruchy sú jedným z negatívnych fenoménov súčasnosti. Tento stav nepriamo zapríčinila dnešná infromatická doba. Pokrok vo vede, inováciách a globálna digitalizácia ukazujú aj svoju odvrátenú tvár. Táto práca sa sústreďuje na analýzu takzvaných anonymných seba esejí, na detekciu kritických stavov a určovanie diagnózy respondenta. Seba eseje sú texty ktoré píše sám používateľ, v ktorých popisuje svoje problémy. Texty týchto príspevkov sú strojovo čitateľné. Model je navrhnutý tak, aby bol schopný diagnostikovať depresívne kritické stavy a podávať respondentom spätnú väzbu, ktorá by poskytla používateľovi odhad jeho diagnózy a nabádala ho k riešeniu problému, čo je spojené s konkrétnymi návrhmi, čo by bolo možné urobiť a na koho by sa dalo obrátiť.

KLúčové slová —kritické stavy, analýza seba-esejí, dolovanie textu, spracovanie prirodzeného jazyka,

I. ÚVOD

Duševné poruchy sú jedným z negatívnych fenoménov súčasnosti. Psychické ťažkosti sú bohužiaľ stále rozšírenejšie a rapidne znižujú kvalitu života obyvateľstva. Prejavovať sa môžu už aj v rannom, adolescentom veku a mladým zabraňujú rozvíjať svoje talenty a schopnosti. Brzdia ich vývin a začlenenie do spoločnosti. Problémy však niekedy zjdu príďaleko a môžu skončiť fatálne. Tento stav nepriamo zapríčinila dnešná doba. Pokrok vo vede, inováciách a globálna digitalizácia ukazujú svoju odvrátenú tvár. Hlad po úspechu, neustále porovnávanie sa a neobmedzene možnosti vytvárajú na spoločnosť tlak, ktorý môže byť ťažko únosný. Práve z toho pramenia problémy s koncentráciou, no vyústením môžu byť aj vážnejšie duševné poruchy.

Častý výskyt a nárast psychických problémov v spoločnosti vzbudzujú obavy. Centrá pre kontrolu a prevenciu chorôb (CDC – Centrum for Decease Control) odhadujú, že do roku 2020 bude zrejme depresia druhou najčastejšou príčinou zdravotného postihnutia na svete. Nevenovanie dostatočnej pozornosti problému môže viesť u postihnutých k užívaniu návykových látok, násiliu, v najhoršom prípade dokonca až k samovražde. Väčšina ľudí s duševnými poruchami je často nediagnostikovaná a neliečená. Je preto nevyhnutné vyvíjať snahu o posilnenie informovanosti verejnosti o tomto probléme a snažiť sa o včasnú diagnostiku. Táto práca má ambíciu stať sa jedným z prostriedkov na podporu včasnej diagnostiky kritických stavov. Mohla by sa stať nástrojom pre uľahčenie diagnostiky kritických psychických stavov pacientmi a taktiež psychológmi. Okrem toho by mohla plniť funkciu informačnú a mobilizačnú.

Zo širokej škály psychických ochorení ma táto práca za cieľ diagnostiku depresie. Podľa svetovej zdravotníckej organizácie (WHO – World Health Organization) [1] trpí celosvetovo 350 miliónov ľudí depresiami. Depresie postihujú všetky vrstvy obyvateľstva. Ťažko sa z nich hľadá východisko. Najzraniteľnejší sú mladí adolescenti, ľudia na vrchole produktivity a seniori. Pri pohľade na čísla počtu pokusov o samovraždu behá mráz po chrbte. Mojim cieľom je hľadať spôsob ako tento trend utlmiť aj za pomoci technológií a analýzy textu. Je potrebné zdôrazniť, že osobné lekárske vyšetrenie špecialistom je stále najlepšou a najpresnejšou formou pomoci a rozpoznania problému. Avšak v našej krajine je návšteva špecialistu psychológa alebo psychiatra tabuizovaná a považovaná za hanbu. Pre plnohodnotný a produktívny život je psychické zdravie nevyhnutné a túto tému už nepovažujú za tabu v mnohých iných krajinách. Alternatívne domáce vyšetrenie analýzou písaného textu, prinesie respondentovi výhodu diskretnosti.

V tejto práci sa snažíme analyzovať takzvané anonymné seba eseje, detegovať kritické stavy a určovať diagnózu respondenta. Seba eseje sú texty ktoré píše samotný používateľ o svojich

problémoch a zároveň sú strojovo čitateľné. Používateľ v nich opisuje svoje pocity, nálady a situácie ktorými práve v živote prechádza. Niekedy sa jedná dokonca o reporty vlastnej osobnosti. Môžu mať formu škálového testu dostupného na internete. Práca sa opiera hlavne o analýzu sentimentu a je inšpirovaná aj metódami strojového učenia. Výsledkom bude model, ktorý bude diagnostikovať depresívne kritické stavy a podávať respondentom spätnú väzbu vo forme doporučení a nabádania k riešeniu problému. V žiadnom prípade sa nejedná o náhradu tradičnej psychológie ale o využitie dostupnej softvérovej kapacity k podpore rozhodovania a pomoci pri diagnostike a nastolení správnej liečby kritických stavov.

II. KRITICKÉ STAVY A DEPRESIA

Aliancia duševného zdravia uvádza, že duševná choroba je stav, ktorý ovplyvňuje myslenie, pocity aj náladu človeka a ovplyvňuje to aj jeho život na dennej báze. Najrozšírenejšou duševnou chorobou je depresia. Depresia je duševná porucha, ktorá postihuje viac ako 350 miliónov ľudí po celom svete. Jednotlivec ktorý je ňou postihnutý, žije život s veľkými ťažkosťami. Toto stále častejšie postihnutie psychiky jedinca býva často podceňované. K prejavom patria sklúčenosť, smútok, nedostatok motivácie, beznádej, strata záujmu o okolie a mnoho ďalších. Depresia spôsobuje aj straty koncentrácie a poruchy pamäte. Ovplyvňuje to ako sa človek cíti, premýšľa a správa sa. Vážne ovplyvňuje pracovný a partnerský život. Môže viesť k vážnym emocionálnych ale aj fyzickým problémom. Neliečená depresia môže viesť k tragickým koncom. Najväčšie riziko prináša v adolescentnom veku, na vrchole produktívneho veku a u seniorov.

Pre ďalší výskum je v prvom rade potrebné roztriediť depresie do základných skupín. V tejto práci je používané rozdelenie depresívnych stavov na základe príčiny a okolností ich vzniku, ktoré zaviedol nemecký psychológ Kielholz [2].

A. Endogénna depresia

Typickým znakom endogénnej depresie je to, že sa vyskytuje bez rozpoznateľnej vonkajšej príčiny. Trpiaci nemá žiadne fyzické príčiny, ako sú choroby. Nepôsobia na neho dokonca ani faktory ako odlúčenie, rozvod, šikanovanie alebo iný spúšťač depresie. Neexistujú teda žiadne významné udalosti v živote postihnutého ako sú straty príbuzných alebo a napriek tomu trpia postihnutí silnou depresiou. Táto depresia je sprevádzaná silnými výkyvmi nálady. Môžu ju mať za následok, metabolické ochorenia mozgových buniek, chyby látkových premien v mozgu ako aj nedostatok hormónov, hlavne norepinefrínu alebo serotonínu. Samozrejme, značne prispievajú aj genetické faktory.

Endogénna depresia začína veľmi pozvoľne a často môže byť prehliadnuteľná okolím. Postihnutý prechádza procesom, kedy sa jeho podvedomie stáva smutnejším. Tento typ depresie je nebezpečný aj vďaka tomu, že sa s ňím spájajú aj riziká samovraždy. Aj keď sa trpiaci po čase sťažuje na mnohé problémy so stravou a výkyvy nálady, je veľmi ťažké nájsť patologické nálezy a diagnostikovať depresiou. Preto ide o veľmi špecifický typ depresie. Duševné príznaky sú nasledovné: smutná nálada a bezradnosť, extrémne zmeny nálady, extrémne výkyvy nálad, energickosť sa strieda s nezáujmom, každodenná únava, ťažké vstávanie, nedostatok sily, slabosť, vnútorný nepokoj, nervozita a napätie, úzkosť, citlivosť na kritiku, precitlivenosť, podráždenosť a agresivita, poruchy pozornosti a koncentrácie, problémy vo vzťahoch, pokles medziľudských kontaktov a tendencia k izolácii a existuje aj dôvodne riziko samovraždy.

B. Psychogénna depresia

Tento typ depresie je pre laika azda najpochopteľnejší. Psychogénna depresia je úzko spätá so životnými udalosťami. Každý človek ma inú odolnosť a schopnosť vysporiadať sa s ranami osudu. Psychogénnu depresiou môžeme nazvať aj neprimeranou, no do určitej miery pochopiteľnou reakciou na udalosti ako sú smrť v rodine, strata blízkeho, strata partnera, nehody, tragédie, rozvod, odlúčenie, strata zamestnania či slabé ohodnotenie a pocity menejcnosti. Ďalšími, pre potreby slovníka veľmi dôležitými príčinami sú ochorenia blízkych, závislosti na hazarde a alkohole, rôzne životné sklamanie, traumy, zážitky a šoky. Taktiež aj nedostatok získanej sebaúcty a sebavedomia v detstve, sa v dospelosti môže prejavovať práve takouto formou. V poslednej dobe sa veľká pozornosť venuje aj traumatizujúcim udalostiam ako je fyzické násilie alebo sexuálne zneužívanie, ktoré je bezpochyby veľkým predpokladom získania nielen depresie z triedy psychogénnych ale aj iných duševných chorôb.

C. Somatogénna depresia

Tento typ depresie je spôsobený fyzickou chorobou. Je paradoxné, že príčina nie je psychologická. Teda príčinou nie je pacientova vedomosť o chorobe ale fyzicky proces samotnej

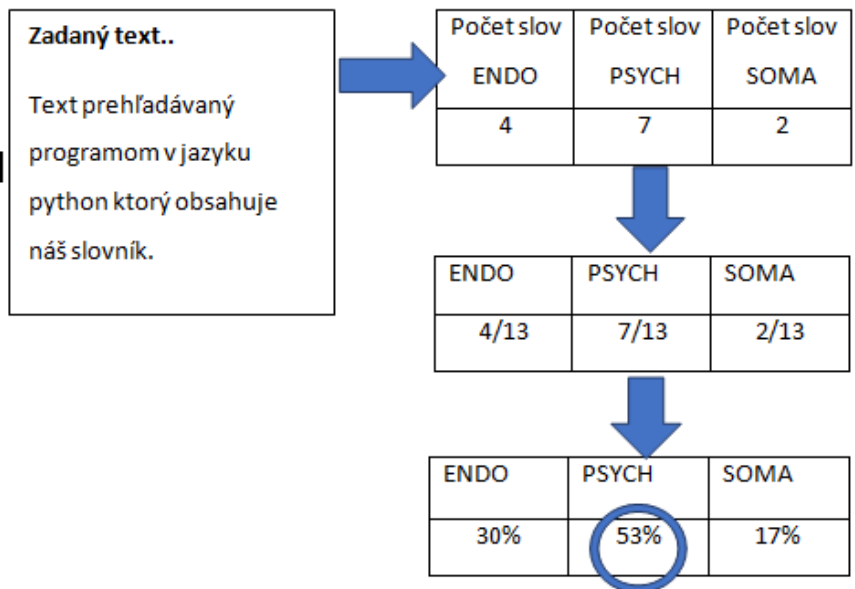
choroby v tele. Choroba pôsobí na látkové premeny v mozgu a ovplyvňujú mozgové centrá ktoré sú zodpovedné za reguláciu nálady. Príkladmi ochorení, ktoré môžu viesť k takejto depresii, sú napríklad hypofunkcia štítnej žľazy, ktorá kvôli nedostatku hormónov štítnej žľazy spôsobuje stav podobný depresii, alebo nádor v prednej časti mozgu, ďalej sa jedná o pacientov postihnutých cukrovkou (diabetes), pacientov trpiacich na chronické zápaly, ochorenia ako Parkinson, Malignóm, tuberkulóza, mononukleóza. Často je sprievodným javom pri ochoreniach ako epilepsia, skleróza multiplex, AIDS, encefalitída mnoho ďalších. Depresia s týmito príčinami nie je tak častá. Pacienti s touto depresiou často trávajú veľa času v nemocnici. Pre ochorenia a časovú náročnosť liečby majú problémy s prácou alebo štúdiom.

III. NÁVRH PRÍSTUPU K DETEKCIÍ KRITICKÝCH STAVOV

Naše riešenie problému zvyšujúceho sa trendu duševných ochorení spočíva vo vytvorení diagnostického modelu, ktorý by mal byť pomocou pri určovaní diagnózy z písaného textu. Zároveň bude model informovať respondenta o diagnóze a poradí mu na koho sa obrátiť. Presnú diagnózu môže určiť jedine skúsený odborník, náš navrhnutý model môže byť nápomocný. V rámci vývoja tohto modelu boli vytvorené slovníky špecificky viazané na jednotlivé typy depresie, bola vytvorená aplikácia na prehľadávanie textov samo esejí, bola vytvorená množina depresívnych textov zozbieraných z webu za účelom testovania diagnostiky. Tieto texty boli anotované psychológom.

Model pozostáva zo slovníka, ktorý má pre každú z troch vybraných diagnóz naviazaný určitý počet špecifických slov. Tieto slová sú jedinečné pre jednotlivé diagnózy, nie je medzi nimi prienik. Slová sú úzko spojené s príznakmi jednotlivých diagnóz a náladovými rysmi trpiacich. Keďže sa jedná prácu s hlbším ľudským vedomím požiadali sme o pomoc a odobrenie slovníka psychológom. Slová z tohto manuálne vytvoreného slovníka slúžia ako identifikátory jednotlivých diagnóz v texte.

Ako analyzované texty sme používali seba eseje respondentov z portálov zdravie.sk a iných portálov venujúcich sa duševnému zdraviu. Tieto texty(seba eseje) sú anotované (značkované) odborníkom podľa príslušnosti k jednotlivým typom depresie. Inak povedané, psychológ rozhodol u každého textu podľa príznakov o diagnóze ktorú vykazuje. (ENDO,PSYCH,SOMA). Slová z každej kategórie depresii ktoré sú nájdené pri prehľadávaní textu, sú zaznamenávané. Z počtu nájdených slov pre každú kategóriu bola odvodená pravdepodobnosť jednotlivých diagnóz, čo je ilustrované na Obr. 1.



Obr. 1 Ilustrácia odvádzania konkrétnej diagnózy v rámci spracovania textov a ich porovnávaní so slovníkmi.

A. Tvorba slovníkov

Vytvorenie správneho slovníka bolo kľúčovým pre správne fungovanie aplikácie. Preto sme tvorili slovník pre analýzu sentimentu s odbornou podporou psychológa. Pochopenie pohnútok a duševného stavu trpiacich bolo nevyhnutnou súčasťou rozhovorov, ktoré viedli k tvorbe slovníka. Balík informácií získaný rozhovormi s psychológom bol stále rozširovaný

a pretransformovaný do troch vybraných skupín slov pre rôzne typy depresie. Po ujasnení hraníc medzi jednotlivými triedami došlo k citlivému výberu slov jedinečných pre každú kategóriu. Snaha bola nájsť čo najviac slov a slovných spojení s veľkou diverzitou, ktoré symbolizujú príčinu, okolnosti vypuknutia, duševný stav pacienta a symptómy diagnózy. Nasledujú príklady voľby slov v jednotlivých kategóriách:

Slovník 1: typickou formou Endogénnej depresie je napr. ochorenie s názvom bipolárna porucha. V slovníku sa to prejaví slovami a slovnými spojeniami: výkyv, výkyvy, výkyvmi, nálad, nálada, nálady, kolísanie, kolíše.

Slovník 2: typickou črtou Psychogénnej depresie, je neprimeraná reakcia na smrť, resp. stratu blízkeho. To reprezentujú slová: smrť, odišiel, odišla, navždy, opustil, opustila, zomrel, tragédia.

Slovník 3: Pri Somatogénnej depresii je trpiaci ovplyvnený procesmi v tele, ktoré sa dejú v súvislosti s určitou chorobou, napríklad poruchou štítnej žľazy. Do slovníka sa dostali slová: štítna žľaza, štítnou, tyroxín, hypotyroxinoza, endokrinný, endokrinológ, endokrinológia.

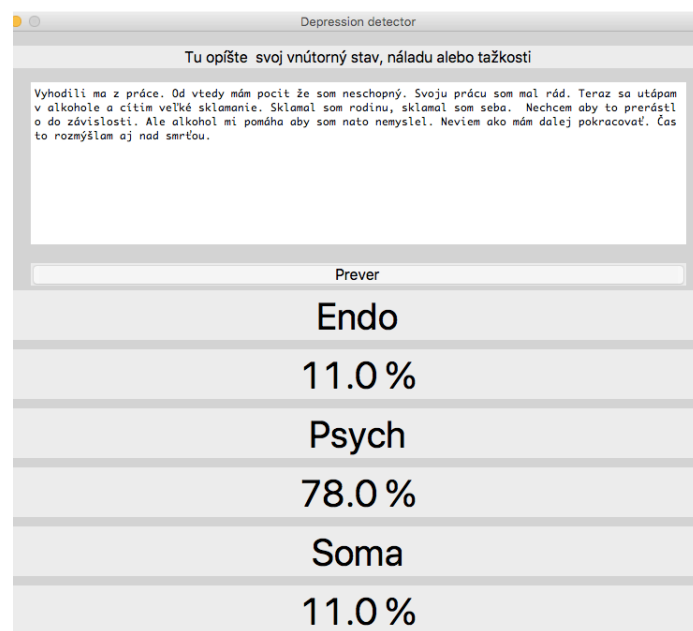
Slovníku boli ďalej upravované použitím Word Count Frequency. V seba esejach sa slová usporiadali zostupne podľa počtu výskytov. Najvyššie umiestnené slová s vysokými frekvenciami sme považovali za relevantné slová a následne ich zaradili aj do slovníka.

Pri predspracovaní sme použili aj N-gramy, ktoré boli vyextrahované zo zozbieraných príspevkov pomocou POS (Part of Speech) tagging [3]. Najčastejšie šlo o bi gramy, teda dvojslovné pomenovania.

Potom sme sa zamerali na mnoho dokumentov zaoberajúcich sa duševným zdravím. Našli sme vedecké slová ktoré sa týkajú rozoberaných kritických stavov. Aj tieto, nie veľmi rozšírené slová boli pridané do slovníka. Taktiež boli použité slová z psychologických testov. Na internete je mnoho prieskumov duševného stavu do ktorých sa dá zapojiť. Jednotlivé otázky sme skúmali s cieľom nájsť pre slovník ďalšie vhodné slová. Takto bol vytvorený slovník o objeme takmer 300 slov týkajúcich sa endogénnej, psychogénnej a somatogénnej depresie.

B. Implementácia

Implementácie bola vytvorená v programe Python. Zvolený bol pre svoju afinitu k analýze textu. Tento programovací jazyk disponuje knižnicou NLTP, ktorá obsahuje množstvo balíkov na prácu s textom. Je tiež vhodná na prácu s prirodzeným jazykom a s prácou s N-gramami. Ako vývojové prostredie bolo použité Pycharm. Aplikácia funguje na princípe rozpoznávania depresívnych stavov z písaného textu. Dokáže analyzovať príspevky z diskusie alebo seba eseje a rozhodnúť či pisateľ príspevku môže trpieť depresiou. Tento diagnostický model sa v budúcnosti bude nachádzať aj na blogoch a webových stránkach ktoré obsahujú rôzne diskusie z oblasti zdravia. Obr. 2 ilustruje činnosť tejto aplikácie, keď model diagnostikoval zo seba eseje respondenta depresiu zo skupiny psychogénnych depresii s vysokou pravdepodobnosťou.



Obr. 2 Ilustrácia odvádzania konkrétnej diagnózy v rámci spracovania textov navrhnutou aplikáciou.

C. Výsledky testov a ich analýza

Na výpočet výsledkov ukazovateľov miery efektívnosti analýzy sentimentu bol použitý nástroj, ktorý pracuje s hodnotami správne pozitívny (True Positive TP), správne negatívny (True Negative TN), chybné pozitívny (False Positive FP) a chybné negatívny (False Negative FN). Uvedené hodnoty boli vnesené do kontingenčnej tabuľky a z nej boli odvodené hodnoty nasledovných mier efektívnosti TPR=Návratnosť, PPV=Presnosť, F1 skóre. Konkrétne výsledky testov podľa jednotlivých typov depresie sú zobrazené v Tabuľke 1.

Tabuľka 1
Výsledky testov implementácie pre rôzne typy depresie

Depresia	Návratnosť	Presnosť	F1 skóre
endogénna	0,880	0,880	0,880
psychogénna	0,780	0,840	0,809
somatogénna	0,895	0,860	0,877

IV. ZÁVER

Článok sa zameriaval na odhaľovanie kritických stavov sprevádzaných najbežnejšími typmi depresie. Implementovaný prístup vo forme webovej aplikácie má relatívne vysokú dôveryhodnosť. Väčšina príspevkov bola správne vyhodnotená a bola určená správna diagnóza. Nesprávne diagnostikované boli len nejasné príspevky alebo tie, ktoré dosiahli pre všetky kategórie rovnaké slovné skóre. V budúcnosti by sme radi zdokonalili tento systém na analýzu kritických stavov použitím analýzy sentimentu so zameraním sa na negatívnu polaritu názorov [4]. Taktiež bolo vhodné zamerať sa prípadný výskyt sarkazmu pomocou pravidiel vytvorených na tento účel v práci [5], ktoré umožňujú zvýšiť presnosť analýzy sentimentu.

Navrhnutý prístup by sa mohol použiť na vytvorenie aplikácie, pre nočný predaj liekov v lekárnach alebo na čerpacích staniaciach. Bolo by zaujímavé používať hlasovú biometriu a metódy konverzie reči na text. Zákazník lekárne by sa v nočných hodinách mohol dostať k liekom. Stačil by k tomu hlasový opis príznakov a potrieb. Aplikácia by mu po analýze požiadaviek vydala liek. Išlo by samozrejme o lieky ktoré je možné kúpiť bez lekárskeho predpisu. Takáto aplikácia by bola vhodnou investíciou pre lekárne, ktoré by lieky predávali aj v noci, bez nákladov na nočné služby personálu.

POĎAKOVANIE

Práca prezentovaná v tomto článku bola podporená Slovenskou agentúrou na podporu vedy a výskumu v rámci projektu APVV-017-0267 "Automatizované rozpoznávanie antisociálneho správania v online komunitách".

LITERATÚRA

- [1] WHO / Fact File. [online] available on http://www.who.int/features/factfiles/global_burden/facts/en/index6.html
- [2] B. Kielholz, Medikal Chemistry. A series of monographs. Vol. 11-X, Academic Press, Inc., New York, USA, 2013, 427 ps.
- [3] E. Brill. A simple rule-based part of speech tagger. In Proceedings of the Third Conference on Applied Natural Language Processing, ANLC '92, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics, pp. 152–155.
- [4] M. Mikula, K. Machová, Spracovanie negácie pre klasifikáciu názorov v slovenskom jazyku. In: Data a znalosti 2015, 1-2.10.2015, Praha, VŠB – TU Ostrava, 2015, 41-45, ISBN 978-80-248-3824-3
- [5] D.G. Maynard, M.A. Greenwood, Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. In: LREC 2014 Proceedings. Language Resources and Evaluation Conference (LREC), 26-31 May 2014, Reykjavik, Iceland [online], 2014, Dostupné na internete: <http://eprints.whiterose.ac.uk/130763/>. ISBN 978-2-9517408-8-4.

Example of the Robotic Receptionist

¹Renát HALUŠKA, ²Peter ŠULAJ, ³Luboš OVSENÍK

^{1,2,3} Department of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹renat.haluska@tuke.sk, ²peter.sulaj@tuke.sk, ³lubos.ovsenik@tuke.sk

Abstract — This paper presents an example of the robotic receptionist, which consists of a humanoid robot and a computer with a webcam. The main part of the project is the face recognition using a webcam connected to the computer and human interaction with robots. In the experiment we evaluated the reliability of the system, which includes correct recognized of human, the time required for human recognizing and correct human interaction with robots.

Keywords — EmguCV, face recognition, Local Binary Patterns Histograms, Nao

I. INTRODUCTION

Today, robots perform more and more human occupation, for example receptionist. In the world of the few robots that perform the function of receptionist. Aiko Chihiro (Fig. 1) is a humanoid robot from Toshiba company. It is dressed in a kimono and welcomes customers at a shopping center in Tokyo. The customer is able to put in a word in Japanese but can be programmed for other languages. It can also identify Japanese sign language. The movement allows 43 motors [1].



Fig. 1 Aiko Chihiro robot [2]

The second example of the robot receptionist (Fig. 2) is looking like a dinosaur at the Hotel Henn in Japan. Its role is to inform about how the customer can check in using the automated baggage trolley, which it is imported to room [3].



Fig. 2 Receptionist robot as dinosaur [3]

II. INTERACTION WITH NAO ROBOT

The main idea is to use the robot Nao as a receptionist, who is able to recognize the person and communicate with him through speech. The interactions works as follows:

1. Face recognition of visitors.
2. Nao greet visitors name.
3. Nao asks visitors: "Looking for someone?" and waiting for an answer yes / no.
4. When a visitor answers „yes“, Nao asks visitors: „Who are you looking for?“.
5. Nao tells room number.
6. When a visitor leaves robot greets him.

The main structure (Fig. 3) of the system consist of face recognition of the visitor, which runs on a personal computer with a connected HD webcam for better recognition. Nao robot that interacts with visitors via speech and communicates with a computer via Wi-Fi using TCP/IP protocol.

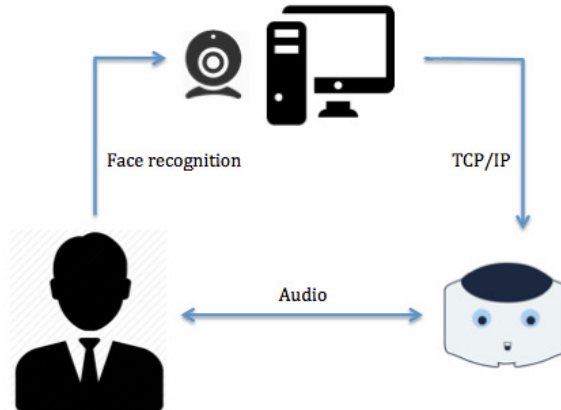


Fig. 3 Structure of the system

It was tested three types of algorithms from EmguCV library: Eigenfaces, Fisherfaces and Local Binary Patterns Histograms. Nao robot that interacts with visitors communicates with a computer via Wi-Fi. The Eigenfaces problem with the image representation high dimensionality. Two-dimensional pxq grayscale images span a $m=pq$ -dimensional vector space, so an image with 100×100 pixels lies in a 10000 -dimensional image space already [4].

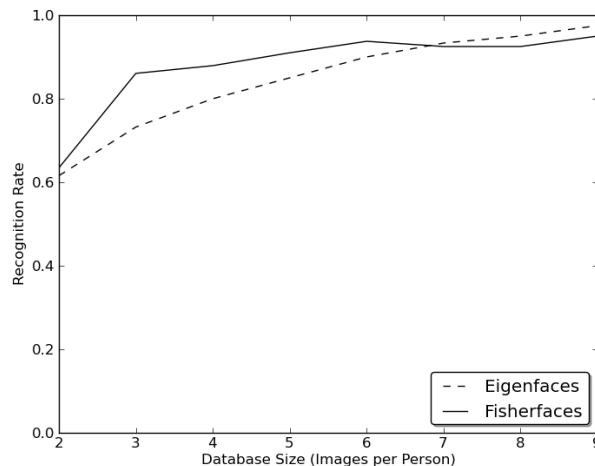


Fig. 4 Graph of recognition rates of the Eigenfaces and Fisherfaces method

The Principal Component Analysis (PCA), which is the core of the Eigenfaces method, finds a linear combination of features that maximizes the total variance in data. While this is clearly a powerful way to represent data, it doesn't consider any classes and so a lot of discriminative information may be lost when throwing components away [4].

It's impossible simply guarantee perfect light settings in images or 10 different images of a person, therefore it was used third algorithm Local Binary Patterns Histograms. The basic idea of Local Binary Patterns is to summarize the local structure in an image by comparing each pixel with its neighborhood. Take a pixel as center and threshold its neighbors against. If the intensity of the center pixel is greater-equal its neighbor, then denote it with 1 and 0 if not [5]. A more formal description of the LBP operator can be given as:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c) \quad (1)$$

with (x_c, y_c) as central pixel with intensity i_c and i_p being the intensity of the neighbor pixel and s is the sign function defined as:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \quad (2)$$

For face recognition purposes, it was necessary to create a face dataset (Fig. 5).



Fig. 5 Example dataset of faces

The function to communicate with the Nao is part of the main program running on a personal computer and is written in C#. The program uses library *System.Net.Sockets*. Aledebaran and was used blocks *custom Python script*. The script uses library *socket*.

It was done 50 attempts to test the reliability of the system which includes correct recognized of human, the time required for human recognizing (time > 4 seconds) and correct human interaction with robots (recognition response from visitors). Great influence on the attempts had the lighting in the building.



Fig. 6 Example of the attempt

III. CONCLUSION

The key idea of the Robotic Receptionist is to replace human for robotic receptionist which can operate nonstop. Disadvantages of the system is wrong face recognition when is bad lightening, overheating servos of the robot after a long time, robot wrong speech recognition of visitor. Advantages of the system are faster face recognition due to non-use Naos system for detection, when the damaged robot system could still work and extensions for storing new people to the database and thus establish a system for the comings and goings of visitors.

ACKNOWLEDGMENT

This work was supported by following research grants: APVV-17-0208 - Resilient mobile networks for content delivery and VEGA 1/0584/20 - Person Monitoring by UWB Sensor Systems Operating in Real Conditions.

REFERENCES

- [1] K. Moskvitch, "Photo Essay," *Engineering & Technology*, vol. 10, no. 6, pp. 54–57, 2015.
- [2] J. McCarthy, "Toshiba's Aiko Chihira humanoid robot debuts as a greeter at Japanese department store," *The Drum*, 15-Aug-2016. [Online]. Available: <https://www.thedrum.com/news/2015/04/21/toshiba-s-aiko-chihira-humanoid-robot-debuts-greeter-japanese-department-store>. [Accessed: 28-May-2020].
- [3] A. Press, "Japan's robot hotel: a dinosaur at reception, a machine for room service," *The Guardian*, 16-Jul-2015. [Online]. Available: <https://www.theguardian.com/world/2015/jul/16/japans-robot-hotel-a-dinosaur-at-reception-a-machine-for-room-service>. [Accessed: 28-May-2020].
- [4] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.
- [5] F. Wang, B. Qi, Y. Xing, and B. Lu, "Research on face detection based on combination of Haar and skin colors features," 2010 3rd International Congress on Image and Signal Processing, 2010.

Example of Hard Switching for Hybrid FSO/RF System

¹Renát HALUŠKA, ²Maroš LAPČÁK, ³Luboš OVSENÍK

^{1,2,3} Department of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

¹renat.haluska@tuke.sk, ²maros.lapcak@student.tuke.sk, ³lubos.ovsenik@tuke.sk

Abstract — This paper is focused on the design and implementation of hard switching for a hybrid FSO / RF system based on the prediction of the received signal strength parameter. Free Space Optics (FSO) fiber less optical communication in combination with a high-frequency RF line represents almost 100% availability. The most important is the indicator of the strength of the received RSSI signal, parameter in the hard switching method of a hybrid FSO / RF system. It is necessary to design right algorithms for the prediction itself with RSSI parameter prediction.

Keywords — FSO, Hard Switching, Hybrid FSO/RF system, RSSI

I. INTRODUCTION

Free Space Optics (FSO) is an optical communication technology in which data is transmitted by the propagation of light in free space, which allows optical connection. FSO works similarly to Optical Fiber Cable (OFC), but the only difference is that optical beams are emitted by air instead of OFC cores, which are glass fibers. The FSO system consists of an optical transmitter and a receiver at both ends, which provides two-way (full-duplex) communication. FSO is a technology that for its proper functioning needs direct visibility of the so-called LOS (from the English Line Of Sight), in which data, voice and video communication is achieved with a maximum data transfer rate of 10 Gb/s full duplex (two-way) connection [1].

The use of lasers is a simple concept similar to optical transmission using fiber optic cables; the only difference is the transmission media. Light passes through air faster than glass, so FSO technology can be classified as optical communication at the speed of light. FSO communication is considered as an alternative to communication systems for long distance radio transmission (LOS) [2].

The theory for FSO systems is essentially the same as the theory of optical fiber transmission. The difference is that the energy beam is collimated and emitted by clean air or space from a source to a destination, and not guided by an optical fiber. If the energy source does not generate a parallel beam enough to cover the required distance, collimation can be performed with lenses [3]. It is visible on the source as infrared energy modulated by the data to be transmitted. At the destination, the beam is captured by a photodetector, the data is extracted from a visible or infrared beam (demodulated) and the resulting signal is amplified and sent to the hardware [4].

II. MEASUREMENT OF PHYSICAL QUANTITIES USING A MONITORING STATION

Communication standards place a high demand on FSO systems throughout the year. The availability and reliability of FSO systems is an overall problem and in solving it is necessary to analyze a sufficient number of auxiliary input physical quantities, which characterize the atmospheric conditions along the entire transmission path in real time. Atmospheric data collection is provided by a weather station and sensors working on a Raspberry Pi 3 microcomputer.

Monitoring the weather conditions between the two heads of FSO systems is an important task in determining and evaluating the reliability and availability of FSO systems. An important condition for securing an optical connection using an FSO line is direct visibility between the receiver and the transmitter. The full-duplex principle applies to this type of communication, so each device is both a transmitter and a receiver.

Visibility measurement is realized in the designed system using a miniOFS sensor (Fig. 1). This sensor measures the amount of water droplets, e.g. fog particles in the air that reduce visibility.



Fig. 1 MiniOFS visibility sensor

Monitoring of the concentration of particles in the fog is performed using a simple optical sensor gp2y1010au. The acquisition of wind speed data is important in the process of analyzing the availability of FSO systems, especially in identifying the occurrence of optical turbulence in the vicinity of the transmission path of the FSO system between the transmitter and receiver. Air temperature was measured using the DS18B20 sensor, which is shown in Fig. 2

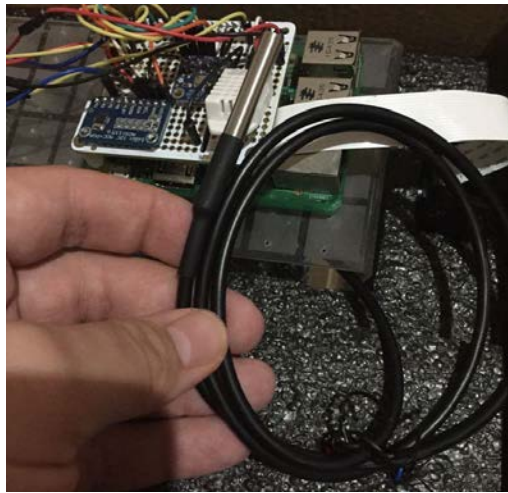


Fig. 2 DS18B20 temperature sensor

III. PREDICTION OF RSSI PARAMETER

Real-time RSSI prediction and diagnostics are very difficult to implement and inefficient in terms of application. Since there are delays in measuring individual parameters and it takes some time to record them in the database, it is more objective to predict the RSSI parameter a few minutes in advance. The prediction of the RSSI parameter should prevent unnecessary switching of the FSO line to the backup RF line and back, or the persistence of the system in communication via the RF line. An important task is therefore the optimal design of the training matrix P and the correct choice of the type of training model, which will reliably describe most scenarios of the case matrix.

The input matrix for the training model is also formed by other data, not only data in logical time sequence. The currently measured data is neglected due to the time delay. The data used are suitably time-shifted backwards by $-x_{min}$. When pre-processing the input data matrix, the optimal selection of the number of offsets is important. It should be noted that with each shift, the input matrix expands by 6 new columns, which increases the time of calculation of the model and also the time required for prediction.

The Python programming language was used to create a hard switching design for a hybrid FSO / RF system. In the first step it was necessary to load the library for importing data from the database, using the command:

```
import xlr.d.
```

Next, the path to the location of the database on the computer was chosen, and then the data was retrieved from the database using the command:


```
wb = xlrd.open_workbook(loc)
sheet = wb.sheet_by_index(0)
```

In this case, a comparison of the previous value with the currently measured value is performed, while the individual states are defined:

```
lastrow = sheet.nrows - 1
prelastrow = sheet.nrows - 2
last = sheet.cell_value(lastrow, 2)
prelast = sheet.cell_value(prelastrow, 2)
```

Since it is a hybrid system working with FSO and RF lines, it was necessary to determine the threshold value of the indicator of the received optical power. It was fixed at a value of - 25 dBm.

```
rss_i = -25
state = ""
```

This proposal is a comparison of values measured with respect to the current time, in the phase of hard switching there is a comparison of states, the last and the previous one. Since it is a hybrid system, there are two states to choose from, either FSO or RF. With the correctly selected threshold limit of the RSSI parameter, it is possible to determine what state will follow for newly acquired data from the database of measured values.

```
if (prelast <= rss_i and last <=rss_i):
    state = "fso"
print ("State is unchanged. State is: " ,state)
elif (prelast <= rss_i and last >rss_i):
    state = "rf"
print ("State is changed. Switch to: " ,state)
elif (prelast > rss_i and last <=rss_i):
    state = "fso"
print ("State is changed. Switch to: " ,state)
else:
    state = "rf"
print ("State is unchanged. State is: " ,state)
```

After this step, it is possible to decide whether the communication will be realized using FSO or RF line.

IV. CONCLUSION

The main disadvantage of fiberless optical communication is the influence of fog, while the connection between the two heads is lost. The second chapter described radio frequency (RF) systems, which in this case form a secondary line for a hybrid FSO / RF system. Unlike FSO lines, these systems are most prone to precipitation, i.e. rain and snow. An important element in the hard switching process is the received signal strength indicator (RSSI).

It is important to note that modifying and improving the case matrices for binary and multistate classification, as well as reworking the regression prediction, would affect the more reliable switching of the hybrid FSO / RF system.

ACKNOWLEDGMENT

This work was supported by following research grants: APVV-17-0208 - Resilient mobile networks for content delivery and VEGA 1/0584/20 - Person Monitoring by UWB Sensor Systems Operating in Real Conditions.

REFERENCES

- [1] J. Mikołajczyk, et al. "Analysis of free-space optics development." Metrology and Measurement Systems 24.4 (2017).
- [2] A. Malik and P. Singh, "Free Space Optics: Current Applications and Future Challenges," International Journal of Optics, 17 November 2015.
- [3] A. G. Alkholidi and K. S. Altowij, "Free Space Optical Communications — Theory and Practices," Sanaa, 2014.
- [4] M. Rouse, "whatis.techtarget.com," Marec 2011. [Online]. Available: <https://whatis.techtarget.com/definition/free-space-optics-FSO>.

Bezpečnosť a riadenie inteligentných sietí v elektroenergetike

¹Jozef HUMENÍK

¹ Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹jozef.humenik@student.tuke.sk

Abstract — Tento článok sa venuje výskumu problémov bezpečnosti a riadenia inteligentných sietí v oblasti elektroenergetiky. Zistilo sa, že s rozvojom inteligentných sietí v oblasti elektroenergetiky a ďalších odvetviach národného hospodárstva sa počas ich integrácie vytvára nástroj na riadenie hospodárstva krajiny na kvalitatívne vyššej úrovni ako tomu bolo v minulosti a je tomu teraz na území Ruskej federácie.

Keywords — bezpečnosť, inteligentné siete, manažment, zníženie rizika

I. ÚVOD DO INTELIGENTNÝCH SIETÍ

V období posledných rokov pre možné vnímať nárast inštalácie inteligentných elektromerov v distribučných oblastiach po celom svete. V Spojených štátoch amerických a západnej Európe sa už teraz využíva veľké množstvo rôznych snímačov, čidiel a zariadení vo veľkých, ale ešte vo väčšom množstve menších projektov v oblasti prenosu a distribúcie elektrickej energie.

Objem finančných investícií do rozvoja inteligentných sietí v období rokov 2009 - 2013 v niektorých regiónoch sveta dosahuje do výšky niekoľkých desiatok miliárd amerických dolárov. Ako sú napríklad:

EÚ - 7 miliárd dolárov na podporu celoeurópskej infraštruktúry vo forme energetického balíčka, kde došlo k 80 % pokrytiu distribučnej oblasti inteligentnými elektromermi);

USA - 19 miliárd dolárov, z toho 14,5 miliárd dolárov na prenos a distribúciu elektrickej energie z obnoviteľných zdrojov energie a 4,5 miliardy dolárov na vytvorenie inteligentnej siete);

Čína - 70 miliárd dolárov na rozvoj alternatívnych zdrojov energie, rozšírenie siete vedení vysokého napätia (VN), inštalácia nových zariadení a vytvorenie rozsiahleho monitorovacieho systému);

India - 10 miliárd dolárov, z toho 8 miliárd dolárov na inštaláciu nových zariadení a 2 miliardy dolárov na nákup softvéru a systémov automatizácie);

Veľká Británia - 3 miliardy dolárov, z toho 500 miliónov dolárov sa investovalo do štyroch projektov inteligentných miest, na legislatívne zmeny v oblasti nárastu energetickej efektívnosti a spoľahlivosť siete);

Austrália - 1 miliarda dolárov, z toho 100 miliónov dolárov sa investovalo do dvoch projektov inteligentných miest, kde v roku 2013 došlo k úplnému pokrytiu distribučnej siete inteligentnými elektromermi v štáte Victoria;

Kanada - 0,5 miliardy dolárov sa investovalo do inteligentnej infraštruktúry distribučnej siete a integrácie obnoviteľných zdrojov energie;

Južná Kórea - 0,3 miliardy dolárov bolo vyhradených do inštalácie inteligentných elektromerov a vytvorenia testovacieho centra. [1]

II. BEZPEČNOSŤ A RIADENIE INTELIGENTNÝCH SIETÍ

Inteligentné siete je potrebné vnímať ako časť celkovej elektrizačnej sústavy. V zásade platí, že inteligentné siete sa zakladajú na systémoch merania a zohľadnenia spotreby elektrickej energie, ktorá je prepojená so systémami na riadenie spotreby elektrickej energie. V inteligentných sieťach by sa mali do maximálnej možnej miery využívať lokálne zdroje energie s použitím rôznych prvkov na akumuláciu elektrickej energie (napr. batérie).

Inteligentné siete by mali byť napojené na riadiace systémy, aby jednotlivé distribučné spoločnosti boli schopné zabezpečiť dodávku elektrickej energie v požadovanom množstve a kvalite. Dispečingy by mali mať vypracované tzv. núdzové plány pre prípad výnimočných situácií.

Ako príklad sa uvádza Ruska federácia, v ktorej prezident Vladimir Putin a vláda RF poverili ministerstvá, štátne služby, a taktiež spoločnosti z oblasti energetiky implementáciou vypracovania a zrealizovania projektov k prechodu z klasických distribučných sietí na inteligentné.

Minister energetiky Ruskej federácie S. I. Šmatko vo svojom prejave na zasadnutí za okrúhlym stolom na tému inteligentných sietí, inteligentnej energetiky a inteligentnej ekonomiky v rámci oficiálneho programu IV. Petrohradského medzinárodného ekonomického fóra, ktorý sa konal ešte v júni roku 2010 poznamenal, že modernizácia národného hospodárstva v oblasti elektroenergetiky zahŕňa tieto úlohy: inovatívny charakter prechodu na moderné technológie 21. storočia a vytvorenie programu modernizácie prostredníctvom implementácie inteligentných sietí. Asistent prezidenta RF A.V. Dvorkovich tiež zdôraznil, že rozvoj inteligentných sietí je jedným z prioritných projektov Komisie pre modernizáciu a technologický rozvoj národného hospodárstva RF, na čele ktorého stojí prezident Ruskej federácie. [4]

Zároveň platí, že v tom čase ani v zahraničí, ani v Rusku neexistuje jednotná komplexná koncepcia pre vytvorenie postupov na zlepšovanie efektívnosti riadiacich technológií určených pre oblasť inteligentných sietí. Popri tom v mnohých vedeckých a firemných výskumoch a rôznych článkoch sa veľká pozornosť udeľuje osobným (čiastkovým) otázkam zvýšenia efektívnosti riadenia vo viacerých oblastiach predmetnej činnosti, akými sú napríklad: správa elektronických dokumentov, rôzne prostriedky pre zber, spracovanie a prenos informácií, implementáciu CPM (Corporate Performance Management), BPM (Business Performance Management) a EPM (Enterprise Performance Management), použitie geografických informačných systémov (GIS), RFID technológií, internetu vecí (IoT) a podobne.

S tým úzko súvisia početné nariadenia, smernice a projekty „Elektronickej vlády“, ktoré sa realizujú na federálnej, ale aj na regionálnej úrovni. V dnešnej dobe je použitie elektronických informačných systémov pre riadenie založené na budovaní a fungovaní organizačnej štruktúry a riadiacich systémoch v rámci inteligentných prostredí. Práve tu je prechod procesov a inteligentných postupov riadenia, v ktorých sa udržiava nová kvalita riadenia akéhokoľvek druhu riadenej činnosti pre konečný výstup systémy pre novú úroveň riadenia. Takéto univerzálne inteligentné manažérske prostredie je tvorené spojením čiastkových inteligentných riadiacich priestorov nižšej úrovne.

V mnohých zahraničných štúdiách a výskumoch je zmena systémov riadenia v tomto smere chápaná ako koncept zameraný na centralizáciu riadenia siete. Centralizácia siete je pojem prevzatý z oblastí telekomunikácií, kde sa používa na prepojenie počítačov s rôznymi operačnými systémami. Pre oblasť energetiky znamená centralizácia siete správu objektov v jednotnom informačnom a komunikačnom riadiacom systéme, berúc ohľad na prepojenie rôznorodých objektov.

III. KONCEPT A ROZVOJ INTELIGENTNÝCH SIETÍ

Počiatočné ustanovenia prijaté počas prípravnej fázy a vývoja konceptu inteligentnej siete v zahraničí:

1) Koncept inteligentnej siete predpokladá systematickú transformáciu energetického priemyslu (energetický systém) a ovplyvňuje všetky jeho základné prvky, akými sú výroba, prenos a distribúcia, vrátane komunálnych služieb, predaj a dispečing;

2) elektrizačná sústava sa bude považovať ako sieť podobná internetovej sieti, ktorá bude určená na podporu energetických, informačných, ekonomických a finančných vzťahoch medzi všetkými subjektami energetického trhu a inými záujmovými stranami;

3) Rozvoj elektroenergetiky by sa mal zamerať na rozvoj existujúcich a na vytvorenie nových funkcionálnych vlastností energetických systémov a ich častí, ktoré by zabezpečovali v čo najväčšej miere zlepšenie kľúčových faktorov novej elektroenergetiky pomocou zdieľania vízií všetkých zúčastnených strán

- 4) elektrická sieť a všetky jej prvky sa považuje za hlavný predmet formácie novej technologickej základne, ktorá umožňuje výrazne zlepšiť dosiahnuté výsledky a vytváranie nových funkčných vlastností energetického systému;
- 5) vývoj koncepcie komplexne pokrýva všetky hlavné oblasti rozvoja od výskumu až po praktické využitie. Vývoj by sa mal vykonávať na vedeckom, právnom, technologickom, technickom, organizačnom, riadiacom základe a informačnej úrovni;
- 6) realizácia koncepcie nesie inovatívny charakter a dáva podnet k prechodu na novú technologickú štruktúru v elektroenergetike a ekonomike [5].

Pojem „inteligentná sieť“ sa dnes vzťahuje na jednu z najdôležitejších oblastí rozvoja trhu a technológií na celom svete v oblasti prenosu a distribúcie elektrickej energie. V podstate ide o technológie, ktoré sú schopné vytvoriť elektrickú sieť transparentnou a plne ovládateľnou:

- zahraničné inteligentné siete sú spojením obojsmernej dátovej výmene dát v digitálnej podobe všetkých účastníkov, akými je výroba, distribúcia, skladovanie a spotreba elektrickej energie;
- Ruské „inteligentné siete“ predstavujú komplexný, modernizovaný a inovačný rozvoj všetkých subjektov elektroenergetického priemyslu na základe moderných technológií v celej krajine [2].

Okrem toho vytvorenie komplexu „Inteligentných sietí“ na území RF v rôznych odvetviach ekonomiky prirodzene vedie k vytvoreniu nového systémového a štrukturálneho vzdelávania, ktoré možno nazvať medziodvetvovým klastrom (angl. „Cluster“), pretože sa bude skladať z počítačového klastera a distribuovaného počítačového systému, čiže mriežky (angl. „Grid“). Hoci prvé prejavy medziodvetvových počítačových klastrov sa začali objavovať už v roku 2005, aj napriek tomu sa vedecký výskum začal len pred niekoľkými rokmi.

Perspektíva rozvoja medziodvetvového zoskupenia sietí založeného na prevažujúcich organizačných a manažérskych potrebách je trendom, ktorý by v konečnom dôsledku mal mať usporiadanú prepojenosť fungovania a prepojenia distribuovaných informácií medzi objektami, informačnými sieťami a užívateľmi informácií.

Výhodami medziodvetvového klastra sú napr. širšie možnosti pre zhromažďovanie, spracovanie, uchovávanie, distribúciu informácií a pod. Ďalej je to schopnosť prispôbiť sa dopytu po informáciách, ich využitiu a efektívnemu poskytovaniu informácií z pohľadu ekonomiky na splnenie požiadaviek a zvýšení spokojnosti zákazníkov.

Na základe vývoja informačných systémov novej generácie, ktoré sú súčasťou medziodvetvových klastrov je možné dôjsť k záveru, že konečný prechod ruskej ekonomiky na novú kvalitatívnu úroveň riadenia formou vytvorenia viacúrovňového celku informačných systémov riadenia, ktorých väčšiu časť tvorí automatizácia a zvýšenie kvantitatívnych a kvalitatívnych charakteristík zberu, spracovania, uchovávania a distribúcie informácií pre potreby prijatia manažérskych rozhodnutí, t. j. k prechodu na inteligentné informačné systémy, ktoré sú súčasťou medziodvetvových klastrov na území Ruskej federácie.

Takýmto spôsobom, výsledkom rozvoja medziodvetvového klastera je nárast efektívnosti riadenia založenej na kvalitatívne vyššej úrovni zberu, spracovania, skladovania a distribúcie informácií, ktoré sa zvykli využívať v procese prijímania manažérskych rozhodnutí na základe paradigmy fungovania siete. Takýto spôsob je v poslednom období dopĺňaný novými technologickými riešeniami.

Pre niekoľkými rokmi sa v mnohých krajinách začali zahajovať projekty, ktoré sa zameriavajú na využitie internetu vecí (angl. IoT – Internet of Things). Podstata využívania IoT spočíva v technológiách rádiových frekvencií identifikácie (angl. Radio Frequency Identification). Rádiová frekvencia identifikácia, zvyčajne nazývaná ako RFID je metóda identifikácie, ktorá používa RFID štítky (tagy) alebo transpondéry za účelom ukladania a čítania informácií pomocou rádiových vln.

Rozvoj „internetu vecí“, ktorý umožňuje prepojenie globálnych telekomunikačných sietí prostredníctvom informačných agentov a komodít pomocou RFID štítkov sa stáva jedným z najperspektívnejších trendov v oblasti rozvoja informačných a komunikačných technológií. [6]

Súčasná štruktúra informačných systémov je v mnohých krajinách veľmi rôznorodá, čiže obsahuje širokú škálu technických a technologických prvkov, čo mimoriadne spomaľuje proces zvýšenia efektívnosti riadenia, z pohľadu zraniteľnosti vytvára možnú „vstupnú bránu“ pre potenciálny pokus o prebratie kontroly nad informačnými a komunikačnými sieťami, čo si vyžaduje vyššiu kvalitu a spoľahlivosť riadiacich systémov. [3]

Ako príklad úspešného pokusu o prebratie kontroly riadenia sa môže uviesť situácia z 14. augusta 2003. Vtedy v USA došlo k nehode s tzv. „domino efektom“, výsledkom ktorej bolo diaľkové odpojenie jedného z viacerých prvkov elektrizačnej sústavy spôsobené neznámym páchatelom. Dôsledkom čoho došlo k preťaženiu a poškodeniu aj ostatných prvkov v sieti. V dôsledku tejto nehody bolo približne 50 mil. odberateľov odpojených od dodávok elektrickej energie v trvaní asi 4 dní.

Počas tohto obdobia by odberatelia spotrebovali viac ako 60 GWh elektrickej energie. Dá sa to porovnať k spotrebe amerického štátu New York, spolu s jeho susednými štátmi, ako aj provincie Ontário v Kanade. Len v USA straty činili viac ako 10 miliárd amerických dolárov a v Kanade sa straty pohybovali na hranici asi 2 miliárd kanadských dolárov.

IV. ZHODNOTENIE A ZÁVER

Zložitosť riešenia danej situácie spočíva v jeho rozšírení, pretože situácia vyžaduje komplexné zváženie rôznych aspektov, akými sú ekonomické, organizačné, technické, informačné a pod. K riešeniu týchto problémov je potrebné spoločné riešenie, ktoré si vyžaduje vývoj nových koncepcií pomocou moderných vedeckých poznatkov. [7]

Tieto ťažkosti je možné prekonať iba zavedením softvéru a moderných organizačných mechanizmov riadenia a prispôsobenia založených na inteligentných technológiách využívajúcich pokročilé metódy analýzy súčasnej situácie a prijímania účinných rozhodnutí, ktoré majú viesť k zníženiu rizika prebratia kontroly nad inteligentnými sieťami.

Navyše, predmetné technológie pracujúce s týmito zdrojmi informácií sa môžu neustále vylepšovať na základe požiadaviek vlády, platnej legislatívy alebo všeobecnými požiadavkami a potrebami na trhu. Keďže potenciál medziodvetvového klastera umožňuje sformulovať súbor opatrení na vyriešenie problému „Riadenie objektov kritickej energetickej infraštruktúry v podmienkach vonkajšieho elektronického deštruktívneho pôsobenia“.

Navrhovaný súbor opatrení na vyriešenie tohto problému musí jasne a zreteľne popisovať štruktúru funkcií, aby bolo možné navrhnúť presný postup a popis všetkých častí. Perspektíva rozvoja inteligentných sietí v oblasti elektroenergetiky a ďalších odvetviach národného hospodárstva Ruskej federácie pri ich integrácii sa vytvára nástroj na riadenie ekonomiky krajiny na kvalitatívne vysokej úrovni, než tomu bolo za čias ZSSR a je tomu teraz. Okrem toho v súvislosti s objektívnym rozdelením podnikovo-regionálnej štruktúry elektroenergetiky iných krajín, predovšetkým USA a rozvinutých krajín západnej Európy v zásade nie je možné dosiahnuť takúto úroveň riadenia sietí.

Zároveň formovanie takéhoto systému na území Ruskej federácie vytvorí mechanizmus pre riadenie energetiky a ekonomiky, ktorý sa nebude zhoršovať z krátkodobého ani dlhodobého hľadiska v krajinách s centralizovaným riadením, tak ako je tomu napr. v Číne. Samozrejme berúc do úvahy geostrategické ciele (priority) a úlohy zamerané na rozvoj energetiky, ktoré si daná krajina stanovila.

Takýto systém sa stáva mimoriadne účinným nástrojom geoekonomického a geopolitického riadenia celej Ruskej federácie, nielen podporou jednoty vzdialených regiónov a hospodárskych komplexov krajiny, ale aj ako nástroj poskytujúci štátnu suverenitu a schopnosť čeliť vonkajšiemu tlaku v energetickom sektore v jeho organizačných alebo technologických formách. Zároveň aj v mimoriadnych situáciách, akým je vojnový stav alebo prírodná katastrofa. Ale aj ako nástroj integrovaný do ruského riadiaceho centra energetických systémov a ekonomík republík bývalého ZSSR, ako aj niekoľko ďalších krajín blízkeho východu a Európy.

REFERENCES

- [1] BERDNIKOV, R.N. Politika innovacionnogo razvitiya i modernizacii OAO „FSK EES“. Available at: <<http://www.fsk-ees.ru>>
- [2] BUDARGIN, O.M. Umnaya set' - platforma razvitiya innovatsionnoy ekonomiki, zasedanie kruglogo stola «Umnyye seti - Umnaya energetika - Umnaya ekonomika» 4-ogo Peterburgskogo mezhdunarodnogo ekonomicheskogo foruma. 2010.
- [3] BUGAEV, A.S., LOGINOV, E.L., RAJKOV, A.N., SARAJEV, V.N. Latentnyy sintez resheniy, Ekonomicheskiye strategii. 2007.
- [4] DOROFEEV, V.V., Intellektual'naya set'. Novyye printsipy postroyeniya. Oborudovaniye sistemy upravleniya intellektual'noy seti. zasedanie kruglogo stola «Umnyye seti - Umnaya energetika - Umnaya ekonomika» 4-ogo Peterburgskogo mezhdunarodnogo ekonomicheskogo foruma. 2010.
- [5] KOBEC, B.B., VOLKOVA, I.O., OKOROKOV, V.R., Smart grid kak kontseptsiya innovatsionnogo razvitiya elektroenergetiki za rubezhom. Energoekspert. 2010.
- [6] LOGINOV, E.L. «Internet veshchey» kak attraktor obyektivnoy ekonomicheskoy real'nosti. Natsional'nyye interesy: priority i bezopasnost'. 2010.
- [7] LOGINOV, E.L., SHKRABLYAK, A.S., Tendentsii razvitiya elektronnykh finansovykh tranzaktsiy i metodov ikh kontrolya v global'nykh telekommunikatsionnykh setyakh. Inzhenernaya fizika. 2009.

Diskový krokový motor s dvojitým rotorom s permanentnými magnetmi

¹Ján KAŇUCH, ²Peter GIROVSKÝ

^{1,2} Katedra elektrotechniky a mechatroniky, Fakulta elektrotechniky a informatiky, Technická univerzita, Letná 9, 04200 Košice, Slovenská Republika

¹jan.kanuch@tuke.sk, ²peter.girovsky@tuke.sk

Abstrakt — Hlavným cieľom tohto príspevku je prezentovať výsledky konštrukčného návrhu a výpočtu hlavných parametrov elektromagnetického obvodu štvorfázového diskového krokového motora s malým krokom ($1,8^\circ$) a vonkajším dvojitým aktívnym rotorom, teda rotorom s permanentnými magnetmi. V úvode príspevku sú stručne popísané klasické krokové motory, možnosti ich riadenia a konštrukčné riešenie troch základných typov. Tento príspevok sa v druhej kapitole zaoberá problematikou diskových krokových motorov s axiálnym magnetickým tokom a permanentnými magnetmi na rotore, vlastnosťami týchto motorov a ich základnými parametrami. V príspevku je bližšie opísaná konštrukcia a princíp motorov typu TORUS s jedným vnútorným diskovým statorom a dvojitým aktívnym rotorom.

Kľúčové slová — krokový motor, diskový rotor, permanentný magnet

I. ÚVOD

Krokové motory (obr. 1) sa rozvíjajú už od polovice 19. storočia, keď bol prvýkrát popísaný princíp a konštrukcia reluktančného motora. V polovici 20. storočia sa začali vyrábať prvé krokové motory s permanentnými magnetmi. Krokové motory sú stále viac konkurencieschopné so servo motormi, keďže majú bezkonkurenčné ceny a výrobcovia postupne znižujú a odstraňujú ich nedostatky, ako je napr. dosiahnutie vyššieho krútiaceho momentu pri vysokých rýchlostiach a zdokonaľovanie riadiacich algoritmov.



Obr. 1 Krokový motor MEZ Z420L177 a) a detail vnútra krokového motora [1] b)

Vinutia statora krokového motora (vyrábajú sa ako jednofázové alebo viacfázové) sú napájané pomocou prúdových impulzov rôznej polarít. Zmenou prúdu v statorovom vinutí sa mení magnetické pole a tým aj silový účinok. Rotor krokového motora sa natočí podľa polarít magnetického poľa statora o presne definovaný uhol – uhol kroku. Pre generovanie prúdových impulzov je potrebná špeciálna riadiaca elektronika, teda tzv. elektronické rozdeľovače impulzov, ktoré generujú impulzy pre postupné natáčanie rotora o definovaný uhol kroku.

V súčasnosti sa pre riadenie krokových motorov používa niekoľko spôsobov [2]:

- Riadenie na plný krok - Pri tomto spôsobe riadenia sa rotor otáča (krokuje) o základný uhol kroku, teda počet krokov na jednu otáčku závisí od počtu zubov rotora. Pri počte 200 zubov to znamená 200 krokov na otáčku, čiže uhol kroku (pootočením rotora) je vtedy rovný $1,8^\circ$. Napája sa vždy len jedna fáza statora.

- Riadenie na polovičný krok - Výsledkom tohto spôsobu riadenia krokovania so súčasným spínaním dvoch fáz je to, že uhol krokovania sa zníži na polovicu. Výhodou tohto spôsobu riadenia je okrem dvojnásobného zníženia kroku aj plynulejšia prevádzka v porovnaní s riadením na plný krok. Nevýhodou je ale o 15% nižší krútiaci moment, než pri plnom krokovaní a zvyšuje sa aj prúd v motore pri napájaní len jednej fázy.
- Mikrokrokovanie - Pri tomto riadení krokovania sa zvyšuje presnosť krokovania tým, že sa v každom vinutí riadi nielen veľkosť ale aj smer toku prúdu. Tvar prúdu vo vinutiach motora je úmerný podľa sinusových alebo kosínusových funkcií. Mikrokrovaním je možné rozdeliť základný krok rotora až 256-krát. Pri klasickom riadení tečie fázami motora rovnaký prúd. Pri mikrokrovaní sa tento prúd mení a tým sa nastavuje ľubovoľná rovnovážna poloha medzi dvoma susednými krokmi. Mikrokrokovanie teda zvyšuje nároky na riadiacu elektroniku. Krútiaci moment sa však zníži až o 30%.

Krokové motory možno rozdeliť do troch základných typov:

1. *Krokový motor s premenlivou reluktanciou* – Stator a rotor je z elektrotechnických plechov. Stator má na vnútornej strane vyjadrené póly, na ktorých je vinutie, pričom ešte každý pól môže byť rozdelený na väčší počet zubov (v závislosti od veľkosti kroku). Rotor má na vonkajšej strane tiež zuby. Počet zubov rotora sa nerovná počtu zubov statora. Avšak uhol kroku nesúvisí s počtom zubov na statore ale len na rotore a s počtom budiacich fáz statora. Tieto motory pracujú na princípe minimálnej reluktancie magnetického obvodu, čo znamená to, že pokiaľ je nejaká cievka budená, tak v motore sa vytvorí taký moment (reluktančný), ktorý sa snaží otočiť rotor do najbližšej rovnovážnej polohy, čiže do polohy, v ktorej je magnetický odpor (reluktancia) motora minimálny. V typickej aplikácii tieto krokové motory dosahujú pomerne vysoké krokové rýchlosti s uhlom kroku od 5 do 15 stupňov.
2. *Krokový motor s permanentným magnetom (PM)* – Stator krokového motora s PM má vyjadrené póly, rotor je však valcový, na ktorom sú umiestnené PM. Permanentné magnety sú bez zubov, ale sú magnetizované kolmo na os otáčania. Rotor sa otáča v dôsledku príťažlivosti magnetických pólov statora a rotora, keď sú napr. štyri fázy pod napätím v určitom poradí. Krokové motory s permanentným magnetom majú typicky vysoký krútiaci moment a lepšie tlmenie.
3. *Hybridný krokový motor* – Tento typ krokového motora má vlastnosti krokových motorov s permanentným magnetom a tiež s premenlivou reluktanciou. Rotor sa podobá rotoru motora s premenlivou reluktanciou. Avšak na hriadeľi je aj osovo magnetizovaný koncentrický permanentný magnet, ktorý umožňuje vyšší krútiaci moment motora, lepšiu dynamiku a vyšší prídržný moment. Takže rotor je teda zložený z valcového permanentného magnetu a na obidvoch koncoch magnetu sú rotorové pólové nástavce, ktoré majú po obvode zuby. Tieto pólové nástavce sú rovnaké, ale sú voči sebe natočené o jednu polovicu zubového rozstupu rotora. Hybridné krokové motory sa vyrábajú s krokom $0,9^\circ$ a $1,8^\circ$ stupňoch (bežne používaných) a tiež s krokom $3,6^\circ$ a $4,5^\circ$.

II. MOTORY S DISKOVÝM ROTOROM A PERMANENTNÝMI MAGNETMI

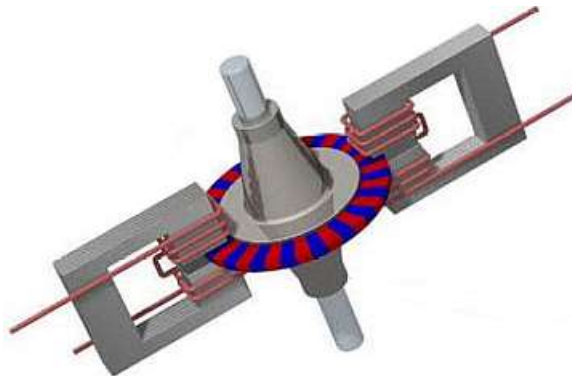
Diskové motory (s axiálnym magnetickým tokom – obr.2) s permanentnými magnetmi boli prvýkrát predstavené už koncom 70-tych rokov minulého storočia. Rastúci záujem o tieto stroje v rôznych aplikáciách je aj v súčasnosti a to kvôli ich vysokému pomeru krútiaceho momentu k hmotnosti a účinnosti, ako alternatíva ku konvenčným strojom.



Obr. 2 Krokový motor s diskovým rotorom a axiálnym magnetickým tokom [3]

Diskové motory s permanentnými magnetmi majú viacero jedinečných vlastností. Vďaka permanentným magnetom sa účinnosť strojov výrazne zlepšila a dosiahla sa vyššia hustota energie. Výkon majú porovnateľný s hybridnými motormi. Na rozdiel od valcových rotorov u bežných motorov s permanentným magnetom, sú rotory v diskových motoroch tenké (zvyčajne aj menej ako 1mm) disky. Bežné krokové motory s permanentnými magnetmi sú obvyčajne obmedzené na minimálny uhol kroku 30° pre maximálne 12 krokov na otáčku [4].

Až do nedávnej doby, mali krokové motory maximálne 200 krokov na otáčku. Najnovšie verzie, ktoré majú uhol kroku $0,9^\circ$, majú teda 400 krokov na otáčku. Diskové motory sú vo všeobecnosti o polovicu menšie ako hybridné motory s rovnakým výkonom a ich hmotnosť je o 60% nižšia. Rotory a ich magnetické obvody sú ľahké a sú vyrábané skôr z tvrdých plastov ako kovu (mimo magnetov). Princiálne konštrukčné riešenie diskového krokového motora firmy Portescap je uvedené na obr. 3.



Obr. 3 Princiálne konštrukčné riešenie diskového krokového motora s PM - Portescap [4]

Aj keď široká rada diskových motorov je k dispozícii už asi 30 rokov, až v posledných rokoch sú stále viac využívané pre ich nízky moment zotrvačnosti a vysoký krútiaci moment. Väčšina diskových motorov sa počíta pre špičkový prúd 10-násobne väčší ako je trvalé zaťaženie na prekonanie zotrvačnosti pri zrýchlení a spomalení. Bežné motory sú obvyčajne navrhované na špičkový prúd iba 2 až 3-násobný k prúdu pri trvalom zaťažení. V opačnom prípade vyššie prúdy vytvárajú silné magnetické pole v rotore, čo spôsobuje demagnetizáciu magnetov.

Vo všeobecnosti diskové stroje s permanentnými magnetmi majú vysokú účinnosť a dobrý pomer krútiaceho momentu k ich hmotnosti a toto podnecuje k výskumu ich rôznych konštrukcií a možnosti využitia pre rôzne aplikácie v regulovaných pohonoch. Diskové stroje s permanentnými magnetmi a axiálnym tokom sú najobľúbenejšie a sú používané a skúmané rôzne typy ich konštrukcie. Existujú však dve základné konštrukčné topológie týchto strojov:

1. diskové stroje s dvoma vonkajšími rotormi a jedným statorom v strede,
2. diskové stroje s jedným rotorom v strede dvoma vonkajšími statormi.

Diskové motory s PM majú veľa významných výhod. Napr. tak ako aj iné motory s PM, tiež môžu byť navrhnuté tak, aby pomer výkonu k hmotnosti bol maximálny. Taktiež napr. hladina hluku a vibrácií je menšia ako pri konvenčných strojoch. Smer hlavného magnetického toku vo vzduchovej medzere sa môže jednoducho meniť (v závislosti od polarizácie permanentných magnetov). Mimo týchto výhod, hustota magnetického toku je redukovaná v dôsledku dvojnásobnej vzduchovej medzery (na rozdiel od valcového rotora), teda tok pri diskovom rotore tečie štyrikrát cez vzduchovú medzeru (pri valcovom rotore dvakrát). Avšak jednou z dôležitých výhod tohto stroja je, že konštrukciou statora sa veľmi ľahko prenáša teplo z motora, a z tohto dôvodu môže byť elektrické preťaženie motora pomerne vysoké. Nakoniec aj vhodný tvar a veľkosť motora má zásadný význam pre využitie v niektorých špeciálnych aplikáciách.

III. MOTORY TYPU TORUS

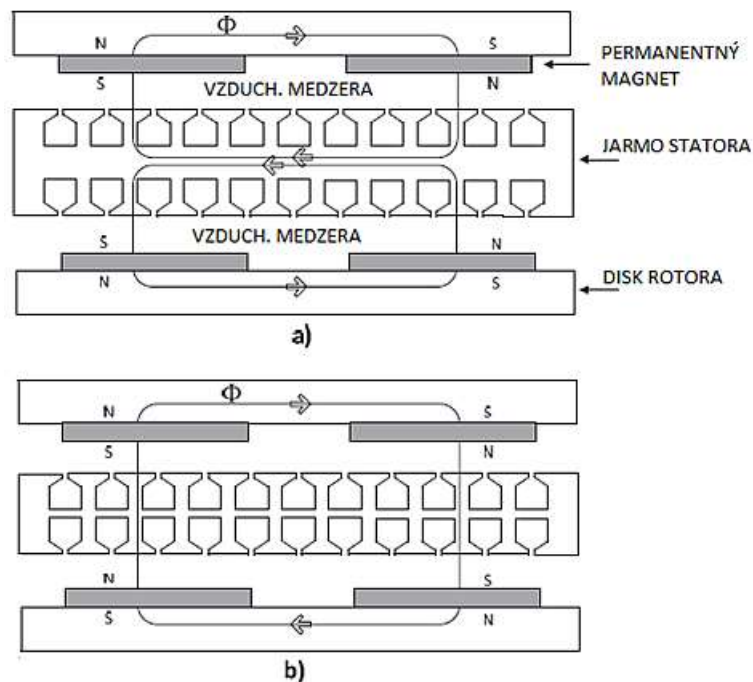
Motory typu TORUS sú veľmi často využívaným variantom pri konštrukčnom usporiadaní diskových krokových motoroch s axiálnym tokom a s permanentnými magnetmi. Ide v podstate o zaužívanú koncepciu dvoch rotorov a jedného statora, ktorý sa nachádza v strede medzi rotorovými diskami. Vinutia na diskovom statore sú orientované v radiálnom smere a existujú rôzne typy v závislosti od usporiadania magnetov [5].

Motory TORUS sú teda špeciálnym typom diskových strojov s permanentnými magnetmi, ktoré sú vhodné pre využitie aj v elektromobiloch kvôli niekoľkým dobrým vlastnostiam. Motor

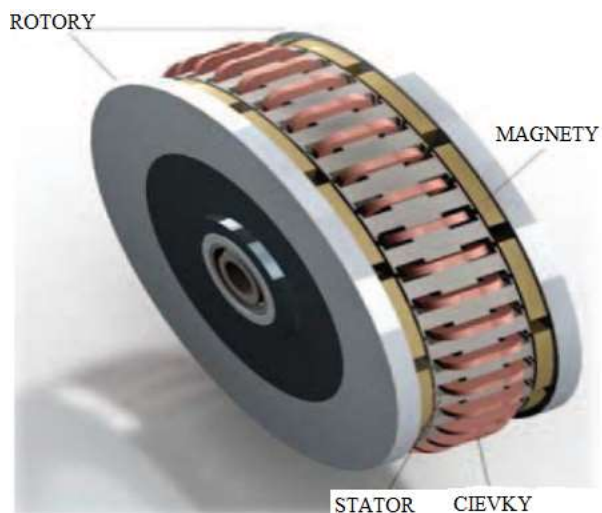
má pomerne veľký priemer rotora, ktorý umožňuje uloženie veľkého počtu pólov, čoho dôsledkom je to, že tieto stroje sú ideálne pre nízke rýchlosti a vysoký krútiaci moment. Okrem toho, diskový tvar aktívnych častí motora umožňuje vytváranie rôznych konštrukčných riešení, napríklad s rôznym počtom vzduchových medzier, s drážkovaným statorom alebo statorom bez drážok, alebo dokonca s rotorom bez železa. Na rotorových diskoch sú upevnené axiálne polarizované magnety, ktoré sú upevnené napr. na vnútornej ploche diskového rotora. Vinutia na statore sú orientované v radiálnom smere .

Princíp fungovania motora typu TORUS je ten istý, ako pri synchronnom stroji. Rýchlosť otáčania rotora závisí od frekvencie napájacieho napätia. Zotrvačnosť systému výrazne filtruje pulzovanie krútiaceho momentu pri vyšších otáčkach. V závislosti od polarít magnetov na rotore sú na obr. 4 znázornené dva základné konštrukčné riešenia motorov typu TORUS s vyznačením cesty magnetického toku v oboch konfiguráciách:

- a) s orientáciou magnetov NN,
- b) s orientáciou magnetov NS.



Obr. 4 Principiálne konštrukčné riešenia diskového krokového motora typu TORUS [5]



Obr. 5 Principiálne konštrukčné riešenia diskového krokového motora typu TORUS [5]

V porovnaní s inými konštrukčnými riešeniami má motor typu TORUS so segmentovanou kotvou niekoľko podstatných výhod [6]. Vzhľadom na použitie permanentných magnetov zo vzácnych zemín s vysokou hustotou energie, krátkych čiel vinutí a vysokých koeficientov plnenia vinutia, je účinnosť týchto motorov výrazne zlepšená. Mimo toho je hmotnosť statora a teda aj straty v železe značne znížené v dôsledku chýbajúceho jarma statora vďaka konštrukčnému riešeniu s orientáciou magnetov NS.

IV. ELEKTROMAGNETICKÝ NÁVRH DISKOVÉHO KROKOVÉHO MOTORU S DVOJITÝM ROTOROM S PERMANENTNÝMI MAGNETMI

Merný ťah klasického valcového rotora sa vypočíta podľa rovnice

$$\sigma = \frac{2 \cdot M}{\pi \cdot d^2 \cdot l} \quad (1)$$

kde M je požadovaný moment motora, parameter d je priemer rotora a l je dĺžka rotora.

V aktívnej dvojvrstve diskového rotora platí pre merný ťah diskového rotora, teda pre dve planárne vrstvy rovnica

$$\sigma = \frac{F}{2 \cdot [\pi \cdot (d-h) \cdot h]} \quad (2)$$

kde F je sila na diskový rotor, parameter h je výška medzikružia a parameter d je vonkajší priemer medzikružia.

Ak silu F nahradíme momentom M , keďže sila F je veličina platná pre celý disk a pre silu platí rovnica

$$F = \frac{2 \cdot M}{d-h} \quad (3)$$

tak po dosadení (3) do (2) a následnej úprave, potom pre merný ťah diskového rotora platí rovnica

$$\sigma = \frac{M}{\pi \cdot (d-h)^2 \cdot h} \quad (4)$$

Z rovnice (4) vyplýva, že veličina v jej menovateli

$$\pi \cdot (d-h)^2 \cdot h \quad (5)$$

je planárna a je rozmerovým kritériom diskového krokového motora.

Teda z porovnania rovníc (1) a (4) nám vyplýva, že diskový rotor môže mať aj malú hrúbku a jej veľkosť môže výhradne závisieť len od konštrukčných možností jeho výroby a od pevnosti materiálu rotorového disku.

Pri návrhu diskového motora je potrebné určiť základné rozmery krokového motora, ako sú jeho vonkajší priemer, výška medzikružia, navrhnuť tvar a rozmery pólov rotora aj statora a vypočítať budiace vinutie.

A. Základné rozmery a návrh magnetického obvodu

Pri určovaní základných rozmerov magnetického obvodu sa použije rovnica (4) upravená na tvar

$$(d-h)^2 \cdot h = \frac{M}{\sigma \cdot \pi} \quad (6)$$

Priemer disku sa vypočítal pomocou upravenej rovnice (6), keď sa predtým zvolila výška medzikružia h a požadovaný moment motora M , teda z rovnice

$$d = \sqrt{\frac{M}{\sigma \cdot \pi} + h} + h \quad (7)$$

Počet krokov na otáčku, teda aj počet magnetov v rotore vychádza zo zadaného počtu fáz $m=4$ a počtu pólov statora $2p$. Počet magnetov v rotore sa zvolil $Q_2=50$. Pri návrhu počtu pólov statora a počtu magnetov rotora sa muselo prihliadať na to, aby nemohlo dôjsť k zlepeniu statora a rotora, preto sa zvolili zadané hodnoty, pri ktorých toto nemôže nastať. Pre počet krokov motora na otáčku platí

$$K = Q_2 \cdot m = 50 \cdot 4 = 200 \quad (8)$$

Pre mechanický uhol kroku platí

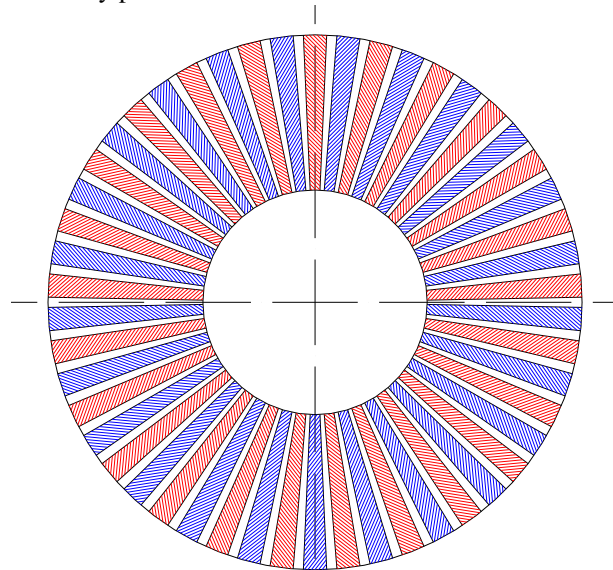
$$\alpha_K = \frac{360}{K} = \frac{360}{200} = 1,8^\circ \quad (9)$$

a pri osemtaktom riadení by bol mechanický uhol kroku ešte zmenšený na polovicu (na $0,9^\circ$).

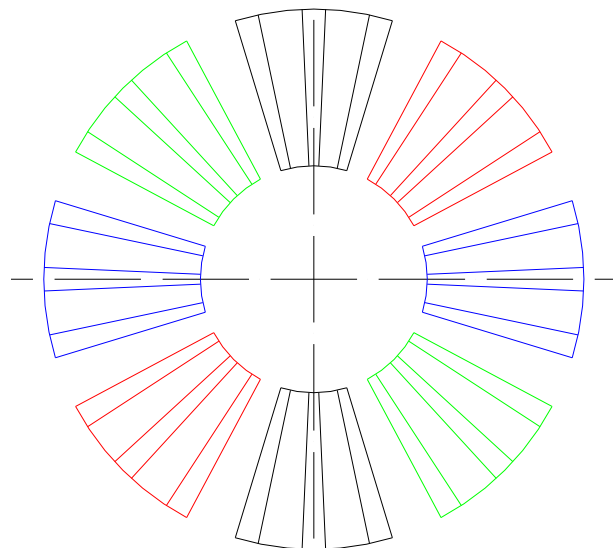
Rozmery rotorového disku sa určili pomocou rovnice (7) a boli vypočítané nasledujúce hodnoty: vnútorný priemer $d_{\text{vnut}} = 70$ mm a vonkajší priemer $D_{\text{vonk}} = 150$ mm. Pre dosiahnutie čo najmenšieho uhla kroku sú klasické zuby nahradené 50-timi osadenými neodýmovými magnetmi (Obr. 6). Rozstup medzi magnetmi bol určený na $7,2^\circ$. Magnety sú magnetizované striedavo a medzi nimi sa nachádza nemagnetický spojovací materiál s hrúbkou 2 mm.

Motor je 4-fázový, čo znamená, že na oboch stranách statorového disku je 8 statorových pólov s rozstupom 45° (obr. 7). Jednotlivé póly majú pólové nadstavce s tromi zubami, medzi ktorými sú drážky. Rozmery a tvar zubov sú rovnaké, ako sú rozmery a tvar permanentných

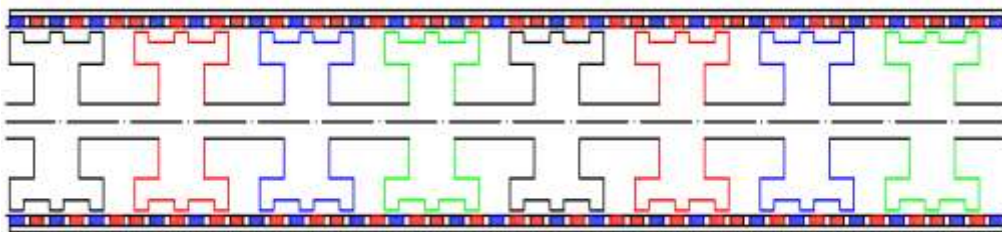
magnetov na rotorových diskoch. Na obr. 8 je zobrazený rozvinutý priečný rez magnetickým obvodom motora cez stredný priemer disku.



Obr. 6 Rotorový disk s PM



Obr. 7 Póly statora

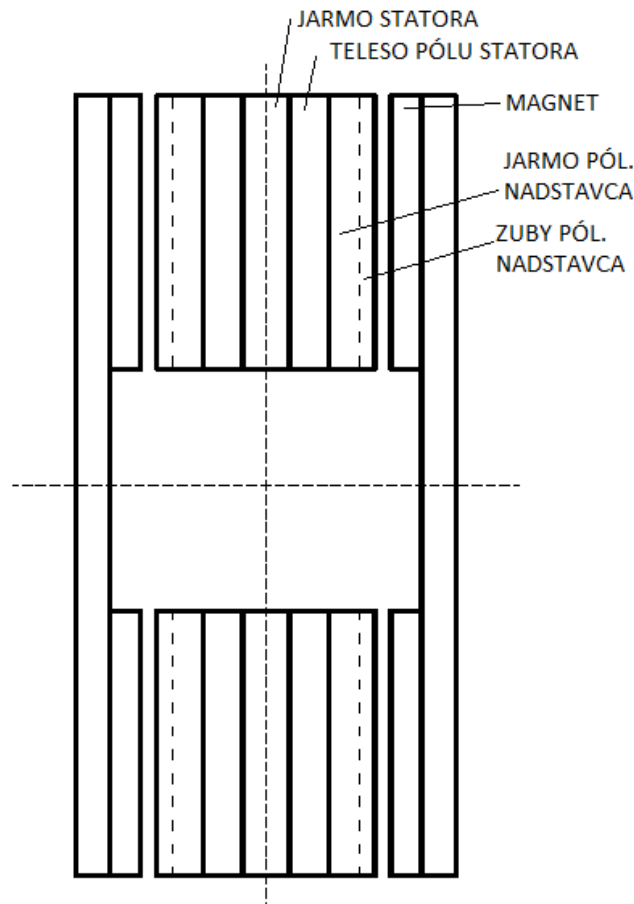


Obr. 8 Rozvinutý priečný rez magnetickým obvodom cez stredný priemer disku

B. Elektromagnetický výpočet motora

Elektromagnetický optimalizačný výpočet motora zahŕňa výpočet parametrov vo vzduchovej medzere (magnetická indukcia vo vzduchovej medzere, intenzita magnetického poľa, magnetické napätie, magnetická vodivosť a hodnota magnetického toku vo vzduchovej medzere), magnetické napätie v rotore motora – permanentný magnet, magnetické napätie v statore motora (jarmo pólového nastavca statora, zuby pólového nastavca statora, teleso pólu statora a jarmo statora), magnetické napätie v jarme rotora a výpočet budiaceho vinutia, čiže počet závitov jednej fázy z celkového magnetického napätia v magnetickom obvode motora a požadovaného fázového prúdu. Rez magnetický obvodom motora s dvojitým vonkajším diskovým rotorom je zobrazený na obr. 9.

Následne sa urobí prepočet indukčnosti vo vzduchovej medzere a porovná sa s pôvodnou hodnotou vypočítanou pre predpokladaný počet závitov. Keď je rozdiel veľký, tak sa urobí prepočet všetkých potrebných veličín v magnetickom obvode pre nový zvolený počet závitov a taktiež sa prepočítajú aj všetky jednotlivé parametre, ktoré táto zmena ovplyvnila. V tabuľke 1 sú uvedené vypočítané hlavné veličiny a parametre elektromagnetického obvodu motora.



Obr. 9 Magnetický obvod motora s dvojitým diskovým rotorom

Tabuľka 1
Vypočítané hlavné parameter elektromagnetického obvodu motora

Symbol	Veličina	Vypočítaná hodnota
N	Počet závitov jednej fázy	200
R_{IFS}	Odpor jeden fázy	0,253 Ω
L_{IFS}	Indukčnosť jeden fázy	4,822 mH
Φ_{00}	Magnetický tok	$8,11 \cdot 10^{-4}$ Wb
B_{PNS}	Magnetická indukcia v jarme póloveho nadstavca statora	0,650 T
B_P	Magnetická indukcia v telese pólu statora	1,228 T
B_{ZPN}	Magnetická indukcia v zuboch póloveho nadstavca statora	1,535 T
B_{JS}	Magnetická indukcia v jarme statora	0,518 T
B_{JR}	Magnetická indukcia v jarme rotora	2,260 T

C. Výpočet statického väzobného momentu krokového motora

Statický väzobný moment je moment, ktorý je v rovnováhe s krútiacim momentom pôsobiacim na hriadeľ stojaceho krokového motora, ktorý je nabudený, a rotor je pootočený z magneticky kľudovej polohy o daný statický uhol záťaže. Pre výpočet momentu je možné odvodiť vzťah

$$M_{sv} = \frac{1}{2} \cdot N_c^2 \cdot I_f^2 \cdot \frac{\partial W}{\partial \beta_m} \quad (10)$$

kde N_c je počet závitov cievky budiacej fázy, I_f je fázový prúd, W je energia magnetického poľa v magnetickom obvode a β_m je statický uhol záťaže.

Z rovnice (10) je zrejmé, že ak je v nabudenej cievke krokového motora konštantný prúd, hodnota statického momentu je závislá iba od zmeny vodivosti v obvode motora. V našom

prípade je to vtedy, keď sa rotor pootočí o mechanický uhol kroku, teda o $1,8^\circ$. Pri zanedbaní nelinearity je možné rovnicu (10) upraviť na tvar

$$M_{sv} = \frac{1}{2} \cdot N_c^2 \cdot I_f^2 \cdot \frac{\Delta\lambda}{\Delta\beta_m} \quad (11)$$

Zmena magnetickej vodivosti $\Delta\lambda$ pre celý obvod sa vypočíta ako

$$\Delta\lambda = 2 \cdot (\lambda_{max} - \lambda_{min}) \quad (12)$$

kde λ_{max} je magnetická vodivosť pre polovicu magnetického obvodu motora v magnetickej kľudovej polohe a λ_{min} je magnetická vodivosť pre polovicu magnetického obvodu motora v polohe pred vykonaním kroku, pričom $\Delta\beta_m$ je zmena uhla záťaže, teda uhla pootočenia rotora pri zmene vodivosti z minimálnej do maximálnej hodnoty.

Pootočenie rotora, teda uhol kroku motora sa prepočíta na radiány

$$\Delta\beta_m = \frac{\alpha_K \cdot \pi}{180} \quad (13)$$

V tabuľke 2 sú uvedené výsledky výpočtov hlavných veličín navrhovaného diskového krokového motora s dvojitým vonkajším rotorom a s permanentnými magnetmi pre určenie statického väzobného momentu motora.

Tabuľka 2
 Výsledky výpočtov hlavných veličín pre určenie statického väzobného momentu motora

Symbol	Veličina	Vypočítaná hodnota
m	Počet fáz motora	4
I_f	Fázový prúd	5,0 A
λ_{max}	Magnetická vodivosť v magnetickej kľudovej polohe	$1,23 \cdot 10^{-6}$ H
λ_{min}	Magnetická vodivosť v polohe pred vykonaním kroku pre polovicu magnetického obvodu	$7,29 \cdot 10^{-7}$ H
$\Delta\lambda$	Zmena magnetickej vodivosti pre celý magnetický obvod motora	$9,97 \cdot 10^{-7}$ H
$\Delta\beta_m$	Pootočenie rotora (uhol kroku motora)	0,03 rad
M_{sv}	Maximálny statický väzobný moment	1,2 N.m

Po niekoľkých prepočtoch a optimalizácii jednotlivých parametrov motora sme dospeli k požadovaným hlavným rozmerom magnetického obvodu a k jednotlivým parametrom motora. Pri návrhu a výpočtoch jednotlivých magnetickej napätí bol použitý ako materiál pre stator a tak isto pre jarmo rotora elektrotechnický plech Eo10, pričom obidva diskové rotory sú osadené permanentnými magnetmi zo vzácnych zemín. Pre vypočítané konečné parametre motora bol vypočítaný aj výsledný statický väzobný moment 1,2 N.m, čiže hodnota, ktorá bola na začiatku predpokladaná.

V. ZÁVER

Krokové motory sú stále viac rozšírené v pohonnárskych aplikáciách, keďže ich parametre a riadenie sa postupne vylepšujú a tým sa rozširuje aj ich možnosť využitia.

V tomto článku je v úvodných kapitolách je prezentovaná a objasnená problematika nie len klasických, ale hlavne diskových krokových motorov s axiálnym tokom [7]-[10]. Obsahom tretej kapitoly je stručný popis konštrukcie krokových diskových motorov typu TORUS s dvoma rotormi a jedným statorom. Pre tento typ motora je v štvrtej kapitole urobený konštrukčný návrh elektromagnetického obvodu štvorfázového krokového motora s malým krokom ($1,8^\circ$) a spracovaný elektromagnetický návrh tohto diskového krokového motora s vonkajším dvojitým aktívnym rotorom, čiže rotorom s permanentnými magnetmi. V poslednej časti je vypočítaný aj výsledný statický väzobný moment motora.

Jednotlivé výsledné elektromagnetické parametre a teda aj moment motora je možné ešte ďalej doladovať úpravou napr. rozmerov jednotlivých častí magnetického obvodu, zmenou veľkosti vzduchovej medzery a pod. Pre kontrolu elektromagnetického návrhu prezentovaného krokového motora by bolo vhodné urobiť aj simuláciu elektromagnetického poľa vo vhodnom simulačnom programe (napr. Motor-CAD od firmy Ansys), ale toto nebolo cieľom tohto príspevku.

POĎAKOVANIE

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy. č. APVV-16-0270.

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-19-0210.

Táto práca vznikla s podporou grantu VEGA č. 1/0187/18.

LITERATÚRA

- [1] Prehľad a zapojenie elektrických motorov, [online, 25.3.2020], dostupné: <https://www.elektrolab.eu/blog/prehľad-a-zapojenie-elektrických-motorov>
- [2] T.A. Khan et al., „Hybrid Stepper Motor and its Controlling Techniques a Survey”, [online, 25.3.2020], *IEEE*, 2014, ISBN: 978-1-4799-2593-3, dostupné: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6839207&queryText%3DHybrid+Stepper+Motor+and+its+Controlling+Techniques+a+Survey>
- [3] P520 Disc Magnet Stepper Motor – Portescap, Disc Magnet Stepper Motor, [online, 19.11.2019], dostupné: <https://www.mclennan.co.uk/product/p520-disc-magnet-stepper-motor>
- [4] Stepper Motor Precision and Accuracy, [online, 20.2.2020], dostupné: <https://www.portescap.com/en/solutions/motor-precision-and-accuracy>
- [5] A. Mahmoudi et al., „Slot-less torus solid-rotor-ringed line-star axial-flux permanent-magnet motor“, *Progress In Electromagnetics Research*, vol. 131, pp. 331-355, 2012, [online, 19.1.2020], dostupné: [gooa.las.ac.cn › external › download](http://www.gooa.las.ac.cn/external/download)
- [6] B. Zhang, M. Doppelbauer, „Numerical Iron Loss Calculation of a New Axial Flux Machine with Segmented-Armature-Torus Topology”, *IEEE*, 2014, E-ISBN: 978-1-84919-815-8, [online, 25.3.2019], dostupné: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6837014&queryText%3DNumerical+Iron+Loss+Calculation+of+a+New+Axial+Flux+Machine+with+Segmented-Armature-Torus+Topology>
- [7] A. Mahmoudi et al., „Axial-flux permanent-magnet machine modeling, design, simulation and analysis”, *Academic Journals*, 2014, ISSN: 1992-2248, [online, 19.2.2020], dostupné: http://www.academicjournals.org/article/article1380714510_Mahmoudi%20et%20al.pdf
- [8] S. M. Mirimani, A. Vahedi, F. Marignetti, „Effect of inclined static eccentricity fault in single stator-single rotor axial flux permanent magnet machines”, *IEEE Trans. Magn.*, Vol. 48, No. 1, pp. 143-149, Jan. 2012.
- [9] A. Mahmoudi, N. A. Rahim, W. P. Hew, „Axial-flux permanent-magnet motor design for electric vehicle direct drive using sizing equation and finite element analysis”, *Progress In Electromagnetics Research*, Vol. 122, pp. 467-496, 2012.
- [10] G. De Donato, F. G. Capponi, F. Caricchi, „No-load performance of axial flux permanent magnet machines mounting magnetic wedges”, *IEEE Trans. Ind. Electron.*, Vol. 59, No. 10, pp. 3768-3779, Oct. 2012.

Úspora Nákladov na Prepravu Prostredníctvom Optimalizácie Trasy

Radka KAČMAREKOVÁ, František BABIČ

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

radka.kacmarekova@student.tuke.sk,frantisek.babic@tuke.sk

Abstrakt — V modernom svete je plánovanie efektívnych trás nevyhnutné pre spoločnosti zamerané na distribúciu rôznych produktov. V tomto článku sa budeme zaoberať problematikou optimalizácie trasy, ktorá si získala popularitu v odvetví dopravy a logistiky, pretože má pozitívny vplyv na chod firmy a to tým, že znižuje čas potrebný na cestovanie, znižuje náklady a podporuje maximálnu produktivitu pracovníkov a nákladných vozidiel. Na tento účel sme analyzovali viacero možností riešenia, od prehľadávania do šírky pomocou Dijkstrovho algoritmu, cez prehľadávanie do hĺbky až k problému obchodného cestujúceho. V poslednom prípade sme vybrali dva konkrétne algoritmy na určenie najkratšej cesty: algoritmus najbližšieho suseda a Littleho metódu. Vhodnosť týchto algoritmov bola experimentálne otestovaná a overená prostredníctvom funkčného prototypu webovej aplikácie.

Kľúčové slová — Optimalizácia trasy, teória grafov, najkratšia cesta.

I. ÚVOD

V dnešnej dobe existuje množstvo spoločností zameraných na distribúciu pohonných hmôt, ktoré chcú získať optimálny plán distribúcie s použitím strojovej sily a vhodných IT technológií. Cieľom týchto spoločností je minimalizovať náklady na prepravu a optimalizovať pracovný čas dispečerov a vodičov cisternových vozidiel. Jedným zo spôsobov ako znížiť náklady na prepravu pohonných hmôt je optimalizácia trasy rozvozu.

Optimalizácia trasy je proces hľadania nákladovo najvýhodnejšej trasy pre skupinu miest, ktoré je potrebné navštíviť. Ide o proces stanovenia najkratších možných trás na dosiahnutie cieľa. Odkedy sa optimalizácia trasy stala digitálnou, výpočet týchto trás sa robí pomocou vhodných algoritmov. Jednu z možností predstavuje teória grafov.

Graf predstavuje určitú grafickú formu modelu rozhodovacieho alebo optimalizačného problému [1]. Pri prehľadávaní postupne navštevujeme všetky vrcholy a hrany grafu. Toto prehľadávanie musí byť systematické a efektívne. Podstatou prehľadávania grafu je postupné označovanie vrcholov. Ak sa navštíví nejaký vrchol prvýkrát, je potrebné si ho zapamätať a označí sa ako „objavený“. Takýmto spôsobom je potrebné pokračovať, až kým nepreskúame všetky vrcholy. Existujú dve hlavné metódy prehľadávania:

- Prehľadávanie do šírky (z angl. Breadth First Search) [2].
- Prehľadávanie do hĺbky (z angl. Depth First Search) [3].

Medzi najznámejšie typy problémov riešených pomocou teórie grafov:

Eulerov cyklus (z angl. Eulerian path) - cyklus, ktorý prechádza každou hranou grafu a cesta končí v začiatočnom vrchole [4]. Tento cyklus využíva Úloha čínskeho poštára, ktorého cieľom je nájsť najkratšiu cestu z východiskového vrcholu po všetkých hranách grafu s návratom do východiskového vrcholu.

Problém štyroch farieb - cieľom tohto problému je zafarbenie vrcholov grafu tak, aby žiadne dva susedné vrcholy nemali totožnú farbu [5]. Za susedné vrcholy sa považujú také, ktoré majú spoločné hraničnú čiaru – hranu. Tento problém sa často využíva v kartografii a to tak, že každému štátu sa priradí jeden vrchol (napr. hlavné mesto) a hranou sa spoja tie vrcholy, ktorých štáty majú spoločnú hranicu.

Hamiltonovský cyklus - ide o cyklus, ktorý prechádza každým vrcholom grafu práve raz [6]. Hamiltonovský cyklus sa úzko spája s pojmom Hamiltonovská kružnica. Hamiltonovská kružnica je podgraf, ktorý obsahuje všetky vrcholy pôvodného grafu. Graf, v ktorom existuje kružnica obsahujúca všetky vrcholy, sa nazýva hamiltonovský graf. Na rozdiel od hľadania

ulerovských ťahov, je hľadanie hamiltonovských cyklov veľmi ťažká úloha. Presnejšie zistenie, či v danom grafe existuje hamiltonovský cyklus, je tzv. NP-úplná úloha. To znamená, že so vzrastajúcim počtom miest rastie čas riešenia úlohy minimálne exponenciálne, čo vedie k tomu, že algoritmus riešenia nemusí poskytovať v reálnom čase optimálne riešenie pre väčší počet premenných.

A. Problém Obchodného Cestujúceho

Problém obchodného cestujúceho (TSP) – je jednou z najznámejších optimalizačných úloh, s cieľom nájsť najkratšiu cestu medzi súborom miest, ktoré je potrebné navštíviť [7]. Z matematického pohľadu je nájsť hamiltonovský cyklus s minimálnym súčtom hodnôt hrán v ohodnotenom grafe za podmienky, že každú hranu navštívi iba raz. Tento typ problému je možné riešiť tak, že sa preskúmajú všetky hamiltonovské kružnice a vyberie sa z nich tá, ktorá má minimálny počet hodnôt hrán. Problém je v tom, že tých kružníc môže byť veľmi veľa. TSP sa využíva v prevádzkach pri plánovaní dopravných úloh, pri plánovaní výroby a riadení dodávateľského reťazca. Riešenie tohto problému môže znížiť výrobné náklady a všetky náklady spojené s plánovaním chodu určitých procesov. Na riešenie tohto problému sme si vybrali 2 metódy.

Algoritmus najbližšieho suseda napriek svojej jednoduchosti pomerne presný a modely na jeho základe generujú optimálne výsledky [8], [9]. Tento algoritmus bol jedným z prvých, ktoré boli použité na vyriešenie problému obchodného cestujúceho. Princíp: k dispozícii množinu údajov, z ktorých každá má číselné hodnoty. Ak je počet prvkov n , môžeme vrcholy reprezentovať ako body v n -rozmernej mriežke. Pri novom vrchole vypočítame vzdialenosť od vrcholu k všetkým ostatným vrcholom v množine. Vyberáme k -najbližších susedov.

Littleho metóda patrí do skupiny metód vetiev a hraníc [10], [11]. Ide o optimalizačnú a kombinatorickú metódu riešenia, ktorá preskúma každé možné riešenie. Ak existuje n miest, cez ktoré má prejsť obchodný cestujúci, a ak sú všetky mestá navzájom poprepájané ako kompletný graf, potom existuje $n!$ rôznych okružných ciest.

B. Prípadová Štúdia

U dopravcov sa plánovanie dodávok distribučnej siete spravidla vykonáva manuálne rutinne bez objektívneho vedeckého kritéria. Okrem toho je táto činnosť založená najmä na skúsenostiach dispečera, ktorý musí mať dobrú znalosť geografie územia, ktoré má byť obsluhované. Sieťou TOTAL sa rozumie skupina čerpacích staníc TOTAL prepojených spoľahlivou cestnou sieťou, ktorých plánovanie činnosti je organizované okolo dvoch uzlových bodov, ktoré sú skladmi Sep Konga (mesto Kinshasa) [12]. V prípade, ak by si spoločnosť vybrala tento spôsob riešenia distribúcie pohonných hmôt s cieľom minimalizácie nákladov, musí sa počítať s viacerými cisternami, ktoré by sa poslali na rozvoz pohonných hmôt. Vo všeobecnosti algoritmus, ktorý bol použitý v tomto prípade je podobný Dijkstrov algoritmu s tým rozdielom, že pracuje aj so zápornými hodnotami hrán. Ak sa použije záporná hodnota, Dijkstrov algoritmus môže zlyhať a pri nájdení najkratšej cesty z jedného bodu do druhého, nemusí prejsť všetkými bodmi, ktoré sú definované na začiatku plánu. Jeho cieľ je len nájsť najkratšiu trasu z bodu A do bodu B. Oba algoritmy sú obľúbené a často používané pri hľadaní najkratšej cesty, ale nedokážu nájsť najkratšiu cestu pre jedného dopravcu s podmienkou navštívenia všetkých naplánovaných čerpacích staníc.

II. VLASTNÉ RIEŠENIE

V našom riešení sme využili dva hore popísané algoritmy. Výstupom je funkčný prototyp vo forme webovej aplikácie. Základom je zoznam destinácií, ktoré zadal používateľ. Z týchto destinácií sa vytvorí graf, v ktorom sú všetky body poprepájané navzájom medzi sebou. Pri vytváraní hrán medzi jednotlivými bodmi sa hneď zisťuje aj vzdialenosť medzi nimi pomocou Google API. Keď je vytvorený graf, tak nasleduje samotné hľadanie optimálnej cesty:

Algoritmus najbližšieho suseda:

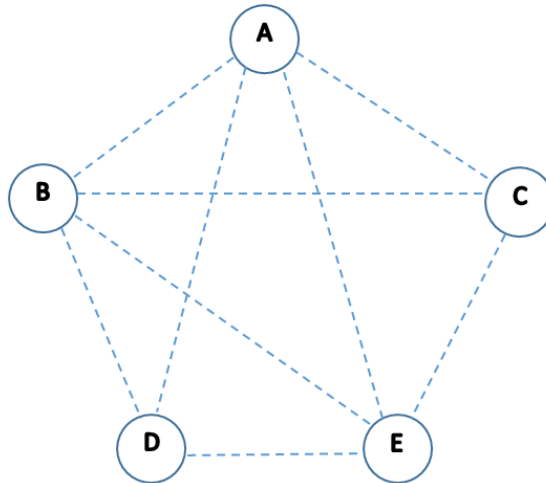
1. Vyber z grafu vrchol ako východiskový bod.
2. Z východiskového bodu choď do vrcholu po hrane s najmenšou vzdialenosťou. Ak existuje viac ako jedna možnosť, vyber náhodne.
3. Pokračuj v budovaní okruhu, vždy po jednom z vrcholov, ktoré ešte neboli navštívené.
4. Zapíš výslednú postupnosť s celkovou dĺžkou trasy.

Littleho metóda:

1. Algoritmus zoberie zoznam zadaných vrcholov.
2. Odstráni počiatočný a koncový vrchol, lebo tie sú vždy na začiatku a na konci – ich pozícia v hľadanej trase je nemenná.

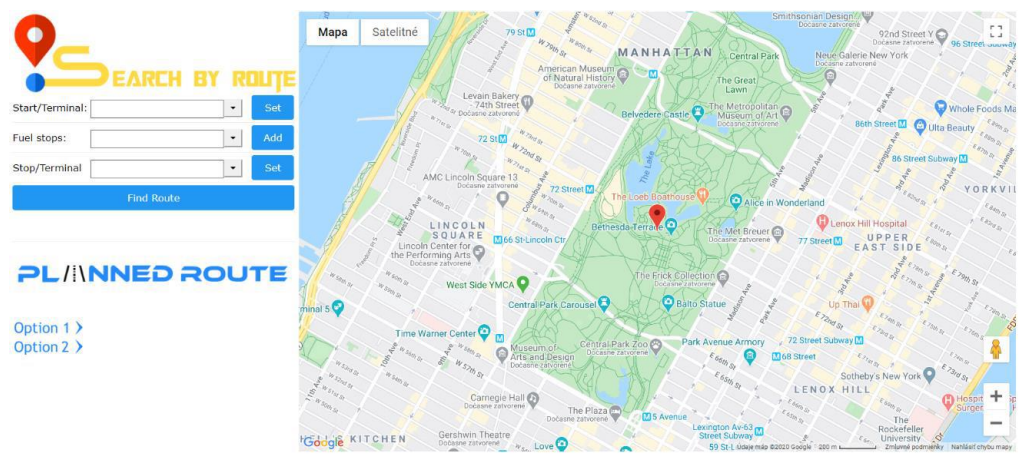
3. Zo zvyšných bodov vytvorí všetky možné kombinácie, teda konkrétne permutácie všetkých vrcholov, okrem začiatku a cieľa, bez opakovania.
4. Ku všetkým týmto kombináciám sa naspäť pripoja počiatočný a koncový bod a spočítajú sa vzdialenosti medzi všetkými vrcholmi.
5. Výsledkom bude kombinácia vrcholov s najkratšou vzdialenosťou. Ak by takých kombinácií bolo viac, vyberie sa prvá v poradí.

Graf na Obr.1 znázorňuje, ako môžu vyzerat' prepojenia medzi jednotlivými bodmi. Hrany sú znázornené čiarkovane, lebo daná hrana môže, ale aj nemusí existovať. Napríklad prepojenie medzi bodmi A a E môže existovať priamo, ale tiež môže prechádzať cez bod C prípadne B alebo BD, BCD alebo inú kombináciu bodov.



Obr. 1 Graf vytvorený zo zadaných destinácií.

Funkčný prototyp obsahuje interaktívnu mapu na zaznamenávanie východiskového a cieľového bodu (Obr.2). Option 1 zobrazí výstup algoritmu najbližšieho suseda; Option 2 zobrazí výstup Littleho metódy. Výstup sa zobrazí na mape a v tabuľke v poradí, v akom má používateľ navštíviť naplánované miesta. Používateľ si vyberie trasu, ktorá viac vyhovuje jeho požiadavkám.



Obr. 2 Prototyp webovej aplikácie využívajúci dva algoritmy na optimalizáciu trasy.

III. ZÁVER

Jedným zo spôsobov ako znížiť variabilné náklady je optimalizácia prepravy tovaru a produktov prepravnej logistiky. Na tento účel sme využili dva algoritmy, ktoré riešia problém obchodného cestujúceho. Okrem nami vybraných existujú aj ďalšie algoritmy, ktoré sa snažia priblížiť optimálnemu riešeniu. Väčšina odborníkov sa však domnieva, že ho nebude možné nikdy riešiť polynomiálnymi algoritmi kvôli jeho zložitosti.

REFERENCIE

- [1] R. Trudeau, *Introduction to Graph Theory*. Dover Books on Mathematics, Courier Corporation, 1993.
- [2] T.H. Cormen, et al. 22.3. *Introduction to Algorithms*. MIT Press. 2009.

- [3] E. Shimon, *Graph Algorithms*. (2nd ed.), Cambridge University Press, 2011, pp. 46-48.
- [4] C. L. Mallows, N. J. A. Sloane, *Two-graphs, switching classes and Euler graphs are equal in number*. SIAM Journal on Applied Mathematics 1975, 28(4), pp. 876-880.
- [5] K. Appel, W. Haken, *Every planar map is four colorable*. Part I. Discharging, Illinois J. Math. 21, 1977, pp.429-490.
- [6] S. Skiena, *Hamiltonian Cycles*. §5.3.4 in *Implementing Discrete Mathematics: Combinatory and Graph Theory with Mathematica*. Reading, MA: Addison-Wesley, 1990, pp. 196-198.
- [7] *The Traveling Salesman Problem*. In: *Combinatorial Optimization. Algorithms and Combinatorics*, vol 21. Springer, Berlin, Heidelberg
- [8] D.S Johnson, L.A McGeoch, *The Traveling Salesman Problem: A Case Study, Local Search in Combinatorial Optimization*. John Wiley & Sons, 1997, pp. 215-310.
- [9] D.L Applegate, R.E. Bixby, V. Chavatal, W.J. Cook, *The Travelling Salesman Problem, A Computational Study*. Princeton Univesity Press, Princeton, 2006.
- [10] J. D.C. Little, K. G. Murty, D. W. Sweeney, C. Karel, *An Algorithm for the Traveling Salesman Problem*. In: *Operations Research*, 1963, 11(6), pp. 972-989.
- [11] A. Cipleu, W. Ștefănescu, I. Vandici, A.I. Gönczi, *A New Method to Improve Computing Time for TSP Little Algorithm*. In: Dumitru I., Covaciu D., Racila L., Rosca A. (eds) *The 30th SIAR International Congress of Automotive and Transport Engineering. SMAT 2019*. Springer, Cham.
- [12] Y. N. Vially, *Optimization of the Fuel Distribution Business by Application of the Bellman-Kallaba Algorithm: A Case study of the Network of the TOTAL Company in Kinshasa-DRC*. *International Journal of Advances in Scientific Research and Engineering*, 2019, 5(10), pp. 271-287.

Párovanie Duplicitných a Takmer Identických Obrázkov Pomocou Percepčného Hashovania

¹Patrik TUŽÁK, ²Ladislav PERDUK, ¹František BABIČ

¹ Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovensko

² bart.sk, Košice, Slovensko

patrik.tuzak@student.tuke.sk, perduk@bart.sk, frantisek.babic@tuke.sk,

Abstrakt — Tento článok sa zaoberá porovnaním rôznych funkcií percepčného hashovania na identifikáciu duplicitných alebo takmer identických obrázkov. Hashovacia funkcia je výpočtovo efektívna funkcia, ktorá mapuje binárne reťazce ľubovoľnej dĺžky na binárne reťazce určitej pevnej dĺžky, ktoré nazývame hashovacie hodnoty. Pre naše porovnanie sme vybrali šesť hashovacích funkcií: aHash, dHash, dhash2, myHash, pHash a wHash. Na porovnanie dvoch percepčných hashov sme použili Hammingovú vzdialenosť. Na každý obrázok sme aplikovali niekoľko transformácií: pridanie vodoznaku, zmena kontrastu, zmena jasu, preklopenie o 90 stupňov, zmena farieb na šedo tónový obraz, a zmena rozmerov obrázka. Dosiahnuté výsledky potvrdili, že percepčné hashovanie je rýchlym a dostatočne robustným riešením pre identifikáciu duplicitných alebo takmer identických obrázkov.

Kľúčové slová — percepčné hashovanie, transformácie, výpočtová náročnosť

I. ÚVOD

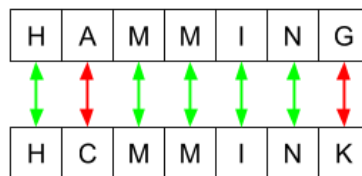
Rozpoznávanie obrazu funguje u počítačov inak ako u ľudí. Ľuďom sa celé roky vyvíjali schopnosti rozoznávať a pomenovať objekty. Keď sa pozerú na akýkoľvek obrázok, vedia popísať, čo sa na obrázku nachádza, aký má tvar, farbu. Ak aj zmeníme niektoré pixely na obrázku, otočíme ho, zmeníme veľkosť alebo jas, prípadne pridáme nejaký vodoznak, stále dokážeme pri pohľade na objekt popísať jeho vlastnosti. Pri počítačovom videní je obrázok definovaný pomocou matice čísel. Ak sa zmení jeden pixel na obrázku, a už to bude pre počítač iný obrázok. Ak by sme zmenili jas obrázku a tým upravili všetky jeho hodnoty, bolo by možné, že pre počítač by bol podobnejší s úplne iným obrázkom ako predtým.

Hashovanie predstavuje proces, kde na vstupné dáta aplikujeme príslušnú hashovaciu funkciu a výstupom je tzv. hash. Vo všeobecnosti sa pri hashovaní aj veľmi malá zmena na vstupe prejaví ako výrazná zmena na výstupe, čo je takzvaný lavínový efekt. Je to však opozitum toho, čo chceme dosiahnuť, ak chceme pomocou hashovania zistiť, či sú obrázky podobné.

Podľa [1] je percepčné hashovanie proces, pri ktorom sa hodnota hashu konštruuje na základe vizuálneho obsahu obrázka. Môžeme sa s ním stretnúť pri obsahovo založenom vyhľadávaní informácií alebo pri reverznom vyhľadávaní podľa obrázka. Pri tomto type hashovania sa zachováva pomer zmeny hashu a zmeny obrázka.

Pri identifikácii duplicitných alebo veľmi podobných obrázkov sme sa zamerali primárne na ich štruktúru a nezvažovali sme detaily. Nízke frekvencie zobrazujú štruktúru objektov, vysoké frekvencie zobrazujú detaily. Čím menší je obrázok, tým menej detailov zobrazuje, čiže prevládajú nízke frekvencie.

Na výpočet podobnosti dvoch hodnôt percepčných hashov sa zvyčajne používa funkcia Hammingovej vzdialenosti. Podľa autora [2] môžeme približné párovanie reťazcov opísať ako textový reťazec $T=t_1, t_2, \dots, t_n$, vzorec $P = p_1 p_2 \dots p_n$ a celé nezáporné číslo k , pre ktoré platí $k \leq m \leq n$. My hľadáme všetky výskyty podreťazca X v reťazci T , pričom vzdialenosť $D(P, X)$ medzi vzorom P a reťazcom X je menšia alebo rovná k (Obr.1).



Obr. 1 Ukážka fungovania výpočtu Hammingovej vzdialenosti (2).

A. Analýza súčasného stavu

Úprava fotografií je dnes oveľa jednoduchšia a sofistikovanejšia ako v minulosti. Dôsledkom rôznych nenáročných, dostupných a voľným okom nerozoznatelných manipulácií s obrázkami môže byť ich nižšia dôveryhodnosť. Tejto problematike sa vo svojej práci [3] venovali Weng a Preneel. Nemohli však použiť obyčajnú kryptografickú hashovaciu funkciu, ktorá aj pri malej zmene bitov alebo rozlíšenia silno reaguje na výsledok. V takom prípade by museli urobiť rôzne transformácie a každý vstupný obrázok zvlášť zahashovať, čo by bolo veľmi náročné. Preto použili metódu závislú od obsahu obrázka tolerujúcu skreslenie, ktorá zachováva obsah bez ohľadu na digitálnu formu. Na základnom obrázku vykonali 3 transformácie: zmena objektu, odstránenie objektu z obrázka a pridanie objektu na obrázok. Obrázok upravili na určitú veľkosť a previedli ho na šedo tónový obraz. Na získanie vstupov do hashovacej funkcie použili 3 rôzne metódy, a to magnitúdu z Fourierovej transformácie, hlavné body z vlnkovej transformácie a nakoniec algoritmus, ktorý vychádzal z momentov vyšších rádov. Z výsledkov zistili, že modifikovaný obsah sa v konečnom dôsledku zdal autentickjší ako ten pravý. Z toho vyplynulo, že percepčné hashovanie v takejto forme nebolo vhodným riešením pre túto úlohu. Autori však navrhli framework, ktorý extrahoval informácie z obsahu obrázka a tie následne kombinoval s tajným kľúčom, aby vytvoril percepčný hash. Týmto spôsobom potom dokázali chrániť aj percepčne nevýznamné informácie.

Thaiyalnayaki a Sasikala vo svojej práci [4] nerozoznávajú duplikáty alebo takmer identické obrázky, ale všetky obrázky jednoducho označili ako duplicitné. Následne použili hashovanie citlivé na lokalitu (LSH). Táto metóda hľadá najbližších susedov v rámci príslušnej dátovej sady. Používa na to rôzne metriky ako Euklidovská, Manhattanská, Jaccardová alebo Hammingová. Autori pracovali so vstupnými dátami predstavujúcimi 20 obrázkov. V ďalšom kroku tieto obrázky upravili tak, aby boli výsledky vhodnejšie na ďalšiu analýzu a identifikáciu kľúčových vlastností, napríklad: odstránenie šumu, zaostrenie alebo rozjasnenie. Na základe výsledkov dospeli k záveru, že vďaka postupu, pri ktorom extrahujú kľúčové vlastnosti obrázkov, následne ich indexujú a hľadajú duplikáty, je tento prístup vysoko efektívny a reálne by mohli efektívne identifikovať duplikáty aj pri niekoľkých tisícoch obrázkov.

II. NÁVRH RIEŠENIA

V našej práci sme sa zamerali na identifikáciu duplikátov obrázkov rôznych produktov z rôznych kategórií. Tieto obrázky dokonca pochádzali z rôznych zdrojov, kde sa často stretávame s tým, že sú v inej kvalite, inom rozlíšení, sú otočené alebo je na nich pridaný vodoznak.

Všetky obrázky sme zahashovali a navzájom porovnávali pomocou Hammingovej vzdialenosti s prahovou hodnotou nastavenou na 2. Tento postup však priniesol problém, kedy mohli byť identifikované ako duplikáty dva rôzne obrázky, vid'. Obr.2. V tomto prípade šlo z pohľadu počítačového videnia o podobné obrázky, keďže majú podobnú textúru, biely okraj a aj podobné smerovanie hrán. Ak by sme použili vhodný detektor hrán, napríklad od Cannyho [5], zistili by sme, že majú aj podobné hrany. Tento problém sme vyriešili tak, aby jednotlivé produkty v datase mali aj informáciu o kategórii, t.j. párovali sa len obrázky z rovnakej kategórie, napríklad skrine.



Obr.2 Počítačovo podobné produkty.

III. TESTOVANIE

Na testovanie sme vybrali vzorku viac ako 150 jedinečných príkladov produktov z rôznych kategórií. Na každý obrázok sme aplikovali nasledovné transformácie: prídanie vodoznaku, zmena kontrastu, zmena jasú, preklopenie o 90 stupňov, zmena farieb na šedo tónový obraz, a zmena rozmerov obrázka (Obr.3).



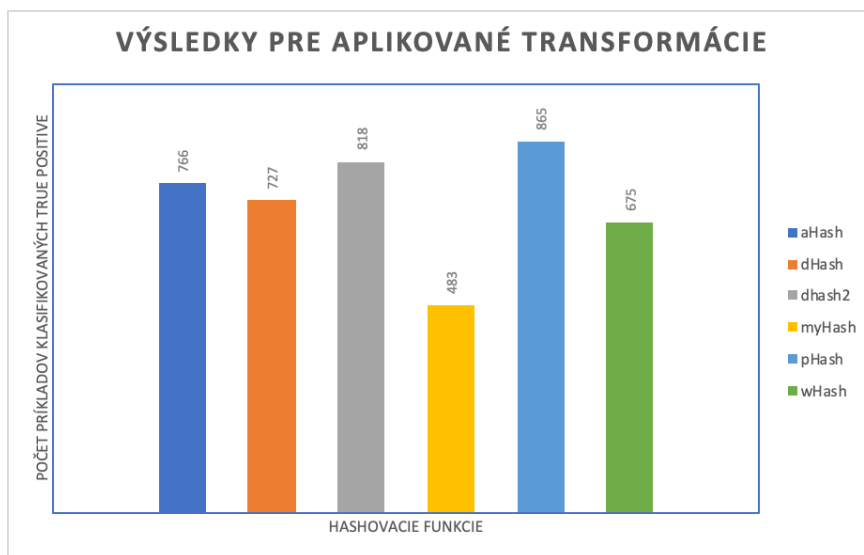
Obr.3 Ukážky aplikovaných transformácií.

Pri každej transformácii a hashovacej funkcii sme vybrali takú hraničnú hodnotu Hammingovej vzdialenosti, ktorá mala pre danú úlohu najlepšie výsledky. Mohli nastať štyri rôzne stavy:

- Obrázok bol klasifikovaný ako pár a naozaj to bol pár – *True Positive*.
- Obrázok nebol klasifikovaný ako pár a naozaj to nebol pár – *True Negative*.
- Obrázok bol klasifikovaný ako pár, ale nebol to pár – *False Positive*.
- Obrázok nebol klasifikovaný ako pár, ale bol to pár – *False Negative*.

Zamerali sme sa na čo najlepšiu klasifikáciu *True Positive* v kombinácii s čo najmenšou hodnotou *False Positive*. Na Obr.4 sú zobrazené výsledky klasifikácie *True Positive* rozdelené podľa jednotlivých hashovacích funkcií. Najlepšie výsledky pri testovaných transformáciách dosiahla hashovacia funkcia *pHash* z balíka *ImageHash*. Veľmi dobré výsledky priniesla aj funkcia *dhash* z balíka *dhash*. Celkovo sa percepčné hashovanie ukázalo ako vhodný nástroj na identifikáciu duplicitných alebo takmer identických obrázkov rôznych produktov. Jediná

transformácia, pri ktorej hashovacie funkcie nedosahovali očakávané výsledky, bolo otočenie obrázka o 90 stupňov.



Obr.4 Sumárne výsledky testovaných hashovacích funkcií pre všetky transformácie.

V prípade že sa na jednom z obrázkov nachádza vodoznak, najlepšie výsledky priniesla hashovacia funkcia pHash z balíka ImageHash. Pri našom testovaní sme ako najvhodnejšiu hraničnú hodnotu pri tejto funkcii zvolili Hammingovu vzdialenosť 6.

Pri 3-násobnej zmene kontrastu dosiahla najlepšie výsledky rovnaká funkcia s hraničnou hodnotou Hammingovej vzdialenosti nastavenou na 8.

Keď sme hodnotu jas nastavili na 50% pôvodnej hodnoty, tak rovnako dobré výsledky dosiahli funkcie pHash (5) aj dhash (13).

Výsledky pri použití inej palety farieb potvrdili účinnosť percepčného hashovania. Takmer všetky hashovacie funkcie dosahli maximálne dobré výsledky na testovacích dátach.

Rovnako aj pri zmene rozmerov obrázka najlepšiu možnú výkonnosť sme zaznamenali pri dhash, dHash, aHash a pHash.

Na testovanie výkonnosti hashovacích funkcií sme použili čas trvania príslušných experimentov. Z tohto pohľadu dosiahla najlepšie výsledky funkcia myHash, ktorá však nedosahovala vysoké hodnoty presnosti. Dosiahnuté časy boli spôsobené štruktúrou tejto funkcie, ktorá bola najjednoduchšia zo všetkých použitých, t.j. bol potrebný najmenší počet kalkulácií. Zahashovať všetky testovacie obrázky v tomto prípade trvalo 0.0774s. V prípade funkcie s najlepšimi výsledkami klasifikácie pHash to bolo 2.7439s, čo je prijateľný čas pre takýto typ úlohy. Najhoršie výsledky dosiahla funkcia wHash s trvaním 20.0339s.

IV. ZÁVER

Z dosiahnutých výsledkov vyplýva, že percepčné hashovanie je rýchlym a dostatočne robustným riešením pre identifikáciu duplicitných alebo takmer identických obrázkov. Väčšina použitých hashovacích funkcií priniesla dostatočne rýchlu a správnu identifikáciu nielen duplicitných obrázkov ale aj aplikovaných transformácií. Problematickým bolo len otočenie obrázka, kde by bolo vhodným riešením zahashovať obrázky s rôznym otočením a potom porovnávať tieto hodnoty hashov.

REFERENCIE

- [1] A. Buldas, A. Kroonmaa, R. Laanoja, *Keyless Signatures' Infrastructure: How to Build Global Distributed Hash-Trees*. In Riis, Nielson H.; Gollmann, D. (eds.). *Secure IT Systems*. NordSec 2013. Lecture Notes in Computer Science 8208, 2013.
- [2] B. Melichar, *Approximate string matching by finite automata*. In: Hlaváč V., Šára R. (eds) *Computer Analysis of Images and Patterns*. CAIP 1995. Lecture Notes in Computer Science 970, 1995. Springer, Berlin, Heidelberg
- [3] L. Weng, B. Preneel, *Attacking Some Perceptual Image Hash Algorithms*. 2007 IEEE International Conference on Multimedia and Expo, 2007, Beijing, China.
- [4] S. Thaiyalnayaki, P.J. Sasikala, *Identifying and Indexing Near-Duplicate Images Using Optimizing Technique in Web Search*. *International Journal of Innovative Science, Engineering & Technology* 2016, 3(7), pp.538-546.
- [5] J. Canny, *A Computational Approach To Edge Detection*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

Vývoj Inovatívneho Pluginu pre Redakčný Systém

Peter DAŇKO, František BABIČ

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Letná 9, 042 00 Košice, Slovensko

peter.danko@student.tuke.sk, frantisek.babic@tuke.sk

Abstrakt — Cieľom je vytvoriť plnohodnotné rozšírenie pre redakčný systém WordPress s funkciami pre malé a stredné realitné kancelárie na základe identifikovaných požiadaviek. Tento plugin ponúkne základnú platformu pre vývojárov, pomocou ktorej môžu pokračovať vývoj očakávaného riešenia na mieru. Týmto spôsobom je možné ušetriť pomerne dosť času a financií, ktoré sa môžu použiť na zložitejšie funkcionality. Navrhnuté riešenie nebude vyžadovať žiadny výkon navyše a teda na nainštalovanie a normálny chod úplne stačia požiadavky, ktoré má samotný WordPress. Proces vývoja bol manažovaný v súlade s modelom životného cyklu vodopád. Okrem vhodného typu obsahu, polí či taxonómií ponúka plugin veľmi intuitívne prostredie, pochopiteľné aj pre laika, ktorý s WordPress platformou ešte neprišiel do styku.

Kľúčové slová — redakčný systém, plugin, realitná kancelária

I. ÚVOD

Wordpress predstavuje bezplatný redakčný systém, ktorý sa vyvinul z pôvodne blogovacieho systému [1]. Aktuálne je najrozšírenejším redakčným systémom, podľa verejných informácií je základom pre vyše tretinu celého Internetu¹. Samozrejme, existujú vývojárske tímy, ktoré svoje produkty pre WordPress predávajú. Takéto platené doplnky veľakrát dotvárajú veľmi komplikované funkcionality, za ktoré by si inak museli koncoví zákazníci zaplatiť oveľa viac, ak by si ich dali zakaždým pripraviť na mieru.

Prostredníctvom dotazníkového prieskumu a osobných konzultácií sme identifikovali reálny problém – pre malé a stredné realitné kancelárie je problematické zaobstarat' si web, ktoré by splnil ich základné požiadavky, používateľské nároky a cena nebola príliš vysoká.

A. Analýza Problému

Na začiatku sme veľký dôraz kládli na podrobnú analýzu problému a potrieb koncových používateľov – zamestnancov a majiteľov realitných kancelárií. V prvom kroku sme pripravili dotazník so všeobecnými otázkami, aké problémy majú s aktuálnou webstránkou, aký redakčný systém [2] využívajú a podobne. Na základe odpovedí sme vytvorili niekoľko hypotéz, ktoré sme si overili na workshopoch priamo v realitných kanceláriách. Okrem toho sme zisťovali a identifikovali jednotlivé modely fungovania týchto firiem, najmä spojené s uverejňovaním inzerátov a správou webovej stránky. Najdôležitejšie zistenia predstavovali: používateľom sa nepáči napr. veľký počet pluginov či sliderov, príliš mnoho vyskakujúcich reklám, chýbajúce typy obsahu, inzeráty spravujú priamo administrátori príslušných webov. Veľmi dôležitým krokom bola analýza konkrétnych typov obsahu, konkrétnych polí, informácií, ktoré by mali obsahovať napr. inzeráty či iný obsah. Na definovanie reprezentantov cieľových skupín používateľov sme použili koncept persón [3].

B. Analýza Súčasného Stav

V rámci analýzy súčasného stavu sme si nainštalovali väčšinu dostupných pluginov redakčného systému Wordpress určených pre realitné kancelárie. Zamerali sme sa na viaceré aspekty, napríklad funkcie vo voľne dostupnej verzii, typy obsahu, spomalenie načítania webu, kritické chyby v administrácii či kóde, ale tiež údaje ako posledná aktualizácia, hodnotenia, počtu stiahnutí a podobne, nakoľko sa developeri či používatelia veľakrát rozhodujú aj na základe týchto údajov (viď Tabuľka 1). Táto tabuľka obsahuje výsledky pre 4 najlepšie hodnotené

¹ https://w3techs.com/technologies/overview/content_management

pluginy, ktoré ale nespĺnili plnohodnotne identifikované požiadavky zo strany cieľovej skupiny. Toto zistenie potvrdilo našu motiváciu pre návrh a vývoj nového riešenia.

Tabuľka 1
 Hodnotová analýza porovnaných riešení

Kritérium	Váha	Cherry Real Estate		Estatik		IMPress Listings	
		Hodnota	Výsledok	Hodnota	Výsledok	Hodnota	Výsledok
Spomalenie webu	0,2	6	1,2	5	1	6	1,2
Počet stiahnutí	0,07	1	0,07	4	0,28	8	0,56
Úroveň hodnotení	0,1	4	0,4	8	0,8	3	0,3
Posledná aktualizácia	0,05	0	0	4	0,2	2	0,1
Funkcie vo free verzii	0,18	6	1,08	1	0,18	3	0,54
Vhodné typy obsahu	0,2	5	1	4	0,8	5	1
Vhodné taxonómie	0,13	2	0,26	4	0,52	2	0,26
Viacero dizajnov	0,07	0	0	2	0,14	3	0,21
Spolu			4,01		3,92		4,17

C. Persóny

Persóny predstavujú fiktívnych reprezentantov identifikovaných cieľových skupín. V našom prípade sme identifikovali 3 kľúčové persóny:

- Realitný maklér (viď Obr.1)- spravidla 25 až 35 rokov, vo viac ako polovici prípadov žena, ktorá je platená na základe provízie z predajov, resp. úspešných ponúk. Jeho motiváciou je čo najlepšie odprezentovať ponuku. V závislosti od nastavenia firmy môže agent (maklér) spravovať ponuky na webe a nahradiť tak administrátora (jeho agenda sa rozdelí medzi programátora webu a agentov).



Obr. 1 Persóna – realitný maklér.

- Programátor webových stránok - freelancer, ktorý sa venuje tvorbe webových stránok niekoľko rokov. Má 25 až 40 rokov, študuje alebo pracuje v IT odvetví. So svojimi klientmi komunikuje pravidelne, keďže poskytuje k svojim projektom aj základnú údržbu. Tento programátor chce mať prístup k celému kódu, čo zabezpečuje GPL-2 licencia pod ktorou beží celý redakčný systém WordPress a teda aj navrhované rozšírenie. Potrebuje kvalitnú dokumentáciu, aby vedel plugin meniť podľa konkrétnych požiadaviek svojich klientov.
- Administrátor webu - plní v realitnej kancelárii základné technické zabezpečenie (základné nastavenie siete, tlačiarňí, počítačov a podobne). Okrem toho sa stará o pridávanie, editáciu a mazanie editorov obsahu (agentov) a rieši základné nastavenia webu ako napr. kontaktné údaje. Má 18-30 rokov a s webom sa najčastejšie dostáva do styku pri pridávaní či úprave ponúk alebo odpisovaní na správy z webstránky. Zabezpečuje aj základnú administratívu a komunikáciu s developerom webu.

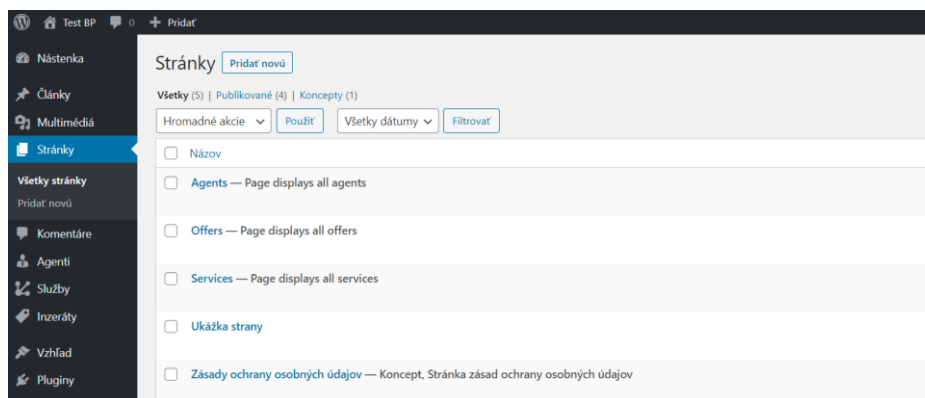
II. VÝVOJ VLASTNÉHO PLUGINU

V procese vývoja sme postupovali v súlade s modelom životného cyklu vodopád [4], t.j. analýza, návrh, implementácia, testovanie a nasadenie/údržba.

Na základe vytvorených persón sme definovali súbor používateľských požiadaviek, ktorým sme prideliťli prioritu pre implementáciu pomocou metódy MoSCoW: must have, should have, could have a won't have.

Pre plugin sme vytvorili aj základnú identitu pozostávajúcu z názvu, loga a grafického kitu (farby a fonty). Zvolili sme živé farby ktoré sa spájajú s domovom, aby čo najlepšie odrážali účel pluginu. Základnou farbou bude zelená, pretože je „farbou krajiny, spája sa s pocitovým chápaním, s pocitom byť prospešný a nápomocný.

Prvým krokom bolo vytvorenie vlastných typov obsahu ako agenti, inzeráty a služby (viď Obr.2). Následne sme implementovali taxonómie, čo sú v podstate vlastnosti jednotlivého obsahu podľa ktorých je ich možné filtrovať (resp. ich môžeme definovať aj ako „vlastné kategórie“). Opäť sme využili znalosti z výskumnej fázy, t.j. taxonómie budú definované len v rámci inzerátov, konkrétne ide o stav nehnuteľnosti, typ nehnuteľnosti a typ inzerátu. Každá taxonómia obsahovala aj základné hodnoty (napr. pre typ inzerátu prenájom a predaj), hodnoty si však môže používateľ voľne vytvárať či odstraňovať.

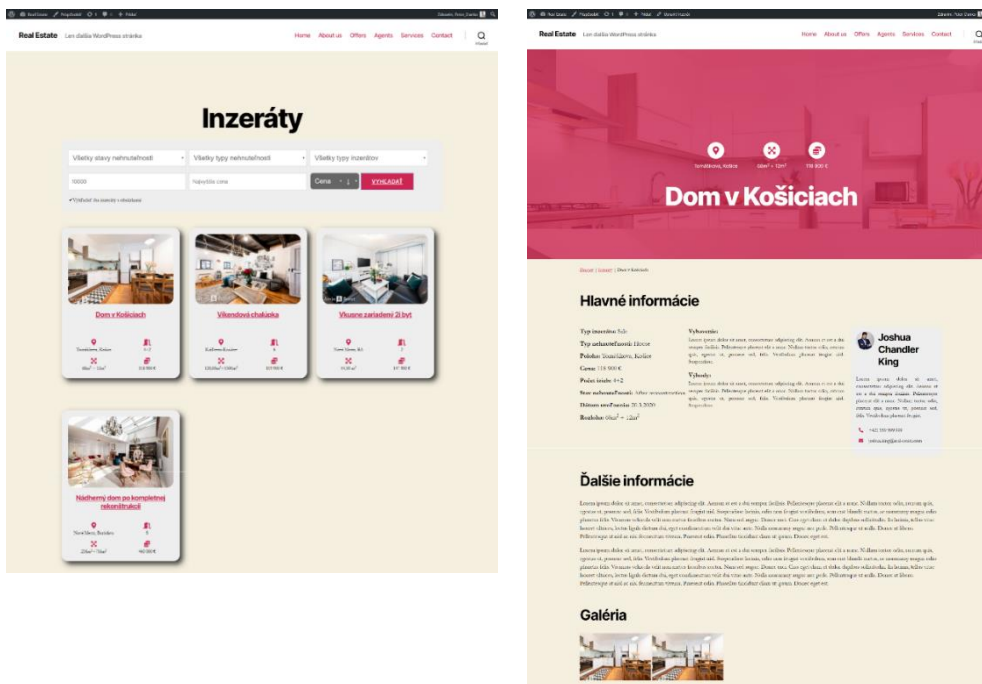


Obr. 2 Ukážka vytvorených vlastných položiek pre plugin.

Druhým krokom bolo vytvorenie šablón. Vytvorili sme archívne stránky pre všetky tri typy obsahu (archívna stránka je taká, kde sa zobrazuje výpis všetkých článkov daného typu); okrem toho aj detail inzerátu. Práve šablóna pre inzeráty bola náročnejšou a to z dôvodu filtrovania ponúk (na archívnej stránke). Následne sme ešte riešili kompatibilitu so základnými WordPress témami a to tým spôsobom, že pre každú tému sa volá iná hlavička súboru, zodpovedajúca danej téme. V šablóne potom testujeme aktívnu tému a voláme priradenú hlavičku. Okrem toho bola šablóna priradená jednotlivým typom obsahu.

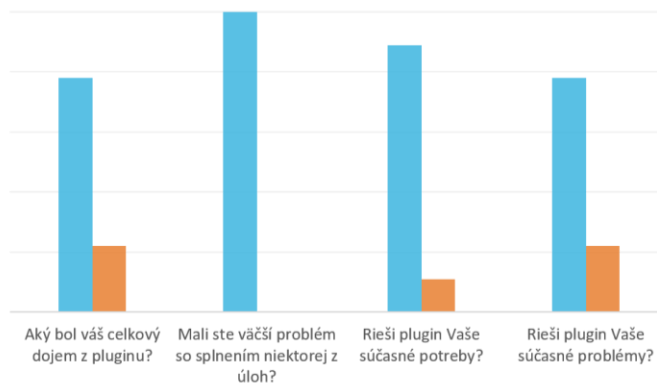
Tretím krokom bola implementácia používateľskej stránky pluginu v administrátorskej časti redakčného systému Wordpress. Následne sme implementovali preklad pluginu ako doplnkovú funkciu s možnosťou prepínať medzi slovenským a anglickým jazykom. Jazyk pluginu sa automaticky prepína spolu so zmenou jazyka administrácie redakčného systému.

Nový plugin poskytuje automatizáciu viacerých krokov, aby sme čo najviac odbremenili programátora a používateľov (viď Obr.3). Preto sa pri aktivácii pluginu okrem vlastných typov obsahu vytvoria aj podstránky s popisom tak, aby používateľovi bolo jasné, na čo slúžia.



Obr. 3 Ukážka vizualizácie inzerátov pomocou nového pluginu.

Vo fáze testovania sme zamerali najmä na použiteľnosť riešenia pre realitné kancelárie z hľadiska administrácie a používania riešenia v každodennej prevádzke. Keďže vytvorenie riešenia môže vytvoriť základ pre vývoj ďalších funkcionalít, nezávisle sme testovali priamo zdrojový kód z pohľadu kvality a modulárnosti., aby bol čo najlepšie rozšíriteľný (modulárny) o funkcie, či položky na mieru. Z výsledkov testovania používateľmi vyberáme:



Obr. 4 Ukážka výsledkov z dotazníka (áno – modrá farba, nie – oranžová farba).

III. ZÁVER

Našou motiváciou bolo priniesť na trh jednoduchý a účelný WordPress plugin pre malé a stredné realitné kancelárie. Jeho použitím tieto kancelárie ušetria čas a financie, ktoré môžu využiť na doplnenie zložitejších funkcií alebo fungovanie firmy. Návrh pluginu bol založený na komplexnej analýze súčasného stavu a výsledkoch intenzívneho používateľského výskumu. Výsledky testovania len potvrdzujú dôležitosť tohto kroku v modeli životného cyklu vývoja softvérového produktu.

POUŽITÉ ZDROJE

- [1] J. Cabot, *WordPress: A Content Management System to Democratize Publishing*. IEEE Software 2018, 35(3):89-92.
- [2] J.M. Martinez-Caro, A.J Alejo-Hernandez, a kol., *A Comparative Study of Web Content Management Systems*. Information 2018, 9(2), 27.
- [3] L. Nielsen, *Personas in Use. In: Personas - User Focused Design*. Human-Computer Interaction Series. Springer, 2013.
- [4] W. Royce, *Managing the Development of Large Software Systems*. ICSE '87: Proceedings of the 9th international conference on Software Engineering, 1987, pp. 328-338.

Porovnanie zariadení pre akumuláciu elektrickej energie pre domácnosť

¹Branislav VANGOR, ²Dušan MEDVEĎ

¹² Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹branislav.vangor@student.tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — V súčasnosti narastá čoraz viac spotreba elektrickej energie. Obnoviteľné zdroje nám ponúkajú množstvo energie najmä počas dňa, keď to najmenej potrebujeme. Akumuláciou vieme túto energiu uložiť v rôznych formách a využiť neskôr keď ju potrebujeme. Tento článok sa venuje popisu vybraných možností akumulácie energie a porovnaniu či sú vhodné pre použitie v domácnosti.

Kľúčové slová — akumulácia elektrickej energie, akumulčné systémy, batérie;

I. ÚVOD

V posledných rokoch stúpla spotreba energie po celom svete a to najmä v ekonomicky vyspelých krajinách, ako aj v rozvojových krajinách. V rámci zvýšenia spotreby, enormne narástol dopyt po elektrickej energii. Ďalším významným faktorom je zvyšovanie počtu obyvateľstva a zvyšujúca sa životná úroveň, ktorá súvisí s ekonomickým rastom. Veľká časť spotreby energie je pokrytá z fosílnych energetických zdrojov, pritom ich dostupný potenciál sa značne znižuje, ich cena zase rastie nepretržite a stále rýchlejšim tempom. Postupne narastá úloha alternatívnych energetických zdrojov ako sú napríklad obnoviteľných zdrojov energie. Okrem toho čelíme v dôsledku využívania fosílnych palív s environmentálnymi problémami už teraz. Výroba elektrickej energie z obnoviteľných zdrojov energie je pomerne ťažko predikovateľná a kolíše počas dňa. Hoci je v obnoviteľných zdrojoch veľký potenciál, nevieme ho dostatočne využiť. Toto je jednou z hlavných nevýhod výroby elektrickej energie z obnoviteľných zdrojov energie, preto je potrebné sa zaoberať čoraz viac myšlienkou akumulácie elektrickej energie. Možnosti akumulácie energie sú rôzne, pričom nie je možné akumulovať priamo elektrickú energiu, ale ju ukladať ako neelektrickú energiu, ktorú v budúcnosti využijeme ako elektrickú energiu.

II. AKUMULÁCIA ENERGIE

Elektrická energia sa využíva vo všetkých oblastiach ľudskej činnosti. Ľudstvo sa stalo závislé od jej aktuálnej ponuky. V elektrárňach, kde sa elektrická energia vyrába, sa pomocou prenosovej a distribučnej sústavy prenáša k miestu spotreby a k spotrebiteľom. Elektrárne musia svoju výrobu prispôbovať aktuálnej spotrebe. Uskladnenie elektrickej energie vo väčších množstvách je ťažké. Ukladanie energie obsahuje premenu energie z foriem, ktoré sa ťažko skladujú, na pohodlnejšie alebo ekonomicky uskladniteľné formy. Žiadna akumulácia nie je bezstratová, pri každom druhu akumulácie dochádza k určitým stratám energie. Akumuláciou elektrickej energie na inú formu a jej spätné využitie vo forme elektrickej energie umožňuje skutočnosť, že už nie je potrebné dodržiavať rovnováhu medzi výrobou a spotrebou elektrickej energie.

Akumulácia energie je taký proces, v priebehu ktorého určitý druh energie sa ukladá za účelom jeho využitia na vykonávanie užitočnej práce v budúcnosti. Pre tento účel je v súčasnosti možné využiť širokú škálu akumulčných zariadení, ktoré keďže slúžia na ukladanie energie, zvyknú sa označovať aj pojmom energetické úložiská.

Energiu treba uskladňovať v zásade v dvoch prípadoch. V prvom prípade je k dispozícii zdroj energie, ktorý ale nemá dostatočný výkon na pokrývanie energetických nárokov určitého dynamického procesu rýchleho priebehu. Iným prípadom je, keď energia nie je k dispozícii v takom množstve, ako to vyžaduje odberateľ.

Dopyt po elektrickej energii sa mení v rôznych časoch, má rôznu dennú, týždennú, ročnú, či sezónnu charakteristiku.

Pre skladovanie elektrickej energie môžu byť vhodné rôzne typy energií:

- **Mechanická**
 - stlačený vzduch (CAES)
 - zotrvačníky
 - prečerpávacie vodné elektrárne
- **Elektrická**
 - superkapacitor (EDLC)
 - magnetické supravodiče (SMES)
- **Tepelná**
 - uloženie merného tepla
 - uloženie latentného tepla
- **Elektrochemická**
 - vodík (palivový článok)
 - tradičné akumulátory (Pb, NiCd, NiMh)
 - moderné akumulátory s vnútornou akumuláciou (Li ion, NaS)
 - moderné prietokové kyselinové akumulátory (VRB, ZnBr)

Pre jadrovú elektráreň, ktorá je finančne nákladná, je najlepšie, aby pracovala na konštantnom výkone. V noci, keď odberatelia spotrebujú asi polovicu energie ako počas dňa, sa výkon elektrárni musí upravovať. Počas noci máme prebytok elektrickej energie, a preto je táto nočná elektrina lacnejšia. Prebytočný výkon v sústave pomáha spotrebovať prečerpávací elektráreň, ktorá čerpá vodu z dolnej nádrže do hornej nádrže.

Batérie patria v súčasnosti k často využívaným prostriedkom pre ukladanie energie. Elektrická energia sa v týchto zariadeniach mení na chemickú. Následne, sa uložená chemická energia mení na elektrickú. Batéria sa skladá z jedného alebo viacerých elektrochemických článkov [1].

Využívanie batérií sa postupne obmedzovalo. Avšak vďaka novým technológiám a materiálom, batérie sa začínajú opäť používať. V elektrochemických článkoch sa na rozdiel od kondenzátorov energia uchováva vo forme chemickej zlúčeniny a nie ako kladné a záporné elektrické náboje v kondenzátore. Nabíjanie a vybíjanie je v porovnaní s kondenzátormi pomalšie, ale hustota energie môže byť až trikrát väčšia [2].

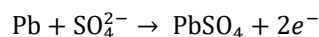
III. MOŽNOSTI AKUMULÁCIE

A. Elektrochemické spôsoby akumulácie

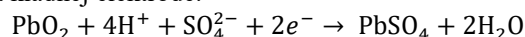
1) Olovený akumulátor

Olovený akumulátor bol vyvinutý v roku 1859. Patrí medzi najstaršie a najčastejšie používané elektrochemické zariadenia. Olovený akumulátor pozostáva z olovených elektród a elektrolytu. Elektrolyt tvorí vodou riedená kyselina sírová H_2SO_4 v koncentrácii 35 %. Elektrolyt môže byť nasiaknutý do vaty zo sklenených vlákien (AGM batéria) alebo vo forme gélu (gélová batéria). V nabitom stave aktívnu hmotu zápornej elektródy tvorí olovo a kladnej elektródy oxid olovičitý PbO_2 . Vybíjaním sa aktívna hmotnosť elektród mení na síran olovnatý $PbSO_4$ a elektrolyt je ochudobnený o kyselinu sírovú a obohatený o vodu. Pri vybíjaní sa teda znižuje koncentrácia elektrolytu a pri nabíjaní narastá [3], [4].

Chemická reakcia na zápornej elektróde:



Chemická reakcia na kladnej elektróde:



Obr. 1 Olovený akumulátor v reze

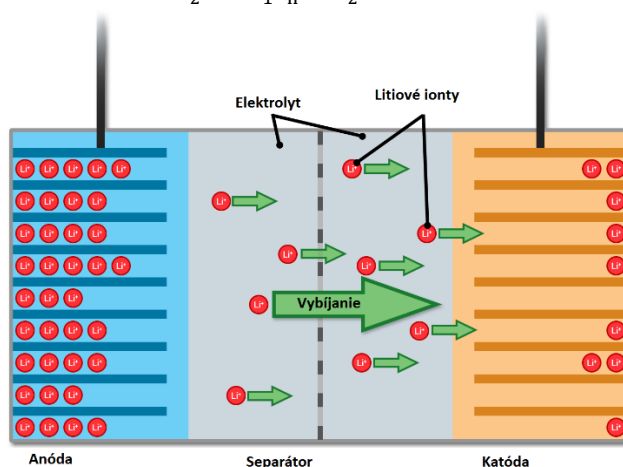
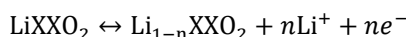
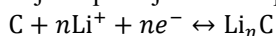
2) Lítium-iónový akumulátor

Prvé lítiové batérie boli použité pre elektroniku, v súčasnosti sa používajú v elektrických vozidlách. Využívajú sa v širokom spektre aplikácií na uloženie energie, od batérií s niekoľkými kWh v obytných systémoch až po sústavy batérií na poskytovanie doplnkových služieb. Lítium-iónové bunky sa vyrábajú vo valcovom alebo obdĺžnikovom tvare. Tieto bunky sú spájané do modulov v sérii alebo paralelne [5].

Katóda je vyrobená z oxidu kovu lítia, ako je napríklad LiMO_2 a LiCoO_2 , anóda je vyrobená z grafitového uhlíka. Elektrolyt je tvorený bezvodnou organickou kvapalinou, ktorá obsahuje rozpustené lítiové soli, napríklad LiClO_4 . V súčasnosti sa vyrábajú v dvoch variantoch a to Li-ion a Li-pol [1].

Na slovenskom trhu sa najčastejšie používajú práve Li-ion batérie (87%), iba 8% používa novší typ Li-pol, ktoré majú nižší objem a hmotnosť.

Reakcie, ktoré prebiehajú na kladnej a zápornej elektróde počas nabíjania/vybíjania:



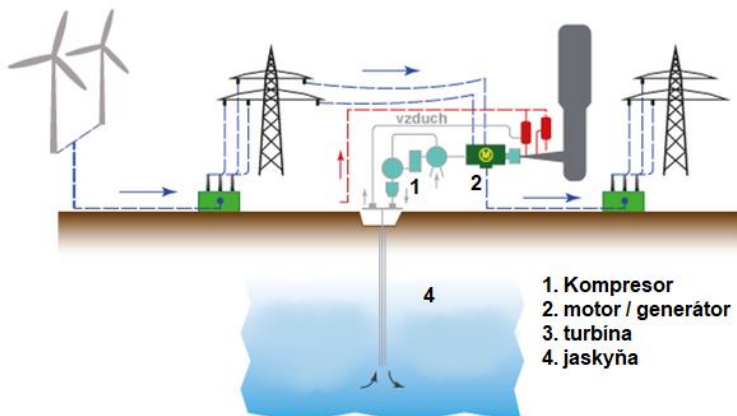
Obr. 2 Princíp činnosti Li-ion akumulátora

Lítiovo-polymérové akumulátory sú najnovším typom, majú vylepšenú technológiu jadra, z tohto dôvodu sú tenšie a ľahšie pri zachovanej kapacite. Majú ale vysokú cenu, preto sa používajú iba pri špičkovej elektronike. Do bežných mobilov či notebookov sa nepoužívajú, pretože ich použitím by došlo k zvýšeniu cien daných elektronických zariadení (hodnota batérie predstavuje 5 až 20 % z ceny zariadenia).

B. Mechanické spôsoby akumulácie

1) Stlačený vzduch

Skladovanie elektrickej energie stlačeným vzduchom (CAES – compressed air energy storage) je jeden z možných spôsobov, ako uložiť elektrickú energiu pre potrebu v inom čase. V čase nízkeho dopytu sa elektrická energia odoberá a ukladá formou stlačeného vzduchu, ktorého potenciálna energia je potom v čase špičky premieňaná naspäť do elektrickej formy a následne využívaná, čo umožňuje doceliť uspokojenie dopytu.



Obr. 3 Schéma CAES

Princíp činnosti je podobný ako u prečerpávacích vodných elektrární. Rozdiel je iba v médiu, s ktorým pracujú. V čase malého odberu sa v zariadení CAES vzduch alebo iný plyn ukladá pod tlakom do jaskyne alebo iného tlakového zásobníka. Ak chceme využiť naakumulovanú energiu, stlačený vzduch sa zahrieva a v expanznej turbíne expanduje, kde roztočí turbínu, ktorá je na spoločnom hriadeli s generátorom. Na uloženie väčšieho množstva energie slúžia ako zásobníky opustené bane či iné vhodné skalné útvary. Pre uskladnenie menšieho množstva energie sa používajú tlakové nádoby. Ak je podzemný zásobník prepojený s vodnou hladinou na povrchu zeme, je to z dôvodu udržania konštantného tlaku v zásobníku [5], [6].

IV. POROVNANIE ZARIADENÍ PRE AKUMULÁCIU

Na porovnanie zariadení bol vytvorený jednoduchý program v prostredí MS Excel. Titulnú stranu programu pre zadávanie vstupných informácií môžeme vidieť na Obr. 4. Pre určenie jednotlivých zariadení bol zvolený rodinný dom, blízko Košíc. Úžitková plocha domu je 105,6 m². V dome je priemerná denná spotreba elektrickej energie 5 kWh. Na ohrev teplej úžitkovej vody, vykurovanie a varenie, je použitý zemný plyn. Elektrická energia v dome je využívaná na osvetlenie a chod ostatných bežných spotrebičov v domácnosti. Akumulácia elektrickej energie bude zabezpečená z fotovoltaickej elektrárne (FVE). V programe je možnosť zvoliť si z dvoch ponúkaných výkonov 1,5 kWp a 2,3 kWp pre FVE. Program na základe zvolenej možnosti určí cenu za zvolenú fotovoltaickú elektrárňu. Tlačidlom „Ukáž údaje FVE“ je možné si pozrieť základné údaje fotovoltaiky. V ďalšom kroku si užívateľ vyberá z možností dlhodobé a krátkodobé využitie uloženej energie. Následne si už môže vybrať akumuláčn é zariadenie zo zoznamu, ktorý je závislý od jeho výberu v predchádzajúcom kroku. Po zvolení zariadenia, si používateľ môže zobrazit' technické údaje akumuláčného zariadenia a informácie pre užívateľa kliknutím na tlačidlo „Technické údaje“.

Program na výpočet akumuláčných zariadení	
<i>Legenda</i>	
Vybrať hodnotu z rolovacieho menu	
Prepojenie na iný hárok	
Importované hodnoty z iných buniek	
Typ budovy	Rodinný dom
Z čoho budeme akumulovať	Fotovoltaika
Fotovoltaický systém	HYBRID
Denná spotreba [kWh]	5
Maximálny výkon panelov [kWp]	1,5
Cena s DPH za FVE [€]	3614
Technické údaje FVE	Ukáž údaje FVE
Čas využitia	Dlhodobé využitie
Akumuláčn é zariadenie	LiFePO4 2,4kWh
Technické údaje zariadenia	Technické údaje

Obr. 4 Ukážka titulnej strany programu pre určenie akumuláčného zariadenia

Ukážka technických údajov a informácií pre používateľa je znázornená na Obr. 5. Informácie pre užívateľa obsahujú informatívne časy počas, ktorých dokáže zariadenie dodávať elektrickú energiu pre domácnosť a čas, za ktorý sa zariadenie nabije. Tieto časy majú iba informatívny charakter, pretože čas, ktorý dokáže zariadenie zásobovať domácnosť elektrickou energiou je vypočítaný pre celkové vybitie zariadenia. Nie je vhodné zariadenia vybiť na 100 %. Výrobca zariadenia udáva počet cyklov pri určitom vybití. Ak sa tento údaj prekročí, klesá životnosť zariadenia. Tento vypočítaný údaj je iba ukazovateľ, aké množstvo energie dokáže zariadenie uchovať pre človeka neznalého tejto problematiky. Čas, ktorý je potrebný k nabitíu je vypočítaný pre optimálne podmienky výroby elektrickej energie z fotovoltaickej elektrárne, vtedy je

dosiahnutý maximálny nabijací prúd, ktorý je závislý od použitej fotovoltiky. Ideálne podmienky máme dosiahnuté počas slnečných letných dní.

 **Návrat na úvodnú stranu**

Technické špecifikácie BMZ ESS X Li-Ion 48V 186Ah 10kWh

Všeobecné vlastnosti	ESS X
Energia (menovitá/reálna)	10,6 kWh/8,05 kWh
Menovité napätie	54 V
Maximálne napätie	61,5 V
Minimálne napätie	45,0 V
Kapacita (menovitá/reálna)	186,3 Ah/ 149,1 Ah
Maximálny prúd pri nabíjaní	90 A
Maximálny prúd pri vybití	300 A (3 sekundy)
Maximálny výkon vybitia	18 kW
Hmotnosť	99 kg
Rozmery ŠxVxH (mm)	638 x 421 x 487
Komunikácia SMA	CAN
Hĺbka vybitia	80%
Počet cyklov	5000



INFORMÁCIA PRE POUŽÍVATEĽOV:

- Teplota článkov pri vybití: 2 ° C do 45 ° C
- Teplota článkov pri nabíjaní: 2 ° C do 45 ° C
- Odporúčaná teplota akumulátora: 10 ° C do 25 ° C
- Samovybitie článkov: cca 2% ročne
- Maximálny počet paralelne spojených akumulátorov: 12 (povinný dodatočné vybavenie)
- Trieda krytia: IP 21
- Označenie CE: áno
- UN test 38.3: áno

Cena **4300 € bez DPH**
 5160 € s DPH

Informácie pre užívateľa



Kapacita použitej batérie je rovná 10 kWh

Doba použiteľnosti pri 100% nabití: **1 deň 14 hodín***

Doba nabíjania na plnú kapacitu: **6 hodín 30 minút****

Výhody:
Veľmi vysoká hustota energie - 200 Wh / kg. .
Relatívne vysoká kapacita s malým objemom a hmotnosťou.
Takmer žiadne samovybitie (do 5%).
Nemá pamäťový efekt.
Životnosť 5000 nabíjajúcich cyklov.

Nevýhody:
Nebezpečenstvo výbuchu alebo vznietenia pri nesprávnom používaní
Vadí jej úplné vybitie.
Recyklácia je zatiaľ veľmi obtiažna a nákladná.

* počítané s hodnotou DOD 80%, kedy výrobca udáva životnosť 5000 cyklov
** uvedený údaj má iba informačný charakter, počítalo sa s maximálnym prúdom nabíjačky, ktorý je dosiahnutý iba pri optimálnych podmienkach

Obr. 5 Ukážka technických údajov a informácii pre užívateľa

Pri použitých akumuláčnych zariadeniach bola pri niektorých typoch uvedená kapacita batérie v ampérhodinách, pri iných bola uvedená energia vo wathodinách. Pri výpočte bolo preto potrebné počítať s energiou a pri iných typoch, bolo potrebné uvažovať s kapacitou batérie. Preto bolo potrebné prepočítavať jednotlivé hodnoty.

A. Príklad prepočtu kapacity batérie na množstvo uloženej energie

Pre výpočet času využitia bolo potrebné poznať uloženú energiu v batérii. Ako príklad je nižšie uvedený výpočet pre olovený akumulátor 230 Ah s napätím 12 V.

Postup výpočtu:

$$230 \text{ Ah} \cdot 12 \text{ V} = 2760 \text{ Wh} = 2,76 \text{ kWh} \quad (1)$$

B. Príklad prepočtu uloženej energie na kapacitu batérie

Pre výpočet doby nabíjania akumuláčného zariadenia bolo potrebné poznať jeho kapacitu. Ako

príklad je nižšie uvedený výpočet pre batériu LiFePO₄ s energiou 3,6 kWh a napätím 48 V.

Postup výpočtu:

$$\frac{3600 \text{ Wh}}{48 \text{ V}} = 75 \text{ Ah} \quad (2)$$

Ak je známe množstvo uloženej energie a kapacita akumuláčného zariadenia, je možné stanoviť dobu, počas ktorej dokáže domácnosť fungovať z akumuláčného zariadenia. Priemerná uvažovaná denná spotreba elektrickej energie v domácnosti je 5 kWh. Príklad, pre olovený akumulátor 230 Ah, 2,76 kWh, 12 V:

$$t_{\text{využitia}} = \frac{2,76 \text{ kWh}}{5 \text{ kWh}} = 0,552 \text{ dňa} = 13 \text{ h } 15 \text{ min} \quad (3)$$

Prepočet energie na kapacitu akumulátora bol dôležitý práve pre výpočet času nabíjania akumuláčného zariadenia. Príklad je uvedený pre olovený akumulátor 230 Ah a nabíjanie bude prebiehať z FVE 1,5 kWp, kde je nabíjací prúd 40 A.

$$t_{\text{nabíjania}} = \frac{(Q \cdot 1,4)}{I_{\text{nab}}} = \frac{(230 \text{ Ah} \cdot 1,4)}{40 \text{ A}} = 8,05 \text{ h} = 8 \text{ h } 3 \text{ min} \quad (4)$$

kde: Q kapacita akumulátora [Ah]

I_{nab} prúd nabíjačky [A]

V. ZHRNUTIE

Tento príspevok sa zaoberal akumuláciou elektrickej energie v rôznych formách. Vo vytvorenom programe je umiestnených spolu 11 zariadení pre akumuláciu energie: olovený akumulátor 230 Ah, AGM akumulátor 230 Ah, li-ion akumulátor v troch možnostiach 6,8 kWh, 8,5 kWh a 10 kWh. Nasleduje zotrvačnickové úložisko, vanádiová redoxová batéria rôznych výkonov a množstvom energie, ktoré dokážu uložiť energiu s nasledovným využitím (P , E) 5 kW 20 kWh, 5 kW 40 kWh, 10 kW 75 kWh. Ako posledné zariadenie je použitý lítium železo fosfátový akumulátor 2,4 kWh a 3,6 kWh. Pri výbere akumuláčného zariadenia je potrebné prihliadať nielen na cenu daného zariadenia, ale aj na životnosť akumuláčného zariadenia. Ako príklad uvádzame olovený akumulátor, kde výrobca udáva 400 cyklov pri 40 % vybití a li-ion akumulátor ponúka až 5000 cyklov pri 80 % vybití. Batéria typu li-ion je v porovnaní s oloveným akumulátorom značne drahšia, li-ion akumulátor stojí v priemere 31 eur za 1 Ah kapacity, pričom olovený akumulátor stojí iba 1,4 eura za 1 Ah kapacity. Li-ion akumulátor má až 12-krát väčší počet cyklov ako olovený akumulátor. Ako najlepšie zariadenie zo všetkých vychádza vanádiová redoxová batéria, ktorá umožňuje vybitie až 100 %, ale vyžaduje si väčšiu investíciu oproti iným zariadeniam. Vo vytvorenom programe vyšlo ako nevhodné zotrvačnickové úložisko z dôvodu, že je potrebné vstupné napätie 400 V AC a zariadenie dokáže dodávať elektrickú energiu iba 15 až 30 sekúnd. Vývoj ide stále dopredu a teda aj technológie akumulácie elektrickej energie sa posúvajú stále dopredu. Čo v dnešnej dobe nie je možné zaobstarať pre určité bezpečnostné riziká (napr. supravodivé magnetické akumulovanie, ktoré je náročné na chladenie), p pár rokov môže byť bežne dostupné v menších výkonoch, ktoré budú vhodné pre použitie aj v domácnosti.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18.

LITERATÚRA

- [1] Kolcun, M., Vojtek, M., „Výskum rôznych možností využitia energetických úložísk v elektrizačnej sústave“. 1. vydanie Košice: TUKE, 2018. ISBN 978-80-553-2711-2.
- [2] Murtinger, K. „Ukladání elektriny z fotovoltaických a větrných elektráren“, 2011. [online] [cit. 2020-05-20]. Dostupné na internete: <<https://www.nazeleno.cz/energie/energetika/ukladani-elektriny-z-fotovoltaickych-a-vetnych-elektraren.aspx>>.
- [3] National natural science foundation of China, „Progress in Natural Science: Materials International“. 2011. [online] [cit. 2020-05-18]. Dostupné: <<https://www.sciencedirect.com/science/article/pii/S100200710800381X>>.
- [4] „Olovený akumulátor“. [online] [cit. 2020-05-18]. Dostupné na internete: <<https://cs.wikipedia.org/wiki/Oloven%C4%9Bn%C3%BDakumul%C3%A1tor>>.
- [5] Energieportal, „Skladovanie energie – všetko, čo potrebujete vedieť o uskladňovaní elektriny“. [online] [cit. 2020-05-20]. Dostupné na internete: <<https://www.energie-portal.sk/Dokument/skladovanie-energie-vsetko-co-potrebujete-vediet-o-uskladnovani-elektriny-104204.aspx>>.
- [6] Petro, Š., „Dôvody a možnosti akumulácie elektrickej energie v súvislosti s rozšírením obnoviteľných zdrojov energie a porovnanie dostupných technológií“. [online] [cit. 2020-05-19]. Komárno: Projektstav Komárno s.r.o., 2013. Dostupné na internete: <<http://www.gesc.sk/wp-content/uploads/2014/09/Dovody-a-moznosti-akumulacie-elektrickej-energie-v-suvlosti-s-rozsirenim-obnovitelnych-zdrojov-energie-a-porovnanie-dostupnych-technologii.pdf>>.
- [7] Vangor, B., „Porovnanie zariadení pre akumuláciu energie pre domácnosť“, Bakalárska práca. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2020.

Simulované prihrávanie lopty v robotickom futbale

Ján VAŠČÁK

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská Republika

jan.vascak@tuke.sk

Abstrakt – Obsahom tejto práce je poukázať na problematiku robotickej kooperácie. Ako testovacia úloha bola zvolená z oblasti robotického futbalu, konkrétne prihrávanie lopty medzi dvomi robotmi. Úloha bola zrealizovaná prostredníctvom simulačného prostredia *Webots*, kde sme zvolili model známeho robota *Khepera III*. V rámci riešenia tejto problematiky sa práca zaoberá tvorbou virtuálneho prostredia, jeho modelovaním ako aj návrhom algoritmu na navigáciu robota. Napokon sa práca venuje samotnej realizácii návrhu a simulácii vyvinutého riešenia. Následne sú popísané vykonané experimenty a ich vyhodnotenie.

KLúčové slová – kooperácia, riadenie, robot, simulácia

I. ÚVOD

Základnou podmienkou pre rozšírenie využitia robotov je ich schopnosť kooperovať, t.j. aby boli schopné medzi sebou vykonávať koordinovaný pohyb [1]. Ako vzorovou úlohou môže v tomto poslúžiť napr. prihrávanie lopty medzi sebou [2]. Za týmto účelom je potrebné zostrojiť systém, ktorý dokáže efektívne analyzovať stav dvoch robotov a lopty na hracom poli, a ktorý je schopný optimálne využiť tieto informácie za účelom robotov a nahrávania lopty medzi nimi.

V [3] je popísaný obrazový prístup, ktorý je jeden z najbežnejších prístupov pri projektovaní pohybu. Daná metóda definuje pohyby ako sadu sérií statických uhlov kĺbu. V [2] je obrazový prístup rozšírený o možnosť reagovať na vonkajšie zmeny prostredia. V danom systéme, sú kľúčové obrazy definované pomocou karteziánskych pozícií aktuátorov (kolesá, končatiny a pod.), ktoré sú následne interpolované. Spoločné body sú vypočítavané podľa inverznej kinematiky a Bézierových kriviek. Nevýhodou oboch prístupov je, že sa berie do úvahy len poloha lopty, podľa ktorej je kop vygenerovaný.

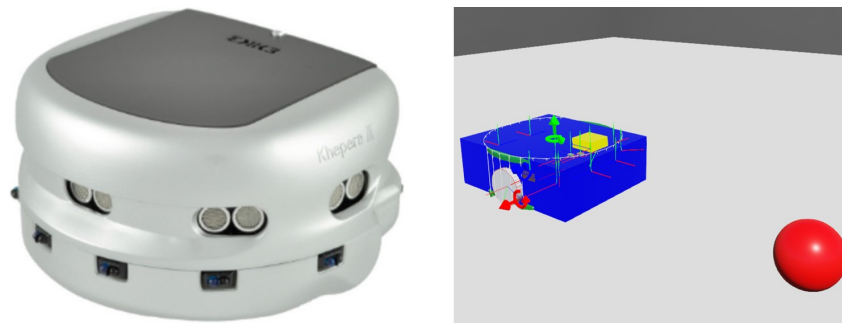
Vylepšením uvedených prístupov je metóda založená na polohe lopty a na základe ďalších informácií získaných z prostredia (napr.: poloha brány a poloha prekážok), ktoré slúžia ako vstupy do rozhodovacieho bloku, ktorý je v tomto prípade realizovaný pomocou tzv. *fuzzy kognitívnej mapy* a na výstupe sa získajú riadiace body pre inverznú kinematiku [4], [5].

II. POUŽITÉ PROSTRIEDKY

V nasledujúcej časti si opíšeme jednak reálne ako aj simulačné prostriedky použité pri realizácii koordinovanej prihrávky lopty.

Hlavným komponentom pri tvorbe projektu bol program *Webots* [6]. Ide o voľne dostupný, open-source, 3D robotický simulátor, ktorý našiel svoje uplatnenie v širokej sfére technických oblastí. Vďaka jeho prehľadnosti a množstvu dostupných modelov rôznych robotov je možné zvolený návrh podstatne zrýchliť. *Webots* umožňuje použiť celú radu programovacích jazykov ako napr. C, C++, Python, ROS, Java a MATLAB. Keďže je potrebné vzhľadom na náročnosť úlohy naprogramovať efektívny kód, tak voľba padla na jazyk C. Navyše poskytuje rozsiahlu podporu pri programovaní regulátorov používaných v reálnom svete. Takisto je to jazyk s najväčšou podporou v samotnom simulátore *Webots*, čo sa prejavuje v množstve príkladov vytvorených práve v tomto jazyku.

Za robotov nachádzajúcich sa na hracom poli boli zvolené malé kolesové roboty *Khepera III* (viď Obr. 1), ktoré sa vyznačujú hlavne ich jednoduchosťou a pohyblivosťou [7]. Nakoľko tieto roboty z hľadiska riadenia procesu prihrávania lopty nevykonávajú žiadnu úlohu, boli z nich využité len motorické a komunikačné prvky, t.j. pravý motor, ľavý motor a komunikačný prijímač. Kvôli jednoduchosti počas simulácie boli obidva roboty zabalené do obalu v tvare kvádra z dôvodu ich oblého tvaru a s ním spojenými problémami pri kopnutí do lopty.



Obr. 1 Robot *Khepera III* – reálny (vľavo) a nasimulovaný v prostredí *Webots* (vpravo).

Riadiacim a výpočtovým centrom celého procesu sa stal robot umiestnený nad hracou plochou. Tento robot pozostáva z kamery snímajúcej hraciu plochu a tiež z komunikačného vysielача. Z dôvodu prehľadnosti a jednoduchosti tomuto robotovi nebol priradený tzv. *Physics node*, čiže našho neplatia fyzikálne zákony a preto nie je nutné do prostredia dopĺňať žiadne oporné stĺpy. Táto možnosť je ekvivalentná s umiestnením robota napr. na strop ihriska v reálnom svete.

III. HLAVNÉ KOMPONENTY SIMULÁCIE

Realizácia tohto projektu sa uskutočnila vo viacerých fázach, ktoré na seba nadväzujú a sú závislé od úspešnosti ostatných.

Prostredie, v ktorom prebieha simulácia, je tvorené jednoduchou štvorcovou arénou *RectangleArena*, ktorá je súčasťou samotného simulátora *Webots*. Počas vývoja projektu boli na hraciu plochu umiestnené dva pásy slúžiace výlučne ako vizuálna pomôcka pre pozorovateľa simulácie a pre analýzu pozície ako aj pohyb robotov nemajú žiadny význam.

A. Rozpoznávanie objektov pomocou kamery

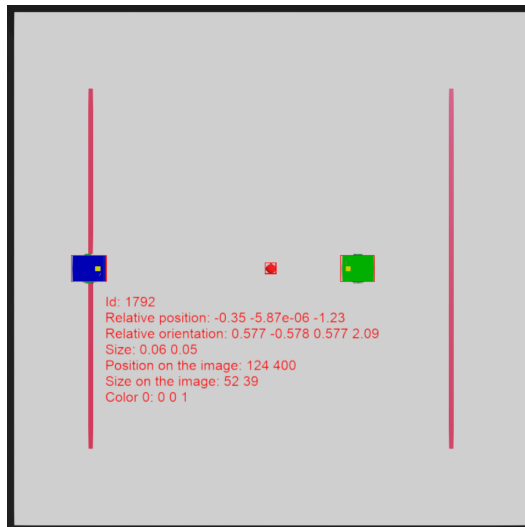
Za účelom zmysluplného riadenia robotov je potrebné najprv zistiť ich pozície a rotácie ako aj pozíciu lopty na hracej ploche. K tomu slúži kamera v podobe „tretieho“ robota, ktorá je umiestnená priamo nad stredom hracej plochy tak, že svojim zorným pol'om pokrýva celú hraciu plochu.

Samotné rozpoznávanie objektov je zrealizované pomocou funkcie kamery zvanej *recognition node*. Aby táto funkcia bola schopná rozoznať objekt v obraze kamery, je potrebné tomuto objektu nastaviť špecifickú farbu, na základe ktorej sa odlišuje od ostatných objektov. Funkcia *recognition node* je ďalej schopná analyzovať relatívnu pozíciu a rotáciu objektov vzhľadom k pozícii a rotácii kamery. Keďže je kamera umiestnená priamo nad stredom hracej plochy, vid' Obr. 2, jej horizontálne súradnice sú nulové, a tým pádom je rozoznaná pozícia objektu totožná s reálnou pozíciou v simulačnom svete, nakoľko pre naše účely bolo potrebné získať iba 2D súradnice na hracej ploche.

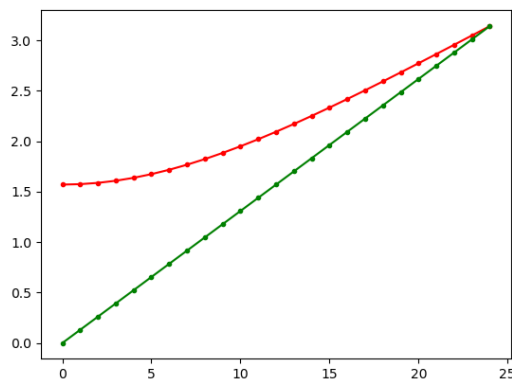
S rotáciou je to už komplikovanejšie, pretože rotácia objektu v prostredí *Webots* je definovaná pomocou osi rotácie (vektor $\{rx, ry, rz\}$) a uhla rotácie v radiánoch. Hoci kamera je schopná rozoznať rotáciu objektu, zaznamenaná os rotácie nie je totožná s požadovanou osou (zvislá os Y v simulačnom prostredí) a tiež nie je ani konzistentná. Z tohto dôvodu bola využitá iba posledná hodnota rotácie, a to samotný uhol rotácie v radiánoch. Experimentálne sa zaznamenalo 24 rôznych uhlov v rozmedzí $[0; \pi]$ a umelo sa vygenerovali pomocné hodnoty (priemer dvoch okolitých uhlov). Takto sa zdvojnásobil počet oporných hodnôt pre vyššiu presnosť. Následne každému nameranému uhlu bola priradená jeho skutočná hodnota v simulačnom svete a táto bola použitá ako výsledok analýzy rotácie objektu, Obr. 3.

B. Pohyb a riadenie robotov

Pohyb robotov na hracej ploche sa uskutočňuje pomocou troch funkcií: *go_forward*, *turn_robot* a *shoot*. Funkcia *go_forward* využíva ako vstupný parameter vzdialenosť v metroch, ktorú daný robot má prejsť z bodu A do bodu B . Na základe odometrie vypočíta najvyššiu možnú rýchlosť pri presune s čo možno najväčšou presnosťou. Funkcia *turn_robot* so vstupným parametrom uhla otočenia funguje na podobnom princípe. Na základe rýchlosti kolies za určitý čas prepočítava rozdiel orientácie robota pred a po otočení a na základe uhla, o ktorý sa má otočiť následne vypočítava najvyššiu rýchlosť otočenia s čo možno najvyššou presnosťou. Funkcia *shoot* má na starosti kopnutie loptičky smerom k druhému robotovi. Vychádza zo vzdialenosti medzi robotmi a na jej základe určuje rýchlosť, s ktorou narazí do loptičky.



Obr. 2 Obrázok kamery s použitím *recognition node*.



Obr. 3 Porovnanie rozloženia hodnôt rotácie – červená: hodnoty zaznamenané kamerou, zelená: požadované hodnoty.

Celé riadenie pohybu robotov sa odohráva na základe vstupných dát získaných z rozpoznávania obrazu. Tieto dáta v sebe zahŕňajú polohu a orientáciu obidvoch robotov a polohu loptičky. Pri prepočtoch predstavuje hracia plocha karteziánsku súradnicovú sústavu s nulovým bodom v centre hracej plochy.

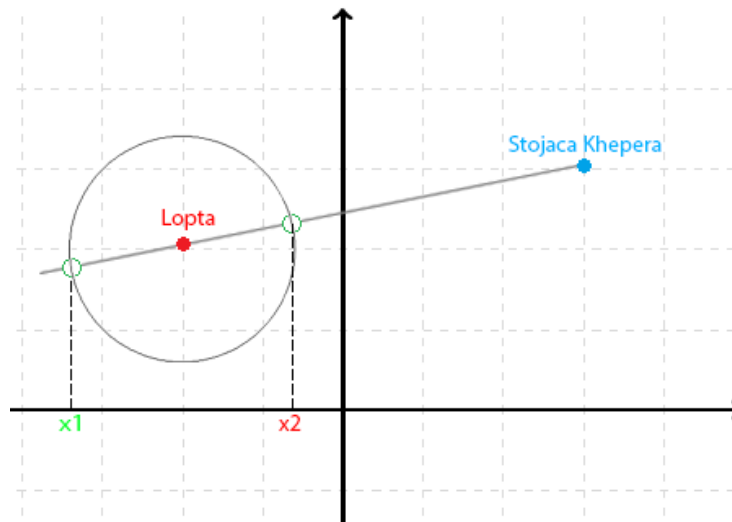
Po analýze vyplynulo, že pre pohyb robota bude postačovať týchto päť základných informácií:

- 1) identifikátor robota (obidva roboty používajú rovnaký regulátor),
- 2) prvé natočenie,
- 3) vzdialenosť pre pohyb smerom vpred,
- 4) druhé natočenie,
- 5) vzájomná vzdialenosť robotov (potrebná pre prepočet rýchlosti kopnutia).

Vzdialenosť loptičky a robota, potrebná pre výpočet vzdialenosti, ktorú má *Khepera* prejsť a tiež vzdialenosti dvoch robotov potrebné pre prepočet sily kopnutia sú počítané cez bežne používaný vzťah na výpočet vzdialenosti dvoch bodov v súradnicovej sústave.

Desired position je pozícia, na ktorú chceme robota za účelom kopnutia do lopty dostať. Je zrejme, že táto pozícia musí ležať na polpriamke spájajúcej stojací robot s loptičkou, no takisto sa musí nachádzať v určitej vzdialenosti od loptičky. Vychádzajúc zo všeobecnej rovnice lineárnej funkcie dostávame predpis priamky spájajúcej loptičku s *Kheperou*. Na vyjadrenie množiny bodov v určitej vzdialenosti od loptičky bola použitá rovnica kružnice a na získanie *desired position* bola vytvorená sústava týchto dvoch rovníc o dvoch neznámych. Keďže rovnica kružnice má kvadratický tvar, riešenie má dva výsledky, čo je zrejme aj z Obr. 4. Pre výber správneho riešenia boli porovnané x súradnice loptičky a tohto bodu.

Prvé a druhé natočenie vychádza z už spomínanej *desired position*. Pri výpočte prvého natočenia sme najprv zistili smernicu priamky spájajúcej kopajúci robot s loptičkou, následne toto číslo bolo prepočítané na uhol (funkcia \arctan), ktorý je zároveň požadovanou orientáciou robota. Výsledné



Obr. 4 Spôsob výpočtu požadovanej pozície.

natočenie robota predstavuje rozdiel tejto požadovanej orientácie od jej aktuálnej. Druhé natočenie bolo počítané podobným spôsobom s rozdielom v počítaní smernice priamky.

Pri klasickom prístupe, keď *Khepera* nahráje loptičku a ostane stáť na mieste, budú sa roboty postupne približovať k sebe, až nastane situácia, keď sa zrazia. Preto sme implementovali funkciu, ktorá má za úlohu dostať robota po nahratí lopty na náhodnú pozíciu vybranú podľa určitých pravidiel a do dostatočnej vzdialenosti od druhého robota. Po prihratí loptičky sa preto robot presunie na toto miesto a natočí sa smerom k ihrisku. Spôsob výpočtu sa od predchádzajúceho líši jedine vo výbere *desired position*, inak je celý proces rovnaký.

C. Komunikácia a prihrávanie lopty

Komunikácia medzi riadiacim centrom a robotmi prebieha na základe *RS_232* vysielača umiestneného na robotovi s kamerou a *RS_232* prijímača, ktorý sa nachádza na robotoch. Komunikácia prebieha v dvoch fázach a sa skladá z už spomínaných piatich základných informácií. V prvej fáze sa odošle prvé natočenie robota. Následne prebieha kontrola, či počas tohto otáčania robot narazil do loptičky. Ak áno, proces počká na zastavenie loptičky a následne opakuje celý prepočet pohybu robotov a opätovne odosiela informáciu o prvom natočení. Ak prebehla prvá fáza bez problémov, nasleduje druhá fáza, kde sa odosiela vzdialenosť, ktorú má robot prejsť, aby sa dostal na určené miesto, čiže druhé natočenie, ktoré natočí robota smerom k loptičke a druhému robotovi. Nakoniec sa odošle vzdialenosť medzi robotmi, ktorá je potrebná pre prepočet sily kopnutia.

Potreba rozdeliť komunikáciu na dve fázy vyplývala hlavne z nekonzistencie sily kopnutia a z občasných anomálií, pri ktorých bola loptička z neznámych fyzikálnych dôvodov kopnutá silnejšie ako obvyčajne a preto zastavila tesne vedľa robota, ktorý ju pri prvom otáčaní posunul. Keďže ide o dávkovú (batch) komunikáciu, tak posunutie loptičky pri prvom otáčaní spôsobilo rozladenie a zlyhanie celého systému.

Jednou z najkritickejších častí tohto projektu je kopnutie do loptičky, nakoľko ako už bolo skôr spomínané, pomer rýchlosti a vzdialenosti, ktorú loptička prejde, je nekonzistentný dokonca aj pri simulácií s rovnakým priebehom a parametrami. Preto sa občas stalo, že pri nižšej rýchlosti robota loptička prešla väčšiu vzdialenosť ako pri vyššej rýchlosti. Tento jav bol sprevádzaný jemným „zatrásením“ robota v momente úderu do loptičky. Nakoľko sa tento jav uskutočnil jedine v prostredí simulátora, je možné, že bol tento neprimerane silný kop spôsobený chybným výpočtom interakcie robota a loptičky a že v realite takýto problém nenastane. Avšak aj pri tomto obmedzení sme sa snažili daný algoritmus optimalizovať na čo možno najlepšie výsledky. Možnosťou takisto zostáva prídanie mechanizmu na odpaľovanie loptičky.

IV. ZÁVER

Uvedeným spôsobom bol vytvorený plne autonómny systém riadenia robotov, ktorých úlohou je spoločne prihrávanie lopty medzi sebou, pričom ich pozície a orientácie sú neustále kontrolované kamerou umiestnenou nad hracou plochou. Systém je taktiež do značnej miery odolný voči miernym anomáliám, ktoré počas behu programu môžu nastať. Týka sa to hlavne sily kopnutia loptičky, prípadne neúmyselného posunutia loptičky pri rotácii robota.

Do budúca jedným zo spôsobov vylepšenia tohto návrhu by mohla byť úprava jeho programovej stránky, t.j. zovšeobecnenie riešenia vzhľadom na to, že momentálne je toto riešenie výrazne obmedzené len na určitý počet a typ robotov. Taktiež čo sa týka zovšeobecnenia, by bolo vhodné upraviť rozpoznávanie rotácie robotov tak, aby nebolo závislé len na niekoľkých oporných bodoch ale aby bolo spojenou funkciou nameranej rotácie. V prípade implementácie spolaľhivého spôsobu kopnutia do lopty (napr. riadeným vystreľovacím modulom), kde by bolo možné hovoriť aj o presnosti strely nielen v dvoch dimenziách, by sa tento systém mohol stať základom pre vytvorenie plnohodnotnej verzie robotického futbalu.

POĎAKOVANIE

Táto práca je finančne podporená Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky prostredníctvom projektu KEGA č. 033 TUKE-4/2018 – Smart Industry/Architektúry inteligentných informačných a kybernetických systémov.

LITERATÚRA

- [1] G. Yasuda, "Behavior-based autonomous cooperative control of intelligent mobile robot systems with embedded Petri nets," in *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*, 2014, pp. 1085–1090.
- [2] J. Müller, T. Laue, and T. Röfer, "Kicking a ball – modeling complex dynamic motions for humanoid robots," in *RoboCup*, ser. Lecture Notes in Computer Science, vol. 6556. Springer, 2011, pp. 109–120.
- [3] T. Laue, T. Röfer, K. Gillmann, F. Wenk, C. Graf, and T. Kastner, "B-Human 2011 – eliminating game delays," in *RoboCup*, ser. Lecture Notes in Computer Science, vol. 7416. Springer, 2012, pp. 25–36.
- [4] J. Vaščák and K. Hirota, "Integrated decision-making system for robot soccer," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 15, no. 2, pp. 156–163, 2011.
- [5] J. Vaščák and R. Michna, "Learning of fuzzy cognitive maps by a PSO algorithm for movement adjustment of robots," in *Emergent Trends in Robotics and Intelligent Systems*, P. Sinčák, P. Hartono, M. Virčíková, J. Vaščák, and R. Jakša, Eds. Cham: Springer International Publishing, 2015, pp. 155–162.
- [6] Cyberbotics, "Webots: Open source robot simulator," [online] <https://www.cyberbotics.com/>.
- [7] F. Mondada, E. Franzi, and A. Guignard, "The development of Khepera," in *Proceedings of the First International Khepera Workshop*, ser. HNI-Verlagsschriftenreihe, Heinz Nixdorf Institut. 64, 1999, pp. 7–14.

Podpora kupónového marketingu prostredníctvom vhodných analytických metód

Jana JAKUBOVÁ, František BABIČ, Ludmila PUSZTOVÁ

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Letná 9, 042 00 Košice, Slovenská republika

jana.jakubova@student.tuke.sk, frantisek.babic@tuke.sk, ludmila.pusztova.2@tuke.sk

Abstrakt — V dnešnej digitálnej dobe, keď je pre ľudí pohodlnejšie, efektívnejšie a rýchlejšie si zaobstarat' veci prostredníctvom nákupu online, je pre obchodníka omnoho ťažšie získať zákazníka. Každý kupujúci je rád, ak narazí na výhodnú ponuku alebo obdrží zľavový kupón, či už prostredníctvom emailu, ktorý pri registrácii na danom e-shope zadal, alebo prostredníctvom iného komunikačného kanálu. Dáta predstavujú zozbierané informácie z internetového obchodu zaoberajúceho sa predajom kníh, audio kníh, CD nosičov, elektronických kníh a čítačiek elektronických kníh. Celý analytický proces bol riadený metodikou CRISP-DM s využitím programovacieho jazyka R. Na upravené dáta sme vo fáze modelovania použili vybrané techniky, konkrétne algoritmy rozhodovacích stromov C4.5, C5.0 a Random Forest, Naivný bayesovský klasifikátor, algoritmus k-NN, metódu SVM a takisto aj logistickú regresiu. Vo vyhodnotení sme zamerali nielen na hodnoty predikcie ale aj použiteľnosť vytvorených modelov.

Kľúčové slová — kupónový marketing, klasifikácia, e-shop

I. ÚVOD

Internetový obchod ako jeden z najväčších a v súčasnosti najpopulárnejších segmentov na trhu predaja výrobkov a služieb, je čoraz viac nútený externými vplyvmi na jeho predaje zaoberať sa správnym riešením marketingu a následnej optimalizácii činností s ním súvisiacimi. Správne nastavenie starostlivosti o zákazníkov môže byť kľúčovým bodom k získaniu ich väčšieho počtu, no takisto aj k udržaniu tých, ktorí už niekedy v minulosti prejavili o služby internetového obchodu záujem. Takéto optimálne nastavenie spomínanej politiky je dobrým odrazovým mostíkom k zvýšeniu predaja, a tým pádom aj navýšeniu celkových tržieb obchodu. Ako jeden z najúčinnějších spôsobov sa javí kupónové odmeňovanie zákazníkov. Avšak práve pri tomto druhu politiky musí internetový obchod dbať na množstvo faktorov súvisiacich s využívaním zľavových kupónov. Nesprávne nastavenie odmeňovacieho riešenia môže viesť k neefektívnemu využívaniu kupónov zákazníkmi, a tým k nulovému efektu prilákania zákazníkov k nákupu tovarov. V horšom prípade, môže zlé riešenie priviesť internetový obchod k stratám. Každý internetový obchod disponuje množstvom údajov a informácií o svojich zákazníkoch realizujúcich kúpu. Tieto dáta sú kľúčové v procese správnej identifikácie zákazníkov a následne úspešne nastavenej stratégii obchodu pri kupónovom marketingu.

A. Prehľad existujúcich prác

Kupónový marketing a analýza dát získaných z historických objednávok zákazníkov patria medzi rýchlo sa rozvíjajúce oblasti. Vybrali sme niekoľko existujúcich štúdií, ktoré potvrdzujú potenciál tejto kombinácie z pohľadu biznis cieľov daného obchodníka alebo firmy. Zároveň ukazujú, ktoré metódy strojového učenia už boli na tento účel použité; alebo ktorých potenciál je potrebné ešte overiť na vhodných dátových množinách, čo bolo aj motiváciou pre naše experimenty.

V práci [1] sa autori zamerali na zvýšenie predajov obchodného domu vhodnou distribúciou kupónov s cieľom nárastu predaja ale aj spokojnosti zákazníkov. Autori si vybrali jednu z jednoduchších a ľahko porozumiteľných techník, a to metódu rozhodovacích stromov, konkrétne algoritmus C4.5 a nástroj WEKA. Použité dáta predstavovali sumár šiestich mesiacov v obchodnom dome a boli popísané 6 atribútmi, ako napríklad vek, pohlavie, profesia a vzdelanie zákazníka. Po použití algoritmu C4.5 vygenerovali z výsledného klasifikačného stromu aj pravidlá ako napríklad: AK je vek zákazníka vyšší ako 30 a zároveň nižší alebo rovný 40, A

príjem zákazníka je vyšší ako 25 000 a zároveň nižší alebo rovný ako 50 000, A ukončené vzdelanie zákazníka je 1. stupeň VŠ, A pohlavie zákazníka je žena, POTOM zákazník kupón použije. Autori však v závere uviedli, že nevýhodou tohto algoritmu je lineárne zväčšovanie s narastajúcim počtom dát, čím sa celý algoritmus výrazne spomaľoval.

Autori Yao a Lu sa vo svojej práci [2] orientovali na optimalizáciu distribúcie digitálnych kupónov v súlade s obmedzením maximálneho počtu ponúk. Na tento účel použili historické transakcie 10 000 náhodne vybraných zákazníkov v období od apríla 2017 do apríla 2018, ktoré obsahovali aj záznamy o priradení kupónu, t.j. dátum, typ, nominálnu hodnotu a cieľové ID zákazníka. Pri modelovaní si vytvorili niekoľko obmedzení. Prvotne ignorovali vlastnosti každého kupónu (jeho hodnotu a typ) a pozerali sa iba na čas poskytnutia kupónu zákazníkom. Ďalej sa rozhodovali iba na základe histórie nákupov zákazníkov za posledných 8 týždňov. Po tretie, stanovili si cyklus rozhodnutia o pridelení kupónu týždenne. Cieľom týchto optimalizácií bolo maximalizovať celkový príjem za nasledujúce 3 mesiace pre každé z týchto obmedzení. Vo fáze modelovania zlúčili transakcie o zákazníkoch a záznamy o priradení kupónu. Dáta si rozdelili v pomere 80 (trénovanie) a 20 (testovanie), na ktoré aplikovali Q-learning algoritmus. Ich výsledky ukázali, že pri obmedzení maximálneho počtu ponúkaných kupónov sa kumulovaná celková odmena zvyšuje o 6% po optimálnej politike v porovnaní s použitím pôvodnej politiky.

V práci [3] mali autori k dispozícii rok staré dáta o nákupoch a údajoch 22 000 zákazníkov, ale iba pri 19 413 kupónoch boli poskytnuté atribúty ako diskontná cena, celková cena alebo umiestnenie predajne. Ich úlohou bolo predikovať, ktoré kupóny si daný zákazník v priebehu budúceho týždňa zakúpi. Na vyhodnocovanie experimentov používali metriku Mean Average Precision (priemerná presnosť). V tejto metrike sledovali predikciu pozretia 10-tich najlepších kupónov, a zároveň ktorí kupujúci si pozreli v danom týždni menej ako 10 kupónov. Zistili, že počet návštevníkov a videní je oveľa vyšší ako počet kupujúcich a nákupov. Vyhodnotením všetkých výsledkov zistili, že práve algoritmus XGBoost priniesol najnižšiu chybu 0.00072. Okrem tohto prístupu autori odporúčali aj iné metódy pre predikovanie ako Random forest alebo k-NN.

B. Použité metódy

V tejto časti sú stručne popísané analytické metódy, ktoré sme použili v našich experimentoch:

Naivný Bayesovský klasifikátor - slovo naivný nie je v názve tejto štatistickej klasifikačnej metódy náhodné. Vychádza z predpokladu nezávislosti atribútov, a teda, že vplyv jedného z atribútov nie je nijako ovplyvnený iným atribútom [4]. V tomto klasifikátore ide o spôsob, pomocou ktorého môžeme vypočítať pravdepodobnosť hypotézy vzhľadom na predchádzajúce znalosti. Hovoríme tak o podmienenej pravdepodobnosti.

Algoritmy C4.5, C5.0 - keďže v prípade algoritmu C5.0 ide o modifikáciu alebo rozšírenie algoritmu C4.5, majú väčšinu podobných vlastností. V oboch prípadoch ide o algoritmus rozhodovacieho stromu. Podľa porovnania¹ je zjavné, že algoritmus C5.0 prevyšuje C4.5 algoritmus vo svojej presnosti, rýchlosti a menšej spotrebe pamäte. Čo sa týka nových funkcionalít, u C5.0 algoritmu nachádzame napríklad možnosť zistenia dôležitosti jednotlivých atribútov podieľajúcich sa na predikcii alebo aj väčšie množstvo podporovaných dátových typov.

Algoritmus Random Forest - patrí medzi techniky modelovania využívané pre úlohy regresné i klasifikačné. Pri tejto technike sa znova zastavíme pri jej názve, ktorý opäť nie je náhodný. Forest (les) vysvetľuje fakt, že tento algoritmus vytvára niekoľko rozhodovacích stromov, ktoré tvoria akýsi les. Každý zo stromov v tomto lese buduje svoje otázky na základe istej podmnožiny všetkých atribútov a každý má tak k dispozícii inú, neúplnú dátovú vzorku [5]. Spojením týchto stromov získavame presnejší a stabilnejší výsledok. V prípade klasifikačných úloh je výsledkom hodnota, ktorá bola predikovaná vo väčšine podstromov tohto modelu. Práve veľké množstvo menších, rozličných rozhodovacích stromov predstavuje faktor, ktorý oproti bežnému samostatnému rozhodovaciemu stromu prináša lepší a presnejší výsledok predikcie.

Metóda podporných vektorov (SVM – Support Vector Machine) - používa sa pri riešení klasifikačných problémov, ale aj problémov regresnej analýzy. Snahou algoritmu je rozdeliť dáta do dvoch tried, ktorých rozdeľuje čiara nachádzajúca sa zhruba v polovici vzdialenosti krajných bodov = nadrovina. Cieľom je umiestnenie tejto čiary tak, aby predstavovala maximálnu vzdialenosť od oboch krajných bodov tried [6]. Tieto krajné body sa nazývajú aj podporné vektory a vzdialenosť týchto bodov môžeme označiť ako kritériálna funkcia.

Algoritmus k-najbližších susedov (kNN – k Nearest Neighbors) - tento algoritmus predpokladá, že podobné prvky sa nachádzajú vo svojej tesnej blízkosti. Podobnosť sa dá definovať ako vzdialenosť medzi dátovými bodmi. Existuje viacero metód pre výpočet tejto vzdialenosti, no najčastejšie využívanou je takzvaná Euklidovská vzdialenosť. Pre rozhodnutie

¹ <https://rulequest.com/see5-comparison.html>

o tom, kam algoritmus zaradí prvok z testovacej množiny je kľúčové poznať hodnotu K . Táto hodnota hovorí o tom, koľko susedov budeme brať do úvahy pri rozhodnutí pri klasifikácii nového prvku. K rozhodnutiu o zaradení prvku do istej z tried sa dospeje zistením, koľko susedov z konkrétnej triedy tento prvok má, a teda ktorý z nich je súčasťou majoritnej triedy [7].

Logistická regresia - slovo regresia môže v tomto prípade vzbudzovať dojem, že sa jedná o algoritmus, ktorého výstupom je istá predikovaná hodnota numerickej premennej. Keďže však ide o logistickú regresiu, výstupom tohto algoritmu je pravdepodobnosť toho, či vstupná hodnota patrí alebo nepatrí do danej triedy. To znamená, že hodnoty sa musia nachádzať v intervale $<0,1>$ [8]. Cieľom tohto algoritmu je nájsť určitý vzťah medzi atribútmi a pravdepodobnosťou istého výsledku.

II. ANALYTICKÝ PROCES

Táto časť príspevku obsahuje riešenie vybranej problematiky rozdelenej do šiestich fáz podľa metodológie CRISP-DM.

A. Pochopenie cieľa

Vytvorenie profilu zákazníka na základe pochopenia jeho minulého nákupného správania používa internetový obchod na správne rozhodnutie o zaslaní/nezaslaní kupónu. Z analytického pohľadu riešime klasifikačnú úlohu, pri ktorej s využitím historických informácií o zákazníkoch budeme predikovať rozhodnutie o uskutočnení ich budúcej objednávky. Z biznis pohľadu je strategicky správne zaslanie ďalšieho zľavového kupónu zákazníkovi, čo vedie internetový obchod k optimálnemu nastaveniu benefítového systému. Na druhej strane, zaslanie kupónu zákazníkovi, ktorý by nákup realizoval aj bez jeho obdržania, je pre internetový obchod neefektívne.

B. Pochopenie a príprava dát

Dáta použité v tomto príspevku pochádzajú zo súťaže Data Mining Cup 2010 a predstavujú zozbierané informácie z internetového obchodu zaoberajúceho sa predajom kníh, audio kníh, CD nosičov, elektronických kníh a čítačiek elektronických kníh. Dostupné dáta sú rozdelené na trénovaciu a testovaciu množinu. Trénovacia množina obsahuje 37 vstupných atribútov, 1 cieľový atribút *target90* (hovorí o tom, či bola objednávka do 90 dní znova zrealizovaná) a 32 428 záznamov, ktoré predstavujú objednávky zákazníkov. Testovacia množina obsahuje len 37 atribútov, nakoľko cieľový atribút je dostupný iba v trénovacej množine.

Pred fázou modelovania sme tieto dáta podrobili niekoľkým úpravám, nakoľko obsahovali niekoľko zvláštnych hodnôt. Pri atribúte *deliverydate* bol v niekoľkých záznamoch dátum uvedený ako 0000-00-00, ktorý logicky neudáva správne časové obdobie. Pri hlbšom preskúmaní dát sme zistili, že takýto formát dátumu sa vyskytuje v prípade, ak ide napríklad o elektronickú knihu, ktorú si zákazník stiahne, alebo napríklad o situáciu, kedy zákazník objednávku zruší. Na základe porovnania atribútov *deliverydate* a *deliverydatepromised* sme vytvorili v našej dátovej množine nový atribút nazvaný *delayed*, ktorý hovorí o tom, či bola alebo nebola zásielka oneskorená. O konkrétnom počte dní omeškania hovorí rovnako nový atribút *days_delayed*.

Ďalšou nezrovnalosťou v dátach bol atribút *points*, pri ktorom sme zistili, že hodnotu 1 nedosahuje v žiadnom zázname. Z toho sme usúdili, že jeho ďalšie využitie by neprinieslo žiaden úžitok a preto sme tento atribút v ďalšej časti práce z dátovej množiny odstránili.

Pri atribúte *deliverypostcode* sme zistili, že počet chýbajúcich hodnôt je veľký, avšak v tomto prípade sme to nepovažovali za chybu. V prípade chýbajúcej hodnoty sme predpokladali rovnaké poštové číslo ako bolo uvedené pre zaslanie faktúry. Takýto prípad nastal pri 31 036 záznamoch.

C. Modelovanie a vyhodnotenie

Na upravené dáta sme vo fáze modelovania použili vybrané techniky, konkrétne algoritmy rozhodovacích stromov C4.5, C5.0 a Random Forest, Naivný bayesovský klasifikátor, algoritmus k -NN, metódu SVM a takisto aj logistickú regresiu. Tieto algoritmy sme postupne vyskúšali na rôzne modifikovaných dátach pre zistenie a zabezpečenie optimálnej dátovej množiny, a následnému dosiahnutiu čo najlepšieho výsledku. Výkonnosť vytvorených modelov sme sledovali na trénovacej množine. Nasledujúcimi experimentami sme predikovali cieľový atribút *target90*, ktorý hovorí o tom, či zákazník do 90 dní od jeho poslednej objednávky, vykoná objednávku znova.

1. Experiment - prvotné modely bez úprav atribútov. Najvyššia presnosť bola dosiahnutá za použitia algoritmu Random Forest (87,01%), naopak najnižšiu presnosť dosiahol Naivný Bayes (74,8%).
2. Experiment - nadzorkovanie cieľového atribútu pomocou SMOTE funkcie. Funkciu SMOTE sme použili na vybalansovanie cieľového atribútu. Presnosť modelov sme

následne overili na nevybalansovanej trénovacej množine, ktorá neprešla procesom vzorkovania. Pôvodné zastúpenie hodnôt cieľového atribútu *target90* bolo 6 051 záznamov pre triedu 1 a 26 376 záznamov pre triedu 0. Po nadvzorkovaní sa hodnoty zmenili, a teda 24 204 záznamov pre triedu 1 a 24 204 záznamov pre triedu 0. Presnosť modelov na takto vytvorenej množine rapídne klesla, pričom Naivný Bayes dosiahol presnosť až menej ako 30%. Z pohľadu špecifickosti sa však táto hodnota zvýšila. Najvyššia presnosť bola znova v prípade algoritmu Random Forest (85,93%).

3. Experiment - nadvzorkovanie cieľového atribútu pomocou funkcie ROSE. Na vybalancovanie cieľového atribútu sme použili taktiež funkciu ROSE, pomocou ktorej sme však dosiahli omnoho lepšie výsledky. Po nadvzorkovaní bolo 26 376 záznamov pre triedu 1 a 26 376 záznamov pre triedu 0. Pri použití tejto metódy sme dosiahli najvyššiu presnosť zo všetkých modelov, konkrétne pomocou algoritmu C4.5 (89,11%).
4. Experiment - podvzorkovanie cieľového atribútu pomocou funkcie ROSE. Pri podvzorkovaní bolo 6 051 záznamov pre triedu 1 a 3 051 záznamov pre triedu 0. V porovnaní s nadvzorkovaním, nám podvzorkovanie prinieslo horšie výsledky. Najvyššia dosiahnutá presnosť bola za použitia Naivného Bayesa (74,97%).
5. Experiment - redukcia atribútov. Pri tomto experimente sme dátovú množinu zredukovali o 2 atribúty na základe výsledkov štatistických testov, s cieľom dosiahnutia vyššej presnosti modelov. Rovnako sme túto dátovú množinu podrobili vzorkovaniu s využitím metódy nadvzorkovania ROSE, ktorá nám pri vyššie vykonaných experimentoch priniesla najúspešnejšie výsledky. V prípade množiny pred nadvzorkovaním bola najvyššia dosiahnutá presnosť u algoritmu Random Forest (87,8%). Po nadvzorkovaní to bolo 87,73% v prípade algoritmu C4.5.

Zaujímavým pohľadom v našej práci bolo aj porovnanie hodnôt cieľového atribútu *target90* s hodnotami pôvodného atribútu *voucher*, ktorý hovorí, či zákazník využil kupón pri predchádzajúcej objednávke. Na základe tohto zistenia môžeme hľadať súvislosť medzi tým, že kupón, ktorý zákazník využil pri predchádzajúcej objednávke, motivoval zákazníka k ďalšej kúpe. Zamerali sme sa na jednu konkrétnu kombináciu vytvorenú z atribútov *target90*, ktorý sme predikovali na testovacej množine, a z atribútu *voucher*, ktorý sa v tejto množine takisto nachádza.

Experiment - kombinácia VOUCHER = 1 & TARGET90 = 0. Pri takomto type zákazníkov uvažujeme, že ak pri ich poslednej objednávke kupón použili, no následne nevideli žiadnu motiváciu objednávku uskutočniť znovu, bude potrebné ich k ďalšiemu nákupu znova motivovať zaslaním ďalšieho zľavového kupónu. Predikciu atribútu *target90* sme tentokrát realizovali na testovacej množine. Na tejto množine sme hľadali už spomínanú kombináciu hodnôt *voucher* = 1 & *target90* = 0. Po vyselektovaní riadkov zodpovedajúcim tejto podmienke sme si vytvorili nový atribút *sendvoucher*, ktorému sme priradili hodnotu 1, ak bola táto podmienka splnená, a naopak hodnotu 0, ak podmienka splnená nebola. Najvyššia dosiahnutá presnosť mala hodnotu 73,58% za použitia algoritmu C5.0.

D. Nasadenie

Novo nadobudnuté informácie získané pomocou našich experimentov, by mohol internetový obchod využiť pre nastavenie svojho kupónového marketingu. Nie sú dôležité len samotné výsledky predikcie, ale aj správne pochopenie správania zákazníkov a spôsoby, akými sme odvodzovali ich možné opakujúce sa nákupy. Takisto sme chceli poukázať na význam dát, akými vo všeobecnosti internetové obchody disponujú a vedia ich tak využiť vo svoj prospech. Naše zistenia nemusia pomôcť len v oblasti kupónového odmeňovania zákazníkov. E-shop sa vďaka predpovedanému počtu zákazníkov znova realizujúcich svoje nákupy môže zamerať aj na iné formy zlepšovania svojej marketingovej stratégie. Pre dosiahnutie čo najvyššej spoľahlivosti predikovaných modelov, a teda dosiahnutie zisku po zavedení kupónového systému, by boli potrebné ďalšie hlbšie analýzy a zamyslenia vzhľadom na komplexnosť problému a dostupných dát.

III. ZÁVER

Našou motiváciou bolo poukázať na potenciál dát, ktorými tradične disponuje internetový obchod. Preto sme na základe dostupnej dátovej vzorky vytvorili klasifikačné modely predikujúce či zákazník do 90 dní opätovne urobí objednávku alebo nie, kde najvyššia presnosť dosahovala 89,11%. Vo fáze vyhodnotenia sme následne najlepšie modely sme otestovali pomocou testovacej množiny definujúcej, či bolo vhodné motivovať zákazníka k tejto opätovnej objednávke kupónom. V tomto prípade sme dosiahli najlepšiu hodnotu presnosti 73,58%.

POUŽITÉ ZDROJE

- [1] S. Kumar, R.K. Arora: *Analyzing Customer Behaviour through Data Mining*. International Journal of Computer Applications Technology and Research, 2015, Vol. 4, s. 884-888.
- [2] G. Duan, X. Ma: *A Coupon Usage Prediction Algorithm Based On XGBoost*. In: Proceedings of 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Huangshan, China, 2018, s. 178-183.
- [3] A. Gupta: *Predicting Coupon Purchase*. Uhuru Data Lab, 2015.
- [4] S. Russell, P. Norvig, Peter: *Artificial Intelligence: A Modern Approach (2nd ed.)*. Prentice Hall, 2003.
- [5] L. Breiman: *Random Forests*. Machine Learning, 2001 45 (1), s. 5-32.
- [6] T.S. Furey, N. Cristiani, B. Duffy, D.W. Bednarski, M. Schummer and D. Hausslet: *Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data*. Bioinformatics. 2001, Vol. 16, s. 906-914.
- [7] N.S. Altman: *An introduction to kernel and nearest-neighbor nonparametric regression*. The American Statistician, 1992, 46 (3), s. 175-185.
- [8] J. Tolles, W. Meurer: *Logistic Regression Relating Patient Characteristics to Outcomes*. JAMA, 2016, 316 (5), 533-4.

Riešenie problému odchodovosti zákazníkov pomocou dátovej analytiky

Diana KLIMONOVÁ, František BABIČ, Ludmila PUSZTOVÁ

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Letná 9, 042 00 Košice, Slovenská republika

diana.klimonova@student.tuke.sk, frantisek.babic@tuke.sk,
ludmila.pusztova.2@tuke.sk

Abstrakt — V príspevku sa zaoberáme riešením problému odchodovosti zákazníkov v oblasti poskytovania elektrickej energie pomocou vhodných metód dátovej analytiky. Schopnosť predvídať, že zákazník uvažuje o odchode, môže v dnešnej dobe predstavovať kľúčový faktor úspechu v rámci podnikania. Náš analytický proces sme riešili v súlade s metodikou CRISP-DM, ktorá predstavuje životný cyklus pre projekty/procesy dátovej analytiky. Vo fáze modelovania sme zvolili sedem klasifikačných modelov: Naivný Bayes, algoritmus rozhodovacieho stromu C4.5 a C5.0, algoritmus Random Forest, k-NN, metódu SVM a logistickú regresiu. Na vyhodnotenie ich výsledkov sme použili úspešnosť, presnosť, špecifickosť, senzitivitu, ROC krivku, hodnotu AUC a taktiež kontingenčnú maticu. Podľa úspešnosti klasifikácie bol najlepší model vygenerovaný algoritmom k-NN (87.8%) a Naivný Bayes (87,08%). Zároveň ROC krivky ukázali preučenie v prípade modelov vygenerovaných pomocou algoritmov rozhodovacích stromov.

Kľúčové slová — odchodovosť zákazníkov, klasifikácia, CRISP-DM

I. ÚVOD

V dnešnej dobe je pre firmy a podnikateľov stále častejšou výzvou udržanie si svojich zákazníkov. Nakoľko prilákanie nových vyžaduje vysoké náklady, je oveľa väčším prínosom zamerať sa na udržanie tých aktuálnych. Dátová analytika je nevyhnutnou súčasťou v procese zvyšovania efektívnosti podnikania v rôznych oblastiach. Orientuje sa na spôsoby čo najefektívnejšieho riadenia biznis procesov a rovnako napomáha aj pri riešení rôznych problémov, či už z oblasti logistiky a plánovania manažmentu. Dátová analytika môže predstavovať kľúčový faktor úspechu aj v spojitosti s riešením problematiky odchodovosti zákazníkov. Schopnosť predvídať, že zákazník uvažuje o odchode, môže v dnešnej dobe predstavovať kľúčový faktor úspechu v rámci podnikania.

Obchodovanie s elektrickou energiou v súčasnosti nie je to, čo to bývalo kedysi. Na trhu existuje silná konkurencia aj v tejto oblasti a zákazníci si môžu slobodne vybrať, koho služby budú využívať. A práve v tomto bode nastáva u poskytovateľov energie problém resp. výzva, ktorú je potrebné vyriešiť, s cieľom prosperity podniku. Medzi odberateľmi energie môžeme definovať 2 typy zákazníkov. Zákazníci, ktorí sú lojálni, čiže stáli. Svojim správaním preukazujú vernosť jednému poskytovateľovi energie. A naopak zákazníci, ktorí majú neštandardné správanie a charakterizuje ich častá zmena dodávateľa energie. Existujú rôzne faktory, ktoré zákazníka vedú k tomu, aby nedodržiaval lojalnosť voči jednému poskytovateľovi energie. Môže ho k tomu viesť výhodnejšia ponuka v podobe nižšej ceny u konkurencie, čo v konečnom dôsledku zapríčiňuje pokles zisku u dodávateľa. Nakoľko prilákanie nových zákazníkov predstavuje vysoké náklady vložené napríklad do reklamy, dodávateľa elektrickej energie sú preto konfrontovaní výzvou, ktorá spočíva v schopnosti klasifikácie zákazníkov medzi lojálnych a nelojálnych – ktorí potenciálne uvažujú o odchode. Na základe tejto klasifikácie budú schopní zefektívniť ponuky pre svojich zákazníkov tak, že zákazník nebude mať dôvod odísť ku konkurencii a zároveň dodávateľa energií budú disponovať ziskom, ktorý nebude výrazne klesať v dôsledku odchodu zákazníkov.

A. Analýza Súčasného Stavú

V nasledujúcej časti je uvedených niekoľko štúdií, ktorých autori sa venovali problematike odchodu zákazníkov v rôznych doménových oblastiach. Analyzovali sme hlavne použité metódy a dosiahnuté výsledky.

Práca čínskych autorov [1] sa zameriava na predpovedanie odchodu zákazníkov v bankovom sektore. Ich cieľom bolo vylepšiť existujúce riešenia tohto problému a poukázať na nevýhody iných algoritmov ako sú napríklad rozhodovacie stromy, neurónové siete, ale aj Bayesovský klasifikátor. Použili metódu IBRF, ktorá predstavuje kombináciu metódy *balanced random forest* a *weighted random forest*. Metódu aplikovali na dáta z reálneho sveta; extrahované z dátového skladu bánk, ktorý obsahoval záznamy viac ako 20 000 zákazníkov popísané pomocou 27 atribútov. Na tvorbu modelu použili len vybrané atribúty ako napríklad vek, vzdelanie, typ zamestnania zákazníka, ako aj výšku úveru, dĺžku splatnosti úveru, atď. V rámci rozdelenia vzoriek použitých pri tréningovej a testovacej množine bolo náhodne vybraných 1 524 vzoriek, kde vo vybraných vzorkách bolo dokopy 73 zákazníkov, ktorí odišli. Použitím metódy IBRF dosiahli až 93,2% presnosť. V porovnaní s existujúcim riešením, metóda IBRF výrazne presahovala presnosť rozhodovacieho stromu (62%) i neurónových sietí (78,1%).

V práci [2] autori vychádzali z porovnania dvoch klasifikačných techník využívaných pri predikcii odchodu zákazníkov, a to konkrétne rozhodovacie stromy (algoritmy CART, C5.0 a CHAID) a logistické regresné modely. Samotný výskum prebiehal za použitia dvoch datasetov, ktoré boli získané z dátového skladu mobilného telekomunikačného operátora vo Veľkej Británii. Datasety obsahovali 15 519 a 19 919 záznamov popísaných 17 atribútmi. Medzi atribútmi bola aj výstupná premenná, ktorá definovala, či ide o odchod zákazníka alebo nie. Pred modelovaním si vytvorili konečné množiny odstránením chýbajúcich hodnôt a nevýznamných atribútov, ktoré identifikovali pomocou Waldovho testu. Na vytvorenie samotných experimentov použili softvér IBM SPSS Modeler. Miera úspešnosti pri algoritme CART dosahovala 68,57%, pri CHAID 68,83% a pri C5.0 70,25%. Použitím logistických regresných modelov dosiahli úspešnosť maximálne 68,1%. Na základe týchto výsledkov dospeli k záveru, že rozhodovacie stromy sú úspešnejšou technikou pri riešení problému odchodovosti zákazníkov.

Štúdia [3] bola zameraná na porovnanie piatich najpoužívanejších metód strojového učenia, ktoré sa používajú pri predikcii odchodu zákazníkov v telekomunikačnom priemysle. Autori sa zamerali najmä na neurónové siete, rozhodovacie stromy, metódu podporných vektorov, Naivný Bayesovský klasifikátor a logistickú regresiu. Štúdia bola vykonávaná na datasete churn z UCI Machine Learning Repository, z ktorého použili len 18 atribútov prevažne numerického typu a jeden binárny atribút churn, ktorý nadobúdal hodnoty yes/no. V prvom experimente na testovaciu množinu aplikovali všetky metódy, v druhom kroku na jednotlivé metódy použili tzv. Boosting algoritmus. Tento algoritmus bol zavedený na neurónové siete, metódu podporných vektorov a rozhodovacie stromy. Z použitých metód dosahovali neurónové siete, metóda podporných vektorov a rozhodovacie stromy presnosť nad 93%. Vo všetkých troch prípadoch sa presnosť zvýšila použitím Boosting algoritmu.

B. Použité metódy

Náš analytický proces sme riešili v súlade s metodikou CRISP-DM [4], ktorá predstavuje životný cyklus pre projekty/procesy dátovej analytiky. Tento procesný model je tvorený šiestimi fázami, konkrétne pochopenie cieľa, pochopenie dát, príprava dát, modelovanie, vyhodnotenie a nasadenie. Vo fáze modelovania sme na naše dáta použili tieto techniky:

Naivný Bayesovský klasifikátor - je klasifikačným algoritmom, ktorý funguje na princípe Bayesovej teórie podmienenej pravdepodobnosti, na základe ktorej klasifikuje záznamy do jednotlivých tried cieľového atribútu. Algoritmus funguje na princípe výpočtu podmienenej pravdepodobnosti, pričom zaradenie objektov do tried cieľového atribútu vykonáva na základe jej maximálnej hodnoty [4]. Ako dôvod prívlastku „naivný“ tohto algoritmu, definuje autor [5] predpoklad nezávislosti všetkých atribútov.

Rozhodovací strom (algoritmus C4.5) - ide o klasifikačný algoritmus generujúci rozhodovacie vychádzajúc z výpočtu entropie a maximálneho pomerového informačného zisku [6]. Samotný rozhodovací strom sa generuje zhora - nadol, teda od základného koreňového uzla až po konečné listové uzly. Vetvenie reprezentuje príslušný atribút, ktorého výber je riadený maximálnou hodnotou pomerového informačného zisku.

Rozhodovací strom (algoritmus C5.0) - predstavuje vylepšenú verziu algoritmu C4.5. Bol navrhnutý tak, aby bol schopný spracovávať aj datasety s veľkým počtom záznamov. Disponuje výhodami ako sú rýchlosť, schopnosť efektívne využívať pamäť (dokáže výrazne redukovat' využívanú pamäť v porovnaní s algoritmom C4.5), a taktiež aj možnosť zavedenia boostingu, vďaka ktorému je možné dosahovať vyššiu presnosť a nižšiu chybovosť dosiahnutú použitím tejto metódy [7]. Kritérium vetvenia samotného rozhodovacieho stromu je rovnako ako pri algoritme C4.5 definované výpočtom entropie a informačného zisku.

Random Forest - je klasifikačnou metódou, ktorej princíp spočíva vo vytváraní niekoľkých rozhodovacích stromov a ich následnej agregácii prostredníctvom spriemerovania. Názov tejto metódy sa spája s náhodným výberom atribútov, a teda aj pre jednotlivé rozhodovacie stromy

platí, že medzi nimi neexistuje žiadna závislosť. Na vygenerovanie každého čiastkovéhostromu sa nevyužívajú všetky atribúty, ale len náhodne definovaná podmnožina týchto atribútov [8]. Algoritmus tejto metódy teda predovšetkým spočíva v stanovení m atribútov z celkového počtu p atribútov.

k-Nearest Neighbors (k-NN) - predstavuje jednu z najznámejších a najčastejšie využívaných metód strojového učenia. Algoritmus k-NN je charakterizovaný zatriedením záznamov s doposiaľ neznámou triedou cieľového atribútu tak, že klasifikátor k-NN hľadá v priestore záznamy tréningovej množiny, ktoré sú k novému záznamu najbližšie. Teda základný princíp spočíva vo vzdialenosti medzi jednotlivými objektami (záznamami). Dôležitou časťou v prípade aplikovania tohto algoritmu je správne stanovenie parametra k, ktoré je v tomto prípade kľúčom k úspechu. Autor [4] poukazuje na fakt, že veľmi malé, ale ani veľmi veľké hodnoty parametra k neprispievajú k správne fungovaniu tejto metódy, resp. môže dochádzať k nepresnej klasifikácii.

Support Vector Machine (SVM) - metóda podporných vektorov, ktorá vo svojom algoritme pracuje s kernelovou metódou [8]. Podstata fungovania algoritmu SVM spočíva v roztriedení lineárne separovateľných dát pomocou priamky, alebo ak ide o dáta, ktoré nie sú lineárne separovateľné, teda viacrozmerné dáta, ide o roztriedenie pomocou nájdenia tzv. deliacej hyperplochy [9]. Je potrebné nájsť takú hranicu, ktorá určuje čo najširšiu vzdialenosť medzi hraničnými bodmi pre jednotlivé triedy cieľového atribútu.

Logistická regresia – regresná metóda, pri ktorej nominálny cieľový atribút (závislá premenná), nadobúda dve hodnoty (0 a 1). Nezávislé premenné môžu byť nominálne, ale aj spojité. Je založená na pravdepodobnosti, na základe ktorej určuje zatriedenie záznamov do konkrétnej kategórie cieľového atribútu [8],[10]. Inak povedané ide o modelovanie závislej premennej, ktorá je závislá na viacerých nezávislých premenných.

II. ANALYTICKÝ PROCES

V tejto časti sme zhrnuli základné zistenia nášho analytického procesu riadeného metodológiou CRISP-DM.

A. Pochopenie cieľa

Obchodným cieľom bola identifikácia zákazníkov, ktorí potenciálne uvažujú o odchode, čo by mohlo znamenať zníženie zisku spoločnosti. Na riešenie tejto situácie by spoločnosť aplikovala cieleň marketing. Z analytického pohľadu išlo o vytvorenie klasifikačných modelov, ktoré dokážu rozdeliť zákazníkov na lojálnych a nelojálnych – potenciálne uvažujúcich o odchode. Na základe analyzovaných prác, zaoberajúcich sa rovnakou problematikou, sme si stanovili minimálnu hranicu úspešnosti, ktorá predstavuje 80%, pričom cieľom analytického procesu bolo túto hranicu nielen prekročiť, ale aj dosiahnuť najvyššiu možnú úspešnosť klasifikácie.

B. Pochopenie a príprava dát

Na riešenie tejto úlohy sme použili dáta dostupné zo súťaže Data Mining Cup 2002, ktoré boli zozbierané od obyvateľov Nemecka a následne agregované na úroveň tzv. mikrobuniek (najmenší úsek ulice s najmenej 5 domácnosťami). Boli to anonymizované údaje, ktoré zaručili dodržiavanie prísnych nemeckých predpisov o ochrane osobných údajov. K dispozícii sme mali ako tréningovú, tak aj testovaciu množinu, pričom obe obsahovali demografické údaje o 10 000 zákazníkoch poskytovateľa elektrickej energie, ako napríklad vekovú štruktúru, sociálne vrstvy, spotrebu elektrickej energie za posledný rok, úverový status, hustotu vozidiel a ďalšie. Datasets boli tvorené 34 atribútmi: 29 nominálnych atribútov, 3 numerické atribúty a cieľový atribút *canceler*. Cieľový atribút charakterizoval typ zákazníka z pohľadu lojálnosti, a teda nadobúdal hodnotu yes (nelojálny zákazník) a no (lojálny/stály zákazník). Distribúcia týchto hodnôt bola nevyvážená, nakoľko lojálnych bolo 9000 zákazníkov a nelojálnych len 1000 zákazníkov.

V rámci pochopenia dát sme sa zamerali aj na sledovanie závislostí medzi atribútmi, ktoré sme v prípade nominálnych atribútov realizovali pomocou *Chi-kvadrát testu nezávislosti*. Rovnako sme pomocou tohto testu sledovali potenciálne závislosti medzi nominálnymi atribútmi a cieľovým atribútom. Závislosť medzi atribútmi je určená na základe p-hodnoty. Ak p-hodnota je menšia ako 0.05, existuje medzi dvoma atribútmi závislosť. Takýmto spôsobom sme dospeli k 12 atribútom ako napríklad **payment_type**, **MTREG0G**, **MTKAU0G**, **MTBEB0G**, **MTBON0G** a ďalšie. V prípade numerických atribútov sme zvolili sledovanie korelácií, pričom vo všetkých prípadoch bola zistená slabá korelácia. Taktiež sme vykonali aj sledovanie závislostí medzi numerickými atribútmi a cieľovým atribútom, načo sme použili *dvojvýberový Welchov t-test*. Týmto testom sme dospeli k záveru, že medzi všetkými numerickými atribútmi a cieľovým atribútom existuje závislosť.

Vo fáze prípravy dát sme sa zamerali predovšetkým na problém nevyvážených dát. Zvolili sme nadvzorkovanie a podvzorkovanie pomocou funkcie *ROSE* a funkcie *SMOTE*. Tabuľka 1 znázorňuje početnosť záznamov pre jednotlivé triedy cieľového atribútu pred a po vzorkovaní. Pôvodné zastúpenie tried cieľového atribútu definuje dáta po odstránení chýbajúcich hodnôt, ktoré sa v dataseete nachádzali.

Tabuľka 1
 Porovnanie distribúcie hodnôt cieľového atribútu

Trieda cieľového atribútu	Pôvodné	Nadvzorkovanie (<i>ROSE</i>)	Podvzorkovanie (<i>ROSE</i>)	Nadvzorkovanie (<i>SMOTE</i>)	Podvzorkovanie (<i>SMOTE</i>)
No	6 874	6 874	734	6 606	734
Yes	734	6 874	734	6 606	734

Taktiež sme sa v tejto fáze venovali redukcii dimenzionality, ktorá spočívala vo výbere atribútov na základe štatistických testov a korelácií, ktorými sme sa zaoberali v časti pochopenia dát. Nový dataset teda obsahoval 15 atribútov a cieľový atribút canceler.

C. Modelovanie a vyhodnotenie

Vo fáze modelovania sme zvolili sedem klasifikačných modelov, konkrétne Naivný Bayes, algoritmus rozhodovacieho stromu C4.5 a C5.0, algoritmus Random Forest, k-NN, metódu SVM a logistickú regresiu. Na vyhodnotenie ich výsledkov sme použili úspešnosť, presnosť, špecifickosť, senzitivitu, ROC krivku, hodnotu AUC a taktiež kontingenčnú maticu.

O tom, že dátová analytika je iteratívny a interaktívny proces sme sa presvedčili hlavne v prípade prípravy dát a modelovania, kde sme postupne krok po kroku sledovali výsledky zvolených klasifikačných modelov, ktoré boli vytvárané na jednotlivých datasetoch obsahujúcich zmeny na tréningových dátach. Išlo o dataset bez chýbajúcich hodnôt, nadvzorkovaný a podvzorkovaný dataset pomocou funkcie *ROSE*, nadvzorkovaný a podvzorkovaný dataset pomocou funkcie *SMOTE* a taktiež dataset, ktorého počet atribútov bol zredukovaný na základe štatistických testov. V prípade, ak model vykazoval najlepšie výsledky pri niektorej z metód vzorkovania, bol taktiež upravený z pohľadu dimenzionality a následne boli dosiahnuté výsledky porovnané. Dataset, na ktorom boli modely aplikované prvýkrát, charakterizoval akýsi počiatočný bod poskytujúci prvotné výsledky, ktoré sme sa snažili ďalšou úpravou dát vylepšiť. Jednotlivé modely boli vytvárané na tréningových a testovacích dátach s rovnakým počtom záznamov, vid' Tabuľka 2.

Tabuľka 2
 Vyhodnotenie vygenerovaných modelov

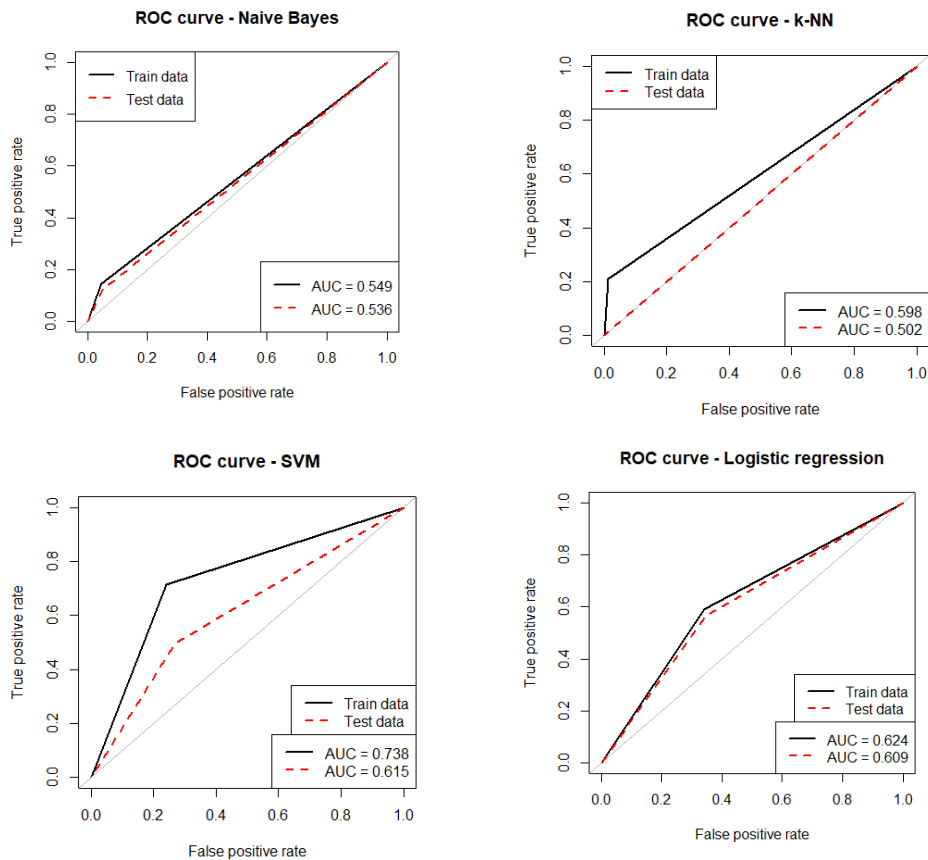
Metrika	Naivný Bayes	k-NN	SVM	Logistická regresia
Úspešnosť	87.08	87.80	71.11	64.38
Presnosť	91.07	90.43	93.21	93.62
Špecifickosť	12.19	3.56	49.59	58.22
Senzitivita	95.03	96.74	73.39	65.03

Hranicu úspešnosti prekročili modely založené na algoritme Naivného Bayes a k-Najbližších susedov. Modely vytvorené pomocou rozhodovacích stromov vykazovali známky preučenia, ich ďalšia adaptácia bude predmetom budúcej práce. Rovnako konštatujeme, že všetky modely mali problém správne klasifikovať minoritnú triedu áno – uvažuje o odchode, vid' príslušné ROC krivky (Obr.1). Redukcia počtu vstupných atribútov a zvolené metódy podvzorkovania a nadvzorkovania prispeli k čiastkovému zlepšeniu niektorých výsledkov.

D. Nasadenie

Poslednou fázou metodiky CRISP-DM je nasadenie dosiahnutých výsledkov do praxe. Ak by sme sa zamerali na cieľový marketing poskytovateľa elektrickej energie, definovali by sme rozdielne cenové ponuky pre jednotlivé typy zákazníkov. Je potrebné si uvedomiť, že z

dosiahnutých výsledkov, ktoré sme zaznamenali pri klasifikácii pomocou jednotlivých modelov napríklad k-NN, Naive Bayes alebo C4.5, síce je klasifikovaných najviac záznamov správne, avšak existuje aj určitý počet záznamov spomedzi všetkých, ktoré boli zatriedené nesprávne. Práve preto je potrebné zamerať sa na cieľový marketing, ktorý by rozlišoval ponuky pre zákazníkov, ktorých zatriedenie nebolo úspešné a rovnako pre zákazníkov, ktorí boli správne klasifikovaní do jednotlivých tried cieľového atribútu.



Obr. 1 ROC krivky najlepších klasifikačných modelov.

Predstavme si situáciu bez aplikovania nami vytvorených klasifikátorov. Každý zákazník dostáva rovnakú cenovú ponuku, pričom poskytovateľ elektrickej energie nie je schopný povedať, ktorý odberateľ sa javí ako potenciálne uvažujúci o odchode. Poskytovateľ elektrickej energie má 100 zákazníkov, ktorí platia rovnakú čiastku za spotrebu elektrickej energie $X\text{€/mesačne}$. Jeho ročný zisk teda predstavuje $100 \cdot X \cdot 12$. Ak teda odíde napríklad 10% zákazníkov, pre poskytovateľa by to znamenalo pokles zisku o 10% z celkového ročného zisku, čo môže predstavovať dosť veľkú čiastku. Ak by však poskytovateľ elektrickej energie disponoval schopnosťou klasifikovať zákazníkov na základe niektorého nami vytvoreného modelu, mohol by upraviť cenovú ponuku tak, aby zákazník uvažujúci o zmene dostával najnižšiu cenovú ponuku, lojálny zákazník by dostával tiež výhodnejšiu cenovú ponuku a zákazník, ktorého klasifikácia nebola úspešná, pôvodnú cenovú ponuku. Na základe takto stanovenej ceny by poskytovateľ pri odchode zákazníka mal zabezpečený menší pokles zisku, ako v prípade keď nebol schopný rozoznávať medzi jednotlivými typmi zákazníkov. Zároveň by bol schopný docieľiť výhodnejšími ponukami udržanie si svojich stálych zákazníkov.

III. ZÁVER

Tento článok popisuje spôsob, akým je možné riešiť problematiku odchodovosti zákazníkov za pomoci klasifikačných metód dátovej analytiky. Jedným z prvých a podstatných krokov procesu bolo správne pochopenie dát a zistenie závislostí medzi jednotlivými premennými. Následne sme vykonali niekoľko zmien na dostupných dátach, ako napríklad odstránenie chýbajúcich hodnôt a nadvzorkovanie/podvzorkovanie datasetu s cieľom dosiahnutia čo najpresnejších výsledkov. Na upravených dátach sme vytvárali klasifikačné modely, ktorých výsledky sme neustále porovnávali.

POUŽITÉ ZDROJE

- [1] X. Yaya, L.Xiu, E.W.T.Ngai, Y. Weiyun: Customer churn prediction using improved balanced random forests, *Expert Systems with Applications* 2009, s. 5445-5449, ISSN 0957-4174
- [2] M. Hassouna, A.Tarhini, E.Tariq, M.A. Saeed: Customer Churn in Mobile Markets A Comparison of Techniques, *Canadian Center of Science and Education* 2015, s. 224-237
- [3] T.Vafeiadis, K.I. Diamantaras, G.Sarigiannidis, K.Chatzisavvas: A comparison of machine learning techniques for customer churn prediction, *Simulation Modelling Practice and Theory* 2015, Vol. 55, s.1-9.
- [4] Stuart; Norvig, Peter (2003) [1995]. *Artificial Intelligence: A Modern Approach* (2nd ed.). Prentice Hall. ISBN 978-0137903955.
- [5] P.Kaviani, S.Dhotre: Short Survey on Naive Bayes Algorithm. *International Journal of Advance Engineering and Research Development* 2017,Vol. 4, s. 607-611, ISSN 2348-4470.
- [6] S.Singh, M.Giri: Comparative Study Id3, Cart And C4.5 Decision Tree Algorithm. *International Journal of Advanced Information Science and Technology* 2014, s. 47-52 s. ISSN 2319-2682.
- [7] M.Arif: Decision Tree Algorithms C4.5 and C5.0 in Data Mining. *International Journal of Database Theory and Application* 2018, Vol. 11, s.1-8, ISSN 2005-4270.
- [8] Cortes, Corinna; Vapnik, Vladimir N. (1995). "Support-vector networks" (PDF). *Machine Learning*. 20 (3): 273–297. CiteSeerX 10.1.1.15.9362. doi:10.1007/BF00994018.
- [9] D.A.Pisner, D. M. Schnyer: Support vector machine, *Machine Learning* 2019,s.102, ISBN 978-0-12-815739-8.
- [10] F.R.Harrell: *Binary Logistic Regression, Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*. International Publishing Switzerland 2015, s. 219.

Metódy hlbokého učenia pre klasifikáciu rádiových galaxií

¹Kristián Hai LE THANH, ²Viera MASLEJ KREŠŇÁKOVÁ, ³Peter BUTKA

^{1,2,3}Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹kristian.hai.le.thanh@student.tuke.sk, ²viera.maslej.kresnakova@tuke.sk,
³peter.butka@tuke.sk

Abstrakt – Tento príspevok sa venuje využitiu metód hlbokého učenia v oblasti astrofyziky. Konkrétne ide o klasifikáciu rádiových galaxií podľa morfológie do štyroch tried. Predkladaná práca sa zameriava na konvulčné neurónové siete a tréningovanie modelov na malej dátovej vzorke. Tento problém je efektívne riešený augmentačnými technikami. Výsledky tejto práce presahujú doteraz publikované výsledky v danej doméne.

Kľúčové slová – rádiové galaxie, hlboké učenie, klasifikácia, augmentácia dát

I. ÚVOD

Úloha vytvorenia algoritmu hlbokého učenia bola vytvorená na základe medzinárodnej spolupráce s doktorkou Khadijou El Bouchefry, vedeckou pracovníčkou observatória South African Radio Astronomy Observatory (SARAO). SARAO momentálne pracuje na projekte vybudovania svetovo najväčšieho rádiového teleskopu, spoločne s oblasťou zberu dát o rozlohe 1 milión km², inak nazývanou Square Kilometer Array (SKA).

V tomto príspevku sa budeme zaoberať klasifikáciou 4 morfológických tried rádiových galaxií Fanaroff Riley typu I, II, Bent-Tail a kompaktných rádiových galaxií pomocou konvulčných neurónových sietí. Klasifikáciou morfológických typov rádiových galaxií, vieme zabezpečiť distribúciu potrebných informácií pre vedcov z oblasti astrofyziky, čím dosiahneme výrazné zlepšenie využitia výpočtových a ľudských zdrojov.

II. RÁDIOVÉ GALAXIE

Rádiové galaxie môžu byť klasifikované viacerými spôsobmi. Rozdelenie môže byť urobené na základe:

- 1) morfológie,
- 2) parametrov (spektrálny index, jas hostiteľskej galaxie na optickej infračervenej optickej dĺžke),
- 3) typu hostiteľskej galaxie.

Morfologická klasifikácia galaxií je prístup, ktorý je založený na zoskupovaní podľa vizuálneho vzhľadu. Takéto členenie vedcom umožňuje pochopiť vznik a vývoj galaxií, alebo ich vlastností, ako napríklad jas, okolité prostredie, hviezdnu hmotu, alebo rýchlosť tvorby hviezd. Jednotlivé morfológické typy rádiových galaxií, ktoré v našej práci klasifikujeme si detailnejšie popíšeme:

• Kompaktné rádiové galaxie

Kompaktné rádiové galaxie (angl. **compact**, skrt. **COMP**) sa vizuálne skladajú z jedného objektu. Prevažná väčšina rádiových zdrojov kompaktného typu má intenzitu žiarenia s frekvenciou 1.4GHz.

• Fanaroff-Riley

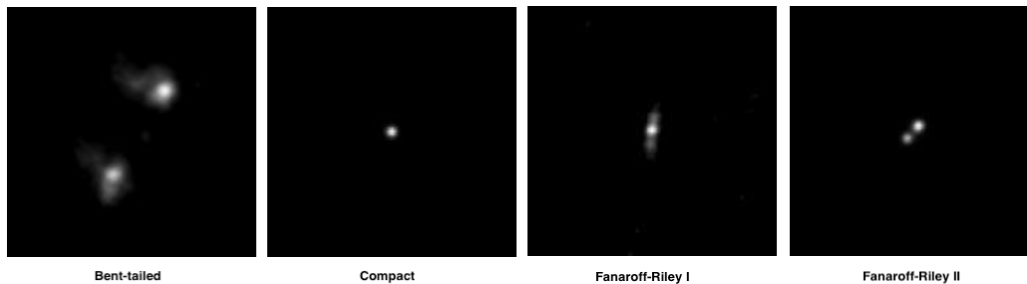
Relatívne polohy regiónov s vysokým a nízkym povrchovým jasom v lalokoch extragalaktických rádiových zdrojov a ich koreláciu s ich rádiovou svietivosťou ako prvý pozoroval Bernie Fanaroff a Julia Riley [1]. Schémou Fanaroff-Riley boli tradične klasifikované rozšírené rádiové galaxie ako **Fanaroff-Riley typu I** (FRI) a **Fanaroff-Riley typu II** (FR II). FRI a FR II sa rozlišujú na základe polohy menej a viac jasných oblastí v rozšírených oblastiach zdroja.

- FRI zdroje majú regióny s nízkym jasom ďalej od centrálnej galaxie a naopak najjasnejšie oblasti sú koncentrované na centrum galaxie.
- FRII trieda obsahuje svetelné rádiové zdroje s aktívnymi bodmi vo svojich lalokoch.

• Bent-Tailed rádiové galaxie

Niektoré rozšírené galaxie vykazujú viac ohýbané morfológie vyplývajúce z faktorov prostredia, alebo vnútorných vlastností. Takýto typ galaxie označujeme, ako Bent-Tailed a môžeme ich ďalej deliť do dvoch hlavných tried podľa výskytu ich prúdov. Širokouhlý chvost (angl. **Wide-Angled Tail**, skrt. **WAT**) je prvý typ, ktorého prúdy emitujú široké rádiové žiarenie v tvare písmena C v dôsledku vysokého dynamického tlaku vyplývajúceho z rýchleho pohybu hostiteľskej galaxie. Úzko-chvosté (angl. **Narrow-Angled Tail**, skrt. **NAT**) sú rádiové galaxie, ktorých zdroj sa nachádza vo vonkajších oblastiach zhluku, no vyznačuje sa difúznym chvostom, ktorý ma užší tvar písmena C [2].

Typy jednotlivých rádiových galaxií sú zobrazené na obrázku 1.



Obr. 1 Typy rádiových galaxií

III. ANALÝZA SÚČASNÉHO STAVU

Počas analýzy súčasného stavu vid' tabuľka 1 sme zistili, že existuje viac publikácií, ktoré sa venovali problematike klasifikácie rádiových galaxií. Ašak len jedna z uvedených publikácií [3] sa venovala klasifikácií práve do štyroch tried – COMPT, BENT, FRI, FRII. Autori publikácie uvádzajú dosiahnutú priemernú presnosť 97%. Naša práca sa tiež venuje klasifikácií rádiových galaxií do štyroch tried a preto považujeme túto publikáciu za state-of-the-art v tejto oblasti.

	Presnosť	Návratnosť	F1 miera	Počet
Výsledky dosiahnuté v [3]				
COMPT	0.98	0.98	0.98	1000
BENT	0.96	0.98	0.97	1000
FRI	0.98	1.00	0.99	1000
FRII	0.96	0.93	0.95	1000
priemer	0.97	0.97	0.97	4000
Výsledky dosiahnuté v [4]				
BENT	0.95	0.79	0.87	77
FRI	0.91	0.91	0.91	53
FRII	0.75	0.91	0.83	57
priemer	0.88	0.86	0.86	187
Výsledky dosiahnuté v [5]				
FRI	0.95	0.85	0.90	80
FRII	0.83	0.94	0.88	117
priemer	0.89	0.89	0.89	197

Tabuľka 1
Porovnanie výsledkov klasifikácie rádiových galaxií

IV. KLASIFIKÁCIA RÁDIOVÝCH GALAXIÍ PODĽA MORFOLOGICKÉHO TYPU

Tento príspevok sa zaoberá klasifikáciou rádiových galaxií podľa morfológie do 4 tried, kompaktné (COMPT) a rozšírené galaxie (FRI, FRII a BENT). Na tréningovanie sme použili novú dátovú množinu z katalógu Faint Images of the Radio sky at Twenty-Centimeters (skr. FIRST) [6]. Dátová vzorka označených obrázkov z katalógu FIRST s ktorou budeme pracovať obsahuje:

- 311 obrázkov triedy **BENT**,
- 278 obrázkov triedy **COMP**,
- 277 obrázkov triedy **FRI**,
- 276 obrázkov triedy **FRII**.

Predspracovanie obrázkov zahŕňalo orezanie obrázka na veľkosť 150×150 , odstránenie šumu pomocou `sigma_clipped_stats` funkcie s hodnotou $\sigma = 3$ a prevedenie obrázka na škálu sivej. Následne sme sa inšpirovali publikáciou [3], a obrázky rotovali o 90, 180 a 270 stupňov. Náhodným rozdelením sme vytvorili tréningovú a testovaciu podmnožinu v pomere 80 : 20.

Počas experimentov sme sa zameriavali hlavne na tvorbu vlastnej siete. Navrhli sme jednoduchú sieť, ktorá pozostáva z troch konvolučných vrstiev, ktoré ako aktivačnú funkciu využívajú ReLU. Sieť obsahuje aj jednu vrstvu Dropout a Max Pooling, pretože sme učenie siete potrebovali regulovať, optimalizovať¹. Architektúru modelu, s ktorým sme pracovali môžete vidieť na obrázku 2.

Layer (type)	Output Shape	Param #
conv2d_13 (Conv2D)	(None, 62, 62, 32)	320
max_pooling2d_13 (MaxPooling)	(None, 31, 31, 32)	0
conv2d_14 (Conv2D)	(None, 29, 29, 64)	18496
max_pooling2d_14 (MaxPooling)	(None, 14, 14, 64)	0
conv2d_15 (Conv2D)	(None, 12, 12, 194)	111938
max_pooling2d_15 (MaxPooling)	(None, 6, 6, 194)	0
dropout_5 (Dropout)	(None, 6, 6, 194)	0
flatten_5 (Flatten)	(None, 6984)	0
dense_9 (Dense)	(None, 194)	1355090
dense_10 (Dense)	(None, 4)	780
Total params: 1,486,624		
Trainable params: 1,486,624		
Non-trainable params: 0		

Obr. 2 Architektúra neurónovej siete

	Presnosť	Návratnosť	F1 miera	Počet
BENT	0.94	0.96	0.95	250
COMP	0.99	0.99	0.99	253
FRI	0.96	0.96	0.96	222
FRII	0.96	0.92	0.94	221
vážený priemer	0.96	0.96	0.96	916

Tabuľka 2
 Výsledky testovania klasifikátora bez použitia augmentačných techník

Ako je uvedené v tabuľke 2 vyššie, rozšírením tréningovej dátovej množiny sme dokázali dosiahnuť úspešnosť modelu 96%, čím sme však neprekonali doposiaľ najlepšie publikované výsledky. Preto sme sa rozhodli viac pozrieť na využitie ďalších augmentačných techník.

A. Augmentácia dát

Umelé zväčšenie tréningovej množiny vytvorením modifikovaných verzií dát vedie k lepšej schopnosti modelu generalizovať to, čo sa naučil na nových dátach. Tento proces nazývame **augmentácia dát**. Keďže sme nedosahovali požadovanú úspešnosť klasifikácie, tak sme sa rozhodli počas výskumu vyskúšať vplyv širokého spektra augmentačných techník, ako prevrátenie snímok v horizontálnom a vertikálnom smere, rotácie, zväčšenie, orezanie a mnoho ďalších. Na augmentáciu dát sme použili `ImageDataGenerator()`. Táto funkcia generuje tenzorové obrazové dáta spolu a rozširuje ich údaje v reálnom čase, opakovane, po dávkach.

Najlepšie výsledky sme dosahovali pomocou:

1) Horizontálne a vertikálne posunutie

Ide o augmentačnú techniku náhodného prevrátenia obrázkov v horizontálnom a vertikálnom smere (Obr. 3).

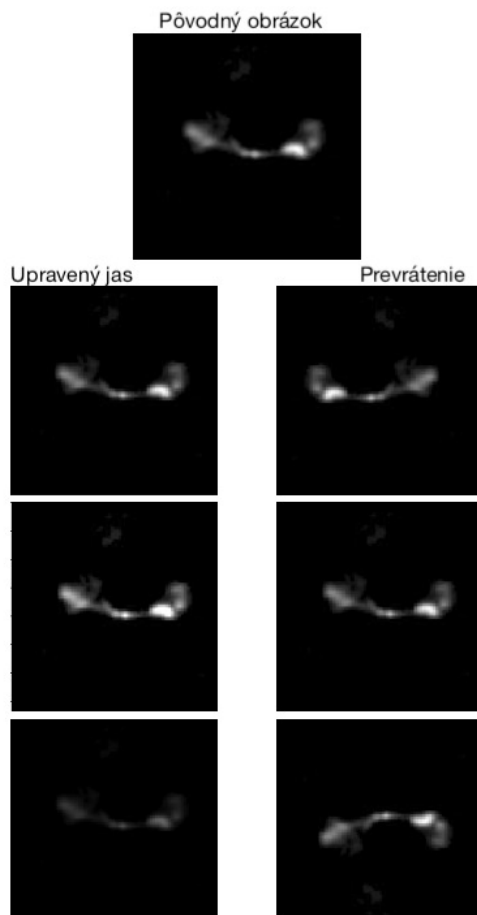
2) Úprava jasu

Táto technika sa nám osvedčila len v prípade, že sme jas pridávali. Pri uberaní jasu sme stratili dôležité informácie. Ukážka použitia týchto techník je zobrazená na obrázku 3.

¹Podrobnejšie o neurónových sieťach v knihe [7]

Pri augmentačných technikách je dôležité dávať pozor, aby boli zachované podstatné črty obrázku, a preto je dôležité dbať na nastavenie generátora nových obrázkov. Skúšali sme množstvo kombinácií typov augmentačných techník s rôznymi nastaveniami, ktoré sme použili ako vstup do modelu s parametrami, ktoré sa nám najviac osvedčili pri tréningu modelu bez augmentačných techník.

Na obrázku 3 je cielene zobrazený príklad, ktorý má zníženú hodnotu jasu, aby sme poukázali na dôležitosť nastavenia augmentačnej techniky. Ak by sme ponechali augmentáciu jasu s hodnotami, kedy je nový obrázok tmavší ako pôvodný, mohli by sme v niektorých situáciách stratiť morfológické vlastnosti galaxie. Preto budeme novým obrázkom jas len pridávať.



Obr. 3 Augmentácia úpravou jasu a prevrátením

Takto sme dokázali rozšíriť dátovú vzorku z pôvodných 1142 obrázkov na 8327 obrázkov. Celú dátovú množinu sme $3\times$ otočili a následne sme vytvorili 2 dátové vzorky, ktoré boli rozšírené o augmentované dáta. Obe dátové vzorky obsahovali 3660 obrázkov. Konvolučnú sieť sme najprv učili na dátovej vzorke s pridanými obrázkami, ktoré boli náhodne obrátené vo vertikálnom a horizontálnom smere. Neskôr sme natrénovaný klasifikátor znova učili na dátovej vzorke s pridanými obrázkami, ktoré mali náhodne pridaný jas o 0-50%. Celý priebeh učenia sme kontrolovali pomocou validačnej dátovej množiny. Výsledný klasifikátor sme otestovali na testovacej dátovej vzorke obsahujúcej 916 obrázkov, pričom výsledná priemerná úspešnosť klasifikácie dosahovala priemerne 99%. Všetky metriky vyhodnotené počas testovania sú v tabuľke 3.

	Presnosť	Návratnosť	F1 miera	Počet
BENT	0.99	1.00	0.99	250
COMP	1.00	1.00	1.00	253
FRI	1.00	1.00	1.00	222
FRII	1.00	0.99	1.00	221
priemer	1.00	1.00	1.00	916

Tabuľka 3
Výsledky klasifikácie pomocou augmentačných techník

	BENT	COMP	FRI	FRII
BENT	249	0	1	0
COMP	0	223	0	0
FRI	0	0	222	0
FRII	2	0	0	219

Tabuľka 4
Kontingenčná tabuľka výsledkov klasifikácie

Pre lepšie pochopenie ako naša sieť klasifikuje, sme si výsledky zobrazili vo forme chybovej matice v tvare vhodnej pre viac-triednu klasifikáciu. Z tabuľky 4 môžeme zhodnotiť, že model sa pomýlil 3 krát. V triede BENT klasifikoval zle 1 príklad do triedy FRII. Naopak, 2 príklady v triede FRII chybné klasifikoval do triedy BENT. Tieto triedy sú vizuálne veľmi podobné, no napriek tomu ich naša sieť dokáže veľmi presne klasifikovať. Na diagonále je počet správne klasifikovaných príkladov.

V. ZÁVER

V prvých fázach návrhu siete sme sa stretli s prvým problémom a to nedostatok anotovaných dát. V úvode sme sa rozhodli tento problém riešiť vytvorením nových obrázkov otáčaním obrázkov. Dostali sme tak 4-násobne veľkú dátovú množinu. Vytvorený model na týchto dátach sa síce priblížil najlepším doposiaľ publikovaným výsledkom, avšak našim cieľom bolo zvýšiť túto klasifikáciu. Keďže tieto teleskopy dokážu generovať obrovské množstvo dát, každá desatina vo výslednej presnosti predstavuje obrovskú stratu. Na základe spomínanej publikácii [3] sme sa rozhodli využiť augmentačné techniky.

Najväčší prínos bola zmena jasu, ktorá spolu s augmentačnou technikou náhodného preklápania obrázkov v horizontálnom a vertikálnom smere nám priniesla celkovú úspešnosť 99%. Tieto experimenty dokážeme zopakovať s maximálnou toleranciou $\pm 0,2\%$. Výsledky nielen presahujú doteraz publikované výsledky, ale poznatky v oblasti augmentácie dát, môžu v budúcnosti slúžiť na vytvorenie návodu pri riešení konvolučných neurónových sietí s nedostatkom označených dát.

POĎAKOVANIE

Táto práca bola podporená APVV grantom – APVV-16-0213.

LITERATÚRA

- [1] B. L. Fanaroff and J. M. Riley, "The morphology of extragalactic radio sources of high and low luminosity," *Monthly Notices of the Royal Astronomical Society*, vol. 167, no. 1, pp. 31P–36P, 1974.
- [2] S. Giacintucci and T. Venturi, "Tailed radio galaxies as tracers of galaxy clusters. serendipitous discoveries with the gmrt," *Astronomy & Astrophysics*, vol. 505, no. 1, pp. 55–61, 2009.
- [3] W. Alhassan, A. R. Taylor, and M. Vaccari, "The FIRST Classifier: Compact and extended radio galaxy classification using deep Convolutional Neural Networks," *Monthly Notices of the Royal Astronomical Society*, 2018.
- [4] A. K. Aniyar and K. Thorat, "Classifying Radio Galaxies with the Convolutional Neural Network," *The Astrophysical Journal Supplement Series*, 2017.
- [5] H. Tang, A. M. M. Scaife, and J. P. Leahy, "Transfer learning for radio galaxy classification," *Monthly Notices of the Royal Astronomical Society*, vol. 488, no. 3, pp. 3358–3375, 07 2019.
- [6] R. H. Becker, R. L. White, and D. J. Helfand, "The first survey: faint images of the radio sky at twenty centimeters," *The Astrophysical Journal*, vol. 450, p. 559, 1995.
- [7] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.

Anotácia a detekcia lineárnych objektov v prehliadke oblohy

¹Peter KULBAGA, ²Viera MASLEJ KREŠŇÁKOVÁ, ³Peter BUTKA

^{1,2,3}Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹peter.kulbaga@student.tuke.sk, ²viera.maslej.kresnakova@tuke.sk,
³peter.butka@tuke.sk

Abstrakt – Táto práca sa zaoberá problematikou anotácie dát, crowdsourcingu a následnom využití metód hlbokého učenia pri klasifikácii lineárnych objektov zo snímok oblohy. Cieľom tejto práce bolo prostredníctvom anotačného nástroja získať čo najväčšie množstvo anotácií v rámci vybraných astronomických dát a následne využiť tieto anotácie na vytvorenie klasifikačného modelu na báze konvolučných neurónových sietí.

Kľúčové slová – crowdsourcing, astronómia, detekcia lineárnych objektov, hlboké učenie, konvolučné neurónové siete

I. ÚVOD

Crowdsourcing je metóda ktorá sa využíva pri riešení problémov, ktoré nie je schopný vyriešiť jednotlivec, alebo malá skupina ľudí v rozumnom čase. Tieto problémy však veľakrát nie sú ani tak veľmi komplikované, ako skôr časovo náročné. V dnešnej dobe sme schopní využívať rôzne technológie, ktoré nám umožňujú automatizovať veľa rutinných operácií, avšak stále existujú situácie, kedy nie je možné nahradiť človeka za umelú inteligenciu [1].

Na riešenie nášho projektu sme si vybrali platformu Zooniverse. Zooniverse je jednou s platforiem pre crowdsourcing, ktorá poskytuje širokú základňu užívateľov. Na rozdiel od platformy ako Amazon Mechanical Truck sa Zooniverse zameriava čisto na vedecké projekty, pri ktorých je potrebné využiť metódu crowdsourcing. Zooniverse vznikol v roku 2007 v spolupráci Adler Planetarium a Oxfordskou univerzitou vo Veľkej Británii. Má viac než 1,5 milióna registrovaných užívateľov na celom svete [2].

II. DETEKCIA LINEÁRNYCH OBJEKTOV

Meteor je svetelný jav vyplývajúci z prechodu meteoroidu, mikrometeoroidu, kométy alebo asteroidu cez zemskú atmosféru. Viditeľnosť meteoru je spôsobená atmosférickým tlakom, ktorý ohrieva meteoroid tak, že svieti a vytvára žiarivú stopu plynov a roztavených častíc padajúceho telesa. Plyny zahŕňajú odparený meteoroidný materiál a atmosférické plyny, ktoré sa zahrievajú, keď meteoroid prechádza atmosférou. Väčšina meteorov svieti asi sekundu. Priemerne jasný meteor má jasnosť približne 0 až 3 magnitúdy [3].

Meteory sú objekty, ktoré sa vyznačujú špecifickými vlastnosťami oproti iným objektom ako napríklad satelity. Dôležitým parametrom je výška v ktorej meteory začínajú byť viditeľné. Meteory sú viditeľné vo výške 120–75 km a zvyknú sa rozpadáť vo výškach 95–50 km nad zemským povrchom. Ak to porovnáme s výškou, v ktorej sa vyskytujú satelity, t. j. 300–32000 km, je to značný rozdiel. Okrem toho, že sa u meteorov očakáva výraznejšia svetelná stopa, môžeme u nich pozorovať taktiež zmenu hrúbky stopy, ktorá zvykne byť v počiatočných štádiách prechodu atmosférou tenšia. Jas ktorý je vyprodukovaný prechodom meteoritu zemskou atmosférou má úplne iné vlastnosti ako jas pochádzajúci zo satelitov.

Satelity vykazujú periodickú zmenu jasu, ktorá je spôsobená odrazom slnečného žiarenia zo solárnych panelov. Dôležitým faktorom je aj rýchlosť akou vstupujú meteory do zemskej atmosféry, v porovnaní s rýchlosťou satelitov obiehajúcich zem. Najpomalšie meteory dosahujú pri vstupe do atmosféry rýchlosť 11 km/s a najrýchlejšie môžu dosahovať rýchlosť až 70 km/s. Satelity zvyknú podľa [4] obiehať vo výške 640 km nad zemským povrchom pričom platí, že čím vyššie sa satelit nachádza v porovnaní so Zemou, tým pomalšia je jeho rýchlosť.

Rýchlosť satelitov je pomalšia v porovnaní s najpomalšími meteormi, ktoré môžeme pozorovať pri vstupe do atmosféry. Vďaka týmto odlišnostiam je možné rozlíšiť, či objekt ktorý zachytila kamera je skutočne meteor, alebo ide len o nejaký iný lineárny objekt, ktorý môžeme sledovať na oblohe.

III. DATASET

Dáta, s ktorými sme pracovali, pochádzajú z projektu SDSS [5]. Na tréovanie kvalitného modelu sme potrebovali stiahnuť čo najväčšiu množinu snímok, obsahujúcu nejaký lineárny objekt. Snímky, na ktorých je zaznamenaný pohybujúci sa objekt na oblohe ako napríklad meteor, satelit, stanica ISS je považovaný za anomáliu resp. chybu. BOSS, skratka pre Baryon Oscillation Spectroscopic Survey, je zložka, v ktorej sa nachádzajú dôležité dáta pre našu prácu, keďže obsahuje dáta, v ktorých sa často vyskytujú chyby a anomálie.

Podľa [6] existujú dva hlavné spôsoby ako pristupovať k údajom v databáze SDSS:

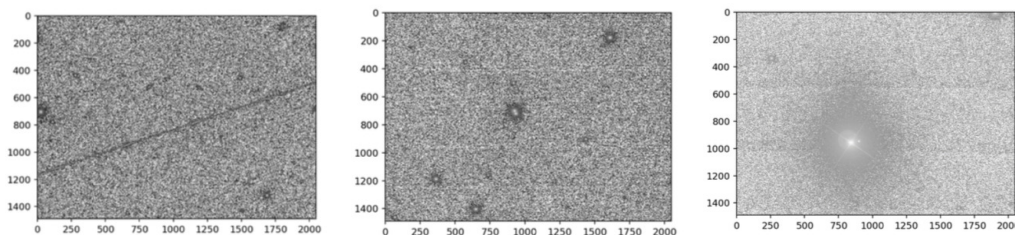
- vedecký archívny server SAS6 pre interaktívne vyhľadavanie spektier a obrazových mozaík,
- SkyServer umožňujúci prístup k archívnemu serveru katalógu CAS.

CAS umožňuje sťahovať surové dáta, ktoré je možné použiť na preberanie kalibrovaných snímok oblohy v oblasti filtrov a optických alebo infračervených spektier určených objektov (ak existujú). Dáta môžu byť stiahnuté ako súbory FITS alebo JPEG ak sa jedná o farebné digitálne snímky oblohy.

Na stiahnutie **pozitívnych snímok** sme využili dáta, ktoré pochádzali z projektu názvom Linear feature detection algorithm (LFDA) [7]. Algoritmus LFDA bol navrhnutý tak, aby poskytoval optimálny výpočtový výkon. Preto algoritmus pozostáva z 3 úrovní, pričom každá úroveň je časovo náročnejšia a taktiež poskytuje menej spoľahlivé výsledky. Prostredníctvom daných úrovní je možné určiť výskyt jasných alebo menej výrazných lineárnych objektov. V prvom kroku sa odstránia všetky objekty, ktoré by sa mohli na fotografii vyskytovať, ako napr. hviezdy či galaxie. Následne sa algoritmus snaží identifikovať jasne viditeľný lineárny objekt. V prípade, že algoritmus identifikuje tento lineárny objekt dôjde k ukončeniu procesu, pričom sa zaznamená výstup. V prípade že algoritmus nedokáže identifikovať žiadny lineárny objekt, prejde do ďalšej fázy v ktorej sa pokúša nájsť objekt, ktorý nie je veľmi jasný respektíve je zahmlený. V oboch krokoch dochádza k viacnásobnej kontrole. Ak jeden z testov preukáže neprítomnosť lineárneho objektu fotografia je automaticky klasifikovaná ako negatívna. Spolu sme získali 30902 snímok.

Pre stiahnutie **negatívnych snímok** sme nemali k dispozícii žiaden zdroj, na základe ktorého by sme mohli získať overené údaje. Preto sme sa rozhodli stiahnuť údaje zo servera SDSS, pričom sme očakávali, že množina údajov bude obsahovať prevažne negatívne snímky, keďže zachytenie lineárneho objektu kamerou SDSS bolo považované za anomáliu, ktorá sa vyskytuje len sporadicky. Vytvorili sme skript, ktorý sťahoval dané údaje v náhodnom poradí, pričom bolo stiahnutých 58000 negatívnych snímok.

Cieľom našej práce bolo overiť dostatočne veľkú množinu údajov, pričom som mal pre tieto účely využiť metódu crowdsourcingu a následne na daných dátach naučiť model na báze konvulčných neurónových sieti.



Obr. 1 Prvý obrázok predstavuje pozitívnu snímku, druhý a tretí negatívnu snímku

A. Vyhodnotenie anotačného projektu

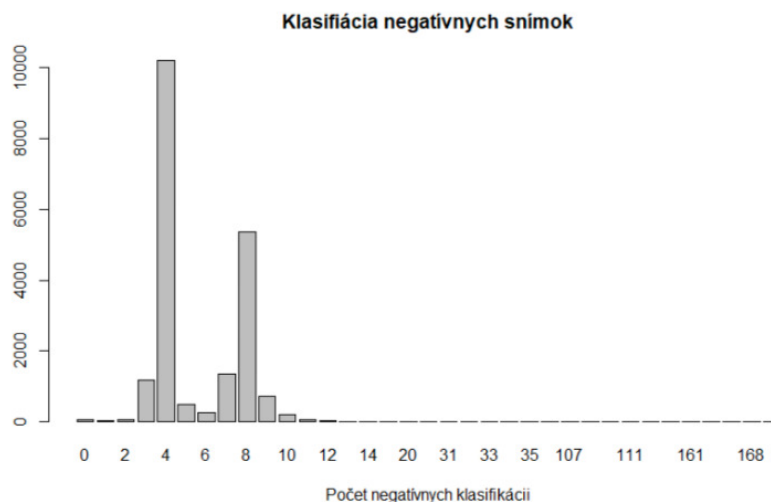
V grafe 2 môžeme vidieť výsledky klasifikácie pozitívnych snímok. Na x -ovej osi sa nachádzajú hodnoty klasifikácii. Pri hodnote 0 sa dobrovoľníci 4 krát zhodli, že sa na danej snímke nenachádza žiaden lineárny objekt aj napriek tomu, že pred anotáciou bola daná snímka považovaná za pozitívnu. Snímky ktoré mali hodnotu menšiu ako 1 bolo potrebné odstrániť. Pri hodnote 2 bolo potrebné znovu manuálne overiť, či ide o negatívny resp. pozitívny snímok. Snímky ktoré mali hodnotu väčšiu ako 2 boli automaticky klasifikované ako pozitívne. Aj napriek tomu, že počet klasifikácii každej fotografie bol nastavený na hodnotu 4, v grafe sa nachádzajú hodnoty v rozsahu 0–149. To bolo spôsobené nárazovým klasifikovaním snímok, pri ktorom dochádzalo k situáciám, kedy bola snímka zobrazená v tom istom čase viacerým dobrovoľníkom. Preto snímky, ktoré mali hodnotu väčšiu ako 4 sú v nasledujúcej tabuľke zarátané v hodnote 4 a sú považované za pozitívne.



Obr. 2 Graf zobrazujúci výsledky klasifikácie pozitívnych snímok

Po overení snímok môžeme skonštatovať, že 13736 pozitívnych snímok bolo klasifikovaných správne a v 1285 pozitívnych snímkach bola detegovaná chyba. Teda 3,55% zo všetkých pozitívnych snímok, ktoré boli detegované algoritmom LFDA bolo chybných a preto ich bolo potrebné odstrániť.

V grafe 3 môžeme vidieť výsledky klasifikácie negatívnych snímok. Na x -ovej osi sa nachádzajú hodnoty klasifikácií. Pri hodnote 0 sa dobrovoľníci 4 krát zhodli, že na danej snímke sa nachádza lineárny objekt, aj napriek tomu, že pred anotáciou bola daná snímka považovaná za negatívnu. Snímky ktoré mali hodnotu menšiu ako 1, bolo potrebné odstrániť. Pri hodnote 2 bolo potrebné znovu manuálne overiť, či ide o negatívny resp. pozitívny snímok. Snímky, ktoré mali hodnotu väčšiu ako 2 boli automaticky klasifikované ako negatívne. Snímky, ktoré mali hodnotu väčšiu ako 4 boli automaticky považované za negatívne.



Obr. 3 Graf zobrazujúci výsledky klasifikácie negatívnych snímok

V množine negatívnych dát sa nachádzalo 613 chybných snímok, ktoré bolo potrebné odstrániť. V percentuálnom vyjadrení z množiny negatívnych dát bolo chybných 1,66% snímok. Teda môžeme skonštatovať, že náš predpoklad o veľmi nízkom výskyte pozitívnych snímok pri náhodnom sťahovaní negatívnych dát z databázy SDSS sa ukázal ako pravdivý.

IV. MODELOVANIE

Model je tvorený z 10 konvolučných vrstiev, v ktorých je použitá maska o veľkosti 3x3 pixlov. V danom modeli bola využívaná aktivačná funkcia ReLu. Aby sme zachovali rozmer vstupných

Architektúra modelu							
Operation group	Operation	Layer name	Number of filters	Filter size	Stride size	Padding size	Output size
Group_0	Input image	Input layer	NA	NA	NA	NA	128x128x1
Group_1	Convolution (2 times)	Convolution layer	8	3x3x1	1x1	1x1	128x128x8
		ReLu layer	NA	NA	NA	NA	128x128x8
	Pooling	Max pooling layer	1	2x2	2x2	0	64x64x8
Group_2	Convolution (2 times)	Convolution layer	32	3x3x1	1x1	1x1	64x64x32
		ReLu layer	NA	NA	NA	NA	64x64x32
	Pooling	Max pooling layer	1	2x2	2x2	0	32x32x32
Group_3	Convolution (3 times)	Convolution layer	64	3x3x1	1x1	1x1	32x32x64
		ReLu layer	NA	NA	NA	NA	32x32x64
	Pooling	Max pooling layer	1	2x2	2x2	0	16x16x64
Group_3	Convolution (3 times)	Convolution layer	128	3x3x1	1x1	1x1	16x16x128
		ReLu layer	NA	NA	NA	NA	16x16x128
	Pooling	Max pooling layer	1	2x2	2x2	0	8x8x128
Group_4	Fully connected layer (FCL)	1st FCL	NA	NA	NA	NA	4096
		ReLu layer	NA	NA	NA	NA	4096
		2st FCL	NA	NA	NA	NA	4096
		ReLu layer	NA	NA	NA	NA	4096
		3st FCL	NA	NA	NA	NA	2
		Softmax layer	NA	NA	NA	NA	2

Obr. 4 Architektúra modelu

dát, pred každou konvolučnou vrstvou rozšírime obrázok o okraj v horizontálnom aj vertikálnom smere doplnením 0 hodnôt (zero-padding). Po prvej a druhej konvolúcii, v ktorej bolo použitých 8 filtrov, prichádza vrstva MaxPooling, ktorá zredukuje veľkosť snímky na polovicu a nastane prvý Dropout váh. Príkaz Dropout slúži na náhodne vylúčenie váh s cieľom zabrániť preučeniu siete. Celý tento proces sa niekoľko krát opakuje, pričom ako môžeme vidieť na obrázku 4, sa postupne zvyšuje počet používaných filtrov. Na konci sa nachádzajú dve plne prepojené vrstvy, v ktorých je použitý vektor rozhodovacích hodnôt o veľkosti 4096 a taktiež Dropout váh. Poslednej časti modelu sa nachádza ešte jedna plne prepojená vrstva, v ktorej sa klasifikujú snímky do jednej z dvoch klasifikačných tried. Podrobnejšie o princípoch hlbokého učenia v knihe [8].

Pri najúspešnejšom experimente sme dosiahli výsledky:

- **Úspešnosť** – 99,12%
- **Presnosť** – 99%
- **Návratnosť** – 99%
- **F1 miera** – 99%
- **Chybová funkcia** – 0,042

V. ZÁVER

Hlavným cieľom práce bolo prostredníctvom techniky zvanej crowdsourcing overiť množinu astronomických dát z projektu SDSS, ktorý sa zameriava na prieskum oblohy. Pri anotovaní dát sme využili existujúcu platformu s názvom Zooniverse, ktorá poskytuje užívateľské rozhranie, v ktorom som realizovali svoj projekt. V rámci daného crowdsourcingového projektu, bolo potrebné overiť čo najväčšiu množinu astronomických dát. Podarilo sa nám klasifikovať 52021 snímok,

ktoré boli následne použité pri zlepšovaní dodaného modelu, ktorý bol postavený na báze konvulučných neurónových sietí. Daný model slúžil pre účely identifikácie lineárnych objektov z astronomických snímok. V percentuálnom vyjadrení z množiny negatívnych dát bolo chybných 1,6% snímok. Teda môžeme skonštatovať, že náš predpoklad o veľmi nízkom výskyte pozitívnych snímok pri náhodnom sťahovaní negatívnych dát z databázy SDSS sa ukázal ako pravdivý. Zo všetkých pozitívnych snímok, ktoré boli detegované algoritmom LFDA bolo chybných 8,55% dát. Najlepší model, ktorý sa nám podarilo vytvoriť dosahoval 99% úspešnosť, pričom metriky presnosť a návratnosť dosiahli hodnotu 99%. Išlo o model, ktorý obsahoval len údaje, ktoré boli získané počas realizácie projektu anotácie dát.

POĎAKOVANIE

Táto práca bola podporená APVV grantom – APVV-16-0213.

LITERATÚRA

- [1] J. Howe, *Crowdsourcing: How the power of the crowd is driving the future of business*. Random House, 2008.
- [2] L. Trouille. (2017, oct) American astronomical society. [Online]. Available: <https://aas.org/posts/news/2017/06/build-your-own-citizen-science-project-zooniverse>
- [3] A. E. Rubin and J. N. Grossman, “Meteorite and meteoroid: New comprehensive definitions,” *Meteoritics & Planetary Science*, vol. 45, no. 1, pp. 114–122, 2010.
- [4] S. Holzner. (2018, feb) Dummies. [Online]. Available: <https://www.dummies.com/education/science/physics/how-to-calculate-a-satellites-speed-around-the-earth>
- [5] D. G. York and col., “The sloan digital sky survey: Technical summary,” *The Astronomical Journal*, vol. 120, no. 3, pp. 1579–1587, sep 2000. [Online]. Available: <https://doi.org/10.1086%2F301513>
- [6] T. Barclay. (2000, mar) Sloan digital sky survey. [Online]. Available: <http://skyserver.sdss.org/dr12/en/sdss/data/data.asp>
- [7] D. Bektešević and D. Vinković, “Linear feature detection algorithm for astronomical surveys–i. algorithm description,” *Monthly Notices of the Royal Astronomical Society*, vol. 471, no. 3, pp. 2626–2641, 2017.
- [8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.

Analýza dát v doméne astronomických pozorovaní

¹Martin HARMAN, ²Viera MASLEJ KREŠŇÁKOVÁ, ³Peter BUTKA

^{1,2,3}Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹martin.harman@student.tuke.sk, ²viera.maslej.kresnakova@tuke.sk,
³peter.butka@tuke.sk

Abstrakt – Cieľom tejto práce je analýza dát z astronomických pozorovaní premenných hviezd pomocou samoorganizujúcich sa máp. S využitím programovacieho jazyka R, predspracovaním dát sme analyzovali svetelné krivky premenných hviezd pomocou lomb-scargle periodogramu a vytvorili klasifikačný model.

Kľúčové slová – premenné hviezdy, analýza dát, klasifikácia, samoorganizujúce sa mapy

I. ÚVOD

Hlavným cieľom našej práce je analýza dát z astronomických pozorovaní oblohy, konkrétne sa jedná o analýzu premenných hviezd [1] z katalógu OGLE pomocou Kohonenových samoorganizujúcich sa máp a využitie takejto neurónovej siete na kategorizáciu hviezd.

Kohonenove siete sú typ samoorganizačných máp (SOM), ktoré vytvoril Tuevo Kohonen v osemdesiatych rokoch 20. storočia. Tieto neurónové siete nepotrebujú učiteľa a pri učení využívajú prístup rozšíreného konkurenčného učenia [2]. Základným princípom ich funkcie je zhuková analýza, t. j. schopnosť algoritmu, siete, nájsť určité vlastnosti a závislosti priamo v predkladaných tréningových dátach bez prítomnosti nejakej vonkajšej informácie. Kohonenove siete využívajú priestorovú reprezentáciu dát väčšinou v dvojrozmernom priestore a sú geometricky usporiadané vo forme vektora alebo matice. Vďaka tomuto usporiadaniu je možné určiť najbližšieho suseda. Susedia, ktorí sú najbližšie pri sebe si sú najviac podobní a naopak tí, čo sú od seba najďalej majú najmenšiu podobnosť. SOM má 2 základné typy topológií. Prvý typ je rektangulárny, ktorý má 4 najbližších susedov a druhý je hexagonálny, ktorý má až 6 najbližších susedov.

II. PREMENNÉ HVIEZDY

Premenné hviezdy sú hviezdy, ktoré menia svoju jasnosť. Zmeny jasností týchto hviezd môžu trvať len pár sekúnd alebo až niekoľko rokov. Poznáme viac než 150 000 premenných hviezd a o mnoho ďalších si myslíme, že by mohli byť tiež premenné [3].

Existuje viacero rozdelení premenných hviezd. Je možné ich rozdeliť do viacerých kategórií, ktoré sa potom delia medzi ďalšie podkategórie. Jedno z rozdelení je na fyzikálne a geometrické premenné hviezdy [4]:

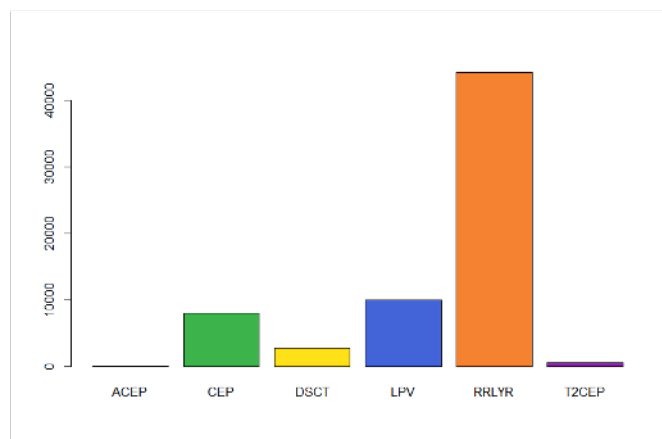
- 1) **Fyzikálne premenné hviezdy** – sú také hviezdy, ktoré menia jasnosť kvôli fyzikálnym vplyvom. Tieto hviezdy rozdeľujeme na dva typy:
 - **Pulzujúce** premenné hviezdy periodicky menia svoju jasnosť.
 - **Expandujúce** premenné hviezdy menia svoju jasnosť náhlou zmenou, ktorá môže byť zapríčinená výbuchom alebo vzplanutím.
- 2) **Geometrické premenné hviezdy** – príčinou zmeny jasností u týchto hviezd býva spôsobená pozorovaním zo Zeme, kedy môže dôjsť k zákrytu tejto hviezdy.

Tento príspevok sa zaoberá klasifikáciou šiestich konkrétnych tried premenných hviezd a to:

- 1) **Cepheids** (CEP) sú pulzujúce premenné hviezdy s periódami v rozmedzí 1 až 135 dní.
- 2) **Anomalous Cepheids** (ACEP) sú podtyp cefeidov s dĺžkou periódy 0,4 až 2 dni. Pretože si sú veľmi podobné s RR Lyrae, sú často zamieňan. Oproti nim majú ale výrazne jasnejšiu svietivosť.
- 3) **Type II Cepheids** (T2CEP) sú podtyp cefeidov, označované ako tzv. „low-mass“, s dĺžkou periódy 10 až 100 dní.

- 4) **RR Lyrae (RRLYR)** sú pulzujúce premenné hviezdy s krátkou periódou, zvyčajne 0,05 až 1,2 dni. Sú staršie a menej masívne ako cefeidy.
- 5) **Delta Scuti Type (DSCT)** sú pulzujúce premenné hviezdy dosahujúce krátkych períod v rozmedzí 0,1 až 0,2 dní. Tvary svetelných kriviek týchto hviezd sa zvyčajne od seba rôzne líšia.
- 6) **Long Period Variables (LPV)** názov odkazuje na rôzne typy premenných hviezd, ktoré dosahujú dlhých períod.

Nakoľko tieto dáta sú nevybalansované (viď. Obrázok 1), rozhodli sme sa robiť experimenty na štyroch menších dátových množinách, v ktorých je rozdelenie tried rovnomernejšie.



Obr. 1 Distribúcia typov klasifikovaných tried

III. MODELOVANIE DÁT

Na vytvorenie modelu sme využili 4 dátové množiny a tú, ktorá mala najlepšie výsledky sme ďalej použili na vytvorenie klasifikačného modelu. Pri vytváraní modelu sme testovali rôzne veľkosti a typy SOM, pričom sme menili počet iterácií.

Pri vyhodnocovaní sme sledovali nasledujúce metriky:

- 1) **Čistota uzlov** – priemerné zastúpenie majoritnej triedy

$$\frac{\sum \text{Zasúpenie majoritnej triedy}}{\text{Počet všetkých uzlov}} \cdot 100 \quad (1)$$

- 2) **Zmiešané uzly** – počet uzlov (v percentách), v ktorých mala majoritná trieda menšie zastúpenie než 75%

$$\frac{\text{Počet zmiešaných uzlov}}{\text{Počet všetkých uzlov}} \cdot 100 \quad (2)$$

DM	seed	map	typ	iter	cistota	neciste
Data 1	42	6x5	hexagonal	250	75,23%	40%
Data 1	42	6x5	rectangular	250	74,55%	43,33%
Data 1	42	6x5	rectangular	1000	78,08%	38,89%
Data 1	42	7x6	rectangular	250	74,45%	52,38%
Data 2	42	8x7	rectangular	250	78,74%	39,29%
Data 2	42	8x7	rectangular	500	79,01%	37,50%
Data 2	42	8x7	hexagonal	1500	79,04%	41,07%
Data 3	42	9x9	rectangular	250	81,71%	27,16%
Data 3	42	10x9	rectangular	250	81,11%	30%
Data 3	42	10x10	rectangular	250	82,72%	28%
Data 4	42	15x15	rectangular	1000	86,33%	24%
Data 4	42	15x15	rectangular	2000	87,27%	22,22%
Data 4	42	15x15	hexagonal	500	86,14%	22,67%

Tabuľka 1
 Porovnanie výsledkov

Tento postup sme použili pre naše 4 dátové množiny. Výsledky týchto pokusov v podobe čistoty uzlov a počet zmiešaných uzlov sme zapisovali do tabuľky spolu s ďalšími údajmi ako použitá dátová množina, typ a rozmery SOM, počet iterácií. V tabuľke 1 sa nachádzajú najlepšie výsledky týchto pokusov.

Na základe výsledkov v tabuľke 1 môžeme povedať, že pokusy s vyšším počtom iterácií dosahovali väčšiu čistotu zhľukov než tie s menším počtom iterácií. Čo sa týka veľkosti mapy, tak to záležalo na veľkosti dátovej množiny. Ak bola mapa príliš malá, tak vzniklo veľa zhľukov s nízkou čistotou. Ak bola príliš veľká, tak sa začali objavovať prázdne zhľuky. Medzi hexagonálnym a rektangulárnym typom neboli veľké rozdiely. Pri veľkosti dátovej množiny sa dalo predpokladať, že čím bude väčšia, tým lepšie sa model dokáže naučiť, a tým bude dosahovať lepšie výsledky.

Najlepšie výsledky dosahuje dátová množina 4, ktorá obsahuje:

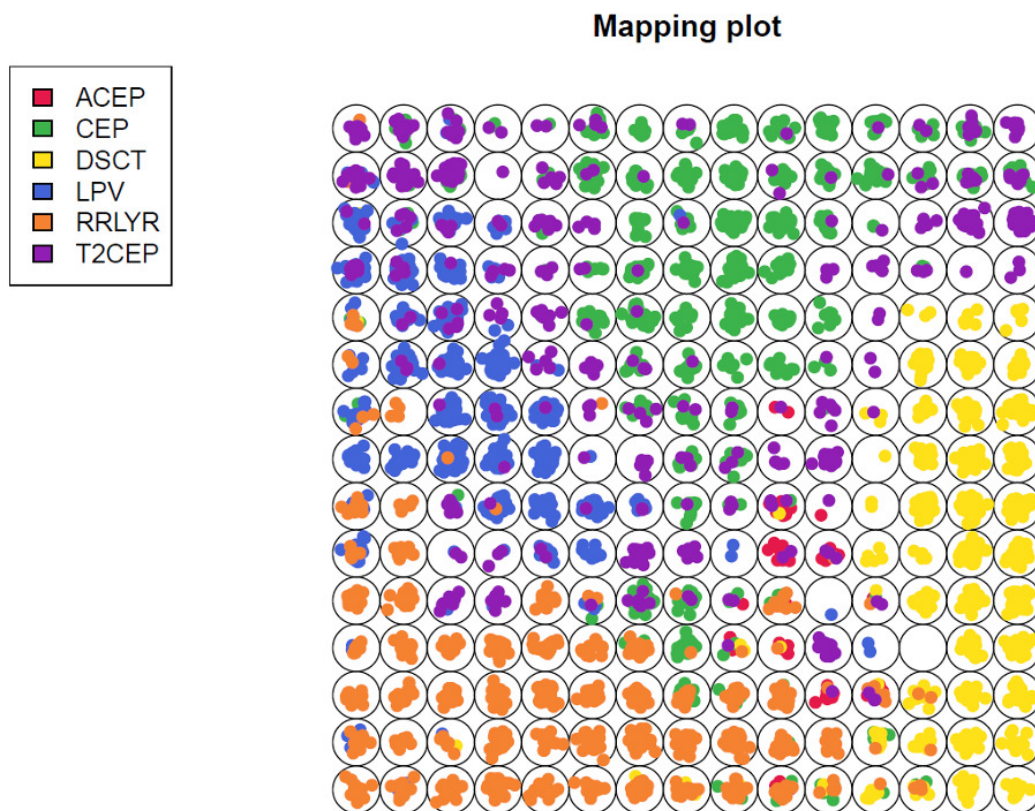
- ACEP – 87 objektov
- CEP – 1000 objektov
- DSCT – 1000 objektov
- LPV – 1000 objektov
- RRLYR – 1000 objektov
- T2CEP – 586 objektov

IV. EXPERIMENTY

Vytvorili sme viacero experimentov, na tomto mieste si však poíšeme najlepšie z nich.

A. Experiment 1

Prvý experiment je rektangulárna SOM s počtom iterácií 2000, veľkosťou mapy 15×15 . Výsledná distribúcia tried je zobrazená na 2.



Obr. 2 Experiment 1 – distribúcia tried

Vďaka obrázku 2 vidíme rozmiestnenie jednotlivých typov hviezd na mape. Môžeme pozorovať, že hviezdy typu RRLYR a DSCT majú najčistejšiu distribúciu v uzloch. Typ hviezd CEP je výrazne ohraničený typom T2CEP. Mohlo to byť spôsobené ich veľkou podobnosťou, keďže T2CEP je podtyp CEP a v niektorých uzloch dochádza k ich zmiešaniu, ako aj v prípade T2CEP a LPV. Napriek tomu, že k dispozícii sme mali iba malú vzorku typu ACEP, tak SOM dokázal väčšinu umiestniť v jednom uzle.

Čistota uzlov v tomto prípade je 86,7%, zmiešané uzly majú zastúpenie 21,33%. Po aplikovaní tohto modelu na tréningovú množinu, ktorá tvorí 20% z pôvodnej množiny sme dospeli k výsledkom:

- **Úspešnosť** – 89,45%
- **Presnosť** – 85,39%
- **Návratnosť** – 84,73%

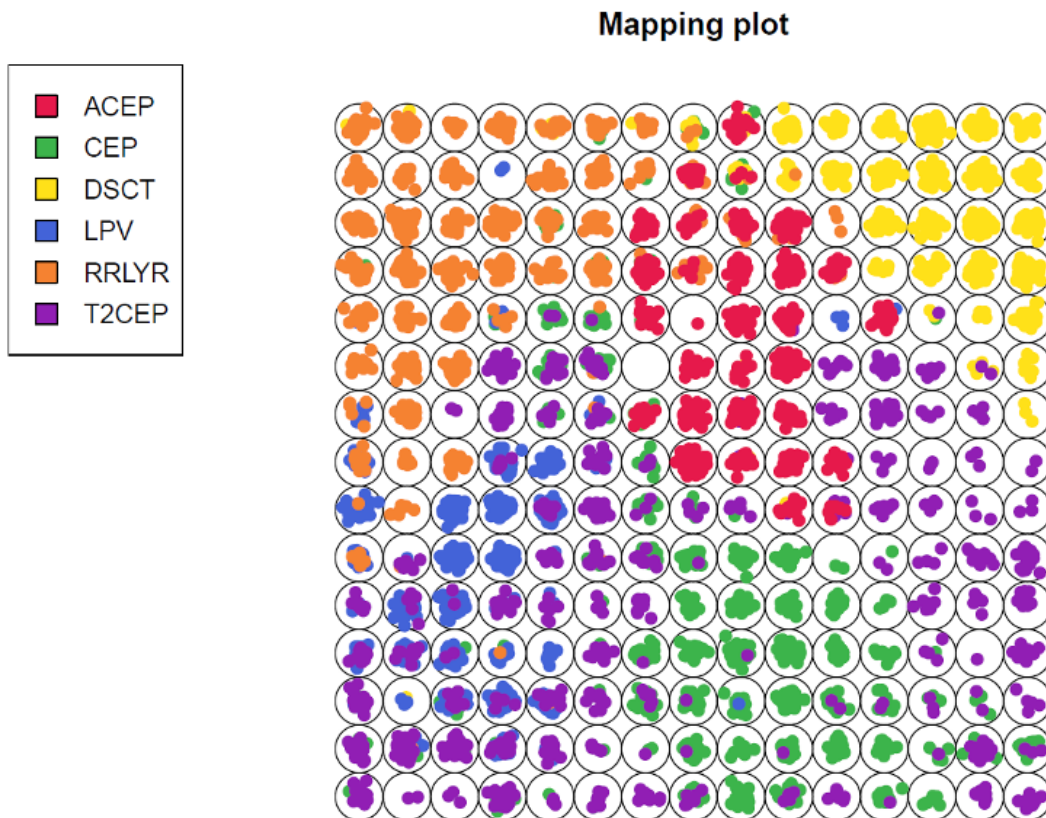
V kontingenčnej tabuľke 2 vidíme, že najmenšiu úspešnosť mala klasifikácia hviezdy typu T2CEP, pri ktorom model nesprávne predpokladal, že patrí k typu LPV alebo CEP. Vzhľadom na početnosť typu ACEP bola jeho klasifikácia pomerne úspešná, aj keď došlo k pár chybným klasifikáciám, kedy boli označené ako RRLYR.

	ACEP	CEP	DSCT	LPV	RRLYR	T2CEP
ACEP	15	1	1	0	0	5
CEP	0	170	0	0	3	19
DSCT	0	0	192	0	4	0
LPV	0	6	0	199	3	26
RRLYR	5	6	1	2	215	0
T2CEP	1	12	2	5	0	74
	71,43%	87,18%	97,96%	96,60%	95,56%	59,68%

Tabuľka 2
 Experiment 1 – kontingenčná tabuľka

B. Experiment 2

Zopakovali sme experiment 1, ale využijeme oversampling na vybalancovanie dát. Získali sme tak dáta so 6000 objektmi, 1000 objektov pre každý typ. Výsledky distribúcie tried sú znázornené na obrázku 3.



Obr. 3 Experiment 2 – distribúcia tried

Výsledné hodnoty SOM v tomto experimente predstavovali 93,71% pre čistotu uzlov a 10,22% pre zmiešané uzly. Výsledná klasifikácia tak dosahovala:

- **Úspešnosť** – 88,62%
- **Presnosť** – 88,65%
- **Návratnosť** – 88,63%

Celkovú úspešnosť klasifikácie sa nám síce nepodarilo zlepšiť, napriek tomu sa nám podarilo dosiahnuť oveľa lepšiu úspešnosť pri klasifikácii typu ACEP a o niečo lepšiu klasifikáciu typu T2CEP. Výrazne nám klesla úspešnosť klasifikácie typu RRLYR, kde najčastejšie dochádzalo k zámene s typom ACEP.

	ACEP	CEP	DSCT	LPV	RRLYR	T2CEP
ACEP	200	7	0	0	20	10
CEP	0	162	4	2	3	26
DSCT	0	0	196	0	5	0
LPV	0	3	1	192	5	12
RRLYR	0	5	2	3	187	0
T2CEP	0	19	2	9	2	153
	100,00%	82,65%	95,61%	93,20%	84,23%	76,12%

Tabuľka 3
Experiment 2 – kontingenčná tabuľka

V. ZÁVER

Cieľom tejto práce bola analýza premenných hviezd pomocou samoorganizujúcich sa máp a vytvorenie klasifikačného modelu. Pri experimentoch sme vybratú dátovú množinu, ktorá obsahovala údaje o fáze premennej hviezdy, rozdelili na tréningovú a testovaciu. Tréningová bola použitá na vytvorenie modelu SOM. Testovacia množina bola použitá pri klasifikácii pomocou natrénovaného modelu. Experiment dosiahol pomerne dobré výsledky. Niektoré typy premenných hviezd ale boli často nesprávne klasifikované, pretože si boli podobné. Tento problém by mohol byť vyriešený pridaním ďalších atribútov, napríklad amplitúdy a pridaním ďalšieho parametra, ktorý by mohol pomôcť tieto podobné typy od seba viac odlíšiť. Pri experimente, kde sme použili dáta obsahujúce údaje o hodnotách magnitúdy vo svetelnej krivke premennej hviezdy sme zaznamenali vysoký počet zmiešaných uzlov. Tento problém sme čiastočne vyriešili vybalansovaním dát pomocou oversamplingu.

POĎAKOVANIE

Táto práca bola podporená APVV grantom – APVV-16-0213.

LITERATÚRA

- [1] P. Ledoux and T. Walraven, "Variable stars," in *Astrophysics II: Stellar Structure/Astrophysik II: Sternaufbau*. Springer, 1958, pp. 353–604.
- [2] P. Šinčák and G. Andrejková, "Neurónové siete inžiniersky prístup (1. diel)," *Elfa: Kosice*, 1996.
- [3] G. Bowen, "Dynamical modeling of long-period variable star atmospheres," *The Astrophysical Journal*, vol. 329, pp. 299–317, 1988.
- [4] H. A. Smith, *RR Lyrae stars*. Cambridge University Press, 2004, vol. 27.

Interaktívna vizualizácia dát zo satelitných pozorovaní

¹Kristián MAŤAŠOVSKÝ, ²Viera MASLEJ KREŠŇÁKOVÁ, ³Peter BUTKA

^{1,2,3}Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹kristian.matasovsky@student.tuke.sk, ²viera.maslej.kresnakova@tuke.sk,
³peter.butka@tuke.sk

Abstrakt – Príspevok sa zaoberá návrhom a implementáciou interaktívnej Rshiny aplikácie vizualizujúcej dáta zo satelitných meraní, ktoré slúžia odborníkom v oblasti skúmania atmosférických dát na tvorbu grafov znázorňujúcich zmeny koncentrácie prvkov v mezofére po výskyte meteorov. Hlavnou úlohou tejto aplikácie je podať interaktívne, ľahko zrozumiteľné grafy, ktoré pomôžu používateľom lepšie pochopiť a vidieť zmeny ozónu a oxidu dusnatého v časoch a miestach blízkyh k zaznamenaniu meteorov a meteorických rojov.

Kľúčové slová – satelitné dáta, meteory, vizualizácia dát, meranie atmosféry

I. ÚVOD

Každý deň preletí zemskou atmosférou veľké množstvo meteorov. Vedci sa domnievajú, že denne to môže byť až 44 000 kilogramov hmoty. Väčšina tohto materiálu sa však odparí v atmosfére Zeme a my tak môžeme vidieť jasnú stopu meteoru, ktorú hovorovo nazývame “padajúca hviezda”. Počas hociktorej jasnej noci v priebehu roka môžeme vidieť až niekoľko takýchto meteorov za hodinu. Niekoľkokrát do roka dokonca nastáva jav nazývaný meteorický roj. V tomto období môžeme vidieť desiatky padajúcich meteorov na nočnej oblohe každú hodinu. Výskumný tím na čele s Elizabeth A. Silber sa zaoberal teoretickým modelom, ktorý skúma zmeny koncentrácie ozónu a oxidu dusnatého v tesnej blízkosti za meteorom po jeho prelete atmosférou [1], [2]. Pre lepšie porozumenie teoretického modelu, ako aj jeho overenie z pohľadu pozorovaných dát, je zaujímavé mať k dispozícii dáta nameraných hodnôt koncentrácií týchto molekúl, a to podľa možnosti jednoduchým a prehľadným spôsobom.

V tejto práci predstavíme aplikáciu pre vizualizáciu dát zo satelitných pozorovaní. Vzhľadom k očakávaniam zo strany fyzikov sme navrhli základné požiadavky, ktoré má aplikácia spĺňať a to:

- výber hodnôt (O_3 , NO), ktoré chce používateľ sledovať
- výber časového obdobia a konkrétneho miesta
- výber nadmorskej výšky
- zabezpečiť interaktívnosť grafov a ukladanie výsledkov do súboru

Okrem toho sme definovali jednoduché scenáre použitia, ktoré sú reprezentované vizualizáciami popísanými v tomto článku.

II. POCHOPENIE DÁT

Dáta sú voľne dostupné a pochádzajú z prístroja SABER (The Sounding of the Atmosphere using Broadband Emission Radiometry) [3] umiestneného na satelite TIMED (Thermosphere Ionosphere Mesosphere Energetics Dynamics) patriaceho spoločnosti NASA. Tento prístroj umožňuje merať vertikálne profily teploty, hustoty a koncentrácie rôznych molekúl v atmosfére, medzi ktoré patria aj ozón a oxid dusnatý. Na serveri poskytujúcom dáta z prístroja SABER sa nachádzajú netCDF súbory za každý deň od roku 2002 po súčasnosť. V našich vizualizáciách sme pracovali s dátami verzie V2.0, Level 2A. Každý súbor obsahuje 55 premenných, 3 dimenzie a 16 globálnych atribútov. Nás však budú zaujímať len dimenzie *Altitude* a *Event*. *Event* závisí od počtu meraní v jednom orbite, približne je to ale 95 Event-ov v jednom súbore (za jeden obeh satelitu, čo je cca 90 minút). Z toho vyplýva, že dátová tabuľka z jedného netCDF súboru bude obsahovať:

- **Počet riadkov:** $EVENT \times ALTITUDE$ (pre každý event pripadá 400 meraní v rôznych výškach).
- **Atribúty:** event, dátum, čas, výška, latitude, longitude, hustota, O_3 , NO.

Rozhodli sme sa stiahnuť dáta pre 3 roky: 2012, 2013 a 2014. Toto obdobie sme zvolili z dôvodu možnosti porovnávať aspoň hodnoty v podobných časových obdobiach troch rôznych rokov, zároveň ale aj pre to aby sa v strede intervalu nachádzal rok 2013, kedy bol zaznamenaný prelet superbolidu (extrémne jasného meteoru) nad územím Ruska, neďaleko mesta Čeljabinsk v roku 2013. Roky 2012 a 2014 tak môžu poslúžiť fyzikom aj na porovnanie hodnôt po prelete tohto superbolidu v roku 2013 s hodnotami v predchádzajúcom a nasledujúcom roku. Dáta je možné neskôr rozšíriť a poskytnúť aj ďalšie roky.

III. VIZUALIZÁCIE – VERTIKÁLNE PROFILY

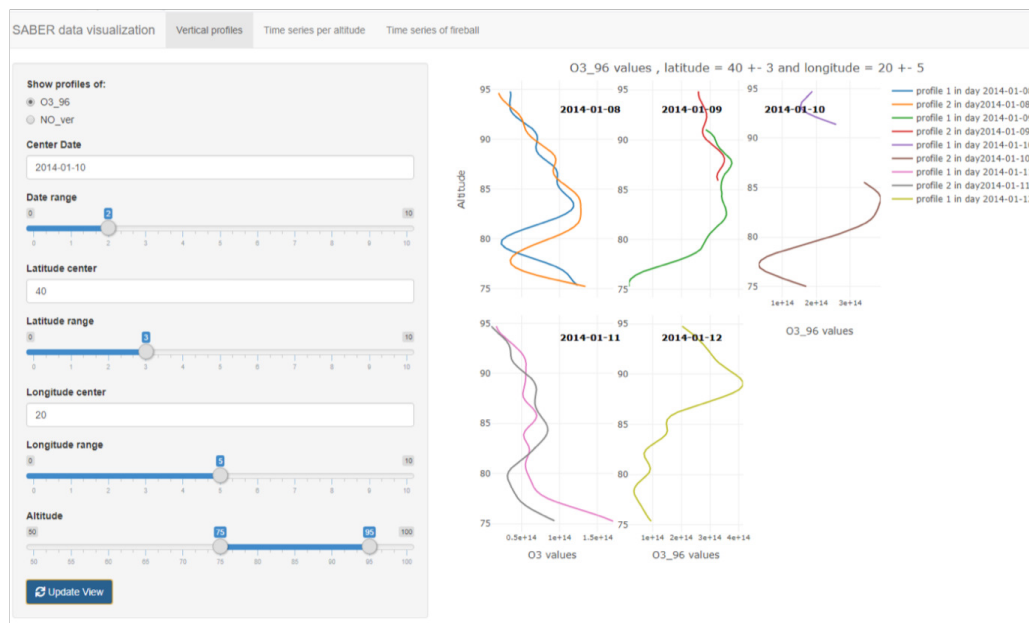
A. Vizualizácia - prvý scenár

Prvým scenárom použitia môže byť situácia, keď používateľ (fyzik) chce zistiť, či silný meteorický roj (napríklad roj Perzeíd, vyskytujúci sa zhruba jeden mesiac s maximom v okolí 12.augusta) ovplyvňuje hodnoty ozónu (O_3) alebo oxidu dusnatého (NO) v atmosfére v časovom okolí ním zadaného dňa v roku (napríklad +/-10 dní okolo 12.augusta) nad vybraným miestom (napríklad nad miestom v Európe). Voľbou viacerých miest môže fyzik sledovať vplyv pre rôzne zemepisné výšky a šírky. Dôležitá je najmä šírka, pretože ovplyvňuje počet meteorov pre daného pozorovateľa vzhľad k dráhe častíc vnikajúcich do atmosféry pre daný meteorický roj (každý ma na oblohe zdanlivý stred odkiaľ ako keby jeho meteory vylietavajú - je to spôsobené práve dráhou roja častíc pred vstupom do atmosféry - tento bod na oblohe sa nazýva radiant meteorického roja).

V rámci navrhutej vizualizácie, v prvom okne aplikácie sa zobrazujú vertikálne profily nameraných hodnôt ozónu a oxidu dusnatého v zadaných výškach. Používateľ si môže vybrať konkrétny dátum a počet dní v rozmedzí od 1 do 10 v okolí vybraného dátumu, pre ktoré chce vidieť grafy. Podobne si vyberie aj zemepisnú šírku a dĺžku, pri ktorých sa tiež nachádza interval rozmedzia. V poslednom rade si môže zvoliť rozsah nadmorských výšok, v ktorých chce sledovať namerané koncentrácie molekúl.

Výstup zobrazený na obrázku 1 je množina interaktívnych grafov, pričom každý graf znázorňuje vývin hodnôt zvoleného atribútu v jeden deň pre rôzne nadmorské výšky v určitých zemepisných súradniciach. Ak sa v jeden deň nameralo viac hodnôt (sonda preletela v medziach zadaných intervalov nad tým istým miestom na Zemi viackrát za deň), tak sa profily farebne rozlíšia.

Vytvorená vizualizácia takto umožňuje zvoliť si rôzne umiestnenia a časové obdobia, a to podľa potreby fyzika (v závislosti od toho pre aký meteorický roj chce vizualizácie vidieť), ktorý môže následne sledovať zmeny v koncentráciach najmä vzhľadom k maximu meteorického roja (kedy je počet meteorov maximálny).



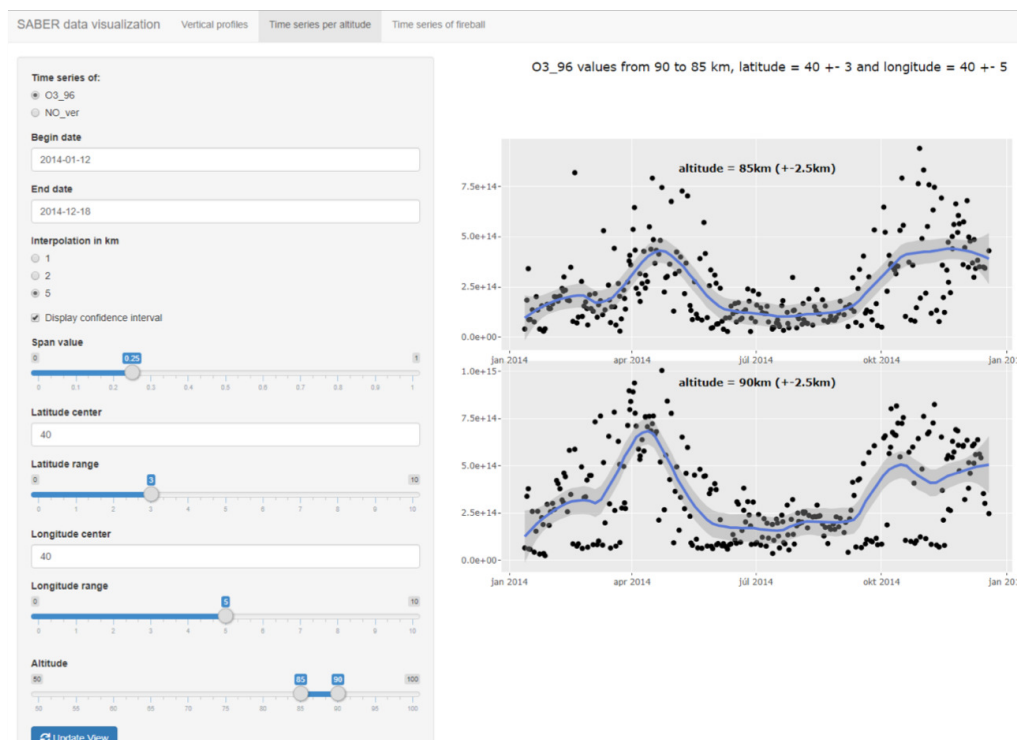
Obr. 1 Vizualizácia I - vertikálne profily vzhľadom k definovanému obdobiu

B. Vizualizácia - druhý scenár

Druhý scenár použitia aplikácie vychádza z potreby sledovať vývoj hodnôt koncentrácie vybranej molekuly pre určitú výšku počas dlhšieho obdobia, ideálne presahujúceho variácie spôsobené ročnými obdobiami a nimi súvisiacimi zmenami atmosféry. Znamená to, že fyzika si chce pozrieť napríklad vývoj (zmeny) koncentrácie ozónu od januára 2012 do decembra 2014 pre výšky od 70 do 95 km po každom piatom kilometri nad daným miestom (definovanom zemepisnou šírkou a zemepisnou výškou). Atmosférické výšky 70-95 km sú zvolené preto, že to je primárne miesto, kde dochádza k meteorickému javu a aj k teoreticky predpovedanému efektu zmeny koncentrácie molekúl v ich stopách.

V druhej časti aplikácie teda môže používateľ pozorovať zmenu hodnôt vybraného atribútu počas ním zvoleného časového obdobia v rôznych nadmorských výškach. Oproti predchádzajúcej časti si môže vybrať interpoláciu pre 1, 2 a 5 kilometrov, t.j., zobrazia grafy po každom jednom, dvoch alebo piatich kilometroch so spriemerovanými hodnotami meraní koncentrácií medzi nimi (v rámci pozorovaní prístroj SABER poskytuje na 1 km cca 3 merania). Výsledkom je graf vývoja koncentrácie v čase a používateľ si ešte môže zvoliť, či v grafoch chce mať viditeľný okrem bodov aj odhad krivky (pomocou metódy LOESS - LOcally Estimated Scatterplot Smoothing s nastaviteľnou hodnotou span v rozhraní) a jej interval spoľahlivosti (confidence interval).

Príkladom je teda výstup zobrazený na obrázku 2, predstavujúci (pre viaceré výšky) množinu interaktívnych grafov, pričom každý graf znázorňuje vývin hodnôt zvoleného atribútu nie v jeden deň, ale v rozsiahlom časovom období (2012-2014) pre jednu nadmorskú výšku v určitých zemepisných súradniciach.



Obr. 2 Vizualizácia II - graf vývoja hodnôt koncentrácie molekúl v čase pre dané výšky v atmosfére

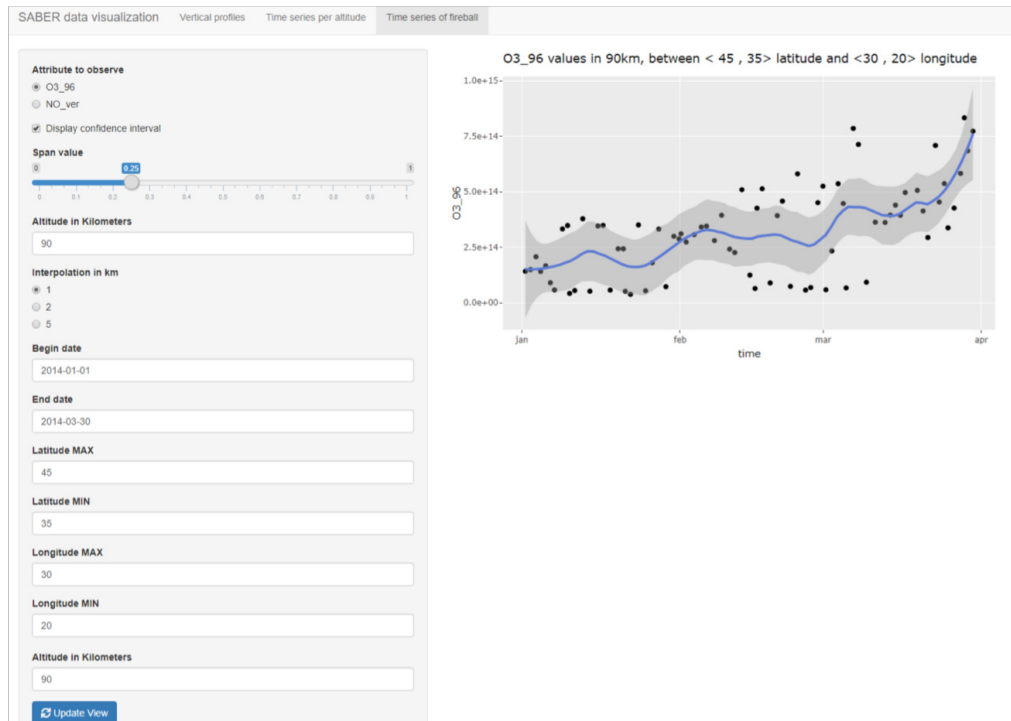
C. Vizualizácia - tretí scenár

Posledný scenár použitia predstavuje podobnú alternatívu k predchádzajúcemu, avšak cieľom je skúmať konkrétnu oblasť preletu jedného meteoru a sledovať jeho vplyv na zmenu koncentrácií molekúl v rôznych výškach nad miestom preletu. Tento scenár je užitočný najmä pre prípady preletu väčších telies, ideálnym príkladom je prípad superbolidu Čeljabinsk, ktorý sa objavil 15.2.2013 a spôsobil dopad meteoritických fragmentov v okolí ruského mesta rovnakého mena. V tomto prípade sa jednalo o najsilnejší bolid od čias dopadu fragmentov tzv. Tunguzského meteoritu v roku 1908.

V poslednej časti aplikácie preto predpokladáme, že používateľ vie konkrétnu výšku a súradnice preletu meteoru. Po zadaní týchto hodnôt sa mu vykreslí graf, v ktorom môže pozorovať vývin hodnôt O_3 a NO v časovom období pred preletom a po prelete meteoru. Na základe toho vie

sledovať či moment preletu extrémne jasného meteoru spôsobil zmeny v koncentráciách. Od predošlej časti sa líši hlavne v možnosti zadať väčší rozsah zemepisných súradníc podľa potreby, v závislosti od dráhy skúmaného meteoru.

Výstup zobrazený na obrázku 3 je jeden interaktívny graf, ktorý znázorňuje vývoj hodnôt vo zvolenom časovom období (2012-2014) pre jednu nadmorskú výšku v určitých zemepisných súradniciach. Tieto súradnice môžu byť zadané s väčším rozmedzím ako v predchádzajúcom rozhraní, aby sa dal pozorovať prelet meteoru, ktorého trajektóriu poznáme.



Obr. 3 Vizualizácia III - sledovanie zmien koncentrácie ozónu v čase v oblasti preletu bolidu pre danú výšku

Okrem možnosti vizualizácie v rozhraní Rshiny, pri všetkých scenároch je možnosť výsledné vizualizácie uložiť a poskytnúť vo forme pdf súborov.

IV. ZÁVER

Cieľom tejto práce bolo navrhnúť a implementovať interaktívnu Rshiny aplikáciu vizualizujúcu dáta zo satelitných meraní, ktorá bude slúžiť odborníkom v oblasti skúmania atmosférických dát na tvorbu prispôbitelných grafov znázorňujúcich zmeny hodnôt koncentrácií molekúl v mezofére po výskyte meteorov. Hlavnou úlohou tejto aplikácie je podať interaktívne, ľahko zrozumiteľné a rýchlo sa načítavajúce grafy, ktoré pomôžu používateľom lepšie pochopiť a vidieť zmeny koncentrácie ozónu a oxidu dusnatého v časoch a miestach blízky k zaznamenaniu meteorov a meteorických rojov. Aplikácia je ďalej ľahko rozšíriteľná, je možné použiť nielen dáta z dlhšieho časového obdobia, ale pri relatívne malých zmenách implementácie by sa dali doplniť aj ďalšie alebo upraviť existujúce scenáre, či sledovať zmeny v ďalších parametroch meraných prístrojmi SABER.

POĎAKOVANIE

Táto práca bola podporená APVV grantom – APVV-16-0213.

LITERATÚRA

- [1] E. A. Silber, M. L. Niculescu, P. Butka, and R. E. Silber, "Nitric oxide production by centimeter-sized meteoroids and the role of linear and nonlinear processes in the shock bound flow fields," *Atmosphere*, vol. 9, no. 5, p. 202, 2018.
- [2] E. A. Silber, W. K. Hocking, M. Niculescu, M. Gritsevich, and R. E. Silber, "On shock waves and the role of hyperthermal chemistry in the early diffusion of overdense meteor trains," *Monthly Notices of the Royal Astronomical Society*, vol. 469, pp. 1869–1882, 2017.
- [3] "Sounding of the atmosphere using broadband emission radiometry." [Online]. Available: http://saber.gats-inc.com/browse_data.php

Metódy hlbokého učenia pre klasifikáciu zákrytových premenných hviezd

¹Lukáš HUMENÍK, ²Viera MASLEJ KREŠŇÁKOVÁ, ³Peter BUTKA

^{1,2,3}Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹lukas.humenik@student.tuke.sk, ²viera.maslej.kresnakova@tuke.sk,
³peter.butka@tuke.sk

Abstrakt – Zákrytové premenné hviezdy vieme definovať ako svetelné krivky zaznamenávajúce svietivosť týchto systémov. Tento príspevok je zameraný na klasifikáciu zákrytových premenných hviezd pomocou metód hlbokého učenia. Využitím konvolučných neurónových sietí, ktoré boli trénované na základe syntetických dát vychádzajúcich z fyzikálnych modelov, sme dokázali klasifikovať observačné krivky podľa morfológie do troch tried a to oddelené, polodotykové a dotykové binárne hviezdy s celkovou úspešnosťou 92%.

Kľúčové slová – zákrytové premenné hviezdy, svetelné krivky, klasifikácia, hlboké učenie

I. ÚVOD

Neurónové siete majú svoje využitie v rôznych oblastiach. Jednou z nich je napríklad aj oblasť astrofyziky, ktorej sa venujeme v tomto príspevku. Zamerali sme sa na klasifikáciu zákrytových premenných hviezd. Je to špecifická skupina premenných hviezd, v ktorej dvojica hviezd tvoriaca binárny systém rotuje okolo spoločného bodu. Rotáciou následne dochádza k zákrytu jednej hviezdy druhou. Tento jav spôsobí pokles intenzity svetla vyžarovaného binárnym systémom.

Svetelné krivky predstavujú základnú formu dát, s ktorou budeme pracovať. Metódy hlbokého učenia [1] používajú architektúru neurónovej siete. Aplikovaním práve neurónových sietí sa pokúsime dosiahnuť čo najvyššiu úspešnosť klasifikácie. Konvolučné neurónové siete sa najčastejšie využívajú na klasifikáciu 2-dimenzionálnych údajov. V praxi sa však stretávame aj s úlohami, kde je lepšie použiť 1-dimenzionálnu konvolučnú neurónovú sieť (1D CNN) [2]. Typickými úlohami pre použitie 1D CNN je napríklad rozpoznávanie reči, spracovanie textu či spracovanie časových radov.

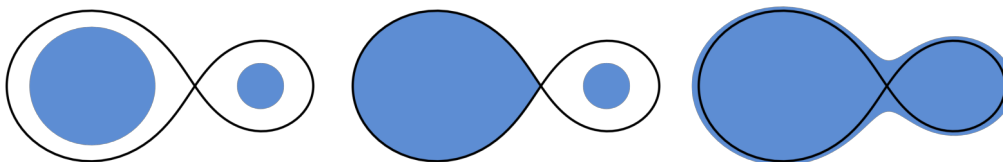
V tomto príspevku sme sa zaoberali práve využitím 1D CNN na klasifikáciu svetelných kriviek zákrytových binárnych systémov. Keďže svetelná krivka je reprezentovaná ako vektor hodnôt svietivosti a má rovnaký počet fáz, tak 1D CNN je vyhovujúcim nástrojom na dosiahnutie požadovaných výsledkov. Cieľom tejto práce je automatizovať proces klasifikácie zákrytových binárnych hviezd.

II. ZÁKRYTOVÉ PREMENNÉ HVIEZDY

Zákrytové dvojité hviezdy, inak nazývané aj zákrytové binárne hviezdy, sú len jedným z viacerých druhov premenných hviezd. Pre pozorovateľa sa tieto hviezdy môžu javiť ako jediný bod svetla, avšak podľa jeho premenlivosti jas a spektroskopických pozorovaní môžeme povedať, že jeden bod svetla v skutočnosti predstavujú dve hviezdy, ktoré obiehajú okolo seba. Zmeny intenzity svetla zákrytových binárnych hviezd sú spôsobené tým, že jedna hviezda prechádza pred druhou vo vzťahu k pozorovateľovi. Ak predpokladáme, že hviezdy sú sférické a majú kruhové obežné dráhy, potom dokážeme ľahko odhadnúť, ako sa mení svetlo v závislosti od času pre zatmenie dvojitých hviezd. Tieto výpočty sa môžu vykonávať relatívne jednoduchým počítačovým programom [3].

Klasifikácia zákrytových premenných hviezd, ktorej sa venujeme v tejto práci, je založená na pomere medzi vzdialenosťou hviezd a ich veľkosťou. Na základe tohto kritéria rozdelujeme dvojité hviezdy do troch tried:

- **Oddelené binárne hviezdy** Oddelené dvojité hviezdy (angl. detached), sú znázornené ako prvé na obrázku 1 a predstavujú takú dvojicu hviezd, v ktorej sa každá hviezda nachádza vo svojom vlastnom Rocheovom laloku. Rocheovým lalokom nazývame oblasť okolo binárnej hviezdy, v ktorej sa nachádza obiehajúci materiál gravitačne viazaný na danú hviezdu. Táto oblasť má tvar kvapky, ktorej vrchol smeruje k druhej hviezde. V Rocheovej oblasti je gravitačná sila pôvodnej hviezdy väčšia v porovnaní s druhou hviezdou. Hviezdy na seba nijako významne nevlývajú a vyvíjajú sa samostatne. Väčšina binárnych hviezd patrí práve do tejto triedy.
- **Polodotykové binárne hviezdy**
Polodotykové binárne hviezdy (angl. semi-detached) sú charakteristické tým, že jedna z hviezd zaplňuje celý svoj Rocheov lalok (darca), pričom druhá hviezda (príjemca) nie. Plyny z povrchu darcovského Rocheovho laloka sa prenášajú na druhú, narastajúcu hviezdu. Pri tomto procese dominuje hromadný prenos. Vo viacerých prípadoch sa stáva, že plyn pritekajúci k prijímacej hviezde vytvára okolo nej disk. Na obrázku 1 je takýto binárny systém znázornený v strede.
- **Dotykové binárne hviezdy**
Dotykové binárne hviezdy (angl. over-contact) sú také, v ktorých obe hviezdy tvoriace binárnu sústavu hviezd, vyplnia svoje Rocheove laloky. Obiehajúci materiál v Rocheových lalokoch oboch hviezd sa spojí a vytvorí nad celou binárnou sústavou obálku. Trením obiehajúceho materiálu v obálke sa brzdí orbitálny pohyb, čo môže viesť k tomu, že sa hviezdy nakoniec zlúčia. Na obrázku 1 ich znázorňuje 3 animácia.



Obr. 1 Oddelené, polodotykové a dotykové binárne hviezdy

III. KLASIFIKÁCIA SVETELNÝCH KRIVIEK ZÁKRYTOVÝCH BINÁRNYCH HVIEZD

Naša práca je vysoko špecifická tým, že ide o experimenty na dátach, ktoré generovala a zozbierala výskumná skupina z UPJŠ. Tieto datasety neboli doposiaľ publikované. Ako jediná publikáciu, ktorá riešila podobný problém považujeme [4].

V publikácii [4] je predstavený systém, ktorý automaticky klasifikuje svetelné krivky zákrytových binárnych hviezd. Cieľom klasifikácie je roztriediť zákrytové binárne hviezdy podľa ich geometrickej konfigurácie v upravenej verzii zaužívanej klasifikačnej schémy. Na klasifikáciu bol v tejto práci použitý bayesovský súbor neurónových sietí natrénovaných na dátach, ktoré boli zozbierané satelitom *Hipparcos*. Tie sú tvorené siedmimi rôznymi kategóriami vrátane excentrických binárnych systémov a dvoch typov morfológií pulzujúcich svetelných kriviek. Táto schéma vychádza z historickej klasifikácie zákrytových binárnych kriviek do troch skupín Algol, Beta Lyrae a W Ursae Majoris, ktoré predstavujú triedy oddelených, polodotykových a dotykových systémov. Percentuálne vyjadrenie chybovosti pre ich 3 najlepšie architektúry sú $8,7\% \pm 3,6\%$, $6,9\% \pm 1,3\%$ a $6,9\% \pm 1,3\%$.

V našej práci sme zvolili metodiku CRISP-DM [5], ktorá nás prevedie celým priebehom riešenia úlohy v našom podaní.

A. Pochopenie cieľa

Zákrytový binárny systém, ktorému sa venujeme v tejto práci, je jedným z typov binárnych hviezd. Obežná rovina tohto systému dvoch hviezd leží približne na úrovni zorného poľa pozorovateľa tak, že komponenty tvoriace systém podliehajú vzájomnému zatmeniu.

Zákrytové binárne systémy patria medzi premenlivé hviezdy, avšak, dôvodom nie je to, že sa svetlo jednotlivých hviezd líši, ale to, že dochádza k zatmeniu medzi hviezdami systému. Svetelná krivka zákrytovej binárnej hviezdy je tvorená z periód prakticky konštantného svetla s

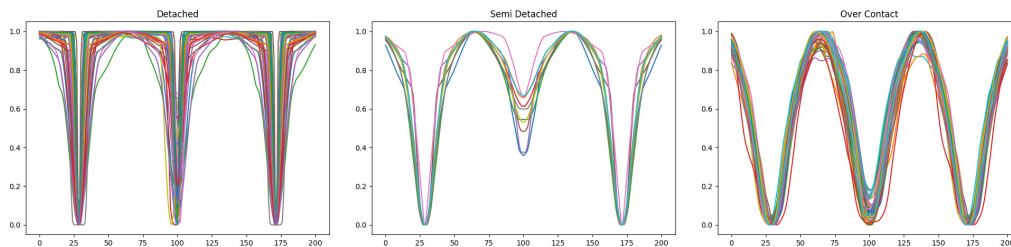
periodickými poklesmi intenzity svetla. Keď je jedna z dvojice hviezd väčšia ako druhá, bude jedno zatmenie úplné, zatiaľ čo druhé bude predstavovať prstencové zatmenie.

V našej práci sa zaoberáme svetelnými krivkami, ktoré predstavujú jeden z najdôležitejších atribútov, ktoré dokážeme získať pri študovaní binárnych hviezd. Svetelné krivky sa zobrazujú vo forme grafu, ktorého y-ová os predstavuje jas objektu, a na x-ovej osi sa nachádza určité časové obdobie. Pri štúdiu predmetov, ktoré majú premenlivú svietivosť v čase, ako sú napríklad novy, supernovy alebo premenné hviezdy, je svetelná krivka jednoduchým, ale za to hodnotným ukazovateľom.

B. Príprava dát

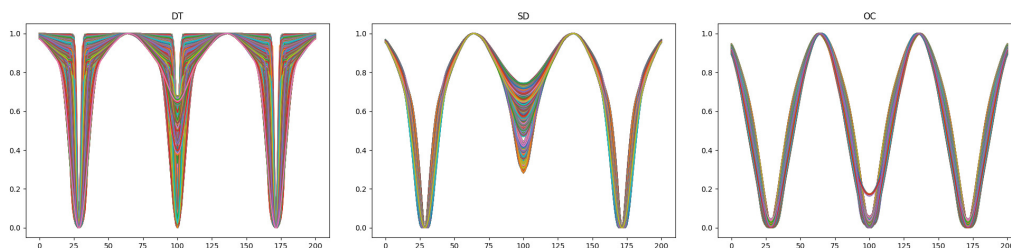
Dáta, s ktorými budeme pracovať, nám poskytla výskumná skupina z Univerzity Pavla Jozefa Šafárika v Košiciach. Ide o syntetické dáta, teda také, ktoré boli vygenerované na základe parametrov binárnych hviezd z reálnych observačných pozorovaní. Najpočetnejšou skupinou sú svetelné krivky oddelených binárnych systémov. Svetelných kriviek triedy oddelených systémov máme k dispozícii 107 780, triedy polodotykových binárnych systémov je 23 900 a najmenej početnou triedou je trieda dotykových binárnych hviezd, ktorých je 15 754.

Naša výskumná skupina sa taktiež podarilo zozbierať a zatriediť dáta z reálnych pozorovaní. Ich počet je však veľmi malý. Celkovo máme k dispozícii iba 100 takýchto svetelných kriviek. 41 z nich patrí triede oddelených binárnych hviezd, 12 je z triedy polodotykových a ostatných 47 svetelných kriviek patrí do triedy dotykových binárnych systémov. Tieto krivky sú zobrazené na obrázku 2.



Obr. 2 Observačné dáta

Na základe observačných dát bez anomálií sme vyseletovali podmnožinu syntetických kriviek pre každú triedu. Výber sme uskutočnili na základe priebehu svietivosti v každej z troch tried osobitne. Do filtrovanej množiny syntetických dát sme zahrnuli iba tie svetelné krivky, ktorých hodnoty sa nachádzali v určenom rozpätí svietivosti pre danú fázu. Tento postup sme opakovali pre všetky tri triedy. Po selekcii sa v triede oddelených systémov nachádza 4497 kriviek, polodotykových je 1809 a dotykových svetelných kriviek je 9115. Tieto krivky sú zobrazené na obrázku 3.



Obr. 3 Filtrované syntetické dáta

C. Modelovanie

V našej práci sme sa rozhodli použiť konvolučné neurónové siete. Zvyčajne sa používajú pri úlohách zameraných na klasifikáciu obrázkov a na prácu s obrázkami všeobecne. Avšak, v našom prípade použijeme konvolučnú sieť, ktorej filter nebude dvojrozmerný ako pri obrázkoch, ale bude obsahovať iba jednu dimenziu. Týmto spôsobom chceme extrahovať príznaky zo svetelných kriviek a na ich základe naučiť model správne klasifikovať binárne systémy.

Architektúru modelu, ktorý klasifikoval binárne systémy s požadovanou presnosťou môžete vidieť na obrázku 4. Použili sme dve konvolučné bloky, ktoré obsahujú CNN vrstvu a vzorkovaciu vrstvu. Vyššiu úspešnosť sme dosiahli aj vďaka regularizačnej vrstve. Rozhodli sme sa použiť regularizačnú techniku dropout s parametrom 0,25, čo znamená, že 25% vstupných jednotiek sa pri každej aktualizácii v rámci tréningu modelu odstránilo, čím sme zabránili preučeniu a prispeli k lepšej extrakcii príznakov.

Layer (type)	Output Shape	Param #
conv1d_26 (Conv1D)	(None, 182, 64)	1344
max_pooling1d_26 (MaxPooling)	(None, 91, 64)	0
conv1d_27 (Conv1D)	(None, 82, 32)	20512
max_pooling1d_27 (MaxPooling)	(None, 41, 32)	0
flatten_15 (Flatten)	(None, 1312)	0
dense_29 (Dense)	(None, 64)	84032
dropout_9 (Dropout)	(None, 64)	0
dense_30 (Dense)	(None, 3)	195
Total params: 106,083		
Trainable params: 106,083		
Non-trainable params: 0		

Obr. 4 Sumárna tabuľka modelu

D. Vyhodnotenie

Výsledky modelu, ktorý popisujeme vyššie nám prinieslo výsledky zobrazené v tabuľke 1. Dosiahli sme najvyššiu úspešnosť 92%. Náš model vyhodnotil nesprávne presne 8 príkladov. Najväčšiu nepresnosť sme zaznamenali v triede polodotykových svetelných kriviek, čo vyplýva z veľkej podobnosti kriviek v hraničných intervaloch.

Skutočnosť ↓, Predikcia →	dotykový	polodotykový	oddelený
dotykový	44	3	0
polodotykový	1	8	3
oddelený	0	1	40
Presnosť	0.9778	0.6667	0.9302
Návratnosť	0.9362	0.6667	0.9756
F1 skóre	0.9565	0.6667	0.9524

Tabuľka 1
 Kontingenčná tabuľka a vyhodnotenie metrick modelu

IV. ZÁVER

Použitím neurónových sietí sme sa pokúsili vyriešiť úlohu zameranú na klasifikáciu zákrytových premenných hviezd, alebo inak nazývaných aj binárnych systémov. Náš model bol natrénovaný na tri triedy binárnych systémov na základe dát obsahujúcich svetelné krivky.

Vytvorili sme niekoľko modelov, ktoré sme porovnávali medzi sebou vyhodnotením na observačných dátach, ktoré predstavovali svetelné krivky binárnych systémov z reálnych pozorovaní. Výsledky boli vo všetkých prípadoch veľmi podobné. Najvyššia úspešnosť, akú sa nám podarilo dosiahnuť bola na úrovni 92%.

POĎAKOVANIE

Táto práca bola podporená APVV grantom – APVV-16-0213.

LITERATÚRA

- [1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [2] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applications: A survey," *arXiv preprint arXiv:1905.03554*, 2019.
- [3] D. Bruton, 1995. [Online]. Available: <http://www.physics.sfasu.edu/astro/ebstar/ebstar.html>
- [4] L. M. Sarro, C. Sánchez-Fernández, and Á. Giménez, "Automatic classification of eclipsing binaries light curves using neural networks," *Astronomy & Astrophysics*, vol. 446, no. 1, pp. 395–402, 2006.
- [5] J. Paralič, *Objavovanie znalostí v databázach.*, 1st ed. Košice: Elfa, spol. s r.o., 2003.

Metódy hlbokého učenia pre detekciu a klasifikáciu antisociálneho správania v online priestore

¹Zuzana DÁVIDOVÁ, ²Viera MASLEJ KREŠŇÁKOVÁ, ³Martin SARNOVSKÝ

^{1,2,3}Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹zuzana.davidova@student.tuke.sk, ²viera.maslej.kresnakova@tuke.sk,
³martin.sarnovsky@tuke.sk

Abstrakt – Prevažná väčšina obsahu na Internete je používateľmi generovaný obsah. Tento obsah predstavujú rôzne články, recenzie, internetové diskusie a fóra, príspevky na sociálnych sieťach a podobne. Polarita tohto obsahu je však rôzna, pretože aj používatelia - ľudia, sú rôzni. Niektorí sú zdvorilí, korektní a konajú zodpovedne aj v online svete. Existujú však aj takí, ktorí sa nesprávajú slušne, vyjadrujú sa vulgárne, šíria nenávistné prejavy a ich správanie tak môžeme označiť ako antisociálne. Predložený článok sa venuje použitiu metód hlbokého učenia pre detekciu antisociálneho správania, konkrétne toxických komentárov v online diskusiách. V rámci riešenia úlohy sme vytvorili niekoľko rôznych prístupov, kde sme Porovnávali napr. použitie predspracovaného a nepredspracovaného textu, ale taktiež aj použitie rôznych typov neurónových sietí. Modely boli trénované a vyhodnotené na dátovej množine Toxic Comments získanom zo súťaže Kaggle.

Kľúčové slová – antisociálne správanie, toxické komentáre, hlboké učenie, neurónové siete

I. ÚVOD

Internet je v súčasnosti prostredie, v ktorom môžu používatelia vytvárať a zdieľať informácie s minimálnymi obmedzeniami. Väčšina z nich používa web zodpovedným a efektívnym spôsobom, existuje však skupina, ktorá preukazuje typ správania, ktorý by sa dal opísať ako antisociálny. V súčasnosti existuje množstvo definícií antisociálneho správania, existujú však dva hlavné typy takéhoto správania:

- **Šírenie dezinformácií, falošných správ** – v rôznych formách, napr. hoaxy, falošné a neobjektívne správy, falošné recenzie
- **Reakcie používateľov** – vyskytujú sa v rôznych formách, napr. manipulácia s diskusiou, kyberšikana, nenávistné výroky, trolling, spamovanie a iné.

Obe formy antisociálneho správania predstavujú vážny problém, pretože ich dôsledky môžu mať veľký dopad na skutočný život. Používatelia internetu spolu často komunikujú v reálnom čase, diskusie zvyčajne zahŕňajú veľké množstvo používateľov. Takáto masívna komunikácia podporovaná modernými technológiami, ktoré umožňujú čiastočnú anonymitu, vedie tiež k novým hrozbám vo forme nesprávnych reakcií používateľov, napr. osobné útoky. Antisociálne reakcie používateľov v online diskusiách často súvisia s toxickými poznámkami. Toxický komentár v online diskusií všeobecne možno definovať ako reakciu agresívnym spôsobom (napr. osobné útoky, slovné šikanovanie).

Existuje potreba účinne odhaľovať a odstraňovať rôzne formy antisociálneho správania. V súčasnosti sa stále používajú manuálne techniky odhaľovania antisociálneho správania v online komunitách. V praxi to znamená, že za nájdenie a odhalenie takéhoto správania je zodpovedný človek, moderátor diskusie. Tento spôsob je neefektívny, preto existuje veľká potreba navrhnuť a implementovať nové metódy využívajúce strojové učenie a umelú inteligenciu. Cieľom takýchto prístupov je využitie takýchto metód na prevenciu a elimináciu negatívnych vplyvov antisociálneho správania v online komunitách, napríklad umožnením plne automatizovaného zisťovania a predpovedania rôznych typov antisociálneho správania.

II. METÓDY HLBOKÉHO UČENIA NA DETEKCIU ANTISOCIÁLNEHO SPRÁVANIA

Detekcia antisociálnych komentárov sa môže realizovať ako klasifikačná úloha, ktorej cieľom je detegovať, či je vstupný text obsahuje antisociálne prvky správania alebo nie. Ďalšou možnosťou je presnejšie zistiť o aký typ antisociálneho správania ide. Pri detekcii sa využíva binárna klasifikácia, zatiaľ čo pri predpovedaní typu antisociálneho správania v komentári sa musia použiť techniky viacnásobnej (angl. *multilabel*) klasifikácie, kde jeden komentár môže spadať do viac ako jednej kategórie antisociálneho správania. V súčasnosti klasifikácia toxických komentárov predstavuje aktívnu oblasť výskumu a navrhuje sa množstvo prístupov s použitím rôznych metód.

Niekoľko publikácií sa taktiež zameriava na odhaľovanie antisociálneho správania v online komunitách [1]. Autori vo svojej štúdií [2] o kyberšikane použili komentáre na YouTube, ktoré sa týkali citlivých tém rasy a kultúry, sexuality a inteligencie.

V štúdií [3] sa autori zaoberali detekciou toxických komentárov pomocou metód hlbokého učenia. Výsledná klasifikácia určovala či je daný komentár toxický alebo nie je. V tejto štúdií autori použili dve rôzne slovníkové metódy - GloVe a FastText. Pri každej z týchto metód použili iný typ neurónovej siete, pričom sa zamerali predovšetkým na dva typy - konvolučné a rekurentné neurónové siete. V prípade rekurentnej LSTM siete, použili aj jej obojsmerný variant. Modely vyhodnocovali použitím štandardných metrick - návratnosť, presnosť a F1 skóre. Spomedzi všetkých experimentov dosiahli rekurentné neurónové siete najvyššie hodnoty návratnosti. Použitím GloVe dosiahla jednoduchá a obojsmerná LSTM sieť hodnotu návratnosti 74%. Úspešnosť tréningu dosiahla 86% použitím konvolučnej a obojsmernej LSTM vrstvy (FastText). Poslednou hodnotiacou metriku je F1 skóre, pri ktorej dosiahli najvyššiu hodnotu 77,7%, a to pri použití LSTM siete (GloVe) a obojsmernej LSTM siete (GloVe).

Autori článku [4] pri riešení problému detekcie toxických komentárov použili logistický regresný klasifikátor a úlohu riešili ako viacnásobnú (angl. *multilabel*) klasifikáciu. Výsledný model teda zaradil daný komentár do jednej zo šiestich kategórií (toxic, obscene, insult,...). Ich cieľom bolo zlepšiť výsledky vtedajšej state-of-the-art, ktoré pri použití slovníkového prístupu FastText dosiahli priemerné hodnoty úspešnosti (angl. *Accuracy*) pre každú triedu 94% a priemerné hodnoty meriky AUC (angl. *Area Under Curve*) pre každú triedu 82%. Autori použili vytvorené používateľské komentáre z e-learning Udemy kurzov, s cieľom porovnať výkonnosť prístupov využívajúcich slovné embeddingy v rozličných doménach. Využili pritom rôzne slovníkové prístupy, pričom najlepšie výsledky dosiahli pri použití FastText. Priemerná hodnota úspešnosti klasifikácie pre každú triedu sa rovnala 96% a priemerná hodnota metriky AUC pre každú triedu dosiahla 88%.

III. POCHOPENIE DÁT

Pracovali sme s dátovou množinou obsahujúcou komentáre zo stránky Wikipédia – internetová encyklopédia s otvoreným obsahom. Táto dátová množina bola zverejnená v rámci Kaggle súťaže – *Toxic Comment Classification Challenge*.

Súbor dát bol rozdelený na tréningové – 159 571 záznamov a testovacie dáta – 153 164 záznamov, ktoré obsahovali nasledovné atribúty: *id*, *comment_text*, *toxic*, *severe_toxic*, *obscene*, *threat*, *insult* a *identity_hate*, ktoré predstavujú cieľové atribúty (angl. *labels*).

Tabuľka 1 poskytuje prehľad početností hodnôt jednotlivých tried nie len samostatne, ale takisto aj v kombinácii s triedou *toxic*, ktorá je najviac zaskúpená v súbore údajov. Z tabuľky je zrejmé, že ak je komentár označený ako veľmi toxický, automaticky sa označí ako toxický. Zaujímavá je tretia časť tabuľky, ktorá zdôrazňuje, že nie každý obscénny, urážlivý a nenávisťný komentár alebo vyhrážka musí byť automaticky klasifikovaný ako toxický.

Percentuálny podiel komentárov v rámci jednotlivých kategórií zobrazuje Obrázok 1.

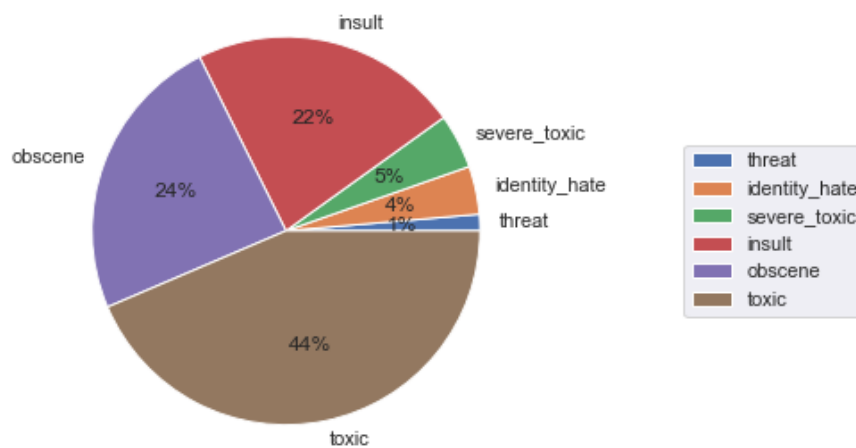
Keďže podstatou viacnásobnej klasifikácie je to, že príklad, v našom prípade komentár, môže patriť do viacerých tried, na Obrázku 2 sme znázornili do akého počtu tried patrí aký počet komentárov. Z obrázka teda vyplýva, že najviac komentárov patrí do jednej triedy a najmenej komentárov patrí do šiestich tried zároveň.

IV. PRÍPRAVA DÁT

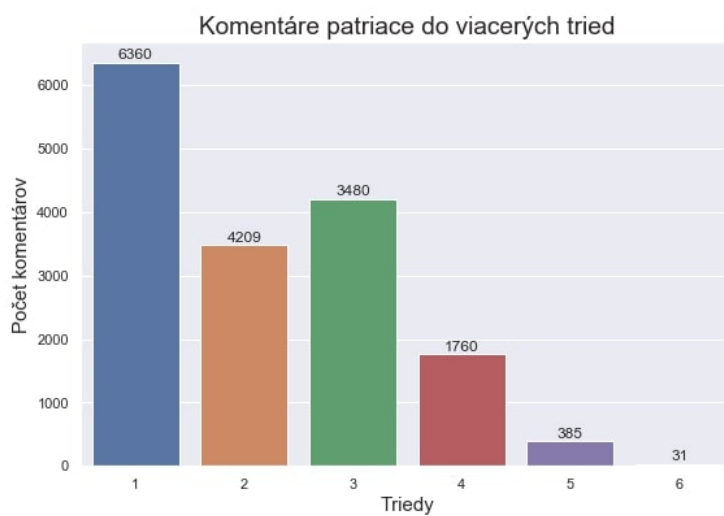
Vo fáze prípravy dát je zvyčajne hlavnou úlohou dáta upraviť – predspracovať. Keďže pracujeme s textom, často sa stáva, že v ňom nájdeme určité skratky, slangové výrazy, neúplné slová, preklepy a podobne, ktoré by sme za bežných okolností z textu odstránili. V tomto prípade by sme ale chceli poukázať na štúdiu [5]. Autori sa v tejto štúdií venujú otázke, či pri doméne akou je antisociálne správanie na Internete a jeho detekcia, je dôležité a dobré dáta predspracovať, resp. či nie je lepšie ponechať ich v pôvodnej podobe. Textové dáta týkajúce sa domény antisociálneho správania totiž úpravou strácajú svoju osobitosť a špecifické črty, ktoré sú príznačné či už pre samotný obsah správy, ktorú chce autor podať, ale aj pre samotného autora. Ak by sme to zhrnuli

Trieda	Početnosť
toxic	15 294
severe toxic	1 595
obscene	8 449
threat	478
insult	7 877
identity hate	1 405
severe toxic AND toxic	1 595
obscene AND toxic	7 926
threat AND toxic	449
insult AND toxic	7 344
identity hate AND toxic	1 302
severe toxic but NOT toxic	0
obscene but NOT toxic	523
threat but NOT toxic	29
insult but NOT toxic	533
identity hate but NOT toxic	103

Tabuľka 1
Početnosti v rámci jednotlivých tried toxických komentárov



Obr. 1 Zobrazenie percentuálneho podielu komentárov v jednotlivých kategóriách prostredníctvom koláčového grafu



Obr. 2 Znáznornenie počtu komentárov, ktoré patria do viacerých tried

– predspracovaním textu v tejto doméne môžeme prísť o dôležité črty, ktoré majú rozhodujúci charakter v procese extrahovania príznakov. Rozhodli sme sa preto do našich experimentov zaradiť aj porovnanie modelov z hľadiska toho, či v nich použijeme predspracovanie alebo nie.

Pri predspracovaní dát sme sa držali niekoľkých dôležitých krokov. Ako prvé bolo potrebné odstrániť záznamy s *null* hodnotami, ktoré pre nás neboli nijako užitočné a v ďalších krokoch by nám spôsobovali iba problémy. Ďalším krokom bolo pripravenie textového korpusu, ktorý sme vytvorili z pôvodných dát aplikovaním nasledovných krokov:

- vytvorenie slovných tokenov,
- prevod textu na malé písmená,
- odstránenie interpunkcie,
- odstránenie stop slov¹.

Slová, ktoré nám ostali v dátovej množine po spracovaní textu do textového korpusu sme následne previedli do vektorovej reprezentácie. Vznikli nám tak vektorové reprezentácie slov (angl. *word embeddings*).

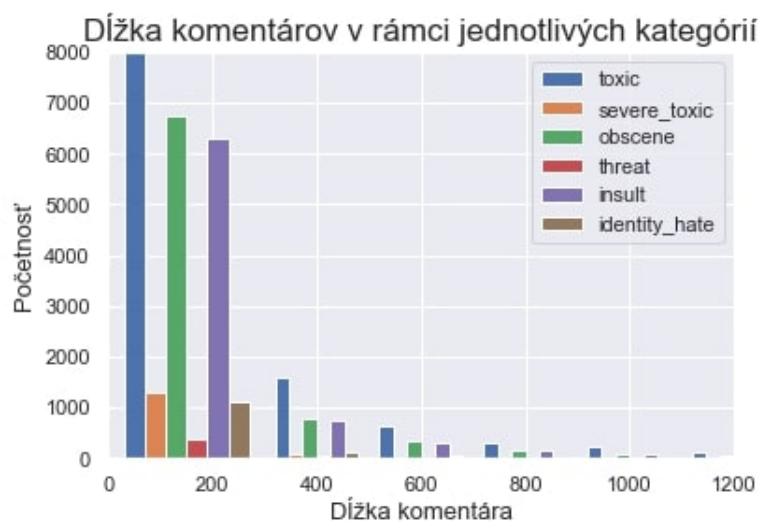
Word embeddings sú vektorovou reprezentáciou slov v prirodzenom jazyku. Keďže zdroj ich významu vyplýva z kontextu, v akom sa nachádzajú v texte, nazývame ich aj *kontextové vektory*. Na základe polohy slov, teda vzdialenosti vo vektorovom priestore, je možné extrahovať syntaktickú a sémantickú podobnosť daných slov. Slová s podobnými vlastnosťami sa tak nachádzajú blízko seba.

V tejto práci sme na vytvorenie slovných vektorov použili nástroj Word2Vec, ktorý je v jazyku Python implementovaný prostredníctvom knižnice Gensim. Takto pripravené vektorové reprezentácie slov sme previedli na tokenizovaný vektor, teda sekvenciu indexov jednotlivých slov v texte. Sekvencie sme zarovnali na rovnakú dĺžku doplnením nulových indexov, tzv. padding. Vo finálnej fáze sme vytvorili maticu s vektormi, ktorú sme použili v neurónovej sieti ako jeden z parametrov Embedding vrstvy.

Implementovali sme aj ďalšie slovníkové prístupy – GloVe a FastText, pri ktorých sme už ale použili vopred vytvorené/predtrénované slovné vektory:

- *GloVe* patrí medzi najnovšie spomedzi metód pre výpočet vektorovej reprezentácie slov. Tento prístup však neprechádza celým korpusom. Učí sa len na základe globálnej štatistiky výskytov slov v spoločnom kontexte. Táto metóda zachytáva rôzne lingvistické vzory a dokáže úspešne riešiť úlohy založené na princípe analógie [6].
- *FastText* je knižnica vytvorená výskumným tímom spoločnosti Facebook pre učenie a výpočet reprezentácie slov a klasifikáciu viet. Jej princíp spočíva v priradení vektorovej reprezentácie k n-gramom znakov, ktoré obsahujú jednotlivé slová [7].

Priemerná dĺžka komentára vo všeobecnosti bola 394 znakov. Obrázok 3 zobrazuje počet komentárov v závislosti od dĺžky komentára v rámci jednotlivých tried. My sme v našich experimentoch pracovali s priemernou dĺžkou komentára na úrovni 200 znakov.

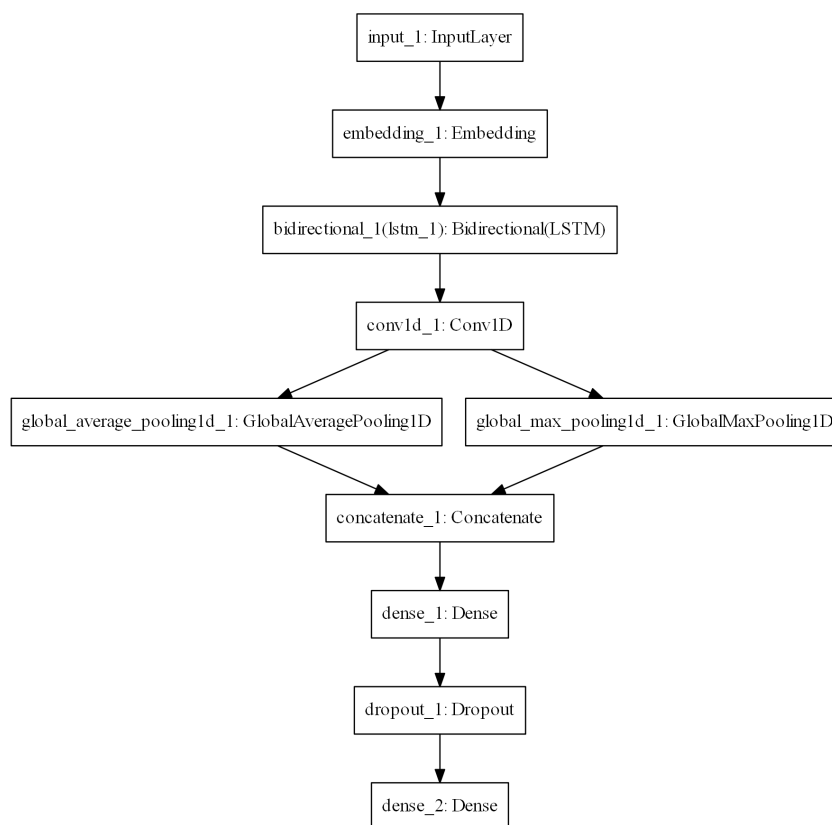


Obr. 3 Zobrazenie priemerných hodnôt dĺžky komentára v závislosti od jednotlivých tried antisociálneho správania

¹Stop-slová (angl. *Stop words*) sú neplnovýznamové slová vyskytujúce sa v danom jazyku. Sú to zvyčajne spojky, predložky, zámená a pod.

V. MODELING AND EVALUATION

Pre detekciu typov antisociálneho správania sme použili viacnásobnú klasifikáciu. Výsledkom bolo určenie konkrétnych tried, ktorých črty obsahuje daný komentár. Po vyskúšaní viacerých architektúr neurónových sietí sa nám ako najvhodnejšia osvedčila architektúra zobrazená na Obrázku 4. Súčasťou tejto architektúry je Embedding vrstva, obojsmerná LSTM vrstva, 1D konvolučná vrstva a vzorkovacie vrstvy. Nasledujú 2 plne prepojené vrstvy, pričom druhá z nich predstavuje výstupnú vrstvu neurónovej siete, na ktorej sa nachádza 6 neurónov – jeden pre každý typ antisociálneho správania.



Obr. 4 Architektúra použitá pri viacnásobnej (angl. *multilabel*) klasifikácii toxických komentárov

Vytvorili sme niekoľko klasifikačných modelov. V experimentoch sme testovali použitie rôznych slovníkových prístupov, ako aj použitie predspracovaného a nepredspracovaného textového korpusu. Modely sme vyhodnocovali použitím štandardných metrick *úspešnosť a chyba* na testovacích dátach, *presnosť, návratnosť a F1 miera*.

Keďže z hľadiska detekcie antisociálneho správania je nevyhnutné minimalizovať hodnotu FN (False negative) pre nás kľúčovou metrika sa stala návratnosť, pri ktorej sme sa pokúšali dosiahnuť čo najvyššie hodnoty. Užitočnou v tomto prípade je aj metrika F1 skóre, ktorá predstavuje istý kompromis medzi presnosťou a návratnosťou. Okrem hodnotenia modelov skrz štandardné metriky, sme taktiež merali efektívnosť tréningu. Pozornosť sme venovali času, ktorý bol potrebný na natréningovanie modelu.

Tabuľka 2 prezentuje najlepšie dosiahnuté výsledky v rámci daného slovníkového prístupu. Prvý z dvojice je vždy model, v ktorom sme použili textový korpus **bez predspracovania**.

	Úspešnosť	Chyba	Presnosť	Návratnosť	F1 miera
GloVe1	0.9831	0.0443	0.8	0.72	0.76
e2GloV e2	0.9822	0.0474	0.79	0.71	0.75
Word2vec1	0.9827	0.0519	0.82	0.68	0.74
Word2vec2	0.9819	0.0493	0.82	0.65	0.73
FastText1	0.9833	0.0454	0.82	0.70	0.76
FastText2	0.9822	0.0523	0.83	0.65	0.73

Tabuľka 2

Zhrnutie najlepších výsledkov neurónových sietí pri detekcii antisociálneho správania s využitím viacnásobnej (angl. *multilabel*) klasifikácie. Slovníkové prístupy označené číslom 1, napr. *GloVe1*, predstavujú modely bez predspracovania. Čas tréningu bol pri všetkých modeloch rovnaký, bolo to 4167s.

Najvyššie miery návratnosti sa nám podarilo dosiahnuť pri použití slovníkového prístupu GloVe, o čosi lepšie však v prípade GloVe1, kde sme textový korpus nepredspracovali. Takisto hodnoty metrik F1 miera a chyba na testovacej množine dosiahli pri tomto modeli najlepšie výsledky.

Čo sa týka úspešnosti, skóre bolo pomerne vyrovnané pri všetkých modeloch. Hodnoty sa odlišovali iba v tisícinách. Možeme konštatovať, že predspracovanie vo všetkých prípadoch prišlo horšie výsledky. Tu sa potvrdzujú tvrdenia uvedené v úvode tejto kapitoly, že predspracovaním sme v tomto prípade prišli o potrebné príznaky. Okrem toho sa vynechaním predspracovania skrátil čas a výpočtová náročnosť experimentov.

Výsledok viacnásobnej klasifikácie v podobe jednotlivých metrik pre každú triedu znázorňuje Tabuľka 3. K tomu prislúchajúce kontingenčné tabuľky zobrazuje Tabuľka 4.

Trieda	Presnosť	Návratnosť	F1 miera	Počet
toxic	0.86	0.75	0.80	3056
severe toxic	0.51	0.34	0.41	321
obscene	0.86	0.77	0.81	1715
threat	0.70	0.35	0.47	74
insult	0.77	0.69	0.73	1614
identity hate	0.60	0.39	0.47	294

Tabuľka 3
Výsledné hodnoty metrik pre všetky triedy

toxic	2289	362
	767	28497
severe toxic	108	102
	213	31492
obscene	1328	218
	387	29982
threat	26	11
	48	31830
insult	1110	329
	504	29972
identity hate	115	76
	179	31545

Tabuľka 4
Kontingenčná tabuľka s výslednými hodnotami pre jednotlivé triedy antisociálneho správania

VI. ZÁVER

Chceli by sme zdôrazniť výsledky, ktoré sme dosiahli s modelmi, pri ktorých sme nepoužili štandardné kroky spojené s bežným spracovaním údajov. Modely trénované bez použitia bežného spracovania boli v priemere lepšie ako modely s predspracovaným textom. Potvrdili sme tvrdenia zo štúdie [5], že v oblasti ako je detekcia antisociálneho správania, nie je vhodné údaje predspracovať. Predspracované údaje tak strácajú svoju individualitu a špecifickosť a modely tak dosahujú horšie výsledky. Zároveň považujeme za výhodu implementáciu a porovnanie rôznych slovníkových metód.

POĎAKOVANIE

Táto práca bola podporená APVV grantom – APVV-17-0267.

LITERATÚRA

- [1] J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec, "Antisocial behavior in online discussion communities," 2015.
- [2] K. Dinakar, R. Reichart, and H. Lieberman, "Modeling the detection of textual cyberbullying," in *fifth international AAAI conference on weblogs and social media*, 2011.
- [3] B. V. Aken, J. Risch, R. Krestel, and A. Löser, "Challenges for toxic comment classification: An in-depth error analysis," 2018.
- [4] R. Saia, A. Corrigan, R. Mulas, D. Reforgiato Recupero, and S. Carta, "A supervised multi-class multi-label word embeddings approach for toxic comment classification," 09 2019.
- [5] F. Mohammad, "Is preprocessing of text really worth your time for online comment classification?" *ArXiv*, 2018.
- [6] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
- [7] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," 2016.

Výpočet parametrov týkajúcich sa dynamickej ampacity vonkajších vedení

¹Nikola ŠTIEBEROVÁ, ²Lubomír BEŇA, ³Anastázia MARGITOVÁ,
⁴Martin KANÁLIK

^{1, 2, 3, 4} Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹nikola.stieberova@student.tuke.sk, ²lubomir.bena@tuke.sk,
³anastazia.margitova@tuke.sk, ⁴martin.kanalik@tuke.sk

Abstract — Populačný rast, industrializácia a rozvoj priemyslu spôsobil neustále narastajúci dopyt po elektrickej energii. Pretože výstavba nových elektrických vedení je veľmi časovo a finančne náročná, pristupuje sa k zvyšovaniu prenosovej schopnosti už existujúcich elektrických vedení, a to zvyšovaním ampacity vodičov vonkajších elektrických vedení. Tento článok sa zaoberá výpočtom ustálenej teploty vodiča pri známej hodnote prúdu pretekajúceho vodičom. V praktickej časti je realizované porovnanie výpočtu teploty vodiča v závislosti od času analytickým riešením a numerickým teplotným krokováním podľa technickej príručky CIGRE 601.

Keywords — teplota vodiča, technická príručka CIGRE 601, skoková zmena

I. ÚVOD

Vodiče vonkajších elektrických vedení sú neoddeliteľnou súčasťou elektrizačnej sústavy. Používajú sa pri vysokých napätiach, a preto ich parametre musia spĺňať určité limity, aby sa zaistila bezpečná prevádzka. Jeden z najdôležitejších faktorov ovplyvňujúcich prenosovú schopnosť vedení je teplota vodičov. V prípade, ak teplo generované prúdom tečúcim vodičom prekročí určitý tepelný limit, dôjde k nezvratnému poškodeniu vodiča. Aby sa zabránilo poškodeniu vodičov elektrického vedenia, je potrebné určiť maximálnu hodnotu prúdu, ktorý môže pretekať vodičom. Ampacita je hlavným parametrom pri návrhu a prevádzke vonkajších elektrických vedení. Inými slovami, je to maximálna hodnota elektrického prúdu, ktorý môže pretekať cez vodič (elektrické vedenie), bez narušenia jeho mechanických a elektrických vlastností [1].

II. TEPelnÉ HODNOTENIE PRENOSOVEJ SCHOPNOSTI VEDENÍ

A. Statické tepelné hodnotenie (STH)

Jedná sa o tradičný spôsob prevádzkovania vedení. V niektorých prenosových sústavách sa používajú rôzne limitné obmedzenia ampacity pre letné a zimné mesiace. Tieto stanovené prúdové obmedzenia v jednotlivých ročných obdobiach predstavujú omnoho nižšie hodnoty ako tie hodnoty prúdu, ktoré môžu pretekať vedením za aktuálnych poveternostných podmienok. STH uvažuje s maximálnou povolenou prenosovou schopnosťou odvodenou od najhorších možných parametrov počasia [1].

STH prenosových vedení môže byť obmedzované tromi hlavnými faktormi [2]:

- stanovený dovolený prúd tečúci vodičom,
- stanovená vzdialenosť vodiča od zeme,
- menovitý výkon transformátora.

B. Dynamické tepelné hodnotenie (DTH)

Poskytuje skutočnú ampacitu prenosových vedení pri uvažovaní prevádzkových podmienok v reálnom čase, ktorá je daná časovou konštantou a rozdielom medzi okamžitou a maximálnou hodnotou teploty vodiča. Hlavným cieľom DTH je zvýšiť prenosovú schopnosť existujúcich vedení, zmierniť preťaženie vedení, uľahčiť integráciu veternej energie, znížiť ekonomické

náklady a zlepšiť spoľahlivosť prevádzky elektrizačnej sústavy [1].

Parametre, ktoré môžu byť merané v reálnom čase, sú [3], [4]:

- poveternostné podmienky, ako napr. okolitá teplota, rýchlosť vetra, smer vetra, slnečné žiarenie a zrážky,

- charakteristiky vedení, ako napr. zaťaženie vedenia, vzdialenosť vodičov od zeme, priehyb vodičov, mechanické namáhanie vodiča a teplota vodiča.

Metódy DTH vonkajších vedení sú rozdelené do dvoch skupín [3], [5]-[8]:

- priame metódy – zariadenia na priame monitorovanie vonkajších vedení zbierajú údaje o charakteristických veličinách vedenia, ako je priehyb vodiča, mechanické namáhanie vodiča, výška vodiča nad zemou a teplota vodiča. Na určovanie DTH vedení sa používajú aj ďalšie údaje z meteorologických staníc.

- nepriame metódy – odhadnuté alebo merané údaje o počasí pozdĺž vedenia sa využívajú na výpočet prúdovej zaťažiteľnosti (ampacity) tohto vedenia. Monitorovanie počasia je najjednoduchší a najmenej invazívny systém monitorovania, pretože na vedení nemusí byť nainštalovaný žiaden monitorovací systém.

III. VYUŽITIE WAMS SYSTÉMOV

Širokoplošné monitorovacie systémy (WAMS) umožňujú realizáciu nových a pokročilých funkcií na podporu rozvoja a riadenia prenosových a distribučných sietí, konkrétne na hodnotenie stability prevádzky elektrizačnej sústavy, ampacity vedení a podobne [9], [10].

Online monitorovanie parametrov vonkajších vedení je pokročilou funkciou, ktorá je založená na predpoklade, že prevádzkové parametre nie sú statické hodnoty, ale sú ovplyvnené zaťažením siete, poveternostnými podmienkami a starnutím vodičov. Pri online výpočtoch prevádzkových parametrov vedenia sa musí brať do úvahy aj vplyv chyby meracích prístrojov prúdu a napätia [10].

Výhody využitia WAMS systémov pri hodnotení ampacity vonkajších elektrických vedení sú [10], [11]:

- vyšší povolený limit zaťaženia vedenia používaný na kontingenčnú analýzu, ktorá poukazuje na rezervy pri využívaní elektrických vedení,
- využitie možnosti vyššej prúdovej zaťažiteľnosti vedení v zime,
- úspora investícií na výstavbu nových vedení,
- využitie úplnej prenosovej zaťažiteľnosti vedení pri riešení kritických a núdzových situácií

IV. VÝPOČET TEPLoty VODIČOV VONKAJŠÍCH ELEKTRICKÝCH VEDENÍ VYPLÝVAJÚCI Z DYNAMICKEJ AMPACITY PODĽA TECHNICKEJ PRÍRUČKY CIGRE 601

A. Tepelná bilančná rovnica pre ustálený stav

Teplota vodičov vonkajších elektrických vedení závisí od prevládajúcich poveternostných podmienok, ako je rýchlosť a smer vetra, teplota okolia alebo slnečné žiarenie. Taktiež aj od elektrického prúdu, ktorý vodičom preteká. Predpokladá sa, že všetky tieto parametre sú konštantné v závislosti od času. V tom prípade je možné hovoriť o tom, že vodič je v ustálenom stave. Inými slovami, tepelný prírastok spôsobujúci oteplenie vodiča je rovný stratám tepla, ktoré sú zodpovedné za ochladenie vodiča, viď (1) [1], [3], [12], [13]:

$$\text{Tepelný prírastok} = \text{Tepelné straty}, \quad (1)$$

$$P_J + P_s + P_m = P_c + P_r, \quad (2)$$

kde

- P_J – oteplenie vodiča Jouleovými stratami (W/m),
- P_s – oteplenie vodiča slnečným žiarením (W/m),
- P_m – oteplenie vodiča magnetickým žiarením (W/m),
- P_c – ochladenie vodiča konvekciou (W/m),
- P_r – ochladenie vodiča radiáciou (W/m).

Postup výpočtu všetkých vyššie uvedených tepelných výkonov je uvedený v [1].

B. Tepelná bilančná rovnica pre prechodný stav vodiča

Tepelná bilančná rovnica pre prechodný stav vodiča berie do úvahy teplo akumulované vo vodiči. Zmenu teploty vodiča vonkajších elektrických vedení v závislosti od času je potom možné vyjadriť nasledovným spôsobom [14], [15]:

$$\text{Teplo akumulované vo vodiči} = \text{Tepelný prírastok} - \text{Tepelné straty} \quad (3)$$

$$m \cdot c \cdot dT_s/dt = P_J + P_m + P_s - P_c - P_r, \quad (4)$$

kde

- m – hmotnosť vodiča na jednotku dĺžky (kg/m),
 c – merná tepelná kapacita vodiča (J/(kg·K)).

Pre súčin hmotnosti na jednotku dĺžky a mernej tepelnej kapacity hliníkových vodičov vystužených ocelou, t. j. AlFe lán zložených z dvoch materiálov, v závislosti od teploty vodiča T_s platí [13]:

$$m \cdot c = m_a \cdot c_a + m_s \cdot c_s, \quad (5)$$

$$c_a = c_{a20} \cdot [1 + \beta_a \cdot (T_s - 20)], \quad (6)$$

$$c_s = c_{s20} \cdot [1 + \beta_s \cdot (T_s - 20)], \quad (7)$$

kde

- m_a – hmotnosť hliníka (Al) na jednotku dĺžky (kg/m),
 c_a – merná tepelná kapacita hliníka (Al) pri teplote T_s (J/(kg·K)),
 m_s – hmotnosť ocele (Fe) na jednotku dĺžky (kg/m),
 c_s – merná tepelná kapacita ocele (Fe) pri teplote T_s (J/(kg·K)),
 c_{a20} – merná tepelná kapacita hliníka (Al) pri teplote 20 °C (J/(kg·K)),
 β_a – teplotný súčiniteľ mernej tepelnej kapacity hliníka (Al) (1/K),
 c_{s20} – merná tepelná kapacita ocele (Fe) pri teplote 20 °C (J/(kg·K)),
 β_s – teplotný súčiniteľ mernej tepelnej kapacity ocele (Fe) (1/K).

Oteplenie a ochladenie vodiča, respektíve teplota vodiča závisí od oteplenia vodiča Jouleovými stratami, magnetickým žiarením a slnečným žiarením, a od ochladenia vodiča konvekciou a radiáciou. Tieto tepelné výkony sú závislé od klimatických podmienok (teplota okolia, rýchlosť a smer vetra, intenzita slnečného žiarenia) a od prúdu tečúceho vodičom. V Tabuľke I je uvedená závislosť medzi zmenou klimatických podmienok a oteplením/ochladením vodiča. Najjednoduchší spôsob, ako modelovať zmeny prevádzkových parametrov vodiča, je skoková zmena.

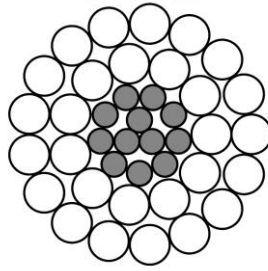
Tabuľka I Závislosť medzi zmenou klimatických podmienok a oteplením/ochladením vodiča

Prevádzkový parameter		ΔP_J	ΔP_s	ΔP_c	ΔP_r
Prúd	Zvýšenie	+			
	Zníženie	-			
Slnečné žiarenie	Zvýšenie		+		
	Zníženie		-		
Rýchlosť vetra	Zvýšenie			+	
	Zníženie			-	
Teplota okolia	Zvýšenie			-	-
	Zníženie			+	+

V. IMPLEMENTÁCIA ZVOLENÉHO DYNAMICKÉHO TEPelnÉHO MODELU NA VODIČE VONKAJŠÍCH ELEKTRICKÝCH VEDENÍ PRI REÁLNYCH PREVÁDZKOVÝCH PODMIENKACH

Ako skúmaný fázový vodič je zvolené lano 352-AL1/59-ST1A (350 AlFe 6, resp. AlFe 350/59), ktoré je bežne používané na Slovensku a jeho základné parametre sú uvedené v Tabuľke II. Konštrukčné vyhotovenie lana 352-AL1/59-ST1A je znázornené na Obr. 1.

Nakoľko má lano 352-AL1/59-ST1A dve hliníkové vrstvy, môže byť vplyv magnetického oteplenia podľa technickej príručky CIGRE 601 zanedbaný. Toto lano je špecifické tým, že má 3 osovú vodiče (zvyčajne je to 1 osový vodič). Sivé vodiče na Obr. 1 predstavujú oceľové jadro a biele hliníkový plášť.



Obr. 1 Konštrukčné prevedenie lana 352-AL1/59-ST1A (350 AlFe 6)

Tabuľka II Technická špecifikácia lana 352-AL1/59-ST1A (350 AlFe 6)

Parameter	Hodnota
Vonkajší priemer	$D = 26,5 \cdot 10^{-3} \text{ m}$
Koeficient absorptivity povrchu	$\alpha_s = 0,35 \text{ (-)}$
Koeficient emisivity povrchu	$\varepsilon = 0,35 \text{ (-)}$
Striedavý odpor pri teplote 20 °C	$R_{ac 20} = 0,0816 \cdot 10^{-3} \text{ ohm/m}$
Teplotný súčiniteľ odporu	$\alpha = 4,03 \cdot 10^{-3} \text{ 1/K}$
Merná tepelná kapacita ocele (Fe) pri teplote 20 °C	$c_{s 20} = 481 \text{ J/(kg} \cdot \text{K)}$
Merná tepelná kapacita hliníka (Al) pri teplote 20 °C	$c_{a 20} = 897 \text{ J/(kg} \cdot \text{K)}$
Teplotný koeficient mernej tepelnej kapacity ocele (Fe)	$\beta_s = 1 \cdot 10^{-4} \text{ 1/K}$
Teplotný koeficient mernej tepelnej kapacity hliníka (Al)	$\beta_a = 3,8 \cdot 10^{-4} \text{ 1/K}$
Hmotnosť ocele (Fe) na jednotku dĺžky	$m_s = 0,5119 \text{ kg/m}$
Hmotnosť hliníka (Al) na jednotku dĺžky	$m_a = 1,116 \text{ kg/m}$
Výška nad hladinou mora	$y = 208 \text{ m}$
Počet hliníkových drôtov (Al) vo vrstvách	11 + 17
Počet oceľových drôtov (Fe) vo vrstvách	3 + 9
Prierez hliníkových drôtov (Al)	351,86 mm ²
Prierez oceľových drôtov (Fe)	58,9 mm ²
Priemer hliníkových drôtov (Al)	4 mm
Priemer oceľových drôtov (Fe)	2,5 mm

A. Výpočet teploty reálneho vodiča vonkajších elektrických vedení podľa technickej príručky CIGRE 601 po zmene prevádzkových podmienok

Táto časť sa zaoberá výpočtom teploty lana 352-AL1/59-ST1A, ktorého parametre sú uvedené v Tabuľke II, a to dvoma metódami prezentovanými v technickej príručke CIGRE 601: analytickým riešením a numerickým teplotným krokovaním. V analytickom riešení do výpočtu vstupuje teplota vodiča pred zmenou, ale aj asymptotická teplota vodiča po skokovej zmene. Asymptotická teplota je maximálna teplota, ku ktorej sa blíži teplota vodiča, ale podľa analytického výpočtu nikdy túto teplotu nedosiahne (asymptota). Súčin hmotnosti vodiča na jednotku dĺžky m a mernej tepelnej kapacity vodiča c sa v prípade analytického riešenia počíta iba raz, pričom sa vychádza z teploty vodiča pred zmenou. Pri numerickom teplotnom krokovaní sa súčin $m \cdot c$ prepočítava v každom kroku (iterácii). Podrobná metodika výpočtu je uvedená v [16].

V Tabuľke III sú uvedené prevádzkové podmienky a tepelná špecifikácia skúmaného vodiča pred a po zmene všetkých prevádzkových parametrov. Ak sa zmenia všetky prevádzkové podmienky, budú sa na zmene teploty vodiča podieľať všetky tepelné výkony z bilančnej rovnice (4). Táto súčasná zmena všetkých parametrov vyvolá ochladenie vodiča v dôsledku najvýraznejšieho vplyvu, t. j. konvekcie, ktorá súvisí hlavne so zmenou rýchlosti vetra. Oteplenie vodiča Jouleovými stratami a slnečným žiarením nie je dostatočné, rovnako ako pokles miery ochladenia vodiča radiáciou.

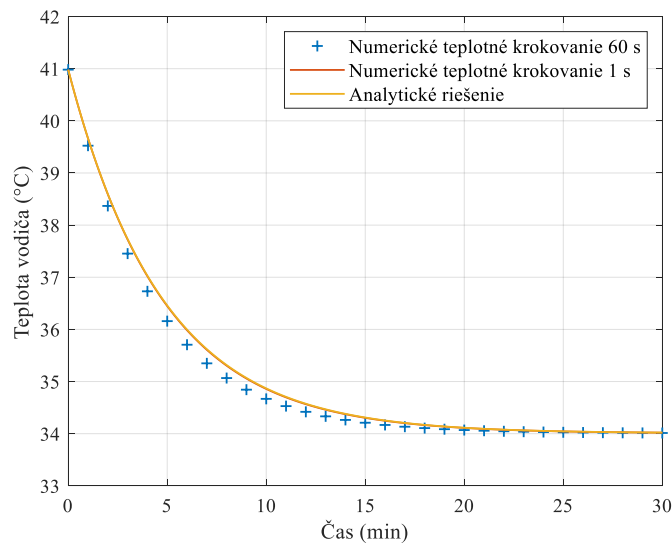
Záporné znamienka pri časových konštantách reprezentujú, že zmena daného prevádzkového parametra nekorešponduje s výslednou zmenou teploty vodiča. Tepelná časová konštanta vyjadruje časový interval, za ktorý sa zmení teplota vodiča z počiatočnej (východiskovej) teploty na cca 63,2 % z asymptotickej teploty. V Tabuľke III je vidieť, že zvýšenie hodnoty oteplenia Jouleovými stratami a slnečným žiarením by malo viesť k otepleniu vodiča, no aj napriek tomu sa vodič ochladí. Časová konštanta prúdu a slnečného žiarenia je z toho dôvodu záporná. Taktiež zníženie miery ochladenia radiáciou (z dôvodu zvýšenia teploty okolia) by malo mať za následok oteplenie vodiča, ale záporné znamienko pri časovej konstante radiácie vyjadruje, že dochádza k opačnému javu (ochladeniu vodiča).

Na Obr. 2 je zobrazená závislosť teploty vodiča od času po súčasnej skokovej zmene všetkých prevádzkových parametrov uvedených v Tabuľke III. Z Obr. 2 vyplýva to, že analytické riešenie

je takmer totožné s numerickým teplotným krokovaním s krokom 1 s. Výpočet numerickým teplotným krokovaním s krokom 60 s má podobne väčší odklon od analytického riešenia. Z uvedeného vyplýva, že aj numerické teplotné krokovanie s vhodne zvoleným krokom prináša dostatočne presné výsledky.

Tabuľka III Prevádzkové podmienky a tepelná špecifikácia vodiča pred a po súčasnej skokovej zmene všetkých prevádzkových parametrov

Parameter	Pred zmenou	Po zmene
Teplota okolia (°C)	21	24
Rýchlosť vetra (m/s)	0	5
Smer vetra (°)	45	45
Intenzita slnečného žiarenia (W/m ²)	0	400
Prúd (A)	400	700
Ustálená/asymptotická teplota vodiča (°C)	40,9822	34,0071
Oteplenie vodiča Jouleovými stratami (W/m)	14,1518	43,3398
Ochladenie vodiča konvekciou (W/m)	10,4383	74,3786
Ochladenie vodiča radiáciou (W/m)	3,7135	3,2019
Oteplenie vodiča slnečným žiarením (W/m)	0	3,7100
Časová konštanta prúdu (min)		-5,0015
Časová konštanta konvekcie (min)		2,2831
Časová konštanta radiácie (min)		-285,3764
Časová konštanta slnečného žiarenia (min)		-39,3489
Globálna časová konštanta (min)		4,7816



Obr. 2 Závislosť teploty vodiča od času po súčasnej skokovej zmene všetkých prevádzkových parametrov

VI. ZÁVER

Cieľom tohto článku je čitateľovi priblížiť problematiku výpočtu teploty vodičov vonkajších elektrických vedení súvisiaci s dynamickou ampacitou podľa metodiky uvedenej v technickej príručke CIGRE 601. Hlavným zámerom je poukázať na skutočnosť, že jednotlivé prevádzkové parametre (klimatické podmienky a prúd tečúci vodičom) majú vplyv na teplotu vodiča a menia sa v čase. Konkrétne sa skúma vplyv prevádzkových parametrov na teplotu vodiča po ich skokovej zmene, t. j. závislosť teploty vodiča od času. Výpočet sa realizuje dvoma metódami, analytickou a numerickou. V prípade skokovej zmeny teploty okolia, intenzity slnečného žiarenia a prúdu platí medzi týmito parametrami a teplotou vodiča priama úmernosť. Medzi teplotou vodiča a zmenou rýchlosti vetra platí nepriama úmernosť: čím je vyššia rýchlosť vetra, tým nižšia je teplota vodiča. V prípade súčasnej zmeny všetkých prevádzkových parametrov sa podieľajú na zmene teploty vodiča všetky tepelné výkony, ktoré prispievajú k otepleniu/ochladeniu vodiča. Výsledné oteplenie/ochladenie vodiča je závislé od najvýraznejšieho vplyvu. V prípade rozoberanom v tomto článku je to vplyv konvekcie, ktorá súvisí hlavne so zmenou rýchlosti vetra.

POĎAKOVANIE

Prácu podporila Vedecká grantová agentúra Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky a Slovenskej akadémie vied grantom VEGA č. 1/0372/18 „Výskum pôsobenia SMART sietí na distribučnú sústavu“.

LITERATÚRA

- [1] M. Kanálik, A. Margitová, L. Beňa, „Temperature calculation of overhead power line conductors based on CIGRE Technical Brochure 601 in Slovakia.“ In *Journal of Electrical Engineering 101(16)*, 2019, s. 921 – 933
- [2] S.D. Kim, M.M. Morcos, „An application of dynamic thermal line rating control system to up-rate the ampacity of overhead transmission lines.“ In *IEEE Transactions on Power Delivery 28(2)*, 2013, s. 1231 – 1232
- [3] S. Karimi, P. Musilek, A.M. Knight, „Dynamic thermal rating of transmission lines: A review.“ In *Renewable and Sustainable Energy Reviews 91*, 2018, s. 600 – 612
- [4] Y. Wang, Yanling et al., „Uncertainty analysis of dynamic thermal rating based on environmental parameter estimation.“ In *EURASIP Journal on Wireless Communications and Networking, 2018*
- [5] M. Ntulí et al., „Increasing the capacity of the transmission lines via uprating: an updated review of benefits, considerations and developments.“ In *Australasian Universities Power Engineering Conference (AUPEC)*, 2016
- [6] R. Adapa, D.A. Douglass, „Dynamic Thermal Ratings: Monitors and Calculation Methods.“ In *IEEE Power Engineering Society Inaugural Conference and Exposition in Africa*, 2005, s. 163 – 167
- [7] D.A. Douglass et al., „A Review of Dynamic Thermal Line Rating Methods with Forecasting.“ In *IEEE Transactions on Power Delivery 34(6)*, 2019, s. 2100 – 2109
- [8] E. Fernandez et al., „Review of dynamic line rating systems for wind power integration.“ In *Renewable and Sustainable Energy Reviews 53*, 2016, s. 80 – 92
- [9] CIGRE, Working Group B2.36, „Guide for Application of Direct Real-Time Monitoring Systems,“ Technical Brochure 498, 2012
- [10] A. Popelka et al., „ADVANCED APPLICATIONS OF WAMS.“ In *22nd International Conference on Electricity Distribution*, 2013
- [11] V. Krištof, S. Kušnír, M. Kolcun, „Základy technológie WAMS.“ In *Electric Power Engineering 2011 : proceedings of the 12th international scientific conference*, 2011, ISBN 978-80-248-2393-5
- [12] B. Pouckovic, Z. Djuricic, „Current carrying capacity of overhead line that connects wind power plant to the grid.“ In *10th International Conference on Environment and Electrical Engineering*, 2011
- [13] CIGRE, Working Group B2.43. „Guide for thermal rating calculation of overhead lines,“ Technical Brochure 601, 2014
- [14] J. Hosek et al., „Effect of time resolution of meteorological inputs on dynamic thermal rating calculations.“ In *IET Generation, Transmission & Distribution*, 2011
- [15] V.T. Morgan, „Rating of bare overhead conductors for intermittent and cyclic currents.“ In *Proceedings of the Institution of Electrical Engineers 116(8)*, 1969, s. 1361 – 1376
- [16] Štieberová Nikola, „Výpočet dynamickej ampacity vonkajších vedení,“ Diplomová práca, 2020, s. 27 – 33

Návrh Konceptie Nabíjacej a Vybíjacej Časti Testera Li-ion Batériových Článkov

¹Juraj BILANSKÝ, ²Milan LACKO

¹Katedra Elektrotechniky a Mechatroniky, Fakulta Elektrotechniky a Informatiky, Technická Univerzita v Košiciach, Slovenská Republika

¹juraj.bilansky@tuke.sk, ²milan.lacko@tuke.sk

Abstrakt – Tento článok obsahuje informácie, ohľadne li-ion batérií, ich základných vlastností. Popisuje vhodné metódy nabíjania a vybíjania li-ion batériových článkov. Takisto obsahuje základné bezpečnostné podmienky, ktoré je počas týchto procesov potrebné dodržať. Výsledkom článku je návrh konceptie pre cyklický tester batérií, na ktorého zostrojenie sa bude ďalej pracovať.

Kľúčové slová – batéria, článok, nabíjanie, vybíjanie, koncepcia

I. ÚVOD

Batérie sa v posledných rokoch stali neoddeliteľnou súčasťou života každého z nás. Poskytujú napájanie pre mobilné zariadenie, ktoré dennodenne používame. Sú to napríklad mobilné telefóny, notebook, hodinky a veľa iného. S rozvojom mobilných zariadení sa rozrástá aj trh s dostupnými batériami. Dnes je na trhu dostupných množstvo rôznych typov batérií ako sú napríklad NiCd (nickel–cadmium), NiMh (Nickel–metal hydride) alebo Li-ion (lithium-ion). Práve posledné menované li-ion batérie tvoria na trhu majoritný podiel. Používajú sa v mobiloch, notebookoch a v mnohých ďalších odvetviach. Tieto batérie sa používajú aj vo vozidlách na elektrickú energiu. Budeme sa venovať práve týmto batériám, ktoré sa nachádzajú v elektrických vozidlách (elektromobily, elektrokolobežky, elektrobicykle, ...). Tieto batérie sú zvyčajne veľké, ťažké a tvoria značnú časť hmotnosti vozidla. Sú umiestnené v podlahe a tým pádom znižujú ťažisko celého vozidla, čo má oproti veľkej váhe pozitívny účinok na jazdné vlastnosti. Takéto batérie sa nazývajú trakčné (energia z nich sa premieňa na pohybovú). Tieto batérie (podľa veľkosti vozidla) sa skladajú z veľkého počtu malých batériových článkov, ktoré vhodným sériovo-paralelným spájaním vytvárajú batériu elektrického vozidla s požadovanými vlastnosťami ako sú výstupné napätie, prúd a kapacita. Výrobcovia pri návrhu batérie musia nájsť správny pomer medzi kapacitou batérie a hmotnosťou, keďže platí čím väčšia batéria tým je vozidlo ťažšie a takisto je aj spotreba energie väčšia. Výsledná kvalita batérie potom definuje základné vlastnosti vozidla ako sú výkon, dojazd a hmotnosť. Z pohľadu cyklického testovania batérií je tento proces energeticky, časovo ale aj finančne náročný (celá batéria je drahá, v prípade elektromobilu aj ťažká a má vysokú kapacitu) preto si testovanie zjednodušíme a budeme testovať základné stavebné prvky týchto batérií a to batériové články.



Obr. 1 Batéria elektromobilu Tesla Model S - 85 kWh [1]

II. LI-ION BATÉRIE

Li-ion batérie dokážu poskytnúť skvelú úroveň výkonu počas dlhej životnosti. Pre dosiahnutie maximálnej životnosti je potrebné ich správne nabíjať a vybíjať. Bežne dostupné nabíjačky sú zväčša vyrábané tak, aby čo maximálne urýchlili nabíjací proces bez ohľadu účinkov na batériu. Okrem toho sa výrobcovia snažia o maximálnu univerzálnosť nabíjačiek, čo nie je správne, keďže rozdielne typy batérií majú rozdielny proces nabíjania. Všetky batérie cyklickým nabíjaním a vybíjaním strácajú postupne svoje vlastnosti v prípade nesprávneho používania to platí dvojnásobne. V prípade nami zvolených li-ion batérií je ich životnosť stanovená na približne 500 až 1000 cyklov v závislosti na presnom zložení batérie. V tejto časti sa budeme venovať procesu nabíjania a vybíjania li-ion batérií. Problémom, ktoré počas týchto procedúr môžu nastať a ako im predísť. Li-ion batérové články sa vyrábajú v rôznych prevedeniach, (cylindrické, prizmatické, gombíkové, ...) väčšinou majú nominálne napätie okolo 3.7V [2].

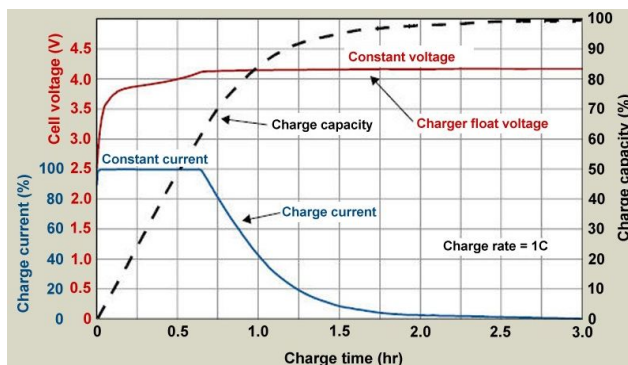


Obr. 2 Rozdielne tvary li-ion batérií

- + Relatívne vysoké operačné napätie okolo 3,7V
- + Vysoká výkonová hustota
- + Bez pamäťového efektu
- + Samovybíjanie je približne 1-2% za mesiac
- + Výborné nabíjacie/vybíjacie charakteristiky
- + Široký rozsah pracovných teplôt od -20°C do +60°C
- Potrebná ochrana článkov
- Starnutie - životnosť 500 až 1000 cyklov, znižovanie kapacity článku
- Vyššia cena

A. Nabíjanie

Štandardný a výrobcom udávaný ako správny proces nabíjania pozostáva z 2 fáz [3].



Obr. 3 Nabíjacia charakteristika li-ion batérie [4].

- 1) Fáza konštantného prúdu - do batérie tečie konštantný prúd. Tento prúd sa pohybuje v hodnotách od 0,5C do 1C v závislosti od toho, aké rýchle nabíjanie požadujeme. Výrobcovia odporúčajú voliť hodnotu nabíjacieho prúdu maximálne 0,8C. Počas tejto fázy bude prúd konštantný a napätie bude narastať. Táto fáza končí, keď napätie na svorkách batérie dosiahne maximálne napätie batérie. Li-ion batérie majú toto napätie stanovené na hodnotu 4,1V - 4,3V v závislosti na type batérie [3], [5].

- 2) Fáza konštantného napätia - na batériu pripojíme konštantné napätie o hodnote maximálneho napätia. Nabíjací prúd začne klesať. Táto fáza končí, keď je prúd tečúci do batérie menší ako 3% z prúdu batérie. Táto druhá fáza nie je vždy žiadúca, keďže je schopná pridať batérii okolo 15% kapacity, avšak stresuje batériu a tým mierne znižuje jej životnosť [3], [5].

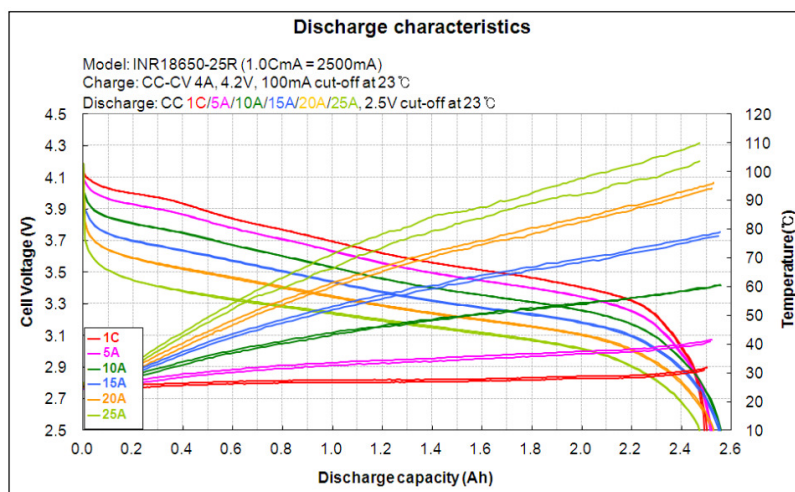
Po ukončení nabíjania je vhodné nechať batériu chvíľku odpojenú bez akejkoľvek záťaže.

1) *Ochrana počas nabíjania:*

- Regulácia nabíjacieho prúdu - Nabíjací prúd musí byť obmedzený a konštantný. Typická hodnota je 0,8C [6].
- Teplota počas nabíjania - teplota batérového článku by sa mala počas nabíjania monitorovať. Batérie sa nesmú nabíjať pokiaľ je teplota článku nižšia ako 0 °C alebo vyššia ako 45 °C.
- Ochrana proti prepätiu - ochrana, ktorá zabezpečí to, že na svorkách batérie nebude vyššie napätie ako 4,3V [6].
- Ochrana proti prepólovaniu - odpojí batériu v prípade nesprávneho zapojenia [6].

B. *Vybíjanie*

Moderné li-ion batérie zvládajú široký rozsah vybíjajúcich prúdov. Výrobcovia zväčša stanovujú optimálnu hodnotu vybíjacieho prúdu batérie. Táto hodnota sa označuje ako 1C (pri jednej batérii môže byť 1C = 500mA pri inej 1C = 1000mA) [7]. Pri tejto hodnote výrobca garantuje životnosť jednotlivých článkov. Li-ion batérie môžeme samozrejme vybíjať aj vyšším prúdom ako 1C, no následkom bude najmä vyššia teplota batérie pri veľkých vybíjajúcich prúdoch a to až 100 °C. Takisto má vyšší vybíjací prúd vplyv aj na kapacitu batérie po 100 a viac cykloch [8].



Discharge current						
	1C	5A	10A	15A	20A	25A
Capa.(Ah)	2.496	2.518	2.556	2.550	2.525	2.472
Temp.(°C)	31.6	41.2	60.6	78.4	95.2	106.8

Obr. 4 Vybíjacie charakteristiky pri rôznych vybíjajúcich prúdoch pre batériu Samsung INR18650-25R [9].

1) *Ochrana počas vybíjania:*

- Ochrana proti skratu - li-ion batérie sú citlivé na skraty, teda je potrebná ochrana, ktorá zakročí v prípade tvrdého skratu [10].
- Teplota počas vybíjania - teplota počas vybíjania by nemala prekročiť hranicu 100 °C. Teplota vyššia ako 100 °C môže spôsobiť poškodenie vnútornej štruktúry článku [10].
- Ochrana proti podvybitiu - zabráni poklesu napätia článku batérie pod minimálnu hodnotu, ktorá je zvyčajne 3V v závislosti od výrobcu [11].
- Ochrana proti prepólovaniu

III. NÁVRH KONCEPCIE

V predošlých sekciách sme zhrnuli spôsob ako vhodne nabíjať a vybíjať li-ion batérie. Spomenuli sme faktory, ktoré ovplyvňujú životnosť batérie a teda je potrebné dať na ne pozor počas procesu nabíjania a vybíjania. Podľa týchto požiadaviek sme navrhli koncepciu nabíjačky/vybíjačky spolu s blokmi na zabezpečenie optimálneho nabíjania a vybíjania.

A. Požiadavky

- Nabíjanie a vybíjanie rôznych li-ion článkov
- Možnosť nabíjať/vybíjať až 8 článkov naraz.
- Možnosť nastavenia nabíjacej krivky a vybíjacieho profilu
- Nabíjací prúd max 5A vybíjací max 30A
- Ochrana proti prepólovaniu, skratu, podvybitiu, prepätiu a prehriatiu
- Monitorovanie napätí, prúdov, teplôt a stavu nabitia článku

B. Návrhy Konceptíí

Pretože chceme, aby bol systém stabilný, časť riadiacich systémov (regulátory) postavíme na analógovej technike. Keďže modul bude slúžiť aj na zaťažovanie, a to nechceme vždy konštantné, potrebujeme mať možnosť meniť žiadanú vstupnú hodnotu do regulátorov. To dosiahneme tak, že ako vstup do regulátora prúdu použijeme mikroprocesor. Tu nastáva jeden problém a to je potreba spojiť analógový regulátor s digitálnym procesorom. Teda je potrebné voliť procesor s DAC (Digital-to-analog converter) prevodníkom, prípadne do obvodu doplniť externý DAC prevodník. Pri takejto konfigurácii vieme nahráť rôzne tvary vybíjacích a nabíjacích charakteristík priamo do pamäte mikroprocesora, ktorý následne za pomoci regulátorov a výkonových komponentov vykoná zvyšok práce. Úlohou mikroprocesora bude okrem riadenia regulátorov aj meranie a zber dát počas činnosti zariadenia. Dáta, ktoré by sa mali zbierať, sú popísané v sekcii požiadavky.

C. Konceptia s Jedným Procesorom

Konceptia je založená na jednom hlavnom, riadiacom procesore, ktorého úlohou je poskytovať aj HMI (Human-machine interaction). Pri takejto koncepcii je potrebné použiť výkonnejší mikroprocesor s väčším počtom GPIO (General-purpose input/output) pinov. Keďže chceme naraz testovať až 8 batérií a na každej chceme merať minimálne napätie, prúd a teplotu. Budeme teda potrebovať minimálne 24 GPIO pinov len na monitorovanie batérií počas testovania. K tomu je potrebné pripočítať požiadavky aj na riadenie regulátorov. Okrem toho musí mať procesor dostatočnú pamäť. Na trhu sú dostupné dostatočne výkonné mikroprocesory, ktoré by túto úlohu zvládli, avšak výhod v tejto koncepcii je menej ako nevýhod.

- + Menej komponentov a nižšia cena celkového zariadenia
- Vysoké požiadavky na MCU, náročnejší program a jeho odladenie
- V prípade, ak zlyhá niečo v obvode s batériou, môže sa to preniesť na procesor, ktorý ak zlyhá, zlyhá celé zariadenie

D. Konceptia s Pomocným Procesorom

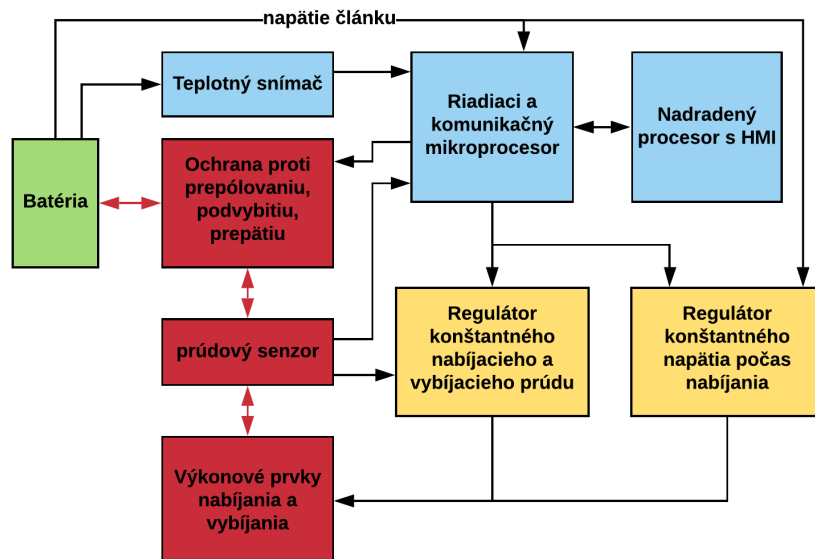
Oproti predošlej koncepcii by nám pribudol jeden pomocný mikroprocesor. Ten by nemusel byť extrémne výkonný a ani nebude potrebovať veľký počet GPIO pinov. Jeho úlohou by bolo riadiť vstupné žiadané hodnoty do regulátorov a monitoring počas činnosti. V procesore by boli takisto nahrané krivky pre nabíjanie a vybíjanie daného článku. Procesor by bol umiestnený na doske s batériou, regulátormi a snímačmi pre jeden článok. Celý tento modul by sa potom pripájal k hlavnému procesoru s HMI. Vďaka takémuto modulárnemu dizajnu by bolo kedykoľvek možné k procesoru pripojiť ďalšie moduly a tak vykonávať ľubovoľný počet nezávislých testov. Ďalším plusom takejto koncepcie je to, že sa zjednoduší programovanie, keďže procesory budú mať menej práce. To bude viesť k stabilnejšiemu chodu zariadenia. Výhodou bude aj to, že pokiaľ sa niečo stane v obvode jedného článku, zasiahne to iba pomocný mikroprocesor a nie hlavný. Takže zvyšok systému bude naďalej fungovať.

- + V prípade zlyhania jedného modulu zvyšok systému funguje
- + Menšie nároky na MCU, jednoduchšie programovanie a ladenie
- + Rýchlejší a stabilnejší systém
- + Možnosť vytvorenia modulárne plug&play riešenia (jednoduché pridanie ďalších článkov).
- Viac komponentov a vyššia cena
- Je potrebné riešiť medziprocesorovú komunikáciu

E. Návrh Výslednej Konceptie

Je jasné vidieť, že výhody v koncepcii s pomocným procesorom prevyšujú nevýhody a zvolíme si ho ako koncept pre ďalší vývoj zariadenia.

Nižšie na Obr. 5 je znázornená blokovaná schéma pre jeden batériový článok. Červenou farbou sú označené výkonové časti a cesty zariadenia, kde sa predpokladá väčší prúd (podľa charakteristík na Obr. 4 aj 25A). Modrou farbou sú označené snímacie, riadiace a komunikačné časti zariadenia. Žltá farba znázorňuje regulátory. Keďže chceme testovať až 8 článkov, bude potrebných 8 takýchto testovacích modulov.



Obr. 5 Bloková schéma nabíjacieho a vybíjacieho modulu s pomocným procesorom

IV. ZÁVER

Článok bol venovaný li-ion batériovým článkom. Poradilo sa v ňom zhrnúť požiadavky pre nabíjanie a vybíjanie daných li-ion batériových článkov. Boli popísané hraničné stavy batérie, ktoré bude potrebné ošetriť ochranami. Taktiež sa podarilo navrhnuť vhodnú koncepciu výkonovej časti zariadenia na cyklické testovanie batérií. Vďaka tomu bude možné pokračovať v ďalšom výskume a to je návrh a zostavenie jednotlivých blokov podľa koncepcie na Obr. 5. Výsledkom práce bude zariadenie pre cyklické nabíjanie a vybíjanie batériových článkov podľa programovateľného profilu.

POĎAKOVANIE

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe zmluvy č. APVV-18-0436.

Práca bola takisto podporovaná výskumným grantom pre mladých výskumníkov do 30 rokov a doktorandov na TUKE č. 04/TUKE/2020 s názvom Zariadenie pre cyklické nabíjanie a vybíjanie batériových článkov s programovateľným profilom.

LITERATÚRA

- [1] QNOVO, "A peek inside the battery of a tesla model s," 2014. [Online]. Available: <https://qnovocom/peek-inside-the-battery-of-a-tesla-model-s/>
- [2] J. Biľanský, "Overview of the Battery Types and their Testing," *Nonconference Proceedings of Young Researchers, SCYR 2020*, pp. 143–146, April 2020.
- [3] Shang-Wen Luan, Jen-Hao Teng, Dong-Jing Lee, Yong-Qing Huang, and Chen-Lin Sung, "Charging/discharging monitoring and simulation platform for li-ion batteries," in *TENCON 2011 - 2011 IEEE Region 10 Conference*, 2011, pp. 868–872.
- [4] Battery University, "Li-ion charging characteristics," 2018. [Online]. Available: https://batteryuniversity.com/_img/content/new.jpg
- [5] A. B. Khan, Van-Long Pham, Thanh-Tung Nguyen, and Woojin Choi, "Multistage constant-current charging method for li-ion batteries," in *2016 IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific)*, 2016, pp. 381–385.
- [6] Z. Ye, X. Wu, Y. Sun, and J. Lu, "A universal protection controller for li-ion battery charger," in *2010 Asia-Pacific Power and Energy Engineering Conference*, 2010, pp. 1–4.
- [7] N. Meena, V. Baharwani, D. Sharma, A. Sharma, B. Choudhary, P. Parmar, and R. B. Stephen, "Charging and discharging characteristics of lead acid and li-ion batteries," in *2014 POWER AND ENERGY SYSTEMS: TOWARDS SUSTAINABLE ENERGY*, 2014, pp. 1–3.
- [8] C. Wey and P. Jui, "A unitized charging and discharging smart battery management system," in *2013 International Conference on Connected Vehicles and Expo (ICCVEx)*, 2013, pp. 903–909.
- [9] L. Samsung SDI Co., "Inr18650-25r," <https://www.tme.eu/Document/d5041798b41b6ad5e98cd9d1377d272d/INR18650-25R.pdf>, 2014, lithium-ion rechargeable cell for power tools.
- [10] Y. Hsieh, J. Wu, and Q. Kuo, "A li-ion battery string protection system," in *2011 International Conference on Applied Electronics*, 2011, pp. 1–4.
- [11] J. Xiong, H. Banvait, L. Li, Y. Chen, J. Xie, Y. Liu, M. Wu, and J. Chen, "Failure detection for over-discharged li-ion batteries," in *2012 IEEE International Electric Vehicle Conference*, 2012, pp. 1–5.

Robotický operačný systém a jeho použitie v evolučnej robotike

¹Michaela TARHANIČOVÁ, ²Marián MACH

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹michaela.tarhanicova@student.tuke.sk, ²marian.mach@tuke.sk

Abstrakt — Cieľom článku je problematika návrhu pomocou prístupu evolučnej robotiky, založenej na využití evolučného procesu postupného generovania a prehľadávania alternatívnych riešení. Pozornosť je venovaná Robotickému operačnému systému a robotickému simulátoru Gazebo, umožňujúcemu vytváranie vhodného simulačného prostredia pre vyhodnocovanie kvality generovaných alternatív. Článok sa zaoberá možnosťou prepojenia robotického simulátora Gazebo s evolučným algoritmom prostredníctvom komunikačných kanálov, poskytovaných Robotickým operačným systémom. V rámci experimentovania boli vykonané dva jednoduché experimenty, zamerané ako na evolúciu parametrov robota tak aj na evolúciu parametrov prostredia, v ktorom sa robot pohyboval.

Kľúčové slová — evolučná robotika, Gazebo, Robotický operačný systém, TurtleBot

I. ÚVOD

Výskum v oblasti robotiky smeruje k návrhu robotov disponujúcich takými schopnosťami ako sú autonómia a inteligencia. Takéto roboty sa dokážu prispôsobiť svojmu okoliu a vďaka tomu uspokojivo fungovať v neustále sa meniacom prostredí. Jeden z prúdov zakladá na vybavení robotov jednoduchým typom správania, ktoré sa bude postupne transformovať na stále zložitejšie a zložitejšie správanie – až do stavu keď je možné tomuto správaniu prisúdiť určitý stupeň inteligencie [1]. Do tohto prúdu patrí aj prístup reprezentovaný evolučnou robotikou, ktorá aplikuje metódu selekcie, rozmanitosti a princípov dedičnosti prirodzenej evolúcie pre návrh robotov.

Pri technickej realizácii sa často využíva Robotický operačný systém (ROS), ktorý nie je v skutočnosti klasickým operačným systémom, ale skôr open-source meta-operačným systémom, poskytujúcim hardvérovú abstrakciu pre prácu s rôznymi periférnymi zariadeniami [2].

Preto sa zdá prirodzeným spojenie systému ROS a metód evolučnej robotiky. Tento článok referuje o jednom takomto pokuse.

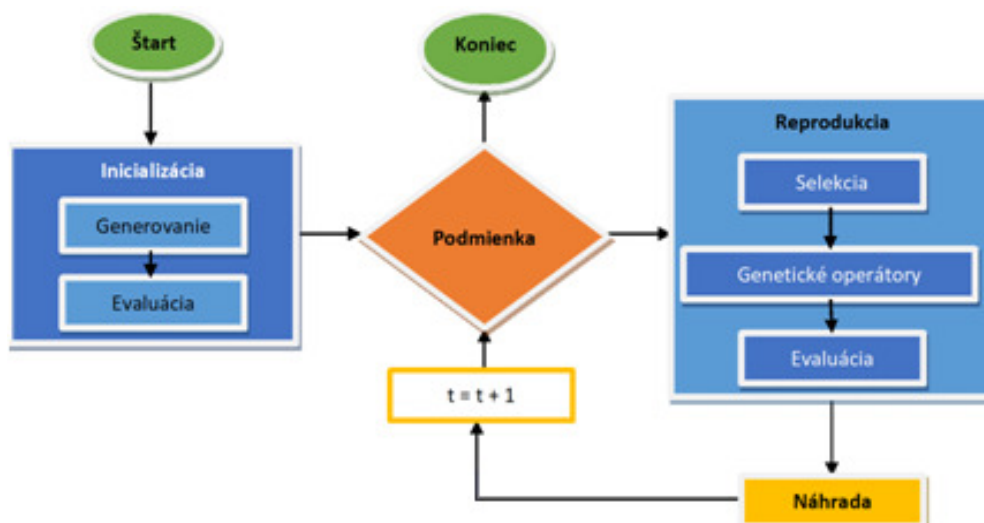
II. EVOLUČNÁ ROBOTIKA

Evolučná robotika sa vyčlenila z odvetvia evolučných výpočtov, ktoré dokážu úspešne riešiť problémy v optimalizácii ako aj modelovaní a dizajnu, počas deväťdesiatych rokov ako metóda automatického vytvárania robotov. Predstavuje metódu návrhu inšpirovanú princípom rôznorodosti a selektívnym prístupom. Je ju možné využiť napríklad pre návrh senzorového aparátu robota alebo aj jeho celkovej morfológie [3] ale tak isto aj pre vývoj kontrolných systémov robota [4].

Evolučná časť je založená na evolučných algoritmoch (EA), inšpirovaných Darwinovou teóriou prirodzeného výberu [5]. Schéma EA je znázornená na obr. 1. Základnou časťou algoritmu je populácia jedincov, z ktorých každý kóduje jednu alternatívu návrhu, pričom kvalita danej alternatívy je ohodnotená jeho vhodnosťou.

Začiatkom celého procesu je inicializácia, kde sa náhodne vygeneruje prvotná podoba populácie a všetky jedince sú ohodnotené. Vytvorená populácia sa postupom času vyvíja v cykle pomocou série iterácií, kde pomocou reprodukčného procesu sú generované nové alternatívne návrhy. Tieto následne súťažia s pôvodnými návrhmi o to, kto nahradí aktuálne

prítomných jedincov v populácii a vytvorí tak novú populáciu, vstupujúcu do ďalšej iterácie evolučného procesu.



Obr. 1 Schéma evolučného algoritmu

Samotný reprodukčný proces pozostáva z troch krokov. Najprv sa z alternatívnych návrhov, ktoré sú zaradené v aktuálnej populácii, vyberá preferovaná skupina (prihliadajúc ku kvalite jednotlivých návrhov). Z jedincov tejto skupiny sú odvodené nové návrhy pomocou genetických operátorov, ktoré sa snažia zachovať niektoré časti selektovaných návrhov a súčasne nahradiť ich ostávajúce časti novými alternatívami. Vygenerované nové alternatívne návrhy sú podrobené evaluácii, kde sa určuje ich kvalita.

V prípade evolučnej robotiky je najprv potrebné určiť, ktorá časť robota má byť vyvíjaná evolučným spôsobom. V závislosti na výbere sa určí potrebná reprezentácia tak, aby každý jedinec (či už generovaný pri inicializácii alebo reprodukčným procesom) mohol reprezentovať jednu alternatívnu podobu vyvíjanej časti robota. Vhodnosť jedinca určovaná evaluačným procesom je závislá od toho, akým kvalitným riešením daná alternatíva je. Dalo by sa to samozrejme testovať priamo na reálnom robotovi v reálnom prostredí, avšak je nutné realizovať viacero vyhodnotení (a nie všetky alternatívne návrhy by robot mohol prežiť) a preto sa prednosť dáva simulovanému robotovi v simulovanom prostredí.

III. NÁVRH ARCHITEKTÚRY

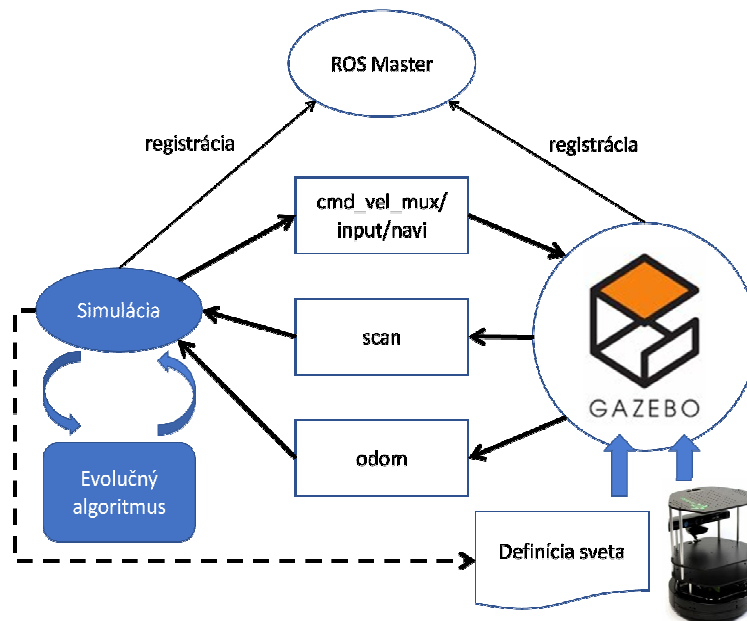
Pre experimentovanie s evolučnou robotikou v prostredí ROS-u sme navrhli architektúru podľa obr. 2. Evolučný algoritmus má klasickú podobu s jednou výnimkou – určovať vhodnosť jedincov nedokáže algoritmus sám tradičným spôsobom (vložením definície funkcie vhodnosti priamo do algoritmu) ale musí sa spoliehať na externý prvok. V našom prípade rolu tohto externého prvku hrá simulátor *Gazebo*, ktorého činnosť musí byť inicializovaná a riadená evolučným algoritmom. Súčasťou algoritmu je preto blok *Simulácia*, ktorý je zodpovedný za celkovú komunikáciu so simulátorom.

Aby blok *Simulácia* bol schopný komunikácie so simulátorom *Gazebo*, sú oba realizované ako uzly systému ROS, ktoré sa klasickým spôsobom registrujú v uzle ROS Master, poskytujúcim menňé a registračné služby v systéme ROS (vrátane registrácie publikovania a odoberania správ v rámci jednotlivých tém).

Simulácia komunikuje s *Gazebo* prostredníctvom správ v rámci registrovaných tém. Použité sú tri témy:

- `cmd_vel_mux/input/navi`,
- `scan`,
- `odom`.

Prvá téma slúži pre prenos správy typu *geometry_msgs/Twist*. Typ správy *geometry_msgs* poskytuje rôzne správy týkajúce sa rôznych geometrických primitív (vrátane pozície a orientácie v priestore). Správa *Twist*, patriaca do tejto skupiny, vyjadruje rýchlosť reprezentovanú dvomi trojrozmernými vektormi – vektorom lineárnej rýchlosti a vektorom uhlovej rýchlosti. Z týchto šiestich hodnôt používame iba dve lineárne rýchlosti (v smere osí *x* a *y*) a jednu uhlovú rýchlosť (otáčanie okolo osi *z*).



Obr. 2 Komunikácia medzi evolučným algoritmom a simulátorom

Téma *scan* je použitá pre prenos správy *sensors_msgs/LaserScan*. Skupina správ *sensors_msgs* definuje správy pre bežne používané senzory, pričom jej správa *LaserScan* je určená pre planárny laserový merač vzdialenosti.

A téma *odom* je určená pre prenos správy *nav_msgs/Odometry*. Rodina správ *nav_msgs* definuje správy používané pri interakcii s navigačným balíkom, kde správa *Odometry* reprezentuje odhad pozície a rýchlosti. Pre naše účely z tejto správy vyberáme informáciu o póze robota vo svete, pričom sa berú do úvahy iba dve hodnoty pre pozíciu a jedna hodnota pre orientáciu robota.

Simulátor Gazebo pre realizáciu simulácie potrebuje dve vstupné definície – definíciu simulovaného sveta (zahŕňajúcu napríklad definovanie geometrie, materiálu a povrchu jednotlivých prvkov sveta alebo osvetlenie scény) a definíciu použitého robota. Keďže svet reprezentuje statické prostredie pre simuláciu, v prípade nutnosti jeho zmeny musí byť upravený pred začiatkom simulácie. V úlohe robota musí byť použitý taký robot, ktorý je schopný komunikovať počas simulácie prostredníctvom uvedených správ.

V priebehu simulácie simulátor publikuje informácie o aktuálne meraných hodnotách laserového merača vzdialenosti a odometrii simulovaného robota. Tieto správy sú odoberané evolučným algoritmom a na ich základe blok Simulácia určuje a publikuje požadované hodnoty lineárnej a uhlovej rýchlosti, ktoré sú odoberané simulátorom a na ich základe je príslušným spôsobom upravovaný priebeh simulácie.

Je zrejmé, že uvedená architektúra umožňuje realizovať simulácie s robotom v rôznych prostrediach, pričom robot je schopný sa v použítom prostredí pohybovať a identifikovať prekážky, ktoré mu bránia v zamýšľanom pohybe.

IV. NASTAVENIE EXPERIMENTOV

Pre experimentovanie v prostredí Gazebo bol použitý robot TurtleBot, pretože má senzor (Kinect) umožňujúci registrovať prekážku, poskytuje informácie o odometrii a aj možnosť prístupovať k svojmu vnútornému svetu. Ten sa mu pri každom zapnutí nastaví na nulovú pozíciu. Nevýhodou je to, že jeho senzor má obmedzený uhlový rozsah – dokáže identifikovať prekážku pred robotom avšak nie vedľa robota.

Robot sa počas experimentov pohyboval v prostredí s prekážkami, ktoré boli reprezentované stenami. Prvotne vytvorené prostredie obsahovalo iba tieto „prekážkové“ steny, čo sa napokon prejavilo ako nepostačujúce. Bolo nutné vytvoriť aj steny, ktoré ohraničovali celý priestor, ktorý mal robot k dispozícii, aby z tohto priestoru počas simulácie neunikal.

Keďže cieľom našich experimentov nebolo vyvíjať riadenie, rozhodli sme sa pre jednoduchú stratégiu. Najprv sme nechali robota ísť rovno až kým nenarazil na prekážku. Po náraze sa robot náhodne rozhodol či sa otočí doprava alebo doľava nehladiac na to, či po otočení bude mať pred sebou voľnú cestu (ak nie, tak sa opäť náhodne otáčal). Tento spôsob sa ukázal ako

nedostatočný a preto bol zmenený. Výsledná stratégia bola nastavená tak, že robot išiel rovno a po náraze na prekážku sa otočil doprava aj doľava, aby zistil v ktorom smere je voľná cesta (toto kompenzovalo neexistenciu bočných senzorov). Ak voľných ciest bolo viac, vybral si podľa svojej preferencie. Táto stratégia bola realizovaná ako pravidlový systém s týmito pravidlami:

1. Chod rovno
2. Ak zastaneš kvôli prekážke pred sebou, otoč sa doprava a zisti akú máš vzdialenosť od prekážky
3. Otoč sa doľava
4. Otoč sa doľava a zisti, akú máš vzdialenosť od prekážky
5. Otoč sa doprava
6. Rozhodni sa, ktorým smerom sa pohneš na základe vzdialeností – otoč sa do toho smeru, kde je prekážka vzdialenejšia
7. Otoč sa zvoleným smerom
8. Chod na pravidlo 1

Takýto súbor pravidiel si vyžiadalo ohraničenie kladené na tvar prostredia, v ktorom sa robot pohyboval. Prostredie už nemohlo byť konštruované ľubovoľne ale muselo rešpektovať tri podmienky:

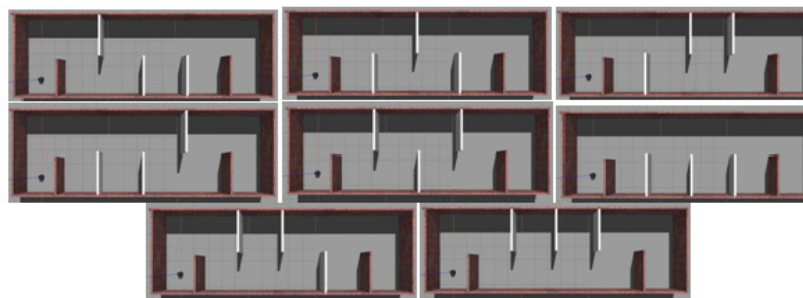
- Steny okolo celého sveta – keďže robot bol riadený vzdialenosťou od prekážky, tak v každom smere musel byť schopný nejakú prekážku identifikovať.
- Po náraze na prekážku nemôže byť prekážka na ľavej strane robota v rovnakej vzdialenosti ako prekážka na jeho pravej strane.
- Prekážky nemôžu byť usporiadané tak, aby sa robot v prostredí zacyklil na nejakej trajektórii prechádzajúcej časťou prostredia.

V. EXPERIMENTY

Pre účely testovania navrhutej štruktúry boli realizované dva podobné experimenty, zamerané na rôzne prvky prostredia s robotom [6]. V prvom experimente sa menilo práve prostredie v ktorom sa robot pohyboval, pričom samotný robot sa nemenil. V druhom experimente bola situácia opačná – menil sa robot, ktorý sa pohyboval v nemennom prostredí. Cieľom oboch experimentov bolo overiť, či Gazebo je použiteľným simulátorom pre účely evolučnej robotiky.

A. Evolúcia prostredia

Pre tento experiment boli použité prostredia, ktoré sa môžu meniť presúvaním stien do rôznych polôh. Rôzne polohy stien nútia robota, ktorý má za úlohu prejsť z počiatočnej pozície na zadanú cieľovú pozíciu, pohybovať sa po rôznych (aj keď podobných) trajektóriách. Jedno z týchto prostredí (jednoduchý svet A) je zobrazené na obr. 3.



Obr. 3 Rôzne konfigurácie sveta A

Svet A mal tri konfigurovateľné steny (na obrázku znázornené bielou farbu). Každá z týchto stien mohla byť v jednej z dvoch polôh – buď v hornej alebo dolnej polohe, takže existovalo osem rôznych konfigurácií sveta. Štartovacia poloha je vyznačená čiernym bodom, cieľová poloha bola v podobnej pozícii avšak v najpravejšom oddelení daného sveta.

Evolučný algoritmus svojimi jedincami reprezentoval polohu prestaviteľných stien. Funkcia vhodnosti bola definovaná ako čas, potrebný k prechodu robota zo štartovacej pozície do cieľovej. Populácia bola tvorená štyrmi jedincami, pričom v každom cykle boli vytvárané štyri nové jedince pomocou pravdepodobnostného kombinovania kríženia (jedince si vymieňali polohu každej druhej steny) a mutácie (poloha náhodne vybranej steny bola zmenená). Iniciálna

populácia bola generovaná náhodne.

Rýchlosť lineárna pri priamom pohybe aj uhlová pri otáčaní bola nastavená na pevné hodnoty, ktoré sa počas evolúcie nemenili.

Získané výsledky sú v tab. 1. Evolučný algoritmus dokázal identifikovať konfiguráciu prostredia, ktorá je z hľadiska času prechodu najvhodnejšia (č. 1) a rovnako aj najmenej vhodná (č. 6). Tieto identifikované konfigurácie sú plne v súlade s intuitívnym očakávaním najrýchlejšej cesty v prípade 1 (všetky steny umiestnené dole, takže nebránili robotovi v priamočiaram pohybe od ľavého okraja sveta k pravému okraju) a najpomalšej cesty v prípade 6 (z hľadiska pohybu robota najčlenitejšia, nútiaca ho kaskádovito mnohokrát prechádzať priestor zhora nadol a naopak).

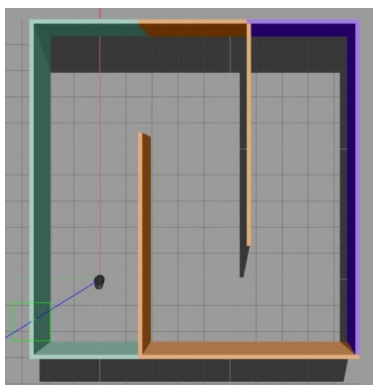
Tabuľka 1
Získané časy prechodov v svete A

Č.	1. stena	2. stena	3. stena	Najlepší čas [ms]
1	Dole	Dole	Dole	525
2	Dole	Dole	Hore	1270
3	Dole	Hore	Dole	1266
4	Dole	Hore	Hore	1260
5	Hore	Dole	Dole	1268
6	Hore	Dole	Hore	1973
7	Hore	Hore	Dole	1262
8	Hore	Hore	Hore	1275

Podobné chovanie evolučného algoritmu bolo identifikované aj v ďalších dvoch svetoch, ktoré boli použité v experimente.

B. Evolúcia parametrov robota

Pre tento experiment boli použité prostredia, ktoré sa počas evolúcie nemenili. Menili sa parametre robota – hodnota lineárnej rýchlosti pre pohyb robota v priamom smere a hodnota uhlovej rýchlosti robota pri jeho otáčaní. Rôzne hodnoty týchto parametrov ovplyvňujú rýchlosť pohybu robota v prostredí a teda aj dobu prechodu z počiatočnej pozície na zadanú cieľovú pozíciu. Jedným z použitých prostredí bol svet A (obr. 3) v konfigurácii 6 (tab. 1), iné je zobrazené na obr. 4 (jedna z konfigurácií sveta E).



Obr. 4 Jedna z konfigurácií sveta E

Evolučný algoritmus svojimi jedincami reprezentoval dva robotove parametre ako dvojicu reálnych čísel – obe z intervalu $<0,1>$. Funkcia vhodnosti bola aj teraz definovaná ako čas, potrebný k prechodu robota zo štartovacej pozície do cieľovej. Podobne ako v predchádzajúcom experimente, populácia bola tvorená štyrmi jedincami, pričom v každom cykle boli vytvárané štyri nové jedince pomocou kombinovania kríženia (jedince si vymieňali uhlové rýchlosti) a mutácie (hodnota náhodne vybraného parametra bola zvýšená alebo znížená o malú hodnotu). Iničiálna populácia bola generovaná náhodne z príslušného definičného oboru.

Získané výsledky sú uvedené v tab. 2. Evolučný algoritmus nedokázal identifikovať takú konfiguráciu rýchlostí, ktorá by mohla byť intuitívne očakávaná (maximálne hodnoty rýchlosti

umožňujúce najrýchlejší pohyb). Dôvodom je to, že v prípade vyšších hodnôt rýchlostí simulovaný robot nezvládal pohyb vo voľnom priestore medzi stenami a narážal do nich. V takýchto prípadoch bola simulácia ukončená a vhodnosť príslušnej kombinácie hodnôt rýchlostí bola nastavená na hodnotu, ktorá daného jedinca nepreferovala do úlohy favorita v evolučnom procese.

Tabuľka 2
Získané časy prechodov v svetoch A a E

	Lineárna Rýchlosť	Uhlová rýchlosť	Najlepší čas [ms]
Svet A	0.4	0.2	1973
Svet E	0.55	0.42	995

VI. ZÁVER

Vo vykonaných experimentoch evolučný algoritmus dokázal nájsť vhodné riešenie. Avšak použitie simulátora Gazebo v spojitosti s jednoduchým riadením pohybu robota bez spätnej väzby z okolia o okamžitej pozícii malo za následok počas simulácie časté vychýľovanie robota z predpokladaného smeru a vrážanie do stien, čím určitý počet simulácií končil neúspechom. V tomto prípade bolo nutné simuláciu opakovať. Značný počet takýchto neúspešných simulácií mal za následok predĺženie doby behu samotného evolučného procesu, čím sa celkový proces stal časovo príliš náročným. Pri ďalších pokusoch je nutné použiť taký spôsob riadenia robota, ktorý dokáže korigovať vybočenia z predpokladaného smeru pohybu a tým zabráni zbytočnému predlžovaniu doby evolúcie kvôli vzniku neúspešných simulácií.

POĎAKOVANIE

Táto publikácia vznikla vďaka podpore Agentúry pre podporu výskumu a vývoja v rámci projektu APVV-15-0731 2016-2020 „Cloud Based Human Robot Interaction“.

LITERATÚRA

- [1] J. C. Bongard: Evolutionary Robotics. Communications of the ACM, 56 (8), 2013, 74-85.
- [2] M. Quigley, B. Gerkey, W. D. Smart: Programming Robots with ROS, O'Reilly Media, Inc, USA, 2015.
- [3] S. Doncieux, N. Bredeche, J. B. Mouret, A. E. Eiben: Evolutionary Robotics: What, Why, and Where to. Frontiers in Robotics and AI, 2 (March), 2015, 1-18.
- [4] E. Haasdijk, N. Bredeche, S. Nolfi, A. E. Eiben: Evolutionary Robotics. Evolutionary Intelligence, 7 (2), 2014, 69-70.
- [5] M. Mach: Evolučné algoritmy – Prvky a princípy. Elfá, Košice, 2009.
- [6] M. Tarhaničová: Evolučná robotika. Diplomová práca, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, 2018, 55 strán.

Možnosti akumulácie tepelnej energie

¹Vladimír JURENKA, ²Dušan MEDVEĎ

^{1,2} Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky Technickej univerzity v Košiciach, Slovenská republika

¹vladimir.jurenka@student.tuke.sk, ²dusan.medved@tuke.sk

Abstrakt — Článok sa zaoberá hľadaním alternatívnych zdrojov tepelnej energie na pokrytie spotreby tepelnej energie na ohrev teplej úžitkovej vody a na vykurovanie v domácnostiach. Dôležitou súčasťou je výpočet tepelnej bilancie materiálov na zistenie či daný materiál v zadanom množstve vyhovuje na pokrytie spotreby tepelnej energie v domácnosti.

KLúčové slová — akumulácia tepelnej energie, solárne kolektory, vykurovanie, teplá úžitková voda

I. ÚVOD

Fosilne palivá sú v dnešnej dobe už palivá, ktoré tzv. „dosluhujú“ našej populácii. Preto je dôležité hľadať nové alternatívy energie, ktorá nám je potrebná pre náš každodenný život.

Zdrojov, z ktorých vieme získať teplo, je mnoho a v tomto článku bolo zvolené získanie tepla zo Slnka pomocou solárnych kolektorov. Získané teplo je potrebné uskladniť do materiálu a neskôr z daného materiálu naakumulované teplo využiť na pokrytie energie na vykurovanie a ohrev teplej úžitkovej vody (TÚV) v domácnosti. Úlohou bolo zistiť, aký materiál je najvhodnejší pre akumuláciu tepla, ktorá by postačovala na pokrytie spotreby tepla pre domácnosť počas celého roka.

II. AKUMULÁCIA CITEĽNÉHO A LATENTNÉHO TEPLA

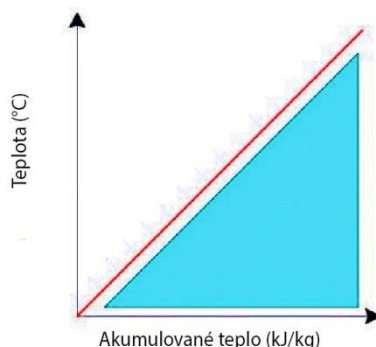
A. Akumulácia citeľného

Pri akumulácii citeľného tepla sa tepelná energia uskladňuje počas ohrievania látky, ktorá má vhodné vlastnosti na tieto účely. Tento spôsob sa využíva predovšetkým na akumuláciu ohriatej pitnej vody v obytných budovách [1].

Fyzikálna podstata akumulácie citeľného tepla je založená na princípe kalorimetrickej rovnice, ktorú definuje nasledujúci vzťah:

$$Q = m \cdot c \cdot (\vartheta_1 - \vartheta_2) \quad (1)$$

kde Q teplo dodané/odobrané látke [J]
 m hmotnosť [kg]
 c merná tepelná kapacita látky [J/(kg · K)]
 ϑ_1 teplota látky [°C]
 ϑ_2 teplota látky [°C]



Obr. 2 Princíp akumulácie citeľného tepla [1]

K výhodám tohto systému patria najmä nízke investičné náklady, netoxičita látky na akumuláciu tepla a dostupný sortiment zásobníkov na akumuláciu teplej vody. Za negatívum možno považovať fakt, že tepelná kapacita akumulovaného citeľného tepla je limitovaná. Na uskladnenie veľkého množstva energie je teoreticky potrebný veľký objem zásobníka, čo znižuje účinnosť z hľadiska tepelných strát. Rovnako platí, že zvyšovaním teploty látky, za účelom uskladnenia väčšieho množstva tepla, narastá tepelná strata na rozhraní materiálu s okolitým prostredím, čím sa znižuje účinnosť procesu [1].

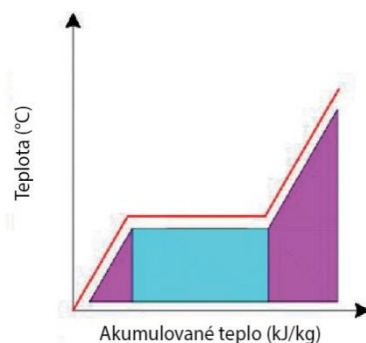
B. Akumulácia latentného tepla

Latentné teplo, inak nazývané aj ako skupenské teplo, je tepelná energia, ktorá sa uvoľní z látky alebo je do látky, dodaná pri transformácii skupenstva. Pri tejto forme akumulovanej energie možno docieľiť veľkú tepelnú kapacitu pri malom objeme a konštantnej teplote. Tento princíp akumulácie tepla sa aplikuje pri dlhodobom uskladnení tepelnej energie v procesoch, kde sa prísne požaduje dodržanie teploty. Nevýhoda akumulácie latentného tepla je v tom, že má nižšiu tepelnú vodivosť materiálu na zvýšenie tepelnej vodivosti a teda k zvýšeniu účinnosti môžeme pomôcť uhlíkovými vláknami alebo kovovou penou [1].

Fyzikálnu podstatu akumulácie latentného tepla definuje vzťah:

$$Q = m \cdot c_1 \cdot (\vartheta_1 - \vartheta_2) + L \cdot m + m \cdot c_2 \cdot (\vartheta_2 - \vartheta_3) \quad (2)$$

Kde Q teplo dodané/odobrané látke [J]
 m hmotnosť [kg]
 c merná tepelná kapacita látky [J/(kg · K)]
 ϑ teplota [°C]
 L špecifické latentné teplo látky [J·kg⁻¹]



Obr. 2 Princíp akumulácie latentného tepla [1]

III. MATERIÁLY PRE AKUMULOVANIE TEPELNEJ ENERGIE

A. Vlastnosti látok pre akumuláciu tepla:

- **Tepelné vlastnosti**
 - vhodná teplota topenia
 - vysoká merná tepelná kapacita
 - vysoké skupenské teplo topenia
- **Fyzikálne vlastnosti**
 - vysoká objemová hustota
 - žiadne alebo malé podchladenie počas tuhnutia
 - nízky tlak pár
- **Chemické vlastnosti**
 - predĺžená chemická stabilita
 - kompatibilita s konštrukčnými materiálmi zásobníka
 - netoxičita
 - nehorľavosť a neexplozívnosť
- **Ekonomické vlastnosti**
 - dostatočná zásoba
 - cenová dostupnosť

B. Parametre vybraných akumulčných materiálov

Tabuľka 1
 Vlastnosti vody

Názov	Voda
Objemová hustota ρ (kg/m ³)	999,972
Merná tepelná kapacita, tuhý stav c (J/kg·K)	4180
Merná tepelná kapacita, kvapalný stav c (J/kg·K)	4200
Tepelná vodivosť, tuhý stav λ (W/m·K)	2,3
Tepelná vodivosť, kvapalný stav λ (W/m·K)	0,55312
Teplota tuhnutia ϑ (°C)	-0,001
Teplota topenia ϑ (°C)	0
Teplota varu ϑ (°C)	100
Latentné teplo z tuhého do kvapalného L (J/kg)	334000
Latentné teplo z kvapalného do plynného L (J/kg)	2257000

Tabuľka 2
 Vlastnosti parafínového vosku

Názov	Parafínový vosk
Objemová hustota ρ (kg/m ³)	0,916
Merná tepelná kapacita, tuhý stav c (J/kg·K)	2384
Merná tepelná kapacita, kvapalný stav c (J/kg·K)	1250
Tepelná vodivosť, tuhý stav λ (W/m·K)	0,346
Tepelná vodivosť, kvapalný stav λ (W/m·K)	0,22
Teplota tuhnutia ϑ (°C)	33,6
Teplota topenia ϑ (°C)	64
Teplota varu ϑ (°C)	300
Latentné teplo z tuhého do kvapalného L (J/kg)	200000
Latentné teplo z kvapalného do plynného L (J/kg)	-

IV. VÝPOČET POTREBNÉHO TEPLA PRE AKUMULAČNÉ MÉDIUM VODA

Predpokladaný objem akumulčnej nádrže: $V_{\text{akumulčnej nádrže}} = 40 \text{ m}^3$

Energia potrebná z tuhého do kvapalného skupenstva

Pre určenie energie potrebnej na prechod medzi skupenstvom potrebujeme poznať hmotnosť, mernú tepelnú kapacitu v tuhom stave a rozdiel teplôt teploty topenia a teploty vnútorného priestoru, teda platí:

$$E_{T \rightarrow K} = m \cdot c_T \cdot (\vartheta_{\text{topenia}} - \vartheta_{\text{vnútorná}}) = -3678 \text{ MJ} \quad (3)$$

kde: m hmotnosť (súčin objemovej hustoty a objemu akumulčnej nádrže) [kg]

Energia skupenskej zmeny

Energia skupenskej zmeny je súčin hmotnosti a latentného tepla z tuhej látky na latentné teplo kvapalnej látky.

$$E_{SZ} = m \cdot L_{T \rightarrow K} = 13,36 \text{ GJ} \quad (6)$$

kde: E_{SZ} energia skupenskej zmeny [J]

Energia potrebná na ohriatie média z teploty topenia na požadovanú teplotu

Keď máme vypočítanú energiu skupenskej zmeny, tak potom si vypočítame energiu potrebnú na zohriatie média z teploty topenia na konečnú teplotu média. Na tento výpočet potrebujeme hmotnosť, mernú tepelnú kapacitu kvapaliny, rozdiel teplôt konečnej teploty média a teploty topenia. Teda platí:

$$E_{\text{ohriatia}} = m \cdot c_K \cdot (\vartheta_{\text{konečná}} - \vartheta_{\text{topenia}}) = 14,28 \text{ GJ} \quad (9)$$

kde: E_{ohriatia} energia potrebná na ohriatie média z teploty topenia na požadovanú teplotu [J]

Celková energia potrebná na ohriatie média z ϑ_1 na ϑ_2 , aj s koeficientom účinnosti akumulácie

Celková energia je súčet energií $E_{T \rightarrow K} + E_{SZ} + E_{ohriatia}$. Teda platí:

$$E_{\text{celková}} = (E_{T \rightarrow K} + E_{SZ} + E_{ohriatia}) \cdot \frac{\eta_{\text{akumulácie}}}{100\%} = 53,25 \text{ GJ} = 14,79 \text{ MWh} \quad (10)$$

kde: $E_{\text{celková}}$ celková energia potrebná na ohriatie média z ϑ_1 na ϑ_2 upravená koeficientom účinnosti 45% [J]

$\eta_{\text{akumulácie}}$ koeficient účinnosti akumulácie [%]

Energia na akumuláciu

Je to rozdiel energie (E_{SK}), ktorú nám dodávajú slnečné kolektory o ploche 50 m² a celkovej energii potrebnej na ohriatie média z ϑ_1 na ϑ_2 ($E_{\text{celková}}$). Teda platí:

$$E_{\text{akumulácie}} = E_{SK} - E_{\text{celková}} = 92,19 \text{ GJ} \quad (15)$$

kde: $E_{\text{akumulácie}}$ energia na akumuláciu [J]

E_{SK} energia slnečných kolektorov s koeficientom účinnosti 75% [J]

Prebytok energie na zabezpečenie vykurovania a ohrevu TÚV budovy z akumuláčného zdroja

Je to energia, ktorá zostala ako prebytok po rozdiel celkovej energie na ohriatie média z ϑ_1 na teplotu ϑ_2 , vrátane koeficientu akumulácie a celkovej energetickej náročnosti budovy ($E_{\text{budova}} = 52,66 \text{ GJ}$). Teda platí:

$$E_{\text{prebytok}} = E_{\text{celková}} - E_{\text{budova}} = 585,71 \text{ MJ} = 162,70 \text{ kWh} \quad (10)$$

kde: E_{prebytok} prebytok energie na zabezpečenie vykurovania a ohrevu TÚV z akumuláčného zdroja [J]

E_{budova} energetická náročnosť budovy na vykurovanie a ohrev teplej vody [J]

V. VÝPOČET PRE PARAFÍNOVÝ VOSK AKO AKUMULAČNÉ MÉDIUM

Predpokladaný objem akumuláčnej nádrže: $V_{\text{akumuláčnej nádrže}} = 40 \text{ m}^3$

Energia potrebná z tuhého do kvapalného skupenstva

$$E_{T \rightarrow K} = m \cdot c_T \cdot (\vartheta_{\text{topenia}} - \vartheta_{\text{vnútorná}}) = 3669 \text{ MJ} \quad (21)$$

kde: m hmotnosť (súčin objemovej hustoty a objemu akumuláčnej nádrže) [kg]

Energia skupenskej zmeny

$$E_{SZ} = m \cdot L_{T \rightarrow K} = 7328 \text{ MJ} \quad (6)$$

kde: E_{SZ} energia skupenskej zmeny [J]

Energia potrebná na ohriatie média z bodu topenia na požadovanú teplotu

$$E_{ohriatia} = m \cdot c_K \cdot (\vartheta_{\text{konečná}} - \vartheta_{\text{topenia}}) = 732,80 \text{ MJ} \quad (22)$$

kde: $E_{ohriatia}$ energia potrebná na ohriatie média z teploty topenia na požadovanú teplotu [J]

Celková energia potrebná na ohriatie média z ϑ_1 na ϑ_2 , aj s koeficientom účinnosti akumulácie

$$E_{\text{celková}} = (E_{T \rightarrow K} + E_{SZ} + E_{ohriatia}) \cdot \frac{\eta_{\text{akumulácie}}}{100\%} = 7,24 \text{ MWh} \quad (24)$$

kde: $E_{\text{celková}}$ celková energia potrebná na ohriatie média z ϑ_1 na ϑ_2 upravená koeficientom účinnosti 45% [J]

$\eta_{\text{akumulácie}}$ koeficient účinnosti akumulácie [%]

Energia na akumuláciu

$$E_{\text{akumulácie}} = E_{SK} - E_{\text{celková}} = 119,37 \text{ GJ} \quad (26)$$

kde: $E_{\text{akumulácie}}$ energia na akumuláciu [J]

E_{SK} energia slnečných kolektorov s koeficientom účinnosti 75% [J]

Prebytok energie na zabezpečenie vykurovania a ohrevu TÚV budovy z akumuláčného zdroja ($E_{\text{budova}} = 52,66 \text{ GJ}$)

$$E_{\text{prebytok}} = E_{\text{celková}} - E_{\text{budova}} = -7,39 \text{ MWh} \quad (28)$$

kde: E_{prebytok} prebytok energie na zabezpečenie vykurovania a ohrevu TÚV z akumuláčného zdroja [J]

E_{budova} energetická náročnosť budovy na vykurovanie a ohrev teplej vody [J]

Do sivých buniek zapíšte požadované hodnoty	
Vypočítané hodnoty sú písané červenou farbou	
V ružových bunkách sú importované hodnoty z iných buniek	
Akumulácia	
typ akumulačného média	Voda
objem	40 m ³
objemová hustota "ρ"	999,972 kg·m ⁻³
hmotnosť "m"	39998,88 kg
merná tepelná kapacita v tuhom stave "c"	4180 J·kg ⁻¹ ·K ⁻¹
merná tepelná kapacita v kvapalnom stave "c"	4200 J·kg ⁻¹ ·K ⁻¹
tepelná vodivosť v tuhom stave "λ"	2,3 W·m ⁻¹ ·K ⁻¹
tepelná vodivosť v kvapalnom stave "λ"	0,55312 W·m ⁻¹ ·K ⁻¹
teplota tuhnutia	-0,001 °C
teplota topenia	0 °C
teplota varu	100 °C
Latentné teplo (z tuhého stavu do kvapalného)	334000 J·kg ⁻¹
Latentné teplo (z kvapalného stavu do plynného)	2257000 J·kg ⁻¹
počiatočná teplota média	22 °C
konečná teplota média	85 °C
Potrebná energia z tuhého stavu do kvapalného stavu	-3678 MJ
Energia skupenskej zmeny (prechod z tuhého stavu do kvapalného)	13359,626 MJ
Energia potrebná na ohriatie média z bodu topenia na požadovanú teplotu	14279,600 MJ
Celková energia potrebná na ohriatie média z teploty 22 °C na teplotu 85 °C:	23,961 GJ
	6,66 MWh
Koeficient zohľadňujúci straty akumulácie	45 %
Celková energia potrebná na ohriatie média z teploty 22 °C na teplotu 85 °C, vrátane koeficientu strát:	53,247 GJ
	14,79 MWh
Energia na akumuláciu	92,193 GJ
	25,609 MWh
Vyhovuje dané množstvo média?	ÁNO
Domácnosť (spotreba)	
Energetická náročnosť budovy počas roka (vykurovanie)	12000 kWh
	43,200 GJ
Energetická náročnosť budovy počas roka (TUV)	2628 kWh
	9,461 GJ
Celková energetická náročnosť budovy počas roka	14628 kWh
	52,661 GJ
Prebytok energie na zabezpečenia vykurovania budovy z akumulačného zariadenia:	585,709 MJ
	162,70 kWh
Vyhovuje dané množstvo média?	ÁNO

Obr. 3 Výstup z MS Excel - akumulačné médium „voda“

Do sivých buniek zapíšte požadované hodnoty	
Vypočítané hodnoty sú písané červenou farbou	
V ružových bunkách sú importované hodnoty z iných buniek	
Akumulácia	
typ akumulačného média	Parafínový vosk
objem	40 m ³
objemová hustota "ρ"	916 kg·m ⁻³
hmotnosť "m"	36640 kg
merná tepelná kapacita v tuhom stave "c"	2384 J·kg ⁻¹ ·K ⁻¹
merná tepelná kapacita v kvapalnom stave "c"	1250 J·kg ⁻¹ ·K ⁻¹
tepelná vodivosť v tuhom stave "λ"	0,346 W·m ⁻¹ ·K ⁻¹
tepelná vodivosť v kvapalnom stave "λ"	0,22 W·m ⁻¹ ·K ⁻¹
teplota tuhnutia	33,6 °C
teplota topenia	64 °C
teplota varu	300 °C
Latentné teplo (z tuhého stavu do kvapalného)	200000 J·kg ⁻¹
Latentné teplo (z kvapalného stavu do plynného)	0 J·kg ⁻¹
počiatočná teplota média	22 °C
konečná teplota média	80 °C
Potrebná energia z tuhého stavu do kvapalného stavu	3669 MJ
Energia skupenskej zmeny (prechod z tuhého stavu do kvapalného)	7328,000 MJ
Energia potrebná na ohriatie média z bodu topenia na požadovanú teplotu	732,800 MJ
Celková energia potrebná na ohriatie média z teploty 22 °C na teplotu 80 °C:	11,729 GJ
	3,26 MWh
Koeficient zohľadňujúci straty akumulácie	45 %
Celková energia potrebná na ohriatie média z teploty 22 °C na teplotu 80 °C, vrátane koeficientu strát:	26,066 GJ
	7,24 MWh
Energia na akumuláciu	119,374 GJ
	33,159 MWh
Vyhovuje dané množstvo média?	ÁNO
Domácnosť (spotreba)	
Energetická náročnosť budovy počas roka (vykurovanie)	12000 kWh
	43,200 GJ
Energetická náročnosť budovy počas roka (TUV)	2628 kWh
	9,461 GJ
Celková energetická náročnosť budovy počas roka	14628 kWh
	52,661 GJ
Prebytok energie na zabezpečenia vykurovania budovy z akumulačného zariadenia:	-26595,267 MJ
	-7387,57 kWh
Vyhovuje dané množstvo média?	NIE

Obr. 4 Výstup z MS Excel - akumulačné médium „parafínový vosk“

VI. ZÁVER

Cieľom tohto článku bolo priblížiť možnosť akumulácie tepelnej energie do akumulčných materiálov. Na výpočet tepelných bilancií bol zostavený výpočtový program v MS Excel.

Tepelná energia bola v tomto prípade získavaná zo slnečných kolektorov a akumulovaná do akumulčného média v akumulčnej nádrži. Energetická náročnosť budovy sa skladá z energie na vykurovanie (12000 kWh) a energie na ohrev teplej úžitkovej vody (2628 kWh). Energetická náročnosť na ohrev teplej úžitkovej vody zodpovedá spotrebe 4 osobám v domácnosti.

Porovnali sme vlastnosti dvoch akumulčných médií, vodu o objeme 40 m³ a parafínový vosk, taktiež o objeme 40 m³. Našou úlohou bolo zistiť či obidva tieto akumulčné médiá budú postačovať na pokrytie tepelnej spotreby domácnosti.

Na Obr. 3 je možné vidieť výstup z výpočtového programu pre akumulčné médium voda. Pre toto akumulčné médium bolo zvolené množstvo akumulčného média o objeme 40 m³. Toto médium o danom objeme postačovalo na pokrytie spotreby tepla v domácnosti. Dane množstvo vody je pre naše potreby predimenzované o 585,71 MJ.

Na Obr. 4 je možné vidieť výstup z výpočtového programu pre akumulčné médium parafínový vosk. Parafínový vosk o objeme 40 m³ nepostačuje svojou akumulčnou schopnosťou na pokrytie celkovej tepelnej spotreby domácnosti. Zistili sme, že dané množstvo tohto akumulčného média by postačovalo počas roka len na ohrev teplej úžitkovej vody. Ak by sme chceli týmto médium pokryť tepelnú spotrebu domácnosti, museli by sme tento zdroj zväčšiť minimálne o energiu 27 GJ.

POĎAKOVANIE

Tento príspevok vznikol vďaka podpore Vedeckej grantovej agentúry Ministerstva školstva, vedy, výskumu a športu SR a SAV podporenej grantom VEGA 1/0372/18.

LITERATÚRA

- [1] Gašparík, M. „Akumulácia tepelnej energie v solárnych systémoch“. [online] [citované: 4.11.2019]. Dostupné na internete: < <https://www.asb.sk/stavebnictvo/technicke-zariadenia-budov/energie/akumulacia-tepelnej-energie-v-solarnych-systemoch> >
- [2] Ukrainczyk, N., Kurajica, S.; Sipusic, J., „*Thermophysical Comparison of Five Commercial Paraffin Waxes as Latent Heat Storage Materials. Chemical & Biochemical Engineering Quarterly*“, [online] [citované: 15.3.2020]. Dostupné na internete: < https://www.researchgate.net/publication/44858087_Thermophysical_Comparison_of_Five_Commercial_Paraffin_Waxes_as_Latent_Heat_Storage_Materials >
- [3] „Voda“, [online] [citované: 10.3.2020]. Dostupné na internete: < <https://sk.wikipedia.org/wiki/Voda> >
- [4] Jurenka, V., „*Možnosti akumulácie tepelnej energie*“. Bakalárska práca. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2020.

Porovnanie vplyvu defuzzifikačnej metódy na priebeh regulácie synchronného motora

¹Peter Girovský

¹ Katedra elektrotechniky a mechatroniky, Fakulta elektrotechniky a informatiky, Technická univerzita, Letná 9, 04200 Košice, Slovenská Republika

¹peter.girovsky@tuke.sk

Abstrakt — Tento príspevok sa zaoberá porovnaním vplyvu rôznych defuzzifikačných metód na priebeh regulácie synchronného motora s permanentnými magnetmi (PMSM). V práci sú porovnané defuzzifikačné metódy – Centroid, Bisector, Middle, Smallest, and Largest of Maximum V rámci práce bol zostavený matematický model synchronného motora s permanentnými magnetmi a navrhnuté riadenie pomocou fuzzy logiky. Návrh a simulácia fuzzy riadenia boli robené v programe MATLAB/Simulink.

Keľúčové slová — synchronný motor, fuzzy logika, riadenie

I. ÚVOD

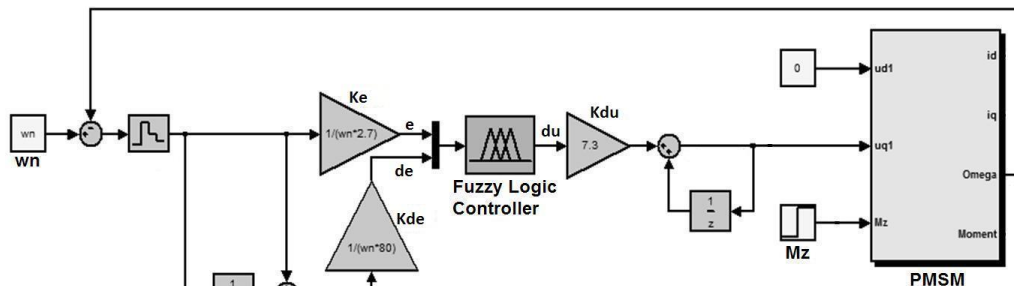
Využívanie fuzzy logiky nachádza čoraz väčšie uplatnenie pri riadení procesov v priemysle. Prednosťou týchto metód je predovšetkým ich jednoduchosť pri tvorbe modelov zložitých nelineárnych procesov a možnosť využitia skúseností aj vedomostí pri návrhu metód riadenia týchto procesov. Fuzzy logika predstavuje alternatívu riešenia pre riadenie, lebo je jednoduchšia, rýchlejšia a zlepšuje kvalitu riadenia všade tam, kde vieme opísať skúsenosti prostredníctvom pravidiel, pre vytvorenie fuzzy systému. Fuzzy logika je metóda založená na pravidlách rozhodovania. Používa sa aj v expertných systémoch a hlavne pri riadení procesov. Pomocou nej môžeme riadiť veľkú časť procesov, pri ktorých nepoznáme presný matematický popis.

Tento článok vychádza z publikovanej štúdie [6], v ktorej sa vyšetroval vplyv rozmiestnenia funkcií príslušností pre fuzzy riadenie synchronného motora s permanentnými magnetmi. Defuzzifikácia vo fuzzy regulátore určuje podľa vstupu a tabuľky pravidiel rovno regulačné zásahy, preto vplyv použitej defuzzifikačnej metódy môže mať veľký vplyv na priebeh regulovanej veličiny.

II. FUZZY REGULÁCIA SYNCHRÓNNEHO MOTORA S PERMANENTNÝM MAGNETOM

A. Regulácia sústavy s PI fuzzy regulátorom

Pre návrh riadenia bol použitý model synchronného motora s permanentnými magnetmi v súradnicovom systéme d,q. Navrhli sme regulačnú sústavu s fuzzy PI regulátorom, ktorý má priradenú FIS maticu, v ktorej sú definované vlastnosti regulátora. Použitá regulačná štruktúra s navrhnutým fuzzy PI regulátorom je zobrazená na obr.1.



Obr. 1 Regulačný obvod PMSM s fuzzy PI regulátorom

Regulačný obvod a priebeh prechodovej charakteristiky je ovplyvnený nastavením periódy vzorkovania, ale aj jednotlivými zosilneniami na vstupe (K_e , K_{de}) a výstupe (K_{du}), kde e predstavuje regulačnú odchýlku, de zmenu regulačnej odchýlky a výstupná premenná du predstavuje akčný zásah. Veľkosť vzorkovacej periódy bola nastavená na hodnotu 0,01.

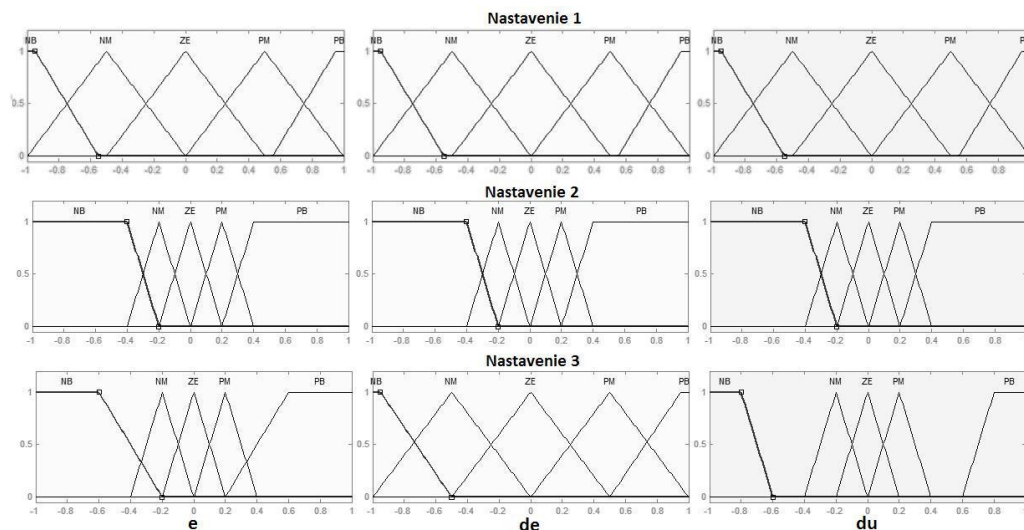
Pre funkcie príslušnosti bol vybraný trojuholníkový tvar a pre okrajové funkcie bol zvolený tvar lichobežníkový. Pre dosiahnutie žiadaných priebehov regulovanej veličiny sme nastavili zosilnenia na vstupe e , de a výstupe du nasledovne: $K_e = 1/(\omega_n * 2.7)$, $K_{de} = 1/(\omega_n * 80)$, $K_{du} = 7.3$. Tieto nastavenie zabezpečili požadovaný priebeh regulovanej veličiny.

III. VPLYV DEFUZZIFIKAČNEJ METÓDY

Ako už bolo spomenuté vyššie, použitá defuzzifikačná metóda ovplyvňuje regulačný zásah a tým aj priebeh regulovanej veličiny. Pre vyšetrenie vplyvu defuzzifikačnej metódy bola použitá FIS matica s piatimi funkciami príslušnosti zobrazenými na obr. 2.

Tabuľka 1
 Báza pravidiel pre päť funkcií príslušnosti

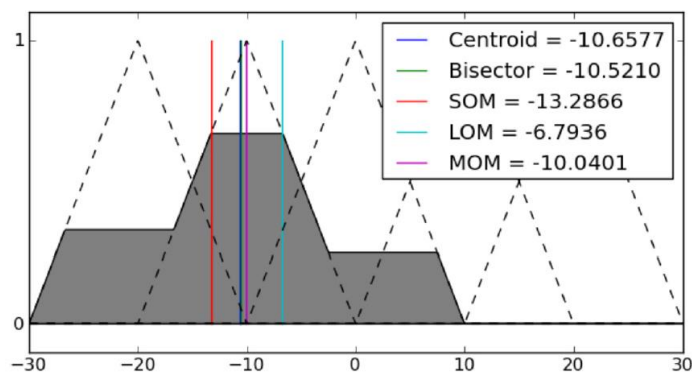
		<i>de</i>				
		NB	NM	ZE	PM	PB
<i>e</i>	NB	NB	NB	NM	NM	ZE
	NM	NB	NM	NM	ZE	PM
	ZE	NM	NM	ZE	PM	PM
	PM	NM	ZE	PM	PM	PB
	PB	ZE	PM	PM	PB	PB



Obr. 2 FIS matice pre jednotlivé nastavenia pre päť funkcií príslušnosti.

V článku je porovnávaný vplyv týchto defuzzifikačných metód:

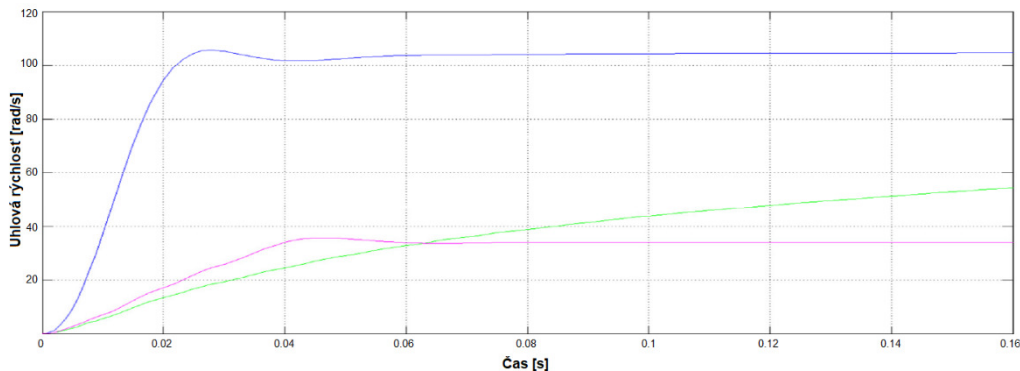
- Centroid defuzzifikačná metóda vracia stred plochy pod krivkou.
- Bisector je čiara, ktorá rozdelí celú oblasť do dvoch podoblastí rovnakých veľkosti. Niekedy, ale nie vždy je totožná s ťažiskom plochy.
- Middle, Smallest, and Largest of Maximum, (MOM, SOM, a LOM) Stredné, najmenšie a najväčšie maximum. Tieto tri metódy určujú maximálnu hodnotu, ktorú zaberá celková funkcia príslušnosti.



Obr. 3 Určenie ostrej hodnoty pri rôznych defuzzifikačných metódach

A. Vplyv defuzzifikačnej metódy pri inferenčnej metóde Min – Max

Na obr. 4 sú zobrazené priebehy regulovanej veličiny SMPM s fuzzy regulátorom pre počiatkové nastavenia hodnôt K_e , K_{de} a K_{du} , ktoré sú uvedené v odseku 2 tohto článku. Z uvedených výsledkov simulácie je vidno, že priebeh uhlovej rýchlosti bol vyhovujúci iba pri použitej defuzzifikačnej metóde Centroid.

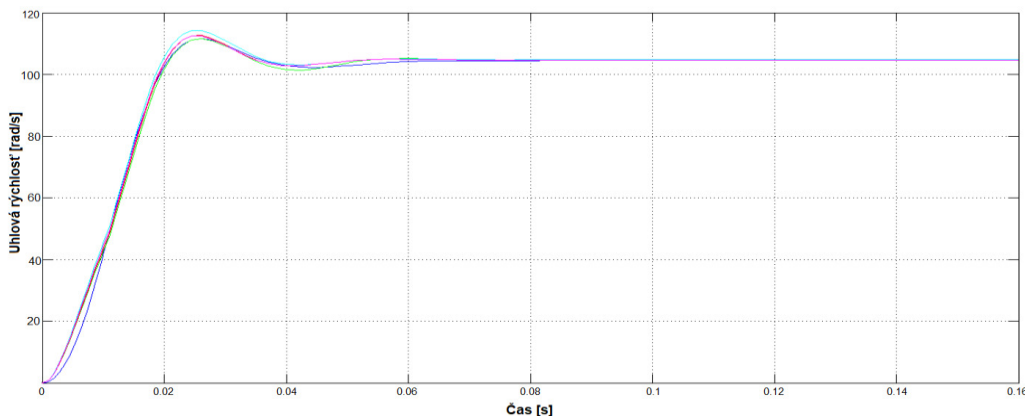


Obr. 4 Priebeh uhlovej rýchlosti SMPM pri počiatkových hodnotách K_e , K_{de} , K_{du}

Prenastavením váh pre jednotlivé defuzzifikačné metódy môžeme dosiahnuť ideálne priebehy uhlovej rýchlosti SMPM. Tieto hodnoty sa nachádzajú v tabuľke 2 a priebeh regulovanej veličiny na obr. 5.

Tabuľka 2
 Nastavenia váh pri inferenčnej metóde Min – Max

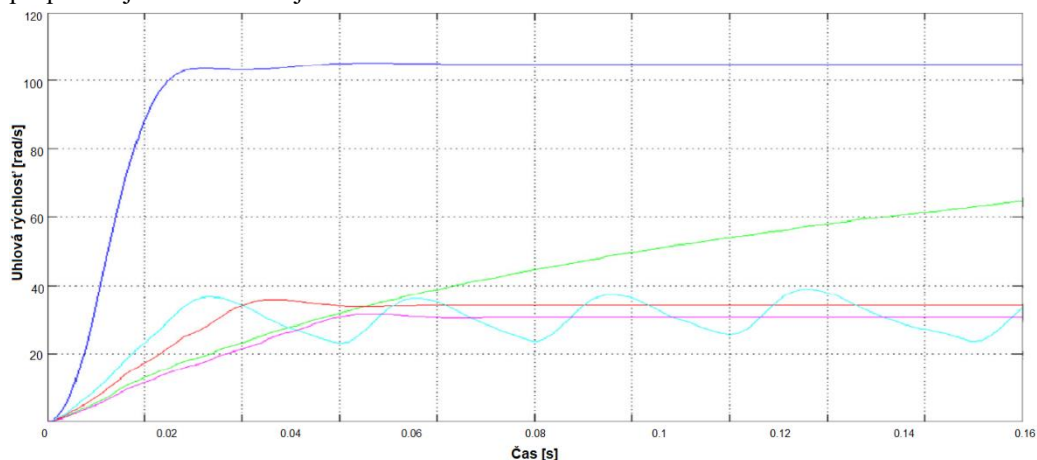
Defuzzifikácia	K_e	K_{de}	K_{du}
Centroid	$1/(w_n * 2)$	$1/(w_n * 30)$	5,2
Bisector	$1/(w_n * 1)$	$1/(w_n * 500)$	47,23
MOM	$1/(w_n * 1)$	$1/(w_n * 20)$	48,15
LOM	$1/(w_n * 1.5)$	$1/(w_n * 50)$	38,24
SOM	$1/(w_n * 1)$	$1/(w_n * 20)$	53,5



Obr. 5 Priebeh uhlovej rýchlosti SMPM po zmene nastavenia váh pri jednotlivých defuzzifikačných metódach

B. Vplyv defuzzifikačnej metóde pri inferenčnej metóde Prod – Max

Na obr. 6 sú zobrazené priebehy regulovanej veličiny SMPM s fuzzy regulátorom pre počiatočné nastavenia hodnôt K_e , K_{de} a K_{du} , ktoré sú uvedené v odseku 2 tohto článku. Z uvedených výsledkov simulácie je vidno, že priebeh uhlovej rýchlosti bol vyhovujúci opäť iba pri použitej defuzzifikačnej metóde Centroid.

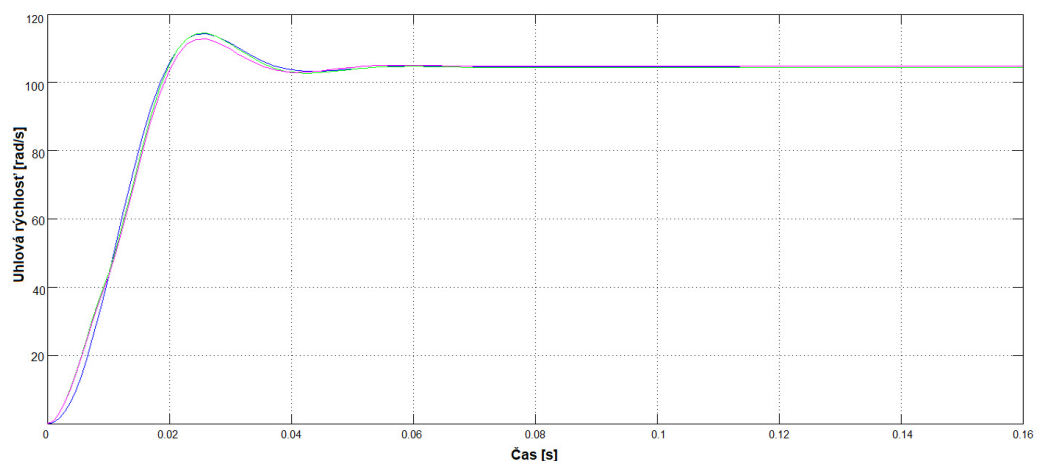


Obr. 6 Priebeh uhlovej rýchlosti SMPM pri počiatočných hodnotách K_e , K_{de} , K_{du}

Prenastavením váh pre jednotlivé defuzzifikačné metódy môžeme dosiahnuť ideálne priebehy uhlovej rýchlosti SMPM. Tieto hodnoty sa nachádzajú v tabuľke 3 a priebeh regulovanej veličiny na obr. 7.

Tabuľka 3
 Nastavenia váh pri inferenčnej metóde Prod - Max

Defuzzifikácia	K_e	K_{de}	K_u
Centroid	$1/(w_n * 2)$	$1/(w_n * 11)$	5,4
Bisector	$1/(w_n * 1)$	$1/(w_n * 50)$	49
MOM	$1/(w_n * 1)$	$1/(w_n * 20)$	48,2
LOM	$1/(w_n * 1)$	$1/(w_n * 20)$	48,2
SOM	$1/(w_n * 1)$	$1/(w_n * 20)$	48,2



Obr. 7 Priebeh uhlovej rýchlosti SMPM po zmene nastavenia váh pri jednotlivých defuzzifikačných metódach

IV. ZÁVER

Z výsledkov simulácii jednotlivých regulačných obvodov vyplýva, že vplyv zmeny defuzzifikačnej metódy pri inferenčných metódach Min – Max a Prod – Max, je veľký, ale len vtedy ak váhy jednotlivých regulačných obvodov sú nastavené na rovnaké hodnoty pri všetkých defuzzifikačných metódach. Ak však prestavíme hodnoty váh pri jednotlivých defuzzifikáciách, môžeme dosiahnuť vhodné priebehy regulovanej veličiny. Z toho vyplýva, že použitá metóda defuzzifikácie ovplyvňuje žiadanú hodnotu na výstupe, ale vhodným nastavením váh, pri ľubovoľnej defuzzifikačnej metóde, sa dá dosiahnuť požadovaný priebeh prechodovej charakteristiky uhlovej rýchlosti.

POĎAKOVANIE

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-15-0750.

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-16-0206.

Táto práca vznikla s podporou grantu VEGA 1/0187/18.

LITERATÚRA

- [1] P. BRANDŠTETTER: Střídavé regulační pohony: moderní způsoby řízení. VŠB TU, Ostrava, 1999.
- [2] V. NOVÁK: Základy fuzzy modelování, 175s., ISBN 80-7300-009-1, 2000 Praha, vydalo nakladatelství BEN
- [3] V. MAŘÍK, O. ŠTEPÁNKOVÁ, J. LAŽANSKÝ: Umělá inteligence (4), ISBN 80-200-1044-0, 2003 Praha, vydala ACADEMIA
- [4] Vilém Novák, Základy fuzzy modelování, BEN, 2000 Praha, ISBN 80-7300-009-1
- [5] Defuzzification Methods – MATLAB & Simulink Example, Try MATLAB, Simulink, and Other Products, [online]. Dostupné na internete: <<http://www.mathworks.com/help/fuzzy/examples/defuzzification-methods.html>>
- [6] P. GIROVSKÝ, J. ŽILKOVÁ: Fuzzy riadenie synchronného motora, In: Electrical Engineering and Informatics 6 : proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice. - Košice : FEI TU, 2015 S. 57-61. - ISBN 978-80-553-2178-3

Paralelization of two-dimensional models of cosmic rays distribution in heliosphere

¹Michal SOLANIK, ²Pavol BOBÍK, ³Ján GENČI

¹Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

²Department of Cosmic Physics, Institute of Experimental Physics SAV Košice, Slovak Republic

³Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

¹michal.solanik@student.tuke.sk, ²bobik@saske.sk, ³jan.genci@tuke.sk

Abstract – The goal of the previous bachelor thesis was to parallelize 1D models of cosmic rays distribution in the heliosphere. During testing 1D F-p model we discovered an error that we've named pulsation. We suggested three hypotheses that probably could cause pulsation, but none of them was confirmed. After analysis of the range of input parameters, we found out that pulsation is present in 1D F-p simulation with step length less than 2.0 s. Successful parallelization of 1D heliosphere models led us to the option to parallelize 2D B-p, 2D F-p, and 2D F-T models on GPU. Despite higher consumption of memory even higher speed up was achieved than in the case of 1D models with acceptable accuracy. GPU implementation of the 2D F-T model achieved 9.83 speed up against the reference multiprocessor system. The average speedup for the 2D F-p model was 11.69 against the reference multiprocessor system. GPU implementation of the 2D B-p model speed up ranged from 82.29 to 473.46 against the reference multiprocessor system. Positive results in terms of speed up and acceptable accuracy led us to design and implement a distributed system. Design and implementation took into account that every single node can have different GPUs. It was reflected in the distribution of simulation iterations for every node of the distributed system. On a distributed system with two different GPUs was achieved speed up that ranged from 24.04% to 34.62% against single reference GPU.

Keywords – CUDA, GPGPU, Heliosphere, Distributed systems

I. INTRODUCTION

Demand after computing power is growing with the complexity of models in physics. In the bachelor thesis, we proved [1] that the parallelization of 1D models of cosmic rays distribution in the heliosphere was successful. Achieved speedup in the case of 1D F-p model was 7.87 against the reference multiprocessor system and in case of 1D B-p model was at least 86.87 against the same reference system. This speedup with acceptable accuracy results led us to parallelization of 2D models of cosmic rays distribution in the heliosphere on GPU with CUDA framework. Later we found out that a distributed system, which uses GPUs, is also an acceptable option for parallelization.

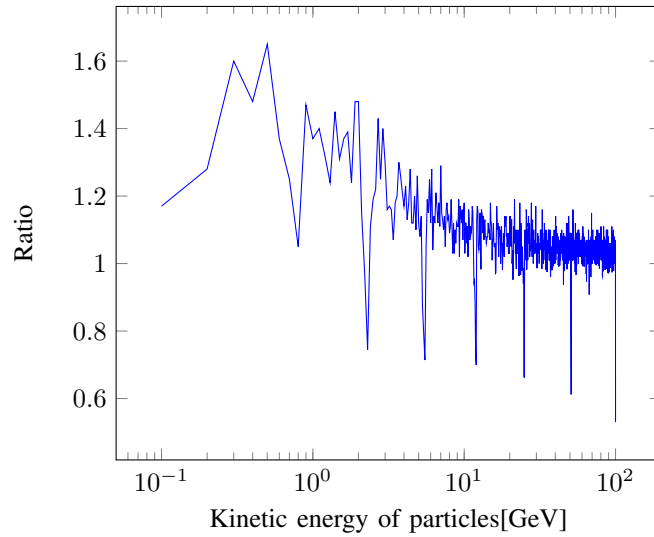
II. METHODS AND MODELS

A. 1D F-p pulsations

During the testing of GPU implementation of the 1D F-p model, we found an error that we named pulsations. An example of pulsations with step length $dt = 0.5$ s can be found in figure 1. We suggested four hypotheses that can cause pulsations:

- problems that are linked to distribution in RNG implementation, testing RNG generators suggested [2] that XORWOW RNG distribution can deviate,
- differences between single and double precision,
- usage of optimization options for compilation,
- incorrect distribution of simulations on GPU.

Fig. 1: Energetic spectrum for 200 billions of particles - F-p model - $dt = 0.5$



B. Parallelization of 2D heliospheric models

Success parallelization of 1D heliospheric models on GPU with the usage of CUDA framework led us into the idea, that 2D heliospheric simulations can be parallelized in the same way. The structure of forward or backward simulation is very similar. It can be divided into:

- pre-simulation, where injection parameters are calculated,
- main simulation, where the main part of the calculation is located,
- post-simulation, where some parameters can be calculated. Intensity w is calculated in post-simulation for backward models. This part is however not used in forward type simulations.

GPU implementation will add additional parts that are depicted with red color in figure 2. GPU pre-simulation consists of memory allocation on GPU and in host memory, initialization structures for RNG. Because simulations themselves are calculated on GPU, results have to be copied from GPU memory to RAM.

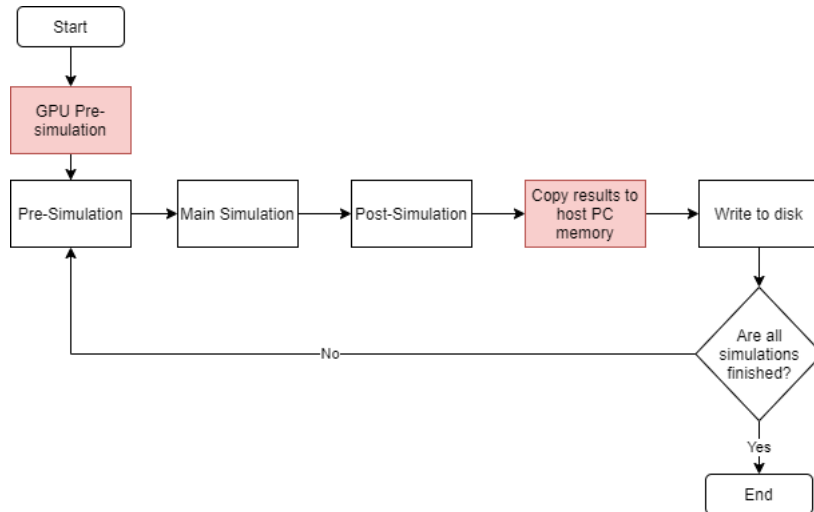


Fig. 2: Flowchart of heliospheric simulation for CUDA implementation

GPU implementation is mostly limited to memory usage. A number of used registers determines [3] effective usage of GPU. To achieve that we have to refactor the structure of CPU code. This will help compilation to optimize memory usage.

C. Distributed system

A distributed system that uses GPU can achieve even better speedup. We proposed a distributed system that is depicted in figure 3. It consists of several modules:

- UI, that allows user to launch new simulations and display results,
- primary node of the distributed system, that provides REST API for UI, control over launching simulations, fail tolerance for failed simulations and parts of simulations, that failed on nodes of the distributed system,
- message broker, that uses AMQP for communication. Message oriented middleware is [4] one from the basic patterns for communications. AMQP provides persistent queues that allow implementing fault tolerance,
- node of the distributed system, that control calculation on GPU.

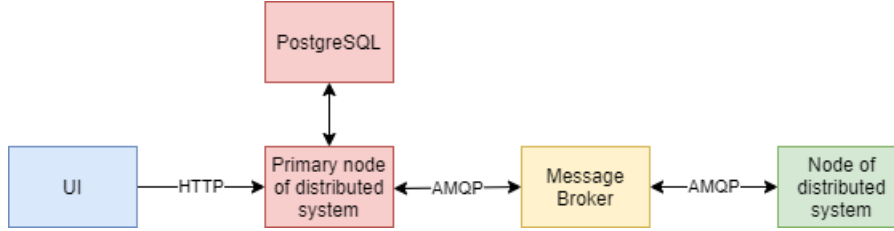


Fig. 3: Flowchart of heliospheric simulation for CUDA implementation

The distributed system won't have registered nodes with only one type of GPU. We decided to use equation 1. This represents the performance of the given GPU. With the calculated performance of every available node in the distributed system, we can proportionally divide simulation between them.

$$Performance = GPU_TFLOPS * GPU_Usage * GPU_Memory_Bandwidth \quad (1)$$

Fault tolerance is an important part of the distributed system. A user of the distributed system should not feel [4] the failure of any node of the distributed system. We decided to use heartbeats from the side of the nodes. Every 2 minutes node of the distributed system will send a message with availability to the primary node. The primary node will check every 15 minutes for inactive nodes. Active simulations from inactive nodes will be distributed to another available node.

III. RESULTS

In this section we will discuss results for 1D F-p pulsations, accuracy and speedup of 2D heliospheric models. Also we will discuss speedup and fault tolerance of the distributed system.

A. 1D F-p pulsations

None of the suggested hypotheses in section II-A have been confirmed. In the case of different random generator implementation and different simulation distribution, there were no changes in the energetic spectrum. In figure 4 is present an energetic spectrum for GPU implementation of 1D F-p model for 200 billions of simulations compiled without optimization parameter – *use_fast_math*. Comparing it against spectrum with step length $dt = 0.5$ noise from 0.1 GeV to 10 GeV has been reduced and pulsation is more visible.

We tested GPU implementation 1D F-p with several step length dt . We discovered that for dt greater than 2.0 pulsation disappeared.

B. 2D heliospheric models

2D heliospheric models were more complex than 1D heliospheric models. 2D forward models produced much more data than 1D F-p model. With solar wind velocity $V = 400$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s output file was 196 times larger in case of 2D F-T model without drift and 211.38 times larger in case of 2D F-p model against 1D F-p model. That means we had enough data to exclude low statistics errors on the whole spectrum. All models were implemented without drift.

2D F-T model without drift

Measurement of speedup was realized with input parameters of solar wind velocity $V = 400$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and K_{per} / K_{par} ratio = 0.01 for 100 billions of simulations. On reference multiprocessor system execution time was 150 hours. On the other

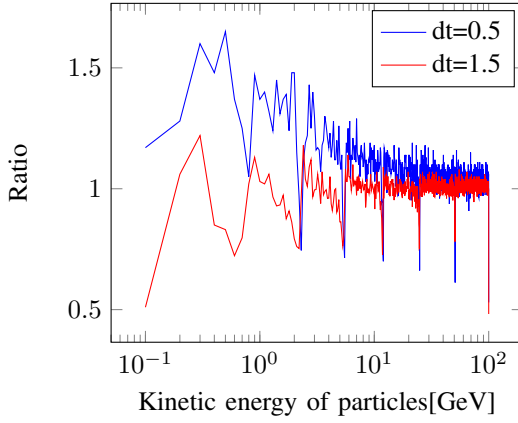


Fig. 4: Energetic spectrum for 200 billions of particles
- F-p model - $dt = 1.5$ without usage of `-use_fast_math`

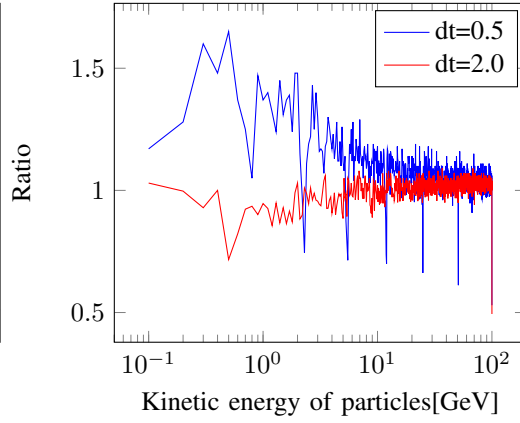


Fig. 5: Energetic spectrum for 200 billions of particles
- F-p model - $dt = 2.0$

hand, execution time on GTX 1080 TI was 15.25 hours. We achieved 9.83x speedup against the reference multiprocessor system.

Accuracy was kept on acceptable level. In figure 6 is present ratio of 2D F-T and 2D B-p energetic specters for 100 billions of particle simulations for solar wind velocity $V = 400$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01. For $\mu = 0.0$, equator region, can be observed most significant difference 16% on 0.8 GeV. For $\mu = 0.95$, polar region, difference decreases under 15% on 6.7 GeV. The same shape of the curve was observed from CPU implementation too. In figure 7 is present ratio of 2D F-T and 2D B-p energetic specters for 25 billions of particle simulations for solar wind velocity $V = 700$ km/h, step length $dt = 0.5$, $K_0 = 1 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01. This ratio of spectrums represents maximal modulation. There are visible pulsations for short step length dt .

2D F-p model without drift

Measurement of speedup was realized with input parameters of solar wind velocity $V = 400$ km/h, step length $dt = 2.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01 for 100 billions of simulations. We used step length $dt = 2.0$ because noticed [5] error with a growing spectrum on high energies. On reference multiprocessor system execution time was 150 hours. On the other hand, execution time on GTX 1080 TI was 12.82 hours. We achieved 11.69x speedup against reference multiprocessor system.

Accuracy was kept on acceptable level. In figure 8 is present ratio of 2D F-p and 2D B-p energetic specters for 100 billions of particle simulations for solar wind velocity $V = 400$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01. For $\mu = 0.0$, equator region, can be observed differences over 10% from 0.1 GeV to 3.3 GeV. For $\mu = 0.95$, polar region, difference decreases under 2% from 10 GeV. In figure 9 is present ratio of 2D F-p and 2D B-p energetic specters for 25 billions of particle simulations for solar wind velocity $V = 700$ km/h, step length $dt = 0.5$, $K_0 = 1 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01. As for 1D F-p and 2D F-T model pulsations are visible in this ratio. Pulsations are related to short step length in forward type simulations on GPU.

2D B-p model without drift

Like for 2D F-T and 2D F-p model, measurement of speedup was realized with input parameters of solar wind velocity $V = 400$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01 for 167.7 millions of simulations. Additional parameter is μ , we used $\mu = 0.0$, equator region, and $\mu = 0.95$, polar region, for simulations. On the reference multiprocessor system execution time for $\mu = 0.0$ was 62.88 hours. The execution time of GPU implementation 2D B-p model on GTX 1080 TI was 13.62 minutes. This means 276.82x speedup against reference multiprocessor system. For $\mu = 0.95$ execution time on the reference multiprocessor system was 27.95 hours while on GTX 1080 TI it was 13.74 minutes. This represents 121.99x speedup.

Big differences between speedup led us to carry out another test. We used input parameters solar wind velocity $V = 700$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01 for 167.7 millions of simulations. For $\mu = 0.0$ multiprocessor reference system execution time was 74.10 hours. Execution time for GTX 1080 TI was 9.39 minutes, which is 473.46x speedup against the reference multiprocessor system. In the case of $\mu = 0.95$ it is 82.29x speedup.

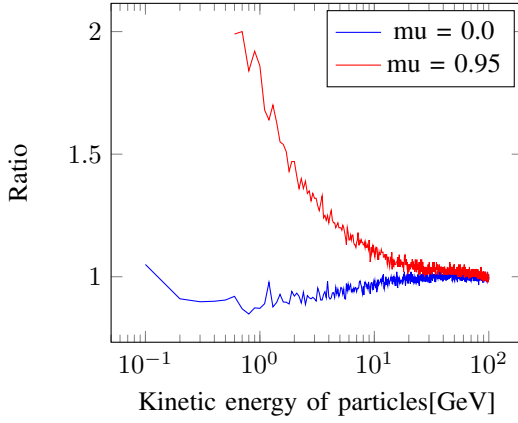


Fig. 6: Ratio of energetic spectra of 2D F-T and 2D B-p for 100 billions of particles - $V = 400$ km/h, $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01

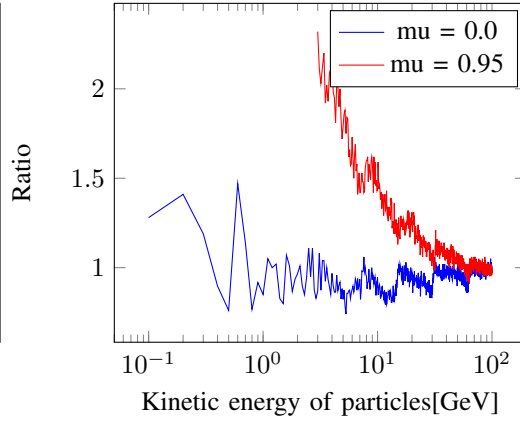


Fig. 7: Ratio of energetic spectra of 2D F-T and 2D B-p for 25 billions of particles - $V = 700$ km/h, $dt = 0.5$, $K_0 = 1 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01

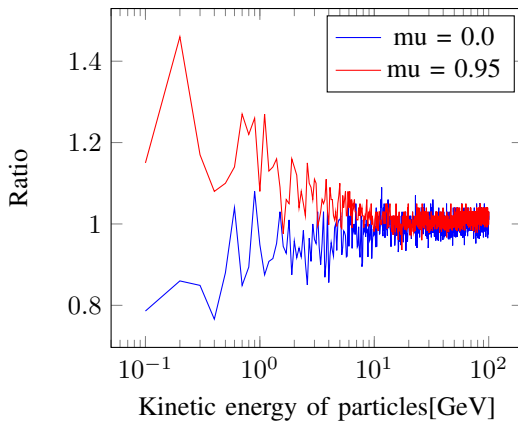


Fig. 8: Ratio of energetic spectra of 2D F-p and 2D B-p for 100 billions of particles - $V = 400$ km/h, $dt = 2.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01

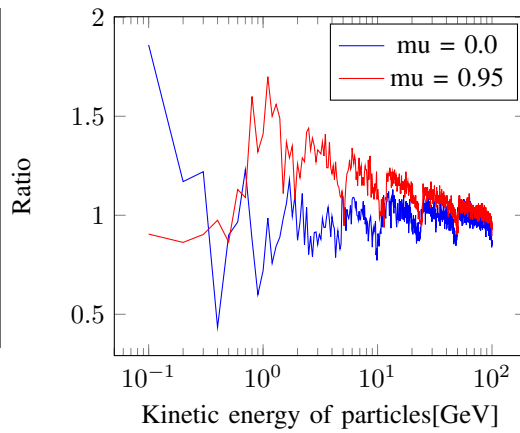


Fig. 9: Ratio of energetic spectra of 2D F-p and 2D B-p for 25 billions of particles - $V = 700$ km/h, $dt = 0.5$, $K_0 = 1 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01

GPU implementation of 2D B-p accuracy was kept [1] on the level with an accuracy of GPU implementation of 1D B-p model. In figure 10 is present ratio of GPU and CPU implementation of 2D B-p model with input parameters of solar wind velocity $V = 400$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01 for 167.7 millions. The maximal difference is 14%, but we suggest that difference is caused by low statistics in the CPU spectrum. In figure 11 is present ratio of GPU and CPU implementation of 2D B-p model with input parameters of solar wind velocity $V = 700$ km/h, step length $dt = 5.0$, $K_0 = 5 * 10^{18}$ m²/s and $K_{per} K_{par}$ ratio = 0.01 for 167.7 millions. The maximal difference is 9%, but we suggest that difference is caused by low statistics in the CPU spectrum too.

C. Distributed system

We used a distributed system with two nodes, one with GTX 1080 TI and second with RTX 2060. In table 1 are present execution time for each type of tested models. In the case of 1D F-p model, we gained about 26.89% speedup against GTX 1080 TI itself. Execution time between GTX 1080 TI and RTX 2060 in distributed system differ in 6 minutes, which is acceptable.

For 2D F-p model we used the same parameters as for 1D F-p model. We gained a smaller speedup against GTX 1080 TI itself, only about 24.04%.

1D B-p model was a different case, we had to change the distribution of simulations. The best option was evenly distributed simulation for each node. We gained 34.62% speedup against GTX 1080 TI itself.

The test of fault tolerance was successful. We used three nodes with GTX 1080 TI, RTX 2060 and GTX 1660. Node with GTX 1660 failed with notifying the primary node about failure. Node with RTX 2060 failed without notifying the primary node. Both failures were recorded and redistributed after to node with GTX 1080 TI after the successful execution of its part of the

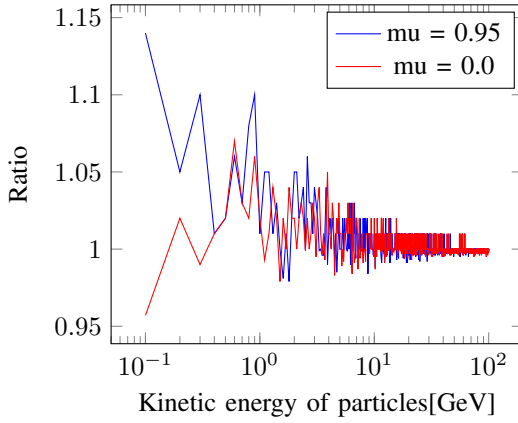


Fig. 10: Ratio of energetic spectra of GPU and CPU implementation of 2D B-p for 167.7 millions of particles - $V = 400$ km/h, $dt = 5.0$, $K_0 = 5 \cdot 10^{18}$ m²/s and K_{per} K_{par} ratio = 0.01

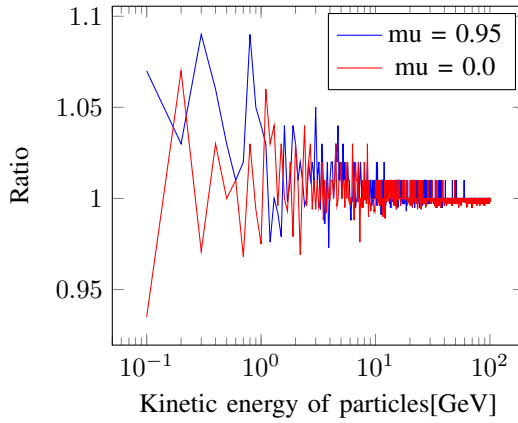


Fig. 11: Ratio of energetic spectra of GPU and CPU implementation of 2D B-p for 167.7 millions of particles - $V = 700$ km/h, $dt = 5.0$, $K_0 = 5 \cdot 10^{18}$ m²/s and K_{per} K_{par} ratio = 0.01

Table 1: Execution times for each model tested in distributed system

System type	1D F-p [hours]	2D F-p [hours]	1D B-p [hours]
Reference multiprocessor system	43.30	150.00	145.57
GTX 1080 TI	5.61	12.82	1.66
Distributed system	4.28	10.50	1.17

simulation. Execution time was 15% longer against GTX 1080 TI itself, but that is a cost for launching two parts of simulation from failed nodes.

IV. CONCLUSION

In the presented paper we focused on three areas - the analysis of the error of the so-called pulsation in the spectra of GPU implementation of 1D F-p model, 2D parallelization F-T, 2D F-p and 2D B-p models and a distributed system that allows the simulation to be distributed to multiple devices.

Suggested hypotheses about pulsation were not proved. After removal of `-use_fast_math` we observed the only disappearance of noise on low energies. After the analysis of input parameters, we found that pulsations are present for step length dt lower than 2.0 for every forward type model that was parallelized on GPU.

GPU implementation of every two-dimensional heliospheric model proved speedup is even greater than in the case of 1D heliospheric models. GPU implementation of 2D F-T gained 9.83x speedup against the reference multiprocessor system. However, large differences in the polar region between 2D F-T and 2D B-p was observed in CPU implementation too. GPU implementation of 2D F-p model gained 11.69x speedup against the reference multiprocessor system. In the case of GPU implementation of 2D B-p model speedup varied from 82.29x to 473.46x.

We used computer systems with GTX 1080 TI and RTX 2060 for testing speedup and the correctness of the distributed system. Achieved speedup varied from 24.04% to 34.62% against execution time on GTX 1080 TI itself. Fault tolerance also proved to be enough for finishing the simulation after 2 from 3 available nodes failed. Launching simulations is implemented via a web interface.

Parallelized heliospheric models on GPU and the distributed system will be used by students of physics on UPJŠ, employees of OKF ÚEF SAV.

REFERENCES

- [1] M. Solanik, *Prepis modelov distribúcie kozmického žiarenia v heliosfére do CUDA jazyka*, 2018.
- [2] Testing curand library distribution. [Online]. Available: <https://docs.nvidia.com/cuda/curand/testing.html#testing>
- [3] Cuda c programming guide. [Online]. Available: <https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html>
- [4] A. S. Tanenbaum and M. Van Steen, *Distributed systems: principles and paradigms*. Prentice-Hall, 2007.
- [5] C. Pei, J. Bieber, R. Burger, and J. Clem, "A general time-dependent stochastic method for solving parker's transport equation in spherical coordinates," *Journal of Geophysical Research: Space Physics*, vol. 115, no. A12, 2010.

The Impact of Electric Vehicles on Voltage Symmetry

¹Maksym Oliinyk, ²Jaroslav Džmura

¹Department of Electric Power Engineering Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

²Department of Electric Power Engineering Faculty of Electrical Engineering and Informatics Technical University of Košice, Slovak Republic

¹maksym.oliinyk@tuke.sk, ²jaroslav.dzmura@tuke.sk

Abstract — Voltage asymmetry is one of the indicators by which the quality of electric energy in three-phase electric networks with a voltage of 0.4 kV is evaluated. According to these standards, the voltage asymmetry coefficient in the zero sequence should not exceed 2%. With the proliferation of electric vehicles, the load increases significantly, which is a variable, depending on the time of day and season, due to which voltage asymmetry may occur during operation. In this paper, we simulate a real network with a gradual increase in the number of connected electric vehicles, as well as a possible solution to the problem of asymmetry.

Keywords — Smart Grid, computer simulation, quality of electric energy, electric cars.

I. SMART GRID CONCEPT

Smart Grids are an automated system that independently monitors and distributes electricity flows to achieve maximum energy efficiency. The concept of smart grids focuses not on the modernization of certain technologies and equipment, but on the revision of the principles of development and creation of new, innovative technical equipment for the power industry. It should provide a much more complete satisfaction of the requirements of consumers and other stakeholders through a significant change in the physical and technological characteristics and functional properties of all components of the energy system [1].

- Measuring instruments and devices for storing information, including in particular smart meters and smart sensors.
- Improved control methods: distributed intelligent control systems and analytical tools to support communication at the level of real-time energy system objects, enabling the implementation of new algorithms and techniques for controlling the energy system, including the control of its active elements.
- Advanced power grid technologies and components: Flexible alternative power transmission systems (FACTS), DC transmission, superconducting cables, microgrids, semiconductor power electronics, power storage, etc.
- Improved interface and methods of adopted solutions - technologies and tools that enable the transformation of data obtained from various network objects into information for decision-making.

Integrated communications that allow the elements of the first four groups to ensure interconnection and interaction, which is in fact an intelligent network as a technological system. In recent years, there has been a clear trend towards electrification in the automotive industry. Virtually all global manufacturers plan to release new cars that will be driven by electrical power plants. Moreover, according to numerous experts, electric cars have a future. That is, in the coming years, more and more cars with zero emissions of harmful substances will begin to appear on the roads, which, according to predictions and forecasts, should eventually completely force out classic cars [2].

According to Wood Mackenzie, about 3.3% of sales of new passenger cars in China accounted for “clean” electric cars in 2018, compared to 0.7% in 2015 [3]. According to the international consulting company Frost & Sullivan, in 2018 two million electric vehicles were sold worldwide, and by 2025 their sales will increase to 25 million, which is expected to be 20.22% of all cars [4]. In March 2019, analysts at

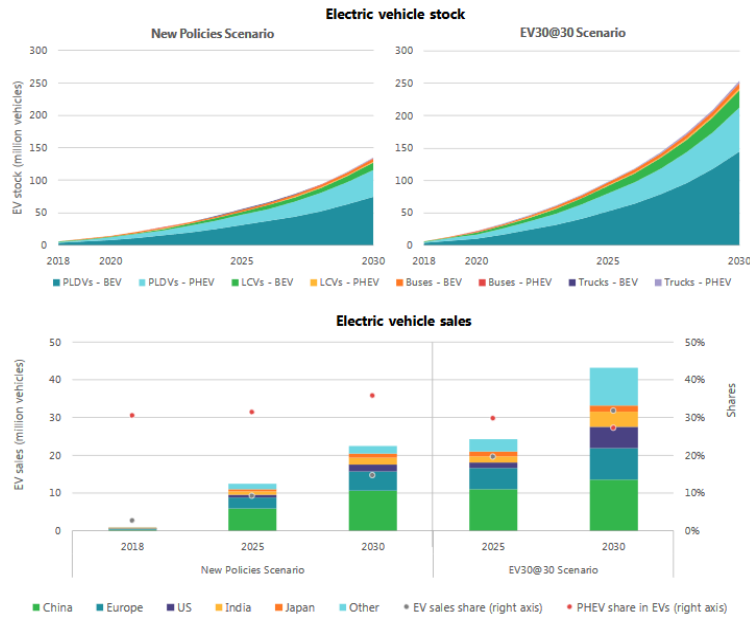


Fig. 1. Future global EV stock and sales by scenario, 2018-30 [5]

Bloomberg New Energy Finance provided data that in 2018 around 1.3 million all-electric cars were sold worldwide (excluding hybrids), and 60% of the volume fell on the Chinese market. According to Bloomberg New Energy Finance, by 2025 almost half of all buses in the world will be electric. The leader in this regard is China. He will have 99% of electric buses. One of the first cities where all buses are electric was Chinese Shenzhen - this is about 16.3 thousand urban electric buses [6].

II. EFFECT OF VOLTAGE UNBALANCE

One of the reasons for the deterioration in the quality of electric energy is the asymmetry of voltages in the electric network [4]. If the three-phase voltage has the same magnitude and is in a phase shift of 120° , then the three-phase voltage is called symmetrical. If the symmetry of the multiphase system is violated, the latter can be decomposed into three symmetric components — the direct sequence system, the inverse, and zero sequence systems superimposed on it [7]. As a rule, a three-phase voltage balance is an ideal situation for a power system. However, single-phase loads, unbalanced three-phase equipment and devices, poor connections to electrical connectors and many other factors cause voltage unbalance in the power system and reduce the quality of electricity. Thus, voltage asymmetry is one of the urgent problems of electric power systems.

In addition to the above, the cause of voltage unbalance can be emergencies in networks, such as asymmetric short circuits or phase failure [8]. Consider the main consequences of stress asymmetry.

1. Additional power loss. Voltage asymmetry always causes additional power loss in the power system. The higher the voltage unbalance coefficient, the greater the power dissipation. This ultimately leads to increased costs when paying for electricity [9], heating the motor windings, which can lead to the destruction of their insulation.
2. Decrease in equipment life cycle. High temperatures exceeding the nominal values for devices significantly reduce the service life of these devices and accelerate the cycle of their replacement, which significantly increases the cost of operation and maintenance of the equipment.
3. The effect on the operation of the relay. High zero sequence current resulting from voltage unbalance can lead to disturbance of the relay or reduce the sensitivity of the ground relay. This can lead to serious problems in terms of security of the power system.
4. Inaccurate measurements. The reverse and zero sequence of voltages or currents lead to inaccurate measurements in many types of measuring instruments. The inaccuracy of the measured values may affect the suitability of the settings and the coordination

of relay protection systems, the correctness of the decisions of some automatic functions of the system.

5. Influence on the operation of the transformer. Three-phase voltage with a high coefficient of voltage unbalance can lead to the fact that the flow inside the core of the transformer is asymmetric. This unbalanced flux will cause additional magnetic losses in the core, an increase in the temperature of the windings and may even damage the transformer.

In general, the consequences of voltage asymmetry in power systems are wide and serious. Voltage asymmetry can significantly shorten the life cycle of equipment, speed up its replacement and increase the cost of operating and maintaining the system. To eliminate non-random voltage asymmetry in the power system, it is necessary to develop a uniform load connection scheme even in the early stages of design. In this case, it is necessary to take into account their capacities and work schedules. Now, many electric cars are powered through 1 or 2 phases. Cars that are powered through 3 phases are extremely small. In addition, with an increase in the number of phases, the cost of charging also increases, which leads to the fact that consumers will prefer more affordable solutions [7]. Table 1 shows the charging parameters of various electric vehicles using various charging solutions [9].

Table 1
 Supported charging options for electric vehicles [9].

Parametrs	Volkswagen e-Up	Skoda CITIGOe IV	Nissan Leaf	Hyundai Kona Electric	Citroen C-Zero
Battery Useable, kWh	32.3	32.3	36	39.2	14.5
3-phase 16A (11 kW)	7.2	7.2	3.7	11	3.7
3-phase 32A (22 kW)	7.2	7.2	6.6	11	3.7
3-phase 16A (11 kW), A	2x16	2x16	1x16	3x16	1x16A
3-phase 32A (22 kW), A	2x16	2x16	1x29	3x16	1x16A
CCS (50 kW DC), kW	30	30	46	35	30

III. ELECTRIC NETWORK MODEL

Modeling is a very effective process of testing theoretical knowledge and the most accurate design in practice [10]. To study the effect of charges on the distribution system, a model was created on data from a classical village, where we know the load, approximate distances and types of power lines, and parameters of transformers. To create a model of the electric network, the parameters of real equipment were used [11]. The model was created using the Python programming language. Approximate distribution scheme of the village can be seen in Figure 2.

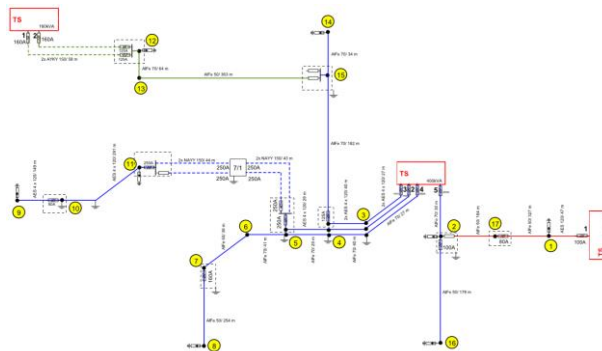


Fig. 2. Schematic single-line diagram of a distributed network of a village

To create a load model, an annual load schedule for each house was used. In Fig 3. You can see the total load curve of the whole village. Loads are represented mainly by private small houses.

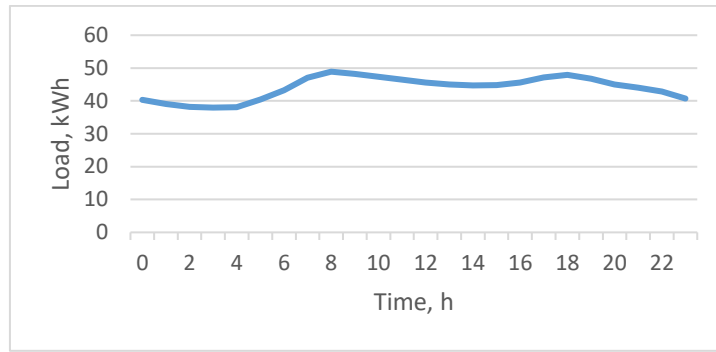


Fig. 3. Annual load schedule

To create a model of the charging station, the parameters of the charging station from Schneider Electric were used: the most affordable EVlink Wallbox Plus - T2 socket outlet - 1 phase - 16A/3.7kW (fig.4) [12] with the declared power of 3.7 kW (single-phase solution). To calculate the duration of charging, an Nissan Leaf electric car was used. The declared battery capacity of this car is 24 kWh [13]. It takes about 7 hours to fully charge this car. The model of the charging station is a load that is included in the given time intervals. For the time the machine was turned on for charging, three periods were selected at 17:00, 19:30, and 22:00. The timing and connection point are made randomly. The number of connected machines increases in increments of 5% of the total number of network loads.



Fig. 4. Charge for electric vehicle (EVlink Wallbox Plus) selected for modeling [12]

IV. SIMULATION RESULTS

As part of the simulation, a smooth increase in the number of connected electric vehicles was carried out. The increment was 5%, as can be seen in graph 5, the maximum current in the network increases with an increase in the number of electric vehicles. When connecting 35% of the total number of consumers, the maximum current value exceeded the permissible value for the cable. Thus, modern networks can only allow electric cars to have 35% of consumers.

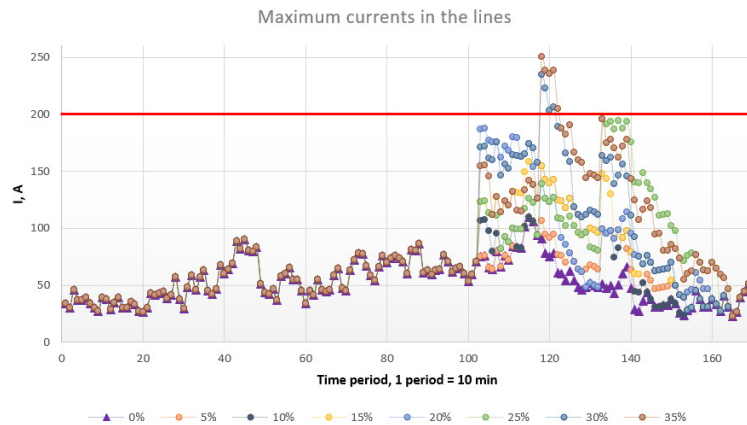


Fig. 5. Graph of maximum currents in the network.

Since the model is three-phase, it is possible to calculate the asymmetry coefficient of the voltage in the network. Since the time it takes to connect an electric vehicle to charge can be different, as well as the charging time (by how many percent the battery was discharged per day), the effect of asymmetry is manifested even when the charges are evenly distributed over the phases [14]. To control the voltage asymmetry in three-phase systems, we used the voltage asymmetry coefficient in the reverse sequence K_{2U} and the voltage asymmetry coefficient in the zero sequence K_{0U} . The voltage asymmetry coefficient in the reverse sequence is [11]:

$$Ku = (U_{2(1)} / U_{1(1)}) \cdot 100 \quad (1)$$

Where $U_{2(1)}$ - is the effective value of the voltage of the reverse sequence of the fundamental frequency of the three-phase voltage system, V; $U_{1(1)}$ - is the effective value of the voltage of the direct sequence of the fundamental frequency, V.

The calculation of asymmetry parameters was carried out for the most difficult mode, that is, when the number of electric vehicles was 35% of the total number of consumers. You can see the results of the calculations on graphs 6 7 8. The phase selection algorithm for the connection has the following steps [15]:

- 1) Phase current load measurement;
- 2) Measurement of the asymmetry coefficient;
- 3) Selection of the least loaded phase;
- 4) Check the asymmetry coefficient whether the phase is selected correctly;
- 5) Load connection.

As can be seen in Figure 8, a simple algorithm for selecting the phase for the connection significantly reduced the voltage unbalance coefficient.

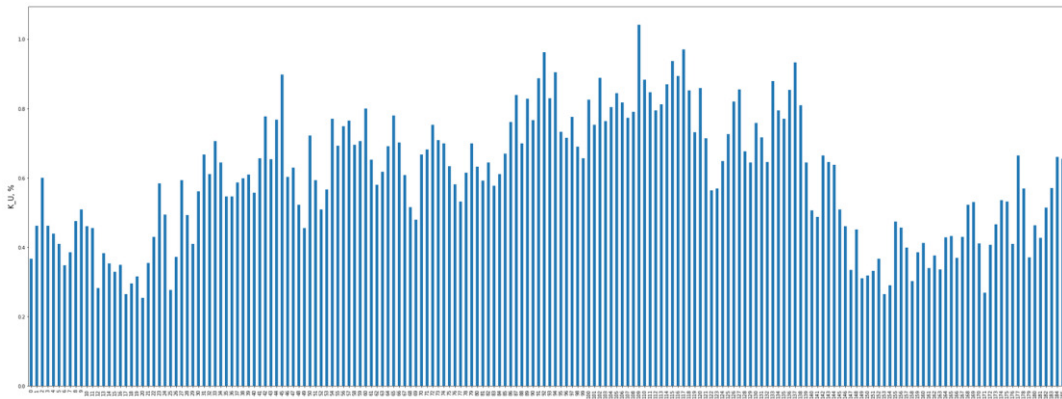


Fig. 6. Asymmetry factor for a network without the use of electric vehicles.

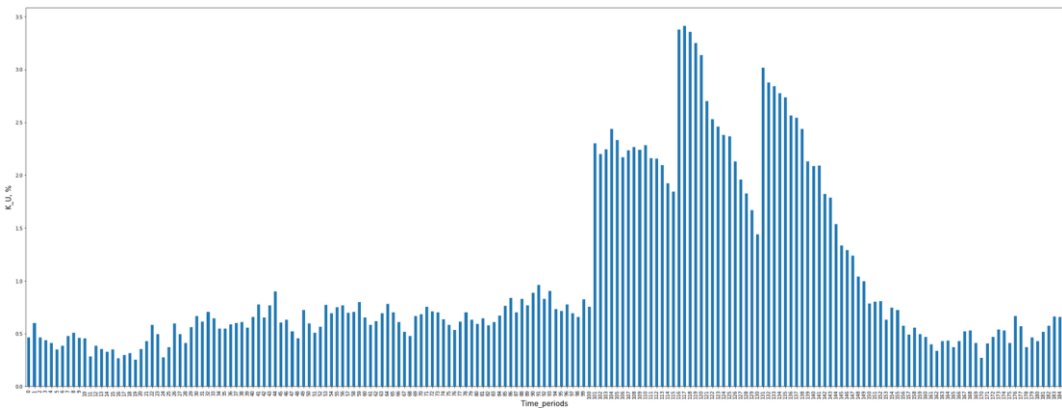


Fig. 7. Asymmetry factor for a network with the use of electric vehicles.

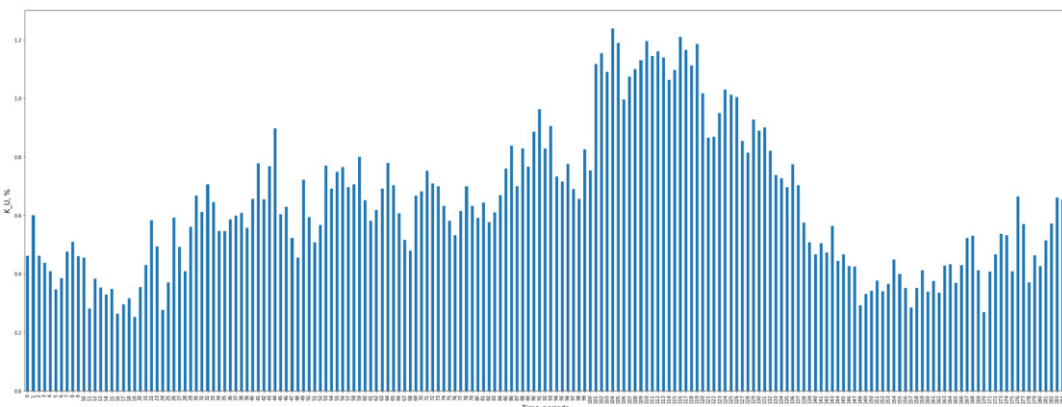


Fig. 8. Asymmetry factor for a network with the use of electric vehicles using the phase distribution algorithm.

V. CONCLUSION

An increase in the number of modern smart elements does not always positively affect the parameters of the electric network, as well as the quality of electric energy. So the increase in the number of electric cars, creates not only an overload of power lines, but also an increased load on the rest of the equipment, but also worsens the quality of electricity. In this regard, modern networks require an integrated approach to modernization. It is the comprehensive implementation of the Smart Grid concept that will allow avoiding the deterioration of the operation of electric networks, increase reliability, and for network companies will avoid unforeseen additional costs.

ACKNOWLEDGMENT

This work was supported by the Scientific Grant Agency of the Ministry of Education of Slovak Republic and the Slovak Academy of Sciences by the projects VEGA No. 1/0372/18

REFERENCES

- [1] S. Borlase, "Smart Grids: Infrastructure, Technology and Solutions", 577, (CRC, 2013) ISBN: 978-1-4398-2905-9
- [2] N. Nikmehr, S. N. Ravadanegh, "Optimal power dispatch of multi-microgrids at future smart distribution grids," IEEE Transactions on Smart Grid, 2015, pp. 1648–1657.
- [3] Electric Vehicle Outlook Data. Wood Mackenzie. August 2018., Available on internet: <https://www.woodmac.com/reports/power-markets-ev-charging-infrastructure-development-global-market-sizing-and-forecasts-29627>
- [4] IEA, Global EV Outlook 2018, May 2018, Available on internet: <https://webstore.iea.org/global-ev-outlook-2018>
- [5] Global EV Outlook 2019, Available on internet: <https://webstore.iea.org/global-ev-outlook-2019>
- [6] J. Ball, "The Global Electric-Car Showdown Is Officially on in China", 2019 Available on internet: <https://fortune.com/2019/03/22/electric-car-showdown-china/>
- [7] M. Kanálik, "Comparison of international power quality standarts", in Proceedings of the 4th International Scientific Symposium on Electric Power Engineering, ELEKTROENERGETIKA 2007, 2007, pp. 237.
- [8] J. Momoh, "Smart grid fundamentals of design and analysis," IEEE P., 2012, ISBN: 978-0-470-88939-8.
- [9] Database of Electric vehicle, unpublished, Available on internet: <https://ev-database.org>
- [10] M. Kolcun, M. Kanalik, D. Medved, Z. Conka, „Measuring of real value of short-circuit power in Island Operation Condition“, in: Electric Power Engineering (EPE). Ostrava: VSB-TU, 2015, pp. 418422. ISBN 978-1-4673-6787-5.
- [11] M. Kanálik, "Power system's modeling for purposes of harmonic voltage and current calculations", in Proceedings of the 9th International Scientific Conference Electric Power Engineering 2008, EPE 2008, pp. 157.
- [12] EVlink Catalog 2019 Electric vehicle charging solutions, unpublished.
- [13] Nissan Leaf manual, unpublished.
- [14] M. Pavlík, "Compare of shielding effectiveness for building materials", in Przegląd Elektrotechniczny, 95(5), 137-140. doi:10.15199/48.2019.05.33
- [15] F. Saffre, R. Gedge, "Demand-Side Management for the Smart Grid", in 2010 IEEE/IFIP Network Operations and Management Symposium Workshops. Institute of Electrical & Electronics Engineers (IEEE), 2010. Pp. 300–303.

Simulovanie synchronizácie činností robotov

Ján VAŠČÁK

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská Republika

jan.vascak@tuke.sk

Abstrakt – Obsahom tejto práce je poukázať na problematiku robotickej kooperácie. Ako testovacia úloha bola zvolená z oblasti robotických manipulačných liniek, konkrétne prekladanie obrobkov medzi dvomi bežiacimi pásmi. Úloha bola zrealizovaná prostredníctvom simulačného prostredia Webots, kde sme zvolili model dvoch príbuzných typov robotických ramien. V rámci riešenia tejto problematiky sa práca zaoberá tvorbou virtuálneho prostredia, jeho modelovaním ako aj návrhom samotného riadiaceho algoritmu. Napokon sa práca venuje samotnej realizácii návrhu a simulácii vyvinutého riešenia. Následne sú popísané vykonané experimenty a ich vyhodnotenie.

Kľúčové slová – kooperácia, riadenie, robot, simulácia

I. ÚVOD

Doteraz sa priemyselné roboty nasadzovali do menej náročných pracovných prostredí na vykonávanie hlavne manipulačných operácií, pričom na seba preberali synchronizáciu činností jednotlivých zariadení. V súčasnosti prechádzajú významným inovačným procesom, ktorého podstatná časť je sústredená na zvyšovanie ich inteligencie a multifunkčnosti [1]. Rozvoj priemyselnej robotiky upúšťa od ich individuálneho a izolovaného nasadzovania a prechádza k skupinovému nasadzovaniu.

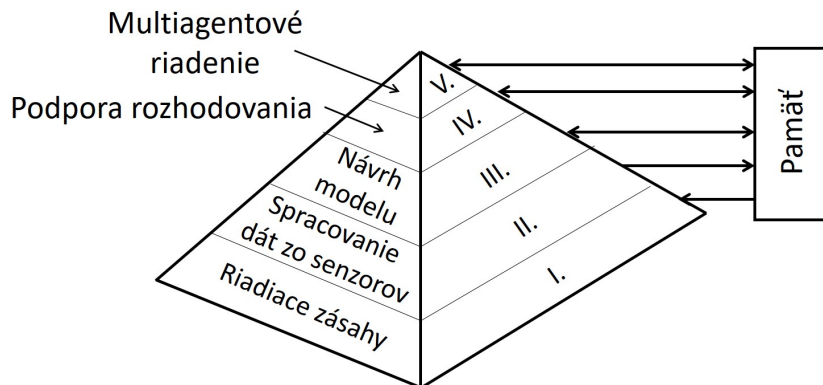
Prínos skupinového nasadzovania je zrejmý. K výhodám patrí, že skupinové roboty dokážu vykonávať viac druhov rôznych činností. Od duo robotov možno v najbližšej budúcnosti očakávať, že prekonajú aj ľudskú zručnosť, a to aj v citlivosti, nielen v sile, rýchlosti a presnosti. Základnou myšlienkou vývoja duo robotov sú činnosti človeka vykonávané obidvoma rukami pri každodennej manipulácii a pri spolupráci s viacerými pracovníkmi. Novými úlohami v multirobotických systémoch a pri duo robotoch sú stanovenie postupu činnosti a manipulačných dráh robotov a synchronizácia ich pohybu a rýchlosti. Kľúčovými aspektmi multirobotických systémov sa stáva zvládnutie paralelného riadenia, synchronizácia a kooperácia ich činnosti.

Dnešná doba ponúka enormné množstvo riešení hľadaného problému, ktoré si vieme pozrieť a využiť, avšak veľmi častý jav je problém s ich pochopením, spojením, resp. využitím, ktoré často odrádza užívateľ a od ich použitia pre málo podrobný popis kódu, funkcionality alebo celého projektu ako celku. Následkom toho je, že užívateľ hľadá čo najpriehľadnejšie a najrýchlejšie riešenie pre jeho konkrétny problém, či už vo forme návodu alebo hotového projektu, ktorý môže využiť s najmenším úsilím.

II. SIMULÁCIA SYNCHRONIZÁCIE ROBOTOV

Prevládajúci prístup v prípade synchronizácie viacerých robotov je založený na tzv. *agentovom* prístupe, t.j. jednotlivé roboty sú prezentované ako agenti, ktoré sú definované ako samostatné entity schopné vykonávať svoje funkcie pri zabezpečení potrebných informácií (napr. snímače, informácie o stavoch ostatných robotov, predpísaných úlohách a pod.). Na základe vykonávaných funkcií, ktoré hierarchicky za sebou nasledujú, je možné vytvoriť riadiacu a informačnú pyramídu, vid' Obr. 1, kde je možné vidieť päť základných úrovní. Prvé štyri sa týkajú samostatného robota. Najvyššia, piata úroveň predstavuje tzv. multiagentový prístup, vedúci k vytvoreniu multiagentového systému (MAS), čiže prípad, ak navzájom interagujú aspoň dva rôzne roboty [2].

V prípade tímových úloh musia byť roboty vybavené ďalšími nadstavbovými schopnosťami, ktoré zastrešujú aj funkcie z predošlých úrovní a definujú MAS. Koncept MAS bol inšpirovaný niekoľkými zjavne odlišnými disciplínami ako napr. ekonómia, sociológia a biológia a predstavuje spoločné prepojenie medzi nimi, kde je bežné využitie prostriedkov umelej inteligencie [3]. Síce neexistuje žiadna všeobecná definícia agenta, pretože tento výraz je príliš rozšírený, existuje



Obr. 1 Riadiaca a informačná štruktúra robota.

na rôznych úrovniach zložitosti a môže sa využívať v mnohých formách. Avšak ho môžeme charakterizovať niektorými minimálnymi vlastnosťami, ktoré musia byť splnené [4]:

- 1) Autonómnosť: Agent je schopný pôsobiť nezávisle od človeka. Má jednotlivé vnútorné stavy a ciele. Autonómnosť nemôže byť skutočná bez schopnosti proaktivity, kde nečaká iba pasívne na vonkajšie podnety, ale sa aktívne ujíma iniciatívy.
- 2) Kooperácia: Je to schopnosť interakcie s inými činiteľmi (prípadne ľuďmi) pomocou komunikačného jazyka. Je to základná vlastnosť, bez ktorej nemôžeme hovoriť o MAS-e.
- 3) Učenie: Možno nie je vždy nevyhnutné, ale je to veľmi užitočná vlastnosť, ktorá umožňuje agentovi byť flexibilný a schopný sa prispôbiť zmenám prostredia.

III. POUŽITÉ SIMULAČNÉ PROSTRIEDKY

Keďže návrh a odskúšanie systému na kooperáciu robotov boli vykonané na simulačnej báze, prvou otázkou je voľba vhodného prostriedku. V súčasnosti sa na takéto účely najviac využívajú prostriedky *Webots* a *Gazebo*. Po vykonaní rozboru danej úlohy bol zvolený prvý menovaný prostriedok.

Webots je tzv. open source a multiplatformová desktopová aplikácia, ktorá sa používa na simuláciu robotov [5]. Poskytuje kompletné vývojové prostredie na modelovanie, programovanie a simuláciu robotov, vid' Obr. 2. Bol navrhnutý pre profesionálne použitie a je široko používaný v priemysle, vzdelávaní a výskume.

Umožňuje jednoducho navrhnuť kompletnú simuláciu z prostredia robotiky vďaka knižniciam, ktoré obsahujú roboty, senzory, ovládače, objekty a materiály. Podporuje import CAD modulov a na editáciu simulácie a regulátorov robotov využíva moderné grafické užívateľské prostredie. *Webots* taktiež obsahuje širokú škálu robotov ako napr. roboty s dvomi kolesami, bipedické roboty, roboty s viacerými nohami, modulárne roboty, automobily, lietajúce roboty, autonómne podvodné vozidlá, pásové roboty, kozmické vozidlá atď.

Systém môže pracovať pod rôznymi operačnými systémami ako MS Windows, Linux a macOS. Simulácie môžu byť exportované ako filmy. Interaktívne HTML scény alebo animácie môžu byť streamované do ľubovoľného prehliadača pomocou *webrgl* a *websockets*.

Robot môže byť programovaný v jazykoch C, C++, Python, Java, MATLAB alebo ROS s jednoduchým API pokrývajúcim všetky základné potreby robotiky. V našom prípade práve táto variabilita programovacích jazykov rozhodla o použití tohto prostriedku. Navyše, *Webots* je známy jeho rozsiahlou a komplexnou dokumentáciou. Okrem toho sa jedná o robustný systém. Aby sa zabezpečila kvalita kódu, každá modifikácia kódu sa podrobí vzájomnej kontrole a odošle sa do automatickej testovacej sady testujúcej všetky API. Spätná kompatibilita je zaručená a dobre zdokumentovaná medzi hlavnými verziami.

IV. NÁVRH SIMULÁCIE KOOPERATÍVNEJ ČINNOSTI

Skôr než pristúpime k samotnému riešeniu danej úlohy, na ktorej si ukážeme možnosti synchronizácie pohybov, je potrebné popísať prostriedky, na základe ktorých sa bude simulovať kooperácia robotov. Bude sa jednať o úlohu synchronizovaného prekladania súčiastok z jedného pásu na druhý, pričom použité roboty budú odlišného typu. V takejto situácii je potrebné si uvedomiť technické vlastnosti robotov, či už priestorový záber alebo kinematické vlastnosti spomínaných zariadení [6], za účelom zistenia kritických oblastí, kde môže dôjsť k zrážke robotov (v tomto prípade ich ramien), ako aj plánovania pohybov s cieľom splnenia danej úlohy. Vráťane toho sa aj rozhoduje, ktorý robot, ktorú súčiastku uchopí a premiestni.

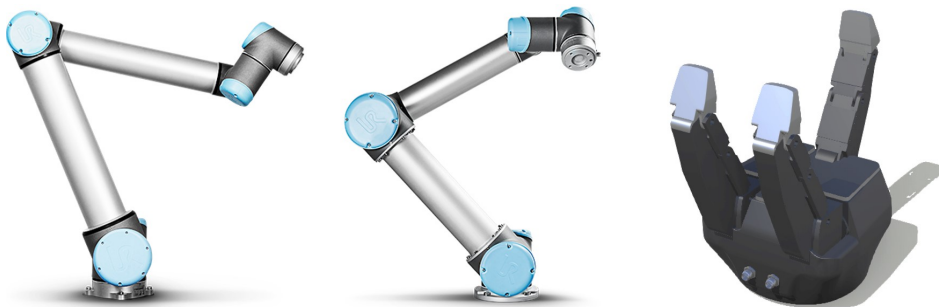


Obr. 2 Príklad ukážky prostredia prostriedku *Webots*.

Universal Robots UR10 firmy Universal Robots je veľký priemyselný robot určený pre väčšie úlohy, kde hrá dôležitú úlohu presnosť a spoľahlivosť, konkrétne pre bremená do 10 kg. UR10 sa dobre hodí pre kooperatívne procesy s vyššou hmotnosťou bremien, ako sú balenie, paletizácia, montáž alebo odoberanie a ukladanie. S dosahom až 1300 mm je robot UR10 efektívnejší pri práci vo väčšom priestore. Môže šetriť čas na výrobných linkách, kde je dôležitým faktorom vzdialenosť. Výhodou je, že sa ľahko programuje a ponúka rýchle nastavenie.

Ľahký a flexibilný robot UR5 toho istého výrobcu je ľahším variantom predošlého typu, ktorá umožňuje automatizovať opakované a nebezpečné úlohy so zaťažením do 5 kg a pracovným dosahom až 850 mm. Jedná sa teda o kolaboratívne procesy s nízkou hmotnosťou ako napr. odoberanie, ukladanie a testovanie.

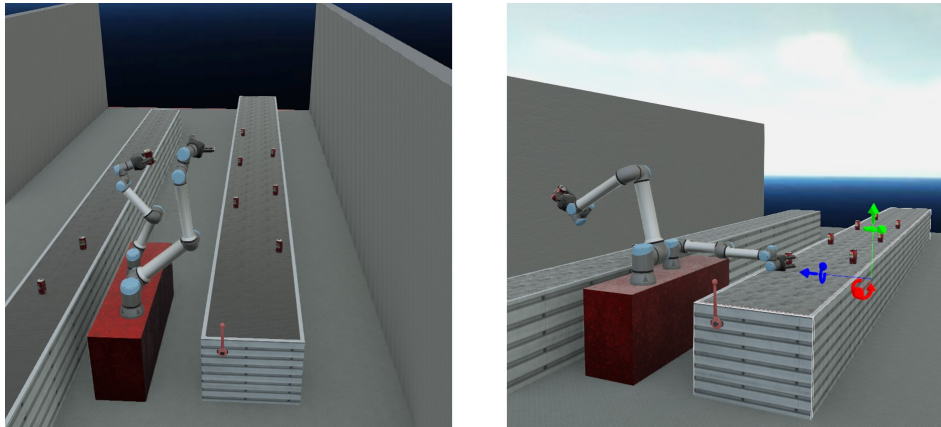
Aby roboty mohli vykonávať manipulačné úlohy, je potrebné ich vybaviť uchopovacím adaptérom. V tomto prípade sa jedná o uchopovač typu ROBOTIQ 3F s tromi prstami, ktorý je schopný sa prispôbiť tvaru objektu, aby bolo možné pevné uchopenie. Samotný proces uchopenia rieši senzorika a riadiaci systém samotného uchopovača, takže odpadá potreba riešiť túto časť problému a je možné sa sústrediť iba na riešenie synchronizácie pohybov oboch robotov. Na Obr. 3 sú zobrazené všetky tri základné prostriedky, pomocou ktorých budeme riešiť danú úlohu.



Obr. 3 Použité robotické prostriedky (zľava doprava): UR10, UR5 a ROBOTIQ 3F

Prvým krokom návrhu je vytvorenie vhodného technologického procesu a s ním súvisiaci výber vhodného robota, senzorov a prostredia, ktoré by boli pre danú úlohu najvhodnejšie a spĺňali predpísané požiadavky. Ďalším krokom je návrh logiky robota v interakcii s prostredím, v ktorom sú zvolené roboty nasadené. V našom prípade sa jedná o premiestňovanie objektov (napr. súčiastok alebo plechoviek) z jedného pásového dopravníka na druhý. Pre túto úlohu je ešte potrebné na roboty umiestniť snímače, ktoré nám budú merať vzdialenosti a pozície premiestňovaných objektov. Vďaka týmto údajom je možné navrhnuť regulátor pohybu robotických ramien. Výhodou simulačného prostredia *Webots* je možnosť si vytvoriť dané prostredie z rôznych uhlov pohľadu,

kde sa dajú podľa potreby zvyrazňovať detaily alebo veličiny (sily, momenty a pod.) riešenej úlohy, ako je to znázornené na Obr. 4.



Obr. 4 Zobrazenie simulácie kooperácie dvoch robotických ramien (UR10 a UR5): vľavo všeobecný náhľad, vpravo zobrazenie otáčavých síl kĺbov.

V. ZÁVER

Uvedené riešenie predstavuje istý programovací rámec, ktorý umožňuje sa sústrediť iba na problémy spojené s návrhom zloženia danej zostavy a riadením jednotlivých komponentov, čo je predmetom ďalšieho výskumu, kde sa budú navrhovať rôzne typy regulátorov a vzájomne porovnávať ich vlastnosti.

Do budúca jedným zo spôsobov vylepšenia tohto návrhu by mohla byť úprava jeho programovej stránky, t.j. zovšeobecnenie riešenia vzhľadom na to, že momentálne je toto riešenie výrazne obmedzené len na určitý počet a typ robotov. Taktiež čo sa týka zovšeobecnenia, by bolo vhodné upraviť rozpoznávanie rotácie robotov tak, aby nebolo závislé len na niekoľkých oporných bodoch ale aby bolo spojenou funkciou nameranej rotácie. Napokon zaujímavým smerom výskumu sa v tejto súvislosti javí aj technológia tzv. *internetu vecí*, keďže sa môže jednať o priestorovo značne rozložené úlohy ako napr. komplexné výrobné linky, kde však na význame narastá aj problematika počítačovej ochrany snímaných dát a komunikácie [7].

POĎAKOVANIE

Táto práca je finančne podporená Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky prostredníctvom projektu KEGA č. 033 TUKE-4/2018 – Smart Industry/Architektúry inteligentných informačných a kybernetických systémov.

LITERATÚRA

- [1] S. K. Khaitan and J. D. McCalley, "Design techniques and applications of cyberphysical systems: A survey," *IEEE Systems Journal*, vol. 9, no. 2, pp. 350–365, 2015.
- [2] J. Vaščák, "Decision-making systems in mobile robotics," in *Autonomous Decision Systems Handbook*, K. Mls, Ed. BEN, 2011, ch. Decision-making Systems in Mobile Robotics, pp. 56–88.
- [3] P. Stone and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," *Autonomous Robots*, vol. 8, no. 3, pp. 345–383, 2000.
- [4] H. S. Nwana, "Software agents: An overview," *The Knowledge Engineering Review*, vol. 11, no. 3, pp. 205–244, 1996.
- [5] Cyberbotics, "Webots: Open source robot simulator," [online] <https://www.cyberbotics.com/>.
- [6] C. Georgoulas, T. Linner, A. Kasatkin, and T. Bock, "An Aml environment implementation: Embedding turtlebot into a novel robotic service wall," in *ROBOTIK 2012 - Proceedings for the conference of ROBOTIK 2012, 7th German Conference on Robotics, 21-22 May 2012, International Congress Center Munich (ICM) in conjunction with AUTOMATICA, Munich, Germany*. VDE-Verlag, 2012, pp. 1–6.
- [7] M. Horton, L. Chen, and B. Samanta, "Enhancing the security of iot enabled robotics: Protecting TurtleBot file system and communication," in *2017 International Conference on Computing, Networking and Communications, ICNC 2017, Silicon Valley, CA, USA, January 26-29, 2017*. IEEE Computer Society, 2017, pp. 662–666.

Vplyv pripojenia nového zdroja energie na vybrané ukazovatele v elektrickej sieti

¹Daniel PÁL, ²Lubomír BEŇA, ³Jakub URBANSKÝ

^{1,2,3} Katedra elektroenergetiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹daniel.pal@tuke.sk, ²lubomir.bena@tuke.sk, ³jakub.urbansky@tuke.sk

Abstrakt — Článok sa zaoberá témou pripojiteľnosti obnoviteľných zdrojov elektrickej energie a vyšetruje, aké vplyvy majú na existujúcu sieť. Objasňuje, ako sa menia kľúčové parametre elektrickej siete ako napätia a straty, ak sú do nej pripojené nové zdroje s nepredikovateľnou výrobou. To je v súčasnosti veľmi aktuálna téma, s ohľadom na neustály nárast dopytu po elektrickej energii. Rozšírenie výroby elektrickej energie z obnoviteľných zdrojov má vplyv na existujúcu elektrickú sieť. Vzniká potreba neustále identifikovať, či sú tieto vplyvy pozitívne alebo negatívne. Prvá časť článku je venovaná téme obnoviteľných zdrojov energie a popisuje, ako narastali celosvetovo inštalované výkony obnoviteľných zdrojov od roku 2010. Nasledujúca časť opisuje model siete, na ktorej bola realizovaná simulácia. Záverečná časť vyhodnocuje výsledok simulácie, zmenu strát a napätia v sieti.

Kľúčové slová — obnoviteľné zdroje energie, simulácia, straty.

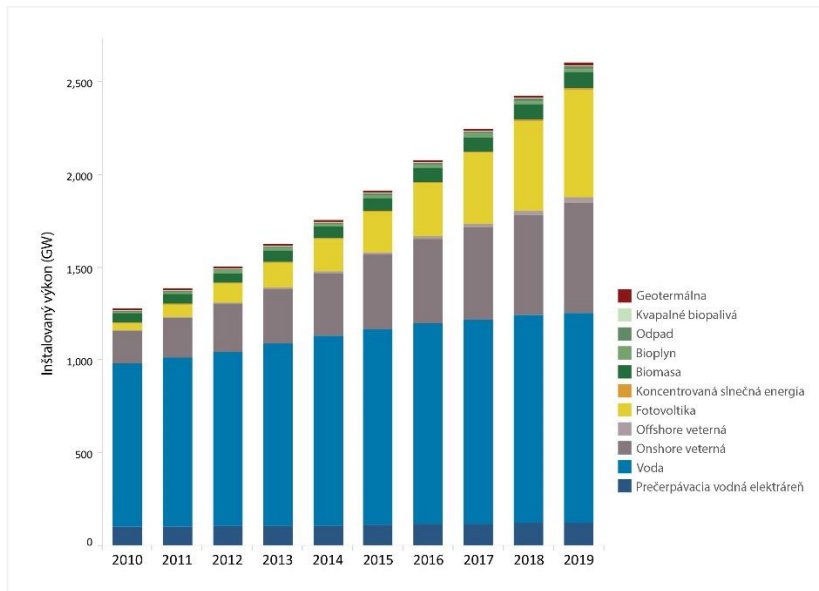
I. ÚVOD

Európska únia (EÚ) kladie v posledných rokoch veľký dôraz na využívanie obnoviteľných zdrojov energie (OZE) oproti konvenčným zdrojom. EÚ sa zaviazala do konca roku 2020, pokryť 20 % z konečnej spotreby energie pomocou OZE. Na rok 2030 sa zaviazala EÚ pokrývať až 32 % konečnej spotreby z OZE. V roku 2023 sa bude daný cieľ prehodnocovať, avšak jedine na ešte vyššiu hodnotu s ohľadom na ratifikáciu Parížskej dohody [1][2][3].

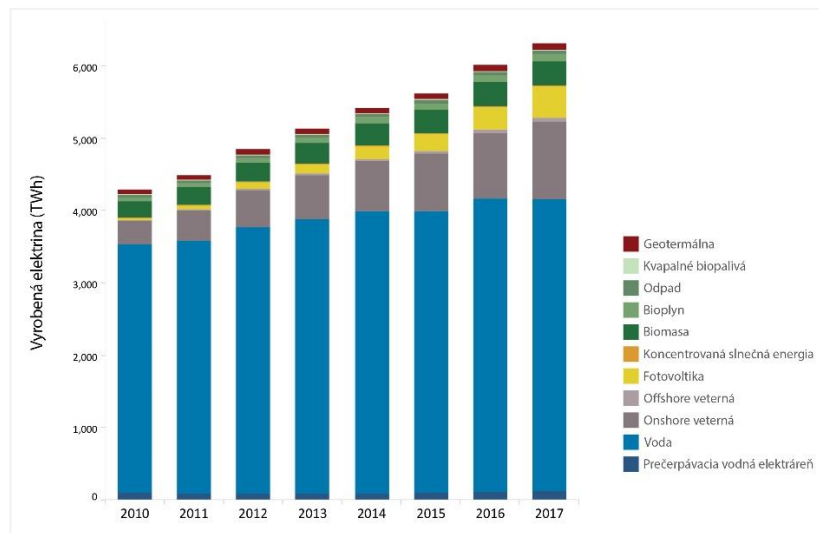
Medzi najčastejšie využívané OZE patrí:

- slnečná energia,
- veterná energia,
- vodná energia,
- geotermálna energia,
- biomasa.

Zvyšovanie celosvetového inštalovaného výkonu a množstvo vyrobenej elektrickej energie z OZE od roku 2010 znázorňuje Obr. 1 a Obr. 2. Obrovský nárast OZE vyžaduje to, aby sa elektrická sieť stala flexibilnejšou a dokázala reagovať na variabilitu a neistotu prevádzkových podmienok v rôznych časových rámcoch. Na Obr. 1 je možné vidieť, že sa inštalovaný výkon OZE na svete od roku 2010 každým rokom zvyšuje v priemere o 150 až 250 GW. Obr. 2 znázorňuje elektrickú energiu vyrobenú z OZE, ktorá sa s ohľadom na ročný prírastok inštalácií logicky taktiež zväčšuje.



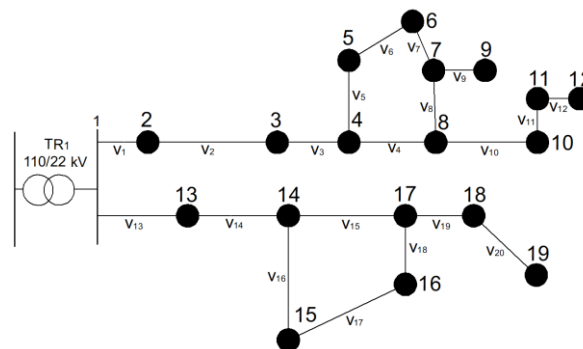
Obr. 1 Inštalovaný výkon obnoviteľných zdrojov energie (OZE) vo svete [4]



Obr. 2 V roba elektrickej energie z obnoviteľných zdrojov energie (OZE) vo svete [4]

II. VY ETROVANÝ PR KLAD

Vy etrovan  sie  je zobrazen  na Obr. 3. Sie  je nap jan  z jedn ho bodu (transform tor TR₁ 110/22 kV). Najhor  i stav nastane v pr pade poruchy na veden   . 1 alebo  . 13, kedy doch dza k  pln mu zastaveniu dod vky do zvy n ch  ast  sie . To mo no ozna i  za najv    iu nev hodu topol gie danej sie . Napr klad v pr pade v padku vedenia  . 10 ostan  bez dod vky len uzly  . 10,  . 11 a  . 12. V padok vedenia  . 10 nebude ma  vplyv na zvy n   as  sie  s v nimkou v   ie spomenut ch uzlov.



Obr. 3 Vy etrovan  sie 

Parametre vetiev a uzlov s  uvedenn  v Tab. 1 a Tab. 2.

Tabuľka 1
 Parametre vetiev

Číslo vetvy	Rezistencia (Ω)	Indukčná reaktancia (Ω)	Číslo vetvy	Rezistencia (Ω)	Indukčná reaktancia (Ω)
1.	0,148	0,134	11.	2,037	0,769
2.	0,923	0,835	12.	2,425	0,915
3.	0,738	0,668	13.	0,886	0,802
4.	0,886	0,802	14.	0,959	0,868
5.	1,002	0,690	15.	1,033	0,935
6.	1,164	0,439	16.	2,004	1,380
7.	1,455	0,549	17.	1,940	0,732
8.	1,002	0,690	18.	1,202	0,828
9.	1,940	0,732	19.	1,455	0,549
10.	1,202	0,828	20.	2,425	0,915

Tabuľka 2
 Parametre uzlov

Číslo uzla	Odoberaný činný výkon v kW	Odoberaný jalový výkon v kVAr	Číslo uzla	Odoberaný činný výkon v kW	Odoberaný jalový výkon v kVAr
1.	0	0	11.	700	140
2.	2500	600	12.	1700	350
3.	1800	400	13.	2500	600
4.	2000	500	14.	2700	600
5.	500	400	15.	1500	350
6.	2000	500	16.	300	0
7.	500	300	17.	1000	200
8.	1000	200	18.	1500	350
9.	1000	0	19.	1300	300
10.	1000	200			

III. VYŠETROVANÉ PRÍPADY

Pri zvyšovaní inštalovaného výkonu z OZE dochádza k decentralizácii výroby elektrickej energie s ohľadom na výrobu v miestach spotreby (napr. malé inštalácie na rodinných domoch, administratívnych budovách, malé fotovoltaické parky a pod.). Tento príspevok vyšetroje vplyv decentralizovanej výroby na vybrané ukazovatele v sieti. Ukazovatele, ktoré boli vyšetrované v príspevku boli napätia v uzloch a celkové činné a jalové straty, ktoré vznikajú pri prevádzke siete.

Úlohou tohto príspevku je preskúmať zmenu napätí v uzloch a celkové činné a jalové straty, v prípade pripojenia OZE do jednotlivých uzlov. Na pripojenie OZE boli vybrané uzly č. 7 a č. 16. Dôvodom výberu daných uzlov bolo optimálne umiestnenie OZE s ohľadom na zlepšenie parametrov celej siete. V prípade umiestnenia na začiatku by sa parametre na konci vedenia zlepšili len minimálne. Z dôvodu topológie siete napätie smerom na koniec vedenia vždy klesne z hodnoty, ktorá bola na začiatku.

V simulácii bol uvažované, že do uzla č. 7 je možné pripojiť len taký OZE, ktorý buď dodáva do siete iba činný výkon s výkonom 1,5 MW, buď dodáva do siete iba jalový výkon s výkonom 0,8 MVar alebo dochádza k ich kombinácii (napr. fotovoltaický panel v kombinácii s vhodným striedačom, ktorý dokáže dodávať aj jalový výkon). V uzle č. 16 bol predpoklad pripojenia zdroja, ktorý vyrába iba činný výkon s hodnotou 1,3 MW.

S ohľadom na pripojenie vyššie spomenutých zdrojov bola sledovaná zmena strát a napätí v sieti. Stanovených bolo 8 variantov (Tab. 1), pričom prvý z nich odkazuje na súčasný stav. Zvyšných 7 stavov rieši kombinácie v prípade pripojenia č. 7 a č. 16.

Tabuľka 3
 Vyšetrované prípady

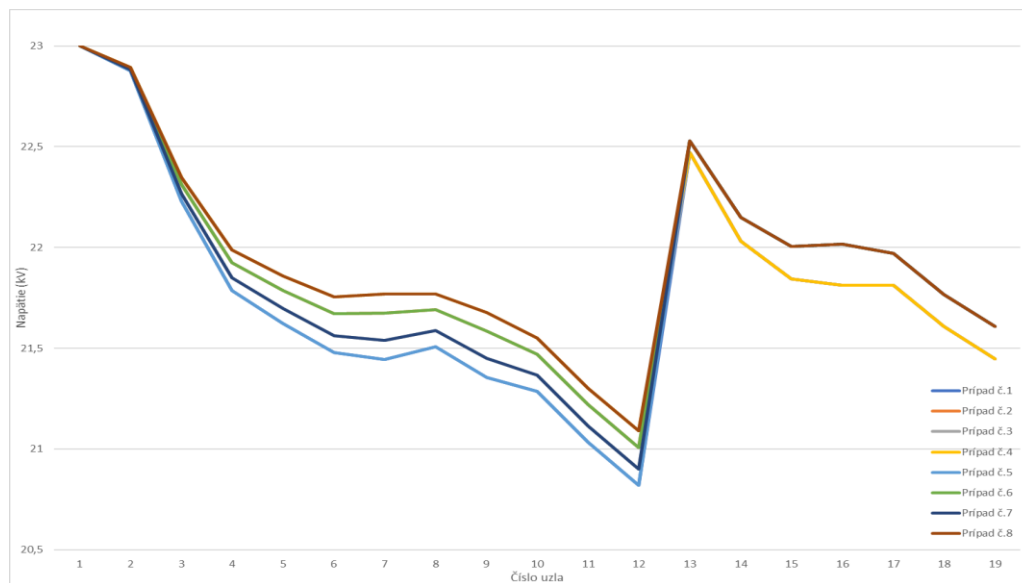
Prípad č. 1	Súčasný stav	Prípad č. 5	16P
Prípad č. 2	7P	Prípad č. 6	7P + 16P
Prípad č. 3	7Q	Prípad č. 7	7Q + 16P
Prípad č. 4	7P + 7Q	Prípad č. 8	7P + 7Q + 16P

7P → v uzle č. 7 je iba výroba činného výkonu
 7Q → v uzle č. 7 je iba výroba jalového výkonu
 7P + 7Q → v uzle č. 7 je výroba činného a jalového výkonu
 16P → v uzle č. 16 je iba výroba činného výkonu
 7P + 16P → v tomto prípade dochádzalo už ku kombinácii. Bolo uvažované, že v uzle č. 7 je vyrábaný iba činný výkon, ale do siete vyrába aj zdroj, ktorý je pripojený do uzla č. 16.
 7Q + 16P → v uzle 7 je iba výroba jalového výkonu a súbežne do siete je pripojený zdroj v uzle 16.
 7P + 7Q + 16P → všetky zdroje súbežne vyrábajú do siete.

IV. VÝSLEDKY SIMULÁCIE

Výsledky simulácie sú znázornené na Obr. 4 a Obr. 5. Pri používaní zdroja, ktorý je pripojený do uzla č. 7 nebol zaznamenaný vplyv na druhú stranu siete. To je možné pozorovať aj na Obr. 4, kde od 13. uzla sú viditeľné len 2 krivky kvôli prekryvaniu sa jednotlivých hodnôt. Na ukážku boli vybrané niektoré uzly a výsledky napätia sú uvedené v Tab. 4. Je možné pozorovať, že niektoré hodnoty sú rovnaké, ako bolo vysvetlené vyššie. Taký prípad nastal v prípade použitia iba zdroja v uzle č. 1, pričom to nemalo vplyv na druhú stranu siete. Napríklad v Tab. 4 v uzle č. 16 sú len 2 hodnoty. Ak nebol používaný zdroj v uzle č. 16, vtedy hodnota bola 21,8133 kV, ale ak už sa uvažovalo s novým zdrojom (od 5. prípadu), vtedy sa hodnota zvyšovala na 22,0179 kV.

Z Obr. 4 je tiež vidno, že napätia sú v súčasnosti najhoršie. Akonáhle dochádza k pripojeniu nového zdroja, tak sa napätie v príslušnej časti zvyšuje. Najlepší variant je v prípade ak sú pripojené oba zdroje (do uzla č. 7 a č. 16). Rozdiel medzi napätiami v danom prípade bol v rozmedzí 0,2 – 0,3 kV.

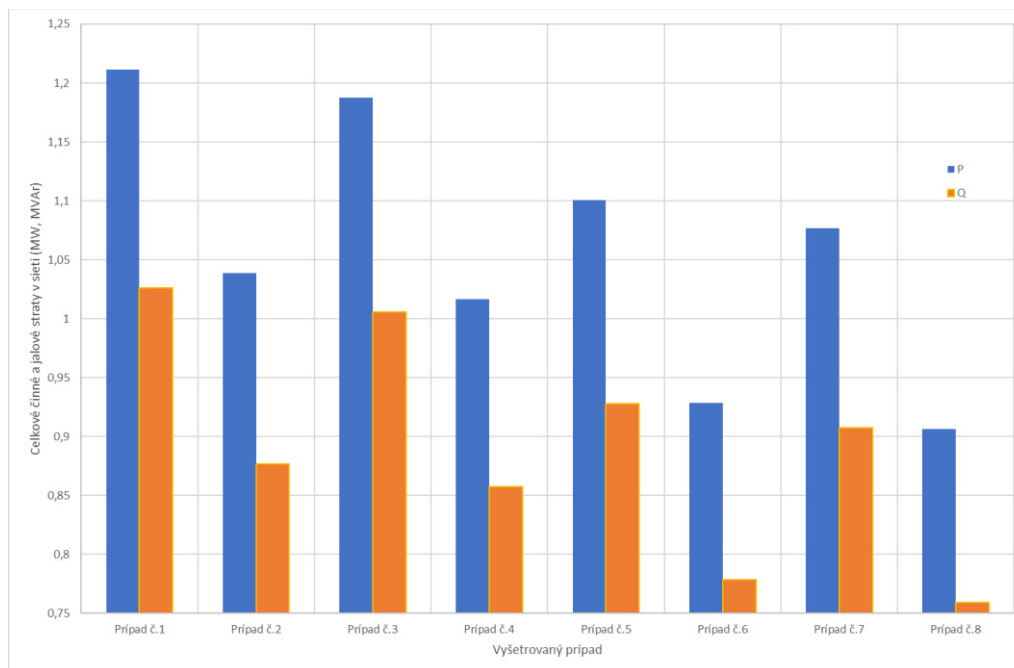


Obr. 4 Vplyv vyšetrovaných prípadov na napätia v uzloch

Tabuľka 4
 Vplyv vyšetrovaných prípadov na napätia vo vybraných uzloch

	Číslo uzla					
	6.	7.	8.	15.	16.	17.
Prípad č.1	21,4779	21,4456	21,5072	21,8431	21,8133	21,8126
Prípad č.2	21,6715	21,6752	21,6902	21,8431	21,8133	21,8126
Prípad č.3	21,5610	21,5395	21,5869	21,8431	21,8133	21,8126
Prípad č.4	21,7530	21,7679	21,7692	21,8431	21,8133	21,8126
Prípad č.5	21,4779	21,4456	21,5072	22,0048	22,0179	21,9717
Prípad č.6	21,6715	21,6752	21,6902	22,0048	22,0179	21,9717
Prípad č.7	21,5610	21,5395	21,5869	22,0048	22,0179	21,9717
Prípad č.8	21,7530	21,7679	21,7691	22,0048	22,0179	21,9717

Pripojenie nového zdroja do siete má vplyv aj na celkové straty v sieti. V prípade pripojenia nového zdroja dôjde vždy k poklesu strát. V najnižšej miere klesnú vtedy, ak v uzle č. 7 pripojený zdroj dodáva iba jalový výkon (prípád č. 3). To nastáva z dôvodu celkového odberu jalového výkonu, ktorý je na úrovni 6 MVar. Celkový odber činného výkonu je 25,5 MW. Preto pripojenie zdroja s činným výkonom má väčší vplyv na straty. Vždy ak pripojený zdroj dodával iba činný výkon, vtedy straty klesli najviac. To bol prípad č. 2, 4, 6 a č. 8. Najlepšie výsledky boli zaznamenané v prípade pripojenia oboch zdrojov. Rozdiel medzi najhorším a najlepším výsledkom v prípade činných strát bol 0,3 MW a v prípade jalových strát 0,25 MVar.



Obr. 5 Vplyv vyšetovaných prípadov na celkové činné a jalové straty v sieti

V. ZÁVER

Príspevok sa venoval zmene strát a napätí v prípade, ak sa do centralizovanej siete pripájajú OZE. S ohľadom na rozsah obnoviteľných zdrojov pripojených do siete, ktorých vplyv je nutné neustále prehodnocovať je to vysoko aktuálna téma.

V príspevku bola sledovaná sieť s celkovým odberom činných výkonov v uzloch na úrovni 25,5 MW a celkovým odberom jalových výkonov 6 MVar. V sieti sa sledovala zmena napätí a strát s ohľadom na pripojenie zdrojov do uzlov č. 7 a č. 16. Najlepšie výsledky s porovnaním so súčasným stavom vznikli vtedy, ak boli zdroje pripojené do oboch uvažovaných uzlov (č. 7 a č. 16) súčasne a zdroj v uzle č. 7 vyrábala činný aj jalový výkon. V porovnaní s celkovým odberom, výkon týchto pripojených zdrojov je veľmi malý, len 2,8 MW a 0,8 MVar. To je len 10 % z celkového výkonu. Dokázali však znížiť celkové činné a jalové straty o 25 % a tiež napätia v uzloch priemerne zvýšili o 0,2 – 0,3 kV. S ohľadom na výsledky simulácie je možné konštatovať nutnosť odsimulovať pripojenie nového zdroja do siete z dôvodu zistenia, či jeho pripojenie nespôsobuje preťaženie vetvy a tiež zvýšenie napätia nad dovolenú hodnotu.

POĎAKOVANIE

Túto prácu podporila Vedecká grantová agentúra Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky a Slovenskej akadémie vied grantom VEGA č. 1/0372/18.

LITERATÚRA

- [1] Energia z obnoviteľných zdrojov [online]. [cit. 2020.05.25]. Dostupné na internete: <https://www.europarl.europa.eu/factsheets/sk/sheet/70/energia-z-obnovitelnych-zdrojov>
- [2] Renewable energy directive. [online]. [cit. 2020.05.25]. Dostupné na internete: https://ec.europa.eu/energy/topics/renewable-energy/renewable-energy-directive/overview_en
- [3] Parížská dohoda [online]. [cit. 2020.05.20]. Dostupné na internete: [https://eur-lex.europa.eu/legal-content/SK/TXT/?uri=CELEX:22016A1019\(01\)](https://eur-lex.europa.eu/legal-content/SK/TXT/?uri=CELEX:22016A1019(01))
- [4] Solar energy. [online]. [cit. 2020.05.17]. Dostupné na internete: <https://www.irena.org/solar>

Analysis of Questions and Answers on UX.StackExchange

¹Samuel BIROŠ, ²Matúš SULÍR

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovakia

¹samuel.biros@student.tuke.sk, ²matus.sulir@tuke.sk

Abstract – Over the last decade, Stack Exchange became a storage of valuable information about information technology. Many researchers study questions and answers on the most popular website among this website network – Stack Overflow. On the contrary, in our research we take a closer look at the UX.StackExchange website, which is offering valuable information about user experience (UX) and topics that are being discussed by designers.

Keywords – questions, answers, Stack Exchange, user experience, design

I. INTRODUCTION

Information technology developers are not seeking new information in printed books anymore. Instead, they exchange their knowledge and experience on sites such as Stack Exchange. The same applies to UX designers, who have separate a website: UX.StackExchange.com. It is a place that leverages the knowledge and expertise of users to provide answers to technical questions. In our work we take a closer look at questions and answers that are being asked by UX developers, we analyze them and use the results to answer our research questions. Even though UX.StackExchange is an incomparably smaller Q&A (questions&answers) website than Stack Overflow, it offers very valuable information. With approximately 106 thousand users, 29 thousand questions, and 76 thousand answers [1] it provides a database of experience and solved problems.

II. RELATED WORK

There are several research papers with a similar focus that analyze Q&A websites. Most of them are focused on the largest site within the Stack Exchange - Stack Overflow. We have summarized the information obtained from these works in this section, as the principle of the Stack Overflow site is the same as the UX website.

The most common subject of research in similar works are topics or areas that users ask about. The question “What are the most common topics asked about by the users?” is included in most research, often targeting only a certain group of users. Rosen and Shihab [2] focus only on mobile developers. While looking for an answer to the question of what mobile software developers ask about, the authors used a topic modeling method resulting in a total of 40 created topics. The resulting model has shown that mobile developers are asking about a number of issues, from device input issues to application distribution.

One of the most cited works dealing with the issue of Q&A websites [3] is dedicated to the general observation and research of all users on Stack Overflow. In the first examined research question authors are likewise asking about areas and topics of the questions asked on the page. The authors obtain data directly from the Stack Overflow database in the form of data feeds. The result of this research question is 40 areas/topics that most frequently occur on Stack Overflow. To achieve this, authors used a topic modeling method, more specifically the LDA (Latent Dirichlet Allocation) method.

Research by Treude et al. [4] is, among others, devoted to categorizing and examining questions on the Stack Overflow based on their answer rate. Examining approximately 38,000 questions from the time of 15 days in 2010, they found that 5,450 questions (14.19%) remained unanswered with the remaining questions ranging from 1 to 23 answers. Approximately the same distribution can be expected for other sites within Stack Exchange, although it should be noted that Stack Overflow is by far the largest site within the network as well as the site with the most active users.

In 2019, a study focusing on the description of UX knowledge [5] was published. This research focuses on community interaction on the same website as we are analyzing in our work – UX on Stack Exchange. The authors used mixed-approach methods, including natural language processing and qualitative content analysis, to collect and analyze nine years of question and answer data. They wanted to analyze concepts and knowledge categories that could characterize the body of UX knowledge and create a view on how UX knowledge shared on the UX website evolved over time. The result is An Emergent UX Vocabulary, which is divided into several main categories according to the content of the site. In contrast to them, we have more specific research questions and we are more focused on quantitative metrics.

III. RESEARCH SETTING

In this section, we provide information about research data, questions, and methodology.

A. Research Data

Stack Exchange makes its data publicly available in form of XML feeds. We used the latest available data dump which was published on 2nd March 2020. The dataset is divided into eight separate XML documents that can be downloaded from the Archive.org website [6]. For our purposes we use mainly Posts.xml file, which contains data about all published posts, as well as the view count, post type, creation date, ID of the user who created each post and some other data, Tags.xml file, which contains mainly tag name and tag count and also Users.xml, where all data about users are stored.

B. Research Questions

In our research we decided to examine the answers to these seven research questions: 1) What are the main topics discussed on the UX.StackExchange? 2) What are the main topics of unanswered questions? 3) Do the unanswered questions have any common features? 4) What types of questions have many views but no answer? 5) Does the presence of the image affect the answer rate of the question? 6) How have UX trends changed over the years? 7) How are the numbers of new posts and users evolving over the years?

C. Research Methods

While examining the research questions we parsed the XML files and processed the data using scripts written in the programming language Ruby. For some questions, we used Latent Dirichlet Allocation for topic modeling. For editing words in titles and bodies of questions to their roots we use stemming and lemmatization methods. The results were summarized using descriptive statistics. The details of the method are different for each research question, so they are mentioned alongside the result description in the following section as necessary.

IV. RESEARCH RESULTS

Now we will present the results of the individual research questions.

RQ1: What are the main topics discussed on the UX.StackExchange?

Firstly, we focused on analyzing tags that users use when publishing questions. From the available data, we chose the Tags.xml file, which contains, among other things, information about the tag name and how many times the tag was used. Based on these data we got 10 the most used tags on UX.StackExchange website, which are stated in Table 1a.

Based on the obtained results, we can say that designers most often ask questions related to *usability*. However, it is a relatively broad concept that is more difficult to define in detail only with the help of tags as they are meant to be more general. The second most used tag is *website-design*, which says that users ask for information related to the design of a website or web applications. The most commonly used tags also include *forms*, which represent the forms and queries associated with them, as well as *gui-design*, which is a design associated with a graphical user interface. The *mobile tag*, which we expected here, is also among the results, as mobile devices are a general trend.

Secondly, we also tried to analyze topics based on the topic modeling method – LDA. We decided to create a total of 5 topics with each topic containing 5 keywords based on the frequently used words in question titles. The results are visible in Table 1b.

Table 1: Main topics on UX Stack Exchange

(a) Topics based on the used tags

Tag	Number of usages
usability	3176
website-design	2635
forms	2499
gui-design	2471
mobile	1608
interaction-design	1605
user-behavior	1503
navigation	1473
buttons	1260
user-expectation	1156

(b) Topics based on topic modeling

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
user	user	good	user	user
ux	good	design	page	good
page	mobile	ux	good	design
button	design	field	button	button
list	display	app	form	mobile

Table 2: The most common tags in unanswered questions and the division of questions in which they are used

Tag	Questions	Answered	%	Unanswered	%
usability	3176	3114	98,05%	62	1,95%
gui-design	2471	2415	97,73%	56	2,27%
website-design	2635	2589	98,25%	46	1,75%
interaction-design	1605	1567	97,63%	38	2,37%
forms	2499	2462	98,52%	37	1,48%
user-expectation	1156	1121	96,97%	35	3,03%
user-behavior	1503	1474	98,07%	29	1,93%
navigation	1473	1445	98,10%	28	1,90%
mobile	1608	1581	98,32%	27	1,68%
ios	704	679	96,45%	25	3,55%

RQ2: What are the main topics of unanswered questions?

In this RQ we examined questions with no answers. The total number of these questions was 587, while the number of tags used in the unanswered questions was 1656 in total. Thus, on average, 2.82 tags were used in one unanswered question.

The list of tags with most unanswered questions is in Table 2. We can see that the results include tags *gui-design*, *website-design*, *mobile*, and *usability*, which are also among the most frequently used tags in general. Up to four of the five most common tags match both answered and unanswered questions.

It is clear from the results that although these tags are mostly used in unanswered questions, the same tags are much more widely used in questions that have received at least one answer. Furthermore, the *ios* tag is the only tag (and therefore the topic it represents) that, when looking at the total number of questions and the number of unanswered questions, exceeded the limit of 3.50% for unanswered questions. However, it should also be noted that this tag has been used much less than the other tags stated in the table.

RQ3: Do the unanswered questions have any common features?

In this RQ, we find out whether the unanswered questions show any similar features. We focus on 3 aspects, which are the number of tags used, the number of views of the questions, and the reputation of the post owners.

Number of tags in unanswered questions: While asking a question, the user must add 1 to 5 different tags to categorize the post. We decided to look at the distribution of unanswered questions based on the number of tags used in them. Most questions, 166, contained 3 tags, 164 questions contained 2 tags. This result is similar to the results that show almost three quarters of all questions contain between 2 and 4 marks [4].

Number of views: The second criterion we followed was the number of views the questions earned. We worked with the Posts.xml file and monitored the parameter showing the number of views of the question. We divided this data into six categories as shown in Fig. 1.

An interesting finding from the Fig. 1 is that none of the unanswered questions has 0–1 views. By merging questions with none and 1 view, we wanted to eliminate the possibility that after adding a question, the given question can be displayed directly to the user, thus setting view count number to 1.

More than 70% of unanswered questions have at most 100 views, while the largest portion of answered questions has more than 500 views. The results in Fig. 1 clearly show a significant discrepancy in the number of views of answered and unanswered questions.

Users reputation: The reputation of users represents how active they are on the site. Therefore we tried to find out if there is a correlation between the reputation of the user asking the question and whether the question gets any answers or not. The result and comparison of answered and

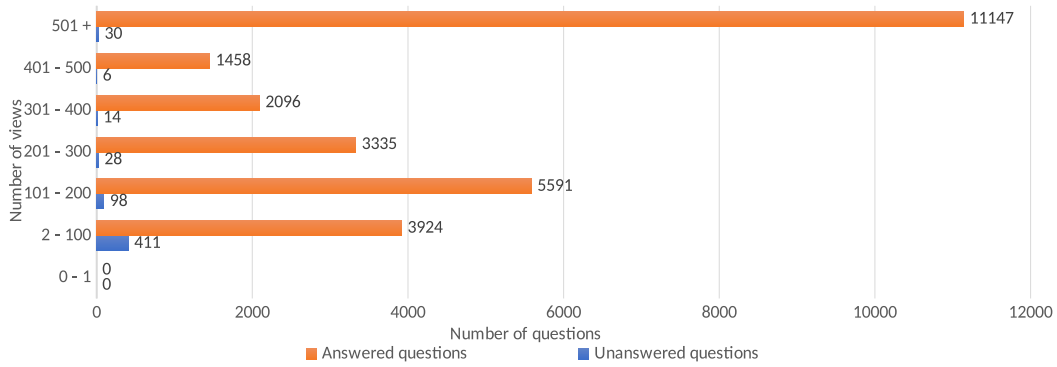


Fig. 1: Comparison of answered and unanswered questions based on view count

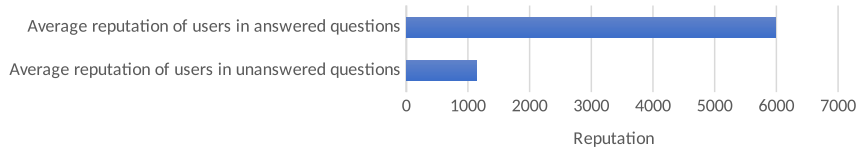


Fig. 2: Comparison of the average value of the reputation of users with unanswered and answered questions

unanswered questions based on the reputation of the users who asked the question is visible in Fig. 2. For unanswered questions, the average user reputation was only 1150, while for questions answered the value was as high as 5982. The difference is so striking that it suggests that the answer rate of the question may indeed be affected by the reputation of the users (or vice versa).

RQ4: What types of questions have many views but no answer?

As we found out, more than 70% of unanswered questions have up to 100 views. Based on this result, we decided to examine unanswered questions with more than 100 views, which we can consider to be higher than usual. The purpose of our research at this point is to find out which questions have many views but no answer and whether this phenomenon is influenced by the topic of the question.

From all the questions that meet our definition, we tried to find out which full words (words conveying meaning on their own) occur most often. The stemming method was applied to these words. First, we worked with the title of the questions, which is the basic text in the question. The results are visible in Table 3a. Second, we took into account the body of the question, which is an expanding text in which the asking user defines his problem in more detail. We proceeded as in the case with the titles and the results are visible in Table 3b.

In general, the most common words in unanswered questions with many views are *user*, *design*, and *app*. However, it is understandable that users are mostly asking about ambiguities associated with users and design. An interesting finding is that in both cases the word *app* also appeared, which showed that these questions are often directed to web or mobile applications. An equally interesting result is the presence of the word *best* in the results of the question titles. This word strongly suggests a high subjectivity of questions when users ask for the opinions of other users.

RQ5: Does the presence of the image affect the answer rate of the question?

For other Stack Exchange websites, such as Stack Overflow, probably the most important aspect of the question is the source code snippet or another demonstration. However, since the designers

Table 3: The most common full words in the titles and bodies of unanswered questions with more than 100 views

(a) in titles		(b) in bodies	
Word	Number of occurrences	Word	Number of occurrences
design	25	user	241
user	17	design	90
best	15	button	74
ux	14	imag	72
app	11	app	71
ui	10	need	68
mobil	10	about	68
select	9	page	62
ui	9	make	61
data	8	list	60

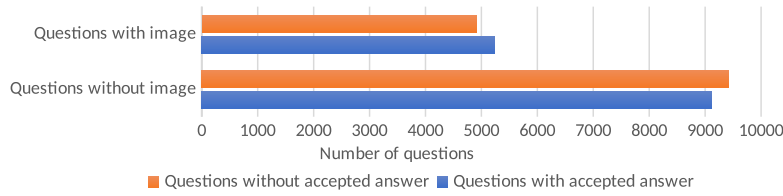


Fig. 3: Comparison of questions with and without an accepted answer based on the presence of an image

usually do not have problems with the code, we focused on tracking the presence of the image. We decided to compare questions with and without an accepted answer and analyzed the presence of an image (as an illustration). We found 14,368 questions with an accepted answer, from which:

- 9,123 questions did not contain an image as an illustration.
- 5,245 questions contain an image as an illustration.

Also, we found a total of 14,344 questions without an accepted answer, which means that the questions are divided in about half according to whether they have an accepted answer or not. Of the total number of questions without an accepted answer:

- 9,430 questions did not contain an image.
- 4,914 questions contain an image.

Thus, as can be seen from the results in Fig. 3, both groups have approximately the same percentage of questions with and without an example in the form of a picture. Therefore, we cannot assume that the presence or absence of an image has any direct connection with the answer rate.

RQ6: How have UX trends changed over the years?

When monitoring trends and their development within the UX website, we decided to monitor specific tags and compare them within a group of other “competing” tags. We monitor the development of the use of tags for the years 2010 - 2019 for the following groups:

- a group of tags representing basic application types,
- a group of brands representing the most used mobile platforms,
- a group of tags representing common design types.

All results for these three groups and linked tags are stated in Fig. 4. An interesting thing that can be seen from all of these figures is that for all monitored tags there is a visible decline in

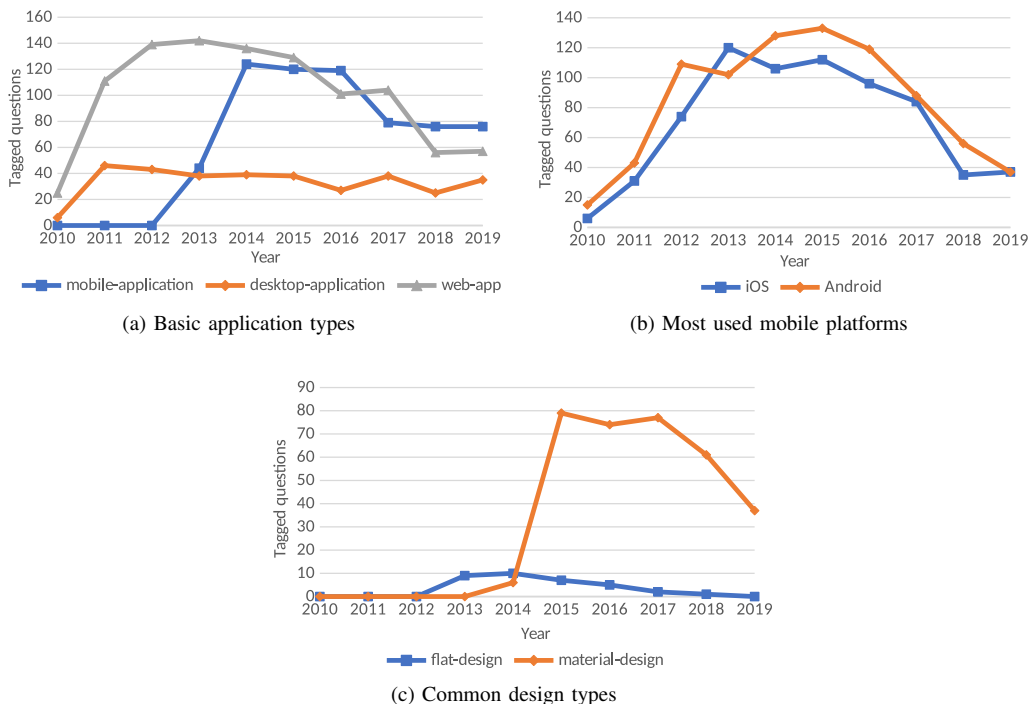


Fig. 4: Trends in years 2010–2019 according to tags used on UX Stack Exchange

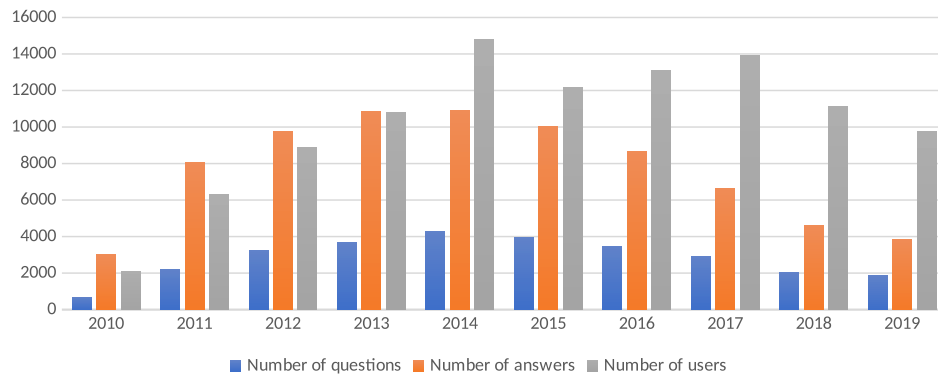


Fig. 5: Development of the number of new questions, answers, and users in 2010–2019

their usage in the recent period, approx. from 2017. This led us to the last research question, the results of which we are showing in the next subsection.

RQ7: How are the numbers of new posts and users evolving over the years?

We decided to monitor how the total number of new posts added and the number of new users registered are evolving between 2010 and 2019.

In Fig. 5 it is understandable that the total number of answers exceeds the total number of questions, as more than one answer is expected in one question. An interesting finding, however, is that while in 2014 the community experienced probably its best period in terms of the total number of added posts, in 2019 these numbers reached approximately the level of when the community was formed. Therefore we can say that the lower number of tags usage is probably related to the overall reduced number of published posts.

Looking at the ever-decreasing number of new questions and answers in recent years, the question of what is the trend in user registrations also arises. As we can see in Fig. 5, although a lower number of new registrations is noticeable in 2018 and 2019 compared to the previous year, this decrease is not as dramatic as with new posts.

V. CONCLUSION

In this article, we defined several research questions connected with the UX website on Stack Exchange. With the help of the data from the website and the created scripts, we brought specific answers to the questions and results in the form of tables, graphs, and verbal descriptions. In these results, we offered the main topics discussed on the UX Stack Exchange website using two independent research methods, but also the main topics of unanswered questions, the main characteristics of questions that remained unanswered or the development of trends and topics on the web over the years and other significant results. These can be used to examine UX as a discipline more closely.

ACKNOWLEDGMENT

This work was supported by Project VEGA No. 1/0762/19 Interactive pattern-driven language development.

REFERENCES

- [1] “All Sites - Stack Exchange,” <https://stackexchange.com/sites?view=list#traffic>.
- [2] C. Rosen and E. Shihab, “What are mobile developers asking about? A large scale study using Stack Overflow,” *Empirical Software Engineering*, vol. 21, no. 3, pp. 1192–1223, 2016.
- [3] A. Barua, S. W. Thomas, and A. E. Hassan, “What are developers talking about? An analysis of topics and trends in Stack Overflow,” *Empirical Software Engineering*, vol. 19, no. 3, pp. 619–654, 06 2014.
- [4] C. Treude, O. Barzilay, and M.-A. Storey, “How do programmers ask and answer questions on the web? (NIER track),” in *Proceedings of the 33rd International Conference on Software Engineering*, ser. ICSE ’11. New York, NY, USA: Association for Computing Machinery, 2011, p. 804–807.
- [5] Y. Kou and C. M. Gray, “A practice-led account of the conceptual evolution of UX knowledge,” in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI ’19. New York, NY, USA: Association for Computing Machinery, 2019.
- [6] “Stack Exchange Directory Listing,” <https://archive.org/download/stackexchange>.

Brief Analysis of Game Development Stack Exchange

¹Natália BAVOĽÁROVÁ, ²Matúš SULÍR

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovakia

¹natalia.bavolarova@student.tuke.sk, ²matus.sulir@tuke.sk

Abstract – Many research papers are studying Stack Overflow (SO) – the most popular Q&A (questions&answers) website among professional or amateur developers. But not many of them are talking about Stack Exchange in general or its subsites. This paper analyses questions and answers on the Game Development Stack Exchange (GDSE) website. Questions about topics, trends, users and other questions are answered. We also found that users with higher reputation do not receive more responses in general.

Keywords – Game development, questions, answers, Stack Exchange, Stack Overflow

I. INTRODUCTION

Stack Overflow is widely used by computer science students, senior programmers, or simply coding enthusiasts. This website is part of a big group of different sites all belonging under Stack Exchange. Stack Overflow is arguably the most used and visited one with over 12 million users and 19 million questions¹. Plenty of researchers analyzed Stack Overflow and similar Q&A websites to observe the most discussed topics [1], [2], the efficiency of badges [3], [4], or unanswered questions [5], [6]. Some of them focused on categorizing questions [7] and others focused mainly on user behavior, such as the usage of tags, how users answer questions [8], their overall activity [9], activity in relation to their reputation [10], or when are users more willing to do revisions [11]. Others focused on code examples in questions [12].

In contrast to the mentioned papers, this paper analyses the Game Development sub-site of Stack Exchange, which – from its official description – is a Q&A site for professional and independent game developers.

II. RESEARCH SETTING

In this section, based on related work, research questions are presented. Then, research data used to answer research questions are detailed. Last, the research methods are described.

A. Research Questions

RQ1: What are the main discussion topics asked by game developers? There are many articles analyzing most discussed topics on the Stack Overflow site but not many of them are analyzing other information-technology related web sites on Stack Exchange. Although SO contains questions from various programming topics including game development, there is a website created specifically for game developers. Game Development Stack Exchange is highly concentrated with skilled developers as well as junior developers. Most discussed topics on such a website can help us understand changing trends in technologies and programming languages – which ones are gaining and which ones are losing popularity.

RQ2: What percentage of questions on the Game Development Stack Exchange website are answered and unanswered? The reason behind the existence of so many Q&A websites is people looking for answers and expecting to find a satisfying answer. If there is no such answer, what is the point then? On an ideal Q&A website, there is a high-quality answer for every question. But in the real world, the answer rate is much lower than 100%. In the case of Stack Overflow, the answer rate is around 70%². By analyzing unanswered questions we can find similarities

¹<https://stackoverflow.com/sites?view=list#users>

²<https://stackoverflow.com/sites?view=list#traffic>

Table 1: Comparison of SO and GDSE ²

	Questions	Answers	Answer Rate	Users	Visits/day	Questions/day
SO	19m	29m	70%	12m	10m	8.8k
GDSE	48k	73k	86%	107k	25k	15

and differences with the answered ones. Then the recommendations for writing questions can be improved to increase the answer rate.

RQ3: Which questions are inactive since last year and are not successfully answered? An answer is considered successfully answered if there is an accepted answer. The authors of questions can mark the best and the most helpful answer as accepted. Stack Overflow is using badges to motivate users to answer or edit old and inactive questions. This system can help answer old questions and increase the overall answer rate. In this question, we are searching for inactive questions that were not successfully answered. In addition, we examine these questions and we detect whether they were edited or not. Editing questions can increase their quality, which can lead to them being answered.

RQ4: What percentage of registered users are inactive since registration? For Q&A websites it is important to have active users, the more active the better. The activity of users is frequently discussed in similarly related analyses. In this question, we are analyzing the activity of registered users since their registration. We are also analyzing the connection of users' reputation to their activity. How many users asked only one question? The reputation of how many users is one? We are not only studying the least active users but also the most active ones.

RQ5: Do users with a higher reputation receive more answers to their questions? Is the reputation of users affecting other aspects, for example, responses to their questions? In this question, we are examining the relation of these variables.

B. Research Data

Stack Overflow currently has around 12 million registered users and 19 million questions ². On the other hand, Game Development Stack Exchange website has significantly fewer users and questions. We can see the comparison in Table 1.

From this table, it is clear that we will work with a smaller amount of data compared to Stack Overflow. For example, GDSE has almost 400-times fewer questions. However, the answer rate on this website is 16% higher. The difference in the number of users is more than 100 times, in the number of daily visits more than 400 times. The most striking difference is in the number of questions per day, with almost 500 times more questions per day on SO than on GDSE. All data used is downloaded from the Stack Exchange archive, where the latest files are published every few months ³. The available file package includes XML files (Badges.xml, Posts.xml, Tags.xml, Users.xml, etc.). The analyzed files contain data of actions performed on the Game Development Stack Exchange website from 2009 to March 2020.

C. Research Methodology

To answer research questions we used scripts in the Ruby programming language. To calculate correlation, Pearson's and Kendall's correlation coefficient was used. As a preparation of the data for topic modeling we used lemmatization and for the topic modeling itself, Latent Dirichlet Allocation (LDA) was used. To visualize the results in graphs we used the Pyplot library. The details of the method used to answer a particular research question are mentioned in the following section whenever necessary.

III. RESULTS

In this section, we summarize the results of our research questions.

RQ1: What are the main discussion topics asked by game developers?

First, we analyzed the main discussion topics based on the tags used in each question. Our results show that ten most used tags are unity (11 739), c# (6528), opengl (3948), c++ (3798), java (3240), xna (3065), 2d (2961), libgdx (2200), collision-detection (2160) and android (2144). We analyzed changes of these tags from 2015 to 2019, as shown in Figure 1.

According to the results from 2019, the popularity of tags java, libgdx, and android decreased. These three tags, which were often used in other years, were outperformed by new tags. Tags

³<https://archive.org/download/stackexchange>

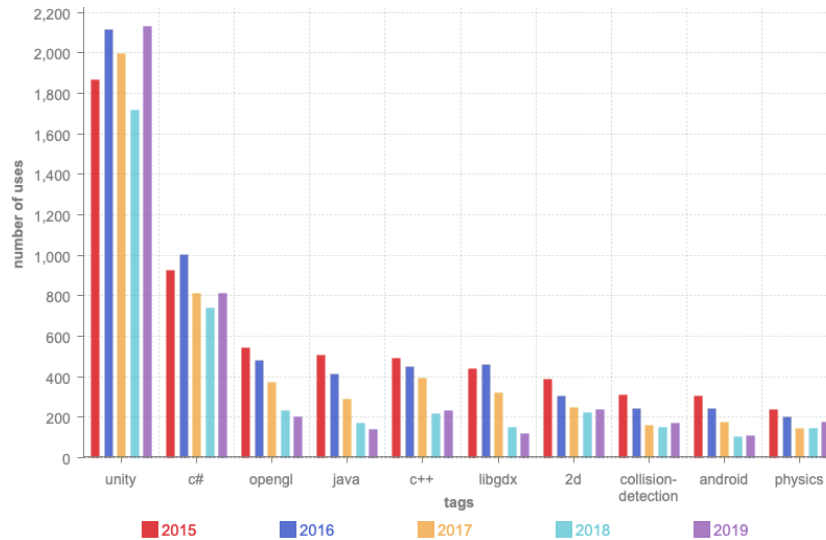


Fig. 1: Usage of top 10 tags in 5 years.

unreal-4, mathematics, and animation appeared on the leading positions in 2019. To summarize, occurrences of tags unity and c# increased in 2019 compared to the previous year. There was a small increase also in the usage of tags c++, 2d, collision-detection, and physics. The remaining tags from the top ten had a declining number of appearances. Out of programming languages, C++, C#, and Java were the most popular. Out of game engines, Unity, Unreal Engine, GameMaker, and Godot were the most used ones.

After analyzing the questions based on the used tags we tried to categorize the questions based on their first word. We came to the conclusion that the most frequent starting word is 'how' (26.53% of all questions). This probably means the users often ask for guidance on how to perform something.

Finally, we used topic modeling (LDA) to find the main topics from the titles and bodies of the questions. We list the word groups determined by the algorithm and our interpretation of them:

Topics from the titles of questions (2012-2020):

- unity game android animation scene change working – *change of scene using animation*
- object unity camera position movement rotation player – *movement of camera and player*
- unity map render image sprite create libgdx – *create and render image/sprite*
- game based engine collision system detection server – *collision detection system*
- game libgdx implement algorithm grid collision loop – *implementing collision algorithm*

Topics from the bodies of questions (2012-2020):

- texture shader render opengl draw vertex code – *code for texture rendering*
- player game collision move code server character – *collision of characters*
- map tile game point make grid terrain – *game map*
- camera object position rotation vector matrix code – *transformation of objects*
- game unity code screen file class project – *class in unity project*

Game, player, and object were among the most common words. After all, all the questions on this website are about game development. In comparison with the analysis of topics based on tags, we can see a certain overlapping of results. In both methods, the words Unity, collision (detection), and Android occur. The main difference is that in the analysis using tags, mostly expressions describing a specific technology or a specific programming language are in the foreground.

RQ2: What percentage of questions on the Game Development Stack Exchange website are answered and unanswered?

We can divide all questions on GDSE website into three groups – answered (with at least one answer: 34.01%), successfully answered (with an accepted answer: 54.06%) and unanswered (11.93%). We took a deeper look at the unanswered questions and the tags used in them. The results show similarities between these tags and the top 10 used tags, as it is shown in Figure 2.

We inspected the presence or rather the absence of a code preview in unanswered questions. However, according to the results, 47.55% of unanswered questions contained a code preview and only 45.62% of answered questions had it.

As a last possible reason for these questions not being answered, the length of individual questions was analyzed. The results showed that the length of unanswered questions slightly

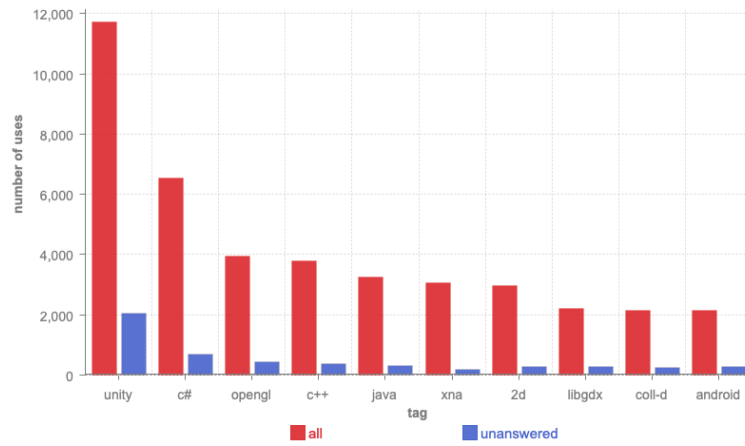


Fig. 2: Comparison of tags in unanswered questions and tags used in all questions.

deviates from the average. The reason for these questions not having any answers could be an inappropriate length of the question (too short or too long).

RQ3: Which questions are inactive since last year and are not successfully answered?

In this RQ we tried to find questions that have not been successfully answered and also did not have any activity in 2020. We excluded questions with activity in 2020 because their fairly recent activity made it impossible to categorize them yet. We also excluded successfully answered questions (for obvious reasons) and closed questions. This led us to 18 259 inactive questions from which 4512 did not have an answer. We analyzed inactivity of these questions in relation to the edits made on them. Edits and corrections from users other than the author can help increase the intelligibility and readability of this question. Our results showed that 71.04% of these inactive questions did not have any edits made on them (from other users).

RQ4: What percentage of registered users are inactive since registration?

The fourth research question addressed the activity of users. The results of this question indicate that 21.21% of registered users are inactive since registration and 78.79% of users have been or are regularly active. User's activity is also reflected in the value of their reputation. There are 46,248 out of 104,769 users with a reputation value of 1. The highest reputation achieved by a single user is 94,364. Users with only one question are 13,909 altogether and the most active user had 105 questions. The biggest group of users based on their reputation is one where users had a reputation of 101–1000. This group includes 68.51% of users and also contributed the largest number of questions (33,328). The second largest group consisted of users with a reputation of 1–100, where 29.47% of users belonged, and they published 12,589 questions. Users with a reputation higher than a thousand represent only 2.02% of all users.

RQ5: Do users with a higher reputation receive more answers to their questions?

The user's reputation should reflect his expertise and ability to provide not only helpful answers but also appropriately asked questions. In this research question, we calculated the correlation coefficient between the value of the user's reputation and the number of answers to the user's questions. First, we visualized data from 2019 to see whether we work with linear or curved data – as we can see in Figure 3.

We used Pearson's (0.03578) and Kendall's (0.0000285) correlation coefficient to find the results. Based on these numbers we can declare that there is no relation between these two values.

IV. THREATS TO VALIDITY

Internal threats to the validity of our study include the used libraries (gems). Despite efforts to use verified gems, it is possible that some of them may have negatively affected the validity of our results. Another threat to internal invalidity is the use of the topic modeling method. In this method, it can be difficult to estimate how many topics with what range of words is ideal to generate. Even by testing and evaluating results with various variables, it is possible that the results may not be accurate.

The data available from the Game Development Stack Exchange page should be mentioned as an external threat to validity. Although the data itself should be accurate, only certain parts

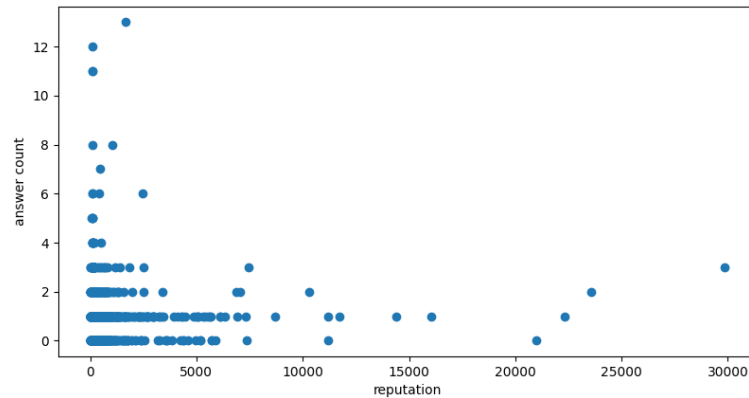


Fig. 3: The relation between the user's reputation and the number of responses.

(year ranges) of this dataset have been used. Using a different range of data in individual research questions could lead to different results.

V. RELATED WORK

Barua et al. [1] were finding answers to questions about most discussed topics, changing interest of developers over time, or correlation of question in one topic with answers in another one. The authors also contemplate the use of tags in users' posts. They come to the conclusion that tags are useful but not perfect. A work by Bajaj et al. [2] focuses specifically on web developers. Similarly to the previous paper, the authors try answering questions about the most discussed topics, the most important topics related to web development, or the existence of temporary trends in the web development discussions.

A study by Bornfeld and Rafaei [3] shows the efficiency of badges on Q&A websites such as Stack Overflow. Based on their results, with the introduction of simple leader boards, users became a lot more active. They found that 18 out of 20 badges had a positive effect, three a negative effect and one was not effective at all. Another research dealing with badges [4] came to equally interesting findings. User behavior on the website was studied and a pattern based on this typical behavior was found. With badges, users tended to break their routine to gain a new badge. After this conclusion, it seems fairly easy to motivate users to do unwanted or unpopular activities.

Asaduzzaman et al. [5] focused on unanswered questions. They found out that fifty percent of all questions are answered in fifteen minutes. On the contrary, if a question did not receive an answer in one month, they considered it unanswered. They tried to find reasons and factors affecting answer rate such as: answer being too short, vague, difficult to understand, a specific program without code sample or correct explanation, too difficult, too specific, or very time consuming, too specific technology with only a few experts in that field, and lastly, questions from impatient, inconsistent, or reckless members. Based on these factors they were able to categorize questions into five groups. Then they tried to predict how long will every question remain unanswered. In a similarly oriented article [6], the authors came to the conclusion that if a question does not receive an answer in a few days, the probability of receiving one is very low. Even if an answer appears then, the likelihood of this answer being helpful and useful is not very high.

Allamanis a Sutton [7] chose a different approach. They categorized questions based on similarities and assigned related tags to each category. Treude et al. [8] observed that questions have 2–4 tags on average. They also distinguished questions based on the existence of an accepted answer. If a question does not have such an answer, this question is considered unsettled.

Regarding the activity of users, Slag et al. [9] used clustering to determine groups of users based on their activity (number of posts and reputation). The smallest group of the most active users created most of the content. A slightly bigger group was somewhat less active but still contributed to some extent. The biggest group comprised of the least active users that contributed with little to no posts. Then they identified a group of users called "one-day flies". These users asked only one question in their "lifetime". Movshovitz-Attias et al. [10] analyzed the activity of users in relation to their reputation. Stack Overflow changed its scoring rules in 2009–2010 which affected all registered users. This change was meant to reward "answerers" more than "askers". The authors also presented a thought that reputation can be considered a user's degree of qualification and reliability.

The importance of users' revisions is discussed in an article by Wang et al. [11]. They found out that users made significantly more revisions and corrections on badge days than on other days. If users made more corrections per day, they were rather smaller ones and simpler adjustments (such as text correction or code formatting). But it was users who made multiple corrections in one day who were more prone to making mistakes. These corrections were then rejected. The authors expressed a thought that the reassessment and introduction of new rules for awarding these badges could lead not only to a quantitative increase in corrections (as has been the case so far) but also to a qualitative one.

VI. CONCLUSION AND FUTURE WORK

The Game Development Stack Exchange site has a higher answer rate compared to the Stack Overflow page. However, in case of user activity, the number of their posts, or their reputation, SO is superior in all of these aspects. This is understandable, given the size of this website. The larger website has more users and daily visits. These users offer others the opportunity to respond, vote, revise, and edit posts to increase their reputation. Although users on GDSE do not achieve as many posts and do not have a reputation as high as those on Stack Overflow, they cannot be labeled inactive. The largest percentage of users has a reputation in the range of 101-1000. This means that these users are active in adding questions, commenting, and answering to some extent.

This work offers a general view of the Game Development Q&A website and can serve as a basis for further in-depth analysis. In addition to modeling the topics on the titles and text of the questions, this method could also be used on answers. The topics generated from answers can be then examined and compared with those generated from questions. Despite multiple tests and adjustments in the number of words, the size of the intervals, and the number of generated topics, it is possible that results from topic modeling can be improved. Scripts can be later executed with the most recent data to get the most up-to-date results. It is also possible to modify existing questions or add new ones according to current needs.

ACKNOWLEDGMENT

This work was supported by Project VEGA No. 1/0762/19 Interactive pattern-driven language development.

REFERENCES

- [1] A. Barua, S. W. Thomas, and A. E. Hassan, "What are developers talking about? An analysis of topics and trends in Stack Overflow," in *Empirical Software Engineering*, vol. 19, no. 3, 6 2014, pp. 619–654.
- [2] K. Bajaj, K. Pattabiraman, and A. Mesbah, "Mining questions asked by web developers," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, ser. MSR 2014. New York, NY, USA: ACM, 2014, pp. 112–121.
- [3] B. Bornfeld and S. Rafaeli, "Gamifying with badges: A big data natural experiment on Stack Exchange," *First Monday*, vol. 22, 06 2017.
- [4] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec, "Steering user behavior with badges," in *Proceedings of the 22nd International Conference on World Wide Web*, ser. WWW '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 95–106.
- [5] M. Asaduzzaman, A. Mashiyat, C. Roy, and K. Schneider, "Answering questions about unanswered questions of Stack Overflow," in *IEEE International Working Conference on Mining Software Repositories*, 05 2013, pp. 97–100.
- [6] L. Yang, S. Bao, Q. Lin, X. Wu, D. Han, Z. Su, and Y. Yu, "Analyzing and predicting not-answered questions in community-based question answering services," in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, ser. AAAI'11. AAAI Press, 2011, p. 1273–1278.
- [7] M. Allamanis and C. Sutton, "Why, when, and what: Analyzing stack overflow questions by topic, type, and code," in *10th Working Conference on Mining Software Repositories (MSR)*, 5 2013, pp. 53–56.
- [8] C. Treude, O. Barzilay, and M.-A. Storey, "How do programmers ask and answer questions on the web? (NIER track)," in *Proceedings of the 33rd International Conference on Software Engineering*, ser. ICSE '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 804–807.
- [9] R. Slag, M. d. Waard, and A. Bacchelli, "One-day flies on StackOverflow – why the vast majority of StackOverflow users only posts once," in *IEEE/ACM 12th Working Conference on Mining Software Repositories*, 2015, pp. 458–461.
- [10] D. Movshovitz-Attias, Y. Movshovitz-Attias, P. Steenkiste, and C. Faloutsos, "Analysis of the reputation system and user contributions on a question answering website: StackOverflow," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ser. ASONAM '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 886–893.
- [11] S. Wang, T.-H. P. Chen, and A. E. Hassan, "How do users revise answers on technical Q&A websites? a case study on Stack Overflow," *IEEE Transactions on Software Engineering*, vol. PP, pp. 1–1, 10 2018.
- [12] S. M. Nasehi, J. Sillito, F. Maurer, and C. Burns, "What makes a good code example? A study of programming Q&A in StackOverflow," in *28th IEEE International Conference on Software Maintenance (ICSM)*, 2012, pp. 25–34.

Návrh systému pre testovanie simulovania lokalizácie mobilných robotov

Ján VAŠČÁK

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská Republika

jan.vascak@tuke.sk

Abstrakt – Obsahom tejto práce je poukázať na problematiku lokalizácie robotov. Úloha bola zrealizovaná prostredníctvom simulačného prostredia *Webots*, kde sme zvolili dva modely mobilných robotov. V rámci riešenia tejto problematiky sa práca zaoberá návrhom meraní vo virtuálnom prostredí, jeho modelovaním ako aj návrhom samotného navigačného algoritmu. Napokon sa práca venuje samotnej realizácii návrhu a simulácii vyvinutého riešenia. Následne sú popísané vykonané experimenty a ich vyhodnotenie.

Príučové slová – lokalizácia, navigácia, robot, simulácia

I. ÚVOD

Na začiatku dvadsiateho storočia s rozvojom priemyslu, vedy a techniky sa roboty začínajú čoraz viac využívať aj v každodennom živote. Ďalším vývojom vznikajú nové odvetvia robotiky ako aj nové možnosti použitia, roboty už nie sú len na linkách vo fabrikách, ale prichádzajú do bezprostredného styku s ľuďmi. Do začiatku deväťdesiatych rokov tieto koncepty narážali na technické limity v oblasti hardvéru a softvéru. Tieto problémy boli ošetrené vývojom výkonných riadiacich systémov, čo značne urýchlilo vývoj ďalších nových technológií. Pokračujúci vývoj senzorov a počítačov zapríčinil, že sa stávajú dokonalejšie a efektívnejšie v každom ohľade a medzi tie najvýznamnejšie patria presnosť merania, výkonnosť a hlavne cenová dostupnosť. Toto viedlo k rozšíreniu mobilných robotov do skladov a na rôzne ďalšie pracoviská, v ktorých sa pohybujú aj pracovníci a iné objekty [1]. Prechod na mobilných robotov prináša so sebou viaceré problémy. Jedným z najväčších z nich je lokalizácia presnej polohy robota v danom priestore a navigácia, teda určenie bezpečnej cesty z bodu A do bodu B. Na lokalizáciu väčšinou používame GPS, ale keďže hlavným pôsobiskom mobilných robotov je interiér budov, musia sa použiť iné postupy určenia ich polohy. Proces navigácie úzko súvisí s lokalizáciou, pretože na stanovenie kurzu potrebujeme vedieť súčasnú polohu, polohu nášho cieľa a ako aj možných prekážok na trase medzi nimi [2]. V súčasnej dobe sú mobilné roboty vybavené veľkým počtom snímačov, ako sú RGBD kamery, 3D laserové diaľkomery alebo ultrazvukové senzory, čo im poskytuje sa čoraz lepšie pohybovať v dynamickom prostredí. Veľkou výhodou týchto periférií a technológií je vysoká adaptívnosť na aktuálne prostredie. Na základe týchto, ale aj rôznych iných snímačov sa rozvíjajú ďalšie spôsoby ovládania mobilných robotov. V tejto práci sme sa zamerali konkrétne na využitie dvoch prístupov, a to odometrie a merania zrýchlenia (rotačného a pozdĺžneho) na určenie polohy (lokalizáciu) mobilného robota.

II. POUŽITÉ TECHNOLOGIE

V ďalšom si uvedieme použité programovacie ako aj technické prostriedky, ktoré sa využívajú pri vnútornej lokalizácii mobilných robotov. Táto práca je síce postavená na simulácii lokalizačného procesu, ale pri jej návrhu musíme poznať vlastnosti snímačov, ktorých údaje by sa v reálnej situácii využívali.

A. Simulátor *Webots*

Webots je bezplatný a open-source 3D robotický simulátor používaný v priemysle, vzdelávaní a výskume. Projekt *Webots* sa začal pôvodne vyvíjať v roku 1996 Švajčiarskom federálnom technologickom inštitúte v Lausanne. *Webots* obsahuje veľkú zbierku voľne modifikovateľných modelov robotov, senzorov, akčných členov a predmetov. Okrem toho je tiež možné vytvárať

nové modely od úplného počiatku. Pri navrhovaní modelu robota užívateľ špecifikuje grafické aj fyzikálne vlastnosti objektov. Grafické vlastnosti zahŕňajú tvar, rozmery, polohu a orientáciu, farby a textúru objektu. Fyzikálne vlastnosti zahŕňajú hmotnosť, koeficient trenia, ako aj konštanty pružiny a tlmenia. V softvéri je prítomná jednoduchá dynamika tekutín. *Webots* používa knižnicu *Open Dynamics Engine* (ODE) na detekciu kolízií a simuláciu dynamiky tuhého telesa, ako je rýchlosť, zotrvačnosť a trenie [3]. Ďalšou výhodou tohto prostriedku je otvorenosť voči viacerých programovacím platformám ako C, C++, Python, Java, MATLAB alebo ROS s jednoduchým API pokrývajúcim všetky základné potreby robotiky.

B. Určovanie polohy pomocou odometrie

Jedným zo základných spôsobov, ako roboty určujú svoju polohu vo svojom prostredí, je odometria. Táto metóda využíva pohybové senzory na určenie zmeny polohy robota vzhľadom na niektorú známu polohu. Napríklad, ak robot sa pohybuje po priamke a ak pozná priemer svojich kolies, môžeme spočítaním počtu otáčok kolesa určiť, ako ďaleko prešiel. Roboty často majú k hnacím kolesám pripojené snímače hriadel'a, ktoré vydávajú pevný počet impulzov na otáčku. Počítaním týchto impulzov môže procesor odhadnúť prejdenú vzdialenosť. Odometria má však svoje obmedzenia. Pretože ide o kumulatívne meranie, s postupom času sa akákoľvek chyba snímania zvýši. Roboty môžu pravidelne potrebovať ďalšie senzory na presné určenie polohy robota, aby sa zabránilo nadmernému zvyšovaniu chýb [4].

C. Určovanie polohy pomocou zrýchlenia

Ak poznáme počiatočnú rýchlosť a smer pohybu, tak pomocou zrýchlenia a jeho zmien vieme tiež určiť polohu robota v nasledujúcom časovom kroku. Keďže ale zrýchlenie môže byť rotačné ako aj pozdĺžne, tak na tento účel potrebujeme dva druhy sníčov, a to gyroskop a akcelerometer. Výhodou tohto prístupu je fakt, že lokalizáciu môžeme rozšíriť na 3D priestor, čo v plnej miere dokážeme v prípade dronov.

Gyroskopický senzor, alebo skráteno gyroskop, meria uhlovú rýchlosť otáčania a dodáva informáciám poskytovaným akcelerometrom ďalší rozmer sledovaním rotácie alebo krútenia. V dnešnej dobe sa používa veľa typov gyroskopov [5]. Najčastejšie používanými zariadeniami sú MEMS, kde môžeme presne zmerať polohové uhly. Tieto zariadenia sú najužitočnejšie pri lietajúcich objektoch, kde malá zmena polohy môže spôsobiť veľké zmeny a používa sa tiež tam, kde potrebujeme zmerať polohu a použiť ju ako vstup do systémov spätnej väzby alebo riadiacich systémov. Napriek svojej jednoduchosti v dizajne sa stali nevyhnutnými súpravami pre čokoľvek od námornej dopravy po raketoplán a samozrejme roboty.

Akcelerometer je elektromechanické zariadenie používané na meranie zrýchľovacích síl. Takéto sily môžu byť statické, ako napríklad nepretržitá gravitačná sila alebo, ako je to v prípade mnohých mobilných zariadení, dynamické na snímanie pohybu alebo vibrácií. Akcelerometer meria pozdĺžne (lineárne) zrýchlenie pohybu. To znamená, že akcelerometer zmeria smerový pohyb zariadenia, ale nebude schopný presne určiť svoju bočnú orientáciu alebo naklonenie počas tohto pohybu, pokiaľ nie je k dispozícii gyroskop na doplnenie týchto informácií.

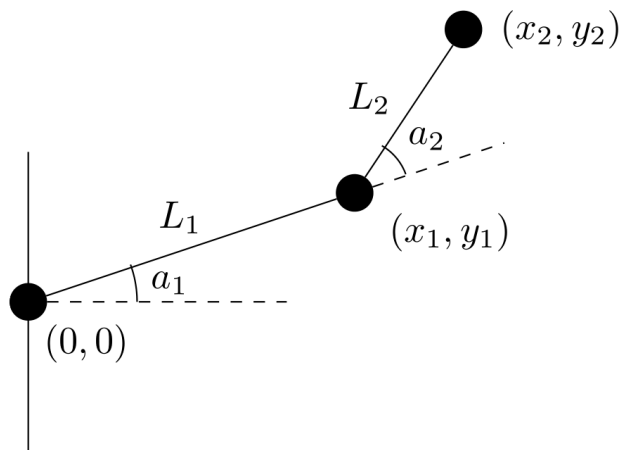
D. Priama kinematika

Úlohou kinematiky je popisovanie správania sa telies a sústav telies bez ohľadu na sily a momenty, ktoré vyvolávajú akýkoľvek pohyb. Inými slovami, jedná sa o popis polohy častí manipulátora a prípadne následnými deriváciami sa môže určiť ich rýchlosť a zrýchlenie. Priamou metódou sa na základe znalosti kĺbových súradníc (natočenia, resp. posuvu) hľadá poloha koncového bodu manipulátora, vid' Obr. 1. Správne vyriešenie priamej kinematiky je dôležitým krokom pre určenie inverznej kinematiky. Pri výpočte polohy efektora je jedinou neznámou iba poloha tohto bodu. Parametre reprezentujúce natočenie v jednotlivých osiach sú všeobecne známe.

E. Modely mobilných robotov

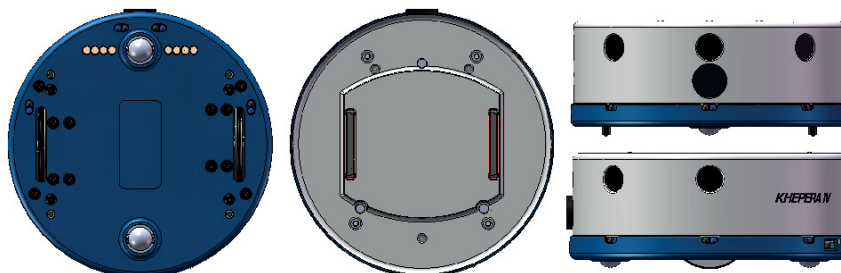
Pre naše experimentálne účely sme využili modely dvoch známych mobilných robotov, ktoré sa vo veľkej miere využívajú vo výskume a vzdelávaní a ktoré sú dostupné v prostriedku *Webots*. Konkrétne sa jedná o roboty *Khepera* a *TurtleBot*.

Robot *Khepera IV* je štvrtou verziou mobilného robota *Khepera* určeného pre vzdelávanie a výskum. Tento kompaktný mobilný robot je navrhnutý pre rovné povrchy. Architektúra robota poskytuje mimoriadnu modularitu a používa systém rozširujúcej zbernice pre takmer neobmedzené konfigurácie. Robot obsahuje pole 8 infračervených snímačov na detekciu prekážok so 4 ďalšími na zabránenie pádu alebo sledovanie čiary ako aj 5 ultrazvukových snímačov na detekciu objektov s dlhým dosahom, vid' Obr. . Robot funguje na operačnom systéme Linux a poskytuje



Obr. 1 Koncept priamej kinematiky.

štandardné prostredie pre vývoj aplikácií. Mobilný robot *Khepera IV* môže byť použitý na výskum a vzdelávanie v rôznych oblastiach, ako sú navigácia, umelá inteligencia, multiagentové systémy, riadenie, kolektívne správanie, či programovanie v reálnom čase [6].



Obr. 2 Model robota Khepera IV.

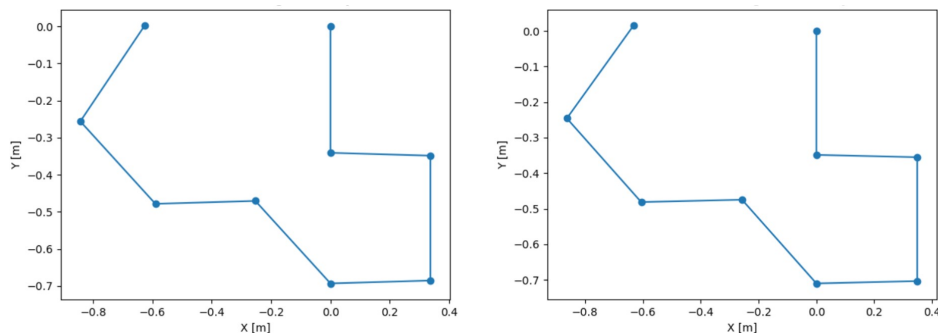
TurtleBot je robot so štandardnou platformou tzv. robotického operačného systému (ROS). Existujú tri verzie série *TurtleBot*. V roku 2017 bol *TurtleBot 3* vyvinutý s funkciami, ktoré dopĺňajú chýbajúce funkcie jeho predchodcov a požiadavky používateľov. *TurtleBot 3* je malý, cenovo dostupný, programovateľný mobilný robot pre použitie vo vzdelávaní, výskume, hobby a produkcii prototypov. Základnou technológiou *TurtleBot 3* je metodológia tzv. simultánnej lokalizácie a mapovania (SLAM), navigácia a manipulácia, vďaka čomu je vhodný pre účely domácich servisných robotov. Cieľom *TurtleBot 3* je dramaticky znížiť veľkosť platformy a znížiť cenu bez toho, aby bolo potrebné obetovať jej funkčnosť a kvalitu a zároveň ponúknuť rozšíriteľnosť.

III. NÁVRH SYSTÉMU NA SIMULÁCIU LOKALIZÁCIE

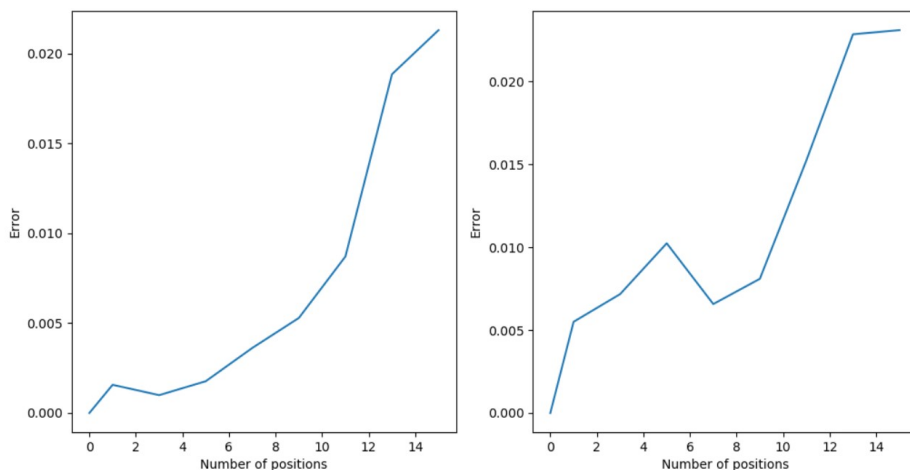
V tejto práci popisujeme návrh matematicko-fyzikálneho modelu softvéru pre mobilné roboty, ktorý využíva prvky odometrie, merania zrýchlenia a priamu kinematiku. Na popísanie priestoru, v ktorom je daný mobilný robot umiestnený, sa použil matematický model dvojdimenzionálneho euklidovského priestoru, ktorý využíva karteziánsku súradnicovú sústavu. Hlavný program pozostáva z počiatočnej inicializácie periférií, sekvencií pohybov, ktoré robot vykoná vo virtuálnom priestore a nakoniec z výsledného prepočtu celkovej lokalizácie do karteziánskej, ale aj cylindrickej súradnicovej sústavy, kde okrem súradníc sa vypočíta aj uhol natočenia koncového bodu z odometrie a merania zrýchlenia voči počiatočnému stavu robota. Výstupom sú relatívne súradnice osí X a Y ako aj relatívne natočenie okolo osi Z voči pôvodnej pozícii z oboch meraní, pričom sa tento výstup vypíše do konzoly. Lokalizácia zanedbáva pozdĺžne (translačné) pohyby v osi kolmej na povrch, teda os Z . Táto os vzhľadom na počiatočný bod nie je cieľom merania. Taktiež zanedbáva rotačné pohyby okolo osi X a Y , ktoré nie sú rovnako ako translačný pohyb v osi Z cieľom merania. Program bol napísaný v interpretovanom multi-paradigmaticom jazyku Python (verzia 3.7.5).

Cieľom bolo lokalizovať robota vo virtuálnom, ale aj reálnom, priestore pomocou dostupných periférií robota a pomocou navrhnutého algoritmu zistiť jeho polohu. Na určenie výslednej polohy

boli použité dve metódy: odometria a meranie zrýchlenia, pričom cieľom bolo porovnať skutočné súradnice v robotickom simulátore *Webots* s vypočítanými súradnicami. Experimenty potvrdili, že presnosť oboch metód je relatívne vysoká. Po ukončení pohybu na porovnanie jednotlivých výsledkov poslušil výpis do konzoly robota, pričom namerané veličiny boli porovnané s „reálnymi“, čiže vypočítanými v simulátore *Webots*. Popri výpise do konzoly sa na konci zobrazí trajektória pohybu robota vykreslená do grafu, Obr. 3 a Obr. 4.



Obr. 3 Získané trajektórie pomocou odometrie (vľavo) a merania zrýchlenia (vpravo).



Obr. 4 Kumulatívna chyba pri odometrii (vľavo) a meraní zrýchlenia (vpravo).

V prostredí *Webots* pri importovaní robota je nutné do programu implementovať modul *controller*, ktorý obsahuje triedu *Robot*. Inštancia triedy *Robot* slúži na získavanie referencií na jednotlivé senzory a aktuátory potrebné pre ovládanie virtuálneho modelu robota *Khepera IV*. Hlavnú triedu programu tvorí back end celého robota, ktorý spája všetky moduly potrebné pre fungovanie programu. Vo front-end-e je vytvorená inštancia objektu triedy *Khepera*. Následne musia byť zavolané tri inicializačné metódy, a to inicializácia aktuátorov, senzorov a samotnej lokalizácie, po ktorých je možné používať metódy zabezpečujúce pohyb: *Forward(time)*, *Backward(time)*, *Right(time)*, *Left(time)*. Back end-ová trieda *Khepera* zahŕňa vlastné moduly s triedami umožňujúcimi výpočet súradníc robota.

IV. ZÁVER

Navrhnutý systém úspešne simuloval mobilného robota *Khepera IV*, ktorý je schopný sa pohybovať v bezbariérovom priestore virtuálneho sveta v prostredí *Webots*. Na základe odometrie a základných senzorov, akými sú gyroskop a akcelerometer bola určená koncová poloha mobilného robota. Systém je jednoducho použiteľný v širokom spektre rôznych druhov mobilných robotov vzhľadom na univerzálnosť vytvoreného algoritmu. V súčasnosti prebiehajú experimenty s modelom robota *TurtleBot*. Takýto aparát je zároveň aplikovateľný aj na reálneho robota, t.j. ak by sme program dokázali nahráť na reálny aparát, namerané výsledky by boli spoľahlivo a prehľadne vykreslené. Využitie senzoričky (gyroskop, akcelerometer, alebo enkóдеры) je síce nevyhnutné na určenie polohy robota, no existujú aj alternatívne metódy umožňujúce dosiahnutie podobných výsledkov. Pri testovaní oboch meraní, teda odometrie a senzormetrie sa podarilo

určiť polohu virtuálneho mobilného robota s presnosťou približne 10^{-2} m po vykonaní rôznych druhov pohybov.

POĎAKOVANIE

Táto práca je finančne podporená Kultúrnou a edukačnou grantovou agentúrou Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky prostredníctvom projektu KEGA č. 033 TUKE-4/2018 – Smart Industry/Architektúry inteligentných informačných a kybernetických systémov.

LITERATÚRA

- [1] Nakju Doh, H. Choset, and Wan Kyun Chung, "Accurate relative localization using odometry," in *2003 IEEE International Conference on Robotics and Automation*, vol. 2, 2003, pp. 1606–1612.
- [2] J. Vaščák and I. Savko, "Radio beacons in indoor navigation," in *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)*, 2018, pp. 283–288.
- [3] Cyberbotics, "Webots: Open source robot simulator," [online] <https://www.cyberbotics.com/>.
- [4] J. Vaščák and J. Hvizdoš, "Vehicle navigation by fuzzy cognitive maps using sonar and RFID technologies," in *2016 IEEE 14th International Symposium on Applied Machine Intelligence and Informatics (SAMII)*, 2016, pp. 75–80.
- [5] V. P. Zhuravlev, S. E. Perelvaev, B. P. Bodunov, and S. B. Bodunov, "New-generation small-size solid-state wave gyroscope for strapdown inertial navigation systems of unmanned aerial vehicle," in *2019 26th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, 2019, pp. 1–3.
- [6] F. Mondada, E. Franzi, and A. Guignard, "The development of Khepera," in *Proceedings of the First International Khepera Workshop*, ser. HNI-Verlagsschriftenreihe, Heinz Nixdorf Institut. 64, 1999, pp. 7–14.

Deep reinforcement learning agents playing nine men's morris

¹Kamil ADAMIŠÍN, ²Michal BUDIŠ, ³Lukáš HRUŠKA, ⁴Jakub KARĽA

¹Department of Cybernetics and Artificial Intelligence, Faculty of Electrical

Engineering and Informatics, Technical University of Košice, Slovak Republic

¹kamil.adamisin@student.tuke.sk, ²michal.budis@student.tuke.sk,

³lukas.hruska@tuke.sk, ⁴jakub.karla@student.tuke.sk

Abstract – This article is describing the issue of creating a neural network to play a table game Nine Men's Morris. To solve this problem, we used a type of learning method called Reinforcement learning, which is based on the principle of reward and punishment, where the agent receives a reward for a correct move and a penalty for bad moves. The goal was to teach the neural network the rules of the game of Nine Men's Morris in the environment we created. To decide appropriately on the next move and bring the game to a winning state. We achieved the best results by implementing the DRQN algorithm and using a multi-agent system.

Keywords – Deep Reinforcement Learning, Nine Men's Morris, Neural Network.

I. INTRODUCTION

The neural networks are becoming more widespread these days than in the few past years in the field of mobile phones, computer games and applications. However, we still face the problem of relatively slow learning. Because of this problem, the development of intelligent agents who can play various games has been an important part. An example is the game of Nine Men's Morris, where two player play against each other, trying to build a mill and remove as many opponent's pieces as possible. To create this type of an intelligent agent, we also need to set up the individual inputs and outputs correctly, as well as the train of neural network, which should be adequate for a real player. This is especially important when playing two-player board games.

II. DEEP REINFORCEMENT LEARNING

Deep Reinforcement Learning (DRL) was created to address one of the main goals in the field of artificial intelligence, it aims to create autonomous agents that interact with the environment in which they find themselves in order to learn the best possible behavior through tests and errors. Learning takes place in such a way that the agent faces the consequences of its actions when selecting and carrying them out. It is able to learn by observing the reward(punishment) credited the environment.

DRL uses deep neural networks to approximate the functions of RL components as a value function, policies or model, which represents a state function and a reward function.

By performing the action, the agent transfers the environment from the current state s_t to the next s_{t+1} . With each change of the environments state, the environment returns feedback in the form of reward or penalty. The goal is to teach the agent a suitable control strategy (called policy) that maximizes the collected reward. In general, these strategies map the distribution of probabilities between actions based on the state of the environment.

Learning takes place on the basis of reward and punishment. Here, the reward function $R_t(s, a)$ is calculated. to prevent the agent from deliberately avoiding to win a game just because he is getting rewards, each reward is discounted using a discount factor $\gamma \in (0, 1)$. This discount factor is applied in every step of environment, so the agent should end the game faster. [1]

$$V_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \quad (1)$$

- V_t - reward function
- γ - discount factor
- k - step
- r_t - reward at the time t

A. Value function

Value function is used to estimate how good the state or the state-action pair is. Represents approximations of functions $\hat{v} = (s; \theta)$ or $\hat{q} = (s, a; \theta)$, where the Q-value is approximated using Q-learning, Deep Q-Network algorithms and their extensions.

$$\forall s \in S : V_{\pi}(s) = E_{\pi} \left[\sum_{t=t_0}^{\infty} \gamma^{t-t_0} r_t | s_t = s \right] \quad (2)$$

where:

- $V_{\pi}(s)$ - value of the state s under policy π
- E_{π} - expected value of a random variable
- r_t - reward at the time t
- γ - discount factor: penalize the actor for taking too many steps.
- $|$ - state of distribution

1) *Q-Learning*: With the Q-Learning algorithm we represent the value function using a so-called Q-table which maps all possible combinations of a state s and an action $Q(s, a)$. Training Q-tables is faster and more effective for winning the games than training Deep neural networks (DNN), but is less suitable for learning e.g. playing chess, as there are more than all possible states and combinations 10^{120} . So the Q-Table would be incredibly large and learning would take a lot of time and become unmanageable.

2) *DQN*: In deep Q-learning we use a multi-layer neural network that approximates the value of each possible action for the current state of the environment. Vanilla DQN algorithm uses convolutional layers in its architecture and needs sequence of images as the input. Therefore the input to the neural network is the state and the output is the Q-value of the action to be performed by the agent. For an n - dimensional space with m number of actions is a function from NSR^n to R^m possible functional rewards. If θ are the parameters of the resulting network, and this network is the same as the connected network with parameters θ , then the resulting network copies each step of the connected network $\theta_{-t} = \theta_t$. Now we can write the goal as:

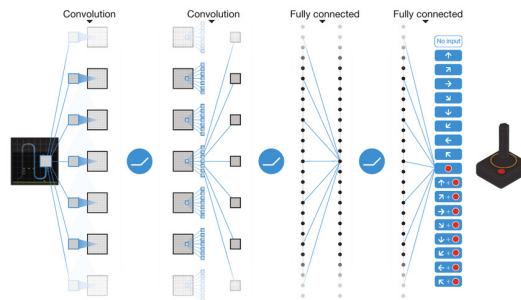


Fig. 1: Deep Q Network Topology

3) *DRQN*: is similar to the DQN, but the difference is that it stores all previous steps taken by the agent. Thanks to their memory, recurring neural networks can select actions based on their previous selections and states. The topology of the DRQN algorithm we can see in Fig.2.

B. POLICY

The purpose of the function policy is to find a map to the state with respect to the action. Such a function uses algorithms such as Actor-critic or policy gradient. [2]

III. GAME RULES

The game of Nine Men's Morris is for two players and consists of a playing area with 24 squares and a set of nine stones for each player. The game begins with both players taking turns placing their stones on the board with the goal to form a mill - a set of three pieces in a row on points connected with a line. Once every piece has been set, players may move them around along the aforementioned lines. Each time a mill is formed, the player may take one of their opponent's pieces out. Once a player is left with only 3 stones, they may move them to any free place regardless of the connecting lines. The game is played until one of the players is no longer able to form a mill (has only two pieces left).

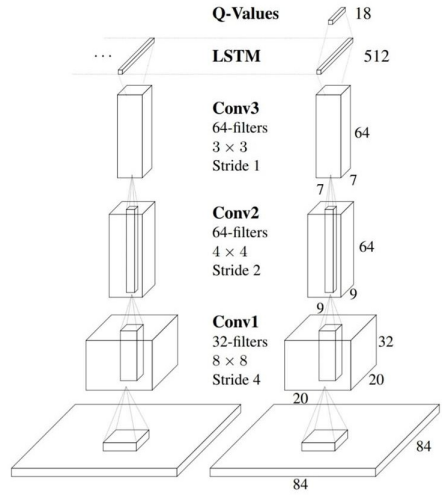


Fig. 2: Deep Recurrent Q Network Topology

IV. DESCRIPTION OF IMPLEMENTATION

First, we implemented the DQN algorithm, which did not achieve satisfactory results, so we chose the DRQN algorithm for learning the neural network as second options. The game is divided into 3 parts:

- 1) Placing stage - in this part, the agents take turn in placing their pieces until each one is put on the board.
- 2) Middle stage environment - this state of the game starts right after the end of the placing stage, when each player already has all their pieces on the board and can move them to neighboring free positions that are interconnected.
- 3) Final stage - occurs after the middle stage, when the player has only three pieces left on the board and it ends when one of the teams builds a mill and takes one of the pieces from the opponent. This is also the end of the game.

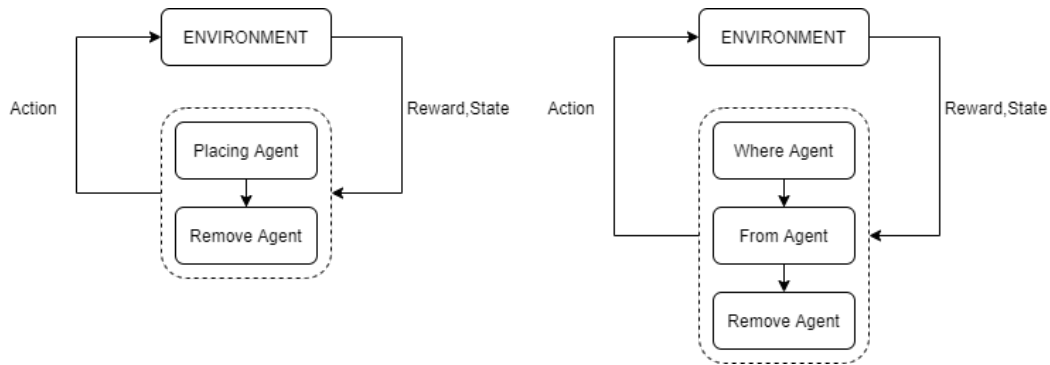


Fig. 3: Diagram

The game environment contains several agents responsible for playing the game, two in the Placing stage and three in the each of other stages.

First of these, called “where agent”, is responsible for selecting a vacant position on the board where the player’s piece can be moved, preferably where the mill can be built. This agent has 24 outputs, each representing a position on the board. Chosen output is used as input for from agent.

Second agent, called “from agent”, is responsible for selecting the piece to be moved to a new position in one of the given teams. This agent has 24 outputs, each representing position on the board.

Third agent called “remove agent” is responsible for removing an opponent’s piece. This agent is only used when a new mill is built. The output of this agent is 25 positions, of which 24 outputs, each representing position on the board and the last one is the possibility when the agent can not take any piece. This case occurs when all the opponent’s piece are contained in a mill, from which the piece cannot be taken according to the rules.

All three agents depend on the progress of the game, which means that after the action of “where agent”, “from agent” is choosing an action and if a new mill is built, then the last of the

agents, “remove agents” is choosing an action.

The neural network used, consists of an input layer, two hidden layers, one Long short-term memory (LSTM) layer and an output layer. Where the input of the neural network is a playing area represented as a 7×7 matrix, where 24 squares are intended for playing the game and the other squares are unavailable. “From agent” has, in addition to the area, also the output from “where agent”. The input layer has a size of 49 neurons, the hidden layers have 256 neurons, the LSTM layer has 5 separate layers where each also has 256 neurons, and the output layer is the number of actions of each agent.

Reinforcement learning is based on reward and punishment, which means that the agent receives a reward for the correct action and a penalty for the wrong action. The agent tries to collect as much reward as possible during the game. The agent receives the highest reward for the position of the mill, exactly 1000. For the correct selection of the correct figure to move or the selection of a free position, where the agent can move the piece will receive a reward of 100. Otherwise, the agent receives a penalty of -100 for a bad move. It will also receive a penalty for too many moves without the right move.

V. RESULTS

We tested whether the agent is able to learn the rules of the game Nine Men’s Morris and then play the game. During testing, we tried to use algorithms like DQN and DRQN to learn the agent. Because the principle of the game works on the basis of two teams, which are represented as agents and to ensure the balance of the algorithm so that one team is not preferred over the other, agents are therefore divided into two teams, white and black. Of the mentioned algorithms, the DRQN algorithm came out of the testing as better alternative, and therefore achieved better results. The results of the DRQN algorithms are shown in the following figures.

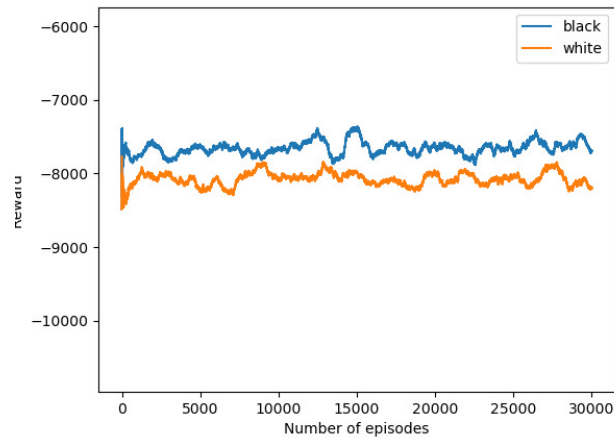


Fig. 4: Last stage - From agent

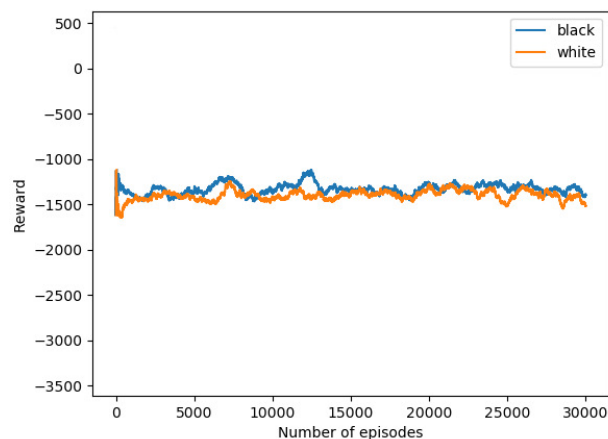


Fig. 5: Last stage - Where agent

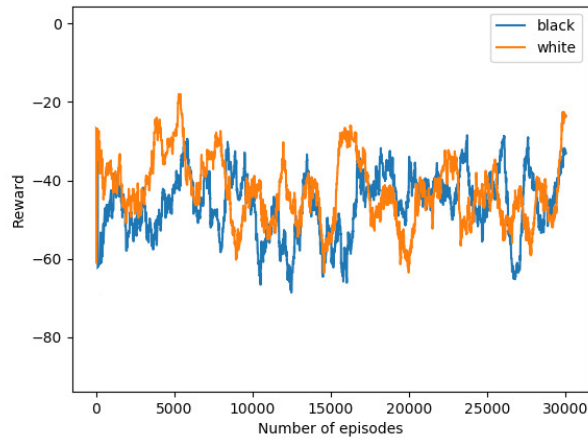


Fig. 6: Last stage - Remove agent

VI. CONCLUSION

In this article, we explained how it is possible to use the basic algorithms of deep reinforcement learning in training agents to play a game for two players. We have described the problems related to the algorithms we used. We simply introduced the basic principle of Nine Men’s Morris, and also explained step by step the whole process we went through in solving our work. The results contained in the work are sufficient to point out the functionality of the solution we created. A possible expansion of research would be based on the use of different deep reinforcement learning algorithms and also on a larger number of games played.

LIST OF FIGURES

1	Deep Q Network Topology	2
2	Deep Recurrent Q Network Topology	3
3	Diagram	3
4	Last stage - From agent	4
5	Last stage - Where agent	4
6	Last stage - Remove agent	5

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
 [2] Y. Li, “Deep reinforcement learning: An overview,” *arXiv preprint arXiv:1701.07274*, 2017.

Smartfón ako periférne zariadenie pre zdieľané virtuálno-reálne prostredie LIRKIS G-CVE

¹Tomáš BALLUCH, ²Štefan KOREČKO, ³Marián HUDÁK, ⁴Branislav SOBOTA

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹tomas.balluch@student.tuke.sk, ²stefan.korecko@tuke.sk, ³marian.hudak.2@tuke.sk, ⁴branislav.sobota@tuke.sk

Abstrakt — Tento článok sa zaoberá rozšírením LIRKIS G-CVE prostredia kolaboratívnej virtuálnej reality určenej pre smartfóny. Toto prostredie je charakteristické možnosťou pripojenia viacerých používateľov a použitím webových technológií. Rozšírenie má ponúknuť podporu pre smartfóny s operačným systémom Android a podporou sledovania orientácie či polohy tohto zariadenia. Používateľ by na web stránkach tohto VR prostredia mal byť schopný vidieť ostatných používateľov a mohol by sa v ňom pohybovať menením pozície smartfónu. Článok sa venuje analýze možných spôsobov sledovania pohybu zariadenia, návrhu a implementácií vhodne zvolenej technológie do webovej VR aplikácie, zhodnoteniu výsledného riešenia a potencionálnym budúcim vylepšeniam tohto riešenia.

Kľúčové slová — A-Frame, JavaScript, sledovanie pohybu, smartfón, virtuálna realita, webová aplikácia, LIRKIS G-CVE

I. ÚVOD

Systémy virtuálnych realít sú na vzostupe a stávajú sa čoraz dostupnejšími pre široké spektrum ľudí, preto je táto téma aktuálnejšia ako kedykoľvek predtým. Prostredia virtuálnych realít sa za posledné desaťročie rozšírili do početnej skupiny domén či priemyslu, ako je napríklad vzdelávanie, medicína, architektúra či zábava. Cieľom tohto článku je opísať vývoj špecifického rozšírenia multi-platformového virtuálneho prostredia LIRKIS Global Collaborative Virtual Environment (LIRKIS G-CVE). Toto rozšírenie do LIRKIS G-CVE pridáva podporu mobilných telefónov použitím aktuálne dostupných technológií. LIRKIS G-CVE ponúka virtuálno-reálne prostredie zdieľané viacerými používateľmi prostredníctvom internetu, ktorí ho môžu vnímať pomocou monitora alebo prílb (headset-ov) pre virtuálnu alebo zmiešanú realitu, ako je Microsoft Hololens, Google Cardboard so smartfónom alebo štandardný osobný počítač. Tu predstavené rozšírenie je zamerané na neustále sledovanie orientácie a pozície smartfónu, spracovanie vstupov od používateľa a vykresľovanie výstupu virtuálnej scény prostredníctvom podporovaných smartfónov s operačným systémom Android. Má poskytovať možnosť kombinácie telefónu s VR okuliarmi typu Google Cardboard. Rozšírením LIRKIS G-CVE o túto a ďalšie časti vznikne prostredie virtuálnej reality, v ktorom sa budú môcť jeho používatelia pohybovať prostredníctvom senzorov zariadenia a budú schopní vzájomnej interakcie. Smartfóny s operačným systémom Android majú vysoký podiel na celosvetovom trhu a čoraz viac týchto zariadení má podporu pre VR aplikácie, čím výrazne klesajú obstarávacie náklady pre prostredie LIRKIS G-CVE.

II. LIRKIS G-CVE A POŽIADAVKY NA ROZŠÍRENIE PRE SMARTFÓNY

Aplikácia LIRKIS G-CVE [1] je v domovskej inštitúcii autorov vyvíjané online VR prostredie, ktorého cieľom je poskytnúť viacpoužívateľský zážitok, multi-platformovú podporu na základe webových technológií. Síce existujú riešenia podobné LIRKIS G-CVE, no žiadne nepokrýva všetky jeho vlastnosti. Azda najpodobnejšie sú práce [2], [3], [4],[5] a [6]. Riešenie [2] je založené na tej istej technológii ako LIRKIS G-CVE, softvérovom rámci A-Frame. Líši sa však v spôsobe realizácie zdieľania prostredia. Systém SimCEC z [3] je postavený na hernom softvérovom rámci Unity, kým naše prostredie na flexibilnejších webových technológiách.

Takýto rozdiel je aj medzi LIRKIS G-CVE a systémom CoVAR [4],[5]. Systém VirtualOulu z [6] síce tiež využíva technológie webovej virtuálnej reality, no je zameraný na klasické prenosné počítače („laptop“, „notebook“). LIRKIS G-CVE okrem toho ráta aj s tabletmi, smartfónmi či špeciálnymi prilbami pre virtuálnu a zdieľanú realitu, kde možno zaradiť aj prilby na báze smartfónov.

Prvá z vyššie uvedených vlastností LIRKIS G-CVE, viacpoužívateľský zážitok, znamená, že sa do prostredia bude môcť pripojiť viacero používateľov súčasne, ktorí sa budú môcť vo virtuálnej scéne navzájom vidieť alebo inak medzi sebou interagovať. Každý používateľ bude reprezentovaný avатарom, ktorý sa vo VR scéne objaví pri pripojení sa používateľa do aplikácie a zmizne po jeho odpojení sa alebo vypnutí.

Druhou vlastnosťou je podpora viacerých platforiem, čo znamená možnosť spustenia tohto prostredia na viacerých platformách poskytujúcich VR zážitok. Nevyhnutne musí LIRKIS G-CVE podporovať moderné smartfóny, v najlepšom prípade s čo najväčším počtom rôznych modelov pre dosiahnutie čo najširšej kompatibility. Preto by mala byť použitá VR platforma pre mobilné telefóny poskytujúca širokú podporu zariadení, akou je napríklad Google Cardboard. Ten je podporovaný operačným systémom Android a iOS a následne bol vyvinutý aj pre web pomocou WebGL API a knižnice Three.js. Okrem smartfónov má prostredie podporovať aj headset Microsoft HoloLens a osobné počítače. Podstatným rysom VR prostredia je jeho vývoj použitím webových technológií. Tento prístup je výhodný v tom, že nie je potrebná inštalácia vlastnej aplikácie VR prostredia. Pri internetovej aplikácii stačí používateľovi otvoriť jej stránku prostredníctvom webového prehliadača. Nevýhodou tohto prístupu môžu byť potencionálne obmedzenia vývojového prostredia alebo použitého softvérového rámca. Tie sú väčšinou väčšie ako pri samostatných aplikáciách a môže sa to prejavovať napríklad na horšej úrovni grafických detailov, obmedzenej funkcionalite alebo oneskorenej spätnej väzbe kvôli pomalému internetovému pripojeniu. Avšak použitím najviac vyhovujúcich technológií sa tieto potencionálne nedostatky môžu značne minimalizovať. Okrem týchto vlastností by malo prostredie disponovať aj jednoduchým a intuitívnym používateľským rozhraním, teda ak by používateľ otočil hlavou alebo sa posunul v určitom smere v skutočnom svete, mal by sa tento pohyb čo najvernejšie preniesť do syntetického prostredia, teda používateľov avatar by mal pohnúť hlavou v čo najpresnejšom uhle alebo sa posunúť o čo najpresnejšiu vzdialenosť v porovnaní so skutočnosťou. Rozhranie aplikácie by takisto malo poskytovať rýchlu spätnú väzbu a používateľovi známe alebo zaužívané gestá a malo by byť responzívne a prispôsobené displejom rôznych veľkostí, keďže samotná aplikácia bude spustiteľná na viacerých druhoch platforiem s rôzne veľkými displejmi. Rôzne podporované platformy by takisto mali ponúkať čo najpodobnejšie vlastnosti a funkcie v rámci svojich možností, a to z dôvodu, aby používateľov rôzne platformy nijako neobmedzovali.

III. TECHNOLOGICKÁ ZÁKLADŇA ROZŠÍRENIA

Na vývoj rozšírenia pre smartfóny bol použitý softvérový rámec A-Frame, v ktorom je vyvíjané aj samotné prostredie LIRKIS G-CVE. A-Frame je webový softvérový rámec s otvoreným zdrojovým kódom (open-source), určený pre vývoj aplikácií a hier v AR a VR. Je založený na Three.js a WebVR, v ktorom je možné vyvíjať 3D a VR scény prostredníctvom jazykov HTML a JavaScript [7]. Výsledná aplikácia je spustiteľná vo webovom prehliadači a nie je potrebné ju inštalovať. Okrem tejto výhody ponúka aj podporu VR headset-ov, mobilných zariadení, stolných počítačov či najpoužívanejších webových prehliadačov. A-Frame je postavený na architektúre Entity-Component-System. To znamená, že každý objekt v scéne je entitou a každá entita sa skladá z jedného alebo viacerých komponentov určujúcich jej správanie, vzhľad alebo funkciu. Týmto spôsobom je možné meniť správanie alebo vzhľad entít za behu programu prostredníctvom pridávania alebo odoberania jej komponentov. Entity sú do scény pridávané buď pomocou HTML súboru obsahujúceho scénu vo forme hierarchickej štruktúry entít, alebo môžu byť do scény pridávané prostredníctvom JavaScript kódu, odkiaľ môžu byť menené aj ich komponenty. Networked-Aframe, skrátene NAF, je nadstavba A-Frame podporujúca zdieľanie entít a ich atribútov, čím umožňuje vývoj viac-používateľských prostredí pomocou synchronizácie entít a komponentov A-Frame scény. NAF je tiež voľne dostupná a ponúka relatívne ľahkú implementáciu a vytvorenie prostredia, preto je použitá v LIRKIS G-CVE.

Ako vývojové prostredie bolo použité IDE Glitch. Glitch ponúka niekoľko výhod, ako je online editor s automatizovaným nasadením, teda poskytuje server pre vyvíjanú aplikáciu. Ďalšou výhodou je okamžité prejavovanie sa zmien v kóde vo výslednej aplikácii bez nutnosti reštartu serveru alebo obnovenie okna s aplikáciou sa pri detekcii zmien v kóde. Má takisto podporu systému na správu verzií Git a vlastné fórum, preto ho používa veľká komunita ľudí pre vývoj webových aplikácií [8].

IV. EXPERIMENTÁLNA ANALÝZA KVALITY SLEDOVANIA POLOHY

Softvérový rámec A-Frame ponúka veľa výhod, avšak na sledovanie a zmenu polohy používateľa v scéne prostredníctvom smartfónu poskytuje iba prostriedky značkovej rozšírenej reality. To je relatívne veľké obmedzenie, pretože by to znamenalo, že by táto funkcionálna bola k dispozícii len v určitých striktných podmienkach, kedy by kamera sledovala špeciálnu značku. Keďže by tento stav značne obmedzoval pohyb používateľa vo fyzickom prostredí, ako aj jeho otáčanie, tak boli experimentálne overené dva spôsoby sledovania polohy telefónom bez potreby značiek. Prvý spôsob sledovania bol prostredníctvom polohových senzorov smartfónu, akými sú akcelerometer, gyroskop a magnetometer. Druhý spôsob bol pomocou fotoaparátu zariadenia, kedy sa zisťovala jeho poloha sledovaním bez značiek (z angl. markerless tracking).

A. Sledovanie polohy prostredníctvom polohových senzorov

Na testovanie tohto spôsobu sledovania bola vytvorená testovacia A-Frame scéna, kde bol pohyb zaznamenaný senzormi aplikovaný na kameru scény v reálnom čase. Existuje viacero spôsobov prístupu k dátam o polohe či orientácii zariadenia prostredníctvom jazyka JavaScript. Prvou možnosťou je prístup pomocou natívne podporovaných udalostí *devicemotion* a *deviceorientation*, alebo pomocou knižníc tretích strán, ktoré ponúkajú väčší počet funkcií a možností a tiež sa softvérovo snažia spresniť údaje zo senzorov pomocou rôznych techník. Na testovanie bola použitá knižnica *motion-sensors.js* implementujúca Generic Sensor API, čo je novodobá alternatíva pre udalosti *deviceorientation* a *devicemotion*. Knižnica implementuje rôzne rozhrania ako *Accelerometer*, *LinearAccelerationSensor* alebo *GravitySensor* [9], cez ktoré sa prístupuje k polohovým údajom. Postupne bolo vytvorených 7 skriptov prístupujúcich k dátam buď pomocou spomínaných natívnych funkcií alebo pomocou objektov a metód tejto knižnice. Tieto skripty riešia sledovanie polohy cez vyššie uvedené senzory, každý iným spôsobom, pričom niektoré skripty sa líšia len menšími zmenami. Niektoré používali filtre a iné techniky na potlačenie šumu zo senzorov. Skripty boli testované a implementované do scény samostatne a cieľom bolo ich porovnať a zistiť, ktorý ponúka najpresnejšie údaje.

Po implementácii, testovaní a experimentovaní s vyššie spomenutými skriptami bolo možné konštatovať, že všetky sú prakticky nepoužiteľné pre účely sledovania drobných zmien polohy telefónu. Jedným z dôvodov bol najpravdepodobnejšie fakt, že akcelerometre, ktoré sú inštalované do súčasných smartfónov, vedú dobre detegovať len náhle a prudké zmeny polohy, ako je trasenie telefónom, ale nevedia presne sledovať pomalší pohyb, ako je napríklad posun telefónu z výšky očí do výšky kolien. Teda žiaden z vytvorených skriptov neponúkal dostatočne presné výsledky, aby mohol byť použitý v samotnej aplikácii LIRKIS G-CVE.

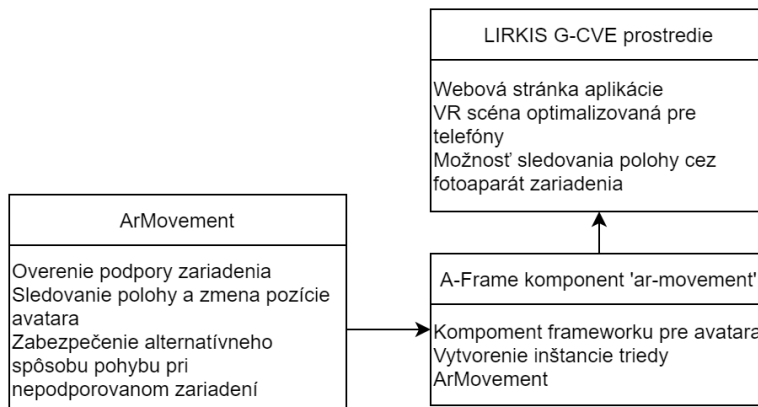
B. Sledovanie polohy prostredníctvom fotoaparátu

Keďže A-Frame počas vývoja VR prostredia nepodporoval sledovanie polohy bez značiek, bola použitá technológia tretej strany, ktorá bola integrovaná do tohto softvérového rámca. Jednou z mála open-source technológií ponúkajúcich sledovanie polohy bez značiek cez web je AR webová aplikácia od spoločnosti Google, ktorá používa knižnice a skripty písané v JavaScript-e a WebXR Device API pre sledovanie zariadenia a generovanie 3D modelov, ktoré sú vložené do výstupu z kamery. Toto API kombinuje funkcionálnu virtuálnej (VR) a rozšírenej (AR) reality vo webovom prostredí pre podporované smartfóny. Okrem tejto technológie využíva táto webová aplikácia ešte Google ARCore. Samotná AR aplikácia mala za úlohu vygenerovať do prostredia zachyteného kamerou zariadenia virtuálne kocky budiace dojem, že sa v prostredí naozaj nachádzajú. Aplikácia bola implementovaná podľa zdroja [10], poskytujúceho návod na vytvorenie takejto aplikácie, ktorý obsahuje postup, požiadavky, zdrojový kód vzorovej aplikácie a iné informácie. Vzorová aplikácia bola importovaná do vývojového prostredia Glitch, kde bol jej zdrojový kód dôkladne preštudovaný. Celá logika sledovania sa nachádzala v jednom skripte obsahujúcom niekoľko metód. Tento skript mal na starosti zistenie, či zariadenie podporuje sledovanie polohy, respektíve či má prehliadač podporu WebXR API, vytvorenie plátna, na ktoré je produkovaný AR obsah, vytvorenie virtuálnej kamery, scény, a ďalších nevyhnutných objektov a nakoniec získavanie údajov o polohe z výstupného prúdu fotoaparátu a následné prispôbenie polohy virtuálnych objektov. Vzhľadom na to, že WebXR API bolo počas vývoja LIRKIS G-CVE experimentálnou technológiou, ktorá nebola vo finálnej podobe, bolo nutné používať v telefóne prehliadač Chrome Canary vo verzii 70-72, nastaviť v ňom príslušné príznaky a nainštalovať „Služby Google Play pre RR“ podľa návodu na [10]. Po splnení všetkých podmienok bolo možné aplikáciu úspešne spustiť. Aplikácia bola testovaná v rôznych podmienkach s veľmi dobrými výsledkami. Vo všetkých prípadoch bola schopná kvalitne mapovať prostredie a správne vykresľovať virtuálne objekty. Tento spôsob sledovania nevykazoval žiadne nedostatky ani problémy, ktoré boli pozorované pri získavaní polohy prostredníctvom senzorov. Pomocou testovania a rozboru skriptu AR aplikácie bolo zistené, akým spôsobom sa získavajú údaje o polohe a kde a ako sú uchovávané a použité.

Preto bude tento spôsob použitý, upravený a importovaný, aby bolo LIRKIS G-CVE schopné sledovania polohy smartfónu jeho kamerou bez značiek.

V. REALIZÁCIA ROZŠÍRENIA

Rozšírenie aplikácie LIRKIS G-CVE pre smartfóny bolo navrhnuté tak, aby obsahovalo HTML stránku so scénou, optimalizovanou pre použitie so smartfónmi s operačným systémom Android. Keďže je prostredie už vyvíjané, HTML stránka má byť skopirovaná a upravená pre podporu mobilných zariadení. Scéna má obsahovať HTML element hlavy avatara používateľa s novo vytvoreným komponentom s názvom *ar-movement*. Následne sa vytvorila trieda *ArMovement*, obsahujúca logiku samotného sledovania polohy telefónu. Tu je vykonávané aj overenie, či zariadenie podporuje možnosť sledovania polohy. Trieda vykonáva sledovanie a cez komponent *ar-movement* prístupuje k HTML elementom scény, ktorých polohu upravuje podľa údajov z fotoaparátu (Obr.1).



Obr. 1 Návrh riešenia jednotlivých častí mobilnej VR aplikácie

Implementácia rozšírenia pre LIRKIS G-CVE vychádza z experimentálnej AR aplikácie, opísanej v časti IV.B. Z nej preberá hlavný skript, no HTML súbor definujúci scénu je vynechaný, pretože rozšírenie má byť použiteľné pre rôzne scény. Zo skriptu boli odstránené časti viažuce sa na konkrétnu scénu, časti nevyhnutné pre sledovanie polohy boli zachované. Skript tak vykonáva cyklus, kde v každej iterácii zistí aktuálnu relatívnu polohu zariadenia a tieto údaje potom použije na výpočet novej polohy avatara vo VR scéne. Cyklus sa začína získaním polohy telefónu. Túto funkcionálnu vykonával skript aj doteraz, avšak následne sa pristúpi k týmto dátam a uloží ich do pomocných premenných na ďalšie spracovanie. Ďalej sa tieto dáta upravujú do vhodnej podoby a budú pripravené na poslednú časť cyklu. V tejto časti sa hodnoty polohy priradia avatarovi v scéne a celý cyklus sa znova opakuje.

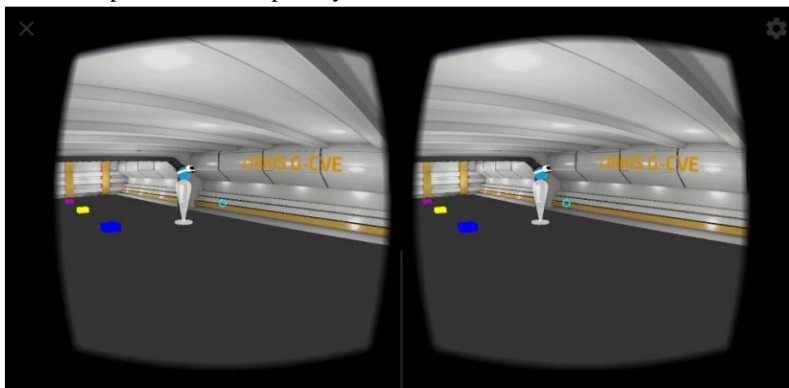
VI. INTEGROVATEĽNOSŤ ROZŠÍRENIA DO SCÉN PROSTREDIA LIRKIS G-CVE

Integrovaťnosť bola overená na dvoch scénach. Prvá, ktorú nazveme *testovacia*, bola upravená iba pre toto rozšírenie. Druhá, tzv. *finálna*, okrem tohto rozšírenia integrovala aj rozšírenie pre použitie holografického počítača MS HoloLens a rozšírenie komunikačných možností LIRKIS G-CVE.

Testovacia scéna, skladajúca sa z HTML stránky a niekoľkých skriptov, musela byť pre podporu smartfónov zmenená, pričom jej štruktúra zostala zachovaná. Z HTML súboru scény boli odstránené komponenty entity avatara slúžiace na jeho pohyb, keďže ten bude riešený cez kameru. Následne bolo do scény pridané tlačidlo spustenia algoritmu sledovania polohy, keďže sledovanie polohy musí používateľ vyvolať pomocou gesta, ako je stlačenie tlačidla. Táto požiadavka je nutná pre správne fungovanie algoritmu a nie je možné ju obísť. Taktiež bol do scény pridaný skript *ArMovement* pre sledovanie polohy. Tento skript zatiaľ obsahoval logiku sledovania a generovania kociek z AR aplikácie spomenutej vyššie, ktorá musela byť upravená. Prvou zmenou skriptu bolo vytvorenie komponentu *ar-movement*, ktorý bol priradený entite hlavy. Keďže pôvodne skript tvorila jedna trieda, tak pri inicializácii komponentu sa vytvoril objekt tejto triedy obsahujúci logiku sledovania. Všetky metódy tejto triedy boli optimalizované pre podporu testovacej VR scény. Samotná funkcionálna sledovania polohy zostala zachovaná, avšak zo skriptu boli odstránené časti generujúce virtuálne objekty do dátového prúdu z kamery. Plátno použité na vykresľovanie dátového prúdu bolo skryté, takže používateľ mohol vidieť VR scénu a nie výstup z kamery. Údaje o polohe vo forme matice boli dekomponované a bola podľa nich upravená poloha avatara. Ak skript detegoval zariadenie nepodporujúce sledovanie polohy (bez podpory WebXR alebo ARCore či zle nastavené návestia), tak bolo pri vstupe do scény vygenerované chybové hlásenie a bol zabezpečený alternatívny pohyb po scéne

vo forme dotyku displeja. V scéne bola použitá verzia A-Frame 0.9.2, ktorá ako jediná podporovala sledovanie aj vo VR móde.

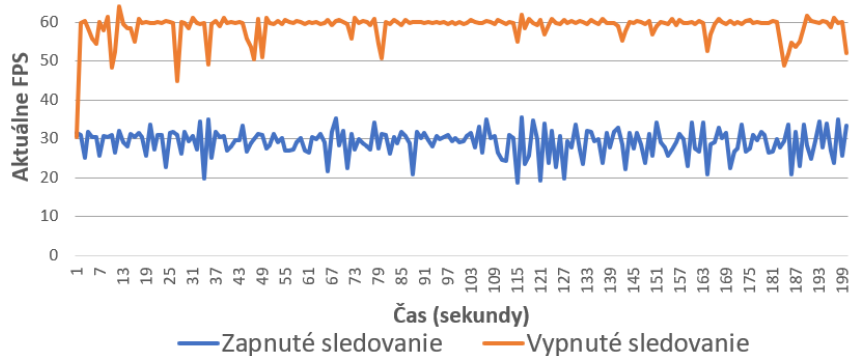
Po implementácii a úspešnom prvotnom testovaní algoritmu sledovania polohy bola dodaná finálna scéna LIRKIS G-CVE (Obr. 2)[1]. Tá ponúkala na základnej obrazovke tri tlačidlá, a to vstúpiť do VR prostredia ako administrátor, robot alebo používateľ. Bežnému používateľovi bola prístupná posledná možnosť. Po jej výbere bol presmerovaný na stránku, kde bolo potrebné zadať používateľovo meno a následne bolo možné vstúpiť do scény. Do hlavnej stránky bol pridaný skript, ktorý deteguje typ zariadenia. Ak zistí, že používané zariadenie je smartfón, otvorí HTML stránku scény pre mobilné zariadenia. Tá bola optimalizovaná a upravená pre tento typ zariadení a pre sledovanie polohy.



Obr. 2 : Ukážka finálnej scény LIRKIS G-CVE

VII. VÝKONOVÉ TESTY ROZŠÍRENIA

Finálna VR scéna v tomto štádiu poskytovala možnosť sledovania polohy pre mobilné zariadenia a bola pripravená na používanie. Testovaním bolo zistené, že počet snímkov za sekundu (FPS) v scéne je pri zapnutom sledovaní polohy zhruba polovičný (25-35 FPS) ako pri vypnutom sledovaní (50-60 FPS) pri použití telefónu Samsung Galaxy S8. To je spôsobené veľkou záťažou pre CPU, ktoré musí generovať grafický obsah scény a zároveň sledovať polohu zariadenia. Avšak aj FPS pri zapnutom sledovaní polohy je dostatočný pre uspokojivý herný zážitok a pri použití novšieho zariadenia môže značne stúpnuť.



Obr. 3 : Tabuľka FPS pri zapnutom a vypnutom sledovaní vo finálnej scéne

Takisto bolo zistené, že sledovanie polohy ponúka veľmi presné výsledky aj za zhoršených svetelných podmienok, ako je pri umelom osvetlení miestnosti, alebo v miestnosti so zrkadlami či s podlahou, ktorá odráža slnečné svetlo. Použitá technológia teda poskytuje veľmi dobré výsledky aj v horších svetelných podmienkach a prestáva byť spoľahlivá až v značne tmavých priestoroch. Za bežného dňa či pri umelom osvetlení miestnosti je sledovanie polohy a mapovanie priestoru vykonávané na veľmi vysokej úrovni.

Posledným testovaným aspektom bolo overenie spojenia. Tým sa zistovalo či a s akým oneskorením vidí používateľ používajúci smartfón ostatných používateľov a naopak. Pohyb bol častokrát veľmi plynulý, takmer so žiadnym oneskorením, no občas bol trhavý alebo s oneskorením niekoľko sekúnd. Ostatní používatelia mali veľmi podobné problémy. Tieto problémy boli najpravdepodobnejšie spôsobené preťaženým serverom, keďže vo väčšine prípadov fungovala scéna plynule. Používatelia mohli v scéne vidieť ostatných avatarov pohybujúcich sa po prostredí a otáčajúcich hlavu smerom akým pozeral jeho používateľ bez väčších problémov.

VIII. ZÁVER

Po splnení všetkých očakávaných cieľov a kritérií je rozšírenie VR prostredia LIRKIS G-CVE pre smartfóny otestované a pripravené na používanie. Toto rozšírenie ponúka zaujímavý a neštandardný zážitok z virtuálneho sveta kombináciou hlavných vlastností VR a AR. Avšak aj napriek tomu existujú určité aspekty, ktoré by sa dali odporučiť na pokračovanie jeho vývoja. Prvým aspektom je funkčnosť sledovania polohy len pri použití prehliadača Chrome Canary a softvérového rámca A-Frame verzie 0.9.2, ktorý vo VR-móde mení sám orientáciu kamery namiesto skriptu sledovania polohy. Preto je nutné na začiatku orientovať telefón správne, aby nevznikol nesúlad medzi jeho orientáciou v skutočnom svete a orientáciou kamery vo virtuálnej scéne. Je možné, že nasledujúce verzie A-Frame prinesú možnosť kompatibility AR a VR módu, kde bude možné sledovanie polohy prostredníctvom kamery vo VR prostredí, alebo tieto verzie len umožnia sledovať polohu pomocou vlastných riešení. Ďalším aspektom je použitie novších modelov smartfónov, pretože telefóny s procesormi novšími o 3 generácie sú schopné priniesť potencionálne viditeľne vyššie FPS scény, vytvárajúc lepší herný zážitok z VR prostredia.

POĎAKOVANIE

Táto práca je dopadom implementácie projektu „Univerzitný vedecký park TECHNICOM pre inovačné aplikácie s podporou znalostných technológií“, ITMS: 26220220182, podporeného z Operačného programu výskum a vývoj, financovaného z Európskeho fondu regionálneho rozvoja.

REFERENCIE

- [1] Marián Hudák, Štefan Korečko a Branislav Sobota. "Enhancing Team Interaction and Cross-platform Access in Web-based Collaborative Virtual Environments". In: Proceedings of 2019 IEEE 15th International Scientific Conference on Informatics (2019), s. 160–164. [cit. 3.5.2020].
- [2] C. Gadea, D. Hong, D. Ionescu, and B. Ionescu, "An architecture for web-based collaborative 3d virtual spaces using dom synchronization," in *2016 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*. IEEE, 2016, pp. 1–6.
- [3] P. V. Paiva, L. S. Machado, A. M. G. Valença, T. V. Batista, and R. M. Moraes, "Simcec: A collaborative vr-based simulator for surgical teamwork education," *Computers in Entertainment (CIE)*, vol. 16, no. 2, pp. 1–26, 2018.
- [4] T. Piumsomboon, A. Day, B. Ens, Y. Lee, G. Lee, and M. Billinghurst, "Exploring enhancements for remote mixed reality collaboration," in *SIGGRAPH Asia 2017 Mobile Graphics & Interactive Applications*, pp. 1-5, 2017.
- [5] T. Piumsomboon, Y. Lee, G. Lee, and M. Billinghurst, "Covar: a collaborative virtual and augmented reality system for remote collaboration," in *SIGGRAPH Asia 2017 Emerging Technologies*, pp. 1-2, 2017.
- [6] T. Alatalo, T. Koskela, M. Pouke, P. Alavesä, and T. Ojala, "Virtualoulou: collaborative, immersive and extensible 3d city model on the web," in *Proceedings of the 21st International Conference on Web3D Technology*, 2016, pp. 95–103.
- [7] Jessica Conditt. "Mozilla makes it easy to create VR websites with 'A-Frame'". In: Engadget [online]. (dec. 2015). url: <https://www.engadget.com/2015-12-17-mozilla-makes-it-easy-to-create-vr-websites-with-a-frame.html>. [cit. 1.5.2020].
- [8] Anil Dash. "What is Glitch?" In: Medium [online]. (nov. 2019). url: <https://medium.com/glitch/what-is-glitch-90cd75e40277>. [cit. 2.5.2020].
- [9] "Motion Sensors Explainer". In: The World Wide Web Consortium [online]. (aug. 2017). url: <https://www.w3.org/TR/motion-sensors/>. [cit. 2.5.2020].
- [10] "Building an augmented reality (AR) application using the WebXR Device API". In: Google [online]. (2019). url: <https://codelabs.developers.google.com/codelabs/ar-with-webxr/#0>. [cit. 3.5.2020].

Použitie MS HoloLens v zdieľanom virtuálno-realitnom prostredí LIRKIS G-CVE

¹Kristína MATIKOVÁ, ²Štefan KOREČKO, ³Marián HUDÁK, ⁴Branislav SOBOTA

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

¹ kristina.matikova@student.tuke.sk, ² stefan.korecko@tuke.sk,
³ marian.hudak.2@tuke.sk, ⁴ brislav.sobota@tuke.sk

Abstrakt — Článok sa venuje vstupno-výstupnej časti zariadenia Microsoft HoloLens v kontexte s kolaboráciou vo virtuálnom prostredí LIRKIS Global - Collaborative Virtual Environment. Z teoretického hľadiska sa zaoberá technickými možnosťami zariadenia MS HoloLens v rámci využitej technológie A-Frame. Z praktického hľadiska opisuje návrh a implementáciu klientskej časti tohto kolaboratívneho prostredia s využitím danej technológie, spolu s vykonanými experimentami. V závere je vyhodnotený aktuálny stav klientskej časti systému a jej budúce potenciálne rozšírenia.

Kľúčové slová — virtuálna realita, A-Frame, kolaborácia, Microsoft HoloLens, LIRKIS G-CVE

I. ÚVOD

Microsoft (MS) HoloLens [1] je náhlavné zariadenie pre zmiešanú realitu, ktoré v sebe spája priehľadné displeje, pre každé oko jeden, senzory na snímanie prostredia a pohybu používateľa a osobný počítač. Preto sa označuje aj ako holografický počítač [1]. Napriek tomu, že je primárne určený pre zmiešanú, alebo presnejšie rozšírenú realitu, kde sú do prostredia v okolí používateľa vkladané virtuálne objekty, môže sa použiť aj pre realitu virtuálnu. V takomto prípade je celá plocha displejov pokrytá virtuálnymi objektmi tak, že reálne prostredie cez ne používateľ nevníma. Tento článok je zameraný práve na takéto využitie MS HoloLens pre multi-platformové zdieľané virtuálne prostredie LIRKIS Global Collaborative Virtual Environment (LIRKIS G-CVE) [2]. Konkrétne sa zaoberá realizáciou softvérového klientskeho modulu systému LIRKIS G-CVE pre cieľové zariadenie MS HoloLens. Na vývoj tohto softvéru bol využitý softvérový rámec A-Frame a jazyk JavaScript. Hlavné problémy s ktorými bolo potrebné sa vysporiadať pri jeho vývoji boli spôsob zobrazovania virtuálneho prostredia, pohyb klienta vo virtuálnom prostredí a jeho interakcia so systémom. Článok sa zameriava na všetky tri hlavné problémy a objasňuje ako boli navrhnuté a implementované konkrétne vstupno-výstupné časti modulu pre MS HoloLens.

II. TECHNOLÓGIA A-FRAME A PODPORA PRE MS HOLOLENS

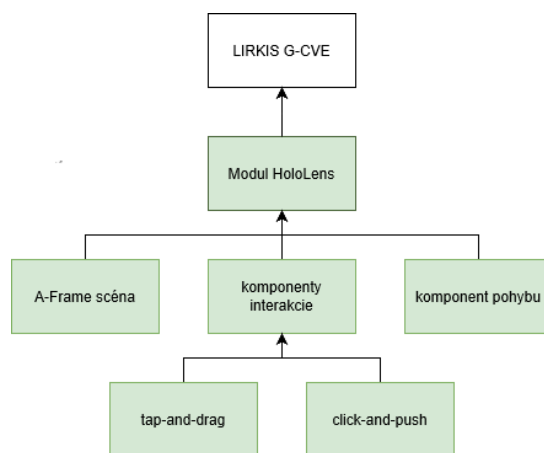
Základom LIRKIS G-CVE je webový softvérový rámec (framework) A-Frame [3], postavený na technológii WebVR. A-Frame je určený pre vytváranie virtuálnych prostredí zobrazovaných pomocou webových prehliadačov. Jeho architektúra je založená na princípe entita-komponent-systém čo poskytuje väčšiu flexibilitu a opakované využitie kódu pri vývoji aplikácie. V tejto architektúre entita predstavuje základ všetkých objektov v scéne bez vytvárania alebo vykresľovania objektov. Na entitu sa pripája komponent, predstavujúci opakovateľne použiteľný modul, ktorý poskytuje vizuálnu podobu, správanie alebo funkčnosť entít. Systém v tejto architektúre poskytuje globálnu správu a služby pre triedy komponentov. Využíva sa na separáciu logiky a dát. Html entity vytvorené pomocou A-Frame je možné ovládať pomocou jazyka JavaScript. Táto podpora jazyka JavaScript je dôležitá na rozširovanie systému o funkcionálnu, na ktorú samotný A-Frame nemá zatiaľ riešenie. Jednou z týchto funkcionalít je aj podpora vstupov z cieľového zariadenia.

V samotnej A-Frame dokumentácii [3] nie je uvedená podpora pre zariadenie MS HoloLens. Avšak uvádza sa podpora pre webový prehliadač Microsoft Edge, ktorým disponuje práve toto zariadenie. Pre správne zobrazovanie A-Frame aplikácii na zariadení HoloLens je potrebné v prehliadači povoliť niektoré experimentálne funkcie v časti *about:flags*. Konkrétne sú to experimentálne JavaScript funkcie, povolenie technológie WebVR a povolenie zobrazovania imerzívneho módu webovým prehliadačom.

III. KLIENSKY MODUL PRE MS HOLOLENS

Klientsky modul pre zariadenie MS HoloLens bol navrhnutý ako modulárny, pozostávajúci z nasledujúcich troch hlavných častí (Obr. 1):

- A-Frame scéna, reprezentovaná html súborom,
- komponenty interakcie klienta so systémom, reprezentované JavaScript súborom a
- komponent pohybu avatara, reprezentovaný JavaScript súborom.



Obr. 1 Modul klientskej časti G-CVE pre zariadenie MS HoloLens

Spomínaný html súbor obsahuje implementáciu scény a všetkých potrebných prvkov pre správne zobrazenie prostredia LIRKIS G-CVE pre zariadenie MS HoloLens. Okrem toho je v tomto súbore vytváraný avatar, ktorý reprezentuje používateľa vo virtuálnom prostredí. Scéna využíva A-Frame verziu 0.9.0 pretože sa ukázala ako najstabilnejšia pre cieľové zariadenie. Komponenty vyvíjané pre podporu prirodzeného pohybu klienta v scéne a jeho interakcie so systémom sú opísané nižšie v tomto článku.

A. Avatar používateľa a podpora pohybu

Samotný návrh avatara je zložený z troch základných častí: hlava, telo a ruky. Tieto časti sa pre daného používateľa nezobrazujú, avšak sú dôležité pre ostatných klientov v scéne. Najdôležitejšou časťou je hlava, pretože zariadenie Microsoft HoloLens získava údaje z reálneho pohybu a orientácie používateľovej hlavy, a teda tieto údaje je vhodné preniesť aj na hlavu avatara. Tieto údaje sú získavané a prenášané na html entitu hlavy pomocou A-Frame komponentu s názvom *look-controls*, ktorý si pre správnu funkcionálnu vyžaduje *camera* komponent. To je pre systém relevantné, pretože kamera definuje perspektívu sledovania scény používateľom. Okrem týchto komponentov je na html entitu hlavy avatara naviazaný aj komponent *cursor*, ktorý je založený na komponente *raycaster* a poskytuje pre používateľa možnosť interakcie pohľadom. Dokáže aj odpočúvať udalosti a zachytiť priesečník s najbližšou viditeľnou entitou, čo je dôležité pre vytvorenie ľahkej interakcie so systémom.

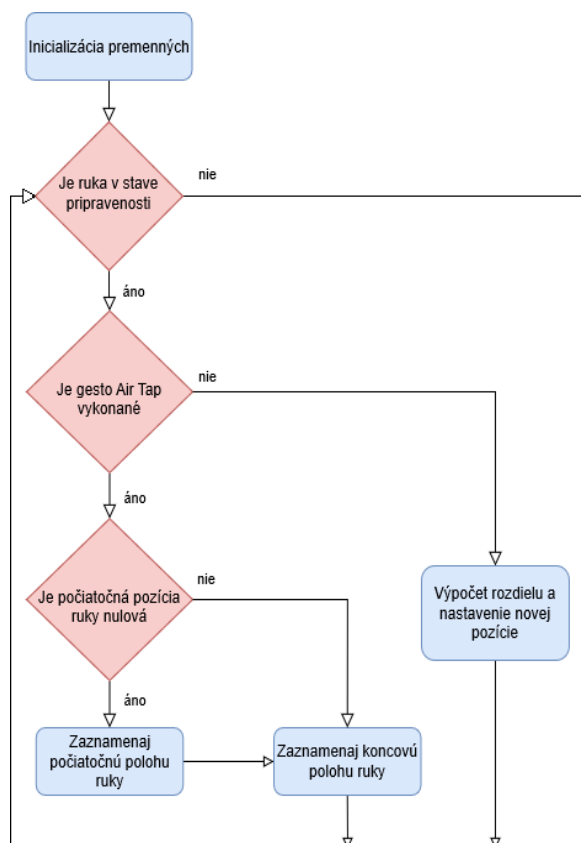
Ďalšími dvoma časťami avatara sú entity predstavujúce telo a ruky, ktoré získavajú údaje o pohybe a orientácii používateľovej hlavy. Následne využívajú tieto údaje aj na nastavenie svojej vlastnej pozície a rotácie.

B. Podpora interakcie klient-systém

Charakteristickou súčasťou zariadenia MS HoloLens je jeho interakcia s hologramami pomocou snímania gest rúk. Z tohto dôvodu bol navrhnutý prvý skript na interakciu, ktorý dokáže spracovať údaje zo snímania používateľových rúk. Princípom tohto komponentu s názvom *tap-and-drag* je využívanie a sledovanie polohy rúk v trojrozmernom priestore, sledovanie odohrania gesta *Air Tap* a nastavovanie novej polohy objektu na základe rozdielu medzi počiatočnou a koncovou pozíciou ruky. Celé kontrolovanie a značenie polohy ruky prebieha len ak je vyko-

návané gesto Air Tap. Po uvoľnení gestu sa presunie objekt, na ktorý sa klient pozeral a interagoval s ním na novú pozíciu, určenú pripočítaním rozdielu polohy ruky pri vykonaní gestu a pri jeho uvoľnení. Pri návrhu tohto komponentu bolo zistené, že ruky sa správajú pre zariadenie ako ovládače, a preto je z nich možné získať stav pripojenia a stlačenia tlačidla pomocou metód uvádzaných v dokumentácii Gamepad API [4]. Toto zistenie výrazne uľahčilo implementáciu prirodzenej interakcie klient-systém.

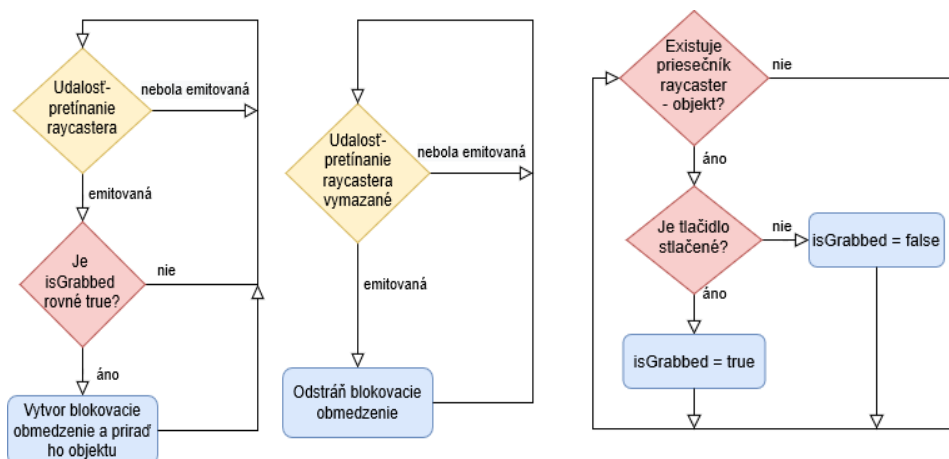
Postup vykonávania jednotlivých procesov tohto komponentu je graficky znázornený pomocou vývojového diagramu (Obr. 2).



Obr. 2 Vývojový diagram interakcie tap-and-drag

Tak, ako prvý komponent určený na interakciu, aj komponent s názvom *click-and-push* využíva sledovanie udalosti pretínania raycastera s objektom a udalosti zrušenia tohto pretínania. Rozdielom je využitie fyziky na vytváranie, priradovanie a rušenie fyzických blokovacích obmedzení pri emitovaní týchto udalostí. Toto obmedzenie odstráni všetky stupne voľnosti medzi dvoma objektami, konkrétne medzi objektom na ktorý sa používateľ pozerá a objektom kurzora. Pre klienta sa toto obmedzenie javí ako ukotvenie objektu na kurzor a hýbanie s objektom smerom, ktorým sa používateľ otočí. Pre zabránenie vytvárania fyzických obmedzení vždy keď sa používateľ pozerá na daný objekt bola implementovaná logika, ktorá vytvárala tieto obmedzenia len v prípade ak tlačidlo na ovládači HoloLens Clicker bolo stlačené. To zabezpečí aby sa obmedzenia vytvárali len vtedy, ak si to používateľ vyžaduje.

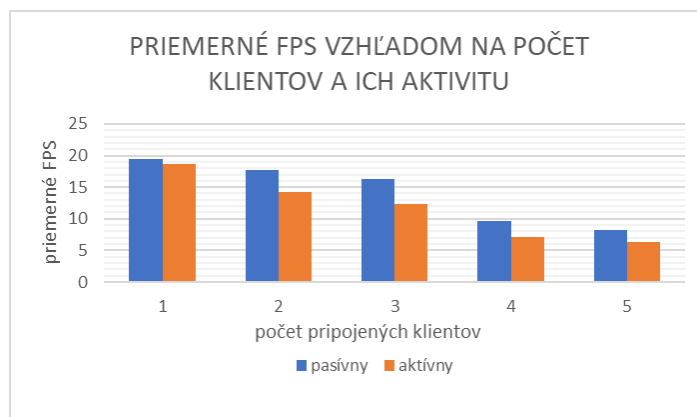
Postup vykonávania jednotlivých procesov tohto komponentu je graficky znázornený pomocou vývojových diagramov zobrazených na Obr. 3.



Obr. 3 Vývojové diagramy interakcie click-and-push

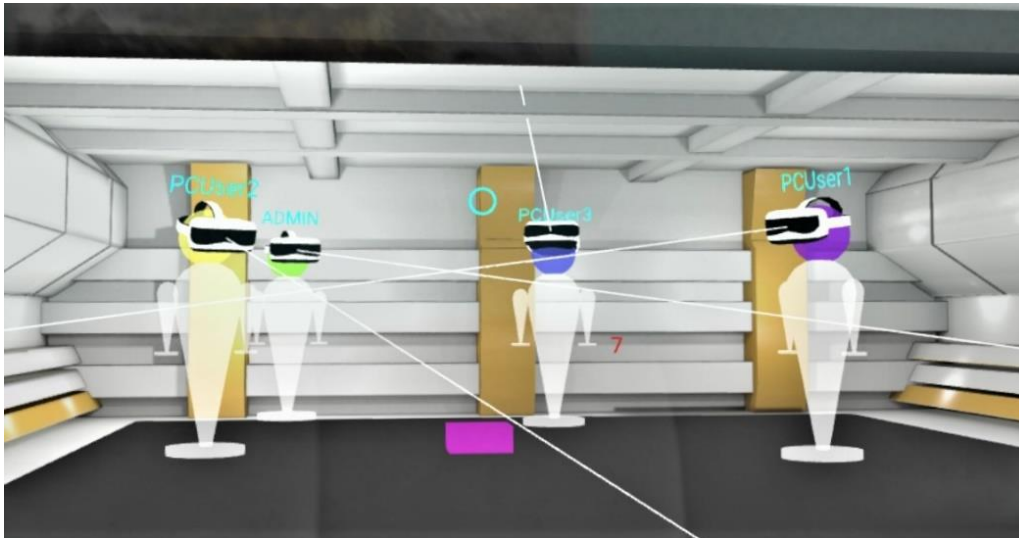
IV. EXPERIMENTÁLNE OVERENIE KLIENTSKEHO MODULU

Experimentálne testovanie a vyhodnotenie klientskeho modulu vo virtuálnom prostredí LIRKIS G-CVE sa sústredilo najmä na schopnosť zariadenia spracovávať vstupné a výstupné údaje. Ide teda o schopnosť prijímania údajov zo servera a ich vykresľovanie zariadením MS HoloLens. Dôraz sa preto dával na priemernú snímkovaciu frekvenciu pri rôznom počte pripojených klientov a v závislosti na ich aktivite v scéne. Experiment pozostával z postupného prihlasovania sa piatich používateľov. Títo pripojení používatelia mali na povel vykonávať dve úlohy po dobu desiatich sekúnd. Prvou úlohou bolo nehybné státie v zornom poli používateľa so zariadením Microsoft HoloLens (Obr. 5) a druhou zasa ich aktívny pohyb, taktiež v zornom poli. Pri každom novom prihlásenom používateľovi a pri oboch ich aktivitách sa nameral priemerný počet snímkov za sekundu. Výsledok tohto experimentu je znázornený na Obr. 4. Z tohto grafu môžeme vidieť, že priemerná snímkovacia frekvencia klesá s pribúdajúcim počtom prihlásených klientov. Avšak pri aktívnych používateľoch klesá ešte rýchlejšie ako pri pasívnych. Tento pokles spôsobuje neplynulé zobrazovanie scény zariadením.



Obr. 4 Graf priemerného FPS pri aktivite klientov

Rovnako sa sledovalo systémové oneskorenie pri vykonávaní týchto experimentov. Pri testovaní sa ukázalo, že po hlasovom potvrdení používateľov o vykonaní pohybu, vznikol časový posun systému spôsobujúci neskorší pohyb avatarov týchto používateľov. Namerané oneskorenie bolo v rozmedzí od 5 až 25 sekúnd. Toto oneskorenie mohlo byť zapríčinené zlou odozvou serveru, keďže „ping“ z využitej siete na server dosahoval priemernú hodnotu 120ms.



Obr. 5 Pasívni klienti v zornom poli zariadenia MS HoloLens, ktorí sa zúčastnili testovania

V. MOŽNOSTI ROZŠÍRENIA A PODOBNÉ RIEŠENIA

Rozšírenie kolaboratívneho prostredia LIRKIS G-CVE o klientsky modul pre zariadenie MS HoloLens zlepšilo podporu systému LIRKIS G-CVE o ďalšie zobrazovacie zariadenie virtuálnej reality. Z pohľadu kolaborácie teda umožnilo ďalšej skupine klientov využívať naplno toto prostredie. Modul opísaný v tomto článku dokáže spracovať údaje o polohe zariadenia, gestikulačné vstupy a vstupy z ovládača HoloLens Clicker, ktoré následne využíva na pohyb po scéne a interakciu používateľa s objektami v tejto scéne.

Z hľadiska budúcich rozšírení vstupno-výstupnej časti pre HoloLens je zámerom posilnenie tejto podpory a dosiahnutia ideálnejších výsledkov pri vizualizácii prostredí vytvorených v systéme LIRKIS G-CVE. To sa dá dosiahnuť elimináciou obmedzení, ktoré negatívne ovplyvňujú proces ovládania a zobrazovania. Okrem toho môže byť cieľom rozšíriť výskumnú činnosť o ďalšie vstupné údaje zariadenia ako je napríklad zvuk. Rozoznávaním hlasových povelov by mohol vzniknúť nový spôsob interakcie klienta so scénou a tým vylepšiť už existujúcu podporu pre zariadenia Microsoft HoloLens.

Tu predstavené prostredie LIRKIS G-CVE a jeho rozšírenie pre MS HoloLens vykazuje istú mieru podobnosti s inými existujúcimi riešeniami. Samotnému LIRKIS G-CVE sú azda najpodobnejšie práce [5], [6] a [7]. Riešenie [5] je založené na tej istej technológii ako LIRKIS G-CVE, softvérovom rámci A-Frame. Líši sa však v spôsobe realizácie zdieľania prostredia. Systém SimCEC z [6] je postavený na hernom softvérovom rámci Unity, kým naše prostredie na flexibilnejších webových technológiách. Systém VirtualOulu z [7] síce tiež využíva technológie webovej virtuálnej reality, no je zameraný na klasické prenosné počítače („laptop“, „notebook“). LIRKIS G-CVE okrem toho ráta aj s tabletmi, smartfónmi či špeciálnymi prilbami pre virtuálnu a zdieľanú realitu, kde patrí aj tu preberaný MS HoloLens.

Ďalším podobným prostredím, najmä s ohľadom na tu opísané rozšírenie pre MS HoloLens je CoVAR [8], [9]. CoVAR je viacpoužívateľský kolaboratívny systém podporujúci viaceré technológie zmiešanej a virtuálnej reality. CoVAR ponúka kolaboráciu v lokálnej sieti, kým LIRKIS G-CVE v globálnej. Oba systémy poskytujú prirodzenú interakciu, ktorá vtiahne používateľa do virtuálneho prostredia počas kolaborácie. CoVAR podporuje uprené hľadanie (gazing) očami a celou hlavou a gestá rukami, snímané zariadením MS HoloLens alebo HTC Vive. Významne sa LIRKIS G-CVE a CoVAR líšia v technológiách implementácie: CoVAR je postavený na hernom rámci Unity, naše prostredie na webových technológiách. Vďaka tomu je LIRKIS G-CVE schopné pokryť všetky súvisiace platformy a zariadenia bez potreby rekompilácie či inštalácie svojho kódu.

VI. ZÁVER

Experimentálnym testovaním uvedeným v tomto článku sa zistilo, že výsledný produkt klientskej časti má isté problémy so zobrazovaním zložitejších scén s vyšším počtom polygónov. Podľa meraní z experimentov priemerný počet nameraných snímok za sekundu nepresiahol hodnotu 20. Pri zložitejších scénach s väčším počtom polygónov táto hodnota mierne klesala. Avšak táto hodnota rapídne klesala pri väčšom počte prihlásených a aktívnych používateľoch až na neprijateľne nízke hodnoty. To spôsobovalo problémy s plynulým behom scény. Z pohľadu komponentov zabezpečujúcich interakciu sa ukázalo, že majú svoje plusy a mínusy. Líšia sa hlavne z hľadiska fyziky, precíznosti a vzdialenosti potrebnej na presun objektu. Zatiaľ čo komponent *tap-and-drag* je lepší na precíznu manipuláciu objektom bez fyziky, druhý je vhodný na rýchlejšiu manipuláciu a na presun o veľkú vzdialenosť s využitím fyziky.

Celkovo podpora systému pre zariadenie zmiešanej reality Microsoft HoloLens je teda dostatočná, ale nie ideálna. V porovnaní s podobnou multiplatformovou webovou aplikáciou Mozilla Hubs [10], ktorá taktiež slúži na kolaboráciu vo virtuálnom prostredí, je výsledok tejto práce lepší o podporu vstupov zo zariadenia MS HoloLens. Mozilla Hubs poskytuje len prezenčnú formu pre toto zariadenie.

POĎAKOVANIE

Táto práca bola podporená z APVV grantu č. APVV-16-0202 „Vylepšovanie kognície a motorickej rehabilitácie s využitím zmiešanej reality“.

REFERENCIE

- [1] Microsoft: Dokumentácia Microsoft HoloLens [online]. [cit. 2020-05-26]. Dostupné na internete: <https://docs.microsoft.com/en-us/hololens/>
- [2] Marián Hudák, Štefan Korečko, Branislav Sobota: Enhancing Team Interaction and Cross-platform Access in Web-based Collaborative Virtual Environments, in *Proceedings of 2019 IEEE 15th International Scientific Conference on Informatics*, IEEE, 2019, pp. 160-164
- [3] Dokumentácia A-Frame [online]. [cit. 2020-05-26]. Dostupné na internete: <https://aframe.io/docs/>
- [4] Dokumentácia Gamepad API [online]. [cit. 2020-05-26]. Dostupné na internete: https://developer.mozilla.org/en-US/docs/Web/API/Gamepad_API
- [5] C. Gadea, D. Hong, D. Ionescu, and B. Ionescu, "An architecture for web-based collaborative 3d virtual spaces using dom synchronization," in *2016 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*. IEEE, 2016, pp. 1–6.
- [6] P. V. Paiva, L. S. Machado, A. M. G. Valença, T. V. Batista, and R. M. Moraes, "Simcec: A collaborative vr-based simulator for surgical teamwork education," *Computers in Entertainment (CIE)*, vol. 16, no. 2, pp. 1–26, 2018.
- [7] T. Alatalo, T. Koskela, M. Pouke, P. Alavesa, and T. Ojala, "Virtualoulou: collaborative, immersive and extensible 3d city model on the web," in *Proceedings of the 21st International Conference on Web3D Technology*, 2016, pp. 95–103.
- [8] T. Piumsomboon, A. Day, B. Ens, Y. Lee, G. Lee, and M. Billinghurst, "Exploring enhancements for remote mixed reality collaboration," in *SIGGRAPH Asia 2017 Mobile Graphics & Interactive Applications*, pp. 1-5, 2017.
- [9] T. Piumsomboon, Y. Lee, G. Lee, and M. Billinghurst, "Covar: a collaborative virtual and augmented reality system for remote collaboration," in *SIGGRAPH Asia 2017 Emerging Technologies*, pp. 1-2, 2017.
- [10] Dokumentácia Mozilla Hubs [online]. [cit. 2020-05-26]. Dostupné z: <https://hubs.mozilla.com/docs/>

Rozšírenie možností komunikácie a správy klientov zdieľaného virtuálno-reality prostredia LIRKIS G-CVE

¹Michal IVAN, ²Štefan KOREČKO, ³Marián HUDÁK, ³Branislav SOBOTA

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹michal.ivan.4@student.tuke.sk, ²stefan.korecko@tuke.sk, ³marian.hudak.2@tuke.sk, ⁴branislav.sobota@tuke.sk

Abstrakt – Článok sa zaoberá rozšírením webového kolaboračného virtuálneho priestoru LIRKIS G-CVE o manažment a obohatenú komunikáciu pripojených klientov. Toto rozšírenie umožňuje získavať informácie o klientoch, rozšírené najmä o ich polohu, iným pripojeným klientom a špeciálnemu klientovi v úlohe administrátora. Klient-administrátor má tiež prístup k štatistikám a riadeniu ostatných klientov. Článok približuje architektúru LIRKIS G-CVE a návrh a implementáciu rozšírenia pre správu a komunikáciu klientov. Taktiež prináša experimentálne vyhodnotenie výkonu rozšírenia pri použití online platformy glitch.com.

Príčné slová – virtuálna realita, A-Frame, Networked-Aframe, správa klientov, komunikácia klientov

I. ÚVOD

Kolaboratívne virtuálne prostredie je počítačový, distribuovaný, virtuálny priestor, v ktorom sa ľudia môžu stretávať a interagovať s ostatnými používateľmi, s virtuálnymi agentmi alebo s virtuálnymi objektmi [1]. Kolaboratívne virtuálne prostredia sa môžu líšiť v oblasti zobrazenia prostredia. Prostredie môže byť 3D grafický priestor, 2D prostredie, ale aj textové rozhranie. Nové médiá sa zameriavajú na vzťah jednotlivca s prostredím, v ktorom sa nachádza a na interakciu jednotlivca s týmto prostredím. Používatelia sa vo virtuálnom prostredí považujú za samostatné bytosti. Inými slovami, používatelia dostanú grafické reprezentácie samých seba označované ako avatar [2]. Avatarom sprostredkujú ostatným svoju identitu, prítomnosť a umiestnenie v priestore.

V tomto článku sa zameriame na problematiku kontroly nad klientmi, ktorí sú pripojení vo virtuálnom prostredí. Odlíšime si bežných klientov od klienta, ktorý má rozšírené právomoci. Takého klienta označíme ako administrátora. Ďalej vytvoríme administrátorské používateľské rozhranie pre použitie vo virtuálnom priestore. Administrátor tak bude mať kontrolu nad ostatnými klientmi, bude vedieť zhromažďovať záznamy o ich aktivitách a bude schopný sledovať ich pozíciu v reálnom čase. Virtuálne prostredie, v ktorom implementujeme naše riešenie sa nazýva LIRKIS G-CVE [3] a sú v ňom použité technológie A-Frame a Networked-Aframe.

II. PODOBNÉ RIEŠENIA

Práca najviac príbuzná LIRKIS G-CVE je [4], kde je prestavená architektúra pre webové kolaboratívne 3D virtuálne prostredia. Naše riešenie ako aj [4] sú postavené na tej istej technológii, konkrétne A-Frame. Práca [4] implementuje zdieľanie pomocou servera zameraného na objektovú synchronizáciu a využíva NoSQL databázu. V LIRKIS G-CVE je synchronizácia objektov riešená pomocou údajových kanálov medzi klientmi a serverom, používajúc protokol WebSocket. To je efektívnejšie pre asynchrónne pripojenie v reálnom čase, kde sa počet klientov a objektov v priebehu kolaborácie mení.

Ďalšou podobnou architektúrou je SimCEC [5], ktorá poskytuje viac-používateľské zdieľané prostredia pomocou Unity3D aplikácie, spúšťanej vo webovom prehliadači. Distribúcia zdieľaných prostredí je zabezpečená na to vyhradeným kladovým serverom. Podobne ako v našom

prípade aj tu systém vyhodnocuje aktivitu používateľa a výsledky spracováva v reálnom čase. Kým SimCEC sa zameriava na medicínske prostredia, záber LIRKIS G-CVE je všeobecnejší. Výhodou LIRKIS G-CVE oproti SimCEC je, že je postavené na webových technológiách. Vďaka tomu nevyžaduje inštaláciu špeciálnych vývojových ani produkčných prostredí.

Systém VirtualOulu [6] používa kolaboratívny 3D model mesta v imersívnej webovej virtuálnej realite. Na rozdiel od LIRKIS G-CVE je tento systém zameraný na prenosné počítače a nie tablety či smartfóny. LIRKIS G-CVE ráta so všetkými týmito zariadeniami ako aj ďalšími ako sú špeciálne prilby pre virtuálnu či zdieľanú realitu.

III. ARCHITEKTONICKÝ ZÁKLAD LIRKIS G-CVE

LIRKIS G-CVE využíva technológiu Networked-Aframe, ktorá má architektúru klient-server. Server je základom architektúry. Musí mať preto dostatočný výkon a kapacitu na zvládanie prijímania, spracovávanie žiadostí a posielania odpovedí, viacerým súčasne pripojeným klientom. Tiež obsahuje komponenty, ktoré poskytujú základnú alebo rozširujúcu funkcionálnu pre klientov.

V prostredí LIRKIS G-CVE je použitý konkrétne Networked-Aframe vo verzii 0.6.0, ktorý využíva WebRTC [7] s nadstavbou EasyRTC [8]. WebRTC je technológia, ktorá umožňuje dvom klientom komunikovať peer-to-peer, to znamená vymieňať si dáta, ktoré neprechádzajú serverom. EasyRTC je rozhranie WebRTC API. EasyRTC využíva klientske knižnice, ktorými riadi signalizáciu a vo veľkej miere izoluje aplikácie od prebiehajúcich zmien v rozhraní WebRTC. EasyRTC nám uľahčuje prácu s WebRTC funkciami.

Vďaka architektúre klient – server môžeme stavať na možnosti jednoduchého pridávania komponentov serveru alebo klientom, ktorými rozširujeme funkcionálnu celého systému. Pridávať komponenty možno buď jednotlivo každému klientovi vlastný, alebo rovnaký komponent môžu používať viacerí klienti. Každý klient si však vytvorí vlastný objekt daného komponentu. Pomocou WebRTC a EasyRTC môžu klienti komunikovať priamo medzi sebou prostredníctvom dátového kanálu.

IV. ROZŠÍRENIE PRE SPRÁVU A KOMUNIKÁCIU KLIENTOV LIRKIS G-CVE

Systém LIRKIS G-CVE podporuje viacerých pripojených klientov v jednom virtuálnom prostredí. Práve preto by mal mať systém aj klientsku rolu administrátora, ktorý má rozšírené práva a má možnosť manažovať bežných klientov. Základom pri návrhu riešenia bolo použitie komponentov a ich znovu použiteľnosť inými klientmi. Napríklad *klient1* využíva vlastný komponent, zatiaľ čo *klient2* až *klientN* využívajú rovnaký komponent. Práve preto sme využili túto vlastnosť a navrhli sme komponenty, ktoré využíva len administrátor a komponenty, ktoré využívajú ostatní klienti v prostredí. Lepšiu predstavu o začlenení rozšírenia do LIRKIS G-CVE možno získať z konceptuálneho diagramu na Obr. 1. V diagrame môžeme vidieť systém ako celok s komponentmi servera, administrátora a bežného klienta.

Server obsahuje všetky koncové body, ktoré jednotlivé komponenty využívajú. Obsahuje tiež súbor s heslom pre prihlásenie administrátora a súbor pre ukladanie a získavanie histórie klientov. Na server sa pripájajú tak klienti ako aj administrátor, no komponenty využíva len administrátor. Služí aj ako signalizačný server pre nadviazanie peer-to-peer spojenia medzi všetkými klientmi a zároveň pomocou EasyRTC a Networked-Aframe vytvára spoločný virtuálny priestor, do ktorého sa všetci klienti pripoja.

Bežní klienti majú len komponenty na úrovni webového prehliadača. Validácia používateľského mena prebieha len v prehliadači klienta a až pri vstupe do spoločného prostredia kontaktuje server. Prijímanie správ je možné až po pripojení klienta do spoločného prostredia. V implementovaných scénach bežných klientov od administrátora odlišujeme použitím inej URL adresy pre pripojenie do scény.

A. Komponenty administrátora

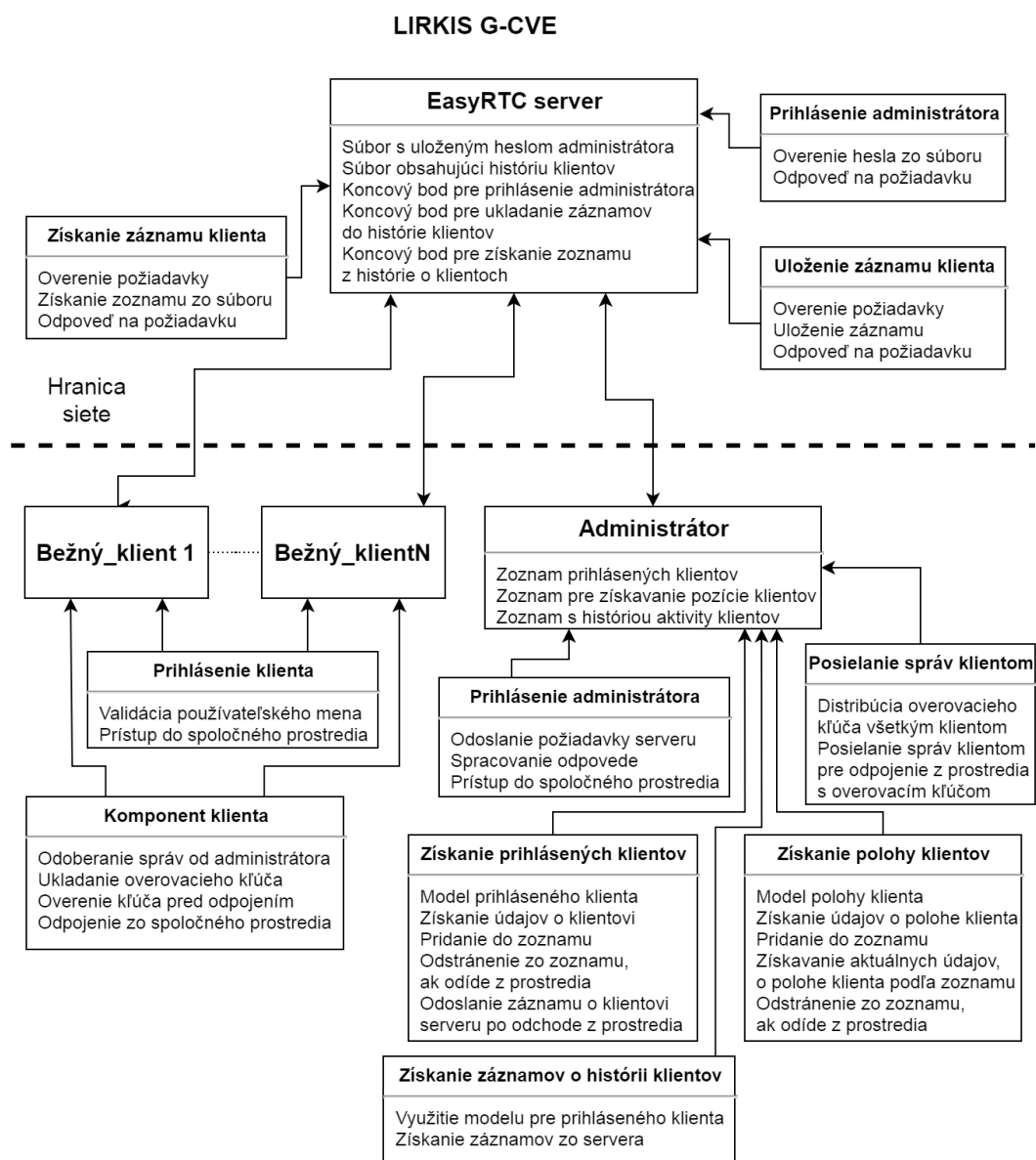
Komponenty sú opakovane použiteľné moduly alebo dátové kontajnery, ktoré môžu byť pripojené k entitám na zabezpečenie vzhľad, správania sa alebo funkcionality. Entity sú objekty, ku ktorým je možné pripojiť komponenty. Komponenty pre administrátora pridáme jeho HTML entite. Tu sa zameriame na komponenty, ktorými administrátor získava údaje o ostatných pripojených klientoch. Najprv si potrebujeme obmedziť prístup do administrátorského rozhrania a to použitím hesla pre vstup do zdieľaného prostredia. Navrhli sme preto komponent pre prihlásenie, ktorý využíva komponent servera pre prihlásenie administrátora.

Pre lepšiu prehľadnosť o pripojených klientoch v prostredí sme navrhli komponent, ktorý obsahuje zoznam pripojených klientov. Samotný Networked-Aframe nám poskytuje metódu, pre získavanie aktuálne pripojených klientov. Naš zoznam je ešte rozšírený o používateľské meno klienta, ktoré klient zadáva pri vstupe do prostredia. Používateľské meno klienta je zobrazené nad hlavou

každého avatara, takže vieme jednoducho identifikovať daného klienta zo zoznamu. Zoznam obsahuje aj čas pripojenia klienta do prostredia, aby mal administrátor prehľad, ako dlho sa klient nachádza v prostredí. Komponent pre posielanie správ klientom slúži na odpojenie klienta z prostredia. Administrátor má možnosť vybrať klienta zo zoznamu pripojených klientov a poslať mu správu, aby sa odpojil z prostredia. Pre overenie, že správa prišla od administrátora, administrátor po svojom vstupe do prostredia a pri vstupe iného klienta do prostredia, odošle všetkým pripojeným klientom overovací kľúč, ktorým sa overí práve správa pre odpojenie klienta z prostredia.

V našom riešení ukladáme záznamy o tom, kedy a ako dlho bol klient pripojený v prostredí. Pre získavanie potrebných údajov sme využili zoznam pripojených klientov. Pri odpojení klienta z prostredia je potrebné do objektu v zozname pripojených klientov pridať záznam o čase odpojenia klienta. Tento objekt sa následne odosiela serveru, ktorý ho uloží. Záznamy o aktivitách administrátora sa neposielajú. Ak máme uložené údaje o aktivite na serveri, môžeme ich poskytnúť administrátorovi v klientskom rozhraní. Preto je vytvorený komponent pre získavanie záznamov o histórii klientov. História klientov je uložená na strane servera. Ihneď po autentifikácii administrátora odošleme požiadavku na server, ktorá nám vráti zoznam posledných päťdesiat pripojených klientov. Vrátený zoznam je administrátorovi dostupný po celý čas pobytu v prostredí. Zároveň sa aktualizuje vždy, po pridaní nového záznamu a to odoslaním novej požiadavky.

Zoznam pripojených klientov a história je bežným prvkom manažmentu klientov vo virtuálnom prostredí. Naše riešenie je špecifické komponentom pre získavanie polohy klientov. Pre získava-



Obr. 1 Konceptuálny diagram systému LIRKIS G-CVE s rozšírením pre správu a komunikáciu klientov.

nie aktuálnej polohy potrebujeme využiť informácie o zdieľaných entitách ostatných klientov. Informácie o zdieľaných entitách poskytuje Networked-Aframe. V našom riešení je vytvorený samostatný zoznam klientov, ktorý obsahuje identifikátor klienta, používateľské meno klienta a identifikačné číslo zdieľanej entity, ktorej polohu chceme sledovať. Následne sa údaje o polohe získavajú v slučke pre každý jeden objekt v zozname. Identifikačné číslo entity je časť avatara klienta, ktorého polohu chceme v prostredí sledovať. Okrem pozície nás zaujíma aj rotácia hlavy avatara.

Všetky komponenty administrátora sa dajú použiť nezávisle od seba. A-frame komponenty priradíme HTML entite administrátora. Jediná ich závislosť je od inicializačného komponentu. Inicializačný komponent má vlastný A-frame komponent s názvom *admin-initialization*. Tento komponent vznikol pri implementácii a zlučuje funkcionality, ktoré by inak museli byť implementované osobitne pre každý jeden komponent ako napríklad odstránenie klienta zo zoznamu po jeho odchode z prostredia. Taktiež sa inicializačný komponent stará o inicializáciu používateľského rozhrania – tabuliek. Naše riešenie sa dá využívať aj bez používateľského rozhrania a to pomocou konzoly webového prehliadača, kde sa vieme prihlásiť do zdieľaného prostredia a tiež si vieme zobrazit' všetky zoznamy, ktoré ponúkajú komponenty administrátora.

Pre ilustráciu si bližšie si priblížime implementáciu komponentu pre získavanie aktuálnej pozície klientov. Inicializácia prebehne v metóde životného cyklu A-frame komponentu *init*, kde získame Networked-Aframe identifikačné číslo zdieľanej entity pre pozíciu a pre rotáciu. Aktualizácia polohy prebieha metóde *tick*, ktorá je tiež metódou životného cyklu A-frame komponentu a volá sa v každom vykresľovaní snímku scény. V týchto metódach voláme Networked-Aframe metódu pre získanie objektu zdieľanej entity. Z vráteného objektu si vyberieme pozíciu alebo rotáciu a priradíme ju objektu v zozname pozícií a rotácií. Takéto získavanie pozície nám zvyšuje nároky na server, pretože sa môže uskutočniť aj šesťdesiatkrát za sekundu. Na druhej strane vieme zistiť presnú pozíciu každého klienta v prostredí bez žiadneho oneskorenia.

B. Grafické používateľské rozhranie administrátora

Pre uľahčenie navigácie a presmerovania používateľa na správnu URL adresu vybraného klienta sme navrhli úvodnú stránku. Tá pozostáva z dvoch tlačidiel pre pokračovanie ako administrátor alebo ako bežný klient. Pri vstupe do prostredia sa vyžaduje zadanie používateľského mena pre bežného klienta a hesla pre vstup do prostredia ako administrátor. Pre potrebu zadania vstupu používateľa sme navrhli modálny dialóg. Obsahuje miesto pre zadanie hesla alebo používateľského mena, tlačidlo pre návrat na úvodnú stránku a tlačidlo pre potvrdenie vstupu. Používateľské rozhranie administrátora zobrazuje údaje získané komponentmi administrátora. Navrhli sme preto tabuľky pre zobrazenie obsahu zoznamov jednotlivých komponentov. Používateľské rozhranie administrátora zachycuje Obr. 2.

V. EXPERIMENTÁLNE OVERENIE ROZŠÍRENIA

Naše riešenie je dostupné online na platforme glitch.com, ktorá poskytuje zdieľaný server. V experimentálnom overení sme hľadali maximálny počet pripojených klientov, pri ktorom budú použiteľné jednotlivé funkcie administrátora. Testovali sme funkciu vytvorenia entity klienta a priradenie používateľského mena, vyhodenie klienta z prostredia a nakoniec rýchlosť odozvy tabuľky s pozíciami klientov. Administrátor disponoval stabilným internetovým pripojením s rýchlosťou viac ako 90 Mbps sťahovania a 9 Mbps nahrávania. Ostatní klienti sa pripájali na server z iných počítačov a z iných sietí.

Obmedzenie servera sme zistili už pri prvom teste, kde sme merali čas priradenia používateľského mena klientovi. Maximálny počet pripojených klientov, ktorým bolo stabilne priradené používateľské meno bol 15. Pri pokuse pripojiť ďalšieho klienta, server prestal pracovať a odpojil všetkých pripojených klientov. Priemerný čas priradenia používateľského mena v prostredí s jedným pripojeným klientom bol 0,338 sekundy. V prostredí s 15 súčasne pripojenými klientmi bol priemerný čas 0,574 sekundy.

Testovanie odpájania klienta z prostredia bolo zamerané na využitie komponentu pre posielanie správ a komponentu zo zoznamom pripojených klientov. Pri jednom pripojenom klientovi bol priemerný čas od kliknutia na tlačidlo pre odpojenie po samotné odpojenie vybraného klienta 0,624 sekundy. Tento komponent pracoval správne s maximálne 12 pripojenými klientmi, kde maximálny čas odpojenia dosiahol hodnotu 0,9 sekundy. Pri 15 pripojených klientoch bol čas odpojenia 10 sekúnd.

Všetky testy prebiehali pri staticky umiestnených klientoch, to znamená, že klient sa po pripojení do prostredia nemenil svoju polohu. Komponent pre získavanie aktuálnej pozície potreboval k otestovaniu klientov, ktorí sa pohybujú po virtuálnom priestore. Maximálny počet pohybujúcich sa klientov, ktorý poskytovaný server vedel spracovať bol 10. Používateľské rozhranie administrátora

LIRKIS G-CVE Admin user interface

Admin UI provides information about connected users, their position and history about user activity.

Connected users

Kick selected users

Search

UserID	Username	Connection time
GyqtH8W54R79un5M	testUser	9 Apr 2020 time 16:39:20

Actual position of users

Search

Username	Position			Rotation		
	X	Y	Z	X	Y	Z
testUser	-0.3221	0.1430	-1.6265	-0.3060	-2.9720	0.0000

History of users activity

Search

UserID	Username	Connection time	Disconnection time
atDuzV9yzih2bA8B	histest	7 Apr 2020 time 11:57:40	7 Apr 2020 time 11:57:56
LHVZah3SNb596mc	l23	6 Apr 2020 time 21:56:53	6 Apr 2020 time 21:56:58
UzDGSuQmTRoQNKD0	PCUser2	5 Apr 2020 time 12:38:05	5 Apr 2020 time 12:43:13
Q8Uwn5wFyn6ZFwxI	PCUser3	5 Apr 2020 time 12:39:11	5 Apr 2020 time 12:43:10
BVCa50MF9CEZsDpJ	Hlo	5 Apr 2020 time 12:41:43	5 Apr 2020 time 12:43:02
kB3HTUycCnIVEFT2	Holo	4 Apr 2020 time 19:17:03	4 Apr 2020 time 19:38:15
nyWnFS9UaloLBFTH	hey	4 Apr 2020 time 15:31:06	4 Apr 2020 time 15:31:30

Previous 1 2 3 4 5 Next Showing 1 to 10 of 50 rows 10 rows per page

Obr. 2 Používateľské rozhranie administrátora systému LIRKIS G-CVE.

pracovalo dobre, bez žiadneho oneskorenia. Po pripojení 11. klienta server prestal pracovať a odpojil všetkých klientov.

Keďže sme implementovali aj používateľské rozhranie, testovali sme aj jeho použiteľnosť a to pomocou empirickej metódy testovania použiteľnosti systému (SUS). Testovania sa zúčastnilo 10 používateľov, ktorí mali za úlohu splniť tri scenáre použitia. Scenáre boli zamerané na prácu s používateľským rozhraním administrátora. Po ukončení testovania každý účastník vyplnil dotazník SUS. Vyhodnotenie dotazníka ukázalo priemernú hodnotu odpovedí 81,5 bodu, čo je v rámci škály hodnotenia SUS nadpriemerné skóre.

VI. MOŽNOSTI BUDÚCEHO VÝVOJA ROZŠÍRENIA

Rozšírením o rozlíšenie bežných klientov a klientov s rozšírenými právomocami sme získali základ pre manažment pripojených klientov. V našom riešení získané údaje o histórii ukladáme do súboru, preto by bola vhodná integrácia databázy na strane servera. Keďže vieme získať presnú pozíciu klientov, môžeme vytvoriť klienta, ktorý sa autonómne pohybuje po scéne a dokáže sa vyhnúť kolízii s iným klientom.

Riešenie je určené pre použitie s čo najviac pripojenými klientmi, no server, ktorý zdarma poskytuje stránka glitch.com nie je dostačujúci ak chceme získať aktuálnu polohu klientov. Vhodnejšie by preto bolo vyčleniť samostatný server iba pre LIRKIS G-CVE, ideálne na pracovisku autorov.

Momentálne prebieha autentifikácia administrátora pomocou hesla, ktoré je uložené na strane servera. V budúcnosti by bolo zaujímavé zvážiť autentifikáciu pomocou Google účtu alebo iného účtu, ktorý by poskytol lepší spôsob zabezpečenia prístupu do prostredia. Samozrejme to isté

by platilo aj pre bežného klienta a mohli by sme obmedziť vstup do prostredia konkrétnemu klientovi.

VII. ZÁVER

Tu predstavené rozšírenie systému LIRKIS G-CVE poskytuje odlíšenie bežného klienta od klienta administrátora. Poskytuje používateľské rozhranie pre prihlásenie a používateľské rozhranie administrátora. To klientovi-administrátorovi umožňuje manažovať pripojených klientov a sledovať históriu pripojených klientov. Komponenty administrátora poskytujú zoznam pripojených klientov s ich používateľským menom a aktuálnu polohou klientov. Komponent histórie klientov slúži na ukladanie záznamu o pobyte klienta v prostredí a na získanie zoznamu záznamom zo servera. Tak má administrátor prehľad o využívaní prostredia.

Jednotlivé komponenty sú navrhnuté a implementované nezávisle na sebe a dajú sa ľahko použiť pre ľubovoľné prostredie vytvorené pomocou technológie A-Frame a Networked-Aframe. Dajú sa použiť s vytvoreným používateľským rozhraním alebo bez neho, prípadne integrovať do vlastného rozhrania. Komponenty sa priradujú HTML entitám jednotlivých klientov. Testovanie nám ukázalo, že riešenie je ľahko použiteľné pre používateľov a zároveň spoľahlivé v rámci možností poskytovaného servera.

POĎAKOVANIE

Táto práca je dopadom implementácie projektu “Univerzitný vedecký park TECHNICOM pre inovačné aplikácie s podporou znalostných technológií – II. fáza”, ITMS: 313011D232, podporeného z Operačného programu výskum a inovácie, financovaného z Európskeho fondu regionálneho rozvoja.

LITERATÚRA

- [1] E. F. Churchill, D. N. Snowdon, and A. J. Munro, *Collaborative virtual environments: digital places and spaces for interaction*. Springer Science & Business Media, 2012.
- [2] S. Benford, C. Greenhalgh, T. Rodden, and J. Pycok, “Collaborative virtual environments,” *Communications of the ACM*, vol. 44, no. 7, pp. 79–85, 2001. [Online]. Available: <https://dl.acm.org/doi/fullHtml/10.1145/379300.379322>
- [3] M. Hudák, Š. Korečko, and B. Sobota, “Enhancing team interaction and cross-platform access in web-based collaborative virtual environments,” in *2019 IEEE 15th International Scientific Conference on Informatics*. IEEE, 2019, pp. 160–164.
- [4] C. Gadea, D. Hong, D. Ionescu, and B. Ionescu, “An architecture for web-based collaborative 3d virtual spaces using dom synchronization,” in *2016 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*. IEEE, 2016, pp. 1–6.
- [5] P. V. Paiva, L. S. Machado, A. M. G. Valença, T. V. Batista, and R. M. Moraes, “Simcec: A collaborative vr-based simulator for surgical teamwork education,” *Computers in Entertainment (CIE)*, vol. 16, no. 2, pp. 1–26, 2018.
- [6] T. Alatalo, T. Koskela, M. Pouke, P. Alavesä, and T. Ojala, “Virtualoulou: collaborative, immersive and extensible 3d city model on the web,” in *Proceedings of the 21st International Conference on Web3D Technology*, 2016, pp. 95–103.
- [7] Google. (2011) Webrtc. [Online]. Available: <https://webrtc.org/>
- [8] P. S. Inc. (2016) Easyrtc. [Online]. Available: <https://easyrtc.com/>

Detekcia trolla typu oponent

¹Kristína MACHOVÁ, ²Matej VASILKO

^{1,2} Katedra Kybernetiky a Umelej Inteligencie, Fakulta Elektrotechniky a Informatiky,
Technická Univerzita v Košiciach, SR

¹kristina.machova@tuke.sk, ²matej.vasilko@student.tuke.sk

Abstrakt — Detekcia trolla sa stala v poslednej dobe dôležitou súčasťou diskusií o internetovej bezpečnosti. Viacero štúdií prinieslo špecifický pohľad na problematiku určenia viacerých typov trollov. V snahe objasniť problematiku detekcie trolla oponenta sa táto práca snaží definovať základný problém prečo je potrebné obmedzovať činnosť trollov. Práca ponúka pohľad na rozdelenie trollov, taktiež popisuje metódy, ktorými sa troll oponent dá v diskusií identifikovať. Práca teoreticky popisuje problematiku analýzy sentimentu a strojového učenia a ich využití pri odhalení trolla oponenta v internetovej diskusií a poukazuje na niekoľko riešení, akými sa dá činnosť trolla obmedziť. Táto práca sa sústreďuje na návrh prístupu k odhaleniu trolla typu oponent v diskusií pomocou určovania polarít jeho komentárov.

Kľúčové slová — anti-sociálne chovanie, detekcia trollov, troll oponent, analýza sentimentu, strojové učenie

I. ÚVOD

Vďaka neustálemu rozširovaniu internetových platforiem ako sú napríklad Facebook, Twitter alebo Youtube, je čoraz viac ľudí schopných prejavovať svoj názor na produkty, služby ale aj politické témy a svetové osobnosti. Nie všetci členovia internetových platforiem, ktoré v posledných rokoch rapídne získali na dôležitosti najmä vďaka uľahčenému šíreniu informácií, však chcú produkovať a rozvíjať zmysluplnú diskusiú. Namiesto toho je ich cieľom vyvolať u ľudí negatívne emócie, čím sa snažia odkloniť diskusiú iným smerom, pričom priamo alebo nepriamo napádajú iných používateľov. Takýchto používateľov nazývame Internetovými trollmi. Internetové trolovanie je predmetom intenzívneho skúmania a existuje už viacero štúdií, ktoré prinášajú svoj špecifický pohľad na túto problematiku a odpovede na otázku „Kto je vlastne troll a čo je jeho motivácia?“ Prax používania a využívania dezinformácií vo svoj prospech nie je vôbec nová. Dezinformácie boli už v minulosti využívané vládcami, diktátormi, autoritami a manipulátormi v celých dejinách ľudstva. Starostlivo vytvorené a vhodne použité dezinformačné kampane sa už niekoľko krát v histórii ukázali ako možný nástroj na dosiahnutie určitého cieľa. Pre súčasnú dobu je ale nové to, v akom prostredí sa všetko odohráva. Internet je už takmer všade, a preto sa stal domovským ekosystémom aj pre trolovanie ako také [1].

Pojem Internetový troll sa široko uplatnil v mediálnych diskusiách o demokracii na webe a o právach užívateľov sociálnych sietí. Vo veľkom množstve prípadov pod pojmom troll rozumieme jednotlivca, ktorý využíva anti-sociálne správanie s cieľom dráždiť používateľov v online komunitách, pričom využíva vo svoj prospech dezinformácie, agresívne slová a slovné útoky na to, aby vykoľajil konštruktívny a užitočný vývoj diskusiie. Kľúčovou črtou trolovania je predstieranie, že príspevok je úprimný a pochádza od obyčajného používateľa, za čím môžeme vidieť snahu o čo najúčinnejšie narušenie online interakcie. Trollovia sa väčšinou snažia ovplyvniť verejnú mienku, ale niekedy sa len chcú zabaviť. O tom, že trolovanie je v súčasnosti fenomén vypovedá aj to, že podľa služby Google Trends, ktorá sa zameriava na porovnávanie popularity vyhľadávaných výrazov v internetovom vyhľadávači Google, je od roku 2011 slovo “troll” omnoho populárnejšie slovo ako tomu bolo kedykoľvek predtým, dokonca tento termín je viac populárnym termínom ako “spam,” čo značí, že internetové trolovanie sa stáva čoraz viac nebezpečným [2].

Detekcia trollov v online priestore sa stáva v posledných rokoch čoraz dôležitejšou. Existujú aj prvé prístupy a metódy na ich detekciu. Najčastejšie sa na tento účel využívajú metódy strojového učenia a analýza sentimentu. Webové portály každým dňom rastú, či už ponúkanými službami, ale hlavne počtom používateľov, ktorí sa zapájajú do diskusií, aby sa niečo dozvedeli ale aj preto aby ovplyvňovali názory ostatných používateľov. Ovplyvňovanie názoru je súčasťou fungovania spoločnosti už od dávnych čias. V dnešnej dobe sa to však deje hlavne v online svete.

Je tomu tak preto, že informácie sa najrýchlejšie šíria internetom. Ovpływňovanie názoru však len jedným z motívov, ktoré môže troll mať. Snaha trollov o znehodnocovanie webových diskusií je jedným z dôvodov aktuálnosti tejto témy. Zaoberanie sa trollom typu oponent nie je také jednoduché, lebo je potrebné rozlíšiť trolla z princípu a nahnevaného používateľa z presvedčenia. V práci sa venujeme správaniu definujúcemu trolla, ako aj motívom, ktoré vedú k takejto činnosti, pričom sa sústreďujeme na trolla typu oponent. Ale venujeme sa aj problematike rozlišovania rôznych typov trollov a riešeniu úlohy detekcie trolla typu oponenta na základe určovania polarít komentárov a analýzy sentimentu. V práci je uvedený návrh prístupu k detekcii trolla oponenta a jeho implementácia, ktorá je vo forme aplikácie vhodná na odhaľovanie trolla oponenta vo webovej diskusií, ako aj rozlišovanie medzi oponentom trollom a oponentom, ktorý trollom nie je. Prístup je vyhodnotený na vzorke používateľov, ktorí môžu byť potenciálnym trollom oponentom.

II. DEFINÍCIA A DELENIE TROLLOV

Online trolovanie sa stal populárnym aj vďaka tomu že je v skutočnosti jednoduchý, každý človek môže potenciálne byť trollom, každý troll sa môže skrývať za svoj počítač, anonymitu svojho účtu a z veľkej časti prípadov nemusí čeliť následkom svojich činov. Vďaka tomu sa títo trollovia cítia silnejšími. K najčastejším platformám na ktorých pôsobí väčšina trollov sú sociálne siete a to hlavne kvôli masívnej prítomnosti používateľov a obrovskému množstvu preberaných tém, vďaka čomu je jednoduché skryť svoju totožnosť za falošne vytvorené účty ktoré sa na sociálnych sieťach dokážu jednoducho šíriť. Falošné účty zvyčajne trollovia používajú často, lebo im dávajú šancu skryť pravú identitu. V diskusiách sú trollovia nepríjemní, preto niektoré sociálne platformy prejavili snahu ručne identifikovať ľudí ktorí degradujú diskusie pomocou falošných účtov. Takéto riešenie je však náročné na čas, taktiež sú sociálne siete prístupné každému a tak je boj proti falošnými účtami celkovo náročný. Taktiež v mnohých prípadoch sa troll snaží neporušovať pravidlá fóra na ktorom pôsobí, aby predišiel skorému zablokovaniu jeho účtu a tým pádom dokáže čo najdlhšie vykonávať trolovanie iných používateľov. Jeden troll môže mať vytvorených viacero falošných účtov naraz, čo dokáže využiť nato, aby pomocou iných falošných účtov vyzdvihoval svoje komentáre a dodával im väčšiu vierohodnosť. Podľa štatistického portálu gdblogs ktorý sa zaberá vyhľadávaním falošných účtov na sociálnej sieti Facebook je až 8.7% profilov na tejto sociálnej sieti falošných alebo duplikovaných [3].

Z pozorovania trollov a ich chovania je zjavné že mnoho trollov sa spája a vytvárajú tak takzvané "farmy trollov." Sú to skupiny ľudí ktorí za svoju činnosť dostávajú zaplatené a trolovanie využívajú v prospech svojho zamestnávateľa. Cieľom farmy trollov je ovplyvniť verejnú mienku a nastoliť ilúziu širšej podpory pre istý názor, určitý produkt alebo určitého politického kandidáta čo môže docieľiť zisku viacerých hlasov vo voľbách alebo k vylepšeniu obrazu firmy ale aj k propagácii iných myšlienok [4].

A. Delenie trollov

Podľa toho do diskusie akého typu sa troll zapája, ale aj podľa toho ako agresívne útočí, aké slová používa v diskusií, či si z ostatných užívateľov uťahuje alebo používa neoverené alebo nepravé fakty je možné trollov rozdeľovať do viacerých skupín [1]:

- Provokačný troll
- Troll oponent
- Social - engineering troll
- Záujmový troll
- Odborný troll
- Satirický troll

Provokačný troll – patrí medzi najčastejších trollov, ktorých na internete nájdeme. Sústreďujú sa na vyprovokovanie ľudí s určitým názorom, často používa urážlivé a agresívne slová. Takýto trollovia sa snažia vyprovokovať ostatných užívateľov a odkloniť tak celú diskusiu od podstatných vecí k obviňovaniu a urážaniu, čím znehodnotia komentáre a zničia tak akýkoľvek prínos, ktorý by diskusia mohla mať.

Troll oponent – je typ trolla, ktorý na prvý pohľad nemusí používať hneď urážky a provokácie, snaží sa ale o to v diskusií prezentovať protikladný názor na základe falošných správ, zavádzania a nepravd.

Social - engineering troll – sa snaží využívať viaceré spôsoby ako sú napríklad klamstvo, hoax alebo polopравdy na to, aby používateľov podnecoval k aktivitám, ktoré by za bežných okolností neurobili. Zvyčajne je to podnecovanie používateľov aby sa pripojili k určitej organizácii, hlasovali vo voľbách za určitého kandidáta, bojkotovali danú akciu a podobne. Ovpływňovanie

politiky je ich hlavným zámerom ale okrem ovplyvňovania verejnej mienky a vyvolávania pochybností sú využívaní aj na vytváranie lepšieho dojmu a zvyšovanie predajnosti produktov.

Zájmový troll – má svoj názor na určitú situáciu a v internetovej diskusii, kde sú používatelia rozdelení na dva tábory. Zájmový troll sa snaží na prvý pohľad prikláňať na jednu stranu, to však len predstiera v domnienke, že ak sa prikloní na jednu stranu, tak ho jej podporovatelia budú brať vážne, následne sa však snaží týchto používateľov presvedčiť o tom, že ich argumenty sú nepresné a pokúša sa vyvolať pocit, že strana, ktorú na prvý pohľad obhajuje nemá pravdivé argumenty a snaží sa takto znehodnotiť jednu argumentujúcu stranu v diskusii.

Odborný troll – je najťažší na odhalenie, pretože odborný troll používa odkazy na rôzne vedecké články a stránky, používa citáty, z ktorých však vyberá to, čo sa mu hodí, využíva odborné znalosti, s ktorými manipuluje tak, ako sa mu to hodí a jeho hlavná sila spočíva v tom, že dokáže bežných komentujúcich presvedčiť, že jeho argumenty majú vedecký podklad. Na odhalenie takéhoto trolla je zvyčajne potrebný odborník na danú problematiku.

Satirický troll – je druh trolla, ktorému ide iba o to aby sa na internete pobavil a tak pomocou satiry, vtipných komentárov, ale aj rafinovaných urážok si uľahuje a znehodnocuje ostatné konštruktívne komentáre. Takýto troll je najmenej nebezpečný troll, ale dokáže ničiť konštruktívnosť diskusie

B. Troll oponent

Troll oponent patrí medzi veľmi často vyskytujúcich sa trollov v diskusiách. Jeho základnou črtou je to, že stojí sám v opozícii za cieľom zvrátiť diskusiu, v ktorej sa väčšina prispievateľov prikláňa na jednu stranu alebo zostáva neutrálnymi. Tento druh trolla je na detekciu pomerne náročný, keďže nie vždy sa musí výhradne jednať o trolla, ale môže sa jednať aj o bežného užívateľa, ktorý má iba opačný názor ako väčšina komentárov. Troll oponent za žiadnu cenu nezmení svoj názor, aj keď ho v skutočnosti nemusí zastávať. Neodradí ho ani to, že je vo svojej úlohe často sám. Svoj názor nemení a na užívateľov, ktorí mu oponujú dokáže vzniesť veľa slovných urážok, len aby ich zdiskreditoval. Pri argumentovaní proti niečomu používa dezinformácie. Aby boli dezinformácie účinné musia byť dobre cieleňé a starostlivo navrhnuté, aby neboli ľahko odhaliteľné. Pre bežných používateľov je dôležité si použité fakty overovať, neveriť všetkému čo takáto osoba napíše alebo ho prosto ignorovať. Na detekciu miery opozície tohto trolla je možné použiť analýzu sentimentu, pomocou ktorej získame hodnotu miery polarity názorov jednotlivých komentárov. Troll oponent je v diskusii ten, ktorého názor je zjavne číselne veľmi odlišný od ostatných komentárov aj celej diskusie. Na detekciu takéhoto trolla je potrebné, aby výsledky overoval expert, ktorý vie posúdiť, či sa v skutočnosti jedná o trolla oponenta alebo o tvrdohlavého používateľa. Detekcia trolla oponenta je dôležitá z hľadiska toho, že ide o jedného z najčastejších ale zároveň aj najrizikovejších trollov. Troll oponent koná svoju činnosť často aj kvôli tomu, že na začiatku diskusie bol obyčajný diskutujúci, ktorý si však za žiadnu cenu nechce priznať chybu, a tak sa snaží za každú cenu presadiť si svoj názor a aj keď už neverí v jeho správnosť.

V praxi je troll oponent využívaný na manipuláciu názorov bežných ľudí, ktorí čítajú a prechádzajú diskusiami na rôznych internetových platformách, pretože ako sa ukazuje názory na internete majú vplyv na názory užívateľov ohľadom produktov, spoločností ale aj politických strán a známych osobností. Využívanie trollov oponentov na internete umožňuje týmto skupinám získať väčšiu popularitu medzi ľuďmi alebo zmierniť na nich vyvíjaný tlak verejnosti. V takýchto prípadoch je motiváciou trolla finančná odmena, ktorá je dostatočne zaujímavá aby túto činnosť vykonávali. Taktiež jeho motivácia môže byť zábava. Pri trollovi oponentovi je potrebné si overovať fakty a nenechať sa strhnúť jeho neustálym oponovaním, v ktorom takýto troll zvyčajne stojí vo veľkej menšine. Troll oponent sa ale nevzdáva za žiadnu cenu názoru ktorý obhajuje a aj keď je spočiatku v menšine, dokáže u ľudí vyvolať pochybnosti v obhajovaní ich názoru.

III. NÁVRH PRÍSTUPU K DETEKЦИИ TROLOVANIA

Táto práca navrhuje a implementuje prístup k detekcii Trolla oponenta, ktorý je založený na analýze sentimentu a určení polarity slov a viet používateľských komentárov vybraných komentujúcich v diskusiách, ktorí by mohli byť potenciálne určení ako troll oponent. Na určenie polarity jednotlivých komentárov od používateľov bol pre účely tejto práce použitý software, ktorý bol vyvinutý na katedre Kybernetiky a Umelej Inteligencie na Technickej Univerzite v Košiciach. Konkrétne sa jedná o aplikáciu Klanaz, ktorá automatizovane dokáže hodnotiť vety a súvetia, ktoré sú do nej vložené. Troll oponent prispieva do diskusie komentármi, ktoré majú za cieľ oponovať ostatným členom diskusie, a tak komentáre trolla oponenta budú tie, ktorých ohodnotenie polarity je veľmi odlišné, či už kladne alebo záporne od polarite ostatných používateľov. Knižnica Klanaz bola použitá na vytváranie vstupnej dátovej množiny pomocou

automatizovaného sťahovania dát z internetových portálov, najmä zo sociálnej siete Facebook. Pomocou nástrojov určených na túto činnosť je možné sťahovať komentáre priamo z diskusií do tabuľkového dokumentu pre Microsoft Excel. Dané komentáre boli následne ohodnotené z hľadiska polarity názorov, tým vznikla dátová množina, ktorá obsahovala meno účtu komentujúceho, jeho komentár a ohodnotenie daného komentáru. Avšak jeden účet má spravidla viacero komentárov, a tak z hodnotenia všetkých komentárov toho istého užívateľa bola odvodená priemerná hodnota polarity jeho komentárov. Následne bola získaná aj priemerná hodnota polarity celej diskusie. Priemerná polarita názorov všetkých komentárov konkrétneho užívateľa bola porovnávaná práve s priemerom polarity názorov celej diskusie. Toto porovnanie je veľmi dôležité pre odhalenie trolla oponenta, u ktorého sa priemerná polarita názorov výrazne líši od polarity názorov celej diskusie. Porovnávanie priemeru komentárov jedného užívateľa s priemerom komentárov celej diskusie je vykonávané automatizovane v jazyku Java, konkrétne vo vývojovom prostredí IntelliJ. Keďže je jazyk Java objektovo orientovaný, tak každý používateľ je braný ako samostatný objekt. Pre vytvorenie priamej komunikácie vstupu, ktorý je vytvorený programe Microsoft Excell a prostredím IntelliJ je použitý Application programming interface Apache POI.

A. Extrakcia dát

Boli testované viaceré prostriedky na sťahovanie dát. Napokon bol použitý softvér, ktorý predstavoval internetovú službu ExportComments. Táto služba je v porovnaní s Facepagerom jednoduchšia. Nedokáže sťahovať údaje až z toľkého množstva portálov ako aplikácia Facepager. Služba ExportComments je zameraná výhradne na niekoľko sociálnych sietí a to Facebook, Instagram, Twitter, YouTube a Tiktok. Obr. 1 ilustruje dáta získané pomocou služby ExportComments.

	A	B	C	E	F	G	H	I	J	K	L
1											
2											
3											
4											
5											
6			Name (click to view profile)	Likes	Comment						
7	1		Anna Ulična	3	Je to smutné, že sa miesteni predávajú aj na miesta, ktoré sú určené pre invalidov. A cestovanie zadarmo je tiež o ničom. Ak si aspoň mesiac dopre						
8	2		Anna Ulična	10	Cestovanie zadarmo treba zrušiť, namiesto toho pridať dôchodcom aspoň 300 eura aby si mohli dovoliť cestovať. Aj tak nikdy nie sú lístky.						
9	3		Mária Mečiarová	2	myslím že to bolo vtedy, keď tu bola čínska delegácia - náš premiér vyhlásil - najprv obchod, potom ľudské práva, čo si volíme, tak sa ku nám správaj						
10	4		Zlatica De Sut	6	ano, je to pravda, na miesta vyznačené piktogramom pre telesne postihnutých ZSSK predáva miestničky zdravým ľuďom, aj studentom....uz by bolo n						
11	5		Viliam Pobežka	2	tie vagony premávali v nemecku este za hitlera						
12	6		Peter Dragon Hoang	0	Naco zadarmo vlak pre preukaz tak zbytocne preto nula peniaze. Stacia preukaz ZTP alebo student.						
13	6-1		Marek Mega	0	Peter Dragon Hoang ?						
14	6-2		Vlado Ado Probsa	0	Peter Dragon Hoang SKús to napísať ešte raz a zrozumiteľne Peter, lebo to nemá ani hlavu ani päťu						
15	6-3		Peter Dragon Hoang	0	Nepotrebujem zadarmo vlak. Suhlasiť platiť lístku na vlakom IC alebo Pendolino alebo Euronight alebo Regiojet alebo RailJet (y) Niechom rychlik I						
16	6-4		Margita Muraniova	0	Peter Dragon Hoang aj tomu rozumiete co ste pisali??? 0000						
17	6-5		Peter Dragon Hoang	0	Margita Muraniova aco????						
18	7		Martin Slávik	3	Nie len pre invalidov. Ani cestujúci s deťmi nemajú častokrát proti študentom vo vozni 2 šancu						
19	8		Kuko Friglik	3	Ja mám ZTP aj so sprievodcom. Teraz musím chodiť vlakom, lebo som operovaný na ruky. Skúste sa ísť pozrieť milí riaditelia, aké vlaky posielate smer						
20	9		Vladimír Kompas	3	poďakujte sa s. Ficovi ZSSK sú taký istý zlodeji ako on						
21	10		Lenka Kubínáková	4	Naše úžasnú železnice sa čudujem ze ešte existujú a dokážu vypraviť jeden vlak. Takmer denne im horia lokomotivy, smradlave vagóny, meškania r						
22	11		Anna Kušnierova Ugorcakova	0	Ďalší problém ...ďalší bordel....ďalší, ktorí len hrábú peniaze za lístky a na cestujúcich, vlaky, vagóny kašú z vysoká....bordel v bordeli....0000						
23	12		Vladimír Kompas	6	a inak prepáčte to sa nehanbite ich takto haníť predsa podľa s. Fica ešte takto dobre nebolo na slovensku vo všetkom sa nám darí za ich vládnutia id						
24	13		Jana Blahová	0	KLAMSTVO....V KAŽDOM VOZNI SÚ PRESNE URČENÉ MIESTA PRE INVALIDOV, TEHOTNÉ ŽENY !!!						
25	13-1		Ján Marosz OL'ANO - Za spol'á	3	Ano sú p. Jana Blahová. Ale ZSSK na ne už niekoľko týždňov predáva miestničky, takže zdravotne postihnutí ľudia nemajú tieto miesta vôbec isté.						
26	13-2		Ján Tarbaj	0	Ano včerajší prípad v rýchliku R601...aj na mieste pre invalidov bola expresná miestnička a museli vyhodiť cestujúcich so ŽiP lebo mali miestničku.						

Obr. 1 Ilustrácia extrahovaných dát použitím prostriedku ExporComments.

B. Detekcia trolla oponenta

Predpokladali sme, že troll oponent bude mať priemer hodnotení komentárov výrazne odlišný od ostatných užívateľov, či už v kladnom alebo zápornom zmysle. Taktiež sa predpokladá, že práve troll oponent bude ten, kto bude mať v diskusií veľké množstvo komentárov v krátkom časovom rozmedzí v porovnaní s bežnými komentujúcimi. Pre určenie, či sa jedná o trolla oponenta, alebo obyčajného diskutujúceho je potrebné, aby program prešiel celou vstupnou množinou a následne každého diskutujúceho rozpoznať ako osobitný objekt s parametrom "Meno" a priradil mu prislúchajúcu hodnotu, ktorá reprezentuje priemer polarity názorov všetkých jeho príspevkov. Parameter, ktorý reprezentuje priemer hodnotení všetkých komentárov užívateľa je porovnaný (pre každý objekt osobitne), s vopred vypočítaným priemerom celej diskusie, od ktorého odpočíta časť, ktorá reprezentuje vplyv daného užívateľa, ktorého skúmame, či je potencionálnym trollom.

Po prekročení vopred určeného prahu, teda maximálnej možnej hodnoty, od ktorej sa môže priemer jeho komentárov odlišovať od priemeru komentárov celej diskusie je programom označovaný každý komentujúci osobitne ako "TRUE" alebo "FALSE." Práh je nastavený na hodnotu dva pri polarite názoru z intervalu <-3, +3>. Príliš malé nastavenie prahu by viedlo k

veľkým počtom identifikovaných trollov a program by určoval ako trollov aj tých, ktorí nimi nie sú. Avšak priveľká hodnota prahu by znamenala, že len málo ktorý, ak vôbec nejaký diskutér by bol identifikovaný ako troll oponent. To hlavne z dôvodu, že hodnoty polarít komentárov sa pohybujú na množine od mínus tri po plus tri. Program ohodnotí používateľa hodnotením "TRUE," ak používateľ prekročil tento prah a program ho identifikoval ako potenciálneho trolla oponenta alebo "FALSE" ak užívateľ tento prah neprekročil, a teda ho program vyhodnotil ako bežného diskutujúceho. Bežného diskutujúceho ktorí v diskusii oponuje je možné rozoznať od trolla oponenta aj na základe počtu príspevkov. Troll prispieva veľa v krátkom časovom rozmedzí, lebo chce zmanipulovať debatu. Oponent z princípu povie svoje v niekoľkých komentároch, a už sa tomu nebude venovať hlavne ak by sa debata zvrhla.

Môžu nastať aj prípady, kedy v diskusii na nejakom webovom portáli nastane situácia, že troll oponent oponuje naozaj všetkému, bez ohľadu či je komentár myslený pozitívne alebo negatívne. Na komentáre, ktoré sú pozitívne, odpovedá troll oponent veľmi negatívne, taktiež ale v rovnakej diskusii na negatívne komentáre odpovedá veľmi pozitívne, a teda v celku je priemer hodnotení jeho komentárov neutrálny. Napríklad, ak troll oponent uverejní komentár, ktorý je z hľadiska polarít ohodnotený hodnotou +3 a potom prispieje ďalším komentárom, ktorého polarita bude -3, tak priemerná polarita týchto komentárov je približne nulová, a teda sa nebude odlišovať od prahu na toľko, aby prístup, ktorý je založený na priemere komentárov mohol určiť takéhoto trolla oponenta ako podozrivého. Takéhoto trolla oponenta prístup založený na priemere komentárov nedokázal identifikovať. Bolo preto potrebné modifikovať tento prístup a miesto priemeru komentárov vytvoriť hodnotu diferenciálu. Táto hodnota predstavuje priemer súčtu všetkých komentárov jedného užívateľa, v ktorom sú však záporné hodnoty obrátené na kladné. Teda každý komentár, ktorý je ohodnotený kladnou hodnotou, zapíšeme do vstupu jeho prevrátenú, teda kladnú hodnotu. Všetky takéto kladné komentáre jedného užívateľa je potrebné spriemerovať a takisto spriemerovať aj celú diskusiu, z čoho dostaneme potrebné hodnoty, ktoré sa budú porovnávať. Tento prístup k detekcii trolla oponenta bol implementovaný a testovaný pomocou obvyklých mier efektívnosti klasifikácie odvodených z kontingenčnej tabuľky [5]. Dosiiahnuté výsledky testov sú uvedené v Tabuľke 1.

Tabuľka 1
Výsledky testov implementácie na detekciu trolla oponenta

	Miera efektívnosti	Hodnota
1	Citlivosť (Sensitivity)	0,8333
2	Špecifickosť (Specificity)	0,9659
3	Presnosť (Precision)	0,7692
4	Negative Predictive Value	0,9770
5	False Positive Rate	0,0341
6	False Discovery rate	0,2308
7	False Negative Rate	0,1667
8	Správnosť (Accuracy)	0,9500
9	F1 skóre (F1 score)	0,8000
10	Matthews Correlation Coefficient	0,7723

IV. ZÁVER

Článok sa zameriaval na detekciu trolla oponenta vo webových diskusiách na základe polarít jeho komentárov. Prínosom tejto práce je návrh, implementácia a otestovanie prístupu založeného na porovnávaní priemernej polarít celej diskusie s priemernou polaritou komentárov podozrivých užívateľov. Výsledky testov ukázali, že toto riešenie je sľubné.

Existuje viacero možností, ako by mohol byť prístup v budúcnosti vylepšený a to napríklad tým, že by bol založený na niektorej z metód strojového učenia. Ďalšia možnosť pre vylepšenie by mohlo byť to, že by prístup pre detekciu trolla bol prístupný aj ako aplikácia vo webovom prehliadači.

POĎAKOVANIE

Práca prezentovaná v tomto článku bola podporená Slovenskou agentúrou na podporu vedy a výskumu v rámci projektu APVV-017-0267 “Automatizované rozpoznávanie antisociálneho správania v online komunitách”.

LITERATÚRA

- [1] H. Berghel, D. Berleant, The Online Trolling Ecosystem : Pages 44-51, August 2018: DOI Bookmark: 10.1109/MC.2018.3191256.
- [2] A. Birkbak, Into the wild online: Learning from internet trolls *First Monday*, Volume 23, Number 5-7, May 2018
- [3] C.E. Andrew, Fraud Social Media Statistics, Research part three. Fake account research 9.October 2015
- [4] F. Jansson,O. Casselryd, Troll detection with sentiment analysis and nearest neighbour search, 2017, [online], Dostupné na internete: < <http://www.diva-portal.org/smash/get/diva2:1112405/FULLTEXT01.pdf>>
- [5] Confusion matrix. . [online] available on [xhttp://onlineconfusionmatrix.com/](http://onlineconfusionmatrix.com/)

Metódy adaptívneho učenia na dynamických dátach

Martin SARNOVSKÝ, Michal KOLÁRIK

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

`martin.sarnovsky@tuke.sk, michal.kolarik@student.tuke.sk`

Abstrakt — Tento článok podáva teoretický prehľad v oblasti spracovania dátových prúdov, využitím algoritmov strojového učenia. Súčasťou práce je analýza súčasného stavu, popis problémov pri analýze dátových prúdov, ako aj návrh vlastného zloženého modelu. Cieľom bolo navrhnutie a implementácia nového modelu pre spracovanie dátových prúdov, ktorý bude kombinovať viaceré techniky a metódy, pre adaptáciu na problémy, ktoré vznikajú pri spracovaní dátových prúdov. Model využíva nové techniky pre udržiavanie diverzity klasifikátorov, ako aj dynamicky aktualizovanú maticu váh. Navrhnutý model bol experimentálne overený na štyroch vybraných datasetoch a porovnaný s existujúcimi metódami.

Keywords — Dátové prúdy, zložený model, konceptový drift, diverzita

I. ÚVOD

S narastajúcim technologickým pokrokom sa strojové učenie stáva čoraz viac súčasťou reálnych aplikácií. V súčasnej dobe už mnohým firmám pomáhajú algoritmy strojového učenia v takých úlohách, ako sú napr. predpovedanie používateľských nákupov, budúcich produktov, filtrovanie spamu, autonómne riadenie, predikcie vývoja trhu, rozpoznávanie tváre, predpoveď počasia, detekcia rakoviny či mnohé ďalšie. Niektoré z týchto aplikácií tolerujú spracovanie dát, ktoré môže trvať niekoľko hodín, dní či týždňov, iné ale vyžadujú spracovanie v reálnom čase alebo takmer reálnom čase, pretože zdrojom týchto dát sú údaje, ktoré neustále pribúdajú a vyvíjajú sa v čase. Takéto dáta predstavujú dátové prúdy. Na ich spracovanie sú potrebné odlišné algoritmy a techniky schopné riešiť nové problémy, ktoré s dátovými prúdmi súvisia.

Dátové prúdy bývajú potencionálne neohraničené a usporiadané sekvencie dátových položiek, ktoré postupne prichádzajú v čase [1]. Časový interval medzi jednotlivými položkami môže byť rôzny. Tieto položky môžu predstavovať dáta typu kľúč-hodnota alebo komplexnejšie štruktúry ako grafy [2]. Veľká časť výskumu v oblasti dátových prúdov sa zameriava hlavne na oblasť klasifikácie. Pri klasifikačných úlohách sa na dynamických dátových prúdov sa tieto správajú často nestacionárne a dáta sa v rámci nich vyvíjajú v čase. Mení sa distribúcia atribútov alebo cieľovej triedy v čase a tieto zmeny predstavujú tzv. konceptový drift.

II. PREHĽAD SÚČASNÉHO STAVU

Detektory konceptových driftov sú metódy, ktoré na základe informácií o výkonnosti klasifikátorov alebo priamo položkách vstupných dát vedú signalizovať či daná distribúcia dátového prúdu sa mení. Tieto signály zvyčajne aktivujú aktualizáciu alebo pretrénovanie klasifikátorov alebo nahradenie starého klasifikátora v modeli novým. Účelom je na jednej strane minimalizovať zníženie výkonnosti modelu a na druhej strane minimalizovať tzv. čas zotavenia. V nasledujúcich riadkoch si predstavíme niektoré adaptívne klasifikačné metódy schopné sa driftu prispôbiť. Aboost [3] nazývaný aj OnlineBoosting, je zložený model spracúvajúci prúdové dáta po dávkach. Vždy, keď dorazí dávka dát, sa najskôr vypočíta chyba modelu. Potom sa mechanizmus založený na štatistických testoch použije na detekciu konceptového driftu. Ak je detegovaný konceptový drift, všetky klasifikátory z ktorých je model zložený sú zahodené. Následne je vytvorený nový klasifikátor, ktorý sa natrénuje na súčasnej dávke dát. Trénovacie príklady majú priradené váhy na základe chyby klasifikácie v modeli. Potencionálnou nevýhodou tohto modelu je zahodenie celého modelu, ak je detegovaný konceptový drift, čo spôsobí oneskorenie pri zotavení modelu z driftu. SEA (Streaming Ensemble Algorithm) je zložený model patrí do kategórie s dávkovým prístupom. Model funguje na princípe pridávania

klasifikátora, pre každú novú dávku dát, pokiaľ sa nedosiahne maximálny limit počtu klasifikátorov. V opačnom prípade sa najskôr vyhodnotí kvalita nového klasifikátora na nasledujúcej dávke dát. Potom tento klasifikátor nahradí iný klasifikátor, ktorého kvalita je horšia ako kvalita nového klasifikátora na tejto dávke dát. AWE (Accuracy Weighted Ensemble) [4] AWE funguje na podobnom princípe ako SEA model. Hlavná myšlienka modelu je priradenie váh každému z klasifikátorov v modeli na základe chyby predikcie na najnovšej tréningovej dávke. Využíva špeciálny variant strednej kvadratickej chyby, ktorá využíva pravdepodobnosti jednotlivých klasifikátorov. Predpokladom je, že najnovšia tréningová dávka bude lepšie reprezentovať aktuálnu testovaciu dávku. DWM (Dynamic Weighted Majority) [5] je jedným z najznámejších modelov, ktorý patrí do kategórie online modelov s pasívnym prístupom. Ako už z názvu vyplýva, tento model dynamicky mení váhy klasifikátorov a na výslednú predikciu využíva väčšinové hlasovanie.

III. NÁVRH MODELU

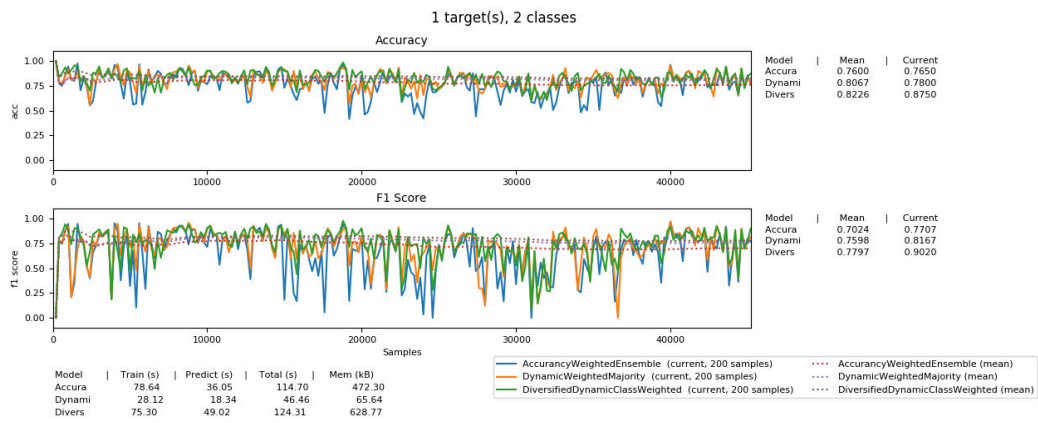
Obsahom tejto časti bude návrh vlastného zloženého modelu a jeho implementácia. Navrhovaný zložený model patrí do kategórie pasívnych online modelov. To znamená, že nevyužíva detektor konceptových driftov, ale dynamicky mení štruktúru modelu v priebehu spracovania dát. Model DDCW (Diversified Dynamic Class Weighted) je založený na myšlienke diverzifikovaného dynamického zloženého modelu s váhovaním pre cieľové triedy. Následne bude model testovaný na vybraných datasetoch, ktoré by mali pokrývať širšiu škálu vlastností a rôzne konceptové driftы.

Návrh modelu sa zakladá na predpoklade, že silný model môže byť tvorený jednoduchými klasifikátormi, ktoré sú veľmi rôznorodé – diverzité. Diverzitu využíva skoro každý zložený model, ale väčšina zložených modelov pre spracovanie dátových prúdov, túto diverzitu predpokladá buď na základe použitia odlišných klasifikátorov alebo sú jednotlivé klasifikátory tréňované na odlišných vzorkách dát. Nový, nami navrhnutý model sa snaží uplatniť možnosť zbierania metriky pre diverzitu a túto metriku ďalej využívať pri aktualizácii v modeli tak, aby bola v modeli udržiavaná diverzita medzi klasifikátormi. To znamená, že model preferuje klasifikátory, ktorých vzájomná diverzita je vyššia a naopak pri klasifikátoroch, ktoré sú si viac podobné, sa znižuje ich dôležitosť. Pre naše účely budeme používať metriku q štatistiky. Pre túto metriku platí že čím je nižšia jej hodnota, tým viac sú klasifikátory od seba rozdielne [6]. Táto metrika je ale párová, čo znamená, že je počítaná vždy pre dvojicu zvolených klasifikátorov. Pre výpočet diverzity celého modelu sa potom spočítajú čiastkové diverzity jednotlivých klasifikátorov a výsledná hodnota sa spriemeruje. Ďalším predpokladom, ktorý vstupuje do návrhu modelu je životnosť jednotlivých klasifikátorov v modeli, kedy sa uplatňuje postupné utlmovanie klasifikátorov s pribúdajúcou životnosťou. To by malo zabezpečiť postupné zabúdanie a obmieňanie starých klasifikátorov, ktoré sú v modeli už dlhší čas, za nové.

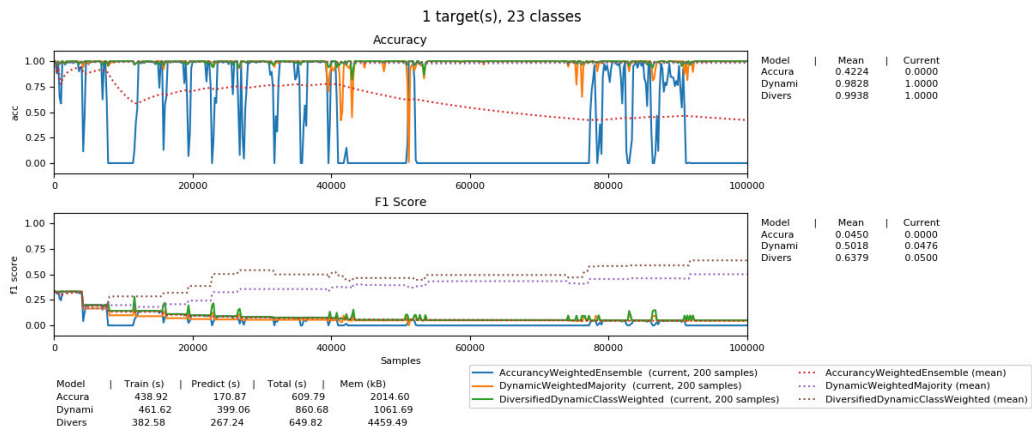
IV. VYHODNOTENIE

Prvou úlohou pre testovanie bude porovnanie jednoduchých (statických) klasifikátorov a navrhnutého modelu. V tomto teste budeme preverovať správanie sa modelu najmä v oblasti konceptových driftov, oproti jednoduchým klasifikátorom. Budeme sledovať hlavne vizuálne, ako modely spracúvajú dáta na reálnych, ale aj syntetických datasetoch (použili sme 2 syntetické - Stagger [7], LED a 2 reálne - KDD99 [8], Elec data).

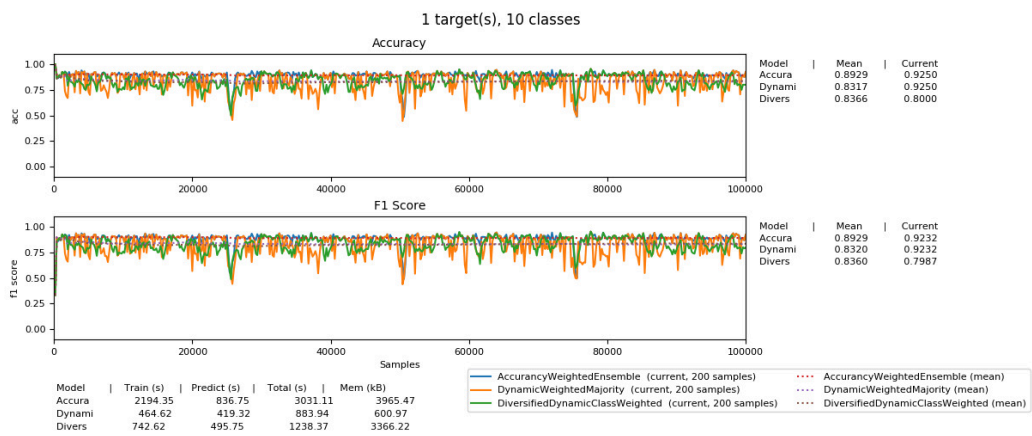
V experimentoch sme sa venovali porovnaniu výkonnosti navrhnutého modelu DDCW s modelmi AWE (Adaptive Weighted Ensemble) a DWM (Dynamic Weighted Majority), ktoré sú svojou štruktúrou podobné s našim návrhom. Cieľom tohto testovania bolo identifikovať oblasti, v ktorých tento model dosahuje lepšie výsledky, ako môže byť výkonnosť modelu pri jednotlivých typoch datasetov alebo schopnosť adaptácie na konceptový drift v porovnaní s inými modelmi. Sledovali sme metriky a ich vývoj počas jednotlivých iterácií dátového prúdu, okrem zvolených metrick porovnávali aj čas tréningovania, vyhodnotenia príkladov, alebo pamäťové nároky. Na Obr. 1-4 vidíme vizualizácie týchto metrick na jednotlivých datasetoch.



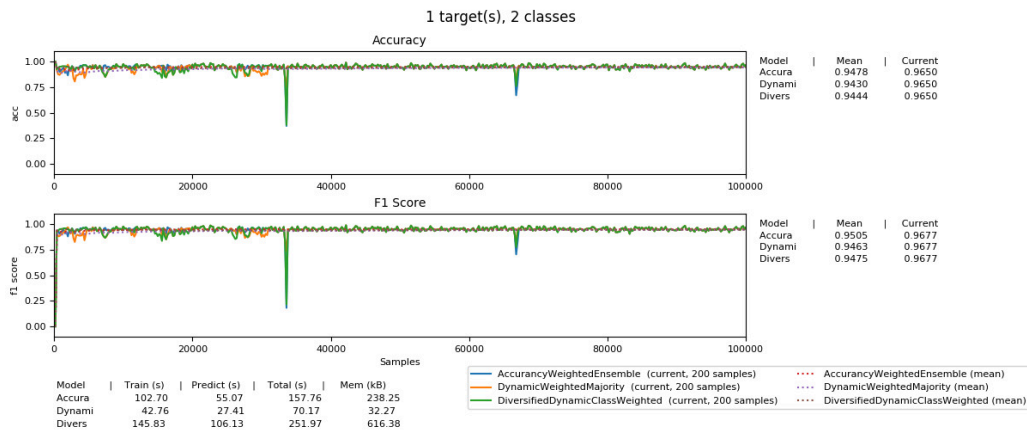
Obr. 1 Vizualizácia výkonnosti modelov DDCW, AWE a DWM na datasete Electricity



Obr. 2 Vizualizácia výkonnosti modelov DDCW, AWE a DWM na datasete KDD 99.



Obr. 3 Vizualizácia výkonnosti modelov DDCW, AWE a DWM na datasete LED.



Obr. 4 Vizualizácia výkonnosti modelov DDCW, AWE a DWM na datasete Stagger.

V experimentoch sme testovali porovnanie výkonnosti modelu DDCW s modelmi AWE a DWM. Celkovo môžeme povedať, že model DDCW bol vhodnejší pri adaptácii na konceptové drifts a na reálnych datasetoch dosahuje lepšie výsledky, ako modely AWE a DWM. K spracovaniu dát, ale vyžaduje väčšie množstvo systémových zdrojov, čoho dôsledkom môže byť použitie zložitejšej štruktúry váhovania (použitie matice váh) a zvolenej funkcie pre výpočet diverzity v modeli. Vplyv na veľkosť modelu má aj parameter, ktorý nastavuje minimálny počet klasifikátorov v modeli na 5. Táto hodnota bola zvolená preto, aby bolo možné výraznejšie uplatniť párové diverzity v modeli (minimálne potrebný počet 3 klasifikátory pre výpočet párovej diverzity).

V. ZÁVER

Cieľom tohoto článku bolo navrhnúť dynamický zložený model, ktorý by bol schopný sa adaptovať na zmeny vznikajúce v dátových prúdoch. Najväčší problém pri spracovaní dátových prúdov tvorí konceptový drift, ktorý sa v prípade statických datasetov nevyskytuje. Na základe analýzy súčasného stavu, existujúcich problémov v oblasti a možností ich riešenia sme navrhli vlastný zložený model. Tento model implementuje nové zistené poznatky, ako aj vlastné návrhy pre dynamickú adaptáciu na zmeny v dátových prúdoch. Model bol podrobený testovaniu v rámci experimentov, v ktorých preukázal schopnosti adaptácie na konceptový drift, ako aj výkonnosť modelu v porovnaní s inými populárnymi modelmi. Z výsledkov testovaní, môžeme povedať, že model svoju úlohu splnil a je schopný sa lepšie adaptovať na zmeny konceptov, ako iné testované modely. Model rovnako ponúka aj možnosti optimalizácie a to zmenou svojich parametrov. Táto práca otvára ďalšie možnosti na skúmanie, ako je úprava tohto modelu pre potreby regresie pri spracovaní dátových prúdov, ale tiež možnosť zvoliť inú metódu diverzity alebo iný prístup k udržiavaniu diverzity v modeli tak, aby bol model ešte lepšie pripravený na spracovanie konceptových driftov a efektívnejšie využil dostupné zdroje.

POĎAKOVANIE

Táto práca bola podporená APVV grantom č. APVV-16-0213.

LITERATÚRA

- [1] J. Gama, J. Aguilar-Ruiz, and R. Klinkenberg, "Knowledge discovery from data streams," *Intell. Data Anal.*, vol. 12, no. 3, pp. 251–252, 2008, doi: 10.3233/ida-2008-12301.
- [2] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," *Inf. Fusion*, vol. 37, pp. 132–156, 2017, doi: 10.1016/j.inffus.2017.02.004.
- [3] A. Beygelzimer, S. Kale, and H. Luo, "Optimal and adaptive algorithms for online boosting," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2016-Janua, pp. 4120–4124, 2016.
- [4] B. Wang and J. Pineau, "Online Ensemble Learning for Imbalanced Data Streams," pp. 1–15, 2013, [Online]. Available: <http://arxiv.org/abs/1310.8004>.
- [5] J. Z. Kolter and M. A. Maloof, "Dynamic weighted majority: An ensemble method for drifting concepts," *J. Mach. Learn. Res.*, vol. 8, pp. 2755–2790, 2007.
- [6] L. Yang, "Classifiers selection for ensemble learning based on accuracy and diversity," *Procedia Eng.*, vol. 15, pp. 4266–4270, 2011, doi: 10.1016/j.proeng.2011.08.800.
- [7] M. G. Elfeky, W. G. Aref, and A. K. Elmagarmid, "STAGGER: Periodicity mining of

- data streams using expanding sliding windows,” in *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2006, doi: 10.1109/ICDM.2006.153.
- [8] M. Tavallae, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in *IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009*, 2009, doi: 10.1109/CISDA.2009.5356528.

Webová aplikácia pre analýzu sekvencií DNA v prostredí R

Martin SARNOVSKÝ, Júlia JACKOVÁ

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,
Technická univerzita v Košiciach, Slovenská republika

martin.sarnovsky@tuke.sk, julia.jackova@student.tuke.sk

Abstrakt — Cieľom tohoto článku práce je podať prehľad o fylogénéze a vytvoriť návrh a webovú aplikáciu v jazyku R, ktorá umožní fylogenetickú analýzu sekvencií. Na to aby sme lepšie pochopili fylogenetickú analýzu na začiatku práce sme podali teoretický prehľad o fylogénéze, ako aj o konštrukcii fylogenetických stromov. V ďalšej časti je analýza bioinformatických prostriedkov v jazyku R, návrh aplikácie nasledovne jej implementácia. Praktická časť je zameraná na vytvorenie aplikácie pre fylogenetickú analýzu. Hlavným cieľom aplikácie je analýza sekvencií vírusov, zarovnanie sekvencií a konštrukcia fylogenetických stromov. Aplikáciu sme vytvárali s využitím databázy SQLite a programu na zarovnanie sekvencií ClustalW. Využívať aplikáciu môže každý zaregistrovaný používateľ. Aplikácia poskytuje používateľovi jednoducho a prehľadne vytvárať fylogenetickú analýzu a následne výsledky analýzy ukladať.

Keywords — webová aplikácia, fylogenetická analýza, fylogenetický strom, ClustalW

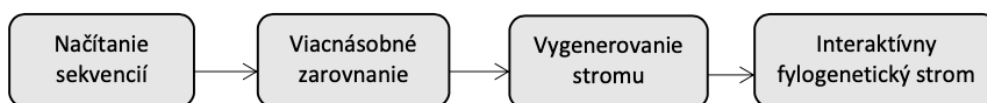
I. ÚVOD

Skúmaním fylogénézy, príbuzenských vzťahov a vývojom organizmov sa zaoberá veda fylogenetika. Fylogénéza definuje evolučný vývoj druhov alebo iných skupín organizmov cez jednotlivé vývojové stupne. Na zobrazenie vzťahov medzi organizmami využíva fylogenetické stromy. Sú to diagramy, ktoré vyjadrujú ako sa jednotlivé druhy alebo organizmy vyvíjali v čase a tiež popisujú ich spoločných predkov. Hlavným cieľom fylogénézy je hľadať podobnosť a vzťahy medzi jednotlivými organizmami. Vzťahy medzi organizmami popisujú sekvencie aminokyselín alebo nukleových kyselín. Nukleové kyseliny sú zložené z nukleotidov, ktoré obsahujú genetický kód organizmov teda gén. Tieto kyseliny sa vyskytujú v dvoch rozličných skupinách. Jedna z nich je kyselina deoxyribonukleová (DNA) a druhá je ribonukleová kyselina (RNA). DNA je dvojitá molekula, ktorá sa nachádza v jadre živých buniek. V DNA existujú štyri rôzne bázy: adenín (A), cytozín (C), guanín (G) a tymín (T). Ribonukleová kyselina sa podobá DNA, avšak na rozdiel od nej je RNA jednovláknová molekula tvorená z ribózy. Štruktúra RNA obsahuje adenín (A), cytozín (C), guanín (G) a uracil (U). RNA je hlavným genetickým materiálom, ktorý sa nachádza v rastlinách a živočíchoch. Vírusové RNA je špeciálny typ RNA, ktorý obsahuje funkcie DNA a mRNA. Tento typ RNA sa zvyčajne používa pri rôznych analýzach. Prvým krokom pri vytváraní fylogenetického stromu je identifikovanie vírusu a jeho pôvodu [1]. Fylogenetická analýza je metóda fylogénézy, ktorá sa využíva na objasnenie evolučnej histórie a vzťahov medzi skupinami organizmov. Typicky zahŕňa identifikáciu homologických sekvencií, ich viacnásobné zarovnanie, fylogenetickú rekonštrukciu a grafické znázornenie vyvedeného stromu. Výsledkom analýzy sú fylogenetické stromy, ktoré sú vytvorené tak, aby popisovali evolučný vzťah medzi organizmami [2]. Fylogenetický strom je schéma (diagram), ktorá zobrazuje línie evolučného pôvodu druhov, organizmov alebo génov zo spoločného predka. Na rekonštrukciu fylogenetického stromu pomocou sekvencií proteínov je potrebné mať súbor homológnych sekvencií. Homológne sekvencie sú tie, ktoré boli vyvinuté zo spoločného predka. Metódy na vytvorenie fylogenetického stromu sú klasifikované do dvoch skupín. Prvá je metóda založená na vzdialenosti, ktorá používa evolučnú maticu vzdialenosti. Výhodou tejto metódy je krátky čas výpočtu, preto je možné pracovať s veľkým množstvom údajov. Druhá metóda je založená na znakoch, ktorá sa využíva priamo počas konštrukcie stromu. Je ťažké rozhodnúť, ktorá metóda je najlepšia na vytvorenie fylogenetického stromu preto sa strom zhotovuje pomocou viacerých metód a výsledky sa porovnávajú. Ak sú výsledky rovnaké, fylogenetický strom sa považuje za spoľahlivý [3], [4].

II. NÁVRH WEBOVEJ APLIKÁCIE PRE PODPORU FYLEGETICKÝCH ANALÝZ

Ako prvé a najdôležitejšie pri vytváraní návrhu aplikácie je potrebné stanoviť ciele a požiadavky. Hlavným cieľom je vytvoriť webovú aplikáciu, ktorá by slúžila každému kto potrebuje vykonať fylogenetickú analýzu a porovnávať vzorky DNA. Požiadavky na aplikáciu sme rozdelili do dvoch skupín. Prvú skupinu tvorili požiadavky, ktoré boli cieleňé na používateľa aplikácie. Používateľ je ten, kto primárne bude využívať aplikáciu na vytváranie fylogenetickej analýzy a ukladanie výsledkov analýzy. Druhu skupinu tvoria požiadavky na aplikáciu, ktoré sú potrebné pre administrátora. Administrátor by mal aplikáciu používať na spravovanie všetkých zaregistrovaných používateľov a na editáciu databázy sekvencií. Po zadelení požiadaviek do týchto kategórií budeme mať lepší prehľad o tom, ktoré požiadavky sú pre nás prioritné.

Návrh používateľského rozhrania tvorí funkčnosť celkovej aplikácie. Požívateľské rozhranie by malo byť vytvorené tak, aby aplikácia bola jednoduchá a zrozumiteľná aj pre používateľa, ktorý nemá toľko informácií o bioinformatike a samotnej fylogenetickej analýze. Aby sme lepšie pochopili celkový cyklus webovej aplikácie rozdelili sme ho do štyroch bodov (Obr. 1):



Obr. 1 Rozhranie používateľa

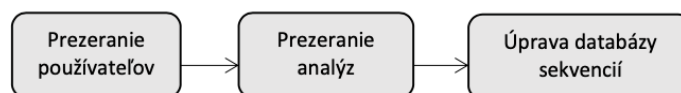
1. Načítanie sekvencií - ako prvý krok bude vytvorenie vstupu pre používateľa, do ktorého zadá sekvenciu, ktorú chce analyzovať. Aplikácia bude požadovať vstupný reťazec sekvencií, ktorý používateľ musí zadať do vytvoreného vstupu aby sa daná sekvencia mohla porovnať s databázou sekvencií v aplikácii.

2. Viacnásobné zarovnanie sekvencií - niektoré sekvencie sa zhodujú alebo len podobajú. Každá vzorka sekvencií ma rozličnú dĺžku, preto musíme upraviť a zarovnať jednotlivé sekvencie na spoločnú dĺžku. Na tento proces využijeme program ClustalW, ktorý má najlepšie výsledky zarovnania sekvencií.

3. Vygenerovanie fylogenetického stromu - po viacnásobnom zarovnaní by nám aplikácia mala vygenerovať fylogenetický strom. Používateľ by mal mať možnosť fylogenetický strom stiahnuť ako obrázok alebo môže výsledok analýzy uložiť v aplikácii.

4. Interaktívny fylogenetický strom – výstupný fylogenetický strom by mal byť interaktívny obrázok. Zobrazenie fylogenetického stromu by sa malo meniť podľa zobrazenia typu grafu a metódy vytvorenia stromu, ktoré bude môcť používateľ meniť. Taktiež tento graf bude mať farebne rozlíšenú vetvu (sekvenciu), ktorú používateľ vložil na začiatku do aplikácie.

Administrátorské rozhranie by sa malo líšiť v základnej funkcionalite od používateľského rozhrania. Hlavnou úlohou administrátora by mal byť prehľad informácií o všetkých zaregistrovaných používateľoch a taktiež o všetkých analýzach, ktoré si používateľ uložil. Okrem tejto funkcionality administrátor by mal mať možnosť upravovať databázu sekvencií. Jednotlivé reťazce sekvencií by mal vedieť editovať, vymazať alebo doplniť nový reťazec do databázy.

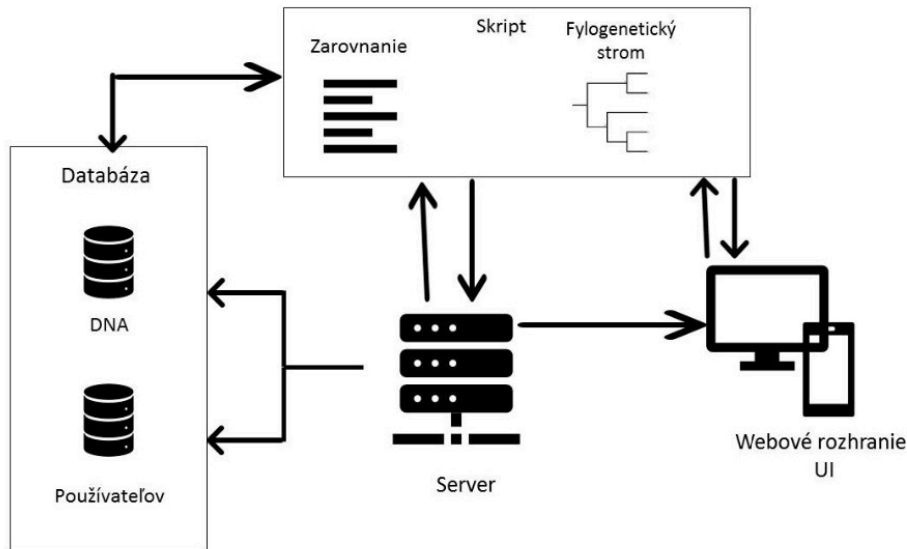


Obr. 2 Rozhranie administrátora

III. IMPLEMENTÁCIA APLIKÁCIE

Architektúra aplikácie nám vyjadruje ako komunikujú navzájom jednotlivé komponenty aplikácie aby dosiahli požadovaný výsledok pre používateľa. Používateľ sa prihlási do aplikácie kedy sa pripojíme k databáze používateľov. V databáze sa overia prihlasovacie údaje a používateľ

sa dostane do aplikácie. Po tomto kroku používateľ načíta sekvenciu, ktorú chce analyzovať a porovnávať do aplikácie. Vstupná sekvencia sa pripojí k databáze sekvencií. Nasleduje volanie vonkajšej aplikácie ClustalW, ktorá spracuje tento súbor a vykoná viacnásobné zarovnanie. Tento súbor zarovnaných sekvencií využíva server na to, aby vytvoril fylogenetický strom. Na serveri potom beží proces, ktorý pomocou vytvorených funkcií balíka APE [5] vygeneruje fylogenetický strom. Následne server vyhodnotí strom pomocou bootstrap metódy a odošle výsledok analýzy (fylogenetický strom) do webového rozhrania.

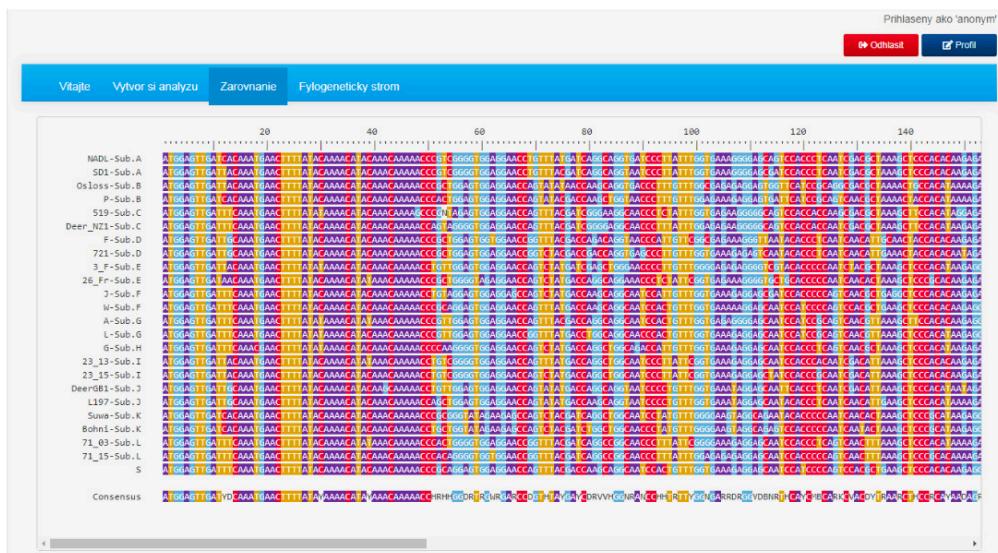


Obr. 3 Architektúra aplikácie.

Pri vytváraní webovej aplikácie sme používali databázu na ukladanie informácií o používateľovi. Na pripojenie k databáze SQLite bolo okrem spomínaných balíkov potrebné nainštalovať knižnicu *sqldf* a ďalšiu knižnicu na šifrovanie používateľských hesiel s názvom *bcrypt*. Databáza obsahuje tabuľku, ktorá má tri stĺpce, ich obsahom je používateľské meno, heslo a dátum kedy sa používateľ zaregistroval.

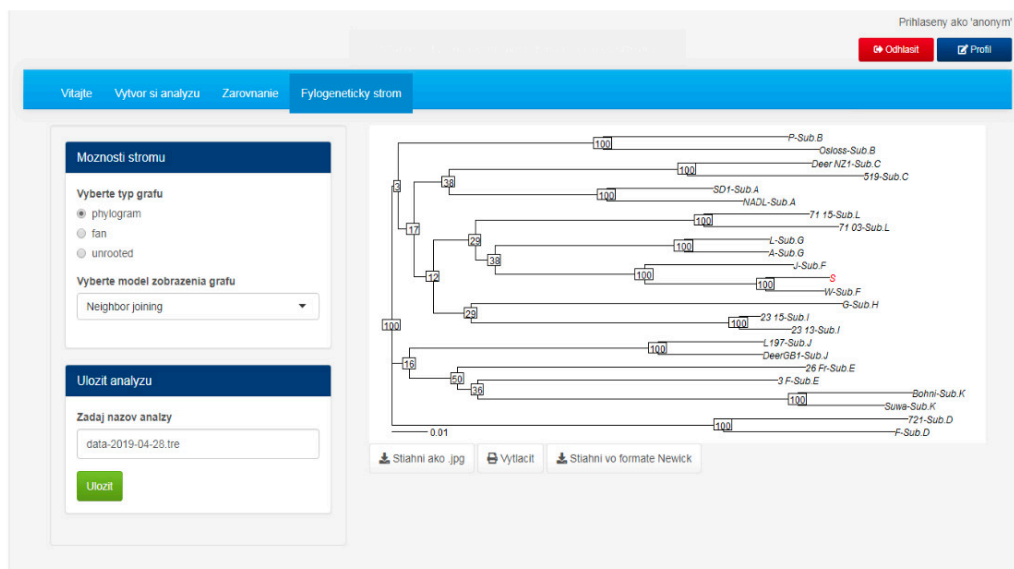
Hlavnou úlohou tejto webovej aplikácie je podpora procesu fylogenetickej analýzy. Prvý krok je vytvoriť priestor v aplikácii, do ktorého budú používatelia vkladat' sekvenciu, ktorú potrebujú analyzovať. Tá sa následne bude porovnávať so sekvenciami DNA uloženými v databáze a pomocou vybraných metód určovať a vizualizovať podobnosť medzi jednotlivými reťazcami. Databáza sekvencií je tvorená z rôznych reťazcov vírusov vo formáte *fasta*.

Po tomto kroku nastáva zarovnanie reťazcov a vytvorenie fylogenetického stromu pomocou rôznych metód. Zarovnanie sekvencií je automatický proces, implementovaný pomocou metódy ClustalW. Využili sme funkciu, ktorú obsahoval použitý fylogenetický balík APE. Vizualizácia zarovnaných sekvencií je zobrazená na Obr. 4.



Obr. 4 Vizualizácia zarovnania reťazcov DNA v aplikácii.

Posledným krokom v aplikácii je konštrukcia a vygenerovanie fylogenetického stromu. Proces vytvorenia fylogenetického stromu sa vykonáva na serveri a používateľ má prístup len k vygenerovanému obrázku fylogenetického stromu. Na vytvorenie fylogenetického stromu sme použili štyri rôzne metódy. Ako prvú predvolenú metódu sme zvolili metódu najbližších susedov (Neighbor-Joining). Po vygenerovaní stromu si používateľ môže zobrazit' strom pomocou ostatných metód, teda UPGMA, maximum parsimony, maximum likelihood. Pre každú metódu zobrazenia stromu sme vytvorili funkciu s názvom metódy. Tieto funkcie sme vytvárali tak, že pri výbere inej metódy sa vygeneruje nový fylogenetický strom so vstupnou sekvenciou. Metódy (funkcie) obsahujú načítanie sekvencií, zarovnanie sekvencií pomocou ClustalW a následne príkazy podľa ktorých sa vytvorí fylogenetický strom. Fylogenetický strom vo webovej aplikácii sme zobrazili ako obrázok. V aplikácii sme implementovali tri možnosti na zobrazenie grafu. Fylogenetický strom môže byť zobrazený ako zakorenený (phylogram), nezakorenený (unrooted) alebo cirkulárny (fan).



Obr. 5 Vizualizácia fylogenetického stromu v aplikácii.

IV. ZÁVER

V tejto práci sme sa venovali návrhu a implementácii webovej aplikácii, ktorá má využitie v bioinformatike. Hlavným cieľom bolo vytvoriť aplikáciu na podporu fylogenetických analýz rôznych sekvencií DNA. Pre implementáciu aplikácie sme vybrali vhodný fylogenetický balík v jazyku R a webovú aplikáciu sme vytvorili v tomto jazyku. Aplikácie, ktoré riešia podobnú problematiku už existujú a sú dostupné. Navrhnutá aplikácia sa líši od ostatných tým, že bola vytvorená v prostredí R. Toto prostredie poskytuje jednoduchšiu použiteľnosť aplikácie, ako aj širšiu podporu metód, ktoré je možné využiť pri vytváraní fylogenetických stromov. Na rozdiel už od vytvorených aplikácii sme sa zamerali tiež na vytvorenie administrátorského rozhrania, ktoré slúži na na editáciu databázy skúmaných sekvencií. Aplikácia je vytvorená tak, aby sa databáza sekvencií mohla jednoducho meniť samotnými používateľmi, z čoho vyplýva, že ju môžeme využiť na analýzu rôznych druhov DNA sekvencií. V budúcnosti by sa aplikácia mohla rozšíriť o rôzne spôsoby zarovnania sekvencií a taktiež o zobrazenie fylogenetického stromu v Newick formáte.

POĎAKOVANIE

Táto práca bola podporená APVV grantom č. APVV-16-0213.

LITERATÚRA

- [1] T. Satyanarayana, A. Prakash, and B. N. Johri, *Microorganisms in sustainable agriculture and biotechnology*. 2013.
- [2] *The Phylogenetic Handbook*. 2009.
- [3] L. Nakhleh and D. A. Morrison, "Phylogenetic Networks," in *Encyclopedia of Evolutionary Biology*, 2016.

- [4] E. Paradis, *Analysis of phylogenetics and evolution with R: Second edition*. 2012.
- [5] E. Paradis, J. Claude, and K. Strimmer, "APE: Analyses of phylogenetics and evolution in R language," *Bioinformatics*, 2004, doi: 10.1093/bioinformatics/btg412.

Riadená plošina pre virtuálnu jaskyňu LIRKIS CAVE

¹Jakub GRIB, ³Branislav SOBOTA, ²Štefan KOREČKO

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, Slovenská republika

¹jakub.grib@student.tuke.sk, ²branislav.sobota@tuke.sk, ³stefan.korecko@tuke.sk

Abstrakt – Článok sa zaoberá návrhom a čiastočnou realizáciou simulátora s dvoma stupňami voľnosti: rotácia okolo osi x a rotácia okolo osi y . Simulátor sa bude nachádzať v laboratóriu LIRKIS v domovskej inštitúcii autorov. Pozostáva zo statickej a pohyblivej základne. Na jeho zhotovenie sa použili dva navijaky, nastaviteľné spojovacie tyče a dve kľuky, pripevnené k navijaku. Je možné pripevniť aj nadstavec s monitormi. Horná základňa je upevnená na homokinetickom kĺbe, pričom ju nakláňajú dva upravené navijaky. Motory kontroluje mikropočítač na báze Arduino prostredníctvom riadiacich obvodov VNH2SP30. Virtuálnu scénu generuje systém CAVE. Z hľadiska polohovania je je objasnené ako ako merať polohu hriadeľa motora potenciometrom. Taktiež je načrtnuté smerovanie budúceho vývoja systému.

Kľúčové slová – simulátor, 2-DOF, navijak, Arduino, virtuálna realita

I. ÚVOD

Simulátory sa využívajú pre viacero účelov a sú predmetom výskumu a vývoja už niekoľko desaťročí. Jedným z hlavných dôvodov pre vývoj týchto zariadení bol v minulosti výcvik pilotov [1]. Simulátory tak zvyšujú bezpečnosť pri tréningu a zároveň dokážu znížiť celkové náklady, ktoré sú s ním spojené. Preto sa simulátor postupne stal neoddeliteľnou súčasťou výcviku. Výhodou simulátorov je, že dokáže vytvoriť podmienky, ku ktorým reálne dochádza len málokedy, alebo je ťažké či nebezpečné ich vyvolať reálne. Napríklad simulácia riadenia pod vplyvom alkoholu. Alebo testovanie sa rôznych senzorov a elektronických systémov. Ďalšie odvetvie priemyslu, ktoré sa zaoberá vývojom simulátorov je herný priemysel. Takéto zariadenia sú síce dostupné širokej verejnosti, no ich cena stále prekračuje hranicu niekoľko tisíc eur. Cena lepších zariadení so šiestimi stupňami voľnosti sa môže pohybovať aj v rozmedzí niekoľko desiatok tisíc. Na internete vznikajú rôzne návody a fóra, v ktorých ľudia diskutujú o spôsoboch, ako je možné skonštruovať podstatnú súčasť takýchto zariadení - pohyblivú plošinu.

Jeden z projektov, využívajúci pohyblivú plošinu, je aj projekt všeobecného simulačného stanovišťa (simulátora), predstavený v tomto článku. Pod všeobecným sa myslí, že sa nebude špecializovať len na konkrétne odvetvie či druh virtuálnej scény. Cieľom projektu je navrhnúť pohyblivú plošinu a celkovo systém, ktorý bude poskytovať pohlcujúci zážitok z virtuálneho prostredia. Okrem všeobecnosti použitia je dôležitou charakteristikou projektu aj relatívne nízka finančná náročnosť. Podobne ako pri leteckých simulátoroch [2] je aj našim cieľom kombinovať fyzický pohyb základne simulátora s vizuálnym a akustickým vnemom. Čo sa týka fyzického pohybu základne, ideálne by bolo implementovať riešenie so šiestimi stupňami voľnosti (6-DoF), či už na báze tradičnej Stewartovej plošiny (Stewart platform) [3], tiež nazývanej pohybová základňa (motion base) [4], alebo exotickejších riešení ako umiestnenie na pohyblivej robotickú ruku [5]. Finančné a priestorové obmedzenia však neumožňovali takúto realizáciu, preto tu predstavené simulačné stanovište ponúka dva stupne voľnosti (2-DoF) pre nakláňanie v smere osí x a y . Konštrukčné riešenie však umožňuje budúce prispôbenie pre viac stupňov.

II. VÝBER VHODNÝCH KOMPONENTOV

V prvom rade bolo potrebné zvážiť, aké komponenty budú použité a či bude vhodné ich použiť vzhľadom k technickému riešeniu nášho simulátora. Veľkosť a tvar komponentov by mal byť prispôbený celkovým rozmerom simulátora. Keďže rozpočet by nemal presiahnuť istú hranicu, bolo potrebné zvažovať aj cenu riešenia. Výber istého komponentu následne dost ovplyvňuje aj výber ďalšieho.

Na začiatku projektu bolo potrebné určiť či nakláňanie hornej plošiny bude riešené na základe otáčavého alebo lineárneho pohybu. Nie každý aktuátor je vhodný pre náš projekt či už veľkosťou alebo cenou. Keďže ťažisko hornej plošiny bude na kĺbe, nakláňanie plošiny sa docieli dvoma zariadeniami, ktoré budú pohybovať plošinou vertikálnym smerom. Preto na začiatku projektu vznikla myšlienka nakláňať plošinu pomocou lineárnych aktuátorov. Pneumatický pohon totiž nie je vhodný kvôli hlučnosti a veľkému objemu kompresora. Navyše nie je ani veľmi presný. Plošina sa musí nakláňať čo najpresnejšie, preto sa táto možnosť vylúčila. Hydraulické sú presnejšie a vo všeobecnosti majú väčšiu silu. Nevýhodou je, že potrebujú čerpadlo a môže tu dôjsť k úniku kvapaliny, čo je v našom prípade neprípustné [6].

Ideálnym riešením pre projekt je použitie servomotorov. Aby sa dosiahla požadovaná rýchlosť, je potrebné použiť aj prevodovku. Alternatívne je možné zakúpiť lineárny servomotor, ktorý by prevodovku nepotreboval. Hlavnou nevýhodou je cena. Spoločne so servomeničom takéto motory stoja aj viac ako 500 Eur. K tomu je potrebné ešte dokúpiť samotnú prevodovku. Krokové motory sú lacnejšie, no stále to presahuje náš rozpočet.

Preto sa použili navijaky. Konkrétne navijaky s výkonom 960 W, ktoré sa napájajú na 12 V. Ak nie sú započítané straty, tak maximálny odber prúdu bude dosahovať hodnotu až 80 A. Keďže nebudú pracovať na plný výkon, ich odber bude nižší. Tieto navijaky majú v sebe už zabudovanú planétovú prevodovku s prevodovým pomerom 153:1. Tá umožňuje dosiahnuť veľký prevodový pomer pričom zaberá málo miesta. Rýchlosť je približne 23 rpm. Takéto nízke otáčky sú pre náš projekt ideálne. Podľa výkonu a otáčok sa vypočítal krútiaci moment, ktorý je 398.58 N m^{-1} . Pri kľuke o dĺžke 20 cm, motor by pôsobil na jej koniec silou 1993 N, čo je v prepočte 203 kg.

Ďalej bolo potrebné vybrať riadiace obvody pre ovládanie týchto elektromotorov. Kvôli vysokému odberu sa musí použiť paralelne viacero riadiacich obvodov. Pre náš projekt sa najviac hodili obvody BTS7960 a VNH2SP30. Napriek tomu, že BTS7960 je výkonnejšie, použili sa kontroléry VNH2SP30, lebo sú viac používané a existuje k nim lepšia dokumentácia. Cena v tomto prípade nie je dôležitá. Oba riadiace obvody sa predávajú aj vo verzii, ktorá má na doske dva čipy. Ak sa použijú motory s napájaním na 24 V nedá sa použiť VNH2SP30. Maximálne napätie pre BTS7960 je totiž $27,5 \text{ V}^1$. Pri VNH2SP30 je to len 16 V^2 .

Dôležitou je aj otázka napájania. Pri použití autobaterie je potrebné ju pred každým použitím nabiť a počas prevádzky stále dobíjať. Lineárny zdroj v porovnaní so spínaným zdrojom je neefektívny a zaberá veľaké miesto [7]. Preto bol vybraný 12 V spínaný zdroj. Jeho výkon musí byť rovný alebo väčší než je výkon nášho navijaku. V našom prípade sa jedna o zdroj s výkonom aspoň 1000 W. Pri kúpe takéhoto typu zdroja si je potrebné uvedomiť, že pre pripevnenie vodičov, ktoré slúžia na napájanie motora sa musí zdroj upraviť. Ďalšia možnosť je zakúpiť adapter, ktorý je presne určený pre náš typ zdroja a obsahuje svorky, do ktorých je možné pripevniť vodiče. Ten je možné jednoducho odobrať. V našom prípade cena adaptéra je vyššia ako cena samotného zdroja.

Pre náš projekt je vhodnejšie použiť lineárny potenciometer. Pri výbere je potrebné dbať ohľad aj na jeho linearitu. Čím je nižšia, tým je aj odchýlka hodnoty nižšia, no cena rastie. Ak by sa potenciometer nahradil enkóderom, bolo by potrebné upraviť zdrojový kód pre čítanie pozície. Hlavnou výhodou enkóderu je nepretržité otáčanie hriadeľa. Inkrementálny enkóder si vyžaduje neustále monitorovať zmeny. To neplatí pri absolútnych enkóderoch, pomocou ktorých je možné merať pozíciu priamo [8]. Cena jedného je v zahraničí okolo 30 Eur, ale nájdu sa aj za 15 Eur.

III. FUNKCIONALITA SIMULÁTORA

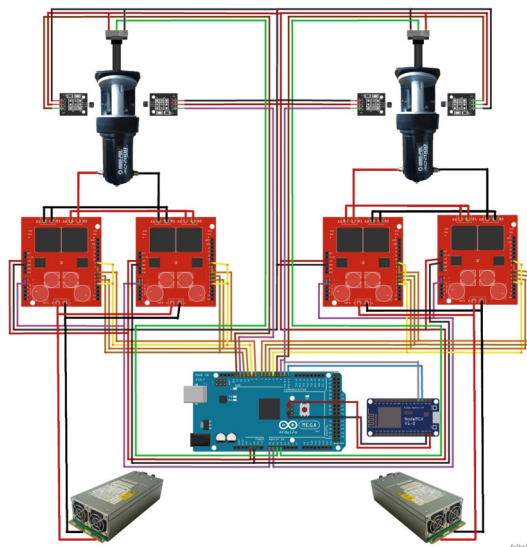
Náš simulátor je možné využívať pre viacero účelov. Dokáže sa nakláňať v smere osi x a y . Pozostáva zo statickej a pohyblivej základne. Horná základňa drží na homokineticom kĺbe, ktorý sa nachádza približne v strede plošiny. Nakláňanie hornej plošiny zabezpečujú dva navijaky, ktoré sú prerobené na polohovateľné motory vďaka potenciometru, ktorý je pripojený na hriadeľ navijaka. Na bubon navijaka je pripevnená kľuka, pomocou ktorej je možné regulovať veľkosť sklonu. Závitová tyč prepája kľuku s hornou plošinou cez kľbové oká. Motory kontroluje Arduino prostredníctvom riadiacich obvodov. Ďalšie senzory, ktoré kontrolujú pozíciu hriadeľa sú hallove sondy. Preto sú na druhej strane bubna pripevnené magnety. Magnety spoločne so sondami vymedzujú rozsah pohybu hriadeľa. Ten je možné ovládať pomocou príkazov, ktoré je potrebné zasielať priamo do riadiaceho mikropočítača typu Arduino. Kompletná schéma celého systému je zobrazená na Obr. 1.

Simulátor sa dá používať dvoma spôsobmi. Prvý spôsob je umiestniť plošinu do virtuálnej jaskyne (CAVE), ktorá pozostáva z 3D obrazoviek rozmiestnených do polkruhu. V tomto prípade bude obraz generovať jaskyňa, ktorá tiež bude ovládať simulátor. Simulátor je možné použiť

¹<https://www.infineon.com>

²<https://www.pololu.com>

aj v kombinácii s inými zariadeniami. Na hornú plošinu bude možné pripevniť aj nastaviteľný stojan pre monitory. V tomto prípade obraz nebude generovať CAVE. Komunikácia s riadiacim mikropočítačom môže prebiehať cez USB kábel alebo bezdrôtovo pomocou UDP protokolu. Nato je potrebné k riadiacemu mikropočítaču pripojiť WiFi modul ESP. Komunikácia cez kábel je všeobecne rýchlejšia. Príkazy budú zahrňovať už konkrétne hodnoty potenciometra, ktoré predstavujú polohu hriadeľa a motora.



Obr. 1: Kompletná schéma zapojenia

IV. ÚPRAVA KOMPONENTOV

Aby bolo možné zostaviť simulátor s požadovanou funkcionalitou, bolo potrebné realizovať viaceré modifikácie získaných komponentov.

A. Polohovateľný motor

Aby bolo možné navijak upraviť na polohovateľný motor, je nutné najprv rozobrať navijak a odstrániť lano. Ďalej sa vyrobí držiak pre potenciometer, ktorého hriadeľ bude prepojený s bubnom navijaku. Pri vyrábaní držiaka sa použil plech s hrúbkou približne 1 mm. Plech by mal byť pružný, aby sa potenciometer kvôli nepresnostiam pri pohybe nepoškodil. Veľkosť držiaka je potrebné prispôsobiť dĺžke oboch hriadeľov. V našom prípade je držiak vyrobený z plechu s rozmermi o veľkosti 2 x 20 cm. Na prepojenie je možné použiť aj flexibilnú hriadeľovú spojku, no v našom prípade nevyhovovala veľkosť žiadnej bežne dostupnej spojky. Preto sa použila káblova spojka.

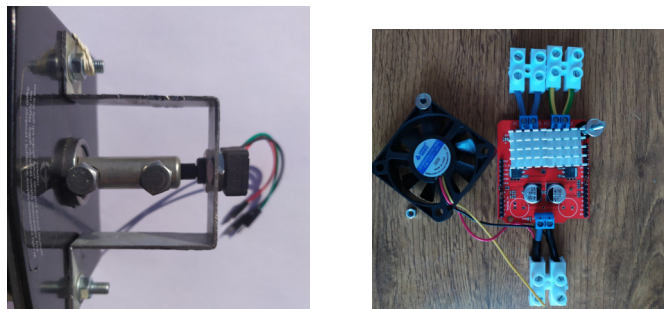
Keďže sa nedá spoliehať na to, že potenciometer bude stále prepojený s hriadeľom, na navijak sa pripevnia aj dve hallové sondy. Jedna bude vymedzovať horný a druhá dolný doraz. Na bubne sú pripevnené dva magnety. Čím je magnet menší, tým sú hranice vymedzené presnejšie. Použili sa neodymové magnety s priemerom 3 mm. Medzi sondou a magnetom bol ponechaný priestor približne 2 - 3 mm. Ako hallové sondy sa použili moduly KY-003. Pre testovanie sa na bubon pripevnili gumičky, ktoré ho priťahujú smerom dole, pretože kvôli nepresnostiam je bubon možné pootočiť rukou o uhol zhruba 5°.

B. Prispôsobenie VNH2SP30 a vodičov

Na riadiaci obvod motora sa najprv priletovali kolíky, ktoré sú súčasťou balenia. Aby sa na Arduino použil čo najmenší počet pinov, bolo potrebné prepojiť konkrétne piny. Na dosku sa priletovali aj 5 mm svorky. Na čipy sa naniesie tenká vrstva teplovodivej pasty. Lepšie je použiť pastu bez metalických častíc, napr. *ARCTIC MX-4*. Na nanosenú vrstvu sa pripevnili hliníkové chladiče. V našom prípade majú veľkosť 20x20 mm.

Ďalej sa použije PVC kábel, ktorého hrúbka jednej žily je 2.5 mm². Nastrihá sa približne na 4 cm kusy tak, aby vzniklo 24 žíl. Aby sa nám vodiče nerozhádzali, pomocou krimpovacích kliešti sa zakrmpuje iba jeden koniec každého vodiča. Môžu sa na to použiť napríklad fastony na 2.5 mm² káble. Tieto konce sa vložia do svorkovnice s prierezom 6 - 10 mm², lebo pri skladaní a rozkladaní je jednoduchšie pripevniť káble do svoriek s väčším priemerom. Svorkovnice sa rozdelia na menšie časti, konkrétne na 12 kusov po dve svorky. Druhý koniec sa vloží do svoriek na riadiacom obvode motora. Na prepojenie riadiaceho obvodu so zdrojom a navijakom sa použili

vodiče z jeho ovládača. Tie je potrebné prerozdeliť do rovnakých zväzkov, konce zakrimpovať a zaizolovať.



Obr. 2: Držiak a VNH2SP30 po prerobení

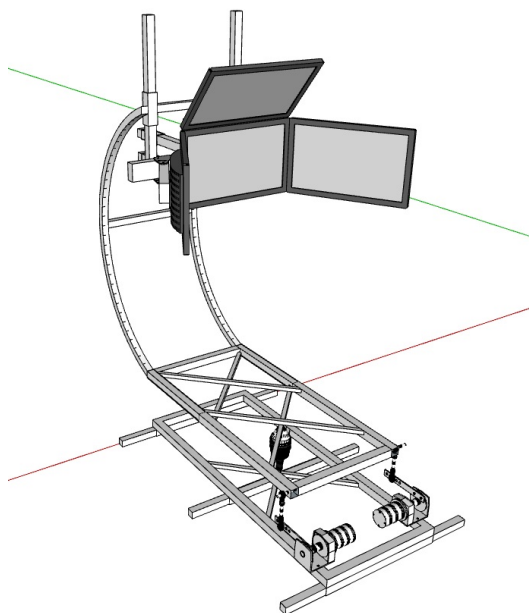
1) *Prerábanie serverového zdroja na 12V napájací zdroj:* Serverové zdroje je najprv potrebné zapojiť podľa schémy. Tú je možné vyhľadať na internete. Aby sa na zdroj mohli pripojiť káblové oká, je potrebné ho ešte ďalej prispôsobiť. Použil sa tenký medený plech a plech zo silnejšieho materiálu s hrúbkou aspoň 1 mm. Napríklad pozinkovaný oceľový plech. Ďalej sa použili skrutky *M4x40*. Je potrebné mať 8 skrutiek a matíc aspoň 20. Ďalej sa vyrezali plechy 25x15 mm. Šírka pinov je približne 10 mm. Do plechov sa vyvrtávajú otvory pre skrutky. Následne sa vyrezané plechy zložia do celkov, tak ako je tomu na obrázku Obr. 3. Celky, ktoré vznikli sa nasunuli na zdroj a zospodu sa prispájkovali.



Obr. 3: Zdroj - pohľad spredu

V. KONŠTRUKCIA SIMULÁTORA

Simulátor je navrhnutý tak, aby bol použiteľný v aj mimo CAVE. Pre použitie mimo CAVE je potrebné navrhnuť aj nadstavec na monitory. Veľkosť a tvar simulátora musia byť vhodné aj pre použitie vo virtuálnej jaskyni LIRKIS CAVE na pracovisku autorov, ktorá má približne rozmery kocky o strane 250cm. Zároveň simulátor musí mať dostatočnú stabilitu na to, aby sa udržal na mieste. Konštrukcia je navrhnutá z oceľových profilov o veľkosti 40x40 mm.



Obr. 4: Kostra simulátora

Rozmery spodnej základne sú 1400x600 mm. Aby sa spevnil rám, uprostred sa umiestnil kríž, na ktorý sa použijú profily s výškou 20 mm. Profily zvierajú pravý uhol, pričom jeden je umiestnený zospodu a druhý zvrchu. Pod spodnú základňu sa umiestnili tri profily, ktoré majú rovnakú šírku ako rám. Kvôli stabilite do týchto profilov bude možné vložiť a upevniť ďalší profil, respektíve nejaký hranol. Na hornej základni sa kríž umiestni bližšie k zadnému profilu základne, pretože používateľ bude pravdepodobne ťažší ako stojan s monitormi. Vzdialenosť kríža od stredu je 150 mm, čo je 450 mm od zadnej steny rámu. Keďže sa jedná o hornú základňu bude na ňu vyvíjaný väčší tlak. Pre spevnenie sa do jej prednej časti pridal ešte jeden kríž. Aby sa na simulátor dalo postaviť, rám bude z vrchu “zakrytovaný”, čo spevní konštrukciu. Pre naše potreby rám musí udržať približne 200 kg.

Stojan má tvar poloblúku. To preto, aby v mieste spájania s hornou základňou tu nevznikal veľký tlak. Poloblúk má výšku 1300 mm a je v základni zasunutý približne 500 mm. Vzdialenosť monitorov od používateľa je možné regulovať v hornej časti držiaka, alebo vysúvaním celého nastavca zo základne. Aby nedošlo k poškriabaniu, na koniec hornej základne zospodu sa pripevnia gumené podložky. Gumené podložky bude potrebné umiestniť aj medzi motor a konštrukciu. Motory sú dosť hlučné a guma dokáže zmierniť prenos vibrácií na konštrukciu.

Kľuka motora je prepojená s hornou plošinou cez kľbové oko s priemerom 12 mm. Kľbové oká sú prepojené so závitovou tyčou, ktorá je dlhá približne 125 mm. Ak je oko pripevnené na “strednom” otvore kľuky a horná plošina je v horizontálnej polohe, závitová tyč musí zvierat s kľukou pravý uhol. Druhá tyč prechádza skrz celú plošinu až na druhú stranu. Tu je pripevnené ďalšie oko. Kľuku je možné vyrobiť napríklad z pásoviny. Môže sa použiť pásovina 30x5 mm, ktorej dĺžka je 240 mm. Otvory sú od seba vzdialené 20 mm, pričom priemer jedného je 12 mm. Najviac vzdialený otvor od stredovej osi bubna je vzdialený približne 190 mm. Vzdialenosť medzi základňami, respektíve výška kĺbu, je 230 mm. Tiež je možné použiť kardanový kĺb. Na obrázku je homokinetický kĺb hnacieho hriadeľa.

VI. PRESNOSŤ MOTOROV

Motor je potrebné vypínať ešte pred tým než dosiahne nastavenú hodnotu. Experimentálne sa zistilo, že je potrebné ho vypnúť vtedy keď je rýchlosť motora rovná približne polovici vzdialenosti od nastavenej pozície. Táto vzdialenosť sa ladí pomocou konštanty. Závislosť tejto konštanty je vyjadrená v nasledujúcej tabuľke.

Tabuľka 1: Tabuľka meraní - závislosť konštanty od rýchlosti

Rýchlosť \ Vzdialenosť	50/L	MAX/L	50/R	MAX/R
60	10	-5	10	5
120	23	-11	23	0
180	23	-5	23	5
255	25	15	25	25

Predchádzajúca tabuľka zobrazuje vzťah konštanty a rýchlosti. Motory boli testované pri 4 rýchlostiach, pričom maximálna rýchlosť je 255. Písmeno L predstavuje ľavý motor a R označuje motor na pravej strane. Pri konštantách, ktoré sa nachádzajú v tabuľke, motor dosahoval odchýlku zhruba 1° . Z predchádzajúcej tabuľky je vidieť, že obidva motory majú rovnaké konštanty pri posune o minimálnu vzdialenosť (pre testovanie sme zvolili minimálnu vzdialenosť 50 jednotiek a maximálnu 350 jednotiek potenciometra). Pri maximálnej vzdialenosti je rozdiel medzi pravým a ľavým motorom približne o 10 jednotiek. Z toho vyplýva, že motory nemajú rovnaký dobeh.

Hodnoty sa zobrazili pomocou internetovej služby *Wolfram Alpha*. Z daných hodnôt sa zistovalo, o aký priebeh funkcie ide. Tento nástroj dokáže vytvoriť okrem lineárnej aj logaritmickú, exponenciálnu, kvadratickú a kubickú funkciu, ktorá sa najviac zhoduje so zadanými hodnotami. Následne zobrazí jej priebeh a odchýlku zadaných bodov od priebehu funkcie. Ďalej sa vyberie priebeh s najmenšími odchýlkami. Pri niektorých priebehoch bolo ťažké určiť matematickú funkciu zo štyroch hodnôt, preto bolo potrebné zistiť ďalšie hodnoty.

V prípade minimálnej vzdialenosti sa zistilo, že ide o logaritmický priebeh. Pri maximálnej vzdialenosti v oboch prípadoch išlo o kvadratický priebeh funkcie, pričom funkcia pri pravom motore je posunutá o desať jednotiek na osi y smerom hore.

Ďalej sa zistovalo, či konštanta rastie lineárne vzhľadom k pozícii pri rovnakej rýchlosti. Aby priebeh funkcie bolo vidno čo najlepšie, vybral sa motor a testovacia rýchlosť, kde je najväčší rozdiel medzi minimálnou a maximálnou vzdialenosťou. Preto sa na ľavom motore nastavila rýchlosť 120. Meraniami sa zistilo, že lineárny priebeh nemá ani závislosť konštanty od rýchlosti a ani závislosť konštanty od vzdialenosti pri rovnakej rýchlosti. Výpočet, pomocou ktorého by sa dosiahlo presné polohovanie, je pre Arduino dosť náročný. Preto sa za konstantu dosadil aritmetický priemer najmenšej a najväčšej hodnoty.

Hriadeľ nášho potenciometra sa dokáže otáčať o 300° . Prevodník na našom Arduine je 10-bitový, teda výstup potenciometra je rozdelený na 1024 hodnôt. Keďže 7 je stredná hodnota znamená to, že presnosť ľavého motora je 5.27° . Na pravom motore je stredná hodnota 13 a krajná 0. Preto je presnosť tohto motora 3.81° . V skutočnosti môže byť pri oboch motoroch aj o niečo vyššia, keďže nie sú otestované všetky rýchlosti, no vo väčšine prípadov by mala byť menšia.

VII. ZÁVER

V budúcnosti kvôli zvýšeniu presnosti je možné použiť aj enkóder, no zatiaľ to nie je potrebné, keďže motory nezastavujú úplne presne. Namiesto prerobených navijakov sa dajú použiť lineárne servomotory, ktoré sú tichšie a presnejšie. Taktiež nie je nutné riadiace obvody motorov napájať len cez mikrokontrolér. Jednoducho je možné využiť piny $+3.3V$ a $+5V$, ktoré sú priamo na zdroji. Takýmto spôsobom sa dá nahradiť Arduino modulom ESP32. Keďže tento čip má dve jadrá s taktom 240 MHz, už len jedno jadro je neporovnateľne rýchlejšie než Arduino Mega alebo UNO. Zatiaľčo jedno jadro by mohlo kontrolovať všetky motory a senzory, druhé môže komunikovať s počítačom, nastavovať novú polohu a meniť vlastnosti motorov.

Ďalším nedostatkom našej práce je, že celý systém sa spolieha iba na jediný mikrokontrolér. Kritický môže byť hlavne moment, ak dôjde k zahlteniu komunikácie. Vykonávanie sa môže natoľko spomaliť, že doska sa prestane venovať kontrole pozície hriadeľa.

Celková cena simulátora by nemala presiahnuť hranicu 2000 Eur. Najdrahšou súčasťou je konštrukcia, ktorá je zväšaná a je nutné ju povrchovo upraviť. Celá konštrukcia sa dá umiestniť ešte na jednu základňu. Simulátor by tak získal ďalší stupeň voľnosti, otáčanie okolo zvislej osi. Pre otáčanie je potrebný ešte jeden motor. Kvôli veľkosti základne by pravdepodobne uhol otáčania plošiny nebol väčší ako 90° . Simulovanie šoférovania, lietania alebo bicyklovania si vo väčšine prípadov ani nevyžaduje väčší uhol.

Implementáciou washout filtra je možné šetriť priestor, keďže simulátor po vykonaní pohybu sa vráti na štartovnú pozíciu. Práve na implementáciu tohto filtra môže byť zameraný ďalší vývoj simulátora. Predtým je potrebné vyskúšať, ako sa budú motory správať pri veľkej záťaži a podľa toho nastaviť konštanty v zdrojovom kóde tak, aby polohovanie bolo čo najpresnejšie. Motory sú testované bez záťaže, preto hodnoty sa môžu líšiť aj s narastajúcou váhou plošiny.

POĎAKOVANIE

Táto práca bola podporená z KEGA grantu č. 035TUKE-4/2019: "Virtuálno-reálné technológie a vzdelávanie postihnutých ľudí".

LITERATÚRA

- [1] A. T. Lee, *Flight simulation: virtual environments in aviation*. Routledge, 2017.
- [2] D. Allerton, *Principles of flight simulation*. John Wiley & Sons, 2009.

- [3] B. Dasgupta and T. Mruthyunjaya, "The stewart platform manipulator: a review," *Mechanism and machine theory*, vol. 35, no. 1, pp. 15–40, 2000.
- [4] M. Becerra-Vargas and E. Morgado Belo, "Application of h theory to a 6 dof flight simulator motion base," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 34, no. 2, pp. 193–204, 2012.
- [5] F. M. Nieuwenhuizen and H. H. Bülthoff, "The mpi cybermotion simulator: a novel research platform to investigate human control behavior," *Journal of Computing Science and Engineering*, vol. 7, no. 2, pp. 122–131, 2013.
- [6] S. Rosenfeld, "Advantages and drawbacks of pneumatic, hydraulic, and electric linear actuators," 7 2017. [Online]. Available: https://www.timotion.com/en/news/news_content/blog_articles/general/advantages_and_drawbacks_of_pneumatic_hydraulic_and_electric_linear_actuators?upcls=1481266229&guid=1499762723
- [7] Anusha, "Switch mode power supply (smps)," 5 2017. [Online]. Available: <https://www.electronicshub.org/switch-mode-power-supply-smps/>
- [8] J. S. Cook, "Encoders vs. potentiometers: Which is right for your project?" 5 2019. [Online]. Available: <https://www.arrow.com/en/research-and-events/articles/encoder-vs-potentiometer-how-to-choose>

Tvorba a vizualizácia troj-dimenzionálnych máp prostredia

¹René FEDORČÁK, ¹Branislav MADOŠ, ¹Anton BALÁŽ

¹Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 01 Košice, Slovak Republic

¹rene.fedorcak@student.tuke.sk, ¹branislav.mados@tuke.sk, ¹anton.balaz@tuke.sk

Abstract – Článok sa zaoberá problematikou vytvárania máp prostredia. Vo svojej úvodnej časti popisuje v rámci tejto práce navrhnuté hardvérové vybavenie mapovacieho zariadenia, zloženého z platformy Arduino Mega 2560, laserového senzora VL53L1X a dvoch servomotorov Tower Pro SG 90. V ďalšej časti popisuje spôsob získavania dát a ich vizualizáciu s využitím na tento účel vyvinutého softvérového vybavenia.

Keywords – Arduino, Arduino Mega 2560, mapovanie prostredia, vizualizácia, simultánna lokalizácia a mapovanie, SLAM

I. INTRODUCTION

Medzi najdôležitejšie úlohy v rámci mobilnej robotiky patrí lokalizácia mobilného robota a vytváranie máp prostredia, v ktorom sa mobilný robot pohybuje. Mobilný robot, ktorého úlohou je bezpečne sa pohybovať vo vnútornom prostredí, musí teda byť schopný vnímať svoje okolie, spoľahlivo lokalizovať svoju polohu a vytvárať a udržiavať si vnútornú reprezentáciu svojho pracovného prostredia [1]. Mapovanie prostredia možno považovať za vytváranie jeho zjednodušeného modelu. Pri vytváraní máp môže dochádzať aj k transformácii dát získaných o troj-dimenzionálnom priestore do dvoj-dimenzionálneho modelu tvoriaceho mapu. Mapy prostredia možno klasifikovať na topologické, metrické a hybridné [2]. Zložitým problémom sa mapovanie stáva, ak je pri vytváraní mapy potrebné integrovať údaje získavané senzormi umiestnenými na mobilnej platforme pri jej pohybe v mapovanom prostredí, keď môže dochádzať k simultánnej lokalizácii a mapovaniu prostredia - Simultaneous Localization and Mapping (SLAM) [3][4][5]. Jednoduché a finančne nenáročné riešenie pre mapovanie prostredia v rámci budovy je popísané v [6].

Cieľom tohto článku je zhrnúť popis zostrojenia mapovacieho zariadenia a počítačovej aplikácie, pomocou ktorej bude možná tvorba a vizualizácia trojdimenzionálnych máp prostredia. Mapovacie zariadenie bude riadené pomocou platformy Arduino Mega 2560, na ktorú budú pripojené dva servomotory a laserový senzor VL53L1X. Senzor umožňuje meranie vzdialeností a servomotory zabezpečia zmenu orientácie senzora ako vertikálne, tak aj horizontálne. Následne sa získané údaje spracujú a využijú na vykreslenie trojdimenzionálnej mapy prostredia pomocou v rámci tejto práce pripravenej počítačovej aplikácie.

II. MAPOVACÍ SYSTÉM

V rámci tejto práce navrhnutý mapovací systém sa skladá z hardvérovej časti, ktorej hlavným prvkom je snímací senzor, aktuátory, platforma Arduino pre spracovávanie signálov v reálnom čase a softvérové vybavenie umožňujúce komunikáciu s host počítačom a tiež spracovávanie nameraných dát a vizualizáciu zostavenej mapy.

A. Hardvérové vybavenie

Na zostrojenie navrhnutého mapovacieho zariadenia boli využité tieto komponenty (Figure 1):

- Platforma Arduino Mega 2560,
- Laserový senzor VL53L1X,
- 2 x servomotor Tower Pro SG 90,
- Pan a tilt kit,
- Prepojovacie káble,
- USB-B kábel.

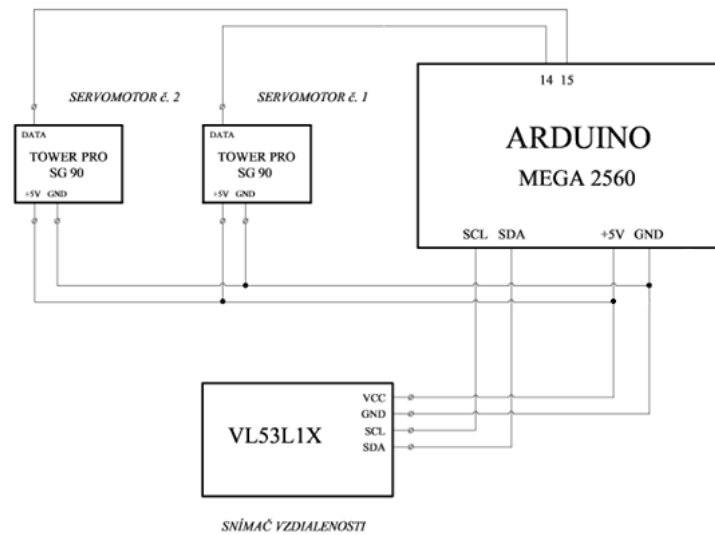


Fig. 1 Celková schéma zapojenia hardvérového vybavenia mapovacieho systému navrhnutého v rámci tejto práce.

Laserový senzor (VL53L1X) patrí medzi modernejšie Time of Flight senzory, pričom patrí do radu ST FlightSense™. Pre získanie údajov o vzdialenosti objektu od senzora využíva meranie času letu laserového lúča k prekážke a späť. Vlnová dĺžka laserového lúča je v infračervenej oblasti a predstavuje 940nm. Maximálna vzdialenosť, ktorú dokáže senzor odmerať, je 4m, pri frekvencii merania na úrovni 50Hz. Technická špecifikácia tohto senzora je uvedená v tabuľke (Tab. 1).

Pripojenie senzora je realizované prostredníctvom I2C zbernice, ktorá využíva dva signály - Synchronous Data (SDA) a Synchronous Clock (SCL). SDA signál tvorí sériovú dátovú linku, ktorá slúži na prenos dát a SCL tvorí sériovú linku, ktorá je využívaná na prenos hodinového signálu. V princípe fungujú tak, že obe linky sú prepojené pomocou pull - up rezistorov na kladný pol napájacieho napätia. Komunikácia realizovaná medzi Arduino Mega 2560 platformou a laserovým senzorom VL53L1X je obojsmerná.

Senzor využíva 6 pinov. Obsahuje dva základné piny, ako u skoro všetkých periférnych zariadení pripojených k platforme Arduino, a to 5V na napájanie a GND pre uzemnenie. Zvyšné 4 piny tvoria vstupné impulzy a výstupné dáta, kde vstupné impulzy majú označenie SCL (I2C clock input) a Xshut (shutdown pin, active low), a výstupné dáta majú označenie SDA (I2C serial data) a GPLO1 (interrupt output). Vzhľadom na to, že sa používa 5V, sú pripojené iba SCL a SDA k platforme Arduino.

Table 1
 Technická špecifikácia senzora VL53L1X

Rozmery	20 mm x 24 mm
Prevádzkové napätie	3.3 V /5V
Veľkosť montážnych otvorov	2.0 mm
Vzdialenosť	+5 mm
Vlnová dĺžka lasera	940 nm
Zorné pole	27 stupňov
Presnosť	Od 15 do 27 stupňov
Prevádzková teplota	-20 80 stupňov celzia

Keďže mapovanie prostredia je realizované zo stabilnej polohy zariadenia, bolo potrebné zabezpečiť dynamickú zmenu orientácie senzora, mapujúceho okolité prostredie, v priestore. Ako aktuátory slúžia dva servomotory typu Tower Pro SG 90. Parametre týchto servomotorov je možné vidieť v Tab. 2.

Table 2
Technická špecifikácia Tower Pro SG90

Model	Tower Pro SG90
Typ	Mikro servomotor
Hmotnosť	9 g
Rozmery	23 × 12.2 × 29 mm
Rýchlosť	0.1 sec/60 stupňov(4.8 V)
Teplotný rozsah	0 - 55 stupňov celzia
Napätie	4.8 V
Šírka mŕtveho pásma	1 μ s
Dĺžka servo vodiča	25 cm
Blokovací moment	1.8 kg/cm(4.8 V)
Gear type	POM gear set

Servomotor má tri základné piny - zem, napájanie a dáta. Hnedý vodič sa pripája na pin GND na Arduino doske. Po pripojení hnedého vodiča, sa pripojí červený vodič na 5V. Posledný oranžový vodič, ktorý slúži na ovládanie servomotora, je pripojený na PWM výstup Arduino dosky.

Pre správne fungovanie zariadenia je nutné, aby bolo možné meniť uhol natočenia snímacieho prvku tak v horizontálnej, ako aj vertikálnej ose. Na tento účel sú použité dva servomotory, ktoré sú vsunuté do pan - tilt kitu. Spodný servomotor umožnil realizovať otáčanie v rámci osi x. Na ňom je potom pripevnený ďalší servomotor, ktorý umožňuje otáčanie v ose y. Na tomto servomote je pripevnená platforma, na ktorej je umiestnený samotný merací prvok, ktorým je laserový senzor vzdialenosti VL53L1X. Celá konštrukcia je potom v záujme jej stability pripevnená k podstavcu, tak ako to ukazuje Figure 2.

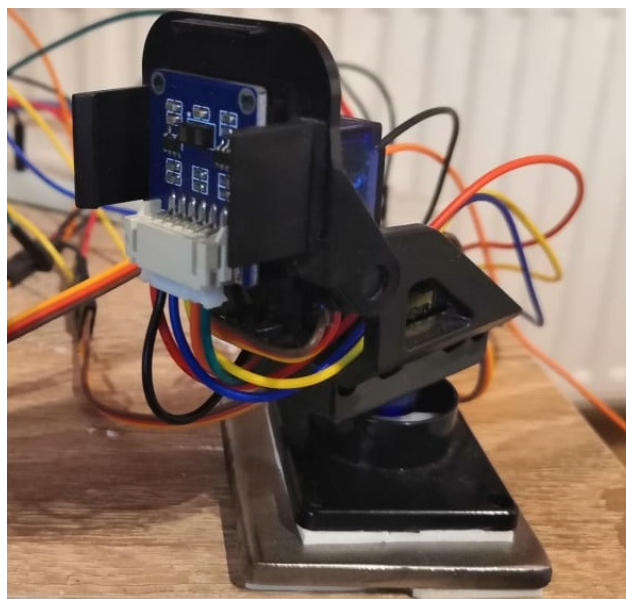


Fig. 2 Pan a tilt kit

B. Získanie, spracovanie a prenos dát

Získavanie informácií z prostredia je založené na systematickom upravovaní uhla natočenia senzora v jednej rovine v uhle z rozsahu $\langle 0; 180 \rangle$ stupňov a podobne v ďalšej rovine, kolmej na predošlú, v uhle z rozsahu $\langle 0; 180 \rangle$ stupňov. Po každej zmene natočenia senzora je realizované meranie vzdialenosti bodu, od ktorého sa odrazí lúč laserového senzora. Vytvorí sa tak trojica číselných údajov, tvoriacich súradnice obsadeného bodu v okolí meracieho zariadenia, pričom je použitý sférický súradnicový systém.

$$\tau(\rho, \phi, \theta) \quad (1)$$

kde

ρ je vzdialenosť bodu od meracieho senzora

ϕ je azimut - odklon sprievodcu bodu od osy x

θ je sklon - odklon sprievodcu bodu od osy z

Následne je táto súradnica bodu prepočítaná zo sférických súradníc na karteziánske súradnice x, y, z :

$$x = \rho * \sin\phi * \cos\theta \quad (2)$$

$$y = \rho * \sin\phi * \sin\theta \quad (3)$$

$$z = \rho * \cos\phi \quad (4)$$

Arduino Mega 2560 ako Single Board Computer (SBC) platforma slúži na nastavovanie uhlov obidvoch servomotorov na požadovanú hodnotu, periodickým generovaním riadiacich signálov pre tieto servomotory v reálnom čase. Medzi ďalšie realizované operácie patrí inicializovanie merania laserovým senzorm a odčítanie dosiahnutých výsledkov. Následne na tomto SBC prebieha prevod získaných sférických súradníc na karteziánske súradnice obsadeného bodu a tieto sú zasielané do host počítača komunikačným káblom typu USB-B.

C. Softvérové vybavenie

Hardvérová časť mapovacieho systému je doplnená softvérovým vybavením, ktoré možno rozdeliť do dvoch častí:

- Arduino IDE softvér, ktorý slúži na riadenie celého zariadenia od zabezpečovania pohybu oboch servomotorov, cez meranie vzdialenosti až po samotnú transformáciu a zasielanie dát do sériového portu.
- Processing softvér, ktorý slúži od získania údajov od Arduino platformy zo sériového portu až po samotnú vizualizáciu trojdimenzionálnej mapy prostredia. V rámci vytvorenej aplikácie je možné nielen sledovať postupné vytváranie mapy, ale taktiež aj uložiť mapu do súboru.

Platforma Arduino obsahuje softvér na riadenie pohybu servomotorov ako vertikálne, tak horizontálne. Nachádza sa tu tiež funkcia na vykonávanie meraní pomocou senzora, ktorá je súčasťou funkcie na prevod zo sférických súradníc do karteziánskych a ich zasielanie. Na začiatku tvorby mapy sa vrchný servomotor nastaví na základnú pozíciu (0) a spodný servomotor, ktorý sa pohybuje po osi X, sa presunie z pozície 0° na pozíciu 180° . V tejto výškovej úrovni sa merajú jednotlivé vzdialenosti pomocou snímacieho zariadenia. Okamžite po dosiahnutí 180° , vrchný servomotor sa posunie na ďalšiu pozíciu a spodný servomotor sa vracia z pozície 180° na 0° , počas tohto presunu sa taktiež merajú jednotlivé vzdialenosti. Tento proces sa opakuje, až kým vrchný servomotor nedosiahne svoju maximálnu hodnotu, v tomto prípade 180° . V tejto časti je zahrnuté taktiež spracovanie a prenos údajov. Komunikáciu medzi počítačom a platformou Arduino zabezpečuje USB-B kábel. Následná komunikácia prebieha pomocou sériového portu.

Po stlačení príslušného tlačidla vo vytvorenej aplikácii a nastavení inkrementu pre pohyb servomotorov, sa začne už vyššie spomínaný proces, pomocou ktorého sa začnú spracovávať údaje a sú zasielané do sériového portu. Údaje v tvare - uhol spodného servomotora, uhol vrchného servomotora a vzdialenosť, sú ešte pred zaslaním upravené (premena zo sférických do karteziánskych súradníc), a tak v spomínanom tvare zaslané do sériového portu (Figure 3).

Na spracovanie a vizualizáciu údajov bola vytvorená aplikácia v Processingu, ktorá spracováva údaje získané cez sériový port, vytvára z nich troj-dimenzionálnu mapu a tú vizualizuje v interaktívnom režime, v ktorom môže používateľ túto mapu prehliadať z rôznych uhlov a realizovať jej priblíženie a oddialenie. Aplikácia umožňuje aj ukladanie získaných informácií do súboru.

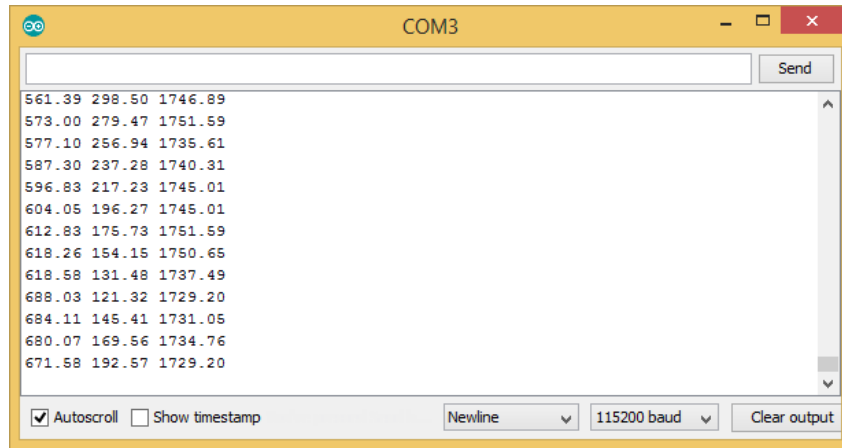


Fig. 3 Výpis dát zasielaných prostredníctvom sériového portu.

III. ZÍSKANÉ VÝSLEDKY

Po zostavení hardvérového a softvérového vybavenia bolo realizované testovanie za rôznych podmienok a s rôznym usporiadaním mapovaného prostredia s cieľom otestovať spoľahlivosť merania a možnosti vizualizácie zostavenej mapy prostredia. Jedno z mapovaní bolo uskutočnené v miestnosti s rôznymi objektmi, pričom fotografiu mapovanej miestnosti možno vidieť na Figure 4. Celkové mapovanie trvalo 9 minút a 35 sekúnd. Výsledné bodové zobrazenie je vykreslené na Figure 5.



Fig. 4 Fotografia miestnosti použitej pri testovaní v práci popisovaného mapovacieho zariadenia.

IV. CONCLUSION

Cieľom tohto článku bolo oboznámiť s komponentmi na vytvorenie mapovacieho zariadenia a príslušnej počítačovej aplikácie, ktoré spoločne umožnia tvorbu a vizualizáciu trojdimenzionálnych máp prostredia. Systém obsahuje hardvérovú a softvérovú časť. Hlavnou časťou hardvéru je platforma Arduino, na ktorú sú pripojené všetky potrebné periférne zariadenia, a ktorá riadi celý systém na mapovanie prostredia. Softvérovú časť tvorí počítačová aplikácia vytvorená pomocou prostredia Processing, ktorá slúži na spracovanie potrebných údajov a na samotné vykresľovanie máp mapovaného prostredia v 3D.

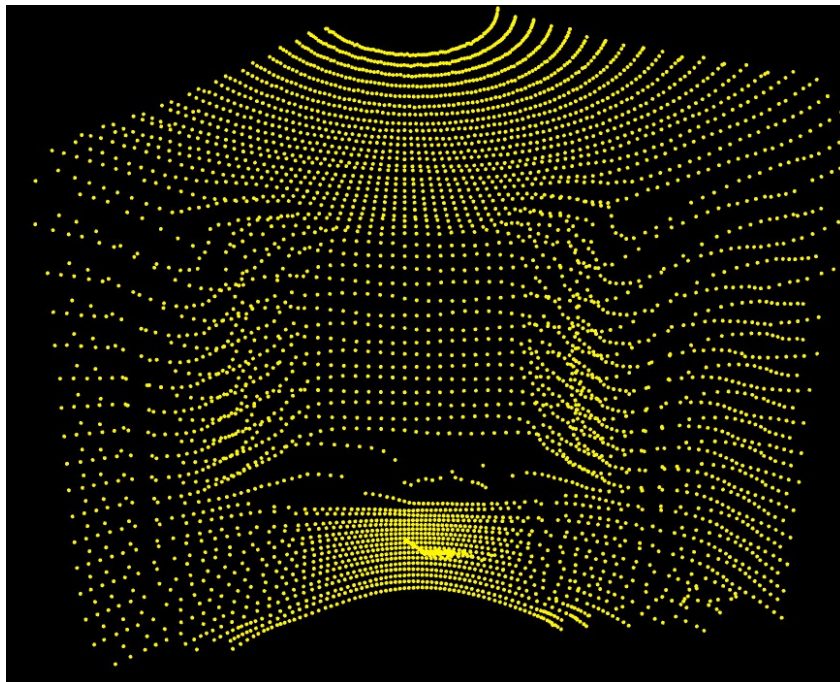


Fig. 5 Troj-dimenzionálna mapa prostredia získaná pomocou v práci popisovaného mapovacieho zariadenia.

ACKNOWLEDGMENT

This research was supported by the Slovak Research and Development Agency, project number APVV-18-0214 and by KEGA Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under Grant No. 003TUKE-4/2017 Implementation of Modern Methods and Education Forms in the Area of Security of Information and Communication Technologies towards Requirements of Labor Market.

LITERATÚRA

- [1] Victorino, A.C: La command référencée capteur: une approche robuste au problème de navigation, lo-calisation et cartographie simultanées pour un robot d'intérieur. PhD. thesis, L'Université de Nice-Sophia Antipolis, INRIA-Sophia Antipolis, September 2002.
- [2] Jurišica, L. and Murár R.: Prieskum neznámeho prostredia mobilným robotickým systémom a tvorba topologickej mapy prostredia, ATandP Journal 2/2004.
- [3] Dinnouti, L.S., Victorino, A.C. and Silveira, G. F.: Simultaneous Localization and Map Building by Mobile Robot Using Sonar Sensors, Proceedings of COBEM 2003, 17th International Congress of Mechanical Engineering, November 10-14, 2203, Sao Paulo, SP.
- [4] Durrant-Whyte, H. and Bailey, T.: Simultaneous localisation and mapping (SLAM): Part I the essential algorithms. IEEE Robotics and Automation Magazine, 2, 2006.
- [5] Castellanos, J.A., Montiel, J.M.M., Neira, J. and Tardos, J.D.: The spmap: a probabilistic framework for simultaneous localization and mapping building. IEEE Transactions on Robotics and Automation, 15(1):948-952, 1999.
- [6] Shahring M.R., Hashim F.H., Zaki W.M., Hussain A., Raj ,T.: 3D Indoor Mapping System Using 2D LiDAR Sensor for Drones. In: International Journal of Engineering & Technology [online]. vol. 7, no.4. 2018. [cit.2020-05-10]. Dostupné z: <https://www.sciencepubco.com/index.php/ijet/article/view/20797/9772>

Automatizovaný odpočet meračov

Radko VARCHOLA, Anton BALÁŽ, Branislav MADOS

Katedra počítačov a informatiky, Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach

radko.varchola@student.tuke.sk, anton.balaz@tuke.sk, branislav.mados@tuke.sk

Abstrakt—Táto práca sa zaoberá problematikou automatizovaných meračov. V úvode práca rozvádza trendy v IoT. Následne analyzuje súčasný stav a použitie automatizovaných meračov vo svete. Opisuje možné hardvérové komponenty a softvérové technológie, potrebné pre konštrukciu prototypu automatizovaného merača a vývoj príslušných aplikácií. Ďalším krokom je špecifikácia požiadaviek, návrh architektúry a algoritmov. Po návrhu nasleduje analýza bezpečnostných rizík. V riešení je detailne zdokumentovaný postup implementácie a prepojenia jednotlivých aplikácií. Vo vyhodnotení je vykonané testovanie a overenie riešenia. V závere sú zhrnuté dosiahnuté ciele a nadobudnuté poznatky.

Kľúčové slová—Automatizovaný merač, IoT, Esp32, OCR, Tesseract, Firebase

I. ÚVOD

V posledných rokoch môžeme evidovať rapídne rastúci trend inteligentných domácností, automatizácie podnikov a ďalších inštitúcií. Za pomoci všadeprítomnej internetovej siete sme schopní z bežných objektov, ktoré používame každý deň, spraviť chytré zariadenia. Spomínaný fenomén sa nazýva IoT alebo Internet vecí [1].

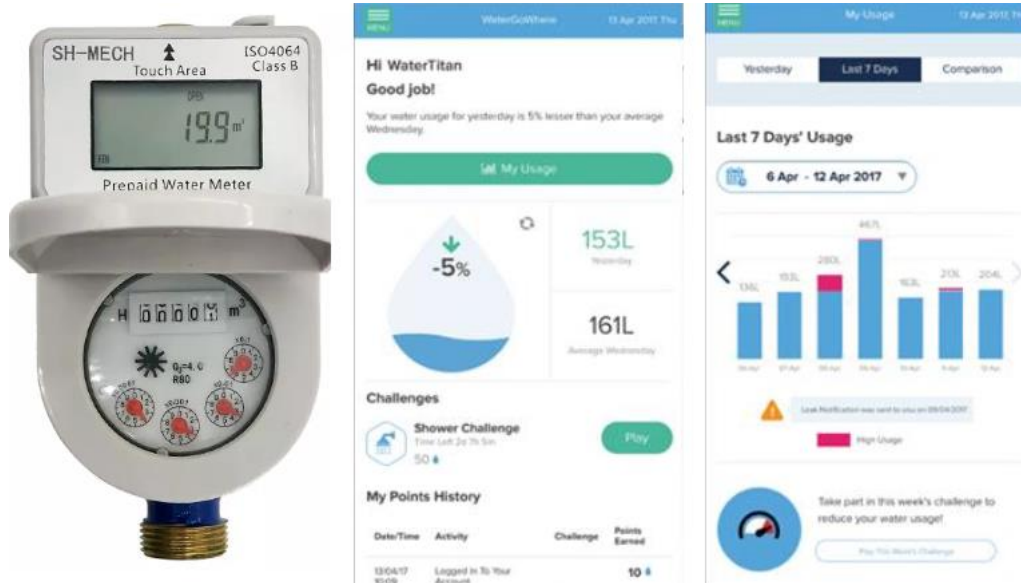
Jedná sa o množinu fyzických zariadení, domácich spotrebičov, vozidiel alebo ďalších prístrojov, ktoré môžu disponovať rôznymi vstavanými senzormi, softvérom, vstupno-výstupnými jednotkami a pohyblivými časmi [1]. Prostredníctvom internetovej siete sú schopné zhromažďovania dát, obojstrannej výmeny dát a vykonávania rozličných operácií aj bez ľudskej interakcie. Dopyt po týchto službách rastie v bežných domácnostiach, armáde, doprave, zdravotníctve, školstve, poľnohospodárstve a v ďalších odvetviach priemyslu.

Jednou z oblastí, kde trend IoT taktiež preniká je správa energií. Mechanické meracie zariadenia sa nahrádzajú automatizovanými meracími zariadeniami alebo tzv. AMR (Automated Meter Reader), ktoré sú schopné v reálnom čase zbierať dáta z merača energií nezávisle od toho, či sa jedná o plyn, elektrinu alebo vodu. Prípadne klasické mechanické merače sú obohatené o senzory alebo prístroje schopné vyhotovovať obrazový záznam v závislosti od typu merača. V mnohých krajinách používanie AMR upravuje alebo dokonca vyžaduje platná legislatíva [2].

II. AKTUÁLNE RIEŠENIA

Jedno z riešení - automatizovaný merač elektriny Sinotimer WDS688, dokáže na základe vstavaného jednofázového senzoru elektrických impulzov, zbierať namerané hodnoty elektrického prúdu, napätia, výkonu a celkovej i mesačnej spotreby elektrickej energie v reálnom čase. Plusom je jednoduchá inštalácia a ovládanie prostredníctvom aplikácie Tayu/Smart Life pre platformy iOS a Android, kde je potrebné sa zaregistrovať, prihlásiť a spárovať sa so zariadením. Súčasná cena je približne 28 EUR [3]. Ďalším z produktov je automatizovaný merač vody ShMech15, ktorý zbiera dáta pomocou internetovej domácej siete dáta z mechanického merača. Vstavaná batéria pri každodennom meraní vydrží v prevádzke až 6 rokov a jej výmena je veľmi jednoduchá. V ponuke sú taktiež verzie napájané priamo z elektrickej siete s neustálym zberom dát. Cena merača sa pohybuje od 50-100 EUR (2020). Singapúrska národná vodohospodárska agentúra prostredníctvom svojho Smart Water Meter programu zavádza tieto vodomery do tristo tisíc domácností. Hlavným cieľom programu je rozšíriť osvetu ohľadom šetrenia vody. K vodomermom používatelia obdržia mobilnú aplikáciu, ktorá deteguje úniky vody a monitoruje aktivitu spotrebičov. Aplikácia ponúka ako bonusovú funkčnosť hranie náučných hier, odoberanie aktualít a možnosť vyhrať rôzne ceny. Domácnosti, ktoré participujú na programe ušetrili v priemere 5% vody [4].

Okrem komerčných riešení možno v hobby komunitách evidovať množstvo podomácky skonštruovaných meračov. Jeden zo spôsobov je nainštalovať na mechanický merač potrebný senzor. Potom pomocou mikropočítača (častokrát sa jedná o Raspberry Pi alebo Arduino) a domácej siete zberať dáta a takto mechanický merač automatizovať. Ďalším častým spôsobom je nasadiť ku klasickému meraču mikropočítač s fotoaparátom alebo kamerou. Blog na portáli Makerbot od autora Jomlolo prezentuje jedno z takýchto riešení. Zvoleným mikropočítačom bol ESP8266 s externou kamerou ArduCAM obohatenou o nočné videnie. Zariadenie je permanentne napojené v elektrickej sieti. Plastové časti pre inštaláciu zariadenia boli vyhotovené 3D tlačiarňou. Na mikropočítači je spustený HTTP server, implementovaný v jazyku Python, ktorý na vyžiadanie zhotoví snímku merača a pošle ju na druhý web server, kde sa obrázok spracuje. Optické rozoznávanie znakov je implementované v jazyku PHP s použitím knižnice OpenCV. V používateľskom rozhraní si užívateľ môže prezrieť konkrétny obrázok. Celá implementácia je zverejnená na portáli Github [5].



Obrázok 1 Shmech a mobilná aplikácia [4]

III. ESP32-S

Mikročip ESP32 bol na trh uvedený v roku 2016 čínskou spoločnosťou Espressif Systems, ktorá je vedúcim inovátorom IoT čipov. Odvtedy má čip množstvo variácií. Pre riešenie nášho problému sme si vybrali na analýzu model ESP32-S AI Thinker, ktorý disponuje vstavaným WiFi modulom, Bluetooth 4.2 modulom s podporou. Na doske sa nachádza PCB anténa s možnosťou zakúpenia externej antény. K doske je možné pripojiť niekoľko typov kamier od výrobcu Espressif, ktoré sú schopné zhotoviť obrázky s ULTRA-HD rozlíšením. Doska taktiež zahŕňa desať GPIO pinov pre prípadne pripojenie ďalších periférií, Micro-SD slot pre pamäťovú kartu a LED flash osvetlenie. Veľkosť internej/externej RAM pamäte je 4MB. Pre ladenie dosky je potrebné dokúpiť FDTI programátor [6].

Napájanie je 5V a čo sa týka výpočtového výkonu, čip zvládne multimediálny stream vo vysokom rozlíšení. Čip je možné programovo prepínať do režimu ľahkého alebo hlbokého spánku. Výrobca tvrdí, že zariadenie zvládne vydržať v režime spánku 2-5 rokov v závislosti od kapacity bežných externých batérií s kapacitou 20000-50000 mAh, keďže spotreba v režime spánku sa pohybuje v hodnotách 10 mikroampérov. Prevádzková hodnota odberu prúdu je zas okolo 200 miliampérov. Veľmi priateľná cena, vrátane kamery je približne 5 EUR(2020) [7]. Testovaním objektívov, sme dospeli k výsledku, že najvhodnejší typ objektívu je Fish-eye lens.



Obrázok 2 Mikropočítač ESP32-S [6]

IV. TESSERACT OCR

Tesseract je softvérový rámec vytvorený spoločnosťou Hewlett Packard, zaoberajúci sa optickým rozoznávaním znakov s podporou pre operačné systémy Windows, Linux a Mac OS. Jedná sa o slobodný softvér, vydaný pod Apache 2.0 licenciou. Technológia bola vytvorená na prelome 90-tych rokov pre jazyky C a C++. V roku 2006 ho začala sponzorovať spoločnosť Google. Tesseract je považovaný za jednu z najpresnejších open-source technológií, ktoré sú momentálne k dispozícii [9].

Súčasná verzia softvéru podporuje rozoznávanie znakov vo viac ako 100 jazykoch s rozličným písmom. Taktiež je kompatibilná s mnohými formátmi obrázkov, akými sú JPG/JPEG, PNG, BMP alebo TIFF vďaka knižnici Leptonica. Ďalšou dôležitou funkcionalitou softvéru je možnosť rozlišovať rozloženie a štruktúru textu obrázku. Rozoznáva súmerne a nesúmerne oddelené znaky v texte. Softvér je možné trénovať pre nový jazyk alebo konkrétny typ obrázka. Každý obrázok je nutné spracovať, inak hrozí riziko výstupu nízkej kvality. Zaujímavým benefitom je aj Tesseract Trainer GUI, kde je možné jednoducho obohatiť tesseract o vlastný dataset [9].

V. PREDSPRACOVANIE OBRAZU S OPENCV

OpenCV je otvorená a slobodná multi-platformová knižnica zameraná na počítačové videnie, manipuláciu a spracovanie obrazu v reálnom čase. Ponúka množstvo funkcionalít, pomocou ktorých spracujeme obrázky. Aplikáciou týchto funkcionalít výrazne zvýšime presnosť pri extrakcii znakov. Jasové transformácie, ako napríklad zvýšenie kontrastu, úprava jasu, prevod obrázka na čiernobiely sú jedny z prvých krokov pri spracovaní. Ďalší typ častých úprav sú geometrické transformácie, medzi ktoré patrí rotácia, priblíženie alebo orezanie nepotrebných častí [10].

Vyhladzovanie: Obrázok sa vyhladzuje pomocou Gaussovho filtra 3x3, ktorý znižuje šum a zachováva okraje v obraze. Vyhladzovanie má dopad na hrany, čím je výraznejšie, tým menej hrán je možné detegovať [11].

Prahovanie: Metóda spracovania, ktorej výstupom sú čierne a biele obrázky. Pixely obrázka majú hodnoty od 0 do 255, kde 0 je čierna a 255 je biela. Prahovaním určíme medznú hodnotu a všetky pixely s nižšou hodnotou nastavíme na 0 a všetky pixely s vyššou hodnotou nastavíme na 255. Cieľom je sa dopracovať k optimálnej hodnote. Inverzné prahovanie obsah farieb navzájom vymení. Adaptívne prahovanie má na základe rozloženia v snímke viac lokálnych prahov [11].

Detekcia hrán: Hrana je hranica medzi dvoma oblasťami s pomerne rozdielnymi hodnotami jasu. Cieľom detekcie hrán je nájsť miesta v snímke, kde sa jas razantne mení. Metódy detekcie hrán sú založené na veľkosti prvej derivácie jasu a druhej derivácie jasu. Jeden z najpopulárnejších operátorov používaný pri detekcii hrán sa nazýva Sobel. Charakteristický je

tým, že pri detekcii hrán pomocou derivácie, ktorá vytvára šum v obraze, vykonáva jemné vyhladzovanie, aby ho zredukoval [12].

Detekcia obdĺžnikov: OpenCV dokáže na základe geometrických tvarov detegovať objekty v obraze. V príslušných metódach stačí špecifikovať tvar objektu, rozsah rozmerov a následne vykonať selekciu. S výslednými oblasťami je vo výsledku možné robiť rôzne operácie, ako spájať, vystrihnúť a iné [13].



Obrázok 3 Modifikácie obrazu

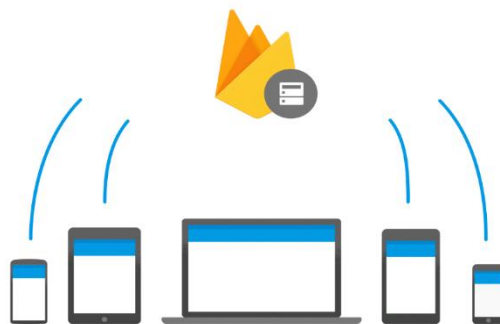
VI. SLUŽBY FIREBASE

V našom produkte bude potrebné zabezpečiť spoľahlivý, stabilný a bezpečný backend. Jednou z možností, ktorá spraví značnú časť práce za nás, je vývojová platforma Firebase. Firebase ponúka množstvo služieb a funkcionalít, ktoré môžeme veľmi jednoducho integrovať do našich aplikácií. Výhodou je jednoduchá integrácia s ďalšími biznis nástrojmi a technológiami od spoločnosti Google a moderná interaktívna dokumentácia. Platforma ponúka bezplatný štart a manažment vytvorených aplikácií dosiahneme pomocou Firebase konzoly [8].

Firestore autentifikácia umožňuje identifikáciu užívateľa a zabezpečenie obsahu aplikácie. Benefitom pre vývojárov je jednoduchá implementácia a množstvo hotových funkcionalít, ktoré je možné na mieru prispôsobiť. Služba podporuje desiatky možností, prostredníctvom ktorých sa môžu užívatelia zaregistrovať a splňa OAuth 2.0 štandard, či už je to email, telefón alebo rôzne sociálne siete. Emailová a dvojfaktorová autentifikácia sú tiež samozrejmosťou. Každý používateľ má vygenerované unikátne UserID. Služba je ľahko integrovateľná s mobilnými platformami Android, iOS a webovými aplikáciami [8].

Firestore Realtime je cloud-hosted databáza, ktorá archivuje svoje dáta v JSON formáte. Klientske aplikácie zdieľajú databázu a automaticky behom niekoľkých milisekúnd obdržia aktualizáciu s najnovšími dátami. Dáta sa ukladajú, aj keď je zariadenie offline a po opätovnom pripojení sa synchronizujú s verziou na serveri. Ďalším benefitom je rýchly návrh schémy vďaka JSON štruktúre. Bezpečnosť a formát dát môžeme dosiahnuť pomocou pravidiel, ktoré môžeme nastaviť v konzole. V pravidlách môžeme špecifikovať, ktoré skupiny používateľov majú prístup k jednotlivým častiam dát. Nevýhodou sú obmedzenia pri zložitejšom filtrovaní dát a komplexných dopytov [8].

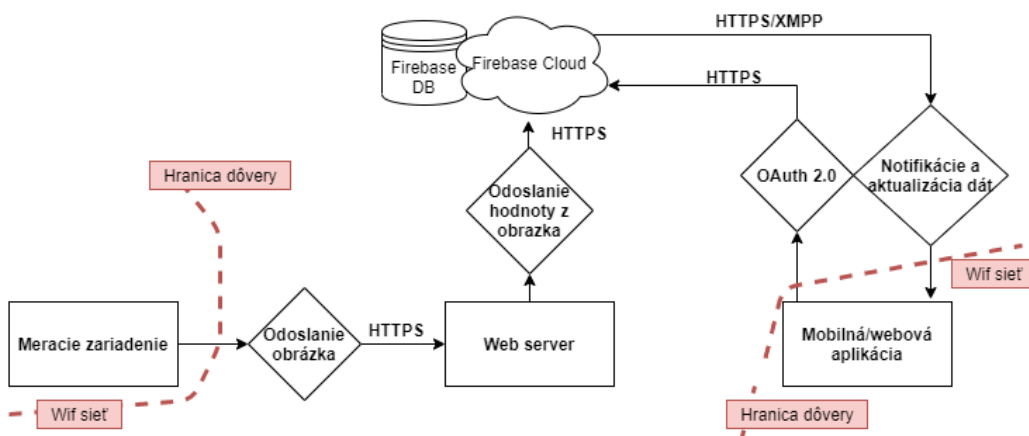
Odosielanie notifikácií o meraniach z našej serverovej aplikácie na mobilné zariadenia cez Firebase Cloud dosiahneme prostredníctvom služby Firebase Cloud Messaging. Služba využíva HTTP a XMPP protokoly a je schopná odoslať správu v JSON formáte až do veľkosti 4 kilobajtov. Dokumentácia uvádza, že priemerný čas doručenia správy je 250 milisekúnd a denne ich môžeme odoslať ľubovoľný počet. Každému mobilnému zariadeniu služba priradí unikátny registračný token. Pomocou týchto tokenov sme schopní zariadenia rozdeliť do skupín a efektívne zacieliť príslušné notifikácie konkrétnym používateľom [8].



Obrázok 4 Služby Firebase [8]

VII. NÁVRH RIEŠENIA

Na základe požiadaviek a analýzy sme vyhotovili návrh štruktúry, ktorým docielime požadované riešenie automatizovaného zberača dát. Dodržanie bezpečnostných kritérií a minimalizáciu bezpečnostných rizík a hrozieb riešime pomocou šifrovanej komunikácie, certifikátov a autentifikácie spĺňajúcej štandardy. Automatizovaný merač bude vyhotovovať snímku a odosielať ju na webový server prostredníctvom HTTPS protokolu. Server obrázok uloží, spracuje a hodnotu zapíše do databázy na Firebase cloude. Taktiež upovedomí cloud, aby odoslal notifikácie určeným cieľným zariadeniam. Klientske aplikácie po OAuth 2.0 autentifikácii odfria z Firebase servera správy a aktualizujú si dáta, ktoré zobrazujú. Prípadná komunikácia web servera s aplikáciami je taktiež ošetrená pomocou HTTPS.

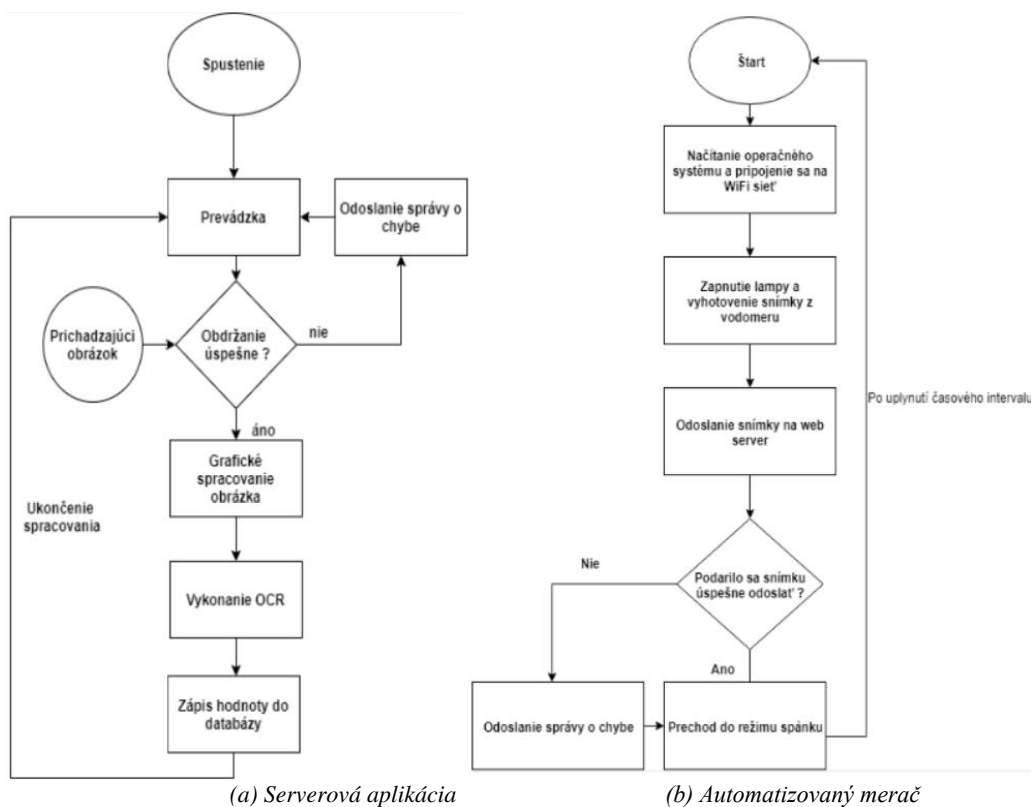


Obrázok 5 Architektúra riešenia

VIII. ALGORITMUS AUTOMATIZOVANÉHO MERAČA A SERVEROVEJ APLIKÁCIE

Algoritmus automatizovaného merača je nasledovný. Zariadenie by sa nachádzalo pri konkrétnom merači energií a keďže berieme do úvahy, že sa jedná o prostredie v tme, mohlo by fungovať nasledovným algoritmom. Po zapnutí by sa zariadenie pripojilo na domácu bezdrôtovú sieť, zaplo osvetlenie, ktorým by disponovalo a následne fotoaparátom spravilo snímku, ktorú by pomocou komunikačného protokolu odoslalo na web server pre ďalšie spracovanie. Následne by prešlo do režimu spánku a po určitom časovom intervale celý proces zopakovalo.

Aplikácia nasadená na webovom serveri by za pomoci mikrorámca Spark vykonala zostávajúce potrebné procesy, uložila by snímku do svojho priečinka. Ďalším krokom by bolo spracovanie a modifikovanie grafických vlastností obrázka pre lepšie rozpoznávanie znakov z meracieho zariadenia. Následne by sa vykonalo rozpoznávanie znakov z obrázka. Extrahovaná hodnota by sa zapísala do databázy, kde by bola prístupná klientskym aplikáciami pre grafické zobrazenie. Prípadná ďalšia funkcionálna by mohla zahŕňať odoslanie notifikácie klientskym aplikáciám o práve vykonanom meraní.



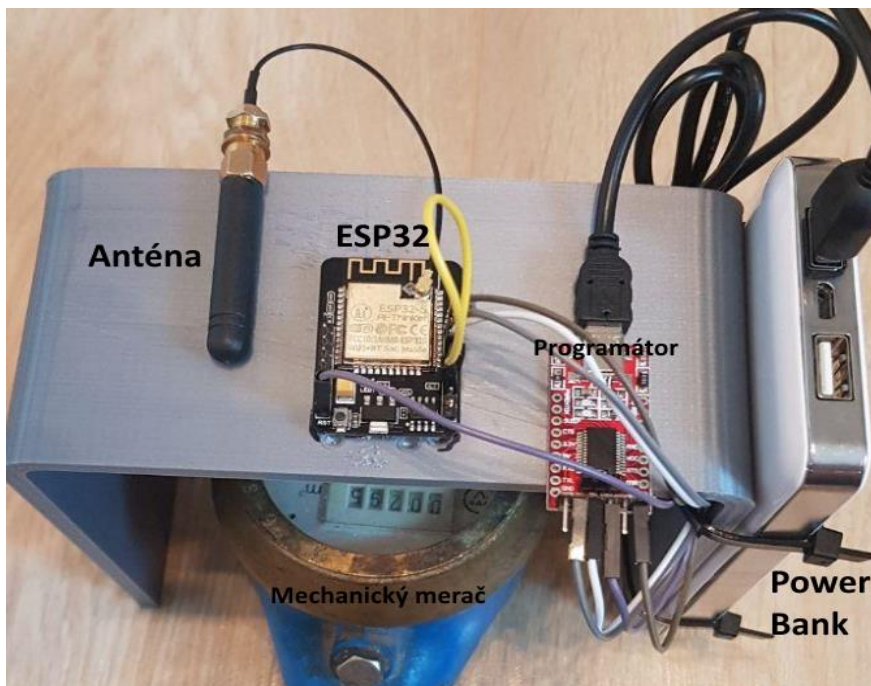
Obrázok 6 Diagram algoritmu serverovej aplikácie a automatizovaného merača.

IX. VYHODNOTENIE

Na základe navrhnutého riešenia bol skonštruovaný prototyp automatizovaného merača. Prvým krokom bolo vyhotoviť podstavec pre automatizovaný merač, ktorý sme vyrobili s použitím 3D tlačiarne. Na stojane je prichytený mikropočítač ESP32, ktorý má vyhotovené otvory pre objektív kamery a LED osvetlenia. Vedľa mikropočítača sa nachádza lepidlom zafixovaný FDTI programátor, ktorý prepája ESP32 so záložným zdrojom. Konkrétne sa jedná o mobilnú power banku. Posledným komponentom pripojeným na prototyp je externá anténa, ktorá má za úlohu zlepšiť pripojenie na domácu sieť. Približná súhrnná cena je 17 EUR (2020).

Automatizovaný merač sa po stlačení tlačidla spustí, pripojí sa na domácu internetovú sieť a vyhotoví snímku so zapnutým osvetlením. Redukciu odlesku dosiahneme prekrytím osvetlenia tenkým matným materiálom. Snímka sa odošle na lokálny server. Občas môžeme pozorovať zlyhanie inicializácie kamery, alebo neúspešné pripojenie na domácu sieť. Tento problém je riešený opakovanými reštartami. Zariadenie po úspešnom vykonaní odoslania prejde do režimu hlbokého spánku a po skončení nastaveného intervalu sa prebudí a celý proces zopakuje. Čo sa týka výdrže batérie, elektrickým multimetrom sme zmerali odber prúdu zariadenia. V hlbokom spánku sa hodnota prúdu pohybovala okolo $10\mu\text{A}$. V aktívnom stave, ktorý trvá 30 sekúnd sa pohybovala v rozmedzí $150\text{--}250\text{mA}$. Podľa výpočtu na portáli GeekTips[14] by batéria s kapacitou 20000 mAh mala výdrž 338 dní pri meraniach vykonávaných každú hodinu.

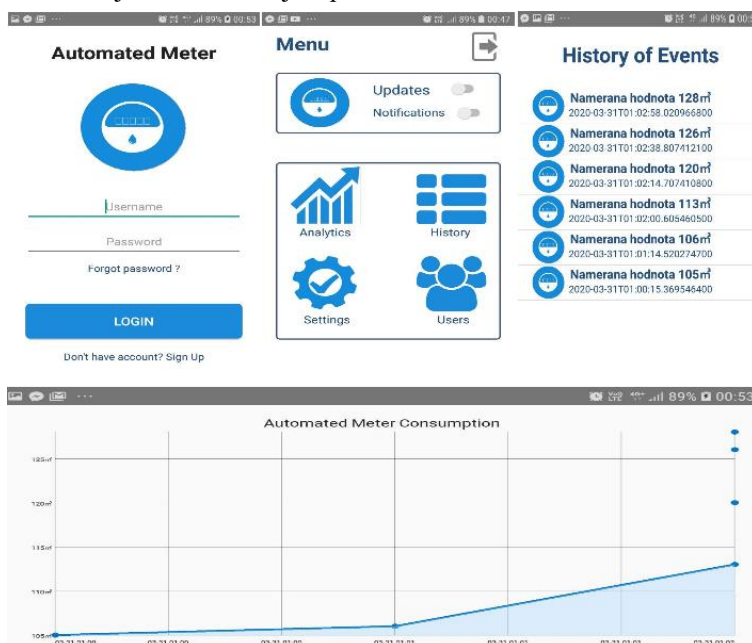
Serverová aplikácia je nasadená v lokálnom Docker kontajneri. Url adresa má HTTPS prefix pred príslušnou URL. Doručený obrázok je uložený, spracuje sa a extrahujú sa z neho znaky. Nameraná hodnota sa zapíše do Firebase NoSQL databázy a odošlú sa notifikácie príslušným zariadeniam. Pri zmene modelu merača, je častokrát nutné pozmeniť implementáciu predspracovania obrázka a rozoznávania znakov, prípadne dotrénovať OCR softvér. Dôvodom je rozdielna štruktúra objektov vo fotografii merača. Iný typ znakov, ako sú napríklad digitálne číslice. Taktiež mnohé ďalšie faktory, ako farba, formát a redundantné znaky v merači.



Obrázok 7 Prototyp automatizovaného merača

X. MOBILNÁ APLIKÁCIA

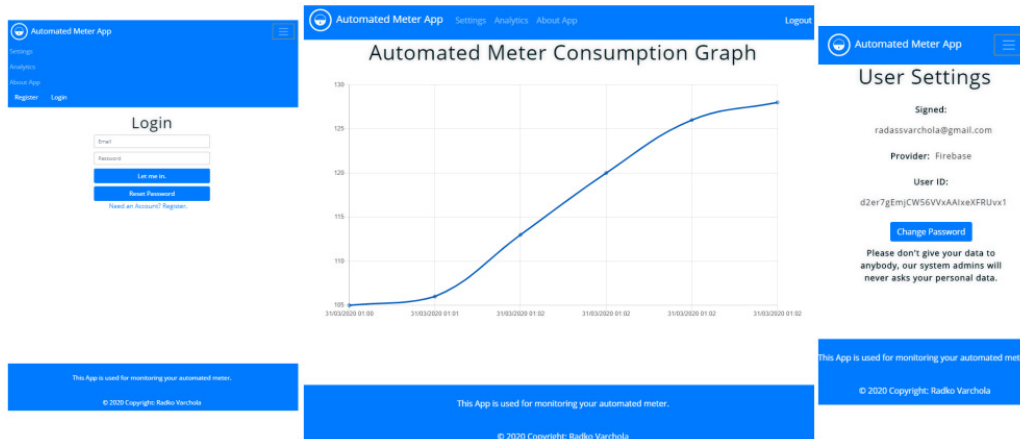
Mobilnú aplikáciu nainštalujeme pomocou APK súboru. Aplikácia je kompatibilná iba s Android 6 a vyšším. Najprv je potrebné sa do aplikácie registrovať. Po registrácii sa užívateľ ocitne v Login aktivite kde sa po verifikovaní emailu môže prihlásiť. V login aktivite sú odkazy na aktivitu pre registráciu a reset hesla. Celý postup registrácie a prihlásenia využíva službu Firebase Autentifikácia. Spolu s doručenou notifikáciou, ktorá príde prostredníctvom služby Firebase Cloud Messaging, sa spraví zápis do Firebase databázy s príslušným časom, ktorý si užívateľ môže pozrieť v histórii, do ktorej sa dostane po kliknutí na ikonu History v hlavnom menu. Notifikácia sa zobrazí aj keď je aplikácia vypnutá, ale užívateľ musí byť prihlásený. Veľkou výhodou je, že notifikácie prídu do jednej sekundy a obsah databázy sa aktualizuje v aplikácii hneď akonáhle sa pripojí k sieti. To isté platí aj pre graf zobrazovaných dát. Ďalšou výhodou grafu, je možnosť listovania a približovanie. V nastaveniach a v hlavnom menu vie užívateľ pomocou spínačov vypnúť alebo zapnúť notifikácie. V nastaveniach je možné aplikáciu aktualizovať, odhlásiť sa a zistiť potrebné informácie o aplikácii. V aktivite Užívateľia sa nachádzajú základne údaje o prihlásenom užívateľovi.



Obrázok 8 Mobilná aplikácia

XI. WEBOVÁ APLIKÁCIA

Webová aplikácia vyvinutá za pomoci rámca Angular, taktiež poskytuje možnosť, ako zobrazíť namerané dáta. Rovnako ako mobilná aplikácia, implementuje služby platformy firebase. Prvým krokom je registrácia. Po verifikovaní emailu, sa užívateľ vie prihlásiť. V prípade neúspechu si vie užívateľ obnoviť heslo. Po prihlásení môže užívateľ vidieť graf nameraných hodnôt. V ďalších sekciách si môže užívateľ aplikáciu prispôsobiť a prezrieť si aktuálny a informácie o aplikácii. Responzivnosť aplikácie bola dosiahnutá pomocou aplikačného rámca Bootstrap a korektné sa zobrazí na osobnom počítači, tablete či chytrým telefóne.



Obrázok 9 Webová aplikácia

XII. ZÁVER

Hlavným prínosom práce je návrh vyhotovenia automatizovaného merača napájaného batériami, ktorý zbiera dáta z mechanického meracieho zariadenia. Namerané hodnoty sú bezpečne zobrazené užívateľom pomocou mobilnej alebo webovej aplikácie. V úvode sme predstavili fenomén IoT technológií, ich použitie v domácnosti a rozličných odvetviach priemyslu. Následne sme sa oboznámili s problematikou automatizovaných meračov a preskúmali ich súčasný stav na dnešnom trhu. Ďalej boli prezentované komunikačné protokoly a vývojové platformy, ktoré zabezpečujú komunikáciu medzi meračom a klientskými aplikáciami. Analyzovali sme si Spark a ďalšie webové mikrorámce ktoré by mohli tvoriť serverovú aplikáciu. Porovnali sme si Tesseract OCR s inými alternatívami a opísali modifikáciu obrazu s použitím OpenCV. Následne sme navrhli štruktúru produktu a vytvorili algoritmy automatizovaného merača a serverovej aplikácie. Pre elimináciu zraniteľností sme vykonali bezpečnostnú analýzu za pomoci Microsoft Thread Modeling. Podrobne sme opísali implementáciu a konštrukciu automatizovaného merača, mobilnej a webovej aplikácie. Vo vyhodnotení sme zhrnuli dosiahnuté výsledky a poznatky.

POĎAKOVANIE

This research was supported by KEGA Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under Grant No. 003TUKE-4/2017 Implementation of Modern Methods and Education Forms in the Area of Security of Information and Communication Technologies towards Requirements of Labor Market.

LITERATÚRA

- [1] Internet of things (IoT).url:<https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT> (cit.27.03.2020).
- [2] Smartmeter. url: https://en.wikipedia.org/wiki/Smart_meter (cit. 27.03.2020).
- [3] WDS688 90-250VACWIFI Smart Switch EnergyMeter. url: <https://www.aliexpress.com/item/4000171379795.html?spm=2114.12057483.0.0.3ec271a6ywFeCC> (cit.27.03.2020).
- [4] 300,000 smartwatermeters to be installed in homes, firms. url: <https://www.straitstimes.com/singapore/300000-smart-water-meters-to-be-installed-in-homes-firms>.
- [5] Wasserzähler erfassen.url: <https://www.thingiverse.com/thing:3238162> (cit.27.11.2018).
- [6] ESP32ADifferent IoTPower andPerformance.url: <https://www.espressif.com/en/products/hardware/esp32/overview> (cit.07.09.2016).
- [7] Insight IntoESP32 SleepModes andTheirPowerConsumption. url: <https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/> (cit. 08.01.2020).
- [8] The Good and the Bad of Firebase Backend Services. url: <https://www.alt-exsoft.com/blog/firebase-review-pros-cons-alternatives/> (cit. 06.11.2019).

- [9] Tesseract: anOpen-SourceOptical Character Recognition Engine. url: <https://www.linuxjournal.com/article/9676> (cit.01.07.2007).
- [10] Introduction to OpenCV.js andTutorials.url: https://docs.opencv.org/3.4.0/df/d0a/tutorial_js_intro.html (cit.27.03.2020).
- [11] Wilhelm Burger.Digital Image Processing:AnAlgorithmic IntroductionUsing Java.2.Springer-VerlagLondon,mar.2016. isbn:9781447166832.
- [12] HaffnerOto. “Detekcia hrán v obraze”. In:MATLAB/Comsol 4.8 (dec. 4). <http://www.posterus.sk/?p=11234>, s.1–2.
- [13] OpenCV shape detection byAdrian Rosebrock. url: https://docs.opencv.org/3.4.0/df/d0a/tutorial_js_intro.html (cit.08.02.2016).
- [14] BATTERY LIFE CALCULATOR FOR PROJECTWITH SLEEPMODE. url: <https://www.geekstips.com/battery-life-calculator-sleep-mode/> (cit.08.02.2020).

Forensic analysis of mobile instant messenger applications

Tomáš MARCZI, Anton BALÁŽ and Branislav MADOŠ

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

tomas.marczi@student.tuke.sk, anton.balaz@tuke.sk, branislav.mados@tuke.sk

Abstract — This paper intends to demonstrate a real-life use case of forensic analysis of WhatsApp instant messaging application on Android smartphones. To clarify the problematics, firstly, we comprehensively describe the different phases of forensic analysis. Then we describe the architecture we established to accomplish the mobile data acquisition, storage, and analysis under forensically sound conditions. In the end, we perform forensic analysis on the obtained instant messaging application's artefacts, and we semantically analyse the available textual artefacts with the hope of providing pieces of information that have evidentiary value in front of the court of law.

Keywords — forensic analysis, forensic science, mobile forensics, semantic analysis, Android, WhatsApp

I. INTRODUCTION

The online communication has become one of the most common forms of communication between people. The reason behind this probably is, that it takes significantly less effort to reach someone through the internet. The claim for instant messaging (IM) applications is enormous, and it is still rapidly increasing every day.

However, with the growth of the IM application users and the possibilities the internet provides, the number of cyber-crimes is growing too. Until a few decades ago, network and computer security were a defence utility for military establishments. Nowadays, network and computer security have become an essential asset of every home computer.

IM application users often share confidential and personal information, photos, and videos amongst themselves, and in the world of the internet, 100% privacy does not exist. Despite the security measures provided by the owners of these applications, there is still a chance for cyber-criminals to reach these private data of the users. Taking advantage of the vulnerability of social networking and instant messaging applications is a frequently used way of hackers.

When such an incident happens, forensic analysis is needed to be performed. However, this is not the only case a forensic analysis is performed. Every incident, every criminal act, every investigation where digital devices are involved, that device needs to be forensically analysed with the hope of providing relevant information or even pieces of evidence related to the given case.

II. GOALS

The primary goal of this work is to construct an architecture for Android mobile data gathering, storage, and analysis. To clarify the concept behind the forensic analysis, we present the phases of it. In this work, we also simulate a real-life use case of forensic analysis. In the analysis phase, our primary target is the WhatsApp IM application and its textual artefacts on which we perform semantic analysis.

III. RELATED WORKS

The digital forensic analysis is a relatively new field of forensic science. Recent scientific research has focused on the acquirable artefacts on iPhones (Zdziarski, 2010), which described the different artefacts found on an iOS device. Kubasiak in his work in 2009, highlighted that the jailbreaking of the phone (getting root access) is fundamental in mobile forensics, although

it does not cope perfectly with the rules of forensic science because the target device's memory gets altered. Al-Mutawa in 2011 demonstrated in his work the available artefacts from the Facebook application on computers. Lessard and Kessler have also done some essential researches in the field of digital forensics. They emphasized in their work (2011) the importance of mobile forensics, researched and described the artefacts of typical social media applications acquirable from mobile smartphones.

IV. DIGITAL FORENSIC ANALYSIS

A. *Basics*

Forensic analysis is a comprehensive examination for identifying the pieces of evidence, consequences, reasons and culprits of a security incident or violation of rules of the state laws or organizations. Digital forensic analysis is the sub-type of forensic analysis, which emphasizes on digital devices such as smartphones, computers, tablets.

The specialists who perform the digital forensic analysis are so-called digital forensic investigators since digital forensic analysis is part of every criminal investigation, where digital devices are involved. These forensic experts are responsible for collecting the pieces of digital evidence and for following every security and data protection procedures when they are handling sensitive financial or private information. The competencies these examiners need to have are comprehensive computer science knowledge, analytical skills, and most of all, attention to details. [1]

B. *Rules and phases of digital forensic analysis*

There are strict rules that need to be followed when performing forensic analysis. A fundamental rule of the forensic analysis is to sustain the integrity of the original data and to protect it from any contagion that would interfere with its approval in court, so this is something that the examiner needs to take care of. To fulfil this rule, the forensic analysis needs to be completed between forensically acceptable conditions. In forensically acceptable conditions, when a forensic analysis is being performed, if any changes occur during an examination, the nature, the extent, and the reason for all changes need to be accurately documented. [2]

The first phase of digital forensic analysis is all about identifying the purpose of the investigation and identifying the required resources a forensic investigator needs for the execution of analysis.

The second phase is the phase of preservation. Preservation of the data is a must in forensic analysis in any phases because the originality of the data needs to remain untouched. Although there is a phase dedicated to the preservation because there is a point in forensic analysis, where the preservation gains considerable importance. In this phase, the forensic investigator needs to make sure to eliminate any activities, any potentially harmful elements that may harm or even destroy them. The suspect digital devices need to be protected. They must be precisely the same how they were found, without any alteration, so that it can be later analysed.

In the third phase, the potential pieces of digital evidence are being collected for analysis. When the forensic examiner recognizes potential evidence, he collects it.

The fourth phase stands for the examination and analysis of the acquired artefacts. This phase includes a comprehensive systematic search of evidence related to the incident being investigated. The examination process needs to be well organised to prevent crucial data from being lost, which may be required and critical to a court case. Several software tools exist to help and fasten the work of forensic investigators in this phase. These tools can be used in this phase to help with file extraction, file filtering, text indexing, e-mail, and NAT recovery and with many other essential processes.

In the final phase of the forensic analysis, the examiner needs to create a report which includes all the results of the analysis and a conclusion. This report must be written according to a specific template. [3]

V. FORENSIC ANALYSIS OF WHATSAPP IM APPLICATION

WhatsApp allows users to chat and make calls. There are several other features provided by WhatsApp that users can use in chat sessions, including sending multimedia files, voice records, contacts, and geographical location sharing is also available for the users. Initiating conference

and video conference calls is also possible. Every user has its profile, which is a collection of data that involves the user's WhatsApp name, status line, and avatar picture. All the profile data is saved on WhatsApp's central system. When a user adds someone to its contact list, all the profile information is downloaded from this central repository, which also presents other services, such as user authentication or registration.

The primary goals of this chapter are the presentation of acquisition, extraction, and analysis of all the artefacts produced by WhatsApp IM application on Android smartphones and the understanding how these artefacts can be correlated mutually to conclude different types of information.

The goal is to approve the acquisition of realistic data corresponding to that obtained in real-world investigations. For this purpose, the first step of the experiment included the creation of several fictional WhatsApp accounts. The next step was the conduction of everyday WhatsApp activities, i.e. generation of chat conversations between the fictional suspect's phone (our phone) and other parties and usage of all the possible features provided by WhatsApp.

For the experiment, the fictional suspect's smartphone we used during the analysis is the Samsung Galaxy Ace 4 model running on Android 4.4.4 (Android KitKat). The fictional forensic workstation we use is a device running on Windows 10 operating system.

The first thing we do is to obtain root access to the phone. After it is done, we have unlimited access to the phone's whole internal memory unit. Next, we gain the logical image of the phone's internal memory using. For this purpose, we use BusyBox, ncat and Android Debug Bridge.

A. *BusyBox*

Android is running on a Linux kernel. However, by default, it does not provide such system commands as copying or removing files and folders in the filesystem. That is because of usability reasons, and everyday users do not need these features. The BusyBox application is basically a set of the basic Linux commands that allow the users to execute Linux commands for filesystem manipulation.

B. *Ncat*

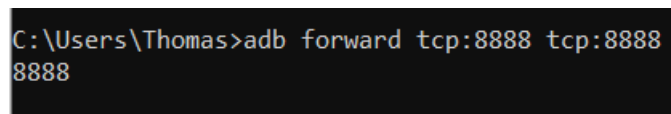
Ncat needs to be present on our forensic workstation. It is a powerful program that has built-in network utilities. With the help of these network utilities, it can write and read data across different networks from a command line. Ncat has numerous useful other features. It utilises both TCP and UDP for communication and is created to be a secure back-end engine to maintain network connectivity to other applications and users directly. It can work with both IPv4 and IPV6. Ncat can redirect TCP and UDP ports to other sites or ports. It can also provide proxy connections, HTTP proxies, and even SSL support.

C. *Android Debug Bridge*

Android Debug Bridge is a handy command-line machine that allows the user to interact with an Android phone. By utilising the adb command, we can initiate a wide range of device operations, such as installing and debugging applications. Android Debug Bridge also grants entrance to a Unix shell that we can use to execute a wide range of commands on an Android mobile phone.

D. *Logical image acquisition and extraction*

Firstly, we enable the Android Debug Bridge's port forwarding feature. With the help of this feature, we can command the adb to forward requests from a specific host port to another specific port on a device or emulator.



```
C:\Users\Thomas>adb forward tcp:8888 tcp:8888
8888
```

Fig. 1. – Android Debug Bridge's port-forwarding

In our case, we want to forward traffic from our phone to one of our forensic workstation's port. This is achieved with the help of the command on Fig. 1. By that command, we are telling adb to forward every traffic from the phone's head unit server to the port 8888 on the workstation.

After this connection is established between the phone and the workstation, we utilize our next mobile forensics architecture component, the dd. The dd is a Linux command-line utility to create disk images among its other features. It is a regularly used tool in mobile forensics, considering it can be used to produce a raw image of memory partitions, folders, or files. The product of the dd tool is a logical image with a .dd extension. Next, we set up a listening connection on the phone's head unit server (port 8888 of the phone).

```
130|root@heatqlte:/ # dd if=/dev/block/mmcblk0 | busybox nc -l -p 8888
```

Fig. 2. – Utilisation of the dd command

The command on Fig. 2. means that the input file (if) is the mmcblk0, which is the phone's primary partition. This command means that the Ncat is going to listen on port 8888 of the phone, and whenever a connection comes in on that port (obviously from the workstation), the phone will send all of the data located in the dd input file through this port.

The last step is to initiate the data transfer on the forensic workstation. For this purpose, we utilise the command on Fig. 3., which makes a connection on port 8888 to the localhost, and because we have the adb port forwarding set up, it will make a connection basically to the phone. The phone will send the workstation back all the data from the partition. To handle that data, we need to pipe it into a .dd file, as the Fig. 3. shows.

```
C:\Users\Thomas\Desktop\platform-tools>ncat.exe 127.0.0.1 8888 > android2.dd  
C:\Users\Thomas\Desktop\platform-tools>_
```

Fig. 3. – Data transfer initiation

At this point, the logical image of the phone is acquired and is ready to be analysed. Part of our architecture is the Autopsy software tool, which is a powerful tool for analysing disk images. Autopsy extracts the image. Every partition goes to a different folder. The user-generated data is located at the userdata partition, e.g. in the userdata folder.

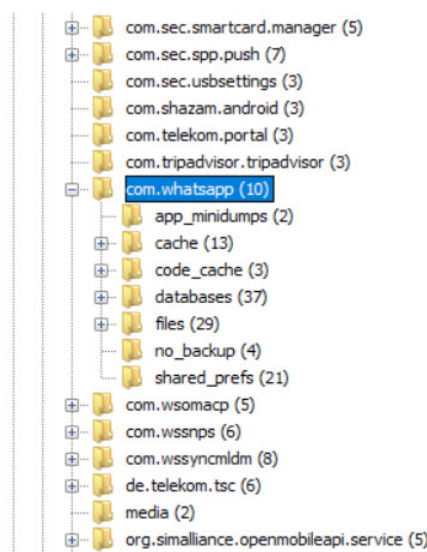


Fig. 4. – WhatsApp data location

The most important artefacts of the WhatsApp IM application are the contacts database, chat database, contact avatars, log files, user settings and preferences. In the contacts database (wa.db), the wa_contacts is the most crucial table. It contains information about the contacts of

the owner of the WhatsApp account such as phone numbers, contact addition timestamp, contact status and the contact's WhatsApp account type (can be business, standard etc.).

The avatars of contacts are all downloaded and available in the files folder. The log files are considered as significantly useful sources of information. Every activity done by the user is recorded in these files such as contact addition, contact deletion, messages sent, messages received, message deletion. For forensic investigators, these are very helpful and valuable artefacts.

The user settings and preferences contain information about the timestamps of the last and first activity of the user and other less relevant information.

The most valuable source of information is the chat database. The messages table contains all the undeleted messages that were sent or received by the user. It also contains the metadata of every single message including timestamps, the data type it carried, the list of recipients involved in the conversation, and even whether the recipients received and saw the particular message.

The primary task when analysing the messages table is to rebuild the conversation in its actual chronological order to be able to understand the transmitted messages. When reconstructing the discourse, all the columns described above need to be considered to get the most accurate picture of the conversation. Although to be able to rebuild the conversation, the contents of the messages table need to be extracted and processed. For this purpose, we use the built-in feature of Autopsy, which allows exporting tables in CSV format. After the table has been exported to our forensic workstation, it needs to be stored securely and in the correct format. To store it, we created a MySQL database on our workstation for this case, and we saved these exported artefacts in there.

Next, we transform the contents of the tables which are not directly readable to readable format because as an example, the timestamps are all stored in Unix epoch time format, which is hard to understand at first look.

Storing all these sensitive data directly on the workstation device, which is equipped with a lot of security measures, is a secure way.

At this point, we can construct the chat conversations of the suspect in chronological order and start the semantic analysis.

E. Semantic analysis

To determine the mindset of the authors of the chat messages and to fully understand the actual meaning of these transmitted messages, it is inevitable to perform a comprehensive semantic (or textual) analysis within the forensic analysis.

In this work, we recreated simulation of a lifelike case that is contemporary nowadays. In our simulation, we created several imaginary subjects, e.g. we created a WhatsApp account for each of our subjects.

The main subject of the forensic investigation is Jack. Jack has gone missing and was last seen on the 15th of March. The authorities have acquired a search warrant for Jack's house, and they seized his mobile phone that was found at his place. After the phone was given to the digital forensic examiners (the examiners are us in this paper), they started the analysis of the phone immediately. They followed the procedure described in this paper to acquire the phone's logical image forensically, to extract the artefacts and to separate the relevant artefacts from the less relevant ones. They found out that the only user-installed application on the phone was the WhatsApp IM application. They examined the contact databases and the settings of Jack's WhatsApp account. Eventually, they started to analyse the chat databases with the hope of finding information about Jack's disappearance in the chat logs. After they accomplished the reconstruction of the chat conversations, the semantic analysis of the messages may have taken place.

According to the source no. 4, the semantic analysis of digitally found textual artefacts has three steps. In the first step, the forensic experts create the criminological ontology and the analysis corpus. In this phase, the examiners must separate all the textual data gathered from the digital device. Additionally, the raw text extraction is being performed from document files if it is needed. Eventually, the separated text data with all its metadata is stored (usually in a database). In the second step, the processing of the textual data takes place. Language recognition is also part of this phase. The texts are logically structured, and, in our case, the chat conversations are reconstructed into chronological order. The determination of sentences or text fragments that are defined as secondary contexts also takes place in this phase. Secondary contexts are sentences that are not explicitly relevant from the investigation's perspective.

The actual extraction process forms the last action inside the primary process. In this phase,

the substantive event sentences that are fitting to instantiate at least one section of the ontology are identified.

After the semantic analysis has been undertaken, we need to make a conclusion about the case and choose a criminological ontology that corresponds for this case, based on the findings and results of the text analysis. The discussions we analysed clearly corroborates that there is a very high probability that the suspect's disappearance can be linked to his suicidal intents.

We pass our results of forensic analysis and semantic analysis (as digital forensics report) of Jack's mobile phone to the authorities that are investigating Jack's disappearance, and hopefully, our findings bring the investigation forward.

VI. CONCLUSION

In this work, we have presented an experiment of a forensic analysis of the WhatsApp IM application artefacts on phones running on Android OS. We proved that the data acquired from the WhatsApp IM application could have relevant evidentiary value in the court of justice. Additionally, we recreated a possible real-life situation of a suspect who may have potential suicide intents. We forensically analysed the artefacts left by the suspect on the phone.

We performed a semantic analysis on the reconstructed chat conversations. Eventually, based on our findings and results of the semantic analysis, we concluded, that the suspect has the potential to have suicide intents and his disappearance can be linked to these intents.

The results of this research may hopefully help the forensic community's perception of the kinds of artefacts that are expected to persist following the use of WhatsApp IM application and services on Android smartphone devices.

ACKNOWLEDGMENT

This research was supported by KEGA Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under Grant No. 003TUKE-4/2017 Implementation of Modern Methods and Education Forms in the Area of Security of Information and Communication Technologies towards Requirements of Labor Market.

REFERENCES

- [1] S. McCombie and M. Warren, Computer forensic: An issue of definition, Proceedings of the First Australian Computer, Network and Information Forensics Conference, 2003
- [2] Curran, K., Robinson, A., Peacocke, S., Cassidy, S. 2010. Mobile Phone Forensic Analysis, International Journal of Digital Crime and Forensics, Vol. 2, No. 2, pp., April-May 2010, ISSN: 1941-6210, IGI Pub
- [3] Bandar Almaslukh, Forensic Analysis using Text Clustering in the Age of Large Volume Data: A Review. International Journal of Advanced Computer Science and Applications, Vol. 10, No. 6, 2019
- [4] M. Spranger, D. Labudde. Semantic tools for forensics: Approaches in forensic text analysis. University of Applied Sciences Mittweida. Mittweida, Germany. ISBN: 978-1-61208-311-7. 2013. pp. 20-22

Aspekty Kybernetickej bezpečnosti v procese vzdelávania počas krízy COVID-19

¹Jana Handriková

¹ Department of Technical Science and Informatics, Faculty of Security Engineering,
University of Zilina, Slovak Republic

¹jana.handrikova@fbi.uniza.sk,

Abstract — Currently, the proportion of users who work with computer technology every day is increasing. However, not everyone is aware that computer systems contain very sensitive data that can be misused. Everyone, even the average user, participates in the protection of data stored in the information system every time they access the computer system. Often without realizing his role. Therefore, even these users should have at least a basic knowledge of cyber security. In addition, graduates of the Faculty of Security Engineering, University of Zilina are prepared to work in job positions where information protection is an important part of the job. In the subject Informatics1, students also acquire basic knowledge in this area. The aim of the survey using the questionnaire method was to find out whether they apply this knowledge in practice - specifically during videoconferencing calls, which have become one of the tools providing distance learning.

Keywords — *Cybernetic security, Distance Learning, Videoconference*

I. ÚVOD

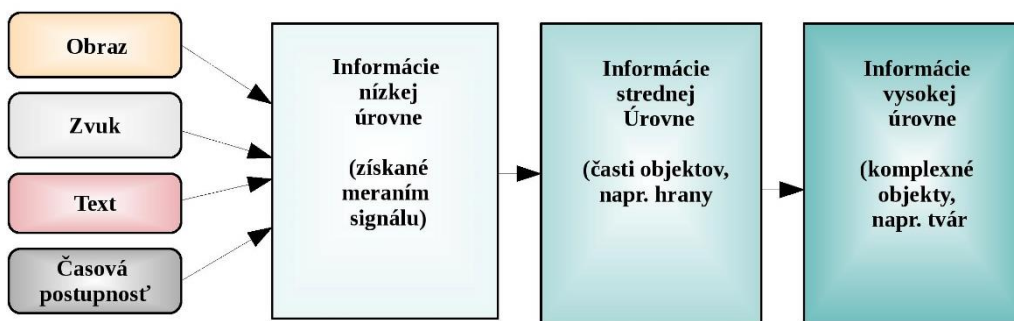
Absolventi Fakulty bezpečnostného inžinierstva Žilinskej univerzity v Žiline sú pripravovaní na pôsobenie v pracovných pozíciách, v ktorých ochrana informácií je dôležitou súčasťou náplne práce. V predmete Informatika1 študenti získavajú i základné poznatky z tejto oblasti. Cieľom prieskumu s použitím dotazníkovej metódy bolo zistiť, či tieto poznatky uplatňujú i v praxi - konkrétne počas videokonferenčných hovorov, ktoré sa stali jedným z nástrojov zabezpečujúcich dištančnú výučbu počas pandémie COVID 19.

II. VIDEOKONFERENCIE AKO SÚČASŤ DIŠTANČNÉHO VZDELÁVANIA A KYBERNETICKÁ BEZPEČNOSŤ

Samozrejme, v súčasnosti je výpočtová technika súčasťou výučby všetkých predmetov na vysokej škole, Fakulta bezpečnostného inžinierstva nie je výnimkou. Dôležitosť nástrojov elektronického vyučovania sa zvýraznila počas obdobia dištančnej výučby kvôli Covid-19. Študentská rada vysokých škôl (ďalej už len ŠRVŠ) v dňoch 20. – 26. marca 2020 zmapovala situáciu výučby pomocou dotazníkového prieskumu o priebehu dištančnej výučby. [1] V tomto dokumente sa konštatuje, že väčšina škôl nebola vopred pripravená na vzniknutú situáciu. Menej ako 10% študentov si myslí, že dištančná forma výučby plnohodnotne nahrádza prezenčnú formu. Takmer polovica študentov sa stretla s výučbou formou videokonferencie.

Videokonferencia umožňuje realizovať komunikáciu medzi dvoma a viacerými účastníkmi v reálnom čase. Tento spôsob elektronického vzdelávania môžeme charakterizovať ako vzdelávanie na základe počítačov. S rozvojom informačných technológií spojených s Internetom vecí, čoraz vyššia dostupnosť mobilných koncových zariadení ako sú tablety a smartfóny, súvisiaca hlavne s ich klesajúcou cenou, zvyšuje pravdepodobnosť dostupnosti tejto formy vzdelávania pre študentov).

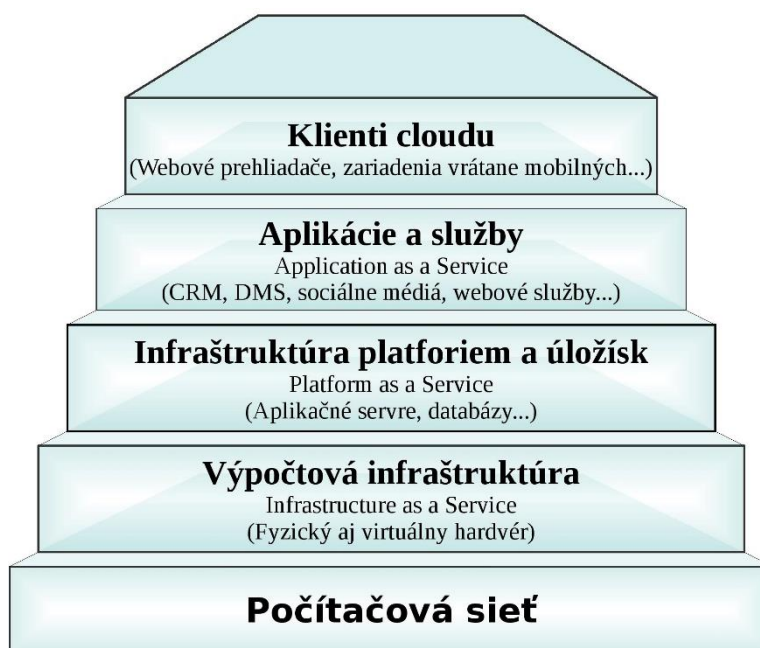
Pri realizácii videokonferencie sú prenášané prostredníctvom počítačovej siete všetky zložky videa – obrazová, zvuková, textová, ktoré sa menia v čase. To zvyšuje angažovanosť študentov v pedagogickom procese. (pozri napr. [2])



Obr. 1 Hierarchia informácií vo videu

Na realizáciu videokonferencie je potrebná počítačová architektúra, ktorá predstavuje výpočtový model umožňujúci veľkému množstvu počítačov prepojených prostredníctvom komunikačnej siete pracujúcej v reálnom čase participovať na riešenej úlohe (videokonferencii), využívať mechanizmy distribuovaného spracovania dát. Takouto architektúrou je cloud architektúra.

Pre cloud architektúru je typické poskytovanie softvéru ako služby, vývojových a výpočtových platforiem ako služby, infraštruktúry ako služby a hardvéru ako služby. Používatelia prístupujú k poskytovaným službám prostredníctvom webového prehliadača, tenkého klienta alebo mobilnej aplikácie zo svojich vzdialených koncových zariadení. Poskytovaný softvér a používateľské dáta sú umiestnené na serveroch, ktoré fyzicky môžu byť veľmi vzdialené. Pri prevádzke v cloude prevádzku zvyčajne zabezpečuje externý dodávateľ, ktorý môže narušiť diskretnosť prenášaných dát - ich nahrávaním a ukladaním



Obr. 2

Hierarchia služieb poskytovaných v Cloude (obrázok modifikovaný podľa [5])

Bezpečnosť v elektronickom vzdelávaní zahŕňa tri oblasti [6]:

- hardvérová bezpečnosť,
- informačná bezpečnosť,
- počítačová bezpečnosť.



Obr. 3 Zložky bezpečnosti elektronického vzdelávania

Hardvérová bezpečnosť je bezpečnosť technických prostriedkov, ktoré sa v používajú v elektronickom vzdelávaní a označuje sa tiež ako fyzická bezpečnosť technických prostriedkov. V prípade e-vzdelávania ide predovšetkým o servery v cloude, ktoré poskytujú softvér a úložiská dát, ale i o koncové zariadenia účastníkov videokonferencie a technické prostriedky infraštruktúry umožňujúce komunikáciu účastníkov videokonferencie (DoS a DDoS útoky).

Informačná bezpečnosť obsahuje počítačovú a sieťovú bezpečnosť. Zahŕňa najmä bezpečnosť a ochranu dát, resp. bezpečnosť komunikačného prostredia

Na svojich stránkach sa téme bezpečnosti počas videokonferenčných hovorov venuje i NCKB SK-CERT [6]. Upozorňuje na spôsoby zneužitia:

- Neoprávnený prístup a narušanie priebehu konferencií (tzv. Bombing).
- Zneužitie zraniteľností v konferenčnom softvéri.
- Slabiny a chyby spôsobené zavádzaním a prevádzkou konferenčného softvéru.
- DoS a DDoS útoky na prebiehajúce videokonferencie.

Videokonferencie sa do zrušenia prezenčnej formy štúdia využívali hlavne v komerčnej oblasti. V tomto prípade boli riešené napríklad tieto otázky bezpečnosti:

- ochrana údajov, ktoré sú predmetom obchodného tajomstva,
- otázka vytvárania trvalých záznamov priebehu videokonferencie,
- bezpečnosť uloženia dát, záznamov z videokonferencií,
- ochrana osobných údajov,
- snímaná tvár účastníkov konferencie, tento údaj patrí do skupiny osobných údajov spadajúcich pod ochranu GDPR,
- využívanie dát z videokonferencií na Data Mining,
- zamedzenie vstupu neautorizovaných osôb do komunikácie,
- jednoznačná identifikácia účastníkov videokonferencie,
- zabezpečenie ochrany pracoviska, na ktorom sa nachádzajú účastníci videokonferencie,

...

V článkoch venujúcich sa bezpečnosti videokonferencií (napr. [8], [9]) je zdôraznená skutočnosť, že len časť otázok bezpečnosti je riešiteľná technickými prostriedkami. Dôležitou a neoddeliteľnou súčasťou bezpečnosti priebehu videokonferenčných hovorov je budovanie kultúry bezpečnosti v organizácii, stanovenie obmedzení pre jednotlivých účastníkov videokonferencie, stanovenie pravidiel používania technických a softvérových prostriedkov.

Pri používaní videokonferencií v akademickom prostredí je dôležité riešiť i tieto otázky bezpečnosti:

- obmedzenie možnosti využitia videokonferencie na neetickú, resp. protiprávnu činnosť,
- ochrana súkromia účastníkov konferencie.

Tieto otázky vyplývajú hlavne zo skutočnosti, že účastníci sa počas konania videokonferencie nachádzajú v prostredí mimo univerzity, často vo vlastnom obydli, v súkromí. Je potrebné zdôrazniť, že na nutnosť chrániť si svoje súkromie by mali byť upozornení všetci účastníci videokonferenčného hovoru.

Na stránke NCKB SK-CERT [7] nájdeme tieto odporúčania pre bezpečnosť videokonferencií:

- Na videokonferenčné hovory používajte známy softvér s dobrou reputáciou a s adekvátnymi prvkami bezpečnosti, ako je napríklad šifrovanie sieťovej komunikácie, dvojfaktorové overenie totožnosti pri prihlasovaní a podobne.
- Najmä v prípade verejnej správy neodporúčame používať aplikáciu Zoom. Odporúčame použiť iné, bezpečnejšie alternatívy.
- Používajte len aktualizovaný softvér a v prípade vydania bezpečnostných aktualizácií neodkladajte ich inštaláciu.
- Svoje konto do softvéru chráňte komplexným heslom a ak je to možné, aj viacnásobnou autentifikáciou (vyhnite sa autentifikáciou prostredníctvom SMS).

- Každý videokonferenčný hovor chráňte komplexným a ťažko uhádnuteľným heslom. Nepoužívajte rovnaké heslo vo viacerých videokonferenčných hovoroch.
- Overujte každého účastníka videokonferencie, najlepšie kontrolovaním a riadením vstupov do prostredia videokonferencie (funkcia „čakárne“).
- Videokonferencie nastavujte ako súkromné a nie verejné.
- Nezdieľajte odkaz na videokonferenciu verejne prostredníctvom sociálnych sietí a podobne, odkaz zdieľajte iba s konkrétnymi ľuďmi, ktorí sa majú videokonferencie zúčastniť.
- Ak chcete s účastníkmi telekonferencie komunikovať citlivé údaje, robte to tak, aby ste časti informácií rozdelili a časť povedali počas hovoru a druhú časť odoslali v správe cez inú aplikáciu.
- Ak máte akékoľvek podozrenie na kompromitáciu videokonferencie, prípadne sa vaše zariadenie správa zvláštno, bezodkladne o tom informujte svojho zamestnávateľa a osobu, ktorá je vo vašej organizácii zodpovedná za kybernetickú a informačnú bezpečnosť.

III. UPLATNENIE POZNATKOV O KYBERNETICKEJ BEZPEČNOSTI ŠTUDENTAMI POČAS DIŠTANČNEJ VÝUČBY

Všetci študenti Fakulty bezpečnostného inžinierstva absolvujú ako prvý predmet zameraný priamo na výpočtovú techniku a informačnú techniku predmet informatika I. Jednou z oblastí, z ktorej sú študentom poskytované základné poznatky, sa stala kybernetická bezpečnosť. Študenti sú oboznamovaní hlavne so základnými pojmami ilustrovanými na konkrétnych prípadoch z praxe.

Každý profesionál, ktorého pracovná náplň súvisí priamo či nepriamo so zabezpečením informačnej bezpečnosti, by mal získať dostatočné poznatky i z oblasti kybernetickej bezpečnosti, aby dokázal byť partnerom pracovníkom, ktorí spravujú informačné systémy a sú zodpovední za ich prevádzku. Musí poznať princípy, metodológiu a technológie, aby dokázal presadzovať bezpečnostné požiadavky organizácie, ktorú zastupuje pri vyjednávaniach so stranami zainteresovaným v procese zabezpečovania a hodnotenia bezpečnosti vrátane kybernetickej bezpečnosti.

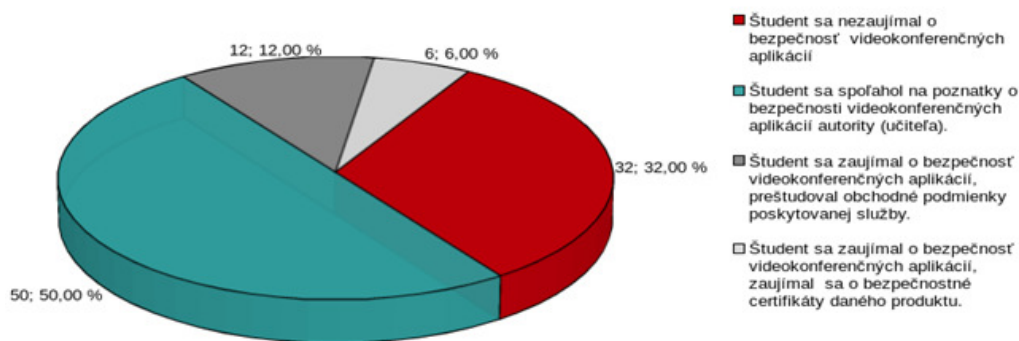
Kvôli pretrvávajúcej dištančnej výučbe nebolo možné realizovať vyplnenie dotazníka na vyučovaní. Preto bol vytvorený anonymný dotazník v systéme elektronického vyučovania – Moodle. Prieskumu sa zúčastnili študenti prvého ročníka všetkých odborov. Počet respondentov (36) bol výrazne ovplyvnený skutočnosťou, že študenti neboli okolnosťami motivovaní, aby dotazník vyplnili. Je možné predpokladať, že prieskumu sa zúčastnili iba tí študenti, ktorí majú k výpočtovej technike, jej používaniu a riešeniu problematiky kybernetickej bezpečnosti kladný vzťah, prípadne neutrálny.

Väčšina respondentov používala počas trvania dištančnej výučby tri videokonferenčné systémy (47 %), menší podiel z nich používal dva systémy (25 %) a časť z nich používala dokonca viac ako štyri videokonferenčné systémy (11 %). Väčšina študentov túto skutočnosť považovala za nepríjemnú, obťažujúcu kvôli nutnosti inštalácie rôznych aplikácií (78 %). Potešujúce je, že vyučujúci boli schopní študentom poskytnúť technickú podporu. Väčšina študentov by uvítala používanie jednotného systému pre videokonferenčné hovory (75 %). Tieto výsledky korešpondujú s prieskumom vykonaným Radou vysokých škôl.

Prekvapujúcim výsledkom prieskumu je skutočnosť, že videokonferenčné hovory väčšina študentov nepoužíva na súkromnú komunikáciu (67 %). To je v rozpore s predpokladom, že získaná skúsenosť s používaním takýchto systémov zvýši pravdepodobnosť, že videokonferenčný hovor bude uprednostnený pri výbere komunikačného kanála i v súkromnej komunikácii. Dôvodom môže byť hardvérové vybavenie a skúsenosti ostatných osôb zúčastňujúcich sa súkromnej komunikácie.

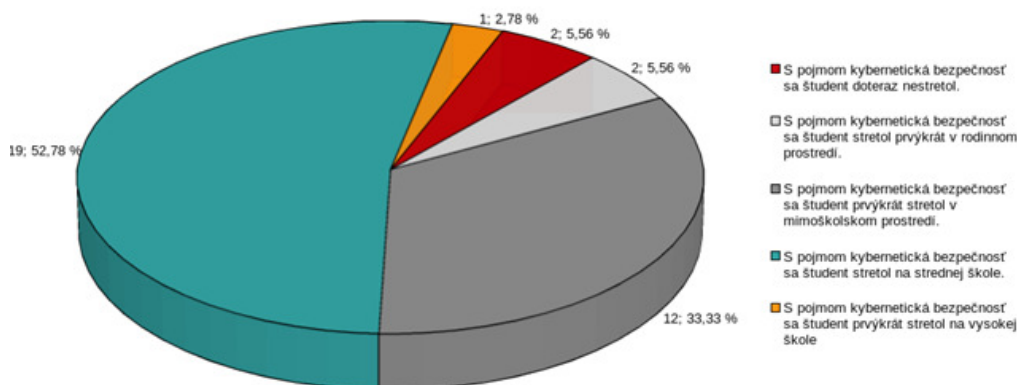
Cieľom bolo zistiť, ako študenti pri takejto komunikácii využívajú, uplatňujú poznatky prezentované, získané na vyučovaní.

Na otázku, či sa zaujímali o bezpečnosť používania takýchto systémov, väčšina odpovedala, že sa spoliehajú na autoritu, v tomto prípade učiteľa (50 %). Učiteľ však môže byť rovnakým laikom v oblasti informačnej bezpečnosti ako študent. Veľká časť odpovedala, že sa o bezpečnosť aplikácie nezaujímalá vôbec (32 %). Toto svedčí o tom, že časový tlak spôsobil, že sa študenti snažili o riešenie problému a nehodnotili, resp. akceptovali aj väčšie riziko.



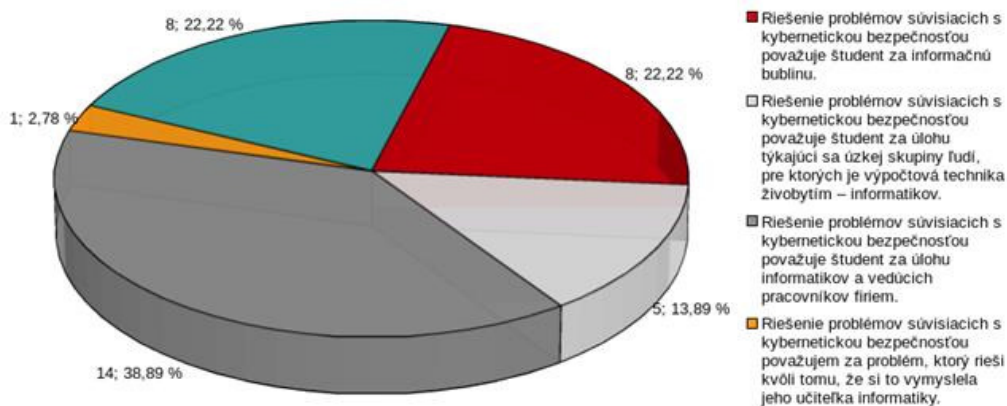
Obr. 4 Záujem študentov o bezpečnosť videokonferenčných aplikácií

Časť študentov sa s pojmom kybernetická bezpečnosť stretla po prvýkrát na nižšom (strednom) stupni škôl (78 %). Toto hodnotím ako veľmi kladnú skutočnosť – vzhľadom na to, že používatelia systémov sa stávajú čoraz mladšie vekové kategórie, ktoré sú vzhľadom na svoje obmedzené skúsenosti veľmi ohrozenou skupinou. V rodinnom prostredí sa prvýkrát s týmto pojmom stretla iba malá časť študentov (5,56 %) – toto môže poukazovať na to, že rodičia nemajú vzdelanie/skúsenosti s touto problematikou, neprikladajú jej dôležitosť. Väčšina rodičov žiakov je v produktívnom veku, preto to môže poukazovať na širší problém s úrovňou poznatkov o kybernetickej bezpečnosti v spoločnosti.



Obr. 5 Miesto prvého stretnutia študenta s pojmom kybernetická bezpečnosť

Väčšina študentov riešenie otázok súvisiacich s kybernetickou bezpečnosťou považuje za problém informatikov a riadiacich pracovníkov (takmer 39 %). To, že tieto otázky rieši každý používateľ pri každej práci v systéme, si uvedomuje len malá časť študentov (2,78 %).



Obr. 6 Kto je zodpovedný za riešenie problémov kybernetickej bezpečnosti podľa názoru študentov

Hlavným zdrojom poznatkov z oblasti kybernetickej bezpečnosti je pre väčšinu respondentov Internet. V tejto otázke mohli študenti označiť len jednu odpoveď z ponúkaných, preto je možné, že ich zdroje informácií sú rôznorodejšie.

Študenti si uvedomujú potrebu ochrany súkromia len čiastočne. Nedoceňujú skutočnosť, že informácie o súkromí je možné získať nielen priamo, ale i nepriamo. Niektorí z respondentov síce riešili prostredie, z ktorého uskutočňovali videokonferenčné hovory (22 %), ale väčšina sústredila svoj záujem na len na to, či obsah komunikácie bude pre nich zaujímavý (67 %). Respondenti majú malé poznatky z oblasti biometrických systémov (len 33 % z nich pozná klady i nevýhody používania takýchto metód) a nezaujímajú sa o to, aké dáta je možné získať z videa. Pri registrácii do systémov väčšina z nich používa iba čiastočne pravdivé informácie (56 %), ale na druhej strane časť z nich využíva možnosť použiť na prihlásenie sa do systémov prostredníctvom účtu na sociálnych sieťach (25 %). S podmienkami používania osobných údajov súhlasí vysoký podiel respondentov automaticky (44 %). Len malú časť z respondentov na základe svojho nesúhlasu s týmito podmienkami hľadá inú alternatívu (17 %), 37 % študentov je presvedčených, že väčšinou nemá zmysel čítať podmienky poskytovania služby, pretože službu bez tohto súhlasu nemôžu využívať..

Na komunikáciu počas dištančnej výučby používali študenti väčšinou súkromné zariadenia. Na týchto zariadeniach majú väčšinou vytvorený len jeden – administrátorský účet (50 %), prípadne viacero účtov, ale všetky s administrátorskými právami (8 %). Účet vytvorený na prácu, ktorý by mal obmedzené prístupové práva k softvéru a hardvéru zariadenia, používa iba 8 % študentov..

V dotazníku respondenti uviedli, že za najdôležitejšiu súčasť ochrany svojich zariadení považujú používanie hesla/PIN (47 %), prípadne viacfaktorovej metódy. Súčasťou dotazníka bola i úloha, v ktorej mali respondenti určiť najbezpečnejšie heslo. 69% z nich bolo úspešných, 22 % z nich si nebolo istých. Nebolo testované, či študenti správne identifikujú viacfaktorovú metódu na prístup do zariadenia, resp. či vedia uviesť nejaký príklad.

IV. ZÁVER

Spätná väzba poskytnutá študentami umožňuje identifikovať oblasti manažmentu kybernetickej bezpečnosti, ktorým je potrebné sa venovať počas výučby podrobnejšie a zdôrazniť ich príspevok ku kybernetickej bezpečnosti ako celku.

Zabezpečenie kybernetickej bezpečnosti sa stáva čoraz dôležitejšou úlohou. Zapojenie čoraz väčšieho počtu zariadení do nehomogénneho prostredia Internetu vecí prináša so sebou čoraz väčšie riziká. Neoddeliteľnou súčasťou systému manažovania kybernetickej bezpečnosti je človek – či už v úlohe experta alebo bežného používateľa. Na tieto potreby praxe musí reagovať i akademické prostredie a pripravovať odborníkov schopných vysporiadať sa s touto úlohou.

REFERENCES

- [1] PŘEDSEDNÍCTVO ŠRVŠ. *Sprievodca výučbou počas COVID-19* [online]. [cit. 2020-06-18]. Dostupné z: <https://srvs.eu/covid-19/>.
- [2] Schindler, L.A., Burkholder, G.J., Morad, O.A. et al. *Computer-based technology and student engagement: a critical review of the literature*. Int J Educ Technol High Educ 14, 25 (2017). <https://doi.org/10.1186/s41239-017-0063-0>.
- [3] Nedeva, Veselina & Dineva, Snejana & Atanasov, Svetoslav. (2014). Effective E-learning Course With Web Conferencing. DOI: 10.13140/2.1.2605.6648
- [4] DR KISHORE KUMAR DAS a AFTAB ARA. *E-learning by cloud computing-challenges_benefits_and_changes_a_case_study_of_almedina_international_university*. INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY. 2014, 7(3), 366-378. DOI: <https://doi.org/10.1186/s41239-017-0063-0>. ISSN 2277-9655..
- [5] *Cloud infrastructure* [online]. [cit. 2020-06-17]. Dostupné z: <https://searchcloudcomputing.techtarget.com/definition/cloud-infrastructure>
- [6] LEVICKÝ, Dušan. *Kryptografia v informačnej a sieťovej bezpečnosti*. Elfa, 2010. ISBN 9788080861636
- [7] *NCKB SK-CERT odporúča vyhnúť sa používaniu rizikových platforiem na uskutočňovanie videohovorov*. Www.sk-cert.sk [online]. [cit. 2020-06-17]. Dostupné z: <https://www.sk-cert.sk/sk/nckb-sk-cert-odporuca-vyhnut-sa-pouzivaniu-rizikovych-platforiem-na-uskutocnovanie-videohovorov/index.html>
- [8] MOSELEY, Sara. *5 Steps for Secure Video Conferencing* [online]. [cit. 2020-06-18]. Dostupné z: <https://highfive.com/blog/5-steps-for-secure-video-conferencing>
- [9] TAYLOR, Ian. *Video Conferencing Security: Four Key Considerations* [online]. 2007 [cit. 2020-06-18]. Dostupné z: <http://www.uctoday.com/collaboration/video-conferencing-security-four-key-considerations/>

Author's index

- A**
Adamišín Kamil 445
- B**
Babič František 316, 320, 324, 339, 344
Bačík Ján 7
Baláž Anton 21, 491, 497, 505
Baláž Róbert 16
Balluch Tomáš 450
Baumgartner Maroš 132
Bavolárová Natália 434
Beňa Lubomír 379, 423
Bernát Ivan 70
Bilanský Juraj 385
Birková Anna 64
Biroš Jozef 235
Biroš Samuel 428
Bobík Pavol 70, 76, 407
Bučko Jozef 27
Budiš Michal 445
Butka Peter 350, 355, 360, 365, 369
- C**
Cimbala Roman 167, 173
Cymbalák Dávid 42
- Č**
Čonka Zsolt 213
Čontofalský Martin 42
- D**
Daňko Peter 324
Dávidová Zuzana 373
Drutarovský Miloš 184
Džmura Jaroslav 52, 413
- F**
Fedor Pavol 283
Fedorčák René 491
Ferenčík František 160
- Frivaldský Jozef 10
- G**
Gadušová Zuzana 115
Genčí Ján 58, 64, 70, 76, 82, 407
Girovský Peter 37, 307, 402
Gladišová Iveta 109, 278
Goliaš Kristián 58
Grib Jakub 484
Grobár Samuel 256
- H**
Hai Le Thanh Kristián 350
Haluška Renát 132, 256, 296, 299
Halcko Martin 229
Handriková Jana 120, 132, 511
Harman Martin 360
Havlice Zdeněk 154, 160
Havran Peter 167, 173
Hruška Lukáš 445
Hudák Marián 450, 456, 462
Humeník Jozef 302
Humeník Lukáš 369
Huszaník Tomáš 241, 248
- Ch**
Chlebovec Martin 184
- I**
Ivan Michal 462
- J**
Jacková Júlia 479
Jakab František 190
Jakubová Jana 339
Janok Michal 291
Jurenka Vladimír 396
Juščík Tomáš 262
- K**
Kačmáreková Radka 316
Kainz Ondrej 190
Kanálik Martin 379
Kancír Patrik 126
Kaňuch Ján 37, 307
Karabaš Dávid 190
Karla Jakub 445
Kindernayová Iveta 154
Kisel Teodor 207
Klembara Erik 93
Klimonová Diana 344
Kohan Vladimír 213, 220
Kokinda Matúš 154
Kolárik Michal 474
Kolcun Michal 213, 220
Korečko Štefan 450, 456, 462, 484
Krištof Vladimír 47
Kulbaga Peter 355
Kužmová Monika 196
- L**
Lacko Milan 385
Lapčák Maroš 299
Ličko Peter 179
- M**
Madoš Branislav 21, 491, 497, 505
Mach Marián 10, 16, 93, 179, 235, 390
Machová Kristína 202, 274, 291, 468
Marcezi Tomáš 505
Margitová Anastázia 379
Marchevský Stanislav 256
Maslej Krešňáková Viera 350, 355, 360, 365, 369, 373
Matiková Kristína 456
Maťašovský Kristián 365
Medved Dušan 115, 126, 142, 207, 229, 328, 396
Mihalík Ján 109, 138, 278
Michalko Miroslav 190

- N**
Nagyová Kinga 148
Nezník Richard 98
- O**
Oliinyk Maksym 52, 413
Oravec Jakub 32
Osvald Dominik 82
Ovseník Luboš 32, 241, 248, 296, 299
- P**
Papaj Ján 98, 104
Pál Daniel 47, 423
Perduková Daniela 283
Petija Rastislav 190
Pleva Matúš 27, 87
Pusztová Ludmila 339, 344
- R**
Rusňák Richard 268
- Rúšová Veronika 87
- S**
Sarnovský Martin 373, 474, 479
Sklenáriková Alžbeta 21
Skyba Marek 64
Smolnický Peter 274
Sobota Branislav 450, 456, 462, 484
Solanik Michal 407
Sulír Matúš 428, 434
- Š**
Šimoňák Slavomír 148, 196
Štieberová Nikola 379
ŠulaJ Peter 120, 132, 256, 296
- T**
Tarhaničová Michaela 390
Tkáč Peter 7
Tomčo Miroslav 142
- Trembulák Ján 154
Turán Ján 32, 241, 248
Tužák Patrik 320
- U**
Urbanský Jakub 47, 423
- V**
Vangor Branislav 328
Vargová Lucia 202
Varchola Radko 497
Vasilko Matej 468
Vaščák Ján 262, 268, 334, 419, 440
Vaško Martin 76
Vašková Gabriela 104
Vrábel Michal 58
- Z**
Záborský Anton 241, 248
Zdravecký Norbert 248

Electrical Engineering and Informatics XI

Proceedings of the
Faculty of Electrical Engineering and Informatics
of the Technical University of Košice

Published by: Faculty of Electrical Engineering and Informatics
Technical University of Košice
Edition I, 520 pages
Number of CD Proceedings: 50 pieces

Editors: prof. Ing. Alena Pietriková, CSc.
Ing. Emília Pietriková, PhD.

ISBN 978-80-553-3585-8