

IT'14
ŽABLJAK

XIX

naučno - stručni skup

**INFORMACIONE
TEHNOLOGIJE**

SADAŠNJOST I BUDUĆNOST

Urednik
Božo Krstajić

IT'14

**INFORMACIONE
TEHNOLOGIJE**

- SADAŠNJOST I BUDUĆNOST -

Urednik
Božo Krstajić

*Zbornik radova sa XIX naučno - stručnog skupa
INFORMACIONE TEHNOLOGIJE - sadašnjost i budućnost
održanog na Žabljaku od 24. do 28. februara 2014. godine*

Zbornik radova
INFORMACIONE TEHNOLOGIJE - sadašnjost i budućnost 2014

Glavni urednik
Prof.dr Božo Krstajić

Izdavač
Univerzitet Crne Gore
Elektrotehnički fakultet
Džordža Vašingtona bb., Podgorica
www.etf.ucg.ac.me

Tehnička obrada
Aleksandra Radulović
Centar Informatičnog Sistema
Univerziteta Crne Gore

Tiraž
150

Podgorica 2014.

Sva prava zadržava izdavač i autori

Organizator

Elektrotehnički fakultet Univerziteta Crne Gore

Skup su podržali:

- * Ministarstvo za informaciono društvo i telekomunikacije
- * Pošta Crne Gore
- * Agencija za elektronske komunikacije i poštansku djelatnost
- * Čikom

Programski odbor

Dr Novak Jauković, Elektrotehnički fakultet, Podgorica
Akademik Dr Ljubiša Stanković, CANU
Dr Zdravko Uskoković, Elektrotehnički fakultet, Podgorica
Dr Vujica Lazović, Ekonomski fakultet, Podgorica
Dr Branko Kovačević, Elektrotehnički fakultet, Beograd
Dr Milorad Božić, Elektrotehnički fakultet, Banja Luka
Dr Miroslav Bojović, Elektrotehnički fakultet, Beograd
Dr Zoran Jovanović, Elektrotehnički fakultet, Beograd
Dr Božidar Krstajić, Elektrotehnički fakultet, I. Sarajevo
Dr Milica Pejanović-Đurišić, Elektrotehnički fakultet, Podgorica
Dr Dejan Popović, Elektrotehnički fakultet, Beograd
Dr Božo Krstajić, Elektrotehnički fakultet, Podgorica
Dr Milovan Radulović, Elektrotehnički fakultet, Podgorica
Dr Budimir Lutovac, Elektrotehnički fakultet, Podgorica
Dr Igor Radusinović, Elektrotehnički fakultet, Podgorica
Dr Igor Đurović, Elektrotehnički fakultet, Podgorica
Dr Miloš Daković, Elektrotehnički fakultet, Podgorica
Dr Milutin Radonjić, Elektrotehnički fakultet, Podgorica
Dr Ramo Šendelj, Fakultet za Informacione Tehnologije, Podgorica
Dr Stevan Šćepanović, Prirodno-matematički fakultet, Podgorica
Dr Sašo Gelev, Elektrotehnički fakultet, Radoviš

Organizacioni odbor

Dr Božo Krstajić, Elektrotehnički fakultet, Podgorica
Dr Milovan Radulović, Elektrotehnički fakultet, Podgorica
Dr Ana Jovanović, Elektrotehnički fakultet, Podgorica
Dr Saša Mujović, Elektrotehnički fakultet, Podgorica
MSc Žarko Zečević, Elektrotehnički fakultet, Podgorica
Vladan Tabaš, dipl.ing., Čikom

Sekretarijat

Aleksandra Radulović, CIS Univerzitet Crne Gore

P R E D G O V O R

Poštovani učesnici i čitaoci,

Pred vama je zbornik radova koji su prezentovani na XIX Naučno-stručnom skupu "INFORMACIONE TEHNOLOGIJE – sadašnjost i budućnost" (IT'14) koji je uspješno održan od 24. do 28. februara 2014. godine na Žabljaku. U cilju sveobuhvatnog i multidisciplinarnog sagledavanja aktuelnosti u oblasti informaciono-komunikacionih tehnologija u Crnoj Gori i svijetu, Programski odbor je izvršio selekciju kvalitetnih radova čiji su rezultati bili prezentovani učesnicima Skupa, dostupni svim korisnicima Interneta preko web sajta www.it.ac.me I konačno vama putem ovog zbornika. Konstantno povećanje broja prijavljenih i prezentovanih radova, sve širi krug eminentnih naučnika uključenih u proces recenzije i izbor najboljih radova koji će se štampati u časopisu Elektrotehničkog fakulteta Univerziteta Crne Gore u Podgorici ("ETF Journal of Electrical Engineering") su jasna potvrda uvećanju kredibiliteta i kvaliteta konferencije.

U okviru Skupa je održana radionica „Razvoj mobilnih aplikacija“, okrugli sto "IXP4ME – potreba i dileme" i prezentovani projekti: "Fostering innovation based research for e-Montenegro – Fore-Mont", "ECDL za digitalnu Crnu Goru", "Wireless Montenegro" i „Open Govrenment Data (OD) inicijative u Crnoj Gori“.

U ime organizatora skupa se zahvaljujem svima koji su na bilo koji način učestovali u radu ovogodišnje konferencije i najavljujem posebno osmišljen i bogat program za naredni-jubiley XX IT.

Sve detalje o ovom, prošlim i narednom skupu možete naći na poznatoj adresi www.it.ac.me.

Prof. dr Božo Krstajić

SADRŽAJ

Vesna Rubežić, Ana Jovanović HAOS U LASERSKOM SISTEMU SA ERBIJUM-DOPIRANIM VLAKNOM.....	1
Žarko Zečević, Igor Đurović, Božo Krstajić DISTRIBUIRANI SET-MEMBERSHIP NLMS ALGORITAM	5
Žarko Zečević, Božo Krstajić BRZI COUPLED LMS ALGORITAM.....	9
Luka Lazović, Ana Jovanović, Vesna Rubežić UPOREDNA ANALIZA PERFORMANSI CAPON I CAPON-LIKE ALGORITAMA U SISTEMIMA PAMETNIH ANTENA	13
Mladen Rašović, Jadranka Radović, Saša Mujović MODELOVANJE SVJETILJKI PRI ANALIZAMA KVALITETA ELEKTRIČNE ENERGIJE U MREŽAMA JAVNE RASVJETE	17
Novica Daković, Milovan Radulović FLATNESS UPRAVLJANJE AUTONOMNO VOĐENIM VOZILOM	21
Balša Femić, Stevan Šćepanović PRIVATNI OBLAK OTVORENOG KODA.....	25
Luka Filipović, Božo Krstajić PREDLOG POBOLJŠANJA MASTER-SLAVE ALGORITMA ZA RASPODJELU OPTEREĆENJA U MPI PARALELNIM APLIKACIJAMA.....	29
Sidita Duli HIBRIDNA MPI/PTHREADS PARALELIZACIJA ZA ESTIMACIJU PARAMETARA WEIBULL DISTRIBUCIJE	33
Uglješa Urošević, Zoran Veljović POBOLJŠANJE ENERGETSKE EFIKASNOSTI OFDM-CDMA SISTEMA	36
Srdjan Jovanovski, Veselin N. Ivanović PIPELINE-OVANI SISTEM ZA ESTIMACIJU VISOKO NESTACIONARNIH FM SIGNALA	40
Borko Drašković INTEGRACIJA SaaS SERVISA U CLOUD TELEKOMA SRBIJA	44
Nemanja Filipović, Radovan Stojanović MONITORING I ANALIZA VITALNIH FIZIOLOŠKIH PARAMETARA PRIMJENOM PDA UREĐAJA	48
Roman Golubovski KONCEPT ZA EKSPERTNI SISTEM ZA AUTOMATIZOVANU EKG DIJAGNOSTIKU.....	52

Nataša Savić, Zoran Milivojević, Darko Brodić ANALIZA EFIKASNOSTI KVADRATNIH KONVOLUCIONIH JEZGARA KOD PROCENE FREKVENCIJE SIGNALA	56
Igor Ivanović, Srđan Kadić UPOTREBA OpenFlow STANDARDA ZA BALANSIRANJE NFS SERVERA SA SISTEMOM POVRATNE SPREGE.....	60
Miladin Tomić, Milutin Radonjić, Neđeljko Lekić, Igor Radusinović VIRTUELIZACIJA MREŽE KORIŠĆENJEM ALATA FLOWVISOR	64
Slavica Tomović, Milutin Radonjić, Igor Radusinović IMPLEMENTACIJA RIP I OSPF PROTOKOLA NA QUAGGA SOFTVERSKOJ PLATFORMI.....	68
Rabina Šabotić, Milutin Radonjić, Igor Radusinović DISTRIBUCIJA UNIVERZALNOG KOORDINISANOG VREMENA.....	72
Neđeljko Lekić, Almir Gadžović, Igor Radusinović V2X SISTEMI KOOPERATIVNE MOBILNOSTI.....	76
Jelena Šuh, Vladimir Čulum PROTOKOLI ZA REDUDANSU U IP MREŽAMA	80
Pero Bogojević, Jasna Mirković MOBILNA APLIKACIJA CRNOGORSKOG TELEKOMA	84
Mirko Kosanović, Miloš Kosanović INTEGRACIJA BEŽIČNIH SENZORSKIH MREŽA U CLOUD COMPUTING-u	88
Aleksandar Trifunović, Svetlana Čičević, Andreja Samčović, Milkica Nešić PRIMENA TABLET TEHNOLOGIJE U SAVLAĐIVANJU REČI ENGLESKOG JEZIKA	92
Aleksandar Ristić, Dalibor Damjanović OPRAVDANOST INICIJATIVE ZA IZMJENU NASTAVNOG PLANA I PROGRAMA INFORMATIKE U SREDNJEM STRUČNOM OBRAZOVANJU EKONOMSKE I TRGOVINSKE STRUKE, ZANIMANJE TRGOVAC'	96
Risto Bojović MODEL STRATEŠKOG RAZMIŠLJANJA U IT OKRUŽENJU	100
Vuko Perišić STRATEŠKO PLANIRANJE ICT-A U SKUPŠTINI CRNE GORE.....	104
Željko Pekić, Stevan Kordić, Draško Kovač, Tatijana Dlabač, Nađa Pekić ANALIZA ONLINE KOMUNIKACIJE I INTERAKCIJE KROZ E-LEARNING.....	108
Dejan Abazović, Budimir Lutovac ITIL - IMPLEMENTACIJA INCIDENT MANAGEMENT PROCESA U SERVICE DESK-U SA PREDLOGOM ZA NJEGOVO UNAPREĐENJE	112

Jelena Končar, Sonja Leković IMPLEMENTACIJA INTERAKTIVNE ELEKTRONSKE MALOPRODAJE U REPUBLICI SRBIJI	116
Obradović Milovan EVOLUCIJA SISTEMA ZA PODRŠKU ODLUČIVANJU I NJIHOVE PRIMENE U ZDRAVSTVU	120
Ilija Apostolov, Risto Hristov, Sašo Gelev IZBOR OPTIMALNE TEHNIKE ZA ENKRIPCIJU I DEKRIPCIJU PODATAKA.....	124
Tamara Pejaković, Miloš Orović, Andjela Draganić, Irena Orović INFORMACIONI SISTEM ZA IZVJEŠTAVANJE O PRODUKTIVNOSTI AGENATA OSIGURAVAJUĆEG DRUŠTVA.....	128
Predrag Raković, Vasilije Stijepović ANALIZA POTREBA ZA PRIKUPLJANJE PODATAKA, EVALUACIJU I PRAĆENJE SPORTSKIH POVREDA U CRNOGORSKIM SPORTSKIM KLUBOVIMA I SAVEZIMA.....	132
Bogdan Mirković ISTRAŽIVANJE NEFUNKCIONALNIH ZAHTJEVA INFORMACIONIH SISTEMA	135
Bogdan Mirković KRITERIJUMI ZA MJERENJE USPJEŠNOSTI INFORMACIONIH SISTEMA.....	139
Sanja Bauk, Tatijana Dlabač, Radoje Džankić O AMOS SOFTVERU NAMIJENJENOM ELEKTRONSKOM UPRAVLJANJU RESURSIMA NA BRODU	143
Predrag Raković IMPLEMENTACIJA MPI ZA UBRZANJE ESTIMACIJE PARAMETARA 2DCPPPS PRIMJENOM 2DCPF-A	147
Milovan Radulović, Vesna Rubežić, Martin Čalasan HAOTIČNI OPTIMIZACIONI METOD SINTEZE PID REGULATORA U AVR SISTEMU....	150
Tamara Bojičić, Vesna Popović-Bugarin UTICAJ KRITERIJUMA MINIMIZACIJE NA UPRAVLJANJE POTROŠNJOM ELEKTRIČNE ENERGIJE	154
Mirjana Božović, Saša Mujović PRIMJENA SAVREMENOG MJERNO-AKVIZICIONOG SISTEMA ZA MONITORING KVALITETA ELEKTRIČNE ENERGIJE NA PRIMJERU ŽELJEZARE NIKŠIĆ	158
Ana Grbović, Bojan Đordan IMPLEMENTACIJA I VIZUELIZACIJA GRUPNE REGULACIJE U HE PERUĆICA	162
Roman Golubovski JEFTINO PIC BAZIRANO REŠENJE ZA AUTO-TRAKING FOTONAPONSKIH PANELA.....	166

Saša Stojanović, Dragan Tošić, Zoran Milivojević SUN TRACKER SISTEM BAZIRAN NA MIKROKONTROLERU.....	170
Dimitrija Angelkov, Cveta Martinovska Bande UPRAVLJANE ROBOTA PREKO INTERNETA.....	173
Isak Karabegović, Ermin Husak, Milena Đukanović APLIKACIJA INTELIGENTNIH SISTEMA-ROBOTA.....	177
Aleksandar Sokolovski, Sašo Gelev UPOTREBA GPU U SISTEMIMA ZA DETEKCIJU E-MAIL SPAM-A I IDS.....	181
Aleksandar Vučeraković PRIMJER UPRAVLJANJA RAČUNARSKIM SISTEMOM PUTEM GRUPNIH POLISA.....	185
Igor Miljanić PRIMJER ADMINISTRIRANJA HETEROGENOG RAČUNARSKOG SISTEMA RTCG.....	189
Zoran Veličković, Miloško Jevtović ADAPTACIJA IZGLEDA WEB STRANICE USLOVLJENA KLIJENSKIM SPECIFIČNOSTIMA U ASP .NET MVC 4 OKRUŽENJU.....	193
Stevan Šandi, Tomo Popović, Božo Krstajić ALATI ZA PODRŠKU MJERENJU SINHROFAZORA.....	197
Zoran Milivojević, Zoran Veličković, Dragiša Balanesković PROCENA INHARMONIČNOSTI KOPIJE ANTONIUS STRADIVARIUS VIOLINE.....	201
Biljana Chitkusheva Dimitrovska, Maja Kukusheva, Vlatko Chingoski LTspice IV KAO EDUKATIVNO SREDSTVO U NASTAVI ANALIZE ELEKTRICNIH KOLA.....	205
Petar Radunović, Tijana Vujičić, Ivan Knežević POREĐENJE FUNKCIONALNOG I IMPERATIVNOG PRISTUPA PROGRAMIRANJU.....	209
Jelena Ljucović, Ivana Ognjanović, Ramo Šendelj INTEGRACIJA ISTORIJSKIH PODATAKA U AHP ALGORITAM.....	213
Tijana Vujičić, Petar Radunović, Ivan Knežević KOMPARATIVNA ANALIZA NOSQL I SQL BAZA PODATAKA, NA PRIMJERU DATOMICA I MSSQL-A.....	217
Dragan Vidakovic, Dusko Parezanovic KRIPTOSISTEMI JAVNOG KLJUČA I GOLDBAHOVA PRETPOSTAVKA.....	221
Srđan Kadić, Milenko Mosurović POBOLJŠANA METODA VALIDACIJE 3N+1 HIPOTEZE PUTEM TRANSFORMACIJA.....	224
Stevan Šćepanović, Marko Grebović PLANIRANJE OBLASTI POKRIVENOSTI WLAN MREŽA.....	228

Milica Medenica, Sanja Zuković, Andjela Draganić, Irena Orović, Srdjan Stanković POREĐENJE ALGORITAMA ZA CS REKONSTRUKCIJU SLIKE.....	232
Marko Asanović, Radovan Stojanović, Igor Đurović METODA DETEKCIJE VATRE U REALNOM VREMENU NA BAZI OBRADU SLIKE.....	236
Nikola Besić, Gabriel Vasile, Budimir Lutovac, Srđan Stanković, Dragan Filipović ANALIZA PERFORMANSI FASTICA ALGORITMA PRIMIJENJENOG NA 2D SIGNAL	240
Bojan Prlinčević, Zoran Milivojević, Darko Brodić EFIKASNOST MDB ALGORITMA KOD FILTRIRANJA SLIKA SA VODENIM ŽIGOM	244
Ratko Ivković, Mile Petrović, Petar Spalević, Dragiša Miljković, Boris Gara UTICAJ LINEARNOG OSVETLJENJA NA NIVO DETALJA I ENTROPIJU SLIKE	248
Alija Dervić, Neđeljko Lekić PREPOZNAVANJE DUŽICE OKA I UPOTREBA BIOCAM VISTA FA2.....	252
Luka Čadenović UTICAJ PUNJENJA VOZILA NA ELEKTRIČNI POGON NA KVALITET ELEKTRIČNE ENERGIJE U DISTRIBUTIVNOJ MREŽI	256

HAOS U LASERSKOM SISTEMU SA ERBIJUM-DOPIRANIM VLAKNOM CHAOS IN ERBIUM-DOPED FIBER LASER SYSTEMS

Vesna Rubežić, Ana Jovanović, *Elektrotehnički fakultet u Podgorici*

Sadržaj - U ovom radu razmatrano je haotično ponašanje koje se pod određenim uslovima može ostvariti u laserskom sistemu sa erbijum-dopiranim vlaknom. U sistemu je primijenjen elektro-optički modulator za modulaciju gubitaka u rezonatorskoj šupljini. Korišćeni su sinusoidalni i pravougaoni signali za modulaciju. Generisanje haosa je ostvareno promjenom parametara signala za modulaciju. Rezultati simulacija prikazani su vremenskom domenu i faznom prostoru.

Abstract - In this paper chaotic behaviour that can appear under certain conditions in erbium-doped fiber laser is considered. Electro-optic modulator is applied for cavity loss modulation in the fiber loop. Sinusoidal and squared signals for modulation are used. Chaos is generated by the change of parameters of signals for modulation. The results of simulation are shown in time domain and in time-varying phase space.

Ključne riječi: Ring laser sa erbijum-dopiranim vlaknom (EDFRL), haos

1. UVOD

Ring laser sa erbijum-dopiranim vlaknom (EDFRL) kao interesantan optički uređaj za generisanje optičkog haosa razmatran je u mnogo eksperimentalnih i teorijskih radova [1]-[4]. EDFRL je sistem sa tri energijska nivoa i pripada laserima klase B. Električno polje i populaciona inverzija ove klase lasera grade dinamičke jednačine date u [5], [6]. Uslov da bi se ova klasa lasera ispoljavala kao haotični sistem je još jedna perturbacija koja mora biti dodata njegovim jednačinama [7]. U literaturi je razmatrano više načina generisanja haotične dinamike u EDFRL-u. Jedna grupa autora generisanje haotične dinamike zasniva na uvođenju ion-pair modela [1], a druga grupa na uvođenju modulacije pojedinih parametara [3].

Haotične oscilacije koje se mogu generisati u EDFRL-u omogućavaju atraktivan način eksploatacije propusnog opsega dostupnog u optičkim komunikacionim sistemima. Sinhronizacija haotičnog predajnika i prijemnika (Pecora i Carroll [8]) otvorila je put upotrebi haosa u sigurnim komunikacijama u električnom i optičkom domenu [9]. Razvijene su tehnike za modulisanje informacija u širokopojasne haotične oscilacije, prenos signala duž optičkog kabla i demodulaciju informacije u prijemniku. Dobro opisane tehnike u sigurnim haotičnim komunikacijama su message masking, message modulation i chaos shift keying [9-14].

Tema ovog rada je generisanje haosa u EDFRL-u koji ima ulogu predajnika u jednoj chaotic masking komunikacionoj šemi i određivanje opsega bifurkacionog parametra za koji se dobijaju haotična rješenja odnosno haotični izlazi.

Ovaj rad je organizovan na sledeći način. Nakon uvoda, u drugoj glavi su date osnove nelinearnih optičkih fenomena. U trećoj glavi je opisan EDFRL haotični sistem. Numerički rezultati su komentarisani u četvrtoj glavi. Zatim su izloženi zaključci.

2. NELINEARNI OPTIČKI EFEKTI

Nelinearni optički fenomeni nastaju usled zavisnosti optičkih osobina materijala od jačine elektromagnetnog polja svjetlosti koje se kroz taj materijal prostire. Za njih ne važi princip superpozicije, što znači da se različiti optički talasi ne prostiru nezavisno kroz nelinearne optičke materijale, već se mora uzeti u obzir njihovo međusobno miješanje i generacija svjetlosti na drugim frekvencijama.

Nelinearnost se modeluje nelinearnom zavisnošću vektora polarizacije materijala \vec{P} od vektora električnog polja svjetlosti \vec{E} i opisuje se u opštoj formi kao [15]:

$$\vec{P} = \epsilon_0 \left(\chi^{(1)} \cdot \vec{E} + \chi^{(2)} \cdot \vec{E} \cdot \vec{E} + \chi^{(3)} \cdot \vec{E} \cdot \vec{E} \cdot \vec{E} + \dots \right) \quad (1)$$

gdje je ϵ_0 permitivnost u vakuumu, a $\chi^{(i)}$ susceptibilnost i -tog reda. Susceptibilnosti su u opštem slučaju tenzorske veličine. U linearizovanim aproksimacijama se jedino uzima u obzir susceptibilnost prvog reda. Susceptibilnosti višeg reda daju različite nelinearne efekte. Jedan od najinteresantnijih efekata je nastanak haosa u optičkim sistemima [16].

3. MODEL LASERSKOG SISTEMA SA ERBIJUM-DOPIRANIM VLAKNOM

Laser sa erbijum-dopiranim vlaknom kao tro-nivoski energijski sistem ima osnovno stanje, jedno metastabilno stanje i jedno pumpom pobuđeno stanje. Opadanje polarizacionih promjenljivih je reda $10^{12} s^{-1}$. Prema tome, polarizacione promjenljive mogu biti adiabatski eliminisane i ovaj laser može biti razmatran kao laser klase B. Populacija u pumpom pobuđenom stanju je zanemarljiva zbog brze neradijativne tranzicije u metastabilno stanje. Na osnovu prethodno navedenog uproštavanjem standardne laserske jednačine [5], [6] dobijene su sledeće jednačine za lasing polje E_L i populacionu inverziju D lasera sa erbijum-dopiranim vlaknom.

$$\begin{aligned} \dot{E}_L &= -kE_L + gE_LD + \xi \\ \dot{D} &= -\frac{1}{\tau}[(1 + I_p + E_L^2)D - I_p + 1], \end{aligned} \quad (2)$$

gdje "." označava derivaciju, E_L lasing polje, D gustina populacione inverzije, τ je vrijeme opadanja meta-stabilnog stanja za erbijum. k i g su, respektivno, koeficijent gubitaka (loss) i koeficijent pojačanja lasing polja. ξ predstavlja spontano-emisioni proces male amplitude slučajne fluktuacije. I_p je snaga pumpe lasera.

Na osnovu relacije (2) slijedi da je EDFRL nelinearni sistem drugog reda. Dinamika sistema opisana je sa dvije jednačine. Jednačine lasera sa dva stepena slobode zahtijevaju dodatnu perturbaciju u vidu optičke povratne sprege sa kašnjenjem [17], spoljašnje injekcije signala [18] ili povratne sprege sa faznom konjugacijom [19] da bi se generisao kaos, jer su prema Poincare-Bendixon-ovoj teoremi [7] neophodna najmanje tri stepena slobode da bi se u dinamičkom sistemu ispoljio kaos.

EDFRL je disipativni sistem i gubici u ovom sistemu su odgovorni za apsorpciju energije tokom lasing akcije. U ovom radu je korišćena modulacija gubitaka u rezonatorskoj šupljini da bi se proizveo haotični izlaz iz sistema. Ovo je postignuto umetanjem elektro-optičkog modulatora u fiber loop. U ovom slučaju koeficijent gubitaka polja je izražen kao:

$$k = k_0(1 + m \sin \omega t), \quad (3)$$

gdje je m modulacioni indeks, a ω ugaona modulaciona frekvencija.

Laser sa erbijum-dopiranim vlaknom i modulisanim gubicima, opisan jednačinama (2) i (3) je osnov za dobijanje predajnika u komunikacionim šemama sa haotičnim signalima.

Ovdje će biti opisan predajnik u message-masking šemi koji je prikazan na slici 1. Glavni dio predajnika je fiber ring laser formiran od sprežnjaka a_2 i jednog segmenta erbijumom-dopirang vlakna. U krug koji formira vlakno umetnut je elektro-optički modulator. Drugi sprežnjak a_3 je uveden u prijemnik da bi dobili haotično polje E_L . Signal poruke S_m je miješan sa E_L kroz sprežnjake a_1 i a_4 . Takođe, on utiče na signal u krugu A kroz sprežnjake a_1 i a_2 . Kombinovani signal za prenos pogodnim izborom sprežnih koeficijenata $a_i, i=1,2,3,4$ se dobija u formi $\propto |E_L + S_m|^2$ [3]. Ovaj signal se prenosi prema prijemniku.

Ponašanje predajnika opisano je, dakle, sledećim jednačinama:

$$\begin{aligned} \dot{E}_L &= -k(E_L - cS_m) + gE_LD + \xi \\ \dot{D} &= -\frac{1}{\tau}[(1 + I_p + E_L^2)D - I_p + 1] \end{aligned} \quad (4)$$

$$k = k_0(1 + m \sin \omega t)$$

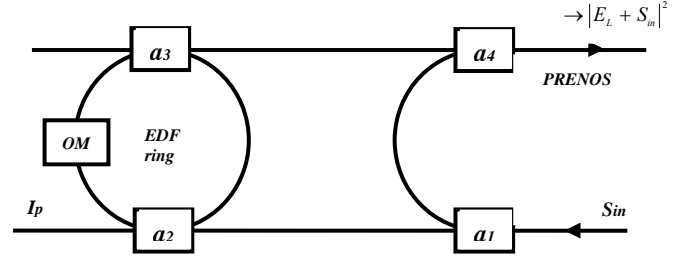
$$S_m = S_0(1 - \sin \omega_s t),$$

gdje su S_0 amplituda poruke i ω_s frekvencija poruke.

Parametri rezonatorske šupljine su optimizovani za

dobijanje haosa. Dobijeno polje E_L je serija pozitivnih haotičnih impulsa u fiksiranim intervalima.

Populaciona inverzija D raste kad je polje E_L jednako nuli, zato što ona nastaje zahvaljujući energiji pumpe koja ulazi u rezonatorsku šupljinu kontinualno. D brzo opada tokom formiranja impulsa, zato što je stimulisana emisija fotona praćena prelaskom elektrona sa višeg na niži energijski nivo.



Sl. 1. Predajnik sa EDFRL-om

4. REZULTATI SIMULACIJA

U našim simulacijama mijenja se modulacioni indeks m u opsegu od 0 do 1, dok ostali parametri imaju vrijednost kao u tabeli I.

Postojanje haosa je kvalitativno verifikovano direktnim opservacionim metodom koji je zasnovan na faznim portretima u faznom prostoru [7]. U slučaju periodičnog kretanja u faznom prostoru se zapaža zatvorena trajektorija. U slučaju haosa u faznom prostoru se vide čudni atraktori sa nezatvorenom trajektorijom.

Parametar	Simbol	Vrijednost
Vrijeme opadanja meta-stabilnog stanja	τ	10 ms
Spontano-emisioni faktor	ξ	10^{-4}
Snaga pumpe	I_p	10 mW
Modulacioni indeks	m	0-1
Brzina opadanja	k_0	$3.3 \cdot 10^7$
Pojačanje	g	$2k_0$
Amplituda poruke	S_0	1
Modulaciona frekvencija	ω	$3.5 \cdot 10^5$
Frekvencija poruke	ω_s	$3.14 \cdot 10^5$
Predstavlja dio signala uveden kroz sprežnjake	c	0-1

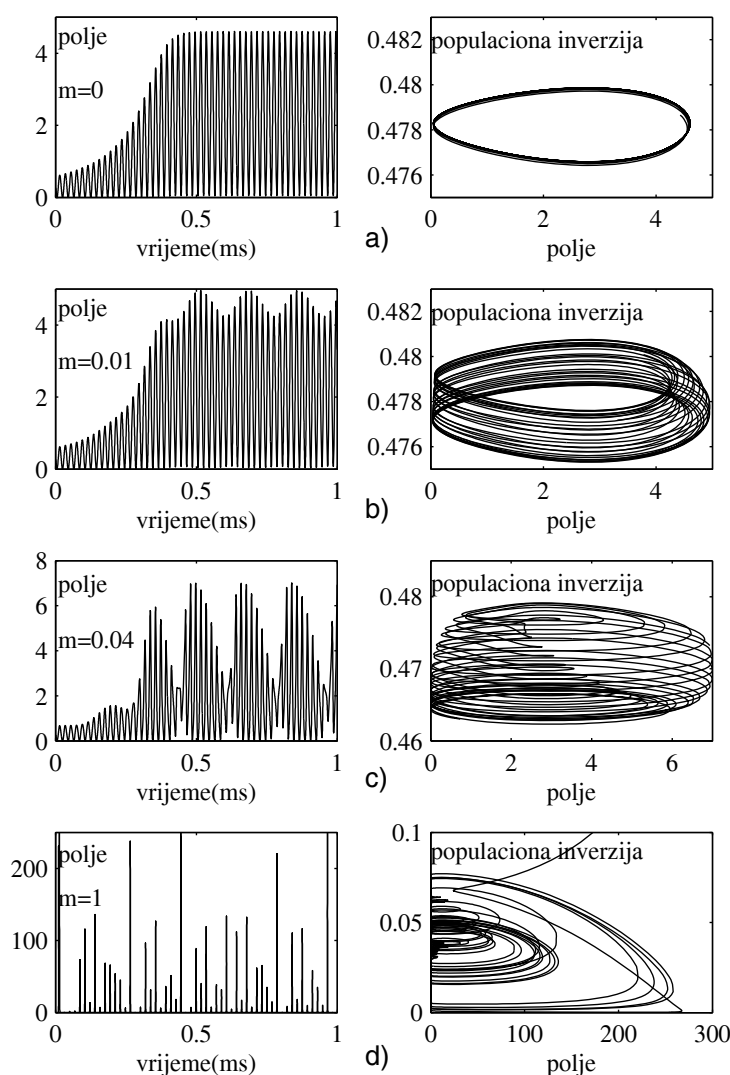
Tabela. I. Vrijednosti parametara u EDFRL-u

Modulacioni indeks m je povećavan od 0 do 1 i izlazi iz predajnika su prikazani na slici 2. Za $m=0$ izlaz je sinusoidalan, kao što se vidi na slici 2a) sa frekvencijom jednakom frekvenciji poruke ω_s . Za $m=0.01$ izlaz postaje kvaziperiodičan i prikazan je na slici 2b). Povećavajući m prema 0.02 i dalje izlazi od kvaziperiodičnih prelaze u haotične. Na slici 2c) prikazan je haotični signal dobijen za $m=0.04$. Daljim povećanjem modulacionog indeksa ka

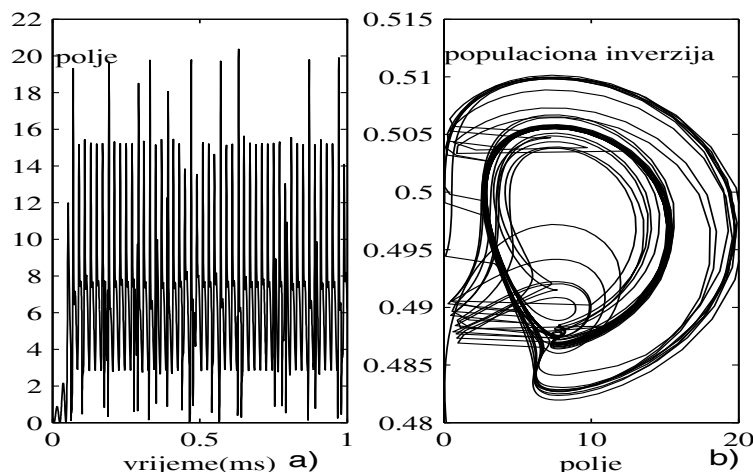
vrijednosti 1 (slika 2d)), signal ostaje haotičan i amplitude haotičnih impulsa povećavaju se 30 puta. Dakle, modulacioni indeks m je kontrolni parametar čijim rastom sistem prelazi iz periodičnog preko kvaziperiodičnog u haotično stanje. Ovo je kvaziperiodični scenario prelaza u kaos. Zapaženo je, takođe, da se dinamički opseg haotičnih impulsa povećava sa povećanjem modulacionog indeksa. Međutim, ovo generiše više impulsa sa amplitudom nula i povećava se razmak između haotičnih impulsa što može imati negativan uticaj na maskiranje signala.

Modulacija gubitaka u vlaknu može biti ostvarena signalom koji nije čisto sinusoidalni. U našim simulacijama koristili smo i pravougaoni periodični signal za modulaciju gubitaka: $k = k_0(1 + \text{square}(\omega t + \phi))$. Na ovaj način

dobijaju se dva dodatna parametra koja se mogu mijenjati u cilju dobijanja složenijih haotičnih signala i većeg stepena sigurnosti haotičnih komunikacija. To su faktor ispune i faza pravougaonog periodičnog signala. Koristeći ovaj oblik modulacije i faktor ispune kao parametar koji se mijenja ostvaruje se intermitentni scenario prelaza u kaos. Intermitencije su naizmjenične periodično-haotične faze u dinamici sistema koje postoje za istu vrijednost kontrolnog parametra. Kako se sistem približava tački prelaza, intervali regularnog kretanja postaju sve kraći, a turbulentni udari sve duži, dok ne ostane samo kaos. Na slici 3a) prikazan je vremenski oblik polja koji odgovara scenariju intermitencija, a na slici 3b) odgovarajući fazni portret.



Sl.2. Generisanje haosa povećanjem indeksa modulacije m - kvaziperiodični scenario prelaza u kaos.



Sl.3. Rezultati simulacija za intermitentni scenario prelaza u kaos.

5. ZAKLJUČAK

U ovom radu razmatrano je generisanje haosa kroz modulaciju gubitaka i zavisnost haotične dinamike u EDFRL-u od vrijednosti modulacionog indeksa. Rezultati simulacija prikazani su u vremenskom domenu i u faznom prostoru. Optički kaos stvoren u fiber laserima je odlična maska za skrivanje signala poruke zahvaljujući kompleksnosti talasnih oblika i dugovremenskoj nepredvidljivosti. U daljem radu razmatraćemo zavisnost haotične dinamike od ostalih parametara sistema. Takođe, mogu se razmatrati i drugi talasni oblici modulacionog signala i njihov efekat na stepen haosa, a time i sigurnost komunikacija.

LITERATURA

- [1] F. Sanchez, P.L. Boudec, P.L. Francois, G. Stephan, "Effects of ion pairs on the dynamics of erbium-doped fiber lasers," *Phys. Rev. A* 48, pp. 2220–2229, 1993.
- [2] H. D. I. Abarbanel, M. B. Kennel, M. Buhl, C. T. Lewis, "Chaotic Dynamics in Erbium-doped fiber ring lasers," *Phys. Rev. A*, vol. 60, no. 3, pp. 2360–2374, 1999.
- [3] L. G. Luo, P. L. Chu, "Optical secure communications with chaotic Erbium-doped fiber lasers," *Journal of Optical Society of America B*, vol. 15, no. 10, pp. 2524–2530, 1998.
- [4] L. G. Luo, P. L. Chu, T. Whitbread, R. F. Peng, "Experimental Observation of synchronisation of chaos in Erbium-doped fiber lasers," *Optical Communication*, vol. 176, no. 1-3, pp. 213–217, 2000.
- [5] L. G. Luo, P. L. Chu, , "Suppression of self-pulsing in an erbium-doped fiber lasers," *Optical Letters*, no. 22, pp. 1174–1176, 1997.
- [6] L. G. Luo, T. J. Tee, P. L. Chu, "Bistability of erbium-doped fiber lasers," *Optical Communications*, no. 146, pp. 151–157, 1998.
- [7] V. G. Ivančević, T. T. Ivančević, "Complex Nonlinearity: Chaos, Phase Transitions, Topology Change and Path," Springer Verlag Berlin Heidelberg, Germany, 2008.
- [8] L. M. Pecorra, T. L. Carroll, "Synchronisation in Chaotic Systems," *Phys. Rev. Lett.*, vol. 64, no. 8, pp. 821–824, 1990.
- [9] M. Cuomo, A. V. Oppenheim, "Circuit implementation of synchronized chaos with applications to communications," *Phys. Rev. Lett.*, vol. 71, no. 1, pp. 65–68, 1993.
- [10] H. Dedieu, M. P. Kennedy, M. Hasler, "Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits," *IEEE Trans. Circuits Syst.*, no. 40, pp. 634–642, 1993.
- [11] U. Parlitz, L. O. Chua, Lj. Kocarev, K. S. Halle, A. Shang, "Transmission of digital signals by chaotic synchronization," *Int. J. Bifurcation Chaos Appl. Sci. Eng*, no. 2, pp. 973–977, 1992.
- [12] P. Colet, R. Roy, "Digital communication with synchronized chaotic lasers," *Opt. Lett.*, no. 19, pp. 2056–2058, 1994.
- [13] G. D. Van Wiggeren, R. Roy, "Communications with chaotic lasers," *Science*, no. 279, pp. 1198–1200, 1998.
- [14] P. Stavroulakis, "Chaos applications in telecommunications," Taylor & Francis Group, 2006.
- [15] A. M. Kugushev, N. S. Golubeva, "Nonlinear electromagnetic processes," Mir Publishers, Moscow, 1979.
- [16] A. Marinčić "Optičke telekomunikacije," ETF Beograd, 1997.
- [17] J. Mork, B. Tromborg, J. Mark, "Chaos in Semiconductor Lasers with Optical Feedback: Theory and Experiment," *IEEE Journal of Quantum Electronics*, vol. 28, no. 1, pp. 93–108, 1992.
- [18] V. A. Lodi, A. Scire, M. Sorel, S. Donati, "Dynamic Behaviour and Locking of a Semiconductor Laser Subjected to External Injection," *IEEE Journal of Quantum Electronics*, vol. 34, no. 12, pp. 2350–2357, 1998.
- [19] V. A. Murakami, J. Ohtsubo, "Dynamics and Linear Stability Analysis in Semiconductor Lasers with Phase-Conjugate Feedback," *IEEE Journal of Quantum Electronics*, vol. 34, no. 10, pp. 1979–1986, 1998.

DISTRIBUIRANI SET-MEMBERSHIP NLMS ALGORITAM DISTRIBUTED SET-MEMBERSHIP NLMS ALGORITHM

Žarko Zečević, Igor Đurović, Božo Krstajić, *Elektrotehnički fakultet*

Sadržaj: U radu je opisan distribuirani adaptivni algoritam koji ima primjenu kod bežičnih senzorskih mreža. S ciljem optimizacije energije, umjesto standardnog LMS algoritma, predloženo je da se koristi Set-Membership normalizovani LMS algoritam. Takođe je dat predlog distribuiranog algoritma sa probabilističkom funkcijom aktivacije senzorskih čvorova. Rezultati simulacija potvrđuju da se predloženim algoritmima vrši značajna ušteda energije.

Abstract: The paper describes the distributed adaptive algorithm which finds application in the wireless sensor networks. Aimed to save the energy, using the Set-Membership normalized LMS algorithm instead the LMS is proposed. Also, the distributed algorithm with probabilistic sensor activation is proposed. Simulation results confirm that the proposed algorithms achieve significant energy savings.

1. UVOD

Bežične Senzorske Mreže – BSM (Wireless Sensor Networks WSNs) su skup prostorno raspoređenih senzorskih čvorova čiji je cilj monitoring nekog fizičkog procesa od interesa (temperatura, vlažnost, zvuk, vibracije, pozicija mete, itd.) [1]. Koristeći mrežnu arhitekturu čvorovi prikupljene informacije šalju baznoj stanici. Generalno, senzorski čvor se sastoji od senzora koji vrši akviziciju, procesora koji vrši obradu podataka i RF transivera koji šalje ili prima obrađene podatke [2]. Senzorski čvorovi se napajaju baterijama i stoga oni imaju ograničen vijek trajanja.

Zadnjih godina problem optimizacije energije BSM-a razmatran je u mnogobrojnim radovima. S obzirom da se najviše energije troši na prenos podataka, predložene su razne tehnike za njegovu redukciju. Ove tehnike se mogu podijeliti u tri grupe: kompresija podataka, procesiranje u mreži i predikcija podataka [1-3]. Koristeći razne tehnike kodiranja podaci koje šalju čvorovi se kompresuju i samim tim se vrši ušteda energije [1-2]. Kod procesiranja u mreži se vrši agregacija podataka koji se na osnovu neke funkcije performanse obrađuju (minimum, maksimum, prosijek, itd.) i nakon toga šalju baznoj stanici. Ova metoda nije pogodna kod aplikacija koje zahtjevaju visoku tačnost mjerenja [2].

Predikcione tehnike obično podrazumjevaju modelovanje procesa koji se mjeri. Model procesa u svakom trenutku posjeduju svi čvorovi mreže, kao i bazna stanica [3-5]. Model u baznoj stanici se može koristiti za estimaciju mjerenih podataka, na taj način vršeći uštedu energije. Da bi se izbjegle veće greške u predikciji model u baznoj stanici se periodično ažurira. Najčešće se proces modeluje kao AR (autoregressive) ili ARMA (autoregressive moving average). Identifikacija parametara procesa se vrši pomoću nekog adaptivnog algoritma. Primjenu metoda najmanjih kvadrata (Least Mean Square - LMS) za redukciju podataka kod BSM prvi je predložio Sanitini u [3]. Prednost LMS algoritma, osim jednostavnosti, je i u tome što nije potrebno poznavati statistički model mjerenog signala.

Senzorske mreže se mogu podijeliti na centralizovane i distribuirane. U [4] je pokazano da distribuirane mreže imaju iste performanse kao i centralizovane, s tim što troše manje energije na prenos podataka. Predložene su razne topologije mreža i adaptivnih algoritama koje se primjenjuju kod njih. U [4] su predloženi Incremental LMS (ILMS) i Spatial LMS (SLMS) - distribuirani algoritmi namjenjeni za topologiju prsten kod koje svaki čvor komunicira sa jednim svojim susjedom u cikličnom maniru. Diffusion LMS (DLMS) je distribuirani algoritam namjenjen za topologije kod kojih svaki čvor komunicira sa više svojih susjeda [5]. ILMS i SLMS sporije konvergiraju, ali troše manje energije u odnosu na DLMS [5]. Postoje i probabilističke, vremenski promjenljive topologije, kod kojih čvorovi komuniciraju sa svojim susjedima na osnovu zadate funkcije vjerovatnoće [6].

Energija senzorskih čvorova troši se i na akviziciju i obradu podataka. Duty cycling je pristup kod kojeg čvorovi periodično prelaze iz aktivnog u mod mirovanja. U modu mirovanja senzori ne vrše mjerenja i ne ostvaruju komunikaciju sa susjednim čvorovima, čime se dodatno vrši ušteda energije [7].

U radu je dat predlog distribuiranog algoritma koji nalazi primjenu kod BSM. U cilju uštede ograničene energije baterija predloženo je da se umjesto Distribuiranog LMS-a koristi Distribuirani Set-Membership NLMS algoritam, čime se značajno smanjuje komunikacija među čvorovima. Predložena je i procedura za prelazak čvorova u mod mirovanja, vršeći na taj način uštedu energije koja se troši na obradu podataka.

2. INCREMENTAL I DIFFUSION LMS

Mjereni signal u k -tom čvoru se može modelovati na sljedeći način [4]:

$$d_k(n) = \mathbf{W}^{*T} \mathbf{X}_k(n) + v_k(n), \quad (1)$$

gdje je $\mathbf{X}_k(n)$ regresivni vektor, $v_k(n)$ bijeli Gausov šum, a \mathbf{W}^* parametri koje treba estimirati.

Cilj distribuiranih algoritama je minimizacija globalne funkcije performanse [4, 5]:

$$J^{glob}(\mathbf{W}) = \sum_{k=1}^N E \left| \mathbf{d}_k(n) - \mathbf{X}_k(n)^T \mathbf{W} \right|^2. \quad (2)$$

Optimalno rješenje je dato jednačinom [4], [5]:

$$\mathbf{W}^o = \left(\sum_{k=1}^N \mathbf{R}_{x,k} \right)^{-1} \left(\sum_{k=1}^N \mathbf{R}_{dx,k} \right), \quad (3)$$

gdje je $\mathbf{R}_{du,k} = E \mathbf{X}_k^T(n) \mathbf{X}_k(n)$, a $\mathbf{R}_{dx,k} = E \mathbf{d}_k(n) \mathbf{X}_k(n)$. Ovo rješenje minimizuje srednju kvadratnu grešku (MSE) širom mreže.

Čvorovi Distribuiranog LMS algoritma međusobno razmjenjuju estimirane vektore u cilju efikasnije minimizacije globalne greške. Iterativna formula Distribuiranog LMS-a je:

$$\mathbf{W}_k(n) = \boldsymbol{\varphi}_k(n-1) + \mu_k (d_k(n) - \mathbf{X}_k^T(n) \boldsymbol{\varphi}_k(n-1)). \quad (4)$$

Senzori međusobno mogu biti povezani na više načina. Ukoliko je cilj postizanje brže konvergencije, poželjna je veća povezanost i razmjena informacija između čvorova (Diffusion LMS-DLMS [4]). Kod DLMS-a vektor $\boldsymbol{\varphi}_{k-1}(n)$ predstavlja kombinaciju adaptivnih vektora koji surađuju sa čvorom k :

$$\boldsymbol{\varphi}_k(n-1) = \sum_{l \in N_k} a_{l,k}(n) \mathbf{W}_l(n-1), \quad (5)$$

pri čemu mora biti ispunjen uslov:

$$\sum_{l \in N_k} a_{l,k} = 1$$

da bi se obezbjedila konvergencija ka optimalnom rješenju. Koeficijenti $a_{l,k}$ mogu biti fiksni, a i adaptivni [5].

Ukoliko je cilj da se uštedi energija koja se troši na komunikaciju, onda je poželjnija topologija kod koje svaki senzor komunicira sa jednim svojim susjedom u cikličnom maniru (Incremental LMS [6]). U tom slučaju vektor $\boldsymbol{\varphi}_k(n-1)$ je jednak adaptivnom vektoru čvora $k-1$:

$$\boldsymbol{\varphi}_k(n-1) = \mathbf{W}_{k-1}(n-1). \quad (6)$$

Postoji i Probabilistički Difuzni LMS (PDLMS) kod kojeg se topologija mreže mijenja u svakoj iteraciji, tj. u svakoj iteraciji čvor k slučajno bira čvorove sa kojima će ostvariti komunikaciju [7].

3. DISTRIBUIRANI SET-MEMBERSHIP LMS ALGORITAM

Senzori u određenim periodima estimiraju parametre mjerenog procesa i o tome obavješavaju baznu stanicu. Bazna stanica na osnovu dostavljnog modela vrši predikciju procesa i donosi odluku o sistemu kojim upravlja.

Većina procesa (vlažnost zemljišta, temperatura) se mogu modelovati kao AR proces [2]:

$$d_k(n) = \sum_{i=1}^N W_i d_k(n-i) + v_k(n) \quad (7)$$

gdje je $d_k(n)$ izmjereni signal, a $v_k(n)$ bijeli Gausov šum.

Ako se regresivni vektor definiše kao:

$$\mathbf{X}_k(n) = [d_k(n-1), d_k(n-2), \dots, d_k(n-M+1)], \quad (8)$$

Distribuirani LMS se može koristiti za estimaciju procesa:

$$\mathbf{W}_k(n) = \boldsymbol{\varphi}_k(n-1) + \mu_k (d_k(n) - \mathbf{X}_k^T(n) \boldsymbol{\varphi}_k(n-1)). \quad (9)$$

U cilju uštede energije, komunikaciju između senzora i ažuriranje adaptivnog vektora treba prekinuti u onom trenutku kada razlika između izmjerene i predviđene vrijednosti padne ispod nekog određenog praga. Kada greška pređe prag treba opet započeti estimaciju i dostaviti novo rješenje glavnom čvoru.

Ako se prethodno opisana logika primjeni na Distribuirani LMS algoritam doći će do degradacije performansi kako u smislu brzine konvergencije, tako i u smislu srednje kvadratne greške u stacionarnom stanju. Bolje performanse se mogu postići ukoliko se umjesto LMS-a koristi Normalizovni LMS algoritam (NLMS). Ovaj algoritam je u literaturi prepoznatljiv kao Set-Membership NLMS (SMLMS) [8]. Iterativna formula Distribuiranog SMLMS-a (DSMLMS) bi bila:

$$\mathbf{W}_k(n) = \begin{cases} \boldsymbol{\varphi}_{k-1}(n) + \mu_k(n) \frac{e_k(n)}{(\mathbf{X}_k(n)^T \mathbf{X}_k(n))} & \text{ako je } |e_k(n)| > \gamma \\ \boldsymbol{\varphi}_{k-1}(n), & \text{ako je } |e_k(n)| < \gamma \end{cases} \quad (10)$$

gdje je $e_k(n)$ signal greške k -og čvora definisan na sljedeći način:

$$e_k(n) = d_k(n) - \boldsymbol{\varphi}_{k-1}(n)^T \mathbf{X}_k(n), \quad (11)$$

a $\mu_k(n)$ promjenljivi korak k -og čvora:

$$\mu_k(n) = 2 - \frac{\gamma_k}{|e_k(n)|} \quad (12)$$

Kada je $|e_k(n-1)| < \gamma_k$ čvor ne ažurira adaptivni vektor, ne šalje i ne prima podatke. Sa druge strane u svakom periodu T (iteraciji) čvor uzima mjerenja i računa grešku $e_k(n) = d_k(n) - \mathbf{X}_k(n)^T \boldsymbol{\varphi}_k(n-1)$ na osnovu estimiranog modela. Parametar γ_k se postavlja na sljedeću vrijednost [8]:

$$\gamma_k = \sqrt{5} \sigma_{v,k}, \quad (13)$$

gdje je $\sigma_{v,k}$ efektivna vrijednost aditivnog šuma k -og čvora.

4. PROBABILISTIČKI DSMLMS

Dodatna ušteda energije se može postići ukoliko se sensorima omogući prelazak u mod mirovanja sa nekom određenom vjerovatnoćom. U modu mirovanja senzor vrši mjerenja, ali ne komunicira sa ostalim sensorima, ne ažurira adaptivni vektor, niti računa signal greške $e_k(n)$. U modu mirovanja senzor mora vršiti mjerenja kako bi u trenutku aktivacije bio dostupan vektor $\mathbf{X}_k(n)$. Kada čvor pređe u aktivni mod, računa se signal greške i ako je on iznad specificiranog praga, ažurira se adaptivni vektor i distribuira ostalim aktivnim čvorovima.

Prelazak čvorova u sleep mode se vrši na osnovu neke funkcije raspodjele koja je čvoru poznata (uniformna raspodjela je pogodna). Vjerovatnoća aktivacije k -tog čvora može se iterativno ažurirati:

$$p_k(n) = \alpha p_k(n-1) + \beta \frac{A_k(n-1)}{n_k}, \quad (14)$$

gdje su α i β konstante koje čiji zbir mora biti manji od jedan, $A_k(n-1)$ broj čvorova koji se bili aktivni u prethodnoj iteraciji, a n_k ukupan broj čvorova sa kojima čvor k može da komunicira. Ideja je da se vjerovatnoća aktivacije povećava sa povećanjem broja čvorova koji su aktivni, i obrnuto.

U cilju postizanja brže konvergencija i manje greške u stacionarnom stanju pogodno je koristiti korak:

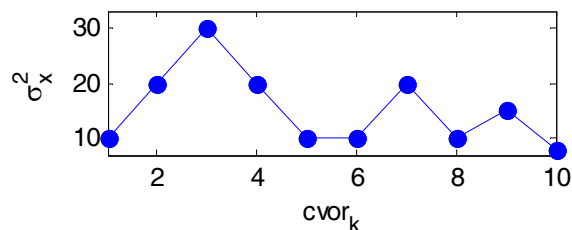
$$\mu_k(n) = 1 + p_k(n) - \frac{\gamma_k}{|e_k(n)|}. \quad (15)$$

Kada je vjerovatnoća aktivacije veća, veći je i korak algoritma čime se ubzava konvergencija. Sa druge strane, niska vjerovatnoća aktivacije indicira da je algoritam konvergirao. U tom slučaju korak je manji, pa će i greška u stacionarnom stanju biti manja.

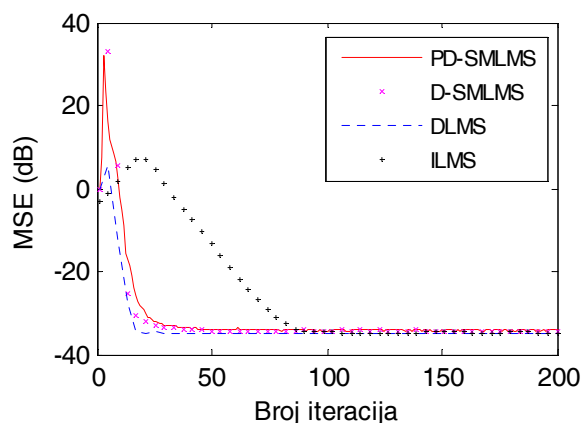
5. REZULTATI SIMULACIJA

Performanse DLMS-a, probabilističkog DSMLMS-a (PDSMLMS), DLMS-a i ILMS-a su poređene na primjeru mreže koja ima 10 čvorova. Radi jednostavnosti je uzeto da

svi čvorovi mogu međusobno da komuniciraju i razmjenjuju estimacije. Mjereni signal svakog čvora je zašumljen bijelim Gausovim šumom čija je varijansa prikazana sa Slici 1.



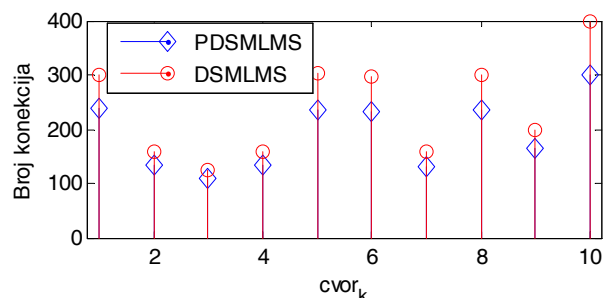
Slika 1) Varijansa aditivnog šuma ($\times 10^{-4}$)



Slika 2. MSE razmatranih algoritama

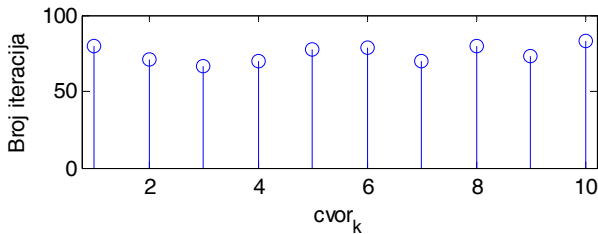
Na slici 2. prikazana je globalna srednja kvadratna greška (MSE) u dB. Za svaki od algoritama je korišćen korak koji obezbjeđuje najbržu konvergenciju. Može se uočiti da PDSMLMS neznatno sporije konvergira od DSMLMS-a i DLMS-a. Sa druge strane ILMS značajno sporije konvergira od razmatranih algoritama. U stacionarnom stanju svaki od algoritama ima približno istu vrijednost srednje kvadratne greške.

Na slici 3 je prikazan broj ostvarenih razmjena podataka u toku 200 iteracija, za svaki čvor mreže pojedinačno. Kod DLMS-a svaki čvor u svakoj iteraciji ostvari komunikaciju sa ostalih 9 čvorova (ukupno 1800 razmjena). Sa druge strane PDSMLMS (DSMLMS) u prosjeku uspostavi 192 (239) konekcije čime je uštedeno 90% (85%) energije koja se troši na komunikaciju sa ostalim čvorovima. ILMS uspostavlja jednu konekciju po iteraciji (200 ukupno).

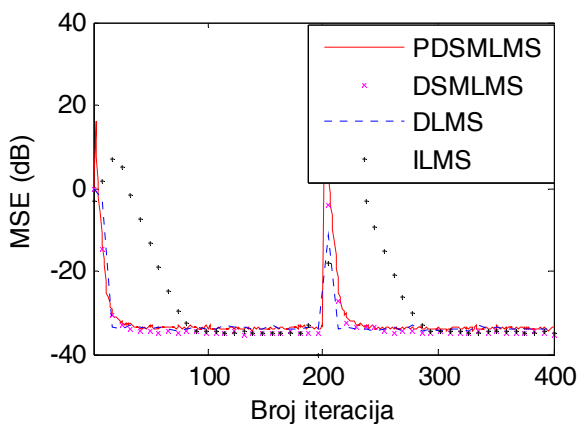


Slika 3. Broj ostvarenih komunikacija

Na slici 4. je prikazan broj iteracija u kojima su čvorovi PDSMLMS-a bili aktivni. Kod ILMS-a, DLMS-a i DSMLMS-a čvorovi su bili aktivni svih 200 iteracija, dok je kod PDSMLMS-a u prosjeku uštedeno 65% energije koja se troši na simulaciju modela (računanje signala greške).



Slika 4. Broj iteracija u kome su čvorovi bili aktivni



Slika 5. MSE razmatranih algoritama

Razmotren je i scenario u kome dolazi do nagle promjene parametara procesa. Sa slike 5 se može uočiti da se predloženi algoritmi brzo adaptiraju na novonastale promjene. Što se tiče uštede energije ona je približno ista kao u prethodnom primjeru, stoga nije dat grafički prikaz podataka (91% uštede energije koja se troši na komunikaciju i oko 70% uštede energije koja se troši na simulaciju modela).

6. ZAKLJUČAK

U radu je prezentovan distribuirani adaptivni algoritam koji može naći primjenu kod BSM. U cilju optimizacije energije predloženo je da se umjesto Distribuiranog LMS-a koristi Distribuirani Set-Membership NLMS algoritam (DSMLMS), čime se postiže 85-90% uštede energije koja se

troši na komunikaciju među čvorovima. Dodatno je predložen probabilistički DSMLMS algoritam kod kojeg čvorovi mreže prelaze u mod mirovanja na osnovu funkcije vjerovatnoće koja se iterativno ažurira. Na taj način se uštedi 65-70% energije koja se troši na simulaciju modela. PDSMLMS i DSMLMS imaju sličnu brzinu konvergencije DLMS-u, a znatno brže konvergiraju u odnosu na ILMS.

LITERATURA

- [1] B. Stojkoska, D. Solev, D. Davcev, "Data Prediction in WSN using Variable Step Size LMS Algorithm", *SENSORCOMM*, 2011
- [2] Y. L. Borgne, S. Santini, G. Bontempi, "Adaptive model selection for time series prediction in wireless sensor networks," *Signal Processing*, vol.87 n.12, pp. 3010-3020, December, 2007.
- [3] S. Santini, K. Römer, "An Adaptive Strategy for Quality-Based Data Reduction in Wireless Sensor Networks," *Proceedings of the 3rd International Conference on Networked Sensing Systems (INSS'06)*
- [4] G. Lopes, A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, Jul. 2008
- [5] N. Takahashi, I. Yamada, A. H. Sayed, "Diffusion Least-Mean Squares With Adaptive Combiners: Formulation and Performance Analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, Sep. 2010
- [6] F. S. Cattivelli, A. H. Sayed, "Analysis of Spatial and Incremental LMS Processing for Distributed Estimation," *Signal Processing, IEEE Transactions on*, vol.59, no.4, pp.1465,1480, April 2011
- [7] N. Takahashi, I. Yamada, "Link probability control for probabilistic diffusion least-mean squares over resource-constrained networks," *Acoustics Speech and Signal Processing (ICASSP)*, 2010
- [8] S. Gollamudi, S. Nagaraj, S. Kapoor, Y. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *Signal Processing Letters, IEEE*, vol.5, no.5, pp.111,114, May 1998

BRZI COUPLED LMS ALGORITAM FAST COUPLED LMS ALGORITAM

Žarko Zečević, Božo Krstajić, *Elektrotehnički fakultet*

Sadržaj: U radu je dat predlog smanjenja računске složenosti Coupled LMS (CLMS) algoritma. Umjesto u svakoj iteraciji koeficijenti adaptivnih filtara se ažuriraju periodično. Sa druge strane estimacija gradijenta se vrši u svakom vremenskom trenutku. Analiza predloženog algoritma i rezultati simulacija pokazuju da predloženi algoritam ima iste performanse kao i originalni algoritam, dok je računska složenost značajno smanjena.

Abstract: In this paper reducing computational complexity of the Coupled LMS (CLMS) algorithm is proposed. Instead in each iteration, the adaptive filter coefficients are updated periodical., On the other hand, the gradient is estimated in each iteration. Analysis of the proposed algorithm shows that the proposed algorithm exhibits the same performances as the original one, while the computational complexity is significantly reduced.

1. UVOD

Filtered-x Least Mean Square (FxLMS) je modifikacija LMS algoritma koja najčešće nalazi primjenu u Active Noise Canceler (ANC) sistemima [1]. Kod ANC sistema brzina konvergencije jedan je od ključnih kriterijuma performansi. Brzina konvergencije FxLMS-a zavisi od raspona sopstvenih vrijednosti autokorelacione matrice filtriranog ulaznog signala [1], [2]. Što je ovaj odnos bliži jedinici algoritam će brže konvergirati. Poznato je da LMS algoritam najbrže konvergira za bijele šumove [3]. Međutim, ako na ulazu FxLMS-a dolazi bijeli šum filter sekundarne putanje (blok S na sl. 1) će ga korelisati. Drugim riječima, raspon sopstvenih vrijednosti će se povećati, dok će se brzina konvergencije smanjiti [1]. Karakteristike konvergencije, kao i uticaj grešaka u modelovanju sekundarne putanje na stabilnost FxLMS-a analizirane su u [4] - [6].

Greška u stacionarnom stanju FxLMS-a je drugi bitan kriterijum za ocjenu kvaliteta rada ANC sistema. Ona zavisi od niza faktora: koraka algoritma, minimalne greške sistema, reda adaptivnog filtra, reda sekundarne putanje, grešaka u estimaciji sekundarne putanje [4]. Kod FxLMS algoritma uvijek postoji višak srednje kvadratne greške, bez obzira na to da li je željeni signal zašumljen [4].

Predložene su brojne modifikacije FxLMS-a čiji je cilj povećanje brzine konvergencije. Jedna od njih, opisana u [7], zasniva se na modifikovanom blok dijagramu FxLMS-a (MFxLMS). Self Orthogonalizing FxLMS algoritam (SOFxLMS), opisan u [8], povećava brzinu konvergencije u uslovima bijelog šuma. Računski jednostavnija varijanta ovog algoritma prezentovana je u [9].

U [10] je predložen novi algoritam za adaptaciju ANC sistema. Predloženi pristup podrazumjeva korišćenje dva adaptivna filtra (CLMS – Coupled LMS). Jedan filter služi za identifikaciju nepoznatog sistema pomoću LMS algoritma. Istovremeno, na osnovu identifikovanih koeficijenta i

koeficijenta sekundarne putanje računaju se koeficijenti drugog adaptivnog filtra s ciljem minimizacije izlazne greške sistema. CLMS ima bržu konvergenciju i manju grešku u ustaljenom režimu od FxLMS-a i njegovih modifikacija.

Poboljšanje performansi ANC sistema CLMS algoritmom postignuto je na račun povećanja računске složenosti. U ovom radu je opisan postupak za smanjenje složenosti CLMS-a. Umjesto u svakoj, predloženo je da se ažuriranje adaptivnih filtara vrši u svakoj L -toj iteraciji. Na ovaj način složenost predloženog algoritma se svodi na red složenosti FxLMS-a, pri čemu performanse originalnog algoritma nijesu degradirane.

Rad je organizovan na sljedeći način. U drugom dijelu je dat kratak opis FxLMS-a i njegovih modifikacija, nakon čega je u trećem dijelu prezentovan predlog smanjenja složenosti CLMS algoritma. U četvrtom dijelu je analizirana računska složenost razmatranih algoritama. Na kraju su dati rezultati simulacija i zaključak, respektivno.

2. FXLMS I NJEGOVE MODIFIKACIJE

Blok dijagram FxLMS-a je prikazan na na sl. 1, gdje je $S(z)$ prenosna funkcija sekundarne putanje, $\hat{S}(z)$ estimacija sekundarne putanje, $P(z)$ nepoznati sistem, a $W(z)$ adaptivni kontroler.

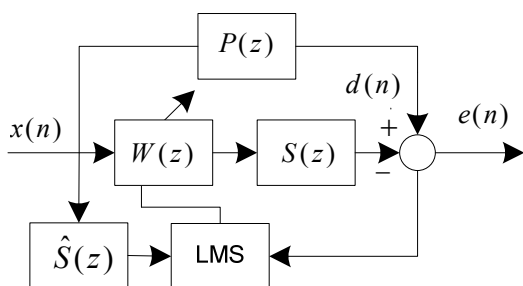
FxLMS algoritam minimizuje signal greške podešavanjem koeficijenata adaptivnog filtra, nakon kojeg slijedi filter sekundarne putanje, (sl. 1). Signal greške se definiše na sljedeći način:

$$e(n) = d(n) - \mathbf{W}(n)^T \mathbf{X}_s(n) \quad (1)$$

gdje je $d(n)$ željeni signal, $\mathbf{X}_s(n)$ vektor filtriranog ulaznog signala, a $\mathbf{W}(n)$ adaptivni težinski vektor (sl. 1). Optimalno rješenje koje minimizuje signal greške je:

$$\mathbf{W}^* = \mathbf{R}_{ss}^{-1} \mathbf{p}_s. \quad (2)$$

Autokorelaciona matrica filtriranog ulaznog signala je označena sa \mathbf{R}_{ss} , a kroskorelacioni vektor između filtriranog ulaznog signala i željenog signala sa \mathbf{p}_s .



Slika 1. Blok dijagram FxLMS-a

S ciljem da minimizuje grešku, FxLMS iterativno ažurira koeficijente adaptivnog filtra koristeći sljedeću iterativnu formulu:

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu e(n) \mathbf{X}_s(n). \quad (3)$$

Korak algoritma je označen sa μ , a sa $\mathbf{X}_s(n)$ estimacija vektora filtriranog ulaznog signala. FxLMS u stacionarnom stanju dostiže rješenje:

$$\mathbf{W}(\infty) = \mathbf{R}_{ss}^{-1} \mathbf{p}_s, \quad (4)$$

gdje je \mathbf{R}_{ss} kroskorelaciona matrica između estimiranog i filtriranog ulaznog signala, a \mathbf{p}_s kroskorelacioni vektor između estimiranog filtriranog ulaznog signala i željenog signala.

Iz (3) se izvodi maksimalni korak algoritma [2]:

$$\mu_{\max} = \frac{1}{N \bar{x}_s^2}, \quad (5)$$

gdje je N red adaptivnog filtra, a \bar{x}_s^2 srednja snaga filtriranog ulaznog signala.

SOFxLMS algoritam, predložen u [8], koristi sljedeću iterativnu formulu:

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu \mathbf{C} \mathbf{X}_s(n) e(n), \quad (6)$$

gdje je \mathbf{C} inverzna autokorelaciona matrica ulaznog signala (bijeli šum) koja se određuje na početku rada algoritma:

$$\mathbf{C} = \begin{bmatrix} \sum_{i=0}^{M-1} S_i^2 & \sum_{i=0}^{M-2} S_i S_{i+1} & \dots & \sum_{i=0}^{M-N} S_i S_{i+N-1} \\ \sum_{i=0}^{M-2} S_i S_{i+1} & \sum_{i=0}^{M-1} S_i^2 & \dots & \sum_{i=0}^{M-N+1} S_i S_{i+N-2} \\ \dots & \dots & \dots & \dots \\ \sum_{i=0}^{M-N} S_i S_{i+N-1} & \sum_{i=0}^{M-N+1} S_i S_{i+N-2} & \dots & \sum_{i=0}^{M-1} S_i^2 \end{bmatrix}^{-1}. \quad (7)$$

Sa M je označen red sekundarne putanje. SOFxLMS algoritam za bijele šumove ima brzinu konvergencije sličnu LMS-u [8].

U [10] je predložen Coupled LMS algoritam (CLMS) koji se sastoji iz dva adaptivna filtra \mathbf{W}_p i \mathbf{W}_F . \mathbf{W}_p je LMS filtar dužine $N+M-1$ čiji je cilj identifikacija nepoznatog sistema \mathbf{P} , dok je \mathbf{W}_F adaptivni filtar čiji je cilj minimizacija izlazne greške sistema $e(n)$.

U cilju identifikacije nepoznatog sistema vrši se estimacija željenog signala:

$$\hat{d}(n) = e(n) + \mathbf{Y}_w^T(n) \hat{\mathbf{S}} = d(n) - \mathbf{Y}_w^T(n) (\mathbf{S} - \hat{\mathbf{S}}), \quad (8)$$

gdje je $\mathbf{Y}_w(n)$ vektor izlaznog signala iz adaptivnog filtra $\mathbf{W}_F(n)$ (dimenzija $N \times 1$).

Estimirani željeni signal $\hat{d}(n)$ se koristi za identifikaciju nepoznatog sistema \mathbf{P} koristeći LMS iterativnu formulu:

$$\mathbf{W}_p(n+1) = \mathbf{W}_p(n) + \mu \mathbf{X} e_m(n), \quad (9)$$

gdje je $e_m(n)$ modifikovani signal greške:

$$e_m(n) = \hat{d}(n) - \mathbf{W}_p^T \mathbf{X}. \quad (10)$$

Drugi filtar $\mathbf{W}_F(n)$ u stacionarnom stanju treba da konvergira ka optimalnom rješenju (2) i on se adaptira na sljedeći način [10]:

$$\mathbf{W}_F(n) = \hat{\mathbf{S}}_k^+ \mathbf{W}_p(n) \quad (11)$$

gdje je $\hat{\mathbf{S}}_k$ konvoluciona matrica definisana na sljedeći način:

$$\hat{\mathbf{S}}_k = \begin{bmatrix} \hat{S}_0 & \hat{S}_1 & \dots & \hat{S}_{M-1} & \dots & 0 \\ 0 & \hat{S}_0 & \dots & \dots & 0 & 0 \\ 0 & 0 & \hat{S}_0 & \dots & \hat{S}_{M-1} & 0 \\ 0 & \dots & \dots & \dots & \hat{S}_{M-2} & \hat{S}_{M-1} \end{bmatrix}_{N \times N+M-1}, \quad (12)$$

a + je oznaka za pseudoinverznu matricu.

3. BRZI CLMS ALGORITAM (FCLMS)

U cilju smanjenja računске složenosti CLMS-a predlaže se da se podaci procesiraju u vidu blokova. Umjesto u svakoj, koeficijenti adaptivnih filtara se ažuriraju u svakoj L -toj iteraciji. Sa druge strane u toku svih L trenutaka se vrši procjena gradijenta koristeći formulu:

$$\Phi(k) = \sum_{i=0}^{L-1} \mathbf{X}(nL+i) e_m(nL+i), \quad (13)$$

gdje je:

$$e_m(nL+i) = \hat{d}(nL+i) - \mathbf{X}(nL+i)^T \mathbf{W}_p(k). \quad (14)$$

Nakon svake L -te iteracije vrši se ažuriranje adaptivnog filtra koristeći jednačinu:

$$\mathbf{W}_p(k+1) = \mathbf{W}_p(k) + \mu \Phi(k), \quad (15)$$

a na osnovu njega se ažurira i drugi adaptivni filter:

$$\mathbf{W}_F(k+1) = \hat{\mathbf{S}}_k^+ \mathbf{W}_p(k+1). \quad (16)$$

Uvrštavanjem (13) i (14) u (15), pod uslovom da nema grešaka u modelovanju sekundarne putanje, dobija se:

$$\begin{aligned} \mathbf{W}_p(k+1) &= \mathbf{W}_p(k) + \mu \sum_{i=0}^{L-1} \mathbf{X}(nL+i) \\ &\quad \times d(nL+i) - \mathbf{X}(nL+i)^T \mathbf{W}_p(k) \\ &= \mathbf{W}_p^*(k) \left(\mathbf{I} - \frac{\mu}{L} \mathbf{R} \right) \end{aligned} \quad (17)$$

Autokorelaciona matrica ulaznog signala ja označena sa \mathbf{R} , dok je \mathbf{W}_p^* optimalno rješenje.

Ako se od (17) oduzme \mathbf{W}_p^* i nađe očekivana vrijednost dobija se:

$$E[\Delta \mathbf{W}_p(k+1)] = \left(\mathbf{I} - \frac{\mu}{L} \mathbf{R} \right) \mathbf{W}_p^*(k). \quad (18)$$

Na osnovu (18) se može zaključiti da će Brzi CLMS (FCLMS) konvergirati istom brzinom kao i CLMS. Maksimalni korak algoritma je dat relacijom:

$$\mu_{\max} < \frac{1}{LN\sigma_x^2}. \quad (19)$$

Ukoliko je red filtra veliki, u cilju dodatnog smanjenja složenosti, za računanje adaptivnog filtra $\mathbf{W}_F(n)$ se može koristiti Brza Furijeova Transformacija (FFT) [2]:

Za računanje izraza **Error! Reference source not found.** potrebno je $O(N \log_2 N)$ operacija množenja sa sabiranjima.

4. SLOŽENOST FCLMS-A

U svakoj L -toj iteraciji potrebno je izračunati izraze (15) i (16), jer je vektor $\mathbf{W}_p(k)$ potreban u narednih L iteracija za računanje izraza (13) i (14). Ovo zahtijeva računsku složenost reda $O(N^2)$, odnosno složenost reda $O(N^2/L)$ po iteraciji.

U cilju ravnomjerne raspodjele računskih operacija po iteraciji, računanje izraza (16) se može podijeliti na L iteracija. Na primjer, za $L=N+M-1$ množenje matrice $\hat{\mathbf{S}}_k^+$ i vektora $\mathbf{W}_p(k)$ se može raspodijeliti na N iteracija na taj način što će se u svakoj iteraciji množiti jedna kolona matrice $\hat{\mathbf{S}}_k^+$ sa vektorom $\mathbf{W}_p(k)$. Sa druge strane za računanje izraza (14) umjesto $\mathbf{W}_p(k)$ može se koristiti $\mathbf{W}_p(k-1)$. Ovo omogućava ravnomjernu upotrebu procesora, ali će prouzrokovati i neznatno sporiju konvergenciju.

Još jedna pogodnost ovakve modifikacije je u tome što se može koristiti paralelno procesiranje. Naime, kod CLMS-a se mora koristiti serijski procesor jer je prvo potrebno izračunati grešku $e(n)$, a zatim tu grešku iskoristiti za adaptaciju filtra. Kod predloženog algoritma ažuriranje koeficijenata se može izvršavati paralelno (nezavisno) sa računanjem gradijenta, što dodatno ubrzava rad algoritma.

U tabeli 1 dat je pregled broja operacija sabiranja i množenja koje su potrebne za izvršavanje FxLMS, SOFxLMS, CLMS i FCLMS algoritma (za $L=N+M-1$). Može se uočiti da svaki od algoritama ima složenost istog reda.

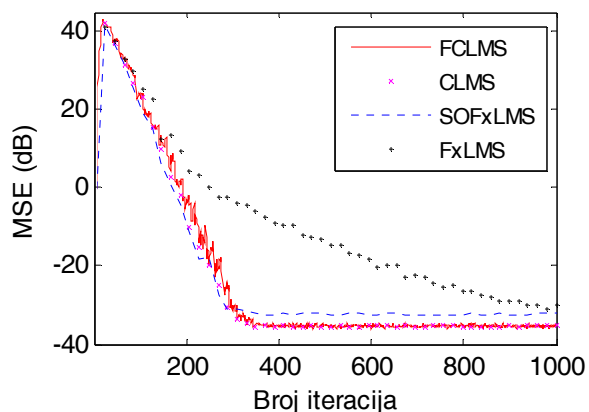
TABELA 1. RAČUNSKA SLOŽENOST RAZMATRANIH ALGORITAMA

Algoritam	+	*
FxLMS	$N+M-1$	$N+M+1$
SOFxLMS	$2N+M-2$	$2N+M$
CLMS	$N(N+M+1)+2M+1$	$N(N+M+1)+2M-2$
FCLMS	$2N+M$	$2N+M+1$

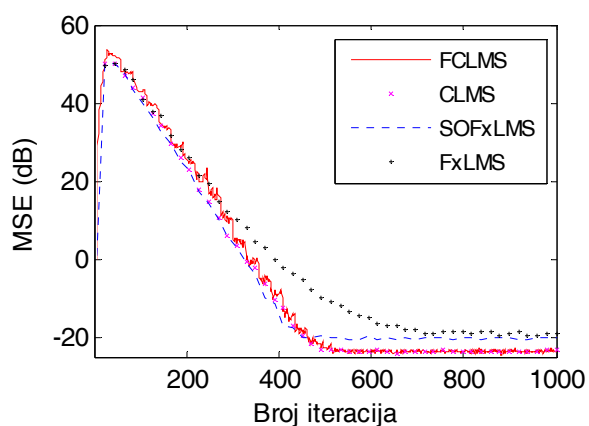
5. REZULTATI SIMULACIJA

Simulacijama je izvršena komparacija FCLMS, CLMS, SOFxLMS i FxLMS algoritma. Na graficima je prikazana srednja kvadratna greska (MSE) za svaki od algoritama.

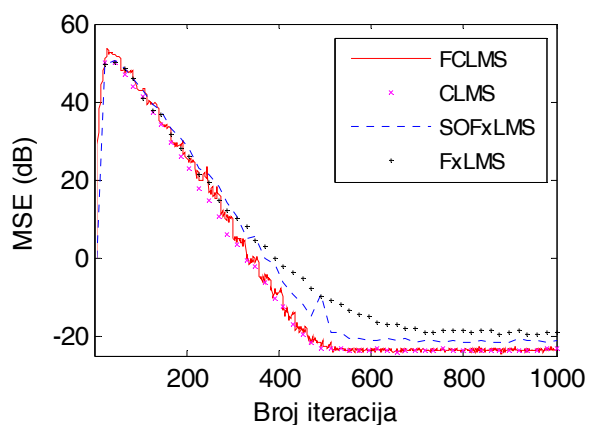
Na slici 2 su prikazani rezultati simulacija za slučaj kada je red adaptivnih filtara \mathbf{W}_F i \mathbf{W} $N = 20$, dok je red adaptivnog filtra \mathbf{W}_p 25. Sekundarna putanja je estimirana bez greške. Može se uočiti da CLMS i FCLMS imaju iste performanse. SOFxLMS ima približno istu brzinu konvergencije (F)CLMS-u, dok FxLMS algoritam nasprije konvergira. (F)CLMS ima manju grešku od stacionarnom stanju od SOFxLMS-a i FxLMS-a.



Slika 2. MSE razmatranih algoritama



Slika 3. MSE razmatranih algoritama



Slika 4. MSE razmatranih algoritama

Do sličnih zaključaka se može doći i u primjerima kada je željeni signal zašumljen, slika 3, i kada postoje greške u estimaciji sekundarne putanje, slika 4 ($N = 20$, $M = 5$). Treba naglasiti da u slučaju kada sekundarna putanja nije modelovana bez greške SOFxLMS sporije konvergira od (F)CLMS-a.

6. ZAKLJUČAK

U ovom radu je prezentovana modifikacija CLMS algoritma (FCLMS) koja ima manju računsku složenost. Analiza algoritma pokazuje da CLMS ima iste performanse kao i originalni algoritam, ali značajno manju složenost. Naime, u radu je pokazano da CLMS ima složenost istog reda kao FxLMS i SOFxLMS.

Rezultatima simulacija je pokazano da, u slučaju idealnog modelovanja sekundarne putanje, FCLMS ima sličnu brzinu konvergencije kao i SOFxLMS i značajno bržu konvergenciju od FxLMS-a. Kada postoje greške u estimaciji sekundarne putanje FCLMS brže konvergira i od SOFxLMS-a. Sa druge strane, predloženi algoritam ima manju grešku u stacionarnom stanju od FxLMS-a i njegovih modifikacija.

LITERATURA

- [1] S. Kuo, D. R. Morgan, "Active Noise Control Systems", New York: Wiley, 1996.
- [2] S. Haykin, "Adaptive Filter Theory", Prentice Hall, 2002.
- [3] S. J. Elliot, "Signal Processing for Active control", Academic Press, New York, 2001.
- [4] I. Tabatabaei, W. H. Abdulla, "Theoretical convergence analysis of FxLMS algorithm", *Signal Processing*, pp. 3046-3055, 2010.
- [5] S. D. Snyder, C. H. Hansen, "The Effect of Transfer Function Estimation Errors on the Filtered LMS Algorithm", *IEEE Trans. Signal Process.*, Apr. 1994.
- [6] I. Tabatabaei, W. H. Abdulla, "Effects of Imperfect Secondary Path Modeling on Adaptive Active Noise Control Systems", *IEEE Transactions on Control Systems Technology*, vol. PP, pp. 1-11, 2011.
- [7] E. Bjarnason, "Active noise cancellation using a modified form of the filtered-XLMS algorithm", in *Proc. Eusipco Signal Processing V*, Brussels, 1992.
- [8] B. Krstajic, Z. Zecevic, Z. Uskokovic, "Increasing convergence speed of FxLMS algorithm in white noise environment", *AEU - International Journal of Electronics and Communications*, May 2013.
- [9] Ž. Zečević, B. Krstajić, "Modifikovani SOFxLMS Algoritam", *TELFOR 2012*, Beograd, Nov. 2012.
- [10] Ž. Zečević, B. Krstajić, "Alternativa FxLMS algoritmu", *ETAN 2013*, Zlatibor, Jun 2013.

UPOREDNA ANALIZA PERFORMANSI CAPON I CAPON-LIKE ALGORITAMA U SISTEMIMA PAMETNIH ANTENA COMPARATIVE PERFORMANCE ANALYSIS OF CAPON AND CAPON-LIKE ALGORITHM FOR SMART ANTENNA SYSTEM

Luka Lazović, Ana Jovanović, Vesna Rubežić, *Elektrotehnički fakultet*

Sadržaj: U ovom radu je predstavljena uporedna analiza Capon i Capon-like algoritama primjenjenih na linearne nizove kao i implementacija Capon-like algoritma na planarne antenske nizove. Razmatran je uticaj broja i rastojanja elemenata antenskih nizova, kao i broja iteracija. Numeričkim analizama je razmatran uticaj upadnog ugla signala na ponašanje algoritma kao i pronalaženje optimalnih parametara niza.

Abstract: This paper presents a comparative performance study between Capon and Capon-like algorithms used in linear and his implementation on planar antenna array. We studied the effect of antenna array parameters in terms of number and element spacing and number of iterations. In simulations we study the influence of angle of arrival on algorithm behavior as well as assessment of the optimal parameters of the array.

1. UVOD

Sistemi pametnih antena predstavljaju antenski niz sa algoritmima za obradu signala (array processing) u cilju automatske optimizacije i prilagođenja predajnog i/ili prijemnog dijagrama zračenja antenskog niza na konkretni scenario signala. Direction of arrival (DOA) i beamforming su ključne tehnologije u sistemima pametnih antena. Cilj DOA algoritma je da procijeni pravac signala željenog i interferirajućeg signala. Rezultat DOA procjene je dalje korišćen za podešavanje težinskih koeficijenata beamforming-a tako da se povećava snaga zračenja u pravcu željenog signala, a poništava u pravcu neželjenog.

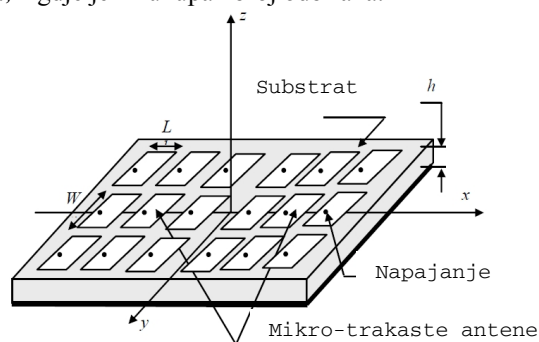
U posljednje vrijeme procjena pravca dolaska signala DOA dobija na važnosti u komunikacionim i radarskim sistemima za komercijalnu i vojnu upotrebu. Procjena pravca dolaska signala pomoću fiksnih antena ima ograničenja. Širina glavne laticije dijagrama zračenja je obrnuto proporcionalna dimenziji antene [1][2]. Povećavajući preciznost procjene ugla povećava se dimenzija antene što nije uvijek praktično. Umjesto korišćenja jedne antene, sistem antenskih nizova sa procesorom za obradu signala povećava preciznost procjene DOA. Planarni antenski niz ima više antenskih elemenata raspoređenih u prostoru na osnovu čega omogućava prostorno odabiranje primljenog talasa.

U ovom radu je predstavljena analiza novog Capon-like algoritma i njegova implementacija na planarnim antenskim nizovima. Cilj je utvrditi da li je moguće postići kvalitetan dijagram zračenja pomoću ovog algoritma čije je računanje značajno jednostavnije od ostalih DOA algoritama. [3][4][5]

2. OPIS PROBLEMA I PRIMJENJENOG METODA

Neka planarni antenski niz bude sastavljen od $N_x \times N_y$ antenskih elemenata raspoređenih u x - y ravni sa rastojanjem između elemenata d_x i d_y , što je prikazano na slici 1, i neka

primaju M uskopojasnih signala $s_m(t)$ od željenog izvora koji dolaze pod uglovima $\theta_1, \theta_2, \dots, \theta_M$ i $\phi_1, \phi_2, \dots, \phi_M$. Niz takode prima I uskopojasnih signala $s_i(t)$ od neželjenih izvora (ili interferirajuće signale) koji dolaze pod uglovima $\alpha_1, \alpha_2, \dots, \alpha_I$ i $\beta_1, \beta_2, \dots, \beta_I$ u određenim vremenskim trenucima $t=t_1, t_2, \dots, K$ gdje je K ukupan broj odabraka.



Slika 1. Planarni antenski niz

Suma signala $x_M(t)$ sa svakog reda antenskog niza se može definisati kao [6] :

$$x_x(t) = \sum_{i=1}^{N_x} s(t) e^{j(i-1)\psi_x}, \quad (1)$$

a faktor niza za svaki red postaje

$$AF_x(\theta, \phi) = \sum_{i=1}^{N_x} e^{j(i-1)\psi_x}. \quad (2)$$

Na isti način se određuje suma signala sa svake kolone niza, pa je faktor niza za svaku kolonu

$$AF_y(\theta, \phi) = \sum_{i=1}^{N_y} e^{j(i-1)\psi_y}. \quad (3)$$

Na osnovu principa množenja nizova, ukupni faktor planarnog niza se definiše kao [7] :

$$AF(\theta, \phi) = \sum_{i=1}^{N_x} \sum_{k=1}^{N_y} e^{j(i-1)\psi_x} e^{j(k-1)\psi_y}, \quad (4)$$

gdje su ψ_x and ψ_y , fazni pomjeraji između elemenata niza i mogu se definirati kao:

$$\psi_x = \frac{2\pi d_x}{\lambda} \sin(\theta) \cos(\phi) \quad (5)$$

$$\psi_y = \frac{2\pi d_y}{\lambda} \sin(\theta) \sin(\phi). \quad (6)$$

gdje je d rastojanje između elemenata, a λ talasna dužina primljenog signala.

Izraz (5) i (6) pokazuju da faktor planarnog niza zavisi kako od azimutnog ugla ϕ tako i od elevacionog ugla θ .

Posmatrajući ovaj matematički model sa stanovišta diskretnog signala, željeni signal $x_M(t)$ se sada može zapisati kao:

$$x_M(t) = A_M(\theta, \phi) s_m(t) \quad (7)$$

gdje je $A_M(\theta, \phi)$ matrica vektora željenog signala (faktor niza) reda $N \times M$ i može se zapisati kao:

$$A_M(\theta, \phi) = [a(\theta_1, \phi_1), a(\theta_2, \phi_2), \dots, a(\theta_M, \phi_M)]. \quad (8)$$

gdje je N broj elemenata antenskog niza i jednak je $N=N_x \cdot N_y$, dok je $s_m(t)$ vektor željenih signala reda $M \times 1$ definisan kao:

$$s_m(t) = [s_1(t) \quad s_2(t) \quad \dots \quad s_M(t)]^T \quad (9)$$

Analogno željenom signalu, na isti način se definiše i neželjeni interferirajući signal $x_i(t)$ kao:

$$x_i(t) = A_i(\theta, \phi) s_i(t), \quad (10)$$

gdje je A_i matrica vektora neželjenih signala reda $N \times I$, a $s_i(t)$ matrica neželjenih signala reda $I \times 1$.

Ukupan signal koji prima planarni antenski niz $x(t)$ dat je superpozicijom željenog signala, neželjenog (interferirajućeg signala) i bijelog Gausovog šuma kojeg detektuje antenski niz. Konačno signal $x(t)$ se može zapisati kao:

$$x(t) = x_M(t) + n(t) + x_i(t). \quad (11)$$

Ukoliko se analiza algoritama vrši na linearnim nizovima ϕ je fiksirano ili je jednako nuli.

A. CAPON algoritam

Capon algoritam, razvijen od strane J. Capon-a, je konvencionalni spektralni metod koji poboljšava karakteristike i rezoluciju Bartlett-ovog algoritma. Ključna ideja Capon algoritma je da minimizira primljenu snagu dolazećeg signala u svim pravcima ujedno održavajući jedinično pojačanje pravcima „gledanja“ [8][9]. Uslov postavljen ovim algoritmom se može zapisati

$$w^H a(\theta, \phi) = 1 \quad (12)$$

Gdje je w težinski koeficijent a H operator Hermitian tj. operator konjugovanog transponovanja.

Primjenjujući Lagranžov metod optimizacije, težinski koeficijent w se može izraziti kao:

$$w_{\text{Capon}} = \frac{\hat{R}^{-1} a(\theta, \phi)}{a^H(\theta, \phi) \hat{R}^{-1} a(\theta, \phi)} \quad (13)$$

gdje je $a(\theta, \phi)$ steering vektora, a \hat{R} kovarijansna matrica.

Korišćenjem optimizovanog koeficijenta w , prostorni spektral snage Capon algoritma se može predstaviti kao:

$$P_{\text{Capon}} = \frac{1}{a^H(\theta, \phi) \hat{R}^{-1} a(\theta, \phi)} \quad (14)$$

B. „CAPON-like„ algoritam

Capon-like algoritam je baziran na modifikovanom Capon algoritmu u cilju korišćenja u nizovima sa direktivnim antenama. Uopšteno, „steering“ vektor se može izraziti [10]

$$A[\theta, \phi] = [g(\theta_1, \phi_1) a(\theta_1, \phi_1), \dots, g(\theta_N, \phi_N) a(\theta_N, \phi_N)]^T \quad (15)$$

Gdje je $g(\theta_i, \phi_i)$ pojačanje niza u određenom pravcu definisanim sa θ i ϕ . Za antenske nizove sa izotropnim elementima $g(\theta_i, \phi_i)$ je jednako 1, za $i=1, 2, \dots, N$. Međutim, u slučaju antenskih nizova sa usmjerenim antenama, $g(\theta_i, \phi_i)$ je određeno dijagramom zračenja što značajno utiče na vrijednosti $a(\theta_i, \phi_i)$. Kako je dijagram zračenja niza pod uticajem dijagrama pojačanja antene u pravcu „gledanja“, uslov postavljen ovim algoritmom se može zapisati kao

$$w^H a(\theta, \phi) = g(\theta, \phi) \quad (16)$$

Primjenjujući Lagranžov metod optimizacije, težinski koeficijenti se mogu izraziti jednačinom:

$$w = \frac{\hat{R}^{-1} a(\theta, \phi) g^H(\theta, \phi)}{a^H(\theta, \phi) \hat{R}^{-1} a(\theta, \phi)} \quad (17)$$

Spektral snage Capon-like algoritma se može izraziti kao:

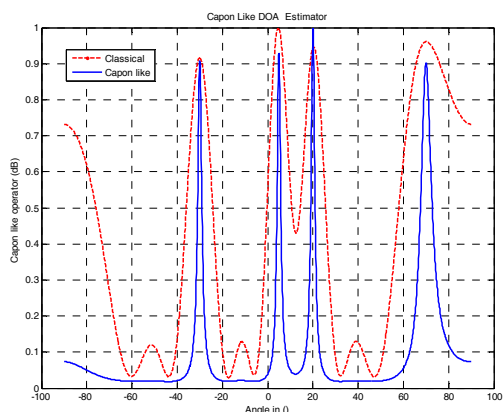
$$P = \frac{a^H(\theta, \phi) a(\theta, \phi)}{a^H(\theta, \phi) \hat{R}^{-1} a(\theta, \phi)} \quad (18)$$

U slučaju da se Capon i Capon-like algoritmi primjenjuju za linearne antenske nizove, važi relacija $a(\theta, \phi) = a(\theta)$ i $g(\theta, \phi) = g(\theta)$.

3. NUMERIČKI REZULTATI

U svim simulacijama, kao ulazni signal, korišćen je signal opisan jednačinom (11). Kao parametri u simulacijama mijenjani su uglovi dolaska signala, broj odbiraka, broj i rastojanje antenskih elemenata.

Na slici 2 je prikazan dijagram zračenja Capon-like algoritma i klasičnog DOA algoritma za upadne uglove $\theta_1 = -30^\circ$, $\theta_2 = 5^\circ$, $\theta_3 = 20^\circ$ i $\theta_4 = 70^\circ$. Simuliran je antenski niz sa 10 elemenata sa 700 odbiraka.

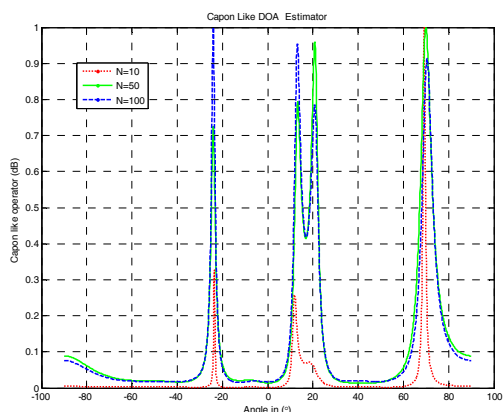


Slika 2. Dijagram zračenja Capon-like algoritma i klasičnog DOA algoritma za upadne uglove $\theta_1 = -30^\circ$, $\theta_2 = 5^\circ$, $\theta_3 = 20^\circ$ i $\theta_4 = 70^\circ$.

Na osnovu slike 2 se može zaključiti da Capon-like algoritam postiže dobre rezultate po pitanju rezolucije i snage zračenja u neželjenim pravcima.

Vrijeme potrebno za računanje klasičnog DOA algoritma je 0,645 s, vrijeme potrebno za računanje Capon algoritma je 0,174 s, dok je vrijeme potrebno za računanje Capon-like algoritma 0,110 na osnovu čega se može zaključiti da je za računanje Capon-like algoritma potrebno šest puta manje vremena od klasičnog DOA algoritma i 37 % manje vremena od Capon algoritma. Malo vrijeme računanja je glavna prednost ovog algoritma.

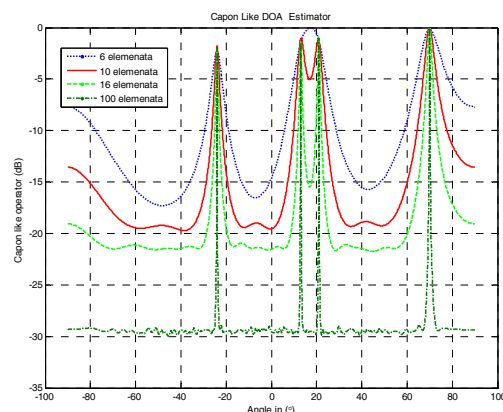
Na slici 3 je prikazana simulacija Capon-like algoritma za upadne uglove $\theta_1 = -24^\circ$, $\theta_2 = 13^\circ$, $\theta_3 = 21^\circ$ i $\theta_4 = 70^\circ$ za različiti broj iteracija algoritma.



Slika 3. Dijagram zračenja Capon-like algoritma za upadne uglove $\theta_1 = -24^\circ$, $\theta_2 = 13^\circ$, $\theta_3 = 21^\circ$ i $\theta_4 = 70^\circ$ za različiti broj iteracija algoritma.

Na osnovu dijagrama sa slike 3 može se zaključiti da algoritam postiže dobre rezultate za mali broj iteracija od svega 50. Za 10 iteracija moguća je procjena tri od ukupno četiri upadna ugla, što je veoma dobro i predstavlja prednost ovog algoritma u odnosu na druge DOA algoritme.

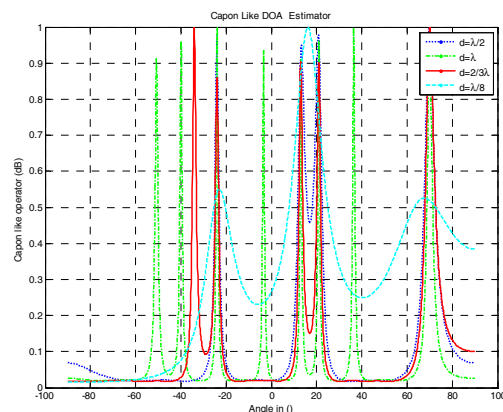
Na slici 4 je predstavljen dijagram zračenja Capon-like algoritma za upadne uglove $\theta_1 = -24^\circ$, $\theta_2 = 13^\circ$, $\theta_3 = 21^\circ$ i $\theta_4 = 70^\circ$ za različite brojeve elemenata antenskog niza.



Slika 4. Dijagram zračenja Capon-like algoritma za upadne uglove $\theta_1 = -24^\circ$, $\theta_2 = 13^\circ$, $\theta_3 = 21^\circ$ i $\theta_4 = 70^\circ$ za različite brojeve elemenata antenskog niza.

Na osnovu slike 4 može se zaključiti da što je veći broj elemenata antenskog niza rezultati procjene pravca su bolji.

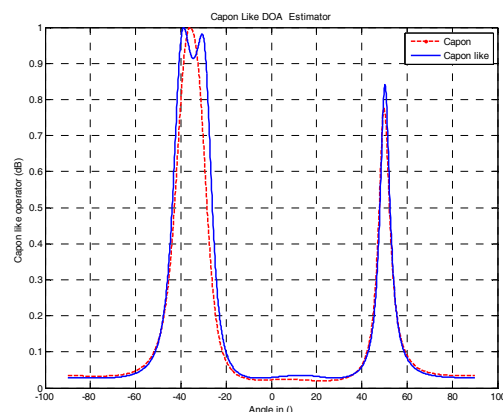
Na slici 5 je prikazan dijagram zračenja Capon like algoritma za upadne uglove $\theta_1 = -24^\circ$, $\theta_2 = 13^\circ$, $\theta_3 = 21^\circ$ i $\theta_4 = 70^\circ$ za različita rastojanja elemenata antenskog niza.



Slika 5. Dijagram zračenja Capon-like algoritma za upadne uglove $\theta_1 = -24^\circ$, $\theta_2 = 13^\circ$, $\theta_3 = 21^\circ$ i $\theta_4 = 70^\circ$ za različita rastojanja elemenata antenskog niza.

Na osnovu slike 5 može se zaključiti da jedino za rastojanje od $\lambda/2$ algoritam daje dobre rezultate. Za ostale kombinacije rastojanja, algoritam ne može da vrši tačnu procjenu.

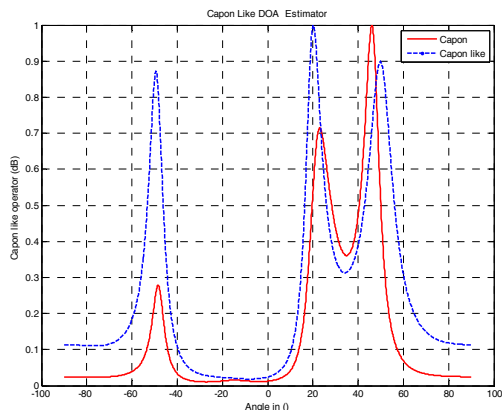
Na slici 6 je prikazan uporedni dijagram zračenja Capon i Capon-like algoritma za upadne uglove $\theta_1 = -30^\circ$, $\theta_2 = -40^\circ$ i $\theta_4 = 50^\circ$.



Slika 6. Uporedni dijagram zračenja Capon i Capon-like algoritma za upadne uglove $\theta_1 = -30^\circ$, $\theta_2 = -40^\circ$ i $\theta_4 = 50^\circ$

Na osnovu dijagrama sa slike 6 može se zaključiti da Capon-like algoritam postiže bolje rezultate kada su dva upadna ugla veoma blizu. Capon algoritam nije procijenio upadne uglove od -30° i -40° .

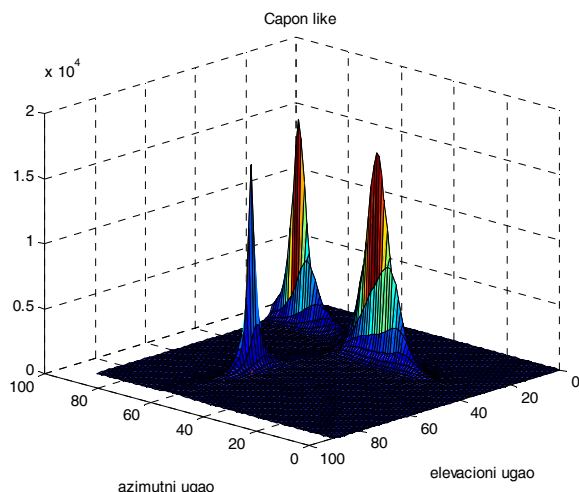
Na slici 7 je prikazan uporedni dijagram zračenja Capon i Capon-like algoritma za 10 iteracija kada su upadni uglovi $\theta_1 = -50^\circ$, $\theta_2 = 20^\circ$ i $\theta_3 = 50^\circ$.



Slika 7 Uporedni dijagram zračenja Capon i Capon-like algoritma za 10 iteracija kada su upadni uglovi $\theta_1 = -50^\circ$, $\theta_2 = 20^\circ$ i $\theta_3 = 50^\circ$.

Na osnovu dijagrama sa slike 7 možemo zaključiti da za mali broj iteracija, što je trend u razvoju adaptivnih algoritama, Capon-like algoritam tačno vrši procjenu upadnih uglova, dok Capon algoritam pravi grešku u procjeni. Capon-like algoritam u slučaju kada su dolazni uglovi bliski pokazuje bolje rezultate od Capon algoritma.

U cilju dalje analize performansi ovog algoritma izvršena je implementacija Capon-like algoritma na planarnim antenskim nizovima. Na slici 8 je predstavljen dijagram zračenja Capon-like algoritma primjenjenog na planarnim antenskim nizovima kada su upadni uglovi $\theta_1 = 20^\circ$, $\phi_1 = 80^\circ$, $\theta_2 = 30^\circ$, $\phi_2 = 40^\circ$ i $\theta_3 = 60^\circ$, $\phi_3 = 60^\circ$.



Slika 8 Dijagram zračenja Capon-like algoritma primjenjenog na planarnim antenskim nizovima kada su upadni uglovi $\theta_1 = 20^\circ$, $\phi_1 = 80^\circ$, $\theta_2 = 30^\circ$, $\phi_2 = 40^\circ$ i $\theta_3 = 60^\circ$, $\phi_3 = 60^\circ$.

4. ZAKLJUČAK

U ovom radu su analizirane performanse Capon-like algoritma primjenjenog na linearnim antenskim nizovima kao i implementacija algoritma na planarnim nizovima u sistemima pametnih antena. Simulacije su imale za cilj utvrđivanje brzine rada poboljšanog Capon-like algoritma u odnosu na Capon algoritam kao i utvrđivanje njegove preciznosti u slučaju bliskih upadnih uglova.

U radu je razmatran uticaj broja antenskih elemenata na rad algoritma, pri čemu je utvrđeno da Capon-like algoritam daje bolje rezultate u slučaju antenskih nizova sa malim brojem elemenata.

Po pitanju broja iteracija, Capon-like algoritam postiže bolje rezultate, pa za svega 10 iteracija uspijeva da izvrši procjenu upadnih uglova. Brzina računanja, mali broj elemenata i mali broj iteracija su glavna prednosti Capon-like algoritma u odnosu na ostale DOA algoritme.

Kada su u pitanju veliki planarni nizovi i veliki broj iteracija rezultati simulacije su gotovo isti kao u slučaju Capon algoritma osim brzine procjene koja je manja.

Simulacijama za bliske upadne uglove pokazano je da Capon-like algoritam postiže bolje rezultate, što je tendencija današnjeg razvoja algoritama.

LITERATURA

- [1] Alan J. Fenn, Adaptive Antennas and Phased Arrays for Radar and Communications.: Massachusetts Institute of Technology Lincoln Laboratory, 2005.
- [2] George V. Tsoulos, Ed., Adaptive antennas for wireless communications.: IEEE Press.
- [3] Lazović Luka, Jovanović Ana, „ Uporedni pregled performansi LMS i NLMS algoritama u sistemima pametnih antena“ IT 2013, Žabljak.
- [4] Lazović Luka, Jovanović Ana, Veljović Zoran, „Analiza performansi i geometrijska optimizacija DOA algoritama primjenjenih na planarnim antenskim nizovima u sistemima smart antena“, ETRAN 2013, Zlatibor.
- [5] Lazović Luka, Jovanović Ana, „Comparative performance study of DOA algorithm applied on linear antenna array in smart antenna systems“, MECO 2013, Budva.
- [6] Momčilo B. Dragovic, Antene i prostiranje radio talasa. Beograd: Akademska misao, 2008.
- [7] Majharul Islam Rajib and Sumon Kumar Biswas Muhammad Mahfuzul Alam, "Design and Performance analysis of smart antenna system for DECT radio base station in Wireless local loop," Journal of communications, vol. 5, pp. 593-603, august 2010.
- [8] Rias Muhamed, Direction of arrival estimation using antenna arrays, Blacksburg, Virginia.
- [9] Petre Stoica, Randolph Moses: Spectral Analysis of Signals, Upper Saddle River, New Jersey, 2005.
- [10] Rahmat Sanudin, Nurul H. Noordin, Ahmed O. El-Rayis, Nakul Haridas, Ahmet T. Erdogan and Tughrul Arslan, "Capon-Like DOA Estimation Algorithm for Directional Antenna Arrays," Loughborough Antennas & Propagation Conference, 14-15 November 2011, Loughborough, UK.

MODELOVANJE SVJETILJKI PRI ANALIZAMA KVALITETA ELEKTRIČNE ENERGIJE U MREŽAMA JAVNE RASVJETE

LUMINAIRES MODELING WHEN ANALAYZING POWER QUALITY IN THE STREET LIGHTING NETWORK

Mladen Rašović, Komunalne usluge d.o.o. Podgorica
Jadranka Radović, Saša Mujović, Univerzitet Crne Gore, Elektrotehnički fakultet u Podgorici

Sadržaj: Javna rasvjeta je jedan od najznačajnijih potrošača električne energije u gradskim elektrodistributivnim sistemima. Osnovni element javne rasvjete je svjetiljka koja se sastoji od sijalice – izvora svjetlosti i odgovarajućih predspojnih uređaja neophodnih za normalan rad i funkcionisanje u uslovima priključenja na distributivnu mrežu. Svjetiljke javne rasvjete kao pojedinačni potrošači, odnosno javna rasvjeta kao grupni potrošač su nelinearni potrošači električne energije. Glavni izvor nelinearnosti su sijalice koje rade na principu električnog pražnjenja u smješi gasova. Sam proces pražnjenja je dosta složen i njegovo modelovanje predstavlja multidisciplinarni izazov jer zahtijeva poznavanje više naučnih disciplina kao što su: fizika plazme, materijali, hemija, elektrotehnika i dr. U literaturi postoji više modela koji opisuju rad ove vrste sijalica. U ovom radu se pokazuje da je model zasnovan na metodi promjenljive provodnosti plazme najadekvatniji za analize strujnih karakteristika opterećenja sa aspekta uticaja na kvalitet električne energije. Uporedna analiza rezultata proračuna izvršenih na osnovu formiranog simulacionog modela u programskom paketu Matlab Simulink i karakteristia dobijenih mjerenjem, potvrđuje visoku tačnost modela.

Abstract: Street lighting is one of the most frequent consumer of electric energy of the towns power distribution systems. Main element of street lighting is luminaire which contains a bulb and necessary control gear for normal operation and operation in cases of connecting to distribution grid. The Street light luminaires like a single consumers, respectively the street lighting like a group consumer, are a nonlinear consumers of electric energy. The main source of nonlinearity is the bulb which works to the principle of electric gas discharge. The gas discharging process is very complex and his modeling is multidisciplinary quest, beacuse it needs a knowledge many science disciplines as: plasma physics, materials, chemistry, electrics etc. In the literature, there are many models which describe the work of this type bulbs. In this paper is showing that to model based to the method of variable plasma conductivity is the most adequate for analysis current characteristics of load with aspect of impact to the quality of electric energy. Comparative analysis of calculated results and simulation model to the program package Matlab Simulink and measured characteristic, confirm high accuracy of model.

1. UVOD

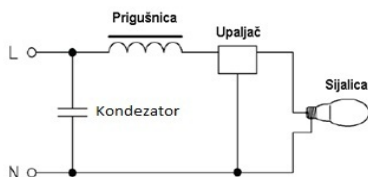
Značajne promjene koje su nastale u funkcionisanju savremenih elektroenergetskih sistema kao i u strukturi konzuma električne energije, doprinijele su aktuelizaciji problematike vezane za kvalitet električne energije. Sa pokretanjem procesa demonopolizacije (deregulacije) elektroprivrede i uvođenjem konkurencije među distributerima električne energije, pitanje kvaliteta električne energije, kao robe, se postavlja u svim svojim aspektima. Prisustvo velikog broja nelinearnih potrošača u distributivnim mrežama može izazvati niz negativnih efekata koji se odražavaju kako na samu mrežu, tako i na ostale priključene potrošače. Stoga je zajednički interes potrošača i proizvođača električne energije posljednjih godina aktuelizovao probleme vezane za kvalitet električne energije, prvenstveno za pojavu, sadržaj i uticaje viših harmonika struja i napona u distributivnim mrežama.

U okviru ovog rada izučava se električno osvjjetljenje kao nelinearni potrošač električne energije. Za analizu je odabrano javno osvjjetljenje (javna rasvjeta), kao najznačajniji grupni potrošač iz ove kategorije. Osnovni element javne rasvjete je svjetiljka sa sijalicom kao električnim izvorom svjetlosti i predspojnom opremom koja omogućava njeno funkcionisanje u uslovima priključenja na distributivnu mrežu. U javnoj rasvjeti se kao električni izvori svjetlosti dominantno koriste sijalice koje rade na principu električnog pražnjenja u gasu. Sam princip rada ovih električnih izvora je osnovni izvor nelinearnosti, a dodatni uticaj imaju i ostali elementi (starter, prigušnica, kondenzator za kompenzaciju reaktivne energije) u električnom kolu svjetiljke. Rad grupe ovakvih potrošača, prisutne upravo u javnoj rasvjeti, može izazvati značajne smetnje, u radu, prvenstveno kod potrošača koji su povezani na iste sabirnice, odnosno električno bliskih potrošača. Takođe, može izazvati i druge neželjene posljedice, kao što su pojave viših harmonika u mrežnom napajanju, povećanje gubitaka u vodovima i transformatorima, preopterećenje struje neutralnog

provodnika i slično. Istovremeno, za optimalno funkcionisanje javna rasvjeta zahtjeva napajanje mrežnim naponom visokog kvaliteta. U suprotnom može doći do neugodnih posljedica, kako za same svjetiljke tako i po komfor i bezbjednost korisnika. Takođe, prisustvo viših harmonika u mrežnom naponu utiče na dodatnu deformaciju talasnog oblika struja priključenih svjetiljki javne rasvjete. Metode modelovanja i simulacije od posebne su važnosti za izučavanje uticaja nelinearnih potrošača, konkretno električnog osvjetljenja, na kvalitet električne energije. Na osnovu sprovedene analize, od mnoštva matematičkih modela sijalice sa pražnjenjem u gasu koji se daju u literaturi, za analize talasnih oblika krivih parametara režima i prisustva viših harmonika, odabran je i prikazan model zasnovan na metodi promjenljive provodnosti plazme. Poređenjem rezultata proračuna izvršenih na osnovu formiranog simulacionog modela u programskom paketu Matlab Simulink i karakteristika dobijenih mjerenjem, potvrđena je visoka tačnost modela.

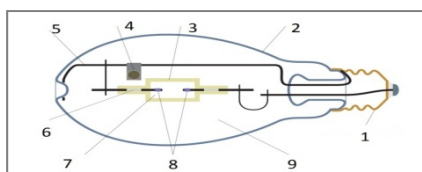
2. PRIRODA NASTANKA NELINEARNOSTI

Električno kolo svjetiljke koja je najzastupljenija u javnoj rasvjeti je prikazano na slici 1. Svi elementi ovog kola su prilagođeni za rad na standardnom mrežnom naponu efektivne vrijednosti 230V i učestanosti 50 Hz.



Slika 1. Električno kolo svjetiljke

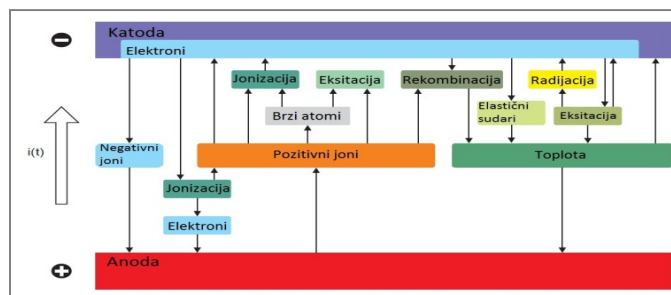
Svjetlosni izvor je sijalica koja radi na principu električnog pražnjenja u gasu. Svjetlost je posljedica (pozitivna) samog procesa električnog pražnjenja, tokom kojeg se električna energija transformiše u kinetičku energiju pokretnih elektrona, odnosno u elektromagnetsko zračenje kao rezultat kolizionih procesa u atomima smješe gasa i metala. Tipična konstrukcija ovakve sijalice je prikazana na slici 2. Sijalica je konstruktivno prilagođena za električno pražnjenje u gasu pod visokim pritiskom i to od 30 do 65 kPa, dok je temperatura plazmenog kanala do 4000 K.



Slika 2. Konstrukcija sijalice sa koja radi na principu električnog pražnjenja u gasu

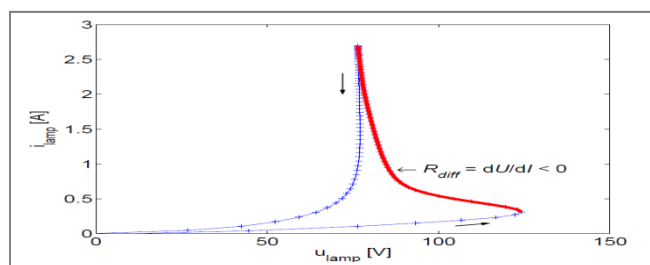
1-podnožak, 2-spoljašnji omotač, 3-gorionik od prozračnog materijala, 4-upijač nečistoća, 5-nosač gorionika, 6-završetak gorionika, 7-unutrašnjost gorionika, 8-elektrode, 9-vakuum ili gas.

Tok procesa prilikom električnog pražnjenja u gasu između elektroda gorionika sijalice, prikazan je na slici 3 [2].



Slika 3. Procesi u sijalici prilikom električnog pražnjenja

Izobličen talasni oblik struje potrošnje, odnosno nelinearna strujno-naponska karakteristika sijalice sa električnim pražnjenjem u gasu (slika 4), negativna je posljedica svih ovih procesa [3], [4].



Slika 4. Tipična strujno-naponska karakteristika sijalice koja radi na principu električnog pražnjenja u gasu

Takođe i ostali elementi električnog kola svjetiljke, prvenstveno prigušnica i kondenzator za popravku faktora snage, doprinose dodatnom izobličenju talasnog oblika struje potrošnje u sistemima javne rasvjete.

3. MODELOVANJE SVJETILJKI JAVNE RASVJETE

U literaturi postoji mnogo vrsta i oblika modela sijalice sa električnim pražnjenjem u gasu [5]. Za njihovo izučavanje sa aspekta kvaliteta električne energije, najadekvatniji su polufizički modeli, tzv. modeli „crne kutije” i to sa parametrima koji se dobijaju mjerenjima u realnim sistemima, odnosno na stvarnim svjetiljkama. Modeli se realizuju u specijalizovanim programskim okruženjima kao što su SPICE ili Matlab [6]. Oni se dominantno zasnivaju na analizi karakteristika ekvivalentne provodnosti odnosno otpornosti. Osnovu modela predstavlja činjenica da pri naizmjeničnom naponu učestanosti 50 Hz, struja koja protiče kroz sijalicu nije sinusoidalnog oblika već je izobličena. To znači da odnos U/I ne predstavlja konstantnu vrijednost već se mijenja po nekoj zavisnosti koja je posljedica procesa u sijalici. Rad sijalice se najadekvatnije opisuje G-modelom [6]. Model se zasniva na analizi električne provodnosti plazme $G(t)$ i principima modelovanja ekvivalentne provodnosti:

$$\frac{dG(t)}{dt} = a_2 i^2(t) - \sum_{i=1}^N b_i G^i(t) \quad (1)$$

Jednačina (1) je diferencijalna jednačina prvog reda, gdje se koeficijenti a i b određuju sa strujno-naponskih

karakteristika realnih svjetiljki. Fizički, značenja ovih koeficijenata se vezuju za energiju koja se predaje plazmi, gubitke u zračenju, termičke gubitke, broj elastičnih sudara i dr. Za modelovanje može se, zavisno od tipa sijalice, primjeniti polinomijalni (2), eksponencijalni (3) i kvadratni model (4):

$$\frac{dG(t)}{dt} = a_2(i(t) + a_1)^2 - [b_3G^3(t) + b_3G^3(t) + b_1G(t) + b_0] \quad (2)$$

$$\frac{dG(t)}{dt} = \frac{a_2}{G(t)}(i(t) + a_1)^2 - [b_2e^{b_3G(t)} + b_1G(t) + b_0] \quad (3)$$

$$\frac{dG(t)}{dt} = \frac{a_2}{G(t)}(i(t) + a_1)^2 - [b_3G^3(t) + b_3G^3(t) + b_1G(t) + b_0] \quad (4)$$

Poželjno je da simulacioni model obuhvati sva tri oblika modela, a što se može realizovati preko sljedeće jednačine:

$$\frac{dG(t)}{dt} = \frac{a_2}{G(t)^{fraG}}(i(t) + a_1)^2 - [(b_2e^{b_3G(t)}eqE) + (b_3G^3(t) + b_2G^2(t))eqE + b_1G(t) + b_0] \quad (5)$$

gdje su:

$a_2, a_1, b_3, b_2, b_1, b_0$ – parametri modela provodnosti koji se dobijaju mjerenjima ili u podacima proizvođača sijalice;

$fraG$ – prva univerzalna konstanta. Ako je 0, tada je izabran polinomijalni model, ako je 1, tada je model provodnosti neodređen i zavisi od sljedeće konstante eqE ;

eqE – druga univerzalna konstanta koja ima sljedeće značenje: ako je 0, tada je izabran polinomijalni ili kvadratni model provodnosti, zavisno od prethodne konstante $fraG$, ako je 1, tada je izabran eksponencijalni model provodnosti.

Jednačina (5) predstavlja matematički model sijalice sa električnim pražnjenjem u gasu, izražen kroz provodnost koja se mijenja u toku vremena uvažavajući sve, ranije pomenute, procese tokom rada sijalice. Za modelovanje svjetiljke, polazi se od kola prikazanog na slici 1. Kolo se može opisati jednačinom:

$$u(t) = R_{ball}i(t) + L_{ball}\frac{di(t)}{dt} + u_{lamp}(t) \quad (6)$$

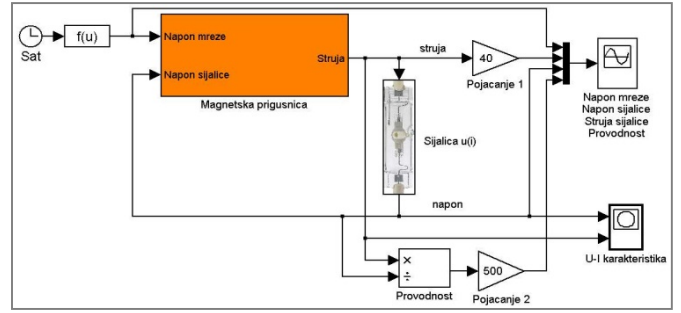
gdje su R_{ball} i L_{ball} parametri prigušnice.

Napon napajanja predstavlja idealni naponski izvor sa sinusnim oblikom signala:

$$u(t) = \sqrt{2}U_{rms}\sin(\omega t) \quad (7)$$

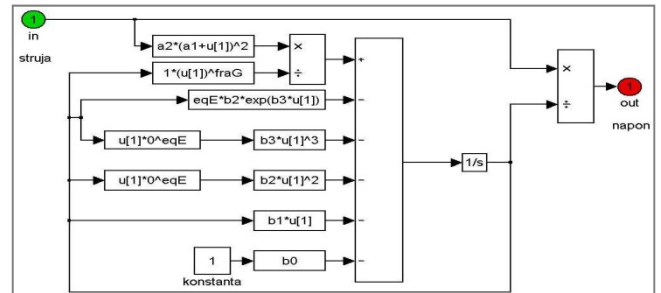
gdje je: U_{rms} – efektivna vrijednost napona mreže (230 V).

Formirani Matlab Simulink model svjetiljke, prikazan je na slici 5.



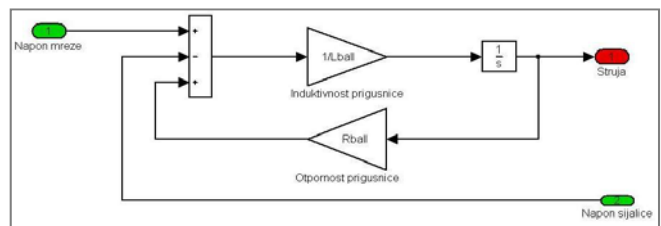
Slika 5. Simulink model svjetiljke koja radi na principu električnog pražnjenja u gasu

Model sadrži sve elemente relevantne za analize karakteristika rada (provodnosti, struje i napona) svjetiljke u ustaljenom režimu, kao i pomoćne elemente za snimanje napona mreže, napona sijalice, struja sijalice, vremenske promjene provodnosti i strujno-naponske zavisnosti ovog nelinearnog potrošača. Simulacioni model sijalice, definisan izrazom (5), prikazan je slici 6.



Slika 6. Simulink model sijalice sa električnim pražnjenjem u gasu, koji uvažava sve procese prilikom njenog rada

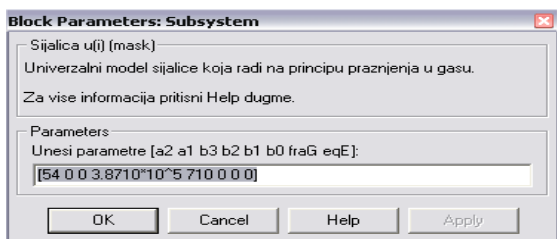
Magnetska prigušnica je predstavljena linearnim modelom kao na slici 7, sa samo dva parametra (otpornost i induktivnost).



Slika 7. Simulink model magnetske prigušnice

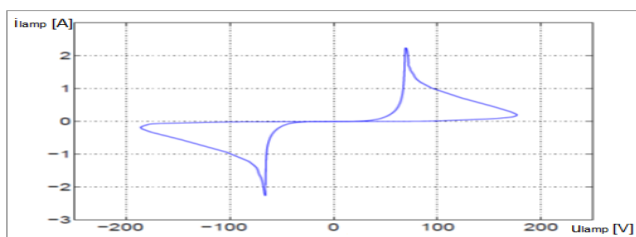
4. VERIFIKACIJA SIMULACIONOG MODELA

Verifikacija formiranog simulacionog modela svjetiljke realizuje se upoređivanjem podataka koje daje simulacioni model sa mjernim podacima. Konkretno se prikazuju rezultati za natrijumovu svjetiljku visokog pritiska, snage 150 W. Dvostrukim klikom na element modela "sijalica" sa slike 5, otvara se polje "Block Parameters" za unos ulaznih parametara sijalice (slika 8). Neophodno je unijeti i vrijednosti parametara prigušnice konkretno $R_{ball}=7 \Omega$ i $L_{ball}=0.35$ H. Takođe se odabiraju i vrijednosti elemenata pojačanja, konkretno $GG=40$ i $GI=5000$.



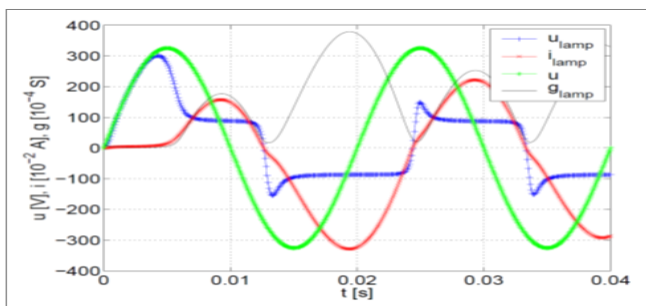
Slika 8. Unos parametra za sijalicu (polinomijalni model, natrijumova sijalice visokog pritiska, snage 150 W)

Formirani simulacioni model daje strujno-naponsku karakteristiku i talasne oblike svih napona i struje svjetiljke. Za razmatrani tip svjetiljke sa električnim prežnjenjem u gasu, strujno-naponska karakteristika dobijena simulacijom prikazana na slici 9. Njen oblik potvrđuje da se radi o nelinearnom potrošaču.

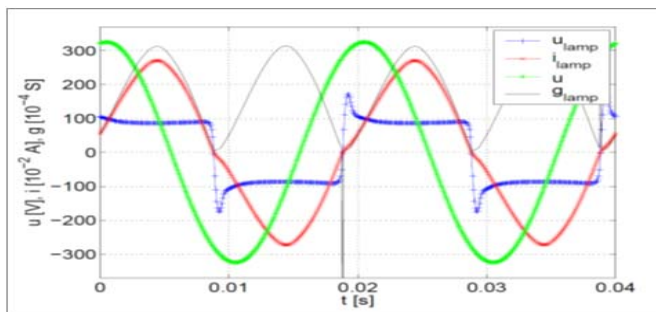


Slika 9. Strujno-naponska karakteristika - simulacija

Taladni oblici karakterističnih veličina (napon mreže, napon lampe, struja lampe i provodnost) prikazani su na slici 10.



Slika 10. Talasni oblici karakterističnih veličina- simulacija



Slika 11. Talasni oblici karakterističnih veličina- mjerenje

Podudarnost simulacionih rezultata i rezultata mjerenja (slika 11), verifikuje formirani simulacioni model za primjenu u analizama kvaliteta električne energije u sistemima javne rasvjete.

5. ZAKLJUČAK

Savremeni elektrodistributivni sistemi u cilju njihove modernizacije, optimizacije rada i minimizacije eksploatacionih troškova, zahtijevaju i poznavanje karakteristika potrošača konzuma sa aspekta kvaliteta električne energije. Za ova izučavanja, posebno su značajni nelinearni potrošači, čije se učešće u strukturi distributivnih konzuma stalno povećava. U grupu nelinearnih potrošača spada i električno osvjetljenje sa svjetiljkama koje su osnovni izvor nelinearnosti. Javno osvjetljenje (javna rasvjeta) je najznačajniji nelinearni potrošač iz ove kategorije. Izučavanja sistema javne rasvjete sa aspekta kvaliteta električne energije se baziraju na mjernim metodama u realnim sistemima i metadama modelovanja. U sistemima javne rasvjete razvijenih zemalja, koje se već odavno bave ovom problematikom, najčešće se primjenjuje kombinacija oba načina. Mjernim metodama se dobijaju kako ulazni podaci za simulacione modele tako i karakteristike za njihovu verifikaciju. Takođe, one su najbolja podloga za konkretne analize kvaliteta električne energije u sistemima javne rasvjete. Metode modelovanja imaju prvorazredni značaj u planovima i primjenama novih tehnologija u ovim sistemima. Osnova modelovanja u analizama kvaliteta električne energije je model sijalice sa električnim praznjenjem u gasu, koje se u javnoj rasveti dominantno koriste. Visoka podudarnost karakteristika dobijenih mjerenjima i karakteristika dobijenih sprovedenom simulacijom sa modelom sijalice koji se zasniva na analizi električne provodnosti plazme $G(t)$ i principima modelovanja ekvivalentne provodnosti, preporučuje ovaj model svjetiljki kao najpovoljniji za analize kvaliteta električne energije u sistemima javne rasvjete.

LITERATURA

- [1] C. Boffito and A. Corazza. „Getters for Lighting Applications“, In Light Sources 2004, Proceedings of the 10th International Symposium on the Science and Technology of Light Sources (LS10), pp 95–104. ISBN 0-7503-1007-3.
- [2] Lj. Milanković „Tehnika visokog napona” izdanje Elektrotehničkog fakulteta u Beogradu, Beograd 1987 g.
- [3] J. B. Ruffet. „Modelisation des lampes a descharge aux halogenures metalliques avec enveloppe en ceramique a l'ade d'un code fluide“. PhD thesis, Université Paul Sabatier de Toulouse III, France, 2005.
- [4] P. Flesch, „Light and Light Sources. High-Intensity Discharge Lamps“, Springer, Berlin, June 2006. ISBN 3-540-32684-7.
- [5] M.T. Au, and J.V. Milanović, „Development of Stochastic Aggregate Harmonic Load Model Based on Field Measurements”, IEEE Transaction on Power Delivery, Vol. 22, No. 1, Jan. 2007, pp.323–330.
- [6] G. Zissis and J.-J. Damelincourt, „Modelling discharge lamps for electronic circuit designers: a review of the existing methods“, In The 29th IEEE International Conference on Plasma Sciences (ICOPS2002), page 318. Banff, Canada, May 26–30 2002.

FLATNESS UPRAVLJANJE AUTONOMNO VOĐENIM VOZILOM FLATNESS BASED CONTROL OF AUTOMATED GUIDED VEHICLE

Novica Daković, Milovan Radulović, *Elektrotehnički fakultet, Univerzitet Crne Gore*

Sadržaj: *U radu je prikazana simulacija kretanja autonomnog vozila primjenom metode flatness upravljanja. Na osnovu kinematičkog modela kretanja vozila projektovan je feed-forward regulator za praćenje referentne trajektorije. Za stabilizaciju sistema u prisustvu smetnji konstruisan je feedback regulator zasnovan na principu linearne dinamike greške. U oba slučaja upravljanja upoređeni su rezultati stvarnog i željenog kretanja.*

Abstract: *This paper presents simulation of the motion of autonomous vehicle using the flatness based control. Based on the kinematic model of the vehicle motion a feed-forward trajectory tracking controller has been designed. For system stabilization in the presence of disturbances a feedback controller based on linear error dynamics has been constructed. In either case, comparative study between real moving and desired one has been presented.*

1. UVOD

U cilju postizanja veće produktivnosti i efikasnosti proizvodnih procesa, koje karakteriše serijska proizvodnja, nametnula se upotreba autonomnih vozila (AV). Osnovni zadatak AV, u procesima ovog tipa, je transport djelova između obradnih centara. Navigacija AV se u posmatranom slučaju može podijeliti na dva zadatka koje je potrebno riješiti. Prvi zadatak je navigacija AV od jedne radne stanice do druge (point to point navigacija), a kao drugi zadatak se postavlja manevar prilaza (dokovanja) AV radnoj stanici. Manevar dokovanja zahtijeva veći nivo preciznosti od zadatka navigacije. Prilikom prilaza radnoj stanici može, zbog nepreciznog manevra, doći do oštećenja AV, radne stanice ili djelova koji se transportuju.

U skladu sa zahtjevom da se kretanje AV uskladi sa zadatim referentnim kretanjem razvijene su razne metode upravljanja. Kompleksnost dinamike AV, problemi određivanja stvarnih parametara i njihova promjenljivost motivišu istraživače da rješenja traže koristeći tehnike soft-computing-a u vidu fuzzy logike [1][2], neuralnih mreža [3], kao i adaptivne tehnike inspirisane biološkim imunim sistemom [4]. Posljednja generacija mobilnih robota, kao i AV navigacioni sistem zasnivaju se na bežičnim tehnikama vođenja, koristeći laserske ili inercijalne senzore u cilju estimacije pozicije AV sa dovoljnom preciznošću da bi se obavio manevar dokovanja [5]. Međutim, industrijski navigacioni sistemi se suočavaju sa problemima niske brzine estimacije pozicije. U cilju rješavanja ovog problema vrši se fuzija dobijene estimacije pozicije sa odometrijskim podacima preuzetih sa enkodera. Ipak, brzina procjene pozicije je i uz ovo poboljšanje još uvijek relativno niska

Kao jedno od mogućih rješenja u cilju poboljšanja brzine AV pri manevaru dokovanja u ovom radu je predložena metoda diferencijalnog Flatness (glatkog) upravljanja. Tehnike Flatness upravljanja su razvijane poslednjih godina i imaju uspješnu primjenu u mnogim industrijskim procesima koji se odnose na rješavanje problema planiranja i praćenja

referentne trajektorije. Značajni rezultati postignuti su u upravljanju termalnim [6] i hemijskim procesima [7], kao i u upravljanju motorima [8], kranovima [9], itd. Ovaj metod su uveli M. Fliess i njegovi saradnici 1992. godine [10] kao način upravljanja kontinualnim nelinearnim sistemima.

Metoda diferencijalnog Flatness upravljanja predstavlja prirodan izbor za planiranje kretanja vozila ukoliko se zahtjeva eksplicitno praćenje zadate trajektorije. Postojanje varijable koja se zove „flat izlaz“ omogućava da se preko nje definišu sve systemske varijable. Dinamika takvog sistema se može izvesti bez rješavanja diferencijalnih jednačina i takav sistem je ustvari flat sistem. Prema tome, moguće je izraziti stanja, kao i ulaze sistema kao funkcije od flat izlaza i konačnog broja njegovih izvoda.

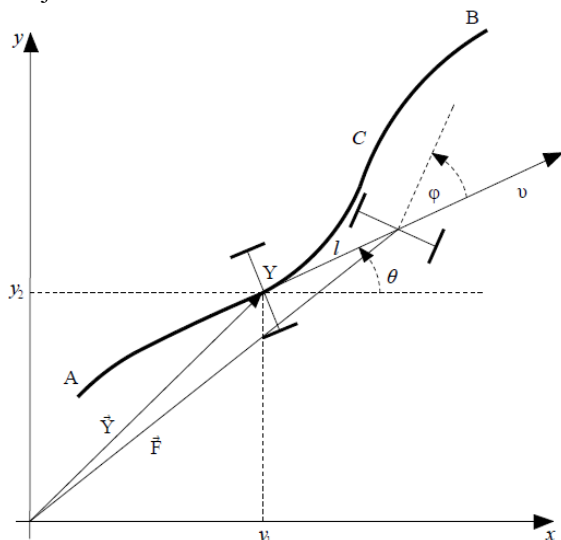
Problem praćenja trajektorije se sastoji u određivanju referentne trajektorije i parametrizaciji systemskih varijabli preko flat izlaza. Zatim se projektuje sistem u zatvorenoj petlji kako bi se dobio stabilni sistem koji prati zadatu trajektoriju sa greškom koja asimptotski teži nuli. Prednost flat sistema je da flat izlaz može da prati proizvodnu trajektoriju pod uslovom da je ta trajektorija dovoljno diferencijabilna (glatka).

U ovom radu je projektovan Flatness zakon upravljanja na primjeru kinematičkog modela vozila. Potrebno je projektovati putanju i to predstavlja rješenje problema praćenja putanje. Kao druga pretpostavka razmatraju se ograničenja u pogledu vremena ili brzine vozila. U tom slučaju, kada vrijeme postaje dodatni parametar, rješava se problem praćenja trajektorije.

U radu su prikazana dva načina upravljanja. Prvi način je feed-forward upravljanje koje je efikasno pod uslovom da je model objekta sistema idealan, kao i da sistem nije podvrgnut nikakvim eksternim smetnjama. Drugi način je feedback upravljanje koje poboljšava performanse i stabilizuje sistem. Projektovanje feedback sistema zahtjeva opisivanje dinamike greške na osnovu koje se proračunava zakon upravljanja.

2. FORMULACIJA PROBLEMA

Šematski prikaz vozila za koji ćemo postaviti kinematički model dat je na slici 1.



Slika 1. Šematski prikaz vozila

Relacije koje opisuju kinematički model vozila u stacionarnom referentnom koordinatnom sistemu su:

$$\dot{y}_1 = v \cos \theta \quad (1)$$

$$\dot{y}_2 = v \sin \theta \quad (2)$$

$$\dot{\theta} = v \frac{\tan \varphi}{l} \quad (3)$$

gdje su y_1, y_2 koordinate tačke Y koja se nalazi na presjeku uzdužne ose i ose pogonskih točkova vozila u referentnom koordinatnom sistemu, a θ, v, φ su ugao u odnosu na x osu, apsolutna brzina i ugao usmjerenja prednje osovine vozila, respektivno.

Flat izlaz ovog sistema je pozicija centra zadnje osovine $y = (y_1, y_2)$. Na osnovu jednačina (1), (2) i (3) moguće je parametrizovati sve varijable sa ovim flat izlazom:

$$\theta = \arctan \left(\frac{\dot{y}_2}{\dot{y}_1} \right) \quad (4)$$

$$v = \pm \sqrt{\dot{y}_1^2 + \dot{y}_2^2} \quad (5)$$

$$\varphi = \arctan \left(\frac{l\dot{\theta}}{v} \right) = \pm \arctan \left(\frac{l(\ddot{y}_2\dot{y}_1 - \ddot{y}_1\dot{y}_2)}{(\dot{y}_1^2 + \dot{y}_2^2)^{\frac{3}{2}}} \right) \quad (6)$$

Varijable v i φ su upravljački ulazi posmatranog sistema koje je potrebno proračunati.

3. FEED-FORWARD UPRAVLJANJE

Feed-forward upravljanje je moguće projektovati ako se smatra da se model vozila podudara sa realnim sistemom, da

na vozilo ne djeluju smetnje i da se početni uslovi potrebni za proračun upravljanja poklapaju sa stvarnim.

Najprije je potrebno definisati željenu putanju. Ovdje će se uvesti eksplicitna relacija koja povezuje koordinate $y_2 = f(y_1)$. Da bi se zadovoljio uslov o dovoljnoj glatkosti ili stepenu diferencijabilnosti putanje najjednostavnije je $f(y_1)$ opisati polinomom:

$$y_2 = f(y_1) = c_0 + c_1 y_1 + c_2 y_1^2 + c_3 y_1^3 + c_4 y_1^4 + c_5 y_1^5 \quad (7)$$

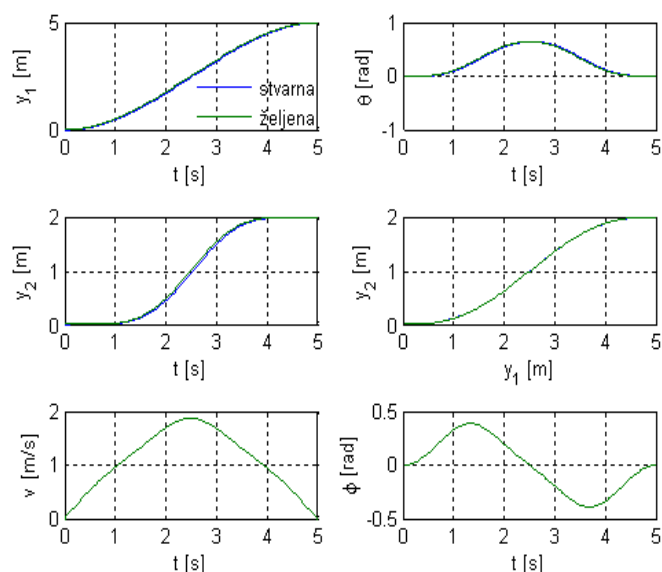
Izraz (7) je polinom petog stepena, jer se nepoznatih šest koeficijenata proračunavaju na osnovu tri početna uslova $(\theta_A, \varphi_A, y_{2,A})$ za poznatu početnu vrijednost $y_{1,A}$ i tri krajnja uslova $(\theta_B, \varphi_B, y_{2,B})$ za poznatu krajnju vrijednost $y_{1,B}$.

Sada je potrebno izvršiti parametrizaciju po vremenu izborom polinoma trećeg stepena za flat izlaz $y_1(t)$:

$$y_1(t) = d_0 + d_1 t + d_2 t^2 + d_3 t^3 \quad (8)$$

Koeficijenti iz relacije (8) se računaju na osnovu početnih uslova za $t = 0$ ($y_1(0) = y_{1,A}, \dot{y}_1(0) = 0$) i krajnjih uslova za $t = t_*$ ($y_1(t_*) = y_{1,B}, \dot{y}_1(t_*) = 0$). Trajektorija flat izlaza vozila $y = (y_1(t), y_2(t))$ određuje i trajektorije upravljačkih ulaza $v(t)$ i $\varphi(t)$ na osnovu relacija (5) i (6).

Simulacioni rezultati dobijeni primjenom Matlab-a prikazani su na slici 2. Početni uslovi za trenutak $t = 0s$ su $(y_{1,A}, y_{2,A}, \theta_A, \varphi_A, v_A) = (0, 0, 0, 0, 0)$, a za $t = 5s$ krajnji uslovi su $(y_{1,B}, y_{2,B}, \theta_B, \varphi_B, v_B) = (5, 2, 0, 0, 0)$.



Slika 2. Simulacija feed-forward upravljanja

Sa slike 2. se uočava da će se, pod pretpostavkom o idealnosti kinematičkog modela vozila, stvarna trajektorija u potpunosti podudarati sa željenom. Time je pokazana opravdanost izbora opisa flat izlaza na predloženi način. Ako je AV dovoljno precizno zauzelo poziciju, prije manevara

dokovanja, moguće je i na ovaj način realizovati upravljanje sa zadovoljavajuće tačnim krajnjim položajem vozila.

4. FEEDBACK UPRAVLJANJE

Feed-forward upravljanje često nije dovoljno da bi se postiglo zadovoljavajuće praćenje trajektorije. To se dešava u slučaju da je vozilo tokom realizacije prethodnog dijela putanje akumuliralo grešku i nije došlo u projektovanu poziciju iz koje treba da otpočne proceduru dokovanja.

Za stabilizaciju kretanja vozila po zadatoj putanji potrebno je projektovati povratnu spregu. Varijabla koja definiše ponašanje praćenja trajektorije je tzv. greška praćenja koja se definiše kao razlika između stvarne i zadate trajektorije:

$$e(t) = (y_1(t) - y_{1,r}(t), y_2(t) - y_{2,r}(t)) \quad (9)$$

Ne postoji samo jedan način da se projektuje regulator asimptotskog praćenja trajektorije. Ovdje će biti opisan jedan specifičan metod [11]. Praćenje trajektorije se može riješiti ako se greške sistema $e_1(t)$ i $e_2(t)$ ponašaju kao dva nezavisna stabilna linearna oscilatora, tj:

$$\ddot{e}_i + 2\delta_i\omega_i\dot{e}_i + \omega_i^2 e_i = 0, \quad i = 1, 2 \quad (10)$$

sa pozitivnim prigušenjem δ_i i učestanošću ω_i , ili drugačije zapisano:

$$\begin{bmatrix} \ddot{e}_1 \\ \ddot{e}_2 \end{bmatrix} + K_1 \begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} + K_0 \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = 0 \quad (11)$$

Pogodnim izborom koeficijenata K_0 i K_1 greška će konvergirati ka nuli. Kao posledica toga, tačka Y će dostići referentnu trajektoriju. Prema tome, upravljanje zavisi od dinamike izabrane greške. To se razlikuje od uobičajenih metoda upravljanja, jer umjesto da izaberemo strukturu regulatora i poboljšavamo njegove karakteristike, biramo željenu dinamiku greške i na osnovu nje proračunavamo zakon upravljanja. Kako bi uspostavili relaciju između željene dinamike greške i modela objekta, prvo treba izraziti ubrzanje tačke $y = (y_1, y_2)$, koje se pojavljuje u \ddot{e} . Prema tome, jednačina (11) se može preformulisati:

$$\begin{bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \begin{bmatrix} \ddot{y}_{1,r} \\ \ddot{y}_{2,r} \end{bmatrix} - K_1 \begin{bmatrix} v \cos \theta - \dot{y}_{1,r} \\ v \sin \theta - \dot{y}_{2,r} \end{bmatrix} - K_0 \begin{bmatrix} y_1 - y_{1,r} \\ y_2 - y_{2,r} \end{bmatrix} \quad (12)$$

gdje su upotrijebljeni izrazi (1) i (2) umjesto stvarne brzine u $\dot{e} = \dot{y} - \dot{y}_r$. Diferencirajući jednačine (1) i (2) i zamjenom $\dot{\theta}$ sa (3) dobija se:

$$\begin{bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \underbrace{\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}}_{R(\theta)} \begin{bmatrix} \dot{v} \\ \frac{v^2}{l} \tan \varphi \end{bmatrix} \quad (13)$$

Kako desne strane (12) i (13) moraju biti jednake slijedi:

$$\begin{bmatrix} \dot{v} \\ \frac{v^2}{l} \tan \varphi \end{bmatrix} = R^{-1}(\theta) \left(\begin{bmatrix} \ddot{y}_{1,r} \\ \ddot{y}_{2,r} \end{bmatrix} - K_1 \begin{bmatrix} v \cos \theta - \dot{y}_{1,r} \\ v \sin \theta - \dot{y}_{2,r} \end{bmatrix} - K_0 \begin{bmatrix} y_1 - y_{1,r} \\ y_2 - y_{2,r} \end{bmatrix} \right) \quad (14)$$

Relacija (14) uključuje poziciju $y = (y_1, y_2)$ i orijentaciju θ , poznatu referentnu trajektoriju i njene izvode $y_r(t)$, $\dot{y}_r(t)$ i $\ddot{y}_r(t)$, kao i upravljačke ulaze v i φ . Takođe, pojavljuje se i vremenski izvod jednog od upravljačkih ulaza. Ovo znači da je prvi red jednačine (14) ustvari diferencijalna jednačina koja se može riješiti u regulatoru ako se mjere pozicija y i orijentacija θ . Za rješavanje ove diferencijalne jednačine prvog reda potreban je početni uslov $v(0)$, koji se može izraziti iz referentne trajektorije primjenom relacije $v(0) = \sqrt{\dot{y}_{1,r}^2(0) + \dot{y}_{2,r}^2(0)}$. Poslije rješavanja diferencijalne jednačine, vrijednost v može se iskoristiti za drugi red jednačine (14) kako bi se proračunao $\tan \varphi$, a samim tim i φ kao drugi upravljački ulaz.

Prema tome, algoritam ovakvog dinamičkog kontrolera sa povratnom spregom je:

$$\dot{v} = [\cos \theta \quad \sin \theta] \lambda \quad (15)$$

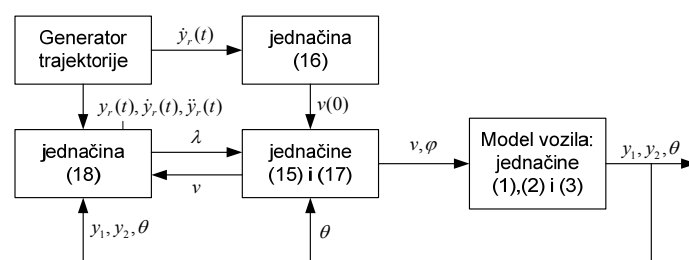
$$v(0) = \sqrt{\dot{y}_{1,r}^2(0) + \dot{y}_{2,r}^2(0)} \quad (16)$$

$$\varphi = \arctan \left(\frac{l}{v^2} [-\sin \theta \quad \cos \theta] \lambda \right) \quad (17)$$

gdje je $\lambda(v, y, \theta, y_r, \dot{y}_r, \ddot{y}_r)$ dato kao:

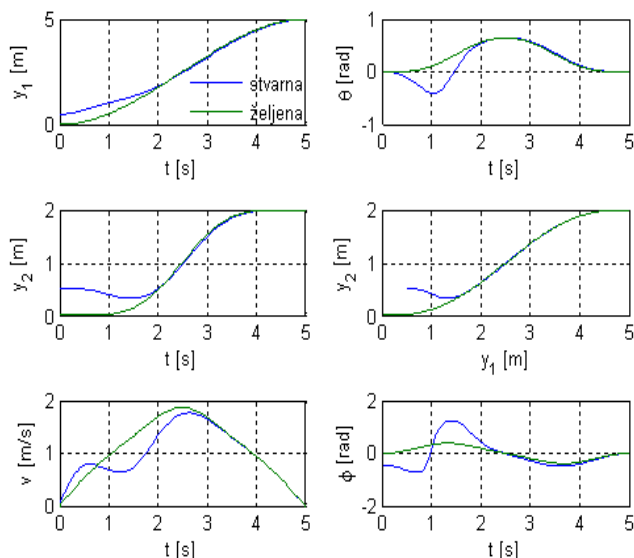
$$\lambda = \begin{bmatrix} \ddot{y}_{1,r} \\ \ddot{y}_{2,r} \end{bmatrix} - K_1 \begin{bmatrix} v \cos \theta - \dot{y}_{1,r} \\ v \sin \theta - \dot{y}_{2,r} \end{bmatrix} - K_0 \begin{bmatrix} y_1 - y_{1,r} \\ y_2 - y_{2,r} \end{bmatrix} \quad (18)$$

Struktura sistema upravljanja u zatvorenoj petlji prikazana je na slici 3.



Slika 3. Struktura sistema u zatvorenoj petlji

Rezultati simulacije prikazani su na slici 4. za iste parametre početnih i krajnjih uslova kao i u slučaju feed-forward upravljanja. Dinamika greške je opisana sa dvostrukim polom u $s = -2$ ($K_0 = 4, K_1 = 4$). U početnom trenutku stvarni položaj vozila je $(y_1, y_2, \theta) = (0.5, 0.5, 0)$.



Slika 4. Simulacija feedback upravljanja

Projektovani zakon upravljanja stabilizuje kretanje vozila, što se vidi sa slike 4. po konvergenciji varijabli sistema ka zadatim trajektorijama. Asimptotska stabilnost je ostvarena izborom asimptotski stabilne i linearne dinamike greške.

5. ZAKLJUČAK

U ovom radu prikazana je jedna od savremenih metoda upravljanja autonomnim vozilima i mobilnim robotima. Zasniva se na parametrizaciji svih sistemskih varijabli pomoću diferencijalno nezavisne varijable koju nazivamo flat izlaz.

Regulator koji treba implementirati je rezultat izabrane dinamike greške i modela objekta. Pri tome ne linearizuje se dinamika objekta, već dinamika greške. Ovakav pristup se može podijeliti u dva koraka. Prvi korak je izbor referentne trajektorije, a drugi izbor dinamike greške praćenja trajektorije flat izlaza i proračun zakona upravljanja.

Flatness metoda je jednostavna i efikasna. Međutim, to ne bi bila veoma važna metoda da postoji samo nekoliko akademskih primjera flat sistema. Činjenica je da postoji veliki broj matematičkih modela u tehnološkim upravljačkim sistemima koji su flat sistemi.

Analiza je sprovedena na osnovu kinematičkog modela vozila, pod pretpostavkom da se točkovi kotrljaju po podlozi bez klizanja i da nema bočnog klizanja točkova. U cilju podrobnije i tačnije analize ponašanja sistema neophodno je razmotriti primjenu predloženog postupka sa uključenom dinamikom vozila. Takođe, u nastavku istraživanja predviđena je analiza ponašanja sistema pri izboru drugih tipova dinamike greške koje će na zadovoljavajući način obezbijediti realizaciju kretanja autonomnog vozila po željenoj trajektoriji.

LITERATURA

- [1] T. Das and I.N. Kar, "Design and implementation of an adaptive fuzzy logic based controller for wheeled mobile robots," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 3, pp. 501–510, May 2006.
- [2] K. R. S. Kodagoda, W. S. Wijesoma, and E.K. Teoh, "Fuzzy speed and steering control of an AGV," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 1, pp. 112–120, Jan. 2002.
- [3] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics," *J. Robotic Syst.*, vol. 14, no. 3, pp. 149–163, 1997.
- [4] Y. J. Lee, J. H. Suh, J. W. Lee, and K. S. Lee, "Driving control of an AGV for an automated container terminal using an immunized PID controller based on cell-mediated immunity," *Artif. Life Robot.*, vol. 9, no. 2, pp. 90–95, 2005.
- [5] H. Martínez-Barberá and D. Herrero-Pérez, "Autonomous navigation of an automated guided vehicle in industrial environments," *Rob. Comput. Integr. Manuf.*, vol. 26, no. 4, pp. 296–311, 2010.
- [6] F. Rotella, F. Carrillo and M. Ayadi, "Digital flatness-based robust controller applied to a thermal process," *IEEE international Conference on Control application*, pp. 936–941, Mexico 2001.
- [7] R. Rothfuss, J. Rudolph and M. Zeitz, "Flatness based control of chemical reactor model," *European Control Conference*, pp. 637–642, Rome, September 1995.
- [8] A. Chelouah, E. Delaleau, P. Martin and P. Rouchon, "Differential flatness and control of induction motors," *symposium on Control, Optimization and Supervision; Computational engineering in system applications, IMACS Multiconference*, pp. 80–85, Lille, 9–12 July 1996.
- [9] J. Lévine, P. Rouchon, G. Yuan, C. Grebogi, B. Hunt, E. Ott, J. Yorke and E. Kostelich, "On the control of US navy cranes," *European Control Conference, ECC'97*, Brussels, July 1997.
- [10] M. Fliess, J. Lévine, Ph. Martin and P. Rouchon, "On differentially flat nonlinear systems," *IFAC-Synposium, NOLCOS'92* pp. 408–412, Borddeaux, 1992.
- [11] J. Rudolph, "Flatness Based Control - Introduction," on October 7–11, 2013

PRIVATNI OBLAK OTVORENOG KODA OPEN SOURCE PRIVATE CLOUD

Balša Femić, Centar informacionog sistema, Univerzitet Crne Gore
Stevan Šćepanović, Prirodno-matematički fakultet, Univerzitet Crne Gore

Sadržaj: Ovaj rad obrađuje temu računarstva u oblaku sa posebnim osvrtom na ownCloud, kao rješenje otvorenog koda za privatni oblak. Opisani su modeli usluga računarstva u oblaku (Infrastructure as a service - IaaS, Platform as a service PaaS, te Software as a service - SaaS), kao i modeli primjene računarstva u oblaku (privatni, javni i hibridni oblak). U praktičnom dijelu detaljno je opisan ownCloud kao privatni oblak otvorenog koda, instalacija istog i njegove mogućnosti. OwnCloud se pokazao kao dobro rješenje za pojedince i firme koji žele da imaju isključivu kontrolu nad svojim podacima.

Abstract: This particular work deals with subject of Cloud computing with special review on an ownCloud as an open source solution for a private cloud. Models of offers in a field of cloud computing are described (such is Infrastructure as a service - IaaS, Platform as a service - PaaS, Software as a service - SaaS), as well as models of usage (private, public and hybrid cloud). In practical part ownCloud is fully described as an open source private cloud, installation and its advantages. OwnCloud is shown as a good solution for individuals and companies who want to have an absolute control over their data.

1. UVOD

Razvojem informacionih tehnologija neprekidno se javlja potreba za unapređenjem trenutnog stanja informacionih sistema. Računarstvo u oblaku, (*Cloud computing*) je nastalo kao rezultat potrebe za povećanjem kapaciteta i novih mogućnosti na postojećim infrastrukturama bez investiranja u novu infrastrukturu i kadrove. Računarstvo u oblaku je omogućilo korisnicima dodatne računarske resurse koristeći samo *web browser*.

Što je uopšte oblak? Iako nije riječ o jednom, jedinstvenom i strogo definisanom konceptu, oblak je lako definisati i prepoznati - riječ je o bilo kojem sistemu ili usluzi, koja podatke ne pohranjuje na klijentskom uređaju niti jedinstvenom serveru, već više servera (ili opštije - računarskih resursa) spojenih mrežom (obično Internetom). Umjesto da se podaci nalaze na jednom, strogo „fizički“ definisanom računaru, oni su raštrkani ili preko čitavog niza računara, koji se mogu nalaziti unutar skupa servera, u susjednim zgradama ili na različitim lokacijama širom planete Zemlje.



Slika 1. Računarstvo u oblaku

2. OSNOVE RAČUNARSTVA U OBLAKU

Težnja za inovacijama i neprekidan razvoj informacionih tehnologija doveli su do nastanka računarstva u oblaku. Javila se potreba za većim kapacitetima i novim mogućnostima bez ulaganja u postojeću infrastrukturu i obuku kadrova.

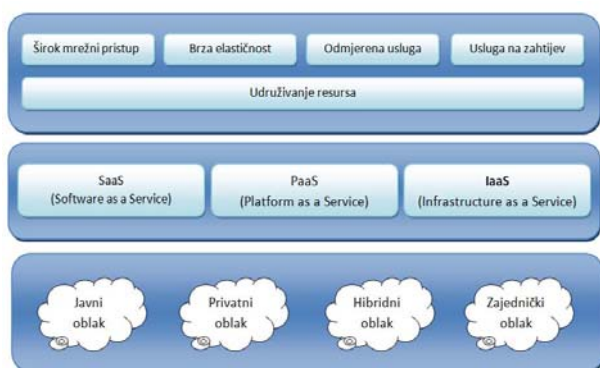
Pored računarstva u oblaku postoje klijent/server i *peer-to-peer* modeli koji su razvijeni prije računarstva u oblaku. U računarskom svijetu, u razdoblju do 1980-ih godina sve je funkcionisalo pomoću klijent/server modela. Sve računarske aplikacije, svi podaci i kontrola izvršavala se preko servera. Ukoliko je korisnik htio pristupiti određenim podacima ili pokrenuti program, morao se spojiti na server kako bi dobio pristup podacima odnosno programu. Korisnici su se spajali na serverski računar pomoću terminala tj. klijent računara. Nije bio moguć pristup više od jednog korisnika na server u isto vrijeme. Model klijent/server iako sličan konceptu računarstva u oblaku nije nudio kvalitetnu korisničku okolinu zbog ograničenosti upravljanja serverskog računara.

Potreba spajanja jednog računara sa drugim rezultirala je razvojem *peer-to-peer* (P2P) tehnologije. P2P tehnologija predstavlja mrežnu arhitekturu u kojoj svaki računar ima jednake mogućnosti i odgovornosti. Dizajnirana je kao suprotnost klijent/server tehnologiji u kojoj je više računara zavisilo od serverskog računara. U P2P okruženju svaki računar je u isto vrijeme i klijent i server. Prepoznajući jednako sve računare u mreži, P2P tehnologija omogućila je direktnu razmjenu resursa i usluga, te više nije bilo potrebe za jednim centralnim serverom.

Potreba da više korisnika zajednički radi na jednom dokumentu dovela je do pojave računarstva u oblaku. Kako bi mogli saradivati na jednom projektu, korisnici najprije

moraju imati osiguranu međusobnu komunikaciju (tekstualnu, zvučnu ili putem videa). Korisnici moraju imati mogućnost međusobne izmjene podataka kako bi se ostvario zajednički način rada. [1]

Američka vladina organizacija koja se bavi standardima i tehnologijom - NIST (engl. National Institute of Standards and Technology) se između ostalog bavi i računarstvom u oblaku. Tačnije, NIST se bavi definisanjem računarstva u oblaku i podjelom na karakteristične modele. Oblak model promovise dostupnost i sastoji se od pet ključnih karakteristika, tri modela pružanja usluga, te četiri modela implementacije koji su navedeni na slici 2. [2]



Slika 2. Ključne karakteristike, modeli pružanja usluga, modeli implementacije računarstva u oblaku

Postoji pet ključnih karakteristika koje pokazuju odnos i razlike sistema zasnovanog na računarstvu u oblaku u odnosu na tradicionalni pristup u računarstvu. Tih pet ključnih karakteristika su:

- **Pružanje usluge na zahtjev korisnika** (engl. *On-demand self-service*) - Korisnik može samostalno odabrati i pokrenuti računarske resurse. Može birati vrijeme korišćenja i mrežni prostor za smještanje podataka bez potrebe za interakcijom s zaposlenima provajdera. Ovaj model računarstva u oblaku pomaže u podržavanju izvršnih i kapacitivnih aspekata objekata koji zavise od nivoa usluge. „Plati po korišćenju“ priroda računarstva u oblaku se može smatrati kao najam opreme koja se plaća zavisno od toga koliko je opreme, na koje vrijeme i s kojim uslugama iznajmljeno.

- **Širok mrežni pristup** (engl. *Broad network access*) - Mogućnosti su dostupne putem mreže i njima se pristupa koristeći standardne mehanizme koji promovisu heterogenu upotrebu „siromašnijih“ i/ili „bogatijih“ klijentskih platformi (na primjer, mobilni uređaji, laptopovi te PDA uređaji) kao i tradicionalnih programskih usluga temeljenih na „oblaku“.

- **Udruživanje resursa** (engl. *Resource pooling*) - Računarski resursi provajdera spajaju se kako bi uslužili sve korisnike koristeći model više zakupljenih jedinica (engl. *Multi-Tenant model*), sa različitim fizičkim i virtuelnim resursima, koji se dinamički dodjeljuju i uklanjaju prema zahtjevima korisnika.

- **Brza elastičnost** (engl. *Rapid elasticity*) - Mogućnosti koje korisnicima nudi računarstvo u oblaku mogu biti ubrzano i elastično pokrenute, u nekim slučajevima i automatski, kako bi se po potrebi ostvarilo proporcionalno povećanje ili smanjenje mogućnosti kada one više nijesu potrebne.

- **Odmjerena usluga** (engl. *Measured service*) - Sistemi koji koriste računarstvo u oblaku automatski provjeravaju i optimizuju upotrebu resursa. Upotreba resursa se može pratiti, provjeravati i o njoj se mogu raditi izvještaji pružajući tako transparentan uvid provajderima usluge i korisnicima.

Način pružanja usluge računarstva u oblaku podijeljen je na tri arhitekturna modela i različite izvedene kombinacije. Tri osnovne klasifikacije često se nazivaju SPI model, pri čemu SPI označava program, platformu ili infrastrukturu (engl. *Software, Platform, Infrastructure*).

- **SaaS** (engl. *Cloud Software as a Service*) - Korisniku je pružena mogućnost upotrebe dostupnih aplikacija koje se nalaze u infrastrukturi oblaka. Aplikacije su dostupne s različitim klijentskih uređaja uz pomoć klijentskog interfejsa (na primjer *web browsera*). Pri tome korisnik ne provjerava infrastrukturu pozadine, uključujući mrežu, servise, operativne sisteme, smještaj podataka. Ovaj model računarstva u oblaku dostavlja jednu aplikaciju preko korisničkog browsera hiljadama korisnika koji koriste arhitekturu predviđenu za mnogo zakupa.

- **PaaS** (engl. *Cloud Platform as a Service*) – PaaS je varijacija SaaS strukture koja kao uslugu donosi razvojni interfejs. Korisnik sam gradi vlastite aplikacije koje se pokreću na infrastrukturi provajdera. Aplikacije se korisnicima dostavljaju preko interfejsa servera putem Interneta. Navedeni serveri su u vlasništvu provajdera.

- **IaaS** (engl. *Cloud Infrastructure as a Service*) – Korisniku je kao usluga pružena mogućnost korišćenja računarske infrastrukture (uglavnom virtuelne platforme). Korisnici ne kupuju servere, programe, prostore za smještaj podataka ili mrežnu opremu, već kupuju navedene resurse kao spoljnu uslugu.

Nezavisno od modela pružanja usluga (SaaS, PaaS ili IaaS) postoje četiri različita modela implementacije usluga računarstva u oblaku. Ta četiri modela su izvedena na četiri različita načina, zavisno od specifičnih potreba.

- **Javni oblak** (engl. *Public Cloud*) – platforma računarstva u oblaku dostupna i otvorena za javnost, nezavisno o tome radi li se o pojedincima ili organizacijama. U vlasništvu je kompanije koja prodaje usluge računarstva u oblaku. U slučaju javnih platformi postavlja se pitanje sigurnosti vlastitih podataka. Aplikacije različitih korisnika često se nalaze na istim serverima, sistemima za skladištenje i mrežama. Ako je javni oblak realizovan s pažnjom usmjerenom na izvođenje, sigurnost i položaj podataka druge aplikacije pokrenute na oblaku ne bi trebale praviti

probleme arhitekturi oblaka i krajnjim korisnicima. Jedna od prednosti javnih oblaka je da oni mogu biti puno veći nego što mogu biti privatni oblaci. Dijelovi javnog oblaka mogu biti i isključivo za upotrebu samo jednog korisnika, čineći tako privatni centar podataka.

- **Privatni oblak** (engl. *Private Cloud*) – infrastruktura računarstva u oblaku dostupna je isključivo jednoj organizaciji. Njome može upravljati sama organizacija ili neko drugi. Organizacije koriste privatne oblake kada trebaju ili žele veći nadzor nad podacima nego što ga mogu imati korištenjem javnog oblaka. Privatni oblaci su napravljeni isključivo za upotrebu jednog klijenta, pružajući mu najveći nadzor nad podacima i najveću sigurnost podataka smještenih na oblaku. Organizacija posjeduje infrastrukturu i ima nadzor nad raspodjelom aplikacija na vlastitoj infrastrukturi.

- **Zajednički oblak** (engl. *Community Cloud*) - Nekoliko organizacija dijeli strukturu oblaka. Infrastruktura podržava posebne zajednice koje imaju zajedničke potrebe, misije, zahtjeve sigurnosti i slično. Njima mogu upravljati same organizacije ili neko drugi (provajder).

- **Hibridni oblak** (engl. *Hybrid Cloud*) - Strukturu oblaka čine dva ili više različitih oblaka (privatni, zajednički ili javni) koji ostaju jedinstveni entiteti, ali su međusobno povezani standardizovanim ili prikladnim tehnologijama koje omogućuju efikasan prenos podataka ili aplikacija. Hibridni oblaci povezuju javne i privatne modele oblaka.

Korišćenjem računarstva u oblaku moguće je izbjeći velike troškove kupovine skupih servera, programa i usluga. Korisnici usluga računarstva u oblaku plaćaju samo ono što koriste. Uglavnom ne postoje zahtjevi za plaćanje unaprijed, a troškovi su jako mali u odnosu na korišćenje vlastite IT infrastrukture

Danas postoji veliki broj korporacija koje pružaju neki oblik računarstva u oblaku. Najpoznatiji su Google, Amazon i Microsoft. Svaka od njih nudi dio koji je besplatan i dio koji se plaća. Besplatni dio obično sadrži ograničene, jednostavne usluge, dok se proširenja resursa dobijaju plaćanjem određene svote novca.

3. OWN CLOUD

OwnCloud je softver otvorenog koda koji omogućava pokretanje privatnog oblaka na vlastitom serveru. Projekat je pokrenuo njemački programer softvera otvorenog koda (engl. *open source*) Frank Karlitschek u januaru 2010. godine i zamišljen je kao besplatna alternativa komercijalnim rješenjima slične namjene (na pr. Dropbox). Ono što ownCloud izdvaja od komercijalnih rješenja je mogućnost instalacije na sopstvenim, lokalnim serverima.

OwnCloud posjeduje napredne funkcionalnosti i lijep korisnički interfejs. Projekat je multiplatformski. Baziran je na PHP-u i SQLite, MySQL ili PostgreSQL bazi, i radiće na bilo kojoj platformi koja zadovoljava ove uslove. [3] OwnCloud je razvijen pod AGPLv3 licencom (slobodna

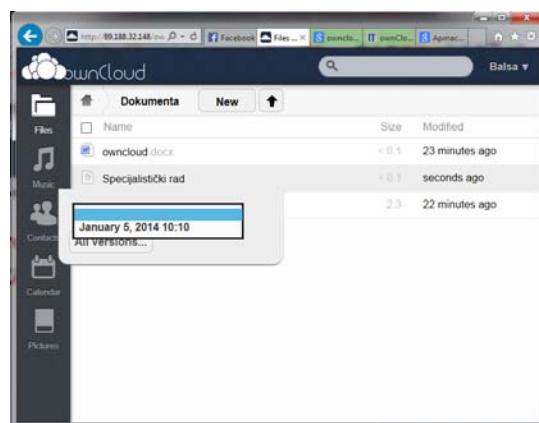
softverska licenca, GNU Affero General Public License, version 3, izdata od *Free Software Foundation* u Novembru 2007.) i može se preuzeti sa adrese <http://owncloud.org>.

Sopstvenim fajlovima, direktorijumima, kontaktima, foto galerijama, kalendaru itd, može se pristupiti sa personalnog računara ili sa mobilnog telefona

Sa računara ownCloud-u se može pristupiti preko aplikacije ili preko web browsera. Zavisno od operativnog sistema postoje klijentske aplikacije za Windows operativne sisteme, Mac OS i Linux.

Svi podaci sa ownCloud servera su sinhronizovani sa bilo kojim klijentom koji podacima prilazi bilo da je u pitanju *web browser*, desktop ili mobilni klijent. Dijeljenje podataka sa drugima je veoma jednostavno. Sam korisnik određuje koje fajlove će dijeliti i koji korisnici odnosno koje grupe korisnika im mogu prilaziti, kao i to da li imaju prava izmjene ili samo pregleda fajlova.

Jako bitna funkcija koju pruža ownCloud je verzionisanje. Data funkcionalnost omogućava čuvanje starijih verzija fajla, tako da u slučaju neželjene promjene na fajlu uvijek možemo vratiti neku od predašnjih verzija. Sam podatak o broju verzija koje se čuvaju je konfigurabilan i zavisi od korisnika.



Slika 3. Verzionisanje na ownCloud-u

Enkripcija se može se opciono uključiti, tako da svi podaci bivaju enkriptovani. To je posebno pogodno, ako koristimo prostor za skladištenje podataka van ownCloud servera. Naravno, nevezano od enkripcije korisničkih podataka, uvijek se može koristiti i SSL konekcija kao dodatna mjera zaštite.

Od mogućnosti koje pruža ownCloud trebalo bi izdvojiti još: teme (potrebom ove funkcionalnosti može se dizajnirati izgled ownCloud klijenta prema potrebama korisnika), pregled ODF fajlova, kalendar, tasks, fajl notifikacija (prilikom dijeljenja fajlova ownCloud se može tako podesiti, da svaki korisnik koji bi trebao da ima prava pristupa datom fajlu bude obaviješten o tome), **galerije** (unaprijedena je kontrola deljenja fotogalerija tako što je omogućeno kategorisanje izborom ownCloud direktorijuma i dijeljenje na principu mail adresa koje imaju prava pregleda).

4. INSTALACIJA I ADMINISTRIRANJE OWNCLOUD SERVERA

Instalacija ownCloud servera moguća na svim distribucijama Linux-a, bilo kao operativnog sistema za PC, bilo kao servera, kao i na operativnim sistemima Windows 7 i Windows 8, te na Windows serveru 2008 ili mlađim verzijama Windows servera.

Da bi zadovoljili sistemske zahtjeve softvera, podrazumijevano je da na sistemu bude instaliran Apache web server kao i PHP verziju 5.3 (ili noviju) sa podrškom za json , xml , zip i gd (paketi php5-json, php-xml, php-mbstring, php5-zip, php5-gd).

OwnCloud se oslanja na WebDAV, što ne zahtijeva da web server posjeduje WebDAV podršku, već sam ownCloud dolazi sa internim WebDAV serverom. Koristeći WebDAV ekstenziju postižu se funkcionalnosti web kolaboracije, rada sa dokumentima, kao i smještanje samih dokumenata na web server prostor za skladištenje.

Za potrebe ovog rada, instaliran je ownCloud server, verzija 5.0.12, na desktop serveru na kojem je Ubuntu server 12.04.

Jednom kada je instaliran ownCloud server na za to predviđenju mašinu, administriranje istim je dosta jednostavno. Upravljanje ownCloud-om, te dodavanje raznih dodatnih funkcionalnosti i mogućnosti, izmjena postojećih postiže se korišćenjem API -ja (Aplikacioni programski interfejs (engl. *Application programming interface*)).

Moguća je i integracija ownCloud-a sa LDAP (*Lightweight Directory Access Protocol*) serverom ili Active Directory instancom, tako da svi korisnici i grupe definisani na LDAP ili *Active Directory* serveru postaju validni ownCloud korisnici. Ili, drugim riječima LDAP server je zadužen za autentifikaciju ownCloud korisnika.

OwnCloud već dolazi sa setom od preko pedesetak predefinisanih aplikacija, koje se mogu omogućiti ili onemogućiti po potrebi. Ukoliko nijedna od predefinisanih aplikacija ne zadovoljava specifične poslovne potrebe korisnika, uvijek može razviti interno svoju aplikaciju korišćenjem API-ja. Neki od predefinisanih aplikaciju koje su podrazumijevano omogućene su: kalendar, tekst editor, aplikacija za verzionisanje, aplikacije za predled .odf i .pdf fajlova, dijeljenje fajlova itd.

Od funkcionalnosti vezanih za administraciju izdvajaju se:

Eksterni skladišteni prostor. Mogućnost neoslanjanja samo na disk prostor ownCloud servera, već korišćenje eksternih kapaciteta za skladištenje podataka. Ukratko, eksterni prostor za skladištenje se mapira u lokalni ownCloud virtuelni fajl sistem. Kao eksterni skladišteni prostor mogu se koristiti: lokalni fajl sistem, FTP , WebDAV, OpenStack Swift i SMB .

Logovanje. Integracija ownCloud bilježenja servisnih aktivnosti u postojeći sistemski log. Pored ove mogućnosti ownCloud može da koristi i sopstveni sistem logovanja.

Migracija i backup. Data funkcionalnosti omogućava laku migraciju korisničkih naloga sa jedne instance ownCloud-a na drugu, kao i pojednostavljeno kreiranje rezervnih kopija podataka.

5. ZAKLJUČAK

Računarstvo u oblaku se svakodnevno sve više i više koristi u privatne, edukacijske i poslovne svrhe. Svoj potpuni potencijal će dosegnuti kada se uklone sigurnosni problemi i moguća zavisnost od jednog provajdera. Računarstvo u oblaku je razvijano desetinama godina istraživanja u virtuelizaciji, računarstvu, mreži i softverskim uslugama. Orjentiranost prema krajnjem korisniku, odlična fleksibilnost, smanjeni troškovi i brojni servisi garantuju uspjeh tehnologije. Iako je tehnologija još u razvoju, svakodnevno se pojavljuju novi servisi i mogućnosti u okviru ponude računarstva u oblaku. Isto će svoj puni kapacitet ostvariti u bliskoj budućnosti.

U radu je poseban osvrt napravljen na ownCloud, kao privatni oblak otvorenog koda koji ne zaostaje za komercijalnom konkurencijom ni kada je u pitanju pitanje sigurnosti, ni jednostavnost korišćenja. Naročito je bitno što je ownCloud softver otvorenog koda. Softver otvorenog koda je softver koji se može koristiti, kopirati, mijenjati i redistribuirati bez ikakvih restrikcija. Temelji se na načelu slobode upotrebe, izmjene, dopune, poboljšanja / unapređivanja i naknadne distribucije programskih rješenja.

OwnCloud krasi napredne funkcionalnosti i lijep korisnički interfejs. Ukoliko se neka kompanija odluči za ownCloud kao privatni oblak, neće osjetiti razliku između vlastitog i komercijalnih rješenja, a podaci će ostati dostupni isključivo korisnicima, tj. bit će pohranjeni na serveru unutar kompanije, što uveliko povećava sigurnost podataka. Iz pomenutih razloga, ownCloud rješenje je primamljivo kako za pojedince tako i za firme. Završavam sa konstatacijom da je velika prednost ovog rješenja mogućnost instalacije na lokalnoj infrastrukturi, pri čemu se ne zavisni od javnih servisa.

LITERATURA

- [1] M.Miller, „Cloud Computing Web Based Applications That Change the Way You Work and Collaborate Online“, Que Publishing, 2009.
- [2] NCERT-PUBDOC-2010-03-293, Nacionalni CERT, www.cert.hr, Nacionalno središte za sigurnost računalnih mreža i sustava u saradnji sa LS&S, www.LSS.hr, Laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.
- [3] <http://owncloud.org>

PREDLOG POBOLJŠANJA MASTER-SLAVE ALGORITMA ZA RASPODJELU OPTEREĆENJA U MPI PARALELNIM APLIKACIJAMA A PROPOSAL FOR MASTER-SLAVE ALGORITHM IMPROVEMENTS FOR LOAD BALANCING IN MPI PARALLEL APPLICATIONS

Luka Filipović, *Centar Informacionog Sistema Univerziteta Crne Gore*
Božo Krstajić, *Elektrotehnički fakultet Univerziteta Crne Gore*

Sadržaj: *U radu je prezentovan modifikovani master-slave algoritam za poboljšanje raspodjele opterećenja u paralelnim aplikacijama. Upotrebom predloženog algoritma smanjuje se vrijeme izvršavanja i povećava stepen iskorištenja distibuiranih računarskih resursa.*

Abstract: *This paper presents modified master-slave algorithm for improvement of load balancing in parallel applications. Use of the proposed algorithm causes a reduction in execution time of parallel applications and increase of utilization of parallel computer resources*

1. UVOD

U poslednjih deset godina, trendovi koji se zasnivaju na sve bržim računarskim mrežama, distribuiranim računarima i višeprocorskim računarskim arhitekturama ukazuju na to da je paralelizam budućnost računarstva. Paralelno računarstvo se smatra najmodernijom granom u IT svijetu čiji je razvoj podstaknut simulacijama kompleksnih sistema [1]. Paralelno programiranje se bazira na konceptu ubrzanja izvršenja programa dijeljenjem programa u više dijelova koji se mogu izvršavati simultano na odvojenim serverima. Glavni razlozi za korištenje paralelnog procesiranja predstavljaju ušteda vremena, rješavanje većih problema i omogućavanje konkurentnosti [2].

Postoji više modela paralelnog programiranja, koji definišu različite nivoe paralelizma kao i različite metode komunikacije između računara koji učestvuju u paralelnoj obradi. MPI (Message Passing Interface) predstavlja jedan od konkretnih programskih modela za razvijanje aplikacija sa paralelnim programiranjem. Njegov glavni zadatak je slanje poruka u domenu paralelne obrade podataka i koristi se kao medijum za komunikaciju. Standardni MPI je izvorno dizajniran za pisanje aplikacija i biblioteka za okruženje sa distribuiranom memorijom. MPI pruža funkcije i rutine za razmjenu potrebnih informacija u homogenom i heterogenom računarskom okruženju; [3].

U realnom distribuiranom okruženju (HPC klaster, Grid klaster ...) opterećenost resursa varira i nije uvijek moguće dobiti na korištenje resurse koji su potpuno slobodni ili jednako opterećeni. Trajanje izvršavanja paralelne aplikacije iznosi koliko i trajanje izvršavanja dijela aplikacije na najsporijem jezgru ili grupi jezgara. Pri tome se mogu desiti veliki gubici zbog toga što paralelna aplikacija čeka najsporije jezgro ili grupu jezgara da bi završila rad. Ova pojava se dešava na klasterima koji se sastoje od servera različite snage (heterogene klasterne) ili klasterima sa promjenljivim opterećenjem, tj. klasterima na kojima više

korisnika u isto vrijeme izvršava paralelne aplikacije i time opterećuje resurse.

U radu će biti prezentovan master-slave algoritama za balansiranje opterećenja (*load balancing*) aplikacija paralelizovanih MPI tehnologijom i predložena njegova modifikovana verzija. Na primjeru paralelne verzije CQ Simulatora [4][5] biće upoređeni principi rada i rezultati paralelizacije aplikacije sa ravnomjernom raspodjelom procesa i aplikacija koje koriste standardni i unaprijedni master-slave algoritam. Testiranje je vršeno na HPCG klaster-u koji se nalazi u Institutu za informacione i komunikacione tehnologije u Sofiji, Bugarska. Klaster se sastoji od 36 servera sa 576 jezgara (dual Intel Xeon X5560 @ 2.8Ghz) objedinjenih u Blade klaster HP Cluster Platform Express 7000 [6].

2. MPI PROGRAMSKI MODEL

MPI [3] predstavlja jedan od najrasprostranjenijih programskih modela za razvijanje aplikacija uz paralelno programiranje. Za specifikaciju MPI modela [7] odgovoran je MPI forum, otvorena grupa koja ima članove iz velikog broja drugih organizacija. Pozitivne karakteristike ovaj model nameću kao standard za komunikaciju među procesima u paralelnom programiranju sa distribuiranim memorijskim modelom.

MPI se najčešće primjenjuje u računarskim klasterima i superračunarima. Po programskom modelu spada u modele prenosa poruka, a prema arhitekturi sistema je namenjen MIMD sistemima. Prilikom razvoja MPI kao cilj su postavljene visoke performanse, dobru skalabilnost i prenosivost. MPI se u vreme pisanja izdvaja kao dominantan model široko prihvaćen i korišćen u računarskim sistemima visokih performansi. Zvanični jezici interfejs standarda su Fortran, C i C++, a MPI implementacije se mogu naći i za C#, Java, Python, Perl i druge jezike.

3. MASTER-SLAVE ALGORITAM ZA PRERASPODJELU OPTEREĆENJA U MPI APLIKACIJAMA

Vrijeme trajanja paralelne aplikacije jednako je trajanju najdužeg procesa, odnosno procesa na najsporišem jezgru. Za vrijeme trajanja aplikacije jezgra su u aktivnom režimu rada ili u stand-by režimu, čekajući da ostala jezgra završe rad. Ukoliko se utvrdi da određeno jezgro ili grupa jezgara duži vremenski period u toku izvršavanja paralelne aplikacije čekaju ostala da završe procese, odnosno rade u stand by režimu, neophodno je izvršiti preraspodjelu zaduženja i optimizaciju procesa izvršavanja. Optimizaciju izvršavanja paralelne aplikacije moguće je izvršiti korištenjem *load balancing* algoritama prilikom pokretanja pojedinačnih procesa. Najzastupljeniji algoritam za *load balancing* paralelnih aplikacija je master-slave algoritam.

Paradigma master-slave obuhvata dva tipa procesora ili jezgara. Na master jezgrima se obavlja *preprocessing*, raspoređivanje procesa i *postprocessing*, dok su slave jezgra zadužena za izvršavanje procesa [8]. Master jezgro, u predloženoj slučaju, na početku izvršavanja paralelne aplikacije generiše listu procesa koje je potrebno izvršiti i šalje jednu ili više instrukcija slave jezgrima. Po završetku svog dijela zadatka slave jezgra signaliziraju master jezgru kraj rada dodijeljenih procesa, a ono im nakon primanja obavještenja šalje instrukcije o sledećim procesima koje treba da izvrše. Ova rutina se ponavlja sve dok se svi procesi ne izvrše. Prednost algoritma se ogleda u dobrom raspoređivanju procesa. Mana algoritma se ogleda u povećanoj komunikaciji između master i slave jezgara i mogućeg čekanja slave jezgra na dodjelu novog seta procesa. Na master jezgru nije moguće izvršavati procese, tako da se i to ističe kao mana ovog algoritma pogotovo kod izvršavanja na manjem broju jezgara. U slučaju izvršavanja aplikacije na 64 jezgra gubitak iznosi svega 1,5%, dok bi gubitak pri izvršavanju na 8 jezgara bio čak 12,5% ukoliko se algoritam pokreće od početka rada aplikacije.

4. MODIFIKOVANI MASTER-SLAVE ALGORITAM

Modifikovanim algoritmom je nastojano da mane koje posjeduje master-slave algoritam svede na minimum i da se poveća efikasnost aplikacije. U predloženoj modifikaciji, paralelni program se od početka izvršava po principu ravnomjerne raspodjele procesa opterećujući sva jezgra uz obavljanje samo neophodne komunikacije između procesa. U trenutku završetka svih procesa na najbržem jezgru šalje se signal svim jezgrima za prekid operacija. Po završetku procesa na svim koji su se izvršavali u trenutku slanja signala prelazi se na master-slave algoritam. Sva jezgra šalju master jezgru izvještaje o procesima koji su im bili dodijeljeni metodom ravnomjerne raspodjele, a koji nisu izvršeni. Master jezgro šalje instrukcije svim slave jezgrima i do kraja vrši monitoring po standardnom master-slave algoritmu.

U idealnom slučaju, kada je izvršavanje procesa na klasteru ravnomjerno i sva jezgra završe dodijeljene procese

u isto vrijeme, modifikovani algoritam se pokreće, verifikuje da su svi procesi izvršeni i obustavlja rad paralelne aplikacije. U tome slučaju su gubici svedeni na par sekundi tokom kojih jezgra šalju izvještaje o završenim procesima.

Predloženim algoritmom je komunikacija među jezgrima koja se tiče preraspodjele poslova svedena samo na kritični period aplikacije, tj poslije kraja rada najbržeg jezgra. Smanjeni su gubici usled neizvršavanja procesa na master jezgru zbog pokretanja algoritma tek u trenutku kada je to neophodno.

5. REZULTATI SIMULACIJA

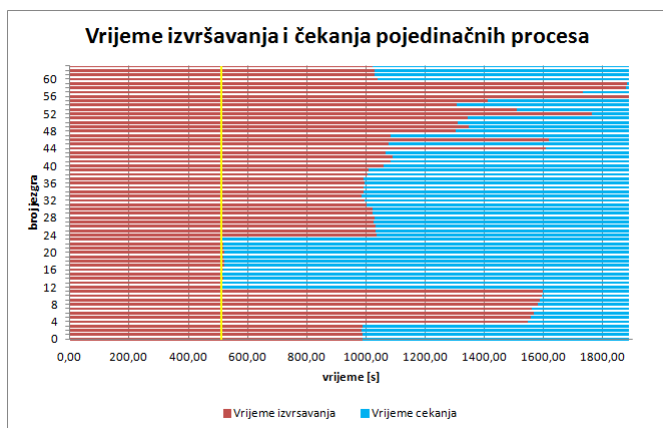
Prezentovani algoritmi optimizacije prilagođeni su paralelnim aplikacijama koje se sastoje od više nezavisnih procesa. Testirani su na primjeru simulatora performansi CQ komutatora paketa, a mogu biti primijenjeni na sve paralelne aplikacije koje se sastoje od više nezavisnih procesa. Predstavljeni su rezultati izvršavanja aplikacije paralelizovane metodom ravnomjerne raspodjele broja procesa po jezgrima, kao i paralelizacije uz upotrebu opisanih algoritama.

Komutatori paketa sa baferima u ukrsnim tačkama komutacione matrice (Crosspoint Queued - CQ) analizirani su u kasnim osamdesetim, ali uglavnom za uniformni saobraćaj i veoma male veličine bafera. Zbog tehnoloških ograničenja prilikom praktične realizacije, u to doba je bilo jako teško implementirati bafere većeg kapaciteta zajedno sa komutatorom, na istom čipu. Nedavno je ova arhitektura aktuelizovana, jer je pokazano da današnja tehnologija omogućava implementaciju velikih bafera unutar komutacione matrice. Prednost rješenja je odsustvo kontrolne komunikacije između linijskih kartica, u kojima se nalaze baferi, i raspoređivača paketa. Koristeći simulator, analiziran je saobraćaj i ispitivan je uticaj veličine redova čekanja (bafera) kao i način njihovog upravljanja, na najvažnije performanse koje se koriste za ocjenu rada komutatora: propusnost (*throughput*), srednje kašnjenje ćelija (*average cell latency*), vjerovatnoća gubitka ćelija (*cell loss probability*) i varijacija u kašnjenju ćelija [4].

Aplikacija za analizu performansi CQ komutatora sastoji se iz dva dijela – generatora uniformnog saobraćaja i simulatora CQ komutatora paralelizovanih uz pomoć MPI tehnologije. [5]. Generator saobraćaja generiše uniformni saobraćaj za 32 različita ulazna opterećenja, odnosno vjerovatnoće pojavljivanja paketa. Simulator vrši analizu za 12 različitih veličina bafera (broj ćelija koje je moguće u njemu smjestiti), 8 algoritama (LQF, RR, ERR, FBRR, EELQF, ELQF, FBLQF i RAND) i 32 fajla generisanog uniformnog saobraćaja. Time se dobijaju 3072 nezavisna procesa koja se uz pomoć mpi-ja pokreću na različitim jezgrima. Testiranje je vršeno za matricu 16x16 i 1.000.000 zahtjeva (time slot-ova). U osnovnoj verziji paralelne aplikacije su procesi ravnomjerno dodjeljivani jezgrima i mjereni rezultati izvršavanja. Ovaj model paralelizacije se pokazao kao najefikasniji u idealnim slučajevima, pri

izvršavanju na slobodnim resursima ili jednako opterećenim resursima.

Sa ciljem dobijanja rezultata u što kraćem vremenskom interval ispitivanja su vršena na 64 jezgra i to na 4, 8 i 16 jezgara po serveru koje se rezervišu prilikom pokretanja aplikacije. Dobijena su vremena izvršavanja koja su varirala od slučaja do slučaja. Izvršena je analiza vremena izvršavanja procesa na svakom jezgru, a ne samo na najsporijem od kog zavisi kraj izvršavanja aplikacije. U većini slučaja je primijećeno veliko odstupanje između trajanja najbržeg i najsporijeg procesa. U nastavku rada će biti analiziran jedan od slučajeva izvršavanja na 64 jezgra, korištenjem 16 servera sa po 4 aktivna jezgra i primjena algoritama na izvršavanje aplikacije.



Slika 1. Raspodjela izvršavanja procesa CQ Simulatora na 64 jezgra, prikazano vrijeme aktivnog rada jezgara i vremena čekanja na primjeru izvršavanja koji je trajao 1889,9 s na 16 servera sa po 4 aktivna jezgra po serveru.

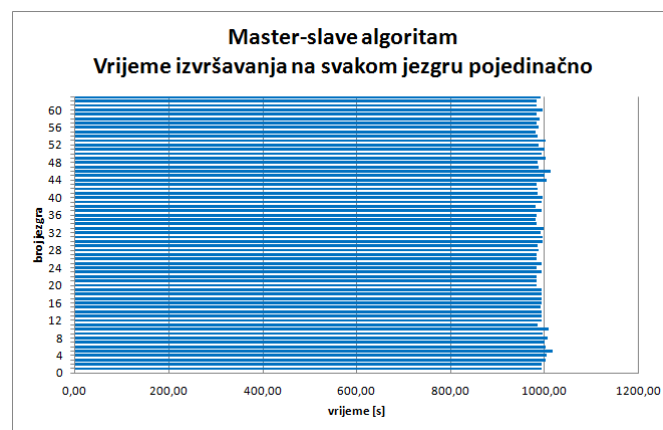
Na slici 1 crvenom bojom prikazano je vrijeme izvršavanja pojedinačnih procesa. Ukupno vrijeme izvršavanja, odnosno najsporijeg jezgra, iznosilo je 1889,86 s, dok najbrže jezgro završilo svoje procese za 512,68 s (označeno žutom linijom). Vrijeme izvršavanja dodijeljenog seta procesa na svakom jezgru je variralo u zavisnosti od trenutne opterećenosti servera na kome se jezgro nalazi, kao i opterećenosti storage-a sa kog se učitavaju ulazni fajlovi. Plavom bojom označeno je vrijeme koje svako jezgro provede u stand-by režimu čekajući da sva jezgra završe svoje procese i da se okonča rad aplikacije. Kod ovog primjera procesorsko vrijeme iznosi 33,57 CPU sati, dok je vrijeme korisnog rada procesora 19,84 CPU sati, tj. svega 59%. Trajanje pojedinačnih procesa je variralo od 9,71 s do 46,95 s.

Master-slave algoritmi opisani u glavi 3 i 4 testirani su u identičnim uslovima kao MPI primjer sa ravnomjernom raspodjelom procesa iz poglavlja 3. Vremena izvršavanja aplikacije uz paralelizovane metodom ravnomjerne raspodjele broja procesa po jezgrima i paralelizacije sa primjene algoritama predloženih u glavi 3 i 4 prikazana su u tabeli 1.

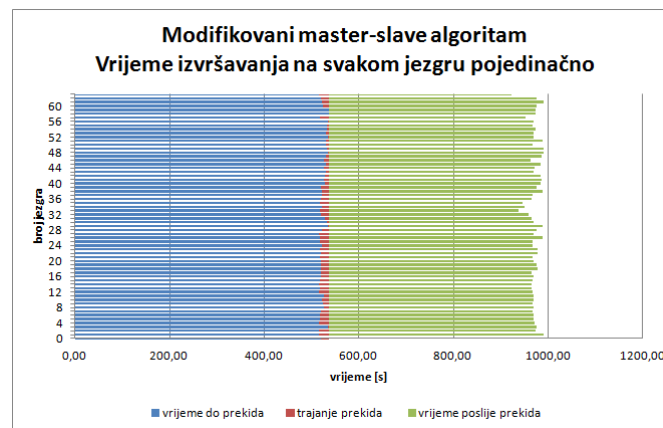
MPI	Master-slave algoritam	Modifikovani master-slave algoritam
1889,86	1016,85	991,69

Tabela 1. Vrijeme izvršavanja aplikacije uz pomoć predoženih algoritama

Analizom rezultata izvršavanja aplikacije utvrđena je značajna ušteda u vremenu izvršavanja prilikom korištenja predloženih algoritama. Vrijeme izvršavanja je smanjeno sa 1889,86 s na 991,69 - 1016,85s, a vrijeme čekanja jezgara da ostala završe svedeno na minimum. Raspodjela zauzetosti jezgara prilikom korištenja osnovnog master-slave algoritma prikazana je na slici 2, a modifikovanog algoritma na slici 3.



Slika 2. Raspodjela izvršavanja procesa po jezgrima tokom vremena uz korištenje master-slave algoritma



Slika 3. Raspodjela izvršavanja procesa po jezgrima tokom vremena uz upotrebu modifikovanog master-slave algoritma

Prilikom izvršavanja obje verzije je nultom jezgru dodijeljena funkcija master jezgra. Master-slave algoritmom je skraćeno vrijeme izvršavanja na 1016,85 s. Utrošeno je 18,06 CPU sati uz stepen iskorištenja 96,07 %.

Izvršavanje aplikacije uz modifikovani master-slave algoritma je trajalo 991,69 s, uz utrošak od 17,63 CPU sati i iskoristivost od 96,29%. Izvršavanje je proteklo u tri faze. U prvoj fazi aplikacija je radila sa svim aktivnim jezgrima i ravnomjerno raspoređenim procesima 515,92 s, kada je svoje procese završilo najbrže jezgro i poslalo signal ostalim jezgrima. U drugoj fazi su preostala jezgra po prijemu signala i kraja trenutnih procesa poslala master jezgru izvještaj o

odrađenim i neodrađenuim procesima. Proces zaustavljanja, slanja podataka i obračuna na master jezgru je trajao 0,70 - 21,08 s, odnosno 0,23 CPU sati (1,32 %). Maksimalno vrijeme prekida aplikacije iznosi maksimalno trajanje pojedinačnih procesa koji se izvršavaju na jezgrima u trenutku slanja signala za prekid izvršavanja. Zaustavljanje procesa i sinhronizacija su označeni crvenom bojom na grafikonu br 3. Nakon toga je izvršavanje aplikacije nastavljeno uz master-slave algoritam i nakon 454,69 s su izvršeni svi procesi. Gubici usled neizvršavanja procesa na master jezgru su smanjeni sa 0,28 CPU sati iz osnovnog master-slave primjera na 0,12 CPU sati kod izvršavanja modifikovanog primjera. Ova ušteda bi bila izražajnije kod primjera gdje se *load-balancing* algoritam izvršava kraće, odnosno gdje najbrže i najsporije jezgro završavaju kraćem vremenskom intervalu.

6. ZAKLJUČAK

U radu je predložen modifikovani master-slave algoritam za raspodjelu opterećenja u MPI paralelnim aplikacijama. Izvršeno je poređenje predloženog algoritma sa sa master-slave algoritmom, koji vrši raspodjelu tokom cijelog rada aplikacije, i algortimom koji vrši ravnomjernu raspodjelu procesa i idealan je za izvršavanje na računarskim klasterima.

Numeričkim primjerom je potvrđena ušteda u vremenu izvršavanja prilikom korištenja *load balancing-a*, pogotovo pri korištenju modikovanog algoritma koji samo vrši raspodjelu tokom kritičnog dijela aplikacije. Potvrđeno je da prekid aplikacije znatno ne utiče na ukupno vrijeme izvršavanja.

LITERATURA

- [1] D. Peleg, "Distributed Computing: A Locality-Sensitive Approach", pp 1-11, Society for Industrial and Applied Mathematics (SIAM), 2000, ISBN 0-89871-464-8
- [2] A. S. Tanenbaum, M. van Steen, "Distributed Systems: Principles and Paradigms", 2nd edition, 2007, ISBN 0132392275 / 9780132392273
- [3] W. Gropp, E. Lusk, A. Skjellum, "Using MPI-2: Advanced Features of the Message Passing Interface", MIT Press, 1999, ISBN 0-262-57133-1.
- [4] M. Radonjić, "Prilog analizi performansi CQ komutatora paketa sa stanovišta veličine i algoritama upravljanja redovima čekanja", doktorska disertacija, 2011.
- [5] L. Filipović, "Optimizacija simulatora CQ komutatora paketa metodom paralelnog programiranja", IT 2012, Zabljak 2012.
- [6] A. Mishev, E. Atanassov, "Infrastructure overview and assessment", HP-SEE project deliverable D 5.2, pp. 21-24, <http://www.hp-see.eu/files/public/HPSEE-WP5-MK-001-D5.2-f-2011-08-31.pdf>
- [7] D. W. Walker. "Standards for message-passing in a distributed memory environment", Technical report, Oak Ridge National Laboratory, August, 1992. First CRPC Workshop on Standards for Message Passing in a Distributed Memory Environment, April 1992.
- [8] S Sahn, G Vairaktarakis, " The master-slave paradigm in parallel computer and industrial settings", Journal of Global Optimization, April 1977, ISBN 0925-5001.

HIBRIDNA MPI/PTHREADS PARALELIZACIJA ZA ESTIMACIJU PARAMETARA WEIBULL DISTRIBUCIJE

HYBRID MPI/PHTHREAD PARALLELIZATION OF THE WEIBULL DISTRIBUTION PARAMETERS ESTIMATOR

Sidita Duli, *Department of Computer Sciences, Faculty of Natural Sciences, University of Shkoder, Albania*

Sadržaj: *U radu je predstavljena implementacija hibridnog programiranja korištenjem MPI i Pthreads tehnologije za estimaciju parametara Weibull distribucije za dati uzorak podataka. Komunikacija između servera se obavlja korištenjem MPI-ja, dok se za komunikaciju jezgara za dijeljenom memorijeom koristi Pthreads. Cilj rada je provjera ubrzanja korištenjem hibridne paralelizacije i poređenje sa tehnikama paralelizacije, korištenjem MPI i POSIX, na primjeru procjene parametara Weibull distribucije za velike ulazne skupove podataka.*

Ključne riječi: *paralelno programiranje, Weibull distribucija, MPI, Pthread*

Abstract: *In this paper it is presented the hybrid programming implementation of MPI and Pthreads to estimate the parameters of Weibull distribution for a given data sample. Message passing communication is managed by MPI to achieved coarse grained parallelism between nodes, and shared memory communication is implemented by Pthreads. The aim of this article is to verify the speedup of the hybrid MPI/Pthread version, and to compare it to the techniques of parallelizing the code, the MPI and the POSIX, in estimating the Weibull parameters of large datasets.*

Keywords: *parallel programming, Weibull distribution, MPI, Pthread*

1. INTRODUCTION

Weibull Distribution is a statistical tool used to model the wind speed. The wind speed data collected in a weather station tend to have the form of a Weibull distribution. The shape of the graphics depends on the parameters of the distribution, which are estimated by the set of wind speed data. Once the Weibull distribution can be used to calculate the probability of a particular wind speed at a particular location, it can be used to work out the number of hours per year that certain wind speeds are likely to record and therefore the likely total power output of a wind turbine per year [1].

Processing voluminous datasets is highly computation intensive. Parallel computing has been fruitfully employed in numerous application domains to process large datasets and handle other time-consuming operations of interest. As a result, unprecedented advances have been made in such areas as biology, scientific computing, modeling and simulations, and so forth.[2]

The principle of parallel programming is to divide the problem in smaller pieces and to distribute the calculations between many processors. In this way can be achieved a faster result than the same code executed in one processor. There are different ways of parallelization the code of this application. Two of them are the message passing interface (MPI) and Posix threads (Pthread). The aim of this research is to represent a hybrid implementation of the parameter estimation of Weibull parameters.

2. METHODS

Two methods used in implementing the parallel version of the estimation of Weibull parameters are MPI and Pthreads.

MPI is a standard developed by the Message Passing Interface Forum (MPIF). It specifies a portable interface for writing message-passing programs, and aims at practicality, efficiency, and flexibility at the same time. MPI is implemented on a great variety of machines, including those "machines" consisting of collections of other machines, parallel or not, connected by a communication network [3].

The programs using MPI libraries may run on distributed-memory multicomputer, shared-memory multiprocessors, networks of workstations, and combinations of all of these. Posix is an international standard with an exact definition and a set of assertions which can be used to verify compliance. A conforming Posix application can move from system to system with a very high confidence of low maintenance and correct operation [4]. The Posix thread library provides an interface to generate and interact with separate threads of execution within a program. This standard is defined by the IEEE and is available across nearly all variants of the UNIX operating systems [5]. The hybrid version MPI/Pthread is implemented by creating two threads for each process involved in the calculations. This number of threads is chosen as the minimum used to form a hybrid version. The tests will show how this minimum number of threads added per processor will affect in the performance of the implemented version.

The application takes a censored set of data, which might be ordered or unordered, taken from a sample of N data. It is

given the location parameter of the Weibull distribution. The data sample and the location parameter are read from a file by each process that is created. The algorithm makes an estimation of the scale parameter and shape parameter of this distribution.

Also for the density function of Weibull distribution is calculated the mode, the mean and the variance for the given set of data. This procedure is repeated R times for different set of data, taken from this N data sample.

3. RESULTS AND DISCUSSIONS

Tests are performed with data samples of 10 000 000 values and 5 000 000 values. The Weibull parameters are calculated for 100 such samples, taken randomly from a bigger amount of data, from 15 000 000. For each of these samples, there are calculated the mode, the mean and the variance of the distribution. The performance is tested for different number of processors, beginning with the serial version with only one processor running a doing the calculations, up to twelve parallel processors doing the calculations. Tests are performed for each implementation, Pthread, MPI and Hybrid version with two threads.

No of Cores	Hybrid 2 threads	MPI version	Pthread Version
2	6421	6022	5960
4	3209	3098	2947
6	2195	2048	1998
8	1632	1567	1502
10	1315	1292	1213
12	1294	1220	1022

Table 1: Time of execution of different version for the sample containing 10 000 000 data.

Table 1 shows the time spent to estimate the Weibull parameters of a dataset containing 10 000 000 data. Tests are made for hybrid two Pthreads and one process, up to two Pthreads and 6 processes. For large dataset, both versions reduce the time spent to estimate the result.

Results show that the time spent to perform the calculations is reduced while increasing the number of cores. For this estimation, hybrid version executed with two threads and one process takes 6421 seconds. For the same dataset, this version takes 1294 second when executed with two threads and six processes. These results are compared with those of MPI version executed with the same number of processes as cores, and those of Pthread version executed with the same number of threads as cores. As it is shown in the table 1, for this range of dataset, the Pthread version needs less time to perform the estimation. Pthread version takes 5960 seconds to estimate the parameters while it is executed with two threads, and 1022 seconds while it is executed with twelve threads.

The results of this research show that the Pthread version is the fastest among pure MPI and hybrid 2 thread MPI/Pthread. Referenced to the previous test made for Pthread version, this is explained because of the cost of managing threads. Compared to the cost of creating and managing a process, a

thread can be created with much less operating system overhead. Managing threads requires fewer system resources than managing processes. [6]

From table 1 we can conclude that MPI pure version nor hybrid MPI/Pthread is not better than Pthreads in terms of execution time [7].

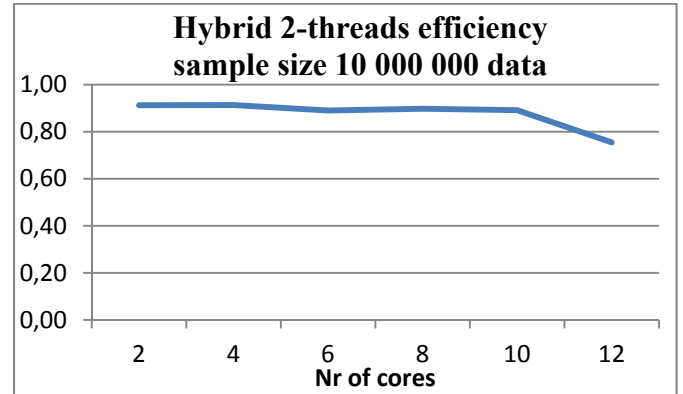


Fig 1. Efficiency hybrid MPI/Pthread version when data sample contains 10 000 000 data.

The figure 2 represents the efficiency of the hybrid MPI/Pthread version. Results show that the efficiency is relatively high when the number of cores is small. Adding new processes after the five process decreases the efficiency. Increasing the number of processes, the communication overhead becomes outstanding and the parallel performance degrades. The hybrid MPI/Pthreads programming mode could obtain a good parallel performance when the processes' number is larger by reducing the communication between processes.

The same implementation is executed applying different sample containing 5 000 000 data.

No of Cores	Hybrid 2 threads	MPI version	Pthread Version
2	3249	3078	2938
4	1618	1585	1471
6	1137	1120	997
8	819	806	741
10	765	752	625
12	729	712	560

Table 2: Time of execution of different parallel versions for a sample with 5 000 000 data.

Results show that hybrid version executed with two threads and one process takes 3249 seconds for estimating the Weibull parameters of a sample containing 5 000 000 data. For the same dataset, this version takes 729 second when executed with two threads and six processes. Results are compared with those of MPI version executed with the same number of processes as cores, and those of Pthread version executed with the same number of threads as cores. For this range of dataset, the Pthread version performs faster the calculations. Pthread version takes 2938 seconds to estimate the parameters while it is executed with two threads, and 560 seconds while it is executed with twelve threads.

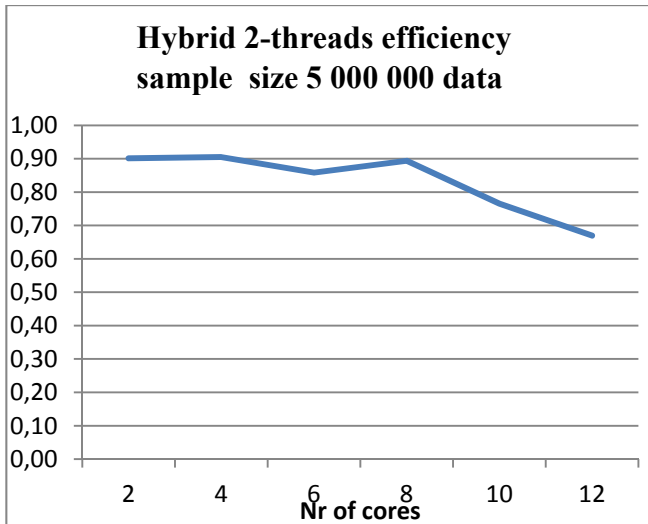


Fig 2. Efficiency hybrid MPI/Pthread version when data sample contains 5 000 000 data.

Figure 2 shows the graphical presentation of the efficiency of the hybrid 2 threads MPI/Pthreads version. Results show that for a small number of processes involved in the parallel calculations the efficiency is high. With increasing the number of processes the parallel performance begins to decrease. The graphic shows that the efficiency is still high when there are involved four processes. They are all combined with 2 threads each, distributed in eight cores, give a good performance.

Both figure 1 and figure 2 show that for a large dataset the efficiency stays high for bigger number of cores, respectively ten cores. For a smaller dataset, in figure 2, the efficiency is high only until the number of eight cores.

4. CONCLUSIONS

The hybrid MPI/Pthread might be as a technique of parallelization. This research analyzes the performance of the implemented hybrid 2 threads MPI/Pthread version of calculating the Weibull distribution parameters for two different given samples. The hybrid implementation performs the calculations in less time, as the number of processes involved is increased. But for the range of cores from two up to twelve, results show that this technique is not the best one

for parallelization of this estimation. The hybrid 2 threads MPI/Pthread needs more time to perform calculations than the pure MPI version. Also the hybrid is slower than the pure Pthread version. This is due to communication needed between processes.

The efficiency of the hybrid 2 threads MPI/Pthread falls when the number of cores is higher than eight, for a sample 5 000 000. For the sample containing 10 000 000 data, the efficiency falls when the number of cores is higher than ten. As future work is analyzing the hybrid 4 threads MPI/Pthreads and comparing the results with this research. Also, it will be paid more attention to other combinations thread/process, comparing each other pair and concluding to a best combination which performs better the calculations.

REFERENCES

- [1] <http://www.reuk.co.uk/Wind-Speed-Distribution-Weibull.htm>
- [2] S.Rajasekaran and J. Reif, "Algorithms" in *Handbook of Parallel Computing Models, Algorithms and Applications*, USA, Chapman & Hall/CRC, 2008
- [3] Y.AOYAMA, J. NAKANO: *Practical MPI Programming, International Technical Support Organization*, 1999
- [4] D. A. Lewine, "POSIX Programmer's Guide", O'Reilly & Associates, 1991
- [5] .D. R BUTENHOF: *Programming with Pthread*, 1997
- [6] S.Duli, B Krstajić, "Pthread in a Weibull distribution parameters estimator", International Conference Information Tehnology IT-2012, 2012
- [7] Huailiang Xuan, Yue Hu, Weiqin Tong, Zhixun Gong and Yan Hou, "Hybrid programming implementation of BiCG method on SMP cluster architecture", IEEE/ACIS 11 International Conference on Computer and Information Science, 2012

POBOLJŠANJE ENERGETSKE EFIKASNOSTI OFDM-CDMA SISTEMA IMPROVING ENERGY EFFICIENCY OF OFDM-CDMA SYSTEM

Uglješa Urošević, Zoran Veljović, *Elektrotehnički fakultet u Podgorici*

Sadržaj - U ovom radu je predložena MISO šema za poboljšanje energetske efikasnosti OFDM-CDMA downlink sistema sa pilot tonom i optimalnim TDC kombinovanjem. BER performanse su analizirane na osnovu originalno razvijenog simulacionog modela u slučaju Rice-ovog fedinga. Predloženi sistem koristi prostorno-vremenski blok kod primijenjen na dvije, tri i četiri predajne antene. Simulacioni rezultati pokazuju da se predloženom šemom ostvaruju značajno bolje BER performanse i veća spektralna efikasnost, uz istovremeno izbjegavanje implementacije optimalnog TDC kombinovanja kao veoma kompleksne metode za povraćaj ortogonalnosti podnosilaca na mjestu prijema.

Abstract - In this paper, MISO scheme is proposed as a solution for improving energy efficiency of the OFDM-CDMA downlink system with pilot tone and threshold detection combining (optimum TDC). BER performance is evaluated through originally developed simulation model in the case of Ricean frequency elective fading. The proposed system uses space-time block coding applied to two, three and four transmit antennas. The simulation results show that the proposed scheme provides significantly better BER performance and greater spectrum efficiency, while at the same time implementation of a very complex subcarriers orthogonality restoration technique at the receiver side, i.e. optimum TDC is avoided.

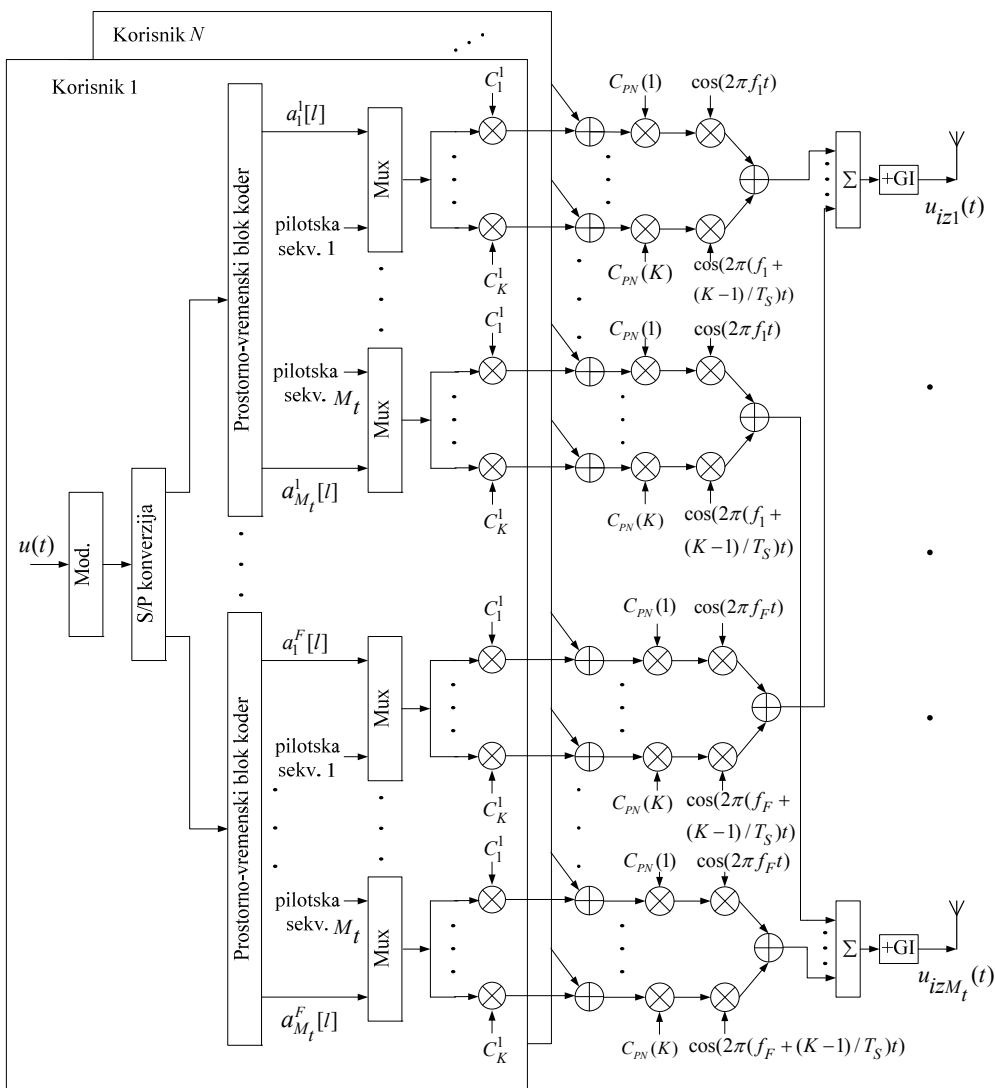
1. UVOD

Buduće širokopojasne mobilne radio-mreže treba da udovolje zahtjevima korisnika koji se ogledaju u stalnom povećanju brzine prenosa, odnosno zahtjevima za sve većim kapacitetom downlink-a. Prenos posredstvom višestrukih podnosilaca (MC - Multicarrier) predstavlja atraktivno rješenje za prenos podataka veoma velikim brzinama. Pri tome, povećanje kapaciteta downlink-a može biti ostvareno kombinacijom MC modulacije i kodnog multipleksa (CDMA - Code Division Multiple Access), [1]-[3]. Ukoliko su upotrebljeni podnosioci ortogonalni, ova kombinacija se naziva OFDM (Orthogonal Frequency Division Multiplexing) - CDMA, [4]-[5]. U slučaju frekvencijski selektivnog fedinga primljeni signal je pod uticajem frekvencijske distorzije, pa dolazi do narušavanja ortogonalnosti podnosilaca i do interferencije između pojedinih korisnika (MUI – Multi User Interference). Radi smanjenja MUI interferencije, a samim tim i broja pogrešno primljenih simbola, neophodno je u prijemniku izvršiti povraćaj ortogonalnosti OFDM podnosilaca. U [1] je pokazano da se optimalnim TDC (Threshold Detection Combining) kao tehnikom kombinovanja na prijemnoj strani postižu bolji rezultati u odnosu na ORC (Orthogonal Restoring Combining) i CEC (Controlled Equalization Combining) kombinovanje. Ipak u pogledu kompleksnosti znatno je zahtjevnije optimalno TDC kombinovanje, pošto je za svaku pojedinačnu vrijednost odnosa signal/šum (SNR – Signal to Noise Ratio) potrebno pronaći optimalnu vrijednost težinskog koeficijenta, na svakom podnosiocu, uz koji se ostvaruje minimalni BER. U slučaju izraženog Doppler-ovog pomjeraja sistem u [1] ima vrlo loše BER performanse. Šema iz [2] je predložena za poboljšanje BER performansi OFDM-CDMA sistema sa pilot tonom i optimalnim TDC kombinovanjem u slučaju izraženog frekvencijskog ofseta.

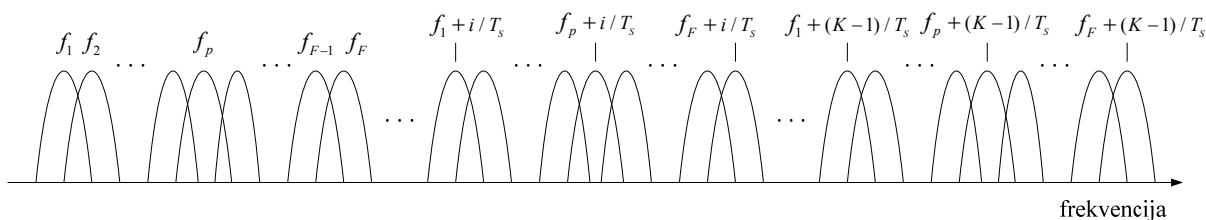
Ipak, u sistemu iz [2] BER performanse opadaju sa smanjivanjem broja podnosilaca, tj. za pouzdaniji prenos je potrebno zauzeti veći frekvencijski opseg, čime se smanjuje spektralna efikasnost. U ovom radu je predložena MISO (Multiple Input Single Output) OFDM-CDMA šema kao solucija za poboljšanje performansi sistema iz [2]. Pokazuje se da predloženi sistem omogućava značajno manji BER. Takođe, predloženi sistem čak i sa manjim brojem podnosilaca, tj. većom spektralnom efikasnošću omogućava manji BER. Pored veće pouzdanosti i veće spektralne efikasnosti još jedna prednost predložene šeme je i izbjegavanje primjene vrlo kompleksnog optimalnog TDC kombinovanja. Rad je organizovan na sledeći način. U drugom poglavlju je prikazan predloženi MISO OFDM-CDMA sistem. Opis originalno razvijenog simulacionog modela i rezultati simulacije su dati u trećem poglavlju. Na kraju je dat zaključak.

2. MODEL SISTEMA

Model predajnika predloženog MISO OFDM-CDMA sistema za downlink komunikaciju je prikazan na slici 1. Pretpostavljeno je da u komunikaciji u jednočelijskom sistemu učestvuje N korisnika. Informacioni biti datog korisnika, koji dolaze na ulaz predajnika se prvo mapiraju saglasno odgovarajućoj modulacionoj šemi. Nakon konverzije iz serije u paralelu (S/P), na svakoj paralelnoj grani ($p=1, \dots, F$; F -broj paralelnih grana) prostorno-vremenski blok koder sakuplja blok od M sukcesivnih simbola i mapira ih u sekvencu od L uzastopnih vektora $\mathbf{a}^p[l] = [a_1^p[l] \dots a_M^p[l]]^T$, $0 \leq l < L$. Kodni odnos je $R_c = M/L$. Smatra se da je tokom trajanja jedne kodne riječi mobilni radio-kanal nepromjenljiv, [6].



Slika 1. Blok šema predajnika MISO OFDM-CDMA sistema za downlink komunikaciju



Slika 2. Distribucija podnosilaca u frekvencijskom domenu

Sekvenca pilot tonova je implementirana na svakom izlazu svih prostorno-vremenskih blok koda i multipleksirana sa kodiranim simbolima podataka. Nakon spredovanja u frekvncijskom domenu korišćenjem ortogonalne spreading sekvence dužine K , $C_k^n = \{-1, 1\}$, $k=1, \dots, K$, signali koji potiču od različitih korisnika se sabiraju. Tako formirani signal se kodira pseudo-slučajnom sekvencom (PN – Pseudo Noise) $c_{PN}(j) = \{-1, 1\}$, $j=1, 2, \dots, R$, $R \gg K$, koja se koristi za smanjenje uticaja MUI interferencije izazvane destrukcijom ortogonalnosti OFDM podnosilaca. Dodatno, u celularnim sistemima se ova sekvenca može iskoristiti za smanjenje interferencije od

strane korisnika iz drugih ćelija. Ortogonalni podnosioci su organizovani u F grupa. U svakoj grupi je K podnosilaca sa separacijom $1/T_s$ (T_s - trajanje mapiranog simbola). Separacija između podnosilaca f_p ($p=1, \dots, F$) koji pripadaju različitim grupama iznosi $1/FT_s$. Na slici 2 je data distribucija podnosilaca u frekvencijskom domenu. Suma od FK moduliranih podnosilaca formira OFDM-CDMA simbol trajanja FT_s . Ovako formiranom OFDM-CDMA simbolu se dodaje zaštitni interval (GI – Guard Interval) trajanja T_{guard} , i formira se rezultatni simbol trajanja $T = FT_s + T_{guard}$.

3. SIMULACIONI MODEL

Za određivanje BER performansi predloženog MISO OFDM-CDMA sistema je razvijen originalni simulacioni model. Pretpostavljeno je ukupno M_t SISO (*Single-Input Single-Output*) mobilnih radio-kanala sa Rice-ovim fadingom i AWGN-om. Za datu vrijednost Rice-ovog parametra izraženog u dB (K_{dB}), maksimalnu vrijednost vremenskog širenja simbola na mjestu prijema (T_{max}) i srednju kvadratnu vrijednost ovog širenja (T_{rms}), simulacioni model generiše jednu direktnu i više reflektovanih putanja. Realizacija ortogonalnih podnosilaca je ostvarena upotrebom inverzne brze Fourier-ove transformacije. U simulacionom modelu se koristi QPSK modulaciona šema, Walsh Hadamard kod, $T_{max} = 150$ ns i $T_{rms} = 150/7$ ns. Razmatrani sistem koristi prostorno-vremenski blok kod primijenjen na dvije, tri i četiri predajne antene, [7]:

$$\mathbf{X}_2 = \begin{bmatrix} a_1 & -a_2^* \\ a_2 & a_1^* \end{bmatrix} \quad (1)$$

$$\mathbf{X}_3 = \begin{bmatrix} a_1 & -a_2 & -a_3 & -a_4 & a_1^* & -a_2^* & -a_3^* & -a_4^* \\ a_2 & a_1 & a_4 & -a_3 & a_2^* & a_1^* & a_4^* & -a_3^* \\ a_3 & -a_4 & a_1 & a_2 & a_3^* & -a_4^* & a_1^* & a_2^* \end{bmatrix} \quad (2)$$

$$\mathbf{X}_4 = \begin{bmatrix} a_1 & -a_2 & -a_3 & -a_4 & a_1^* & -a_2^* & -a_3^* & -a_4^* \\ a_2 & a_1 & a_4 & -a_3 & a_2^* & a_1^* & a_4^* & -a_3^* \\ a_3 & -a_4 & a_1 & a_2 & a_3^* & -a_4^* & a_1^* & a_2^* \\ a_4 & a_3 & -a_2 & a_1 & a_4^* & a_3^* & -a_2^* & a_1^* \end{bmatrix} \quad (3)$$

\mathbf{X}_2 , \mathbf{X}_3 i \mathbf{X}_4 su kodne riječi za sisteme sa dvije, tri i četiri predajne antene, respektivno. Vrste predstavljaju simbole koji se šalju sa pojedinih antena, dok kolone čine simboli koji se šalju u određenim signalizacionim intervalima. U predloženoj šemi estimirana vrijednost koeficijenta mobilnog radio-kanala na poziciji k -tog podnosioca za p -tu paralelnu granu, između i -te predajne i prijemne antene za n -tog korisnika je $\tilde{H}_{n,i}^p(k)$. Nakon prostorno-vremenskog blok dekodiranja težinski koeficijent za rekonstrukciju ortogonalnosti podnosilaca n -tog korisnika na poziciji k -tog podnosioca, za p -tu paralelnu granu je:

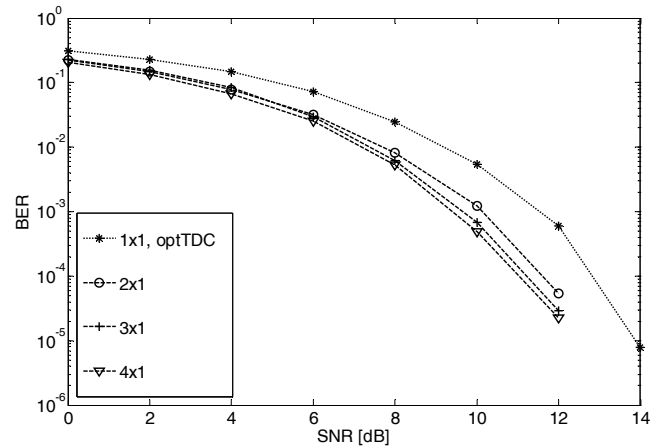
$$w_n^p(k) = \frac{1}{\sum_{i=1}^{M_t} |\tilde{H}_{n,i}^p(k)|^2} \quad (4)$$

Simulacioni parametri su dati u tabeli 1.

Tabela 1 - Simulacioni parametri

Modulacija	QPSK
Kapacitet sistema	200 Mb/s
Broj korisnika	64
Broj paralelnih grana	4
Separacija između podnosilaca	$1/(4T_S)$
Rice-ov parametar (K_{dB})	4 dB

BER performanse predloženog MISO OFDM-CDMA sistema i OFDM-CDMA sistema iz [2] su analizirane i upoređene u različitim uslovima, opisanim sa dva simulaciona scenarija. U prvom scenariju je zanemaren uticaj Doppler-ovog pomjeraja dok je u drugom scenariju uzet u obzir i njegov uticaj na BER performanse. U zavisnosti od prijemnog SNR-a BER performanse za predloženu šemu ($M_t = 2, 3$ i 4 predajne antene) i sistema iz [2], uzetog kao konkurentski sistem, su prikazane na slici 3. Za oba sistema je broj podnosilaca po paralelnoj grani 128. Pretpostavljeno je da je frekvencijski ofset, kao posledica Doppler-ovog pomjeraja, zanemarljiv. Sa slike 3 se može uočiti da se predloženom šemom postižu značajno bolji rezultati, što je naročito izraženo u području srednjeg i većeg SNR-a. Očigledno je da se predloženom šemom ostvaruje značajno manji BER i u slučaju primjene samo dvije predajne antene. Na primjer, za vrijednost BER-a od 10^{-4} , predloženi sistem sa $M_t = 2$ predajne antene, u poređenju sa konkurentskim modelom, redukuje potrebni SNR za oko 1,5 dB, dok u slučaju sa $M_t = 4$ predajne antene ova SNR redukcija iznosi oko 2 dB.



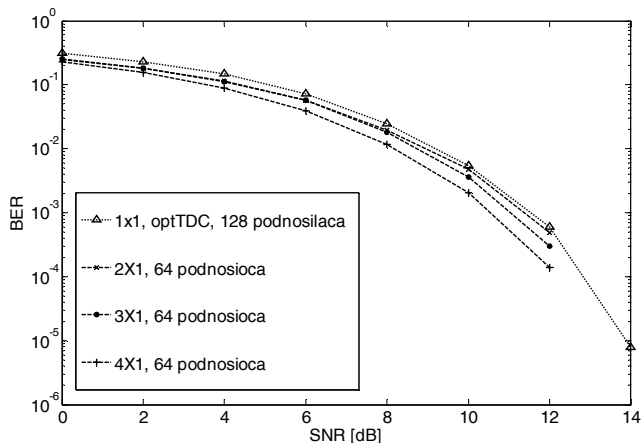
Sl. 3 - BER performanse predloženog sistema ($M_t = 2, 3$ i 4) i sistema iz [2]

Na slici 4 su prikazane BER performanse predloženog sistema ($M_t = 2, 3$ i 4 predajne antene) i šeme iz [2]. Pretpostavljeno je da se 64 podnosilaca po paralelnoj grani koriste u predloženoj šemi, dok je broj podnosilaca po paralelnoj grani u konkurentskom sistemu 128. I u ovom slučaju je pretpostavljen zanemarljiv Doppler-ov pomjeraj. Sa slike 4 se vidi da iako se u predloženom sistemu koristi dva puta manji broj podnosilaca bolje BER performanse se postižu u poređenju sa konkurentskim sistemom.

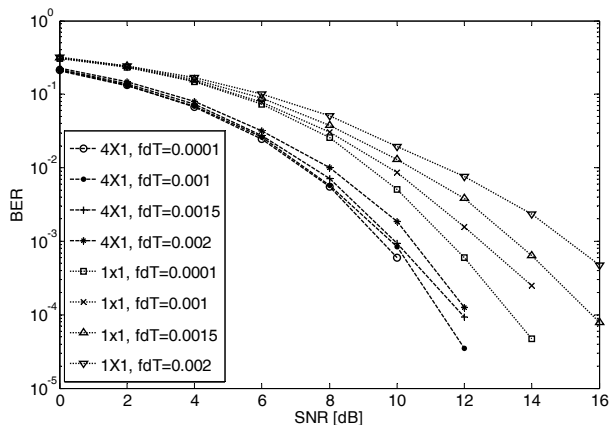
Za različite vrijednosti normalizovanog maksimalnog Doppler-ovog pomjeraja ($f_d T = 0.0001, 0.001, 0.0015$ i 0.002) BER performanse predložene šeme sa $M_t = 4$ i sistema iz [2] su prikazane na slici 5. Broj podnosilaca po paralelnoj grani za oba sistema je 128.

Da predloženi sistem i u uslovima Doppler-ovog pomjera omogućava bolju energetska i spektralnu efikasnost je prikazano na slici 6. Prikazane su BER performanse predloženog modela sa $M_t = 4$, uz 64 podnosilaca po

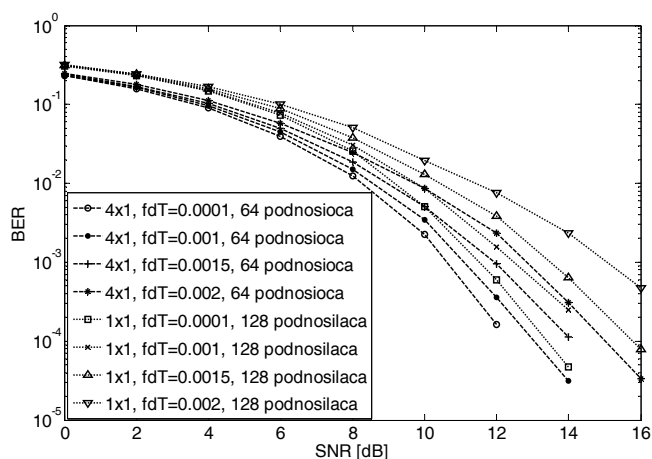
paralelnoj grani i sistema iz [2] uz 128 podnosilaca po paralelnoj grani.



Sl. 4 - BER performanse predloženog sistema ($M_t = 2, 3$ i 4) sa 64 podnosioca i sistema iz [2] sa 128 podnosilaca



Sl.5 - BER performanse predloženog sistema ($M_t = 4$) i sistema iz [2], u slučaju Doppler-ovog pomjeraja



Sl. 6 - BER performanse predloženog sistema ($M_t = 4$) sa 64 podnosioca i sistema iz [2] sa 128 podnosilaca, u slučaju Doppler-ovog pomjeraja.

4. ZAKLJUČAK

U ovom radu je predložena MISO šema za poboljšanje energetske efikasnosti OFDM-CDMA sistema sa pilot tonom i optimalnim TDC kombinovanjem. Za analizu BER performansi je razvijen originalni simulacioni model. Razmatrana su dva scenarija. U prvom je smatrano da je Doppler-ov pomjeraj zanemarljiv, dok se u drugom scenariju njegov uticaj na vjerovatnoću greške uzima u obzir. Pokazano je da se predloženom šemom ostvaruje značajno manji BER, što je očigledno i u slučaju primjene samo dvije predajne antene. Takođe, predloženi sistem ima veću spektralnu efikasnost, tj. i u slučaju kad se koristi dva puta manji broj podnosilaca, odnosno užu frekvencijski opseg predloženi model ostvaruje manji BER. U propagacionim uslovima sa izraženim frekvencijskim ofsetom BER se povećava značajno sporije u odnosu na konkurentski sistem. Na osnovu prikazanih rezultata očigledno je da se predloženi model može smatrati dobrim rešenjem za poboljšanje energetske efikasnosti OFDM-CDMA sistema sa pilot tonom jer se isti BER dobija uz značajno manji SNR. Pored veće pouzdanosti, energetske i spektralne efikasnosti, još jedna prednost predloženog sistema je i izbjegavanje primjene optimalnog TDC kombinovanja.

LITERATURA

- [1] T. Sao and F. Adachi, "Pilot-aided threshold detection combining for OFDM-CDMA downlink transmission in a frequency selective fading channel", IEICE Trans. Commun., vol. E85-B, No.12, pp. 2816-2826, Dec.2002.
- [2] Z. Veljovic, M. Pejanovic and I. Radusinovic, "Performance analysis of a new OFDM-CDMA system with pilot tone for multimedia communications", IEICE Trans. Commun., vol. E88-B, No.8, pp. 3480-3483, August.2005.
- [3] U. Urosevic, Z. Veljovic, E. Kocan: "BER Performance of MIMO OFDM-CDMA System in Ricean Fading Channel", EUROCON conference, Lisbon, Portugal, April 2011.
- [4] S. Hara, R. Prasad, Multicarrier Techniques for 4G Mobile Communications, Artech House, 2003.
- [5] H. Schulze, C. Lüders, Theory and Applications of OFDM and CDMA - Wideband Wireless Communications, Wiley, 2005.
- [6] H. Jafarkhani, Space-Time Coding, Theory and Practice, Cambridge University Press, 2005.
- [7] Kühn, Wireless Communications over MIMO Channels - Applications to CDMA and Multiple Antenna Systems, pp. 283-287, Wiley, 2006.

PIPELINE-OVANI SISTEM ZA ESTIMACIJU VISOKO NESTACIONARNIH FM SIGNALA PIPELINED SYSTEM FOR HIGHLY NONSTATIONARY FM SIGNALS ESTIMATION

Srdjan Jovanovski, *Faculty of Information Technology, Mediterranean University*
Veselin N. Ivanović, *Department of Electrical Engineering, University of Montenegro*

Sadržaj: Prikazan je dizajn signal adaptivne hardverske implementacije optimalnog vremenski-frekvencijskog (VF) filtra koji se bazira na rezultatima VF analize i estimaciji trenutne frekvencije dobijenoj VF analizom. Implementirana tehnika pipeline-ovanja omogućava preklapanje u bezuslovnim koracima filtriranja u susjednim VF trenucima i značajno poboljšava performanse. Postignuto je poboljšanje od jednog takta po VF tački što znači da poboljšanje po VF tački može iznositi i 50%. Dizajn je testiran na višekomponentnom signalu.

Abstract: Pipelined signal adaptive hardware implementation of an optimal time-frequency (TF) filter has been designed. It is based on the real-time results of TF analysis and on the TF analysis-based instantaneous frequency (IF) estimation. The implemented pipelining technique allows the filter to overlap in execution unconditional steps performing in neighboring TF instants and, therefore, to significantly enhance time performance. The improvement corresponding to the one clock cycle by a TF point is achieved, which means that the improvement by a TF point can reach even 50% in some TF points. The design is tested on multicomponent signals.

1. INTRODUCTION

Efficient processing of nonstationary signals requires time-varying approaches that can be defined by using common time-frequency distributions (TFDs). Classical TF filters, related to the Richaczek distribution, [1], short-time Fourier (STFT), [1]-[3], and Gabor transform, [4], as well as the Wigner distribution (WD), [5]-[8], exhibit serious drawbacks that limit their applicability. Their extended versions, suppress drawbacks of classical solutions, but they are numerically quite complex and require significant time, which makes them unsuitable for real-time analysis. Hardware implementations, when possible, can overcome these problems. Single-clock-cycle implementations of TF filters, [1], [2], [9], are quite complex and require repeating of basic calculation elements if they need to be used more than once. Their complexity strongly depends on the estimated signal duration, so they are capable of filtering of signals with the predefined duration only. By considering drawbacks of these solutions, here we develop a pipelined multiple-clock-cycle signal adaptive design of a WD-based optimal TF filter suitable for multi-component FM signals and the real-time implementation.

Optimal nonlinear TF filter can be defined by, [7], [8]:

$$(Hx)(n) = \sum_{k=-N/2+1}^{N/2} L_H(n,k) STFT_x(n,k) \quad (1)$$

where Weyl symbol, $L_H(n,k)$, represents filter's region of support (FRS), [1], [5]-[9], $STFT_x(n,k) = DFT_m[w(m)x(n+m)]$ is the STFT of the q -component noisy signal $x(n)$, $x(n) = \sum_{i=1, \dots, q} f_i(n) + \varepsilon(n)$, $DFT_m[\cdot]$ is the discrete FT in m , $w(m)$ is real-valued lag window, and N is the signal duration.

Following the procedure for the stationary Wiener filter design, [10], and considering the case of FM signals $f_i(n)$, $i=1, \dots, q$, highly concentrated in the TF plane around their instantaneous frequencies (IFs), and of the additive, widely spread white noise $\varepsilon(n)$, not correlated with the estimated FM signals, the FRS of the optimal TF filter corresponds to the combination of IFs of signals $f_i(n)$ [8], [11]. Then, the

optimal filtering of nonstationary FM signals can be reduced to the IF estimation in a noisy environment.

In TF analysis framework, the IF estimation is performed by determining frequency points k_i , $i=1, \dots, q$, where TFD of the noisy signal has local maximum, [8], [11]-[13],

$$IF_i(\vec{n}) = \arg[\max_{k \in Q_{k_i}} TFD_x(n, k)]. \quad (2)$$

Q_{k_i} is the basic frequency region in TF plane around $f_i(n)$, the IF of which is $IF_i(n)$. Among all quadratic TFDs, the WD produces the best IF estimation characteristics in the case of highly nonstationary monocomponent signals, [12], but also emphatic cross-terms in the multicomponent signals case. Cross-terms-free WD (CTFWD) retains the desired IF estimation characteristics of the WD in the monocomponent signals case, and, in the non-overlapping multi-component signals case, IF estimation characteristics of the CTFWD, obtained for each signal's component separately, remain the same as for the case of only that particular component exists, [13]. It is defined based on the same STFTs from (1), has already been implemented in real-time, [14], and, can be used as a base in an optimal nonstationary TF filter development, as performed in [11]. The design from [11] is additionally improved by the pipelining technique application.

2. PIPELINED IMPLEMENTATION

Complete pipelined hardware design of an optimal TF filter, principally following (1) and based on the CTFWD hardware design, [14], and on the real-time CTFWD-based IF estimation, [11], implemented through the sliding vector operation, is given in Fig.1. Modifying input STFT_IN data, the STFT-to-CTFWD gateway, [14], implements the CTFWD calculation, producing an improved TFD of the estimated noisy signal. By each STFT_Load/CTFWD_Store cycle, the calculated CTFWDs are stored into the ShMemBuff, used to move through the CTFWDs, to produce basic frequency

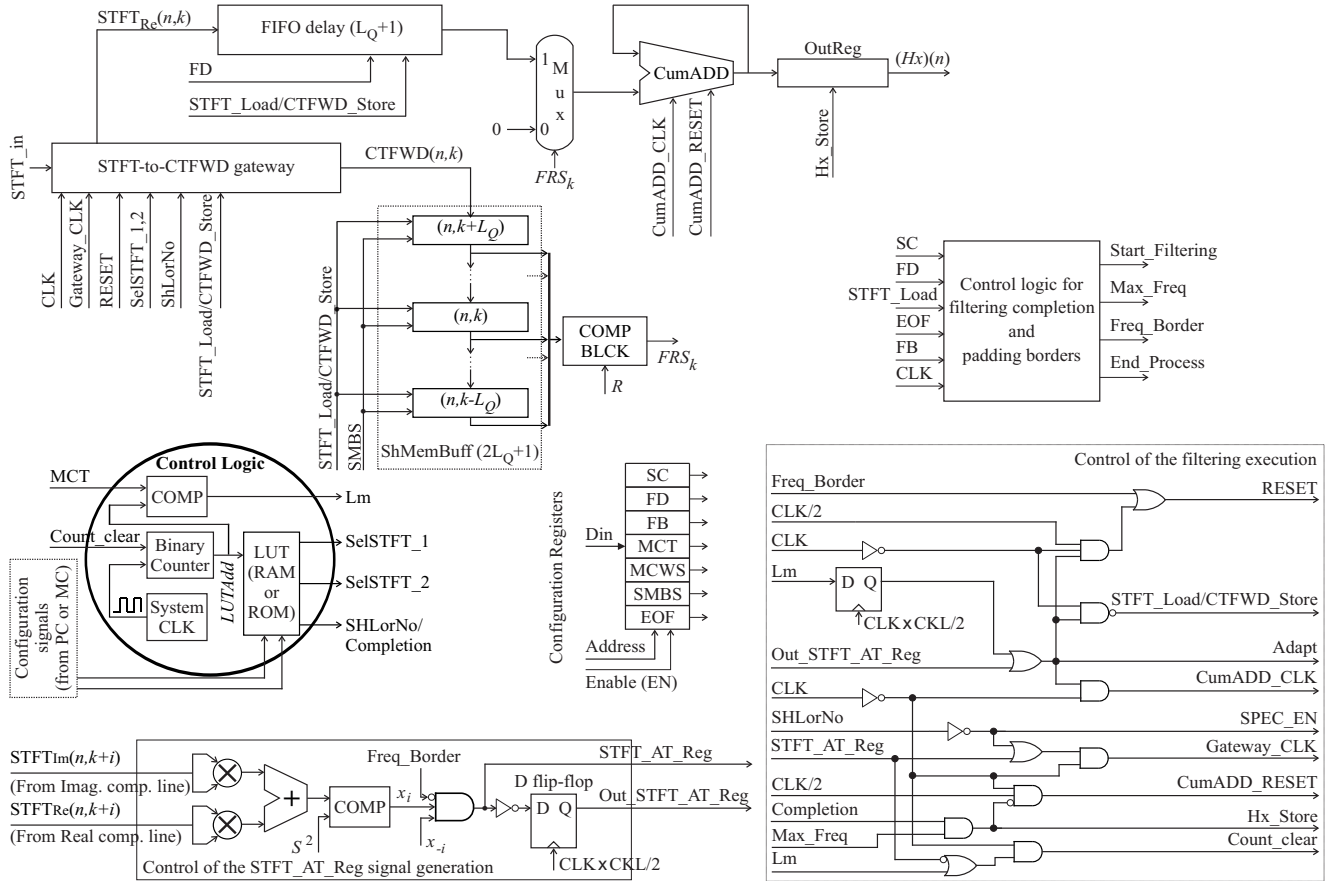


Fig. 1. Pipelined hardware implementation of the TF optimal filter based on the CTFWD-related IF estimation.

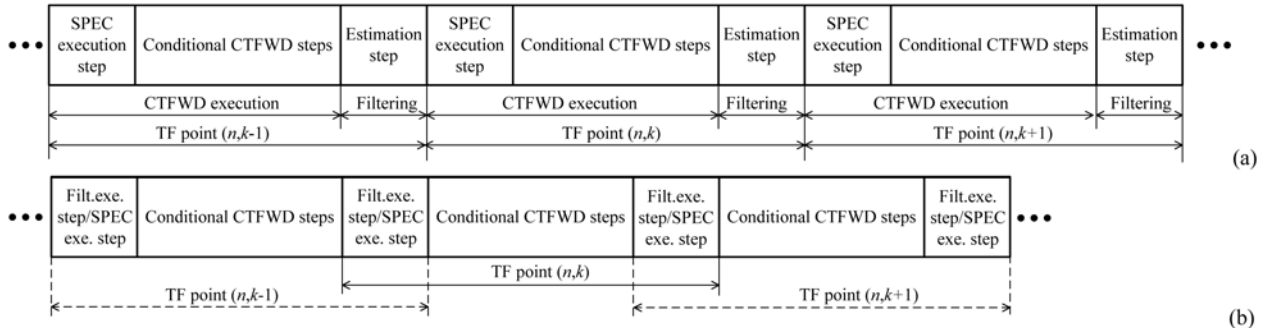


Fig. 2. Pipe stages in the TF filter implementation. (a) Non-pipelined implementation from [11], (b) Proposed pipelined implementation

region Q_k and to take a part in real-time IF estimation procedure implementation, [11]. The COMP BLCK tests conditions of the IF estimation procedure and, if they are satisfied, detects FRS in frequency point k , $k=-N/2+1, \dots, N/2$, for the observed n , determined by $FRS_k=1$. The FIFO delay is used to hold input STFT data and to provide output sample that corresponds in frequency to the ShMemBuff central element.

The design performs the estimation in $L(n,k)+2$ steps per frequency point, where each of these steps is executed in the corresponding CLK of the filtering execution. In the first $L(n,k)+1$ steps (0-th, 1-st, ..., $L(n,k)$ -th one), the CTFWD sample is calculated in the STFT-to-CTFWD gateway, [14]. The TF filter function is then implemented in the $(L(n,k)+1)$ -st-estimation-step, that is overlapped in execution by the 0-th step of the next frequency point $k+1$. Only 0-th-SPEC

execution-step and the estimation one are unconditional. They provide the SPEC-based IF estimation. Residual steps are conditional and depend on the estimated signal shape. They are used to improve the IF estimation quality up to CTFWD-based one and are taken only in TF points existing inside the STFT auto-terms' domains, determined by the signal adaptive period of the $STFT_AT_Reg$ signal. Through the participation in the $Gateway_CLK$ signal generation, zero value of the $STFT_AT_Reg$ signal disables the i -th term ($i=1, \dots, L(n,k)$) to enter the CTFWD calculation in i -th CLK and makes the estimation CLK from the next conditional ($(i+1)$ -st) one. In this way, the $STFT_AT_Reg$ signal allows the proposed design both to optimize the number of CLKs taken in different TF points within the execution and to produce the CTFWD-based IF estimation. It also controls the filtering completion in the observed frequency point.

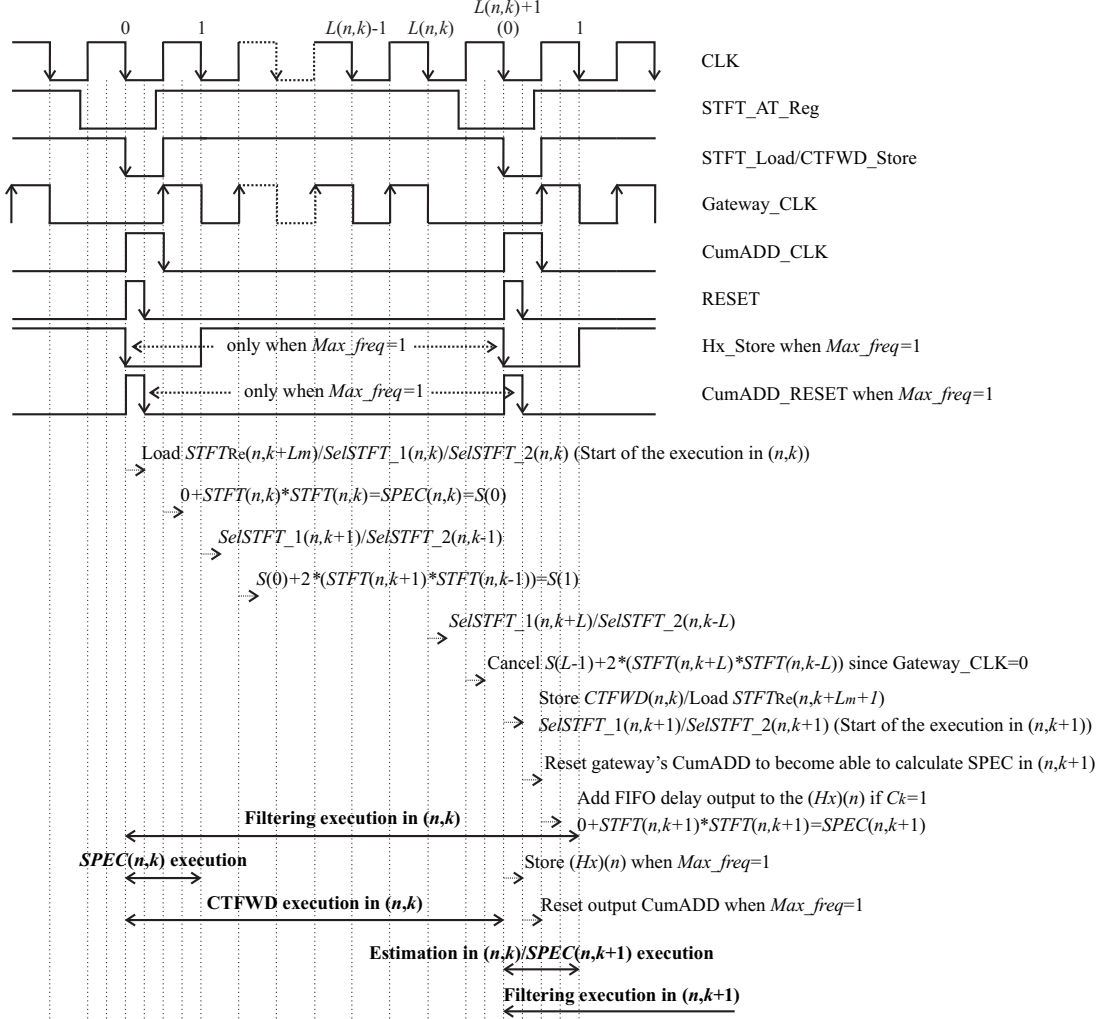


Fig.3. Timing diagram of signals that control execution in the proposed pipelined implementation.

Following the development from [11], in our design the final-completion-step of a time instant n is performed after the execution in each frequency point from the observed n and is overlapped in execution with the SPEC execution step from the next TF point $(n+1, -N/2+1)$, improving the execution time by only one CLK per a time instant n .

The proposed design allows overlapping in execution of the unconditional steps between the substantial frequency points $k, k+1, k=-N/2+1, \dots, N/2$, Fig.2, improving the execution time by one CLK, per a frequency point. This can be a significant development compared to the design from [11] because each time instant contains N frequency points. Residual steps cannot be included in pipelining, because they are conditional and do not have to exist. Signals *STFT_Load/CTFWD_Store*, *CumADD_CLK*, *Gateway_CLK*, *RESET*, and *CumADD_RESET* control the calculation in the STFT-to-CTFWD gateway, the summation in the CumADD in a frequency point $k, k=-N/2+1, \dots, N/2$, the filtering completion in an instant n , as well as the pipelining, as shown Fig.3. Generation of these signals, shown in Fig. 1, slightly increases hardware complexity, but also decreases capacity of the look-up-table memory required by the implementation.

By using pipelining technique, the design improves throughput of the implementation that corresponds to a CLK per a TF point. In comparison to the design from [11] and

depending on the sliding vector size L_Q and on the normalized signal rate, the improvement can reach values of about 15% (for $L_Q=7$) in frequency points existing around IFs, up to the 50% in frequency points existing outside STFT auto-terms. The design has been implemented in the Stratix II family EP1S10F780C5 device.

3. TESTING AND VERIFICATION

Verification of the proposed pipelined design has been performed through filtering of the 3-component test signal:

$$f(t) = e^{-45(t-2/25)^2} \cos(900(t+1.3)^2) + e^{-(t-2/5)^2} \times \cos(1200(t+0.3)^2) + e^{-45(t-2/3)^2} \cos(900(t-1/22)^2) \quad (3)$$

masked by high white noise such that $SNR_m = -0.34$ [dB] and considered within the time interval $[-0.15, 1]$. Parameters $T_w = 0.25$, $R = 0.05 \max_{n,k} \{CTFWD_x(n,k)\}$, $N = 256$, $L_Q = 5$, and reference level applied in the CTFWD calculation, [14]-[16], of $0.1 \max_{n,k} \{|STFT_x(n,k)|^2\}$ are used. Output SNR of 17.03 [dB] and the SNR improvement of 17.37 [dB] have been achieved. Knowing that (3) is highly nonstationary signal (normalized signal rates of its components are 0.828, 0.844, 0.828), improvement can be considered as very high. Besides,

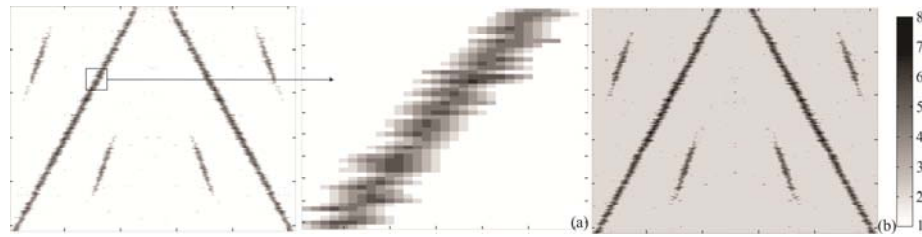


Fig.4. Distribution of CLKs performed per a TF point. (a) Pipelined implementation, (b) Non-pipelined implementation

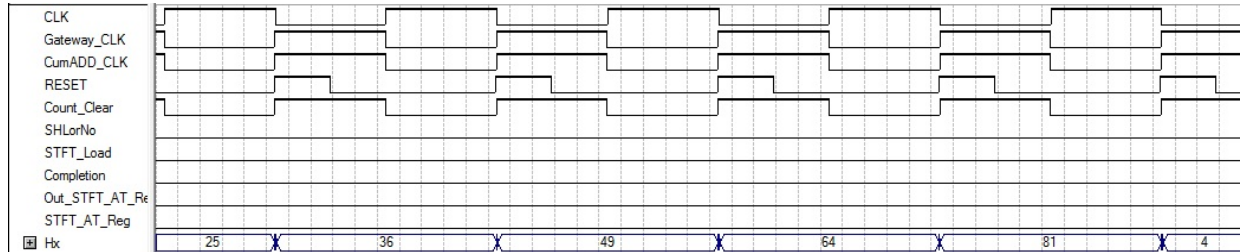


Fig.5. Verification of the pipelined implementation.

maximum improvement of up to approximately $(156/N) \times 10 \log(N/4) + (100/N) \times 10 \log(N/2) = 19.24$ [dB] is theoretically expected in a partly 2-component (in 100 instants) and a partly 4-component (in 156 instants) signal case.

To visually represent the achieved improvement, distribution of CLKs taken by the proposed design per a frequency point in the signal (3) case is shown in Fig.4, and is compared with the design from [11]. For the observed case, the improvement can easily be noted, computed, and numerically expressed by 45.358%.

In Fig. 5 are given simulation results that verify logic of pipelined implementation. As expected, the highest possible improvement in execution time of 50% in the TF points existing outside the STFT auto-terms domains can easily be observed. This observation is in full correspondence with the improvement of a CLK cycle per a TF point achieved by the proposed pipelined design. The observation can be noted by consideration the Hx signal in Fig. 5, whose duration corresponds to the duration of one CLK cycle.

4. CONCLUSIONS

The proposed design retains desirable characteristics of multiple-clock-cycle signal adaptive designs from [11] and [14], regarding calculation and implementation complexity, as well as the execution time. Besides, by applying the pipelining technique, the design additionally improves TF filtering execution time approximately up to the about 45%, depending on the estimated signal shape. In this way, it overcomes the corresponding TFD-based filters regarding almost all critical design performances. Moreover, it enables high quality real-time TF filtering, based on the highest quality signal adaptive CTFWD-related IF estimation. Non-adaptive designs and stationary filters, cannot produce so high quality results.

REFERENCES

[1] G. Matz, F. Hlawatsch, "Linear time-frequency filters: Online algorithms and applications", *App. in Time-Freq Sig. Proc.* (A. Papandreou-Suppappola, Ed.), CRC Press, pp.205-271, 2002.
 [2] F. Hlawatsch, G. Matz, H. Kirchauer, W. Kozek, "Time

frequency formulation, design and implementation of time-varying optimal filters for signal estimation", *IEEE Trans. Signal Processing*, vol. 48, no. 5, pp. 1417-1432, May 2000.

[3] G. Matz, F. Hlawatsch, "Linear time-frequency filters", in *Time-Frequency Sig. Anal. and Proc.* (B. Boashash, Ed.), Prentice Hall, 2002.
 [4] Y.Y. Zeevi, M. Zibulski, M. Porat, "Multi-window Gabor schemes in signal and image representations", in *Gabor Analysis and Algorithms: Theory and Applications* (H.G. Feichtinger, T. Strohmer, Eds.), Boston (MA): Birkhäuser, pp. 381-407, 1998.
 [5] W. Kozek, "Time-frequency signal processing based on the Wigner-Weyl framework", *Signal Processing*, vol. 29, no. 10, pp. 77-92, 1992.
 [6] R.G. Shenoy, T.W. Parks, "The Weyl correspondence and time-frequency analysis", *IEEE Trans. Signal Processing*, vol. 42, no. 2, pp. 318-331, Feb. 1994.
 [7] G.F. Boudreaux-Bartels, "Time-varying signal processing using Wigner distribution synthesis techniques", in *Wigner Distribution—Theory App. in Sig. Proc.* (W. Mecklenbrauker, F. Hlawatsch, Eds.), Elsevier, pp. 269-317, 1997.
 [8] LJ. Stanković, "On the time-frequency analysis based filtering", *Annals of Telecommunications*, vol. 55, no. 5/6, pp. 216-225, May/June 2000.
 [9] S. Stanković, LJ. Stanković, V.N. Ivanović, R. Stojanović, "An architecture for the VLSI design of systems for time-frequency analysis and time-varying filtering", *Annals of Telecomm.*, vol. 57, no. 9/10, pp. 974-995, Sept/Oct. 2002.
 [10] A. Papoulis, *Signal Analysis*, McGraw-Hill, New York, USA, 1997.
 [11] S. Jovanovski, V.N. Ivanović, "Signal adaptive pipelined hardware design of time-varying optimal filter for highly nonstationary FM signal estimation", *J. Signal Processing Systems*, vol. 62, no. 3, pp.287-300, 2011.
 [12] V.N. Ivanović, M. Daković, LJ. Stanković: "Performances of quadratic time-frequency distributions as instantaneous frequency estimators", *IEEE Trans. Signal Processing*, vol.51, no.1, pp.77-89, Jan.2003.
 [13] M. Daković, V.N. Ivanović, LJ. Stanković, "On the S-method based instantaneous frequency estimation", in *Proc. of the Int. Sym. on Sig. Proc. and its Apps.*, Paris, France, June 2003.
 [14] V.N. Ivanović, S. Jovanovski, "A signal adaptive system for time-frequency analysis", *Electronics Letters*, vol. 44, no. 21, pp. 1279-1280, Oct. 2008.
 [15] S. Stanković, LJ. Stanković, "An architecture for the realization of a system for time-frequency analysis", *IEEE Trans. Circuits & Sys. II, Exp. Briefs*, vol. 44, no. 7, pp. 600-604, July 1997.
 [16] V.N. Ivanović, R. Stojanović and LJ. Stanković, "Multiple clock cycle architecture for the VLSI design of a system for time-frequency analysis", *EURASIP J. Appl. Sig. Processing*, Spec. Issue Design Methods for DSP Sys., pp. 1-18, 2006.

INTEGRACIJA SaaS SERVISA U CLOUD TELEKOMA SRBIJA SaaS SERVICE INTEGRATION IN THE TELEKOM SRBIJA'S CLOUD

Borko Drašković, Telekom Srbija

Sadržaj: *Cloud computing predstavlja novi trend u informacionim tehnologijama koji savremena telekomunikaciona kompanija želi da implementira i ponudi krajnjim korisnicima. U ovom radu dat je pregled osnovnih Cloud servisa, prikaz Cloud platforme Telekoma Srbija i integracija SaaS servisa upotrebom odgovarajućeg API.*

Abstract: *Cloud computing represent new trend, which telecommunications companies wants to implement and offer to end users. This document presents basic Cloud services, overview of Telekom Srbija Cloud platform and integration of SaaS service using appropriate API.*

1. UVOD

Ideja o računarstvu u oblaku (*Cloud computing*) se temelji na ideji plaćanja po potrebi. Napretkom računarske tehnike, razvojem Interneta i povećanjem brzina pristupa omogućeno je da se određeni zadaci obavljaju udaljeno.

Cloud computing predstavlja novi trend u informacionim tehnologijama koji savremena telekomunikaciona kompanija želi da implementira i ponudi krajnjim korisnicima. *Cloud* možemo posmatrati kao skup hardvera, mreža, skladišta podataka, servisa i interfejsa koji zajedno čine uslugu. U osnovi to je iznajmljivanje IT servisa i usluga u skladu sa potrebama i zahtevima korisnika.

2. VRSTE I OSOBINE CLOUD

Postoje četiri vrste *Cloud*:

- Javni *Cloud*,
- Privatni *Cloud*,
- Zajednički *Cloud*,
- Hibridni *Cloud*.

Javni *Cloud* je onaj u kome pružalac usluge omogućava javni pristup resursima kao što su aplikacije, skladišta za podatke i drugi resursi, nezavisno od toga da li se radi o pojedincima ili organizacijama, putem Interneta. Usluge mogu biti besplatne ili se koristi model plaćanja po korišćenju (*pay-per-usage*).

Privatni *Cloud* je napravljen isključivo za upotrebu jednog klijenta, koji može biti unutar organizacije ili hostovan od strane pružaoca *Cloud* usluge. Organizacije koje poseduju privatni *Cloud* imaju potpunu kontrolu nad njegovom strukturom.

Zajednički *Cloud* je *Cloud* koji deli nekoliko organizacija. Infrastruktura podržava posebne zajednice koje imaju zajedničke potrebe, misije, zahteve sigurnosti i slično.

Strukturu hibridnog *Cloud* čine dva ili više različitih tipova *Cloud* (privatni, zajednički ili javni), koji mogu biti hostovani kod različitih pružaoca usluga. Svaki od ovih *Cloud* ostaje jedinstveni entiteti, a u isto vreme su međusobno povezani tako da je omogućen efikasan prenos podataka ili upotreba aplikacija između njih. Jedan od primera ovakvog *Cloud* je slučaj kada organizacija koja ima privatni *Cloud* koristi javni *Cloud* u slučaju kada je potrebno kratkotrajno i značajno povećanje kapaciteta (*Cloud*

bursting). Glavna prednost *Cloud bursting* i hibridnog *Cloud* modela je da kompanija plaća proširanje resursa samo u vremenu kada im je to potrebno.

Jena od karakteristika *Cloud* je pružanje mogućnosti korisnicima da po potrebi, najčešće putem *web* pretraživača pristupaju i upravljaju računarskim resursima. Ovaj proces je potpuno automatizovan. *Cloud* resursi postaju dostupni preko mreže korišćenjem standardnih protokola, HTTP, HTML, XML, Java, SOAP ili drugih. Ovo omogućuje široku upotrebu heterogenih platformi.

Udruživanjem računarskih resursa, bez obzira gde se nalazili, omogućeno je opsluživanje većeg broja korisnika. *Cloud* korisnici mogu pristupiti skupu servera koji se percipiraju kao jedinstvena celina, i što je još važnije, korisnici nisu svesni konfiguracije fizičkih servera koji omogućuju uslugu. Elastičnost *Cloud* omogućava dinamičku promenu ili preraspodelu računarskih resursa zavisno od korisničkih zahteva.

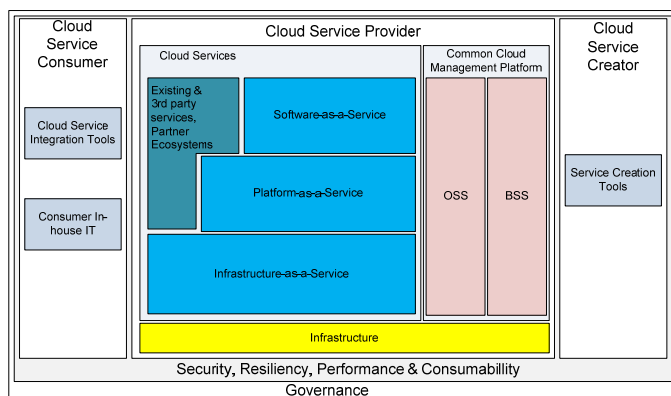
Praćenjem usluge, *Cloud* resursi mogu biti nadgledani, kontrolisani i dati na uvid i korisniku i administratoru. *Cloud* korisnici prate svoje pojedinačne usluge dok administratori nadziru ukupne performanse. Na ovaj način mogu se obračunavati i troškovi po korisniku ili po odeljenju/organizaciji.

Ceo sistem mora da bude tako dizajniran da korisniku omogući pristup računarskim resursima bez obzira na moguće hardverske otkaze i otkaze komunikacione opreme.

Smanjenje troškova je prva dobra osobina *Cloud*. Druga važna osobina *Cloud* je nezavisnost uređaja i lokacije. Odnosno, mogućnost korisnika da pristupi sistemu i podacima sa bilo kog uređaja (telefon, prenosni računar, radna stanica) i bilo kog mesta. Održavanje *Cloud* aplikacija i opreme je jednostavnije jer ne zahteva instalaciju na svakom korisnikovom uređaju. Korišćenjem API (*Application Programming Interface*) omogućena je olakšana interakcija između softvera na različitim platformama sa softverom na *Cloud* platformi na isti način kao što korisnički interfejs olakšava interakciju između čoveka i računara.

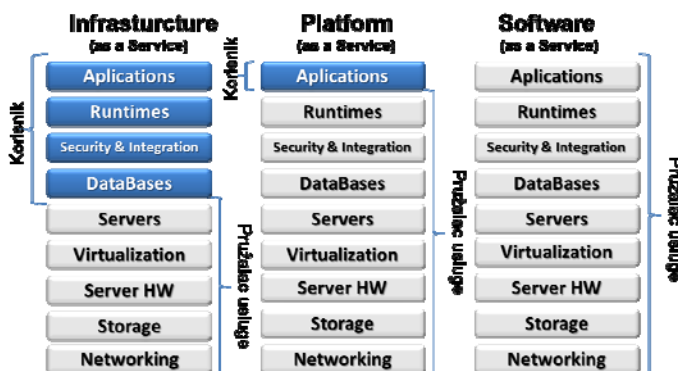
3. CLOUD COMPUTING SERVISI

Za realizaciju *Cloud computing* rešenja potrebno je obezbediti efikasnu, visoko skalabilnu i fleksibilnu arhitekturu (Slika 1.).



Slika 1. Referentna arhitektura

Pružaoi usluga, *Cloud computing* nude u skladu sa nekoliko osnovnih modela: *Infrastructure-as-a-Service* (IaaS), *Platform-as-a-Service* (PaaS) i *Software-as-a-Service* (SaaS). Razlike između ova tri servisna modela su prikazane na Slici 2.



Slika 2. Cloud servisni modeli

IaaS, PaaS i SaaS se implementiraju na elastičnoj, skalabilnoj i visoko performantnoj arhitekturi zasnovanoj na virtualnoj serverskoj infrastrukturi i redundantnom mrežnom okruženju pružaoca usluge.

Infrastructure-as-a-Service treba da omogućiti iznajmljivanje infrastrukturnih resursa, kao što su računarska mreža, storage i virtualna serverska infrastruktura. IaaS kao i svi ostali servisi treba da bude bazirana na virtualnoj infrastrukturi. Izborom predefinisanih paketa korisnik će biti u prilici da kroz automatski *provisioning* locira određene hardverske resurse, na kojima kasnije može podići svoj operativni sistem, bazu, aplikaciju. Korisnik u svakom trenutku mora imati uvid u svoju potrošnju, odnosno iskorišćenost resursa (CPU, memorija,...) kao i materijalne troškove korišćenja servisa.

Platform-as-a-Service (PaaS) je skalabilan, elastičan i visoko dostupan servis na *Cloud computing* infrastrukturi. PaaS je samo jedan od servisa koji će biti ponuđeni u *Cloud* okruženju kao samostalan servis. Korisniku treba omogućiti da pored infrastrukturnih servisa, koje nudi pružalac usluge, odabere odgovarajući operativni sistem (Windows, Linux,...), aplikativni server, bazu podataka (Microsoft's SQL, MySQL, Oracle...), *Database management systems* (DBMSs), *messaging* i mnoge druge oblike aplikativne infrastrukture. PaaS može prvenstveno biti namenjen za

razvoj i testiranje aplikacija, gde su baze uglavnom male, a bezbednosni zahtevi i deljeni resursi od strane velikog broja korisnika nisu od posebnog značaja. Posebnu pažnju treba posvetiti operativnom sistemu koji su izuzetno interesantni na klijentskom tržištu, posebno u kombinaciji sa novim klasama uređaja kao što su *web* knjige i *media* tableti.

Software-as-a-Service (SaaS) je tip servisa gde pružalac usluge samostalno ili u saradnji sa partnerom obezbeđuje aplikacije za krajnje korisnike, kojima korisnici prilaze preko Interneta. SaaS podrazumeva razvoj i hostovanje softvera namenjenog krajnjim korisnicima.

SaaS rešenja moraju biti zasnovana na multi-korisničkom okruženju. Ista verzija aplikacija, na istoj konfiguraciji (hardver, mreža, operativni sistem) se koristi za sve registrovane korisnike. Da bi se obezbedila skalabilnost, aplikacija se instaliraju na više mašina (*horizontal scaling*). U nekim slučajevima, druga verzija aplikacije može biti postavljena i ponuđena odabranoj grupi korisnika koji imaju prava pristupa beta verzijama aplikacija. Dakle, SaaS treba da bude koncipiran za centralizovan pristup, za razliku od tradicionalnih rešenja, gde se više kopija softvera, potencijalno različitih verzija, sa različitim konfiguracijama prilagođavaju i instaliraju na lokacijama klijenata. S obzirom da će se SaaS unapređivati mnogo češće nego klasični softveri, nedeljno ili mesečno potrebno je ispoštovati osnovne postulate servisne arhitekture na kojima će se bazirati SaaS ponuda:

- Hostovanje aplikacija je centralizovano, svaki novi *release* se instalira bez potrebe za bilo kakvom aktivnošću (denstalacija postojećeg/instalacija novog software) na strani korisnika.
- Jedno aplikativno rešenje zasniva se na jednoj verziji softvera, što omogućava jednostavniji razvoj i testiranje.
- Proizvođači aplikacija imaju jednostavniji pristup korisničkim resursima, što ubrzava i olakšava testiranje i promene u dizajnu rešenja.
- Pružalac usluge (aplikacije) mora imati uvid u aktivnosti korisnika na samoj aplikaciji, što olakšava lociranje problema i omogućava dodatna unapređenja na aplikaciji.

Pružanje IaaS/PaaS/SaaS je prilično kompleksno i zahteva dizajniranje, implementaciju i integraciju *Cloud computing* rešenja u postojeće okruženje pružaoca servisa.

4. CLOUD PLATFORMA TELEKOMA SRBIJA

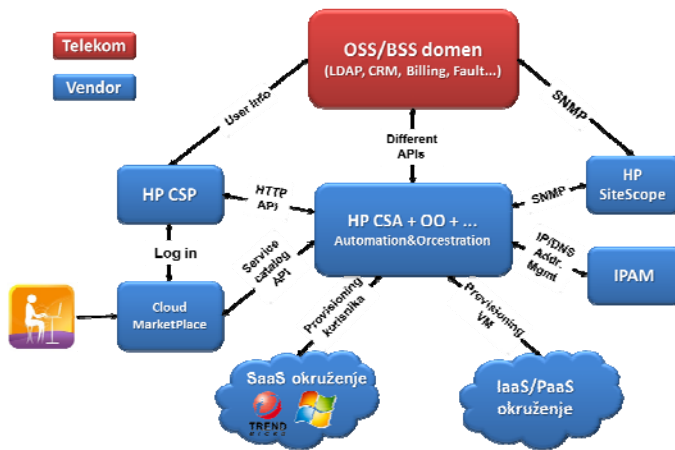
U okviru Telekoma Srbije, *Cloud* usluge postoje od 2007. godine (*web* hosting i hosting virtualnih servera). Najveći nedostatak u pružanju navedenih usluga je odsustvo automatizacije u procesu prodaje, orkestracije i integracije sa postojećim IT platformama. Telekom Srbija je krajem 2012. godine raspisao tender za unapređenje postojećih *Cloud* usluga. Glavni motiv nabavke je unapređenje procesa prodaje, automatizacija i integracija SaaS servisa kroz automatizaciju procesa. Pobjednik tendera je konzorcijum koji čine kompanije Teri Inženjering i HP Solutions.

Telekom Srbija se opredelio za *multi-tenancy* strategiju koja omogućava krajnjim korisnicima rad u izolovanom i nezavisnom okruženju baziranom na najnovijim

tehnologijama. S druge strane Telekom Srbija je prepoznao potrebe tržišta za sve većim brojem aplikativnih rešenja koja bi mogla biti interesantna malim i srednjim preduzeća kao krajnjim korisnicima.

Takođe, sve veći broj nezavisnih softverskih kuća prilagođava svoje postojeće aplikacije *multi-tenancy* okruženju. Na ovaj način se između ostalog postiže kvalitetno pružanje IT servisa uz sniženje cena i opstanak na IT tržištu. Proizvođači softvera traže pouzdanog partnera sa distribuiranom infrastrukturom, što Telekomu Srbija pruža ogromne mogućnosti na IT tržištu.

Uprošćeni prikaz *Cloud* platforme i njene integracije sa postojećim sistemima Telekoma Srbija prikazan je na Slici 3.



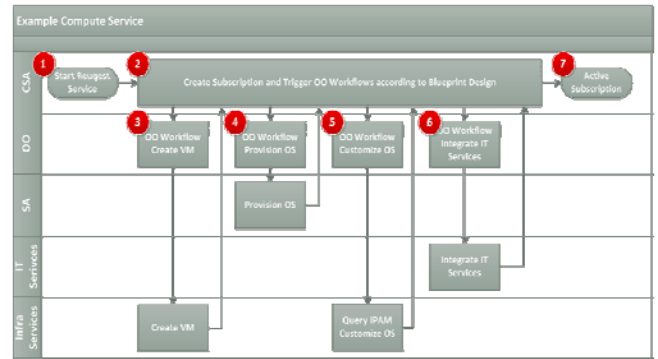
Slika 3. *Cloud* platforma Telekoma Srbija

U okviru platforme postoje dva dela *Cloud* portala: *Cloud Marketplace portal* kompanije Teri (*FrontEnd portal*) i *Cloud SelfCare portal* kompanije HP (*BackEnd portal*). Oba portala preko HTTP API imaju vezu sa katalogom servisa koji se nalazi u okviru HP CSA platforme. *BackEnd* portal pruža korisniku mogućnost *on line* upravljanja svojom infrastrukturom i kupovinu dodatnih elemenata sistema, po potrebi.

Komponenta koja omogućava sinhronizaciju i automatizaciju svih aktivnosti tokom aktivacije servisa je *HP Orcestrator*. Telekom Srbija koristi HP CSA + OO rešenje kao glavni deo automatizacije procesa orkestracije i kao tačku integracije sa ostalim OSS/BSS sistemima.

Orkestrator poseduje komponentu dizajnera servisa koji u grafičkom alatu omogućava kreiranje algoritma aktivacije. Koristi CMDB bazu u kojoj se nalaze informacije o svim raspoloživim resursima koji se mogu koristiti za aktivaciju servisa. Orkestrator ima adaptore za automatsku konfiguraciju sledećih elementarnih komponenti: VMware, Hyper-V, Linux KVM, Linux XEN, Cisco switches, Cisco Routers, Cisco firewall, EMC Clariion storage, HP Proliant server, IBM x server, MS AD, MS Exchange, (MS Lync), LDAP,...

Na Slici 4 je prikazan primer automatskog konfigurisanja jedne virtuelne mašine.

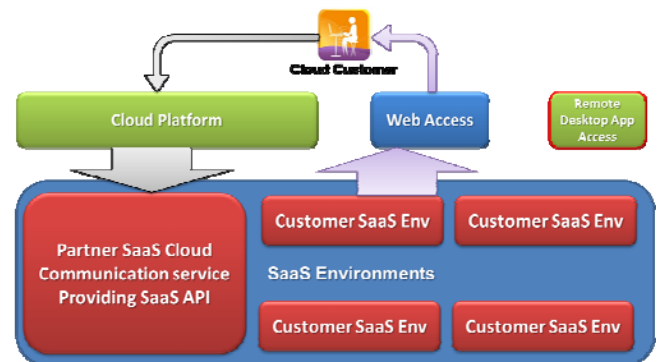


Slika 4. Primer automatskog konfigurisanja virtuelne mašine

5. INTEGRACIJA SaaS SERVISIA

SaaS treba da bude koncipiran kao distribuirani softverski model u kom se aplikacije hostuju na strani pružaoca usluge, administriraju od strane partnera, a dostupnost obezbeđuje preko Internet-a. *Software-as-a-Service* će u prvo vreme biti ponuđen u saradnji sa partnerima.

API o kojem će ovde biti reč je namenjen za korišćenje od strane partnera Telekoma Srbija koji imaju nameru da svoj softver prodaju koristeći *Cloud* infrastrukturu. Pošto različiti softverski servisi funkcionišu na različite načine, Telekom Srbija je pokušao da standardizuje SaaS API koji se koristi u *Cloud* kako bi novi servisi mogli da budu implementirani jednostavno i efikasno. Na slici 5. je prikazana arhitektura i pregled procesa upotrebe SaaS servisa.



Slika 5. Arhitektura i upotreba SaaS servisa

Kada korisnik pride *Cloud Marketplace* portalu u okviru *Cloud* platforme i želi da koristi SaaS, *Cloud* platforma koristi standardni SaaS API za upravljanje SaaS servisom. Partner treba da implementira SaaS *Cloud* komunikacioni servis (*Communication Service*) u okviru ovog API. Partnerova SaaS platforma, na osnovu korišćenih API metoda, može da vrši *provisioning* novog korisničkog okruženja. SaaS platforma treba korisniku da obezbedi *web* pristup samoj aplikaciji, odnosno aplikativnom okruženju. U slučaju da postoji dovoljan broj zahteva za korišćenjem servisa, pristup se može realizovati upotrebom *Remote Desktop* pristupa. Ova vrsta pristupa se još koristi kod *Windows* baziranih aplikacija koje nemaju *web* pristup.

Svaki SaaS partner treba da definiše:

1. SaaS komunikacioni servisi
2. SaaS okruženje

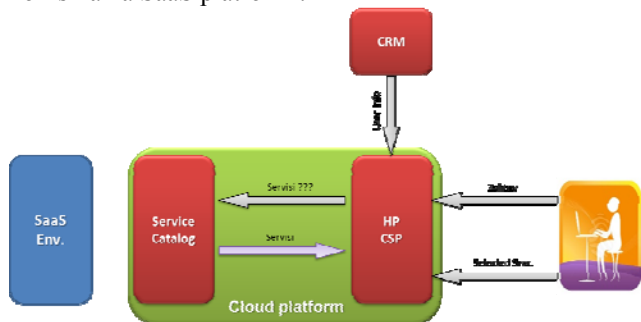
SaaS komunikacioni servis može da bude servis koji obezbeđuje REST API interfejs koji definiše sve potrebne metode definisane SaaS API. Kao zamena REST API može da se koristi CLI. Svaki partner mora da obezbedi sledeće informacije kako bi komunikacija između *Cloud* platforme i SaaS platforme partnera mogla da bude uspostavljena:

1. Pristupna tačka
 - a. REST API, definiše server sa kojim je potrebno ostvariti komunikaciju (primer: `https://SaaSAPIAccessPoint:443/soap/api/v1`)
 - b. CLI API, ime servera koji ima dostupne CLI komande za komunikaciju upotrebom SSH ili RDP (primer: `ssh://CLIAPIAccessPoint:22/Cloud_Command`)
2. Korisničko ime i lozinku

SaaS okruženje (*SaaS Environment*) predstavlja samu aplikaciju koja ima posebno okruženje za svaku kompaniju. SaaS okruženje treba da obezbedi *web* pristup gde se korisnici mogu logovati i koristiti aplikaciju. Preporuka je da svaka kompanija ima drugačiju pristupnu tačku, a da kao korisničko ime budu korišćenje email adrese korisnika pošto su email adrese jedinstvene. U pogledu sigurnosti koja će biti primenjena u partnerskoj aplikaciji, separacija okruženja je moguća na nekoliko nivoa. Na nivou baze podataka, servera, storage-a.

Sve komande u okviru SaaS API-ja možemo podeliti u 5 kategorija. To su: Utility API, Service API, Company API, Licence API, User API. Sve funkcije SaaS API mogu se svrstati u 2 scenarija. Prvi je *provisioning* krajnjeg korisnika koji vodi računa o njegovom menadžmentu i menadžmentu licenci. Drugi je prijava korisnika kod koga vodimo računa o njegovoj autentifikaciji i povratnoj informaciji koja ide od sistema ka korisniku. Povratna informacija ne mora biti samo URL za pristup aplikaciji već može biti i neko korisničko uputstvo, link za dodatni download i slično.

Na Slici 6. i Slici 7. su prikazani koraci prilikom kreiranja korisnika na SaaS platformi.



Slika 7. Primer kreiranja korisnika – deploying



Slika 6. Primer kreiranja korisnika – portal info

Preduslov ovom scenariju je da je korisnik poznat, odnosno da CRM ima podatke o korisniku. U prvom koraku korisnik šalje zahtev za nekim SaaS servisom. *Cloud* platforma komunicira sa CRM i dobija informacije o korisniku. Nakon toga *Cloud* platforma šalje upit ka katalogu servisa i traži servise koje dati korisnik može da kupi. Servisni katalog šalje tražene informacije ka *Cloud* platformi koja to prezentuje korisniku. Pošto se korisnik odluči za kupovinu nekog od mogućih SaaS servisa, *Cloud* platforma šalje ka SaaS platformi komande za kreiranje kompanije, usera i informaciju o tome koji servis je korisnik tražio. SaaS platforma vraća URL koji može biti link za download software-a, link do informacije o licenci ili link do web pristupa aplikaciji. U opštem slučaju to može biti link do bilo kog sadržaja koji je relevantan za konzumaciju servisa. Korisnik dobija imejlom link koji je *Cloud* platforma dobila od SaaS platforme. *Cloud* platforma u taj imejl može da ubaci i adekvatno uputstvo za korišćenje aplikacije ili bilo koji drugi dokument.

6. ZAKLJUČAK

SaaS koncipiran kao distribuirani softverski model u kom se aplikacije hostuju na strani pružaoca usluge, administriraju od strane partnera, a dostupnost obezbeđuje preko Internet-a, omogućuje telekom operaterima izuzetno kratak put od ideje do puštanja u produkciju različitih tipova softvera. Upotrebom ovog modela svi učesnici na telekomunikacionom tržištu mogu da imaju koristi. Telekomunikacione kompanije su u mogućnosti da prošire ponudu svojih ICT servisa, bez značajnog ulaganja u lansiranje svakog od novih servisa. Nezavisni proizvođači softvera, u SaaS, vide veoma efikasan kanal za distribuciju svojih proizvoda. Prodajući svoje proizvode preko *Cloud* rešenja telekom operatera, nezavisni proizvođači imaju daleko širi dijapazon potencijalnih korisnika nego što bi to mogli da imaju u slučaju direktne prodaje. Na kraju, korisnici imaju mogućnost korišćenja IT rešenja bez bilo kakvog početnog ulaganja, i kasnijeg održavanja, IT infrastrukture. Ukoliko ponudjena rešenja ne zadovoljavaju kriterijume korisnika, korisnik će jednostavno otkazati uslugu i pronaći alternativnog pružaoca usluge. SaaS API predstavlja ključni element za jednostavnu i efikasnu primenu SaaS modela.

LITERATURA

- [1] MI SANU, *Dokumentacija za projekat unaprjeđenja IT bezbednosti i zaštite ICT servisa*, Beograd, 2013.
- [2] Telekom Srbija, HP, Teri Engineering, *Dokumentacija za Cloud projekat Telekoma Srbija*, Beograd, 2013.
- [3] Samčović A., *Cloud computing okruženje u mrežama sledeće generacije*, PosTel 2012, Beograd, 2012.
- [4] Weiss A., *Computing in the Clouds*, netWorker, Vol.11, No.4, pp. 16-25, 2007

MONITORING I ANALIZA VITALNIH FIZIOLOŠKIH PARAMETARA PRIMJENOM PDA UREĐAJA

MONITORING AND ANALYSIS OF VITAL PHYSIOLOGICAL PARAMETERS USING PDA DEVICES

Nemanja Filipović, Radovan Stojanović, Elektrotehnički fakultet Podgorica

Sadržaj: U ovom radu predstavljena je upotreba pametnih telefona u procesu obrade i analize EKG, PPG i temperaturnih signala. Ovi signali se, nakon prikupljanja sa odgovarajućih senzora postavljenih na pacijentu i dodatne analogne obrade, snimaju preko standardnog mikrofonskog porta uređaja. Snimljeni signali se potom obrađuju posebno razvijenom Android aplikacijom, koja pored prikupljanja i vizuelizacije pomenutih signala, obavlja detekcije QRS kompleksa i maksimuma PPG signala, računanje pulsa i PTT intervala, kao i „on-demand“ mjerenje temperature. Aplikacija radi u realnom vremenu, uz malo zauzimanje sistemskih resursa.

Abstract: This paper describes the use of smart phones in the processing and analysis of ECG, PPG and temperature signals. These signals, after collecting from the appropriate sensors placed on the patient and additional analog processing, are recorded via a standard microphone port of the device. The recorded signals are then processed by a specially developed Android applications, which in addition to the collection and visualization of the mentioned signals, performs detections of the QRS complexes and maximums of the PPG signal, calculating the pulse and PTT interval, as well as on-demand temperature measurement. The application runs in real time, with a small occupation of the system resources.

1. UVOD

Mobilni uređaji poput pametnih telefona i tablet računara konstantno napreduju, kako u pogledu procesorske snage, tako i u pogledu memorijskih kapaciteta, postajući svakim danom sve više sastavni dio modernog života. U skorije vrijeme ovakvi mobilni uređaji nalaze svoju primjenu i u obradi biomedicinskih signala. Fiziološke karakteristike poput broja otkucaja srca, ritama disanja, zasićenosti krvi kiseonikom i druge, mogu biti detektovane [1]. Čak i složeni algoritmi za obradu biomedicinskih slika mogu se implementirati na ovim uređajima zahvaljujući sve bržim procesorima i adekvatnoj softverskoj podršci.

Biomedicinski signali na kojima će biti fokusirano istraživanje u okviru ovog rada su elektrokardiogramski (EKG), fotopletizmografski (PPG) i signali tjelesne temperature. Njihova upotreba je široko rasprostranjena : EKG se već dugo koristi kao osnovni signal za detekciju i dijagnostikovanje različitih srčanih oboljenja [2], na osnovu PPG signala mogu se detektovati srčani ritam, zasićenost krvi kiseonikom, krvni pritisak i drugo [3], dok je tjelesna temperatura važan indikator stanja organizma.

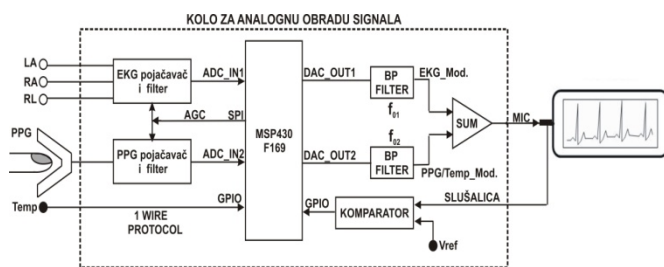
Često se za povezivanje pametnih telefona sa senzorskim uređajima koriste bežični vidovi komunikacije, poput „bluetooth“-a. Međutim, iako ovaj pristup ima brojne prednosti, postoje i mane poput povećane potrošnje energije, uticaj šuma, nemogućnost „plug-and-play“ načina korišćenja, relativno velika cijena komunikacionih modula. Zbog toga će se u ovom radu koristiti drugačiji pristup.

Naime, jedini komunikacioni interfejs koji je uvijek dostupan na svim pametnim telefonima je mikrofonski port. Često se zanemaruje da ovi uređaji mogu prikupljati analogne signale sa pomenutog porta uz brojne prednosti : stabilnost i jednostavnost komunikacije, niska potrošnja, visoka rezolucija (do 24 bita po odbirku), podesiva frekvencija odabiranja (do 44 KHz), ugrađeni pojačavači, jednostavna softverska kontrola. Međutim, treba imati na umu da „headset“ port ima ugrađene filtre propusnike opsega učestanosti, kako na ulaznoj tako i na izlaznoj strani. Donja granična frekvencija kod oba filtra varira od 50 Hz do 300 Hz zavisno od uređaja, dok je gornja granica kod ulaznog filtra obično 3.4 KHz. Kako se kod EKG signala može smatrati da se većina spektralnih komponenti nalazi ispod 50 Hz, kod PPG signala ispod 10 Hz, dok je temperaturni signal gotovo jednosmjerni (DC), jasno je da se ovi signali ne mogu vjerno reprodukovati direktnim snimanjem sa mikrofonskog porta.

Način da se premoste ova ograničenja i omogući da pametni telefoni i tablet računari pod Android operativnim sistemom preko mikrofonskog porta prikupljaju sporo promjenjive signale poput fizioloških, predstavljen je u ovom radu. Ovi signali se prvo modulišu i podižu na viši frekventni opseg, a zatim prikupljaju i obrađuju u realnom vremenu pomoću odgovarajuće Android aplikacije.

2. ARHITEKTURA SISTEMA

Osnovna arhitektura sistema sastoji se od EKG, PPG i temperaturnog senzora, kola za analognu obradu signala i pametnog telefona sa aplikacijom, Slika 1.



Slika 1. Arhitektura sistema

A. Kolo za analognu obradu signala

Signali se sa odgovarajućih senzora uvode u kolo za analognu obradu signala. Tu se vrši filtriranje i pojačanje EKG i PPG signala. Implementirana je automatska kontrola pojačanja (AGC), pri čemu stepen pojačanja računa i zadaje mikrokontroler komunicirajući sa odgovarajućim digitalnim potencimetrima u pojačavačima putem SPI protokola. Temperaturni senzor je digitalni i povezan je na standardni ulazno-izlazni pin mikrokontrolera (GPIO), a komunikacija između njih je „1 WIRE“.

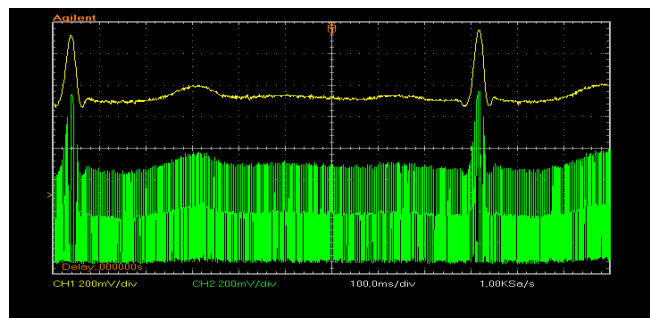
Pojačani EKG i PPG signali se potom šalju na ulaz analognog-digitalnog konvertora (ADC) niskopotrošačkog mikrokontrolera MSP430F169 [4], proizvođača „Texas Instruments“. Mikrokontroler, osim određivanja stepena pojačanja za AGC, vrši i impulsno-amplitudsku modulaciju EKG, PPG i temperaturnog signala. Na taj način ovi signali se transliraju na viši frekventni opseg, te se bez slabljenja mogu snimati preko mikrofonskog porta Android uređaja.

Modulacija se obavlja pozivanjem tajmerskih interaptata odgovarajućom frekvencijom. Za modulaciju EKG signala inerapt Tajmera A mikrokontrolera se poziva sa frekvencijom 2.040 KHz, pri čemu se naizmjenično obavljaju jedna od sljedeće dvije funkcije. Prvo se vrši AD konverzija i vrijednost digitalno-analognog konvertora (DAC) se postavlja na vrijednost dobijenu AD konverziom i zadržava do narednog poziva interaptata. Pri sljedećem pozivu rutine, vrijednost DAC-a se postavlja na 0 i zadržava do sljedećeg poziva, i tako u petlju. Na taj način se obavlja impulsno amplitudska modulacija, sa frekvencijom nosioca jednako polovini frekvencije pozivanja interaptata, dakle 1.020 KHz. Na isti način se obavlja i modulacija PPG signala, pri čemu se koristi drugi par AD/DA konvertora, a frekvencija pozivanja interaptata Tajmera B mikrokontrolera je 3.2 KHz, odnosno frekvencija nosioca je 1.6 KHz.

Modulacija temperaturnog signala je unekoliko specifična, jer na korišćenom mikrokontroleru postoje samo dva para AD/DA konvertora, a mjerenje se obavlja „on demand“, tj. samo kada se pritisne odgovarajuće dugme na korisničkoj aplikaciji. Zbog toga nije potrebno neprekidno slati informacije o temperaturi, a kako je EKG signalu dat veći prioritet, temperaturni i PPG signali dijele isti modulacioni kanal. To znači da se tim kanalom moduliše i šalje PPG signal sve dok se ne pritisne dugme za mjerenje temperature. Tada se generiše komandni signal koji se putem slušalica šalje do mikrokontrolera kako bi ga obavijestio da treba da prikupi podatke sa temperaturnog senzora i umjesto PPG signala pošalje informacije o temperaturi pacijenta.

Senzor šalje jedanaestobitnu vrijednost temperature mikrokontroleru, koji potom generiše tri pravougaona impulsa čije je trajanje srazmjerno izmjerenoj temperaturi. Ti impulsi se zatim modulišu i prosljeđuju na izlaz DAC-a. Nakon toga se opet prelazi na slanje PPG signala.

Nakon modulacije, signali se propuštaju kroz filtre propusnike opsega učestanosti ($f_{01}=1.020$ KHz, $f_{02}=1.6$ KHz) U sumatoru se sabiraju modulirani EKG i PPG/temperaturni signali i kao jedinstven signal šalju do mikrofonskog porta pametnog telefona. Na Slici 2 prikazan je modulišući (žuto) i modulirani (zeleno) EKG signal snimljen na osciloskopu.



Slika 2. EKG modulišući i modulirani signal

B. Android „multi-thread“ aplikacija

Android aplikacija, razvijena u programskom jeziku JAVA, zadužena je za prikupljanje, obradu i vizuelizaciju signala. Podijeljena je u tri paralelne grane. Glavna grana zadužena je za kontrole na korisničkim interfejsu. Druga grana vrši snimanje signala sa mikrofonskog porta, dok treća obavlja demodulaciju, vizuelizaciju i detektovanje vitalnih parametara signala [5]. Zahvaljujući „multi-thread“ pristupu, omogućeno je malo zauzimanje sistemskih resursa i spriječeno usporavanje ili blokiranje aplikacije (Poglavlje 3).

Snimanje signala se vrši pomoću klase AudioRecorder, koja dopušta direktan pristup odbircima snimljenog materijala i pruža mogućnost podešavanja niza parametara. Razvijena aplikacija snima signal sa frekvencijom odabiranja (FO) 8 KHz, u mono formatu sa rezolucijom od 16 bita po odbirku. Izabrana frekvencija zadovoljava teoremu o odabiranju, budući da je najveća frekvencija nosioca 1.6 KHz, a frekventni opseg njemu odgovarajućeg modulišućeg signala 10 Hz. Aplikacija radi u realnom vremenu, zahvaljujući primjeni „ping-pong“ tehnike. Naime, aplikacija konstantno snima signal u paketima od po 8000 odbiraka, odnosno u trajanju od po 1 sekunde. Dok jedna grana snima trenutnu sekundu materijala, druga ima na raspolaganju 1 sekundu da obradi prethodno snimljenih 8000 odbiraka, što je više nego dovoljno (Poglavlje 3).

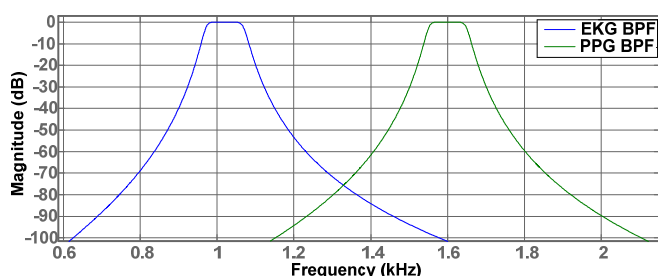
C. Demodulacija signala

Demodulacija signala se obavlja softverskom realizacijom detektora anvelope. U tu svrhu su u Android aplikaciji implementirani digitalni filtri primjenom diferencnih jednačina (1) i (2), te odgovarajućih koeficijenata (a_i , b_i). Ovi koeficijenti dobijeni su korišćenjem Matlabovog paketa „Filter Design & Analysis“. Zadržani su u formatu dvostruke

preciznosti i nisu dalje optimizovani zbog dovoljne procesorske snage i raspoloživih memorijskih kapaciteta Android uređaja.

Prvi korak u demodulaciji je propuštanje signala kroz filtre propusnike opsega učestanosti (BPF), sa propusnim opsezima ± 50 Hz oko nosećih učestanosti (1.020 KHz za EKG i 1.6 KHz za PPG/temperaturni signal). Na ovaj način se vrši izdvajanje EKG i PPG/temperaturnog signala, kao i otklanjanje neželjenih spektralnih komponenti. Ovi filtri su implementirani kao filtri sa beskonačnim impulsnim odzivom (IIR) desetog reda, sa diferencnom jednačinom (1) i amplitudskim karakteristikama prikazanim na Slici 3.

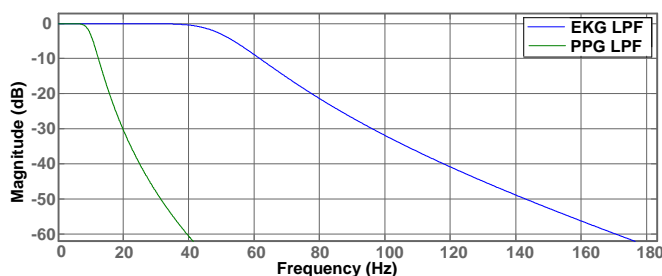
$$y[n] = \sum_{i=0}^{10} b_i \cdot x[n-i] - \sum_{i=0}^{10} a_i \cdot y[n-i] \quad (1)$$



Slika 3. Amplitudska karakteristika EKG i PPG BPF

Nakon toga, razdvojeni signali se propuštaju kroz zasebne niskopropusne filtre (LPF), sa graničnim frekvencijama od 50 Hz za EKG i 10 Hz za PPG signal. Oni su takođe implementirani kao IIR filtri petog reda, sa diferencnom jednačinom (2) i amplitudskim karakteristikama prikazanim na Slici 4.

$$y[n] = \sum_{i=0}^5 b_i \cdot x[n-i] - \sum_{i=0}^5 a_i \cdot y[n-i] \quad (2)$$



Slika 4. Amplitudska karakteristika EKG i PPG LPF

Nakon LPF-a envelope signala su detektovane i možemo pristupiti njihovom crtanju. Da bi obezbijedili dobar odziv aplikacije, crta se svaki deseti odbirak demodulisanog signala, što vizuelno ne degradira signal.

D. Detekcija QRS kompleksa

EKG signal se sastoji iz P, Q, R, S, J i T talasa. U kardiologiji, od posebnog značaja je grupa Q, R i S talasa koja se naziva QRS kompleksom. Postoje brojni algoritmi za njegovu detekciju, od kojih su neki optimizovani za upotrebu u „embedded“ sistemima [6]. U ovom radu implementiran je metod prvog i drugog izvoda, koji je razvio Balda sa saradnicima 1977.god. [7], a kasnije je više puta unapređivan (Ahlstrom i Tomkins 1983.god., Friesen 1990.god.). Treba napomenuti da su ovi algoritmi razvijeni u vrijeme kada su signali u MIT-BIH bazi bili snimani sa FO=200 Hz. Budući

da je u ovom radu FO=8 KHz, potrebno je izvršiti pododabiranje signala sa faktorom 40.

Dakle, nakon demodulacije EKG signala, pristupa se detekciji QRS kompleksa koja se sastoji iz sljedećih koraka :

- Ulazni signal $x[n]$ se normalizuje dijeljenjem sa svojom najvećom vrijednošću
- Prvi izvod u tri tačke ulaznog signala pomnožen odgovarajućim težinskim koeficijentom (OTK)

$$y_0[n] = 2 \cdot |x[n]| - x[n] \quad (3)$$

- Drugi izvod aproksimiran kao prvi izvod u tri tačke od prvog izvoda u tri tačke, pomnožen sa OTK

$$y_1[n] = 1.5 \cdot |x[n]| - 2x[n-2] + x[n] \quad (4)$$

- Kombinovanje dva rezultata uz OTK

$$y_2[n] = 1.3 \cdot y_0[n] + 1.1 \cdot y_1[n] \quad (5)$$

- Propuštanje $y_2[n]$ kroz filtar sa kliznim usrednjavanjem u 8 tačaka

$$y_3[n] = \frac{1}{8} \cdot \sum_{k=0}^7 y_2[n-k] \quad (6)$$

- Rezultat $y_3[n]$ se upoređuje sa pragom 1. Ako je prag pređen, narednih 8 odbiraka se takođe upoređuje sa pragom. Ako je bar njih 6 prešlo prag, smatra se da je detektovan QRS kompleks. Broj QRS kompleksa u jednom minutu predstavlja vrijednost srčanog pulsa.

E. Detekcija maksimuma PPG signala

Za detekciju maksimuma PPG signala (maxPPG) koristi se metod prvog izvoda. Ukratko, algoritam se sastoji iz sljedećih koraka :

- Prvo se računaju pragovi demodulisanog signala i njegovog prvog izvoda. Oni se postavljaju na 70 % vrijednosti maksimuma ovih signala u prethodnih 1.5 sekundi.
- Ukoliko je neki od odbiraka prvog izvoda jednak nuli ili mijenja svoj predznak sa pozitivnog na negativan u odnosu na prethodni, znači da njemu odgovarajući odbirak demodulisanog signala predstavlja lokalni maksimum.
- Da bi utvrdili da li taj lokalni maksimum odgovara maksimumu PPG signala, provjeravamo sljedeće :
1) da li je neposredno prije tog odbirka prvi izvod prešao prag
2) da li je u tom odbirku PPG signal veći od praga

Broj detektovanih maxPPG-a u jednom minutu odgovara vrijednosti srčanog pulsa. Takođe, važan parametar koji ova aplikacija izračunava je i PTT interval. On se računa kao vremenska rezlika između QRS kompleksa i neke od karakterističnih tačaka PPG signala (25 %, 50 % ili 100 % maksimalne vrijednosti) u toku istog srčanog pulsa. Karakteristična tačka u ovom radu izabrana je da bude upravo maxPPG. Da bi uopšte bilo moguće detektovati PTT interval, EKG i PPG signali moraju biti obrađivani paralelno, što se upravo postiže „multi-thread“ pristupom.

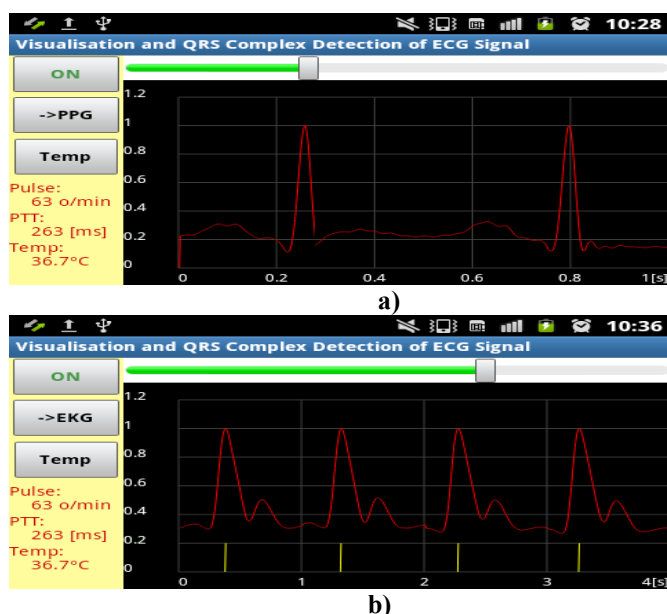
F. Mjerenje temperature

Mjerenje temperature se obavlja pritiskom odgovarajućeg dugmeta na ekranu Android aplikacije. Kao što je već rečeno, tada se šalju tri istovjetna modulirana pravougaona impulsa čije je trajanje srazmjerno izmjerenoj temperaturi. Android

aplikacija vrši demodulaciju ovih impulsa, a potom računanje njihovog trajanja na osnovu detektovanja uzlazne i silazne ivice. Ukoliko trajanje bar dva od tri impulsa ne odstupa međusobno više od 5%, proračunava se i prikazuje nova vrijednost temperature. Ukoliko je odstupanje veće, znači da temperatura nije dobro izmjerena i postupak se ponavlja.

G. Izgled Android aplikacije

Izgled korisničkog interfejsa Android aplikacije osmišljen je da bude jednostavan, intuitivan i lak za korišćenje, Slika 5. Tu se nalazi taster za pokretanje i zaustavljanje monitoringa, taster za prikazivanje EKG ili PPG signala, kao i taster za mjerenje temperature. Ispod njih se prikazuju informacije o srčanom pulsu, PTT intervalu i temperaturi. Pomoću klizača bira se između nekoliko vidova zumiranja po vremenskoj osi (1s, 2s, 4s), kao i indikacije detektovanih QRS kompleksa ili maxPPG-a.



Slika 5. a) Demodulisani EKG signal b) Demodulisani PPG signal sa indikacijom detekcije maxPPG

3. REZULTATI

U ovoj fazi istraživanja, sistem je testiran prema sljedećim kriterijumima : ispravnost rada, detekcije QRS kompleksa, zauzimanje sistemskih resursa (radne memorije i procesorskog vremena), vremena potrebnog za izvršavanje pojedinih operacija aplikacije. Signali su uzimani od volontera i iz MIT-BIH i MIMIC II baze podataka EKG i PPG signala. Signali iz baza su generisani korišćenjem ELVIS II⁺ LabView platforme i posebno razvijenih virtuelnih instrumenata.

Preliminarno testiranje tačnosti detekcije QRS kompleksa na polučasovnom aritmijском signalu broj 100 iz MIT-BIH baze, sa 2273 označenih QRS kompleksa, pokazalo je zadovoljavajuću tačnost detekcije od 97.49%.

Kao što je već rečeno, aplikacija zauzima veoma malo sistemskih resursa. Testni Android pametni telefon je

Samsung Galaxy Ace Plus S7500 (512 MB RAM, procesor 1 GHz). Iako ovaj uređaj spada u nižu srednju klasu pametnih telefona, razvijena aplikacija na njemu zauzima svega 4.47 MB RAM-a trošeći pri radu 2.31 % procesorskog vremena. U Tabeli 1 dat je prikaz vremena potrebnog za izvršavanje pojedinih operacija aplikacije.

Tabela 1. Vrijeme potrebno za pojedine operacije

Operacija	Potrebno vrijeme
Vizuelizacija- crtanje 800 odbiraka	26-33 ms
Detekcija anvelope	45-75 ms
Detekcija QRS kompleksa	570-700 us

4. ZAKLJUČAK

U ovom radu prikazan je način mjerenja vitalnih fizioloških parametara primjenom pametnih telefona. EKG, PPG i temperaturni signali se pojačavaju, filtriraju, modulišu, a zatim prosljeđuju na mikrofonski ulaz uređaja. Android aplikacija vrši snimanje, demodulaciju i vizuelizaciju signala, detekcije QRS kompleksa i maksimuma PPG signala, računanje srčanog pulsa i PTT intervala, kao i mjerenje temperature. Preliminarno testiranje u pogledu tačnosti detekcije QRS kompleksa i zauzimanja sistemskih resursa pokazalo je zadovoljavajuće rezultate. Ovakav sistem bi mogao naći primjenu u prenosnim uređajima za zdravstvenu negu i telemedicinu.

LITERATURA

- [1] C. G. Scully, J. Lee, J. Meyer, "Physiological parameter monitoring from optical recordings with a mobile phone", IEEE Trans Biomed Eng 59 (2), 2012, pp.303-306.
- [2] A. T. Reisner, G. D. Clifford, and R. G. Mark, "The Physiological Basis of the Electrocardiogram", in "Advanced Methods for ECG Analysis", (Eds. G. D. Clifford, F. Azuaje and P. E. McSharry), Artech House, Boston, 2006.
- [3] J.G. Webster : "Design of Pulse Oximeters", Taylor and Francis Group, 1997.
- [4] John H. Davies, "MSP430 microcontroller basics", Burlington, MA, 2008.
- [5] G. Milete, A. Stroud, "Professional android sensor programming", Wrox, 2012.
- [6] R. Stojanović, S. Knežević, D. Karadaglić, and G. Devedžić, "Optimization and Implementation of the Wavelet Based Algorithms for Embedded Biomedical Signal Processing", Journal of Computer Science and Information Systems, ComSIS, Vol. 10, No. 1, 2013, pp. 503-523.
- [7] R. A. Balda , "The HP ECG analysis program", Trends in Computer-Processed Electrocardiograms, J. H. van Bommel and J. L. Willems, Eds. North Holland, 1977, pp.197-205.

KONCEPT ZA EKSPERTNI SISTEM ZA AUTOMATIZOVANU EKG DIJAGNOSTIKU A CONCEPT OF EXPERT SYSTEM FOR AUTOMATED ECG DIAGNOSIS

Roman Golubovski, Faculty of Electrical Engineering, University "Goce Delčev" - Štip,
Republic of Macedonia

Sadržaj: Ekspertni Sistem (ES) je softver koji koristi bazu znanja ljudske ekspertize za rešavanje problema, ili rasčiščavanje nejasnoća gde se uobičajeno konsultuju eksperti. Ekspertni sistemi se uobičajeno koriste za specifične uske domene veštačke inteligencije (AI) gde je problematika sveobuhvatno pokrivena "znanjem". Ekspertni sistemi mogu implementirati i učenje. Jedna od fundamentalnih implementacionih područja je medicinska dijagnostika. Među najprominentnijim njenim aplikacijama je automatizovana EKG dijagnostika, jer EKG morfologija je potpuno determinirana. Predloženi ES se oslanja na standardni 12-kanalni EKG set - relevantni naponski parametri i vremenski segmenti i intervali. Preliminarno testiranje na EKG snimke idu u prilog visokoj tačnosti dijagnostike konzistentno sa ekspertizom iskusnih kardiologa.

Abstract: An expert system (ES) is software that uses a knowledge base of human expertise for problem solving, or clarify uncertainties where normally one or more human experts would need to be consulted. Expert systems are most common in a specific problem domain, and are a traditional application and/or subfield of artificial intelligence (AI). Expert systems may or may not have learning components. One of the fundamental implementation areas is the medical diagnostics, most prominent being the automated ECG diagnostics since the ECG morphology is completely determined. The proposed ES relies on the standard 12-lead ECG set of relevant voltage deflections (amplitudes) and time segments and intervals (durations). Preliminary testing against ECG records promise high accuracy consistent with diagnostic opinions of expert cardiologists.

1. INTRODUCTION TO EXPERT SYSTEMS

A common and acceptable definition of the Expert System (ES) is that it is a special type of system built upon detailed experience and knowledge acquired by human's brain, and formatted in such a way that allows a computer to solve problems from within a specific domain, that normally need human expertise. In the context of this paper the "problem" is the ECG diagnosis and the "expert" is a skilful cardiologist.

The fundamental concept behind the ES is trying to mimic the human reasoning which is still impossible to achieve in full. The human mental process is internal, and it is too complex to be represented as an algorithm. However, most experts are capable of expressing their knowledge in the form of rules for problem solving. The term rule in AI, which is the most commonly used type of knowledge representation, can be defined as an IF-THEN structure that relates given information or facts in the IF part (the condition) to some action in the THEN part (the action). A rule provides some description of how to solve a problem. Rules are relatively easy to create and understand. Rules can represent relations, recommendations, directives, strategies and heuristics. In the context of this paper rules test ECG parameters with threshold values in their conditional part and draw partial diagnostic conclusions.

The foundation of the modern ES is the **production** rule system model proposed by Newell & Simon (figure 1).

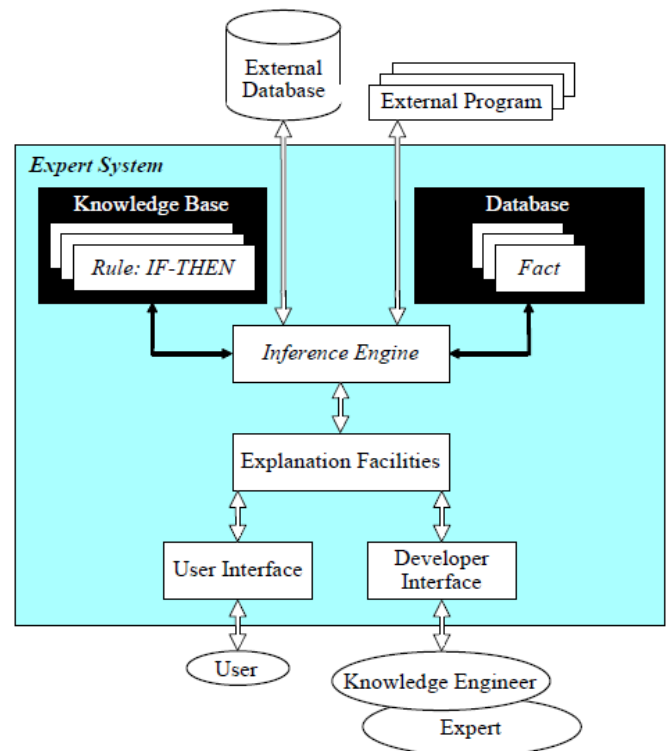


Figure 1. Complete structure of a rule-based ES

The production model is based on the idea that humans solve problems by applying their knowledge (expressed as production rules) to a given problem represented by problem-

specific information. The **knowledge base** contains the domain knowledge useful for problem solving. When the condition part of a rule is satisfied, the rule is said to **fire** and the action part is executed. The **database (working memory)** includes a set of facts used to match against the IF (condition) parts of rules stored in the knowledge base. The **inference engine** carries out the reasoning whereby the expert system reaches a solution. It links the rules given in the knowledge base with the facts provided in the database. The **explanation facilities** enable the user to ask the expert system **how** a particular conclusion is reached and **why** a specific fact is needed. An expert system must be able to explain its reasoning and justify its advice, analysis or conclusion. The **user interface** is the means of communication between a user seeking a solution to the problem and an expert system.

ES is widely implemented in modern ECG devices, but it is still just a helping tool to assist physicians in interpreting ECG (it must not be used as unverified diagnostic source). So being able to explain the deductive reasoning (forward or data-driven inference chaining) helps the physician to validate the conclusions.

2. IMPLEMENTATION OF THE ECG DATA

The standard 12-lead ECG is consisted of the six limb leads in the vertical plane - aVR, aVL, aVF, DI, DII and DIII (figure 2), and the six precordial leads in the horizontal plane - V1, V2, V3, V4, V5 and V6 (figure 3).

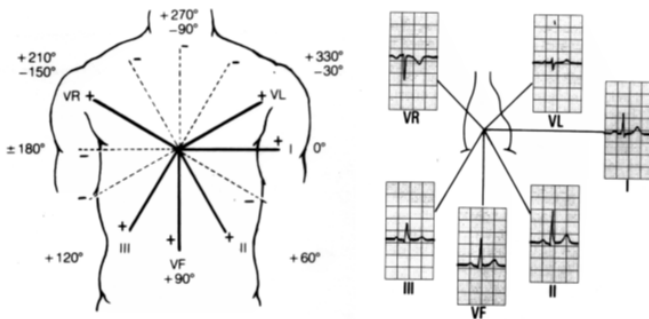


Figure 2. The limb leads

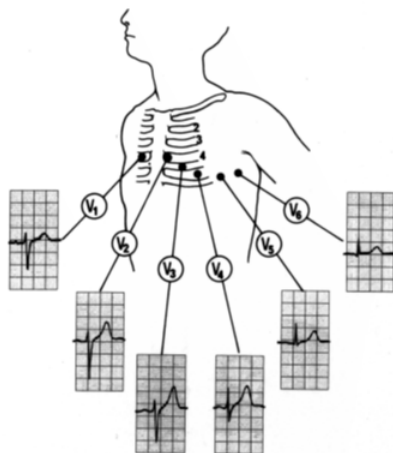


Figure 3. The precordial leads

An ECG record is a non-invasive diagnostic tool used for the assessment of a patient’s heart condition. The features of the ECG, when recognized by simple observations, and combined with heart rate, can lead to a fairly accurate and fast diagnosis.

So far there have been a number of successful developments in the automated diagnosis domain. Like the ES for ECG analysis [1] that works by hierarchically organizing the knowledge in a context tree, where diseases are recognized by traversing the tree having symptoms as nodes and diseases as leafs. Others [2] have used time and frequency domain parameters and correlation constants derived from ECG signals as inputs for their expert system. A software for ECG beat detection and classification [3] had been developed and made available as an open source system for use by researchers. A technique for analyzing ECG signals using hidden Markov models for beat segmentation and classification [4] had also been proposed. The use of neural networks for automatic ECG analysis for the classification of different cardiac abnormalities [5] had also been explored.

Typical ECG morphology is presented on figure 4:

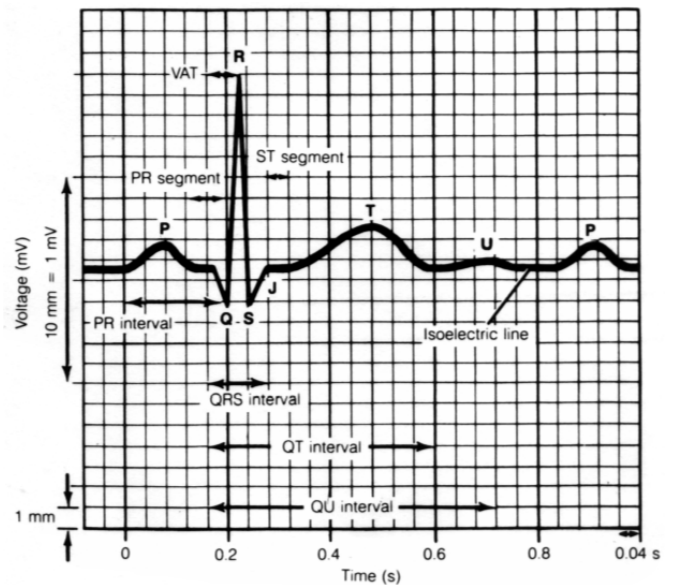


Figure 4. Typical ECG morphology

The proposed expert system concept/algorithm is a rule-based decision support system to aid physicians in the diagnosis of heart diseases. The set of ECG parameters of all 12 leads is related to voltage deflections and time segments/intervals (figure 4) of the:

- P wave
- QRS complex (Q, R and S strokes)
- ST segment
- T wave
- PR interval
- QRS duration
- QT interval
- RR interval
- PP interval

3. THE EXPERT SYSTEM CONCEPT

The concept of the proposed ES is shown in figure 1. The framework of the rule based expert system consists of:

- Facts - input obtained from derived parameters of the 12-lead ECG
- Knowledge Base - a set of rules developed in consultation with experts based on heart rate and ECG wave characteristics (parameters)
- Inference Engine - matches the input (facts) with a rule in the rule-base to identify the abnormality
- Database - stores the patient's personal details, inputs, diagnosed results and user's comments (suggestions)
- Explanation Facilities - provides the forward inference chaining in support of the proposed diagnosis

After thorough analysis of the available relations between the diagnostic conclusions and the corresponding sets of input values, all ECG diagnoses are divided in the following list of groups, which are chained for the process of determining diagnosis, whereas each of the group is consisted of *familiar* diagnostic statements:

- Preliminaries
- Conduction Abnormalities
- Hypertrophy
- Myocardial Infarct
- ST Elevation
- ST Depression
- T Wave Abnormalities
- Rhythm Statements

The algorithm passes through the chain of groups performing lists of tests within them and passing results to their next group of tests. Before evaluating the group list of tests it first performs the group's *skip tests* (figure 5). Skip tests decide weather group's evaluation is feasible or available data is insufficient resulting in skipping the current group tests.

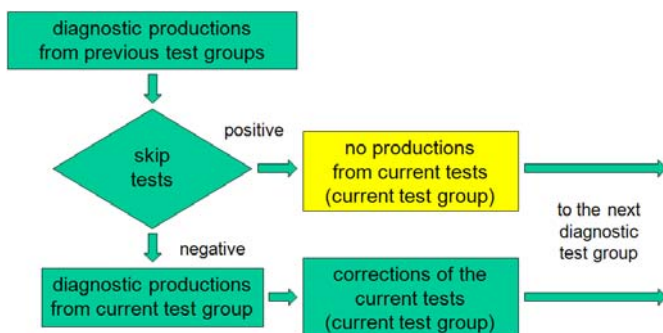


Figure 5. Algorithm of diagnostic chain evaluation

A condition statement follows each interpretive statement. Conditions and their meanings are listed in the table on figure 6.

The *Preliminaries* group identifies following conditions: Arm Lead Reversal and Dextrocardia; Wolff-Parkinson-

White (WPW); Atrial Enlargement; Axis Deviation; Low Voltage; S1-S2-S3 Pattern; Pulmonary Disease.

The *Conduction Abnormalities* are: Right Bundle Conduction; Left Bundle Conduction; Non Specific Conduction Abnormality.

The *Hypertrophies* are: Right Ventricular Hypertrophy; Left Ventricular Hypertrophy.

The *Myocardial Infarctions* are: Anterior Infarct; Septal Infarct; Anteroseptal Infarct; Lateral Infarct; Anterolateral Infarct; Inferior Infarct; Inferior Infarct with Posterior Extension; Infarct Suppressions.

The *ST Elevation* diagnoses are: ST Segment Elevation; Early Repolarization; Pericarditis; Anterior and Septal Epicardial Injury; Lateral Epicardial Injury; Inferior Epicardial Injury.

The *ST Depression* diagnoses are: ST Depression; T Wave Abnormality and Ischemia.

The *T Wave Abnormalities* are: T Wave Abnormality, Nonspecific.

Condition	Meaning
Normal ECG	Normal
Atypical ECG	An unusual pattern has been observed but has no specific significance.
Borderline ECG	Criteria have limited specificity or prognostic significance or where only minimal criteria are met.
Abnormal ECG	Abnormal
Abnormal Rhythm ECG	Abnormal Rhythm
No Further Interpretation Possible	Upon detecting the phenomenon in question, no further useful interpretation of the record is possible.
No Condition Associated	Used with statement prefixes and suffixes.

Figure 6. General condition statements and explanation

Rhythm Statements: Sinus-, Atrial-, Junctional-, Supraventricular- (Tachycardia / Rhythm / Bradycardia); Undetermined (regular) rhythm; Atrial fibrillation; Atrial flutter; Electronic ventricular pacemaker. Known modifiers are also used for recognized specific condition details.

4. ES PRODUCTION EXAMPLES

First Preliminaries test checks for possible lead reversal:

IF	THEN
No Q in lead I and R amplitude < 150uV in lead I or Q amplitude > 0 in lead I and P axis > 90 and PR duration >= 110 ms and QRS axis > 90	PRINT "Arm leads reversed" REASON: <i>Inverted P & QRS in lead I</i>
If above criteria are met and R amplitude < 500 uV in lead V6 and Maximum S amplitude > Maximum R amplitude in lead V6 and P amplitude < 20 uV in lead V6 and P' amplitude < -20 uV in lead V6	PRINT "Dextrocardia" REASON: <i>Inverted P & QRS in V6</i>

Figure 7. Test for Arm Lead Reversal and Dextrocardia

IF	THEN
R amplitude > 100 uV in V1 & V2 and R duration > 20 ms in V1 and V2 and no S in V1 or V2 or R' amplitude > 100 uV in V1 & V2 and R' duration > 20 ms in V1 & V2 and no S' in V1 or V2	PRINT "RsR' (QR) in V1/V2 consistent with right ventricular conduction delay"
Either of the above is true and QRS duration > 90 ms and QRS duration < 120 ms and S duration >= 40 ms in any 2 leads of I/aVL/V4/V5/V6	PRINT "Incomplete right bundle branch block" REASON: 90+ ms QRS duration, terminal R in V1/V2, 40+ ms S in I/aVL/V4/V5/V6
QRS duration >= 120 ms and S duration >= 40 ms in any 2 leads of I/aVL/V4/V5/V6 and R duration < 100 ms in any 4 leads of I/aVL/V4/V5/V6 and QRS area > 0 in V1 and V1 does not terminate in S or S' or QRS duration > 105 ms and S duration >= 60 ms in any 3 leads of I/aVL/V4/V5/V6 and R duration > 60 ms in V1 and QRS area > 0 in V1	PRINT "Right bundle branch block" REASON: 120+ ms QRS duration, upright V1, 40+ ms S in I/aVL/V4/V5/V6
The test for right bundle branch block is positive and R amplitude > 1500 uV in V1 and QRS axis > 110	PRINT "Right bundle branch block plus possible Right Ventricular Hypertrophy" REASON: RBBB, 1.5 mV R in V1, RAD

Figure 8. Right Bundle Conduction test

Figure 8 shows the amount of processing needed for RBBB determination. Figure 9 shows the *skip* tests for the Right Ventricular Hypertrophy:

SKIP TEST IF
The test for Right Bundle Branch Block is positive
or the test for Left Bundle Branch Block is positive
or age < 16
or S amplitude < 250 uV in I
or S amplitude > 1000 uV in V1
or QRS axis < 60
or QRS duration > 140 ms and net QRS amplitude < 0 in V1
or Q amplitude > S amplitude and R exists in I

Figure 9. Skip test for Right Ventricular Hypertrophy

Test for Inferior Infarct with Posterior Extension:

SKIP TEST IF	
The test for an inferior infarct is negative	
The test for Right Bundle Branch Block is positive	
IF	THEN
Q amplitude > 0 in V1 or V2 and R duration >= 40 ms in V1 & V2 or R duration >= 35 ms and QRS net amplitude > 0 in V1 or V2 or R duration >= 30 ms and QRS net amplitude > 0 in V1 and V2	Append "with posterior extension" "prominent R Wave in V1/V2" to the inferior infarct statement

Figure 10. Testing for Inferior Inf. with Posterior Ext.

5. CONCLUSION

The proposed ES conceptualizes the standard cardiologists' reasoning, following a cardiology expertise supported by the vast clinical ECG experience. The data-chaining of the parameters tests follow the well established ECG diagnostic procedures, therefore high accuracy and reliability is expected upon thorough clinical performance investigation.

REFERENCES

- [1] N. E. Mastorakis, N. J. Theodorou, E. S. Rota, "EKG.PRO: An Expert System for ECG Analysis," *Third IEEE Mediterranean Conference on Control and Automation*, July, 1995
- [2] S. M. Patil, S. D. Bhagwat, "ECG Analysis - An Expert System Approach", *Proceedings of first regional Conference of IEEE – EMBS*, 1995, pp. 2/48-2/49
- [3] P. Hamilton, "Open Source ECG Analysis", *Computers in Cardiology*, Vol.29, 2002, pp.101-104
- [4] R. V. Andreao, B. Dorizzi, J. Boudy, "ECG signal analysis through Hidden Markov Models", *IEEE Transactions on Biomedical Engineering*, Volume 53, Issue 8, 2006, pp. 1541-1549
- [5] R. Silipo, C. Marchesi, "Artificial neural networks for automatic ECG analysis", *IEEE Transactions on Signal Processing*, Vol. 46, Issue 5, 1998, pp. 1417-1425

ANALIZA EFIKASNOSTI KVADRATNIH KONVOLUCIONIH JEZGARA KOD PROCENE FREKVENCIJE SIGNALA

ANALYSIS OF EFFICIENCY QUADRATIC CONVOLUTION KERNELS TO ESTIMATION FREQUENCY OF SIGNALS

Nataša Savić, Zoran Milivojević, *Visoka tehnička škola, Niš, Srbija*
Darko Brodić, *Univerzitet u Beogradu, Tehnički fakultet Bor, Srbija*

Sadržaj: U prvom delu rada opisana su parametarska kvadratna konvoluciona jezgra i to Dodgsonovo jezgro i jezgro predloženo u ovom radu. Određeni su analitički izrazi za poziciju maksimuma interpolacione funkcije. U drugom delu rada prikazani su rezultati primene konvolucione interpolacije sa opisanim parametarskim jezgrima kod procene frekvencije signala u spektralnom domenu. Kao mera uspešnosti procene koristi se srednja kvadratna greška. Analizom rezultata koji su prikazani grafički i tabelarno određena je optimalna vrednost parametra jezgra i odgovarajuća prozorska funkcija.

Ključne reči: interpolacija, parametarsko konvoluciono jezgro, procena frekvencije.

Abstract: The first section are described parametric quadratic convolution kernel, Dodgson's kernel and kernel proposed in this paper. Certain analytical expressions for the position of the maximum interpolation function. In the second part of the paper presents the results of applying the convolution ininterpolacije described parametric kernel code evaluation frequency signal in the spectral domain. As a measure of performance evaluation using the mean squared error. The analysis results are presented graphically and in tabular form is determined by the optimal value of the kernel parameter and the corresponding window function.

Key words: interpolation, parametric convolution kernel, estimation frequency

1. UVOD

Teorija aproksimacije (*lat. approximation*-približnost) i interpolacije (*lat. interpolatio*-umetanje) privlači pažnju mnogih matematičara i naučnika od davnina. 1853. godine veliki ruski matematičar Čebišev dok je radio na problemu veze uređaja koji prevodi linearno kretanje parne mašine u kružno kretanje točka razmatrao je pitanje inerpolacije funkcije. Prilikom obrade podataka često se javlja potreba za procenom nekih vrednosti. Rezultati merenja uglavnom su izraženi na diskretnom skupu tačaka. Na osnovu tih podataka procenjuju se međuvrednosti, odnosno data diskretna funkcija zamenjuje se neprekidnom funkcijom na tom skupu ili na još širem skupu. Ovaj postupak naziva se interpolacija, a neprekidna funkcija naziva se interpolaciona funkcija Tačke u kojima se diskretna i neprekidna funkcije podudaraju su interpolacioni čvorovi. Kao interpolacione funkcije se koriste polinomi, segment polinomi (*eng. piecewise polynomials*), racionalne funkcije, eksponencijalne i trigonometrijske funkcije. Interpolacija polinomima je najjednostavnija, međutim funkciju zadatu diskretnim vrednostima nije praktično aproksimirati na celom intervalu jednim polinomom, jer se javljaju oscilacije na krajevima intervala. Kako bi se povećala tačnost i brzina izračunavanja pribegava se podeli intervala na segmente sa različitim polinomima u segmentima.

U mnogim oblastima tehnike, a naročito kod obrade slike javlja se potreba za interpolacijom. Karakteristični slučajevi

su kod promene dimenzije slike, rotacije slike i dr. Pri obradi slike intenzivno se primenjuje konvoluciona interpolacija [2]. Za potrebe konvolucione interpolacije razvijeno je mnogo interpolacionih jezgara. Počev od jezgra drugog reda (*eng. quadratic kernel*), preko trećeg reda (*eng. cubic kernel*) i na dalje [1-3]. Uvode se parametarska jezgra, jer omogućavaju primenu kod različitih problema izborom optimalnog parametra [4,5]. Za potrebe slike pored kvadratnih intenzivno se koriste parametarska kubna jezgra. Kvadratna parametarska jezgra zbog svoje male numeričke složenosti odlikuju se velikom brzinom izvršavanja konvolucione interpolacije. Sa druge strane zbog male dužine jezgra (broja interpolacionih čvorova) kvadratna jezgra iskazuju manju preciznost u proceni interpolacione vrednosti. Kubna jezgra, u mnogim slučajevima predstavljaju kompromis između numeričke složenosti i preciznosti koja odlikuje jezgra većeg stepena. U literaturi su posebno popularna jezgra koja je predložio Keys u svom radu [6]. Zbog toga su, jezgra koja su izvedena iz predloženog parametarskog kubnog jezgra nazvana Kejsovim jezgrima. Pored njih, za potrebe obrade slike zapaženu ulogu imaju Grevilleova jezgra [2,7]. Kod diskretnih električnih signala često se javlja potreba za interpolacijom u realnom vremenu (gubitak sempla, promena frekvencije odmeravanja, itd.). Posebno su aktuelni problemi procene parametara kao što su frekvencija i faza, gde je neophodno vršiti interpolaciju.

U ovom radu analizirana je procena frekvencije sinusoidnog signala i to u spektralnom domenu. Da bi se izvršila

obrada u spektralnom domenu potrebno je najpre izvršiti diskretnu Furijeovu transformaciju (**eng.** *Discrete Fourier Transforms*, **DFT**). DFT kao rezultat daje aproksimaciju spektra vremenskog signala. Osnovna frekvencija predstavlja samo jednu komponentu u frekvencijskom domenu. Međutim, u slučaju kada je realna frekvencija različita od frekvencije izračunavanja DFT-a procena spektra biće pogrešna. Grešku dodatno povećava pojava curenja spektra. Problem estimacije tačne pozicije u spektralnom domenu obavljen je pomoću kvadratnih parametarskih interpolacionih jezgara. Prvo jezgro je obrađeno u [1]. Drugo jezgro su autori formirali po ugledu na ovo jezgro uz postavljanje uslova koji su definisani u [8]. Preciznost procene frekvencije merena je pomoću srednje kvadratne greške (**engl.** *Mean Square Error*, **MSE**). Povećanje preciznosti postignuto je obradom vremenskog diskretnog signala nekim klasičnim prozorskim funkcijama (Hann-ova, Hamming-ova, ...) i izborom optimalnog parametra konvolucionog jezgra. U radu su prikazani analitički izrazi za poziciju maksimuma, koji su originalni rezultati autora. Primena analitičkih izraza otklanja potrebu za konvolucijom što bitno ubrzava process.

Dalja organizacija rada je sledeća. U sekciji 2 opisana je konvoluciona interpolacija. U sekciji 2.1 prikazano je Dodgson-ovo interpolaciono jezgro. U sekciji 2.2. izvršena je konstrukcija kvadratnog interpolacionog jezgra. U sekciji 2.3 prikazan je algoritam procene frekvencije. Eksperimentalni rezultati i analiza prikazani su u sekciji 3.

2. KONVOLUCIONA INTERPOLACIJA

Konvolucija neprekidnih funkcija $f(x)$ i $g(x)$ definiše se sledećim izrazom:

$$f(x) * g(x) = \int f(t)g(x-t)dt. \quad (1)$$

Za diskretne funkcije konvolucija je određena izrazom:

$$f(n) * g(n) = \sum_k f(k)g(n-k). \quad (2)$$

Neka je f semplovana funkcija a g odgovarajuća interpolaciona funkcija. Tada je u interpolacionim čvorovima x_k vrednost semplovane funkcije jednaka vrednosti interpolacione funkcije, tj. $f(x_k)=g(x_k)$. Za ravnomerno raspoređene podatke mnoge interpolacione funkcije mogu se zapisati kao:

$$g(x) = \sum_k c_k u\left(\frac{x-x_k}{h}\right). \quad (3)$$

U (3), c_k su parametri koji zavise od semplovanih podataka, a u je interpolaciono jezgro, h uzorkovani priraštaj.

2.1. Dodgson-ovo parametarsko jezgro

U radu [1] definisano je kvadratno parametarsko interpolaciono jezgro sa:

$$r(f) = \begin{cases} -2\alpha|f|^2 + \frac{1}{2}(\alpha+1), & |f| \leq \frac{1}{2} \\ \alpha|f|^2 + \left(-2\alpha - \frac{1}{2}\right)|f| + \frac{3}{4}(\alpha+1), & \frac{1}{2} < |f| \leq \frac{3}{2} \\ 0 & |f| > \frac{3}{2} \end{cases}. \quad (4)$$

2.2. Predloženo parametarsko jezgro

Posmatrajmo simetrično jezgro definisano po delovima polinomima drugog stepena na intervalima $[-2,-1]$, $[-1,1]$, $(1,2]$:

$$r(f) = \begin{cases} a|f|^2 + b, & |f| \leq 1 \\ c|f|^2 + d|f| + e, & 1 < |f| \leq 2 \\ 0 & |f| > 2 \end{cases}. \quad (5)$$

Neka je interpolaciono jezgro neprekidno i jednako nuli u interpolacionim čvorovima, a neka ima vrednost jedan u nuli. Kako je:

$$r(0) = 1 \Rightarrow b = 1, \quad (6)$$

$$r(1) = 1 \Rightarrow a = -1, \quad (7)$$

$$r(2) = 0 \Rightarrow 4c + 2d + e = 0, \quad (8)$$

$$\lim_{x \rightarrow 1^-} r(f) = \lim_{x \rightarrow 1^+} r(f) \Rightarrow a + b = c + d + e. \quad (9)$$

Kako sistem linearnih jednačina(6)-(8) ima 4 jednačine, a 5 nepoznatih to će jedna nepoznata biti proizvoljna. Otuda je:

$$r(f) = \begin{cases} -|f|^2 + 1, & |f| \leq 1 \\ -2\alpha|f|^2 + 6\alpha|f| - 4\alpha, & 1 < |f| \leq 2 \\ 0 & |f| > 2 \end{cases}. \quad (10)$$

2.3. Algoritam procene frekvencije

Agoritam procene frekvencije diskretnog sinusoidalnog signala realizuje se u sledećim koracima:

Korak 1: Vremenski kontinualan signal $x(t)$ konvertuje se u diskretni signal $x(n)$ vremenskim odmeravanjem sa periodom T :

$$x(nT) = x(t)|_{t=nT}. \quad (11)$$

Korak 2: Signal $x(n)$ deli se na frejmove dužine N :

$$x_l(0 : N-1), \quad (12)$$

gde je $l=1:L$ redni broj frejma.

Korak 3: Vršiti se obrada prozorskom funkcijom $w(n)$ dužine N .

Korak 4: Nad frejmom x_l primenjuje se DFT dužine NFT , čime se generiše spektar:

$$X_l(f) = DFT(x_l, NFT). \quad (13)$$

Korak 5: U spektru X metodom pikovanja određuje se maksimalna vrednost amplitudske karakteristike spektra, koji se nalazi na frekvenciji f_{max} :

$$X_{\max} = X(f_{\max}). \quad (14)$$

Korak 6: Primena interpolacije nad delom spektra:

$$X(f_{k-1}, f_k, f_{k+1}, f_{k+2}). \quad (15)$$

Rekonstruisana funkcija je:

$$X_r(f) = \sum_{i=-1}^2 p_i r(f-i), \quad (16)$$

gde je $p_i = X(i)$, $r(f)$ je interpolaciono jezgro, i $k \leq f \leq k+1$. Pozicija maksimuma, koja predstavlja procenjenju osnovnu frekvenciju je:

$$f_e = \frac{-B}{2A}. \quad (17)$$

Kod primene interpolacionog jezgra (10) koeficijenti su:

$$A = -2p_{k-1} - p_k - p_{k+1} - 2p_{k+2}, \quad (18)$$

$$B = 2\alpha p_{k-1} + 2p_{k+1} + 2\alpha p_{k+2}. \quad (19)$$

dok su za interpolaciono jezgro (4) koeficijenti:

$$A = \alpha(p_{k-1} - 2p_k + p_{k+1}), \quad (20)$$

$$B = \frac{1}{2}(p_{k-1} + p_{k+1}). \quad (21)$$

Korak 7: Određivanje srednje kvadratne greške procene između realne f_0 i procenjene f_e frekvencije:

$$MSE = \overline{(f_0 - f_e)^2}. \quad (22)$$

3. EKSPERIMENTALNI REZULTATI I ANALIZA

3.1 Eksperiment

Algoritam za procenu frekvencije vremenskog sinusoidalnog signala opisan u sekciji 2.3, primenjen je nad sinusnim vremenskim signalom kome se frekvencija menja u opsegu $f_0=125-140$ Hz. Frekvencija semplovanja je $f_s=8$ kHz, blokovi dužine 32 ms, odnosno $N=256$. Primenjeno je Dodgsonovo i predloženo kvadratno konvoluciono jezgro uz primenu Hamming-ovog, Hann-ovog, Blackman-ovog, pravougaonog, Kaiser-ovog i trougaonog jezgra.

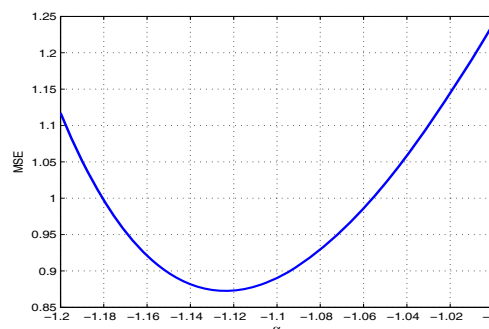
3.2 Rezultati

Vrednosti srednje kvadratne greške u zavisnosti od parametra α za neke prozorske funkcije prikazane su na: sl.1 (Hamming), sl. 2 (Hann), sl. 3 (Blackman), sl. 4 (pravougaoni), sl. 5 (Kaiser) i sl. 6 (trougaoni). U tabeli 1 prikazane su optimalne vrednosti α i minimalne vrednosti MSE.

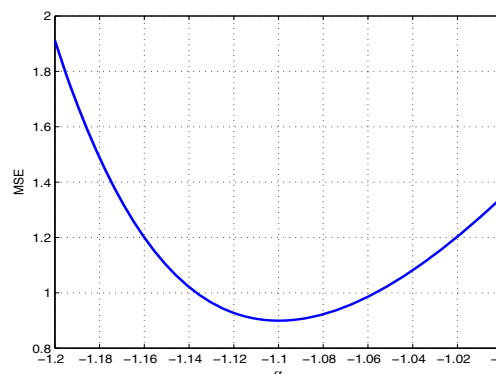
Tbl.1 Minimalna vrednost MSE sa primenom kvadratnog predloženog jezgra.

PROZOR	MSE _{min}	α_{\min}
Hamming	0,8727	-1,125
Hann	0,899	-1,1
Blackman	0,6014	-0,915
Pravougaoni	0,0726	0,01

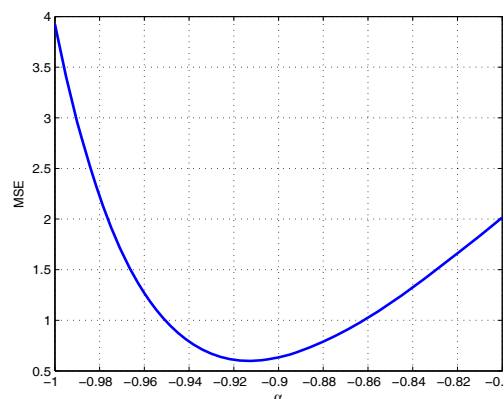
Kaiser	0,963	-1,065
Trougaoni	1,0026	-1,14



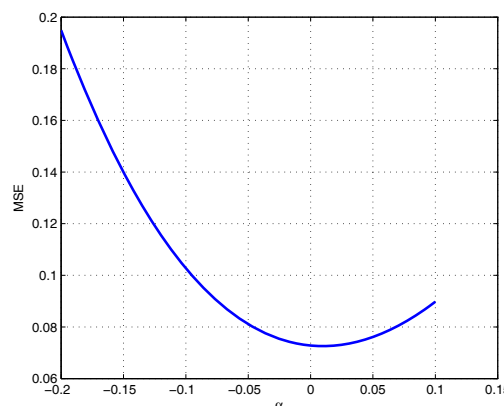
Sl.1. MSE(α) za slučaj primene Hamming-ovog prozora.



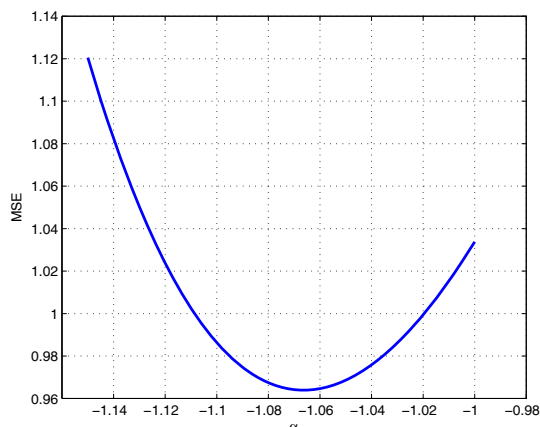
Sl.2. MSE(α) za slučaj primene Hann-ovog prozora.



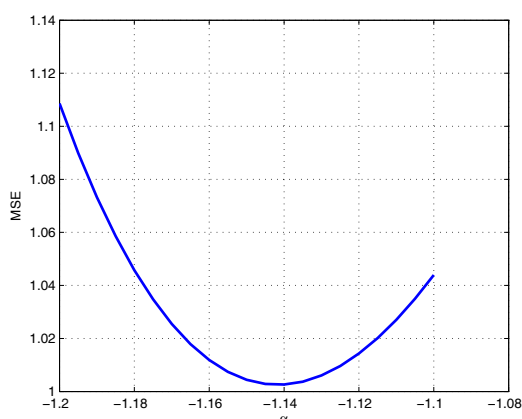
Sl.3. MSE(α) za slučaj primene Blackm-ovog prozora.



Sl.4. MSE(α) za slučaj primene pravougaonog prozora.



Sl.5. $MSE(\alpha)$ za slučaj primene Kaiser-ovog prozora.



Sl.6. $MSE(\alpha)$ za slučaj primene trougaonog prozora.

3.3. Analiza rezultata

Na osnovu rezultata prikazanih na sl.1-6 i tabele 1 zaključuje se da:

a) primena Dodgsonovog konvolucionog jezgra generiše neprihvatljivo velike vrednosti MSE ($MSE > 20$),

b) primena predloženog jezgra dala je optimalne rezultate za pravougaonu prozorsku funkciju. U odnosu na druge prozorske funkcije pravougaona je pokazala bolje rezultate: 91% (Hamming), 92% (Hann, Kaiser), 88% (Blackman), 93% (trougaoni),

c) u poredjenju sa rezultatima primene Kejsovog kubnog interpolacionog jezgra [9], zaključuje se da primena predloženog kvadratnog jezgra dovodi do srednje kvadratne greške koja je: 37.94 (Hamming), 224.75 (Hann), 48.15 (Kaiser), 0.14 (pravougaoni), 601.4 (Blackman) i 358.07 (trougaoni) puta veća.

4. ZAKLJUČAK

U radu su prikazani rezultati primene parametarskih kvadratnih konvolucionih jezgara Dodgsonovog i

predloženog. Dati su analitički izrazi za poziciju maksimuma, kod procene frekvencije u spektralnom domenu, koji su originalni rezultati autora. Detaljnom analizom pokazano je da je Dodgsonovo jezgro proizvodilo neprihvatljivo veliku grešku ($MSE > 20$) dok je predloženo jezgro uz primenu pravougaone prozorske funkcije dalo bolji rezultat. U poredjenju sa Kejsovim kubnim interpolacionim jezgrom predloženo jezgro je proizvodilo znatno veće greške. S obzirom da proces konvolucije sa kvadratnim jezgrom ima manji nivo numeričke složenosti od konvolucije sa kubnim jezgrima primena u sistemima za rad u realnom vremenu predstavlja kompromis između zahtevane brzine i preciznosti.

LITERATURA

- [1] N. A. Dodgson, "Quadratic interpolation for image resampling", *IEEE Trans. Image Processing*, Vol. 6, no. 9, pp. 1322-1326, Sept. 1997.
- [2] E. Meijering, M. Unser, "A Note on Cubic Convolution Interpolation", *IEEE Transactions on Image Processing*, Vol. 12, No. 4, pp. 447-479, April 2003.
- [3] E. Meijering, K. Zuiderveld, M. Viergever, "Image Reconstruction by Convolution with Symmetrical Piecewise nth-Order Polynomial Kernels", *IEEE Transactions on Image Processing*, Vol. 8, No. 2, pp. 192-201, Feb. 1999.
- [4] H. S. Pang, S. J. Baek, K. M. Sung, "Improved Fundamental Frequency Estimation Using Parametric Cubic Convolution", *IEICE Trans. Fundamentals*, Vol. E83-A, No. 12, pp. 2747-2750, Dec. 2000.
- [5] K. S. Park, R. A. Schowengerdt, "Image reconstruction by parametric cubic convolution", *Computer Vision, Graphics & Image Processing*, Vol. 23, pp. 258-272, 1983.
- [6] R. G. Keys, "Cubic convolution interpolation for digital image processing", *IEEE Trans. Acoust. Speech, & Signal Processing*, Vol. ASSP-29, pp. 1153-1160, Dec. 1981.
- [7] Z. Milivojević, M. Mirković, S. Milivojević, "Procena Fundamentalne Frekvencije Primenom Parametarske kubne Konvolucije sa Grville-ovim jezgrom", *XIII Telekomunikacioni forum TELFOR 2005*.
- [8] S. E. Reicherbach, "Two-Dimensional Cubic Convolution", *IEEE Trans. Image Processing*, Vol. 12, no. 8, pp. 857-865, Aug. 2003.
- [9] Z. Milivojević, D. Brodić, "Estimation Of The Fundamental Frequency Of The Real Speech Signal Compressed By MP3 Algorithm", *Archives of Acoustics*, Vol. 38. No. 3, pp. 363-373, 2013.

UPOTREBA OpenFlow STANDARDA ZA BALANSIRANJE NFS SERVERA SA SISTEMOM POVRATNE SPREGE

USING OpenFlow STANDARD FOR FEEDBACK-BASED NFS SERVER BALANCING

Igor Ivanović, Srđan Kadić, *Prirodno-matematički fakultet Podgorica*

Sadržaj: *OpenFlow je standard u razvoju koji predstavlja SDN (Software Defined Networking) princip upravljanja mrežama. U ovom radu predložena je platforma za evaluaciju metoda i algoritama za balansiranje opterećenja NFS servera koja se zasniva na ovom standardu. U radu se predlaže upotreba emuliranog mrežnog okruženja upotrebom Mininet platforme za evaluaciju algoritama za balansiranje.*

Abstract: *Openflow is an emerging standard which implements the SDN (Software Defined Networking) principle of network management. In this paper a platform for evaluation of algorithms and methods of load-balancing is proposed, which is based on this standard. Mininet emulated network environment is envisaged for evaluation of load balancing algorithms.*

1. UVOD

U distribuiranom fajl sistemu NFS (Network File System) protokol je jedan od najstarijih mehanizama za mrežnu razmjenu fajlova [1]. Iako ima određene nedostatke, posebno po pitanju bezbjednosti, njegova uloga u *data-centrima* je još uvijek veoma velika, jer se često koristi kao alternativa skupim SAN sistemima za skladištenje virtuelnih diskova. Performanse koje se postižu prilikom razmjene fajlova su bliske maksimalnoj praktičnoj brzini mrežnog interfejsa. Pored toga, NFS je stekao reputaciju veoma robustnog i pouzdanog servisa. Za razliku od CIFS (Common Internet File System) protokola koji ima mogućnost da pored fajlova dijeli i uređaje (npr., štampače, skeneri i sl.), NFS je koncentrisan samo na fajlove, a dijeljeni resursi se prikazuju kao direktorijumi u okviru klijentskog sistema.

NFS servis se sastoji od sledećih komponenti:

- NFS demon - servis koji prima zahtjeve od NFS klijenata,
- *Mound* demon - servis koji obezbjeđuje montiranje diskova na udaljenim lokacijama.
- *Rpcbnd* - omogućava NFS klijentima da otkriju koje portove NFS koristi.

U slučaju kada se koristi kao dijeljeno skladište virtuelnih diskova raspoloživost, efikasnost i pouzdanost su od ključnog značaja. Da bi se to postiglo u praksi se koristi nekoliko rešenja. Jedno rešenje je da se poveća kapacitet linkova tako što se koristi LACP agregacija linkova (ili *bonding* u okviru operativnog sistema). Ovim se povećava raspoloživost sistema, jer se omogućava da se više klijenata poveže maksimalnom brzinom mreže. U protivnom, svi klijenti bi dijelili brzinu jednog linka. Uobičajena brzina na današnjim sistemima je 1Gb/s, s obzirom da su veće brzine (10Gb/s i više) još uvijek skupe. Efikasnost podrazumijeva optimalno opsluživanje klijenata, što se postiže pažljivim podešavanjem parametara operativnog sistema. Pouzdanost se postiže na nekoliko načina. Jedan način je da se koristi DRBD replikacioni protokol (poznat i kao mrežni RAID 1) ili neki sličan.

Ono što ostaje kao problem kod prethodnog protokola jeste podešavanje. Kod manjih instalacija broj NFS servera i klijenata, hostova virtuelnih mašina, je mali. Tada se ručno

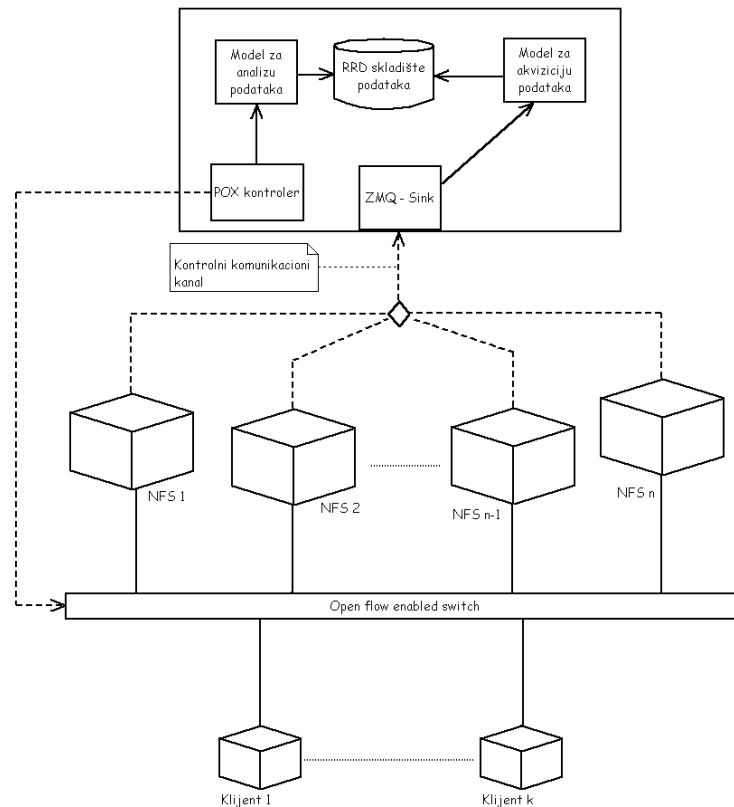
podešavanje isplati i moguće ga je sprovesti relativno jednostavno. U većim sistemima, odnosno sistemima u kojima je tehnička podrška neadekvatna za ovakvu intervenciju, dodavanje novog servera u sistem i samo optimalno podešavanje je otežano.

2. BALANSIRANJE SERVERA

Balansiranje opterećenja servera se najčešće vrši na dva načina: statički i dinamički. Kod statičkog načina predviđa se ponašanje servera pod određenim opterećenjima i u tom slučaju ne postoji povratna sprega koja bi dala informaciju da su parametri odgovarajući. Drugi način koristi povratne podatke, dobijene putem SNMP protokola ili u nekom drugom obliku komunikacije. Dobijeni podaci se obrađuju, a zatim se pomoću nekog od algoritama vrši odlučivanje o rutiranju.

U [2] dat je prikaz implementacije jednog ovakvog sistema koji se koristi u klasterima i zasniva se na servisu za nadgledanje (*monitor daemon*) koji raspoređuje dalja opterećenja. Može se reći da je ovo rešenje *proxy* tipa jer servis za nadgledanje prima zahtjeve, a onda ih prosleđuje na dalju obradu serverima pomoću IPVS (IP Virtual Server) servisa koji radi u kernel zaštićenom režimu rada. Ovaj servis implementira balansiranje opterećenja na transportnom nivou (*layer-4 switching*) i ima mogućnost biranja jednog od 10 algoritama za raspoređivanje:

- Kružno raspoređivanje (*Round-robin scheduling*)
- Težinsko kružno raspoređivanje (*Weighted Round-Robin Scheduling*)
- Raspoređivanje po najmanjem broju veza (*Least connection scheduling*)
- Težinsko raspoređivanje po najmanjem broju veza (*Weighted Least-connection scheduling*)
- Raspoređivanje po najmanjem broju veza bazirano na lokaciji sa replikacijom (*Locality-Based Least-connection with Replication scheduling*)
- Raspoređivanje bazirano na heširanju destinacije (*Destination Hashing Scheduling*)



Slika 1 Prikaz eksperimentalnog okruženja

- Raspoređivanje bazirano na heširanju izvora (*Source Hashing Scheduling*)
- Raspoređivanje zasnovano na najkraćem kašnjenju (*Shortest Expected Delay Scheduling*)
- Raspoređivanje bez smiještanja u red (*Never Queue Scheduling*)

U [3] je dato komercijalno rešenje uz upotrebu mrežnih skretnica koji rade na OSI nivou 7 (tzv., *content switch*). Balansiranje se vrši na osnovu parametara koji su prosljednjeni od strane DFP (*Dynamic Feedback Protocol*) agenata koji nadgledaju jedan ili više servera i prave procjenu opterećenja i kategorisanje definisanjem težine za svaki čvor - server. Podaci se periodično šalju DFP upravljačkom servisu u obliku vektora opterećenja koji na osnovu toga vrši distribuiranje veza na osnovu težina koje su pridružene balansiranim serverima. Algoritmi za raspoređivanje koji se koriste su:

- Kružno raspoređivanje (*Round-robin scheduling*)
- Težinsko kružno raspoređivanje (*Weighted Round-Robin Scheduling*)
- Raspoređivanje po najmanjem broju veza (*Least-connection scheduling*)
- Raspoređivanje bazirano na opterećenju servera i težini (*Arrowpoint Content Aware*)

Dobre osobine prvog rešenja su to što se koristi hardverski nezavisan sistem koji omogućava izbor velikog broja algoritama za raspoređivanje. Zasnovanost na Linux platformi omogućava relativno jeftinu vertikalnu skalabilnost. Sa druge strane, *proxy* model funkcionisanja dovodi do potencijalne tačke zagušenja, kao i tačke otkaza. Drugo rešenje garantuje bolje performanse sistema, jer se sama kontrola predaje mrežnom uređaju, ali se radi o

zatvorenom sistemu koji podrazumijeva, uz veće troškove, vezanost za proizvođača hardvera.

Poznato je da različite vrste servisa imaju različit uticaj na opterećenje hardverskih komponenti. Pored toga, u zavisnosti od samog načina korišćenja servisa obrazac opterećenja se znatno razlikuje. Na primjer, postoji prilična razlika u opterećenju kada se NFS server koristi kao skladište virtuelnih mašina u odnosu na slučaj kada se koristi za opsluživanje korisnika fajlovima. Ako se ovome doda činjenica da, kada se prave klasteri, jedan od zahtjeva je da se dodavanje novih čvorova maksimalno pojednostavi, to mora postojati mehanizam koji će omogućiti automatsku procjenu mogućnosti čvorova i na osnovu toga odrediti sam proces balansiranja.

Jedan od glavnih zahtjeva koji se postavlja pred današnju infrastrukturu je skalabilnost. Stoga većina modernih sistema koristi tehnologije koje su upravo tome prilagođene, jer je količina podataka koja se skladišti u sistemima sve veća. Zbog toga se zahtijeva da se uz minimalna podešavanja vrši dodavanje serverskih čvorova.

3. PREDLOG NOVOG PRISTUPA

U ovom radu se predlaže hibridni princip balansiranja u kome će biti kombinovana dva prethodno razmotrena principa uz upotrebu *OpenFlow* standarda [4]. Sistem će imati formiranu povratnu spregu koja se koristi za transfer mjernih podataka. Kontrolna komponenta će imati u sebi podsistem za smještanje mjernih podataka sa servera, zatim komponentu za analizu podataka i na kraju sam POX *OpenFlow* kontroler koji će vršiti dinamičko podešavanje mrežne skretnice. Ovakvim pristupom se očekuje

kombinovanje fleksibilnosti spoljašnjeg sistema koji može biti implementiran u bilo kojoj tehnologiji sa jednostavnim mrežnim uređajem koji ima visoke performanse.

U [5] je pokazano da je moguće uz pomoć mrežnih skretnica koje imaju implementiran *OpenFlow* standard dobiti funkcionalnosti slične specijalizovanim balanserima opterećenja. Zato se u ovom radu predlaže princip automatske dinamičke konfiguracije sistema na mrežnom sloju, korišćenjem *OpenFlow* standarda za postizanje optimalnih rezultata pri čemu NFS serveri mogu biti heterogeni. Ovaj princip se zasniva na dinamičkom balansiraju servisa na osnovu primljenih podataka o opterećenju sa NFS servera.

4. OPIS PREDLOŽENOG SISTEMA

Na slici 1. prikazan je logički dijagram predloženog sistema. On se sastoji iz više klijentskih čvorova koji koriste neki servis. Servis se opslužuje pomoću više čvorova kako bi se povećala redundantnost, ali u prvom redu raspoloživost servisa. Servisni čvorovi su generalno heterogeni u smislu hardverskih performansi kao i kapaciteta. Najvažniji cilj balansiranja u ovakvim okruženjima jeste da se postigne ravnomjerno i optimalno opterećenje čvorova u cilju optimizacije odziva sistema. Svi komunikacioni kanali su limitirani na gornju granicu prenosa od 1Gb/s.

Svaki čvor posjeduje kontrolni kanal pomoću koga prosljeđuje podatke kontroleru (kontrolni kanali su označeni na slici isprekidanom linijom). Za dobijanje mjernih podataka na Linux baziranim sistemima može se koristiti puno metoda i alata (npr., proc fajl sistem), ali će za potrebe ovih eksperimenata biti korišćen alat nmon [6]. Svaki čvor ima softversku komponentu koja pomoću imenovanog kanala (*named pipe*) preuzima izmjerene podatke iz nmon servisa. Ti podaci se pretvaraju u oblik pogodan za prenos podataka, kako bi se kontroler oslobodio od dodatne obrade podataka, kao i od poznavanja izvornog oblika podataka. Ovdje je definisan JSON (JavaScript Object Notation) format za prenos podataka. Ovakav način organizacije omogućava da se kasnije promijeni mehanizam mjerenja, a da to ne utiče na ostatak sistema. Komunikacija se obavlja pomoću ZeroMQ (Zero Message Queue) komponente [7], koja omogućava veoma brzu komunikaciju između klijenta i servera uz primjenu jednostavnog protokola. Kada kontroler primi podatke, on ih upisuje u RRD bazu podataka (Round Robin Database) [8], koja je specijalizovana za snimanje vremenskih serija podataka. Ovi podaci se koriste prilikom odlučivanja koji čvor će se koristiti prilikom balansiranja opterećenja. Na osnovu ovih podataka je moguće dobiti i dijagrame opterećenja.

Kontroler u sebi sadrži komponentu koja obrađuje dobijene podatke i na osnovu njih vrši odlučivanje. Ova komponenta može da bude predmet daljih istraživanja. S obzirom na relativno nov protokol, nema puno proizvođača *OpenFlow* mrežnih skretnica, u eksperimentima se koristi *openvswitch* softverska skretnica [9] koja se koristi za povezivanje virtuelnih mašina. Za virtuelizaciju su korišćeni kontejnerski čvorovi koji su sastavni dio Mininet sistema, kao i KVM (Kernel Virtual Machine) virtuelizator.

5. OPIS KORIŠĆENIH TEHNOLOGIJA

U ovoj sekciji je dat kratak opis tehnologija koje se koriste u implementaciji predloženog sistema.

5.1 OpenFlow protokol

OpenFlow je implementacija relativno nove ideje da se mreža posmatra kao servisno orijentisan sistem koji dozvoljava dinamičku kontrolu mrežnih uređaja pomoću softverskih komponenti - SDN (*Software Defined Networking*). Osnovnu ulogu u ovom sklopu ima kontroler koji komunicira sa najmanje jednim mrežnim uređajem. Ovdje se, naravno, podrazumijeva da uređaj može biti stvarni ili virtuelni. *OpenFlow* standard definiše API koji se koristi za komunikaciju sa uređajem. Kontroler upravlja sa *flow* tabelom mrežnog uređaja omogućavajući promjenu načina na koji se uređaj ponaša pod prethodno određenim uslovima.

5.2 Mininet

Mininet je mrežni emulator koji uz pomoć kontejnerske virtuelizacije i korišćenjem virtuelnih mrežnih uređaja omogućava stvaranje okruženja u kojem je moguće vršiti mrežne eksperimente, bez potrebe za korišćenjem stvarne mrežne opreme. Ovakav pristup znatno umanjuje finansijske troškove potrebne za testiranje mrežnih topologija, a posebno kada je u pitanju *OpenFlow* protokol, s obzirom da su ovi uređaji još uvijek relativno slabo dostupni. S obzirom da je mreža virtuelno definisana u potpunosti, moguće je eksperimentisati sa protokolima, mrežnim topologijama i servisima. Cio sistem je napravljen u Python okruženju, tako da ga je moguće veoma jednostavno integrisati u aplikacije i dinamički upravljati topologijama i podešavanjima. Poslije završenih eksperimenata virtuelna mreža se jednostavno realizuje fizički.

5.3 RRD mehanizam skladištenja mjernih podataka

Već je pomenuto da se za potrebe skladištenja mjernih podataka koristi RRD softverska komponenta. RRD je nastao kao unaprijeđena verzija MRTG skladišta za smještanje mjernih podataka sa opreme, dobijenih putem SNMP protokola. Jedna od interesantnih karakteristika ovog alata jeste da se direktno može vršiti iscrtavanje grafika. RRD kružni bafer radi isključivo sa vrijednostima koje su neka vrsta stope (*rate*). To znači da vrijednosti koje se smještaju predstavljaju odnose veličina, u ovom slučaju mjerena veličina u odnosu na interval vremena. Na primjer mjeri se opterećenje u odnosu na sekundu (broj prenesenih bajtova u sekundi ili broj prenesenih paketa u interval vremena). Većina ovih veličina u odgovarajućem domenu predstavljaju brzinu izraženu odgovarajućim jedinicama. U slučaju mrežnih podataka obično se izražava u B/s i packet/s. RRD bafer je sastavljen od sledećih djelova:

- RRD (*Round Robin Database*) cirkularne baze podataka koja je fizički predstavljena fajlom na fajl sistemu
- RRA (*Round Robin Archive*) arhiva cirkularne baze koja se sastoji od slotova za snimanje podataka.
- DS (*Data Source*) predstavlja promjenljivu čija se vrijednost snima. Prilikom kreiranja mora se definisati tip

(da li se radi o vrijednosti koja je već neka stopa ili se radi o brojačima).

- CF (*Composite Function*) je kompozitna funkcija koja vrši agregaciju više vrijednosti u jednu.

6. PREDLOG BUDUĆEG ISTRAŽIVANJA

Predložena platforma će u daljem periodu biti implementirana u potpunosti. Potom će se na platformi biti izvršen niz eksperimenata kako bi se potvrdilo da je zaista moguće pomoću ove konfiguracije zaista izvršiti efikasno balansiranje. Ovo rešenje će biti upoređeno sa dostupnim praktičnim rešenjima kako bi se evaluirao domet i efikasnost *OpenFlow* sistema, kao i njegove prednosti i mane u odnosu na klasična rešenja.

Jedan otvoreni problem koji bi mogao da se pojavi je konzistentnost upisa podataka u trenutku prelaza sa jednog virtuelnog skladišta na drugo. Drugi mogući problem su ograničene performance kontrolera i njegovog komunikacionog kanala. Kao što je rečeno, za komunikaciju se već koristi veoma brz ZMQ protokol, ali sa druge strane je potrebno utvrditi granicu do koje se može ići sa opterećenjem jednog kontrolera.

Kao što je poznato, postoji odstupanje u rezultatima eksperimenata kada se mjerni podaci dobijaju iz sintetičkih opterećenja u odnosu na realna opterećenja. Zbog toga će se za potrebe testiranja prvo izvršiti snimanje opterećenja metodom prisluškivanja komunikacije klijenata i servera u toku dužeg intervala vremena. Zatim će tako dobijeni rezultati biti prevedeni u odgovarajući oblik, a potom zahtjevi ponovljeni u eksperimentalnom okruženju. Posebno interesantni podaci se očekuju od mjerenja opterećenja NFS dijeljenog skladišta virtuelnih mašina u sistemu bez povratne sprege.

7. ZAKLJUČAK

U radu je dat kratak uvod u NFS protokol i njegove implementacije. Napomenut je značaj ovog protokola u modernim sistemima i pored njegove relativno duge prisutnosti. Potom je definisana potreba za balansiranjem u velikim sistemima i prikazani su uobičajeni generalni algoritmi koji se koriste u postojećim sistemima. Novi pristup problemu balansiranja bazira se na primjeni algoritama nad

tzv. ravni podataka unutar mrežne skretnice koja podržava *OpenFlow* sadržaj. Da bi se to moglo uraditi, predložena je posebna platforma nad koja će poslužiti za eksperimentisanje. Ovaj sistem posjeduje i mogućnost skladištenja podataka, kako bi algoritmi za balansiranje mogli da koriste ne samo tekuće, već i ranije zapamćene podatke za podršku algoritmima. Na kraju, prikazani su pravci daljih istraživanja korišćenjem predložene platforme.

LITERATURA

- [1] S. Shepler, M. Eisler, D. Noveck, "Network File System (NFS) Version 4 Minor Version 1 Protocol (RFC)", *Internet Engineering Task Force (IETF)*, 2010
- [2] Dynamic Feedback Load Balancing Scheduling http://kb.linuxvirtualserver.org/wiki/Dynamic_Feedback_Load_Balancing_Scheduling
- [3] Configuring Dynamic Feedback Protocol for Server Load Balancing http://www.cisco.com/en/US/docs/app_ntwk_services/data_center_app_services/css11500series/v8.20/configuration/content_lb/guide/DFP.pdf
- [4] Open networking foundation <https://www.opennetworking.org/>
- [5] H. Uppal, D. Brandon "OpenFlow Based Load Balancing" University of Washington CSE561: Networking Project Report
- [6] Nigel's performance Monitor for Linux <http://nmon.sourceforge.net/pmwiki.php>
- [7] P. Hintjens "Code connected Volume 1, Learning ZeroMQ, Professional edition for C/C++", *iMatrix Corporation*, 2013.
- [8] RRDtool logging and graphing <http://oss.oetiker.ch/rrdtool/>
- [9] Open vSwitch an open virtual switch <http://openvswitch.org/>

VIRTUELIZACIJA MREŽE KORIŠĆENJEM ALATA FLOWVISOR NETWORK VIRTUALIZATION WITH FLOWVISOR TOOL

Miladin Tomić, Milutin Radonjić, Nedeljko Lekić, Igor Radusinović, *Elektrotehnički fakultet Podgorica*

Sadržaj: *Mrežna virtuelizacija omogućava simultano korišćenje mrežnih resursa, kroz njihovo dijeljenje i kreiranje individualnih mreža koje funkcionišu nezavisno jedna od druge. U ovom radu je opisan FlowVisor, koji predstavlja jedan od alata za mrežnu virtuelizaciju. FlowVisor je kontroler specijalne namjene koji omogućava virtuelizaciju u OpenFlow mrežama, dijeleći fizičku mrežu na više logičkih mreža. On osigurava da svaka mreža može koristiti samo svoje resurse. U radu je dat uvid u način funkcionisanja FlowVisor-a, osnovne zadatke koje mora da ispuni, princip rada i način na koji obavlja virtuelizaciju mrežnih resursa.*

Abstract: *Network virtualization allows network resources to be shared, by dividing them and creating individual networks, which can operate independently of each other. This paper describes FlowVisor, which is a tool for network virtualization. FlowVisor is a special purpose controller, which enables virtualization in OpenFlow networks, by dividing a physical network into multiple logical networks. It ensures that each virtual network uses only its own resources. Paper provides FlowVisor description, main goals which need to accomplish, work principles and explains how it provides virtualization of network resources.*

1. UVOD

Iako je mrežna virtuelizacija dugo vremena interesantna istraživačkoj zajednici, tek je SDN (*Software Defined Networks*) koncept, kroz razdvajanje kontrolne ravni i ravni podataka, doveo do značajnijeg pomaka na tom polju [1]. Tradicionalna segmentacija LAN (*Local Area Networks*) mreža, podjelom jednog na više nezavisnih *broadcast* domena (*Virtual Local Area Networks*) predstavlja primitivnu podjelu saobraćaja na nivou linka po osnovu interfejsa *switch*-a. Pored toga što ne nudi fleksibilnost u pogledu podjele saobraćaja, ona ne nudi ni mogućnost virtuelizacije ostalih resursa koji čine jednu mrežu. Potpuna virtuelizacija mreža je prije pojave SDN koncepta mogla biti postignuta korišćenjem specijalizovanog hardvera, sa najčešće dupliranim komponentama [2]. To je bilo skupo i nepraktično rješenje. Veći pomak na virtuelizaciji komercijalnih mreža nije mogao biti ostvaren, prije svega zbog činjenice da se proizvođači opreme trude da očuvaju *black box* koncept svojih uređaja. *OpenFlow* kao ključna tehnologija za razvoj SDN koncepta rješava ovaj problem [3]. *OpenFlow* kroz razdvajanje kontrolne ravni od ravni podataka i apstrahovanjem ravni podataka setom odgovarajućih parametara, relativizuje limitirajući uticaj *black box* koncepta na mrežnu virtuelizaciju. *OpenFlow* je zamišljen kao tehnologija koja bi trebalo da dovede do ubrzanog razvoja novih mrežnih kontrolnih rješenja. Jedan od najvećih problema pri razvoju mrežne kontrole je njena evaluacija, kroz proces testiranja. Kako se kontrolna rješenja uglavnom kreiraju za komercijalne mreže, poželjno je razvoj i testiranje sprovoditi na njima. Izgradnja posebnih mreža za tu namjenu je neisplativa, a korišćenje komercijalnih mreža za potrebe testiranja sa stanovišta vlasnika mreža je neprihvatljivo. *OpenFlow*, uz pomoć *FlowVisor*-a, virtuelizacijom rješava ovaj problem. *FlowVisor* je *OpenFlow* kontroler specijalne namjene, koji omogućava kreiranje više virtuelnih mreža na istoj fizičkoj infrastrukturi. *FlowVisor* osigurava da svaka od virtuelnih mreža može da koristi i utiče samo na resurse koji

su joj dodijeljeni. Na taj način, paralelno sa komercijalnom mrežom, mogu postojati mreže koje se koriste u svrhu istraživanja, dijeleći istu fizičku infrastrukturu.

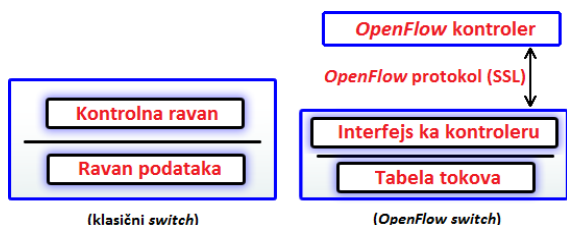
Fleksibilnost *OpenFlow*-a u pogledu kreiranja novih mrežnih kontrolnih rješenja je uspješno demonstrirana u ranijem istraživanju [4]. Imajući u vidu velike mogućnosti koje mrežna virtuelizacija može da pruži, nastavak istraživačkog rada na temu *OpenFlow*-a je bio usmjeren u tom pravcu. Kreirana je *OpenFlow* mreža u kojoj je implemetiran *FlowVisor* i sprovedena virtuelizacija, a time omogućeno koegzistiranje dvije nezavisne mreže. Ovaj rad će pružiti uvid u način na koji *FlowVisor* omogućava virtuelizaciju mreže.

Rad ima 7 poglavlja. U poglavlju 2 je dat kratak opis *OpenFlow* tehnologije a posle toga je pružen uvid u način funkcionisanja *FlowVisor*-a, kroz prikazanu paralelu sa virtuelizacijom u svijetu računara. U poglavlju 4 su dati glavni zahtjevi koje je trebalo ispuniti pri njegovom dizajniranju kako bi mogao naći odgovarajuće mjesto u *OpenFlow* mrežama. Poglavlje 5 opisuje arhitekturu *FlowVisor*-a, dok poglavlje 6 opisuje mehanizme dijeljenja i izolacije mrežnih resursa. Na kraju su data zaključna razmatranja.

2. OPENFLOW TEHNOLOGIJA

Glavna promjena koju *OpenFlow* donosi u odnosu na klasičnu arhitekturu mrežnih uređaja je razdvajanje kontrolne ravni od ravni podataka. Time je omogućeno eksterno kontrolisanje ravni podataka, preko sigurnog kanala, kako je to prikazano na slici 1. Kako se sa slike može uočiti, *OpenFlow switch* ima tri osnovna elementa: tabelu tokova, kontroler i kanal za komunikaciju između njih. Tabela tokova predstavlja generalizaciju tabele prosljeđivanja *Ethernet switch*-a, tj. apstrakciju njegove ravni podataka. Strukturno gledano, ona je set zapisa koji se koriste da bi se utvrdio

način postupanja sa saobraćajem. Kontroler predstavlja kontrolnu ravan *switch*-a. To može biti personalni računar, server, ili neka druga vrsta računarskog sistema na kojem se izvršava kontrolni proces. Ovaj proces definiše ponašanje *switch*-a manipulišući zapisima u tabeli tokova. Sigurni kanal se koristi za komunikaciju između kontrolera i tabele tokova. Ta komunikacija počiva na *OpenFlow* protokolu, koji definiše format poruka koje se razmjenjuju između tabele tokova i kontrolnog procesa, kao i akcije koje slijede [5].

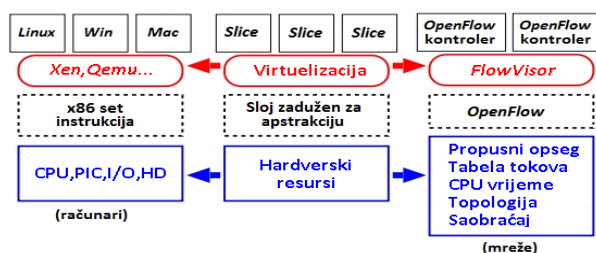


Slika 1. Arhitektura klasičnog *switch*-a i *OpenFlow switch*-a.

Dakle, umjesto integrisanih ravni podataka i kontrolne ravni, *OpenFlow* razdvaja te dvije ravni. Ravni podataka *switch*-eva su apstrahovane tabelama tokova. Pomoću sigurnog kanala i kontrolera omogućeno je centralizovano manipulisanje tabelama tokova, a time i saobraćajem. Na taj način je *OpenFlow* donio daleko veću fleksibilnost u pogledu kontrole u odnosu na klasične uređaje i omogućio virtuelizaciju mrežnih resursa.

3. PRINCIP RADA FLOWVISOR-A

Da bi se bolje razumjeli principi na kojima *FlowVisor* počiva, potrebno je napraviti uvid u principe virtuelizacije u svijetu računara. Kod računara, set instrukcija omogućava apstrakciju hardverskih resursa. Povrh toga, neki alati za virtuelizaciju (npr. *Xen Qemu* [6]) izvode dijeljenje apstrahovanih fizičkih resursa (memorije, procesora i sl.). Na tim resursima mogu potom biti paralelno pokrenuti različiti operativni sistemi, kao tzv. virtuelne mašine. Oni funkcionišu nezavisno, koristeći samo resurse koji su im dodijeljeni. Na ovaj način, različiti operativni sistemi mogu biti istovremeno podržani na istoj fizičkoj infrastrukturi, kao što je prikazano na slici 2. Slično tome, *FlowVisor* kao alat za mrežnu virtuelizaciju zahtijeva neku vrstu hardverske apstrakcije. Apstrakcija bi trebala da bude takva da je lako napraviti virtuelne mreže i da bude dovoljno generalna kako bi mogla obuhvatiti različite vrste uređaja. Uzimajući u obzir da *OpenFlow* ispunjava ove zahtjeve, on je odabran kao alat za hardversku apstrakciju.



Slika 2. Poređenje virtuelizacije u svijetu računara i *OpenFlow* mreža.

Imajući *OpenFlow* kao alat za apstrakciju, *FlowVisor* se smješta između *OpenFlow switch*-eva, apstrahovanih

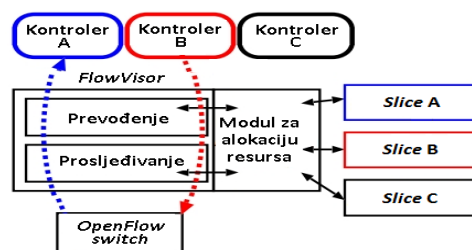
tabelama tokova i kontrolera. Na taj način se dijeli i izoluje apstrahovani hardver, kako bi ponudio izolovane djelove infrastrukture kontrolerima iznad sebe. Mrežna virtuelizacija zahtijeva apstrahovanje sljedećih resursa: propusnog opsega, tabele tokova, CPU-a uređaja, topologije i saobraćaja. Virtuelizacija ovih pet resursa znači da svaka logička mreža koju obezbjeđuje *FlowVisor*, treba da ima na raspolaganju određenu količinu svakog od njih. Virtuelizacija, odnosno dijeljenje mreže po ovih pet resursa se naziva *slice*-ovanje, a jedan *slice* predstavlja definisanu količinu svakog od njih.

4. ZAHTJEVI KOJE FLOWVISOR TREBA DA ISPUNI

Pri dizajniranju *FlowVisor*-a, vođeno je računa da on treba da pruži mogućnost fleksibilne definicije virtuelne mreže, transparentnost prema kontrolerima i *switch*-evima kao i snažnu izolaciju između virtuelizovanih resursa. Kako je glavni motiv mrežne virtuelizacije stvaranje okruženja za testiranje novih ideja, veoma je bitno da politika alokacije resursa, odnosno definisanja *slice*-ova bude što je moguće više fleksibilna. Doza fleksibilnosti koja stoji na raspolaganju u tom pogledu, determiniše mogućnost kreiranja virtuelnih okruženja, a time i opseg ideja koje na njima mogu biti testirane. Slično prvom zahtjevu, zahtjev za transparentnošću je nametnut potrebom da se osigura što dinamičniji razvoj novih kontrolnih rješenja. Ukoliko ovaj zahtjev ne bi bio ispunjen, i mrežni hardver i kontrolna rješenja bi morala biti prilagođena činjenici da se između njih nalazi virtuelizujući sloj. Na ovaj način nove ideje ne bi mogle biti razvijane i evaluirane u realnim uslovima, već bi se moralo voditi računa o virtuelizaciji. Prema tome, veoma je bitno da virtuelizujući sloj bude transparentan i prema kontrolnoj ravni i prema ravni podataka, kako bi se pojednostavio i ubrzao razvoj novih rješenja za menadžment mreže. Posljednji zahtjev je izolacija, jer postojanje više virtuelnih mreža samo po sebi nije veliko dostignuće, ako nisu međusobno izolovane. Samo u tom slučaju, one mogu biti nezavisne, i samo tada je moguće govoriti o više mreža koje koegzistiraju na istoj fizičkoj infrastrukturi [7].

5. ARHITEKTURA FLOWVISOR-A

Kao što je prikazano na slici 3, *FlowVisor* ima tri funkcionalna entiteta: modul za alokaciju resursa, modul za prevođenje i modul za prosljeđivanje.



Slika 3. Arhitektura *FlowVisor*-a i njegova lokacija u *OpenFlow* mreži

Pošto se nalazi iznad *OpenFlow switch*-eva, *FlowVisor* nadzire *OpenFlow* tabele, kojima su fizički *switch*-evi apstrahovani, kao i tokove kojima je apstrahovan saobraćaj. Ova dva resursa zajedno sa propusnim opsegom, topologijom i CPU-om *switch*-eva su *slice*-ovana, odnosno podijeljena i

stavljena na raspolaganje različitim kontrolerima. Na osnovu kreiranih *slice*-ova *FlowVisor* virtualizuje mrežne resurse, odnosno kreira virtuelne mreže [8].

Kada poruka od kontrolera dođe do *FlowVisor*-a, on zna kojem *slice*-u taj kontroler pripada (*slice*-ovi i kontroleri imaju odnos 1:1). Ukoliko kontroleru nije dozvoljeno da pristupi datom *slice*-u, odnosno ako akcije *OpenFlow* poruke nisu dozvoljene jer utiču na resurse koji *slice* politikom nijesu dodijeljeni tom kontroleru, *FlowVisor* prepravljajući poruku da bude u skladu sa *slice* politikom kontrolera. Na primjer, ako *slice* politika kontrolera definiše da on ima pravo na određen broj portova *switch*-a, *OpenFlow* poruka kojom bi kontroler želio da napravi zapis koji vrši *broadcast* bi bila presretnuta od strane *FlowVisor*-a i modifikovana tako da umjesto *broadcast*-a to bude *multicast*, na interfejsu na koje po *slice* politici ima pravo. Na taj način, kontroler može uticati samo na resurse koji su mu dodijeljeni. Ovo prepravljavanje se obavlja od strane modula za prevođenje. Slično tome, kada *FlowVisor* primi *OpenFlow* poruku od *switch*-a, on provjerava sve politike kako bi utvrdio kojem *slice*-u poruka pripada. Poslije provjere, poruka se isporučuje kontroleru čija *slice* politika odgovara opisu poruke. Ovo se obavlja uz pomoć modula za prosljeđivanje.

Na ovaj način, koristeći sve tri funkcionalne jedinice, *FlowVisor* osigurava da svaki kontroler dobija i kontroliše samo saobraćaj koji pripada njegovom *slice*-u, odnosno osigurava izolaciju *slice*-ova, tj. virtuelnih mreža. *FlowVisor* se ponaša kao transparentni *OpenFlow* proksi, koji komunicira korišćenjem *OpenFlow* protokola sa *OpenFlow switch*-evima i kontrolerima. Presrijetanjem poruka od obje strane, i prepravljajući, odnosno prosljeđujući ih, moguće je obezbijediti koegzistenciju više izolovanih mreža na istoj fizičkoj osnovi.

6. MEHANIZMI ZA VIRTUELIZACIJU RESURSA

Kako *FlowVisor* virtualizuje pet resursa, postupak virtualizacije može biti podijeljen na pet zadataka, virtualizaciju saobraćaja, topologije, CPU-a u *switch*-u, tabele tokova i propusnog opsega.

Kako je saobraćaj predstavljen tokovima, ukupan saobraćaj je podijeljen dodjeljivanjem različitih grupa tokova različitim *slice*-ovima. Pojam tok, u *OpenFlow* mreži označava pakete koje karakteriše ista vrijednost polja u zaglavlju paketa, koja su od interesa. *FlowSpace*, sa druge strane, predstavlja grupu tokova koji pripadaju jednom *slice*-u, odnosno virtuelnoj mreži. *FlowSpace*-ovi se definišu kao set zapisa. Svaki zapis se sastoji od pravila i odgovarajućih akcija. Pravila definišu tokove, dok akcije definišu šta bi trebalo biti urađeno sa tim tokovima. Moguće akcije su: *allow*, *deny* i *read-only*. Akcija *allow* daje svu kontrolu nad definisanim tokom, *deny* znači njegovo odbijanje, a *read-only* njegovo primanje, ali ne i mogućnost kontrole nad njim.

Da bi se dobila jasna slika o akcijama koje sprovodi *FlowVisor* u postupku dijeljenja saobraćaja i njegove izolacije, biće razmotren jedan od realizovanih primjera. *OpenFlow* mreža je virtualizovana i kreirane su dvije logičke, odnosno virtuelne mreže, kako je to prikazano na slici 4.

Jednoj virtuelnoj mreži je dodijeljen HTTP saobraćaj koji potiče sa jedne IP adrese, dok druga mreža kontroliše ostali saobraćaj. Dijeljenje saobraćaja se obavlja definisanjem *FlowSpace*-a prvog *slice*-a na sljedeći način:

Allow : tcp_port = 80 ip = adresa_1;

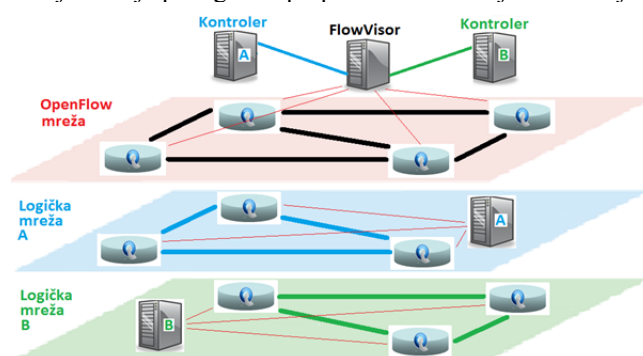
Na ovaj način, kontroler virtuelne mreže čiji *FlowSpace* posjeduje ovaj zapis, ima potpunu kontrolu nad HTTP saobraćajem sa definisane adrese. Pošto kontroler nije svjestan postojanja *FlowSpace*-ova, on misli da kontroliše sav saobraćaj. Kontroler bi mogao pokušati da sprovede akcije koje utiču na tokove koji nisu u definisanom *FlowSpace*-u. Takva poruka će biti presretnuta od strane *FlowVisor*-a. U ovom konkretnom slučaju *FlowVisor* će osigurati da svaka poruka od posmatranog kontrolera može da utiče samo na HTTP saobraćaj koji se tiče IP adrese definisane njegovim *FlowSpace*-om. Pokušaj kontrolera da utiče na neki drugi protokol ili druge adrese, rezultuje njihovim blokiranjem, tj. odbacivanjem takvih *OpenFlow* poruka. Pored toga što *FlowVisor* vodi računa o porukama koje stižu sa kontrolera, on presrijeće i poruke koje generišu *switch*-evi. Tako se osigurava da će sve poruke koje generiše *switch*, a izazvane su HTTP saobraćajem sa definisane IP adrese, završiti na odgovarajućem kontroleru.

Sa druge strane, da bi se osiguralo da saobraćaj koji pripada prvom, ne završi u drugoj mreži, potrebno je definisati *FlowSpace slice*-a druge mreže na sljedeći način:

Deny : tcp_port = 80 ip = adresa_1; Allow : all;

Ovo će rezultirati odbijanjem HTTP saobraćaja sa odgovarajuće adrese, i prihvatanjem ostalog saobraćaja.

Može se definisati i treći *slice*, odnosno virtuelna mreža koja se može koristiti za potrebe monitoringa. *FlowSpace* te mreže bi trebao biti, *Read-Only : all*. To znači da mreži za monitoring može pripadati sav saobraćaj, ali da nad njim ne može biti sprovedena nijedna kontrolna akcija. Na ovaj način *FlowVisor* dijeli saobraćaj, dok presretanje i prepravljavanje, odnosno odbacivanje poruka obezbjeđuje njegovu izolaciju. Na taj način je postignuta potpuna virtualizacija saobraćaja.



Slika 4. Virtuelizovana *OpenFlow* mreža

Postupak virtualizacije topologije može biti podijeljen na dva dijela. Prvi se sastoji od otkrivanja mrežnih uređaja, a drugi od otkrivanja linkova koji ih povezuju. *Switch*-evi bivaju otkriveni kada se povežu na kontroler. Imajući u vidu da se *FlowVisor* nalazi između *switch*-eva i kontrolera, on može spriječiti ili omogućiti uspostavljanje veze sa

kontrolerima u zavisnosti od toga da li kontroleri imaju pravo na taj *switch*. Otkrivanje linkova, se sprovodi korišćenjem LLDP paketa. Za ove pakete se ne pravi zapis u tabeli tokova. Kontroler šalje pakete na *switch* zahtijevajući da oni budu proslijeđeni kroz svaki od interfejsa. Kada se to desi, i paket sa interfejsa preko linka stigne do direktno povezanog *switch*-a, on zbog nepostojanja zapisa u tabeli tokova biva proslijeđen kontroleru. Kontroler onda može da vidi sa kog *switch*-a i sa kog interfejsa je taj paket stigao. Na ovaj način se vrši otkrivanje linkova koji povezuju *switch*-eve. Ukoliko je jedan *switch* dijeljen od strane više kontrolera, svaki od njih ima pravo samo na interfejse koji su mu dodijeljeni. To se postiže kroz sprečavanje kontrolera da šalje svoje LLDP pakete kroz interfejse na koje nemaju pravo. Na ovaj način se vrši virtualizacija mrežne topologije.

Pored ova dva resursa, potrebno je virtualizovati i procesor *switch*-a, kako bi se spriječio njegovo preopterećenje od strane jedne virtualne mreže. Da do ovakve situacije ne bi dolazilo, potrebno je spriječiti da jedan *slice* ima monopol nad CPU-om *switch*-a. Glavni izvor opterećenja procesora su poruke koje kontroler i *switch* međusobno razmjenjuju. Ukoliko je brzina njihovog pristizanja velika, može doći do njegovog preopterećenja. Iz tog razloga se za svaki *slice* vodi računa o brzini razmjenjivanja ovih poruka. Kada njihova brzina dosegne maksimalnu vrijednost, poruke će biti odbacivane izvjesno vrijeme. Dakle, virtualizacija procesora se sprovodi limitiranjem brzine slanja poruka između kontrolera i *switch*-a.

Slično virtualizaciji CPU-a *switch*-a, virtualizacija tabele tokova se obavlja limitiranjem broja zapisa koje svaki *slice* može imati u njoj, kroz postavljanje brojača za svaki *slice*. Kada broj zapisa *slice*-a dosegne maksimalnu vrijednost, dalje unošenje biva onemogućeno.

Virtualizacija propusnog opsega je bazirana na mapiranju tokova različitih *slice*-ova u različite QoS klase. Ove klase su definisane korišćenjem trobitnog *Priority Code Point* polja VLAN tag-a. Polje definiše osam različitih klasa saobraćaja. Karakteristike ovih klasa nisu određene *OpenFlow* specifikacijom, već je njihovo definisanje ostavljeno mrežnim administratorima, a u sklopu mogućnosti koje pruža *OpenFlow switch*. Sam mehanizam obezbjeđenja QoS-a varira u zavisnosti od verzija *OpenFlow switch*-eva i konstantno evoluiru [9].

7. ZAKLJUČAK

Jedan od glavnih motiva za razvoj *OpenFlow* tehnologije, a samim tim i *FlowVisor*-a, jeste pravljenje otklona od trenutno krute i u smislu kreiranja novih kontrolnih rješenja, nefleksibilne mrežne arhitekture. *OpenFlow* omogućava pojednostavljeni razvoj, a prije svega evaluaciju novih mrežnih menadžment rješenja, tako što obezbjeđuje postojanje eksperimentalnih, testnih mreža paralelno sa komercijalnim mrežama. Ovo je neophodno, kako bi se nova mrežna menadžment rješenja razvijala i testirala u okruženju koje je identično komercijalnom. Njihovo koegzistiranje je omogućeno virtualizovanjem *OpenFlow* mreže, korišćenjem *FlowVisor* na način opisan u radu. Time se pružaju garancije da više mreža može paralelno koegzistirati na istoj fizičkoj infrastrukturi, međusobno se ne

ugrožavajući. To bi, u konačnom, sa većim probojem *OpenFlow* tehnologije na polju komercijalnih mreža, trebalo da dovede do stvaranja jednog realnog okruženja za testiranje novih ideja, a samim tim i naglog porasta broja novih mrežnih menadžment rješenja. Imajući u vidu velike mogućnosti koje pruža mrežna virtualizacija, i slijedeći trendove u oblasti SDN-a, uspješno smo sproveli virtualizaciju *OpenFlow* mreže. Za sada je demonstrirana mogućnost virtualizacije saobraćaja i topologije. Zbog malog saobraćajnog opterećenja realizovanih virtualnih mreža, i u odnosu na tako malo opterećenje, prilično velikih hardverskih resursa virtualizovane *OpenFlow* mreže, nije demonstrirana mogućnost virtualizacije preostala tri resursa. Dalji rad će biti usmjeren na omogućavanje i demonstriranje mogućnosti *FlowVisor*-a da virtualizuje preostala tri resursa, od kojih je najinteresantniji i najizazovniji virtualizacija propusnog opsega. Na ovaj način će biti testirana mogućnost *FlowVisor*-a da osigura virtualizaciju i u slučajevima saobraćajnog opterećenja koje prevazilazi dodijeljene mrežne resurse.

ZAHVALNICA

Rad je dijelom finansiran sredstvima projekta „Razvoj *Future Internet* rutera na programabilnim hardverskim platformama otvorenog koda“, sufinansiranim od strane Ministarstva za nauku Crne Gore i Ministarstva za informaciono društvo i telekomunikacije.

LITERATURA

- [1] Open Networking Foundation, “Software-defined networking: The new norm for networks,” ONF White Paper, April 2012.
- [2] J. S. Turner, P. Crowley, J. DeHart, et al., “Supercharging planetlab: a high performance, multi-application, overlay network platform.” ACM SIGCOMM '07 New York, NY, USA, 2007.
- [3] N. McKeown, et al., “OpenFlow: enabling innovation in campus networks”, ACM SIGCOMM, Computer Communication Review, Vol. 38, Issue: 2, Mart 2008.
- [4] Igor Radusinović, Miloš Odalović, “Improving and Simplifying Control in OpenFlow Networks”, Proc. of 21th Telecommunication Forum TELFOR 2013, pp. 86-89, Belgrade, Serbia, Novembar 2013.
- [5] OpenFlow Switch Specification v1.0. Web site: <http://www.openflow.org>
- [6] Xen, Web site: <http://www.xen.org>
- [7] R. Sherwood et al., “Carving Research Slices Out of Your Production Networks with OpenFlow,” ACM SIGCOMM Computer Communication Review, vol. 40, pp. 129–130, Januar 2010.
- [8] R. Sherwood et al., “Flowvisor: A Network Virtualization Layer,” tech. rep., OpenFlow Switch Consortium, Oktobar 2009.
- [9] Balázs Sonkoly, et al., “OpenFlow Virtualization Framework with Advanced Capabilities,” EWSDN, 2012.

IMPLEMENTACIJA RIP I OSPF PROTOKOLA NA QUAGGA SOFTVERSKOJ PLATFORMI IMPLEMENTATION OF RIP AND OSPF PROTOCOLS ON QUAGGA PLATFORM

Slavica Tomović, Milutin Radonjić, Igor Radusinović, *Elektrotehnički fakultet Podgorica*

Sadržaj: *Quagga je softverska platforma otvorenog koda koja pruža TCP/IP servise rutiranja. Podržava sledeće protokole rutiranja: RIPv1, RIPv2, RIPng, OSPFv2, OSPFv3, IS-IS, BGP4 i BGP4+. Quagga je jedinstvena u svom dizajnu jer koristi poseban proces za izvršavanje svakog od navedenih protokola. Programski kod ima modularnu strukturu koja omogućava da se svaki od njenih modula modifikuje i nadograđuje nezavisno od drugih. Trenutno Quaggu podržavaju Linux, FreeBSD, NetBSD, OpenBSD i Solaris sistemi. U ovom radu izvršeno je testiranje podrške ove platforme za RIP i OSPF protokole, pri čemu je dodatno analizirana brzina njihove konvergencije.*

Abstract: *Quagga is an open-source software platform that provides routing services. It supports the following IP routing protocols: RIPv1, RIPv2, RIPng, OSPFv2, OSPFv3, IS-IS, BGP4 and BGP4+. Quagga is unique in its design because it uses a separate process for each running protocol. This software offers true modularity since each protocol module can be upgraded or configured independently of the others. Currently, Quagga is supported by Linux, FreeBSD, NetBSD, OpenBSD and Solaris. In this paper, support of this platform for RIP and OSPF is investigated. In addition, the results of convergence time measurements are presented.*

1. UVOD

Velika popularnost informacionih i komunikacionih tehnologija i ekspanzija širokopojasnih aplikacija i servisa nameću sve strožije zahtjeve u pogledu performansi mrežne opreme i protokola rutiranja na Internetu. Zahvaljujući tehnološkom napretku na polju mikroelektronike, ruteri implementirani u hardveru nude sve veće brzine komutacije. Najčešće korišćena oprema za rutiranje u današnjim mrežama je vlasništvo velikih kompanija (Cisco, Juniper, ...). Ove rutere odlikuje izuzetno sofisticiran hardver koji u najvećem broju slučajeva pruža sve potrebne funkcionalnosti i zadovoljava potrebe i najvećih mreža. Međutim, tržište softvera za rutiranje je zatvoreno u smislu da ovi ruteri rade samo sa softverom njihovog proizvođača tako da je testiranje novih protokola u realnim mrežnim uslovima praktično nemoguće. Stoga su provajderi servisa često primorani da čekaju proizvođače opreme da razmotre implementaciju novih funkcionalnosti u svojim sistemima, što zahtijeva dosta vremena i novca. Ovaj problem je moguće prevazići upotrebom softverskih platformi otvorenog koda (tzv. *open-source*) za rutiranje. Ove platforme podrazumijevaju da ruteri koriste programski kod koji je besplatno dostupan i kao takav može biti modifikovan na bilo koji način, kako bi ispunio zahtjeve korisnika. Instalacijom softvera za rutiranje, običan računar sa većim brojem mrežnih kartica postaje tzv. *softverski ruter*, i može vršiti usmjeravanje paketa u mreži. Iako se po performansama, u principu, ne mogu mjeriti sa hardverskim rješenjima rutera, softverske alternative su u poslednje vrijeme zbog svoje cijene koja je i do 1000 puta niža postale veoma popularne. Pored toga, softverske platforme značajno pojednostavljaju postupak razvoja nekog projekta, jer se dogradnja, korekcija i testiranje vrše neuporedivno lakše i brže. Softver za rutiranje je tipično baziran na Linux-u, mada u poslednje vrijeme počinje razvoj *open-source* alata koji mogu biti implementirani i na drugim operativnim sistemima, kao što je Windows.

Open-source rutiranje još uvijek nije implementirano u nekim većim mrežama, i obično se posmatra sa dozom skepticizma od strane ljudi koji nisu familijarni sa ovom tehnologijom, tako da se *open-source* platforme danas najčešće koristi u istraživačke i obrazovne svrhe. Najveći broj kritika softverskih platformi odnosi se na ograničene performanse, nestabilnost softvera, nedostatak systemske podrške, skalabilnost sistema i mali broj funkcionalnosti. Međutim, ograničenja u pogledu performansi se mogu kompenzovati prirodnom evolucijom performansi računarskih arhitektura. Danas zastupljene varijante softverskih rutera, čija se kontrolna ravan izvršava na personalnom računaru, omogućavaju brzine komutacije od nekoliko Gb/s, što je više nego dovoljno za veliki broj aplikacija. Takođe, *open-source* priroda doprinosi postepenom prevazilaženju najvećeg broja problema vezanih za skalabilnost i dostupnost softverskih funkcionalnosti.

Jedna od najpopularnijih *open-source* softverskih platformi za rutiranje je Quagga, koja je u fokusu ovoga rada. Rad je organizovan na sledeći način. U drugom poglavlju dat je pregled osnovnih mogućnosti Quagga platforme. Postupci konfiguracije RIP i OSPF protokol, kao dva najzastupljenija protokola rutiranja na Internetu, opisani su u trećem i četvrtom poglavlju. U petom poglavlju je opisana procedura izvršenog testiranja brzine konvergencije i predstavljeni su dobijeni rezultati. Na kraju su dati zaključci rada.

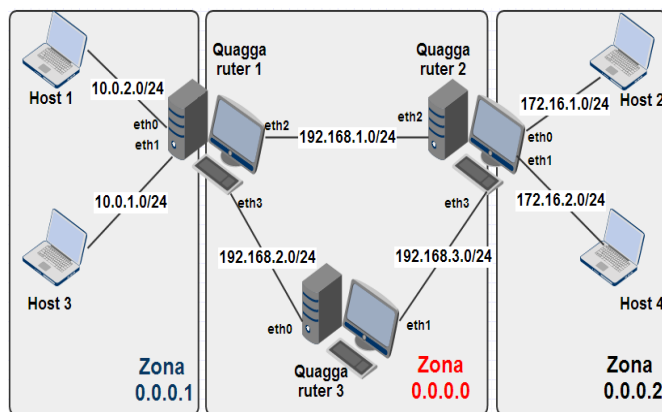
2. QUAGGA SOFTVERSKA PLATFORMA

Quagga je *open-source* softverska platforma koja pruža TCP/IP servise rutiranja. Za sada, Quagga podržava samo *unicast* protokole: RIPv1, RIPv2, RIPng, OSPFv2, OSPFv3, BGP4, BGP4+ i IS-IS na UNIX platformama kao što su: FreeBSD, Linux, NetBSD, OpenBSD i Solaris. Očekuje se da će *multicast* protokoli biti podržani u narednim verzijama. Arhitektura Quagga-e se prilično razlikuje od ostalih softverskih platformi za rutiranje, kod kojih obično jedan proces obavlja sve funkcionalnosti protokola. Quagga koristi

drugačiji pristup. Njen rad se zasniva na nekoliko *daemon* procesa čija je osnovna uloga inicijalizacija i održavanje kernel tabele rutiranja, odnosno komunikacija sa *daemon* procesima koji se izvršavaju na drugim sistemima, u cilju razmjene informacija o rutiranju. Pet *daemon* procesa se vezuju za određene protokole rutiranja i oni se pokreću nezavisno, dok *zebra daemon*, predstavlja glavni proces koji upravlja kernel tabelom rutiranja i vrši redistribuciju ruta između različitih protokola. Potrebno je pokrenuti samo one *daemon*-e koji su pridruženi protokolu rutiranja koji se želi koristiti, dok je *zebra daemon* potrebno aktivirati prilikom svake konfiguracije, jer preko njega ovi protokoli ostvaruju vezu sa kernelom. Svaki od *daemon* procesa ima svoj konfiguracioni fajl i svoj terminal interfejs kome se može pristupiti preko Telnet. Terminal interfejs realizovan je po ugledu na Cisco-ve rutere. Kada je potrebno konfigurisati statičku rutu, to se mora izvršiti u *zebra* konfiguracionom fajlu. U slučaju kada je potrebna konfiguracija RIP, OSPF, IS-IS ili BGP, to se radi u konfiguracionom fajlu pridruženom *ripd*, *ospfd*, *isisd* ili *bgpd daemon* procesu.

3. KONFIGURACIJA RIP PROTOKOLA

RIP je prvi protokol rutiranja koji je korišćen u TCP/IP baziranim mrežama. Iako ovaj protokol posjeduje značajna ograničenja koja su podsticala razvoj novih, kompleksnijih i naprednijih protokola rutiranja, zahvaljujući svojoj stabilnosti i danas je veoma rasprostranjen, naročito u srednjim i malim autonomnim sistemima koji nemaju veliki broj redundantnih putanja. Pokretanje ovog protokola na Quagga platformi zahtijeva aktivaciju *ripd* i *zebra* procesa i kreiranje konfiguracionih fajlova za svaki od njih. Ilustracije radi kreirana je testna platforma prikazana na slici 1.



Slika 1: Testna topologija

Primjer *zebra* konfiguracionog fajla za ruter 1 testne topologije, izgleda ovako:

```
hostname ruter1
password primjer
enable password primjer
interface eth0
interface eth1
interface eth2
interface eth3
ip forwarding
```

U *zebra* konfiguracionom fajlu definišu se interfejsi koji će učestvovati u prosleđivanju i lozinke za VTY (*Virtual Teletype*) konekciju koja omogućava pristup Quagga *daemon*-ima preko Telnet. Komanda **ip forwarding** je

neophodna da bi se omogućilo prosljeđivanje IP datagrama između mrežnih interfejsa. Za konfiguraciju *ripd daemon*-a na istom ruteru može se koristiti fajl sledećeg sadržaja:

```
hostname ruter1
password primjer
key chain test
  key 1
    key string test
interface eth0
interface eth1
interface eth2
  ip rip authentication mode md5 auth-length old-ripd
  ip rip authentication key-chain test
interface eth3
  ip rip authentication mode md5 auth-length old-ripd
  ip rip authentication key-chain test
router rip
  version 2
  network eth0
  network eth1
  network eth2
  network eth3
  timers basic 30 180 120
  offset-list lista in 4 eth2
  distribute-list private in eth2
access-list lista permit 172.16.2.0/24
access-list private deny 172.16.1.0/24
access-list private permit any
```

Osnovna *ripd* konfiguracija zahtijeva navođenje interfejsa preko kojih će ruter periodično obavještavati svoje susjede o zapisima u lokalnoj tabeli rutiranja. Nakon prijema obavještenja ruteri određuju optimalne putanje do svih ostalih destinacija u mreži u skladu su Bellman-Ford algoritmom [1] i ažuriraju svoje tabele rutiranja. Optimalnom rutom kod RIP protokola podrazumijeva se najkraća ruta, odnosno ruta sa najmanjim brojem skokova. Kako je ovo često nepogodno jer onemogućava korišćenje onih ruta sa većim brojem skokova čiji su linkovi rasterećeniji ili imaju veću propusnu moć, Quagga omogućava manipulisanje metrikama ruta. U navedenom primjeru to je urađeno **offset-list** komandom, kojom je ruter podešen da prilikom prijema informacija o ruti 172.16.2.0/24 na *eth2* interfejsu njenu metriku uveća za 4. Quagga ruter podržava obje verzije RIP protokola. Prilikom navođenja verzije protokola u konfiguraciji treba imati u vidu da RIPv1 ne podržava mehanizme za autentifikaciju. U okviru RIPv2 standarda definisane su dvije metode zaštite: jednostavna autentifikacija tekstualnom lozinkom i kriptografska autentifikacija. Quagga podržava oba mehanizma zaštite. Prvi metod je dosta problematičan jer se prisluškivanjem na nivou linka relativno jednostavno može doći do informacije koja se lozinka koristi u sistemu i kasnije lažirati ispravna autentifikacija napadačkog rutera. Zbog ovih problema preporučuje se korišćenje kriptografske autentifikacije bazirane na MD5 (*Message-Digest 5* [2]) heš algoritmu kao što je to urađeno u tesnoj platformi.

Na Quagga platformi je moguće promijeniti i vrijednosti osnovnih RIP tajmera, pomoću **basic timers** komande. Prvi parametar komande (*update tajmer*) definiše interval između uzastopnih oglašavanja rutera i njegova podrazumijevana vrijednost je 30s. Drugi parametar odnosi se na *timeout tajmer* koji se pokreće za svaki od unosa u tabeli rutiranja. Kako se unesene informacije ne mogu trajno smatrati ispravnim, ruteri brišu informacije o ruti nakon isticanja vremena definisanog njenim *timeout tajmerom*. Ukoliko u toku intervala definisanog vrijednošću *timeout tajmera*

(tipično 180s) ruter ponovo primi podatke o toj ruti, ona se smatra "osvježenom" i njen *timeout* tajmer se resetuje. Treći tajmer (*garbage-collection timer*) odbrojava vrijeme nakon kojeg će ruta koja je označena kao nevalidna biti stvarno izbrisana iz tabele rutiranja. Njegov smisao je da obezbijedi dovoljno vremena za propagaciju informacije o gubitku rute ostatku mreže preko redovnih razmjena RIP poruka, i ukoliko se drugačije ne definiše iznosi 120s. Kada su u pitanju mehanizmi za izbjegavanje petlji rutiranja koje predstavljaju veliki problem u RIP mrežama, Quagga podržava samo *split-horizon* [3], koji je aktivan po *default*-u. Filtriranje ruta može se izvršiti **distribute-list** komandom i definisanjem odgovarajuće pristupne liste koja sadrži informacije o tome koje rute treba prihvatiti a koje ne, ali i na druge načine koje nude veću fleksibilnost [4].

4. KONFIGURACIJA OSPF PROTOKOLA

Kod OSPF protokola svaki od rutera u autonomnom sistemu održava bazu podataka koja sadrži informacije o trenutnom stanju mreže. Ruteri određuju stablo optimalnih putanja izvršavanjem Dijkstra algoritma [5]. Optimalnom putanjom smatra se putanja sa najmanjim težinskim faktorom. Vrijednost težinskog faktora se bazira na propusnosti prenosnog medija i to tako što se težinski faktor smanjuje kada se brzina linka povećava. Vrijednosti se mogu definisati i u skladu sa drugim kriterijumima. Pokretanje OSPF protokola na Quagga platformi zahtijeva aktivaciju *ospfd* i *zebra daemon*-a. Kao što je već rečeno, *zebra* proces se mora pokrenuti bez obzira o kojem protokolu rutiranja je riječ, pa se prilikom implementacije OSPF-a na topologiji iz prethodnog primjera mogu koristiti identične konfiguracije za *zebru* kao u slučaju RIP protokola. U nastavku poglavlja, radi sagledavanja osnovnih mogućnosti Quagga-e za implementaciju OSPF protokola, biće razmatran sledeći primjer konfiguracije za ruter 1 testne topologije:

```
hostname ruter1
password primjer
interface eth0
 ip ospf cost 3
interface eth1
 ip ospf cost 3
interface eth2
 ip ospf cost 7
 ip ospf dead-interval 60
 ip ospf hello-interval 10
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 ABCDEFGHIJK
interface eth3
 ip ospf cost 7
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 ABCDEFGHIJK
router ospf
 ospf router-id 1.1.1.1
 network 10.0.1.0/24 area 0.0.0.1
 network 10.0.2.0/24 area 0.0.0.1
 network 192.168.1.0/24 area 0.0.0.0
 network 192.168.2.0/24 area 0.0.0.0
 area 0.0.0.0 authentication message-digest
 area 0.0.0.1 stub no-summary
 default-information originate
 area 0.0.0.1 default-cost 1
 area 0.0.0.1 range 10.0.0.0/22
```

Možda najbitnija prednost OSPF-a u odnosu na RIP je mogućnost hijerarhijskog rutiranja, odnosno podjele autonomnog sistema na više zona. Svaka od zona je

numerisana i kontroliše se od strane rutera koji joj pripadaju. S obzirom da zone mogu razmjenjivati sažete informacije o rutiranju, na ovaj način se značajno smanjuje količina kontrolnog saobraćaja u mreži. Tip informacija o rutiranju koje zona može primiti određuju karakteristike koje joj se dodjeljuju. Kao što se iz navedenog primjera konfiguracije vidi, prilikom definisanja interfejsa koji učestvuju u izvršavanju protokola navodi se i identifikator zone kojoj pripadaju. Kada su u pitanju karakteristike koje se mogu dodijeliti zoni, Quagga podržava standardne, ograničene i potpuno ograničene zone. Ukoliko se zoni eksplicitno ne dodijeli nijedan atribut, ona se smatra standardnom i primaće sve tipove informacija o rutiranju. Zona okosnice (zona 0), koja uvijek mora postojati i na koju su povezane sve ostale zone, mora biti standardna zona. Ograničene zone se definišu dodavanjem *stub* atributa i one ne prihvataju informacije o rutama koje su eksterne u odnosu na autonomni sistem kojem pripadaju. Ukoliko se dodatno upotrijebi **no-summary** atribut, kao u navedenom primjeru kod zone 0.0.0.1, zona se deklarise kao potpuno ograničena i kao takva neće prihvatiti informacije o rutiranju ni iz drugih zona istog autonomnog sistema. Pristup mrežama u ostalim zonama ostvaruje se definisanjem *default* rute prema graničnom ruteru oblasti.

Još jedna interesantna mogućnost je *sumiranje ruta*, odnosno konsolidacija većeg broja ruta u jednu *super rutu*. Sumiranjem se direktno utiče na mrežnu propusnost i resurse koje OSPF proces koristi na ruterima, s obzirom da kao posledicu imamo manju količinu kontrolnog saobraćaja, a samim tim i manju učestanost rekalkulacija tabela rutiranja. Sumiranje ruta se može izvršiti samo na graničnim ruterima zona, što je u navedenom primjeru učinjeno komandom:

```
area 0.0.0.1 range 10.0.0.0/22
```

koja sumira rute 10.0.1.0/24 i 10.0.2.0/24 u super rutu 10.0.0.0/22.

Iz primjera se intuitivno može zaključiti da se prilikom definisanja interfejsa koji učestvuju u OSPF procesu, **ip ospf cost** naredbom manipuliše težinskim faktorima na njihovim priključenim linkovima. Autentifikacija se realizuje na sličan način kao kod RIP protokola, pri čemu autentifikacijski ključ funkcioniše kao lozinka među OSPF ruterima u istoj zoni. Takođe, postoji mogućnost podešavanja *hello* i *dead* intervala. *Hello* interval definiše vremenski razmak između uzastopnih slanja *Hello* paketa koje OSPF koristi za uspostavljanje i održavanje susjedstva sa ostalim ruterima (tipično 10s). Ukoliko ruter prestane da prima *Hello* pakete, nakon isticanja vremena definisanog trajanjem *dead* intervala (tipično 40s) taj susjed se proglašava nefunkcionalnim.

5. KONVERGENCIJA RIP I OSPF PROTOKOLA NA QUAGGA PLATFORMI

Vrijeme konvergencije je vrijeme koje je potrebno da se saobraćaj preusmjeri optimalnom putanjom nakon nekog događaja u mreži. Kako mreža nije u potpunosti funkcionalna tokom ovog vremena, brzina konvergencije predstavlja jedan od ključnih kriterijuma za izbor protokola rutiranja. Mrežna konvergencija zahtijeva da svi "pogođeni" ruteri obrade nastali događaj i ažuriraju svoje tabele prosljeđivanja. Što je vrijeme koje ova operacija zahtijeva manje, to se protokol rutiranja može smatrati optimalnijim po ovom pitanju.

U cilju ispitivanja brzine konvergencije RIP i OSPF protokola korišćena je testna topologija sa slike 1. Na ruterima su korišćene samo osnovne konfiguracije protokola, koje ne uključuju nikakvo filtriranje, promjene metrika ili podjelu autonomnog sistema na više zona (kod OSPF protokola). U analizi je korišćena **ping** komanda za slanje paketa od hosta 1 do hosta 2, a brzina konvergencije je računata na osnovu broja izgubljenih paketa tokom procesa ažuriranja tabela rutiranja i uspostavljanja alternativne putanje preko rutera 3 nakon deaktivacije eth2 interfejsa rutera 1.

Rezultati mjerenja vremena konvergencije RIP protokola prikazani su u Tabeli 1. Spora konvergencija RIP protokola je, između ostalog, posledica velikih *default* vrijednosti njegovih tajmera. Brzina konvergencije može se značajno uvećati smanjivanjem ovih vrijednosti. Na primjer, promjenom vrijednosti *update*, *timeout* i *garbage collection* na 10, 30 i 20 sekundi, respektivno, vrijeme konvergencije se svelo na 30s. Međutim, ove vrijednosti potrebno je pažljivo birati u zavisnosti od topologije. Ukoliko su tabele rutiranja velike onda smanjivanje *update* tajmera može dovesti do zagušenja na sporim linkovima. Takođe, velike vrijednosti *timeout* tajmera uzrokuju sporu reakciju na promjene, dok previše male vrijednosti dovode do nestabilnosti.

Tabela 1. Rezultati mjerenja vremena konvergencije RIP protokola nakon isključenja interfejsa eth2 rutera 1

Redni broj mjerenja	Broj poslanih paketa	Izgubljeni paketi (%)	Vrijeme konvergencije (s)
1	250	72	181
2	250	66	164
3	250	69	172
4	250	79	197
5	250	67	168
6	250	76	191
7	250	70	174
8	250	62	155
9	250	75	191
10	250	76	187
Srednja vrijednost:	250	71	178

Za razliku od RIP-a čija konvergencija može potrajati nekoliko minuta s obzirom da se tabela rutiranja svakog rutera kopira i dijeli direktno spojenim susjedima, konvergencija kod OSPF-a je mnogo brža pošto se prenose samo informacije o promjeni drugim ruterima unutar mreže. Prilikom mjerenja brzine konvergencije interval između uzastopnih slanja ICMP paketa podešen je na 100ms. Rezultati deset izvršenih mjerenja pri istim uslovima pokazuju da potrebno vrijeme za konvergenciju iznosi svega 200ms. Međutim, OSPF ne konvergira uvijek ovako brzo. Znatno više vremena je potrebno da ruteri ponovo presumjere saobraćaj optimalnom putanjom nakon povratka mreže u prvobitno stanje. U ovom konkretnom primjeru, kada je testiranje vršeno deaktivacijom eth2 interfejsa rutera 1, to vrijeme iznosi oko 8 sekundi. Pored ovog scenarija, ista mjerenja su izvršena i nakon isključenja linka između rutera 1 i rutera 2. Iako su ova dva eksperimenta djeluju slično, procesi konvergencije se bitno razlikuju. Nakon ponovnog

uključenja linka mreži je u prosjeku potrebno više od 40 s da konvergira. Razlog je taj što ruter 1 briše rutera 2 iz „liste susjeda” pa se nakon uključivanja linka odnosi između njih postepeno razvijaju od inicijalnog stanja do stanja potpune povezanosti [6], dok se kod prvog slučaja u stanje potpune povezanosti prelazi odmah nakon prijema prvog *hello* paketa. U ovom radu usled nedostatka prostora nijesu objašnjene faze kroz koje se odnosi između susjednih rutera razvijaju, ali riječ je o kompleksnim i vremenski zahtjevnim procedurama.

6. ZAKLJUČAK

Protokoli rutiranja predstavljaju kritične tačke IP mreža. Kao alternativa specijalizovanom hardveru, sve više se koriste softverske platforme za rutiranje, u kombinaciji sa računarnom opšte namjene. Njihovom instalacijom, od običnog računara sa većim brojem mrežnih kartica moguće je napraviti softverski ruter čije su performanse prihvatljive za veliki broj aplikacija. Pored ovog izuzetno ekonomičnog rješenja, mogu se koristiti i za realizaciju kontrolne ravni rutera većeg kapaciteta koji koriste specijalizovani hardver za prosljeđivanje paketa. Zahvaljujući svojoj *open-source* prirodi, softverske platforme naročito su interesantne u istraživačkim krugovima koji na ovaj način mogu razvijati nove protokole i testirati ih u realnom mrežnom okruženju. Quagga predstavlja jednu od najpopularnijih *open-source* softverskih platformama za rutiranje. Rezultati izvršenih testiranja koji su prikazani u ovom radu ilustruju mogućnosti RIP i OSPF protokola na testnoj platformi. Podrška za ove protokole je od velike važnosti imajući u vidu njihovu rasprostranjenost na Internetu. U radu su takođe predstavljene rezultati mjerenja brzine konvergencije koji pokazuju nesumnjivu prednost OSPF protokola nad RIP-om po tom pitanju. U toku rada sa platformom identifikovano je nekoliko nedostataka aktuelnih implementacija RIP-a i OSPF-a. Nadogradnja koda u cilju njihovog prevazilaženja plan je daljeg istraživanja.

ZAHVALNICA

Rad je dijelom finansiran sredstvima projekta „Razvoj *Future Internet* rutera na programabilnim hardverskim platformama otvorenog koda“, sufinansiranim od strane Ministarstva za nauku Crne Gore i Ministarstva za informaciono društvo i telekomunikacije.

LITERATURA

- [1] Richard Bellman, “On a routing problem”, *Quarterly of Applied Mathematics* 16: 87–90, 1958.
- [2] Internet Engineering Task Force, “RFC 1321 – The MD5 Message-Digest Algorithm”, April 1992.
- [3] Internet Engineering Task Force, “RFC 2453 – RIP Version 2”, Novembar 1998.
- [4] Kunihiro Ishiguro, “Quagga 0.99.18, A routing software for TCP/IP networks”, Mart 2011.
- [5] Dijkstra, E. W., “A Note on Two Problems in Connection with Graphs”, *Numerische Math.* 1, 269-271, 1959.
- [6] Internet Engineering Task Force, “RFC 2328 – OSPF Version 2”, April 1998.

DISTRIBUCIJA UNIVERZALNOG KOORDINISANOG VREMENA DISTRIBUTION OF UNIVERSAL COORDINATED TIME

Rabina Šabotić, Milutin Radonjić, Igor Radusinović, *Elektrotehnički fakultet Podgorica*

Sadržaj: *Različiti zahtjevi korisnika u pogledu tačnog vremena uslovljavaju primjenu različitih metoda za distribuciju vremena. U ovom radu su prikazane najvažnije metode za distribuciju univerzalnog koordinisanog vremena, njihove karakteristike i tačnost koja se njima postiže. Posebno je opisana distribucija vremena preko Interneta, korišćenjem NTP (Network Time Protocol) protokola, koja predstavlja najčešće korišćenu metodu. Na kraju, opisan je i PTP (Precision Time Protocol) protokol, kojim se postiže veća tačnost sinhronizacije vremena.*

Abstract: *Different user requirements regarding to the time accuracy force the use of different methods for time distribution. In this paper, the methods for UTC time distribution is presented, their characteristics and accuracy that is achieved by them. Distribution over the Internet, using the NTP (Network Time Protocol), is the most frequently used method, and it is particularly described in this paper. Finally, PTP (Precision Time Protocol), which achieves higher accuracy of time synchronization, is presented.*

1. UVOD

Vrijeme predstavlja koncept koji je čovjek razvio kako bi pratio promjene. Praktično od postanka čovječanstva ljudi su vrijeme mjerili na razne načine. Međutim, pod pojmom modernog mjerenja vremena se podrazumijeva mjerenje pomoću uređaja koji pokazuju period pojave određenih periodičnih promjena u prirodi, kao što su rotacija Zemlje oko svoje ose, rotacija Zemlje oko Sunca, promjena atomskih stanja elemenata i dr. Istorijski razvoj mjerenja vremena i primjena novih tehnika, koje omogućavaju preciznije mjerenje periodičnih pojava u prirodi sa jedne i zahtjeve različitih grupacija korisnika sa druge strane podrazumijevaju definisanje različitih vremenskih skala.

Na Generalnoj konferenciji za tegove i mjere, CGPM (*Conférence Générale des Poids et Mesures*) 1975. godine, preporučena je upotreba Univerzalnog koordinisanog vremena, UTC (*Universal Time Coordinated*) kao osnove za zakonsko vrijeme. UTC nema fizičku realizaciju, već postoje samo lokalne realizacije institucija koje učestvuju u njegovom formiranju UTC(k), gde k predstavlja instituciju u kojoj je vremenska skala generisana, a to su pretežno nacionalni metrološki instituti. Odstupanje UTC(k) od UTC-a je reda 1 μ s, a samo jedan manji broj lokalnih aproksimacija UTC-a odstupa manje od 100 ns. Greška UTC(k) skale je poznata i objavljuje se od strane Međunarodnog biroa za tegove i mjere, BIPM (*Bureau International des Poids et Mesures*) u mjesečnom biltenu Circular T. Ona je zanemariva u odnosu na grešku nastalu uslijed preuzimanja UTC(k) vremena od strane korisnika [1]. Korisnici imaju pristup nekoj od lokalnih aproksimacija UTC-a preko nacionalnih sistema za distribuciju, satelitskih sistema ili preko Interneta. Precizno mjerenje vremena se zahtjeva u svim aspektima života, a prije svega u telekomunikacijama, bankarskim transakcijama, raznim mjernim sistemima koji se koriste u nauci itd. Tačnost vremena koja je potrebna korisniku određuje metodu za distribuciju UTC(k) vremena.

NTP (*Network Time Protocol*) protokol predstavlja najčešće korišćeni protokol za distribuciju UTC(k) - a koji se primjenjuje u Ethernet mrežama. NTP je u stanju da izmjeri

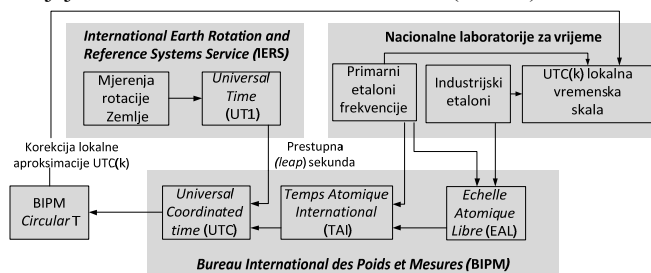
kašnjenje TCP/IP paketa, veoma je robustan, i njegovim korišćenjem se postiže odstupanje od UTC-a od 1 do 100ms. U ovom radu su predstavljeni rezultati odstupanja lokalnog vremena personalnog računara pri sinhronizaciji sa NTP serverom. Akvizicija rezultata je izvršena programom Wireshark koji služi za prikupljanje podataka radi lakše analize. Rezultati koji su dobijeni na osnovu podataka snimljenih Wireshark-om i jednačina koje služe za izračunavanje vremenskog odstupanja i kašnjenja paketa su upoređeni sa rezultatima dobijenim upotrebom komande **ntpq -p** na Linux platformi. Dobijeni su isti rezultati. Da bi se prikupili podaci o poslatim i primljenim paketima, potrebno je da klijent izvrši niz uzastopnih sinhronizacija u određenim vremenskim intervalima. Na osnovu rezultata ustanovljeno je da se primjenom ovog protokola u Ethernet mreži dobijaju vrijednosti za odstupanje lokalnog vremena personalnog računara od NTP servera manje od 100 ms.

Zahtjevi u pogledu preciznosti frekvencije i vremena se povećavaju sa razvojem telekomunikacionih mreža. NTP protokol nije u stanju da ispuni zahtjeve koji su potrebni u mobilnim 3G/4G mrežama. PTP (*Precision Time Protocol*) protokol je, za razliku od NTP protokola, nadvladao uticaj kašnjenja i *jitter*-a, čime je postignuta tačnost sinhronizacije reda veličine nanosekunde. PTP protokol predstavlja odgovor na izazove vremenske sinhronizacije u mrežama sa komutacijom paketa. Ovaj rad prikazuje osnovne principe funkcionisanja PTP protokola.

Rad je organizovan na sljedeći način. U drugom poglavlju je opisano Univerzalno koordinisano vrijeme. O pojedinim metodama distribucije UTC(k) vremena, osobinama i tačnošću koje ih karakterišu se govori u trećem poglavlju. NTP protokol, njegovo funkcionisanje i tačnost sinhronizacije koja se njime postiže, je obrađen u četvrtom poglavlju. S ciljem da se unaprijede postojeće sinhronizacione metode, kao što je NTP, i omogući bolja tačnost vremenske sinhronizacije, razvijen je PTP protokol. U petom poglavlju je opisan PTP protokol i njegove prednosti u odnosu na NTP protokol. Na kraju su, u šestom poglavlju, date zaključne konstatacije.

2. UNIVERZALNO KOORDINISANO VRIJEME

UTC predstavlja kompromisno rješenje između dvije vrste vremenskih skala: univerzalnih, zasnovanih na rotaciji Zemlje, i atomskih, zasnovanih na radu atomskih etalona. UTC vrijeme predstavlja atomsko vrijeme sa, po potrebi, umetnutom prestupnom sekundom, da bi se obezbijedilo da maksimalna razlika između UTC i astronomske UT1 (*Universal Time*) vremena bude 0,9 sekundi. Prestupna sekunda se dodaje ili oduzima u ponoć 30. juna ili 31. decembra. UTC je zvanična vremenska skala od 1. januara 1972. godine. Ova skala je proračunska skala čiji se rezultati dobijaju nakon statističke obrade u BIPM (slika 1).



Slika 1. Pojednostavljena šema formiranja UTC - a

UTC ima za cilj da ne odstupa mnogo od definicije sekunde, da bude stabilan i pouzdan. Ovo se postiže obradom podataka o vremenu sa približno 450 atomskih etalona u 70 laboratorija. Koristeći algoritam ALGOS, BIPM računa srednju vrijednost vremenskih razlika sa pomenutih atomskih etalona, i na taj način se formira slobodna atomska skala, EAL (*Echelle Atomique Libre*). Da bi se poboljšala stabilnost frekvencije slobodne atomske skale, realizuje se procedura pridruživanja težinskih faktora etalonima. Težinski faktor se razmatra kao konstanta za vrijeme od 30 dana računanja, a kontinuitet sa prethodnim periodom je obezbijeden predikcijom frekvencije etalona [2]. Maksimalna relativna težina je fiksirana za svaki period računanja, da bi se izbjegla mogućnost da veoma stabilni etaloni beskonačno povećaju svoje težine i postanu dominantni u skali. Algoritam je u mogućnosti da detektuje abnormalno ponašanje etalona i da ga zanemari ako je neophodno. Ova osobina je realizovano u vidu iterativnog procesa koji počinje sa težinskim faktorima dobijenim u prethodnom mjesecu i koriste se kao indikator ponašanja etalona u mjesecu računanja. Dužina sekunde računata na ovaj način se poredi sa podacima primarnih etalona frekvencije. Danas je najbolja realizacija SI sekunde realizovana primarnim etalonima frekvencije koji su razvijeni u nekoliko laboratorija za vrijeme, sa tačnošću reda 10^{-15} / definisani vremenski period. Ovaj vremenski period može biti 10, 15, 20, 25 ili 30 dana. Da bi se frekvencija održala što bliže SI sekundi, na prethodno računanje se primjenjuju korekcije frekvencije. Rezultujuća vremenska skala predstavlja Međunarodno atomsko vrijeme - TAI (*Temps Atomique International*) [2].

3. METODE ZA DISTRIBUCIJU VREMENA

Postoje tri načina za prenos vremena i frekvencije u telekomunikacijama: jednosmjerni, dvosmjerni i prenos istovremenog posmatranja.

U slučaju jednosmjernog prenosa vremena i frekvencije, jedna strana komunikacionog linka prenosi drugoj strani

njegovu frekvenciju ili vrijeme, ili oboje. Predajna strana može simultano da šalje frekvenciju i vrijeme različitim prijemnicima. Neki od primjera jednosmjernog prenosa vremena i frekvencije su GPS, LORAN i DCF77 [3]. Jednosmjerna GPS metoda mjerenja koristi signal sa GPS prijemnika kao referencu za kalibraciju etalona. Metoda sinhronizacije vremena sastoji se u tome da 1 pps ili signal 10MHz iz GPS prijemnika predstavlja ulaz za frekvencijometar. Primjenom ovog sistema moguće je postići usaglašavanje sa UTC vremenom bolje od 50 ns.

Kod dvosmjernog prenosa vremena i frekvencije obje strane komunikacionog kanala istovremeno prenose i primaju informacije. Na taj način se formiraju dva različita kanala jednosmjernog prenosa u cilju određivanja razlike između lokalnog i referentnog etalona. Primjeri ovog prenosa su TWSTFT (*Two-way satellite time and frequency transfer*) i NTP [3]. TWSTFT predstavlja jednu od najtačnijih metoda za komparaciju etalona sa tačnošću reda 1 ns.

Vremenska razlika između dva etalona se može odrediti simultanim poređenjem svakog etalona sa zajedničkim referentnim signalom, čija tačnost nije bitna jer obje strane primaju signal sa istog satelita u isto vrijeme. GPS metoda istovremenog posmatranja nudi 20 do 30 puta bolje rezultate nego druge slične metode, i danas predstavlja ključnu metodu kod mjerenja i računanja UTC vremena od strane BIPM-a. Ova metoda može koristiti jedan ili više satelita. [3].

U Tabeli 1. su navedeni potencijalni izvori i metode za distribuciju vremena, predstavljene su njihove karakteristike, i prikazana njihova tačnost [1].

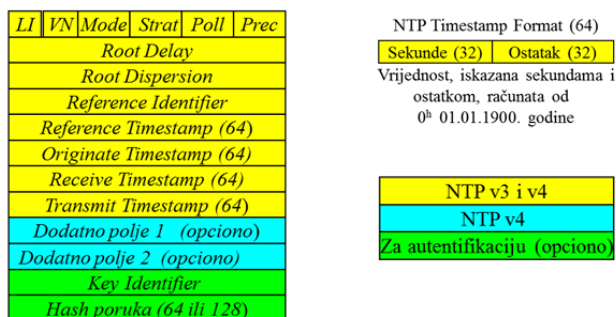
Tabela 1. Karakteristike pojedinih metoda

Tip	Vremenska (frekvencijska) tačnost	Pokrivenost (Sistem)
LF broadcast	1 ms (10^{-10} do 10^{-11})	Globalno (DCF 77)
Broadcast TV signala	0,5 do 10 μ s (10^{-10} do 10^{-11})	Regionalno (DBS)
	10 do 100 ns (10^{-12} do 10^{-13})	Regionalno (DBS)
Navigacioni satelit, broadcast	20 do 500 ns (10^{-9} do 10^{-13})	Globalno (GPS i GLONASS)
Navigacioni satelit, common view	5 do 20 ns (10^{-13} do 10^{-15})	Globalno (GPS i GLONASS)
Komunikacioni satelit, two - way	1 do 10 ns (10^{-14} do 10^{-15})	Regionalno (TWSTFT)
Sistem sa optičkim vlaknima	10 do 50 ps (10^{-16} do 10^{-17})	Lokalno, do 50km
	100 ns (10^{-13} do 10^{-14})	Do 2000 km (SDH mreže)
Mikrotalasni link	1 do 10 ns (10^{-14} do 10^{-15})	Lokalno
Koaksijalni kabl	1 do 10 ns (10^{-14} do 10^{-15})	Lokalno

4. NTP PROTOKOL

Najnovija varijanta NTP protokola je NTPv4 [4]. Osnovu sistema distribucije vremenskog podatka čine mrežni vremenski server - NTS (*Network Time Server*). Generatori

takta su podijeljeni na nivoe, stratum, koji definišu udaljenost od primarnog referentnog izvora. Najviši nivo je stratum 0. Kao eksterne reference, za generisanje što tačnijeg vremenskog podatka, NTS koriste: 1 pps, 10MHz, GPS, kodovani IRIG B signal i neki drugi referentni NTS. Format NTP poruke je prikazan na slici 2.



Slika 2. Format NTP poruke

LI (*Leap Indicator*) predstavlja upozorenje za predstojeću prestupnu sekundu s ciljem da se ubaci/izbriše u posljednjem minutu tekućeg dana.

VN (*Version Number*) se odnosi na verziju NTP protokola.

Mode ukazuje na odgovarajuću varijantu NTP protokola.

Stratum predstavlja nivo i ima sljedeće vrijednosti: 0 ukoliko je nedefinisan ili pogrešan; 1–primarni server; (2–15) – sekundarni server; 16–nesinhronizovan; (17–255)– rezervisan.

Poll predstavlja maksimalni interval između slanja paketa.

Precision, predstavlja preciznost lokalnog generatora takta.

Root delay se odnosi na kašnjenje, RTD (*roundtrip delay*), u odnosu na primarni referentni izvor.

Root Dispersion predstavlja nominalnu grešku u odnosu na primarni referentni izvor.

Reference ID označava primarni referentni izvor.

Reference Timestamp je vrijeme kada je lokalni generator takta posljednji put sinhronizovan.

Originate Timestamp je vrijeme kada je klijent poslao zahtjev.

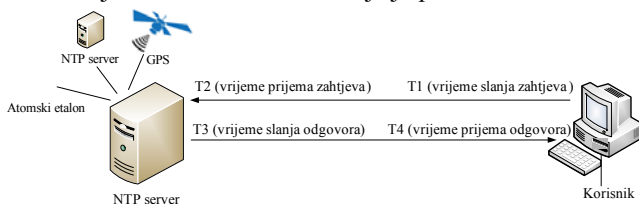
Receive Timestamp je vrijeme kada je server primio zahtjev.

Transmit Timestamp je vrijeme kada je server poslao odgovor.

Destination Timestamp je vrijeme kada je klijent primio odgovor od servera.

Key Identifier i Hash poruka sadrže MAC (*Message Authentication Code*) informaciju [4].

Primjena NTP protokola je jednostavna. Ona se odvija u skladu sa specifičnim algoritmom baziranim na periodičnoj razmjeni *timestamp*-ova između NTP klijenta i NTP servera. Proces klijent – server komunikacije je prikazan na slici 3.



Slika 3. Proces komunikacije klijent - server

NTP klijent šalje zahtjeve i na osnovu odgovora od NTP servera periodično podešava svoje tačno vrijeme. Što se češće vrši podešavanje generatora takta klijenta to će i odstupanje

biti manje. NTP serveri koriste UDP protocol i port 123 za komunikaciju sa drugim serverima i klijentima.

On – *wire* protokol predstavlja mehanizam po kojem se razmjenjuju paketi između klijenta i servera, i obezbjeđuje da se taj proces odvija bez gubitaka ili lažnih paketa [5]. Proces slanja paketa se obavlja u intervalima koji se nalaze u opsegu od 8s do 36h. Za sinhronizaciju je neophodno razmijeniti nekoliko paketa između klijenta i servera, pri čemu se svaki paket provjerava. Paketi koji ne ispune uslove testova, odbacuju se. Referentno vrijeme se dobija kombinovanjem vremena servera koji su zadovoljili testove. U najvećem broju slučajeva, potrebno je oko pet dobrih paketa da bi se NTP server prihvatilo kao izvor prihvatljiv za sinhronizaciju [6].

Za sinhronizaciju sa NTS-om, klijent mora da zna kašnjenje u mreži (*delay*) i vremensko odstupanje (*offset*), koji se računaju na osnovu *timestamp*-ova, po formuli:

$$offset = [(T2 - T1) + (T3 - T4)] / 2 \quad (1)$$

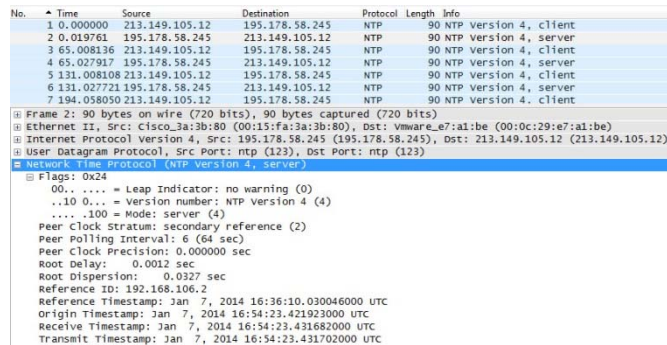
$$delay = (T4 - T1) - (T3 - T2) \quad (2)$$

pri čemu je (slika 3):

T1 - *Origin Timestamp*; T2 - *Receive Timestamp*;

T3 - *Transmit Timestamp*; T4 - *Destination Timestamp*.

Na slici 4. prikazana je NTP poruka snimljena *Wireshark*-om.



Slika 4. NTP poruka snimljena *Wireshark*-om

Na osnovu vrijednosti odgovarajućih *Timestamp*-ova i jednačina (1) i (2) dobijaju se vrijednosti za *offset* i *delay* sinhronizacije klijenta sa NTP serverom: *offset* = -0.12ms i *delay* = 19.758ms.

Poređenja i verifikacije radi, kompletna podrška za NTP je implementirana na personalnom računaru sa CentOS operativnim sistemom. Upotrebom komande `ntpq -p` dobijeni su podaci o sinhronizaciji klijenta sa NTP serverom (slika 5), gdje vrijednosti za *offset* i *delay* (date u ms), odgovaraju vrijednostima dobijenim na osnovu podataka snimljenih *Wireshark*-om i upotrebom jednačina (1) i (2).

remote	refid	st	t	when	poll	reach	delay	offset	jitter
195.178.58.245	192.168.106.2	2	u	1	64	3	19.758	-0.120	0.140

Slika 5. Vrijednosti pri sinhronizaciji klijenta sa serverom

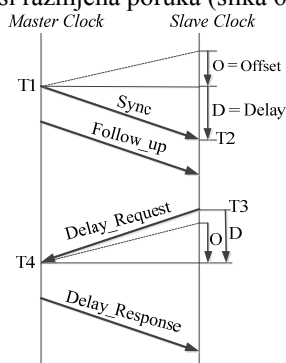
Tačnost sinhronizacije koja se može postići NTP protokolom zavisi prije svega od izvora koji se koristi. Klijent nikad ne može biti tačniji od servera na koji je sinhronizovan, pri čemu tačnost na strani klijenta zavisi u najvećoj mjeri od kompleksnosti infrastrukture mreže,

opterećenosti mreže i njene simetričnosti. Da bi se obezbijedila NTP sinhronizacija potrebno je da vremenska razlika između servera i klijenta ne bude veća od 128ms. Pomoću ovog protokola moguće je održavati tačnost unutar intervala od 1ms do 100ms, a pod idealnim uslovima u LAN mrežama se mogu postići tačnosti bolje od 1ms.

Distribucija vremenskog podatka se može, u zavisnosti od tipa korisnika, podijeliti na restriktivnu i javnu. U prvoj grupi se nalaze oni korisnici koji vremenski podatak koriste u javnoj, poslovnoj i zvaničnoj komunikaciji i čiji su zahtjevi, prvenstveno u pogledu tačnosti i zaštite, od vitalnog značaja za njih. Oni su obično povezani na NTS stratum 1. U drugu grupu spadaju svi ostali i nalaze se u mreži nižeg stratuma.

5. PTP PROTOKOL

IEEE1588v2 je najnovija varijanta PTP koja je standardizovana 2008. godine. Funkcionisanje ovog protokola je zasnovano na transferu poruka i uspostavljanju MS (*Master - Slave*) hijerarhije. PTP protokol koristi BMC (*Best Master Clock*) algoritam, kojim se određuje generator takta sa najboljim karakteristikama u mreži. Odabrani generator takta, GMC (*Grand Master Clock*), potom vrši sinhronizaciju svih podređenih generatora takta u mreži. Ukoliko se utvrdi da odabrani generator takta nema najviši nivo kvaliteta u mreži, tada algoritam određuje novi generator takta. Nakon formiranja MS hijerarhije, između master generatora takta (*master clock*) i podređenih generatora takta (*slave clock*) se vrši razmjena poruka (slika 6).



Slika 6. Razmjena PTP poruka

Master šalje ciklično, u definisanim intervalima (svake 2s), *Sync* poruke, koje sadrže tekuće vrijeme master generatora takta. Informacija o vremenu u *Sync* poruci nije u potpunosti tačna, pa se to vrijeme mjeri i šalje preko naredne poruke, *Follow_up*. Nakon prijema *Sync* i *Follow_up* poruka, *slave* izračunava korekcionu *offset*. Još jedna razmjena poruka je potrebna da bi se izračunalo kašnjenje između *master* i *slave* generatora takta. *Slave* šalje paket *Delay_Request*, a *master* nakon prijema paketa šalje *Delay_Response*. *Slave* određuje *delay* i *offset* preko sledećih formula:

$$delay = [(T2 - T1) + (T4 - T3)] / 2 \quad (3)$$

$$offset = [(T2 - T1) - (T4 - T3)] / 2 \quad (4)$$

Prilikom izračunavanja ovih parametara pretpostavlja se da su kašnjenja simetrična. Ove pretpostavke se mogu uzeti u obzir za potrebe sinhronizacije lokalnih mreža, za koje je PTP namijenjen. Na taj način se dobija tačnost bolja od 100ns. U opštem slučaju, postiže se tačnost manja od 1μs i

zavisi od intervala *Sync* poruke, stabilnosti generatora takta, opterećenosti mreže i njene simetrije u oba smjera prenosa. Na tačnost bitno utiče i upotreba BC (*Boundary Clock*) generatora takta koji se implementira umjesto *switch*-eva ili rutera i TC generatora takta (*Transparent Clock*) kojim se postiže da karakteristike podređenih generatora takta više ne zavise od karakteristika generatora takta koji im prethode [7].

6. ZAKLJUČAK

Vremenski sistemi predstavljaju skup sredstava i uređaja, objedinjenih u okviru jedinstvenog zadatka, čija je uloga realizacija i distribucija vremenskog podatka, njegova konverzija, obrada, čuvanje i predstavljanje u obliku pogodnom za čovjeka i/ili mašinu. Izuzetno dinamičan tehnološki razvoj u oblastima elektronike, telekomunikacija i informacionih tehnologija značajno je uvećao ulogu podatka o tačnom vremenu i postavio veoma stroge zahtjeve za vremenske sisteme. Različiti zahtjevi korisnika u pogledu tačnosti uslovljavaju primjenu različitih metoda za distribuciju vremena, i neke od njih su prikazane u radu.

NTP je najčešće korišćen protokol zato što je široko dostupan, pristupačan, pokazuje dobre performanse u LAN i WAN mrežama, i zahtijeva minimalne sistemske resurse. NTP protokol se, iako je prevashodno namijenjen za distribuciju informacije o vremenu, može koristiti i za distribuciju informacije o frekvenciji. Ipak, glavni nedostaci NTP protokola su evidentno pogoršanje karakteristika usled zagušenja koje se pojavljuju u mrežama sa komutacijom paketa i tačnost reda milisekunde. Zahtjevnije potrebe za sinhronizacijom, koje se prije svega javljaju u 4G mobilnim mrežama, dovele su do razvoja PTP protokola kojim se postiže tačnost bolja od 1μs.

Dalji rad u ovoj oblasti usmjeren je na proučavanje novih metoda kojima bi se postigla tačnost sinhronizacije bolja od 1ns. Jedna od tih metoda, koja zadovoljava ovaj uslov, je kombinacija PTP protokola, SyncE (*Synchronous Ethernet*) i DDMTD (*Digital Dual-Mixer Time Difference*). Ovo rješenje omogućava povezivanje na udaljenosti do 10km preko hiljadu čvorova u optičku mreži [8].

LITERATURA

- [1] Dawid W. Allan, Neil Ashby, Clifford C. Hodge, "The Science of Timekeeping", Hewlett Packard Application Note 1289, 1997.
- [2] Elisa Felicitas Arias, "The Metrology of time", Philosophical Transactions of The Royal Society, 363, 2289-2305, 2005.
- [3] J. Levine, "A review of Time and Frequency Transfer Methods", Metrology, vol.45, no. 6, pp. 162-174, 2008.
- [4] RFC 5905, Network Time Protocol Version 4 Protocol and Algorithms Specification.
- [5] <http://www.eecis.udel.edu/~mills/onwire.html>
- [6] <http://www.ntp.org/ntpfaq/NTP-s-algo.htm/>
- [7] Hans Weibel, "The Second Edition of the High Precision Clock Synchronization Protocol", University of the Applied Sciences, Zurich 2009.
- [8] <http://www.whiterabbitsolution.com/>

V2X SISTEMI KOOPERATIVNE MOBILNOSTI V2X SYSTEMS OF COOPERATIVE MOBILITY

Nedeljko Lekić, Almir Gadžović, Igor Radusinović *Elektrotehnički fakultet, Podgorica*

Sadržaj: U ovom radu su predstavljeni V2X sistemi kooperativne mobilnosti. V2X je zajednički naziv za V2V (Vehicle-to-Vehicle) i V2I (Vehicle-to-Infrastructure) komunikaciju. V2X sistemi omogućavaju vozilima da razmjenjuju podatke između sebe, kao i sa putnom infrastrukturom. V2X komunikacija je bazirana na 5.9GHz DSCR (Dedicated Short Range Communication) radio komunikaciji. To je dvosmjerna, bežična komunikacija kratkog dometa specijalno dizajnirana za objekte u pokretu. V2X komunikacija je slična Wi-Fi komunikaciji uz korištenje efikasnih ad-hoc funkcija umrežavanja. Krajnji cilj ove tehnologije jeste da se kontrola rukovanja automobilom polako prenese sa vozača na samo vozilo.

Abstract: In this paper we present V2X systems of cooperative mobility. V2X is the common name for V2V (Vehicle-to-Vehicle) and V2I (Vehicle-to-Infrastructure) communication. V2X systems allow vehicles to exchange data with other vehicles and roadside units. V2X is based on 5.9GHz DSCR (Dedicated Short Range Communication) radio communication. It is a two-way, short-range wireless communication technology designed especially for moving objects. V2X communication is alike to Wi-Fi communication, but using efficient ad-hoc networking function. The ultimate goal of this technology is to slowly transfer the control over the vehicle from the driver to the vehicle itself.

1. UVOD

U cilju obezbjeđenja sigurnijih i komfornijih uslova na putu, putni operateri, infrastruktura, vozila, vozači i ostali učesnici u saobraćaju moraju međusobno razmjenjivati podatke. Od V2X sistema se očekuje da pruži ključni doprinos u razvoju koncepta kooperativne mobilnosti. Uspjeh koncepta zavisi od broja vozila opremljenih V2X tehnologijom. Sa V2X opremom svaki automobil je u stanju da prati situaciju i opaža dešavanja u svojoj okolini. Informacije o vozilima u svojoj blizini, i situaciji na putu, automobil može iskoristiti za određivanje tekuće i budućih pozicija, stvarajući situacionu svijest. Ovo omogućava realizaciju mnogih sigurnosnih servisa za izbjegavanje saobraćajnih udesa i predviđanje opasnih situacija.

Cilj ovog rada je da čitaocu pruži osnovne informacije o V2X tehnologiji, njenim mogućnostima i izazovima koji se pred nju postavljaju. U radu je dat i poseban osvrt na V2X komunikacione standarde, kao i pregled značajnijih naučno istraživačkih projekata koji se bave ovom tematikom.

Rad sadrži 5 poglavlja. Prvo poglavlje je tekući uvod. U drugom poglavlju, uvodu, date su osnovne informacije o V2X tehnologiji. U poglavlju 3 dat je pregled komunikacionih protokola i standarda prilagođenih primjeni u V2X senzorskim mrežama. U poglavlju 4 predstavljeni su značajni naučno-istraživački projekti fokusirani na razvoj V2X koncepta. Konačno, poglavlje 5 rezimira izneseno, sadrži zaključke i očekivanja u pogledu dalje implementacije.

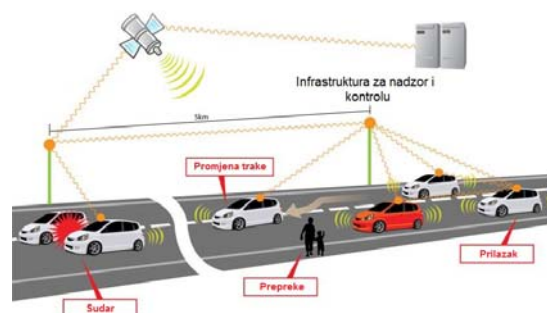
2. V2X SISTEMI: MOGUĆNOSTI I IZAZOVI

V2X tehnologija obuhvata dva osnovna vida komunikacije, i to: V2V (Vehicle-to-Vehicle), komunikacija

između vozila (Slika 1), i V2I (Vehicle-to-Infrastructure), komunikacija sa infrastrukturom na putu (Slika 2).



Slika 1. V2V komunikacija – razmjena podataka između vozila.



Slika 2. V2I komunikacija – razmjena podataka sa putnom infrastrukturom.

V2X tehnologija može unaprijediti efikasnost saobraćaja i podići ekološku prihvatljivost istog. V2X putna infrastruktura može da izda upozorenja vozačima, preporuči alternativni pravac ili pomogne prilikom parkiranja u prometnim gradskim sredinama. Semafori opremljeni V2X tehnologijom mogu informisati vozača o vremenu do sljedeće promjene

signalizacije, ili ukazati na optimalnu brzinu za prolazak raskrsnice kroz zeleno svjetlo.

Stručnjaci u automobilske industriji *V2X* tehnologiju posmatraju kao dugo očekivani iskorak u autonomnoj navigaciji [1].

Sredinom 2013. godine objavljeni su prvi, preliminarni rezultati terenskih ispitivanja u kojim je učestvovalo stotine specijalno opremljenih vozila u Njemačkoj (*simTD*) [2] i Sjedinjenim Američkim Državama (*Safety Pilot*) [3]. Ispitivanja su pokazala da je brza, i prije svega pouzdana, komunikacija između svih učesnika u saobraćaju ključni faktor za sigurnije i komfornije putovanje. Kao nastavak rada u ovoj oblasti Evropska Unija je planirala instalaciju *V2I* uređaja na koridoru između Beča i Roterdama, počev od 2015. godine. Ti uređaji, u kombinaciji sa *V2V* tehnologijom, bi trebali da omoguće autonomnu navigaciju za sva vozila koja se kreću tim koridorom.

Istraživanja pokazuju da upotreba *V2X* komunikacije može da dovede do smanjenja potrošnje goriva između 10 i 20% [4]. Mogućnosti smanjenja potrošnje demonstrirane su u [5]. Pokazuje se da *V2X* komunikacija može redukovati srednju potrošnju goriva posebno u sredinama sa velikom gustinom saobraćaja i dugotrajnim ciklusima semeforske signalizacije.

Zahvaljujući autonomnoj navigaciji, primjenom *V2X* sistema, moguće je smanjiti prosječnu udaljenost između vozila i samim tim povećati broj vozila na autoputevima. U [5] je pokazano da se na ovaj način, kapacitet autoputa može povećati za čak 273%. Povećanje je moguće postići bez ikakvih izmjena putne infrastrukture, u smislu povećanja broja saobraćajnih traka, i slično.

Pojavom *V2X* tehnologije, počinje da se nazire budućnost u kojoj ljudima neće biti potrebna vozačka dozvola, već će automobil upravljati samim sobom.

V2X tehnologija donosi i brojne poslovne i tehničke izazove koji se moraju riješiti. Među glavnim izazovima su tačno pozicioniranje vozila, održiv sigurnosni concept, skalabilnost sveukupnog sistema na milione vozila i brz prodor tehnologije u sve automobile. Tako složen sistem podrazumijeva poznavanje osnovnih aspekata tehnologije duž cijelog obuhvaćenog lanca, od opreme u vozilu, putne infrastrukture, sigurnosnog sistema, centralnog sistema do neophodno prisutnih poslovnih procesa.

Da bi *V2X* tehnologija postala standard i obavezni dio opreme svakog vozila, potrebno je zadovoljiti i sljedeće izazove:

- Privatnost u komunikaciji. Neophodno je zaštititi komunikaciju preko radio kanala i omogućiti anonimnost tokom razmjene podataka.
- Dostupnost tehnologije. Potrebno je ulagati u infrastrukturu i što više puteva i raskrsnica pokriti *V2I* uređajima.
- Pristupačnost korisnicima. Potrebno je obezbijediti isplativost vozila opremljenih ovom tehnologijom i

zadržati kompatibilnost tehnologije za sve proizvođače automobila.

3. V2X KOMUNIKACIONI STANDARDI

Federalna komisija za komunikacije Sjedinjenih Američkih Država i Evropska Unija opredijelile su *DSRC* (*Dedicated Short Range Communication*) frekvencijski opseg od oko 5.9GHz za komunikaciju *V2X* uređaja [5]. Komunikacija je bazirana na *IEEE 802.11p* [6] i *IEEE 1609* [7] standardima.

IEEE 802.11p standard je dizajniran da omogući razvoj *VANET* (*Vehicular Ad Hoc Networks*) u brzo promjenjivom okruženju. Standard predstavlja dopunu *IEEE 802.11-2007* standarda.

Fizički (*PHY*) sloj *IEEE 802.11p* standarda koristi istu obradu signala i specifikacije kao *OFDM PHY* upotrijebljen u *IEEE 802.11a* standardu. Ipak, da bi se ostvarila pouzdana konekcija u uslovima velike brzine, uvedene su manje modifikacije [8]. Umjesto upotrebe punog takta od 20MHz, upotrijebljen je polu takti mod sa širinom propusnog opsega od 10MHz. Time su udvostručeni parametri u vremenskom domenu i prepolovljena brzina prenosa. Na ovaj način je dobijen robustniji signal. Kao posljedica smanjenja širine propusnog opsega, umanjen je efekat Doplerovog pomjeraja. Širi zaštitni interval je redukovao inter-simbolsku interferencu uzrokovanu *multipath* propagacijom.

U cilju obezbjeđenja većeg komunikacionog dometa za automobilske scenarije, definisane su četiri *EIRP* (Effective Isotropic Radiated Power) klase, sve do 44.8 dBm (30W). Najveću vrijednost mogu koristiti jedino urgentna vozila. Tipična vrijednost, koja se smatra bezbjednom je 33dBm (2W), sa dometom 750 metara, što je i evropski standard. Ove vrijednosti su značajno veće nego *IEEE 802.11a EIRP* vrijednosti koje se koriste u SAD-u (800mW) i Evropi (500mW) [8].

Da bi se izbjegla interferencija sa drugom opremom, upotrijebljen je namjenski frekvencijski opseg. U SAD-u je opredijeljen opseg od 5.850-5.925GHz, širine 75MHz. Opseg je podijeljen u 7 kanala širine 10MHz. Na donjem kraju opsega ostavljeno je 5MHz sigurnosne margine. U Evropi je opredijeljen frekvencijski opseg od 5.850-5.905GHz. Opseg širine 30MHz podijeljen je u 3 kanala. Države članice su zadržale mogućnost rezervacije, na nacionalnom nivou, preostala četiri kanala američkog propusnog opsega.

U cilju pokrivanja ostalih protokolskih nivoa *IEEE 802.11p* je dopunjen drugim standardima, konkretno *IEEE 1609* setom standarda. Zajedno, *IEEE 802.11p* i *IEEE 1609* se nazivaju *WAVE* (*Wireless Access in Vehicular Environments*).

IEEE 1609 se sastoji od četiri dokumenta, od kojih je najinteresantniji *IEEE 1609.4*. *IEEE 1609.4* definiše multikanalne *WAVE* operacije, koje su implementirane na vrhu *MAC* (*Medium Access Control*) nivoa. U *OSI* (*Open System Interconnection*) modelu *MAC* je niži podnivo nivoa

linka. Viši podnivo link nivoa je LLC (Logical Link Control) (Slika 3).

Nivo 7	Aplikacija		IEEE 1609
Nivo 6	Prezentacija		
Nivo 5	Sesija		
Nivo 4	Transport		
Nivo 3	Mreža		
Nivo 2	Link	LLC	IEEE 801.11p
		MAC	
Nivo 1	Fizički		

Slika 3. IEEE 1609 i IEEE 802.11 u OSI modelu

MAC nivo obezbjeđuje interfejs i kontrolni mehanizam koji čvorovima mreže omogućava međusobnu razmjenu podataka. Njegova namjena je da komunikaciju između vozila učini bržom i efikasnijom. Stoga se glavna specifičnost standarda odnosi na MAC nivo. U V2X sistemima komunikacija se često odvija u brzo promjenjivom okruženju. Usljed toga, vrijeme koje se ima za razmjenu podataka može biti veoma kratko. Glavni zadatak modifikacije MAC mehanizma je smanjenje utroška vremena. U takvoj situaciji, uveden je tzv. "WAVE mod". Ovaj mod donosi novi BSS (Basic Service Set) tip, i to WAVE BSS (WBSS). WBSS omogućava stanicama da se povežu, kompletiraju WBSS process i razmjenjuju podatke, prijemom samo WAVE objave. Ne zahtijevaju se dalje interackije, kao što su autentifikacija i pridruživanje. Ovom promjenom unapređuje se efikasnost prenosa i prijema podataka u automobilskim ad-hoc mrežama (VANET) [9].

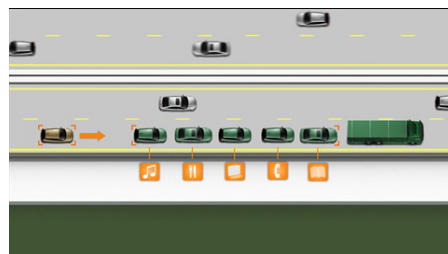
4. STEPEN RAZVOJA V2X SISTEMA I ZNAČAJNI NAUČNO-ISTRAŽIVAČKI PROJEKTI

Može se konstatovati da već, postoje automobili opremljeni nekim vidom V2X tehnologije. Takvi automobili su u stanju detektovati druga vozila oko sebe. Posjeduju senzore koji prate oznake na putu. Upozoravaju vozača ukoliko je prešao na suprotnu traku ili ukoliko je vozilo ispred njega naglo promjenilo brzinu. U stanju su da automatski održavaju vozilo u svojoj traci. Samostalno kontrolišući brzinu, brinu da vozač ne udari u vozilo ispred njega. Iako, serijska proizvodnja ovih vozila još uvijek nije počela, ona su komercijalno dostupna [10], [11].

Razvoj V2X tehnologije je značajno podstakut brojnim naučno istraživačkim projektima. U daljem tekstu će biti dat osvrt na nekoliko njih, koji se mogu svrstati u značajnije. To su SARTRE, sim^{TD}, PRESERVE i Safety Pilot.

SARTRE (Safe Road TRains for the Environment) [12] je evropski projekat koji fokusiran na razvoj autonomnih vozila. Finansiran je od strane Evropske komisije u okviru sedmog okvirnog programa (FP7 - 7th Research Framework Programme) u periodu od 2009 do 2012 godine. Cilj projekta je bio razvoj strategija i tehnologija za bezbjedniji i komforniji saobraćaj. Jedan od pravaca istraživanja projekta

je postojanje takozvanih autonomnih kolona vozila, koje samostalno prate vodeće vozilo. Autonomne kolone funkcionišu tako što sva vozila u koloni komuniciraju i međusobno se dogovaraju o kretanju. Automobili automatski podešavaju svoju brzinu i pravac kretanja, sve u zavisnosti od vodećeg vozila. U takvoj koloni, vozač može da se odmori i posveti pažnju nekim drugim stvarima, kao što su čitanje novina, telefoniranje, slušanje muzike i slično (Slika 4).



Slika 4. SARTRE projekat – vozila u koloni prate vodeće vozilo.

sim^{TD} (Sichere Intelligente Mobilität Testfeld Deutschland, engl. Safe and Intelligent Mobility Test Field Germany) [3] je FP7 projekat koji proučava uticaj V2X komunikacija na sigurnost u saobraćaju. Po broju učesnika, ovo je najveći projekat u Evropi koji se bavio ovom tematikom. U projektu je učestvovalo više stotina posebno opremljenih vozila. Za mjesto testiranja odabran je grad Frankfurt. Grad je izabran zbog najgušćeg saobraćaja u zemlji. Istraživanja su obuhvatila sve tipove vozila (motore, putnička vozila, teretna vozila i autobuse) na gradskim saobraćajnicama i seoskim putevima. Za prenos podataka između vozila upotrebljeno je više različitih protokola: IEEE 802.11p, IEEE 802.11b/g, GPRS, EDGE, UMTS i HSPA. Kao najbolje rješenje pokazao se IEEE 802.11p protokol zbog svoje niskog kašnjenja i manje potrošnje. Projekat je predložio standardizaciju protokola i algoritama za V2X komunikaciju.

PRESERVE projekat (Preparing Secure Vehicle-to-X Communication Systems) [13] je još jedan u nizu FP7 projekata koji finansira Evropska komisija, u period od početka januara 2011. do kraja decembra 2014. godine. Projekat istražuje načine i metode koje će dovesti do sigurnije V2X komunikacije. Glavni zadatak projekta je zaštita prenosa podataka od malicioznih napada i očuvanje anonimnosti učesnika. Oslanjajući se na rezultate sim^{TD} projekta, PRESERVE teži ostvarivanju cjelokupne zaštite V2X sistema, počev od senzora na vozilu, preko mreže u koju se povezuju automobili pa sve do same V2X komunikacije. Pored naučne saradnje sa mnogim univerzitetima, projekat ima i značajnu podršku od automobilske industrije. Naime, projektu se pridružilo nekoliko proizvođača automobila, kao što su: Audi, BMW, Daimler, Denso, Infineon i Volkswagen. Od projekta se očekuje:

- Standardizacija V2X arhitekture;
- Hardverska realizacija V2X sistema kao izolovani podsistem (komponenta) automobile;
- Implementacija V2X sistema u ASIC (Application Specific Integrated Circuit) čip niske cijene;

- Testiranja V2X sistema u realnim uslovima.

Safety Pilot [4] je zajednički projekat NHTSA (*National Highway Traffic Safety Administration*) i RITA (*Research and Inovative Technology Administration*). Projekat finansira vlada Sjedinjenih Američkih Država u okviru Ministarstva za transport (*U.S. Department of Transportation*). Sastoji se iz dvije faze. Prva faza u razvoju ovog projekta bila je prezentacija V2X tehnologije javnosti, zatim edukacija vozača i njihovo upoznavanje sa karakteristikama V2X sistema. Prva faza programa je obuhvatila oko 700 vozača u šest različitih država (Mičigen, Minesota, Florida, Virdžinija, Kalifornija i Teksas) i trajala godinu dana. Podaci koji su prikupljeni tokom ove faze programa su pokazali da većina vozača (njih 9 od 10) bi željela da ovu tehnologiju posjeduju u svom vozilu. Druga faza programa, koja još uvijek traje, proučava razvoj modela V2X sistema koji će biti predstavljen auto industriji i njegovo testiranje sa sličnim rješenjima pojedinih auto proizvođača. Po završetku ovog projekta, vlada će odlučiti o daljoj primjeni V2X tehnologije.

5. ZAKLJUČAK

U ovom radu je dat pregled V2X tehnologije od koje se se očekuje pruži ključni doprinos u obezbjeđenju sigurnijih i komfornijih uslova na putu. Ukazano je na brojna područja primjene kao i na ključne prednosti koje ta primjena donosi. Pokazano je da V2X tehnologije obuhvata dva osnovna vida komunikacije i to V2V (komunikacija između vozila) i V2I (komunikacija između vozila i infrastrukture).

U okviru rada dat je kratak osvrt na komunikacione standarde razvijene za potrebe V2X tehnologije. To su IEEE 802.11p i IEEE 1609, koji se zajedno nazivaju WAVE (*Wireless Access in Vehicular Enviroments*) standard.

Na kraju rada ukazano je na trenutni stepen razvoja V2X tehnologije i dat je pregled značajnijih naučnoistraživačkih projekata na ovom polju.

Istraživači i njihovi industrijski partneri, očekuju da vozila opremljena V2X tehnologijom budu spremna za serijsku proizvodnju za nekoliko godina. Najveće prepreke više nijesu tehnološke, već ekonomske, pravne i socijalne.

ZAHVALNICA

Rad je dijelom finansiran sredstvima projekta „Razvoj *Future Internet* rutera na programabilnim hardverskim platformama otvorenog koda“, sufinansiranim od strane Ministarstva za nauku Crne Gore i Ministarstva za informaciono društvo i telekomunikacije.

LITERATURA

[1] Bernd X. Weis, Astrid Sandweg, “*A New Paradigm for navigation Support based on V2V Communication*”, Springer Berlin Heidelberg, 2010.

[2] *simTD: Safe and Intelligent Mobility – Test Field Germany*, European Center for Information and Communication Technologies (EICT), June 2013.

[3] *Intelligent Transportation Systems – Safety Pilot*, U.S. Department of Transportation, Research and Inovative Technology Administration (RITA), July 2013.

[4] Dominik Lang, Thomas Stanger, Luigi del Re, “*Opportunities on Fuel Economy Utilizing V2V Based Drive Systems*”, SAE Technical Paper, January 2013.

[5] A. Widodo, T. Hasegawa, and S. Tsugawa, “*Vehicle fuel consumption and emission estimation in environment-adaptive driving with or without inter-vehicle communications*,” IEEE Intelligent Vehicles Symposium, 2000, Proceedings Paper, pp. 382–386.

[6] Patcharinee Tientrakool, Ya-Chi Ho, Nicolas F. Maxemchuk, “*Highway Capacity Benefits from Using Vehicle-to-Vehicle Communication and Sensors for Collision Avoidance*”, IEEE Vehicular Technology Conference, September 2011.

[7] Waleed Alasmay, Weihua Zhuang, “*Mobility Impact in IEEE 802.11p infrastructureless vehicular ad hoc networks*”, University of Waterloo, Waterloo, Canada, June 2010.

[8] Jyh-Hong Wen, Chien-Erh Weng, “*Performance Evaluation of IEEE 1609 WAVE for Vehicular Communications*”, International Journal of Vehicular Technology, 2013.

[9] IEEE WG802.11 - Wireless LAN Working Group, “802.11p-2010, <http://standards.ieee.org>, 2010.

[10] IEEE 1609 WG, “1609.4-2010 – IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE) – Multi-channel operation”, <http://standards.ieee.org>, 2010.

[11] *Crash Factors in Intersection-Related Crashes*, U.S. Department of Transportation, National Highway Traffic Safety Administration, September 2010.

[12] T. Nothdurft, and others, “*Google car: First fully autonomous test drives in urban traffic*”, 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), October 2011.

[13] Eric Chan, Peter Gilhead, Pavel Jelinek, Petr Krejčí, Tom Robinson, “*Cooperative control of SARTRE automated platoon vehicles*”, 19th ITS World Congress, Vienna, Austria, October 2012.

[14] Frank Kargl, “*PRESERVE Project*”, University of Twente, The Netherlands, 2013.

PROTOKOLI ZA REDUDANSU U IP MREŽAMA REDUNDANCY PROTOCOLS IN IP NETWORKS

Jelena Šuh, Vladimir Čulum, *Telekom Srbija a.d.*

Sadržaj: Projektovanje računarskih mreža je kompleksan proces, koji ima za cilj da obezbedi mrežnu infrastrukturu visoke dostupnosti i skalabilnosti kao podršku za različite servise i aplikacije. Primenom principa hijerarhije, modularnosti i redundanse hijerarhijski model mrežne infrastrukture ostvaruje ove ciljeve. Cilj ovog rada je da ukaže na važnost primene redundanse, ali i odgovarajućih protokola za default gateway redundansu (*First Hop Redundancy*). U ovom radu su opisane karakteristike HSRP i GLBP protokola i izvršena komparativna analiza performansi ovih protokola na primeru računarske mreže u laboratorijskom okruženju.

Abstract: Computer network design is a complex process, which aims to provide a high availability and scalability network infrastructure as a support for various services and applications. Hierarchical network design model achieves these goals by implementing hierarchy, modularity and redundancy principles. The aim of this paper is to highlight the importance of redundancy implementation and appropriate default gateway redundancy protocols (*First Hop Redundancy*). This paper describes the characteristics of the HSRP and GLBP protocols and gives comparative analysis of their performance in the example of a computer network in laboratory environment.

1. UVOD

Svaka računarska mreža ima zadatak da obezbedi savremene informaciono-komunikacione servise i bez obzira na namenu mora da ispuni visoke zahteve u pogledu pouzdanosti, skalabilnosti i bezbednosti. Ovi zahtevi se na najbolji način mogu ispuniti primenom hijerarhijskog modela mreže. [1]

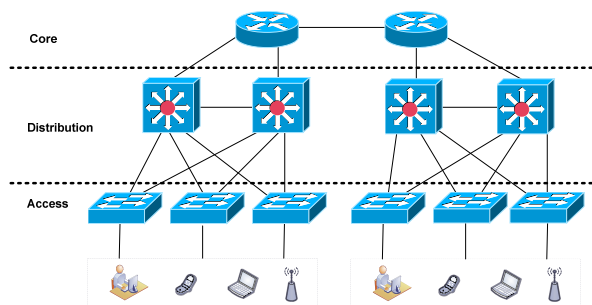
Principi hijerarhije i modularnosti su dominantni kod ovog modela, ali neophodna je i primena principa redundanse. Na ovaj način se u velikoj meri doprinosi fleksibilnosti i dostupnosti same mreže i realizovanih servisa. [2] Pored implementacije redundantnih uređaja i konekcija između uređaja, neophodna je i primena FHRP (*First Hop Redundancy Protocol*) protokola, koji imaju zadatak da obezbede redundansu *default gateway* uređaja u mreži. [3]

U ovom radu je izvršena komparativna analiza performansi dva FHRP protokola: HSRP (*Hot Standby Router Protocol*) i GLBP (*Gateway Load Balancing Protocol*). U drugom poglavlju su opisane karakteristike hijerarhijskog modela mreže i značaj primene redundanse u računarskim mrežama. U sledećem poglavlju su opisane karakteristike HSRP i GLBP protokola, dok su u četvrtom poglavlju prikazani rezultati analize performansi ova dva protokola na primeru konkretne IPv4 i IPv6 računarske mreže. U poslednjem poglavlju dat je zaključak i smernice za buduća istraživanja.

2. HIJERARHIJSKI MODEL MREŽE

Jedan od najvažnijih zadataka prilikom projektovanja računarske mreže je izbor mrežnog modela. Kod *flat* dizajna

mreže ne postoji hijerarhija i svaki uređaj ima jedan i jedinstven zadatak u mreži. Ovakve mreže su jednostavne za projektovanje, ali sa povećanjem broja uređaja javlja se problem sa skalabilnošću i održavanjem. Upravo iz ovih razloga danas je većina računarskih mreža zasnovana na hijerarhijskom modelu. [4] Ovaj model počiva na principima hijerarhije i modularnosti i kompleksne računarske mreže deli na slojeve.



Slika 1. Hijerarhijski model mreže

Access sloj predstavlja vezu korisničkih uređaja sa ostatkom mreže i interfejs ka korisnicima. Pored realizacije povezivanja na mrežu, ovaj sloj vrši i kontrolu pristupa mreži.

Distribution sloj predstavlja interfejs između *access* i *core* sloja i ima zadatak da obezbedi agregaciju podataka sa *access* sloja pre prosleđivanja *core* sloju, koji vrši slanje podataka do krajnjeg odredišta. Na ovom sloju se vrši kontrola toka saobraćaja. Redundansa se ostvaruje multipliciranjem uređaja, ali i realizacijom višestrukih konekcija između uređaja. Još jedna važna karakteristika

ovog sloja je obezbeđivanje *default gateway* redundanse, pa se iz tog razloga primenjuju FHRP protokoli. [5]

Core sloj predstavlja okosnicu mreže i obezbeđuje konektivnost između uređaja *distribution* sloja. Upravo iz tog razloga, na ovom sloju su visoka dostupnost i redundansa od velikog značaja, kao i primena uređaja (ruteri i svičevi) visokih performansi.

Kako je redundansa princip koji neizostavno mora da se primeni u svakoj računarskoj mreži, u daljem tekstu će biti predstavljeni FHRP protokoli: HSRP i GLBP.

3. FHRP PROTOKOLI

FHRP protokoli su kreirani sa ciljem da obezbede redundansu *default gateway*-a i transparentan *failover* u slučaju otkaza uređaja koji ima ulogu *default gateway*-a u mreži. Ovi protokoli nalaze primenu u *shared* mrežama, kao što je *Ethernet*. Suština funkcionisanja ovih protokola se ogleda u sledećem [3]:

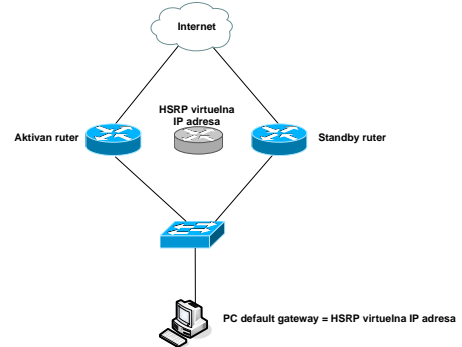
- U mreži postoje dva ili više *default gateway* uređaja.
- Na krajnjim korisničkim uređajima se definiše jedna *default gateway* adresa.
- FHRP protokolom je definisano koji uređaj ili uređaji vrše prosleđivanje saobraćaja.
- U slučaju otkaza *default gateway* uređaja, automatski se vrši preusmeravanje saobraćaja na redundantni *default gateway* uređaj, pri čemu je uticaj na korisnički saobraćaj minimalan (zavisi od vremena konvergencije FHRP protokola). U ovom slučaju, na korisničkim uređajima nije potrebna nikakva izmena *default gateway* adrese.

3.1. HSRP PROTOKOL

HSRP je Cisco *proprietary* protokol koji obezbeđuje redundansu i mehanizam za *failover* u slučaju nedostupnosti *default gateway*-a u mreži. Ukoliko bi iz nekog razloga *default gateway* postao nedostupan, bez ovog protokola, uređaji u mreži ne bi mogli da komuniciraju van lokalne mreže, čak i u slučaju postojanja drugog rutera koji bi mogao da bude *default gateway*. [6]

HSRP definiše grupu *standby* rutera i jedan aktivan ruter. Suštinu ovog protokola predstavlja razmena informacija o IP i MAC adresama između redundantnih *gateway*-a. Informacije o virtuelnim IP i MAC adresama se razmenjuju između rutera koji pripadaju istoj HSRP grupi. Virtuelni ruter predstavlja MAC i IP adresa koja se konfigurira na krajnjim uređajima kao *default gateway*. U okviru HSRP grupe postoji samo jedan aktivan ruter i on je taj koji prima saobraćaj namenjen virtuelnom ruteru i vrši prosleđivanje paketa. Uloga *standby* rutera je da aktivno prati status HSRP grupe i automatski preuzme ulogu aktivnog rutera ukoliko on iz nekog razloga postane nedostupan. U grupi mogu da postoje i drugi ruteri koji nemaju ulogu ni aktivnog ni *standby* rutera (*listen* stanje). [7]

Kada aktivan ruter postane nedostupan, *standby* ruter preuzima njegovu ulogu, a ostali ruteri učestvuju u izboru za novi *standby* ruter. Pošto novi aktivan ruter preuzima IP i MAC adresu virtuelnog rutera, krajnji korisnici ne vide prekid servisa. Krajnji uređaji šalju pakete virtuelnom ruteru, a novi aktivan ruter ih fizički prosleđuje ka odredištu.



Slika 2. HSRP protokol

Kod HSRP protokola postoji mogućnost konfiguracije dodatnih parametara kao što su prioritet, *preempt* i *tracking*. Konfiguracija prioriteta omogućava administratoru mreže da definiše redosled po kome će ruteri preuzimati ulogu aktivnog rutera, dok opcija *preempt* omogućava ruteru da povrti svoju ulogu posle vraćanja u operativno stanje, čak i u slučaju da postoji aktivan ruter u mreži. Pored ovoga, postoji mogućnost praćenja statusa interfejsa (*tracking*), pri čemu se u slučaju „pada” ovakvog interfejsa dinamički smanjuje HSRP prioritet. Promenom navedenih parametara može se u velikoj meri izvršiti optimizacija HSRP protokola za konkretnu računarsku mrežu, čime se doprinosi poboljšanju performansi i dostupnosti. [8]

HSRP protokol nema mogućnost dinamičke realizacije *load-balancing* funkcionalnosti. Međutim, postoji način da se konfiguracijom prioriteta i definicijom HSRP grupa za različite VLAN-ove i različite rutere postigne određeni stepen *load-balancing*-a. [9]

Postoje dve verzije HSRP protokola, pri čemu je u IPv6 mrežama neophodno implementirati verziju 2. Sam princip funkcionisanja HSRP protokola je isti u IPv4 i IPv6 mrežama, a razlika postoji u *multicast* adresi koja se koristi, UDP portu i MAC adresama.

3.2. GLBP PROTOKOL

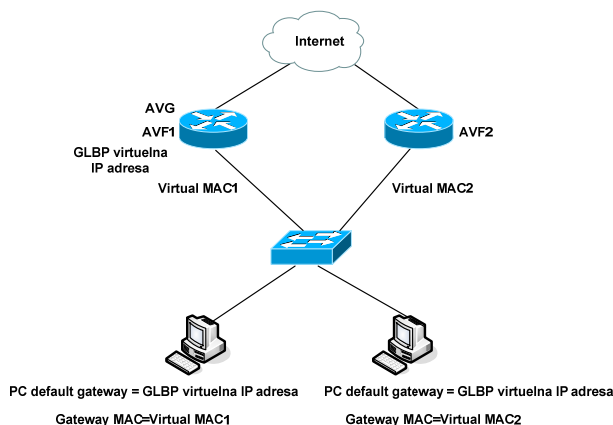
GLBP je drugi Cisco *proprietary* FHRP protokol. Ovaj protokol ima dosta sličnosti sa HSRP protokolom, ali postoji jedna velika razlika, koja se ogleda u funkcionalnosti dinamičkog *load-balancing*-a. Kod GLBP protokola svi ruteri u GLBP grupi su aktivni i vrše prosleđivanje saobraćaja. [10]

GLBP protokol definiše GLBP grupu *default gateway* rutera, u kojoj postoji jedan aktivan virtuelni *gateway* (AVG), a svi ostali predstavljaju rezervu. AVG je zadužen za

dodelu virtuelnih MAC adresa ostalim članovima GLBP grupe, koji se nazivaju AVF ruteri (*Active Virtual Forwarder*). Svaki AVF prosleđuje pakete poslate sa određenom virtuelnom MAC adresom koja je njemu dodeljena od strane AVG rutera.

Default gateway redudansa se kod GLBP protokola ostvaruje na isti način kao i kod HSRP protokola. Jedan *gateway* je AVG, drugi je *standby* virtualni *gateway*, a ostali su u *listen* stanju. AVF redudansa se takođe ostvaruje na sličan način. Ako AVF postane nedostupan, jedan od preostalih AVF rutera u *listen* stanju preuzima ulogu prosleđivanja paketa za tu virtuelnu MAC adresu. [7]

GLBP daje mogućnost izbora uloge koju će *default gateway* imati definisanjem prioriteta. Takođe, postoji i opcija *preempt*, koja omogućava *backup* virtuelnom *gateway*-u da postane AVG, ne samo u slučaju da AVG postane nedostupan, već i u slučaju da ima definisan viši prioritet od njega.



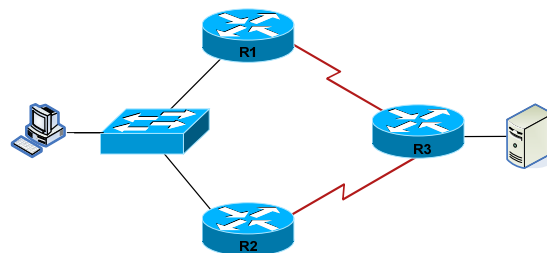
Slika 3. GLBP protokol

Još jedna važna funkcionalnost, koju GLBP podržava, jeste primena težinskih faktora (*weighting*). Na ovaj način je moguće definisati količinu opterećenja svakog rutera u GLBP grupi. Opcija praćenja statusa interfejsa (*tracking*) se takođe može povezati sa težinskim faktorima, pa tako „pad” nekog interfejsa može automatski uticati na promenu težinskog faktora za određeni ruter.

GLBP protokol se može primeniti i u IPv4 i u IPv6 mrežama. Način funkcionisanja je isti, a razlika postoji u *multicast* adresi koja se koristi.

4. ANALIZA PERFORMANSI HSRP I GLBP PROTOKOLA

Analiza performansi HSRP i GLBP protokola izvršena je u laboratorijskom okruženju korišćenjem topologije prikazane na slici 4. Implementirani su Cisco 2801 ruteri i Cisco 2960 svič. Međuruterske konekcije realizovane su preko serijskih interfejsa, dok su veze sa svičem realizovane korišćenjem *FastEthernet* interfejsa. Kako je cilj ispitati performanse FHRP protokola, na ruterima je primenjena osnovna konfiguracija: naziv uređaja, adresiranje i OSPF protokol rutiranja.



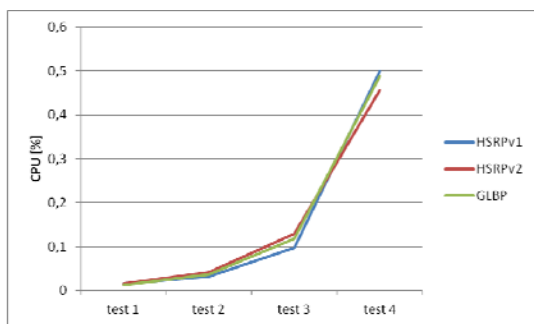
Slika 4. Topologija mreže u laboratorijskom okruženju

Na ruterima R1 i R2 je prvo konfigurisan HSRP, a zatim GLBP protokol. Analiza performansi je vršena posebno za svaki FHRP protokol, prvo u IPv4, a zatim i u IPv6 mrežnom okruženju. Na ruteru R1 je definisan viši prioritet, tako da je on imao primarnu ulogu u prosleđivanju paketa između računara i servera. Parametar, koji je svakako najvažniji, kada je reč o FHRP protokolima je vreme konvergencije. Gašenjem interfejsa na ruteru R1 simuliran je prekid linka između rutera R1 i sviča, čime je inicirano preusmeravanje saobraćaja na drugu putanju preko rutera R2. Na osnovu analize ispisa odgovarajućih *show* i *debug* komandi utvrđeno je vreme konvergencije za različite vrednosti *hello* i *hold timer*-a, koji su konfigurisani na ruterima R1 i R2. Merenja su ponavljana u seriji od po 10 puta, a dobijeni rezultati su prikazani u tabeli 1.

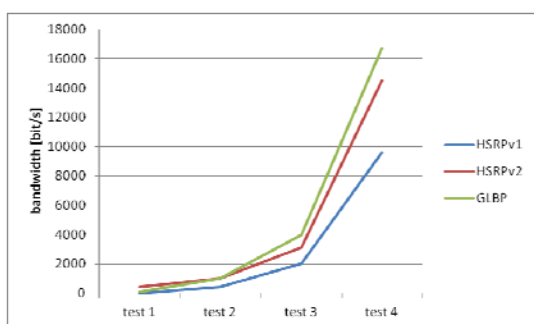
Zatim je analiziran uticaj FHRP protokola na performanse mreže. Posmatran je uticaj HSRP i GLBP protokola na zauzetost propusnog opsega i opterećenje procesora. Prilikom ove analize vršena je promena vrednosti *hello* i *hold timer*-a, kao u prethodnom slučaju, a na slikama 5. i 6. su prikazani dobijeni rezultati u slučaju IPv4 mreže.

Tabela 1. Vreme konvergencije HSRP i GLBP protokola

Vreme konvergencije [sec]	Timer		IPv4			IPv6	
	Hello[sec]	Hold[sec]	HSRPv1	HSRPv2	GLBP	HSRPv2	GLBP
Test 1	3	10	8,7542	8,5546	8,7212	8,5985	8,5859
Test 2	1	3	2,3469	2,3572	2,3254	2,5386	2,5302
Test 3	0,250	0,750	0,6364	0,6387	0,6329	0,6124	0,6014
Test 4	0,050	0,150	0,1253	0,1236	0,1218	0,1352	0,1314

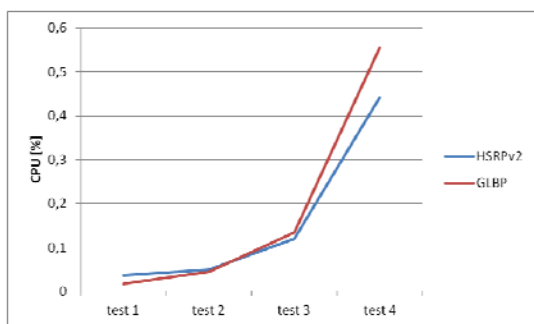


Slika 5. Uticaj FHRP protokola na zauzetost procesora u IPv4 mreži

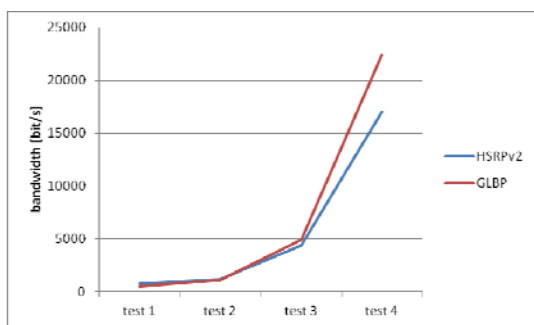


Slika 6. Uticaj FHRP protokola na zauzetost propusnog opsega u IPv4 mreži

Ista metodologija je primenjena kako bi se utvrdio uticaj HSRP i GLBP protokola na performanse u IPv6 mreži, a rezultati su prikazani na slikama 7. i 8.



Slika 7. Uticaj FHRP protokola na zauzetost procesora u IPv6 mreži



Slika 8. Uticaj FHRP protokola na zauzetost propusnog opsega u IPv6 mreži

5. ZAKLJUČAK

Cilj ovog rada je da ukaže na važnost i kompleksnost izbora FHRP protokola u računarskim mrežama. Na osnovu izvršene analize performansi može se zaključiti da GLBP protokol ima nešto manje vreme konvergencije u odnosu na HSRP protokol. Međutim, analiza uticaja FHRP protokola na performanse mreže (zauzetost procesora i propusnog opsega) ukazuje na činjenicu da GLBP protokol ima veći uticaj na performanse mreže u odnosu na HSRP protokol. Ipak, treba imati u vidu da oba protokola u velikoj meri doprinose dostupnosti servisa realizovanih u mreži i da je njihova primena neophodna u modernim računarskim mrežama, a uticaj na zauzetost procesora i propusni opseg minimalan. Zajedničko za oba protokola je da se ne mogu primeniti u *multi-vendor* mrežama, pa bi zato buduća istraživanja trebalo usmeriti na analizu performansi VRRP protokola, koji predstavlja FHRP protokol otvorenog standarda. Pored ovoga, trebalo bi ispitati mogućnosti optimizacije mreže konfiguracijom dodatnih funkcionalnosti, kao što su *preempt*, *tracking*, *load-balancing* itd.

LITERATURA

- [1] W. Mogeji, X. Wang / Z. Liu, „A new complex network model with hierarchical and modular structures,“ *Chinese Journal of Physics* 48, pp. 805-813, 2010.
- [2] A. Guimaraes, H. Oliveira, R. Barros / P. Maciel, „Availability analysis of redundant computer networks: A strategy based on reliability importance,“ u *IEEE 3rd International Conference on Communication Software and Networks (ICCSN)*, Xi'an, 2011.
- [3] W. Odom, Cisco CCNA Routing and Switching ICND2 200-101 Official CertGuide, Cisco Press, 2012.
- [4] D. Teare, Designing for Cisco Internetwork Solutions (DESGN), Cisco Press, 2007.
- [5] K. T. Hutton, M. D. Schofield / D. Teare, Designing Cisco Network Service Architectures (ARCH), Cisco Press, 2008.
- [6] T. Li, B. Cole, P. Morton / D. Li, „RFC 2281: Cisco Hot Standby Router Protocol (HSRP),“ 1998. [Na mreži]. Available: <http://www.ietf.org/rfc/rfc2281.txt/>.
- [7] D. Hucaby, CCNP BCMSN Official Exam Certification Guide, Cisco Press, 2007.
- [8] A. K. Singh / A. Kothari, „HSRP (Hot Stand by Routing Protocol) Reliability Issues Over the Internet Service Provider's Network,“ *Oriental Journal of Computer Science & Technology*, t. 4, br. 2, pp. 399-404, 2011.
- [9] T. Deng-long, F. Yong / L. Qian, „Analysis and Improvement of Virtual Fault-tolerant Routing Protocol,“ *Communications Technology*, t. 4, 2011.
- [10] P. Dubey, S. Sharma / A. Sachdev, „Review of First Hop Redundancy Protocol and Their Functionalities,“ *International Journal of Engineering Trends and Technology (IJETT)*, t. 4, br. 5, 2013.

MOBILNA APLIKACIJA CRNOGORSKOG TELEKOMA CRNOGORSKI TELEKOM MOBILE APPLICATION

Pero Bogojević, Jasna Mirković, *Crnogorski Telekom*

Sadržaj: *Mobilna aplikacija Crnogorskog Telekoma predstavlja nov način komunikacije sa korisnicima. Ona je odlična podrška korisnicima za sve što im je potrebno: pregled potrošnje i računa, prijavu problema, aktivacije servisa i slično. Realizovana internim snagama i iskustvom, odlikuje se najnovijim tehnologijama, visokim sigurnosnim standardima i sjajnim idejama.*

Abstract: *Mobile application from Crnogorski Telekom stands for new way of communication with customers. It represents excellent support for customers for their every need: usage balance and bills overview, problem reporting, services activation etc. Made fully in house with own knowledge and experience it combines newest technologies, high security standards and great ideas.*

1. UVOD

Mobilna aplikacija je prva ovakvog tipa u ponudi Crnogorskog Telekoma i predviđena je da podrži sve smart phone i tablet platforme na tržištu.

Cilj projekta je da se korisnicima na jednom mjestu pruži širok spektar usluga i funkcionalnosti koje su im uvijek pri ruci. Aplikacija će pojednostaviti i unaprijediti komunikaciju sa korisnicima, smanjiti broj nepotrebnih odlazaka u T centre kao i smanjiti broj poziva ka operaterima zbog informacija, aktivacija dodatnih usluga i prijave problema. Aplikacija je predviđena je za smart phone i tablet uređaje i u prvoj varijanti biće lansirana za Android i IOS operativne sisteme. Zbog tehnologija koje su korišćene u izradi aplikacije lako se može lansirati i na ostalim platformama smart phone i tablet uređaja, kao: Windows Mobile, Black Berry i druge.

Najvažnije osobine su jednostavan i čist dizajn, po uzoru na svjetske trendove i vodeće aplikacije na tim poljima, lakoća upotrebe, visoki sigurnosni standardi, mogućnost upotrebe od strane bilo kog korisnika smart phone ili tablet uređaja i druge brojne korisne opcije.

2. MOBILNA APLIKACIJA

Mobilna aplikacija Crnogorskog Telekoma, u prvoj verziji, nudi se svim korisnicima smart phone ili tablet uređaja sa pristupom internetu i Google play ili iTunes online prodavnicama. Korisnici aplikacije mogu koristiti servise mobilne ili fiksne telefonije bilo kog operatera, ali mogu biti i oni koji nisu korisnici usluga niti jednog od operatera u Crnoj Gori. Aplikaciju je potrebno preuzeti sa online prodavnica i instalirati na svom uređaju kako bi se pristupilo korišćenju iste. Poslije preuzimanja i instalacije aplikacije, potrebno je da korisnik kreira svoj nalog. Nalog se kreira jednostavnim popunjavanjem par osnovnih polja, kao što su: korisničko ime, lozinka, mail adresa i matični broj građana. Sva polja su obavezna a korisnik takođe treba da se složi sa uslovima korišćenja aplikacije kako bi nalog bio kreiran. Poslije uspješnog prvog koraka, korisniku se šalje mail na unijetu

mail adresu sa verifikacionim kodom za svoj nalog. Važno je napomenuti da je, sa funkcionalnog aspekta, ovo prvi stepen zaštite, čime korisnik potvrđuje da je on vlasnik unijete mail adrese. Pošto je nalog verifikovan kodom koji je došao putem maila, korisnik se može prijaviti na aplikaciju.

U trenutku kreiranja korisničkog naloga, matični broj građana koristi se u svrhe obrade podataka pa tako aplikacija u pozadini provjerava da li je vlasnik naloga koji se kreira i korisnik Telekom usluga. Na ovaj način se radi diferencijacija i određivanje tipa korisnika, koji može biti Gost ili Korisnik. Ukoliko vlasnik naloga nije korisnik Telekom usluga, njegov tip korisnika postavlja se na Gost. Ukoliko jeste, relevantni podaci se automatski preuzimaju za korisnika (ime, prezime, adresa i drugo) i njegov tip korisnika je Korisnik. Do ovog trenutka u vremenu, izgled i funkcionalnosti aplikacije su iste za sve korisnike.

Poslije prijavljivanja na aplikaciju Gost korisnik vidi skraćenu verziju aplikacije, tj. onu za čiju upotrebu nisu potrebni specifični servisi Telekoma. Ukoliko je prijavljeni korisnik tipa Korisnik, to znači da ima neki od Telekomovih servisa i da potencijalno može pridružiti svoj broj (brojeve) telefona svom nalogu u mobilnoj aplikaciji. U procesu pridruživanja postpaid ili prepaid brojeva telefona, korisniku će na unijeti broj biti poslat verifikacioni kod putem SMS-a. Kao i ranije i ovdje je bitno napomenuti da je ovo drugi stepen zaštite i provjere ispravnosti korisničkih podataka. U slučaju pridruživanja fiksnog broja telefona svom nalogu, korisnik treba da pozove Telekom call centar, 1500, sa tog fiksnog broja, i odabere opciju za aktivaciju broja. Poslije ovog koraka korisniku su dostupne sve funkcionalnosti aplikacije.

Sa stanovišta tehničke realizacije, mobilna aplikacija je rađena po principima troslojne arhitekture, čime su fizički odvojeni "front-end" ka korisniku i "back-end" sistemski dio. Posebno mjesto zauzima srednji sloj koji je odgovoran za komunikaciju ova dva navedena dijela. Ovaj princip arhitekture, poznat i kao SOA (service oriented architecture) kombinovan je sa standardima i smjernicama DT-a (Deutsche

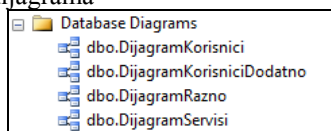
Telekom), čiji je Telekom dio. Ovakav način projektovanja omogućava pažljivo planiranje i dizajniranje grafičkog interfejsa orijentisanog ka korisniku, uz naglasak na jednostavnost, brzinu odziva, raspored kontrola i slično. U svjetlu tehnologija koje su na raspolaganju nije se išlo na „native“ aplikaciju već na web varijantu. Naime, „native“ aplikacije su aplikacije pisane za specifičan operativni sistem, kao IOS, Android, Windows mobile i druge. Web aplikacije, ili njihov prezentacioni dio, pisane su u HTML jeziku i mogu se učitati u bilo kom web pretraživaču nezavisno od operativnog sistema. Oba principa programiranja imaju prednosti i mana, pa poslije izmjerenih benefita odlučeno je ići sa web varijantom.

Aplikacija, tj. „front-end“, je rađena u HTML5 jeziku, što je poslednja verzija HTML-a za web i donosi značajne razlike i unapređenja u odnosu na raniji HTML. Ovakva aplikacija je optimizovana i zapakovana posebnim alatima za IOS, Android i ostale operative sisteme. Ovo je jedana od glavnih prednosti ovog pristupa, jer je ovim putem rađen samo jedan kod, jedan kod se održava, unapređuje i proširuje i po potrebi pakuje za specifične sisteme. Kod „native“ pristupa morali bi se razvijati posebni kodovi za IOS i Android, pa redom i ostale operative sisteme po potrebi. Ovo znači da je održavanje lakše, izmjene su lakše, smanjena je šansa za propustima i previdima. Ovakvim pristupom, čak, moguće je raditi izmjene u aplikaciji u dijelu sadržaja, fonta, tekstova i drugih stvari koje značajnije ne mijenjaju strukturu aplikacije a bez njenog ponovnog postavljanja na online prodavnice u vidu update-a. Ovo je veliki dobitak i omogućava dnevno praćenje i korekcije u sadržaju, nove ponude ka korisnicima, informacije i slično.

Nasuprot korisničkom interfejsu, pozadina aplikacije, „back-end“, je temeljena na skupu funkcija i procedura baznog nivoa. Povezani sistemi su sa jedne strane biling sistemi (fiksne, postpaid mobilne i prepaid mobilne) a sa druge strane lokalna baza same mobilne aplikacije za korisničke podatke, istoriju pristupa i akcija, tabele šifarnike i drugo. Sve je praćeno dnevnim procedurama i zakazanim aplikacijama koje rade presipanja i kontrolu podataka između biling sistema i lokalne baze aplikacije.

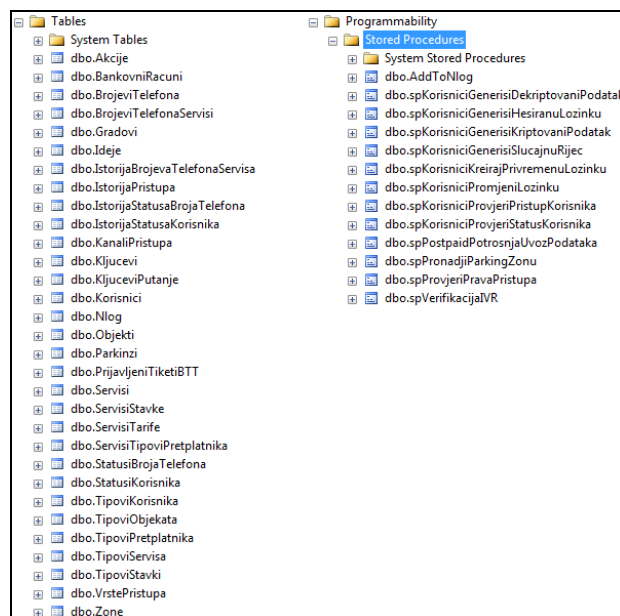
Lokalna baza mobilne aplikacije je pod Microsoft SQL Serverom 2008, a sastoji se od:

- 4 bazna dijagrama



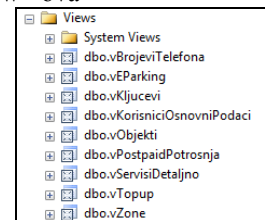
Slika1. Bazni dijagrami

- Tabele i procedure u bazi



Slika 2. Tabele i procedure u bazi

- Baznih „view“-ova



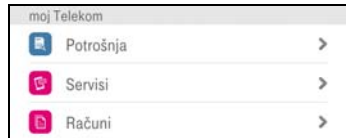
Slika 3. Bazni „view“-ovi

Na kraju srednji sloj je skup web servisa koji na najbolji način povezuje dva kraja aplikacije, orkestrira komunikacijom i obezbjeđuje transakcionu atomičnost. Dio gdje srednji sloj mobile aplikacije Telekoma ide i malo dalje je sam način organizacije ovog sloja. Naime, osim što predstavlja skup web servisa, taj sloj drži i određenu logiku u načinu planiranja, pozicioniranja i implementacije samih servisa. Radi se o standardima DT grupe, koji su do detalja razrađeni i isplanirani da podrže sve procese unutar kompanije, od bazčnih do mnogih kompleksnih. Ovakav pristup omogućava da se implemetirani web servisi koriste od strane bilo kojeg organizacije unutar kompanije, bilo kojeg treće strane kojoj bi trebali procesu u kompaniji ili samih aplikacija koje se razvijaju za kompaniju.

Ovaj koncept po kojem je rađeno zove se Group Domain Model ili skraćeno GDM. GDM predstavlja klaster zatvorenih, stabilnih biznis funkcija koji omogućava razdvajanje biznisa i IT-ja. Logika je ta da je sam domen podijeljen na funkcionalne komponente, a u njima se nalaze agregirani biznis entiteti. Smisao izgradnje srednjeg sloja na ovom principu je pažljivo određivanje svake pojedinačne funkcije, svakog web servisa unutar pravilnog biznis entiteta i funkcionalne komponente. Na taj način pravi se organizacija koja ima strukturu i smisao i lako je raspoloživa trećim stranama, pa čak bile one i biznis. Nije potrebna IT vještina da bi se pronašla funkcionalnost potrebna za neki biznis proces, već je dovoljan sam opis biznis entiteta i komponenti.

Pošto su tehnologije za izradu u Telekomu Microsoft Visual Studio .NET (VB, C#) upareni sa IIS serverom, to znači da je po hijerarhiji potrebno kreirati “solution”, unutar njega odgovarajući “project”, ispod njih web servise i web metode. Koristeći principe GDM-a ideja je bila pravilno upotrebiti taj koncept i uklopiti ga u hijerarhiju tehnologija za izradu. Posmatrano kroz prizmu funkcionalnosti aplikacije iste su podijeljene u sekcije: *moj Telekom*, *opcije*, *opšte* i izdvojeni *donji meni*. Kao što je rečeno na početku, funkcionalnosti aplikacije posmatraju se dinamički i zavise od korisnika koji koristi istu, njegovog profila i servisa koje ima kod Crnogorskog Telekoma.

Moj Telekom sekcija je isključivo rezervisana i vidljiva korisnicima Telekom usluga. Nudi specifičnosti vezane za Telekom servise kao što su:



Slika 4. Sekcija moj Telekom

Potrošnja, mjesto gdje korisnik u svakom trenutku može da vidi ne samo svoju trenutnu potrošnju i stanje na računu već i detaljne stavke te potrošnje. Radi lakše preglednosti potrošnja je organizovana u cjeline i to prema tipu saobraćaja koji je napravljen. Za prepaid može se vidjeti trenutno stanje na računu i datum poslednje dopune računa, sa iznosom. Po GDM-u potrošnja je organizovana kao:

Solution	Billing
	Project Billing Account Management
	Web Service Consumption
	Method GetCurrentPostpaidConsumption
	Method GetCurrentPrepaidConsumption
	Method GetCurrentFixedConsumption

Slika 5. Potrošnja po GDM-u

Servisi je stranica koja korisniku, zavisno od usluga koje ima kod Telekoma, nudi dodatne opcije za postpaid i prepaid, a dodatni Extra TV paketi za fiksnu. Za svaku od dodatnih opcija može da se vidi status i odradi aktivacija ili deaktivacija. Po GDM-u organizacije je:

Solution	Customer Relationship Management
	Project Customer Order Management
	Web Service FixedPackages
	Method GetServiceStatus
	Method ManageService
	Web Service PostpaidServices
	Method GetServiceStatus
	Method ActivateService
	Method DeactivateService
	Web Service PrepaidServices
	Method GetServiceStatus
	Method ActivateService

Slika 6. Servisi po GDM-u

Računi stavka daje korisniku uvid u svoje plaćene i neplaćene račune. Osim iznosa mogu da se vide detaljne stavke pojedinog računa. Isti je moguće poslati na mail adresu. Na istom mjestu aplikacija automatski detektuje status e-bill opcije u sistemu i adekvatno tome nudi korisniku aktivaciju ili deaktivaciju elektronskog računa. Veliki korak naprijed je plaćanje računa koje je moguće elektronskim putem, a podržani su svi tipovi kreditnih i debitnih kartica. Organizovano po GDM-u kao:

Solution	Billing
	Project Bill Calculation & Presentment
	Web Service Bills
	Method GetFixedAvailableBills
	Method GetFixedBill
	Method GetPostpaidAvailableBills
	Method GetPostpaidBill

Slika 7. Računi po GDM-u

Sekcija opcije je vidljiva svim korisnicima, ali varira u opcijama koje su moguće za regularnog i gost korisnika i ima:



Slika 8. Sekcija opcije

Podešavanja, putem kojih korisnik može da vidi informacije o svom profilu (ime, prezime, adresu i grad), da izmijeni lozinku i mail adresu i doda registarske tablice koje želi da koristi u eParking opciji. Uklopljeno po GDM-u kao:

Solution	Unified Communication
	Project Identity Management
	Web Service ManageUser
	Method RegisterUser
	Method VerifyUser
	Method ManageUserAccess
	Method GetUserData
	Method RetrieveUsername
	Method RetrievePassword
	Method ChangePassword
	Method ChangeEmailAddress

Slika 9. Podešavanja po GDM-u

Moji brojevi opcija omogućava da korisnik svom nalogu pridruži brojeve telefona koje ima kod Telekoma. Svaki broj se u prvom koraku inicira za pridruživanje a kasnije verifikuje. GDM:

Solution	Customer Relationship Management
	Project Customer Management
	Web Service ManageSubscription
	Method GetSubscribersPhoneNumbers
	Method AssignPhoneNumberToSubscriber
	Method VerifySubscribersPhoneNumber
	Method VerifySubscribersPhoneNumberByIVR
	Method RemoveSubscribersPhoneNumber
	Method CheckSubscriberType

Slika 10. Moji brojevi po GDM-u

Podrška je najbrži i najlakši način da korisnik prijavi problem u radu servisa. Jednostavnim odabirom servisa na kojem ima problem i tipa problema, uz kontakt telefon i opis problema će biti prijavljen a u pozadini mu dodijeljen najviši prioritet u rješavanju. GDM:

Solution	Customer Relationship Management
	Project Customer Incident Management
	Web Service Support
	Method CreateTicket
	Method GetIssueTypes
	Method GetTicketsPerUser

Slika 11. Podrška po GDM-u

Posebne ponude za korisnika određuju se po broju telefona koji korisnik ima a preračunavaju se u pozadini po definisanom algoritmu.

Solution	Customer Relationship Management		
Project	Offer & Campaign Management		
Web Service	Offering		
Method	GetSpecialOffers		

Slika 12. Posebne ponude po GDM-u

Sekcija opšte na raspolaganju je svim korisnicima:



Slika 13. Sekcija opšte

Kako do nas je interaktivna mapa koja se u trenutku startovanja pozicionira na trenutnu GPS lokaciju na kojoj se korisnik nalazi. Na mapi je moguće vidjeti lokacije i detalje Telekom šopova, Telekom partner šopova i Hot Spot-ova za pristup bežičnom internetu.

Solution	Enterprise Support		
Project	Content Management		
Web Service	Map		
Method	GetMapObjects		

Slika 14. Kako do nas po GDM-u

Telekom sajt je link do zvaničnog Telekom sajta, u ovom slučaju do njegove mobilne verzije.

Ostale Telekom aplikacije stranica daje korisniku prikaz ostalih aplikacija za mobilne uređaje koje nudi Telekom, kao što su: Extra TV mobile, HBO GO, pick'n'play.

Uslovi korišćenja su isti oni koje je korisnik prihvatio prilikom registracije i dostupni su korisniku u svakom trenutku.

Izdvojeni donji meni je sekcija koja je opet dostupna samo korisnicima Telekom usluga i izdvojena je u posebnoj meniju što zbog specifičnosti to i zbog učestalosti korišćenja. Donji meni je uvijek vidljiv a u njemu su:



Slika 15. Donji meni

Dopuna je mjesto sa kojeg korisnik može da pokrene proces dopune prepaid broja telefona i da vidi istoriju ranijih dopuna. Korisnik može da dopuni neki od svojih prepaid brojeva ili da unese neki drugi prepaid broj. Po GDM-u:

Solution	Billing		
Project	Billing Account Management		
Web Service	Topup		
Method	AssignPrepaidSubscribersPhoneNumber		
Method	RemovePrepaidSubscribersPhoneNumber		
Method	GetTopupPhoneNumbers		
Method	GetAllowedTopupAmounts		
Method	ExecuteTopup		

Slika 16. Dopuna po GDM-u

eParking spaja korisnu uslugu sa mogućnostima koje jedna mobilna aplikacija nudi. Moguće je plaćanje parkinga na teritoriji Podgorice i Herceg Novog. Korisnik može da izabere sa kojeg od svojih brojeva želi da plati parking. Velika prednost je ta što se prilikom startovanja eParkinga pokupe GPS koordinate korisnika i take se automatski odredi

zona u kojoj se nalazi. Na korisniku je da provjeri podešavanja i plati. Po GDM-u:

Solution	Customer Relationship Management		
Project	Customer Order Management		
Web Service	SMSParking		
Method	GetRegistrationPlates		
Method	AssignRegistrationPlateNumber		
Method	RemoveRegistrationPlateNumber		
Method	ExecuteSmsParking		
Method	GetServiceHistory		

Slika 17. eParking po GDM-u

Ritam klub je dio standardne ponude Telekomu za lojalne korisnike i tu je moguće vidjeti status i broj poena u Ritam klubu. Osim toga mogu se vidjeti nagrade koje su na raspolaganju za mobilne i fiksne korisnike kao i istorija zamjene ranijih nagrada. Po GDM-u:

Solution	Customer Relationship Management		
Project	Offer & Campaign Management		
Web Service	Loyalty		
Method	GetServiceStatus		

Slika 18. Ritam klub po GDM-u

Ideje je način da korisnik doprinese budućem razvoju ne samo mobilne aplikacije već bilo kojeg servisa ili oblasti u Telekomu tako što će dati ime i opis svojoj ideji i poslati je. Po GDM-u:

Solution	Enterprise Support		
Project	Enterprise Risk Management		
Web Service	CustomersIdeas		
Method	CreateIdea		

Slika 19. Ideje po GDM-u

Social je link do Telekom facebook strane.

3. ZAKLJUČAK

Spoj modernih tehnologija i klasičnih pozadinskih sistema može donijeti puno na polju inovacija, ideja i primjenljivosti mobilnih aplikacija. Iako su baze podataka i sistemi koji su naslijeđeni iz generacija kad mobilne aplikacije nisu bile ni u povoju, nove tehnologije i pravci daju mogućnost da se ostvari sprega između klasičnog i modernog i na taj način postignu velika dostignuća u cilju olakšavanja svakodnevnih aktivnosti, pomaganja korisniku i potpunog uvida u stanje nekih svojih servisa i profila, u ovom slučaju telekomunikacionog operatera.

Učešće u ovom projektu za autore donosi sveobuhvatnu sliku i iskustvo, od vođenja projekta, planiranja i dizajna, analize, kasnije i faze programiranja, nadgledanja i testiranja i na kraju implementacije. Sve faze su vođene i kontrolisane, izvještavane i praćene sa odgovarajućim rizikom. Mobilna aplikacija je odličan E2E (end to end) projekat za rad i iskustvo.

LITERATURA

- [1] <http://msdn.microsoft.com/en-us/library/ms123401.aspx>
- [2] http://docs.oracle.com/html/B16022_01/ch3.htm
- [3] <http://msdn.microsoft.com/en-us/library/ms187910.aspx>

INTEGRACIJA BEŽIČNIH SENZORSKIH MREŽA U CLOUD COMPUTING-u INTEGRATION WIRELESS SENSOR NETWORKS IN CLOUD COMPUTING

Mirko Kosanović, Miloš Kosanović, *Visoka tehnička škola strukovnih studija Niš*

Sadržaj: U ovom radu daćemo pregled i načine povezivanja dve tehnologije, koje zadnjih godina zaokupljaju veliku pažnju IT stručnjaka i od kojih se u budućnosti mnogo očekuje, Bežične Senzorske Mreže (BSM) i Cloud Computing (CC). Od ovih tehnologija u budućnosti se očekuje da u potpunosti izmene način rada, razmišljanja, pa i celokupnog života ljudi. BSM predstavljaju velike mreže malih senzorskih čvorova, koje su sposobne da samostalno prikupljaju, obrađuju i međusobno razmenjuju veliki broj različitih podataka. Mogućnost da se ti podaci distribuiraju putem TCP/IP protokola na Internet, na kome postoji mogućnost korišćenja velikog broja naprednih tehnologija, kao i povezivanje sa njima, daje neslućene mogućnosti za razvoj velikog broja distribuiranih aplikacija koje se mogu primeniti na gotovo svim poljima ljudske delatnosti. Zato i ne čudi činjenica da se zadnjih godina istraživanja na polju povezivanja BSM sa Internetom i korišćenje njegovih prednosti, sve više i više rastu. Ovaj rad prikazuje pregled dosadašnjih dostignuća na ovom polju sa posebnim osvrtom na predlog jedne arhitekture za povezivanje BSM i CC, kao i mogućnosti primene iste kao i probleme koje u budućnosti treba rešavati na ovom polju.

Abstract: In this paper, we present an overview and a ways of connecting the two technologies, which in recent years deal of attention in IT professionals and from which, in future, are much anticipated, Wireless Sensor Networks (WSN) and Cloud Computing (CC). Of these technologies in the future is expected to completely change the method of working, thinking, and even the whole human life. WSN are a large network of small sensor nodes, which are capable of independently collected, processed and exchanging a wide variety of data. The possibility that the data is distributed via the TCP/IP protocol on the Internet, where there is the possibility of using a large number of advanced technologies, as well as connect with them, gives unprecedented possibilities for the development of a large number of distributed applications that can be applied to almost all fields of human activity. Therefore, it is not surprising that in recent years, research in the field of WSN connect to the Internet, and use its advantages, more and more growth. This paper presents an overview of the achievements in this field with particular reference to the proposal for a single architecture for connecting WSN and CC, as well as the possibilities of the same and the problems that need to be addressed in the future in this field.

1. UVOD

Kraj XX veka obeležio je veliki razvoj Interneta koji je doveo do trenutnog povezivanja velikog broja ljudi i to na gotovo neverovatnim brzinama, reda Gb/s, a koje se svake godine sve više povećavaju. Sledeća etapa u njegovom razvoju biće da se to isto ostvari i sa velikim brojem objekata u našem okruženju. Već tokom 2008 godine broj Internet povezanih objekata na našoj planeti je pretekao ukupan broj ljudi koji na njoj živi. Trenutno ima preko 10 milijardi međusobno povezanih objekata a prema tvrdnjama futuriste Cisco Sistema, Dejv Evansa, očekuje se da krajem 2020 godine ta brojka dostigne oko 50 milijardi međusobno povezanih računarskih objekata [1]. Njegova vizija predstavlja da će u budućnosti svet biti pun umreženih, pametnih uređaja koji će posedovati jedinstvenu identifikaciju (IP broj ili radio-frekvencijsku identifikaciju - RF broj). Svi ti uređaji biće opremljeni velikim brojem različitih senzora i moći će da potpuno samostalno sakupljaju veliki broj različitih podataka, koje će bežičnom vezom međusobno razmenjivati ili prosleđivati nadređenom uređaju (*sink-u*), koji će te podatke omogućiti da budu dostupni na Internetu. On je nastao kao krajnji proizvod tri osnovne tehnologije koje su zadnjih godina pretrpele najveći napredak

i razvoj: bežične komunikacije, mikro-elektromehanički sistemi (MEMS) i Internet tehnologije. Jedna od osnovnih prednosti IoT je ta da omogućava velikom broju ljudi da sa veoma malo znanja i iskustva kreira jako moćne a jeftine aplikacije. Sve to omogućeno je razvojem velikog broja mikro platformi, kao što su *Arduino*, *Propeller* i *Microchip PIC* familija, i njihovom integrisanju putem bežične komunikacije sa Internetom. Drugim rečima, ova tehnika zahteva od inženjera da podjednako raspolažu sa znanjima iz hardvera i softvera. Ovo je potrebno kako bi to znanje primenili na koherentan i integrisan način u cilju dobijanja primenjive IoT aplikacije [2, 3].

Sa druge, Internet strane, došlo je do velikog razvoja Internet tehnologija koje su doprinele da se veliki broj podataka prikupi u realnom vremenu (*real time*), i to sa velikog broja računara, i to gotovo trenutno prezentuje klijentima. Tu se pre svega misli na nove generacije distribuiranih računarskih tehnologija: *Cluster computing*, *Grid computing* i *Cloud computing*. One su sada bile preduslov za razvoj jednog novog pogleda na softverske aplikacije tkz. WEB servis orijentisani softver. Razvijen je veliki broj softverskih alata i protokola koji su bili strogo orijentisani ka rešavanju problema u ovoj oblasti. To su pre

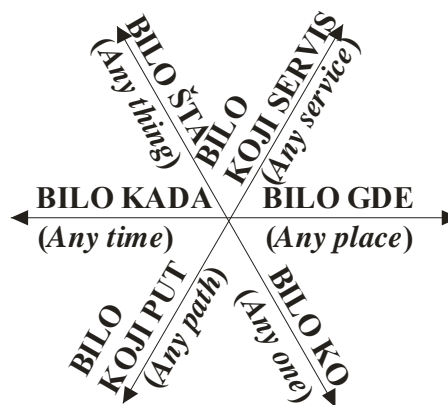
svega XML (*Extensible Markup Language*), WSDL (*Web Service Description Language*), UDDI (*Universal Description, Discovery and Integration*), LINQ (*Language-Integrated Query*) i dr. Pored toga razvijene su i dve tehnologije koje su omogućavale da se preuzimaju podaci sa različitih izvora i isti šalju ka WEB servisima: *Representational State Transfer* (REST) i *Simple Object Access Protocol* (SOAP). Ove tehnologije imaju i mogućnost da koriste mehanizme za razmenu poruka kao što su E-mail, SMS ili da razmenjuju poruke putem društvenih mreža i blogova [4]. Sve te osobine bile su idealne da se njihove pogodnosti iskoriste za povezivanje sa jednim distribuiranim izvorom velikog broja podataka a to su BSM.

Problem povezivanja BSM i Cloud Computing-a razmatran je u ovom radu. Ova konekcija nam omogućava da dobijemo informaciju koju sakuplja svaki SČ u BSM, sa bilo kog mesta, vremenski bilo kada i bilo koju informaciju. Nakon uvoda u problem u poglavlju 2 opisali smo dostupne tehnologije koje su doprinele povezivanju BSM sa *Cloud Computing*-om. U poglavlju 3 date su opšte karakteristike i vrste *Cloud Computing*-a. Poglavlje 4 daje predlog jedne arhitekture za povezivanje ove dve tehnologije i navodi prednosti iste. Poglavlje 5 zaključuje ovaj rad.

2. DOSTUPNE TEHNOLOGIJE

Velikim razvojem objektno orjentisanih tehnologija (OOT), koje su u mnogome diktirale razvoj računarskih sistema i razvojnih okruženja, postalo je neophodno da dostupnost računara bude "svuda prisutno" (*ubiquitous*). Dobro poznato pravilo 3A postalo je osnovni preduslov i neophodnost svih klijent-server aplikacija. Ono kaže da računari moraju uvek da budu dostupni, tj. *any one* (bilo ko), *any place* (bilo gde) i *any time* (bilo kada). Drugim rečima to bi značilo da serveri treba da budu potpuno transparentni i dostupni za sve, da ne zavise od tehnologije izrade, mesta gde se oni postavljaju tj. gde se nalaze korisnici (*users*) ili programeri (*developers*) kao i da budu uvek dostupni. Razvoj IoT duplirao je ove zahteve tako da sada imamo pravilo 6A koje ima još tri dodatna zahteva u odnosu na predhodno pravilo i to: *any thing* (bilo šta), što podrazumeva da svaka stvar ima svoju jedinstvenu identifikaciju preko koje je uvek jednoznačno dostupna, *any path/network* (na bilo koji način) i *any service* (bilo koji dostupan servis) (Slika 1.).

Da bi zadovoljio gore pomenute zahteve IoT mora u sebi da objedini veliki broj naprednih tehnologija koje međusobno moraju da budu povezane kako bi omogućile njihova ispunjenje, tj. primenljivost. Sve te tehnologije mogu se svrstati u tri osnovne kategorije i to: i) minijaturan hardver koji se sastoji od senzora, aktuatora, mikrokontrolera i bežičnog primopredajnika; ii) srednji sloj (*Middleware*) koji je zadužen za skladištenje i obradu/analizu prikupljenih podataka i iii) prezentacioni sloj koji je zadužen za prihvatanje tih podataka i njihovo predstavljanje krajnjim korisnicima na različitim platformama. Navešćemo sada nekoliko tehnologija čiji je razvoj, po našem mišljenju, u mnogome doprineo pojavi IoT [5, 6]:



Slika 1. Šest A – opšte karakteristike IoT.

Radio frekvencijska identifikacija - RFID (*Radio Frequency Identification*) predstavlja veoma zastupljenu tehnologiju jednoznačne identifikacije objekata koja je danas široko u upotrebi. Predstavlja tehnologiju daljinskog slanja i prijema podataka koja koristi radio talase. Svakom objektu se ugradi/zalepi jedan mirkočip (RFID pločica), izuzetno malih dimenzija, koji u sebi ima jedinstveni ID broj promenljive dužine (1-256 bita). RFID pločica u sebi sadrži antenu koja joj omogućava prijem i slanje podataka. Radiofrekvencijska komunikacija zasniva se na stvaranju elektromagnetnih talasa koji dolaze do RFID pločice od strane RFID čitača. Inicirana ovim talasima RFID pločica emituje sopstvene elektromagnetne talase prema primopredajniku u kojima je nalazi jedinstveni ID očitane RFID pločice. Posebna prednost ove tehnologije je da pasivne RFID pločice ne zahtevaju nikakvo napajanje već potrebnu energiju crpu iz primljenih radio talasa od strane RFID čitača. Ova osobina omogućila je praćenje velikog broja objekata a da pri tome nemamo potrebe za napajanjem tih objekata.

Bežične senzorske mreže - Bežične senzorske mreže (BSM) se sastoje od velikog broja bežičnih senzorskih uređaja koji potpuno samostalno formiraju mrežu, putem koje oni prikupljaju, obrađuju i razmenjuju podatke. Neke od ključnih karakteristika BSM su: a) senzorski čvorovi (SČ) su gusto raspoređeni u regionu i veoma su podložni otkazima (kvarovima), b) proizvoljan i nepredvidljiv raspored SČ-ova, c) česte promene topologije senzorskih mreža, d) uglavnom se ne koristi jedinstvena globalna identifikacija (ID) SČ, e) SČ-ovi su ograničeni u pogledu napajanja, komunikacione i računarske snage. U principu, BSM može da radi kao samostalna mreža ili da bude povezana sa drugim mrežama. Ali za mnoge aplikacije, BSM ne radi efikasno u punoj izolaciji. Mora da se omogući način za spoljašnje nadgledanje događaja kao i za pristup podacima koje nam daje mreže senzora. Dakle, javlja se potreba da BSM povežemo na već postojeću mrežnu infrastrukturu, kao što su lokalne mreže (LAN), gradske mreže (Metropolis), ili mrežu svih mreža Internet čime ostvarujemo osnovne zahteve IoT. Imajući u vidu da skup TCP/IP protokola postaje *de facto* standard u mrežnom okruženju nameće se rešenje da se ovaj skup protokola primeni i u BSM. Međutim, ovaj zadatak nije baš jednostavan sa gledišta energetske efikasnosti u BSM tako da se pribegava drugim rešenjima [7].

Tehnologija virtuelizacije - Virtuelizacija predstavlja vrstu softverske tehnologije koja može da se primeni u mnogim oblastima. Virtuelna mašina (VM) predstavlja softversku implementaciju nekog hardvera, na primer računara, koja se u stvarnosti ponaša isto kao i taj računar. Na taj način na jednom računaru (na kome je pokrenut program *hypervisor*, koji omogućava virtuelizaciju) možemo pokrenuti više VM. Kako svaka VM ima sopstveni kernel, operativni sistem, biblioteke i aplikacije, ona u suštini ne zavisi od hardvera na kome se instalira već od vrste *hypervisor-a*. Ranije virtuelizacione tehnologije bile su dosta nestabilne i nedostajala im je fleksibilnost što je ograničavala njihovu široku primenu u računarskim mrežama. Međutim, zadnjih godina dolazi do velikog razvoja ovih tehnologija, a sa druge strane došlo je do povećanja propusnog opsega u mrežama što je sve omogućilo da se concept *Cloud Computing-a* u potpunosti zaživi [8]. Sada, zahvaljujući virtuelizaciji, klijenti više nisu zavisni od snage svojih lokalnih računara već mogu da koriste snagu *Cloud Computing-a* koja je sada centralizovana i znatno nadmašuje klijentske mogućnosti.

3. PRINCIPI CLOUD COMPUTING-A

Cloud computing predstavlja termin koji objedinjuje gotovo sve raspoložive resurse na Internetu: računare, aplikacije, podatke, memoriju, servise i usluge. Za pristup *Cloud* aplikacijama dovoljno je da klijent ima odgovarajuću Internet konekciju i bilo koji Internet pretraživač. *Cloud Computer* servisi omogućavaju fleksibilan, moćan i ekonomičan okvir u pružanju usluga korisnicima. *Cloud* se sastoji od hardvera, mreže, usluge, skladištenje i interfejsa koji omogućavaju isporuku gotovo svih računarskih usluga korisnicima. On se zasniva na ekonomskom principu "plati-za-korišćenje" (*Pay-Per-Use*) i omogućuje velikom broju klijenata laku dostupnost raspoloživim resursima u realnom vremenu i to sa bilo kog mesta i bilo kada. Klijent ne može da vidi resurse ili odredi fizičku lokaciju, kao i organizaciju svih resursa koje koristi. Kada su odobrena klijentu, resursi se samo izvuku iz "oblaka" resursa i nakon korišćenja vraćaju se ponovo u oblak. Neke od glavnih prednosti koje nam *Cloud Computing* računarstvo donosi možemo svrstati u:

1. Smanjenje početnih troškova.
2. Laka alokacija resursa na zahtev, bez ograničenja.
3. Održavanje i nadgradnja resursa se radi u pozadini.
4. Lako brz razvoj i saradnja sa drugim resursima u Cloud-u.
5. Više mogućnosti za globalni razvoj usluga

U CC-u postoje tri osnovne usluge na raspolaganju i to su: Softver kao usluga, platforma kao servis i infrastrukturu kao servis.

1. PaaS (*Platform as a Service*) – omogućava klijentima da pristupaju softverskim razvojnim platformama koje će mu omogućiti pisanje sopstvenih aplikacija. Klijenti razvijaju i izvršavaju svoje aplikacije u okruženju koje im nudi provajder *Cloud* servisa. U okviru njih mogu da se nađu već gotovi podaci ili strukture koje *Cloud* servis poseduje. Klasičan primer predstavlja korišćenje nekog operativnog sistema od strane klijenta koji on ne poseduje. *Google App Engine* je jedan od popularnih PaaS primera.

2. SaaS (*Software as a Service*) – omogućuje korišćenje velikog broja već gotovih softverskih aplikacija bez kupovine licence, preuzimanja i instaliranja na klijentskoj strani. Provajder vodi računa oko instaliranja, novih verzija i dopuna ponuđenih aplikacija. Često se ovaj servis naziva "softver na zahtev" i njegovi predstavnici su korišćenje standardnih aplikacija kao *Office* paketi, *Corel* i td. Jedan od provajdera koji nudi ove usluge je *Google*.

IaaS (*Infrastructure as a Service*) – omogućava zakupljivanje materijalnih i nematerijalnih resursa na mreži. Obuhvata računare, memorijske kapacitete, ali i različite usluge u vidu praćenja i preusmeravanja mrežnog saobraćaja. Klijenti u odabranom Cloud-u prave svoj virtuelni klaster, koji samostalno održavaju tj. instaliraju softver, upravljaju i izvršavaju aplikacije u okviru njega. Ova usluga predstavlja i najveću prednost *Cloud Computing-a*, jer organizacije najviše ulažu u potrebnu mrežnu infrastrukturu. Amazon je jedan od najpoznatijih provajdera koji nude ove usluge.

Navešćemo neke od bitnih karakteristika koje definišu osnovne osobine *Cloud Computing-a* [9]:

- 1) usluga na zahtev (*Service on demand*): zahtev klijenata da koristi resurse mogu se automatski ispuniti bez direktne ljudske interakcije.

- 2) elastičnost zahteva (*Elasticity of demand*): Ne postoji formalna saglasnost ili ugovor o vremenskom periodu za korišćenje resursa. Klijenti mogu da koriste resurse kad god to požele i mogu ih osloboditi kada završe.

- 3) Apstrakcija (*Abstraction*): klijenti ne vide stvarne fizičke lokacije korišćenih resursa. Međutim, oni mogu da koriste resurse bez znanja o lokaciji izvora odakle će podaci biti preuzeti i gde će se oni čuvati.

- 4) mrežni pristup (*Network access*): klijent aplikacija može da se izvrši na raznim platformama, uz pomoć mobilnih telefona, laptopa ili PDA uređaja, korišćenjem pouzdane Internet konekcije.

- 5) kontrola korišćenja servisa (*Service measurement*): bez obzira što resurse u *Cloud-u* koristi više klijenata, *Cloud* infrastruktura ima mogućnost da za svakog klijenta pojedinačno kontroliše vremenski period korišćenja resursa.
- 6) udruživanje resursa (*Resource pooling*): resursi se dinamički dodeljuju iz *Cloud-a* na osnovu zahteva klijenata.

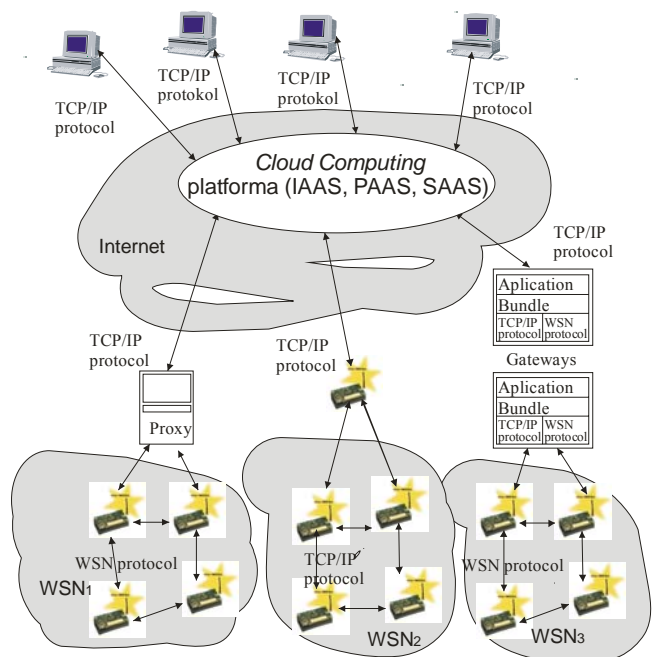
4. PRINCIP PREDLOŽENOG REŠENJA

Proces prikupljanja podataka iz nagledanog prostora nije ni malo lak posao, jer se treba suočiti sa svim osobenim karakteristikama BSM, o kojima je ranije bilo reči. Kako se BSM obično primenjuju u nepristupačnim terenima one su poznate po velikom procentu loše primljenih paketa. Ako tome dodamo da u ovde obično zastupljene višeskokovite (*multy-hop*) topologije problem se još više uvećava. U BSM, u zavisnosti od aplikacije, za prenos paketa primenjuje se jedna od sledećih tehnika: klasično rutiranje ili princip plavljenja (*flooding*). Osnovna razlika između ovih tehnika je da klasično rutiranje zahteva jedinstvenu identifikaciju svakog SČ za razliku od plavljenja kod koga to nije potrebno.

Glavni problem koji treba rešiti kod povezivanja BSM sa *Cloud-om* sastoji se u tome da je neophodno da se obezbedi:

- a) pristup svakom SČ sa TCP/IP bazirane mreže,
- b) efikasna

komunikacija sa aspekta potrošnje energije SČ; i c) transparentnost u radu između TCP/IP protokola i protokola koji se primenjuju u BSM.



A. Proxy architecture B. Overlay based architecture C. Gateway architecture
Slika 2. Tipična arhitektura povezivanja BSM sa CC

Slika 2. prikazuje tipičnu arhitekturu koja omogućuje da BSM povežemo sa *Cloud Computing*-om. SČ-ovi prikupljaju podatke iz regiona i te podatke prosleđuju glavnom SČ, koji predstavlja uređaj koji treba da omogući slanje podataka ka Internetu. U literaturi se navode nekoliko komunikacionih modela za povezivanje BSM sa TCP/IP mrežama tj. Internetom. U zavisnosti od primenjene komunikacione arhitekture, oni se dele na: *Proxy* arhitekture, *Overlay* arhitekture i *Gateway* arhitekture [10]. Osnovna uloga ovog uređaja je da prihvaćene podatke koji su došli iz BSM prosledi ka *Cloud*-u gde će se oni upamtiti. *Cloud* aplikacija vrši analiziranje tih podataka i na osnovu njihove vrednosti preduzima odgovarajuće korake: izdaje neke naredbe izvršnim SČ ili nadležnim računarima, šalje SMS ili E-mail poruke, i td.

Povezivanje BSM sa *Cloud Computing*-om omogućuje nam da na jednostavan način prikupljamo veliku količinu podataka koje SČ-ovi šalju na osnovu očitavanja različitih senzora. Osnovna prednost ovakve arhitekture sastoji se u tome da se sada informacije koje mogu da se izvuku na osnovu prikupljenih podataka ponude kao jedan Internet servis. Naime, svi prikupljeni podaci mogu da se analiziraju na bilo kom stvarnom ili virtuelnom računaru u *Cloud*-u i da se na osnovu dobijenih rezultata preuzimaju dalji koraci. Tako su i nastali termini *Sensing as a Service* (SaaS) i *Sensor Event as a Service* (SEaaS) koji označavaju jedan Internet process, od trenutka prikupljanja informacija pa sve do generisanja događaja koji je zahtevao klijent koji koristi taj process. Ova mogućnost obezbedila je da se ova arhitektura može primeniti u mnogim aplikacijama jer se gotovo automatski vrši prikupljanje podataka, njihova analiza a kao krajnji rezultat svega toga je neki unapred definisani događaj tj. izvršna radnja.

5. ZAKLJUČAK

Do pre nekoliko godina, krajnja tačka Interneta bio je kućni računar ili laptop. Danas, Internet adresibilni uređaji mogu se pronaći u mnogim drugim uređajima kao što su: mobilni telefoni, internet radio prijemnici, televizija, tablet računari i navigacioni sistemi. Broj uređaja sa Internet vezom konstantno će rasti u narednih nekoliko godina, a samim tim i količina podataka kao i zahtevi korisnika. Upravo zbog toga, primena Internet kompatibilnih SČ, postaće neizbežna nužnost. Međutim, IP komunikacija između SČ-ova u okviru BSM predstavlja jako težak zadatak, pre svega zbog jako ograničenog propusnog opsega, malog memorijskog prostora i ograničene energije kojom raspolažu SČ-ovi. Povezivanjem BSM sa nekom strukturom kod koje ovo nisu ograničavajući faktori ovi problemi se donekle mogu prevazići. *Cloud Computing* upravo predstavlja jednu takvu strukturu jer ima veliki propusni opseg, gotovo neograničen memorijski prostor, kao i konstantan, stabilan izvor napajanja. Kombinovanjem ovih arhitekture, BSM i *Cloud Computing*-a, dobijamo jednu snažnu arhitekturu koja zbog svojih pozitivnih osobina daje gotovo neograničene mogućnosti u primeni, a da pri tome ne traži neka velika ulaganja kako materijalna tako i ulaganja u neke skupe softvere.

LITERATURA

- [1] D. Evans, The Internet of Things [INFOGRAPHIC], <http://blogs.cisco.com/diversity/the-internet-of-things-infographic/> pos. 25.01.2014
- [2] Ian G. Smith, *The Internet of Things 2012 New Horizons*, Halifax, UK, 2012.
- [3] J. Gubbia, R.Buyyab, S.Marusic, M.Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions", *Future Generation Computer Systems* 29, 2013, pp.1645–1660
- [4] M. Beraka, H. Mathkour, S. Gannouni, H. Hashimi, "Applications of Different Web Service Composition Standards," *IEEE International Conference on Cloud and Service Computing (CSC)*, pp.56-63, Nov. 2012
- [5] D. Guinard, V. Trifa, T. Pham, O. Liechti, "Towards Physical Mashups in the Web of Things", *Proceedings of INSS 2009*, Pittsburgh, USA, 2009.
- [6] J. Rellermeier at all, "The Software Fabric for the Internet of Things", In *Proceedings of the First International Conference on the Internet of Things*, Zurich, Switzerland, March 2008.
- [7] M.Kosanovic, M.Stojcev, "Connecting Wireless Sensor Networks to Internet", *Facta Universitatis, Series: Mechanical Engineering Vol. 9, No 2*, 2011, pp.169-182
- [8] F.B Shaikh, S Haider, "Security threats in cloud computing," *IEEE Internet Technology and Secured Transactions (ICITST)*, pp. 214-219, Dec. 2011
- [9] A. Alamri at all, "A Survey on Sensor-Cloud: Architecture, Applications, and Approaches", *International Journal of Distributed Sensor Networks*, Volume 2013, Article ID 917923, 18 pages
- [10] M.Kosanović, "Povezivanje bežičnih senzorskih čvorova sa TCP/IP mrežama", XVIII IT, Žabljak 26.02-1.03.2013, pp. 97-100

PRIMENA TABLET TEHNOLOGIJE U SAVLAĐIVANJU REČI ENGLSKOG JEZIKA IMPLEMENTATION OF TABLET TECHNOLOGY TO ENHANCE ENGLISH WORDS RECOGNITION

Aleksandar Trifunović, Svetlana Čičević, Andreja Samčović, *Saobraćajni fakultet, Beograd*
Milkica Nešić, *Medicinski fakultet, Niš*

Sadržaj: *Predavači su danas pred ozbiljnim izazovom da usklade propisane norme u pogledu usvajanja novih znanja sa interesovanjima savremenog učenika. Mobilne tehnologije podstiču razvoj specifičnih mentalnih sposobnosti i omogućuju studentima da pri učenju koriste više čula, kako bi poboljšali rezultate učenja. U radu su predstavljeni rezultati eksperimenta u kome je upotrebljen Tablet PC za učenje reči engleskog jezika. Generalno, studenti su pokazali bolji učinak nakon sesije učenja korišćenjem tablet tehnologije.*

Ključne reči: Učenje jezika, mobilne tehnologije, Tablet PC

Abstract: *Nowdays educators are faced with a serious challenge to adjust the interests of the modern students to the norms of the level of adoption of new knowledge. Due to its multimedial content, mobile technologies facilitate the development of specific mental skills, along with employing almost all senses to improve learning outcomes. This paper presents the results of an experiment in which Tablet PC was implemented for English words recognition and learning. Generally, students showed improved performances after the learning sessions.*

Key words: English language learning, mobile technologies, Tablet PC

1. UVOD

Tokom poslednjih nekoliko godina, dogodila se nagla ekspanzija multimedijalnih komunikacionih uređaja sa ekranom osetljivim na dodir, u prvom redu pametnih telefona i tablet računara. Dostupnost velikog broja modela, uporedo sa drastičnim padom cene i sveopštom prisutnosti brzog širokopojasnog bežičnog interneta, doveli su do toga da ovi uređaji uđu u svakodnevnu upotrebu u svim sferama svakodnevnih ljudskih delatnosti, pa time i u obrazovanju. Mobilni uređaji sa ekranima osetljivim na dodir prevazilaze nedostatke tradicionalnog obrazovanja, upotpunjuju ga, ili ga potpuno zamenjuju.

Metodika nastave jezika struke i nauke u najnovije vreme stavlja procese učenja u centar istraživanja, gde se posebna pažnja posvećuje aktiviranju samih studenata, motivaciji za učenje i usklađivanju potreba onih koji uče s onim šta, kada, koliko, gde i kako treba to da savladaju. Pitanje kako se uči jezik struke i nauke dobija važno mesto [1]. Upotreba novih tehnologija u učenju mora kreirati izazovne aktivnosti koje omogućuju studentima da povežu novu informaciju sa starom, da usvoje novo značenje i upotrebe svoje kognitivne sposobnosti, jer je strategija nastave ta, a ne tehnologija, koja utiče na kvalitet učenja [2]. Tablet računari pružaju studentima velike mogućnosti, pre svega interaktivnost u procesu učenja [3, 4].

Veliki broj univerziteta je u poslednjoj dekadi započeo pilot projekte uvođenja tablet računara u nastavu, sa ciljem proučavanja njegove efikasnosti. Istraživanje sprovedeno na Univerzitetu u Vašingtonu, pokazalo je da su pažnja i motivacija studenata na predavanjima bila na daleko višem nivou i da su bolje razumeli sadržaj kada je on bio prezentovan uz pomoć Tablet PC-ja, a olakšana je i

realizacija nastave i proces ocenjivanja [5, 6, 7, 8]. Postoji više efikasnih metoda učenja reči stranog jezika, kao na primer, učenje putem edukativnih igrica, zatim učenje slušanjem i uz izgovor, kao i implementacijom slikovnih prikaza odgovarajućih reči [9, 10, 11].

Činjenica da, iako studenti mnogo vremena provode na društvenim mrežama ili posećujući sajtove koji su na stranom jeziku, prevashodno engleskom, ipak pokazuju funkcionalnu računarsku nepismenost, i probleme pri učenju, odnosno pamćenju stručnih termina, kao i da relativno retko koriste mobilne uređaje u obrazovne svrhe [12] motivisala nas je da sprovedemo eksperiment u kome bismo ispitali efikasnost primene Tablet računara u pamćenju i učenju stručnih termina na engleskom jeziku.

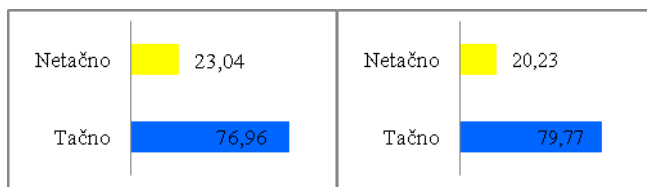
2. MATERIJAL I METODE

U istraživanju je učestvovalo 20 ispitanika dobrovoljaca, od toga 40% ženskog pola, dok su 60% činili ispitanici muškog pola.

U eksperimentu je korišćen tablet računar dijagonale 7 inča, putem koga je ispitanicima prikazivana *power point* prezentacija koja je sadržala materijal za učenje. U prvom delu eksperimenta zadatak ispitanika je bio da povlačenjem linija povežu imenicu na engleskom jeziku sa njenom odgovarajućom slikom. U drugom delu eksperimenta ispitanici je trebalo da povežu reč na engleskom jeziku sa odgovarajućim terminom na srpskom jeziku. Upotrebljeno je ukupno 22 pojma čije je značenje trebalo naučiti. Primarni cilj drugog dela prezentacije je bio da proveriti da li su ispitanici zaista naučili značenje novih pojmova na engleskom jeziku uz upotrebu tablet računara.

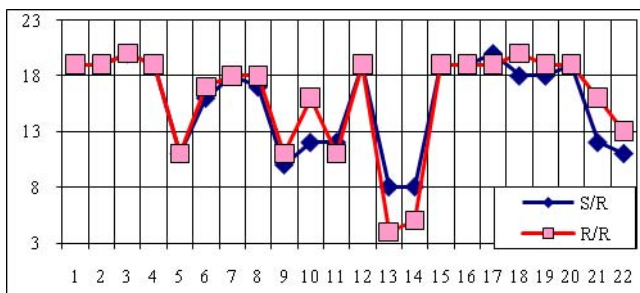
3. REZULTATI

Rezultati eksperimenta će biti predstavljeni na osnovu analize odgovora ispitanika, ali će biti prikazan i opis odgovora koji su ispitanici davali na svaku pojedinačnu reč. Ako analiziramo rezultate eksperimenta sa stanovišta broja tačnih, odnosno pogrešnih odgovora, vidimo da se, generalno, u drugom delu eksperimenta povećava procenat tačnih odgovora (Slika 1).



Slika 1. Procenat tačnih/netačnih odgovora u prvom delu eksperimenta (slika- reč), i na kraju eksperimenta (povezivanje reč-reč), respektivno

Ako izvršimo analizu odgovora po rečima, ustanovićemo da postoje tri kategorije reči. Prvoj kategoriji pripadaju reči za koje je dat tačan odgovor u oba dela eksperimenta (10 reči). Možemo pretpostaviti da to može biti zbog toga što su te reči ispitanicima bile poznate od ranije, zato što su ih naučili već u prvom delu eksperimenta, ili su jednostavno, lake za učenje. Druga kategorija reči su one za koje se daje veći broj tačnih odgovora u drugom delu eksperimenta (8 reči). To može ukazivati da je došlo do poboljšanja performansi na zadatku, odnosno do učenja. Procenat poboljšanja za ovu kategoriju reči se kreće između 5 i 20%. Najzad, trećoj kategoriji pripadaju reči za koje se daje manji broj tačnih odgovora u drugom delu eksperimenta, kada treba povezati dve reči (4 reči) (Slika 2).



Slika 2. Broj tačnih odgovora za povezivanje slike i reči (S-R) i reči sa rečima (R-R) iz Tabele 1.

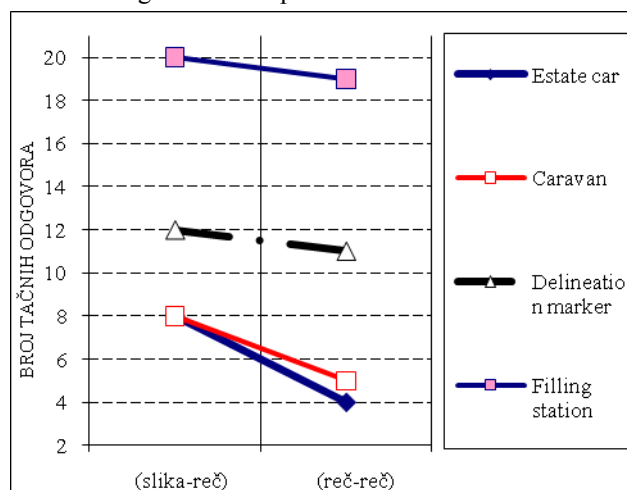
Tabela 1. Reči predstavljene na x osi, na Slici 2.

1	2	3	4	5	6
Multilevel garage	Wind cone	Suspension bridge	Angle collision	Barrier line	Wheelbase
7	8	9	10	11	12
Chassis	Guardrail	Cross slope	Curb	Delineation marker	Direction indicator
13	14	15	16	17	18
Estate car	Caravan	Driving mirror	Edge of carriageway markings	Filling station	Gearshift
19	20	21	22		
Highway code	Motor driven cycle	Abutment	Buffer amortizer		

Ovakav rezultat bi se mogao objasniti na više načina. Najpre, možemo zaključiti da nije došlo do učenja, ili je

zaboravljeno ono što je naučeno, ili se radi o reči koja je teška za učenje, i najzad, moguće je da ispitanicima više odgovara prezentacija slikovnih, nego verbalnih sadržaja. Lateralizacija hemisfera i razvijenost prostornih sposobnosti takođe mogu da daju svoj doprinos ovakvom rezultatu. Da sumiramo, za približno 37% prezentovanih reči došlo je do učenja (poboljšanje učinka u drugom delu eksperimenta), dok je za 18.8% reči učinak lošiji kada treba povezati dve reči (Slike 3 i 4).

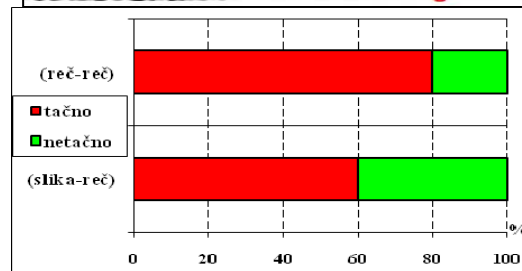
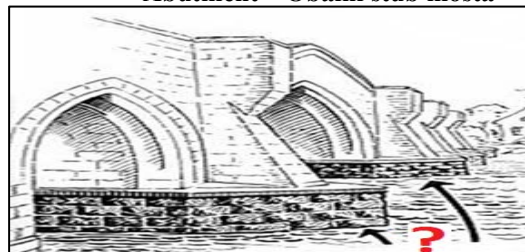
Slika 3. Reči kod kojih je izražen efekat učenja, bolji rezultat u drugom delu eksperimenta



Slika 4. Reči kod kojih je zabeležen lošiji učinak u drugom delu eksperimenta

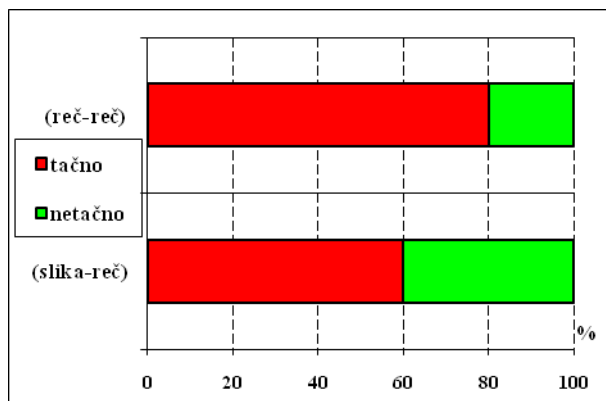
Prosečno, studenti su u drugom delu tačnije odgovarali za sledeće reči: *Curb* (+20%), *Abutment* (+20%) (Slike 5a. i 5b.), *Gearshift* (+10%), *Buffer amortizer* (+10%), *Wheelbase* (+5%), *Guardrail* (+5%), *Cross slope* (+5%) i *Highway code* (+5%). Manje uspeha pri povezivanju dve reči studenti su zabeležili kod sledećih termina: *Estate car* (-20%), *Caravan* (-15%), *Delineation marker* (-5%) i *Filling station* (-5%). Za ostale reči, procenat tačnih odgovora, jednak je u prvom i drugom delu eksperimenta: *Multilevel garage*, *Wind cone*, *Suspension bridge*, *Angle collision*, *Barrier line*, *Chassis*, *Direction indicator*, *Driving mirror*, *Edge of carriageway markings* i *Motor driven cycle*.

Abutment – Obalni stub mosta



Slika 5a.

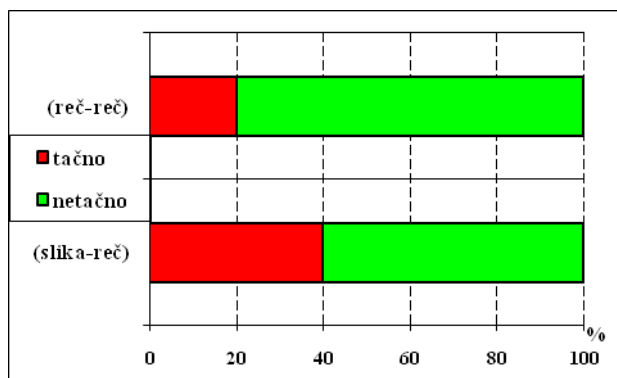
Curb – Ivičnjak



Slika 5b. Reči koje se najlakše uče (najveća ostvarena razlika između prvog i drugog dela eksperimenta, 20%)

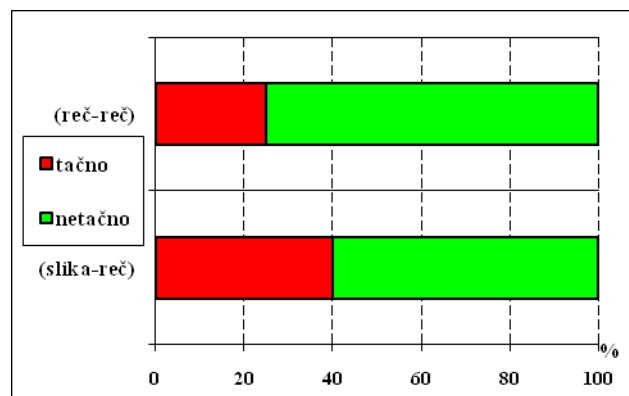
Najmanje tačnih odgovora, u oba dela zadatka, studenti su dali kod imenice *Estate car* - 40% tačnih odgovora u prvom delu zadatka kada su povezivali reč sa slikom, 20% tačnih odgovora u drugom delu zadatka (reč-reč) (Slika 6.), dok je *Caravan* druga reč koja je zadavala probleme ispitanicima-40% tačnih odgovora u prvom delu zadatka, odnosno 25% u drugom delu, (Slika 7.). Studenti koji su pravili grešku kod dva navedena pojma (*Caravan* i *Estate car*) u svim slučajevima pogrešnog odgovora su permutovali značenja ove dve reči.

Estate car – Karavan



Slika 6. Primer reči koja se najteže uči, ostvaren najbolji učinak u drugom delu eksperimenta – 20%)

Caravan – Prikolica za stanovanje



Slika 7. Primer reči koja se teško uči, ostvaren najbolji učinak u drugom delu eksperimenta – 15%)

4. DISKUSIJA I ZAKLJUČCI

Generalno, procenat tačnih odgovora je veliki i u slučaju kada se povezuju slika i reč, i kada se sparuju dve reči. Za približno 37% prezentovanih reči došlo je do učenja-poboljšanje učinka u drugom delu eksperimenta. Uzimajući u obzir navedeno, možemo zaključiti da korišćenje tableta ima veliki potencijal za unapređenje kvaliteta nastave engleskog jezika.

U daljim fazama razvoja kurseva engleskog jezika, prvenstveno tokom usvajanja određene stručne terminologije, neophodno je osmišljavanje i kreiranje novih aplikacija koje bi bile prilagođene korisnicima, različitih nivoa početnog znanja i individualnih preferencija kako bi se obezbedila što veća efikasnost ishoda učenja.

Uz pomoć tablet računara iste sadržaje učenici mogu usvajati na različite načine, slušanjem snimljenog predavanja, analizom slika, grafika i tabela, ili pak igranjem igara, simulacija i rešavanjem kviza. Nastavnik može pripremiti sadržaje u različitim formatima i lako ih učiniti dostupnim učenicima, kojima je pružena veća mogućnost izbora i iskorišćenje punih potencijala.

Dalje pravce istraživanja treba usmeriti ka utvrđivanju limita broja reči koje pojedinac može da nauči kroz jednu prezentaciju, kao i na ispitivanje sklonosti različitih starosnih grupa ka učenju engleskog jezika putem tablet računara. Na primer, [13] sugerišu da je broj ajtema po lekciji koje je razumno prezentirati da bi učenje bilo efikasno između 8 i 12 zavisno od stepena prethodnog znanja. Takođe, vremenski intervali koji proteknu između perioda učenja i vežbanja i provere znanja imaju uticaja na subjektivni doživljaj učenja i količinu zapamćenog materijala [14, 15]. Upravo su tablet računari veoma pogodni za izvođenje ovakve prakse zbog svoje dostupnosti i nezavisnosti od vremena i prostora.

Konačno, potrebno je prikupiti kompletnije podatke o uzorku, od računarske pismenosti, do stilova učenja pojedinaca kako bi se dobili što pouzdaniji rezultati i zaključci, i prilagodili nastavni sadržaji karakteristikama i potrebama svakog pojedinog korisnika.

LITERATURA

Ovaj rad je podržan i finansiran od strane Ministarstva prosvete i nauke Republike Srbije (Projekti br. 32025, 32048, 36022, 36006 i 179002).

- [1] G. Dimković-Telebaković, "Kultura jezika struke i nauke," *Zbornik Matice srpske za filologiju i lingvistiku*, XLI/1, pp.129-134, 1998.
- [2] M. Ally, *Foundations of educational theory for online learning*, In T. Anderson & F. Elloumi (Eds.), *Theory and practice of online learning*, *Athabasca, Canada: Creative Commons: Athabasca University*, 2004.
- [3] X. B. Chen, "Tablets For Informal Language Learning: Student Usage And Attitudes," *Lang Learn & Technol*, (17)1, pp. 20–36, 2013.
- [4] M. Warschauer, and C. Meskill, *Technology and second language learning*. In J. Rosenthal (Ed.), *Handbook of undergraduate second language education* (pp. 303-318). Mahwah, New Jersey: Lawrence Erlbaum, 2000.
- [5] R. Anderson, Ru. Anderson, B. Simon, S. Wolfman, T. VanDeGrift, & K. Yasuhara, Experiences with a tablet PC based lecture presentation system in computer science courses, In *Proc. of the 35th SIGCSE technical symposium on Computer science education*, Norfolk, Virginia: ACM Press, New York, NY, USA, 2004.
- [6] S. Clark, L. Taylor, and J. Pickering Understanding the impact of Tablet PCs on Students' Learning and Academics Teaching, In *Proc. of the Second Innovation in Accounting and Corporate Governance Education Conference*, 2007.
- [7] S. W. Haugland, "The effect of computer software on preschool children's developmental gains," *J Comp Child Educ*, 3, pp. 15-30, 1992.
- [8] L. C. Jones, "Supporting listening comprehension and vocabulary acquisition with multimedia annotations: The students voice," *CALICO J*, 21, pp. 41-65. 2003.
- [9] A. Wright, *Pictures for Language Learning*, Cambridge: Cambridge University Press, 1989.
- [10] M. Kamioka, "How English words will be forgotten after their memorization?," *English Educ*, 31(8), 42–47, 1982.
- [11] N. Schmitt and M. McCarthy (eds.), *Vocabulary: Description, Acquisition and Pedagogy*, Cambridge University Press, 1997.
- [12] S. Čičević and A. Trifunović, The implementation of mobile technology in the English vocabulary learning in the field of transportation and traffic, *Foreign Language In Transport And Traffic Engineering Profession And Science*, pp. 111-129, 2013.
- [13] G. Ruth and R. Stuart, *Working with Words: A Guide to Teaching and Learning Vocabulary* (Cambridge Handbooks for Language Teachers), Cambridge: Cambridge University Press, 1986.
- [14] C. O. Fritz, R. Duffy and P. E. Morris, *Distributed practice learning foreign vocabulary – The effects of interval length*, Summary report of an experimental study, pp, 6, 2010.
- [15] W. Thalheimer, *Spacing Learning Events Over Time: What the Research Says*, 2006. Retrieved from <http://www.work-learning.com/catalog/>

OPRAVDANOST INICIJATIVE ZA IZMJENU NASTAVNOG PLANA I PROGRAMA INFORMATIKE U SREDNJEM STRUČNOM OBRAZOVANJU EKONOMSKE I TRGOVINSKE STRUKE, ZANIMANJE TRGOVAC
JUSTIFICATION OF THE INITIATIVE FOR CHANGING OF THE CURRICULUM FOR THE INFORMATION TECHNOLOGY IN THE SECONDARY VOCATIONAL EDUCATION IN ECONOMIC AND TRADE PROFESSIONS, AVOCATION: SALESMAN

*Aleksandar Ristić, Javna ustanova Gimnazija Prnjavor, Prnjavor
Dalibor Damjanović, Visoka poslovno-tehnička škola, Doboj*

***Sadržaj:** Analiza nastavnog plana i programa informatike važan je i složen posao. Važnost i složenost proizilaze iz stalnih i brzih promjena na polju aktualnosti informaciono-komunikacionih tehnologija kao i specifičnosti zemlje u kojoj se izvodi nastavni plan i program (Republika Srpska, Bosna i Hercegovina). Učestale promjene zakonskih regulativa, isprepletanost entitetskih i državnih zakona i pravilnika naveli su nas na razmišljanje o stanju nastavnog plana i programa informatike u odnosu na navedene činioce. S druge strane, posmatrali smo nastavni program u didaktičko-metodičkom smislu jer isključivo poštovanje Zakona neće biti dovoljno za kvalitetno i valjano obrazovanje učenika. Uvidjevši krupne propuste, predložili smo dozvoljenu izmjenu nastavnog programa koja bi mogla poslužiti za dalju reformu nastavnog programa informatike u srednjem stručnom obrazovanju ekonomske i trgovinske škole, zanimanje trgovac.*

***Abstract:** The analysis of informatics curriculum is important and complex job. The importance and complexity comes from permanent and rapid changes in the field of actuality of the information and communication technologies as well as specifics of the country in which the curriculum is being performed (Serbian Republic, Bosnia and Herzegovina). Frequent changes in tax legislation, interweaving entities and state laws and also regulations have led us to reflect on the state of curriculum informatics in relation to these mentioned factors. On the other hand, we observed the curriculum in didactic - methodological terms because obeying the Law will not be enough for a good and proper education of the students. Seeing key failures, we have proposed a permissible change of the curriculum that could be used in further reform of curriculum informatics in a high school education of economic and trade school, occupation trader.*

1. UVOD

Nastava u srednjem obrazovanju Reublike Srpske izvodi se po nastavnim planovima i programima koji su stari više od deset godina. Prisutni su izuzeci kod kojih je vršena revizija nastavnog plana i programa. Nastava informatike u srednjoj stručnoj školi ekonomske i trgovinske struke, zanimanje trgovac, posmatrana iz ugla aktualnosti uveliko je zastarila jer nastavni plan i program informatke nije revidiran više od deset godina. U tih deset godina promijenjeni su brojni zakoni i pravilnici, donesena je Strategija razvoja obrazovanja, prihvaćene su krovne obaveze na nivou BiH, dok je s druge strane ubrzani razvoj informaciono-komunikacionih tehnologija doveo do pojave novih vrijednosti čija aktualnost i primjenjivost zaslužuju temeljno izučavanje. Zadatak nam je bio da utvrdimo u kojoj su mjeri u nastavnom planu i programu informatike uvažavani didaktički principi vezani za nastavu informatike kako bismo uvidjeli potrebu za reformama u tom pravcu koji je nezavisan, a podjednako značajan u odnosu na aktualnost i zakonsku utemeljenost.

2. TEORIJSKI PRISTUP PROBLEMU

Prema Obrazovnom pojmovniku Agencije za predškolsko, osnovno i srednje obrazovanje [1] definišaćemo sljedeće pojmove:

Cilj obrazovanja je vaspitna/odgojna i obrazovna namjera kako za individualni tako i za društveni aspekt. Cilj se konkretizuje nizom zadataka koji su vaspitni/odgojni i obrazovni.

Ishodi učenja su znanja, sposobnosti, vještine i stavovi koje učenici treba da imaju na kraju razreda/ciklusa/nivoa obrazovanja.

Modul je odvojeni i koherentni blok učenja. Kao takav predstavlja zaokruženi set rezultata učenja sa jasno definisanim kriterijem za ocjenjivanje koji može stajati sam i biti izolovan ili biti u vezi sa drugim modulima.

Nastavni plan je dokument kojise propisuju predmeti i časovi/sati za pojedine obrazovne predmete.

Nastavni program je dokument kojim se propisuju obrazovni sadržaji koje treba realizovati na odgovarajućem stepenu (nivou) obrazovanja.

Prilikom analize uvaženi su osnovni didaktički principi, a posebna pažnja posvećena je ishodima učenja, sponi između tradicionalnog nastavnog plana i kurikularnog oblika.

Analizirajući nastavni plan i program za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac [2], vidimo da se u prvoj godini srednjeg obrazovanja izučava informatika kao jedini predmet koji se odvija putem vježbi. Posmatrajući drugu i treću godinu, zaključujemo da se osim informatike učenici u obliku vježbi bave isključivo praktičnom nastavom, koja slijedi u drugoj godini i nastavlja se tokom treće godine školovanja. Primjećujemo da je nastava planirana globalno, kroz dva modula, koja jasno ne određuju koliko je časova potrebno za svaki modul, odnosno za svaku jedinicu.

3. CILJEVI I ZADACI ANALIZE

Cilj:

Utvrđiti u kojoj mjeri nastavni plan i program informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac uvažava osnovne principe didaktike sa posebnim osvrtnom na didaktičke principe od posebne važnosti za nastavu informatike. Ukoliko utvrdimo značajne nedostatke i propuste, dati preporuke izmjene u okviru prisutnih mogućnosti.

Zadaci:

Na osnovu prethodno navedenog cilja, definišemo sljedeće zadatke:

Izvršiti analizu nastavnog plana i programa informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac u skladu sa opštim didaktičkim principima nastave informatike;
U skladu sa zaključcima utvrditi potrebne izmjene postojećeg nastavnog programa.

4. METODOLOGIJA

Analiza nastavnog plana i programa informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac obavljena je kroz dvije faze.

Prva faza

Imajući u vidu suštinsku bitnost analize nastavnih sadržaja, odnosno nastavnih planova i programa, prema [3], upotrijebili smo sljedeće kriterijume:

opseg nastavnog plana,
organizacija nastavnih sadržaja,
nastavni ciljevi i sadržaji učenja,
struktura sadržaja.

Prilikom analize vodili smo računa o didaktičkim principima, posebno se osvrćući na didaktičke principe od posebnog značaja za nastavu informatike. Prema [4], to su:

princip postupnosti,
princip primjerenosti uzrastu učenika,
princip očiglednosti,
princip individualizacije nastave.

Druga faza

U drugoj fazi izvode se zaključci i daju se preporuke nastale iz prethodne analize.

5. REZULTATI ANALIZE

Opseg nastavnog plana

Pregledajući nastavni plan informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac, zaključujemo da se predmetu Informatika dodjeljuje prostor isključivo u prvom razredu u odnosu na ukupno trogodišnje školovanje.

Nastava informatike je zastupljena sa 2 časa sedmično, odnosno 72 godišnje i izvodi se u računarskom kabinetu u obliku vježbi.

Organizacija nastavnih sadržaja

Nastavni sadržaji su organizovani kroz dva modula:

Računarski sistemi,

Primjena računara.

Modul Računarski sistemi realizuje se kroz četiri jedinice:

Računarski sistem,

Operativni sistem,

Rad sa datotekama,

Rad sa pomoćnim programima i alatima.

Modul Primjena računara realizuje se takođe kroz četiri jedinice:

Obrada teksta,

Obrada tabela,

Internet servisi,

Primjena računara u struci.

Za svaku jedinicu u modulu Računarski sistemi definisani su rezultati učenja (ishodi) i date su smjernice za nastavnika. Navedene jedinice, ishodi i smjernice za nastavnika predstavljaju osnov za dalje planiranje gradiva.

Modul Primjena računara takođe daje rezultate učenja, odnosno ishode kao i smjernice za nastavnika. Međutim, četvrta jedinica Primjena računara u struci daje isključivo nagovještaj o mogućnosti izbora softvera, bez dodatnih uputstava, smjernica i ishoda. Ovakav pristup smatramo nedopustivim jer ne daje čak ni osnovno polazište koje bi bilo podložno daljoj razradi.

Nastavni ciljevi i sadržaji učenja

Posmatrajući nastavni program kroz dva modula, primjetno je da prvi modul [5], definiše sljedeće ciljeve:

shvatanje značaja i uloge informacionih tehnologija u modernom društvu,

ohrabrivanje učenika za sticanjem vještine rada na računaru,

razvijanje sposobnosti samostalnog rada,

shvatanje potrebe rada u grupi,

razvijanje komunikacionih sposobnosti,

razvijanje pravilnog odnosa prema opremi,

sticanje potrebe za daljim učenjem,

ohrabrivanje učenika za primjenu novih tehnologija i komunikacija.

Navedeni ciljevi predstavljaju opšte prihvaćene ciljeve didaktike, bez uporišta u konkretnom programu koji je diktiran rezultatima učenja (ishodima). Zanimljivo bi bilo obaviti uporednu analizu ciljeva nastavnog programa informatike i ciljeva srednjeg obrazovanja iz Zakona o srednjem obrazovanju i vaspitanju.

Drugi modul [5], definiše sljedeće ciljeve:

ohrabrivanje učenika za sticanjem vještine rada na računaru,

razvijanje sposobnosti samostalnog rada,

shvatanje potrebe rada u grupi,
razvijanje komunikacionih sposobnosti,
razvijanje pravilnog odnosa prema opremi,
sticanje potrebe za daljim učenjem,
ohrabivanje učenika za primjenu novih tehnologija i komunikacija.

Ciljevi drugog modula takođe nisu u bliskoj povezanosti sa rezultatima učenja (ishodima), već donose samo blagu povezanost sa tematikom koju je potrebno izučavati kako bismo zadovoljili rezultate učenja.

Ukoliko u nastavnom programu pokušavamo definisati jasne sadržaje učenja, vidimo da se oni zasnivaju na ishodima koji diktiraju sadržaje koji bi trebali biti planirani i obrađeni. Nastavni sadržaji okvirno su definisani jedinicama u svakom modulu. Oba modula imaju po četiri jedinice. Prvi modul sadrži:

Računarski sistem,

Operativni sistem,

Rad sa datotekama,

Rad sa pomoćnim programima i alatima.

Drugi modul definiše sljedeće jedinice:

Obrada teksta,

Obrada tabela,

Internet servisi,

Primjena računara u struci.

Razmatrajući navedene jedinice zaključujemo da se prvi modul bazira na osnovama računarskih sistema, pružajući učenicima mogućnost usvajanja osnovnih znanja o računarskim sistemima, dok u drugom modulu preovladava primjena računara, odnosno priprema učenika za konkretnu primjenu računara.

Struktura sadržaja

Govoreći o strukturi sadržaja potrebno je posebnu pažnju obratiti na didaktičke principe od posebnog značaja za nastavu informatike.

Zasnovanost strukture sadržaja bitna je i sa polazišta ciljeva nastave informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac.

Princip postupnosti

Prema [7] princip postupnosti predviđa dovođenje učenika u stanje primjenjivog znanja postupnim putem, odnosno korak po korak, koji bi u samoj konačnici imao odliku primjenljivog, svrsishodnog.

Posmatrajući module nastavnog programa osnova informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac, zaključujemo da su sadržaji struktuirani po principu postupnosti u zadovoljavajućem nivou. Naime, u nastavnom programu osnova informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac, u prvom modulu i prvoj temi (Računarski sistemi), učenik najprije upoznaje osnovnu ulogu računarskog sistema, pojmove hardvera i softvera, dok na kraju teme uspješno razlikuje aplikativni i sistemski softver, a potom pristupa konkretnom upoznavanju računarskih sistema.

Slična je situacija i u drugom modulu gdje, na primjer, imamo u trećoj temi izučavanje internet servisa kod kojih se učenik prvo upoznaje sa osnovnim pojmovima, a potom upoznaje način povezivanja na internet i čitanja interneta, da

bi na kraju uspješno ovladao značajnijim servisima kao što su elektronska pošta, učenje na daljinu itd.

Analizirajući i ostale teme unutar prvog i drugog modula jasno se primjećuje da je princip postupnosti uvažen u velikoj mjeri, odnosno u svim temama izuzev četvrte teme drugog modula (Primjena računara u struci), gdje se princip postupnosti u potpunosti zapostavlja i ne navodi se nijedan sastavni element, nego se isključivo precizira korištenje programa iz struke bez ikakvih daljnjih pojašnjenja.

Princip povezanosti teorije i prakse

Vrlo je jednostavno afirmativno govoriti o važnosti povezanosti teorije i prakse. S obzirom na nasušnu potrebu konkretnih, stručnih znanja, teorijske osnove unutar predmeta Informatika nužno je potkrijepiti stručnim i svrsishodnim činjenicama. Posmatrajući ranije navedene module nastavnog programa osnova informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac, primjećujemo da se časovi izvode putem vježbi i da su gotovo sve teme potkrijepljene praktičnim primjerima. Međutim, ponovo uočavamo istovjetni problem koji u potpunosti zanemaruje temu Primjena računara u struci, unutar drugog modula, ne dajući ni osnovne smjernice. Ovakav pristup u potpunosti odbacujemo smatrajući da je nedopustivo da upravo u temi koja predstavlja sinergiju teorije i prakse kompletna metodička razrada (pa tako i princip povezanosti teorije i prakse) biva zapostavljena i najblaže rečeno nedorečena. Mišljenja smo da je važno pomenuti pristup promijeniti i dati kompletnu metodičku razradu koja će u potpunosti uvažavati princip povezanosti teorije i prakse.

Princip očiglednosti

Princip očiglednosti ključan je za razumijevanje nastavnih sadržaja. Nastavni plan i program osnova informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac, unutar smjernica za nastavnike na indirektan način poziva na poštovanje principa očiglednosti, no ipak ne navodi konkretne korake koji bi omogućili stvaranje jasnije predstave o implementaciji nastave koja će u potpunosti sadržati očiglednu nastavu.

Princip individualizacije nastave

Posmatrajući informatiku kao nastavni predmet koji se odvija putem vježbi lako zaključujemo da je individualizacija potrebna i vrlo moguća jer doprinosi boljem napredovanju i stvaranju ličnih kompetencija. Međutim analizirajući nastavni plan i program osnova informatike za srednje stručno obrazovanje, ekonomske i trgovinske struke, zanimanje trgovac, unutar ciljeva izučavanja nastavnog predmeta nalazimo da je postavljen cilj razvijanja sposobnosti samostalnog rada i rada u grupi koji poprimaju naličje afirmisanja principa individualizacije nastave. Izuzev pomenutog ne nalazimo druge upute koje bi doprinijele zadovoljavanju ovog značajnog didaktičkog principa.

6. ZAKLJUČCI I PREPORUKE

Nastavni sadržaji su organizovani kroz dva modula. Oba modula sadrže iste ciljeve učenja, što je nedopustivo, s

obzirom na diferencije među tematskim cjelinama. Svaki modul sadrži četiri nastavne teme, za koje su dati rezultati (ishodi) učenja i smjernice za nastavnika. Četvrta tema drugog modula, Primjena računara u struci, ne sadrži detaljne ishode učenja, nego se svodi na jedan ishod u kojem se govori da će učenik biti spreman da koristi program iz struke po želji odjeljenja i nastavnika. Ovakva formulacija je u potpunosti neprihvatljiva u pogledu svih didaktičkih principa i osnovnih pravila. Nedopustivo je da se ne definišu nivoi znanja, da se ne prate zahtjevi struke i da se na taj način u potpunosti minorizuje značaj navedene tematske cjeline koja predstavlja suštinsku novost u odnosu na ranije izučavanje informatike koje su učenici imali tokom obaveznog osnovnog obrazovanja. S toga, kao vrlo važnu i hitnu mjeru, predlažemo izmjenu unutar postojeće teme, a sve u okviru preporuke Ministarstva prosvjete i kulture RS koja dopušta izmjene nastavnog programa u opsegu od 15%, u kojoj će kao primjer biti obrađene neke nastavne jedinice, a za ostale koje se preporučuju za navedenu temu, biće date detaljne smjernice za nastavnike i predloženi jasni i nedvosmisleni ishodi koji će biti u skladu sa Zakonom o srednjem obrazovanju i vaspitanju, Osnovama kvalifikacijskog okvira u BiH za relevantni obrazovni profil.

Posmatrali smo nastavne ciljeve i sadržaje učenja. Tom prilikom smo uočili da svi ciljevi i rezultati (ishodi) učenja po kojima je potrebno planirati sadržaje nisu usklađeni. S toga je veoma važno napraviti korekcije koje će dovesti do potpunog sklada između nastavnih ciljeva i sadržaja učenja, odnosno njegovih rezultata (ishoda).

Bavili smo se odnosom strukture sadržaja i poštovanjem pedagoških principa od posebnog značaja za nastavu informatike. Iz navedenog zaključujemo da je princip postupnosti poštovan u zadovoljavajućem nivou, osim u temi Primjena računara, gdje je princip postupnosti zanemaren jer nije naveden nijedan postupni ishod učenja ili su date bilo kakve smjernice koje imaju elemente postupnosti. Slična situacija je i kod poštovanja principa povezanosti teorije i prakse. Razmatrali smo oba modula i u njima sve nastavne teme. Zaključujemo da je situacija na ovom polju prihvatljiva i da postoji značajna povezanost teorije i prakse, izuzev u temi Primjena računara, gdje zbog već pomenutog preskakanja planiranja i davanja smjernica za nastavnike i rezultata (ishoda) učenja, imamo situaciju u kojoj se navodi da učenici treba da koriste program po izboru iz struke, ne objašnjavajući u kojoj mjeri su potrebne teorijske osnove, u kojoj mjeri su potrebna praktična znanja itd. Poštovanje principa očiglednosti utemeljeno je u samim ciljevima

izučavanja informatike, bez datih konkretnih smjernica koje bi omogućile jasniji pristup daljem planiranju gradiva. Slična je situacija i kod principa individualizacije. U ovom slučaju, on nije jasno naveden, ali iz ciljeva izučavanja informatike i isticanja samostalnog rada i rada u grupi, zaključujemo da je vođeno računa o uvažavanju ovog principa za nastavu informatike. Navedeni razlozi daju potpunu opravdanost za inicijativu revidiranja nastavnog plana i programa. S druge strane, takvoj inicijativi treba da prethodi temeljna analiza usklađenosti postojećeg nastavnog plana i programa i važećih zakonskih regulativa i prihvaćenih krovnih obaveza. Kvalitet nastavnog plana i programa moguće je postići potpunim uvažavanjem kako pravila didaktike, tako i važećih zakonskih regulativa i prihvaćenih generičkih deskriptora kvalifikacijskih nivoa po ishodima učenja.

LITERATURA

- [1] http://www.aposo.gov.ba/wp-content/uploads/2012/08/RJECNIK_ODABRANIH_OBRADOVNIH_TERMINA.pdf, posjećeno: 6.2.2014.
- [2] <http://www.rpz-rs.org/sajt/doc/file/Nastavni%20planovi%20za%20SREDNJE%20SKOLE/Zadnja%20izmjena%2030.08.2013/11.Ekonomija%20pravo%20i%20trgovina.xls>, posjećeno: 5.2.2014.
- [3] N. Pastuović, *Edukologija*, Zagreb: Znamen, 1999.
- [4] K. Voskresenski, D. Glušac, *Metodika nastave informatike*, Zrenjanin: Tehnički fakultet „Mihajlo Pupin“, 2007.
- [5] http://www.rpz-rs.org/sajt/doc/File/sr/01/06695_20061016054721_Modul_1_-_Osnove_racunarskih_sistema.doc, posjećeno: 5.2.2014.
- [6] http://www.rpz-rs.org/sajt/doc/File/sr/07/06696_20061016054733_Modul_2_-_Poslovna_primjena_racunara.doc, 5.2.2014.
- [7] M. Vilotijević, *Didaktika*, Beograd: Zavod za udžbenike i nastavna sredstva, 1999.

MODEL STRATEŠKOG RAZMIŠLJANJA U IT OKRUŽENJU STRATEGIC THINKING MODEL IN IT ENVIRONMENT

Risto Bojović, *Fakultet tehničkih nauka, Kosovska Mitrovica*

Sadržaj: U radu je objašnjena uloga strateškog razmišljanja pri realizaciji različitih IT projekata. Predložen je konkretan model strateškog razmišljanja, navedeni i objašnjeni njegovi osnovni elementi kao i rezultati koji treba da proizadju.

Abstract: In this paper the role of strategic thinking in the process of an IT project realization is described. A certain model of strategic thinking is proposed. The basic elements of this model are explained as well as expected results.

1. UVOD

Pojmovi kao što su „strateško razmišljanje“, „strateško planiranje“ i „strateško odlučivanje“ vrlo se često pominju i gotovo uvek asociraju na određene delatnosti koje se sprovode na nekim višim nivoima (npr. vlasti ili menadžmenta kompanija). To je verovatno zbog toga što na tim nivoima postoji izrazita potreba za donošenjem različitih strategija koje služe kao smernice u radu, ali i kao osnova za procenu uspešnosti rada. Teme tih strategija su različite: ekonomski i privredni razvoj, finansije, tržište, prosveta, zdravstvo, kultura, i slično. Naravno, i oblast informacionih tehnologija (IT) je predmet različitih strateških razmišljanja na pomenutim višim nivoima, ali je to obično u okviru širih tema, čak i kada su u pitanju kompanije iz oblasti IT-a. Najčešće se na IT gleda kao na sredstvo pomoću koga se mogu ostvariti neki „viši“ (uglavnom finansijski i ekonomski) ciljevi.

Slično je i sa izrazom „kreativno razmišljanje“, koji se kod nas najčešće pominje u vezi sa obrazovanjem dece.

Međutim, imajmo na umu sledeće činjenice:

- a) da se po mnogim mišljenjima, već uveliko nalazimo u post-industrijskom dobu [1], čiji su simbol upravo IT,
- b) da se količina informacija u svetu udvostručava svakih 18 meseci ili češće [2],
- c) da se oblast IT-a ubedljivo najbrže razvija,
- d) da su IT snažan pokretač razvoja i svih drugih oblasti,
- e) da je IT oprema koja se koristi sve složenija i moćnija, dok joj cena generalno opada, zbog čega postaje sve dostupnija,
- f) da se brzina tehnološkog napredovanja neprekidno povećava, tako da možemo govoriti ne o brzini, nego o ubrzanju tehnološkog razvoja.

Planiranje u pomenutom okruženju predstavlja pravu noćnu moru. Jer, posmatrano sa investicionog aspekta, investirati u nešto što će koliko sutra biti sigurno manje ili više tehnološki prevaziđeno predstavlja nezahvalan i stresan posao. Utoliko stresniji ukoliko se radi o većem ulaganju [3]. Štaviše, stresnost pogodja obe strane, kako investitora, tako i onoga ko pruža uslugu investitoru. Jedini pravi odgovor na sve ovo jeste primena principa i algoritama strateškog razmišljanja, strateškog planiranja, usklađivanja i implementacije.

Pitanja koja se, takođe, postavljaju su i: 1) kako da se pojedinac i organizacija snađu u svemu ovome i 2) kakvi treba da budu ljudi čija je profesija oblast IT-a? Jer, sigurno je da je uloga IT profesionalaca u savremenom društvu veoma značajna, budući da se radi o ljudima koji su važni nosioci razvoja društva u celini, da je njihovo obrazovanje nadprosečno i da raspolazu izuzetno moćnim oruđima.

Takođe, suprotno raširenom mišljenju da je strateško razmišljanje rezervisano samo za lidere na najvišem nivou i one organizacione jedinice koje se bave strateškim planiranjem, danas sve više počinje da prevladava shvatanje da strateško razmišljanje predstavlja konstantnu aktivnost u koju treba da budu uključeni svi zaposleni [4]. Jer, jasno je da samo postojanje strateškog plana nije dovoljno za ostvarivanje svih ciljeva koji su njime definisani. Zaposleni koji ga sprovode moraju na prvom mestu da poseduju sposobnosti prilagođavanja različitim konkretnim situacijama i da u kratkom roku odgovore na različite izazove. Ovo je naročito karakteristično za oblast IT-a čije svakodnevne inovacije ni jedan strateški plan ne može unapred da predviti. Jedino što je moguće, to je da izvršioci budu osposobljeni da u skladu sa planskim opredeljenjima u konkretnim situacijama pronalaze najbolja moguća rešenja. Neophodno je da dobro poznaju ne samo svoj deo posla, već i generalni plan u celini, njegove suštinske ciljeve, svrhu i širi značaj koji ima za društvo i konkretnu organizaciju.

Zbog toga je potrebno da realizatori projekata ovladaju osnovnim algoritmima i principima strateškog razmišljanja na različitim nivoima. Pri tome, mora se voditi računa o specifičnostima pojedinih delatnosti, budući da svaka od njih

ima svoje karakteristične algoritme razmišljanja (kao rezultat prirode posla, istorijskog, obrazovnog i životnog bekgraunda) kojima se treba prilagoditi i u čijem svetlu treba omogućiti primenu strateškog razmišljanja. Pomenute specifičnosti potrebno je sistematizovati i jasno istaći jer se najčešće one nesvesno usvajaju tokom školovanja i života.

2. OSNOVNI ZADACI STRATEŠKOG RAZMIŠLJANJA

Osnovne teme kojima je potrebno da se bavi strateško razmišljanje u oblasti IT-a proizilaze iz dve osnovne prethodno pomenute karakteristike IT-a:

- a) da je to oblast koja se veoma brzo razvija i
- b) da je IT oprema koja se koristi sve moćnija.

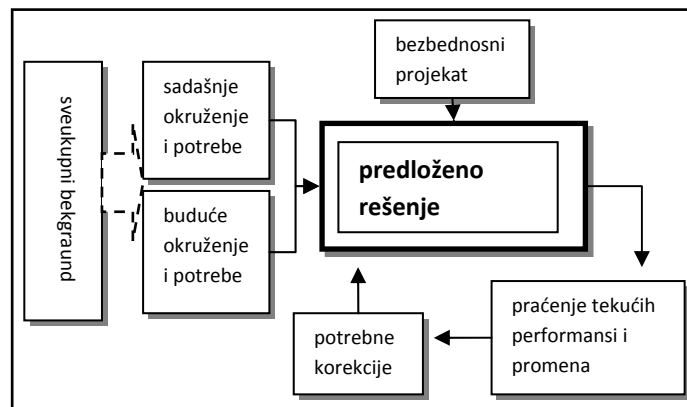
Brz razvoj se može smatrati osnovnom karakteristikom IT sektora. I, kako je prethodno rečeno, tu više ne treba govoriti o brzini, već o ubrzanju razvoja. Ovaj razvoj je, doduše, skokovit sa testerastim oblikom krive, ali krive koja ima izrazito uzlaznu putanju.

Usavršenost IT opreme je dostigla takav nivo da jednostavno „nije sve za svakoga“. Dovoljno je samo de se setimo bezbednosnih problema i nekoliko skorašnjih međunarodnih afera s tim u vezi.

Stoga bi zadaci strateškog razmišljanja bili:

- 1) Jasno definisanje cilja koji se želi ostvariti;
- 2) Analiza okruženja u kome je potrebno ostvariti postavljeni cilj;
- 3) Predviđanje budućih karakteristika okruženja;
- 4) Projektovanje bezbednosnog sistema;
- 5) Definisanje osnovnih karakteristika rešenja koje treba predložiti;
- 6) Koncipiranje sistema praćenja i analize funkcionisanja rešenja koje treba realizovati;
- 7) Definisanje skupa aktivnosti koje moraju biti na raspolaganju u slučaju potrebnih korekcija.

Shematski prikaz ovih zadataka dat je na Slici 1. Postojanje povratne sprege je od suštinske važnosti za uspešno funkcionisanje bilo kog rešenja. Sveukupni bekgraund podrazumeva obrazovno-kulturološke osobenosti kako izvođača, tako i korisnika. Od njih umnogome zavisi kako će biti definisane osnovne karakteristike rešenja, kako će ono konačno izgledati, kakav će smisao imati i kakve će posledice ostaviti na okruženje.



Slika 1. Struktura strateškog razmišljanja

3. ELEMENTI STRATEŠKOG RAZMIŠLJANJA

Da bi strateško razmišljanje dalo dobar rezultat neophodno je da se ono sprovede u skladu sa njegovim osnovnim načelima. Analizom različitih teorijskih razmatranja i praktičnih iskustava opisanih u literaturi [5], možemo navesti sledeće bitne elemente strateškog razmišljanja koje karakterišu uspešne pojedince:

- 1) Perspektiva;
- 2) Kapaciteti u pogledu znanja i veština;
- 3) Samosvest;
- 4) Metakognicija [6];
- 5) Konceptualni kapacitet;
- 6) Izvršna fleksibilnost;
- 7) Kreativno i kritičko razmišljanje;
- 8) Preuzimanje rizika;
- 9) Intuicija.

Perspektiva omogućuje postizanje strateškog nivoa. Ona podrazumeva postojanje intelektualnog okvira za pogled u budućnost kao smernicu za unošenje promena. Bez ovog okvira promene neće biti u središtu pažnje, biće slučajne i bez izgleda na uspeh [6].

Znanja i veštine predstavljaju preduslov za obavljanje svakog posla, pa i strateškog razmišljanja. Stoga su znanja i veštine iz oblasti koja je tema strateškog razmišljanja nešto što se samo po sebi podrazumeva i bez njih jednostavno nema kompetentnosti.

Samosvest se definiše kao sposobnost sagledavanja i realne procene sopstvenih sposobnosti bitnih za obavljanje određene delatnosti. Uobičajeno je da se samosvest izražava takozvanim „koeficijentom samozablude“ koji predstavlja meru (ne)podudarnosti samoocenjivanja i ocenjivanja od strane drugih. On je pozitivan kada su ocene drugih više od

sopstvenih ocena, a negativan kada su ocene drugih niže od sopstvenih. Negativan koeficijent samozablude javlja se kod ljudi koji nisu svesni uticaja i posledica svog ponašanja.

Metakognicija je proces koji nadgleda i kontroliše proces kognicije (spoznaje). Tehničkim rečnikom, to bi bila povratna sprega u procesu poimanja stvarnosti, što je od izuzetne važnosti kada se radi o složenim problemima koji zahtevaju pronalaženje novih rešenja.

Konceptualni kapacitet predstavlja sposobnost nalaženja alternativnih rešenja na bazi sagledavanja problema u širem kontekstu. Veći konceptualni kapacitet je garancija boljeg razumevanja svih pitanja i problema vezanih za oblast u kojoj se deluje i preduslov za uspešno obavljanje postavljenih zadataka. Smatra se da konceptualni kapacitet pojedinca raste sa godinama i iskustvom.

Izvršna fleksibilnost je tesno povezana sa konceptualnim kapacitetom. Ona predstavlja sposobnost izvršioca da sagleda problem iz više različitih uglova, čime se omogućuje preformulisanje složenih problema i dolaženje do jednostavnijih rešenja uz istovremeno izbegavanje nedoumica i mogućih dvosmislenosti. Izvršna fleksibilnost je direktno proporcionalna sa konceptualnim kapacitetom, stručnim interesovanjima i kreativnošću, a obrnuto je proporcionalna sa sa mentalnom rigidnošću.

Smatra se da su kreativno i kritičko razmišljanje dve strane jednog istog novčića. Teško ih je razdvojiti, ali je, takođe, i teško naći osobu kod koje su podjednako zastupljeni. Većina ljudi je sklonija kritičkom razmišljanju i provodi znatno manje vremena u traženju različitih rešenja za složene probleme. Kreativnost je za mnoge misteriozni dar koji poseduju pojedinci i koji je teško naučno opisati. Za druge je to, opet, karakteristika pojedinca koja se može pokrenuti društvenom dinamikom i odgovarajućom motivacijom.

Preuzimanje rizika je odlika strateškog razmišljanja koja se najčešće vezuje za odlučivanje na najvišim nivoima. Sa aspekta rešavanja konkretnih tehničkih problema, težnja je da se rizik svede na minimum. On se, naravno, često ne može izbeći i posledica je, pre svega, brzih promena u okruženju. Takođe, rizik je imanentan kreativnim rešenjima, budući da ona odstupaju od već poznatih i proverenih.

Najzad, intuicija je nešto što predstavlja rezultat delovanja kompletnog bekgrounda pojedinca. Ona je rezultanta celokupnog njegovog iskustva i njegova specifičnost.

4. OKVIRNI SADRŽAJ STRATEŠKOG IT PLANA

Nezavisno od toga o kom klijentu se radi, strateški IT plan danas mora da da odgovore i predvidi rešenja za neke od usluga koje su ili već postale standard, ili će to biti u najskorijoj budućnosti. Osim toga, kao sadašnji standard, ove usluge i infrastruktura koja ih podržava u budućnosti će najverovatnije predstavljati osnovu na kojoj će se vršiti nadogradnja.

Posmatrano sa stanovišta korisnika IT-a, tehnički zahtevi kojima je danas najčešće potrebno odgovoriti su:

- pristup uslugama sa bilo kog mesta, bilo kojim uređajem i u bilo koje vreme,
- što veća brzina prenosa podataka
- raspoloživost usluga i podrške 24 časa, 365 dana u godini,
- operabilnost svih usluga i aplikacija na mobilnim uređajima,
- mogućnost pristupanja udaljenim računarskim resursima,
- mogućnost obezbedjenja dodatnih kapaciteta na zahtev korisnika,
- bezbedan pristup aplikacijama i podacima,
- jednostavan i bezbedan oporavak u slučaju havarije.

Sve ove zahteve moguće je realizovati sa različitim nivoom kvaliteta usluge, a moguće je da korisnici imaju i još neke druge, specifične, zahteve o kojima je, takođe, potrebno voditi računa. Međutim, i na osnovu ovih pomenutih, može se videti sva složenost posla koji je potrebno obaviti i veličina intelektualnog napora koju realizator posla mora da uloži.

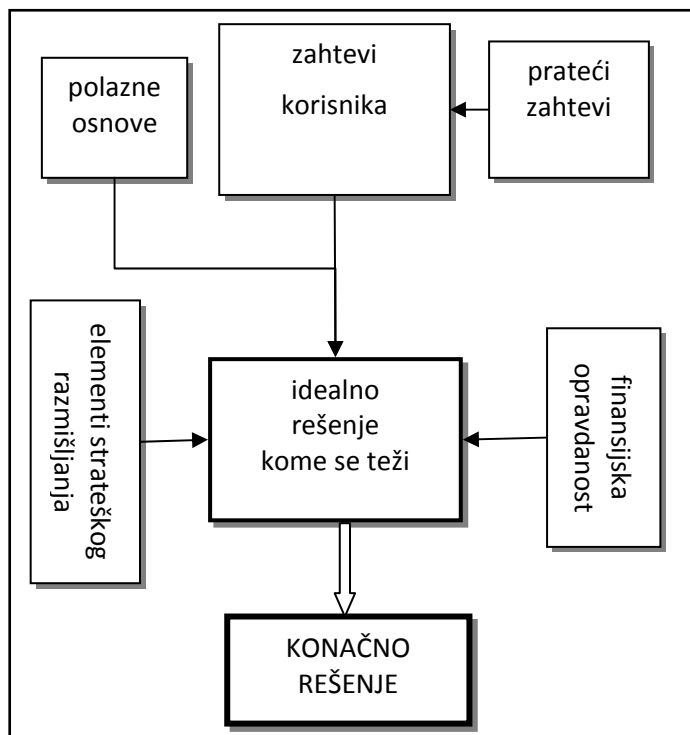
Pored toga, veoma važni aspekti u realizaciji nekog IT projekta su i:

- obuka osoblja, koja može biti kompleksna i višeznačna uz usvajanje novih pojmova, znanja, veština, uverenja, vrednosti, stavova i navika [7],
- osećaj zadovoljstva obavljenim poslom, kako investitora, tako i izvođača,
- uticaj na kvalitet uslova rada.

Najzad, ono što investitor očekuje od izvođača jeste i uspešna finansijska realizacija IT projekta. Ona podrazumeva da se u okviru raspoloživih sredstava: a) dobije najsavremenija i najefikasnija oprema, čija će aktuelnost biti što dugovečnija, a koja će, sa tim u vezi, pružati mogućnost određenog stepena nadogradnje i b) koja će se u određenom (doglednom) vremenskom periodu isplatiti.

Na osnovu iznetog, očigledno je da je strateško razmišljanje pri realizaciji IT projekata neophodno sprovoditi praktično na svim nivoima radi usklađivanja velikog broja zahteva. To za posledicu ima pravljenje velikog broja kompromisa a što pretpostavlja postojanje kompetentne radne snage.

Na Slici 2 dat je shematski prikaz osnovnih elemenata strateškog IT plana.



Slika 2. Struktura strateškog IT plana

Na kraju, navedimo i neke osobine koje poseduju pojedinci skloni strateškom razmišljanju od kojih se neke mogu vremenom naučiti i usvojiti:

- a) vizija,
- b) sklonost timskom radu,
- c) osećaj za odvajanje bitnog od nebitnog,
- d) razlikovanje poslova za individualno i kolektivno razmišljanje,
- e) organizacione sposobnosti,
- f) umeće pregovaranja,
- g) sposobnost povezivanja događaja i pojava,
- h) prosuđivanje i izvođenje zaključaka,
- i) čvrst karakter.

5. ZAKLJUČAK

Osnovni razlog primene strateškog i kreativnog razmišljanja u oblasti IT leži u izuzetnoj moći i značaju IT opreme. Njen značaj je kako upotrebn, tako i bezbednosni. On se svakodnevno povećava sa njenim usavršavanjem i širenjem broja oblasti u kojima se primenjuje. Zato je strateško razmišljanje na izvršnom nivou svakoga dana sve značajnije.

Sušтина je u činjenici da je u svakodnevnom radu neophodno praviti vešestruke kompromise, i to na svim nivoima. Na menadžersko-liderskom nivou to i ne treba posebno obrazlagati. Na operativnom nivou, međutim, to bi bilo sledeće: da oprema bude što jeftinija, što aktuelnija, da njena aktuelnost što duže traje (npr. u smislu mogućnosti nadogradnje), da njene performanse zadovoljavaju sve potrebe korisnika, da je obezbeđeno održavanje, da se dobro uklopi u enterijer, da se savetuju izvođači drugih radova (npr. građevinskih) o tome šta je potrebno obezbediti za IT opremu itd.

LITERATURA

- [1] A. Toffler, „The Third Wave“, *Bantam Books, USA*, 1980.
- [2] J. Petersen, „The Road to 2015“, *Waite Group Press, USA, CA*, 1994.
- [3] P. Carayon, P. Hoonakker, S. Marchand and J. Schwarz, “Job Characteristics and Quality of Working Life in the IT Workforce: the Role of Gender”, *Proceedings of the 2003 ACM SIGMIS CPR Conference, edited by E. Trauth, Philadelphia, Pennsylvania, pp.58-63, April 10-12, 2003.*
- [4] Strategic Thinking - A Task for All Employees, <http://www.sergaygroup.com/Smart-Talk/Strategic-Thinking.html>
- [5] Collins, James C. and Jerry I. Porras. “Organizational vision and visionary organizations.”, *California Management Review*, pp. 30-52, Fall 1991.
- [6] Sullivan, G. R., and M. V. Harper. “Hope is Not a Method”, *Random House, New York*, 1996.
- [7] D. Velat, R. Radić Dudić, “Aktivna obuka”, *DIAL, Beograd, str. 9, 2008.*

STRATEŠKO PLANIRANJE ICT-A U SKUPŠTINI CRNE GORE STRATEGIC PLANNING ICT IN THE PARLIAMENT OF MONTENEGRO

Vuko Perišić, Skupština Crne Gore

Sadržaj: Skupština Crne Gore je zainteresovana za razvijanje efikasnog, transparentnog i isplativog informacionog i komunikacionog sistema za sve korisnike (poslanike, službenike, građane, itd.) koji će koristeći najnovije standarde u ovoj oblasti biti model za druge parlamente u regionu. Naime, Skupština Crne Gore je zainteresovana za uspostavljenje elektronskog sistema za upravljanje „životnim ciklusom“ bilo kojeg dokumenta koji je sačinjen u Skupštini ili je dobijen po bilo kojem osnovu spolja, tj. put dokumenta otkad se prvi put pojavi u sistemu, pa dok se njegovo procesuiranje završi i arhivira. U cilju ostvarivanja navedenog, neophodno je kvalitetno ICT strateško planiranje, čiji je osvrt dat u ovom radu.

Abstract: The Parliament of Montenegro is interested in developing an efficient, transparent and cost effective information and communication system with all respective users (MPs, staff, citizens, etc.) using the latest standards in this area and to be the model for other parliaments in the region. Namely, the Parliament of Montenegro is interested in having an electronic system for managing the "lifecycle" of any document whether created in the Parliament or received from outside on any basis, i.e. the path of a document from the time of its first appearance in the system, until the completion of its processing and archiving. In order to achieve this, quality ICT strategic planning is necessary, whose review in this paper.

1. UVOD

Skupština Crne Gore ima zakonodavnu, nadzornu i kontrolnu funkciju. Njene glavne nadležnosti se odnose na usvajanje Ustava, Zakona i drugih akata, kao i nadgledanje aktivnosti i politike Vlade. To je jednodomni parlament sa 81 poslanikom.

Skupština radi u dva redovna zasijedanja. Prvo redovno zasijedanje Skupštine (prolječno zasijedanje) počinje prvog radnog dana u martu i traje do kraja jula. Drugo redovno zasijedanje (jesenje zasijedanje) počinje prvog radnog dana u oktobru i traje do kraja decembra. Tokom zasijedanja Skupština radi u sjednicama. Izvan ovih perioda, Skupština može da održi sjednice vanrednog zasijedanja.

Parlamentarna zakonodavna aktivnost je intenzivirana u poslednjih nekoliko godina. Tako je, u 2013 godini nastavljen trend pojačanih aktivnosti Skupštine Crne Gore. Tako je, u 2013 godini, donešeno ukupno 206 akata: amandmani na Ustav Crne Gore, 113 zakona, jedna rezolucija, 66 odluka, 24 zaključka i jedno mišljenje. Održane su ukupno 24 sjednice redovnog, osam sjednica vanrednog zasijedanja i jedna svečana sjednica Skupštine, u trajanju od 318 sata i četiri minuta, na kojima je razmotreno 179 tačaka dnevnog reda, kao i 369 sjednica radnih tijela, u trajanju od 691 sat i 30 minuta, na kojima je razmatrana

1.131 tačka dnevnog reda (među kojima 355 predloga zakona sa 2.071 amandmana) [1].

Skupština Crne Gore ima aktivnu ulogu u pristupnim pregovorima sa EU, koji su započeti na proljeće 2012. To znači da će broj zakona koji će se razmatrati rasti u narednim godinama, zajedno sa jačanjem parlamentarnih nadzornih aktivnosti.

Iz istih razloga, broj zaposlenih u Skupštini (zaposleni, organizovani kao Služba Skupštine, obezbjeđuju administrativnu, tehničku i ekspertsku podršku radu poslanika) će vjerovatno rasti, sa trenutnih 140 na 250 u narednih nekoliko godina.

U poslednjih nekoliko godina, mnogi parlamenti nastoje da uvedu nove ICT za poboljšanje parlamentarnih procesa, sa različitim rezultatima. Dok su neki postigli visok nivo uspjeha, drugi se znatno razlikuju u svojoj sposobnosti da podrže funkcije kroz upotrebu ICT-a. Tako se u svjetskom e-parlament izvještaju za 2010 godinu, rezultati ankete o korišćenju ICT-a u zakonodavnom procesu kreću između 14 do 83 od mogućih 100[2].

Pored toga, svi svjetski e-parlament izvještaji 2008, 2010 i 2012 podvlače činjenicu da je jačanje parlamenata kao transparentne, odgovorne i reprezentativne institucije kroz uvođenje i primjenu ICT-a veliki poduhvat. Uspjeh ovog

poduhvata u velikoj mjeri zavisi od zajedničkog angažovanja i podrške parlamentarnog rukovodstva, poslanika, drugih relevantnih funkcionera i parlamentarnog osoblja[2] [3] [4].

Neophodno je uskladiti jasnu viziju ovog zajedničkog angažovanja koje pružaju informaciono – komunikacione tehnologije sa najvišim ciljevima ustanove. Jedan od osnovnih alata, za menadžere da se kreću od vizije do cilja, je strateški proces planiranja. Tako, ICT strateški proces planiranja, treba da poveže viziju sa strateškim ciljevima i ciljevima programa, projekata, predloga, zadacima i aktivnostima članova, svih zainteresovanih strana. Proces strateškog planiranja ICT omogućava parlamentu da uspostavi prioritete i shodno tome raspoređuje resurse. Strateško planiranje omogućava da se prave kompromisi između konkurentnih potreba, uzimajući u obzir moguće posljedice tih odluka.

Na nedavnom 22. godišnjem zasjedanju Parlamentarne skupštine OEBS-a u Istanbulu, Skupština Crne Gore je prihvatila rezoluciju i deklaraciju da pozdravlja transparentnost i otvorenost kao ključ za jačanje povjerenja između parlamenata i birača [5].

2. CRNA GORA KAO INFORMACIONO DRUŠTVO

Crna Gora je rastuće informaciono društvo. Nacionalni ciljevi postavljeni od strane Ministarstva za informaciono društvo i telekomunikacije Crne Gore u vezi sa e-učešćem (npr. pristup i učešće nacionalne populacije na internetu) uključuju cilj je da se dostigne nivo penetracije korisnika interneta od 75% do 2015, i to 65% od osetljivih društvenih grupa treba da dobije informatičke pismenosti do kraja 2014, a 85% do 2016. Preduzeti su i koraci da se poveća e-učešće marginalizovanih grupa fokusirajući se na e-dostupnosti za sve i digitalnu pismenosti za sve.

Od strane Vlade, tokom 2012 godine, donijeta je odluka (Uredba o pravnom informacionom sistemu) prema kojima sve državne institucije, uključujući i Skupštinu Crne Gore su u obavezi da Službenom listu Crne Gore dostavljaju sve materijale elektronski potpisane.

3. ICT ORGANIZACIJA SKUPŠTINE CG

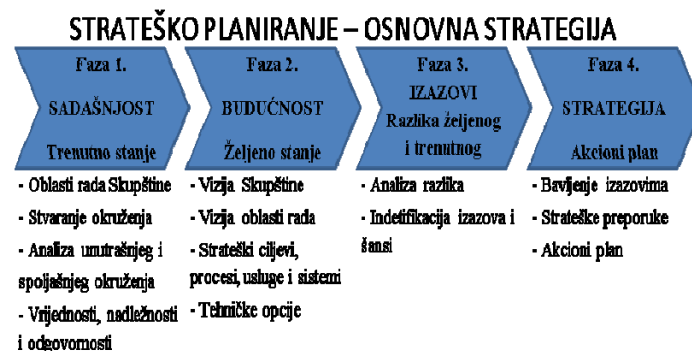
ICT organizacija u Skupštini Crne Gore, organizovana je u vidu Odsjeka pri Sektoru za istraživanje, dokumentaciju i informatičku mrežu. Odsjek za IT, pruža sledeće usluge:

- raspoređivanje i upravljanje mrežnom infrastrukturom
- obezbjeđivanje računara (uključujući i notebook i tablet uređaje)
- pružanje intranet/internet servisa (internet, mail, web,...).

ICT nema svoj redovan budžet, tako da nabavka hardvera i usluga zavisi od odobrenja viših instanci.

4. METODOLOGIJA STRATEŠKOG PLANIRANJA

Uprkos velikim poboljšanjima od 2006 godine, Skupština Crne Gore i dalje zakonodavni proces u najvećoj mjeri realizuje na klasičan način u papirnoj formi. U cilju prevazilaženja situacije neophodno je bolja i efikasnija primjena ICT-a, što uzrokuje ICT strateško planiranje.



Slika 1. Faze strateškog planiranja [6]

Na slici 1. prikazane su četiri faze strateškog planiranja, koje treba da daju odgovore na sledeća pitanja:

I faza: Gdje smo mi? Ili procjena sadašnjeg stanja ICT-a uključujući sve interne i eksterne faktore.

II faza: Gdje želimo da idemo? Ili, razumijevanje buduće ambicije, uključujući viziju i strateške ciljeve za ICT.

III faza: Koji su izazovi? Odnosno, procjena izazova kroz analizu.

IV faza: Kako ćemo krenuti napred? Što predstavlja, formulisanje strategije koja se bavi izazovima kroz preporuke i akcije, uključujući raspodjelu odgovornosti i resursa.

Dakle, strateški okvir ima sljedeće ciljeve:

- da se unaprijedi ICT organizacija, kapacitet i infrastruktura;
- da se obezbjedi detaljna analiza trenutnog stanja i budućih potreba;
- da se razjasne uloge i olakša učešće glavnih aktera;
- da se omogući usmjeravanje trenutnih oskudnih sredstava na ICT prioritete i definiše vremenski okvir za dodatne investicije;
- da se definišu “pravni okviri” – razvoj bezbjednosne politike za IS i set pravila i procedura važnih za IT system; i

- da se definiše neophodan nivo znanja i vještina kao i plan obuke za poslanike, zaposlene u Službi Skupštine i zaposlene u IT sektoru.

- slabe / ograničene IT vještine krajnjeg korisnika i mogućnostima:

5. STRATEŠKI PLAN 2014 – 2016

Strateški plan za ICT Skupštine Crne Gore je razvijen prema e-Parlament konceptu, koji se zasniva na međunarodnim dobrim praksama od strane drugih parlamenata širom svijeta na korišćenju i primeni ICT-a. Osim toga plan je razvijen kroz strukturiranu metodologiju strateškog planiranja i sastoji od tri dijela.

Prvi dio daje pregled trenutnog stanja ICT-a u parlamentu i njegovog uticaja na parlamentarni rad.

Drugi dio sagledava potrebe zainteresovanih strana i definiše viziju ICT-a u Skupštini Crne Gore.

Treći dio predlaže strateške ciljeve i realizaciju kroz konkretne akcije.

6. VIZIJA

U cilju prikupljanja strukturisanih informacija kako bi se obezbijedio jasan opis postojeće ICT infrastrukture, usluga i procesa dostupnih u svakoj funkcionalnoj oblasti, koristile su se metode i tehnike: interna istraživanja, intervjui/sastanci sa rukovodstvom i šefovima službi, upitnici i neposredni obilasci.

Kombinujući iznesene stavove i iskazane potrebe od strane rukovodstva, poslanika i skupštinske službe, može se formulisati vizija ICT-a u Skupštini Crne Gore: „ICT je **adekvatan** da omogući **povezanu, otvorenu, pristupačnu i efikasnu** Skupštinu Crne Gore, vodeći računa o zaštiti životne sredine“.

7. IZAZOVI I MOGUĆNOSTI

U ostvarenju gore navedene vizije parlament se suočava sa brojnim izazovima:

- slaba IT organizacija i ograničeno/nepoznato planiranje i mogućnosti implementacije,
- nepostojanje formalnih procedura za upravljanje IT
- nedostatak kritičnih kompetencija npr. analiza podataka / modeliranje, softver za upravljanje životnim ciklusom,
- nepotpuna / ne potpuno pouzdan IT mrežna infrastruktura,
- nedostatak upravljanje identitetom, odnosno sistematsko snimanje i profilisanje korisnika (npr. osoblje, poslanici), i

- projekat parlamentarnog informacionog sistema,
- parlamentarna pravila koja se mogu prilagoditi za korišćenje IT,
- sadašnje IT osoblje razumije procese parlamenta, i
- naučene lekcije i najbolje prakse iz drugih parlamenata
 - o ICT planiranja, koordinacije i upravljanja,
 - o XML otvoreni standardi,
 - o Upotreba mobilnih uređaja, i sl.

8. STRATEŠKI CILJEVI

Za svaki element vizije ICT-a razvijeni su specifični strateški ciljevi i zadaci:

Adekvatnost: gdje ICT usluge i infrastruktura su uvek isplativo i korisnički orijentisani (tehničke nadležnosti IT organizacije će biti ojačani, grupisani i mobilisani u skladu sa najboljom praksom i standardima);

Povezanost: svim članovima parlamenta, parlamentarnom osoblju i ostalim učesnicima u procesu omogućiti pristup iz bilo kog mjesta u bilo koje vrijeme (parlamentarne aplikacije i informacije će biti bezbjedno dostupne ovlaštenim korisnicima bilo kad i bilo gdje);

Otvorenost: svi dokumenti primljeni ili proizvedeni od strane Skupštine će biti dostupni u elektronskom obliku pogodnom za dalju obradu; parlament će usvojiti standardni format za zakonodavna dokumenta; građani će moći vidjeti sva dokumenta vezana za zakonodavni proces; transkripti sa sjednica će biti dostupni u otvorenom formatu; u komunikaciji sa građanima koristiće se i društvene mreže;

Pristupačnost: sve informacije je moguće lako pronaći i (re)upotrebljavati (ključni dokumenti potrebni članovima i osoblju parlamenta će biti organizovani u strukturu sa opisom metapodataka koji će olakšati njihovo pronalaženje; katalog biblioteke će biti dostupan online, a javni sajt će sadržati sve potrebne informacije u skladu sa smjernicama za parlamentarne sajtove)

Efikasnost: gdje se prate mjere i iskazuju svi parlamentarni procesi (svi relevantni parlamentarni poslovi će biti dokumentovani, a metrika oko troškova, performansi i dosljednosti će biti prikupljena i analizirana)

9. ZAKLJUČAK

Sa prvim ICT strateškim planom, koji se odnosi na period od 2014 do 2016 godine, Skupština Crne Gore ulazi u novu fazu razvoja, implementacije i upravljanja IT uslugama namijenjenim kako internim, tako i eksternim korisnicima. Opšti cilj, strategiskog plana je stvaranje ICT uslova za poboljšanje kvaliteta i održivosti parlamentarne demokratije u Crnoj Gori.

U okviru ICT strateškog okvira, planiran je razvoj posebnih projektnih zadataka za dizajn, razvoj i implementaciju parlamentarnog informacionog sistema. Projektni zadaci trebalo bi da iznesu detalje ciljeva projekta; neophodne usluge, uključujući modalitete implementacije; očekivane ishode i neophodni profil eksperata. Cilj je da IT sistem zakonodavne procedure bude u mogućnosti da postigne transparentnije, isplativije i brže procese. Trebalo bi da omogući pristup informacijama svim činiocima – poslanicima, službenicima, građanskom sektoru, medijima i građanima, kao i da omogući da proces donošenja odluka bude brži kroz informacioni sistem koji obezbjeđuje pristup

informacijama svim učesnicima u procesu bilo gdje i sa bilo kojeg uređaja.

LITERATURA

- [1] Skupština Crne Gore - Izvještaj o radu za 2013 godinu
- [2] The Global Centre for ICT in Parliament – “The World e-Parliament Reports 2010”
- [3] The Global Centre for ICT in Parliament – “The World e-Parliament Reports 2008”
- [4] The Global Centre for ICT in Parliament – “The World e-Parliament Reports 2012”
- [5] <http://www.neurope.eu/article/time-parliaments-commit-openness>
- [6] Manual for ICT Strategic Planning, Global Centre for ICT in Parliament, 2012

ANALIZA ONLINE KOMUNIKACIJE I INTERAKCIJE KROZ E-LEARNING ANALYSIS OF ONLINE COMMUNICATION AND INTERACTION THROUGH E-LEARNING

Željko Pekić, Stevan Kordić, Draško Kovač, Tatijana Dlabac, *Fakultet za pomorstvo Kotor*
Nađa Pekić, *Centar za socijalni rad Kotor, Tivar i Budva*

Sadržaj: *U ovom radu smo se prvenstveno bavili analizom online komunikacije i interakcije profesora i studenata u e-Learningu, kroz tri dimenzije. To su socijalna, didaktička i kognitivna. Svaka od dimenzija je sagledavana kroz svoje kategorije. Postulat informatičke pismenosti u e-Learningu predstavlja korišćenje programskih šema, kroz koje se na univerzalan način plasira gradivo, ocjenjuju radovi, komunicira na nivou profesor-student, doživljava kohezivnost ili individualnost u radu. Cilj nam je bio da ispitamo kakav pristup ovakvom načinu komunikacije imaju profesori, a kakav studenti. Rezultati pokazuju da je više nego dvostruko učestalije posjećivanje online kurseva od strane studenata.*

Abstract: *In this paper, we are primarily engaged in the analysis of online communication and interaction between teacher and studenata in e-learning, in three dimensions. These are social, and cognitive learning. Each dimension is observed through their categories. Postulate literacy in e-learning is the use of program schemes, through which the universal way of placing the material, graded papers, and communicates the professor-student level, perceived cohesion or individuality in the work. Our aim was to examine how access to this mode of communication with teachers and how students. The results show that more than twice as frequent visits to online courses by students.*

1. UVOD

Termin komunikacija potiče od latinskog glagola *communicare*, koji znači učiniti zajedničkim, odnosno saopštiti. Imenica *communicatio* koja je izvedena iz pomenutog glagola označava saobraćanje ili opštenje.

Kao jedna od dvije dileme koje su se prožimale kroz početak prošlog vijeka je pitanje da li se termin komunikacije u podjednakom smislu može vezati i za živa bića i tehničke sisteme. Miler je 1951. godine dao najšire određenje da je komunikacija prenošenje informacija sa jednog mjesta na drugo. Implikacija ovakvog određenja je da se komunikacija ne odnosi samo na čovjeka i druga živa bića, već i na djelovanje nekog sredstva koje služi kao posrednik. Petnaest godina kasnije (1967) Miler iznosi stav da komunikacija, ipak, nije tehnička pojava već aktivnost organizma. Tehnički sistemi se definišu kao sredstva kojima se ljudi pri komuniciranju koriste.

Danas, tehnologija je u velikom progresu i samim tim nam omogućuje drugačiji oblik nastave u odnosu na standardni, a uz to i nove korake u sveopštem obrazovnom sistemu. E-Learning platforma je najbolji indikator te vrste tehnologije.

2. PROCESI I KOMUNIKACIJE

Kreč i saradnici (1962) komunikaciju definišu kao razmjenu značenja među ljudima. MekDejvid i Harari (1974) smatraju da je komunikacija razmjena iskustva i ideja među

pojedincima. Ili kako smatra Pejčak (1975) da je to saopštavanje informacija drugim osobama o stvarima i pojavama.

U različitim definisanjima komunikacije postoje i različiti kriterijumi. Najčešće se koriste kriterijum interakcije, kriterijum sredstva i kriterijum razmjene. Ukoliko je kriterijum interakcija, komunikologija se definiše kroz sam proces komunikacije. U definisanju komunikacije, kod kriterijuma sredstva, koriste se forma, medijum i medij. Forma, kao način na koji se nešto saopštava, medijum kao kombinovanje različitih sredstava i medij kao tehnološko sredstvo masovne komunikacije. Kriterijum razmjene za osnovu ima razmjenu na međuljudskom nivou.

U elemente komunikacija spadaju: izvor, poruka i recipijent. U e-Learningu izvor je profesor, poruka je sadržaj ili materijal koji eminent (student) treba obraditi. U procesu komunikacije postoje dva procesa: enkodiranje (oblikovanje) i dekodiranje (obrađa) informacija.

Svakodnevna dostignuća u oblasti informacionih tehnologija su dovela do velikog progressa obrazovni sistem, i to na svim nivoima obrazovanja. Može se reći da je time ojačan i učvršćen svojevrsan standard na akademskom nivou. Osim što se svakodnevno povećava broj informacija, modifikuje se i način obrade istih. Na taj način se i pospešuju vještine i dijapazon načina usvajanja znanja. Za savremeni vid nastave se često kaže i „brži oblik učenja”.

Ono što je relevantan podatak je to da u današnje vrijeme za bilo koji opis posla se neizostavno traži poznavanje rada na

računaru. I to ne poznavanje osnovnih računarskih operacija, već poznavanje posebnih programa predviđenih za određenu naučnu oblast.

3. PROCESI INTERAKCIJE

U psihologiji interakcija je odnos dvije ili više osoba, koji dovodi do međuzavisnosti u njihovom ponašanju ili doživljavanju, a interakcija u filozofiji veže se za mišljenje o prirodnim i društvenim fenomenima, kao uzajamno povezanim uzrocima.

U tehničkim naukama postoje fundamentalne interakcije, a dijele se na gravitacijske, elektromagnetske, jake, slabe... Interaktivan u informatici je onaj koji se odlikuje dvosmjernom komunikacijom između korisnika i računara. Termin dolazi od latinske riječi *inter*, kao prvog dijela riječi, koji znači da se nešto nalazi između djelova onoga što je rečeno drugim dijelom, ili između toga i drugog čega (*inter* može biti i prvi dio riječi vezane za nešto internacionalno), te latinske riječi *actio* - agere, a znači djelovati.

4. DIMENZIJE ONLINE INTERAKCIJE I NJIHOVE KATEGORIJE

Prema Andersonu (2001) postoje tri osnovne dimenzije analize online komunikacije i interakcije, a to su: socijalna, didaktička i kognitivna. Socijalna dimenzija podrazumijeva sve izjave studenata i održavanje kohezije grupe, zajedničke dinamike, timskog rada, društvenih odnosa, emocionalne interakcije.

Socijalna dimenzija se analizira kroz tri kategorije (Gunawardena, 1997):

- afektivnu (emocije i smisao za humor);
- interaktivnu (citiranje drugih, pravljenje pitanja) i
- kohezivnu (grupa govori o nama).

Didaktička dimenzija se odnosi na korišćenje prilike da profesor usmjeri učenje studenta u virtuelnom okruženju, da pospješi komunikaciju, postavlja pitanja, izrazi ideje, odgovori na pitanja. Potrebno je analizirati dimenziju sa obrazovne tačke gledišta. Anderson sugerira tri glavne uloge profesora:

- kao dizajnera nastave;
- kao planera i ocjenjivača i
- kao facilitatora.

Posljednja dimenzija je kognitivna. Ona se odnosi na konstruisanje i potvrdu održivog diskursa u zajednici kritičkog istraživanja. Odnosi se na visoki nivo kritičkog razmišljanja iz perspektive pojedinca. To uključuje i kreativnost, rješavanje problema, intuiciju...

5. CILJEVI ISTRAŽIVANJA

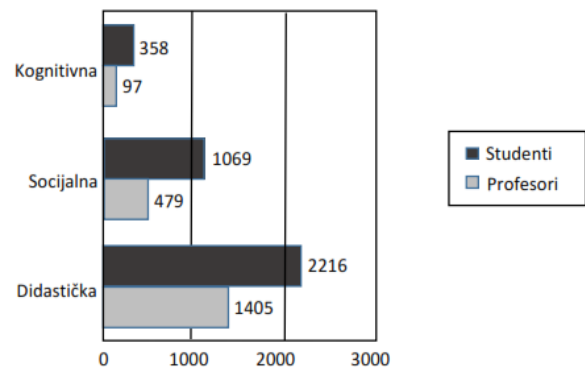
Polazni predmet istraživanja je sagledavanje prirode doživljaja online učenja, od strane profesora i studenata, kroz tri komponente: kognitivnu, socijalnu i didaktičku.

Naučni cilj je utvrditi kako se profesori i studenti odnose prema online učenju, kroz te tri dimenzije.

Praktični cilj je da dobijeni podaci doprinesu efikasnijem i praktičnijem radu u obrazovnim ustanovama koje funkcionišu preko e-Learninga.

6. ANALIZA REZULTATA ISTRAŽIVANJA

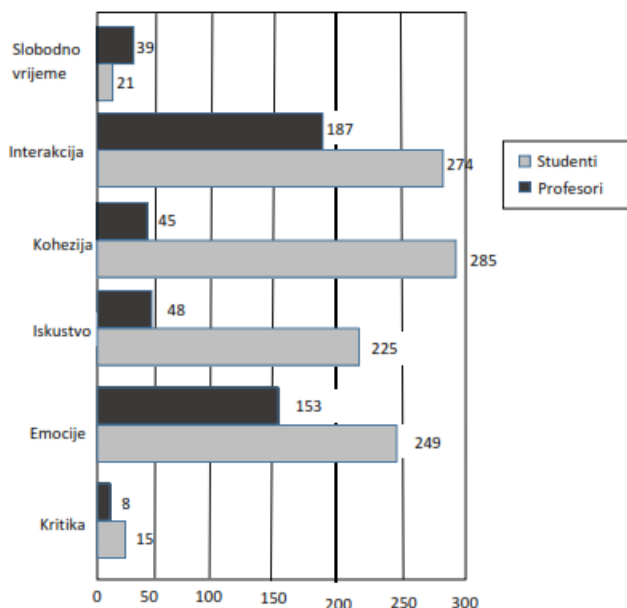
U nastavku će biti prvo prikazani rezultati sve tri dimenzije zajedno, a zatim svaka kategorija svake dimenzije ponaosob. U istraživanju su korišćeni instrumenti konstruisani od strane autora rada. Za socijalnu dimenziju Alfa Krombah koeficijent pouzdanosti iznosi 0,67, didaktičku 0,71 i kognitivnu 0,65. Uzorak sačinjava 154 studenta Fakulteta za pomorstvo u Kotoru i 27 profesora i saradnika u nastavi iz iste institucije.



Slika 1. Frekvencija poruka od strane profesora i studenata

Na slici 1 se jasno vidi da u opštoj frekvenciji slanja onlajn poruka na e-Learning kursevima veću frekventnost slanja poruka imaju studenti. Učestalost slanja poruka je 2-3 puta veća kod studenata u odnosu na profesore. To implicira činjenicu da ovakav stil nastave profesori doživljavaju kao fleksibilniji u odnosu na aktivni što se tiče vremena, i na taj način i pristupaju online kursevima rjeđe. To mimoilaženje u vremenu jeste i jedan od većih nedostataka ovakvog tipa nastave. Npr. student kojem se žuri za određenu informaciju (za koju je postavljeno online pitanje) treba da čeka odgovor sve do momenta kada profesor odgovori na to, a zatim i kada taj isti student opet dodje do računara i potraži taj odgovor. To može djelovati inhibirajuće na motivaciju studenta.

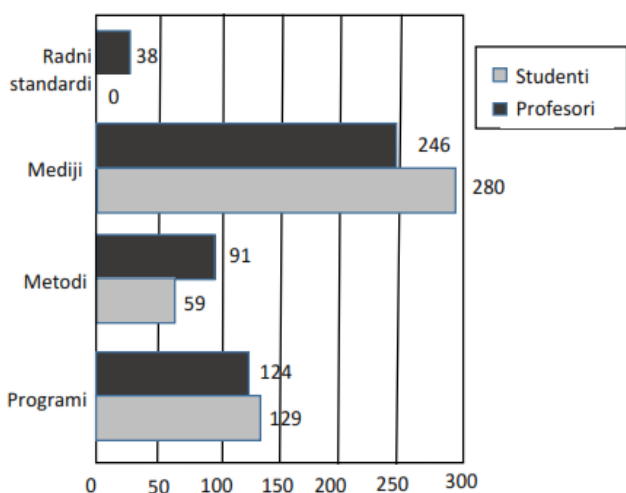
U okviru socijalne dimenzije ajtemi su regulisani kroz doživljaj kohezije. Ova kategorija obuhvata one intervencije koje se odnose na grupni identitet, kroz izraze *nama/vama*, *grupa*, *drugovi*.



Slika 2. Frekvencija poruka u socijalnoj dimenziji

Na slici 2 se vidi frekvencija poruka u socijalnoj dimenziji. Najdrastičnija razlika je doživljaj kohezivnosti profesora i studenata, gdje studenti više od pet puta imaju jači doživljaj grupe. Ako se vodimo time da je Moodle platforma konstruisana tako da pravi nastavu kao proces u kojem svaka karika lanca koja učestvuje ima jednaku važnost u sistemu, onda to znači da je potrebno da svaka od njih ima doživljaj zajedništva tog procesa. To se manifestuje kroz osjećaj kooperativnosti, sudjelovanja i interakcije.

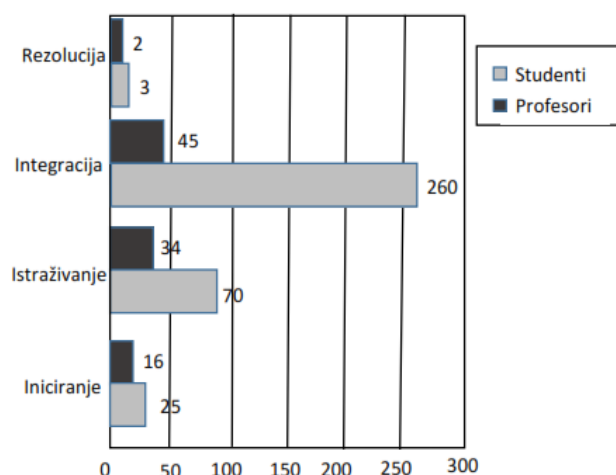
Didaktička dimenzija se odnosi na instrukcijski dizajn, kao i na intervencije aspekata planiranja, organizovanja i upravljanja kursevima.



Slika 3. Frekvencija poruka u didaktičkoj dimenziji

Na slici 3 su prikazani rezultati didaktičke dimenzije. Značajan je podatak da apsolutne radne standarde postavljaju profesori, dok su rezultati kod kategorije programa izjednačeni.

Koncept kognitivne dimenzije koristimo za analizu kognitivne prisutnosti, povezane s izgradnjom znanja u obrazovnom kontekstu na online forumima. Ovaj model se sastoji od četiri osnovna koraka, od kojih svaki pokazuje jedan deskriptor odražavajući opštu kondiciju. Kao što smo naveli, koncept kognitivne prisutnosti bi trebalo da bude podjednako stimulativan i za profesore i za studente, bez obzira na obrazovni nivo.



Slika 4. Frekvencija poruka u kognitivnoj dimenziji

Integracija kao kategorija, sama po sebi je u pozitivnoj korelaciji sa doživljajem kohezivnosti, te stoga nije čudan ovakav rezultat.

7. ZAKLJUČAK

Cjelokupni rezultati ukazuju na različitu frekvenciju prisutnosti online komunikaciji kod profesora i studenata putem e-Learninga. Zanimljiv je podatak dobijen kroz istraživanje da u virtuelnom okruženju i uobičajnoj nastavi se potpuno razlikuje pristup nastavi. Kod virtuelnog okruženja studenti su ti koji češće posjećuju online kurseve, započinju rasprave i postavljaju pitanja. Mada profesori su ti koji postavljaju apsolutne radne standarde. Poznato je da bi e-obrazovanje ili obrazovanje na daljinu trebalo da podrazumijeva da je glavni nosilac komunikacije između predavača i studenta razdvojenost (u različito vrijeme i na različitom mjestu – razdvojenost instruktora - tutora od studenta). Ali to mora da obuhvati uspostavljenu dvosmjernu komunikaciju između predavača i studenta koja ima za cilj da olakša i podrži proces edukacije.

Sve to skupa implicira, praktični cilj, da virtuelno okruženje u obrazovnim institucijama treba da stimuliše aktivnost studenata i njihovu međusobnu saradnju, sudjelovanje, komunikaciju i interakciju, ali da profesori budu jednakim dijelom ekscitatori, kao i studentova motivacija. Suština je postizanje konzistencije obostranih interesa, bez obzira na hijerarhiju.

Ovakvi oblici istraživanja predstavljaju plodno tlo za nastavak daljih istraživanja u području e-Learninga. Na taj

način saznajte više o takvim komunikacijskim procesima i u kom smislu takvi diskursi teku u virtuelnim okruženjima.

LITERATURA

- [1] K. D., K. R. S., B. I., “Pojedinac u društvu. Beograd”, *Zavod za udžbenike i nastavna sredstva*, 1962.
- [2] M. J. W., H. H., “Psychology and Social Behavior”, *Harper & Row: ASIN: B003U7OPY2*, 1974.
- [3] M. G. A., “Language and communication”, New York: McGraw-Hill, 1951.
- [4] G. C., L. C., A. T., “Analysis of a global online debate and the development of an interaction model for examining social construction of knowledge in computer conferencing”, *en Journal of Educational Computing Research*, 17, 397-431, 1997.

ITIL - IMPLEMENTACIJA INCIDENT MANAGEMENT PROCESA U SERVICE DESK-U SA PREDLOGOM ZA NJEGOVO UNAPREĐENJE ITIL - IMPLEMENTATION OF THE INCIDENT MANAGEMENT PROCESS IN THE SERVICE DESK WITH PROPOSAL FOR IMPROVEMENT

Dejan Abazović, *Centralna banka Crne Gore*
Budimir Lutovac, *Elektrotehnički fakultet*

Sadržaj: *U radu se opisuje implementacija Service Desk-a (Help Desk-a) po ITIL v3 2011 frameworku koji definiše šta treba da se uradi unutar organizacije za određene procese. Kao alat za implementaciju Service Desk-a po ITIL frameworku koristi se softversko rješenje ManageEngine: ServiceDesk Plus. Implementacija se bazira na ITIL procesu Incident management iz Service Operation faze. Dodatno, predložene su nove podprocedure i indikatori performansi u cilju poboljšanja standardnog modela.*

Abstract: *The aim of this paper is to describe the implementation of the Service Desk (Help Desk) ITIL v3 2011 framework that defines what needs to be done within the organization for certain processes. Software solution ManageEngine: ServiceDesk Plus is used as a tool for the implementation of Service Desk by ITIL Framework. The implementation is based on ITIL Incident Management process from the Service Operation phase. In addition, new key activities and key performance indicators have been proposed in order to improve the standard model.*

1. UVOD

Da bi informacioni sistem pružio adekvatnu podršku poslovnim procesima organizacije, obezbijedio potreban protok podataka i omogućio obradu raličitog tipa informacija, neophodno je pridržavati se propisanih procedura. Navedene procedure treba da obezbijede kvalitetne servise i isporuku usluga korisnicima kroz dobro implementirane, prilagođene i predefinisane procese rada.

Service Desk (Help Desk) kao servis za podršku rada korisnika se pokazao neophodna funkcija svakog poslovnog okruženja čiji rad je podržan IT-em. Osim što upravlja pregled funkcionisanja sistema i servisa, korisnicima daje mogućnost da direktno komuniciraju sa IT-em prijavljujući primjećene smetnje i prekide. Sa druge strane Service Desk IT stručnjacima prikupljaju potrebne informacije za rješavanje prijavljenog problema kao što pruža osnovne smjernice za definisanje koraka i procedura pri radu[11].

Jedan od načina za uređenje rada Service Desk-a je implementacija Information Technology Infrastructure Library (ITIL) frameworka. Incident management proces, po ITIL v3 2011 frameworku, predstavlja prvi i ključni proces koji je potrebno implementirati u Service Desk-u[6].

2. ITIL V3 2011 FRAMEWORK

ITIL je skup procesa za upravljanje Information Technology Service Management-om (ITSM), a koji se fokusira na usklađivanje IT servisa sa poslovnim potrebama[9]. ITIL opisuje "best practice" pristup za ITSM i pruža podršku organizacijama da razviju odgovarajući framework za ITSM, bez da daje prijedloge o korištenju konkretnih alata ili infrastrukture, nego se bazira na definisanju i pojašnjenju procesa koji treba da se

implementiraju. ITIL, pri tome, ne nudi gotove procese, nego daje generalnu sliku procesa koji mogu da se prilagode individualnim potrebama krajnjeg korisnika.

Ključni segment ITIL-a je pružanje IT podrške krajnjem korisniku sistema. Najefikasniji i najjednostavniji način IT servise podrške je uvođenje i korištenje nekog Service Desk-a, a koji je baziran na ITIL procesima. ITIL v3 2011 se sastoji iz 5 ključnih faza:

- Service Strategy,
- Service Design,
- Service Transition,
- Service Operation,
- Continual Service Improvement.

Svaka od ovih faza ima svoju grupu procesa koje mogu biti nezavisno implementirane. Na taj način mogu se dobro prilagoditi potrebama i okruženju u kojem se implementiraju. Kreirane su tako da mogu pratiti promjene tj. poslovne zahtjeve koje se dešavaju u okruženju.

Service Strategy pruža smjernice i ciljeve servisa, određuje strategiju za implementaciju Service Managementa. Service Design određuje dizajn za sprovođenje strategije, Service Transition opisuje kako service treba pustiti u produkcijsko okruženje, te Service Operation koji objašnjava kako treba da se implementira razvijeni servis. Na kraju, Continual Service Improvement obuhvata sve prethodno navedene oblasti. Ova faza pruža podršku pri kontinuiranom poboljšanju procesa, a tome podliježu svi procesi i aktivnosti koji prate poslovni razvoj i nove korisničke i procesne zahtjeve.

Da bi se servis poboljšao i izmijenio prema zahtjevima, životni ciklus servisa počinje ponovo sa fazom Service Strategy i to predstavlja kružni tok razvoja servisa. U srcu životnog ciklusa servisa, ključni princip je – svi servisi

moraju da pružaju mjerljive vrijednosti za poslovne ciljeve i rezultate.

ITIL Service Management se fokusira na mjerljive poslovne vrijednosti kao primarni cilj. To je postalo veoma važno danas kada IT organizacije moraju same da djeluju kao kompanije da bi pokazale jasan povraćaj ulaganja u njih.

3. MANAGEENGINE: SERVICEDESK PLUS

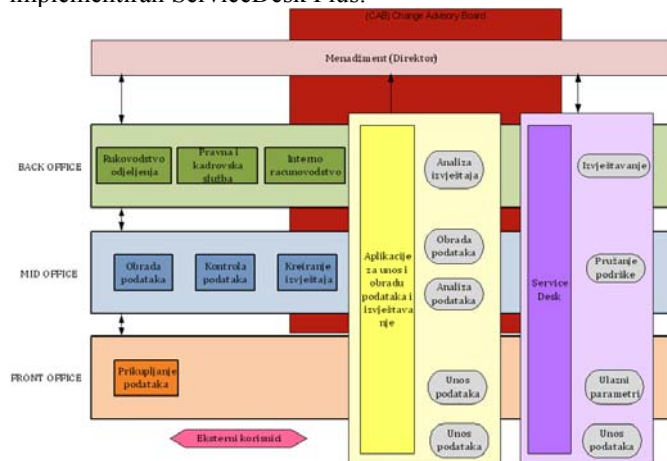
ManageEngine: ServiceDesk Plus je sveobuhvatan help desk IT softver za upravljanje sredstvima i inventarom kao i radom krajnjih korisnika. ServiceDesk Plus pruža serviserima podršku a IT manager-ima integrisanu konzolu za praćenje, nadgledanje i održavanje sredstava i inventara. Takođe, ovaj softver upravlja zahtjevima koje generišu korisnici IT resursa u organizaciji[5].

ServiceDesk Plus je web bazirano rješenje što znatno olakšava korištenje alata krajnjim korisnicima, a predstavlja direktan kontakt sa korisnicima sistema. Korisnici mogu biti interni, odnosno svi zaposleni, ili eksterni, odnosno svi ostali korisnici koji imaju pristup određenom dijelu sistema koristeći za to prethodno definisane načine pristupa sistemu. Na taj način, krajnjem korisniku je jednostavnije pružiti što kvalitetniju i što bržu podršku pri osiguravanju funkcionisanja njemu potrebnih IT servisa. ServiceDesk Plus radi i na Windows i Linux platformi, te podržava više ITIL procesa kao što su: Incident Management, Problem Management, Change Management, Service Catalog i Configuration Management Database (CMDB). U ovom radu detaljnije će biti razmatran.

U ovom radu ServiceDesk Plus će biti korišten kao alat za implementaciju Incident Management-a u Service Desk-u po ITIL v3 2011 frameworku. Implementacija istog će biti prikazana na referentnom modelu finansijske organizacije, koji će prethodno biti ukratko opisan.

4. IMPLEMENTACIJA SERVICEDESK PLUS-A NA REFERENTNOM MODELU

U nastavku će biti predstavljena struktura referentnog modela finansijske organizacije (Slika 1), na kojoj će biti implementiran ServiceDesk Plus.



Slika 1: Referentni model finansijske organizacije

Ključni, a ujedno i primarni, procesi svake finansijske organizacije su prikupljanje finansijskih podataka, obrada

prikupljenih podataka i kontrola prikupljenih i prijavljenih podataka.

Na osnovnom nivou Service Desk predstavlja sredstvo komunikacije korisnika sa IT podrškom sistema. Evidentirani problemi se prijavljuju i evidentiraju u Help Desk aplikaciju.

Na srednjem nivou funkcija Service Desk-a je da analizira prijavljene zahtjeve i da ih riješi, što znači da je ovo mjesto osnovne operativne funkcije aplikacije koja podržava rad Service Desk-a.

Na najvišem nivou Service Desk vrši izvještavanje o stanju sistema i prosljeđuje zahtjeve za izmjene da bi dobio odobrenje za implementaciju. Na tom nivou se, u saradnji sa Change Advisory Board-om (CAB) i krajnom odlukom menadžmenta, donose ključne, strateške odluke i rješavaju se problemi koji su nastali kao rezultat problema u krovnim procesima, krovnim softverskim ili hardverskim komponentama.

IT profesionalci, koji su zaduženi za održavanje sistema i pružanje podrške korisnicima, su uglavnom zaduženi za prva dva nivoa. Pored njih, veoma važani za IT podršku su i spoljašni saradnici organizacije. Pod tim se podrazumijevaju organizacije koje su dobavljači softvera i hardvera, odnosno isporučiocin nekog softverskog rješenja, koji igraju važnu ulogu u IT podršci korisnicima sistema za one dijelove sistema koji su u njihovoj nadležnosti.

Pri rješavanju zahtjeva podnešenog Service Desk-u veoma je važno pridržavati se postupaka definisanih pratećim procedurama da bi se mogao pratiti tok zahtjeva i vrijeme rješavanja zahtjeva.

Prema hijerarhiji poslova, za svaku vrstu i nivo zahtjeva, određena je odgovorna osoba koja pokušava da riješi problem. Ukoliko to ne uspije, uz obrazloženje, zatražit će pomoć od odgovorne osobe na višem nivou hijerarhije poslova.

Svaki korisnik IT sistema (interni i eksterni) na svim nivoima ima pristup aplikaciji ServiceDesk Plus. Preko svog profila pristupa dozvoljenom nivou informacija. Na taj način, on može da vidi podnesene zahtjeve, dodijeljene zahtjeve, izvještaje, prati status prijavljenog problema, kao i da pročita dato rješenje. Na isti način, IT serviser može da vidi sve njemu dodijeljene zahtjeve i da tim putem komunicira sa korisnikom koji je prijavio smetnju ili prekid.

Incidenti i problemi se prijavljuju kreiranjem novog zahtjeva kroz aplikaciju, ili slanjem maila na prethodno definisanu e-mail adresu koji je direktno vezan na ServiceDesk Plus, i koji automatski kreira novi zahtjev. Svaki zahtjev koji se kreira, odgovorni serviser pregleda i delegira na odgovornu osobu koja dalje rješava prijavljeni problem.

U zavisnosti od prirode, nivoa i lokacije problema, problem se delegira na odgovarajući nivo hijerarhije u strukturi organizacije. Worklow dodjeljivanja i rješavanja incidenata svaka organizacija definiše i kreira za sebe obuhvatajući njime sve svoje potrebe i zahtjeve.

5. IMPLEMENTACIJA SERVICE DESK-A PO ITIL FRAMEWORKU

Osnovni cilj Incident managementa je da korisniku, koji je prijavio incident, što je prije moguće, ponudi trenutno ili

trajno rješenje za nesmetan rad. Incident management, uz Problem management iz Service Operation faze i Change management iz Service Transition faze, predstavlja jedan od tri ključna procesa pri otklanjanju grešaka i smetnji na sistemu[11]. Svaki od ova tri procesa ima podproces, specifične ključne aktivnosti i ključne indikatore performansi, tj. Key Performance Indikatore (KPI).

Za Incident managementa, po ITIL-u postoji unaprijed definisani workflow, sa sledećim podprocesima[6]:

- Identifikacija i evidentiranje incidenta,
- Kategorizacija incidenta,
- Prioritetizacija incidenta,
- Inicijalna dijagnoza,
- Eskalacija incidenta,
- Istraga i dijagnoza,
- Rješenje i oporavak,
- Zatvaranje incidenta.

Skup realnih ključnih aktivnosti koje treba da se sprovedu unutar organizacije za neki servis po ITIL-u su[6]:

- Detekcija i evidentiranje incidenta,
- Klasifikacija incidenta,
- Inicijalna podrška korisnicima koristeći Service Desk,
- Istraživanje i dijagnoza,
- Rješenje i oporavak servisa,
- Zatvaranje incidenta,
- Vlasništvo nad incidentom, nadgledanje i komunikacija.

Kao ključni faktor dobrog upravljanja procesima je uspostavljanje tzv. SMART metrike koja govori kakav KPI treba da bude[7]:

- Specific – određen, konkretan,
- Measurable – mjerljiv,
- Achievable – ostvariv,
- Realistic – realan,
- Time Driven – vođen vremenom.

Da li su te ključne aktivnosti dobro implementirane utvrđuje se preko ključnih indikatora performansi (KPI). Svaka organizacija, pored preporučenih KPI-eva može da definiše za sebe specifične KPI-eve koje prilagođava svojim potrebama i okruženju. Ukoliko ti KPI-evi budu zadovoljavajući znači da je implementacija bila uspješna. Ključnih indikatora performansi (KPI) po ITIL-u su [8]:

- Broj prijavljenih incidenta,
- Broj incidenta koji se ponavljaju, a za poznato rješenje,
- Broj incidenta koji mogu da se riješe udaljenim pristupom,
- Broj eskalacija,
- Prosječno vrijeme čekanja na odgovor na zahtjev,
- Prosječno vrijeme rješavanja incidenta,
- Rješavanje incidenta odmah nakon prijavljivanja,
- Rješavanje incidenta u okviru SLA-a,
- Uloženi trud u rješavanje incidenta.

6. IMPLEMENTACIJA INCIDENT MANAGEMENT-A KORIŠTENJEM SERVICEDESK PLUS-A

ServiceDesk Plus podržava ITIL v3 2011 Incident management. Koraci Incident managementa u ServiceDesk Plus su [10], [11]:

a) Prepoznavanje incidenta

Incident se prijavljuje putem e-maila, telefona ili web interfejsa na ServiceDesk Plus.

b) Evidentiranje incidenta i detalja incidenta

Na ServiceDesk Plus-u incident može da se prijavi, evidentira i precizira sa što je moguće više detalja vezanih za nastanak incidenta. Na taj način, serviserima za podršku je jednostavnije klasificirati incident i proslijediti odgovornoj osobi na rješavanje.

c) Kategorizacija i razvrstavanje incidenta

Kategorizacija incidenta je uvijek zavisi od organizacije na koju se odnosi. To podrazumijeva da se definiše lista kategorija, odgovarajućih podkategorija i komponenata koji pripadaju toj podkategoriji (Slika 2).

Slika 2: Izbor kategorije i komponenti incidenta[5]

Prioritet rješavanja incidenta se određuje prema matrici uticaj/hitnost. Uticaj predstavlja vrijednost koliko incident utiče na poslovanje, odnosno koliko je krajnjih korisnika time ugroženo, a hitnost, u zavisnosti od SLA (Service Level Agreement), određuje se prema tome koji je servis ugrožen (Slika 3).

Slika 3: Izbor uticaja, hitnosti i prioriteta incidenta[5]

ITIL kaže da Prioritetizacija treba da bude odnos Impact/Urgency matrice. Jedan od najjednostavnijih načina određivanja prioriteta na osnovu uticaja i hitnosti prikazan je u Tabeli 1:

PRIORITY		Time to resolve [h]	IMPACT		
1	2	2	High	Mid	Low
1	2	2	High	Mid	Low
2	8	8	High	2	3
3	24	24	Mid	2	3
4	36	36	Mid	3	4
5	160	160	Low	3	4
			Low	4	5

Tabela 1: Primjer definisanja prioriteta zahtjeva[12]

Za svaki incident koji bi se mogao ponovo javiti, serviser koji je riješio isti, može da upiše rješenje u bazu podataka ServiceDesk Plus-a kao preporučeno rješenje, koje prethodno administrator revidira i odobri.

d) Prijedlog rješenja

Odgovorna osoba vrši inicijalnu analizu incidenta i uspostavlja inicijalnu dijagnozu (Slika 4).

Slika 4: Dodjeljivanje zahtjeva grupi i IT stručnjaku[5]

Nakon analize incidenta, tehnička podrška može da ponudi trajno rješenje prijavljenog incidenta ili neki workaround. Kada se javi poznat incident, serviser za podršku može da pošalje korisniku predefinisano rješenje, ali i sam korisnik može da pretražuje bazu podataka te na taj način sam nađe odgovor na incident. Serviser koji je riješio isti, može da upiše rješenje u bazu podataka ServiceDesk Plus-a kao preporučeno rješenje, koje prethodno administrator revidira i odobri.

e) Eskalacija incidenta

Ukoliko incident nije moguće otkloniti, incident se eskalira na viši nivo odgovornosti da bi se našlo, što je moguće prije, adekvatno rješenje. U grupu serviseru za podršku moguće je dodati i eksterne saradnike kojima se incidenti mogu delegirati te time smanjiti vrijeme objašnjavanja incidenta i ubrzati proces otklanjanja incidenta. U slučaju eskaliranja incidenta, važno je pridržavati se predefinisanih SLA pravila.

f) Zatvaranje incidenta

Incident se zatvara kada se nađe rješenje za nastali incident. Krajnji korisnik treba da potvrdi da li je zadovoljan ponuđenim rješenjem i da li je to rješenje otklonilo njegov incident. Korisnik će putem e-maila biti obaviješten o toj aktivnosti, te ukoliko se ne slaže sa tim da se zahtjev treba zatvoriti, može putem aplikacije ponovo otvoriti zahtjev i poslati serviseru svoje primjedbe.

Rezultati mjerenja prema definisanim KPI-evima, za period od 7 dana, prikazani su u Tabeli 2:

Broj prijavljenih incidenata	48
Broj incidenata koji se ponavljaju sa poznatim rješenjem	14
Broj incidenta koji mogu da se riješe remote pristupom	9
Broj eskalacija	0
Prosječno vrijeme čekanja na odgovor na zahtjev	00:20
Prosječno vrijeme rješavanja incidenta	04:50
Rješavanje incidenta odmah nakon prijavljivanja	32
Rješavanje incidenta unutar SLA-a	0
Uloženi trud u rješavanje incidenta	n/a

Tabela 2: Rezultat KPI-eva na referentnom modelu

7. ZAKLJUČAK

U radu je prikazana implementacija Incident management procesa po ITIL v3 2011 frameworku, teoretski i na konkretnom primjeru referentnog modela jedne finansijske organizacije primjenom softverskog alata ManageEngine: ServiceDesk Plus. Ovaj proces je implementiran kroz devet podprocesa koji korisnika vode korak po korak, tako Service Desk bude apsolutno prilagođen poslovnim procesima organizacije, te adekvatnim prioritetima procesa, kao i redosljedu rješavanja incidenata.

Njegov kvalitet leži u činjenici da su incidenti detaljno i precizno definisani i da se predlaže uspostavljanje Service Desk-a kao Single Point of Contact za sve incidente.

Tokom procesa implementacije, ustanovljeno je da:

- a) unutar ITIL v3 2011 Incident Management procesa nije definisan konkretan podproces kojim bi se definisale osnovne karakteristike Service Desk funkcije;
- b) je neophodno dodati podproces "Izveštavanje i analiza trendova", koji je pokriven nekim drugim dijelovima ITIL-a, kojim bi se unaprijedilo praćenje prijavljenih incidenata, incidenta koji se ponavljaju i uzroka javljanja incidenata.

Analizom dobivenih rezultata KPI-eva, može se vidjeti da je većina KPI-eva mjerljiva, na osnovu izvještaja koje nudi ServiceDesk Plus[1],[2],[3],[4]. Isto tako, utvrđeno je da se mjerenje efikasnosti implementacije može unaprijediti dodavanjem dva nova KPI-a:

- a) zadovoljstvo korisnika sa podrškom, i
- b) broj neriješenih zahtjeva.

LITERATURA

- [1] Antti Lahtela and Marko Jantti, Jukka Kaukola: „Implementing an ITIL-based IT Service Management Measurement System“, *IEEE Fourth International Conference on Digital Society - ICDS '10*, pp: 249-254, St. Maarten, 2010
- [2] Fernando B. A., Jorge M. F., Raquel de Bragança V. da P., José C. C.: „Definition and Validation of Metrics for ITSM Process Models“, *IEEE Seventh International Conference on the Quality of Information and Communications Technology, QUATIC*, pp: 79-88, Porto, 2010
- [3] A. S. Lima, J. Neuman de Souza, A.C. Callado, J. Augusto Oliveira, Jacques Sauve, Antao Moura: „A Business-Driven IT Services Improvement Model“, *4th IFIP/IEEE Workshop on Distributed Autonomous Network Management Systems*, pp: 1095 – 1102, Dublin 2011.
- [4] Ben Clacy and Brian Jennings: „Service Management: Driving the Future of IT“, *Computer*, Volume:40, Issue: 5, pp: 98-100, may 2007
- [5] <http://www.servicedeskplus.com/>
- [6] http://www.itlibrary.org/?page=Incident_Management
- [7] http://www.ucisa.ac.uk/~media/Files/members/activities/ITIL/service_operation/incident_management/ITIL_IM%20KPIs%20and%20reports%20pdf
- [8] http://wiki.en.it-processmaps.com/index.php/ITIL_KPIs_Service_Operati on#ITIL_KPIs_Incident_Management
- [9] Jan van Bon: *Foundation IT Service management based on ITIL v3*, Van Haren Publishing, 2017.
- [10] Alex D. Paul, *ITIL Heroes Handbook*, www.manageengine.com
- [11] ARINC Managed Services: <http://www.arincmanagedservices.com/blog/2010/11/se rvice-desk-or-help-desk/>
- [12] Drago Topalovic, *All About Incident Classification*, <http://www.20000academy.com/Blog/March-2013/All-About-Incident-Classification>

IMPLEMENTACIJA INTERAKTIVNE ELEKTRONSKE MALOPRODAJE U REPUBLICI SRBIJI

IMPLEMENTATION OF INTERACTIVE ELECTRONIC RETAILING IN REPUBLIC OF SERBIA

Jelena Končar, Sonja Leković, *Ekonomski fakultet Subotica, Univerzitet u Novom Sadu*

Sadržaj: *Elektronska maloprodaja predstavlja potencijalnu šansu za obezbeđivanje konkurentnosti maloprodavaca putem interaktivnog kontakta sa potrošačima, kao i mogućnost za potrošače da pronađu adekvatnu ponudu. Trendovi razvoja elektronske maloprodaje u zemljama u okruženju koje su članice Evropske unije, predstavljaju faktor vučenja daljeg razvoja elektronske maloprodaje u Srbiji. Mogućnost za dalji razvoj elektronske maloprodaje u Republici Srbiji ogleda se u dosezanju udaljenih potrošača, kao i prodaji van granica zemlje, na tržišta koja su slična po karakteristikama potrošača. Za elektronsku maloprodaju u Srbiji od ključnog značaja je implementacija infrastrukture koja će omogućiti dalji razvoj primene elektronske maloprodaje od strane potrošača. U ovoj fazi razvoja maloprodavci treba da prepoznaju prednosti elektronske maloprodaje u interaktivnom, direktnom kontaktu sa potrošačima, kao i prednosti primene inovativne strategije prodaje koja će biti način za diferenciranje od konkurencije.*

Abstract: *Electronic retailing represents a potential opportunity for securing the competitiveness of retailers through interactive contact with consumers, as well as the ability for consumers to find an adequate goods and services. Development trends of electronic retailing in the neighboring countries which are members of the European Union, are drawing factor for the further development of electronic retailing in Serbia. Opportunities for further development of e-retailing in the Republic of Serbia lies in reaching distant customers, and growing sales outside the country, in markets that has similar characteristics of consumers. For electronic retailers in Serbia the crucial is implementation of an infrastructure that will enable the further development of the useage by consumers. At this stage of development retailers should recognize the benefits of electronic retail in an interactive, direct contact with customers, as well as the advantages of innovative sales strategy that will be a way to differentiate from the competition.*

1. UVOD

Za nesmetano funkcionisanje elektronske maloprodaje neophodna je implementacija uslova za njeno odvijanje i razvoja, kao i usvajanje od strane potrošača. Elektronska maloprodaja u tom smisli postaje značajan faktor za postizanje konkurentnosti maloprodaje, što donosi pitanja kao što su:

- Kakav je potencijal elektronske maloprodaje u Evropskoj uniji?
- Šta karakteriše elektronsku maloprodaju u R. Srbiji?
- Koji su pravci daljeg razvoja elektronske maloprodaje u Republici Srbiji?
- Koji su preduslovi za dalji razvoj primene elektronske maloprodaje u Republici Srbiji?

Odgovori na ova i slična pitanja treba da postave smernice za dalji razvoj elektronske maloprodaje Srbije i zemalja u okruženju, kako bi pratile korak sa savremenim trendovima u oblasti maloprodaje.

2. KARAKTERISTIKE ELEKTRONSKE MALOPRODAJE U EVROPSKOJ UNIJI

Elektronska maloprodaja omogućuje preduzećima da prošire svoje prisustvo na domaćem tržištu, a takođe i svoju aktivnost preko nacionalnih granica, što predstavlja šansu na drugim tržištima. Na taj način, elektronska maloprodaja ima potencijal da razvije evropsko jedinstveno tržište kako za preduzeća, tako i potrošače koji imaju mogućnost da porede cene i proizvode na tržištu bez granica.

Učešće elektronske trgovine u Evropskoj uniji u prometu ukupne trgovine iznosio je 15% u 2011. godine, i 14% u 2012. godini [6]. Više od polovine preduzeća naručivalo je elektronski, u odnosu na one koji su prodavali elektronski. Elektronska maloprodaja ima brojne strategijske prednosti za preduzeća [2]:

- Troškovi poslovanja su smanjeni, inicijalna investiranja su mala, može biti održan smanjeni nivo zaliha, nisu potrebne nekretnine, bez regularnog radnog vremena i bez personala za prodaju na malo;
- Moguće je da organizacije nude niže cene (zbog nižih troškova poslovanja);
- Veliko geografsko područje može biti pokriveno efikasno i jeftino;

- Potrošaču je pružen prikladniji način kupovine: nema gužvi, problema sa parkiranjem, redova na kasi i slično;
- Specifični segmenti potrošača odabiraju se putem računara;
- Maloprodavci koji su bazirani na prodavnicama mogu koristiti elektronske medije da dopune svoje regularno poslovanje i prošire geografsko trgovinsko područje.

U Evropskoj uniji u 2011. godini 58% potrošača koristi Internet da bi se informisali o proizvodima, dok je raspoloživi podatak za Srbiju u 2009. godini iznosio 22%, za Makedoniju i Hrvatsku iz 2010. godini 30%, odnosno 43%. [1]. Podaci pokazuju da je u Srbiji u 2011. godini 58,5% potrošača tražilo informacije o proizvodima i uslugama na Internetu. Taj procenat za 2012. godinu je iznosio 72,5% što je najčešća aktivnost na Internetu u toj godini [4]. Potrošači koji su naručivali proizvode putem Interneta za ličnu upotrebu u Evropskoj uniji od 2009. do 2011. godine je iznosio oko 40% (zemlje sa najvećim učešćem od oko 70% je Ujedinjeno Kraljevstvo, kao i Švedska, Danska, Norveška, Holandija, Luksemburg), značajno manje učešće je u Španiji, Grčkoj, Italiji, a u zemljama u okruženju Hrvatskoj, Makedoniji i Srbiji manje od 5% [1]. U Evropskoj uniji jedan od ciljeva Digitalne agende je da se do 2015. godine 50% stanovništva kupuje putem Interneta, a 30% malih i srednjih preduzeća trguje na ovaj način [5].

Navedeni podaci ukazuju na značaj elektronske maloprodaje i njene trendove u Evropskoj uniji, kao i značajan zaostatak u razvoju ove oblasti u Srbiji i zemljama u okruženju. Stoga je neophodno praćenje stanja elektronske maloprodaje, kao i sprovođenje strategije njene implementacije i afirmisanje progama za usvajanje od strane potrošača i maloprodavaca.

3. STANJE I PRAVCI RAZVOJA ELEKTRONSKE MALOPRODAJE U REPUBLICI SRBIJI

Elektronski maloprodavac predstavlja on-line maloprodajni objekat koji se, po svojoj veličini, razlikuje od izuzetno velikog, kao što je Amazon.com, do sitnih malih objekata, koji imaju svoj Web sajt. Neki elektronski maloprodavci, tzv. „cigla i malter“, u svom on-line objektu imaju potpuno iste proizvode kao i u svom fizičkom objektu (u Srbiji poznati maloprodavac Univerexport i njegov sajt elakolije.rs, WinWin kompjuter šop i njegova elektronska prodavnica koji nastupa sa jedinstvenim sloganom „i na Web-u i na zemlji“, Idea, Tehnomanija i sl.). Nasuprot njima postoje i takvi maloprodavci koji posluju samo u virtualnom tržištu, virtualni ili „čisti“, bez ikakve povezanosti sa svojom fizičkom lokacijom (npr. e-Dučan.rs). U Srbiji je 98 privrednih subjekata registrovano pod šifrom trgovine na malo putem pošte i Interneta, dok ostali subjekti koji se bave elektronskom maloprodajom imaju neku drugu primarnu delatnost, a za obavljanje elektronske maloprodaje nije neophodno da se posebno registruju, tako da nema tačne evidencije koliko privrednih subjekata postoji u obavljanju elektronske maloprodaje [9].

Učešće elektronske u ukupnoj maloprodaji u Srbiji procenjuje se na manje od 10% (po nekim procenama oko 1%), dok je u Evropi i preko 15% [5]. Značaj učešća elektronske maloprodaje u rastu BDP u razvijenim zemljama je i do 20%. Prosečna potrošnja po stanovniku u Srbiji na Internetu je oko 58e, dok je taj prosek znatno veći u Evropi oko 887e. U Srbiji postoji oko 1000 elektronskih prodavnica, ali svega 10% pruža mogućnost elektronskog plaćanja [3], što ukazuje na nedostatak integracije ključnih učesnika u realizaciji elektronske transakcije.

Osim navedenih oblika, elektronski maloprodavci mogu biti [2]: kataloški trgovci, on-line trgovinski centri, i on-line direktna prodaja proizvođača. Elektronski trgovinski centri predstavljaju jedan od oblika poslovnih modela elektronske maloprodaje. Slično kao i off-line trgovinski centar oni kroz oblike on-line prisustva ostvaruju prihod po osnovu rentiranja, od naplate za korišćenje usluga od strane maloprodavaca koji plasiraju svoje proizvode putem trgovinskog centra. Prednost on-line elektronskih centara vezana je za eliminisanje velikih izdataka za izgradnju fizičkih trgovinskih centara. On-line trgovinski centri imaju prednost u odnosu na on-line katalošku prodaju jer nude širok asortiman proizvoda.

U Srbiji je od 2010. godine izuzetno razvijena oblast grupne kupovine i aukcijske kupovine. Trenutno na srpskom tržištu funkcioniše 50 elektronskih prodavnica za grupne kupovine koje nude ponudu robe i usluge po cenama sniženih od 50 do 70%. Ovakav način predstavljanja ponude omogućuje potrošačima povoljniju ponudu, a maloprodavci dobru promociju ponude i unapređenje dalje prodaje.

Za elektronsku maloprodaju je od značaja dolazak potrošača do traženih informacija o ponudi, pri čemu potrošači uz pomoć različitih pretraživača pronalaze informacije o proizvodima. Kada se odlučuju o kupovini potrošači u Srbiji sve češće koriste informacije koje im se pružaju na Internetu na elektronskim prodavnicama. Maloprodaja sve više pronalazi mesto na Internetu, što pruža brojne prednosti kako za maloprodavce tako i za potrošače. Potrošači u Srbiji najčešće pretražuju informacije na Internetu pre putovanja i kupovine tehničkih uređaja i to nedelju dana pre donošenja odluke o kupovini, dok kupovinu radije obavljaju u prodajnom objektu. U Republici Srbiji 60,8% korisnika Interneta primarno koriste Internet za pretraživanje informacija o proizvodima i uslugama [4]. Čak 69% potrošača u Srbiji koji su kupovali u maloprodaji prethodno su pretraživali na Internetu, a samo 18% je kupilo u elektronskoj prodavnici, dok je taj procenat u Sloveniji veći za 5% [8].

Ako se posmatra vrednost potrošačke korpe kod pojedinih maloprodavaca koji imaju tradicionalnu i elektronsku maloprodaju u Srbiji (kao što su Univerexport, Idea, WinWin, Tehnomanija), ona je prema procenama veća u elektronskoj maloprodaji nego u tradicionalnoj maloprodaji u Srbiji. Učešće elektronske maloprodaje u rastu BDP u razvijenim zemljama je 20%, a za svako izglubljeno radno mesto u tradicionalnoj maloprodaji, ona kreira 2,6 novih radnih mesta u elektronskoj maloprodaji, i to u poslovima

obrade narudžbenica, slanja pošiljki, dostavljanja i sl. Elektronske prodavnice se u Evropi razvijaju po stopi dva puta većoj nego kompanije u drugim oblastima [5].

Da će razvoj elektronske maloprodaje imati rast i u narednom periodu ukazuje 2.400.000 stanovnika koji koristi Internet svakodnevno (povećanje od 300.000 u odnosu na 2012. godinu), i 900.000 onih koji su kupovali na Internetu (povećanje od 300.000 u odnosu na 2012. godinu) [4]. Međutim, ne treba zanemariti činjenicu da 64,5% nikada nije kupovalo na Internetu, ali taj procenat iz godine u godinu opada (2009. godine 87,4%, 2010. godine 87%, 2011. godine 81,9%, 2012. godine 73,3%). Ukoliko se nastavi ovaj trend u 2014. godini se može očekivati da će biti oko 50% onih koji kupuju na Internetu u Srbiji.

4. IMPLEMENTACIJA INFRASTRUKTURE ELEKTRONSKE MALOPRODAJE U REPUBLICI SRBIJI

Maloprodaja putem Interneta sve je češća pojava u Srbiji. Tržište elektronske maloprodaje ima visok raste svake godine, ali doprinos daljeg razvoja primene elektronske maloprodaje zahteva implementaciju infrastrukture i širokopojasnog Interneta, kao i regulisanje oblasti elektronskog novca, elektronskih ugovora, zaštite potrošača.

U Srbiji Internet priključak ima 55,8% u 2013. godini, što je rast u odnosu na prethodnu godinu [4]. Ovaj procenat u 2012. godini bio je 48% domaćinstava., dok je prosek u EU 76% (najveći je u Švedskoj 92%) [1]. Širokopojasni Internet raspoloživo 43,4% domaćinstava u Srbiji [4]. Procenat lica koje je koristilo Internet u poslednjih godinu dana u Srbiji je 52%, dok je prosek EU 75%, najveći u Danskoj 93%, a najniži u Rumuniji 50% [1].

Prema istraživanjima MasterCard-a u Srbiji je broj korisnika kartica koji kupuju na Internetu povećan za 10% tokom poslednjih tri godine, sa 14% u 2010. godini, na 24% u 2012. godini. Kada je reč o vrsti plaćanja: 32% plaća svoje računa online, koliko i onih koji kupuju odeću i obuću, dok 21% kupuje poklon, knjige 16,9%, karte za sportske i kulturne događaje 16,4%, a kućne aparate 7,2%, 24% kupuje avionske karte, 19% hotelske troškove i troškove smeštaja, a 7,7% plaća putne aranžmane [7]. Broj oni koji veruju u bezbednost online plaćanja je poraslo u odnosu na 2010. godinu, što ukazuje na potencijal daljeg razvoja elektronske maloprodaje u Srbiji.

Prema podacima Narodne banke Srbije u prvih šest meseci u 2013. godini, na domaćim sajtovima je plaćeno elektronskim putem, karticama, roba u vrednosti od 499,5 miliona dinara (što je 89.700 transakcija), a na stranim 109 miliona evra (707.000 transakcija), što iznosi u proseku 154e po transakciji [10]. Ovo ukazuje da potrošači u Srbiji manje troše na domaćim elektronskim prodavnicama i to u proseku 5566 dinara po transakciji. U poređenju sa podacima iz prethodne godine pokazalo se da su potrošači u Srbiji manje trošili na domaćim elektronskim prodavnicama, a više na stranim. Prosečna transakcija u prethodnoj godini je bila

5.298 dinara, dok je prosek preko stranih sajtova 169e po transakciji. Za prvih šest meseci u 2013. godini, premašena je polovina prometa ostvarena prilikom plaćanja karticama na Internetu tokom cele 2012. godine. Najveća potrošnja je zabeležena prvenstveno na stranim sajtovima, koja je za prvih šest meseci u 2013. godini iznosila 64% od prometa u 2012. godini.

Od 2013. godine su otklonjene sve prepreke za funkcionisanje PayPal-a u Srbiji, što će olakšati elektronsku maloprodaju prvenstveno robe iz inostranstva. Plaćanjem i naplatom preko PayPal sistema moguće je obavljati platni promet sa inostranstvom i preko strane institucije elektronskog novca za plaćanje i naplatu po osnovu elektronske kupoprodaje robe i usluga. Ovo omogućuje da se vrši naplata iz inostranstva i plaćanje prema inostranstvu u stranoj valuti za robu kupljenu putem Interneta preko PayPal sistema. S druge strane, plaćanje i naplata robe u Srbiji između domaćih privrednih subjekata se izvršava u domaćoj valuti.

Za funkcionisanje elektronske maloprodaje od značaja je nesmetano dostavljanje naručene robe do potrošača. U tu svrhu se razvija usluga Postekspres, koja je u 2011. godini u Srbiji zabeležila više od milion evra izvoza robe manje vrednosti, dok DHL u Srbiji godišnje ima 620.000 pošiljki [5]. Najznačajnija pitanja zaštite potrošača regulisana su Zakonom o zaštiti potrošača [11] čiji je strateški cilj unapređenje prava i zaštite interesa potrošača u kontekstu integracije Srbije u Evropsku uniju.

5. ZAKLJUČAK

Elektronska maloprodaja ima uticaj na dalji razvoj evropskog jedinstvenog tržišta, i pruža prednosti kako za preduzeća, tako i potrošače koji imaju mogućnost da porede cene i proizvode na tržištu bez granica. Ono što je neophodno je praćenje trendova elektronske maloprodaje u Evropskoj uniji. U Srbiji i zemljama u okruženju primetan je zaostatak u razvoju ove oblasti. Značajan pokazatelj elektronske maloprodaje u Srbiji je vrednost potrošačke korpe, koja je prema procenama veća u elektronskoj maloprodaji nego u tradicionalnoj maloprodaji kod maloprodavaca koji imaju i tradicionalnu i elektronsku maloprodaju (kao što su Univerexport, Idea, WinWin, Tehnomanija i drugi). Tržište elektronske maloprodaje u Republici Srbiji ima tendenciju rasta, a doprinos daljem razvoju elektronske maloprodaje zahteva implementaciju adekvatne infrastrukture koja podrazumeva širenje širokopojasnog Interneta, regulisanje oblasti elektronskog novca, zaključivanje elektronskih ugovora, kao i zaštite potrošače u elektronskoj maloprodaji.

LITERATURA

- [1] EuroStat, Business and trade statistics, European Commission, Luxemburg, december 2012.

- [2] J. Končar, *Elektronska trgovina*, Ekonomski fakultet Subotica, Univerzitet u Novom Sadu, 2008.
- [3] J. Končar, Z. Petrović-Katai, S. Leković, "Strategy and implementation of B2C electronic commerce as function of Serbian competitiveness", 20th Annual Conference on Marketing and Business Strategies for Central & Eastern Europe, Vienna, Austria, December 2012., pp. 177-190.
- [4] Republički statistički zavod, *Upotreba informaciono-komunikacione tehnologije*, Beograd, 2013.
- [5] www.economy.rs, dostupno 26.12.2011.
- [6] www.epp.eurostat.ec.europa.eu, dostupno 17.01.2014.
- [7] www.masterintelligence.com, MasterCardIndex, dostupno 2012.
- [8] www.nops.org.rs, dostupno 11.10.2013.
- [9] www.trzistesrbije.com, dostupno 05.01.2014.
- [10] www.nbs.rs, dostupno 05.01.2014.
- [11] www.mtt.gov.rs/download/1/zakon-o-zastiti-potrosaca.pdf, dostupno 05.01.2014.s

EVOLUCIJA SISTEMA ZA PODRŠKU ODLUČIVANJU I NJIHOVE PRIMENE U ZDRAVSTVU

THE EVOLUTION OF DECISION SUPPORT SYSTEMS AND THEIR APPLICATION IN HEALTHCARE SYSTEM

Obradović Milovan, Vojnomedicinska akademija, Beograd

Sadržaj: Tokom proteklih decenija savremeni poslovni sistemi su se suočavali sa velikim problemom nedovoljne iskorišćenosti prikupljenih podataka. Od početka primene sistema za podršku odlučivanju proteklo je oko pola veka. U poslednjoj deceniji 20. veka učinjen je pomak na planu stvaranja softverskih rešenja orijentisanih na velike skupove podataka prikupljenih u on line informatičkoj podršci procesima. U ovom radu data je evolucija sistema za podršku odlučivanju i neke njihove primene u zdravstvu.

Abstract: During the last few decades modern business systems faced significant problem of not being able to use the gathered data sufficiently enough. It has been about half a century since the beginning of the decision support systems. In the last decade of 20th century, great progress was made in that field by creating software solutions oriented to large data records, collected in an online IT support to processes. This paper describes the evolution of decision support systems and some of their applications in healthcare system.

1. UVOD

Odavno je prošlo vreme donošenja odluka na bazi iskustva, intuicije ili osećanja, pa se danas odluke donose na osnovu kompleksnih analiza gotovo svih faktora poslovanja. Oštra konkurentnost na globalnom tržištu sa ciljem sticanja i očuvanja konkurentne prednosti uslovljava analiziranje velikog broja neprecizno izraženih faktora. Borba sa konkurencijom i trka za pridobijanje i očuvanje klijenata danas nameće potrebu za primenjivanjem inteligentnih metoda i tehnika kojima bi se analizirao što veći broj podataka koji opisuju faktore poslovanja na osnovu kojih bi se proizvele informacije iz kojih bi se sintetizovalo znanje koje bi omogućilo strategijsko usmeravanje preduzeća.

Razmatranju odlučivanja kao problema može se pristupiti na različite načine. Situacija odlučivanja se najčešće zasniva na logičkoj konstrukciji koja nastaje kao rezultat odgovora na niz pitanja. Cilj svih tih odgovora doprinosi opštem situacije odlučivanja. Moramo se zapitati, kome su odluke potrebne, šta je odlučivanje, šta predstavlja i šta ono sadrži? Odgovori na ova pitanja su gotovo očigledni. Pod odlučivanjem se podrazumeva izbor jedne od dve ili više akcija koje se mogu preduzeti radi postizanja cilja. Izbor jedne od mogućih akcija radi postizanja cilja predstavlja problem odlučivanja. Ono što svakom odlučivanju prethodi je neizvesnost u pogledu izbora najboljeg načina za postizanje željenog cilja. Odabrani tok akcija ili plan akcija čini strategiju a njena primena, u konkretnim uslovima, sa izborom sredstava za realizaciju čini taktiku sprovođenja u delo.

Postoje dva nesaglasna pristupa procesu donošenja odluka:

- normativni, koji precizira kako i zašto donosilac odluka treba da se ponaša,
- deskriptivni, koji polazeći od cilja definiše alternative i vrši izbor radi postizanja "zadovoljavajućeg rešenja".

Uočava se da postoji bitna razlika kod ova dva pristupa. Dok se kod normativnog primenom kvantitativnih metoda uz optimizaciju činilaca odlučivanja na racionalan način teži cilju, dotle se deskriptivni pristup bazira na znanjima, iskustvu, intuiciji i heuristici donosioca odluke. U praksi je najčešće prisutna kombinacija ova dva pristupa gde se pri odlučivanju uzimaju dobre osobine svakog od njih.

Razvrstavanje odlučivanja vrši se po različitim kriterijumima:

- u odnosu na uslove: u uslovima izvesnosti, rizika, neizvesnosti i kombinovano
prema ponovljivosti: ponovljive - višekratne odluke, neponovljive - jednokratne odluke
- prema struktuiranosti: struktuirano - po pravilima i algoritmima, nestruktuirano - bez pravila i polustruktuirano - sa i bez pravila.

Uzimajući u obzir osobine normativnog i deskriptivnog pristupa kao i razvrstavanje odlučivanja prema struktuiranosti jasno je da su struktuirani problemi pogodni za normativni a nestruktuirani za deskriptivni pristup. Polustruktuirani problemi se mogu rešavati primenom jednog ili (i) drugog pristupa.

Fenomeni odlučivanja su oduvek zaokupljali pažnju čoveka. Činilo se to na različite načine i u različitim stepenima razvoja nauke i tehnologije. Od Aristotela, koji je tvorac formalne logike, preko niza logičara, filozofa, matematičara, do današnjih dana kada se može reći da postoji dominacija tehnologije i informatike, razvila se kao posebna naučna disciplina teorija odlučivanja. Značajan doprinos tom razvoju dale su informaciono komunikacione tehnologije (na engl. Information and Communication Technology – ICT). U teoriji odlučivanja se nalazi osnova za izgradnju sistema za podršku odlučivanju.

U literaturi iz menadžmenta se navodi da tačnost predviđanja prvenstveno zavisi od nivoa razvijenosti

informacionog sistema, dužine vremenskog perioda predviđanja, stabilnosti uslova poslovanja, uticaja na okolinu i metoda i tehnika koje se koriste za predviđanje [1,2].

2. SISTEMI ZA PODRŠKU ODLUČIVANJU

Početna faza u korišćenju računara bila je automatska obrada podataka (na engl. Electronic Data Processing - EDP) i ona datira od pedesetih godina prošlog veka. Karakterišu je procesi prikupljanja i unosa – pripreme podataka koji obuhvataju prikupljanje, dostavu, kontrolu i prenos (memorisanje) podataka sa dokumentacije na medijume koji čine ulaze u procese EDP. Negativne strane ovakvog pristupa ogledale su se u ograničenosti memorijskih resursa, sporim obradama velikih količina podataka, ograničenim mogućnostima kontrole podataka, velikim vremenskim i prostornim razmacima u odnosu na mesta nastanka dokumentacije, potrebi prilagođenja dokumentacije EDP, relativno slaboj komunikaciji službi za informatiku i ostalih, velikim količinama "izlaznih" rezultata stvorenih za rad brojnih službi ali i kao obavezna (propisana) dokumentacija.

EDP je pogodna za obrade velikih količina podataka, pri čemu obezbeđuje sumarne i pojedinačne pokazatelje za sve nivoe upravljanja: strateški, operativni i taktički. Rezultati koji se stavljaju na raspolaganje korisnicima najčešće su unapred tipizirani, obrade dugo traju a fleksibilnost u pogledu realizacije novih zahteva je direktno zavisna od raspoloživosti i sposobnosti informatičara da brzo odreaguju na njih. Načini iskazivanja kvantitativnih pokazatelja su najčešće tabelarnog ili izveštajnog tipa i zahtevaju dodatne obrade kojima se uobličavaju u forme prikladne za menadžere svih nivoa.

Pristup upravljačkih informacionih sistema (na engl. Management Information Systems – MIS) baziran je na integrišućoj i planskoj funkciji informacionog sistema [1]. Javili su se sedamdesetih i ranih osamdesetih godina prošlog veka sa naglim razvojem hardvera, uvećanjem obradnih resursa i brzine rada, višekorisničkim radom i novim organizacijama podataka. Sredinom 70-tih godina prošlog veka sa razvojem baza podataka i sistema (softvera) za njihovo rukovanje (na engl. Data Base Management Software - DBMS) pojavili su se komercijalni modeli baza podataka (hijerarhijski, mrežni, relacioni) kao osnova za transakcione obrade podataka u realnom vremenu (engl. Online Transaction Processing – OLTP).

MIS predstavlja viši nivo u odnosu na EDP, ali nivo koji je orijentisan na informacione tokove i podatke. Dajući podršku radnim procesima na mestu i u vremenu njihovog odvijanja MIS obezbeđuje informacije i podatke za sve druge nivoe rada i upravljanja. On postaje integrator procesa, eliminiše višestruke unose istih podataka, obezbeđuje pravovremenu – neposrednu kontrolu i njihovo jedinstvo u sistemu i okruženju.

Šta su ograničenja za strateški menadžment kada je mogućnost upotrebe OLTP kao njegove podrške? Odgovor na to pitanje sastoji se u realnom postojanju: velikog broja entiteta, kvalitetnoj organizaciji podataka sa minimalnom ili bez ikakve redundancije, potrebe za kompleksnim pretraživanjima velikog broja entiteta od kojih neki mogu biti

u okruženju. Sve je to uslovljeno: dužim čekanjem na rezultate pretraživanja – veće je vreme odziva, konkurentnim radom sa korisnicima poslovnih transakcija, potrebom postojanja moćnije tehnološke osnove, zavisnošću korisnika strateškog nivoa od profesionalnih informatičara i dr.

Sistemi za podršku odlučivanju (na engl. Decision Support Systems - DSS) pojavili su se kasnih šezdesetih i početkom sedamdesetih godina prošlog veka. Javili su se kao prirodna posledica: naglog razvoja hardvera i softvera, istraživanja i učenja na univerzitetima, potrebi podrške odlučivanju, potrebe za kvalitetnijim informacijama, za konkurentnošću i dr. Ispravna je tvrdnja da su sistemi za podršku odlučivanju integrator informatičkih i upravljačkih procesa. Svi navedeni razlozi inicirali su stvaranje snažne analitičke pozadine koja ima izvorišta u mnogim OLTP sistemima i dostupnost brojnim korisničkim aplikacijama strateškog nivoa [2].

Koncept DSS definisan je sedamdesetih godina od M. S. Morton-a pod nazivom "Upravljački sistemi za odlučivanje" [2]. Nekoliko ustanova i škola je otpočelo razvoj i istraživanje DSS. Kao polazna osnova za razvoj uzeti su sledeći zahtevi: interaktivni rad korisnika, pomoć pri donošenju odluka korišćenjem podataka i modela i rešavanje nestruktuiranih problema.

Dajući karakteristike DSS autor Alter Keen [3] između ostalog, navodi da oni:

- služe za podršku manjim strukturama, rešavanju nestruktuiranih problema i za više nivoe upravljanja
- dozvoljavaju kombinovanje u korišćenju modela i metoda sa tradicionalnim pristupom i funkcijama pretraživanja
- specificiraju moguće poglede za neinformatičke kadrove pri interaktivnom radu
- poseduju fleksibilnost i prilagodljivost na izmene okruženja uz istovremeno prilagođenje korisniku.

DSS kao viši stepen razvoja informacionih sistema od MIS i EDP donosi niz prednosti i koristi od kojih sve ne mogu biti izražene kvantitativno. Računar je u procesima odlučivanja dobio novu i reklo bi se nezamenljivu ulogu. Rane faze razvoja DSS imale su arhitekturu koja se sastojala od komponenti za upravljanje podacima, upravljanje modelima i komponente za korisnički interfejs.

Komponenta za upravljanje podacima sadržala je bazu podataka za odlučivanje i softver za njeno upravljanje i održavanje. Baza podataka za podršku odlučivanju posedovala je sledeće mogućnosti:

- kombinovanje različitih izvora podataka
- brzo i lako dodavanje i brisanje izvora podataka
- prikaz logičke strukture podataka korisniku radi razumevanja mogućnosti i specifikacije izmena
- samostalan rad i biranje podataka, eksperimentisanje sa alternativama zasnovanim na ličnom zaključivanju
- upotrebu velike raznolikosti podataka.

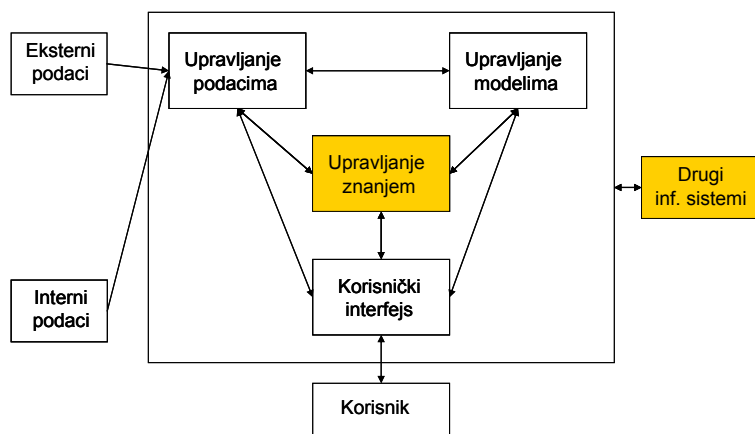
Veoma važna osobina svih baza podataka za podršku odlučivanju sastoji se u većem bogatstvu setova podataka u odnosu na obične baze podataka. DSS baze podataka zahtevaju i posebne pristupe i DBMS koji će ih podržati uz brze odgovore na korisničke zahteve.

Komponenta za upravljanje modelima sadržala je bazu modela i softver za njeno održavanje. U terminologiji DSS ovaj softver je poznat pod imenom sistem (softver) za rukovanje bazom modela (na engl. MBMS - Model Base Management Software). Ovi modeli mogu biti: stratezijski, taktički, operacioni i modeli za izradu blokova i rutina. Podsystem modela ima ulogu da u okviru DSS omogućiti: brzo i lako kreiranje novih modela, smeštanje i održavanje velikog broja modela za sve nivoe upravljanja i korišćenja, povezivanje modela sa bazom podataka, i korišćenje funkcije održavanja baze modela.

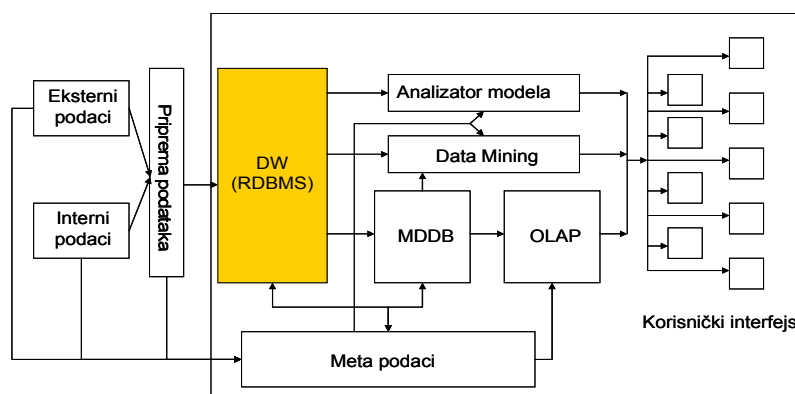
Softver za povezivanje korisnika sa svim segmentima sistema bio je treća komponenta u arhitekturi DSS. Njegova uloga se svodila na korisnički interfejs sa mašinom (računarom). Zahtevi definišu šta i kako korisnik može da uradi uz pomoć sistema. Dobijeni odgovori odnose se na ono što sistem nudi korisniku da vidi, koristeći za to monitor, printere, plotere, grafički monitor, audio izlaze i dr. Odgovori sistema se dobijaju iz podsistema podataka i podsistema modela interaktivnim radom, sa raznovrsnim i efektivnim oblicima prezentacije uz lako i jednostavno korišćenje.

Najveći broj prednosti primene DSS se ne može izmeriti ali se mora istaći da su takve prednosti sadržane u: porastu broja alternativa, boljem razumevanju posla, učenju, kontroli, boljim i kvalitetnijim odlukama, poboljšanju timskog rada, većim potencijalima za donošenje odluka, efikasnijem korišćenju radnog vremena menadžera, unapređenju planiranja, unapređenju poslovanja i dr. Manji deo prednosti primene DSS može se meriti kroz uštede vremena i sredstava kroz brze odgovore na nepredviđene situacije, interaktivan rad, sprovođenje ad hoc analiza, pomoć donosiocima odluka strateškog i taktičkog nivoa, boljem i efikasnijem korišćenju podataka kao resursa, ukrštanju podataka iz drugih OLTP sistema i okruženja i dr. i sve to uz samostalnost korisnika u odnosu na informatičare, nezavisno od OLTP softvera i baza podataka.

Globalna arhitektura savremenih sistema za podršku odlučivanju (Slika 2.) je kompleksnija i sadrži komponentu koja se odnosi na upravljanje znanjem i niz savremenih tehnika i tehnika i alata koje su date na Slici 3. Osnovne karakteristike DDS se sastoje u tome:



Slika 2. Globalna arhitektura DSS (prema [4]).



Slika 3. Detaljna arhitektura savremenog DSS

- da su nezavisni od OLTP softvera i baza podataka
- da imaju oslonac na velika skladišta podataka (Tb)
- da sadrže višedimenzionalne baze (na engl. Multi Dimensional Data Base - MDDB) agregiranih podataka
- da sadrže nove metode i alate (OLAP, Data Mining, ...)
- da imaju veći stepen ukrštanja podataka
- da otkrivaju znanja i informacije iz istorijskih podataka
- da su upiti usmereni na odlučivanje i predviđanje budućnosti
- da su namenjeni za konkurentan rad korisnika
- da obezbeđuju podršku rešavanju ad hoc zahteva
- da zadovoljavaju potrebe strateškog, taktičkog i operativnog nivoa.

Komponenta za upravljanje podacima sadrži nekoliko oblika baza podataka i softvera za njihovo rukovanje. Baza tipa velikog skladišta podataka (na engl. Data Warehouse - DW) se definiše kao "subjektu orijentisan, integrisan, nenarušiv, vremenski promenljiv skup podataka organizovan da zadovolji potrebe menadžmenta" [5]. U poređenju sa transakcionim bazama podataka DW koncepti filtriraju, selektuju i agregiraju tako prikupljene podatke i posle toga ih prevode i smeštaju u tzv. dimenzione baze podataka, modelirane tako da zadovolje širok spektar potreba (ocena stanja, trendovi, projekcije, alternative).

Za analitičku obradu podataka u realnom vremenu (na engl. On Line Analytical Processing - OLAP) bazu podataka čine izdvojeni podaci iz različitih relacionih i drugih transakcionih baza podataka. Postoje različiti koncepti OLAP baza podataka kao što su: relacioni (na engl. Relational OLAP), višedimenzionalni (na engl. Multi Dimensional OLAP), kombinacija relacionog i višedimenzionalnog (na engl. Hybrid OLAP) itd. Ove baze podataka su po svojim veličinama u rasponu od nekoliko desetina Gb do nekoliko desetina Tb.

Razvoj alata i tehnika za dubinsku analizu podataka (na engl. Data Mining - DM) kao nove tehnologije ima prevashodno potporu u naglom razvoju i mogućnostima ICT. DM kao posebnost se vezuje za sredinu 90-tih godina prošlog veka. Velike količine podataka sačuvane u elektronskom obliku predstavljaju polaznu osnovu za primenu sofisticiranih alata za analizu, statističke modele, matematičke algoritme, baze podataka i sl. kao osnove za dobijanje podataka za odlučivanje. DM u podršci odlučivanju razlikuje se od klasičnih statističkih metoda po tome što se ne odvija po unapred utvrđenim pravilima, već kroz kreativnost u analizi podataka može da otkrije nova, nepoznata pravila ponašanja koja obezbeđuju otkrivanje potencijalnih opasnosti ili prednosti. Upotrebom DM vrši se analiza podataka bez startnih pretpostavki koje postavljamo za analizu neke pojave ali sa pitanjem na koje tražimo odgovor koji treba da da DM [4, 6]. Razlog za to se sastoji u razlici u pristupu jer se sa DM utvrđuju veze i međuzavisnosti velikog broja pojava, obezbeđuje znanje za rešavanje problema, unapređivanje poslovanja i predviđanje sa osloncem na softversko modeliranje.

3. PRIMENE DSS ZDRAVSTVU

Vođenje zdravstvene politike predstavlja ključni zadatak, u zdravstvenom sistemu (ZS) države, koji karakteriše: donošenje odluka za sprovođenje aktivnosti radi ostvarivanja postavljenih ciljeva, određivanje prioriteta između ciljeva, sagledavanje uticaja okruženja i politike društveno ekonomskog razvoja, sagledavanje materijalnih mogućnosti društva i građana, određivanje obima zdravstvene zaštite, obezbeđenje dostupnosti zdravstvene zaštite celokupnom stanovništvu, promocija zdravlja i informisanje građana

Da bi se ovako složeni zadaci mogli ostvariti neophodno je da se raspolaze kvalitetnim, selektivnim, agregiranim i blagovremenim informacijama, jer se jedino tako može postaviti dobra organizacija ZS i planiranje usklađeno sa materijalnim mogućnostima [7].

Informacije potrebne za strateški menadžment odnose se na: zdravstveno stanje stanovništva, pojave i širenja zaraznih i drugih bolesti, obezbeđenje i trošenje lekova i sanitetskih materijala, obim izvršenih usluga u zdravstvenoj zaštiti, korišćenje resursa, obezbeđenje i trošenje sredstava zdravstvenog osiguranja, održavanje objekata i opreme i dr. Izvori podataka za izradu svih analiza su podaci za bilo koju teritoriju i zemlju u celini koji se nalaze se u operativnim OLTP bazama podataka o: stanovništvu, prostoru, zdravstvenim ustanovama, zdravstvenoj zaštiti, lekovima i dr. Značajna prednost ovakvih izvora podataka sadrži se u tome da postoje u elektronskom obliku i da su u standardizovani i kontrolisani tokom odvijanja transakcija [8]. Za podršku specifičnim potrebama lekara u dijagnostici, lečenju i medikamentoznoj terapiji kvalitetan doprinos daju DM alati koji se mogu koristiti za predviđanje: medicinskih dijagnoza na osnovu simptoma i znakova zdravstvenog stanja, ranog otkrivanja i (ili) prevencije bolesti, verovatnoće da pacijent dobije srčano oboljenje, ishoda kod obolelih sa izlivom krvi u mozak, ishoda kod infarkta srca, oboljevanja od karcinoma dojke, potrebe bolničkog lečenja, prevremenog rađanja, prevencije rizika od samoubistava, proteinske reakcije kod obolelih od dijabetesa i dr. [9, 10].

4. ZAKLJUČAK

Uzimajući u obzir sve navedene prednosti jasno je da uloga DSS raste svakim danom a sve jeftiniji hardver i softver (prilagođen lakom i jednostavnom korišćenju) utiču da ta uloga brže i efikasnije dolazi do izražaja. Realno je očekivati da će primena savremenih DSS u zdravstvu rasti, jer se u svakodnevnoj podršci procesima lečenja i upravljanja resursima stvaraju ogromne količine podataka koji trebaju biti iskorišćeni radi uspešnijeg vođenja zdravstvene politike, racionalnog korišćenja raspoloživih resursa i pomoć u dijagnostici i terapiji.

LITERATURA

- [1] G.B. Davis, M.H. Olson, *Management Information Systems*, New York: McGraw-Hill Book Company, 1985.
- [2] M.S. Morton, *Management Decision Systems*, Boston: Harvard University press, 1971.
- [3] R.H. Sprague, Watson H.J., *Decision Support Systems*, New York: Prentice Hall International, 1989.
- [4] E. Turban, E. J. Aronson, *Decision Support Systems and Intelligent Systems, 5th Edition*, New York: Prentice Hall International, 1998.
- [5] W. H. Inmon, *Building Data Warehouse, 3th Edition*, New York: John Wiley & Sons, 2002.
- [6] J. Han, M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2006.
- [7] P. Dovijanić, *Organizacija zdravstvene delatnosti i ustanova*, Beograd: Agora, 1995.
- [8] J.H. Bommel, M.A.Musen, *Handbook of Medical Informatics*, Houten/Diegem: Springer, 1997.
- [9] J. Tan., F.C. Payton, *Adaptive Health Management Information Systems*, Boston Toronto London Singapore: Jones and Bartlett Publishers, 2010.
- [10] J. M. Hardin, D. C. Chhieng, "Data Mining and Clinical Decision Support Systems", Available at: www.researchgate.net

IZBOR OPTIMALNE TEHNIKE ZA ENKRIPCiju I DEKRIPCiju PODATAKA SELECTION OF OPTIMAL TECHNIQUES FOR ENCRYPTION AND DECRYPTION OF DATA

Ilija Apostolov, Risto Hristov, *Evropski Univerzitet Republika Makedonija –Skopje*
Sašo Gelev, *Elektrotehnički fakultet - Radoviš*

Sadržaj: *Enkripcija predstavlja proces konvertiranja jednostavne tekst poruke u šifrirani tekst koji može biti dekodiran u originalnom tekstu. Enkripcija i dekrepcija se sprovodi putem algoritma kojeg možemo implementovati u programskom jeziku. Razgledaćemo potrebe za zaštitu podataka, kao i nacine enkripcije i njihove algoritme. Daćemo predlog za enkripciju e-mail poruke. Zatim ćemo pričati o potrebama za enkripciju i predstavimo način podsticaja za korišćenje enkripcijskih tehnika. Veliki broj softverskih rešenja uspešno se spravljaju sa zaštitom prilikom prenosa podataka, e-mail enkripcije i enkripcije celih diskova.*

Abstract: *- Encryption is the process of converting simple text message in the encoded text that can be decoded back to the original message. The method of encryption and decryption of data is implemented by an algorithm that can be implemented in a programming language. We will review the needs for data protection and encryption types and their algorithms. We will also make suggestions for encrypting e- mail messages. Then we will talk about the need for encryption and we will demonstrate the way to encourage the use of encryption techniques. Many software solutions are successfully deal with the protection of data transmission, e- mail encryption and decryption of entire hard disks.*

1. UVOD

Pre pedeset godina algoritme za enkripciju su koristili samo najbolji računari u svetu. Danas imamo potrebu da obezbedimo podatke u našim računarima. Podaci se moraju obezbediti od krađe ili gubljenje. Troškovi povezani sa gubitkom podataka su nepredvidivi, a osim vremena i novca, može da uključi i gubitak pouzdanosti kompanije i iznošenje tajne u javnost. Potreba za sigurnijom zaštitom je sve veća i zato IT lica troše sve više vremena i resurse na ovoj problematici. Programeri za enkripciju obezbeđuju rešenja koja nude sveobuhvatnu skalabilnu zaštitu koja ne dozvoljava otvoreni pristup do sve tačke kompanije, organizacije ili do podataka koji su spodeljeni na mreži [1]. Kriptografija je nauka koja proučava bezbedno zapisivanje. U svim novim tehnologijama i tehnikama stari način kriptiranja i dekriptiranja podataka zamenjen je novim sofisticiranim načinom koji velikom lakoćom bezbedno čuva naše podatke[2]. U suštini, kriptografski algoritam predstavlja sklop matematičkih funkcija koje koristimo za enkripciju i dekripciju podataka [3]

A. Kodiranje naspram dekodiranja



Slika 1. Proces enkripcije i dekripcije

Kodiranje predstavlja interpretacija reči ili fraza u druge reči i fraza. Ova dva procesa omogućavaju šifrovano prevođenje pisama ili simbola pojedinačno. U suštini, enkripcija kao termin obuhvata enkripciju i šifrovanje. Naspram toga, dekripcija obuhvata dekodiranje i dešifrovanje [3].

B. Principi enkripcije

Kriptografija može biti i jaka i slaba. Rezultat jake kriptografije je šifrovan tekst kojeg je teško dešifrovati ako nemaš specijalnu alatku za dekodiranje ili ako ne poznaješ algoritam koji je korišten za šifrovanje poruke [4]. Na internetu možemo naći veliki broj besplatnih softvera za enkripciju, ali za bolja softverska rešenja kompanije naplaćuju za uloženi rad u izradi ovih moćnih programa koji se oslanjaju na matematičke i ostale korisne algoritme koji su pričina za njihovo odlično funkcionisanje. Danas je većina ovih algoritama implementirana u većinu popularnih programskih jezika.

Prve primene kriptografije nalazimo još u vreme Julije Cezara. On je svoj algoritam definisao na taj način što pomera alfabetu za tri pozicije u desno i odbacuje zadnja tri slova, u ovom slučaju X,Y, i Z. Danas se ovaj način ne računa bezbednim, ali u to vreme je dobro funkcionisao. Onaj koji prima poruku da bi je pročitao mora pomeriti slova za tri mesta unazad da bi je pročitao [5].

Primer : A B C D E...
 ↓ ↓ ↓ ↓ ↓
 D E F G H...

Radi obezbeđivanje poverljivosti kriptografija koristi sledeće principe:

Autentifikacija: U ovom procesue primač enkriptirane poruke mora utvrditi poteklo poruke.

Integritet: Primaoc poruke mora biti sposoban da vidi dali je poruka promenjena za vreme njenog putovanja. Napadač ne može zamenite original lažnom porukom.

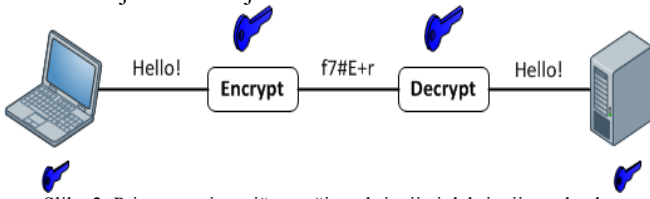
Nepoštovanje: Ni u jednom slučaju pošiljalac ne sme da negira da je poruka od njega poslata.

Poverljivost: Poruka mora biti tako enkriptovana da je nitko drugi sem nas ne može pročitati [5].

2. VIDOVI ENKRIPTIRANJA PODATAKA

Simetrična enkripcija

Simetrična enkripcija ili poznata i kao spodeli ključ. U simetričnoj enkripciji za kriptiranje i dekriptiranje podataka se koristi jedan isti ključ.



Slika 2. Primer za simetričan način enkripcije i dekripcije podataka

Simetrična enkripcija sadrži brye algoritme jer imaju relativno malu kompleksnost koja dozvoljava lako prilagođavanje hardveru [6]. Ipak, ovi algoritmi traže da svi glavni računari koji se koriste u ovom procesu, prethodno budu konfigurisani tajnim ključem. Tajni ključevi se koriste u procesu enkripcije da bi izbegli neovlašćeni pristup napadača, koji kasnije može sebe da predstavi kao pošaljioa poruke. Zatim se dekripcija vrši kod primaoca poruke, koji dekriptira poruku kopijom ključa pošaljioa [7].

1.1 DES (Data Encryption Standard)

Krajem šesdesetih godina dvadesetog veka razvojem finansijskih transakcija, kriptografija je bila sve više interesantna korisnicima. Javila se potreba šifrovanja kojeg bi koristili ljudi širom sveta, tj javila se potreba poverljivosti koja bi bila uspostavljena uvođenjem standarda u kriptografiji [8].

1973 godine Američki institut za standardizaciju i tehnologiju (NIST) imao je za cilj da kreira program za zaštitu računarskih i komunikacijskih podataka. Raspisali su tender za razvoj standardnog sistema kriptografije. Kompanija IBM je ponudila algoritam. Ovaj se algoritam zasnovao na šifrovanje Horsta Feistela. Predloženi algoritam je nakon nekoliko modifikacija bio prihvaćen kao standar 1976 godine pod imenom Data Encryption Standard (DES). Ovaj standar enkriptira i dekriptira blokove od 64 bita i koristi ključ veličine 56 bita [9].

Kasnije kao zamena za DES standar ponuđen je standard 3DES (Triple Data Encryption Standard) koji je bio odobren od FIPS (Federal Information Processing Standards Publications) koji obuhvata kriptografske zahteve za kriptografse module. Sve više vladinih organizacija u Americi i širom sveta prihvataju 3DES standar kao osnovu razumne strategije koja preuzima zaštitne mere povezane sa rizikom od napadača i neželjenih efekata[10][9].

1.2 AES (Advanced encryption standard)

Ovaj standard za enkripciju podataka odobren je od FIPS (Federal Information Processing Standards Publications). [10] Enkriptira i dekriptira 128 bitne blokove i koristi ključeve veličine 128, 192 ili 256 bita.

Kao i većinom algoritama simetrične enkripcije, i AES koristi iste korake u procesu enkripcije i dekripcije. Ovaj algoritam radi na bajtima i to ga čini lakšim za implementaciju i objašnjavanje. Za vreme šifrovanja, ključ

se proširuje u individualne podključeve, koji se koriste za svuku operaciju zasebno. Ovaj process nazivamo ekspanzijom ključeva [11].

Operacije se izvodi više puta od strane standarda AES koji radi sa fiksnim brojem bajti [12].

Ove operacije mogu biti probijene upotrebom nekoliko funkcija koji će biti deo našeg budućeg istraživanja:

- ADD ROUND KEY
- BYTE SUB
- SHIFT ROW
- MIX COLUMN

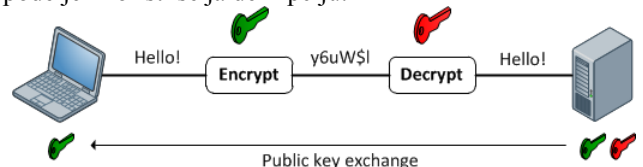
1.3 Drugi poznati simetrični algoritmi

- Triple DES (ECB, CBC), DESX, GDES, RDES – Ključ dužine 168 bita
- (Rivest) RC2, RC4, RC5, RC6 – Promenljiva dužina ključeva do 128 bita
- IDEA – Osnovni algoritam za PGP – ključ dužine 128 bita
- Blowfish – Promenljiva dužina ključeva do 448 bita
- Twofish – Šifrira 128-bitne blokove otvorenog teksta sa dužinom ključeva do 256 bita

Asimetrična enkripcija

Asimetrična enkripcija, poznata još i kao kriptografija javnim ključevima. Razlikuje se od simetrične enkripcije time što se koriste dva ključa, jedan u vreme enkripcije, a drugi za vreme dekripcije. Najčešće korišteni algoritam asimetrične enkripcije je RSA algoritam.

U poređenju sa simetričnom enkripcijom, asimetrična enkripcija nameće veće opterećenje u vreme računanja. Najavljuje se da će biti usavršen. Ali, glavni zadatak ove enkripcije je da uspostavi betbedan medij. Ovo se postiže razmenom javnih ključeva koji se mogu koristiti samo za enkripciju podataka. Dodatni privatni ključ, koji nikada nije spodeljen koristi se za dekripciju.



Slika3. Primer asimetrične enkripcije i dekripcije podataka

Najprije, krajnje tačke razmenuju javne ključeve, što omogućava spor ali siguran protok. Zatim, pošaljilac i primaoc odlučuju da razmene ključeve simetrične enkripcije, čime so omogućava brz protok podataka kroz ove kanale [12].

Algoritmi asimetrične enkripcije

1.1 RSA algoritam

Ovaj algoritam je zasnovan na težini razlaganja velikih brojeva koji imaju samo 2 prosta broja. Javni ključ je dostupan svakome i ovim ključem možemo kriptirati podatke ali ih ne možemo dekriptirati. Ove podatke može dekriptovati samo onaj koji poseduje privatni ključ. Teorijski je moguće, ali u praksi je mnogo teško generirati privatni ključ na bazi znanja javnog ključa. Sve ovo pravi da RSA algoritam bude mnogo popularan izbor za enkripciju podataka [13].

Hashing

Hashing predstavlja forma kriptografske bezbednosti koja se razlikuje od enkripcije. Enkripcija u suštini predstavlja

process u dva koraka, enkripcija i dekripcija. Hashing kondenzuje poruku fiksne dužine. Dva najupotrebljavana algoritma za Hashing su MD5 i SHA-1. Hashing se upotrebljava da bi promenili podatke u bezbednijem obliku, a originalna poruka se ne može vratiti nazad.



Slika 4. Proces Heširanja

Kada koristimo haš za autentifikaciju za bezbedno komuniciranje, on predstavlja rezultat originalne poruke plus tajni ključ. Haš algoritam se koristi i bez tajnog ključa za pruvru grešaka [14].

3. PREGLED PROGRAMA ZA ENKRIPCiju E-MAIL PORUKA

Potreba upotrebe softvera za enkripciju e-mail poruka javlja se bez obzira dali smo uključeni u osetljive finansiske transakcije, dali imamo privatnu bižnis potrebu ili smo deo neke kompanije koja radi sa poverljivim podacima. Cilj je da se uspostavi poverljiva komunikacija sa drugom osobom.



Slika 5. Pregled najboljih enkripcijskih programa za enkripciju e-mail poruka

Ovi softveri omogućavaju besprekornu zaštitu nama i našim podacima. Iako su unešeni algoritmi teški za shvatanje, najvažnije je to što imamo zaštićene poruke.³

Svi programi imaju slične koncepte održavanja privatnosti poruka, ali nisu svi isto efikasni. Programi sa jakom enkripcijom omogućavaju da enkriptujemo e-mail poruke velikom lakoćom. Deo najboljih programa u maju 2013 godine na osnovu rangiranja web sajta www.toptenreviews.com su: Voltage SecureMail, Entrust, Comodo, Symantec i dr. Na slici 5 prikazana su 10 najbolja softverska rešenja koji nude zaštitu e-mail poruka. Obuhvaćeni su delovi: Zaštita podataka, Mogućnosti, Klijenti koji mogu da se povežu (pr. Microsoft Outlook) kao i podržane platforme. Nakon izvršenih ispitivanja kao

najsigurniji se pokazao Voltage koji naplaćuje svoje usluge. Ovaj softver koristi svoj algoritam (IBE Algorithm). Ali, iako je spored njih najsigurniji, ipak nije i najisplativiji. Kao predlog za enkripciju nudimo softver Comodo. Comodo je besplatan i daje jaku enkripciju. Comodo pruža enkripciju putem digitalnih sertifikata i koristi se kao dodatak u nekim klijentima koji se sinhronizuju sa našim e-mailom (Pr. Microsoft Outlook). Uradili smo prakticno ispitavanje programom Comodo i pokazalo se da je lak za upotrebu.

4. POTREBA ZA ENKRIPCiju PODATAKA U R. MAKEDONIJI

Veliki je broj softverskih alata koji se nude za enkripciju na internetu. Deo njih je besplatan, a deo nakon isteka probnog perioda od 30 dana naplaćuju svoje usluge. Sva ova softverska rešenja kreirana su implementacijom nekih od algoritama o kojima smo pisali. U principu skoro svi programi koriste slične algoritme. Ipak je potrebno da se sporede. Ova upoređenje od aspekta preopterećenja računarske konfiguracije, efikasnog rada i potrebnih finansijskih sredstava za nabavku softvera pomoći će nam u odlučivanje koji programa i koja tehnika za enkripciju i dekripciju podataka je najoptimalnija.

U R. Makedoniji je neophodno istraživanje kolika je potreba od upotrebe kriptiranih programa, što računamo kao naš sledeći izazov. Takođe se mora ponuditi i rešenje za korisnike koji nisu upućeni u ovu problematiku. Kao sledeći problem nameće se finansijska mogućnost onih koji imaju potrebu da kriptiraju bilo kakav podatak.

5. PODSTICAJ ZA KORIŠĆENJE ENKRIPCIRANIH TEHNIKA ZA BEZBENOST PODATAKA

Sve nas interesuje koliko korporacijskih podataka se šalje bezbedno i kako se šalju. Često smo svedoci da se u kancelarijama zaboravljaju otvorene e-mail poruke koje mogu da sadrže senzitivne podatke. Odavde dolazi i potreba o upotrebi najoptimalnijih tehnika i zaštita podataka [15].

Urađeno je istraživanje u opštini Negotino i Agenciji za ruralni razvoj R. Makedonije i dobivena je slika koliko su sigurni korporacijski podaci u ovim institucijama i koliko se koriste tehnike enkripcije i dekripcije podataka.

Opština Negotino opšte ne koristi softver za enkripciju i dekripciju podataka, e-mail poruka ili softver za enkripciju celog diskovnog prostora.

U Agenciji za ruralni razvoj koriste napredna softverska rešenja za enkripciju podataka. Koriste softver „PGP“ za enkripciju prilikom razmene podataka, ali koriste i druga softverska rešenja za enkripciju diska i podataka „TrueCrypt“ i „Boxcryptor“. Program Boxcryptor se upotrebljava za mauntiranje disk particije bilo kojeg računara u instituciji, i kasnije se pravi backup (rezervna kopija) na specijalnom „Arkeia backup appliance“.

Na osnovu našeg istraživanja dali smo predlog za izbor optimalne tehnike za enkripciju i dekripciju podataka. Za enkripciju e-mail poruka u obe institucije smo predložili rešenje Comodo, koje nudi jaku zaštitu, ali smo im predstavili i program Voltage koji pruža daleko bolju enkripciju e-mail poruka. Naišli smo na pozitivnu reakciju rukovodstva i zapošljenih, i očekujemo da će isto biti sprovedeno i u drugim institucijama i kompanijama. U agenciji ruralnog razvoja trenutno je dobra enkripcijska zaštita, ali u opštini

Negotino smo predložili više rešenja za enkripciju particije hard diska ili celog diska. U obe institucije smo predložili jednostavan način tekst enkripcije i dekripcije kojeg nude Windows XP, Vista i Windows 7, što dodatno može uticati na zaštitu tekstualnih poruka ili mnogu važnih elektronskih dokumenata. U suštini to su dva VBScript files koji u sebe sadrže kod, jedan za enkripciju, a drugi za dekripciju.

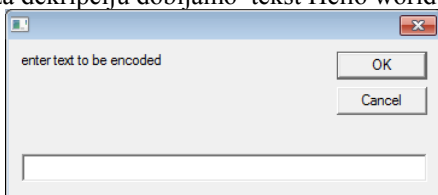


Slika 6. VBScript filesa tekst enkripciju i dekripciju

Ove fajlove kreiramo nakon unošenja u najobicniji text file i snimamo kao „encode.vbs“. Enkriptiracki kod je:

```
'SIMPLE VB ENCRYPTION PROGRAM'
'Create a dialogue box that asks for the text to encode'
set x = WScript.CreateObject("WScript.Shell")
mySecret = inputbox("enter text to be encoded")
'Reverse the submitted text'
mySecret = StrReverse(mySecret)
'Open up an instance of Notepad to print the results after waiting for 1 second'
x.Run "%windir%notepad"
wscript.sleep 1000
x.sendkeys encode(mySecret)'this function encodes the text by advancing each character 3 letters'
function encode(s)
For i = 1 To Len(s)
newtxt = Mid(s, i, 1)
newtxt = Chr(Asc(newtxt)+3)
coded = coded & newtxt
Next
encode = coded
End Function
```

Slično izgleda i program za dekripciju. Primer ako u programu za enkripciju ubacimo tekst Hello world dobijamo kriptiran tekst gourz#roohK. Ako ovaj tekst ubacimo u program za dekripciju dobijamo tekst Hello world.

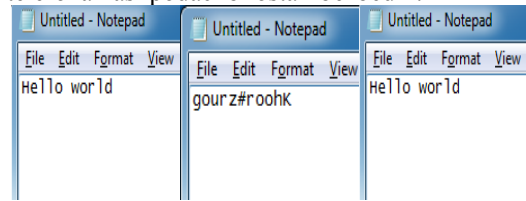


Slika 7. VBScript jednostavan program za tekst enkripciju

VBScript fajlovi su bazirani na RC4 enkriptirackom algoritmu. RC4 se koristi i u WEP, WPA, SSL, BitTorrent, PDF i sl. On je jednostavan za razumevawe i implementaciju. Veliki broj ljudi u R.Makedoniji koji rade sa korporativnim podacima nusu svesni o mogućnosti neželjenih napadača koji za cilj imaju zloupotrebu nekih podataka.

Razgledana je mogućnost funkcioniranja ovih tehnika na više operativnih sistema i platformi, a ne samo na računare, već i za smart telefone, a kao predlog za Android platformu nudimo Secret Space Encrytor. On bezbedno čuva lozinke naših profila na e-mail adresama, kreditnim karticama ili lozinke drugih profila. Takođe, može da enkriptira cele

datoteke telefonskih uređaja, tako da u slučaju gubljenja našeg telefona naši podaci bi ostali bezbedni.



Slika 8. Prikaz tekst enkripcija i dekripcija

6. ZAKLJUČAK

U današnje vreme kada su računari i računarske mreže svuda oko nas, zaštita podataka je sve značajnija. Veliki broj ljudi pokušavaju na razne načine da naruše bezbednost računarskih mreža, da bi nekome naštetili ili imali nekakvu dobit od toga.

Napadači pokušavaju da pročitaju ili da promene sadržaj podataka koja su im dostupna na mreži. Zato je neophodno razviti inteligentne tehnike kako bi računarske mreže i podaci u njima bili bezbedni. Zadovoljavajući rezultati dobiveni upotrebom raznih algoritama za enkripciju su dobra osnova za dalja istraživanja. Tehnike za enkripciju, protokoli autentifikacije i pojava digitalnih potpisa i SSL protokola, omogućavaju siguran način izvođenja bankarskih transakcija i razmena dokumenata velike važnosti.

Danas kada je čitav svet povezan globalnom mrežom, neophodna je kriptografija kako bi imali svoji privatnost.

LITERATURA

- [1] Phil Zimmermann, The Basics of Cryptography, An Introduction to Cryptography, Copyright 1990–2000 Network Associates, Inc. and its Affiliated Companies.
- [2] Peter Mathys, Introduction to Cryptography, ECEN 1200, Telecommunications 1, Fall 2006.
- [3] H.Lee Kwang, Basic Encryption and Decryption, Department of Electrical Engineering & Computer science, KAIST.
- [4] Paul Krzyzanowski, Lectures on distributed systems, Cryptographic communication and authentication, 1997-2009 .
- [5] Dennis Luciano and Gordon Prichett, Cryptology: From Caesar Ciphers to Public-Key Cryptosystems, The College Mathematics Journal, (1987, January, Volume 18.
- [6] Gustavus J. Simmons, Symmetric and Asymmetric Encryption, Sandman Laboratories, Albuquerque, New Mexico 87185.
- [7] KetuFile White Papers, Symmetric vs. Asymmetric Encryption, 2003-2004.
- [8] U.S. Department of commerce/National Institute of Standards and Technology, Data Encryption Standard (DES), (1999, October, 25).
- [9] Proceedings of the IEEE, VOL. 76, NO. 5, The First Ten Years of Public-Key Cryptography, May 1988.
- [10] Randall J. Easter and Carolyn French, Approved Security Functions for FIPS PUB 140-2, Security Requirements for Cryptographic Modules, Information Technology Laboratory, National Institute of Standards and Technology, (2012, May, 30).
- [11] FIPS 197, Announcing the Advanced Encryption Standard (AES), 2001, November, 26.
- [12] Adam Berenet, Advanced Encryption Standard by Example v.1.7, ABI Software Development.
- [13] Evgeny Milanov, The RSA Algorithm, 2009, June, 03.
- [14] Steven M. Bellovin and Eric K. Rescola, Deploying a New Hash Algorithm, Columbia University, 2005. Available: <https://www.cs.columbia.edu/~smb/papers/new-hash.pdf>.
- [15] Chris Brooks, "Computers and Society, Introduction to Encryption, Department of Computer Science, University of San Francisco.

INFORMACIONI SISTEM ZA IZVJEŠTAVANJE O PRODUKTIVNOSTI AGENATA OSIGURAVAJUĆEG DRUŠTVA

INFORMATION SYSTEM FOR PRODUCTIVITY REPORTING OF THE INSURANCE COMPANY AGENTS

Tamara Pejaković, Miloš Orović, Andjela Draganić, Irena Orović, *Elektrotehnički fakultet*

Sadržaj: *Razvoj računara i informacionih sistema dovodi do rasta potreba savremenih preduzeća za postojanjem funkcionalnih i modernih informacionih sistema. Moderni informacioni sistemi treba da omoguće povećanje zarade preduzeća i povećaju njihovu međusobnu konkurentnost. Implementacija i razvoj informacionih sistema se ne može odvojiti od razvoja računara i računarskih tehnologija. Podrška za e-komerc i e-poslovanje prisutna je u sve većem broju savremenih preduzeća. Informacioni sistemi se koriste za obavljanje pojedinačnih zadataka, kontrolu i integraciju podsistema u veću cjelinu. U ovom radu data je analiza sistema za izvještavanje o produktivnosti agenata, a kao dodatak opisana je i programska implementacija sistema. Sistem je implementiran u C++ programskom jeziku, sa podrškom Oracle baze podataka.*

Abstract: *Computer and information systems development increase the need for the existence of functional and modern information systems in the contemporary enterprises. Information systems should facilitate increasment of the profit, as well as to increase competitiveness of the companies. Implementation and information systems development could not be separated from the development of the computers and computer technologies. Support for e-commerce and e-bussines is present in an number of modern enterprises. Information systems are used for specific task performance as well as for control and integration of the subsystems into the whole. In this paper the analysis of the system for agent productivity reporting of incurance companies is described, as well as software implementation of the system. System is implemented in C++ programming language, using the Oracle database.*

1. UVOD

Savremena nauka i njena praksa su danas, više nego ikada ranije, vezane za dostignuća interdisciplinarnih nauka a posebno u oblastima računarske tehnike i informacionih tehnologija. Zbog toga ove tehnologije dobijaju sve značajnije mjesto u poslovanju modernih preduzeća [1]-[3]. Izmijenjeni uslovi poslovanja, i sve brže promjene u načinu funkcionisanja preduzeća akcenat stavljaju na komuniciranje, u cilju ostvarenja interakcije i povezivanja svih elemenata sistema. Intenzivnim razvojem informacionih tehnologija razvijenije zemlje današnjice ulaze u tzv. informatičko društvo koje se značajno razlikuje od industrijskog društva. Uporedo sa energetsom i materijalnom integracijom neophodno je sprovesti i informacionu integraciju. Osim razvoja informacionih tehnologija, aktuelni trend u svijetu je i razvoj vještačke inteligencije, zbog nemogućnosti softverskih aparata da zadovolje određene potrebe savremenog društva.

Neophodno je istaći da su mišljenja o ulozi i značaju informacionih tehnologija u savremenim uslovima poslovanja podijeljena. Sa jedne strane, uvođenje informacionih tehnologija odlaže sagledavanje problema sa kojima smo suočeni u poslovanju. Sa druge strane, ne može se poreći da informacione tehnologije bitno smanjuju troškove a povećavaju profitabilnost preduzeća. Procesi se „informatizuju“ a i sama informacija se posmatra kao roba, pa se naglasak stavlja na način na koji se ona distribuira

krajnjim korisnicima. Pri projektovanju informacionih sistema od velikog je značaja selektovati informacije tako da poruke koje one nose imaju upotrebnost. Takođe, vrijeme potrebno za pristup ovim informacijama treba da bude veoma malo i to je ono čime se upravljamo pri izradi i realizaciji informacionog sistema.

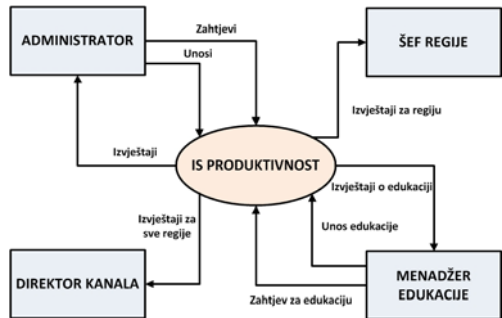
U radu su opisane i analizirane potrebe informacionog sistema za produktivnost agenata nekog osiguravajućeg društva. Informacioni sistem je implementiran korišćenjem C++ programskog jezika i Oracle baze podataka [4]-[8].

Rad je organizovan na sljedeći način: U drugom dijelu opisan je najviši nivo dekompozicije, kao i neki složeniji procesi u sistemu. Treći dio sadrži opis nekih nižih nivoa dekompozicije, dok je programska implementacija sistema opisana u četvrtom dijelu. Zaključak je dat u petoj sekciji.

2. PRVI NIVO DEKOMPOZICIJE SISTEMA

Namjena informacionog sistema je da riješi određeni poslovni problem, a sve u cilju smanjenja troškova, poboljšanja proizvoda i usluga kao i ubrzanju administrativnih poslova [1]. Uspješna primjena informacionih sistema i informacionih tehnologija ima integrativnu funkciju u proizvodnji, menadžmentu, marketingu i računovodstvu. U ovom odrađena je analiza pojedinih djelova i predstavljeno je softversko rješenje

informacionog sistema koji služi za praćenje aktivnosti agenata nekog osiguravajućeg društva.



Slika 1. Dijagram najvišeg nivoa dekompozicije

Na Slici 1 prikazan je najveći nivo dekompozicije informacionog sistema (IS). IS komunicira sa četiri interfejsa: *Administratorom*, *Šefom regije*, *Direktorom kanala* i *Menadžerom edukacije*, dok se glavni proces *IS produktivnost* dijeli na sedam potprocesa, i to: *Baza agenata*, *Produktivnost*, *Eksterni ugovori*, *Edukacija*, *Izmjena naloga*, *Odjavljivanje i prijavljivanje* i *Dodavanje poslovnice* (Slika 2). U nastavku će biti opisani samo neki od procesa, a u narednoj sekciji daćemo detaljniji prikaz programske implementacije sistema.

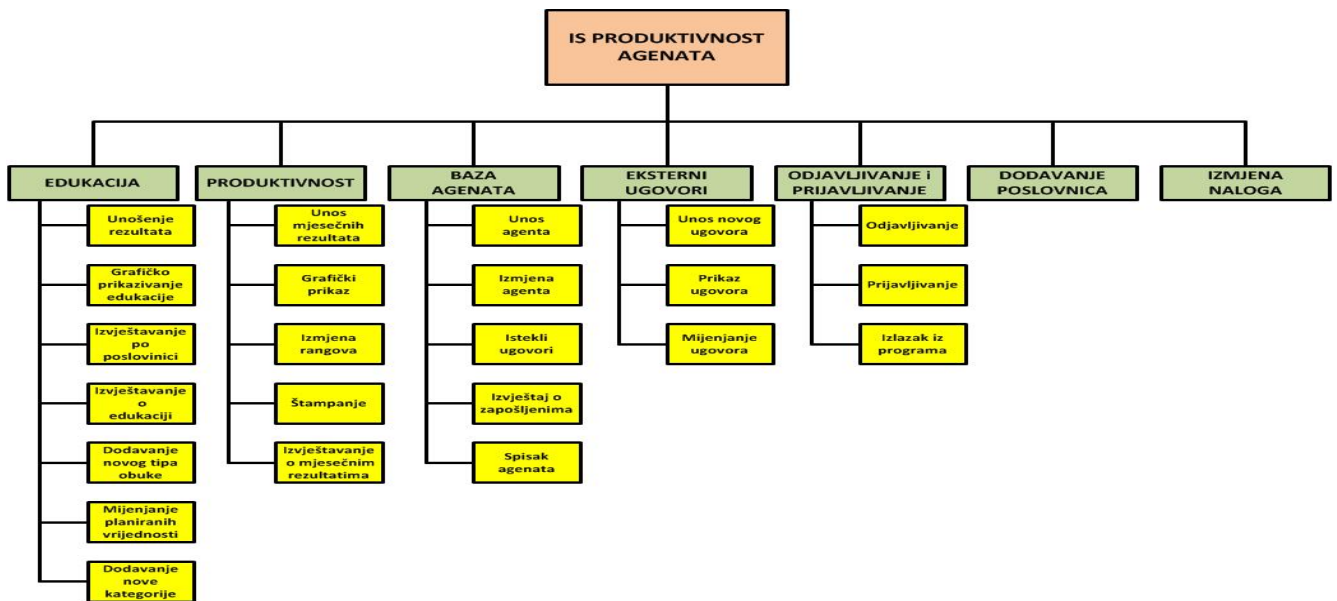
Interfejs *Administrator* od sistema traži određene informacije o agentu, kao što su rezultati testova edukacije, grafički prikaz produktivnosti i sl. (tok podataka *Zahtjevi*), a sistem šalje odgovor, odnosno izvještaj administratoru (tok podataka *Izveštaji*) – Slika 1. Interfejs *Šef regije* dobija izvještaj za svoju regiju, a *Direktor kanala* može dobiti izvještaj za regiju koju želi. *Menadžer edukacije* unosi rezultate edukacije i novi tip edukacije, a može dobiti i grafički prikaz edukacije agenta – Slika 1. Važno je napomenuti da interfejs *Administrator* ima pravo dodavanja nove poslovnice kao i pravo promjene rangova agenata, preko posebnih potprocesa u informacionom sistemu. *Šef regije* ima pravo pregleda izvještaja o mjesečnim rezultatima

za svoju regiju kao i grafičkog prikaza produktivnosti za svoju regiju. *Direktor kanala* ima ista ovlaštenja kao *Šef regije*, za sve postojeće regije.

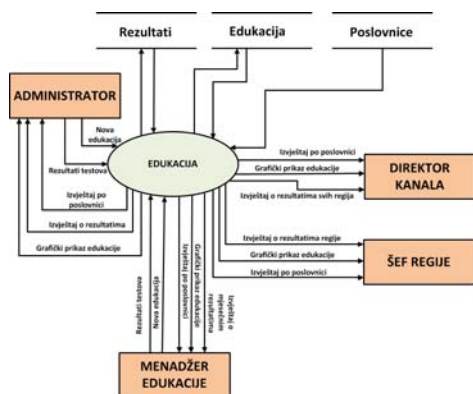
3. NIŽI NIVOI DEKOMPOZICIJE SISTEMA

U nastavku su dati neki niži djelovi arhitekture informacionog sistema za izvještavanje o produktivnosti agenata. Proces *Edukacija* je šematski prikazan na Slici 3. Ovo je proces sa kojim komuniciraju svi interfejsi i kao takav je odabran za šematski prikaz i detaljniju analizu. *Administrator* ima pravo unosa novog tipa edukacije i upisa rezultata testova, nakog čega ažurira baze podataka *Edukacija* i *Rezultati*. Interfejs *Administrator* ima pravo da zatraži izvještaj po poslovnicu, izvještaj o rezultatima testova i grafički prikaz edukacije. Interfejs *Šef regije* nema pravo unosa rezultata, kao ni pravo unosa novog tipa edukacije ili nove kategorije. On dobija rezultate za svoju regiju, grafički prikaz i izvještaj po poslovnicu. *Direktor kanala* se razlikuje od interfejsa *Šef regije* u tome što dobija rezultate po svim regijama. Interfejs *Menadžer edukacije* ima ista prava kao i *Administrator*, pa su i tokovi podataka isti.

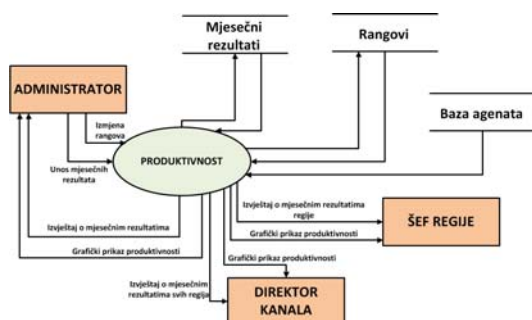
Proces *Produktivnost* (Slika 4) prima, obrađuje i izvještava o svim informacijama koje se tiču učinka agenta u toku mjeseca/godine i namijenjen je rukovodiocima firme. Dijeli na potprocese: *Izmjena rangova*, *Unošenje mjesečnih rezultata*, *Izveštavanje o mjesečnim rezultatima*, *Grafičko prikazivanje*, *Štampanje*. Osim brojanog prikaza, proces *Produktivnost* omogućava i grafički prikaz o produktivnosti agenta (kroz poseban potproces *Grafičko prikazivanje* koji komunicira sa potprocesom *Štampanje* šaljući mu grafike koji se nalaze na formi za štampanje). Slanjem zahtjeva za štampanje potprocesu *Štampanje* korisnik dobija od potprocesa adekvatnu formu sa graphicima i podacima o produktivnosti u pdf formatu pogodnom za štampu.



Slika 2: Podjela procesa IS produktivnost na podprocese



Slika 3. Dijagram procesa Edukacija



Slika 4. Dijagram procesa Produktivnost

4. PROGRAMSKA IMPLEMENTACIJA SISTEMA ZA IZVJEŠTAVANJE O PRODUKTIVNOSTI AGENATA

Informacioni sistem za izvještavanje o produktivnosti agenata predstavlja kompleksan sistem sa mogućnošću upisivanja, iščitavanja, grafičkog prikaza podataka i sl. Kompletan model sistema, idejno rješenje i implementacija predloženo je u [9], a programsko rješenje je kasnije djelimično prevedeno u C++ [10]. Nakon pokretanja programa aktivira se prijavna forma u čija tekstualna polja je potrebno unijeti korisničke podatke. Ukoliko je korisnik pravilno unio administratorske podatke, otvara mu se glavna prelazna forma.

Slika 5: Forma za unos novog agenta

Prelazna forma posjeduje četiri dugmeta: „Baza agenata”, „Produktivnost”, „Edukacija” i „Eksterni ugovori”. Klikom na dugme „Baza agenata” otvara se još jedna prelazna forma na kojoj se nalaze takođe četiri dugmeta: “Unos agenta”, “Izmjena agenta”, „Istekli ugovori” i “Izveštaj o

zaposlenima”. Forma za unos novog agenta prikazana je na Slici 5. Klikom na dugme „Produktivnost” otvara se prelazna forma na kojoj se nalaze dva dugmeta: „Mjesečni rezultati” i „Grafički prikaz produktivnosti”. U formi za unos mjesečnih rezultata (Slika 6) moguće je unijeti rang, broj ugovora, ostvarenu premiju agenta za određeni mjesec i godinu, provjeriti rezultate agenta za određenu godinu, kao i kumulativan broj ugovora i kumulativnu premiju. U formi za grafički prikaz produktivnosti (Slika 7) može se odabrati jedan od ponuđenih grafičkih prikaza i dobiti informacije o broju ugovora sa pravnim licima, prosječnom premijom po agentu u poslovnicima i prosječnoj premiji agenta. Na glavnom meniju mogu se odabrati opcije prikaza štampe i štampanja forme na kojoj se nalaze svi ponuđeni grafički prikazi.

Mjesec	Rang	Broj ugovora	Premija	Br. ugovora procent	Premija procent	Odstupanje ugovora	izmj
Januar	1	7	900	20%	70%	0%	0%
Februar	2	17	1000	39%	50%	0%	0%
Mart	1	56	900	50%	50%	0%	0%
April	2	16	800	20%	24%	0%	0%
Maj	1	4	500	4%	50%	0%	0%
Jun	1	43	900	40%	40%	0%	0%
Jul	3	43	1000	1433%	34%	0%	0%
August	2	49	1000	1633%	340%	0%	0%
Septembar	2	35	900	17%	47%	0%	0%

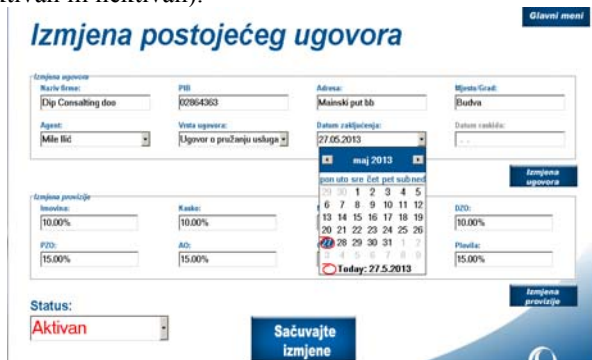
Slika 6: Forma za unos mjesečnih rezultata

Dio informacionog sistema koji se odnosi na ugovore treba da olakša korisniku manipulaciju i pregled svih podataka vezanih za eksterne ugovore, tj. ugovore koje je firma potpisala sa pravnim licima. Korisnik u okviru dijela Eksterni ugovori bira da li želi unos novog ugovora ili prikaz postojećih a unutar prikaza postojećih ugovora korisnik može i izmijeniti postojeći ugovor (Slika 8).

Slika 7: Grafički prikaz produktivnosti agenata

Pri ulasku u formu padajući meni „Agent” se popunjava imenima i prezimenima svih agenata iz baze podataka Baza agenata, padajući meni „Vrsta ugovora” se popunjava iz baze podataka Eksterni ugovori a datum zaključenja se bira iz kalendara. Drugi dio forme služi za izmjenu provizije za pojedine vrste ugovora (Imovina, Kasko, Nezgoda, DZO, PZO, AO, Granično, Plovila) koju korisnik unosi u procentima kao što mu je naznačeno u tekstualnom polju. Na

kraju korisnik bira status ugovora iz padajućeg menija (aktivan ili nektivan).



Slika 8: Izmjena postojećeg ugovora

Edukacija je zasebna cjelina unutar ovog Informacionog sistema sa kojom komuniciraju interfejsi Administrator i Menadžer edukacije koji imaju ista pristupna prava. Agenti nakon određenog vremena stiču pravo da polažu određene tipove obuke kako bi napredovali u karijeri. Agenti mogu više puta polagati iste testove. Sve informacije o rezultatima testova, mentorima, tipu obuke, datumu polaganja i sl. se obrađuju u ovom procesu. Pri ulasku u formu za edukaciju otvara se prelazna forma sa dugmadima „Edukacija agenata”, „Grafički prikaz edukacije” i „Edukacija po poslovnici”.



Slika 9: Edukacija agenata

Dio sistema “Edukacija agenta” prikazana je na Slici 9 i sastoji se iz dva dijela. Dio sa poljima „Agent”, „Kategorija” i „Tip obuke” se automatski popunjava iz baza podataka po otvaranju forme a padajući meni „Planirano” (sa vrijednostima DA i NE) služi za evidenciju da li agent može da polaže određeni test. Drugi dio služi za izvještaj o rezultatima koji korisniku za izabranog agenta i kategoriju vraća tabelarni prikaz vrste osiguranja, datuma polaganja testa, rezultata testa i da li je test odobren a kolona „Planirano” se automatski popunjava. Edukacija po poslovnici omogućava uvid u to koji je tip obuke planiran za određenu poslovnicu, koliko je njih urađeno i koji je broj planiranih obuka. Grafički prikaz edukacije agenata je dat tabelarno u cilju lakšeg snalaženja. Omogućava uvid u rezultate pojedinih kategorija edukacije za odabranog agenta (padajući meni se popunjava automatski iz baze podataka po otvaranju forme) kao i podatke o tome u kojoj poslovnici i regiji radi, ko mu je mentor i koliko dugo je u radnom odnosu (Slika 10).



Slika 10: Grafički prikaz edukacije agenata

5. ZAKLJUČAK

U radu je analiziran informacioni sistem za poslovanje osiguravajućeg društva. Data je analiza prvog nivoa dekompozicije kao i nekih nižih djelova dekompozicije sistema. Dat je i prikaz pojedinih djelova softverske implementacije ovog sistema. Na primjeru ovog informacionog sistema, jasno je koliko je dobar informacioni sistem bitan za funkcionisanje i razvoj malih, srednjih, a naročito velikih preduzeća. Informacioni sistem umnogome olakšava rad preduzeća i čini ga efikasnijim i konkurentnijim na tržištu. Zbog modularnog pristupa u izradi samog sistema, korišćenje istog je mnogo pojednostavljeno i ne zahtijeva posebnu obuku kadra. Jednostavnost u dizajnu je opravdana formalnošću samog sistema i ozbiljnošću poslovanja preduzeća kome je namijenjeno. Integracija informacionih sistema u poslovanju ili bilo kakvoj razmjeni informacije nekog preduzeća predstavlja samo korak naprijed bez obzira na mane koje navode protivnici ove ideje, jer te rijetke greške, koje se i mogu desiti, ne treba da predstavljaju prepreku već motiv ka još boljoj implementaciji informacionih sistema.

LITERATURA

- [1] “IT-Business Alignment: Part I, Effectively aligning IT Systems to your Business Operations”, John Kyriazoglou.
- [2] “Business Information Systems”, Elizabeth Hardcastle.
- [3] “Business Information Systems: Design an App for That,” Raymond Frost, Jacqueline Pike, Lauren Kenyo and Sarah Pels, March 2011 eISBN: 978-1-4533-2949-8.
- [4] B.Swart, M.Cashman, P.Gustavson, „Borland C++ Builder Developer’s Guide”.
- [5] M.McLaughlin, „Oracle Database 11g PL/SQL Programming”.
- [6] S.Meyers, „Effective C++: 55 Specific Ways to Improve Your Programs and Designs”.
- [7] S.Kelby, „The Photoshop Book for Digital Photographers”.
- [8] D.C.Kreines, „Oracle SQL: the Essential Reference”.
- [9] I. Orović, V. Marković, “Program za praćenje produktivnosti agenata osiguravajućeg društva – Uniq osiguranje”, idejno rješenje i implementacija.
- [10] Seminarski rad iz predmeta Informacioni sistemi: “Informacioni sistem za izvještavanje o produktivnosti agenata osiguravajućeg društva”, mentor Doc. dr Irena Orović, studenti: Boris Đurđić, Miloš Orović, Tamara Pejaković, Radomir Starčević.

ANALIZA POTREBA ZA PRIKUPLJANJE PODATAKA, EVALUACIJU I PRAĆENJE SPORTSKIH POVREDA U CRNOGORSKIM SPORTSKIM KLUBOVIMA I SAVEZIMA THE ANALYSIS OF A NEED FOR ACQUIRING DATA, EVALUATION AND MONITORING OF SPORTS INJURIES IN MONTENEGRIN SPORTS CLUBS AND SPORTS FEDERATIONS

Predrag Raković, *Elektrotehnički fakultet, Univerzitet Crne Gore*
Vasilije Stijepović, *Institut za sport i sportsku medicinu Crne Gore*

Sadržaj: *Institut za sport i sportsku medicinu Crne Gore je prikupio podatke o stanju zdravstvene zaštite sportista u Crnoj Gori na osnovu kojih se došlo se do jasne "slike" o stanju sistema zdravstvene zaštite sportista u sportskim klubovima i savezima. Zaključak do kojeg se došlo jeste da je zaštita prepuštena na sprovođenje samim savezima i klubovima i direktno je zavisna od finansijskog stanja u istim. U ovom radu se upoređuje postojeće stanje protoka i čuvanja informacija o povredama sportista u klubovima i savezima i daje se unapređenje ovoga sistema kroz prijedlog informacionog sistema. Ponuđeno rješenje, umnogome bi olakšalo prikupljanje i transfer informacija, kao i samu evaluaciju i praćenja sportskih povreda.*

Abstract: *Institute for Sports and Sports Medicine of Montenegro has acquired data of the state of health of athletes in Montenegro, and led to a clear insight in the state of the health care of athletes in sports clubs and federations. The conclusion to which research led was that the implementation of health protection was left to the federations and clubs, but it is directly dependent on the financial situation in them. In this paper we compare the current system on flow and storage of information on injuries in sports clubs and federations and we propose the improvement of this system through implementation of information system. Offered solution would greatly relieve collection, transfer and evaluation and monitoring of sports injuries.*

1. UVOD

Veliki broj naših vrhunskih sportista, ali i ostalih mladih ljudi koji se bave sportom, izložen je velikim zdravstvenim rizicima zbog nepostojanja adekvatne i organizovane zdravstvene zaštite. Ovo veoma često rezultira ozbiljnim zdravstvenim poslasticama, preranim završetkom karijere i postizanju znatno slabijih takmičarskih rezultata [1],[2],[6],[7],[9]. Problemi evaluacije i praćenja sportskih povreda najjasnije se sagledavaju iz ugla sportkog ljekara. Kada znamo da ovu djelatnost obavljaju ljekari različitih specijalnosti, uloga i odgovornost klupskog ljekara mora biti jasno definisana. Sportska povreda je definisana kao povreda muškarca i žene u svim životnim dobima koji učestvuju pojedinačno ili grupno u organizovanim i neorganizovanim sportskim aktivnostima, a uzrokuju najmanje jedan dan odsustvovanja sa sportskog terena [2],[3],[8]. Ista povreda u opštoj populaciji ne mora imati istu težinu po rad i život pojedinca, kao povreda u sportu, koja uzrokuje odsustvovanje sa terena najmanje jedan dan. U svijetu profesionalnog sportiste dvadeset četiri časa odsustvovanja znače mnogo.

Zdravstvena zaštita sportiste obuhvata:

- preventivni zdravstveni pregled (regulisan Zakonom o sportu),
- funkcionalna testiranja, borbu protiv dopinga i
- djelokrug rada klupskog ljekara.

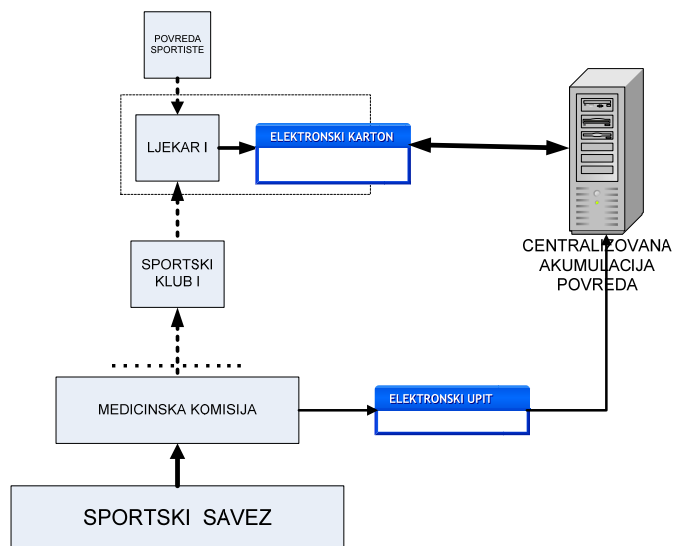
Sportski klub je relativno zatvorena sredina, ima specifičnosti koje su vezane za vrstu sportske aktivnosti, veliki broj članova, određenu hijerarhiju u kolektivu, zajedničke i pojedinačne interese, prije svega sportiste, a zatim i uprave kluba.

Timski doktor u ovako specifičnom sistemu mora sam da se izbori za svoje mjesto, i to znanjem, autoritetom i angažovanjem u timu, čuvajući osnovne principe medicinske doktrine. Klupski ljekar je odgovoran za zdravstveno stanje i bezbjednost sportiste, poštovanje pravila medicinske profesije, poštovanje pravila upravnih tijela kluba i saveza, kao i čuvanje povjerljivosti podataka.

Dužnosti klupskog ljekara su:

- da kontroliše zdravstveno stanje sportiste,
- da po potrebi pruži prvu pomoć, izvrši trijažu povreda na terenu, obezbijedi adekvatno liječenje bolesti i povreda,
- da koordinira proces rehabilitacije i na kraju, da procijeni mogućnost povratka sportiste treninzima i takmičenjima.

Visoka fizička, psihološka i ekonomska cijena povrede sportiste je razlog da na međunarodnom nivou bude prepoznata potreba za adekvatnim sistemom praćenja povređivanja sa ciljem smanjenja incidence i ozbiljnosti povreda. Van Mechelen i ostali 1992.,[3], predlažu algoritam za organizovanje istraživanja na polju prevencije sportskih povreda, a 2006. godine Finch C. [4] predlaže proširenje tog modela da bi se prevazišao problem implementacije naučno potvrđenih preventivnih mjera u praksi.



Slika 2. Prijedlog inofmacionog sistema za prikupljanje podataka, evaulaciju i praćenje sportskih povreda u sportskim klubovima i savezima u Crnoj Gori

Na Slici 2. je dat prijedlog informacionog sistema kojim se unaprijeđuje postojeće stanje u klubovima i savezima. Ovakav sistem predviđa:

- uvođenje elektronskog formulara za unos podataka o tipu, prirodi i poslasticama sportske povrede napravljen na osnovu okvira predloženog u [3],[4],[9],
- popunjavanjem ovog formulara od strane klubbkog ljekara se prikupljene i unešene informacije direktno upisuju u centralizovanu akumulaciju podataka
- centralizovana akumulacija podataka bi bila dostupna kako svim klubbkim ljekarima tako i medicinskoj komisiji u okviru sportskog saveza,
- uvođenje elektronskog formulara za upit o vrsti i broju sportskih povreda.

Centralizovana akumulacija podataka, dakle, bi bila baza podataka iz koje bi medicinska komisija sportskog saveza putem elektronskog formulara za upit o vrsti i broju sportskih povreda dobijala kontinuiranu i pravovremenu informaciju. Sa druge strane, prilikom transfera sportiste iz jednog kluba u drugi ažuriranje podataka o sportskim povredama samog sportiste bi preuzeo klubbki ljekar novog kluba. Ovim predloženim sistemom se obezbjeđuje konzistentnost, trajnost i validnost prikupljenih podataka o sportskim povredama.

4. ZAKLJUČAK

Danas u Crnoj Gori ne postoje validni statistički podaci koji su objavljeni i javno dostupni o povređivanju sportista. Da bi se unapredili zdravlje i sigurnost u sportu potrebno je započeti prikupljanje podataka o povredama na uniforman i sistematski način. Predloženo IT rešenje prikupljanja podataka o sportskim povredama kao i njihova evaulacija i praćenje primjenom standardizovanih statističkih parametara i formulara za prikupljanje podataka može samo doprinijeti razvitku sporta u Crnoj Gori i njegovom prosperitetu kako na nacionalnom tako i na internacionalnom planu. Predloženo idejno rešenje omogućilo bi redovno prikupljanje podataka o povredama kao i analizu mehanizama povređivanja, uzroka i

faktora rizika a sve u cilju prepoznavanja i odabira adekvatnih preventivnih mjera i medicinskih tretmana. Krajnji cilj ovoga rada jeste predlog IT rješenja kojim bi se olakšalo stvaranje dostupne centralizovane akumulacije podataka koja bi bila u službi unapređenja zdravlja sportista i sigurnosti u sportu.

LITERATURA

- [1] Drawer S, Fuller CW. „Evaluating the level of injury in English professional football using a risk based assessment process.” *Br. J. Sports Med*, 2002.
- [2] Fuller CW, Ekstrand J, Andersen TE, Bahar R, Dvorak J. „Consensus statement on injury definitions and data collection procedures in studies of football (soccer) injuries.” *Br. J. Sports Med.*,2006.
- [3] Van Mechelen W, Hlobil H, Kemper HC.” Incidence, severity, aetiology and prevention of sports injuries. A review of concepts.”, *Sports Med*. 1992.
- [4] Finch C. “A new framework for research leading to sports injury prevention.” *Journal of Science and Medicine in Sport* 2006.
- [5] Goldberg AS, Moroz L, Smith A, Ganley T. “Injury Surveillance in Young Athlete.” *Sports Med.*, 2007.
- [6] Knowles SB, Marshall SW, Guskiewicz KM. “Issues in estimating Risks and Rates in sports injurie research.” *Journal of Athletic Training*, 2006.
- [7] Dick R, Agel J, Marshall SW. “National Collegiate Athletic Association Injury Surveillance System Commentaries: Introduction and Methods.” *Journal of Athletic Training* 2007.
- [8] Hagglund M, Walden M, Ekstrand J. “Previous injury as a risk factor for injury in elite football: a prospective study over two consecutive seasons.” *Br. J. Sports Med*, 2006.
- [9] Oblaković-Babić J., Dikić J.,“ Evaluacija i praćenje povreda u sportu – pristup KK FMP i UEFA”, *Sportska medicina*, 2008.

ISTRAŽIVANJE NEFUNKCIONALNIH ZAHTJEVA INFORMACIONIH SISTEMA NON-FUNCTIONAL REQUIREMENTS ELICITATION OF INFORMATION SYSTEM

Bogdan Mirković, Fakultet za informacione tehnologije, Slobomir P Univerzitet

Sadržaj: Razvojni timovi informacionih sistema (IS) su pod stalnim pritiskom razvoja programskog koda na vrijeme i u okviru predviđenih finansijskih sredstava. Kako složenost IS raste, a zahtjevi naručioca postaju složeniji, kao i zahtjevi za kvalitetom IS-a, nefunkcionalna svojstva IS više se ne mogu smatrati sekundarnim. Proces istraživanja zahtjeva predstavlja jednu od kritičnih faza u inženjeringu zahtjeva. U radu su predstavljene osnovne karakteristike nefunkcionalnih zahtjeva, pristupi i tehnike za njihovo istraživanje s posebnim osvrtom na četvoronivovski ciljno-orjentisani pristup.

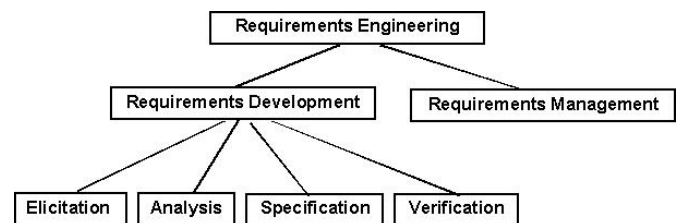
Abstract: Information systems developers are constantly under pressure to deliver code on time and on budget. As information system grows and clients demand higher and higher quality of information system, non-functional properties can no longer be considered to be of secondary importance. The process of requirements elicitation is generally accepted as one of the critical activities in the Requirements Engineering process. In this paper are presented the basic characteristic of non-functional requirements, techniques and approaches for non-functional requirements elicitation with special reference on four-layered goal-oriented approach.

1. UVOD

Razvojni timovi informacionih sistema (IS) su pod stalnim pritiskom razvoja programskog koda na vrijeme i u okviru predviđenih finansijskih sredstava. Kao rezultat toga, mnogi projekti razvoja IS se fokusiraju na razvoj funkcionalnosti IS na račun zahtjeva koji nisu funkcionalne prirode, kao što su, između ostalih, pouzdanost, bezbjednost, aspekti održavanja, prenosivost. Kako složenost IS raste, a zahtjevi naručioca postaju složeniji, kao i zahtjevi za kvalitetom IS-a, nefunkcionalna svojstva IS više se ne mogu smatrati sekundarnim. Mnogi IS koji su razvijeni povučeni su iz upotrebe upravo zbog neadekvatnih ispunjenja kvalitativnih zahtjeva. Iako su ovi aspekti IS-a bili istraživani od strane naučne zajednice od samih početaka primjene IS-a, raniji naučni radovi iz ove oblasti su težili da ih vide kao osobine ili attribute gotovog IS-a i vršili su njihovu procjenu i mjerenje. U radovima novijeg datuma, istraživači zauzimaju komplementaran stav da se nefunkcionalnim zahtjevima treba baviti i izučavati ih od najranijih faza razvoja IS, a zatim pratiti kroz cijeli životni vijek [1], [2].

Proces istraživanja (iskopavanja, traženja, izučavanja, iznošenja na vidjelo, engl. elicitation) zahtjeva predstavlja jednu od kritičnih faza (kao poddisciplina) u inženjeringu zahtjeva (engl. Requirement Engineering) (Slika 1.) [3]. Nefunkcionalni zahtjevi IS-a (NFZ) su takođe u literaturi poznati i pod pojmom kvalitativni zahtjevi (zahtjevi za kvalitetetom) [2], [4] i, za razliku od funkcionalnih zahtjeva, NFZ sadrže ograničenja na IS-u, kao i pojedinim kvalitativnim dijelovima sistema kao što su preciznost, upotrebljivost, bezbjednost, performanse, pouzdanost. Svaki zahtjev može se najšire prikazati preko dva povezana izraza: 1) svaki sistem treba da omogući uslugu korisniku sistema. Svaka od usluga mora zadovoljavati određena ograničenja u cilju zadovoljavanja potreba korisnika IS-a, 2) zahtjevi

sadrže šta sistem i kako da uradi. Može se reći da funkcionalni zahtjevi su povezani sa odgovorom na pitanje “šta“ sistem treba da uradi, dok su NFZ povezani sa odgovorom na pitanje “kako“ sistem to treba da uradi. Kao posljedica toga, NFZ su uvijek povezani sa funkcionalnim zahtjevima [5], [6].



Slika 1. Inženjering zahtjeva i poddiscipline [3]

Nefunkcionalni zahtjevi predstavljaju ograničenja i kvalitete. Kvaliteti su osobine ili karakteristike IS koje naručilac potražuje i koje će uticati na stepen njegovog zadovoljstva sistemom. Ograničenja nisu predmet pregovaranja, za razliku od kvaliteta, jer su striktna i ne mogu se davati ustupci za vrijeme dizajna. Pored navedenog postoji više drugih termina za nefunkcionalne zahteve, kao što su ograničenja, kvalitativni atributi, kvalitativni ciljevi, kvalitet servisnih zahtjeva [6].

2. NEFUNKCIONALNI ZAHTJEVI

Funkcionalnim zahtjevima definišu se konkretni problemi i zbog toga se obično njihovo zadovoljavanje sprovodi pojedinim modulima ili komponentama IS-a. Iako se često navode neformalno (usmenih putem ili u obliku opisnog teksta) oni mogu biti formalizovani kada god je to potrebno uraditi. S druge strane, NFZ definišu globalna ograničenja IS ili dijela IS, ograničenja na funkcionalnim zahtjevima, na

proces razvoja ili proces primjene IS. Oni su globalni u smislu da su nastali iz svih dijelova sistema ili pojedinih komponenti. Za predstavljanje funkcionalnih zahtjeva postoje razvijene notacije (Use Cases/UML). U poređenju sa njima, NFZ se mnogo teže formulišu ili formalno označavaju. Kao rezultat toga, NFZ su često neformalno predstavljeni u projektnoj dokumentaciji, što ih čini teškim za identifikovanje i sprovođenje u toku razvoja i teško ocjenjivim od strane naručioca IS prije njegove isporuke i rada u realnim uslovima. Naučni radovi ranijeg perioda koji su se bavili NFZ razmatrali su mjerenja u cilju dobijanja rezultata koji su pokazivali koliko je razvijeni IS u skladu sa NFZ koji su postavljeni pred IS koristeći određene forme kvantitativne analize [8]-[10]. U posljednje vrijeme, u većem broju radova predloženi su pristupi koji se eksplicitno bave NFZ i prije mogućnosti njihovog mjerenja kvantitativnim metodama u realnom okruženju [2], [6], [11]. Ovi radovi predlažu korištenje tehnika koji omogućavaju donošenja odluka o uključivanju ili isključivanju uslova koji mogu uticati na razvoj IS. Za razliku od pristupa koji su podrazumijevali mjerenje ispunjenosti zahtjeva, pristupi i tehnike predloženi u ovim radovima se tiču i definisanja i objašnjavanja važnosti ispunjavanja NFZ kao relevantnih i važnih dijelova procesa razvoja IS. Kirner [11] je definisao opšte NFZ koje IS treba da zadovolji:

- Performanse (performance) sistema se odnose na vrijeme odziva (response time) zahtjevanih funkcija, propusnu moć (throughput) mreže kroz koju prolaze podaci, tačnost (accuracy) izvršenja funkcija, mogućnost korišćenja odnosno raspoloživost (availability) funkcija sistema i način korišćenja raspoloživih resursa (resource usage) sistema.
- Pouzdanost (reliability) predstavlja sposobnost sistema da može uspješno obraditi problem (failure) koji se dešava u toku izvršenja pojedinih komponenti sistema ili kompletnog sistema. U tom smislu sistem mora da obezbedi način oporavka (recoverability) podataka u slučaju nasilnog prekida rada sistema. Takođe sistem treba da omogući predviđanje (predictability) mogućih ponašanja sistema.
- Bezbjednost (safety) sistema predstavlja skup fizičkih (video nadzor, zaštita prostorija, fizička kontrola pristupa), administrativnih (sigurnosne politike, pravilnici, procedure) i tehničkih (lozinke, programsko šifrovanje, prava pristupa) mjera koja je povezana sa zaštitom (security) sistema.
- Održavanje (maintability) sistema predstavlja skup osobina sistema koje se odnose na jednostavnost otklanjanja grešaka koje su ostale u sistemu nakon implementacije u realnom okruženju, omogućavanje rada sistema, minimiziranje troškova usljed zastoja ili prekida rada sistema. Sa ovim zahtjevima povezani su i podrživost (adaptability) sistema koja se odnosi na lakoću njegovog prilagođavanja novim zahtjevima, internacionalizaciju (internationalization) u smislu njegove prilagodljivosti različitim znakovnim sistemima koji se koriste u svijetu i načinu konfigurisanja (configurability) sistema [12].
- Upotrebljivost (usability) predstavlja sposobnost sistema da se može jednostavno koristiti. To se postiže pomoću

raznih uputstava i dokumentacije koji opisuju način njegovog korišćenja.

Proces istraživanja zahtjeva obuhvata skup aktivnosti koje moraju omogućiti komunikaciju, određivanje prioriteta, pregovaranje i dogovaranje, kao i saradnju svih zainteresovanih strana. Pod pojmom zainteresovane strane u projektima razvoja, implementacije i primjene IS podrazumijevamo bilo kojeg pojedinca, grupu ili organizaciju koja može uticati ili na koju utiče (pozitivno ili negativno) posmatrani informacioni sistem i koji imaju direktan ili indirektan uticaj na definisanje zahtjeva sistema [13]-[15]. Takođe, proces istraživanja mora pružiti snažan osnov za otkrivanje i pronalaženje zahtjeva kao dio visoko interaktivnog istraživačkog procesa. Tipične aktivnosti u procesu istraživanja zahtjeva mogu se podijeliti u pet grupa: razumijevanje domena primjene IS, identifikovanje izvora zahtjeva, analiza zainteresovanih strana, izbor tehnika i pristupa, primjena tehnika i pristupa u istraživanju zahtjeva.

Osnovna klasifikacija pristupa istraživanju NFZ podrazumijeva dva načina istraživanja: proizvodno-orjentisana i procesno-orjentisana. Proizvodno orjentisani pristupi su oni koji se bave mjerenjem kojima se ustanovljava stepen u kome IS zadovoljava definisane NFZ. Tehnike iz ovog pristupa ne služe za prevenciju problema koji nastaju u raznim životnim fazama IS (posebno u razvojnoj fazi), ali su korisne za ocjenjivanje stepena uspješnosti implementacije NFZ. Procesno-orjentisani pristupi se fokusiraju na proces razvoja IS. Njihova svrha je omogućavanje razvojnim timovima IS-a dovoljno dobre alternative ispunjavanja NFZ u samom procesu razvoja IS. Za ovu vrstu pristupa postoji više okvirno definisanih pristupa kao što su ISO9126 ili ciljno-orjentisani pristup kao što je KAOS. Jedna od osnovnih prednosti ciljno-orjentisanih pristupa je ta da ovaj način pristupa može biti korišten i za istraživanje funkcionalnih, ali i nefunkcionalnih zahtjeva. Pored ciljno-orjentisanog, najčešće korišteni pristupi i tehnike su:

- Analizom domena (domain analysis) ispituje se postojeća dokumentacija iz domena primjene IS, kao i postojeći IS ili aplikativna rješenja. Ovo je koristan način da bi utvrdili rani zahtjevi IS-a, razumijevanje domena sistema i komponenti sistema koji se višestruko ponavljaju.
- Rad u grupama (groupwork) predstavlja jednu od osnovnih metoda i pristupa u istraživanjima zahtjeva. Ova metoda izvršava se u grupama u kojima grupa stručnjaka iz oblasti informacionih sistema i/ili drugih učesnika kroz interakciju među učesnicima istražuje problem.
- Intervju (interview) predstavlja metodu za prikupljanje informacija od zainteresovanih strana za razvoj IS koja služi za pronalaženje konkretnih informacija koje projektanti sistema potražuju od jedne osobe. Poznata je i kao metoda licem u lice (face-to-face). Potencijalna zainteresovana strana mora biti upoznata sa ciljevima koji se očekuju od sistema. Kao priprema za ovakve intervjue potrebno je pripremiti listu pitanja kojim će se od intervjuisane osobe dobiti dodatne informacije oko kojih je u ranijem postupku identifikacije zainteresovanih strana bilo konflikata. Prednost ove metode je u tome što

intervjuisana lica nemaju pritisak koji se može pojaviti kada se nalaze u grupi.

- Etnografija (ethnography) predstavlja skup tehnika u kome se posmatraju budući korisnici IS u njihovom okruženju (radnom, životnom). Ona se koristi za kvalitativni opis pojava i radnih procedura. Predstavlja holističko istraživanje zasnovano na ideji da se osobine sistema ne mogu nužno razumjeti na pouzdan način nezavisne jedne od drugih. Metoda se koristi i za istraživanje zavisnosti među zahtjevima.
- Prototipski (prototyping) pristup: omogućavanje zainteresovanim stranama da prototipom IS i njihovim korištenjem daju sugestije i prijedloge koje krajnji IS treba da zadovolji.
- Scenariji su široko rasprostranjeni u istraživanjima zahtjeva i, kao što samo ime govori, narativnog su tipa i njima se specificiraju opisi trenutnih i budućih procesa uključujući i interakcije između sistema i korisnika.
- Pristupima gledišta (viewpoint) modeluje se domen u kome se razvija IS sa različitih gledišta da bi se razvio kompletan i konzistentan opis ciljanog sistema.

Svaka od nabrojanih tehnika, metoda i pristupa ima svoje prednosti i nedostatke i može biti manje ili više prihvatljiva u inženjeringu zahtjeva što zavisi od okolnosti i vrsti IS koji se razvija. Određene tehnike i pristupi su prilagodljiviji u procesu realizacije aktivnosti u istraživanju zahtjeva što zavisi od tipa informacija koje su potrebne u određenom dijelu procesa istraživanja zahtjeva. U Tabeli 1. prikazane su grupe tehnika i pristupa koje su najbolje za specifične aktivnosti u istraživanju zahtjeva.

Tabela 1. Tehnike i pristupi u istraživanju zahtjeva [16]

	Intervju	Domen	Rad u grupi	Etnografija	Prototipiranje	Cilj	Scenario	Tačka gledišta
Razumijevanje domena	X	X	X	X		X	X	X
Izvori zahtjeva	X	X	X			X	X	X
Analiza zaint. strana	X	X	X	X	X	X	X	X
Izbor tehnika i pristupa	X	X	X					
Istraživanje zahtjeva	X	X	X	X	X	X	X	X

Velika većina projekata razvoja i implementacije IS zahtjeva angažovanje većeg broja tehnika i pristupa ali su rijetki slučajevi kada ih je sve moguće upotrijebiti u jednom projektu. Zbog toga je potrebno izabrati tehnike i pristupe koje smatramo da će dati najbolje rezultate. U nekim slučajevima, alternativne tehnike mogu biti od koristi a neke se mogu koristiti kao komplementarne. U Tabeli 2. dat je prikaz komplementarnih (označenih u presjeku vrsta i kolona u tabeli sa K) i alternativnih tehnika i pristupa (označenih u presjeku vrsta i kolona sa A).

Tabela 2. Komplementarne i alternativne tehnike i pristupi [16]

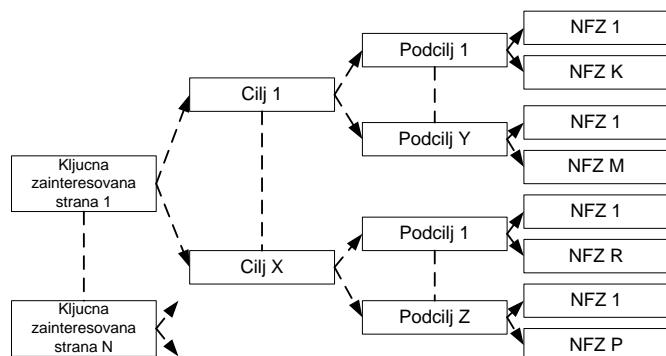
	Intervju	Domen	Rad u grupi	Etnografija	Prototipiranje	Cilj	Scenario	Tačka gledišta
Intervju		K	A	A	A	K	K	K
Domen	K		K	A	A	A	A	A
Rad u grupi	A	K		A	K	K	K	K
Etnografija	A	A	A		K	K	A	A
Prototipiranje	A	A	K	K		K	K	K
Cilj	K	A	K	K	K		K	K
Scenario	K	A	K	A	K	K		A
Tačka gledišta	K	A	K	A	K	K	A	

3. 4-NIVOVSKI CILJNO-ORJENTISANI PRISTUP

Osnovna karakteristika ciljno-orjentisanih pristupa je da se ciljevi predstavljaju na visokom nivou koji predstavljaju ciljeve definisane kompletnim sistemom dekomponuju (koristeći najčešće relacije I i ILI) i razrade (koristeći pitanja Zašto i Kako) u podciljeve na osnovu kojih se vrši istraživanje i definisanje pojedinih zahtjeva. Rezultati ovakvih procesa su komplikovaniji ali i daju kompletniji prikaz nego tradicionalne metode prikazivanja ciljeva koje koriste strukturu stabla. Ovim pristupima se predstavljaju detaljni odnosi domena sistema, ciljeva i zahtjeva. U principu, jedan od osnovnih rizika koji je javljaju korištenjem ovih pristupa jeste mogućnost pogrešnog definisanja ciljeva na visokom nivou iz kojih slijede greške u podciljevima što čini veoma teškim za upravljanjem kasnije aktivnosti koje slijede u realizaciji procesa istraživanja. Korištenje ciljno-orjentisanih pristupa u konjukciji sa scenarijima omogućava da se u praksi ova dva pristupa koriste u situacijama kada su poznati i dobro definisani ciljevi na visokom nivou a nedostaje saznanja o specifičnim problemima i zahtjevima u njihovoj implementaciji.

U četvoronivovskom pristupu (four layered approach) koristi se generalizovan pristup zasnovan na gledištu zainteresovanih strana. Korištenjem ovog pristupa definišu se ciljevi, podciljevi i na kraju NFZ. Koristi se za identifikovanje svih NFZ koji se definišu od strane ključnih zainteresovanih strana (key stakeholder). Pod pojmom ključnih zainteresovanih strana definišu se oni pojedinci, grupe i organizacije koje imaju značajan uticaj na uspjeh sistema. Pristup koristi četvoronivovsku nalaizu kao što je to prikazano na Slici 2. i to: identifikovanje ključnih zainteresovanih strana, definisanje ciljeva zainteresovanih strana, dekompozicija ciljeva u podciljeve, identifikacija NFZ za svaki od podciljeva. Nivovski pristup ima svoje prednosti u implementaciji [17]: skalabilnost (bolje prikazuje važnosti pojedinih nivoa nego horizontalni pristup), bolja fleksibilnost (omogućava se bolji pregled opcija i izbora), cijena (nivovskim pristupom cijena razvoaj i implementacije sistema se smanjuje), segmentacija (dijeljenjem kompleksnijih pitanja u manje kompleksne komponente omogućava brži razvoj i implementaciju sistema), poboljšano razumijevanje

(nivo omogućavaju ispitivanje pojedinačnih komponenti potrebnih za svaki nivo kao i za kompletan sistem).



Slika 2. Četvoronivovski ciljno-orjentisani pristup [17]

Za pronalaženje kritičnih NFZ u ovom pristupu koristi se matrica. Primjer matrice dat je u Tabeli 3. U prikazanoj matrici sa C1, C2,..., Cm prikazani su ciljevi, dok su sa NFZ 1, NFZ 2, ..., NFZ k prikazani nefunkcionalni zahtjevi. Sa matrice se može vidjeti da je kritični NFZ u ovom ilustrativnom primjeru NFZ 2 jer predstavlja NFZ dobijen analizom ciljeva C1, C2, C3, C5, C6, Cm.

Tabela 3. Matrica za pronalaženje kritičnih NFZ [17]

	C1	C2	C3	C4	C5	C6	C7	Cm
NFZ 1	X				X			X
NFZ 2	X	X	X		X	X		X
NFZ 3		X				X	X	
NFZ 4			X	X				
NFZ k							X	X

Prednosti ovakvog pristupa su: nije potrebno modelovanje što omogućava uštedu vremena za analizu i razvoj, kako su identifikovani NFZ veća je mogućnost prihvatanja IS od strane korisnika, svi zahtjevi su analizirani sa stanovišta i gledišta zainteresovanih strana, funkcionalni zahtjevi koji se utvrđuju na osnovu ciljeva su povezani sa NFZ za mogućnost validacije, ovakav pristup može biti korišten nezavisno od metodologije i životnog ciklusa IS.

4. ZAKLJUČAK

Prihvatanje IS od strane naručioca ili korisnika zavisi u mnogome od ispunjenja kako funkcionalnih, tako i nefunkcionalnih zahtjeva te načina njihove implementacije u IS. Svaka od prikazanih tehnika i pristupa u istraživanju zahtjeva ima svoje prednosti i nedostatke koje treba uzeti u obzir prilikom njihovog izbora u procesu inženjerstva zahtjeva. Mogućnosti korištenja tehnika i pristupa kao alternativnih nekim drugim tehnikama i pristupima prikupljanja informacija u procesu istraživanja zahtjeva omogućava brži i jefinitiji razvoj i implementaciju IS. Međutim, njihova implementacija mora biti zasnovana na odličnom poznavanju njihovih karakteristika i mogućnostima njihove komplementarne primjene.

LITERATURA

- [1] L.M. Cysneiros, J.C.S.P. Leite, "Integrating Non-Functional Requirements into data model", 4th International Symposium on Requirements Engineering – Ireland June 1999.
- [2] L.M. Cysneiros, J.C.S.P. Leite, J.S.M. Neto, "A Framework for Integrating Non-Functional Requirements into Conceptual Models" Requirements Engineering Journal – Vol 6, Issue 2, pp. 97-115, Apr. 2001.
- [3] K. Wiegers, "Automating Requirements Management," Software Development, Vol. 7, No. 7, pp. S1-S5, July 1999.
- [4] L. Chung, "Representing and Using Non-Functional Requirements: A Process Oriented Approach" Ph.D. Thesis, Dept. of Comp. Science. University of Toronto, June 1993.
- [5] Evaluation of Natural Language Processing Systems, <http://www.issco.unige.ch>
- [6] L. Chung, B. Nixon, E. Yu, J. Mylopoulos, "Non-Functional Requirements in Software Engineering" Kluwer Academic Publishers, 2000.
- [7] A. Stellman, J. Greene, „Applied Software Project Management“, O'Reilly Media 2005. p. 113., 2005.
- [8] B. Boehm, "Characteristics of Software Quality" North Holland Press, 1978.
- [9] N.E. Fenton, S.L. Pfleeger, "Software Metrics: A Rigorous and Practical Approach" 2nd ed., International Thomson Computer Press, 1997.
- [10] S.E. Keller, "Specifying Software Quality Requirements with Metrics" in Tutorial System and Software Requirements Engineering IEEE Computer Society Press, pp. 145-163, 1990.
- [11] T.G. Kirner, A.M. Davis, "Nonfunctional Requirements of Real-Time Systems", Advances in Computers, Vol 42 pp. 1-37, 1996.
- [12] S. Vlajić, "Projektovanje softvera“, Fakultet organizacionih nauka, Univerzitet u Beogradu, Beograd 2009.
- [13] G. Kotonya, I. Sommerville, "Requirements engineering processes and techniques“, Wiley, New York, 2003.
- [14] A. Pouloudi, "Aspects of the stakeholder concept and their implications for information systems development“, 32nd Annual Hawaii international conference on system sciences, 1999.
- [15] H. Sharp, A. Finkelstein, G. Galal, "Stakeholder identification in the requirements engineering process“, DEXA Workshop, pp. 387-391, 1999.
- [16] D. Zowghi, C. Coulin, "Requirements Elicitation: A Survey of Techniques, Approaches, and Tools“, in Engineering and managing software requirements, Springer Berlin Heidelberg, pp. 19-46, 2005.
- [17] A.A. Rao, M. Gopichand, "Four Layered Approach to Non-Functional Requirements Analysis“, International Journal of Computer Science Issue, Vol. 8, Issue 6, No. 2, November 2011.

KRITERIJUMI ZA MJERENJE USPJEŠNOSTI INFORMACIONIH SISTEMA CRITERIA FOR MEASURING SUCCESS OF INFORMATION SYSTEM

Bogdan Mirković, *Fakultet za informacione tehnologije, Slobomir P Univerzitet*

Sadržaj: *Razvojni timovi informacionih sistema moraju znati šta su ciljevi razvoja IS, dok naručilac IS mora znati u kojoj su mjeri ispunjeni zahtjevi koji su postavljeni. U svakoj od faza razvoja IS, potrebno je definisati skup kriterijuma kojima se utvrđuje ili procjenjuje stepen usaglašenosti zadovoljenja zahtjeva prije, u toku i nakon završetka faze projektovanja informacionog sistema. U literaturi iz oblasti informacionih sistema su definisani mnogi kriterijumi koji definišu dobar informacioni sistem. U radu su prezentovana ranija istraživanja iz ove oblasti, kao i kategorizacija kriterijuma za mjerenje uspješnosti informacionog sistema.*

Abstract: *Information systems developers needs to know what the goals of information system are, and a buyer of the information system needs to know whether the goals have been fulfilled. In each of the phases of the information system life cycle, it is necessary to define a set of criteria that determine or assess the degree of compliance to satisfy the user requirements before, during and after each phase in information system life cycle. Literature defines many criteria for what characterizes a good information system. The purpose of this paper is to provide a framework for criteria for the measuring and comparing of information systems.*

1. UVOD

Razvojni timovi informacionih sistema (IS) moraju znati šta su ciljevi razvoja IS, dok naručilac IS mora znati u kojoj su mjeri ispunjeni zahtjevi koji su postavljeni. U svakoj od faza razvoja IS, potrebno je definisati skup kriterijuma kojima se utvrđuje ili procjenjuje stepen usaglašenosti zadovoljenja zahtjeva prije, u toku i nakon završetka faze projektovanja IS: da li sistem radi ono što je trebalo da radi, kao i da se upoređi sa drugim IS. Uspješnost informacionih sistema (IS) ne može se mjeriti direktno, nego može biti procijenjena koristeći mnoge mjere koje su relevantne za uspjeh. Još od sedamdesetih godina XX vijeka mnogi autori su razvijali prilaz procjeni uspješnosti IS-a. Predlagali su mnoštvo promjenljivih, pokazatelja i mjera, kao što su: zadovoljenje korisnika ili prihvatljivost sistema, angažovanost korisnika, učešće korisnika, umješnost korisnika, (opipljivi) kvalitet informacije ili kvalitet sistema, opipljivi kvalitet usluge: zadovoljenje korisnika sa funkcijom informacionih usluga (SERQUAL). Pored ovoga, potrebno je izvojiti i korisnost IS-a, potreba IS-a za potporu specifičnih zadataka, kao i spregu zadatak-tehnologija. Uspješnost specijalizovanih IS-a se ogleda u uticaju na pojedinačne, grupne ili organizacione performanse kao što su sistemi za podršku pri odlučivanju, sistemi za podršku pri grupnom odlučivanju i grupnoj komunikaciji, kancelarijski sistemi, sistemi za podršku kreativnosti, komunikacija putem računara ili korisnička procjena [1], [2].

Postoji više modela kojima se vrše mjerenja uspješnosti IS-a i većina od njih se bavi ekonomskim pokazateljima korištenja IS-a, mjerenjima upotrebljivosti (većinom interfejsa IS-a) i mjerenjima korisničkog zadovoljstva. U radu su prikazane osnovne kategorije kriterijuma za mjerenje uspješnosti IS, kao i dva modela mjerenja uspješnosti.

2. KATEGORIZACIJA KRITERIJUMA USPJEŠNOSTI IS

U literaturi iz oblasti IS su definisani mnogi kriterijumi koji definišu dobar IS. Na osnovu OITP (organization-information-technology-policy) modela [3], može se izvršiti kategorizacija kriterijuma za mjerenje i procjenu uspješnosti IS [4] kao što je prikazano na Sl. 1. Po ovoj kategorizaciji kriterijumi su podijeljeni u sljedeće kategorije:

- Organizacija: kriterijumi iz ove kategorije su oni kriterijumi koji su interesantni sa stanovišta upravljanja organizacijom. Oni opisuju koliko dobro IS podržava organizacione performanse. U ove kriterijume ubrajamo grupe kriterijuma:
 - performanse (transmisija - koliko često se vrši distribucija informacija u organizaciji, greške - koliko često se dešava gubitak informacija, overflow informacija – koliko često se informacije gomilaju na jednom mjestu) [5], [6],
 - kontrola (menadžment znanja – koliko dobro su informacije dostupne u organizaciji i ne zavise od pojedinaca, pregled – kako su istaknuti informacioni kanali, fleksibilnost – koliko lako je moguće izvršiti modifikaciju sistema za nove potrebe organizacije, upravljanje – kako dobro IS podržava upravljačke strukture organizacije, brzina odlučivanja – koliko dobro IS podržava donošenje odluka što je povezano i sa činjenicom i kriterijumom definisanja koliko često su donešene pogrešne odluke) [7]-[9],
 - ekonomija (ROI (return of investment) - koliko je vremena potrebno da se povrate ulaganja u IS, kompetitivnost – da li IS čini organizaciju u kojoj je implementiran kompetentnijom i konkurentnijom u poslovnom okruženju, zadovoljstvo komintenata – koliko su uvođenjem IS-a zadovoljni komintenti

organizacije, produktivnost – u kojoj mjeri IS povećava produktivnost organizacije) [7]-[9].

- **Pojedinac:** kriterijumi iz ove kategorije su oni koji se tiču pojedinaca na čiji rad utiče IS:
 - **emancipacija:** (satisfakcija – koliko je pojedinac zadovoljan IS-om, demokratičnost – koliko glas pojedinca može uticati na organizaciju i IS, uticaj – koliko je uticaj pojedinca da preko IS-a prikaže svoju situaciju, učenje – u kojoj mjeri je pojedincu omogućeno učenje primjenom IS-a) [9].
 - **ergonomija:** (stres – koliko IS doprinosi povećanju stresa kod pojedinca, preopterećenje – koliko IS doprinosi prevelikom obimu informacija pojedincu a ovaj kriterijum je povezan i sa manjkom informacija pojedincu za njegov rad, kontrola – koliko pojedinac može konfigurirati IS za svoje potrebe) [6]
 - **komunikacija:** (informisanost – koliko je pojedinac informisan o dijelovima organizacije IS koji nisu po njegovom direktornom ili indirektnom kontrolom, društvena interakcija – koliko pojedinac preko IS može ostvariti društvenu interakciju sa ostalim korisnicima IS, pripadnost – koliko IS podržava osjećaj pripadnosti organizaciji) [9]
- **Informacija:** kategorija kriterijuma koja se odnosi na kvalitet i pristup informacijama:
 - **pristup:** (dostupnost – koliko su informacije dostupne korisniku, mogućnost pretrage – kolika je mogućnost pretrage informacija koje su sadržane u IS-u, format – da li su informacije koje su sadržane u IS odgovarajućeg formata koji omogućava lakše donošenje odluka) [10]
 - **kvalitet:** (tačnost – u kojoj mjeri su informacije u IS tačne, precizne, relevantnost – stepen relevantnosti informacija u IS-u, pouzdanost – u kojoj mjeri u informacija pouzdane, značaj – stepen značaja dostupnih informacija u IS-u, jedinstvenost – koliko se u IS-u ponavljaju informacije (redundansa podataka), bez predrasuda – u kojoj mjeri su informacije koje su dostupne u IS nalaze u obliku koji je izvorni i nad

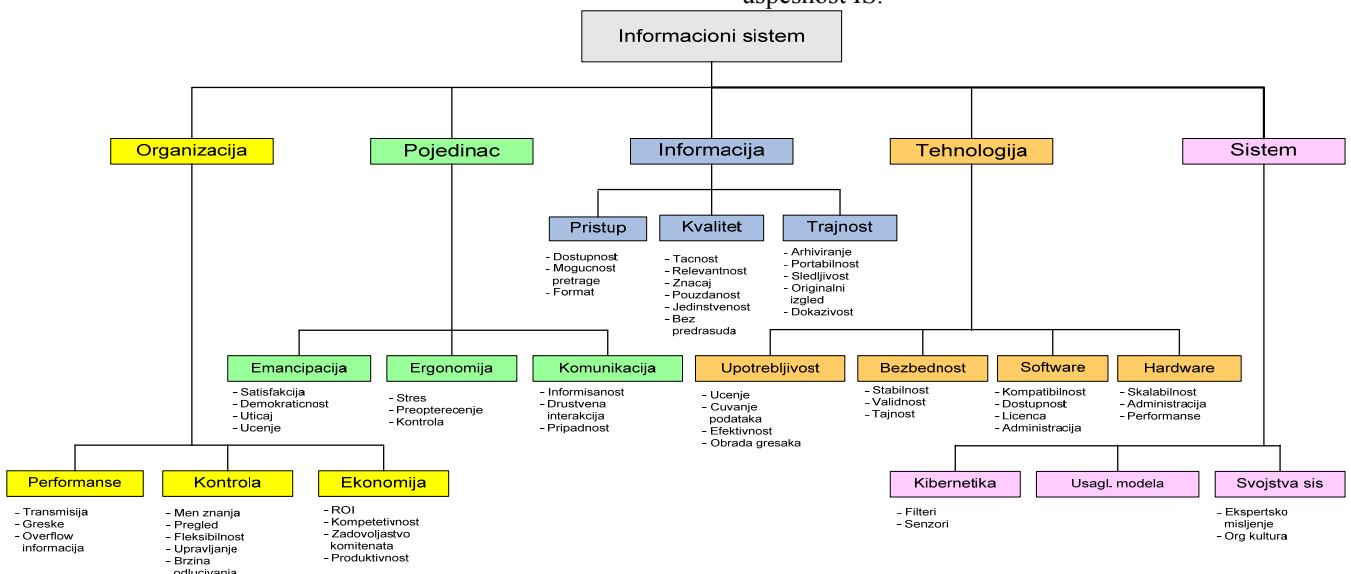
kojima nije bilo nečijeg uticaja (interpretacije korisnika na koji utiče njegov lični stav) [9]

- **trajnost:** (arhiviranje – koliko je potrebno informacijama da se smatraju arhivskim što je povezano i sa kriterijumom arhiviranja informacija, portabilnost – koliko je jednostavno informacija iz jednog IS-a prosljediti u drugi IS, sledljivost – koliko dobro arhivirane informacije se mogu pretraživati na osnovu svog sadržaja i porijekla, originalni izgled – koliko su arhivirane informacije vjerne svom originalnom izgledu, dokazivost – koliko dobro arhivirane informacije pokazuju stvari slijed stanja koje opisuju) [10]
- **Tehnologija:** svi oni kriterijumi koje se odnose na materijalne artefakte koji učestvuju u distribuciji i upravljanju informacijama, ponekad se zovu i uopštenim imenom informaciona tehnologija. Ovi kriterijumi se odnose na korišteni software, hardware, njihovu upotrebljivost i bezbjednost.
- **Sistem:** ovi kriterijumi se odnose na sistemsku tačku gledišta IS-a. Ovim kriterijumima se proučava kako dobro je IS usaglašen sa pojmom dobar sistem uopšte.

3. MODELI MJERENJA USPJEŠNOSTI

Na osnovu komunikacione teorije koju su razvili Shannon i Weaver [11], njegov prilagođavanja kontekstu IS-a [12], te analize empirijskih istraživanja iz područja upravljanja IS, DeLone i McLean [9] su predložili taksonomiju i model uspjeha IS. Taksonomija je služila kao okvir za kategorizaciju mjera identifikovanih pregledom literature u šest međusobno povezanih cjelina:

- **Kvalitet sistema:** mjere kvaliteta sistema opisuju sam sistem i obuhvataju kriterijume poput pouzdanosti, raspoloživog vremena, iskorišćenosti resursa ili fleksibilnost sistema. Ovaj kriterijum više odražava inženjerski podrazumijevane performanse razvoja sistema.
- **Zadovoljenje korisnika:** zadovoljenje korisnika, zajedno sa korišćenjem sistema je najšire primjenjeno mjerilo za uspešnost IS.

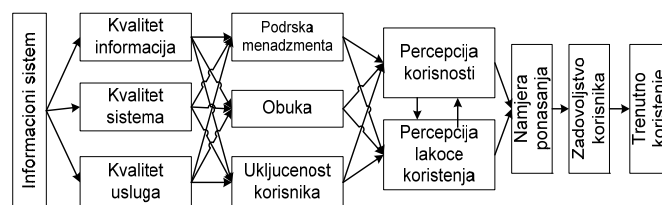


Slika 2. Kategorizacija kriterijuma za ocjenu uspešnosti IS-a (adaptirano prema [4])

- **Kvalitet informacije:** ova kategorija mjeri izlaz informacionog sistema, u obliku izvještaja ili rezultata istraživanja. Kriterijumi su značajnost, tačnost, pouzdanost, savršenost, informisanost. Većina kriterijuma koji se koriste u ovoj komponenti su procijenjeni da budu razumljivi od strane korisnika, prije svega opisni. Prema tome ova komponenta ima bliske veze sa zadovoljenjem korisnika i veliki broj kriterijuma je razvijen kao dio alata za mjerenje zadovoljenja korisnika.
- **Korišćenje:** promjenljive opisuju uspješnost korišćenje IS. Praksa uključuje objektivne kriterijume: vrijeme prijavljivanja, broj funkcija IS. To su kvantitativne mjere korišćenja. Mora da postoji povratna sprema između zadataka i tehnoloških karakteristika.
- **Individualni uticaj:** ovaj elemenat mjeri pojedinačne uticaje, ali u lancu događaja obuhvata događaje koje su prethodile dati uticaj. Riječ je o opisnoj mjeri, mjeri se opseg korišćenja IS-a koji ujedno utiče, eventualno mijenja ponašanje pojedinca. Većina mjera je razvijena i primjenjena u kontrolisanim, laboratorijskim uslovima. Mjeri se: produktivnost donošenja odluka, prosječno vreme donošenja odluka, broj razmatranih alternativa, pouzdanost odluka, uvećanje produktivnosti korisnika ili spremnost da se plate određeni izlazi IS-a.
- **Organizacioni uticaj:** ova komponenta ocjenjuje uticaj korišćenja IS na organizacionom nivou (strategijska poslovna jedinica, fabrička ili cjelokupna organizacija). Izazov za mjerenje na organizacionom nivou jeste izolovanje efekata IS od ostalih efekata koji su od uticaja na organizacione performanse

Model je sačinjen na osnovu nekoliko pretpostavki. Prva i druga pretpostavka definišu da kvalitet sistema i kvalitet informacija su prediktori korištenja i zadovoljstva korisnika. Treća pretpostavka implicira da korištenje i zadovoljstvo korisnika mogu međusobno pozitivno i/ili negativno uticati jedan na drugog. Četvrta pretpostavka je da korištenje i zadovoljstvo korisnika direktno utiču na individualni učinak. Posljednja pretpostavka implicira da je individualni učinak prediktor organizacionog učinka. Model je često bio predmet kritika, a najveća zamjerka se odnosila na činjenicu da dimenzija kvaliteta usluge nije bila uvrštena u model koji je vrlo važna činjenica u transakcionim IS. Ostale kritike su se odnosile na račun toga što dimenzijom korištenja sistema nije precizno definisano na koju se vrstu korištenja odnosi, što nije definisano da na uspjeh IS osim individualnog i organizacionog učinka utiču još i učinci radnih grupa, učinci korisnika, industrijski i društveni učinci. Kao odgovor na kritike, DeLone i McLean [13] su predstavili nadograđen model. Nadograđeni model u odnosu na početnu verziju je dopunjen i proširen sa kvalitetom usluge koja je uz kvalitet sistema i informacija prediktor korištenja i zadovoljstva korisnika. U njemu su detaljnije objašnjeni odnosi korištenje i zadovoljstvo korisnika, umjesto da se svaka vrsta učinka vrednuje posebno predloženo je njihovo objedinjavanje pod pojmom neto korist (net-benefit). Model je obogaćen novim povratnim vezama preko kojih neto korist utiče na korištenje i zadovoljstvo korisnika. Na osnovu Technology Acceptance Model (TAM) [14] i proširenog DeLone i McLean modela, Zaid [15] je definisao model kojim je predloženo deset

dimenzija koje treba mjeriti za analizu uspješnosti IS kao što je prikazano na Sl. 2. Nove dimenzije su: uključenost korisnika koja se definiše kao stvar od značaja i lične važnosti koje pojedinci daju sistemu, percepcija korisnosti koja se definiše kao individualna percepcija koristi IS za poboljšavanje performansi organizacije, percepcija lakoće korištenja koja se definiše kao stepen individualnih vjerovanja da će učenje o načinu rada IS biti lako za korisnika, da stepen obuke korisnika pomaže uspješnosti IS-a, podrška menadžmenta u vidu odobravanja razvoja i implementacije IS-a i njegovog eksploatiranja pomaže uspješnosti IS-a. Mjerenje namjere ponašanja obuhvata mjerenje izbjegavanja neizvjesnosti, vremena odgovaranja na zahtjeve IS-a, broja izvršenih transakcija.

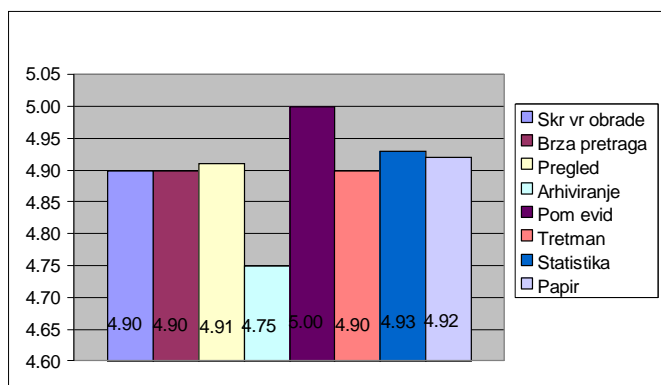


Slika 2. Integrirani model uspješnosti [15]

4. PRIMJER MJERENJA USPJEŠNOSTI

Sa početkom reforme kompletnog zatvorskog sistema Republike Srpske, a u okviru reforme zatvorskog sistema Bosne i Hercegovine, pokrenuti su određeni projekti u saradnji sa sistemima za izvršenje krivičnih sankcija u širem okruženju (prevashodno Mađarska i Škotska). U skladu sa tim, odlučeno je da se razvije poseban IS na području cijele Republike Srpske i da se promijeni sistem razmjene informacija. Nakon implementacije IS-a u sistemu za izvršenje krivičnih sankcija Republike Srpske, pristupilo se analiziranju uspješnosti IS-a sa stanovišta korisnika IS gdje su prikupljeni kvantitativni podaci. U istraživanju su korištena dva mjerna instrumenta za prikupljanje podataka za svakog korisnika u zavisnosti od mjesta zaposlenja: anketni upitnik za prikupljanje opštih podataka, kao i upitnik o rezultatima implementacije IS-a za zaposlene u sistemu ili upitnik o rezultatima implementacije rešenja za ostale državne organe. Anketni upitnik za prikupljanje opštih podataka obuhvata nekoliko kategorija o obilježjima ispitanika: pol, starost, radno mjesto, dužina staža i stepen formalnog obrazovanja. Za svako od ponuđenih pitanja ispitanici su odgovarali zaokruživanjem jedne od ponuđenih opcija. Upitnik o rezultatima implementacije IS i novog načina rada je napravljen tako da je prilagođen ispitanicima koji su radili u sistemu za izvršenje krivičnih sankcija i prije ali i poslije implementacije IS. Za zaposlene u sistemu za izvršenje krivičnih sankcija prikupljeni su odgovori korisnika koji se tiču njihovog mišljenja o: skraćenju obrade podataka za novo lice koje dolazi u ustanovu, o mogućnosti bržeg pronalaska podataka o licima koji se nalaze u ustanovi, mogućnosti IS-a da se prikažu svi podaci koji se nalaze o određenom licu u ustanovi, mogućnosti IS-a za arhiviranje podataka o licima, svrsishodnosti i lakšem vođenju pomoćnih evidencija u IS, olakšavanje zaposlenim o određivanju tretmana lica na osnovu informacija u IS, olakšavanju statističke obrade podataka i smanjenju papirne

dokumentacije. Za lica koja su ispitivana van ustanova pažnja je posvećena njihovom mišljenju o uniformnosti izvještaja iz različitih ustanova, kao i mogućnostima pretrage podataka koji su u njihovom opsegu rada i preglednosti dokumentacije vezane za lica u ustanovama. Na svaki od odgovora ispitanici su se izjašnjavali zaokruživanjem odgovora u skali od 1 (potpuno se ne slažem) do 5 (potpuno se slažem). Uzorak na kome je izvršeno ispitivanje obuhvata ukupno 73 osobe zaposlene u ustanovama-zatvorima, 8 osoba zaposlenih u sistemu za izvršenje krivičnih sankcija (Ministarstvu pravde RS) što ukupno čini 81 osobu iz sistema za izvršenje krivičnih sankcija u RS, kao i 18 osoba zaposlenih u ostalim državnim organima (pravosudni i policijski organi). Na ovaj način je obezbjeđeno prikupljanje validnih podataka od korisnika koji su u poziciji da daju kvalitetne odgovore na pitanja iz upitnika. Anketiranje je izvršeno bez prethodne najave učesnicima. Pregled ukupnih rezultata dat je na Sl. 3.



Slika 3. Ukupni rezultati mjerenja

5. ZAKLJUČAK I DALJE SMJERNICE

Kriterijumi koji su predstavljeni u ovom radu su predstavljeni pojedinačno, bez njihove međusobne povezanosti koje se javljaju u realnim primjerima implemenetacije IS-a. Pored ovoga, data je kategorizacija koja nije uslovljena faktorima uticaja pojedinih kriterijuma na ukupnu ocjenu IS-a a što je uslovljeno tipom IS koji se razvija i implementira. U svakom slučaju, potrebno je da se svaki kriterijum bude: mjerljiv, eksplicitno definisan, da se može kvantifikovati i izvršiti validacija rezultata. Predstavljani kriterijumi i modeli uspješnosti predstavljaju jednostavnu teorijsku osnovu za empirijska istraživanja oblasti mjerenja uspješnosti IS-a. U prikazanom primjeru izvršene analize uspješnosti IS korišteni su kriterijumi koji se odnose na mišljenje korisnika razvijenim IS-om nakon njegovog kratkotrajnog korištenja. Analiza je prikazala da su korisnici zadovoljni razvijenim rješenjem. Ipak, iako je anketiranje učesnika vršeno anonimno, pitanje koje se postavlja u ovom i svim sličnim analizama, nakon dobijanja rezultata je njihova vjerodostojnost iskazanih odgovora sa stanovišta moguće naknadne identifikacije učesnika analize (na primjer, povezivanje godina staža, vrste posla i pojedinih učesnika) što bi trebalo biti izbjegnuto. Finansijska sredstava potrebna za detaljnu analizu uspješnosti IS-a predstavljaju ograničavajući faktor, te i sa te strane treba biti veoma

oprezan prilikom definisanja kriterijuma koji će biti ispitivani.

LITERATURA

- [1] O. Sedlak, T. Kiš, M. Čileg, A. Marcikić, "Modeli merenja uspešnosti sistema upravljanja znanjem", Naučni skup Novi metodi menadžmenta i marketinga u podizanju konkurentnosti srpske privrede, Ekonomski fakultet Subotica, oktobar 2011.
- [2] Z. Ćirić, I. Ćirić, N. Mirkov, M. Peranović, "Merenje uspešnosti sistema upravljanjem znanjem proširenim modelom DeLone/McLean", Zbornik radova sa međunarodnog naučnog skupa Infoteh-Jahorina, Vol. 12, pp. 583-586, March 2013.
- [3] S. Holmberg, "Taking Syntegrity-4 from assumption mode to reflection mode", Systems research and behavioral science. No 18, 2001.
- [4] J. Palmius, "Criteria for measuring and comparing information systems", Proceedings of the 30th Information Systems Research Seminar in Scandinavia IRIS 2007.
- [5] C. Shannon "A mathematical model of communication", Bell System Technical Journal, vol. 27, pp. 379-423, 623-656, 1948.
- [6] J. Palmius, "A theoretical base for the High-level simulation of organisational information systems", Systems Research & Behavioral Science, No 22, 2005.
- [7] Z. Irani, "Information systems evaluation: navigating through the problem domain", Information & Management. vol 40. Elsevier, 2002.
- [8] G. Torkzadeh G, W. Doll, "The development of a tool for measuring the perceived impact of information technology on work", Omega - the international journal of management science, No 27, Pergamon, 1999.
- [9] W. DeLone, E. McLean, "Information Systems Success: The Quest for the Dependent Variable", Information Systems Research. Vol 3, no 1., 1992.
- [10] E. Borglund, "A Predictive Model for Attaining Quality in Recordkeeping", Mid Sweden, University Licentiate Thesis 12. Härnösand, Sweden, 2006.
- [11] C.E. Shannon, W. Weaver, "The Matematical Theory of Communication", University of Illinois Press, Urbana, 1949.
- [12] R.O. Mason, "Measuring information output: A communication systems approach", Information & Management, Vol. 1, No. 5, pp. 219-234, 1978.
- [13] W. DeLone, E. McLean, "The DeLone and McLean Model of Information System Success: A Ten-Year Update", Jorunal of Management Information Systems, Vol. 19, No 4, pp. 9-30, 2003.
- [14] F. Davis, "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology", MIS Quarterly 13(3), pp. 319 – 339, 1989.
- [15] A.H.N. Zaid, "An Integrated Success Model for Evaluating Information System in Public Sectors", Journal of Emerging Trends in Computer and Information Science, Vol. 3, No 6, 2012.

O AMOS SOFTVERU NAMIJENJENOM ELEKTRONSKOM UPRAVLJANJU RESURSIMA NA BRODU ABOUT AMOS SOFTWARE FOR ELECTRONIC MANAGEMENT OF THE ASSETS AT SHIP

Sanja Bauk, Tatijana Dlačić, Radoje Džankić, *Fakultet za pomorstvo, Univerzitet Crne Gore*

Sadržaj: *U radu je okvirno prikazana modularna struktura AMOS softvera, uključujući neke ključne funkcije ovih modula, namijenjenih elektronskom upravljanju resursima na brodu. Takođe je dat kratak opis nekoliko ekstenzija AMOS softvera namijenjenih njegovom prilagođavanju uslovima u korporativnom okruženju. Ovo je urađeno s namjerom da se dijelom nadomjesti to što prezentiranje ovog sofisticiranog, kompleksnog i dinamičnog softvera nije bilo uopšte, ili barem u odgovarajućoj mjeri, zastupljeno u nastavnim planovima predmeta iz održavanja i informacionih sistema koji se izučavaju na Fakultetu za pomorstvo u Kotoru (UCG).*

Abstract: *This paper presents a framework of AMOS software modular structure, including some key functions of these modules, designed for electronic ship asset management. It gives a brief description of several AMOS software extensions designed to adapt it to the enterprise (corporate) environment. This has been done with the intent to partially compensate the fact that the presentation of this sophisticated, complex and dynamic software was not at all or, in the appropriate extent, presented in the curricula of maintenance and information systems courses which are taught at the Faculty of Maritime Studies in Kotor (UoM).*

1. UVOD

AMOS je, u širem smislu, sinonim za elektronsko upravljanje resursima. Izvorno, ovaj akronim je bio izveden od pojmova: „Administration, Management, Operations and Spares” (eng), što bi se u doslovnom prijevodu odnosilo na administraciju, upravljanje, operiranje i rezervne dijelove. To je bila neka vrsta zajedničkog imenitelja za njegove primarne funkcije. S obzirom na razvoj sistema koji je vremenom uslijedio, naziv je revidiran i sada je: „Asset Management Operating System“ (eng), što se može prevesti kao - operativni sistem za upravljanje resursima. Ovaj složeni softver omogućuje kontrolu i upravljanje tehničkih i administrativnih aspekata u domenu: održavanja, rezervnih dijelova i zaliha, nabavki, kvaliteta i sigurnosti, upravljanja putovanjem (u brodarstvu), kao i u domenu administrativnih poslova vezanih za zaposlene [1].

Prije svake analize ovog složenog i veoma sofisticiranog softvera, treba reći da je za njegov nastanak i razvoj zaslužna SpecTec Group kompanija. Riječ je o multinacionalnoj kompaniji, sa predstavništvima u preko dvadeset zemalja širom svijeta, specijalizovanoj za razvijanje i distribuciju AMOS-a. Ova kompanija radi za potrebe brodarstva, industrije nafte i gasa, odbrane, proizvodnje i distribucije energije i dr. U ovom radu, akcenat je stavljen na korišćenje AMOS-a u brodarstvu.

Pored razvijanja softvera, SpecTec Group takođe pruža usluge tipa izrade baza podataka, obuke korisnika, konsultantske usluge i dr. Nesumnjivo je IT lider u brodarstvu kada je u pitanju elektronsko administriranje na nivou broda i kompanije. SpecTec trenutno opslužuje oko 630 brodarskih kompanija (preko 7000 brodova), od kojih je

skoro polovina na Loyd-ovoj listi [1]. U prvoj fazi razvoja, AMOS je bio relativno jednostavan program sačinjen od svega nekoliko hiljada redova i radio je pod tekst procesorom u MS-DOS okruženju. Sada ovaj dinamički softver ima milione redova koda i koristi ga nekoliko najvećih brodara i *offshore* kompanija, te je tako postao standard u pomorstvu. SpecTec već godinama uspješno saraduje sa svim ključnim subjektima u pomorstvu: brodovlasnicima, brodarima, menadžerima, regulatornim tijelima i klasifikacionim društvima. Sistem uspješno odgovara na sve veće zahtjeve u domenu legislativne i olakšava sprovođenje propisa IMO-a (International Maritime Organization, eng), uključujući MARPOL (International Convention for the Prevention of Pollution from Ships, eng), ISM (International Safety Management, eng), STCW (Standards of Training, Certification and Watchkeeping, eng), ISPS (The International Ship and Port Facility Security Code, eng) i dr. Dodatno, AMOS utiče na poboljšanje operativne efikasnosti i profitabilnosti broda i kompanije.

Bez obzira na aktuelnu ekonomsku krizu u brodarstvu, nijedna od kompanija koje koriste AMOS nije potpala pod stečaj. Razlog je dijelom to što AMOS omogućuje strogu kontrolu svih troškova. Kompanije opremljene AMOS-om već niz godina imaju prednosti, u smislu da posjeduju mogućnost uvida u sve svoje troškove, te ih tako mogu lakše kontrolisati. Postoje dokazi da kompanije koje koriste AMOS uštede i do milion dolara godišnje na istim rutama, u poređenju sa brodovima (kompanijama) koji(e) to ne čine. Kompanije koje ulažu u ovaj softver i njegovu efikasnu primjenu, imaju veće izgleda da prevaziđu aktuelni krizni period u brodarstvu. Radi se o promjeni metodologije, redu, poštovanju procedura i kontroli svih aktivnosti.

2. PODRUČJA PRIMJENE AMOS-a

Prva primjena AMOS-a je bila u pomorstvu, prije više od 25 godina. Međutim, danas se ovaj složeni softverski sistem, može se slobodno reći – standard za ekeltronsko upravljanje resursima, primjenjuje i u nekoliko drugih oblasti. Neke od njih su navedene u nastavku.

(a) Platforme za naftu i gas: Upravljanje i održavanje resursa na platformama za ispitivanje nafte i gasa je komplikovan zadatak, pogotovu kada se radi o resursima raspoređenim na više lokacija širom svijeta. SpecTec, na bazi najnovijih IT rješenja, to obezbjeđuje veoma uspješno. Podaci se razmjenjuju između centralne baze podataka u centrali kompanije i udaljenih baza na platformama. Razmjena se odvija putem telefonskih linija i/ili satelitskih veza, uključujući i Internet putem satelita (ISAT – Internet over satellite, eng). Podaci na taj način mogu biti u potpunosti i redovito ažurirani po pitanjima: održavanja, rezervnih dijelova, troškova i dr. Dakle, AMOS omogućuje potpuno praćenje eksploatacionog ciklusa resursa od nabavke, održavanja, logistike rezervnih dijelova i ostalog.

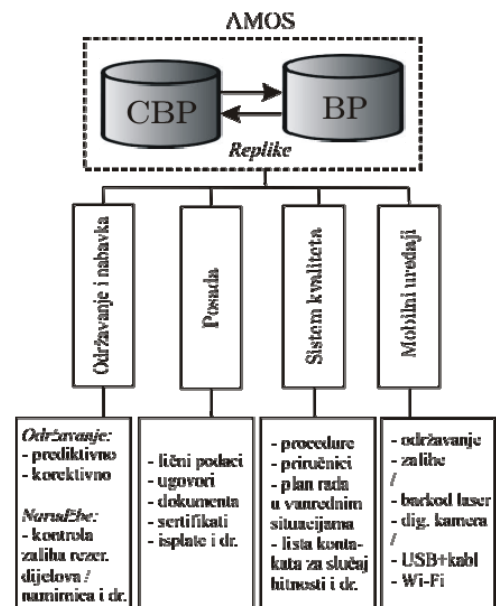
(b) Energetika i industrija: AMOS je široko rasprostranjen u raznim energetske sistemima, gdje se uglavnom koristi za kontrolu održavanja i nabavke rezervnih dijelova za veliki broj elektrana širom svijeta. AMOS sistem održavanja i nabavke raspolaze funkcijama projektnog planiranja i upravljanja dokumentacijom, a takođe je integrisan sa ilustrovanim instrukcionim knjigama proizvođača i katalogima rezervnih dijelova. Sistem koristi AMOS replike i automatski kontroliše sve transfere podataka između udaljenih (lokalnih i regionalnih) baza i centralne baze podataka. Zahvaljujući ažuriranju u realnom vremenu i centralizovanom planiranju održavanja, kao i efektivnom upošljavanju međunarodnih servisnih timova, postiže se veća pouzdanost u radu elektrana, kao i bolja kontrola logističkih procesa i obezbjeđivanje rezervnih dijelova u bilo kom trenutku. Tako je AMOS kroz svoju funkcionalnost u domenima održavanja i nabavke, kao i u procesu upravljanja radnim procesima, potvrdio potencijal i u drugim, brojnim područjima proizvodnje i transporta. Korisnički interfejs je često prilagođen određenim postrojenjima/uređajima, omogućujući tako operaterima da putem PC-a ili PDA (Personnal Digital Assistant, eng) odmah intervenišu u slučaju potrebe kod službe za održavanje, te tako zajednički riješe problem. Ovdje treba naglasiti da AMOS može biti integrisan sa drugim sistemima i tako obezbijediti i održavanje prema stanju. Dodatno, sistem može da bude integrisan sa sistemima za finansijske, odnosno, *e-commerce* transakcije.

(c) Odbrana: SpecTec obezbjeđuje planiranje i upravljanje održavanjem, kao i kontrolu zaliha i na zahtjev mornarice, ili na zahtjev bilo koje slične odbrambene organizacije. AMOS radi samostalno da brodu (održavanje, kompletna logistika oko nabavke rezervnih dijelova, zaliha i sl.), dok se svi podaci repliciraju u centralnoj mornaričkoj bazi (Central Naval Base, eng), što ga čini veoma djelotvornim po pitanjima mornaričkih logističkih zahtjeva. Podaci se ažuriraju na brodu, a potom se kriptovani šalju centralnoj stanici, putem posebno zaštićenih kanala. Kada je u pitanju

upravljanje održavanjem i zalihama, sve se odvija u skladu s NATO standardima. Ovdje treba napomenuti da AMOS omogućuje monitoring vibracija i analize rotirajuće opreme, zašto razvija softver i hardver, kao i baze podataka, uključujući tabele sa karakterističnim parametrima vibracija/rotacija i razne izvještaje.

3. AMOS APLIKACIJE NA BRODU

AMOS se na brodu koristi za održavanje (preventivno i korektivno) brodskih tehničkih (pod)sistema, uključujući sve njihove sastavne dijelove; zatim, za nabavke: rezervnih dijelova, opreme, namirnica i dr; za sprovođenje mjera sistema kvaliteta, uključujući pohranjivanje svih priručnika za praćenje i sprovođenje procedura kvaliteta, kao i najnovije konvencije u pomorstvu; za unos podataka automatski u bazu sa bilo kog mjesta na brodu (skladišta; stroja i sl.) putem mobilnog ručnog uređaja sa ugrađenim laserskim barkod čitačem ili digitalnom kamerom. Na slici 1 su prikazane, putem blok dijagrama, ključne aplikacije AMOS-a na brodu.



Slika 1. Moduli i funkcije AMOS softvera na brodu

3.1. Održavanje i nabavka (M&P – Maintenance and Purchasing, eng)

Ovdje se radi o Windows aplikaciji za integralno upravljanje održavanjem na brodu, nabavkama (kontrola zaliha i narudžbe), kao i troškovima. SpecTec je razvila ovu aplikaciju za potrebe kompanija/organizacija sa dislociranim instalacijama (npr. sjedište broderske kompanije - brodovi). Aktivnosti vezane za održavanje, praćenje zaliha i troškova izvršavaju se na udaljenim lokacijama, ali se prate iz centralnog sjedišta kompanije/organizacije. Sve izmjene kada je u pitanju stanje na zalihama broda, takođe se automatski registruju u sjedištu kompanije. Veoma je važno da AMOS omogućuje korisnicima da vide koliko su novca potrošili na održavanje i rezervne dijelove, nabavku zaliha i potrošnih dobara, te kakvi su izgledi i planovi za buduću potrošnju.

Ono što se može uraditi putem AMOS M&P softverske aplikacije, opisano je detaljnije u nastavku [2].

Planiranje održavanja:

- Specifikacija poslova koji se moraju redovno izvršavati;
- Utvrđivanje rasporeda održavanja;
- Štampanje radnih lista za poslove koji se moraju izvršiti u skorije vrijeme, sa ček-listama ili detaljnim opisom posla koji treba uraditi;
- Planiranje vanrednih pregleda uz obezbjeđivanje radnih naloga (Work Orders, eng) i sl.

Kreiranje izvještaja o održavanju:

- Kreiranje izvještaja o realizovanom održavanju;
- Memorisanje izvještaja za potrebe inspeksijskog nadzora;
- Štampanje izvještaja;
- Kreiranje izvještaja o realizaciji neplaniranih (vanrednih) popravki i rutinskim provjerama i sl.

Kontrola zaliha:

- Pregled aktuelnog stanja na zalihama u svim skladištima;
- Direktno ažuriranje stanja nakon pregleda;
- Automatski prikaz svih transakcija iz zaliha pri održavanju i u zalihe pri nabavkama;
- Prikaz tekućeg stanja na zalihama u odnosu na maksimalno, minimalno, ili stanje pri prethodnom naručivanju;
- Automatsko izračunavanje količine zaliha, neophone da se one popune u skladu sa nekim od prethodno navedenih nivoa;
- Memorisanje podataka o dobavljačima, cijenama i dr.

Kreiranje trebovanja:

- Kreiranje trebovanja za zalihe manuelno ili automatski na osnovu željenih količina i preferentnih dobavljača;
- Kreiranje trebovanja za potrošna dobra po istom principu kao u slučaju zaliha.

Praćenje narudžbina (automatsko generisanje odgovora na pitanja, od kojih su neka nevedena u nastavku):

- Jeli dobavljač primio narudžbu?
- Da li je narudžba procesuirana i potvrđena?
- Kada se očekuje isporuka? i sl.

Evidencija prijema:

- Markiranje isporuka prema narudžbama i automatsko ažuriranje stanja na zalihama.

Praćenje troškova (putem automatskog generisanja odgovora na pitanja koja slijede):

- Kolika je potrošnja (na brodu) za tekući mjesec?
- Kolika je potrošnja za tekuću fiskalnu godinu?
- Kolika je preostalo od planiranog budžeta za potrošnju u tekućoj godini? i sl.

Poželjno je takođe i automatsko praćenje troškova za neplanirane poslove, s ciljem poboljšanja programa preventivnog održavanja u perspektivi.

3.2. Posada (Personnel, eng)

Ovo je AMOS alat za upravljanje podacima o posadi. Osnovni podaci o posadi se unose u posebnu formu u sjedištu kompanije, a mogu da se ažuriraju i na brodu, u skladu sa stepenom ovlašćenja administratora. Sve izmjene i dopune podataka se realizuju u skladu sa AMOS sistemom repliciranja.

Forma za upravljanje podacima o posadi sadrži sljedeće stavke [3]:

- prezime i ime; ugovor; zvanje; adresa; aerodrom; fotografija; dužina staža; obrazovanje; sertifikati; zdravstveni dokument; pasoš; pomorska knjižica i sl.

Ovim osnovnim podacima mogu se pridruženi podaci o članovima porodice i sl. AMOS aplikacija vezana za posadu, uključuje i vođenje platnog spiska, kao i prateće računovodstvene operacije, koje se odnose na troškove u kantini, isplate posadi unaprijed i dr. AMOS prati i vremenske periode angažovanja svakog od članova posade. Ove podatke prate obično i grafički prikazi. Takođe, AMOS kontroliše (automatski) validnost ličnih dokumenata članova posade, kao i sertifikata koje posjeduju.

3.3. Kvalitet i bezbjednost (Q&S - Quality & Safety, eng)

Ovo je AMOS modul za integralno upravljanje sistemom kvaliteta i bezbjednošću, uključujući procedure, cirkularna pisma, informacije o eventualnim neusklađenostima sa procedurama, informacije o akcidentima i sl. Q&S daje mogućnost kompaniji da sama razvija svoj (interni) sistem kvaliteta, ali koji u principu mora biti usklađen sa zahtjevima regulatornih tijela i klasifikacionih društava. Demo baza podataka Q&S modula sadrži listu predefinisanih procedura, uključujući i mogućnost praćenja njihovog statusa (npr. aktuelna/zastarjela). Ove procedure mogu poslužiti kao polazna osnova za kreiranje sopstvenog sistema kvaliteta i načina njegovog praćenja [4]. Procedure, ustvari, opisuju mjere koje treba primijeniti na određeni posao na brodu ili brodski (pod)sistem, kako bi brod bio funkcionalan i bezbjedan, odnosno, kako bi kompanija poslovala u skladu sa predhodno postavljenim ciljevima kvaliteta.

AMOS Q&S modul sadrži informacije o akcidentima (tj. incidentima vezanim za teret, posebno opasan, povredama članova posade, oštećenjima na brodu, zagađenjima okoline i dr). AMOS Q&S omogućuje pridruživanje odgovarajućih priručnika svakoj od procedura u sistemu kvaliteta. Ovi priručnici mogu biti sa neograničenim ili ograničenim pristupom, odnosno rezervisanim samo za određene grupe korisnika na nivou kompanije. AMOS Q&S sadrži i IMO konvencije, sortirane prema hijerarhijskoj strukturi: SOLAS (Safety of Life at Sea, eng), SAR (Search and Rescue, eng), COLREG (Collision Regulations, eng) i dr. Takođe, postoji i poseban funkcionalni segment ovog modula namijenjen raznim zahtjevima (npr. za izmjenu uslova osiguranja i dr). Postoji i mogućnost kreiranja i praćenja rasporeda radnih zadataka u okviru sistema kvaliteta. Još jedan važan funkcionalni segment AMOS Q&S modula je onaj koji je vezan za sistem hitnih intervencija (ERS – Emergency Response System, eng). Ovaj segment sadrži liste kontakata relevantnih tijela za slučajeve hitnosti. Ovoj listi mogu biti dodati kontakti korisnika (naručilaca prevoza), regulatornih tijela, osiguravajućih društava i dr. ERS modul je dizajniran u skladu sa međunarodnim IMO standardima SOPEP (Ship Oil Pollution Emergency Plan, eng), OPA (Oil Pollution Act, eng) i dr.

3.4. Mobilni uređaj (Mobile handset, eng)

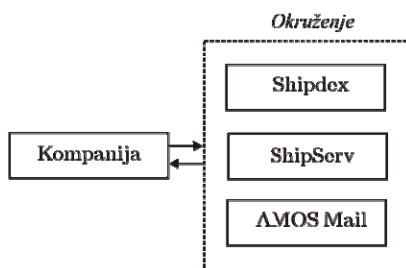
AMOS mobilni (ručni) uređaj omogućuje ažuriranje podataka u sistemu, direktno sa lica mjesta, bilo da je u

pitanju održavanje ili praćenje stanja na zalihama. Uređaj omogućuje direktno učitavanje stanja nekog brojača ili vrijednosti sa neke određene mjerne pozicije na brodu. Radi pod Windows Mobile operativnim sistemom. Verzija 2.0.00 podržava: engleski, italijanski, kineski, norveški, njemački i švedski jezik. Dakle, dvije ključne funkcije koje podržava ovaj uređaj su: održavanje i registrovanje stanja na zalihama. U ove svrhe AMOS mobilni uređaj koristi laserski barkod čitač ili digitalnu kameru. Na AMOS PC stanicu se ovaj uređaj može priključiti putem kabla i USB priključka ili putem Wi-Fi-a [5].

4. AMOS KORPORACIJSKE APLIKACIJE

Osim onih na brodu, AMOS podržava i brojne korporacijske aktivnosti. Neke od njih su samo okvirno prikazane u nastavku, dok je na slici 2 dat njihov blok dijagram.

- **ShipdexTM**: se odnosi na međunarodna pravila poslovanja (International Business Rules, eng) ili ShipdexTM protokol, razvijen je s ciljem standardizacije generisanja i razmjene tehničkih i logističkih podataka na nivou kompanije. Ovaj protokol se može jednostavno implementirati nad različitim IT platformama. Ono što je posebno važno, obezbjeđuje automatsko (paralelno) ažuriranje dislociranih AMOS baza podataka [6].
- **ShipServ**: je složena aplikacija koja omogućuje da svi zahtjevi i narudžbe kreirani u AMOS M&P modulu, budu automatski proslijedeni *TradeNet* mreži. Takođe, obezbjeđuje da se pristigle ponude automatski učitavaju, izbjegavajući tako potrebu za manuelnim unosom i omogućujući bolje i iscrpnije analize. Ovo obezbjeđuje brže i preciznije naručivanje, bržu i povoljniju isporuku, pojednostavljuje i ubrzava postupak reklamacija i sl [7].
- **AMOS Mail**: je servis koji obezbjeđuje uštede u satelitskim komunikacijama i do 80%, čak kada na raspolaganju stoje *Fleet Broadband* komunikacioni kanali velike brzine prenosa. AMOS Mail je namijenjen prije svega brod-obala komunikacijama i kompatibilan je sa GSM, Iridioum, HF-radio, Inmarsat-B, -B HSD, -C, -M, -mini M, FleetNet 32, 55 & 77 telekomunikacionim sistemima. Zahvaljujući AMOS Mail modulu, brod je danas neka vrsta "ploveće kancelarije" [8].



Slika 2. Neke AMOS korporacijske ekstenzije

5. ZAKLJUČAK

U radu je dat okvirni prikaz AMOS softvera, sa posebnim osvrtom na njegovu brodsku komponentu. Opisane su ukratko aplikacije vezane za održavanje i nabavke, upravljanje posadom, praćenje izvršavanja procedura sistema kvaliteta, osnovne telekomunikacije i sl. Načinjen je takođe i kratak osvrt na ključne ekstenzije AMOS-a u multinacionalnom, dinamičnom koroprativnom okruženju. Dalja istraživanja bi trebalo usmjeriti u pravcu detaljnijeg ispitivanja pojedinih funkcija u okvirima prethodno navedenih aplikacija. Isto tako, trebalo bi veću pažnju posvetiti npr. satelitskim komunikacijama koje su nezaobilazni segment ovog sistema, upravljanju svim zaposlenima na nivou kompanije (ne samo posadom), sistemu podrške u ispunjavanju standarda za tankere i sl. Praćenje publikacija SpecTec-a, kao i ispitivanje na bazi demo verzije softvera, su potencijalni načini da se sprovedu dalja istraživanja u ovim pravcima.

Izjava zahvalnosti: Autori žele da izraze zahvalnost gosp. Marinu Sobinu, koji je kao zastupnik SpecTech korporacije, 16. oktobra 2013. godine, održao informativno pozivno predavanje o AMOS-u i instalirao na Fakultetu za pomorstvo u Kotoru (UCG) demo verziju ovog softvera za edukativne svrhe.

LITERATURA

- [1] SpecTec, Sure Solutions – Applications, Internet izvori: <http://www.suresolutions.eu/html/about.html> (pristup: novembar, 2013)
- [2] SpecTec, AMOS Maintenance and Purchasing, Internet izvor: http://www.suresolutions.eu/AMOS_BS_M_P.pdf (pristup: novembar, 2013)
- [3] SpecTec, Sure Solutions – Personnel, Internet izvor: http://www.suresolutions.eu/AMOS_BS_Personnel.pdf (pristup: novembar, 2013)
- [4] SpecTec, Sure Solutions – QMS, Internet izvor: http://www.suresolutions.eu/AMOS2_QMS.pdf (pristup: novembar, 2013)
- [5] SpecTec, Sure Solutions – Mobile, Internet izvor: http://www.suresolutions.eu/AMOS_BS_Mobile.pdf (pristup: novembar, 2013)
- [6] SpecTec, Sure Solutions – ShipdexTM, Internet izvor: http://www.suresolutions.eu/AMOS_BS_Shipdex.pdf (pristup: novembar, 2013)
- [7] SpecTec, Sure Solutions – ShipServ, Internet izvor: http://www.suresolutions.eu/AMOS_BS_ShipServ.pdf (pristup: novembar, 2013)
- [8] SpecTec, Sure Solutions – Mail, Internet izvor: http://www.suresolutions.eu/AMOS2_Mail.pdf (pristup: novembar, 2013)

IMPLEMENTACIJA MPI ZA UBRZANJE ESTIMACIJE PARAMETARA 2DCPPS PRIMJENOM 2DCPF-A

IMPLEMENTATION OF MPI FOR SPEEDING UP THE ESTIMATION OF PARAMETERS OF 2DCPPS USING 2DCPF

Predrag Raković, Univerzitet Crne Gore, Elektrotehnički fakultet Podgorica

Sadržaj – Paralelno računanje je oblik računanja u kojem se proračuni izvode istovremeno i po principu da se veliki problemi često mogu podijeliti na veći broj manjih koji se tada rješavaju istovremeno (paralelno). Message Passing Interface (MPI) je de facto standard za implementaciju programa na više procesora. Cilj ovog rada je da se pokaže da implementacija MPI-ja može u velikoj mjeri skratiti vrijeme estimacije parametara kod dvo-dimenzionih kubičnih signala sa fazom polinomijalnog oblika (2DCPPS).

Abstract – Parallel computing is a form of computation in which the calculations are performed simultaneously and operates on the principle that large problems can often be divided into smaller, which then are computed simultaneously (in parallel). Message Passing Interface (MPI) is the de facto standard for implementing a program on multiple processors. The aim of this paper is to show that implementation of MPI can greatly shorten the time of estimation of parameters of 2D-cubic polynomial phase signals (2DCPPS).

1. UVOD

Paralelno računanje kao oblik računanja u kojem se proračuni sprovode istovremeno, i djeluju po principu da se veliki problemi često mogu podijeliti na manje koji se tada rješavaju paralelno. Paralelno izračunavanje je moguće izvršiti na različitim nivoima kao što su: nivo-bita, nivo instrukcije, nivo podataka kao i nivo samog zadatka koji je potrebno izvršiti. Upotreba paralelizama se često javlja kada problem koji je potrebno izvršiti je u dijelu high-performance computing-a, odnosno oblasti statističkih proračuna. Međutim, paralelno računanje je postala dominantna paradigma u računarskim arhitekturama i to uglavnom u obliku višezgrenih procesora.

Paralelno računanje se grubo može podijeliti na: hardver koji podržava paralelizam, odnosno multi-core i multi-procesorske računare koji imaju višestruke obrade podataka unutar jednog računara; klusterske i grid arhitekture [1] koje koriste više računara za rad nad istim zadacima; kao i specijalizovane paralelne računarske arhitekture koje se ponekad koriste zajedno s tradicionalnim procesorima za ubrzavanje izvršavanja određenih zadataka.

Paralelizovani računarski programi implementiraju drugačiju logiku pisanja nego što je ona kod programa sa sekvencijalnom logikom. Jedan od načina da se implementira ovakav način razmišljanja je upotreba MPI u pisanju samog programa.

MPI [3] iako još nije standard u velikoj mjeri se implementira u programima na više procesora. MPI već ima razvijene funkcije za programске jezike C i Fortran, i pokazao

se kao učinkovit model za sprovođenje programa u paraleli, tako da je svoju primjenu u svijetu pronašao u razvoju najzahtjevnijih aplikacija kao što su aplikacije za modeliranje promjene vremena, simulacija rada raznih vrsta oružja, dizajn aviona, itd.

Matlab, kao i matlaboliki programski alati kao što je Octave [2], postali su dominantni programski alati pronašli su svoju primjenu u numeričkim proračunima i koriste se za razvoj algoritama, simulacije, obradu podataka, testiranje, kao i ocjenjivanje modela. Mnogi od programa ovog tipa proračuna mogu imati koristi od bržeg izvršavanja na paralelnim računarima. Jedan od načina da se ostvari ovakav interes jeste implementacija MPI u takvim proračunima.

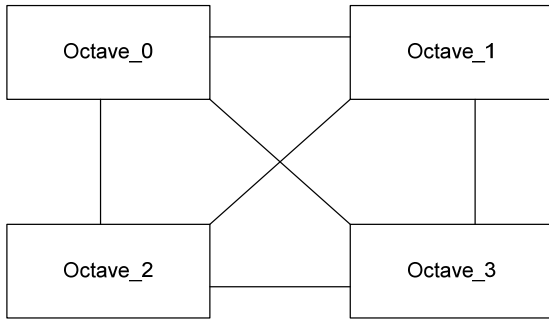
U ovom radu primjena MPI je usmjerena na estimaciju parametara dvo-dimenzionih kubično faznih signala, upotrebom 2DCP funkcije, čija je faza polinomijalnog oblika (2DCPPS). Izazov ove 2DCF tehnike jeste da je potrebno izvršiti tro-dimenziono pretraživanje u cilju estimacije parametara 2DCPPS i kroz ovaj primjer ćemo dati potvrdu upotrebljivosti MPI. Rješavanjem ovog problema primjenom MPI u Octave programskom jeziku prikazani rezultati će dati potvrdu primjene MPI kroz odnos vremena izvršavanja nad različitim brojem procesa.

Rad je izložen na sledeći način. U Sekciji 2 dati su principi MPI u Octavi. U Sekcija 3 je izložena 2DCPF(CPF) tehnika, dok su u Sekciji 4 su dati rezultati primjene MPI u Octavi za estimaciju parametara dvo-dimenzionih kubično faznih signala čija je faza polinomijalnog oblika (2DCPPS) upotrebom 2DCP funkcije. Na kraju slijedi Zaključak dobijenih rezultata.

2. UKRATKO O OCTAVE I MPI

Octave je programski alat koji je prvobitno zamišljen da bude prateći softver za dodiplomski nivo studija na predmetima dizajna hemijskih reaktora. Razvoj dizajna Octave započinje oko 1988. godine ali se ubrzo uviđaju nedostaci i započinje se sa radom na izgradnji puno fleksibilnijeg alata. Danas je Octave opsežan alat vrlo sličan Matlab alatu i slične je namjene, dok za razliku od Matlab-a GNU Octave nije potrebna komercijalna licenca.

MPI se realizuje kroz programsku biblioteku i grupu funkcija koje omogućavaju izvršavanje programa na više procesora, jednog procesora sa većim brojem jezgara, kao i na arhitekturama tipa cluster i grid. Osnovna ideja je da se program pisan sekvencijalnom logikom na jedan jednostavan način prevede u paralelnu logiku. Shematski prikaz MPI komunikacije na četiri računara je prikazan na Slici 1.



Slika 1. Shematski prikaz MPI modela nad četiri računara.

Sa Slike 1 moguće je zaključiti da se poruke razmjenjuju između četiri računara. Na svim računarima se izvršava isti program unutar koga se nalaze MPI funkcije za slanje i primanje poruka kojim se razmjenjuju rezultati izvršavanja dijelova koda. Važno je uočiti da iako se svi rezultati objedinjuju na računaru sa oznakom Octave_0 međurezultati se porukama se mogu razmjenjivati i između ostalih uvezanih računara.

Implementacija MPI se realizuje tako što se piše samo jedan program ali od zavisno od identifikacionog broja procesa se izvršava određeni dio koda. Prosleđivanjem poruka između procesa razmjenjuju se i podaci do kojih se došlo izvršavanjem dijela koda koji je izvršio određeni proces koji dalje šalje poruku. Proces koji je primalac poruke nadalje može da u dalju obradu uvrsti primljene podatke. Takve poruke se najčešće mogu podijeliti na blokirajuće i neblokirajuće poruke. Za blokirajuće poruke je karakteristično da kada program dođe do koraka kada je potrebno poslati podatke drugome procesu njegovo izvršavanje se prekida sve dok se podaci u potpunosti ne pošalju. Dok kod neblokirajućih poruka se samo signalizira da postoje podaci za slanje ali da se ne čeka na njihovo potpuno slanje. Za vrijeme trajanja slanja i primanja podataka, istima se ne smije pristupati.

3. TEORIJSKA POZADINA ESTIMACIJE PARAMETARA 2DCPPS PRIMJENOM 2DCPF

Posmatrajmo model 2DCPPS:

$$\begin{aligned} y(n, m) &= x(n, m) + v(n, m), \\ (n, m) &\in [-N/2, N/2) \times [-M/2, M/2), \end{aligned} \quad (1)$$

gdje je

$$x(n, m) = A \exp\left(j \sum_{p=0}^3 \sum_{q=0}^3 c(p, q) n^p m^q\right), \quad (2)$$

sa A data amplituda signala, dok je $v(n, m)$ bijeli kompleksni Gausov šum osobina $E\{v(n, m)\} = 0$ i $E\{v(n, m)v^*(n, m)\} = \sigma^2 \delta(n - n_1, m - m_1)$. Predloženi pristup estimacije parametara je zasnovan na CPF-u predloženom od strane O'Shea i dr. za estimaciju čirp-rata CPPPS [3]-[4]. Generalizaciju pristupa za PPS većeg reda je moguće naći u [5]-[6]. U radu [7] je data magnituda 2DCPF kao

$$\begin{aligned} f_y(n, m; \Psi) &= \sum_{\tau_n = -\min(N/2-n-1, N/2+n)}^{\min(N/2-n-1, N/2+n)} \sum_{\tau_m = -\min(M/2-m-1, M/2+m)}^{\tau_m = -\min(M/2-m-1, M/2+m)} \\ & y(n + \tau_n, m + \tau_m) \times y(n - \tau_n, m - \tau_m) \\ & \times \exp\left(-j(\psi_n \tau_n^2 + \psi_m \tau_m^2 + 2\psi_{nm} \tau_n \tau_m)\right)^2 \end{aligned} \quad (3)$$

gdje je $\Psi = [\psi_n, \psi_{nm}, \psi_m]$ vektorski prostor koji se koristi za estimaciju izvoda drugog reda faze signala kao

$$\begin{aligned} \Omega(n, m) &= [\omega_n(n, m), \omega_{nm}(n, m), \omega_m(n, m)] \\ &= \arg \max_{\Psi} f_y(n, m; \Psi). \end{aligned} \quad (4)$$

Estimaciju parametara 2DCPPS je dalje moguće dobiti rješavanjem matricnih jednačina:

$$\begin{bmatrix} c(2, 0) \\ c(3, 0) \\ c'(2, 1) \end{bmatrix} = \begin{bmatrix} 2 & 6n_1 & 2m_1 \\ 2 & 6n_2 & 2m_2 \\ 2 & 6n_3 & 2m_3 \end{bmatrix} \begin{bmatrix} \omega_n(n_1, m_1) \\ \omega_n(n_2, m_2) \\ \omega_n(n_3, m_3) \end{bmatrix}$$

$$\begin{bmatrix} c(0, 2) \\ c(0, 3) \\ c'(1, 2) \end{bmatrix} = \begin{bmatrix} 2 & 6m_1 & 2n_1 \\ 2 & 6m_2 & 2n_2 \\ 2 & 6m_3 & 2n_3 \end{bmatrix} \begin{bmatrix} \omega_m(n_1, m_1) \\ \omega_m(n_2, m_2) \\ \omega_m(n_3, m_3) \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} c(1, 1) \\ c''(2, 1) \\ c''(1, 2) \end{bmatrix} = \begin{bmatrix} 1 & 2n_1 & 2m_1 \\ 1 & 2n_2 & 2m_2 \\ 1 & 2n_3 & 2m_3 \end{bmatrix} \begin{bmatrix} \omega_{nm}(n_1, m_1) \\ \omega_{nm}(n_2, m_2) \\ \omega_{nm}(n_3, m_3) \end{bmatrix}$$

Kao što se može uočiti konačni estimati za $c(2, 1)$ i $c(1, 2)$ u zavisnosti od problema mogu se dobiti izborom jedne od dvije vrijednosti dobijene rješavanjem sistema datim sa (5) ili traženjem njihove srednje vrijednosti. Nedostajući estimati se mogu naći na jednostavan način opisan u [7].

Važno je naglasiti da 2DCPF postaje 3D funkcija zavisna od $[\psi_n, \psi_{nm}, \psi_m]$ za jednu tačku (m, n) . Preatraga po 3D prostoru je neophodna da bi se pronašao maksimum 2DCPF.

4. PRIMJER IMPLEMENTACIJE MPI

Numerička provjera brzine izračunavanja je odrađena na primjenu signala datim sa (2) čiju su parametri $A=1$, $c00=1$; $c10=4.5e-1$; $c01=8.2e-2$; $c20=-1.5e-3$; $c11=6e-3$; $c02=-2.2e-3$; $c30=1.7e-5$; $c21=4e-5$; $c12=3.73e-5$; $c03=-1.35e-5$. 2DCPF se odvrđuje u tri trenutka i to (50,50), (50,40) i (40,50). Rezultati izvršavanja su dati u tabelama I i II gdje su prikazane odnosi vremena izvršavanja od broja procesora i odnosi vremena izvršavanja od broja procesa, respektivno. Pretraga se izvršava nad 3D prostorom dimenzija $100 \times 100 \times 100$. Uzeto je da se broj procesa dijeli na istobrojane grupe procesa, zavisno od broja procesora. Realizacija je urađena u Ocava alatu primjenom MPI i upoređivana su normalizovana vremena izvršavanja na jednom računaru sa procesorom sa jednim jezgrom, na jednom računaru sa procesorom sa dva jezgra i na clusteru sastavljenog od računara sa procesorima sa jednim jezgrom nad istim prostorom sa tri dimenzije.

TABLE I

BROJ PROCESA	TIME
Per	/
COR : 1	TIME _{NA JEDNOM JEZGRU}
1 CPU - 1 CORE per CPU	1
1 CPU - 2 CORE per CPU	0,47
2 CPU - 1 CORE per CPU	0,51
10 CPU -1 CORE per CPU	0,14

Lako se uočava da se vrijeme izvršavanja uveliko samanjuje i teži teoriskim granicama, odnosno vremenu izvršavanja na jednom procesoru podijeljeno sa brojem procesora. Odstupanja od očekivanih teoriskih vremena se obajšnjavaju gubitkom vremena koje se odvaja na komunikaciju između procesora unutar samog klastera kao i upotrebom računara čiji su procesori različitih tipova.

5. ZAKLJUČAK

U radu je predložen način za skraćivanje vremena izračunavanja parametara 2DCPPPS. Algoritam za izračunavanje 2DCPPPS je realizovan paralelnim programiranjem primjenom MPI. Rezulati su sumirani u

tabelama I i II gdje su prikazani odnosi vremena izvršavanja od broja procesora i od broja procesa. Na osnovu dobijenih rezultata se može zaključiti da paralelizacijom računarskih programa se mogu ostvariti mjerljive prednosti u izvršavanju istih, i to samo sa implementacijom drugačije logike pisanja programa za razliku od opšte prisutne logike sekvencijalnog programiranja.

TABLE II

BROJ PROCESA	TIME
Per	/
COR : 10	TIME _{NA JEDNOM JEZGRU}
1 CPU - 1 CORE per CPU	1
1 CPU - 2 CORE per CPU	0,43
2 CPU - 1 CORE per CPU	0,41
10 CPU -1 CORE per CPU	0,11

LITERATURA

- [1] Evan Marcus, Hal Stern, "Blueprints for High Availability: Designing Resilient Distributed Systems," John Wiley & Sons, 2003.
- [2] <http://www.gnu.org/software/octave/>.
- [3] P. O'Shea, "A new technique for instantaneous frequency rate estimation," *IEEE Signal Processing Letters*, Vol. 9, No. 8, pp. 251-252, Aug. 2002.
- [4] P. O'Shea, "A fast algorithm for estimating the parameters of a quadratic FM signal," *IEEE Tran. Signal Processing*, Vol. 52, No. 2, pp. 385-393, Feb. 2004.
- [5] M. Farquharson, P. O'Shea, G. Ledwich, "A computationally efficient technique for estimating the parameters of polynomial phase signals from noisy observations," *IEEE Tran. Signal Processing*, Vol. 53, No. 8, pp. 3337-3342, Aug. 2005.
- [6] M. Farquharson and P. O'Shea, "Extending the performance of the cubic phase function algorithm," *IEEE Tran. Signal Processing*, Vol. 55, No. 10, pp. 4767-4774, Oct. 2007.

HAOTIČNI OPTIMIZACIONI METOD SINTEZE PID REGULATORA U AVR SISTEMU A CHAOTIC OPTIMIZATION APPROACH FOR OPTIMUM DESIGN OF PID CONTROLLER IN AVR SYSTEM

Milovan Radulović, Vesna Rubežić, Martin Čalasan, *Elektrotehnički fakultet u Podgorici*

Sadržaj - Funkciju upravljanja naponom na krajevima sinhronog generatora, kao i upravljanje tokovima reaktivnih snaga, obavljaju pobudni sistemi sinhronih generatora preko automatskih regulatora napona (*Automatic Voltage Regulator - AVR*). U ovom radu je predstavljen jedan efikasan metod za određivanje proporcionalno-integralno-diferencijalnih parametara AVR sistema upotrebom Haotičnog optimizacionog algoritma (HOA), zasnovanog na jednačinama koje opisuju dobro poznati haotični Chua-in oscilator. Predloženi metod ima veoma dobru efikasnost u proračunu, kao i stabilne karakteristike konvergencije. Poredjenjem rezultata predloženog algoritma sa rezultatima dobijenim Genetičkim algoritmom (GA) i Ziegler Nichols-ovim (ZN) postupkom pokazuje se da predloženi algoritam daje bolje karakteristike sa aspekta oscilatornosti napona i trajanja prelaznog procesa.

Abstract - A synchronous machine excitation system used automatic voltage regulation (*Automatic Voltage Regulator - AVR*) to control generator voltage, as well as, a reactive power flow control. In this paper, the effective method for determining the proportional-integral-derivative (PID) controller parameters of the AVR system using the equations of the well known chaotic Chua's oscillator is presented. The proposed method has very good computational efficiency, as well as stable convergence characteristics. By comparing the results of the proposed algorithm with the results obtained by using Genetic Algorithm (GA) and Ziegler Nichols's (ZN) method it is shown that the proposed algorithm gives better performance from the aspect of the voltage oscillations as well as from the transient period duration.

1. UVOD

Svi potrošači električne energije zahtijevaju neprekidno napajanje električnom energijom, sa strogim zahtjevima po pitanju njenog kvaliteta. Međutim, energetske potrebe tih potrošača se mijenjaju iz trenutka u trenutak, pa se samim tim mijenjaju i tokovi snaga u elektroenergetskim sistemima (EES).

Regulacija napona i tokova reaktivnih snaga predstavljaju prioritetne zadatke u EES-ma. Funkciju održavanja napona na krajevima sinhronne mašine, a istovremeno i regulaciju tokova reaktivnih snaga, obavljaju pobudni sistemi, posredstvom automatskih regulatora napona (*automatic voltage regulator - AVR*). Od pouzdanosti i brzine reagovanja AVR-a ne zavisi samo napon na krajevima generatora, već zavisi i stabilnost cjelokupnog EES-a [1].

Centralni dio savremenih AVR-a predstavlja proporcionalno-integralno-diferencijalni (PID) regulator [2]. Razlika između podešene vrijednosti napona i stvarnog napona generatora, zajedno sa signalima iz brojnih limitera, predstavlja ulaz u PID regulator. Izlaz iz PID regulatora kontroliše ugao paljenja tiristora u ispravljačkom mostu, pa se na taj način povećava ili smanjuje napon pobude. Tačnije, na taj način se reguliše vrijednost pobudne struje sinhronog generatora, a samim tim i vrijednost njegovog izlaznog napona. Detaljna podjela pobudnih sistema i njihove karakteristike prikazane su u [1], [3], [4].

Najstariji, ali i najčešće korišćeni metod za određivanje parametara PID regulatora jeste Ziegel-Nichols-ov (ZN) metod [2],[5]. Osim njega, u literaturi se mogu naći i metode Cohen-Coon-a (CC) [6] kao i Chien-Hrones-Reswick-a (CHR) [2] koje, može se reći, predstavljaju unaprijeđene verzije ZN metode. Međutim, posljednjih decenija, kao

alternativa klasičnim metodama, moderne heurističke optimizacione tehnike, zasnovane na: fazi logici [7-8], genetskim algoritimima [9-10], neuralnim mrežama [11], teoriji rojeva [12-13], počele su da pridobijaju sve veću pažnju istraživača u cilju određivanja parametara PID regulatora, prije svega zbog njihove pouzdanosti u pronalaženju globalnih optimuma.

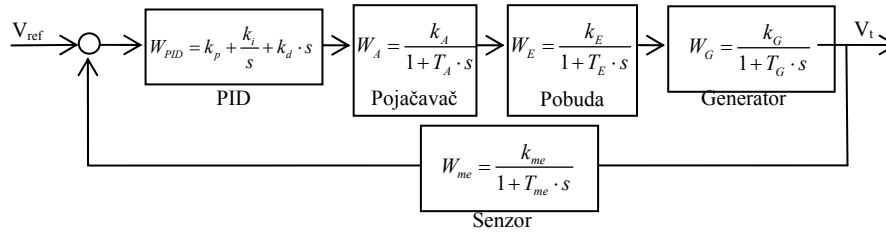
Među modernim heurističkim optimizacionim tehnikama ističu se one zasnovane na haotičnim sistemima. U literaturi postoji nekoliko algoritama koji koriste haotične sekvence za rješavanje optimizacionih problema u inženjerskim aplikacijama [14]-[18].

U ovom radu modifikovan je optimizacioni metod iz rada [14], baziran na Lozi preslikavanju. Modifikovani metod zasnovan je na jednačinama koje opisuju dobro poznati haotični Chua-in oscilator.

Rad je organizovan u nekoliko sekcija. U prvom dijelu, prikazan je uprošćeni blok dijagram AVR-a sinhronog generatora. Nakon toga opisan je haotični optimizacioni metod. Na kraju rada navedene su prednosti upotrebe modifikovanog optimizacionog modela u određivanju PID parametara AVR sistema sinhronne mašine u odnosu na ZN i metod predložen u [22], kao i smjernice za budući rad .

2. MATEMATIČKI MODEL POBUDNOG SISTEMA

Uticaj napona pobude na vrijednost izlaznog napona generatora se može predstaviti preko izuzetno kompleksne prenosne funkcije. Međutim, ako se posmatra neopterećeni generator, tj. rad generatora u praznom hodu, veza izlaznog napona generatora i napona pobude se može linearizovati. U takvom linearnom modelu, prenosna funkcija koja povezuje izlazni napon generatora sa ulaznim naponom pobude može



Slika 1. Blok dijagram AVR sistema sa PID kontrolerom

biti predstavljena preko pojačanja k_G i vremenske konstante T_G :

$$W_G = \frac{k_G}{1 + T_G \cdot s}. \quad (1)$$

Navedene konstante zavise od opterećenja i variraju između 0.7 i 1.0 i između 1.0 i 2.0, respektivno, u zavisnosti da li generator radi pri punom opterećenju ili u prazanom hodu.

Sa druge strane, uređaj za mjerenje napona generatora se sastoji od mjernog transformatora, ispravljačkog kola, filtra i drugih komponenti. Trofazni izlazni napon generatora se prvo smanjuje pomoću mjernog transformatora, a zatim se kroz ispravljačko kolo i filter pretvara u jednosmjerni napon. Zbog toga se prenosna funkcija mjernog uređaja može približno opisati kao element kašnjenja prvoga reda

$$W_{me} = \frac{k_{me}}{1 + T_{me} \cdot s}, \quad (2)$$

gdje je T_{me} - vremenska konstanta mjernog elementa (najčešće je reda desetine milisekunde).

Kod statičkih pobudnih sistema, pobudni namotaj sinhrona mašine se napaja sa krajeva same mašine koju pobuđuje, nakon što se taj izlazni napon smanji pobudnim transformatorom i ispravi kroz ispravljač. Prema tome, izlazni napon (struja) generatora snabdijeva pobudu, tj. pobudni namotaj. Taj pojačavački model može biti opisan preko pojačanja k_A i vremenske konstante T_A :

$$W_A = \frac{k_A}{1 + T_A \cdot s}. \quad (3)$$

Uobičajne vrijednosti za pojačanje k_A se kreću između 10 i 400, a vrijednost vremenske konstante T_A se kreće između 0.02 i 0.1 sekunde.

Prenosna funkcija savremene pobude je predstavljena pomoću pojačanja k_E i vremenske konstante T_E :

$$W_E = \frac{k_E}{1 + T_E \cdot s}. \quad (4)$$

Uobičajne vrijednosti za pojačanje k_E se kreću između 1 i 400, a vrijednost vremenske konstante T_E se kreće između 0.1 i 1 sekunde.

PID kontroler u sistemima pobude sinhronih generatora se koristi za poboljšanje karakteristika dinamičkog odziva u vremenskom domenu, kao i da se smanji ili eliminiše greška u stanju mirovanja (ustaljeni režim).

Prenosna funkcija PID kontrolera AVR-a se može napisati u sljedećem obliku:

$$W_{PID} = k_p + \frac{k_i}{s} + k_d \cdot s. \quad (5)$$

Blok dijagram AVR sistema sa PID kontrolerom je predstavljen na slici 1.

3. HAOTIČNI OPTIMIZACIONI ALGORITAM (HOA)

3.1. Haotični sistemi

Haos je oblik ustaljenog stanja koje, pod određenim uslovima, ispoljavaju mnogi prirodni i od čovjeka stvoreni sistemi. Osnovne osobine haotičnih signala su velika osjetljivost na početne uslove, dugovremenska nepredvidivost i šumu sličan izgled signala. Mala promjena parametara ili početnih vrijednosti može dovesti u budućnosti do sasvim različitih ponašanja kao što su: ravnotežna tačka, periodične oscilacije ili haotično rješenja. Detalji o haotičnom ponašanju mogu se naći u [19-21].

U optimizacionim procedurama haotične sekvence mogu biti uspješno primijenjene, zato što haotična trajektorija u prostoru stanja ne može sjeći samu sebe i haotični atraktor neće nikada posjetiti istu tačku [19-21]. Pretraživanje pomoću haosa predstavlja alternativu statističkom pretraživanju koje zavisi od vjerovatnoće. Haotično pretraživanje je brže, povećana je brzina konvergencije i preciznost rezultata.

U literaturi se mogu naći različiti tipovi haotičnih jednačina koje su primijenjene u optimizacionim metodama. To su: Logističko, Tent, Gausovo, Lozi i druga iterativna preslikavanja kao i jednačina koja opisuje Lorencov sistem [14]-[18]. Takođe, primijenjene su različite haotične optimizacione metode.

U ovom radu predložena je haotična optimizaciona procedura zasnovana na jednačinama koje opisuju Chua-in oscilator [19].

Chua-in oscilator ispoljava bogato dinamičko ponašanje. Za određene vrijednosti parametara u prostoru stanja se zapažaju ravnotežna tačka, granični krug, spiralni ili duplo-spiralni haotični odnosno čudni atraktor [19].

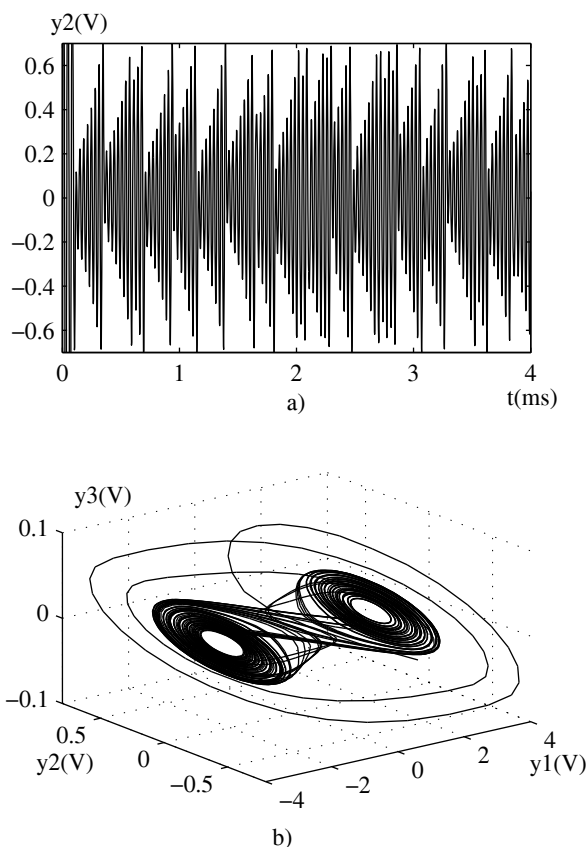
U ovom radu korišćen je duplo-spiralni haotični atraktor koji dobro posjećuje prostor stanja. Jednačine koje opisuju Chua-in sistem kao i parametri koji odgovaraju duplo-spiralnom atraktoru mogu se naći u [19].

Na slici 2a) prikazan je vremenski oblik signala y_2 , a na slici 2b) odgovarajući duplo-spiralni haotični atraktor.

U ovom radu vrijednosti promjenljivih $y_i, i=1,2,3$ su normalizovane u opsegu [0,1]. Ova transformacija je data sa:

$$z_i(t) = \frac{y_i(t) - a}{b - a},$$

gdje je a maksimalna, a b minimalna vrijednost signala.



Sl. 2. a) Vremenski oblik signala y_2 ; b) Duplo-spiralni atraktor.

3.2. Predloženi haotični optimizacioni algoritam

Predloženi haotični optimizacioni metod zasnovan je na jednačinama koje opisuju Chua-ino oscilatorno kolo.

Mnogi optimizacioni problemi sa kontinualnim promjenljivim mogu biti formulisani kao sledeći funkcionalni optimizacioni problem:

- Odrediti X koje minimizuje kriterijumsku funkciju $f(X)$.

Vektor $X = [x_1, x_2, \dots, x_n]$ sadrži promjenljive $x_i \in [D_i, G_i]$, $i = 1, 2, \dots, n$, koje su ograničene donjom (D_i) i gornjom (G_i) dozvoljenom vrijednošću.

U našem radu, usvajamo $n = 3$ i $X = [x_1, x_2, x_3] = [k_p, k_i, k_d]$.

Procedura pretraživanja zasnovana na haotičnim Chua-inim jednačinama sastoji se iz dva dijela: globalnog i lokalnog pretraživanja.

Globalno pretraživanje

Globalno pretraživanje može se opisati u nekoliko koraka.

1. a) Odrediti parametre za Chua-ine jednačine.
b) Odrediti početne uslove: $y_1(0), y_2(0), y_3(0)$.
c) Odrediti maksimalni broj iteracija M_g za haotično globalno pretraživanje.
2. Promjenljive z_1, z_2, z_3 u k -toj iteraciji množimo sa faktorom 1.5: $x_i(k) = 1.5z_i(k)$ zato što su $0 \leq k_p, k_i, k_d \leq 1.5$.

3. U k -toj iteraciji za koordinate tačke $(x_1(k), x_2(k), x_3(k))$ Chua-inog atraktora u prostoru stanja računa se kriterijumska funkcija $f(X(k))$.

4. Koordinate tačke $X^* = (x_1^*, x_2^*, x_3^*)$ za koje se dobije najmanja vrijednost kriterijumske funkcije f unose se u algoritam za lokalno pretraživanje.

Lokalno pretraživanje

Lokalno pretraživanje je pretraživanje u okolini tačke $X^* = (x_1^*, x_2^*, x_3^*)$. Može se opisati u nekoliko koraka.

1. Odrediti broj iteracija za lokalno pretraživanje M_L .
2. U k -toj iteraciji važi: $x_i(k) = x_i^* \pm \alpha z_i(k)$. Znak + ili - se bira slučajno.
3. Koordinate tačke x_1, x_2, x_3 za koje se dobije najmanja vrijednost kriterijumske funkcije f se proglašavaju za k_p, k_i, k_d , respektivno.

Parametar α je važan parametar u haotičnom lokalnom pretraživanju. Manje α omogućava fino pretraživanje oko tačke X^* , dok veće α omogućava pretraživanje u širem opsegu.

4. REZULTATI SIMULACIJA

U prikazanim simulacijama korišćeni su parametri modela sinhronog generatora dati u [14]. Pretraživanje je vršeno u opsegu $0 \leq k_p, k_i, k_d \leq 1.5$. Dakle, $D_i = 0$, $G_i = 1.5$, $i = 1, 2, 3$. Broj iteracija za globalno pretraživanje je $M_g = 1000$, broj iteracija za lokalno pretraživanje je $M_L = 300$. Izvršeno je 50 nezavisnih ponavljanja sa 50 različitih skupova početnih uslova. U našim simulacijama α je iz opsega od 0.015 do 0.15.

Rezultati prikazani u radu rezultat su simulacije u programskom paketu MATLAB. U cilju određivanja optimalnih parametara PID regulatora, primjenom predloženog HOA, izabrana je kriterijumska funkcija koja predstavlja linearnu kombinaciju ITAE kriterijuma (Integral of Time-multiplied Absolute Error) proširenog članom koji obezbjeđuje smanjenje preskoka u odzivu sistema. Kriterijumska funkcija data je izrazom:

$$f(X) = \sum_{t=1}^N t |V_{ref}(t) - V_i(t)| + \omega \{ \max[V_i(t)] - V_{ref}(t) \} \quad (6)$$

gdje su: $V_{ref}(t)$ i $V_i(t)$ referentni naponski ulaz i napon na izlazu sistema, respektivno; N je broj odbiraka, a ω pozitivan težinski faktor. U ovom radu je primjenjena jedinična step funkcija, pa je $V_{ref}(t) = 1$, $t = 1, \dots, N$, a težinski faktor je $\omega = 600$.

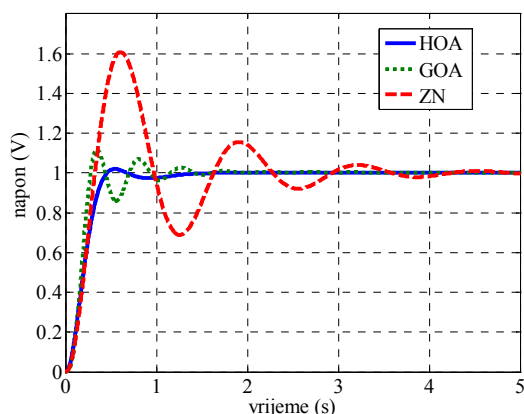
Rezultati dobijeni simulacijom su prikazani u Tabeli I. Izvršeno je poređenje karakterističnih parametara odziva na step ulaz sa druga dva tipa AVR PID regulatora. Primjenom ZN metode dobijeni su parametri prvog PID regulatora. Kao drugi primjer za poređenje kvaliteta predloženog rješenja preuzeti su parametri PID regulatora dati u [22] dobijeni Genetskim optimizacionim algoritmom (GOA). Na kraju su dati rezultati dobijeni predloženim Haotičnim optimizacionim algoritmom (HOA).

Tabela I : Vrijednosti parametara PID kontrolera za različite optimizacione algoritme

Metod	k_p	k_i	k_d	$t_u(s)$	$t_k(s)$	$t_s(s)$	$\Pi(\%)$
ZN-PID	1.155	2.252	0.142	0.251	0.204	3.64	60.8
GOA-PID [22]	4.918	0.657	0.78	0.175	0.175	1.33	11.6
HOA-PID	0.6272	0.4542	0.2085	0.239	0.196	1.09	1.2

t_u - vrijeme uspona, t_k - vrijeme kašnjenja, t_s - vrijeme smirenja, Π - preskok

Na slici 3. prikazan je odziv na jediničnu step funkciju sistema za sva tri regulatora.



Sl. 3. Odziv sistema za analizirane tipove PID regulatora

Kao što se može vidjeti na osnovu rezultata datih u Tabeli I i slici 3. predloženi HAO obezbeđuje kvalitetniji odziv sistema po svim parametrima u odnosu na ZN-PID regulator. U odnosu na GOA-PID [22] naš algoritam ostvaruje aperiodičnost odziva sa manjim preskokom i manjim vremenom smirenja.

5. ZAKLJUČAK

U ovom radu prikazan je postupak određivanja parametara PID regulatora AVR sistema primjenom Haotičnog optimizacionog algoritma. Predloženi metod integriše HOA sa modifikovanim ITAE kriterijumom performanse. Simulacija na modelu AVR sistema pokazuje da predloženi metod uspješno nalazi optimalne parametre PID regulatora.

Pokazano je da AVR sistem sa parametrima PID regulatora, određenim pomoću predloženog HOA, obezbeđuje otklanjanje oscilatornosti odziva, sa zanemarljivim preskokom. Takođe, u odnosu na metode podešavanja regulatora predloženim u [22] i klasičnim ZN metodom obezbeđuje se manje vrijeme smirenja.

U daljem istraživanju navedenog problema izvršice se detaljna analiza poređenja robusnosti posmatranog sistema pri djelovanju poremećaja, kao i uticaj izbora različitih kriterijumskih funkcija na promjenu karakteristika dinamičkog ponašanja sistema.

LITERATURA

[1] IEEE Recommended Practice for Excitation System Models for Power System Stability Studies, IEEE Std 421.5-1992, IEEE, New-York, NY, USA, 1992.

[2] K. Astrom, T. Hagglund, PID Controllers: Theory, Design, and Tuning, (second edition), Department of Automatic Control Lund Institute of Technology, Box 118, Sweden

[3] A.N.A Alla: Simulation Model of Brushless Excitation System, *Amer. J Appl Sci*, pp. 1079-1083, 2007.

[4] I. C Report, „Excitation system models for power system stability studies.“ *IEEE Trans Power Appar. and Syst*, Vol. PAS-100, No. 2, February 1981.

[5] Ziegler, J. G., Nichols, N. B., Optimum Settings for Automatic Controllers, *Trans. ASME*, Vol. 64, pp. 759-768, 1942.

[6] Cohen, G. H., Coon G.A., Theoretical consideration of retarded control, *Trans ASME*, Vol. 75, pp. 27-34., 1953.

[7] S.J. Ho, L.S. Shu, S.Y. Ho, “Optimizing fuzzy neural networks for tuning PID controllers using an orthogonal simulated annealing algorithm OSA,” *IEEE Trans Fuzzy Syst*, 2006, Vol. 14, No. 3, pp. 421-34.

[8] H.X. Li, L. Zhang, G. Chen, “An improved robust fuzzy-PID controller with optimal fuzzy reasoning,” *IEEE Trans Syst, Man, Cybernet – Part B: Cybernet*, Vol. 35, No. 6, pp. 1283-1294, 2005.

[9] C. Vlachos, D. Williams, J.B. Gomm, “Solution to the Shell standard control problem using genetically tuned PID controllers.” *Contr Eng Pract*, Vol. 10, No. 2, pp.151-63., 2002.

[10] Varsek, A., Urbacic, T., Filipic, B., “Genetic Algorithms in Controller Design and Tuning,” *IEEE Trans. Sys. Man and Cyber*, Vol. 23, No. 5, pp. 1330 – 1339, 1998.

[11] P.C. Chen, J.K. Mills. “Synthesis of neural networks and PID control for performance improvement of industrial robots. *J Intell Robot Syst*, Vol. 20, No 2-4, pp.157-80., 1997.

[12] C.C. Kao, C.W. Chuang, R.F. Fung, “The self-tuning PID control in a slider-crank mechanism system by applying particle swarm optimization approach”, *Mechatronics*, Vol 16, No. 8, pp. 513-22. 2006

[13] Z.L. Gaing, “A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System,” *IEEE Transactions on Energy Conversion*, Vol. 19, No. 2, pp. 684-391, June 2004.

[14] L.S. Coelho, “Tuning of PID controller for an automatic regulator voltage system using chaotic optimization approach”, *Chaos, Solitons and Fractals*, Vol. 39, pp.1504-1514, 2009

[15] B. Li, W. Jiang, “Optimizing complex functions by chaos search,” *Cybernet Syst*. vol. 29, no. 4, pp. 409-419, 1998.

[16] XQ. Zuo, YS. Fan, “A chaos search immune algorithm with its application to neuro-fuzzy controller design,” *Chaos, Solitons & Fractal*, vol. 30, no.1, pp.94-109, 2006.

[17] B. Liu, L. Wang, YH. Jin, F. Tang, DX. Huang, “Improved particle swam optimization combined with chaos,” *Chaos, Solitons & Fractal*, vol. 25, no.5, pp.1261-1271, 2005.

[18] D. Yang, G. Li, G. Cheng, “On the efficiency of chaos optimization algorithms for global optimization,” *Chaos, Solitons & Fractal*, vol. 34, no.4, pp.1366-11375, 2007.

[19] M. P. Kennedy, “Three Steps to Chaos-Part II: A Chua's Circuit Primer,” *IEEE Transactions on Circuits and Systems*, vol. 40, no.10, pp. 657-674, 1993.

[20] J. C. Sprott, “Chaos and Time-Series Analysis,” *Oxford University Press*, 2003.

[21] V. G. Ivančević, T. T. Ivančević, “Complex Nonlinearity: Chaos, Phase Transitions, Topology Change and Path,” *Springer Verlag Berlin Heidelberg*, Germany, 2008.

[22] M. P. Calasan, T. M. Dlabac, M. M. Ostojic “PID parameters determination of synchronous machine AVR system,” *IREACO*, vol. 6, no. 4, pp 425-430, July 2013.

UTICAJ KRITERIJUMA MINIMIZACIJE NA UPRAVLJANJE POTROŠNOM ELEKTRIČNE ENERGIJE

MINIMIZATION CRITERION INFLUENCE ON ELECTRICITY DEMAND MENAGEMENT

Tamara Bojičić, Vesna Popović-Bugarin, *Elektrotehnički fakultet, Podgorica*

Sadržaj: *Algoritmi za upravljanje potražnjom/potrošnjom električne energije imaju za cilj smanjenje iznosa koji potrošači plaćaju za utrošenu električnu energiju i dobijanje što nižeg odnosa maksimalnog i srednjeg opterećenja mreže u toku dana. U radu je analiziran uticaj načina obračuna cijene utrošene električne energije na performanse algoritma za optimizaciju potrošnje. Pokazalo se da rezultati optimizacionih algoritama zavise od načina računanja cijene utrošene električne energije. Analizirana je mogućnost direktne primjene aktuelne funkcije cijene. Predložen je adekvatan optimizacioni algoritam, te date smjernice za modifikaciju aktuelne funkcije cijene električne energije, kako bi se dobila bolja optimizacija uz što manje izmjene iste.*

Abstract: *Algorithms for optimization of electricity demand/consumption are aimed at reducing the amount paid by customers for consumed electricity, as well as at obtaining the lowest possible peak-to-average load ratio of network during the day. Impact of function for consumed electricity price calculation on performances of algorithm for optimization of electricity consumption is analysed in this paper. It is shown that results of optimization algorithms depend on this price function. Possibility of direct application of the current price function is analysed. Adequate linear optimization algorithm is proposed and guidelines for possible modification of the current price function of the consumed electricity are given in order to achieve the best optimization possible with the slightest modification of the concerned function.*

1. UVOD

Način funkcionisanja energetskog sistema u nekoj zemlji postaje njen ključ razvoja i integracije u savremene tokove tehnologije. Smart grid (pametna mreža) predstavlja novu interaktivnu mrežu upravljanja električnom energijom kod koje "pametni" uređaji, napredni softver i dvosmjerna komunikacija predstavljaju oslonac funkcionisanja [1]. Digitalna tehnologija omogućava prenos podataka u oba smjera, od potrošača do snabdjevača električnom energijom i obratno, dok su senzori postavljeni duž prenosnih linija ono što mrežu čini "pametnom". Dostupnost informacija, koje ne ugrožavaju privatnost potrošača, dovodi do veće efikasnosti korišćenja električne energije. Pomoću pametnih brojlara potrošači u svakom trenutku imaju uvid koliko, kada i po kojoj cijeni koriste električnu energiju.

U [2] je predložen algoritam za optimizaciju potražnje električne energije koji se zasniva na automatskom raspoređivanju potražnje, baziranom na osnovama teorije igara. U [2] se koristi kvadratna funkcija cijene utrošene električne energije, koja se veoma razlikuje od funkcije cijene trenutno korišćene u praksi. U radu istražujemo mogućnost korišćenja aktuelne funkcije cijene u optimizacionom algoritmu predloženom u [2]. Algoritam podrazumijeva kompleksnu optimizacionu funkciju, te je ovdje modifikovan korišćenjem linearne optimizacije, koja je adekvatna za aktuelnu funkciju cijene. Cilj rada je da se ukaže na potrebu za izmjenom aktuelne funkcije cijene i daju smjernice za izmjenu iste, kako bi te izmjene bile prihvatljive potrošačima, ali i kako bi učinili mogućim i produktivnim primjenu optimizacionih algoritama.

2. SMART GRID

Pametna mreža predstavlja budućnost elektrodistribucije. Jedna od njenih prednosti jeste omogućavanje snabdjevačima električne energije, ne samo da prate kolika je dnevna potražnja u određenom periodu dana, već i da upravljaju njome putem DSM (sistem za upravljanje potražnjom električne energije – Demand Side Management) [3]. Većina DSM programa je razvijano za sistem u kojem se komunikacija vrši između snabdjevača i svakog krajnjeg potrošača. S obzirom na prednosti pametne mreže, DSM program bi trebao uzeti u obzir, ne samo zahtjeve krajnjih korisnika, već i njihov uticaj međusobno. Na taj način se može analizirati i kontrolisati problem PAR-a (odnos maksimalnog i srednjeg opterećenja mreže, Peak to Average Ratio), koji zavisi od rasporeda ukupnog opterećenja mreže.

Model, kojim možemo predstaviti energetski sistem, sastoji se od većeg broja domaćinstava i izvora električne energije povezanog na električnu mrežu. Pretpostavka je da se u svakom brojilu nalazi automatski optimizator električne potrošnje, koji bi upravljao rasporedom potrošnje. Svako brojilo povezano je na liniju napajanja koja dolazi iz izvora električne energije, međusobno, kao i sa izvorom električne energije. Efikasnost algoritma predloženog u [2] se zasniva na kvadratnoj funkciji računanja cijene utrošene električne energije, koja je takva da cijena koju jedno domaćinstvo plaća u nekom satu zavisi, po kvadratnom pravilu, ne samo od njegove potrošnje, već i od potrošnje svih ostalih domaćinstava. Na ovaj način se otvara mogućnost korišćenja teorije igara u optimizaciji potražnje električne energije.

3. TEORIJA IGARA

Teorija igara je nauka koja se bavi strategijama. Predstavlja studiju konflikta i saradnje. Pokušava da definiše matematičke i logičke akcije koje bi igrači trebali preduzeti kako bi sebi obezbijedili najbolji profit. Profit svakog igrača zavisi od odabira strategije. Osnovna podijela igara je na kooperativne i nekooperativne. Kooperativne igre su one kod kojih se takmiče grupe igrača ("koalicije"). Nasuprot njima, nekooperativne igre su one u kojima igrači donose odluke samostalno i svaka saradnja je dobrovoljna, ali ne i podrazumijevana. Cilj teorije igara je odrediti ponašanje, djelovanje, akcije i odluke učesnika igre koje će im donijeti najbolji profit. Uzevši u obzir ove činjenice, korišćenje teorije igara u analizi kontrole i upravljanja potrošnjom električne energije i te kako ima smisla. Potrošači – učesnici igre, sa svojim strategijama – dnevnim rasporedom opterećenja čine osnovu igre optimizacije rasporeda potrošnje električne energije, koja za cilj ima smanjenje iznosa koji svaki potrošač pojedinačno plaća, ali i smanjenje PAR-a. Nešov (John Forbes Nash, Jr.) ekvilibrijum je koncept rješenja nekooperativnih igara u kojima učestvuju dva ili više igrača. Ako je svaki igrač izabrao strategiju, i nijedan igrač ne može da profitira promjenom svoje strategije, pod pretpostavkom da ostali igrači ne promjene svoje strategije, onda trenutni skup izabranih strategija i odgovarajućih dobitaka predstavlja Nashov ekvilibrijum. U našoj analizi, postizanje Nešovog ekvilibrijuma bi značilo onemogućavanje prijavljivanja lažnog rasporeda potrošnje nekog domaćinstva, jer bi to rezultovalo većim iznosom koje to domaćinstvo treba da plati. Poželjno je da svaki DSM garantuje postizanje Nešovog ekvilibrijuma.

4. UTICAJ FUNKCIJE CIJENE NA DSM

Uloga svakog optimizatora jeste utvrđivanje optimalnog rasporeda potrošnje električne energije u toku 24 sata, u skladu sa potrebama domaćinstva. Jedan od glavnih ciljeva je podstaći domaćinstva da koriste prednosti pametne mreže, samim tim i karakteristike optimizatora, kao i da poštuju raspodijelu potrošnje određenu na ovaj način. Najbitniji korak u realizaciji algoritma jeste adekvatno definisanje načina računanja cijene. Svako domaćinstvo $n \in N$, posjeduje određen broj uređaja $a \in A_n$ gdje je A_n skup uređaja domaćinstva n , dok N predstavlja ukupan broj domaćinstava. Za svaki od tih uređaja definiše se vektor rasporeda potrošnje električne energije $x_{n,a} = [x_{n,a}^1, \dots, x_{n,a}^H]$, gdje je $x_{n,a}^h$ potrošnja uređaja a , domaćinstva n u času h . Posmatraju se 24 časa. Ukupna potrošnja n -tog domaćinstva u času h je $l_n^h = \sum_{a \in A_n} x_{n,a}^h$. Ukupna potrošnja svih domaćinstava u času h , u oznaci l_n je $L_h = \sum_{n \in N} l_n^h$. Uloga optimizatora potrošnje električne energije u svakom pametnom broju je da odredi raspored potrošnje svih uređaja jednog domaćinstva u svakom času, tj. raspored članova vektora $x_{n,a}$, u cilju minimizacije iznosa koji se plaća za utrošenu električnu energiju, a posredno i PAR-a. Navedena optimizacija se vrši uzimajući u obzir zaštitu privatnosti potrošača i pod usovima

koje definišu sama domaćinstva. Ti uslovi čine skup X_n , kojeg definiše svako domaćinstvo $n \in N$ i koji predstavlja skup validnih rasporeda potrošnje n -tog domaćinstva, dok x_n predstavlja vektor rasporeda potrošnje svih uređaja domaćinstva n :

$$X_n = \left\{ x_n \mid \begin{array}{l} \sum_{h=\alpha_{n,a}}^{\beta_{n,a}} x_{n,a}^h = E_{n,a}, \\ x_{n,a}^h = 0, \forall h \in H \setminus H_{n,a}, \\ \gamma_{n,a}^{\min} \leq x_{n,a}^h \leq \gamma_{n,a}^{\max}, \forall h \in H_{n,a} \end{array} \right\} \quad (1)$$

Svako domaćinstvo za svaki uređaj definiše period dana u kada je poželjno da uređaj radi, sa $\alpha_{n,a}$ je označen početni, a sa $\beta_{n,a}$ krajnji čas toga perioda. $E_{n,a}$ je ukupna dnevna potrošnja električne energije uređaja a , domaćinstva n . Ona mora biti smještena u dozvoljenom periodu, a jednaka nuli u periodu van njega. Takođe, potrošnja uređaja u času h mora biti veća od minimalne $\gamma_{n,a}^{\min}$, a manja od maksimalne $\gamma_{n,a}^{\max}$ potrošnje uređaja dok je isti u stanju pripravnosti (standby režim). Sada se optimizacija svodi na minimizaciju cijene

$$\text{minimize}_{x_n \in X_n, \forall n \in N} \sum_{h=1}^H C_h \left(\sum_{n \in N} \sum_{a \in A_n} x_{n,a}^h \right) \quad (2)$$

gdje je C_h funkcija cijene, uz zadovoljenje uslova definisanih u (1). Optimizacioni problem (2) ćemo rješavati pomoću IPM metoda (algoritam za rješavanje linearnih i nelinearnih-konveksnih problema optimizacije - Interior Point Method) [4]. Pod pretpostavkom da sva ostala domaćinstva podese svoj raspored potrošnje prema domaćinstvu n , i sa datim vektorom rasporeda potrošnje svih ostalih domaćinstava izuzev n -tog, lokalni optimizacioni problem svodi se na

$$\text{maximize}_{x_n \in X_n} P_n(x_n : x_{-n}) \quad (3)$$

Zarad uvođenja elemenata igre u [2] se uvode dvije pretpostavke. Ukoliko sa b_n označimo ukupan iznos koji snabdjevači električne energije dnevno naplate domaćinstvu, prva pretpostavka je

$$\sum_{n \in N} b_n \geq \sum_{h=1}^H C_h \left(\sum_{n \in N} l_n^h \right) \quad (4)$$

Lijeva strana označava ukupnu dnevnu naplatu domaćinstvima, a desna ukupnu dnevnu cijenu. Ukoliko je odnos naplate i cijene jednak jedinici ($k=1$), budžet sistema je izbalansiran, dok odnos veći od jedinice ($k>1$) indicira na profit kompanije – snabdjevača. Druga pretpostavka je:

$$\frac{b_n}{b_m} = \frac{\sum_{h=1}^H l_n^h}{\sum_{h=1}^H l_m^h}, \forall n, m \in N \quad (5)$$

Dakle, naplata domaćinstvima je upravo srazmjerna njihovoj ukupnoj dnevnoj potrošnji. Efikasni model naplate mora uzeti u obzir obje pretpostavke (4) i (5) na osnovu kojih slijedi:

$$b_n = \frac{\sum_{h=1}^H l_n^h}{\sum_{m \in N} \sum_{h=1}^H l_m^h} = \frac{k \sum_{h=1}^H l_n^h}{\sum_{m \in N} \sum_{h=1}^H l_m^h} \left(\sum_{h=1}^H C_h \left(\sum_{m \in N} l_m^h \right) \right) = \Omega_n \sum_{h=1}^H C_h \left(\sum_{m \in N, a \in A_n} x_{n,a}^h \right) \quad (6)$$

Iznos cijene koju domaćinstvo n plaća zavisi od dnevnog rasporeda potrošnje svih ostalih domaćinstava, što vodi definisanju igre u kojoj domaćinstva predstavljaju igrače, dok njihovi rasporedi potrošnje predstavljaju strategije igrača. Sada se lokalna optimizacija (3) svodi na

$$\text{minimize}_{x_n, X_n} \sum_{h=1}^H C_h \left(\sum_{a \in A_n} x_{n,a}^h + \sum_{m \in N \setminus \{n\}} l_m^h \right) \quad (7)$$

što predstavlja efikasan model naplate električne energije. U [2] je korišćena kvadratna funkcija cijene $C_h(x) = a * x^2$, za $a = \text{const}$. Dakle, troškovi svakog domaćinstva zavise od sopstvenog, ali i rasporeda potrošnje svih ostalih domaćinstava. Svaki igrač teži takvom rasporedu potrošnje koji predstavlja rješenje problema minimizacije cijene, tj. optimizacionog problema. Pokazano je da se sa ovakvom realizacijom funkcije cijene postiže Nešov ekvilibrijum.

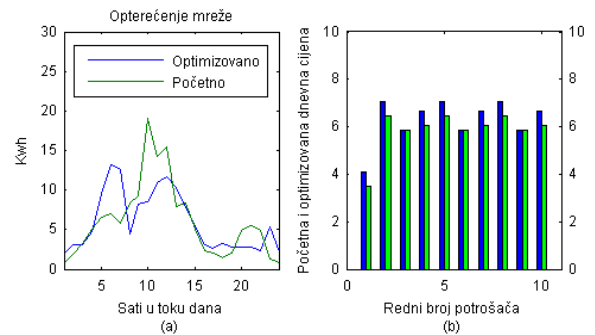
Ideja algoritma predloženog u [2] je da svaki optimizator rješava lokalni optimizacioni problem (7) koristeći IPM metod [4]. Za realizaciju IPM metoda u Matlabu koristimo funkcije `fmincon` ili `linprog`, `fmincon` ukoliko je funkcija koja se optimizuje kvadratna, tj. `linprog` ukoliko je ona linearna. Na početku optimizacije vrši se inicijalizacija vektora l_n , a nakon toga se vrši naizmjenična lokalna optimizacija pojedinačnih domaćinstava. Svaki korisnik definiše i uslove optimizacije za sve uređaje, tj skup X_n . Nakon završetka lokane optimizacije n -tog domaćinstva, provjerava se da li je optimizacijom izmijenjen raspored njegove potrošnje x_n . Ukoliko jeste, na osnovu novih vrijednosti ažurira se vektor l_n , i obavještavaju ostala domaćinstva o izmjeni. Lokalna optimizacija se ponavlja sve dok nijedno domaćinstvo ne objavi promjenu u rasporedu potrošnje x_n , odnosno l_n . Na ovaj način korisnici ne otkrivaju detalje o svojoj potrošnji, već samo ukupnu potrošnju po času, čime se štiti njihova privatnost. Razmjena informacija među korisnicima, kao što je slanje poruke o promjeni vektora l_n , se vrši putem LAN-a ili nekim drugim komunikacionim interfejsom. U [2] je vršena slučajna inicijalizacija l_n , dok smo u našoj realizaciji smatrali da se i na početku optimizacije ta informacija može

razmijeniti među korisnicima i time ubrzati konvergencija algoritma.

Dokazano je da algoritam u [2] konvergira Nešovom ekvilibrijumu igre, kao i da bi svaki korisnik, objavom pogrešnog rasporeda potrošnje l_n , imao veći račun za električnu energiju. Kada bi se navedeni algoritam direktno primijenio, uz način računanja cijene koji je trenutno aktuelan u većini zemalja, usljed njegove linearnosti, cijene koju plaćaju pojedinačna domaćinstva bi se računale potpuno nezavisno. Ne bi bilo elementa igre, pa samim tim se ne bi moglo govoriti o postizanju Nešovog ekvilibrijuma. Da bi se proučilo do kakvih bi se rezultata došlo direktnom primjenom linearne funkcije cijene, pored primjene algoritma uz IPM za linearnu optimizaciju, analizirao se i rezultat dobijeni primjenom istog algoritma uz kvadratnu cijenu utrošene električne energije, ali bez pretpostavki (4) i (5). Dakle, cijena koju pojedinačno domaćinstvo plaća neće zavisiti od rasporeda potrošnje ostalih domaćinstava.

5. PRIMJERI

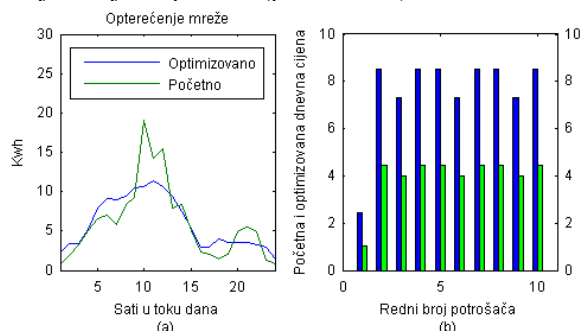
U svim simulacijama je analiziran uzorak od 10 domaćinstava. Svako od njih ima kombinaciju uređaja sa vremenski fiksiranim vremenom aktivacije (frižider, sijalica i td.) i uređaja sa vremenom aktivacije koje se može pomjerati duž 24h (mašina za veš, mašina za suđe itd.). Optimizator ima uticaj samo na uređaje čije se vrijeme aktivacije može pomjerati. U simulaciji je uzeto u obzir da domaćinstva koriste dvotarifna brojila, te da je opterećenje mreže veće u popodnevnom, a manje u večernjim časovima.



Slika 1 (a) Opterećenje mreže prije i nakon optimizacije u toku 24h. (b) Početna (plavi barovi) i optimizovana (zeleni barovi) dnevna cijena pojedinačnih domaćinstava koristeći linearnu funkciju cijene.

Prvo je analizirana mogućnost optimizacije direktnom primjenom aktuelne funkcije cijene utrošene električne energije $C_h(x_n) = a * x_n$. a je koeficijent koji iznosi 0.5 centi/kwh za višu (VT), tj. 0.2 centa/kwh za nižu tarifu (NT). Funkcija je ekvivalentna računanju cijene električne energije u Crnoj Gori. U periodu od 23h – 7h aktivna je naplata niže, a u periodu van, više tarife. Na Slika 1 prikazan je rezultat optimizacije ukupnog opterećenja mreže u toku 24h (plave linije), ali i njegova početna vrijednosti (zeleno linije). Ukupno opterećenje mreže je u časovima više tarife smanjeno u odnosu na početnu vrijednost, što je direktna posljedica odabira funkcije cijene. Došlo je i do peglanja pikova u časovima više tarife, što mrežu čini stabilnijom, ali i

do očekivane pojave pikova u časovima niže tarife. PAR se smanjio za 30.7%, sa 0.1326 na 0.0924. Optimizovana dnevna cijena pojedinačnih domaćinstava (zeleni barovi) je i ovdje manja od početne (plavi barovi).



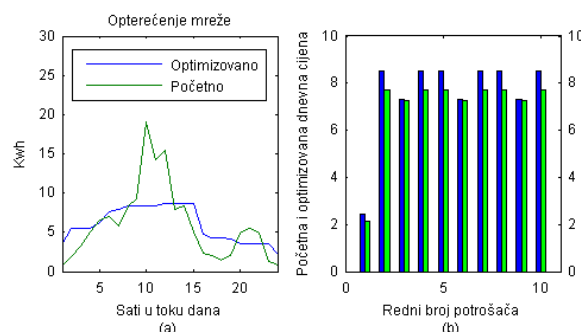
Slika 2 (a) Opterećenje mreže prije i nakon optimizacije u toku 24h. (b) Početna (plavi barovi) i optimizovana (zeleni barovi) dnevna cijena pojedinačnih domaćinstava koristeći modifikovanu kvadratnu funkciju cijene.

U nastavku su dati rezultati optimizacije kada bi se koristila kvadratna funkcija cijene, koja zavisi samo od rasporeda lokalne potrošnje pojedinačnog domaćinstva.

$$\text{minimize}_{x_n, X_n} \sum_{h=1}^H \left(C_h \left(\sum_{a \in A_n} x_{n,a}^h \right) \right) + C_h \left(\sum_{m \in N \setminus \{n\}} I_m^h \right) \quad (8)$$

Cijene pojedinačnih domaćinstava se računaju nezavisno. IPM metod za svoju realizaciju, kada je optimizaciona funkcija kompleksna (3), koristi i gradijent i Hesijan funkcije koju optimizuje. Gradijent i Hesijan se mogu obezbijediti funkciji *fmincon* na više načina. Ukoliko se gradijent i hesijan funkcije koja se optimizuje mogu izračunati, oni se moraju realizovati u okviru nje, što je i učinjeno u simulacijama. Međutim, ako ih ne obezbijedi korisnik, *fmincon* funkcija ima ugrađen način njihovog računanja. Algoritam radi najbrže uz upotrebu Hesijana kojeg realizujemo u okviru funkcije optimizacije. Na slici 2 se vidi da je optimizovana cijena pojedinačnih domaćinstava (zeleni barovi) gotovo za 50% manja od početne (plavi barovi). PAR-a je nakon optimizacije manji za 46.15% što ukazuje da bi stabilnost mreže bila povećana, a potrošači podstaknuti na upotrebu predloženog metoda.

Na slici 3 prikazan je rezultat optimizacije kada je funkcija cijene realizovana kao kvadratna funkcija (7) na način opisan u radu [2]. Ovakav način računanja cijene utrošene električne energije je pažljivo biran. Cijena koju plaća jedno domaćinstvo zavisi ne samo od njegove potrošnje, već i od potrošnje svih ostalih domaćinstava. Na ovaj način se uvodi element igre. Primjećujemo da je, u odnosu na ostale načine realizacije funkcije cijene, došlo do najfinijeg peglanja pikova u mreži, tj. najnižeg PAR-a. On je smanjen za 53.85%. Ukupna optimizovana dnevna cijena pojedinačnih domaćinstava (zeleni barovi) je i ovdje manja od početne (plavi barovi).



Slika 3 (a) Opterećenje mreže prije i nakon optimizacije u toku 24h. (b) Početna (plavi barovi) i optimizovana (zeleni barovi) dnevna cijena pojedinačnih domaćinstava koristeći kvadratnu funkciju cijene iz [2].

Imajući na umu tri analizirane funkcije cijene utrošene električne energije, zaključujemo da direktna primjena trenutno aktuelne funkcije ne bi dala optimalne rezultate. Doveda bi do automatskog uključivanja uređaja u vrijeme kada je cijena električne energije niža, što sada većina domaćinstava sam čini. S druge strane, ne bi optimalno iskoristila veliki dio perioda u kojem je niža tarifa, pa bi moglo doći do povećanja opterećenja mreže na početku niže tarife. Stoga je potrebno da se ova funkcija modifikuje tako da uzme u obzir ukupno opterećenje. U budućem istraživanju planiramo ispitivanje modifikacije uz što manje odstupanje od trenutnog načina računanja funkcije cijene. Mogao bi se zadržati trenutni način računanja funkcije cijene utrošene električne energije sve dok korisnik ima uključen samo jedan uređaj čije je vrijeme uključivanja klizno, a povećavanje cijene kada se uključi više ovakvih uređaja istovremeno. Isti pristup se može proširiti uzimajući u obzir ukupan broj uključenih uređaja sa kliznim vremenom uključivanja u svim domaćinstvima.

6. ZAKLJUČAK

Pokazano je da predloženi sistem upravljanja potrošnjom električne energijom donosi benefite, kako pojedinačnim korisnicima, tako i stanju i kvalitetu električne mreže uopšte. Funkcija cijene utrošene električne energije igra veliku ulogu, te se njenom odabiru mora posvetiti posebna pažnja.

LITERATURA

- [1] F. Berman, G. Fox and A. J. G. Hey *Grid Computing: Making the Global Infrastructure a Reality*, 2003 :Wiley
- [2] A. Mohsenian-Rad, V. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game theoretic energy consumption scheduling for the future smart grid," *Smart Grid, IEEE Transactions on*, vol. 1, no. 3, pp. 320 –331, dec. 2010.
- [3] C. W. Gellings and J. H. Chamberlin *Demand Side Management: Concepts and Methods*, 1993 :PennWell Books
- [4] H. Hindi. A tutorial on convex optimization. In *American Control Conference*, Boston, USA, June 2004.

PRIMJENA SAVREMENOG MJERNO-AKVIZICIONOG SISTEMA ZA MONITORING KVALITETA ELEKTRIČNE ENERGIJE NA PRIMJERU ŽELJEZARE NIKŠIĆ APPLICATION OF CONTEMPORARY DATA ACQUISITION SYSTEM FOR POWER QUALITY CONTROL ON EXAMPLE OF IRONWORK PLANT NIKSIC

Mirjana Božović, *Crnogorski elektroprenosni sistem*
Saša Mujović, *Elektrotehnički fakultet*

Sadržaj: Željezara Nikšić je među najvećim potrošačima priključenim na prenosnu mrežu elektroenergetskog sistema Crne Gore. Prisustvo elektrolučne peći, poznate kao potencijalnog uzročnika mrežnog zaprljanja, iniciralo je potrebu za ispitivanjima parametara kvaliteta električne energije. Mjerenje su sprovedena pomoću trofaznog mrežnog analizatora (Dranetz PG4400) i njegovog kompatibilnog softvera Dran-View 6.0. Dobijeni rezultati su upoređeni sa propisanim mrežnim pravilima prenosnog sistema Crne Gore i relevantnim međunarodnim standardima. Snimanja su izvršena u različitim režimima rada elektrolučne peći.

Abstract: Iron work plant "Niksic" is one of the biggest consumers in Montenegrin electrical transmission system. Presence of an electric arc furnace, known as potential causer of bad powerquality, indicates the need for examination parameters related to power quality. Measurements were conducted with three-phase power analyzer (Dranetz PG4400) with its compatible software Dran-View 6.0. The results have been compared with the permitted values defined by the standards. Measurements were conducted during different operating modes.

1. UVOD

Problematika kvalitet električne energije je danas veoma aktuelana, naročito u uslovima razvoja upravljivih mreža (smart grids). Problemi vezani za kvalitet električne energije su se javili tokom sedamdesetih godina prošlog vijeka sa ekspanzijom energetske elektronskih pretvarača. Njihova masovna upotreba i negativan uticaj na naponske prilike u sistemu, kao i potreba za optimalnim kvalitetom istih u cilju njihovog nesmetanog funkcionisanja, pitanju kvaliteta električne energije daju na posebnom značaju [1]. Elektrolučne peći su dodatna opasnost za sistem i one su predmet interesovanja ovog rada, sa osvrtom na primjenu savremenog mjerno akvizicionog sistema.

2. TRENUTNA ISTRAŽIVANJA KVALITETA ELEKTRIČNE ENERGIJE U CRNOJ GORI

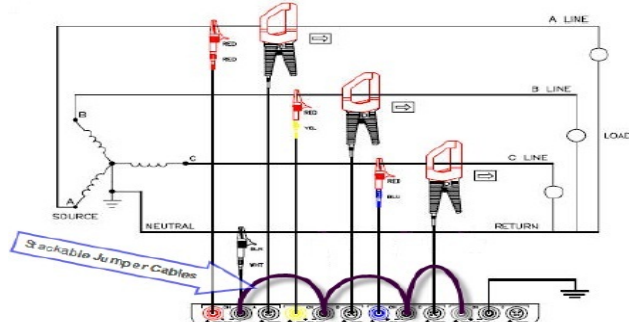
Stanje kvaliteta električne energije u Crnoj Gori nije dovoljno istraženo. Pravila za rad prenosnog sistema Crne Gore propisuju da električni sistemi korisnika prenosne mreže ne smiju imati uticaj na kvalitet električne energije [4]. Javila se potreba za konstantnim monitoringom uticaja veliki potrošača. Usled nedostatka opreme to nije bilo moguće u prošlosti. Prva istraživanja su sprovedena u Željezari.

3. OPIS MJERNO-AKVIZICIONOG SISTEMA

Mjerenja su sprovedena pomoću trofaznog mrežnog analizatora Dranetz Power Guide 4400. Za analizu mjerenja upotrebljen je softver Dran-View 6.0. Dranetz PG 4400 je prenosni mjerni instrument za praćenje kvaliteta napajanja sa osam kanala. Dizajnirani su sa kolor displejom od tečnog

kristala (LCD) ¼ VGA, korišćenjem tehnologije ekrana na dodir. Ovaj uređaj vrši monitoring, snimanje i jednovremeni prikaz podataka na četiri naponska i četiri strujna kanala. Uređaj je dizajniran tako da zadovolji zahtjeve za tačnost i mjerenje dva standarda IEEE 1159 i IEC 61000-4-30 Class A. On može vršiti PQ-optimalnu akviziciju kvaliteta napajanja. Uređaj je dizajniran sa statističkim paketom (QOS), sa monitoringom i protokolima za podešavanje koji su namjenjeni za određivanje saglasnosti mjerenja napona zahtjevanih u standardu EN50160 za monitoring. Evropski standard EN50160 zahtjeva da mjerni parametri moraju biti unutar specificiranog procenta za 95% vremena.

Instrument može snimati stanje prilikom puštanja u rad, može vršiti statističke analize tokom dužeg vremenskog perioda i ispitivanje opreme na terenu radi procjena i održavanja. Njegov kompatibilni softver je Dran-View 6.0. i on omogućava analize izvještaja, štampanje i arhiviranje podataka. Dranetz PG4400 može da radi sa frekvencijom napajanja od 50/60 Hz, i 120/230 AC izvorom napajanja sa ili bez instaliranih baterija. Njegova potrošnja iznosi 20 W.



Slika 3.1 Šema priključivanja mjernog sistema

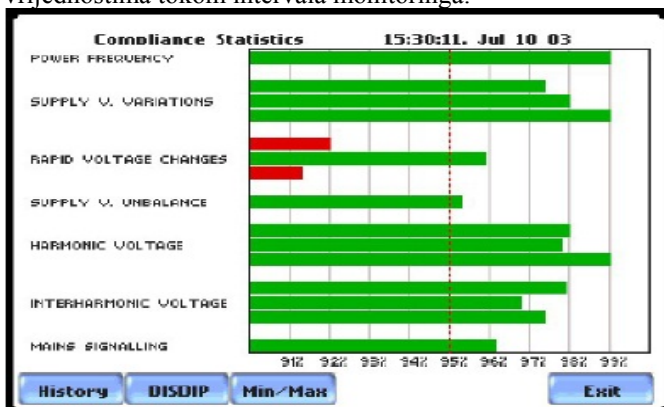
Prilikom startovanja uređaja javlja se početni ekran:



Slika 3.2 Izgled početnog menija

Početni ekran je nudi informacije o statusu monitoringa, koji tip kola je pod monitoringom, statusu memorijske kartice, nazivu fajla sa podacima, trenutnim vrijednostima parametara i prikazu skladištenih podataka pomoću dijagrama trenda, talasnih funkcija i rms dijagrama. Mjerni mod omogućava funkcije pravog rms voltmetra i pravog rms ampermetra. Korisniku se nudi široka paleta različitih opcija za prikazivanja izmjerenih parametara. U slučaju nekarakterističnih pojava, kada vrijednosti veličina koje se prate pređu dozvoljene vrijednosti, uređaj nudi izuzetno detaljan prikaz događaja.

Dranetz PG4400 omogućava i EN50160 tip izvještaja, gdje ukoliko je parametar ispod 95% granice za EN50160, bar graf će biti crvene boje. Neki od parametara imaju detaljnije informacije o sebi koje mogu biti prikazane pritiskom na ime parametra. Postoji i opcija za prikaz istorije mjerenja od prethodnih nedjelja, zajedno sa amplitudom i tabelom raspodjele trajanja (DISDIP), kao i min/max vrijednostima tokom intervala monitoringa:



Slika 3.3 EN50160 tip izvještaja

4. REZULTATI I ANALIZA MJERENJA

Ispitivanje parametara kvaliteta električne energije je vršeno u TS 110/35 kV Nikšić na prvom sistemu sabirnica u trafo polju 110 kV transformatora T3, sa kog se napaja elektrolyčna peć.

Tokom ispitivanja, praćeni su sljedeći parametri:

- potrošnja aktivne i reaktivne snage
- faktor snage
- naponi i njihova nesimetrija

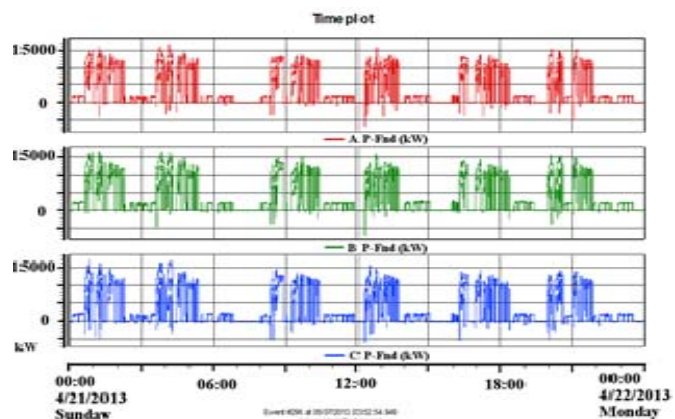
- harmonici
- flikeri

4.1. Potrošnja aktivne i reaktivne snage, faktor snage

Proces rada elektrolyčne peći se može podijeliti na tri faze: topljenje, oksidacija i rafinacija [1].

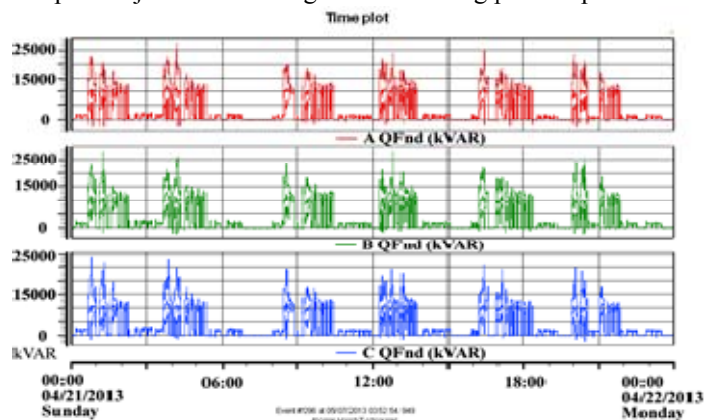
Tokom prve faze rada elektrolyčne peći javlja se visoka vrijednost aktivne snage. Tokom ovog intervalu rad peći je veoma nemiran. Ako je Q_n nominalna reaktivna snaga elektrolyčne peći onda u fazi topljenja dolazi do promjene reaktivne snage u granicama $Q = Q_n (1 \pm 0.7)$ [2]. U drugoj fazi rada peć radi sa 1/3 snage. Promjene opterećenja su mnogo manje nego u prvoj fazi. U trećoj fazi rada nema opterećenja.

Kao karakterističan dan za posmatranje odabran je 21.04.2013., a preostali dan su vrlo slični izabranom danu. Na slici 4.1 je prikazana aktivna snaga kod sve tri faze za taj dan.



Slika 4.1 Aktivna snaga tokom rada peći

Sa Slike 4.1 mogu se uočiti ciklusi rada peći, kao i pomenuti režimi rada. Na slici 4.2 je prikazana reaktivna snaga sve tri faze za odabrani dan. I ovdje se može uočiti značajna potrošnja reaktivne snage tokom prvog režima rada, zatim nešto niža potrošnja tokom drugog režima kao i period bez potrošnje reaktivne snage tokom mirnog perioda peći.



Slika 4.2 Reaktivna snaga tokom rada peći

Proračuni maksimalne/prosječne aktivne/reaktivne snage u jednoj minuti su dati u tabeli 4.1., dok su potrošnja prividne snage i faktor snage prikazani u tabeli 4.2.

Tabela 4.1 Prikaz jednodimenzionalnih vrijednosti zabilježene aktivne i reaktivne snage

Režim	Faze	Max prividna snaga $S_{maxImin}$		Max prosječna prividna snaga $S_{avgImin}$		Faktor snage
I	A	28425	83502	18931	56727	0,71-0,8
	B	29521		19086		0,7-0,78
	C	30684		18710		0,68-0,78
II	A	3377	10472	2905	9163	0,71-0,75
	B	3627		3178		0,74-0,78
	C	3565		3293		0,72-0,73
III	A	84	239	82	235	/
	B	89		88		/
	C	67		66		/

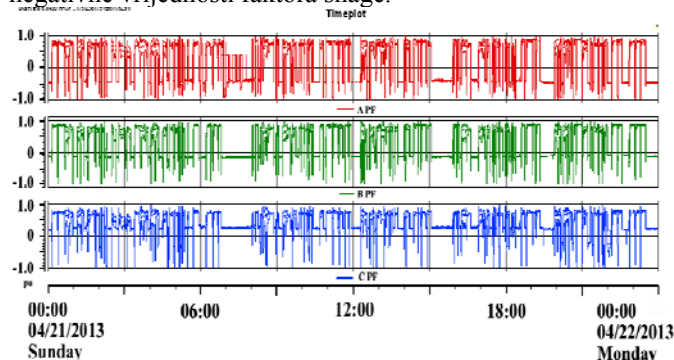
Tabela 4.2 Prikaz jednodimenzionalnih vrijednosti zabilježene prividne snage i faktor snage

Režim	Faze	Max aktivna snaga $P_{maxImin}$	Max prosječna aktivna snaga $P_{avgImin}$	Max reaktivna snaga $Q_{maxImin}$	Max prosječna reaktivna snaga $Q_{avgImin}$
I	A	16130	13746	27499	13274
	B	16558	13883	28171	13182
	C	16716	13401	29055	13019
	Σ	47513	41030	78585	39089
II	A	2161	2008	2801	2227
	B	2428	2219	2963	2123
	C	2378	2035	2996	2589
	Σ	6863	6244	8270	6763
III	A	36	35	/	/
	B	14	13	/	/
	C	20	18	/	/
	Σ	69	65	115	69

Na osnovu tabela 4.1 i 4.2 možemo zaključiti sledeće:

- U režimu rada označenom sa I evidentata je velika potošnja kako aktivne i reaktivne snage, pri čemu reaktivna snaga brže dostiže maksimalnu vrijednost od aktivne.
- U režimu rada označenom sa II postoji uglavnom konstantna potrošnja aktivne snage sve tri faze, što nije slučaj i sa reaktivnom snagom.
- U režimu rada označenom sa III dolazi do velikog smanjenja potrošnje aktivne i reaktivne snage jer je to period mirnog rada peći.

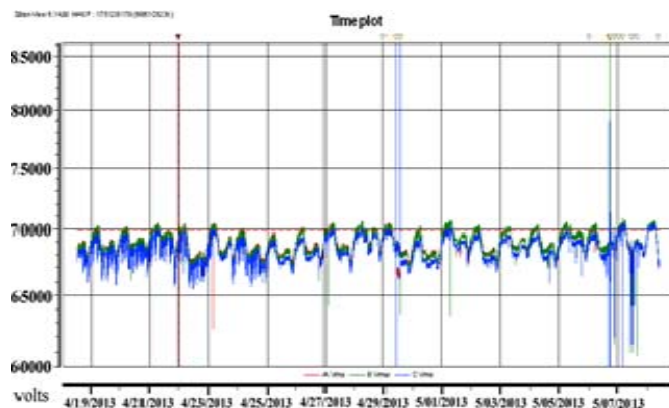
Vrijednost faktora snage u tabeli 4.2 ne uključuje negativne vrijednosti faktora snage.



Slika 4.3 Faktor snage tokom posmatranog dana rada peći

4.2. Naponi

Slika 3.4 prikazuje jednodimenzionalne vrijednosti napona sve tri faze tokom 10 dana praćenja rada peći. Varijacije napona su primjetne tokom prvih dva režima rada peći, dok su pri mirnom pogonu vrlo male.



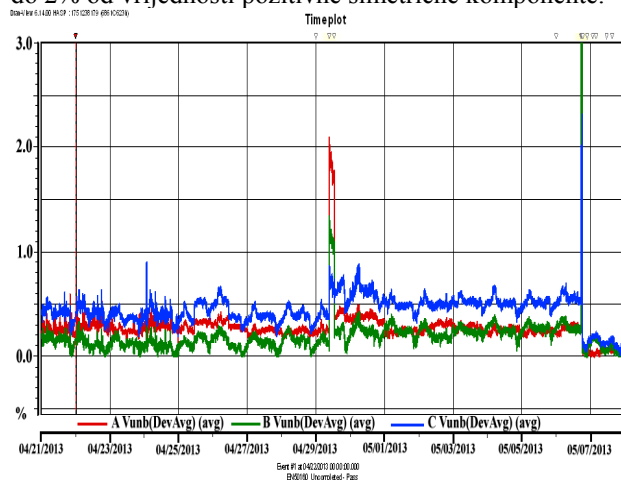
Slika 4.4 – Vrijednosti napona faza tokom perioda praćenja rada peći

Visoke vrijednosti napona tokom mjerenja su nastale kao direktna posljedica položaja i angažovanja HE „Perućica“, što je doprinjelo poboljšanju naponskih prilika. Naponi su u okviru dozvoljenih granica.

Na osnovu slike 4.4 vidi se i događaj od 29. aprila u 12:55, gdje je došlo do pojave prenapona usled atmosferskog pražnjenja. Mjerni uređaji su zabilježili i kvar drugog sistema sabirnica 6. maja, kada je napon faze B iznosio svega oko 35 % U_n . [3]

4.3. Nesimetrije napona

Prema standardu EN 50160 95% vremena u toku jedne nedjelje, 10-to minutna efektivna vrijednost negativne simetrične komponente napona treba da bude u opsegu od 0 do 2% od vrijednosti pozitivne simetrične komponente.

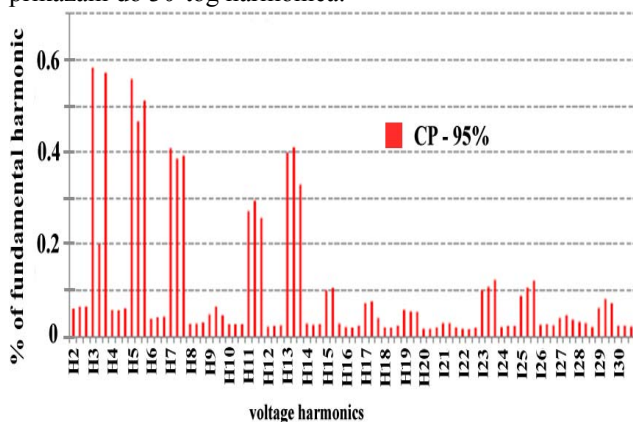


Slika 4.5 – Nesimetrija napona tokom praćenja rada Željezare

Slika 4.5 prikazuje nesimetriju napona sve tri faze i napon faze C tokom ovog perioda ima veću nesimetriju u odnosu na napone faza A i B, obzirom na nejednako opterećenje faza.

4.4 Harmonici

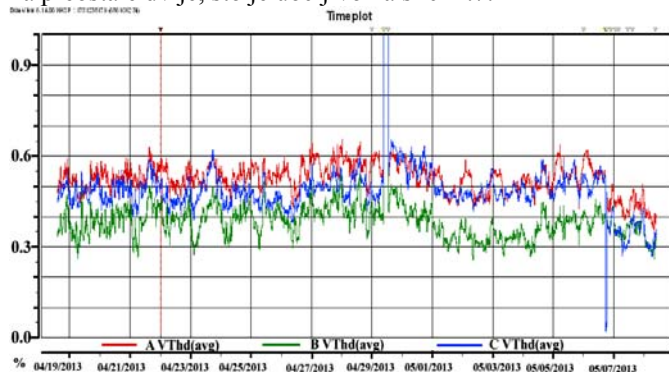
Jedan od parametara koji je vrlo bitan za kvalitet električne energije su i naponski harmonici. Oni su na slici 4.6 prikazani do 30-tog harmonica.



Slika 4.6 Harmonici napona na 110 kV sabirnicama TR 3 u TS 110/35 kV Nikšić [3]

Tokom prelaznih procesa javlja se veća THDv, usled usled atmosferskih pražnjena i manipulacije dalekovodom, kao i u periodima kada je TS Nikšić bila u beznaponskom stanju.

Viši harmonici od 30-tog do 50-tog imaju vrijednosti koje su manje od 0,1 % i ispod su dozvoljenih vrijednosti kod sve tri faze [3]. Faza B ima malo niže vrijednosti THDv u odnosu na preostale dvije, što je uočljivo na slici 4.7.



Slika 4.7 Prosječna vrijednost THDv sve tri faze tokom deset dana

Standard EN50160 koji definiše parametre napona koji su bitni za kvalitet električne energije propisuje da tokom jedne sedmice, 95 % vrijednosti ne smije preći 3 %, što je i ispunjeno u periodu u kom je izvršeno mjerenje.

4.5. Flikeri

Kao mjera djelovanja flikera koriste se dva parametra: indeks nivoa kratkotrajnih flikera Pst i indeks nivoa dugotrajnih flikera Plt. U standardu EN50160 definisana je granica samo za parametar Plt jer se smatra mjerodavnijim za definisanje napona napajanja.

U normalnom radu nivo flikera prouzrokovanih promjenama napona napajanja treba zadovoljiti relaciju

$Plt \leq 1$ i to za 95% vremena ispitivanja koje traje jednu sedmicu. Utvrđeno je da indeksi jačine flikera rastu uporedo sa operativnim ciklusom peći i oni su iznad dozvoljenih vrijednosti. Vrijednosti Pst i Plt faktora su u dozvoljenim granicama, iako su trenutne vrijednosti ovih faktora u 95 % slučajeva iznad dozvoljenih vrijednosti [3].

5. ZAKLJUČAK

Mjerenjima u Željezari je konstatovan neadekvatan kvalitet električne energije i posebno izraženi sljedeći problemi:

- potrošnja reaktivne snage
- flikeri
- propadi napona i tranzijenti
- faktor snage
- harmonici
- nesimetrija napona

Mjerenja su sprovedena pomoću trofaznog mrežnog analizatora Dranetz Power Guide 4400. Ovaj uređaj nudi veliki broj opcija za praćenje parametara koji su od značaja za kvalitet električne energije. Za analizu mjerenja koristi se softver Dran-View 6.0. Softver omogućava vrlo opsežne analize mjerenja i njihovo automatsko upoređivanje sa internacionalnim standardima.

Na osnovu ispitivanja potvrđena je opravdanost ugradnje filtera, čime bi se nivo flikera sveo na dozvoljene vrijednosti. To je ujedno i najbrži način za rješavanje ovog problema. Na pojavu flikera je moguće uticati i povećanjem snage kratkog spoja što podrazumijeva značajne investicione troškove u trafostanici. Neophodno je i postavljanje uređaja za kompenzaciju, koji će uticati na faktor snage. Pojava tranzijentnih prenapona može dovesti do oštećenja osjetljive opreme, pa je poželjna i ugradnja odvodnika prenapona na izvodima 110 kV dalekovoda. Dodatna mjerenja su planirana tokom remonta HE Perućica na drugom sistemu sabirnica.

Sve mjere koje je potrebno preduzeti podrazumijevaju i duže vremensko praćenje uticaja ovog potrošača na kvalitet električne energije i naponske prilike.

LITERATURA

[1] V. Katić, "Kvalitet električne energije – viši harmonici", Novi Sad, 2002.

[2] www.etf.unsa.ba

[3] Elektroenergetski koordinacioni centar, Crnogorski elektroprenosni sistem, „Uticaj rada elektrolučne peći u „Željezari“ Nikšić na kvalitet električne energije - Preliminarno ispitivanje“, maj 2013.

[4] Crnogorski elektroprenosni sistem, „Pravila za funkcionisanje prenosnog sistema električne energije“, oktobar 2011.

[5] Dran-View 6.0, „User’s guide“, DRANETZ – BMI

IMPLEMENTACIJA I VIZUELIZACIJA GRUPNE REGULACIJE U HE PERUĆICA IMPLEMENTATION AND VISUALITION OF JOINT CONTROL IN HPP PERUĆICA

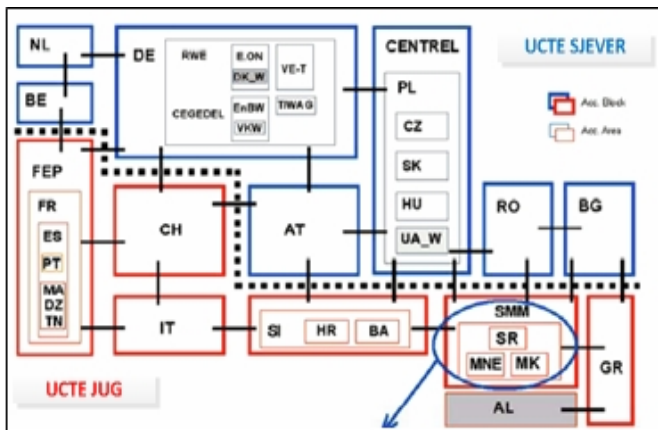
Ana Grbović, Bojan Đordan, HE Perućica, EPCG

Sadržaj: U ovom radu prikazana je implementacija i vizuelizacija Grupne regulacije aktivne i reaktivne snage u HE Perućica. Detaljno je opisan razvoj Grupne regulacije preko WinCC funkcija, pri čemu su složeniji zahtjevi za vizuelizacijom riješeni preko ANSI-C programskog jezika. Takođe, pojašnjena je realizacija sekundarne regulacije frekvencije kao jedne od funkcija grupne regulacije.

Abstract: This paper describes the implementation and visualization of Joint control of active and reactive power in HPP Perućica. The paper describes the development of Joint control over WinCC functions, where are a more complex requirements for visualization resolved through the ANSI-C programming language. It also clarified the implementation of Secondary frequency regulation as a function of the Joint control.

1. UVOD

Najveća elektroenergetska interkonekcija u Evropi je UCTE (Union for Coordination of the Transmission of Electricity). Na slici 1 prikazana je struktura i organizacija kontrolnih blokova i regulacionih oblasti u UCTE interkonekciji, sa koje se vidi i položaj SMM (Serbia, Macedonia, Montenegro) bloka u okviru ove interkonekcije.



Slika 1 – Položaj SMM bloka unutar UCTE interkonekcije

U elektroenergetskoj interkonekciji (EI) koja u cjelini radi u sinhronom paralelnom radu, vrijednost učestanosti predstavlja jedan od bitnih kriterijuma za ocjenu kvaliteta rada tretirane interkonekcije. Osnovni cilj regulacije učestanosti i snage razmjene u EI je održavanje ravnoteže između proizvodnje i potrošnje električne energije. To se postiže promjenom proizvedene snage u cilju kompenzacije neizbežnih promjena u potrošnji, [1], [2].

Elektroprivreda Crne Gore (EPCG), Crnogorski elektroprenosni sistem (CGES) i Crnogorski operator tržišta električne energije (COTEE) su po prvi put u 2013. godini zaključili Ugovor o pružanju sistemskih usluga (predviđen zakonom o energetici) kojim su, između ostalog, obezbjeđeni kapaciteti i energija za potrebe: primarne, sekundarne i

tercijalne regulacije učestanosti, primarne regulacije napona i uspostavljanja sistema nakon raspada.

U 2013. godini završena je implementacija Grupne regulacije za hidroelektranu Perućica. Nakon prihvatanja projekta od strane investitora, kompanija Siemens je u saradnji sa Službom za modernizaciju elektrana i Službom za nadgledanje sistema upravljanja razvila veoma složen softver koji upravlja sa aktivnim i reaktivnim snagama sedam agregata i naponima sabirnica 220kV. U sljedećim poglavljima opisana je realizacija Grupne regulacije na PCS7 i WinCC serverima u HE Perućica.

2. GRUPNA REGULACIJA (GR)

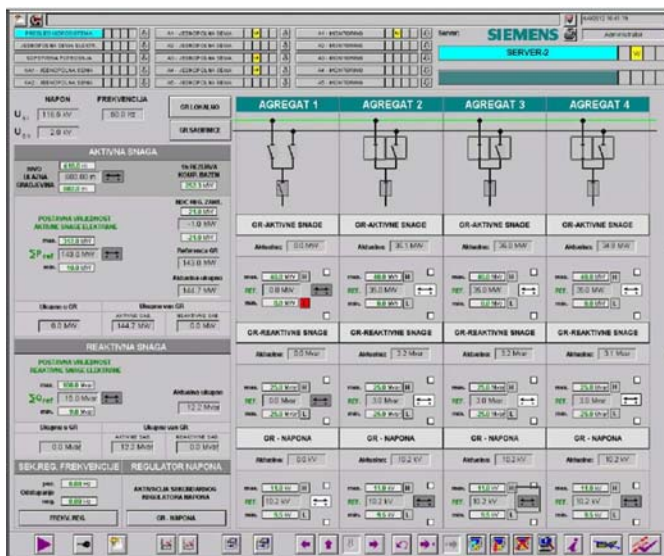
Razvoj GR sa novim funkcijama koje su usklađene sa zahtevima savremenog tržišta električnom energijom zahtjevao je razvoj i realizaciju prema potrebama i podešavanjima u samoj elektrani.

Grupni regulator (GR) je upravljački sistem na nivou elektrane koji operateru pruža mogućnost da u procesu upravljanja opterećenjem cijelu elektranu tretira kao jedan agregat. Naime, pri grupnoj regulaciji operater zadaje samo aktivno i reaktivno opterećenje elektrane i Grupni Regulator na osnovu tih referentnih vrijednosti, kao i informacija o stanju i raspoloživosti svih agregata vrši proračun i podešavanje referentnih vrijednosti opterećenja za svaki agregat pojedinačno. Pri raspodjeli opterećenja regulator vodi računa o ravnomjernom rasporedu opterećenja, ali uzima u obzir i ograničenja, nepovoljne režime rada, kao i pogonsku kartu generatora. Funkcija grupne regulacije je implementirana na redundantnim PCS7 i WINCC serverima. Grupni regulator obavlja četiri međusobno spregnute funkcije, i to:

- Grupna regulacija aktivne snage (GRAS)
- Grupna regulacija reaktivne snage (GRRS)
- Sekundarna regulacija frekvencije (SRF)
- Sekundarna regulacija napona (SNR)

2.1 Grafički prikaz

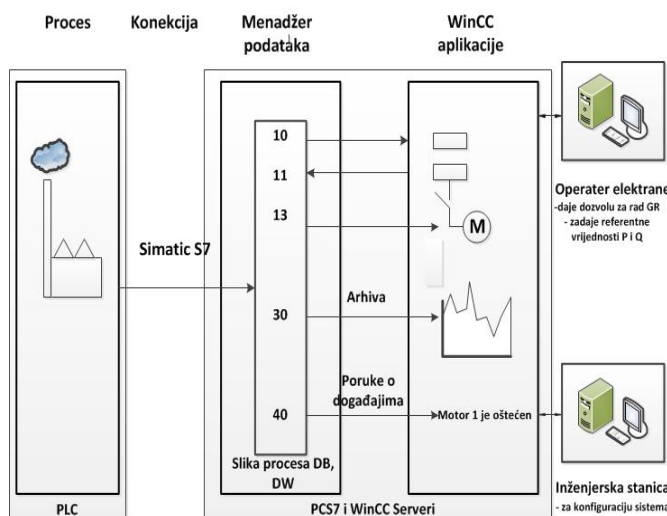
Komunikacija operater-regulator se vrši preko prikaza implementiranih na operatorskim stanicama u komandi.



Slika 2 – Glavni prikaz GR

Na glavnom prikazu operater unosi referentne vrijednosti opterećenja, vrši izbor aktivnog sistema sabirnica, kao i uključivanje/isključivanje prva četiri agregata iz grupe regulacije. Osim upravljanja, na ovom prikazu je moguće pratiti i aktivne i reaktivne snage prva četiri agregata, njihove referentne vrijednosti, kao i referentne vrijednosti i trenutno opterećenje za cijelu elektranu. Na sporednom prikazu moguće je pratiti i aktivne i reaktivne snage za 5,6 i 7 agregat, kao i pregled planirane i ostvarene proizvodnje za tekući sat i prethodni sat. Jedan od načina za postizanje visokih performansi na slikama i u komunikaciji je podešenost vremena osvježavanja koje iznosi 2s.

3. SLIKA PROCESA



Slika 3 – Slika procesa

Grupni regulator aktivne i reaktivne snage fizički se sastoji od (slika 3) PLC (Programabilnih logičkih

kontrolera), PCS7 i WinCC Servera (realizovanih kao redundantni sistem), inženjerske stanice (ES) i daljinskih operatorskih terminala. Sve funkcije GR su implementirane na redundantnom paru servera i PLC-ovima. Serveri su komunikacionim kanalom povezani sa daljinskim terminalom. Na daljinskom terminalu dispečer elektrane izvršava zadavanje svih veličina, parametara, ograničenja vezanih za grupnu regulaciju. Promjena ovih parametara biće omogućena i na ES (inženjerskoj stanici).

3.1 Konekcija WinCC-a sa procesom

WinCC "Data manager" vodi brigu o WinCC tagovima tokom "Runtime"-a, softvera koji služi za vizualizaciju procesa. WinCC aplikacije zahtjevaju tagove od "Data manager"-a koji ima zadatak preuzimanja vrijednosti tih tagova od procesa putem komunikacionih drajvera integrisanih u WinCC projekat. Komunikacioni drajver predstavlja interfejs između WinCC-a i procesa korišćenjem jedne od komunikacionih jedinica. Hardverska konekcija sa procesom se obično ostvaruje korišćenjem komunikacionog procesora. WinCC komunikacioni drajver tada šalje zahtjeve PLC-u putem komunikacionog procesora. PLC šalje zahtjevanu procesnu vrijednost nazad prema WinCC-u.

"Data manager" vrši administraciju podataka WinCC-a i ovo nije vidljivo korisniku. "Data manager" radi sa podacima kreiranim u WinCC projektu i smještenim u bazu podataka projekta. On vodi brigu o svim WinCC tagovima u toku "Runtime"-a. WinCC aplikacije zahtjevaju podatke od "Data manager"-a u vidu WinCC tagova. Ove aplikacije su "Graphics Runtime"-a, "Alarm Logging", "Runtime", "Tag Logging Runtime", [3].

3.2 Eksterni (procesni), interni i sistemski tagovi

Procesni tagovi uspostavljaju vezu između procesa i HMI stanice. Oni se podešavaju nakon podešavanja PLC konekcije. WinCC dozvoljava odvajanje tagova u tehnološke grupe. Grupa tagova predstavlja strukturu koja je jednostavna za čitanje. Prije početka definisanja tagova ("Novi tag") potrebno je kreirati novu grupu ("Nova grupa"). Nazivi tagova koji počinju sa znakom @ rezervisani su samo za sistemske tagove dok oni koji počinju sa znakom \$ nijesu vidljivi u "Tag Managment"-u.

Interni tagovi se koriste za čuvanje nekih vrijednosti u okviru WinCC-a i nemaju nikakve veze sa procesom.

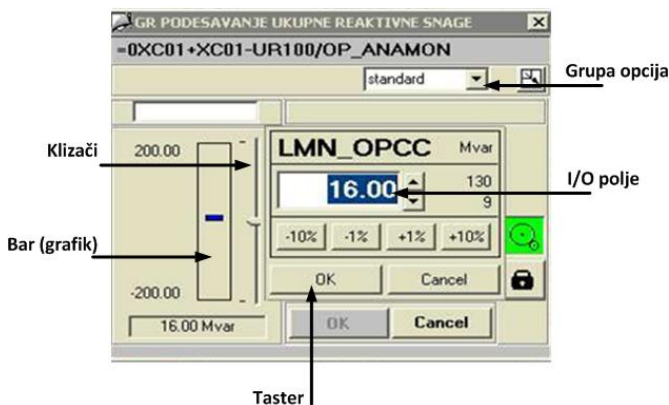
Sistemske tagovi se automatski generišu i koriste za pojedine funkcije WinCC-a. Sistemske, kao i procesne tagovi, sumiraju se i ulaze u licencu koja se određuje prema broju tagova.

4. GRAFIČKI EDITOR

Grafički editor, kao jedna od osnovnih funkcija WinCC-a, koristi se za kreiranje neophodnih ekrana – slika (statičkih i dinamičkih komponenti) u skladu sa definisanim zadatkom. Slike se sastoje od objekata koji mogu biti "Standardni", "Smart" i "Windows" objekti.

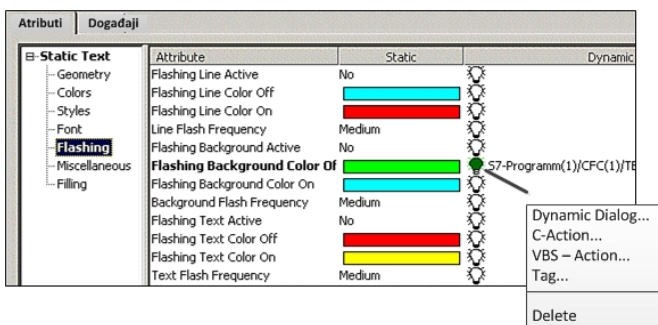
- Standardni objekti se sastoje od **grafičkih objekata i statičkog teksta**.

- Smart objekti se koriste za implementaciju složenijih elemenata, npr. **I/O polje** se koristi kao ulazno, izlazno ili kombinovano polje. Pomoću njega moguće je prikaz binarnih, heksadecimalnih, decimalnih ili stringovnih tipova podataka. Takođe, moguće je zadavanje graničnih veličina, skriveni unos ili transfer nakon završenog unosa. Drugi tip smart objekta je **Bar (grafik)**, koji prikazuje procesnu veličinu u analognoj formi. Polje prikazuje trenutnu vrijednost.
- Windows objekti mogu biti **Tasteri** koji služe za izdavanje procesnih komandi, **Opcione grupe**, kojima biramo jednu od operacija koja može biti izvršena, **Klizači**, za podešavanje analognih procesnih veličina itd. Neki od navedenih objekata dati su na slici 4.



Slika 4 – Primjer implemenatacije različitih vrsta objekata

Karakteristike objekta mogu biti “Atributi” i “Događaji” i oni se konfigurišu različitim alatima. Tip i broj karakteristika objekta zavisi od izabranog objekta. Na slici 5 dat je primjer dinamizacije jedne od karakteristika atributa. Sve dok se ne izvrši dinamizacija i ne konfiguriše određeni triger u dinamičkoj koloni uvažava se podešavanje iz statičke kolone, [3].



Slika 5 – Atributi kao karakteristike objekta

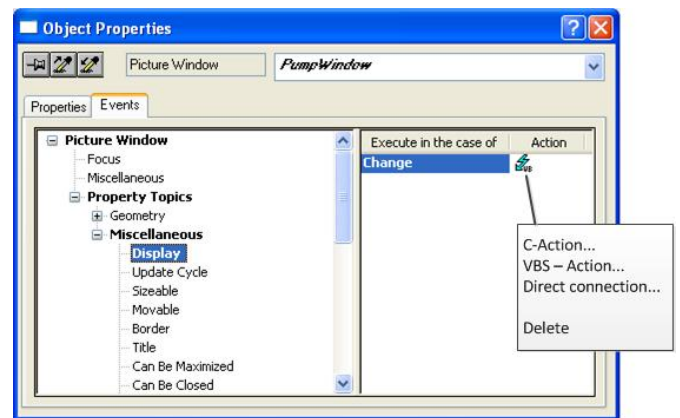
Moguće je izvršiti dinamizaciju na 9 načina. Jedni se koriste za attribute, drugi za događaje. Kao primjer statistike količine dinamizacija objekata, navešćemo glavnu procesnu sliku GR1 koja sadrži 396 dinamizacija za sve objekte sa slike.

“Dynamic dialog” nam omogućava da na “user-friendly” način učinimo naše objekte dinamičkim. Na primjer, veoma je jednostavno iskonfigurirati promjenu boje objekta na slici

pri dostizanju definisanih limita. Grafički simbol crvene munje ⚡ označava upotrebu ovog dijaloga.

“C-Action” nam omogućava dinamizaciju grafičkih ekrana preko akcija u C kodu. Mnoge karakteristike se mogu učiniti dinamičkim, kao što je boja, veličina objekta, pozicija, veličina fonta, koristeći interne ili procesne tagove. Koriste se kada se trigerovanje akcije vrši preko tajmera ili nakon promjene taga. Takođe, možemo koristiti i napredne “VBS action”. Skripte se koriste kada ne želimo da opteretimo PLC, tj. kada iščitamo vrijednost taga, odradimo određenu operaciju nad njom i tagu vratimo novu vrijednost. C skripta vraća vrijednost kada je pisana za attribute, dok se za događaje ne vraća. Simbol zelene munje ⚡ označava upotrebu C skripte, dok simbol svijetlo plave munje ⚡ označava upotrebu VB skripte. Ako C skripta nije iskompajlirana simbol munje je žute boje ⚡. Uz standardne C funkcije (ANSI C), postoje i WinCC specifične funkcije za čitanje i pisanje tagova i procesuiranje svih mogućih objekata.

“Tag Connection” se koristi za direktno povezivanje sa internim i i procesnim tagovima. Simbol zelene lampice 💡 u koloni za dinamizaciju označava upotrebu ove akcije.



Slika 6 – Događaji kao karakteristike objekta

Na slici 6 prikazane su mogućnosti konfiguracije nad događajima.

“C-Action” i “VBS action” se kao napredne akcije izvršavaju kada ih trigeruje neki događaj. U ovom slučaju nije potrebno definisati triger, pošto je to sam događaj, npr. promjena neke karakteristike objekta ili klik mišem.

“Direct connection” je specijalan tip akcije, ima simbol tamno plave munje ⚡. Zbog visokih performansi u “Runtime”-u koristi se kad god je moguće i pomoću njega, npr., možemo veoma brzo prenijeti procesne veličine na tag.

5. SEKUNDARNA REGULACIJA FREKVENCIJE

Dok primarna regulacija učestanosti predstavlja spontano dejstvo regulatora pogonskih mašina sinhronih generatora, koji su osjetljivi na promene učestanosti, sekundarna regulacija predstavlja naknadno, superponirano dejstvo na primarnu regulaciju, preko ulaza za tzv. spoljnu naredbu turbinskih regulatora regulacionih agregata, radi eliminisanja statičke greške učestanosti i/ili snage razmjene, [4].

Sekundarna regulacija frekvencije se kao funkcija GR sa strane elektrane aktivira kroz dijalog koji se poziva klikom na dugme FREK. REG. u okviru Prikaza za grupnu regulaciju kada smo u lokalnom modu. SRF, kao funkcija GR, ima za cilj da održava frekvenciju mreže djelujući kroz referentne vrijednosti aktivne snage agregata koji su uključeni u GRAS. Pri aktivaciji ove funkcije, operater postavlja referentnu vrijednost frekvencije, a takođe i definiše minimalno i maksimalno odstupanje u kom djeluje SRF. Naime, GR u slučaju kad je aktivna SRF, uvećava ili smanjuje referentnu vrijednost aktivne snage agregata uključenih u GRAS do 5% od trenutne aktivne snage u srazmjeri sa udjelom trenutnog odstupanja frekvencije u odnosu na maksimalno odstupanje definisano za SRF. Ukoliko je u određenom vremenskom periodu stvarno odstupanje veće od definisanih limita, SFR se automatski isključuje sve dok se frekvencija mreže ponovo ne vrati u definisani opseg.

Za razliku od ostalih funkcija GR, u slučaju sekundarne regulacije frekvencije ne postoje posebni uslovi za uključenje jer ona djeluje kroz GRAS a ne na agregate pojedinačno.

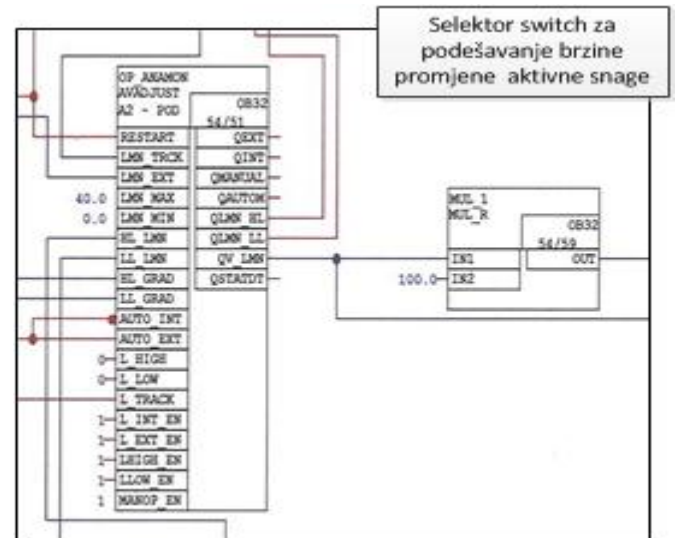


Slika 7 – Prikaz izbora moda upravljanja i GRAS

Dio prikaza koji se odnosi na GRAS služi za definisanje referentnih vrijednosti za aktivnu snagu elektrane, kao i za pregled raspodjele aktivnog opterećenja između agregata. Takođe je sa ovog prikaza moguće pratiti trenutni nivo vode na ulaznoj građevini kao i informaciju o maksimalnoj aktivnoj snazi elektrane za koju postoji jednočasovna rezerva vode u kompenzacionom bazenu. Ograničenja koja potiču od nivoa vode na ulaznoj građevini odnosno rezerve u kompenzacionom bazenu imaju uticaj samo u slučaju da je za mjesto upravljanja GR izabran DALJINSKI mod – odnosno kada elektrana radi za potrebe sekundarne regulacije. U LOKALNOM modu ovaj tip ograničenja je samo informativnog karaktera, a operateru je prepušteno da vodi računa o opterećenju i ograničenjima elektrane.

Daljinsko upravljanje sekundarnom regulacijom se aktivira klikom na dugme SEKUNDARNA REG. Naime, NDC šalje regulacioni zahtjev, tj. tačnu grešku sistema, koja

se pokušava “ispeglati” sa prva četiri agregata. Greška se prosljeđuje do PLC-a, tj. u određeni DB, sa određenom referencom i brzinom promjene. Brzinu promjene podešavamo na selector switch-u CFC (Continuous function chart), koji je prikazan na slici 8. Ona iznosi 100 kW/s kada smo u GR, odnosno 400 kW/s kada smo van GR. Sada PLC šalje referencu ka turbinskom regulatoru, koju turbinski regulator prati po svojoj krivoj.



Slika 8 – CFC za podešavanje brzine promjene aktivne snage

6. ZAKLJUČAK

U radu je izložena analiza realizacije grupne regulacije u HE Perućica, pri čemu su sve funkcije realizovane u skladu sa važećom regulativom i zahtjevima tržišta. Predstavljene su različite mogućnosti dinamizacije procesih slika u WinCC-u, kao i njihovi grafički prikazi. Iako postoje povremeni kratkotrajni prekidu u radu aplikacije, ovi problemi se sistemski rješavaju, tako da se može reći da je pouzdanost rada GR na zadovoljavajućem nivou, a samim tim i kvalitet “peglanja” regulacione greške.

LITERATURA

- [1] Milan Ivanović, Dragan P. Popović, Saša Minić, "Primarna, sekundarna i tercijalna regulacija učestanosti u analizama dinamičke sigurnosti elektroenergetskih interkonekcija", Elektrotehnički institut Nikola Tesla, Univerzitet u Beogradu, pp. 93-115
- [2] "UCTE Operation Handbook, Policy 1: Load-Frequency Control and Performance; A. Primary Control", Final Version (approved by SC on 19 March 2009)
- [3] Siemens, SIMATIC HMI WinCC V6 Communication Manual, 2004
- [4] "UCTE Operation Handbook, Policy 1: Load-Frequency Control and Performance; B. Secondary Control", Final Version (approved by SC on 19 March 2009)

JEFTINO PIC BAZIRANO REŠENJE ZA AUTO-TRAKING FOTONAPONSKIH PANELA AFFORDABLE PIC BASED SOLUTION FOR PHOTOVOLTAIC PANEL AUTO-TRACKING

Roman Golubovski, Faculty of Electrical Engineering, University "Goce Delčev" - Štip,
Republic of Macedonia

Sadržaj: Fotonaponska konverzija je među najbitnijim aplikacijama obnovljivih izvora zbog direktne konverzije solarne energije u električnu. Iako su fotovoltaići skuplji od ostalih tipova energetske konvertora, oni imaju prednosti kao mala težina, i skalabilnost i za manje aplikacije. Zato je bitno da se postiže veća efikasnost maksimirajući absorpciju solarnog zračenja. Sama absorpcija je nekompletna uglavnom zbog napadnog ugla radijacije na površinu PV panela (sunčeva putanja tokom dana i razlike u sezonskim trajektorijama), kao i zbog refleksije istih. Jedan od načina podizanja efikasnosti je optimizacija napadnog ugla kontinuiranim trackingom. Ovaj trud predlaže jeftinu PIC realizaciju trackinga sa četiri LDR (light dependent resistor) senzora za planarnu prostornu orijetaciju sa dva stepena slobode i steperskim pogonom.

Abstract: Photovoltaic (PV) conversion is one of the most important applications of renewable energy sources because of the direct conversion of solar energy to electric energy. Although PVs are more expensive compared to other types of energy converters, they have some advantages such as low weight and feasibility of small scales. Therefore, it is important to absorb the maximum solar energy in order to increase the efficiency of the energy converter. PV's absorption of the solar energy is incomplete mainly due to angle of incidence (resulting from both the daily sun motion and the seasonal trajectory differences) as well as the partial loss due to reflection off PV's surface. One way to increase efficiency is to optimize the angle of incidence by employing continuous solar tracking. This paper proposes a cheap PIC based embedded design calculating next optimal spatial orientation by using four LDR (light dependent resistor) sensors, and tracking that optimal orientation with a two-degrees of freedom stepper drive.

1. INTRODUCTION

Solar energy is a type of renewable energy source that is relatively easy to convert directly into electrical energy by the use of photovoltaic (PV) converters [1], [2]. Solar cells convert the energy of light's photons to electric energy with efficiency between 5% to 25% without using thermodynamic cycle or active fluid. Solar cells can be light collector directly or can use light concentrators like mirror or convex lens.

The PV has advantages like light weight suitable for transport and installation, silent operation, little maintenance, etc. However, it is still more expensive than most of the rest energy converters like diesel generators. So efficiency in terms of optimal operation with maximized solar absorption is essential. Beside the high costs, another problem is the low efficiency of the solar cells themselves (14%-17% for monocrystalline silicon, 13%-15% for polycrystalline silicon and 5%-7% for amorphous silicon). Therefore an optimal PV conversion efficiency requires optimal angle of incidence for as long as possible during daylight radiation.

This paper depicts an affordable (PIC based) concept with four cheap LDR sensors that can accurately and timely follow the sun providing best solar absorption. The algorithm can adjust the daily rate of corrections taking into account the cost of driving the moves against the efficiency loss due to incidence angle's deviation from its optimum. The analysis

behind the algorithm is a typical set of power-voltage characteristics recorded for movable and fixed (reference) PV arrays during a sunny summer day.

2. PV OUTPUT POWER DEPENDENCE OF THE SOLAR RADIATION INTENSITY AND ANGLE

Typical current-voltage-power characteristic of a solar cell is given on figure 1:

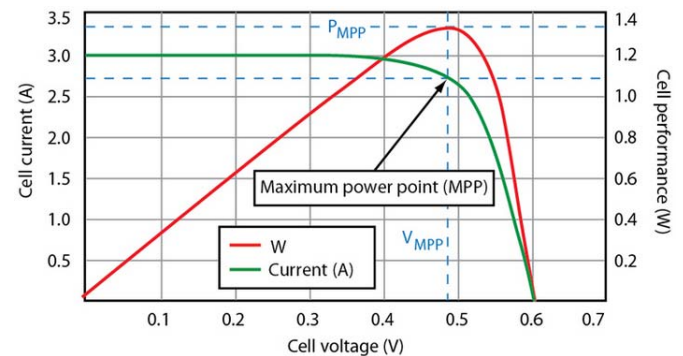


Figure 1. I-V-P characteristics of a PV

Figure 2 shows how generated current varies with light intensity changes thus resulting in output power changes, which can go below its nominal value.

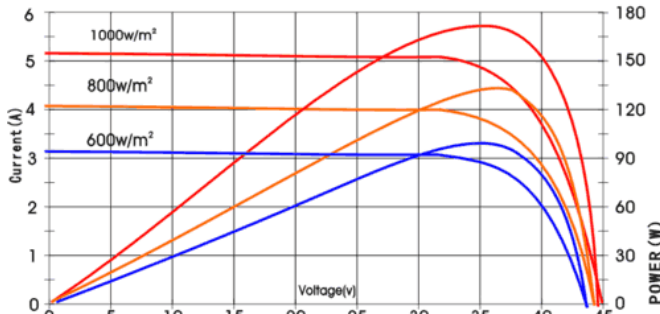


Figure 2. Current variations due to radiation intensity

Daily motion of the sun changes the radiation angle constantly, so the required perpendicular angle of incidence can be achieved only for movable PV arrays. Seasonal trajectory changes also imply need for angle adjustment so 2D planar freedom is required, as depicted on figure3:

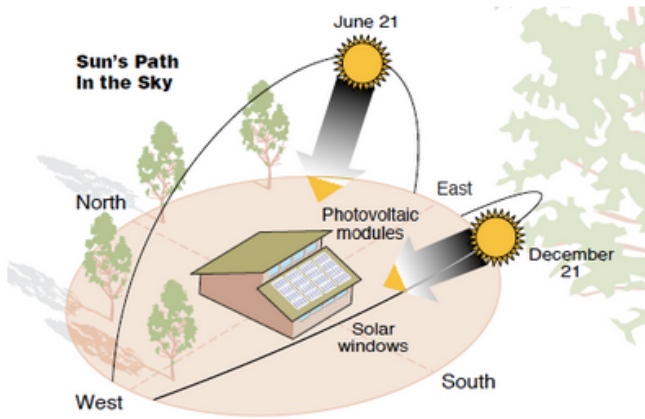


Figure 3. Variation of sun motion path during the year

The angle affects not only the direct absorption of radiation photons by the PV cells, but also the PV panel surface reflection which increases proportionally, thus lowering the effective absorption.

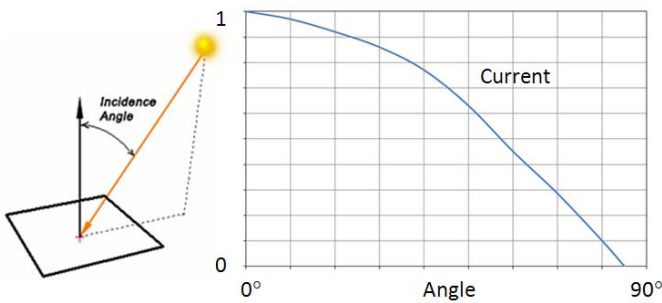


Figure 4. Radiation angle and its effect on the current

3. COMPARISON OF EFFICIENCY BETWEEN MOVABLE AND FIXED PV

The presented output power characteristics on figure 5 are typical and recorded on two identical PV panels - one being fixed in its optimal orientation, the other being movable and following the sun in two degrees of freedom (azimuth and latitude) [3]. The movable PV tracker locks on the MPP (maximum power point).

The comparative study is made on a sunny July day in the north hemisphere, hourly, from 0900h (AM) to 1800h (PM).

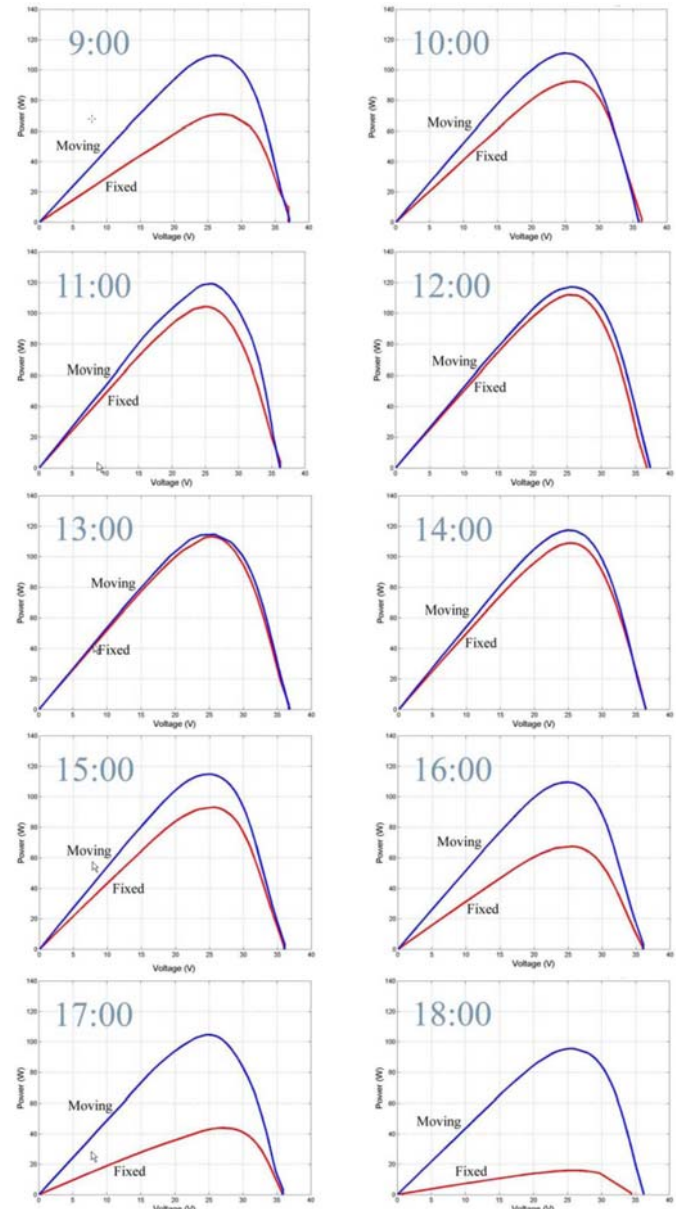


Figure 5. Power characteristics of fixed and moving PVs

At 0900h the movable PV (MPV) is 55.2% more efficient than the fixed (FPV). At 1000h FPV's output power raises significantly compared to MPV's which changes a little. Still MPV is 19.5% more efficient. From 1100h to 1400h FPV's output power approaches the MPV's, as their orientation matches. MPV is still more efficient (4% to 13%). After 1400h sun height starts decreasing and so is the FPV's output power. MPV also decreases its output power but slower as it still tracks best orientation. MPV's efficiency compared to FPV's raises from 7.5% to 61.4% at 1600h. As sunset approaches both MPV and FPM decline in their output power, but MPV is much more efficient - 236% at 1700h and 605% at 1800h.

This comparative analysis shows that a tracker can improve daily energy production of a fixed PV array significantly.

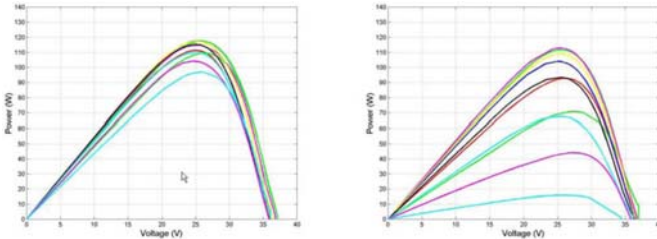


Figure 6. All-day curves for moving and fixed PVs

Figure 6 shows the power curves of both MPV (left graph) and the FPV (right graph). Obviously, during the day the moving system has less variations in the generated power than the fixed. Therefore, it can be concluded that in terms of electric power supply, by using the movable photovoltaic array the power quality will be higher and by using the fixed array, the large power variation should be dumped due to a larger energy storage unit.

4. EFFICIENT AUTO-TRACKING WITH TWO DEGREES OF FREEDOM

A PV tracker is consisted of three components:

- sensor
- controller
- actuator drive

As stated above, its goal is to lock on the sun maintaining minimal angle of incidence (hopefully 0°) by driving panel's actuators accordingly. Objective problem are the latitude (effecting the light radiation angle) and the average count of sunny days around the year. A significant parameter is actuators drive's own power consumption against the generated power, effecting the orientation update rate. Of course, fundamental for bringing the decision of automating a PV array with a tracker is the economic viability.

The proposed prototype tracker is biaxial (polar coordinates - azimuth and latitude) and it drives two step motors. The sensor is almost costless and made of four LDRs (light dependent resistor) which need to be equally calibrated for proper operation. They are planar and divided from each other with opaque plates for the purpose of highest accuracy in proper orientation (figure 7).



Figure 7. The 4x LDR sensor and a single LDR

The controller (figure 10) is PIC based, also very cheap. It acquires the four LDR inputs and generates command signals for the steppers drivers directly. For larger scaled applications different drivers (according to the actuators) can be used with the same controller.

The algorithm in the controller uses the sensors to calculate the adjustments of both axes for the next optimal position. Figure 8 shows the upper view of the sensor and figure 9 shows a side view of sensors LDR#1 (upper left) and LDR#2 (upper right):

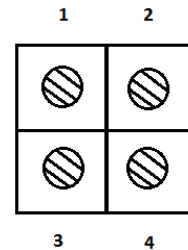


Figure 8. Upper view of the sensor

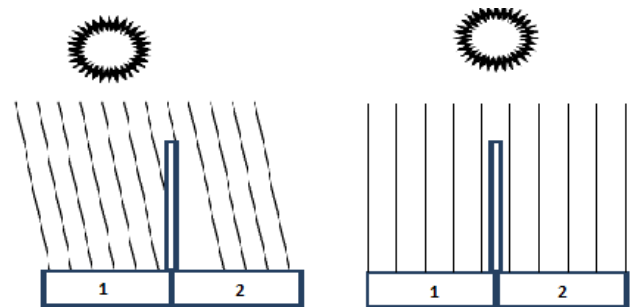


Figure 9. Side view of upper left-right LDRs

The high walls between the LDRs assure sensitivity to minimal differential excitation which allows for precise calculation of sun's azimuth and altitude above the horizon.

The azimuth is detected from the more excited pair of horizontal LDRs (the upper 1-2 or the lower 3-4). The principle is illustrated on figure 9. The microprocessor (PIC) acquires simultaneously all four LDRs and determines which one is most excited. Than it decides which horizontal pair to use for the azimuth correction. Figure 9 depicts the situation where LDR#1 is most excited, therefore horizontal pair 1-2 is used. The controller will then move horizontally in direction of LDR#1 until both LDR#1 and LDR#2 become equally excited, meaning the pair 1-2 is facing the sun perpendicularly.

The altitude is detected in the same manner, by determining the more excited vertical pair (the left 1-3 or the right 2-4). Once it is determined, the controller drives the sensor vertically (simultaneously with the horizontal movement) in the direction of the more excited LDR until both LDRs are equally excited.

Testing with ordinary lamps under daylight show that LDRs appear to provide a very stable (noiseless) signal

which allows for smooth, fast and accurate response to small illumination changes.

Prototypes' schematics given in figure 10 is simple and affordable.

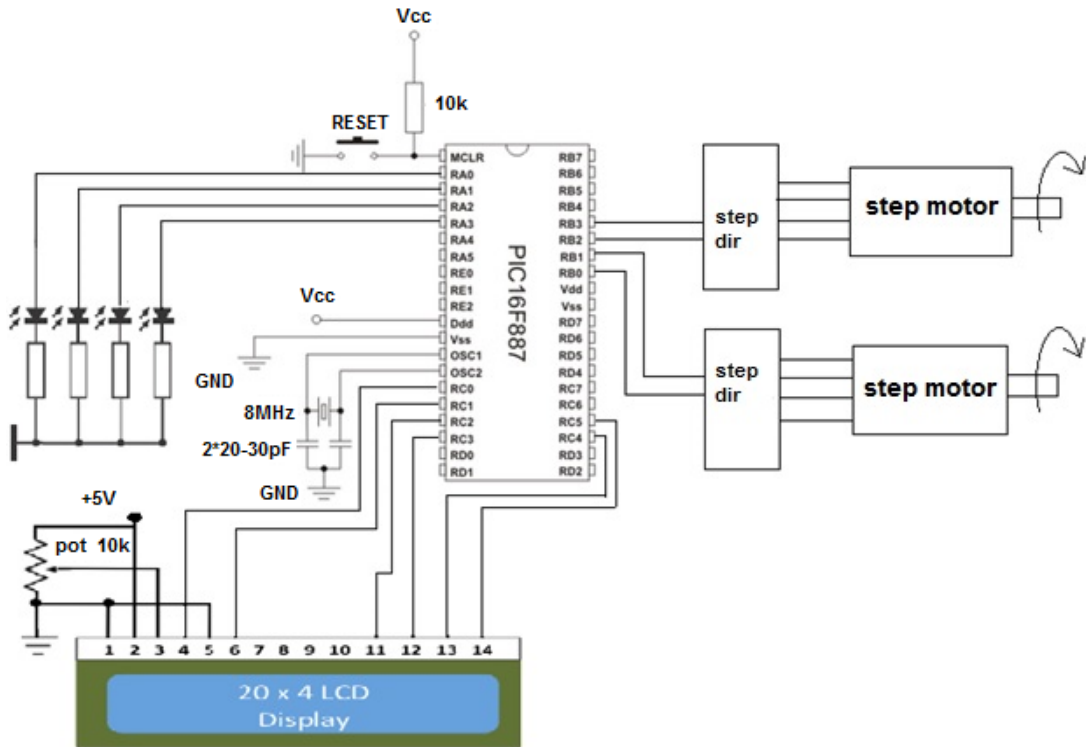


Figure 10. The PIC based controller

The microprocessor is PIC16F887 with sufficient functionality for acquisition of four 10-bit analog channels (LDRs) and four digital outputs for driving the two steppers.

Depending on the drive's power consumption an optimal strategy for the update period can be calculated so that the cost of the tracking remains insignificant against the power production.

5. CONCLUSION

The preliminary assessment of the proposed auto-tracker prototype shows some advantages that seem significant if taken into consideration:

- ability to maintain optimal performance/cost ratio achieved with the simple mechanical design and intelligent control strategy

- minimized self consumption of both the simple electronics and the fast and short lasting updates
- maximized productivity of the PV array by continuous maintenance of the optimal orientation

- the presence of the microprocessor allows both software modifications and upgrades, as well as implementation of communication protocols for networking with other controllers of large PV arrays
- ability for continuous monitoring of significant parameters and algorithm optimization

REFERENCES

- [1] L.D. Partain, "Solar Cells and Their Applications", *John Wiley & Sons, New York, 1995*
- [2] M.R. Patel, "Wind and Solar Power Systems", *CRC Press, 1995*
- [3] H.Moghbelli and R.Vartanian, "Implementation of the Movable Photovoltaic Array to Increase Output Power of the Solar Cells", *Proceedings of the International Conference on Renewable Energy for Developing Countries, 2006*

SUN TRACKER SISTEM BAZIRAN NA MIKROKONTROLERU SUN TRACKER SYSTEM BASED ON MICROCONTROLLER

Saša Stojanović, Dragan Tošić, Zoran Milivojević, *Visoka Tehnička Škola, Niš, Srbija*

Sadržaj: U ovom radu prikazana je hardverska realizacija uređaja za pozicioniranje solarnih panela u odnosu na poziciju Sunca. U prvom delu rada opisani su funkcionalni zahtevi uređaja. U drugom delu rada na nivou blok dijagrama opisana je hardverska struktura uređaja.

Ključne reči: mikrokontroleri, ugrađeni sistemi, optički senzor.

Abstract: This paper presents the hardware realization of the device for positioning solar panels in relation to the position of the Sun. The first part describes the functional requirements of the device. The second part describes hardware structure of the device at the block level.

Keywords: microcontrollers, embedded systems, optical sensor.

1. UVOD

Solarni paneli predstavljaju osnovnu komponentu u korišćenju solarne energije kao alternativni izvor energije, gde se primljena solarne energija pretvara u električnu energiju. Najzastupljeniji solarni paneli su sačinjeni od silicijumski-baziranih kristalnih (monokristalni, polikristalni) ili amorfnih fotonaponskih ćelija povezanih u jednu električnu sredinu, čija efikasnost zavisi od intenziteta Sunčeve svetlosti i od ugla padanja Sunčevih zraka [1]. Zbog toga solarne panele u toku dana treba postavljati u odgovarajući položaj prema Suncu kako bi dobili veću efikasnost fotonaponskih ćelija. Osnovni problem predstavlja fiksna montaža solarnih panela, čime se maksimalna iskorišćenost fotonaponskih ćelija ograničava samo na određeni ugao Sunčevog zračenja u toku dana.

Sun tracker sistem predstavlja automatizovan sistem za usmeravanje solarnih panela prema Suncu sa ciljem povećanja efikasnosti fotonaponskih ćelija, koja u zavisnosti od lokacije može biti 20-60% [2-7]. Osim usmeravanja solarnih panela, moguće je korišćenje Sun tracker sistema i za usmeravanje drugih objekata prema Suncu, kao što su ogledala, sočiva i drugi optički uređaji [8-10].

U ovom radu prikazano je hardversko rešenje Sun tracker sistema čija je realizacija predmet specijalističkog rada na Visokoj tehničkoj školi strukovnih studija u Nišu. Uređaj ima za cilj da pronalazi i prati poziciju Sunca postavljajući solarne panele u idealnom položaju prema Suncu. Platforma se sastoji od tri hardverskih celina: senzora, upravljačke jedinice i pogonske jedinice. Upravljačka jedinica je bazirana na mikrokontroleru ATMEL AT89C4051 [11,12] i ima ulogu da na osnovu signala dobijenih iz senzora pokrene pogonsku jedinicu koja solarne panele postavlja u odgovarajućem položaju.

Organizacija rada je sledeća: u sekciji 2 opisani su funkcionalni zahtevi uređaja, u sekciji 3 predstavljeno je hardversko rešenje uređaja kroz blok šemu sistema.

2. FUNKCIONALNI ZAHTEVI

Osnovni funkcionalni zahtevi koje uređaj treba da izvrši su sledeći:

- 1) detekcija i praćenje pozicije sunca,
- 2) postavljanje platforme u idealan položaj,
- 3) ignorisanje alternativnih izvora svetlosti (ulične svetiljke, farovi automobila...).

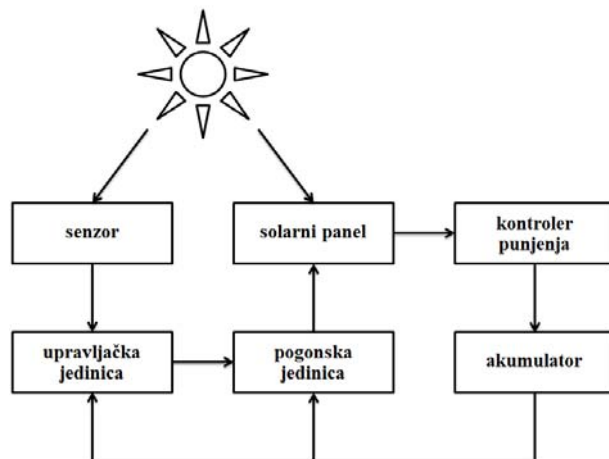
Pred uređaj se mogu postaviti i dodatni funkcionalni zahtevi:

- 1) daljinsko praćenje položaja platforme putem računara,
- 2) daljinsko upravljanje sistemom.

3. HARDVERSKO REŠENJE UREĐAJA

Sun tracker sistem čine tri osnovne celine:

- 1) senzor,
- 2) upravljačka jedinica,
- 3) pogonska jedinica.

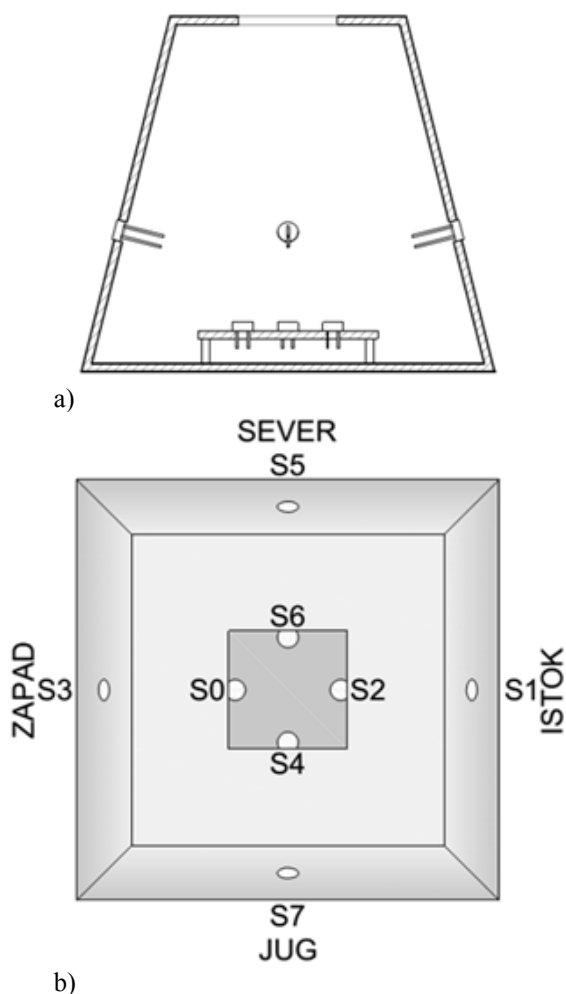


Slika 1. Blok šema sistema.

Na slici 1 prikazana je blok šema celog sistema. Solarni paneli i senzor, koji su konstrukcijski spregnuti u istoj ravni, primaju Sunčeve zrake. Senzor šalje podatke upravljačkoj jedinici o položaju platforme u odnosu na Sunce. Upravljačka jedinica na osnovu dobijenih podataka radi korekcije položaja određuje smer kretanja motora pogonske jedinice koja postavlja solarne panele i senzor u idealni položaj prema Suncu. Električna energija generisana u solarnim panelima smešta se u akumulatoru odakle se koristi za napajanje električnih uređaja i samog upravljačkog sistema.

3.1 SENZOR

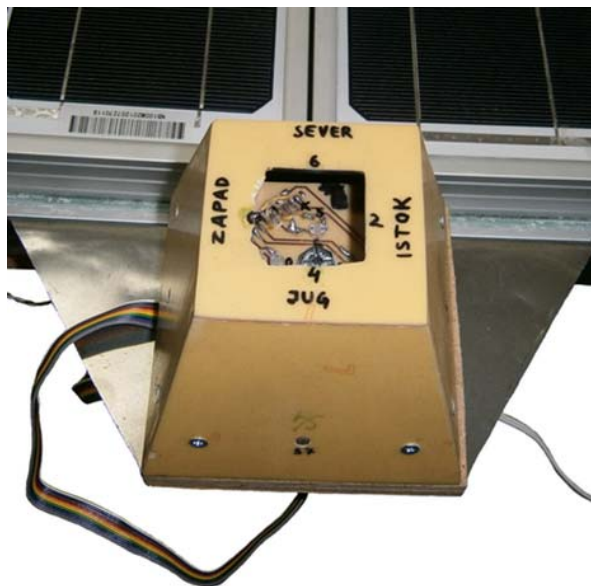
Uloga senzora je da detektuje Sunčeve zrake, odnosno da određuje položaj solarnih panela u odnosu na Sunčeve zrake. To se postiže upotrebom fotootpornika i odgovarajućom geometrijom samog senzora.



Slika 2. Šematski prikaz senzora: a) Poprečni presek; b) Pogled odozgo.

Konstrukcija senzora je u obliku četverostrane zarubljene piramide, kao na slici 2 i 3. Na svakoj strani piramide nalazi se po jedan fotootpornik. Na vrhu piramide, na zarubljenoj strani, nalazi se kvadratni otvor kroz koji prolazi svetlost do dna piramide, gde se nalaze četiri fotootpornika. Fotootpornici sa spoljašnje strane piramide služe za grubo navođenje, dok fotootpornici u unutrašnjem delu služe za precizno navođenje. Razlika u osvetljenosti fotootpornika na

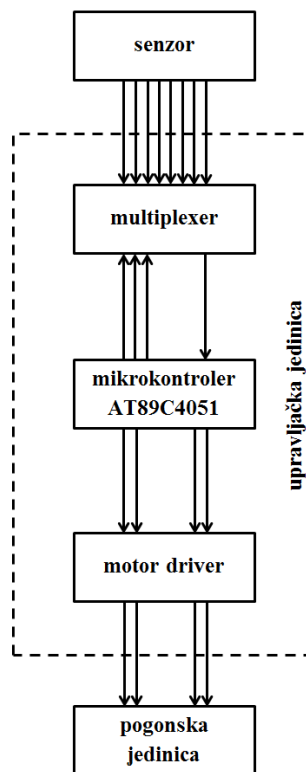
suprotnim stranama piramide se koristi kao signal za određivanje smera u kom treba da se kreću paneli. U slučaju da su svi spoljašnji, odnosno unutrašnji, fotootpornici približno jednako osvetljeni, program zaključuje da je mrak ili da je oblačno i u tom slučaju uređaj ne pokreće motore.



Slika 3. Senzor za detekciju svetlosti.

3.2 UPRAVLJAČKA JEDINICA

Upravljačka jedinica predstavlja centralni deo uređaja. Njeni osnovni elementi su multiplexer [13], mikrokontroler i motor driver [14]. Mikrokontroler koji je korišćen za ovaj uređaj je AT89C4051, proizvod kompanije ATMEL.

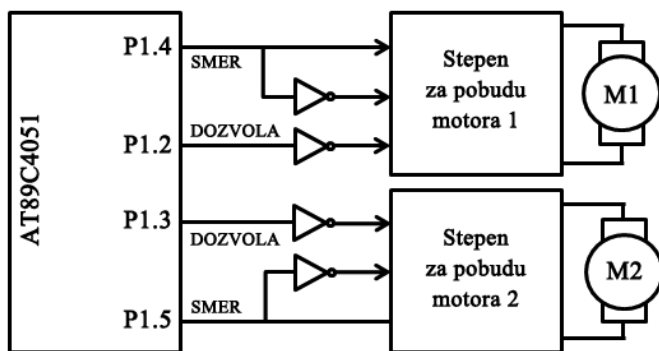


Slika 4. Blok šema upravljačke jedinice.

Upravljačka jedinica prima od senzora osam naponskih signala. Svaki naponski signal predstavlja nivo osvetljenosti određenog fotooptornika. Na ulazu upravljačke jedinice nalazi se analogni multiplexer koji na svom izlazu propušta naponski nivo sa jednog od svojih osam analognih ulaza. Mikrokontroler vrši izbor ulazne linije, koju će multiplexer proslediti na svoj izlaz, slanjem kombinacije na tri adresne linije multiplexera (slika 4).

Mikrokontroler saglasno implementiranom algoritmu vrši sva proračunavanja i određuje da li će i kojom brzinom pokrenuti motore. Za određivanje brzine motora mikrokontroler koristi PWM (Pulse Width Modulation) modulaciju koja radi na principu promene odnosa impuls-pauza.

Mikrokontroler šalje dva signala Motor Driver-u, za svaki motor posebno. Jedan signal služi za određivanje smera motora, a drugi za dozvolu pokretanja motora. Signal za određivanje smera motora grana se na dva ulaza motor drajvera gde jedan od signala prolazi kroz logičko NE kolo (Slika 5).



Slika 5. Šematski prikaz upravljanja motora.

Motor Driver ima mogućnost da pokreće dva motora u smeru koji zavisi od kombinacije signala koje šalje mikrokontroler. Motor Driver radi u mosnom režimu. Kada je na izlazima koji idu na isti motor različit napon (na jednom 0V, na drugom 12V) motor se pokreće u smeru koji zavisi od toga koji kraj motora je na 0V a koji na 12V.

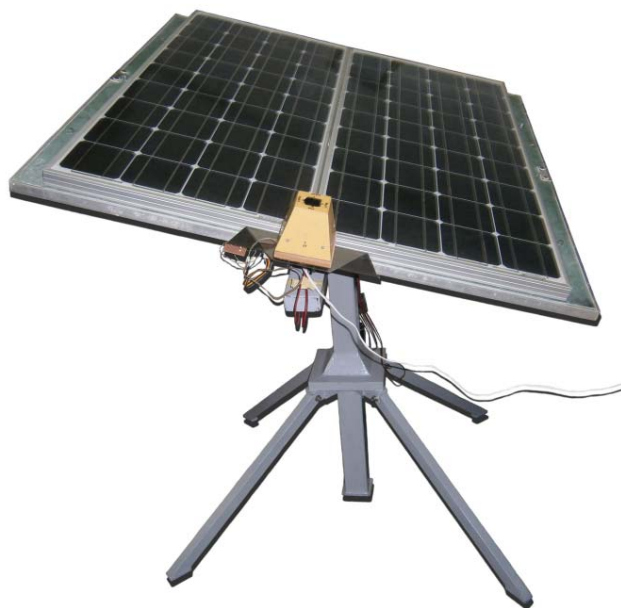
3.3 POGONSKA JEDINICA

Pogonsku jedinicu čine dva motora jednosmerne struje, odnosno dva linearna aktuatora sa pužnim prenosnim mehanizmom i pokretni nosač solarnih panela dvostruke osovine. Na svakoj osovinu je montiran motor koji je pokreće. Jedna osovina pomera nosač solarnih panela po horizontalnoj osi, odnosno u pravcu istok-zapad, a druga osovina pomera nosač solarnih panela po vertikalnoj osi, odnosno u pravcu sever-jug.

4. ZAKLJUČAK

U radu je predstavljena hardversko realizacija uređaja za praćenje Sunca baziranog na mikrokontroleru ATMEL AT89C4051. Uređaj detektuje, prati položaj Sunca i postavlja solarne panele u idealnom položaju prema Suncu čime

povećava njihovu efikasnost. Uređaj je realizovan i testiranjem je ispunio sve postavljene funkcionalne zahteve.



Slika 6. Izgled platforme.

LITERATURA

- [1] Chiras Daniel, Solar electricity basics: a green energy guide, *New Society Publishers, Gabriola Island, Canada, 2010.*
- [2] K. Sreenivasa Rao, M. Mahesh, ARM Based Solar Tracking System, *International Journal of Modern Engineering Research, Vol.2, Issue.4, pp-2504-2507, 2012.*
- [3] Liping Guo, Jingbo Han, Andrew W Otieno, Design and Simulation of a Sun Tracking Solar Power System, *American Society for Engineering Education, Paper ID #7854, 2013.*
- [4] Chniba Saïd, Ben Attia Yassine, Bouajila Youssef and Frigui Ali, Design of a Photovoltaic Panel Orientation System, *International Renewable Energy Congress, ID #71, Tunisia, 2010.*
- [5] C. Praveen, Design of Automatic Dual-Axis Solar Tracker using Microcontroller, *Published by Coimbatore Institute of Information Technology, ISBN 978-1-4675-2248-9, 2012.*
- [6] Nader Barsoum, Pandian Vasant, Simplified Solar Tracking Prototype, *Transaction in Controllers and Energy, ES-E11/GJTO, 2010.*
- [7] <http://www.solar-tracking.com>
- [8] <http://www.redrok.com>
- [9] <http://www.builditsolar.com>
- [10] <http://www.heliotrack.com>
- [11] <http://www.atmel.com/Images/doc1001.pdf>
- [12] Z. Milivojević, Mikrokontroleri-ArHITEKTURA 8051; *Punta, Niš, Srbija, 2005.*
- [13] <http://www.ti.com/lit/ds/schs047g/schs047g.pdf>
- [14] https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf

UPRAVLJANE ROBOTA PREKO INTERNETA CONTROLLING ROBOTS ONLINE

Dimitrija Angelkov, Cveta Martinovska Bande, University Goce Delcev, Computer Science Faculty, Stip, Macedonia

Sadržaj: *Da bi bio uspešan u realnom vremenu upravljaju roboti zasnovane na internetu potrebno je visok stepen slobode i autonomije sa lokalnom inteligencije. Sistem mora da ima standardni mrežni protokol i mašinskog interfejsa. Glavni ceo je korisnik koji koristi veb browser na daljinskom operatera lako da kontroliše mobilnog robota. Sa povećanjem robota i robotizovanih aplikacija javlja se potreba da se pronade način kroz koji će moći da se realizuje upravljanje daljinski kontrolisano. Kontrola robota preko interneta je od posebnog značaja. Danas postoje pouzdani protokoli kroz koje dobijamo povratne informacije i tako možemo da ostavimo interakciju između čoveka i mašinom. Problem daljinsku kontrolu i upravljanje robota sistema arhitekture i interfejsa za kontrolu jednog ili više roboti u različitim aplikacijama je tema stalnih istraživanja u poslednjoj deceniji. Ovaj naučan trud doprinosi da se pronade inovativne metode za upravljanje više robota istovremeno u realnom vremenu putem interneta.*

Abstract: *To be successful in real-time applications run Internet-based robots require a high degree of freedom and autonomy with local intelligence. The system has a standard network protocol and interactive man machine interface. Remote operator can easy control mobile robot into interactive way. With the increase of robots and robotic applications is an increasing need to find a way through which they will be controlled remotely. Control of robots via the Internet is of particular importance because there are reliable protocols through which we receive feedback and have interaction man and machine remotely. The problem of remote control and management of robots, systems architecture and interfaces to control one or more robots in various applications is a topic of constant research in the last decade. This paper contributes to find innovative methods for managing multiple robots simultaneously in real time through the Internet.*

1. INTRODUCTION

Control of robots from a distance have traditionally been designed and managed locally by a single man. The need for mechanical manipulation device distance dating from distant fifties when he first used the remote management in nuclear power plants. The need for managing nuclear fuel is of particular importance because the individual is exposed to the radiation dose that is threatening his life. To eliminate the presence of man was needed to make the system to be operated remotely, and so incurs the beginnings of remote control. Pioneers in this area include M-r. Raymond Goertz from Argon National Laboratory near Chicago and M-r. Jean Vertut with his engineers of nuclear engineering laboratory near Paris. They construct a robotic arm with 6 degrees of freedom which in itself has electromechanical and electro hydraulic system actuators and feedback to the dashboard. Each command has given feedback about whether it is successfully executed. Their aim was for the shortest time to transfer radioactive waste from nuclear power to canisters for storing. The next vision was to make a system that will be operated remotely. This system was built managed to close the distance. The need to eliminate any effect of radiation was very important and necessary to do any management remotely. In this intention failed to make a system that works and distance will be minor delay from the time of setting commands until the execution of the command. Web-based education and control of robots distance presents a growing field in engineering education with a substantial amount of educational material and many tools and platforms for

education. Many universities have adapted their educational programs to learn the distance and direct communication is replaced with the new technologies for transmitting video, audio, and multimedia data print. Professor has interactive teaching large numbers of students and reliably accomplished relationship between the teacher and the listener almost identical spot. Today, distance education is implemented through the Internet and World Wide Web (WWW). Most innovative in how the Internet is a relatively short time has grown into the most powerful computer network in the world. In the area of control and educational robotic engineering have so far achieved important results in the development of interactive web - based platform that is produced materials. Online courses with theoretical and laboratory exercises that are made through the web-based physical laboratories.

2. ROBOT DETECTION

To reduce the time delay from the end user to the robot we created a program that uses technology with short time response. The program "robotdetection.swf" is create in adobe flash cs6 professional. Program includes five scripts "Color32.as", "CustomVideo.as", "DragObject.as", "BmpGenerator.as", "ColorTracker.as" main program script "Main.as" which invites all the functions required for proper detection of robots. To successfully distinguish one robot from each of the other four robots brings cube with different colors on it . The first robot has a red cube, blue cube has a second, third robot has a green cube and fourth robot has a

yellow cube. Blocks are attached to the robot and consider the movements of robots. The script is used ColorTracker.as order "threshold" filter through which we make the previously chosen color of the visual display screen . We set limits on who can do the tolerance previously selected color in order to reduce noise arising when moving object. Also use a command `getPixel32 (loc2, loc1)` which gives us the value of the pixel that is `loc2` and `loc1` respectively in the x-axis and y-axis in pre initialized dimensions of the video do the order Custom Video (640,480). This command defines the dimensions that can be come from video image, in our case the webcam.

Short example of how to use this command on all four robots.
`private var _searchRange: int = 25`
`// rank of the search is set to 25`
`while (loc1 <_position.y + _searchRange) // means doing`
`analysis of camera image pixel by pixel distance of 25 pixels`
`so you can quickly send the desired robot that was previously`
`chosen image .`

```

    {
        loc2 = _position.x - _searchRange;
        while (loc2 <_position.x + _searchRange)
        {
            If (_trackBmd.getPixel32 (loc2, loc1)) // we take
            the value of the corresponding pixel if no such value does not
            increase the amount in X and Y axis
            {
                _xSum = _xSum + loc2;
                _ySum = _ySum + loc1;
                var loc3: *;
                _xCnt ++;
                _yCnt ++;
            }
            ++ loc2;
        }
        ++ loc1;
    }
    _trackBmd.threshold (_trackBmd, _trackBmd.rect, new Point
    (), "<", _r1.color, _fillClr, 4294901760 )
    // define the boundaries in which the requirement applies
    _trackBmd.threshold (_trackBmd, _trackBmd.rect,
    new Point (), ">", _r2.color, _fillClr, 4294901760 )
    // define the boundaries in which the requirement applies

    _trackBmd.threshold (_trackBmd, _trackBmd.rect,
    new Point (), "<", _g1.color, _fillClr, 4278255360 )
    // define the boundaries in which the requirement applies

    _trackBmd.threshold (_trackBmd, _trackBmd.rect,
    new Point (), ">", _g2.color, _fillClr, 4278255360 )
    // define the boundaries in which the requirement applies

    _trackBmd.threshold (_trackBmd, _trackBmd.rect,
    new Point (), "<", _b1.color, _fillClr, 4278190335 )
    // define the boundaries in which the requirement applies

    _trackBmd.threshold (_trackBmd, _trackBmd.rect,
    new Point (), ">", _b2.color, _fillClr, 4278190335 )
    // define the boundaries in which the requirement applies

```

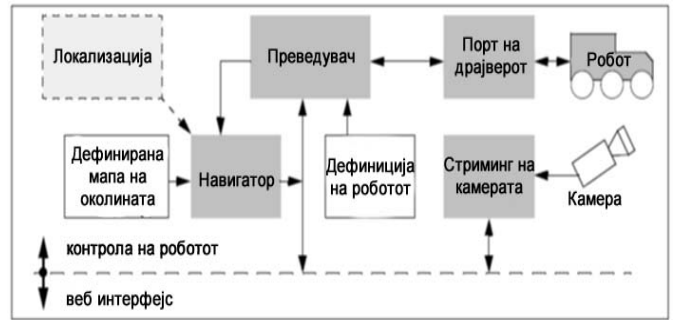


Figure 1. Block diagram through which demonstrates the information from the camera to the currently controlled robot.

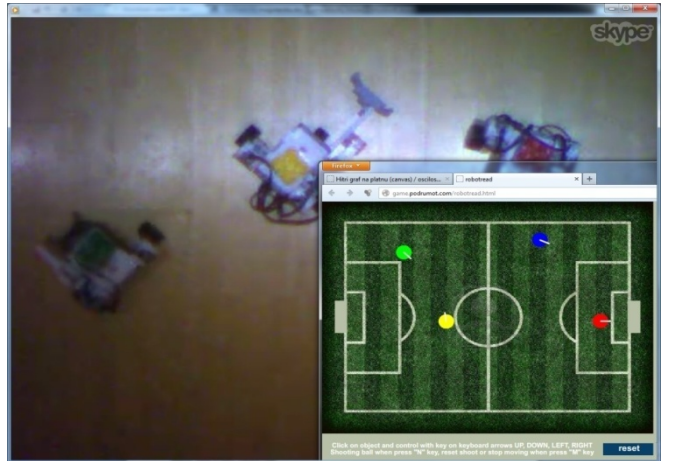


Figure 2. Form activating robots and activation commands and transfer photos from real robots via Skype video streaming service

In the program `Main.as` activate all functions necessary for proper detection of each of the individual robots. In this program the selection of the desired robot do by clicking on any of the colored circles. In our example, colored with blue, green, red and yellow. After four successive clicks on the window of the program is activated detection of objects in color. Then occurs following each of the robots. The program uses modules and color retention hysteresis. These two modules are the fastest way to recognizing the positions of each of the robots. Programs running on each of four consecutive robots. This program is made to create a virtual environment so that we and no image from the camera to direct the robots each of them individually. The end user may not know that operated robots but have an idea that moving objects in the user interface. The advantage of this visual representation is to have the ability to monitor a huge number of robots at the same time. This program can be used in video surveillance, where a large number of cameras and moving objects will be presented in a separate visual graphic interface . In our football bunch of robots is used in accurate selection of the desired robot challenge. If you would like a set amount then we will need more time to select the desired robot. Another application would find this program in tracking the ball a football field. Both modules are sufficient for identifying and tracking the object as a soccer ball. In addition, the program can find application in defending the goal of achieving the goals.

3. PROGRAM FOR SELECTION OF FREE ROBOTS

In this program, we have a visual representation of each of the robots and their current positions. Accordingly, each of the robots is marked with a specific color. The user selects a free robot and pressing the buttons on the left, right, up, down, button H and M button controls the desired robot. If a robot previously occupied by, another user will then be able to select. Each of the objects in the program is initialized at each new frame of flash program. Therefore, if a user decides to cancel the movement of things in the next frame the robot can be selected and further managed. Right from the playground is display the status of robots. It shows the current commands and running status of the robots is free. The picture with row number 3 is displayed for selection and program management of robots.

Program code is optimized in the minimum number of rows to reduce the time of sending commands from the program for each of the robots. The program makes permanent renewal of the coordinates of each of the robots and their status. The data read from the database which is located on the server side. It set the values for each of the robots. These data come from the program for the detection and tracking of robots. This program is installed on the local computer that is responsible for executing commands that arrive from this program and parallel charging base values that the right program for recognition and detection.



Figure 3. Program for the selection and management of the desired robot

4. CONTROL OF THE ROBOT VIA BLUETOOTH CONNECTION

We create "robotonline.exe" program is setting the robot commands by reading data from mysql base which is set by the server . This data base kept by the coordinates of each of the robots and that appropriate command is activated . Each of the robots to operate independently of the other four programs made specifically robotonline1.exe, robotonline2.exe, robotonline3.exe and robotonline4.exe. Each of these programs is configured individually through

separate text file "config.cfg". It is set for each of the robots particular port. This port when do we get a pairing of each of the robots local computer. Communication library in c + + allows the easy way to communicate with Lego Nxt Robot communication via Bluetooth or network connection .

To be able to communicate with Lego Nxt Robot via Bluetooth communication it is necessary to install device drivers and checked all the robots that are included devices for wireless communication in our case Bluetooth device. Once you establish a connection, you can see which ports to use the desired robot and thus the configuration file to insert them. In this way the startup programs specifically each robot will initialize a separate gate and there will be no mixing of commands. Robots previously registered password to verify that the desired coupling.

This C + + communication library can be used in most c + + compilers on Windows operating systems. Also possible is the handhelds running Windows Mobile. In addition to programming and compile the program use MS Visual Embedded C + +.

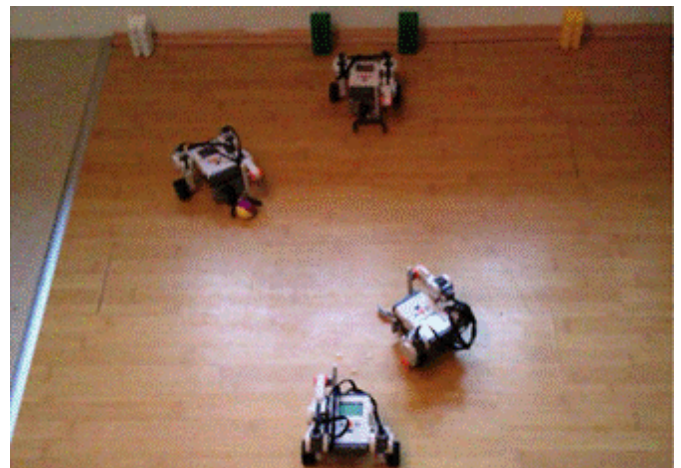


Figure 4. Display of robots in action

5. CONCLUSION

We do daily interaction with machines and created the so-called human entities. I put the card, hit the button and follow the instructions of the device for raising money. Include washing machine or microwave oven nearly every day. You must admit that this is the era of man- made machines that are interactive and easy to operate. The real question is how the individual perceives robots and what are the current relationship and the relationship between man and machine. Today robotics takes increasingly important place in the life and work of the man. At the beginning of the 21st century began expansion of practical application of robotics and intelligent machines. To be man enough informed technical environment and to better use needs to be technically educated enough in this area. This education must be flexible with constant update. It is therefore very important role of higher education institutions must create educational content that robotics will provide the necessary minimum knowledge in this area. This master thesis contains basic technologies to be used to be able to have the management of robots and

intelligent machines remotely. One of the major challenges in the development of artificial intelligence cooperation between man and robot. Cooperation unlike interaction involves a process of intelligently predict the next actions of the Contractor, and acting in accordance with those expectations. Thus, it is possible to support the Contractor or to point to a different opinion, in order to avoid mistakes and get the best consensus opinion in the team. Therefore, cooperation implies a good knowledge of colleagues, team building, compassion, a sense of mutual success and so on. , Which is much more than simple interaction through human or robotic senses? Hence, intelligence to be manifests to ensure successful cooperation in solving complex problems is often difficult to reach and the people, and therefore represents a major challenge in the development of artificial intelligence. Given the fact that the robotics industry is developing very fast, very important to define the rules for cooperation between robotic systems, in order to intercept a number of ethical issues for people living together in harmony and robots. Researchers often see mobile robots as quasi- static devices. Numerous examples of robots with four, six or more wheels are also developed to maximize the mobility of bad pitches. Unlike these egzo skeleton robots that may have greater potential for mobility, built and demonstrated mainly so you have a low center of gravity and plate set as a supporting base , which together with intelligent control algorithm is designed to retain the gravitational vector axis through the base polygon (ie slope motor , foot coordination) . In many concepts, we attempt to maximize mobility with large wheels or legs, with extended wheel rims or soles of the feet, more rundles driver, great body / open space area, articulated body configurations etc... Such robots are usually limited in terms of mobility planned so that is designed to operate at low speeds, which typically extend to 1 km/h or less. Dynamic factors have little impact on such systems, and are therefore often ignored. To overcome these limitations in robotics must use the controller that will gradually outweigh the problems. One of the many controllers used currently EyeBot controller for mobile robots with wheels, walking robots or flying robots. This controller allows us to increase mobility, smart our mobile robots. Through simulation programs developed controllers quickly reveal mistakes when developing an appropriate algorithm that are free of unnecessary disposal means that robots are not sure whether the current program will be suitable for the physical robot . The current level of support and EyeBot Lego Nxt is the leading mobile robotic platforms mostly due to its direct link with the camera and embedded wireless module for communication with other devices. This is because as the camera plays a very important role in solving the problems of robotics especially in football bunch of robots. Through practical simulation, robocup seen his huge potential in being able to solve problems in a dynamic part of automated systems. Another to mention is that the technologies used in this master thesis are open and can be used on multiple platform system. Algorithms of this paper

have been implemented in website www.robotsonline.info. Through this site measurements were performed on the speed of response of each of the robots individually. Measurements are shown in Table 1. This program is free to use and develop exclusively for scientific research purposes. The table with sequence number 1 we can conclude that there is little time difference is maximum plus minus 9 milliseconds. This deviation is quite satisfactory for football bunch of robots. The whole system is dynamic and constantly changing positions all the robots. To reduce the time of execution of server commands need to be closer to where all events are executed.

Table 1. Recorded values of time execution commands

Ordinal test	Total number of successfully executed commands in a minute	Average time of execution of a command in one second(s)	Unlike the mean execution	Mean execution
1	95	0.631	0.008	0.623
2	95	0.631	0.008	0.623
3	96	0.625	0.002	0.623
4	97	0.618	-0.005	0.623
5	95	0.631	0.008	0.623
6	98	0.612	-0.011	0.623
7	97	0.618	-0.005	0.623
8	96	0.625	0.002	0.623

REFERENCES

[1] Giulio Ferrari, Andy Gombos, Soren Hilmer, Jurgen Stuber, Mick Porter, Jamie Waldinger, LEGO MINDSTORMS NXT G Programming Guide 2nd Edition Jun 2010

[2] M. Micire, J.L. Drury, B.Keyes, H. A. Yanco, Multi-touch interaction for robot control, IUI '09 Proceedings of the 14th international conference on Intelligent user interfaces, ACM New York, NY, USA 2009

[3] S. Yang, R.R. Igorevich, Y. Park, D. Min, E. Choi, Reliable robot teleoperation service architecture for various controllers and robots, In the Proc. of the 7th International Conference Networked Computing and Advanced Information Management (NCM), 2011

[4] Suzuki, T., Ohya, A., Yuta, S.: Operation Direction to a Mobile Robot by Projection Lights. In: IEEE Workshop on Advanced Robotics and its Social Impacts, TPO-2-2, (2005)

APLIKACIJA INTELIGENTNIH SISTEMA-ROBOTA U PROIZVODNIM PROCESIMA

Isak Karabegović, Ermin Husak, *Tehnički fakultet Bihać*
Milena Đukanović, *Elektrotehnički fakultet Podgorica*

Sadržaj: *Automatizacija proizvodnih procesa u svim industrijskim granama se ne može zamisliti bez inteligentnih sistema. Modernizaciju i automatizaciju proizvodnih procesa nemoguće je analizirati bez primjene industrijskih robota u svijetu. Industrijski roboti se primjenjuju u svim granama industrije. Stalni trend rasta primjene industrijskih robota vezan je za njegovu funkcionalnu opravdanost, a razloga za to ima dosta. Nabrojimo neke od razloga: zaštita radnika u neuslovnim prostorima rada ili od teških fizičkih poslova, ušteda vremena, postizanje odgovarajućeg kvaliteta, povećanje produktivnosti i sl. Sa razvojem novih tehnologija postoje stalne promjene u robotici koje dovode do novih funkcionalnih rješenja i mogućnosti primjene robota. U ovom radu izvršena je analiza primjene inteligentnih sistema-robota u proizvodnim procesima u svijetu. Analizirana je njihova primjena u onim zemljama u kojim je razvijena automobilska industrija iz razloga što u ovoj industriji je njihova najveća zastupljenost.*

Abstract: *Automation of production processes in all industrial areas could not be assumed without intelligent systems. Modernization and automation of production processes is impossible to analyze without the use of industrial robots in the world. Industrial robots are used in all areas of industry. Permanent growth trend of industrial robot application is connected to its functional validity and there are many reasons for that. Lets count some of the reasons: worker's protection in unsuitable working spaces or from hard physical jobs, saving of time, achieving appropriate quality, productivity increase etc. With the development of new technologies there are permanent changes in robotics which lead to new functional solutions and possibilities for robot applications. In this paper an analysis of intelligent system-robot applications in production processes in world has been done. Their use in countries in which automotive industry is developed has been analysed because in this industry is their highest representation.*

1. UVOD

Automatizacija proizvodnih procesa u svim industrijskim granama u svijetu zaslužuje analizu, a samim tim i primjena industrijskih robota jer bez nje nema automatizacije proizvodnih procesa. Industrijski roboti se primjenjuju u svim granama industrije [1,7,8,9]. Primjene industrijskih robota u proizvodnim procesima raznih industrija se iz godine u godinu povećava. Stalni trend rasta primjene industrijskih robota vezan je za njegovu funkcionalnu opravdanost, a razloga kao što se zna ima dosta. Jedan od razloga je stalna automatizacija i modernizacija proizvodnih procesa, a drugi razlozi su zaštita radnika u neuslovnim prostorima rada ili zaštita radnika od teških fizičkih poslova, ušteda vremena, postizanje odgovarajućeg kvaliteta, produktivnosti, cijene i sl. Njihova primjena još uvijek je najveća u automobilskoj industriji, ali i druge grane industrije ne zaostaju; na primjer elektro-industrija u povećanju broja implementiranih industrijskih robota.

Sa razvojem novih tehnologija postoje stalne promjene u robotici koje dovode do novih funkcionalnih rješenja i mogućnosti primjene robota. Inteligentne mašine i sistemi različitog nivoa složenosti su danas sve prisutniji za obavljanje različitih procesa. Inteligentne mašine i sistemi kao što su: roboti, tehnološke ćelije i slično čine stub CIM-sistema koji predstavlja temelj svake koncepcije fabrike

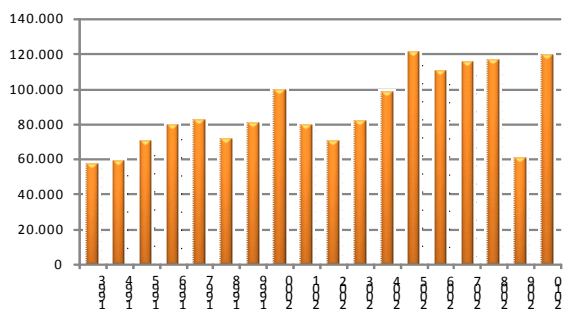
budućnosti. Razlog za sve veću motivaciju primjene industrijskih robota leži u nekoliko sljedećih osnovnih činjenica: povećanje produktivnosti, smanjenje troškova, savlađivanje nedostataka stručnosti čovjeka (preciznost), veća fleksibilnost kod određenog stepena proizvodnje, poboljšanje kvaliteta proizvodnje, oslobađanje čovjeka od monotoni i ponavljajućih zadataka.

2. TREND PRIMJENE INDUSTRIJSKIH ROBOTA U SVIJETU

Statistički podaci za broj primjena industrijskih robota u proizvodnim procesima u svijetu preuzeti su od *International Federation of Robotics (IFR)*, te podataka *Ekonomске Komisije pri UN za Evropu (UNECE)* i *Organizacije za ekonomsku kooperaciju i razvoj (OECD)* [4,5,6,9]. Ova analiza treba da ukaže na primjenu industrijskih robota u svijetu i po kontinentima Evrope i Amerike, Azije/Australije i Afrike.

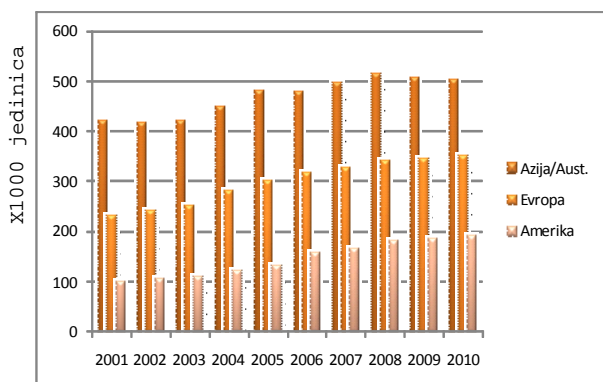
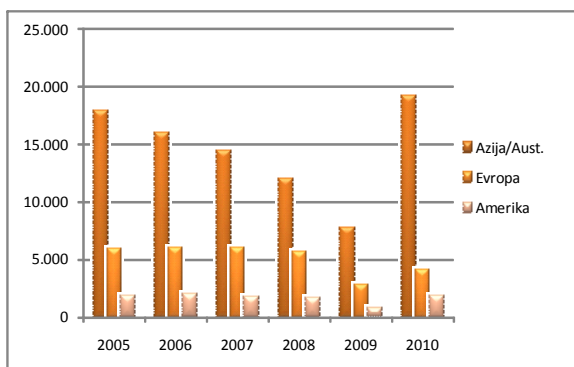
Godišnji trend primjene industrijskih robota u svijetu u periodu 1993-2010 prikazan je na slici 1. iz koje možemo zaključiti da je minimalna primjena robota zabilježena 1998., 2002. i 2009. godine. Od 1998-2000 godine imamo uspon u primjeni robota, a isto tako od 2002-2005 godine trend primjene je rastući. U periodu 2006-2010 godine možemo reći da je primjena industrijskih robota konstantna ako

izuzmemo 2009. godinu gdje je bila minimalna primjena industrijskih robota - oko 60.000 jedinica.



Slika 1. Godišnja primjena industrijskih robota u svijetu od 1993 – 2010 godine

Na osnovu slike 2. zaključujemo da je na prvom mjestu po primjeni industrijskih robota Azija/Australia sa 69.833 jedinice u 2008. godini, 30.117 jedinica u 2009. godini i 60.294 jedinica u 2010. godini. Drugo mjesto zauzima Evropa sa 30.630 jedinice u 2008. godini, 20.483 jedinica u 2009. godini i 34.695 jedinica u 2010. godini.

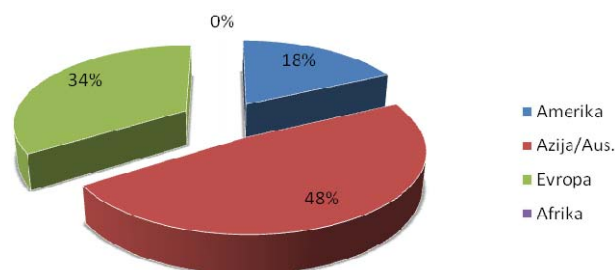


Slika 2. Godišnja i ukupna primjena industrijskih robota po kontinentima od 2001 - 2010 godine [4,5,6]

Na trećem mjestu je Amerika sa 17.114 jedinice u 2008. godini, 8.992 jedinica u 2009. godini i 17.192 jedinica u 2010. godini. Posljednje mjesto pripada Africi sa 256 jedinice u 2008. godini, 196 jedinica u 2009. godini i 454 jedinica u 2010. godini. Na sva četiri kontinenta u 2009. godini bila je minimalna primjena, a razlog je finansijska industrijska kriza.

Kada je analiziran procentualni odnos primjene industrijskih robota po kontinentima u 2010. godini redosljed

je sljedeći: Azija/Australia ima 48%, Evropa 34% i Amerika 18%, a u Africi je toliko zanemarljiv broj robota u odnosu na druge kontinente da je 0% u odnosu na ukupni broj primijenjenih robota u svijetu u 2010. godini (Slika 3).



Slika 3. Ukupni procentualni odnos primjene industrijskih robota po kontinentima u 2010. godini [4,5,6]

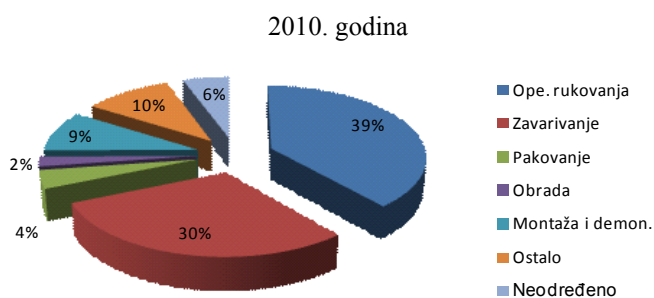
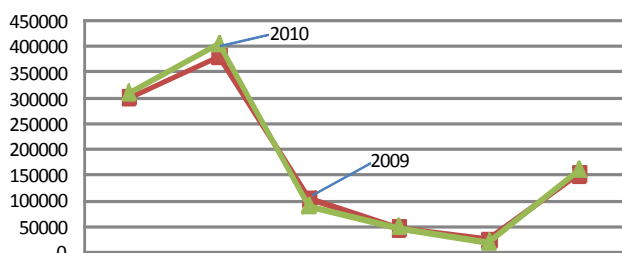
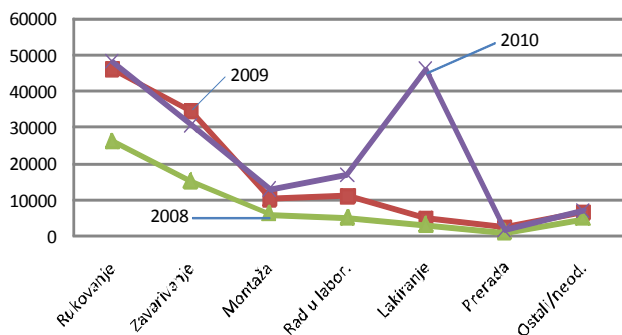
3. PRIMJENA INTELIGENTNIH SISTEMA-ROBOTA U PROIZVODNIM PROCESIMA U SVIJETU

Analiza primjene industrijskih robota u različitim industrijskim granama prikazana je u tabeli 1. Za analizu su uzete sljedeće industrije: poljoprivreda, šumarstvo, ribarstvo, rudarstvo, proizvodnja kako prehrambena tako i duvanska, tekstilna, papirna, drvna, kožna, metalna industrija, plastika i hemijska industrija, elektro/elektronska industrija, automobilska industrija, građevinska industrija itd. Kao što se iz tabele i grafikona vidi aplikacija robota u 2010. godini je porasla u svim industrijskim granama u odnosu na 2009. godinu zbog gospodarske krize koja je bila u 2009. godini.

Tabela 1. Ukupna i godišnja primjena industrijskih robota u Svijetu u različitim područjima primjene 2008-2010. godine [4,5,6]

PRIMJENA	Godišnja primjena robota u Svijetu			Ukupna primjena robota u Svijetu		
	2008	2009	2010	2008	2009	2010
Aplikacija/God	2008	2009	2010	2008	2009	2010
Ope.rukovanja	45.683	25.854	48.162	389.438	391.216	401.295
Zavarivanje	34.547	15.853	30.326	306.975	300.528	308.670
Pakovanje	4.307	2.733	3.907	40.501	41.060	42.298
Obrada	1.991	989	1.792	24.637	23.693	22.448
Mont. i demo.	10.311	5.903	12.984	114.004	105.075	95.586
Ostalo	13.277	6.222	17.879	100.575	102.366	108.165
Neodređeno	2.856	2.464	3.287	59.171	56.793	56.554
UKUPNO Σ	112.972	60.018	118.337	1.035.301	1.020.731	1.035.016

Ako pogledamo područja primjene 41% instalirano je u rukovanju materijalom, 26% u zavarivanju, 11% u montaži, 28% u automobilskoj industriji. Gledajući ukupan broj instaliranih robota u svijetu vidimo da se 39% odnosi na rukovanja materijalom, 30% u procesima zavarivanja svih vrsta i 9% je montaža u svim granama proizvodnje (Slika 4). Najveću primjenu doseglo je rukovanje materijalom - gotovo 48.200 jedinica, a razlog tome je automobilska industrija i želja privrednika za automatizacijom i modernizacijom proizvodnih procesa gdje se najviše traži rukovanje materijalom tokom proizvodnog procesa [1,5,9,11,13]. Kroz ovu analizu smo obuhvatili sljedeće proizvodne procese: operacije rukovanja, zavarivanje, montaža, pakovanje, obrade i ostale procese koji nisu svrstani u navedene proizvodne procese.



Slika 4. Godišnja, ukupna i procentualna primjena industrijskih robota u svijetu u raznim proizvodnim procesima u 2008- 2010 godini

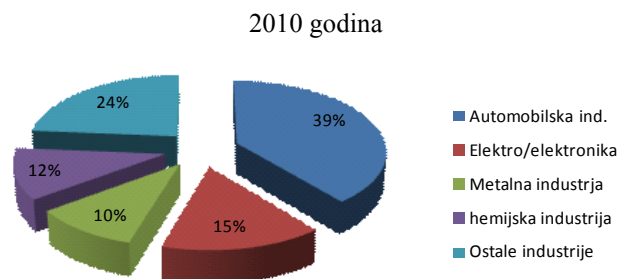
Na slici 5. prikazana je aplikacija inteligentnog sistema-industrijskog robota u procesu zavarivanja u automobilskoj industriji. Na osnovu tabele 1. i tabele 2. zaključujemo da je u sve proizvodne procese u svijetu u 2010. godini primijenjeno 118.335 jedinica robota, a ukupan broj koji je primijenjen u proizvodne procese u svijetu u 2010 godini iznosi 1.035.016 jedinica. Od ukupnog broja 39% je primijenjeno u operacijama rukovanja, 30% procesima zavarivanja, 4% procesima pakovanja, 9% procesima montaže i demontaže, 2% procesima obrade i 16% u ostalim proizvodnim procesima.



Slika 5. Aplikacija inteligentnog sistema-industrijskog robota u procesu zavarivanja u automobilskoj industriji

Tabela 2. Primjena industrijskih robota u svijetu po industrijskim granama u 2008-2010. godine [4,5,6]

INDUSTRIJA	Godišnja primjena rob.u Svijetu			Ukupna primjena rob.u Svijetu		
	2008	2009	2010	2008	2009	2010
Polj. i Rud.	72	91	80	683	737	762
Proizvodnja	94.262	49.162	91.176	810.098	806.034	807.410
- Plas.i ken.ind	12.575	6.852	10.518	114.899	113.700	110.064
- Metalna ind.	14.778	5.253	7.934	98.515	98.245	97.427
- Elek/elektro	16.439	10.855	30.745	147.755	144.457	146.539
-Automobilsk	40.437	19.260	32.702	363.785	364.852	373.223
ELen.pli. i vo.	45	16	21	240	256	246
Konstrukcija	232	226	340	1.264	1.417	1.485
Neodređeno*	28.396	16.867	35.970	223.016	212.257	225.513
UKUPNO Σ	112.972	60.018	118.337	1.035.301	1.020.731	1.035.016



Slika 6. Godišnja procentualna primjena industrijskih robota u svijetu u raznim industrijama i proizvodnim procesima u 2010. godini

Tabela 3. Primjena industrijskih robota u 2010. godini u različitim proizvodnim procesima

Zem.ins.	Zava.	Op.ruk.	Pako.	Obrada	Monta.	Čišće.	Ostal.	UKUP
China	8.000	3.725	864	173	892	1.200	124	14.078
Germany	4.129	6.742	526	421	468	180	1.534	14.000
N.Amer.	3.883	6.244	511	286	1.806	3.360	266	16.356
R.Korea	3.800	11.309	854	60	1.706	5.674	104	23.508
Japan	3.609	7.072	309	175	6.320	4.129	289	21.903
Thailand	1.449	831	36	6	109	13	6	2.450
Italy	807	3.118	79	120	299	-	85	4.517
Taiwan	605	985	91	177	356	1.032	44	3.290
Spain	563	991	56	34	46	-	208	1.897
France	496	1.111	80	61	106	-	195	2.049

Ukoliko analiziramo tabelu 2., i tabelu 3., te sliku 6. dolazimo do zaključka da proces zavarivanja predstavlja proces gdje primjena industrijskih robota ima veliku

primjenu, posebno u zemljama koje su vodeće u proizvodnji automobila odnosno motornih vozila.

Tabela 4. Primjena industrijskih robota u 2010. godini u različitim industrijama [4]

Zemlja	Auto.	Elek/Ele	Hemij.	Metal.	Hra.	Ostal.	Neodr.	UKUP
China	8.300	1.800	1.452	1.800	00	1.300	26	14.978
R.Korea	7.056	12.644	727	253	84	252	2.492	23.508
Germany	6.929	700	1.343	1.705	913	781	1.629	14.000
N.Amer.	5.368	3.555	1.551	1.189	935	249	3.509	16.356
Japan	5.223	9.940	1.832	1.556	429	2.923	-	21.903
Thailand	1.500	28	491	2	4	11	416	2.450
Italy	978	141	697	802	478	443	978	4.517
Spain	941	64	100	119	304	213	156	1.897
France	848	128	241	154	282	201	195	2.049

Oko 77% industrijskih robota instalirano je u procesu zavarivanja proizvodnje automobila širom svijeta, a prednjače sljedeće zemlje: Kina, Njemačka, Sjeverna Amerika, Republika Koreja i Japan.

4. ZAKLJUČAK

Na osnovu sprovedene analize možemo zaključiti da je po prvi put Kina vodeća u instaliranju industrijskih robota u procesu zavarivanja oko 8.000 jedinica u 2010. godini, što predstavlja 26% od ukupno instaliranih robota u procesu zavarivanja, dok je na drugom mjestu Njemačka sa 4.129 instaliranih jedinica robota. Ovo predstavlja normalan trend jer Kina ima velike investicije u automobilskoj industriji. Poslije Kine i Njemačke po primjeni robota u procesu zavarivanja su Sjeverna Amerika, Republika Koreja i Japan. Republika Koreja daje sve veću važnost automatizaciji proizvodnje, ne samo u automobilskoj industriji već i ostalim procesima tako da je u 2010. godini instalirala najveći broj industrijskih robota 23.508 jedinica. Poslije Republike Koreje su zemlje Japan, Sjeverna Amerika, Kina i Njemačka [1,4,9,13].

Analizom tabele 2. i slike 6. dolazimo do zaključka da je najveća primjena industrijskih robota u automobilskoj industriji sa 32.702 jedinice na godišnjem nivou u 2010. godini, zatim elektro industrija sa 30.745 jedinica i na trećem mjestu je hemijska industrija sa 10.518 jedinica. Kada je ukupni broj primjene robota u 2010. godini na prvom mjestu je isto automobilska industrija sa 373.223 jedinica ili 39%, na drugom mjestu je elektro/elektronika sa 146.539 jedinica ili 15% i treće mjesto zauzima hemijska industrija sa 101.046 jedinica ili 12%, zatim metalna industrija sa 97.427 jedinica ili 10% i sve ostale industrije koje nisu nabrojane 225.513 jedinica ili 24% od ukupnog broja primijenjenih robota u svijetu. Iz tabele 3. gdje je izvršena analiza samo vodećih zemalja po broju primjene industrijskih robota u različitim industrijskim granama dolazimo do zaključka da je u automobilskoj industriji najviše instalirano robota u Kini, R. Koreji, Njemačkoj, N. Americi i Japanu. U Kini je instalirano 8.300 jedinica što predstavlja 20% od ukupno instaliranih robota u automobilskoj industriji gdje očit ova zemlja povećava investicije u proizvodne kapacitete automobilske industrije, a samim tim i automatizaciju i modernizaciju

proizvodnih procesa što utiče na povećanje primjene robota u proizvodnim procesima. Kad je u pitanju elektro/ elektronska industrija vodeće su u primjeni robota u ovoj industriji R. Koreja, Japan, Sjeverna Amerika i Kina. U hemijakoj industriji prednjače Japan, Sjeverna Amerika, Kina i Njemačka u pogonima proizvodnje plastike i gume. Ako pogledamo primjenu robota u metalnoj industriji vidjećemo da je redosljed sljedeći: Kina, Njemačka, Japan i Sjeverna Amerika.

LITERATURA

- [1] Karabegović I., Doleček V., Primjena robota u 21. stoljeću, *4th International Scientific Conference on Production Engineering RIM 2003*, Bihać, BiH, (ISBN 9958-624-16-8), pp. 3-22, Septembar 2003.
- [2] Karabegović I., Doleček V., Primjena industrijskih robota u automobilskoj industriji, *5th International Scientific Conference on Production Engineering Development and Modernization of Production RIM 2007*, Plitvička jezera, pp.49-50, Oktobar 2007.
- [3] Karabegović I., Karabegović E., Husak E., Comparative analysis of the industrial robot application in Europe and Asia, *International Journal of Engineering & Technology IJET-IJENS*. Vol: 11, No:1, pp.200-2004, Februar 2011.
- [4] World Robotics 2010, United Nations, New York and Geneva, 2010.
- [5] World Robotics 2008, United Nations, New York and Geneva, 2008.
- [6] World Robotics 2006, United Nations, New York and Geneva, 2006.
- [7] Karabegović I., Jurković M., Doleček V., Primjena industrijskih robota u Evropi i Svijetu, *30. Savetovanje proizvodnog mašinstva*, Vrnjačka Banja, SCG, pp.29-45, Septembar 2005.
- [8] Doleček V., Karabegović I., Robotika, Tehnički fakultet Bihać, Bihać, 2002.
- [9] Doleček V., Karabegović I., Roboti u industriji, Tehnički fakultet Bihać, Bihać, 2008.
- [10] Karabegović I., Hodžić D., Application scenario of robot industry, CENT, M2.Br.2.(ISSN1986-5201), pp.31-40, Septembar 2010.
- [11] Doleček V., Karabegović I., Diseminacija robota, uvodni referat, *5th International Scientific Conference on Production Engineering Development and Modernization of Production RIM 2005*, Bihać, pp.3-20, Septembar 2005.
- [12] Karabegović I., Doleček V., Primjena industrijski robota u automobilskoj industriji, *5th International Scientific Conference on Production Engineering Development and Modernization of Production RIM 2007*, Plitvička jezera, pp.49-50, Oktobar 2007.
- [13] Karabegović I., Čatović F., Hodžić D., Industrial Robot Applications in the Process Industries, *12th International Research/Expert Conference Trends in the Development of Machinery and Associated Technology TMT 2008*, Istanbul, Turkey, (ISBN 978-9958-617-41-6), pp. 1317-1321, August 2008.

UPOTREBA GPU U SISTEMIMA ZA DETEKCIJU E-MAIL SPAM-A I IDS USING GPU FOR QUERY OF EMAIL SPAM DETECTION SYSTEMS AND IDS

Aleksandar Sokolovski, Neotel – Macedonia, R&D Department, Skopje, Macedonia
Saso Gelev, Faculty of Electrical engineering, University of Goce Delcev, Stip, Macedonia

Sadržaj: Cilj ovog istraživačkog rada je jedan od najvažnijih aspekata današnjice, a to je bezbednost i menadžment jednog na najvažnijih servisa email i svih srodnih servisi. Ovim radom pokušavamo da iskoristimo sve pogodnosti standardnih metoda baziranih na potpisu u kombinaciji sa algoritmima za mrežnu detekciju napada. Primarni cilj je potvrditi dali će predloženo rešenje biti efektivna strategija za email spam. Glavni smer je odredite najbolju moguću integraciju standardnih metoda za detekciju spam-a sa logikom baziranu na potpisu u kombinaciji sa algoritmima za mrežnu detekciju napada. Glavni zadatak je efektivnije spravljanje sa email spam-om u poređenju sa prethodno ponuđenim rešenjima. Ovo ćemo postići testiranjem efikasnosti predloženog rešenja na NS'3 mrežnom simulatoru i drugih programskih alata.

Abstract: The scope of this research paper is one very important aspects nowadays, the security and management of one of the most important services the email and all of the alike online services today. This paper attempts to investigate the possible benefits of using standard signature-driven spam detection logic in combination with algorithm for network intrusion detection system (NIDS). The primary objective is to verify that proposed solution (standard signature-driven spam detection logic and NIDS) will be an effective strategy for dealing with email spam detection. The main aim is to determine best possible integration of standard signature-driven spam detection logic in combination with algorithm for NIDS, for creating more effective solution for dealing with email spam and priority mail compared to the previous solutions available. This will be achieved by testing the effectiveness of the solution compared to other solutions until today, by using network simulators NS-3 and other programming tools.

1. INTRODUCTION

The email was one of the most important communication tools and it also is one of the most important communication tools nowadays. The email is also one of the most used services on the World Wide Web. According [1] to here are the general data about email:

- In 2012 144 billions emails were send per day worldwide
- In 2012 there were 2.2 billion email users worldwide.
- In 2012 4.3 billion email clients for sending email worldwide
- In 2012 and 69% of worldwide email was spam.
- In 2011 also 71% of email was email spam.

The importance of fast email delivery has change over the year and from being a reliable and fast delivery system, it has become an immediate delivery system. Many enterprises, companies and organization use various techniques and methods for dealing with spam:

- DNS whitelist and blacklists
- Load Sharing: Load is shared among several servers using Layer 4 switches or DNS round-robin.
- DNS Blacklisting: Identify whether the sender is going to send spam or not according to the sender's IP address with a help of DNS.
- Bayesian spam filtering

- Hybrid Filtering

Firewall is hardware or software based network security systems that monitors and controls inbound and outbound network traffic by analyzing the packets.

The first generation of firewalls only filter packets.

The second generation of firewalls are known as statefull, they analyze packets like the first generation (Layer 3 – IP address) but the difference is they operate up to Layer 4 (Ports) of the ISO OSI network model.

The Third generation of firewall are known as application firewall and they work on Layer 7 (ISO OSI).

The Next Generation of firewalls are Intrusion Detection Systems (IDS) monitors network traffic for suspicious activity and alerts in case of intrusion. Intrusion Detection Systems that only detects intrusions are known as passive IDS, the IDS's that detects intrusions and act to prevent the intrusion are known as active IDS or Intrusion Prevention Systems. The IDS's can also be categories depending on the placement of the IDS (host or network) or depending on the way it detects intrusions (signature or anomaly based). Host Intrusion Detection Systems work on the side of the host and monitor inbound and outbound packets. Network Intrusion Detection Systems are monitoring the entire network (LANs or VLANs usually) all traffic inbound and outbound from any host within the network. Signature based IDS monitors' packets within entire network and compares the possible intrusions against a precompiled database of rules or signatures. Anomaly based IDS monitors' packets within

entire network and compares the possible intrusions against an established baseline. The baseline detected “unusual” behavior within the packets and detected intrusions.

Many of the previously mentioned solutions for spam detection and IDS are great and in the beginning when the list is small and manageable work fast and efficient. When the lists becomes larger the processing becomes slower.

In Conclusion the main drawback in today’s email spam solutions, priority mail delivery and IDS solutions is the query processing. In our paper we present solution for the problem using GPU (Graphical processing Units) more specifically CUDA programming. In part 2 and 3 of this paper we present the GPU and CUDA. In part 4 of this paper we explain the DARPA dataset. In part 5 of this paper we present our solution.

2. GPU COMPUTING

GPU (Graphical Processing Unit) computing or the usage of graphical processors for all-purpose computing, began less than 10 years ago. The research in the beginning only allow programming via graphics language, which made in un-flexible. The NVIDIA's CUDA platform according to [2], provides massively multithreaded general-purpose architecture with up to 128 processor cores and thousands of threads. The platform is programmable in the programming language C and capable of hundreds of billions of floating-point operations each second and supports running on all current and older versions of NVIDIA GPUs, this includes the HPC oriented Tesla product line and Kepler core GPUs. The NVIDIA GPU's are sold in millions worldwide which makes them a very good platform for accelerating high-performance computing (HPC) Applications. GPU computing speeds up the real-worlds science and engineering codes up to 10 or 100 times (depending on the GPU's hardware performance). It's domains of usage range from Computer Technology and MRI, computational chemistry, astrophysics to gene sequencing. Here are few examples of CUDA usage: Friedrich-Alexander- Universidad Erlangen-Nurnberg works on biomedical imaging to accelerate CT reconstruction, working on CUDA using the FDK algorithm. University of Illinois works on accelerating advanced MRI reconstruction techniques using CUDA [3]. The last mentioned work focuses on reconstruction for non-Cartesian scan paths, which reduces image-space error from 45% for conventional reconstruction to 12% but has been considered computationally infeasible in practice. This approach uses NVIDIA Quadro FX5600 GPU and works 13-times faster than on Intel Core 2 Extreme quad-core CPU. This means that the reconstruction times is under 2 minutes for 128 volume. This important types of speedup can change science. According to [4] in medicine this types of speedups can make a significant change in clinical practice, for example an analysis in a lab can be done in hours or minutes instead of days. In biomedical imaging the processes also speedups. Another science field where GPU CUDA speeds up the work is astrophysics (mapping of stars, image processing from telescopes like Hubble and other similar tasks). In short, the potential to greatly accelerate computational techniques opens exciting avenues for biomedical imaging research, astrophysics and other fields.

3. CUDA PROGRAMMING MODEL

The key strength of the GPU is its extremely parallel nature. The CUDA programming model enables developers to use the parallelism by using programming code written in C/C++ programming language. The code will run in thousands or millions of parallel invocations (depending on the GPU) or threads. We will present a simple example of Matrix Addition. For adding two N x N matrices on the CPU in C/C++, we would use double nested loop like in the next code:

```
void addMatrix(float *a, float *b,
float *c, int N)
{
int i, j, index;
for (i = 0; i < N; i++) {
for (j = 0; j < N; j++) {
index = i + j * N;
c[index]=a[index] + b[index];
}
}
void main()
{
..... addMatrix(a, b, c, N);
}
```

Example no.1

The same example written in CUDA will be like this. We write one C function, called a kernel function, to calculate one element in the matrix and invokes as many threads to run that function as the matrix has elements. In each thread the kernel runs with a predefined structure called “threadIdx” indicating which of the many threads is running (example no.2):

```
_global_ void addMatrix(float *a,float *b, float *c, int N)
{
int i= threadIdx.x;
int j= threadIdx.y;
int index= i + j * N;
c[index]= a[index] + b[index];
}
void main()
{
dim3 blocksize(N, N),
addMatrix<<<1, blocksize>>>(a, b, c, N);
}
```

Example no. 2

In example no.2 the “_global_” declaration specifier indicates a kernel function that will run on the GPU, the “<<N, N>>” syntax indicates that the “addMatrix()” function will be invoked across a group of threads run in parallel, this is called a thread block. Thread block may be one, two or three dimensional, which provides a natural way to invoke computation across the elements in domains like the following: matrix, fields and vectors.

CUDA makes critical improvements to the core components of running kernel function across many parallel

threads: hierarchical thread blocks, shared memory and barrier synchronization. Next we explain in details.

Thread block can contain up to 512 thread if it is a NVIDIA Tesla architecture GPU, but kernels are invoked on a grid which consists of many thread block that are scheduled independently. In every thread in every block in a grid executes the kernel and then exits. Additional code will be added (example no.3):

```
_global_ void addMatrix(float *a,float *b, float *c, int N)
{
int i=blockIdx.x*blockDim.x+threadIdx.x;
int j=blockIdx.y*blockDim.y+threadIdx.y;
int index = i + j * N;
if ( i < N && j < N)
c[index]= a[index] + b[index];
}
void main()
{
dim3 dimBlock (16, 16);
dim3 dimGrid (N/dimBlk.x, N/dimBlk.y);
addMatrix<<<dimGrid, dimBlock>>>(a, b, c,N);
}
```

Example no.3

In the example above the size of the thread block which was chosen randomly is 256 (16 times 16). The created grid has enough block to have one thread per matrix element, which is same as the example before. All the threads in a sector block run in parallel. Whether multiple thread block can run one after another (in serial mode) or they can run in parallel, it all is dependable on GPU.

In the matrix threads can run independently, no threads needs to know the elements beings accessed by other threads, but when threads need to cooperate (share results the results of computations or memory fetches) in order to be more efficient. CUDA provides shared memory for this purpose. In the shared memory, the kernels can store data (variables or arrays) that is visible to other thread block. Simple example is calculating the sum value of the elements within an array, this can be archived when thread in a block places the element into an array in shared memory (adding the next and the next and so on). All the threads in a block do the same in parallel. They are calculating sum value of elements by cooperating using the shared memory. The shared memory is small because it is on a chip (16K in 2010 Models of NVIIDA GPU), but it is extremely fast, therefore it is perfect for this type of operations like calculating sum value of array elements.

When threads are operating in parallel on same memory it is important to have a mechanism that ensures that one thread will not attempt to read a result before another thread has ended writing it or changing it. CUDA provides the `__syncthreads()` intrinsic function for this purpose. "`__syncthreads ()`" acts as a barrier at which all threads in the block must wait before any are allowed to proceed. [2]

4. TEST DATA SETS

In 1998 and 1999 The Information Systems Technology Group of MIT Lincoln Laboratory with the support of the Defense Advanced Research Projects Agency and the Air Force Research Laboratory (all from the USA), had worked on a new innovative experiment in the field of intrusion detection systems. They had done a cutting edge experiment for the time, creating an Intrusion detection system that monitors the state of an active computer network, looking for some form of attack like denial of service, form of abuse like unauthorized usage, or rear and strange behavior like some forms of so called anomalous behavior.

The experiment was set in a real military base with real computers, but the attack were simulate (it was known what was attack what was a normal connection, this was used later on to evaluate the effectiveness).

The experiment in 1999 (1999 DARPA Intrusion Detection Evaluation Data Set [5]) was small improvement on the experiment done in 1998. In the 1999 the "simulated" attacks lasted 5 weeks, the first and third week was normal traffic, the second week Contained Labeled Attacks. The attacks were divided into five main categories: Denial of Service Attacks, User of Root Attacks, and Remote to Local Attacks, Probes and Data. The full list of attacks is presented on [REF-04]. Then the system was tested with random network packets (some attacks, some normal traffic), there were 201 instances of about 56 types of attacks distributed throughout these two weeks. At the time the main purpose of the experiment was creating the intrusion detection system, but the real "hidden" value of this experiment was the 1998/1999 DARPA Intrusion Detection Evaluation Data Set.

The collected data from this are made available for all the researchers that needed a test data set for their intrusion detection system. This data set had made possible the creation of many future intrusion detection systems. Proof of the value of this data set is the number of publications using this data set in their research project. This is the reason why we intended to use the 1998/1999 DARPA Intrusion Detection Evaluation Data Set. [5]

The 1998/1999 DARPA Intrusion Detection Evaluation Data Set, is a very data set containing around 4-5 GB of data, our main purpose was testing our intrusion detection genetic algorithm, so in order to minimize the time for analyzing the data set and maximizing the testing type, we used the optimized versions of the DARPA data set, the KDD CUP 99 Data Set. KDD CUP 99 Data Set is compiled from the 1998 DARPA Intrusion Detection Evaluation Data Set. The database contains normal connections and 24 types of attacks, the types of attacks are presented on [5]. Evaluation on the KDD CUP 99 Data Set and Summary report with type of attacks and number of connections for each type of attacks is presented on the table in [6]. For this paper and our algorithm it is very important to present the KDD CUP 99 Data Set Schema properly and precisely, for this we will use the tables from the tasks for the KDD CUP 1999 [6].

5. ALGORITHM AND IMPLEMENTATION

In [7] we presented signature based IDS which can dynamically allocate the number of rules that are kept in the

detection system based on the LAN complexity and number of packets in the network at the moment. The idea is to process and analyze the packet in the distributed agents IDS before the time of arrival at the destination computer and processing the packet by the host computer. This is done using complex mathematical mechanism for calculating the packet travel time. Using Least Recently Used algorithm to remove the not used rules are after the buffer of rules in the DB of the distributed agents is over its pre-defined limit. This makes the hosts in the LAN protected with intrusion detection and prevention for the more recent and frequent type of attacks, but leaves the hosts vulnerable to new attacks (that are not in the current buffer of rules in the DB of the distributed agents). The new attacks will be detected with deeper analyses at the IDS server side (after analyzing the whole “historical” packet in the LAN) and the rule for that attack will be added in the buffer of rules in the DB of the distributed agents. The main problems is that the first new type of attack or intrusion will be only detected but not prevented. In order to increase the efficiency we need to keep more rules in the buffer of rules in the DB of the distributed agents. This can be done by using CUDA programming. To explain the process of using CUDA in IDS analyses, we will use simple matrix (matrix no.1).

Matrix no.1: [1 4 5]; [1 -1 -1]; [9 7 -1].

Firstly in order to be processed by GPU the matrix must have same number of elements in every row. In order to do that in the shared memory (of GPU) every thread adds the number 1 for every element in the row. Then we use the Max_Number() function, that way we calculate the most elements per row, and then the value is placed (MAX_NUMBER) in the shared memory in GPU. Afterward we process the matrix using thread for each row if there is a missing number of elements (compared to the compared to the MAX_NUMBER) we add “empty element” (the number -1 is ignored in processing). Then we get this matrix (matrix no.2). Adding the “empty element” must be done if we want be process the data with CUDA.

matrix no.2: [1 4 5]; [1 -1 -1]; [9 7 -1].

This in our simplistic example representing the buffer of rules in the IDS distributed agent. Then we and packet s from the network traffic (matrix no.3) in the shared memory, pre-process them (adding “empty element” if the number of elements is lower the MAX_NUMBER).

The last step is comparison of the DB IDS rules (matrix no.2) with the packets from the network traffic recorded (array [7, 9,-1]) placed in the shared memory of the GPU in order to be accessed by every thread at once. This speedup the comparison by 10 times at least.

6. RESULTS

The system for testing is the following: One PC: (Intel® Core™ i7-3770K with 8M Cache @ 3.90 GH; Mobile Intel® QM77 Express Chipset; 16 GB DDR3 @ 1600 MHz; HDD 500 GB @ 7200 RPM; GPU: NVIDIA Quadro K2000M - 384 CUDA CORES), RHEF Linux 6 with C/C++, NS-3 and Windows 8.1 with C# (VS 2010).

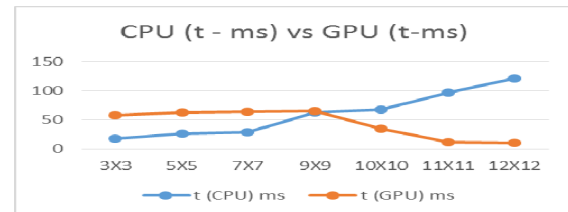


Figure No.1 CPU vs GPU compared (time – ms)

From figure no.1 we can clearly see that GPUs work much faster with larger data sets, compared to CPUs. Therefore they are ideal candidates for using them as main processing power for big data analytics (IDS, email spam).

7. CONCLUSION

NVIDIA GPUs provide massive computation potential in the field of computer science. Our proposal is a new type of using the GPU in the field of computer science more specifically in the field of system administration. Our Future work includes creating an interface for data exchange between CUDA and OPENGL in order to use Intel GPUs as well, also we will try to create dataset of mail for testing email spam (like DARPA DATA for IDS).

REFERENCES

- [1] Internet 2012 in numbers - Royal Pingdom U.K., <http://royal.pingdom.com/2013/01/16/internet-2012-in-numbers/>
- [2] NVidia CUDA platform, <http://www.nvidia.com/>
- [3] W. Jeong, P. T. Fletcher, R. Tao, and R.T. Whitaker, “Interactive Visualization of Volumetric White Matter Connectivity in DT-MRI Using a Parallel-Hardware Hamilton-Jacobi Solver, 2007.
- [4] F. Xu and K. Mueller. “Real-Time 3D Computed Tomographic Reconstruction Using Commodity Graphics Hardware,” Medicine and Biology, 2007.
- [5] MIT Lincoln Labs DARPA DATA SET 1998/1999 http://www.ll.mit.edu/mission/communications/cyber/CS_Tcorpora/ideval/data/
- [6] KDD Cup 1999 Data <https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>
- [7] Aleksandar Sokolovski, Saso Gelev ALGORITHM FOR DISTRIBUTED AGENT BASED NETWORK INTRUSION DETECTION SYSTEM (NIDS), CIIT 2011 Bitola, Macedonia.

PRIMJER UPRAVLJANJA RAČUNARSKIM SISTEMOM PUTEM GRUPNIH POLISA AN EXAMPLE FOR CONTROL OF INFORMATION SYSTEM BY GROUP POLICY

Aleksandar Vučeraković, *Centralna banka Crne Gore*

Sadržaj: *Predmet ovog rada je prezentovanje grupnih polisa kao jednog od važnih alata za administriranje računarskih sistema koji su bazirani na Windows platformi. Nakon analize funkcionalnosti, opsega i načina procesuiranja grupnih polisa, dat je praktični primjer impementacije ovog alata. Na kraju rada dat je akcenat na planiranju primjene grupnih polisa pri administriranju informacionog sistema.*

Abstract: *The subject of this paper is the presentation of Group Policy as one of the important tools for the administration of computer systems that are based on the Windows platform. After analyzing the functionality, scope and method of processing Group Policy, is shown a practical example of the implementation of this tool. At the end of the paper is given the emphasis on planning group policies in the administration of information systems.*

1. UVOD

Razvijanjem i unapredjenjem informacionih sistema povećavao se broj radnih stanica koje treba administrirati, podešavati i konfigurirati. Bilo je potrebno ići od lokacije do lokacije i posebno podešavati svaku radnu stanicu zasebno. Takav pristup održavanja jednog informacionog sistema bio je veoma naporan i iziskivao je puno vremena i radne snage. Zbog svih problema koje je donosio ovakav pristup održavanja informacionog sistema počelo se sa razvijanjem udaljenih metoda upravljanja.

Razvojem operativnih sistema kao što je Windows u sve većoj mjeri se išlo ka centralizovanim i udaljenim metodama upravljanja informacionim sistemom. Za takav pristup upravljanja postojalo je mnogo razloga. U tom pravcu se išlo zbog sve većeg širenja informacionih sistema, zbog lakšeg načina upravljanja sistemom sa jednog mjesta, zbog bezbjednosti sistema i zbog sve većeg auomatizovanja sistema.

Razvijanjem Windows operativnog sistema došlo se do rešenja razvijanjem metode server-klijent gdje bi se sa servera upravljalo svim klijentima. Server je Aktivni direktorijum, a njegov najbolji alat za udaljeno i centralizovano upravljanje klijentima su grupne polise.

Koristeći se [1], [2], [3] i [4], kao i sopstvenim iskustvom autora u radu će biti detaljno objašnjeno funkcionisanje grupnih polisa. Biće riječi o tome zašto su grupne polise toliko bitne za upravljanje jednim informacionim sistemom, kako se kreira objekat grupne polise, kako se konfiguriraju grupne polise. Govoriće se o tome na koje se sve objekte Aktivnog direktorijuma primjenjuju grupne polise, odnosno o opsegu grupnih polisa, kao i o načinu procesuiranja grupnih polisa i o njihovoj

implementaciji sa posebnim naglaskom na praktičnoj implementaciji. Posljednja tema rada biće planiranje primjene grupnih polisa, odnosno najbolji načini iz prakse za podešavanje grupnih polisa i za njihovu implementaciju.

2. GRUPNE POLISE

Objekti Aktivnog direktorijuma kojima se u najvećoj mjeri manipuliše korišćenjem grupnih polisa su: korisnici (*Users*), kompjuteri (*Computers*) i grupe (*Groups*).

Svi navedeni objekti: *Users*, *Computers* i *Groups* se mogu grupisati u organizacione jedinice zbog lakšeg upravljanja tim objektima. Na organizacione jedinice se mogu primijeniti grupne polise.

Grupnim polisama se mogu odrediti posebna podešavanja za cijeli domen, sajt ili za određene grupe korisnika ili računara.

Mogu se napraviti specifična podešavanja koja će se kasnije primijeniti na određenu grupu računara ili korisnika. Upotrebom grupnih polisa može se instalirati određeni softver na udaljenom računaru za potrebe korisnika, mogu se konfigurirati skripte za startovanje sistema ili gašenje sistema računara kao i za prijavljivanje ili odjavu korisnika. Grupnom polisom se može podesiti i bezbjednosna konfiguracija računara kao što je politika lozinki. Takođe se može raditi i redirekcija foldera tako što će korisnik umjesto lokalno na računaru svoje podatke čuvati u nekom folderu koji se nalazi na nekom serveru u mreži. Konfiguracija internet pretraživača *Internet explorer* se isto tako može podešavati grupnom polisom. Podešavanjem *registry* baze se preko administrativnih obrazaca može podesiti radno okruženje korisnika. Grupnim polisama je moguće blokirati ili omogućiti korišćenje određenih aplikacija i resursa računara kao što su USB il CD ROM.

Kompjuteri koji na sebi imaju neki *Windows* operativni sistem počev od *Windows-a* 2000 pa na dalje imaju lokalnu

grupnu polisu kojom možemo upravljati da bi podesili konfiguraciju kompjutera. Lokalna polisa postoji nezavisno od toga da li je računar u domenu ili nije i ona utiče samo na podešavanje računara na kojem je smještena.

Domenske polise se kreiraju u Aktivnom direktorijumu i čuvaju se na domen kontrolerima. Koriste se da se centralizovano upravlja korisnicima i kompjuterima u domenu. Kada se server 2008 proglašava za domen kontroler odnosno kada se instalira uloga *Active Directory Domain Services* (ADDS) predefinisano su kreirane dvije polise i to: *Default Domain Policy* i *Default Domain Controllers*.

GPO ima dvije komponente *Group Policy Container* (GPC) i *Group Policy Template* (GPT). GPC je objekat Aktivnog direktorijuma i čuva se u *Group Policy Objects* kontejneru i predstavlja osnovni atribut GPO, ali ne sadrži podešavanja. Podešavanja se čuvaju u GPT koji se čuva u `%SystemRoot%\SYSVOL-u\Domain\Policies\GPOGUID`.

Kreiranje polise vrši se tako što se u GPMC konzoli desnim tasterom miša klikne na *Group Policy Objects* kontejner i pritisne *New*. Da bi se mijenjala kreirana polisa potrebno je desnim tasterom miša kliknuti na nju i odabrati opciju *Edit* i otvara nam se prozor za editovanje polise GPME.

GPME prikazuje mnoštvo podešavanja polisa koji su dostupni u GPO-u. Ta podešavanja su organizovana hijerarhijski. Prva podjela je napravljena tako što su odvojena kompjuterska i korisnička podešavanja pa u GPME postoje dva dijela i to *Computer Configuration* dio i *User Configuration* dio.

Polisu zatim treba povezati sa domenom, sajtom ili organizacionom jedinicom tako što se desnim tasterom miša klikne na željenu lokaciju i izabere opcija *Link An Existing GPO*.

Grupne polise mogu biti vezane za sajt, domen ili organizacionu jedinicu u Aktivnom direktorijumu što znači da sajt, domen ili organizaciona jedinica postaju opseg polise pa će se polisa primijeniti na sve kompjutere i korisnike unutar opsega uključujući i organizacione jedinice koje se nalaze na nivoima ispod posmatrane organizacione jedinice. Jedna grupna polisa može biti povezana sa više sajtova ili organizacionih jedinica.

Da bi se kreirao link potrebno je desnim tasterom miša kliknuti na organizacionu jedinicu koja će biti opseg polise i izabrati opciju *Link an Existing GPO*. Ako polisa još nije kreirana može se kreirati i povezati na organizacionu jedinicu jednim putem i to tako što se desnim tasterom miša klikne na organizacionu jedinicu i izabere opcija *Create A GPO IN This (Domain | Site | OU) And Link It Here*.

Kada je podešavanje polise postavljeno na *Enable* ili *Disable* u GPO-u sa najvećim prioritonom konfigurisano podešavanje se aktivira. Ako je polisa podešena na *Not Configured* polise sa manjim prioritonom će se redom aktivirati. Domen, sajt ili organizaciona jedinica mogu imati više polisa povezanih na sebe. Redosljed linkova grupnih

polisa određuje prioritet grupnih polisa. GPO sa najvećim brojem linka preuzima prioritet nad polisama koje imaju manji redni broj.

Osnovno podešavanje grupnih polisa je to da se polise nasljeduju sa kontejnera na koje su povezane na kontejnere koji su na nižem nivou. Nasljeđivanje se može blokirati. Polisa se može i forsirati i tada ona dobija najveći prioritet i preovladava nad svim drugim polisama pa čak i poništava opciju blokiranja nasljeđivanja.

Opseg se može definisati i putem filtriranja i to pomoću *security filtera* koji određuju globalne bezbjednosne grupe i *WMI filtera* koji određuju opseg na osnovu karakteristika sistema.

3. PROCESUIRANJE GRUPNIH POLISA

Kada počinje primjenjivanje grupnih polisa bilo da se polisa prvi put primjenjuje ili se obnavlja servis koji je pokrenut na *Windows* sistemima (*Group Policy Client*) određuje koja se polisa odnosi na kompjutere, a koja na korisnike. Povlače se sve polise koje nijesu već keširane. Zatim se kroz seriju procesa nazvanih *client-side extensions* obavlja prevođenje tih polisa i primjena polisa odnosno prave se promjene na lokalnom računaru ili trenutno prijavljenom korisniku. Svaka važnija kategorija podešavanja polise ima svoj CSE kao što su bezbjednosni CSE ili CSE koji pravi promjene u registru. Važno je znati da server ne šalje polise klijentima već da klijenti povlače polise sa servera puštajući CSE-ove da vrše promjene lokalno na računaru.

Kompjuterske polise se primjenjuju prilikom startovanja sistema računara, a zatim na svakih 90 do 120 minuta. Korisničke polise se primjenjuju prilikom prijavljivanja korisnika i onda na svakih 90 do 120 minuta. Osvježavanje polise se može forsirati korišćenjem *GPUpdate* komande.

4. PRIMJER IMPLEMENTACIJE

Za potrebe HelpDesk-a, odnosno odjeljenja koje se bavi podrškom korisnicima u radu sa informacionim sistemom organizacije neophodno je bilo da na svim radnim stanicama u informacionom sistemu bude postavljen jedan lokalni administratorski nalog koji bi se koristio za intervencije na računarima kada se nebi moglo prijaviti na radne stanice sa nalozima sa domena koji imaju administratorske privilegije na radnim stanicama, odnosno kada je domen nedostupan.

Odlučeno je da se taj zahtjev odradi tako što će se napraviti grupna polisa koja će skriptu koja kreira administratorski nalog, na primjer nalog *acoadm*, isporučiti i pokrenuti na svim radnim stanicama. Domen administratori su dobili skriptu koja je prevedena u *exe* fajl pomoću besplatnog alata *Bat to Exe Converter* i koja sadrži sledeće linije koda:

```
@echo off
net user /add acoadm Test#123
net user acoadm Test#123
net user acoadm /fullname:"Aco Vucerakovic"
```

net localgroup administrators /add acoadm

Exe fajl je iskopiran na putanji *C:/Windows/SYSVOL/domain/Scripts* zato što je SYSVOL folder predefinisano *share-ovan* za sve članove domena i *Authenticated Users* imaju privilegije *Read & Execute, List Folder Context i Read* tako da svi članovi domena imaju pristup skripti koja je postavljena na već datoj putanji.

Sada se može napraviti grupna polisa koja će skriptu isporučiti i pokrenuti na svim radnim stanicama u domenu. Potrebno je otvoriti GPMC, kliknuti desnim tasterom miša na naziv domena i odabrati opciju *Create a GPO in this domain and Link it here*. Nakon toga se pojavljuje prozor gdje treba ukucati naziv polise. Polisa je nazvana *local_administrator*.

Pošto je polisa kreirana potrebno je desnim tasterom miša kliknuti na samu polisu i izabrati opciju *Edit*, zatim proširiti dio *Computer Configuration*, proširiti dio *Policies*, proširiti dio *Windows Settings* i odabrati *Scripts*. Zatim duplim lijevim tasterom miša treba kliknuti na *Startup* i dobija se prozor u kojem se na opciju *Add* doda skripta sa lokacije *C:/Windows/SYSVOL/domain/Scripts* koja treba biti primijenjena na sve radne stanice u domenu. Kada je skripta dodata potrebno je pritisnuti *OK* i polisa je kreirana i podešena. Prilikom sledećeg startovanja radnih stanica primijenice se polisa i na svakoj radnoj stanici u domenu će postojati lokalni administratorski nalog *acoadm*.

Nakon nekog vremena primijećeno je da se na korisničkim radnim stanicama nalaze softveri koje niko iz IT sektora nije instalirao na tim radnim stanicama i otkriveno je da su na neki način korisnici uspjeli sami da instaliraju te softvere. Detaljnom istragom utvrđeno je da su korisnici pronašli način da otkriju lozinku lokalnog administratorskog naloga *acoadm* i tako koristeći taj nalog instaliraju softvere na radnim stanicama.

Korisnici su lozinku uspjeli da otkriju koristeći komandnu liniju i pretragu skrivenih foldera sistema i to na sledeći način:

- Prvo su u komandnoj liniji kucali *assoc .bat = "bilo šta"*
- Zatim su pristupili *share-ovanom* folderu *SYSVOL*

Aktivnog direktorijuma i od tamo pokrenuli skriptu *local_administrator* gdje im se pokazala putanja gdje se skripta trebala izvršiti

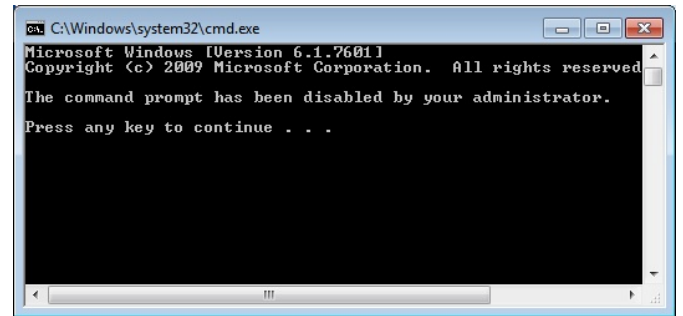
- Pretragom skrivenih fodera došli su do mjesta gdje se skripta trebala izvršiti, otvorili skriptu i pročitali lozinku.

Nakon ovog incidenta administratori Aktivnog direktorijuma su odlučili da svim korisnicima zabrane korišćenje komandne linije i pretragu skrivenih foldera tako što će napraviti polisu koja će zabraniti korišćenje komandne linije i upotrebu *Folder and search options*.

Polisa se kreira tako što se u GPMC pritisne desnim tasterom miša na naziv domena i izabere opcija *Create a GPO in this domain and Link it here*. Zatim se pojavi prozor gdje se ukuca naziv polise. Polisa je nazvana *disable_cmd_and_folder_options*.

Potrebno je desnim tasterom miša kliknuti na kreiranu polisu i odabrati opciju *Edit*, a zatim proširiti stablo do *User Configuration/Policies/Administrative Templates/System*, duplim lijevim tasterom miša kliknuti na *Prevent access to the command prompt*, odabrati opciju *Enabled* i pritisnuti *OK*.

Ovim dijelom polise napravljeno je podešavanje za blokiranje komandne linije svim korisnicima domena. Kada korisnici pokrenu komandnu liniju dobiće obavještenje da je onemogućena od strane administratora što je prikazano na slici 4.1.

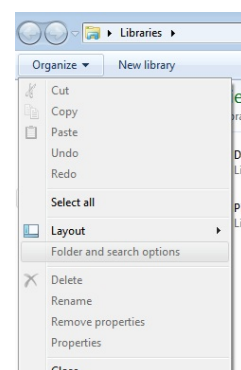


Slika 4.1

Sada polisu treba podesiti tako da zabranjuje korišćenje opcije *Folder and search options*.

Potrebno je proširiti stablo polise do dijela *User Configuration/Policies/Administrative Template/Windows Components/Windows* i duplim lijevim tasterom misa kliknuti na *Remove the Folder Options menu item from Tools menu* kada se dobija prozor gdje odabere opcija *Enabled* i pritisne *OK*.

Kada korisnici pokušaju da otvore *Folder and search options* pojaviće im se zatamljena opcija *Folder and search options* što je prikazano na slici 4.2.



Slika 4.2

Ovom polisom se postiglo da se prilikom sledećeg prijavljivanja korisnika ili osvežavanja korisničkih polisa korisnici onemogućće da koriste komandnu liniju i *Folder and search options*. Postignuto je da korisnici samim tim što ne mogu koristiti ove dvije finkcije ne mogu ni otkriti lozinku lokalnog administratora.

Kroz prikazani scenario objašnjeno je kreiranje i podešavanje grupnih polisa kako kompijuterskih tako i korisničkih. Prikazani primjer je dobar primjer kako se mogu

rješavati problemi putem grupnih polisa i kako se može stvoriti problem implementacijom grupnih polisa ako se ne predvide sve posljedice koje polisa može da proizvede. Problem koji je nastao implementacijom grupne polise koja kreira lokalni administratorski nalog administratori domena nijesu predvidjeli, ali su ga relativno lako riješili takođe implementacijom druge polise. Poenta je u tome da prije implementacije treba sve dobro istestirati, isplanirati i predvidjeti da bi se sa što manjim brojem polisa postigao željeni rezultat.

5. PLANIRANJE IMPLEMENTACIJE I PREPORUKE ZA IMPLEMENTACIJU GRUPNIH POLISA

Dobar dizajn grupnih polisa obezbjeđuje lakše upravljanje i implementaciju polisa i omogućuje lakše naknadne promjene u grupnim polisima. Ako se implementacija grupnih polisa dobro isplanira omogućeno je i lakše praćenje objekata Aktivnog direktorijuma koji su pod opsegom polisa kao i lakša nadogradnja polisa. Da bi se napravio dobar plan za implementaciju grupnih polisa potrebno je:

- sakupiti i analizirati informacije koje su potrebne da se olakša dizajn polisa
- napraviti dizajn polisa i implementirati ga
- napraviti dizajn procesuiranja polisa
- isplanirati upravljanje polisama

Sakupljanje informacija o organizaciji Aktivnog direktorijuma je prvi korak u planiranju implementacije grupnih polisa. Prije dizajniranja okruženja grupnih polisa mora se imati jasna ideja o tome šta vaša organizacija očekuje od grupnih polisa. Takođe je potrebno da se vrlo dobro upozna sa strukturom Aktivnog direktorijuma i kako se može unaprijediti struktura Aktivnog direktorijuma implementacijom grupnih polisa. Moraju se prikupiti informacije o organizaciji AD-a, informacije koje se tiču bezbjednosnih zahtjeva, informacije koje se tiču upravljanja desktop okruženjem i informacije o administrativnim procesima.

Broj objekata grupne polise i kompleksnost svakog od njih utiče na dizajn i implementaciju grupnih polisa. Polise bi trebalo kreirati i imenovati po njihovoj funkciji ili po resursima u organizaciji ili možemo koristiti oba modela.

Grupne polise povećavaju vrijeme potrebno za startovanje sistema kompjutera i za prijavljivanje korisnika. Manjim brojem grupnih polisa dobija se na brzini korisnika koji radi. U praksi je najbolje napraviti generalne polise koje se odnose na sve računare i korisnike u organizaciji, a zatim pojedinačne polise za posebna podešavanja određenih kompjutera ili korisnika.

Uvijek treba testirati podešavanje u nekom testnom okruženju prije nego se postavi polisa u produkciju. Kada se utvrdi da nema problema u radu polise i uvidi da je postignut

željeni efekat grupna polisa se instalira u produkcionom okruženju.

Tokom dizajniranja infrastructure grupnih polisa treba uzeti u obzir nasljeđivanje grupnih polisa. Objekte grupne polise treba povezati sto višoj u strukturi organizacionih jedinica. Blokiranje nasljeđivanja treba koristiti u što manjoj mjeri. Forsiranje grupnih polisa treba koristiti samo u kričnim situacijama. Potrebno je razmotriti korišćenje filtriranja grupnih polisa i to na što višojem nivou. Potrebno je dokumentovati nasljeđivanje grupnih polisa, blokiranje, filtriranje i forsiranje.

GPMC se može koristiti da se odradi *backup* ili *restore* objekta grupne polise. *Backup* omogućuje čuvanje svih informacija koje sadrži objekat grupne polise. *Backup* može kasnije da se koristi da se povrati polisa koja je izbrisana, da se iz *backup-a* vrata podešavanja u neku polisu ili da postojeću polisu koja je *backup-ovana* vratimo u prethodno stanje. Prilikom instaliranja novog okruženja Aktivnog direktorijuma potrebno je migrirati objekte grupne polise, podešavanja i hijerarhiju objekata grupne polise sa starog u novo okruženje. Postoje dva tipa migracije objekata grupnih polisa: migracija preko testnog okruženja i migracija sa starog okruženja u produkciju novog okruženja.

6. ZAKLJUČAK

U radu su prikazane grupne polise kao alat za udaljeno i centralizovano upravljanje informacionim sistemom njihova svojstva za administriranje, način planiranja dizajniranja grupnih polisa i na praktičnom primjeru je prikazano kako se kreiraju i konfiguriraju grupne polise i kako putem njih možemo rešavati probleme koji nastaju u radu informacionog sistema. Svaka veća organizacija koja brine o bezbjednosti svojih podataka ima potrebu za imlementacijom grupnih polisa i na taj način udaljeno i centralizovano upravlja svojim informacionim sistemom.

LITERATURA

- [1] 70-640 Configuring Windows Server 2008 Active Directory, Dan Holme, Nelson Ruest, Danielle Ruest, Jason Kellington
- [2] 6436A Designing a Windows Server 2008 Active Directory Infrastructure and Services
- [3] Group Policy: Fundamentals, Security, and the Managed Desktop, Jeremy Moskowitz
- [4] Microsoft Windows Group Policy Guide, Darren Mar-Elia, Derek Melber, William Stanek, MS Group Policy Team

PRIMJER ADMINISTRIRANJA HETEROGENOG RAČUNARSKOG SISTEMA RTCG EXAMPLES OF ADMINISTRATION HETEROGENEOUS COMPUTING SYSTEM RTCG

Igor Miljanić, Radio Televizija Crne Gore - Podgorica

Sadržaj: U radu je prikazana analiza postojećeg stanja u kompaniji koja nije vršila plansku implementaciju informacionog sistema. Biće prikazana struktura operativnih sistema. Analizirani su sigurnosti problemi u sistemu. Date su konkretne preporuke za unapređenje bezbjednosti sistema.

Abstract: This work presents an analysis of the current situation in the company, which did not perform the planned implementation of the information system. It will be shown in the structure of the operating system. We analyzed the security problems in the system. Given specific recommendations for improving the security system.

1. UVOD

Problem zaštite postoji kod svih računarskih sistema, a posebno kod sistema koji su povezani na Internet. Pitanje zaštite je postalo jedno od najvažnijih pitanja u svakom poslovnom sistemu. Operativni sistem ima značajnu ulogu u rješavanju problema zaštite.

U radu je prikazana analiza postojećeg stanja računarske opreme, aktuelni način administriranja operativnih sistema u firmi u kojoj radim kao glavni IT administrator od 2003. godine. Cilj rada je ukazivanje na bezbjednosne propuste u računarskom sistemu. Date su i konkretne preporuke za unapređenja sistema. Ovaj rad ima za cilj i da skrene pažnju na primjere propusta viđenih u praksi i na taj način doprinese smanjenju grešaka lica koja bitno utiču na funkcionisanje sistema, a prije svega IT administratora.

Apsolutna zaštita računarskih sistema se ne može ostvariti. Osnovni cilj je obezbijediti visok nivo zaštite i zato pristup rješavanju problema zaštite mora biti sveobuhvatan sa stalnim razvijanjem novih mehanizama zaštite u skladu sa bezbjednosnim problemima koji nastaju.

2. ANALIZA POSTOJEĆEG STANJA OPREME U TELEVIZIJI CRNE GORE

Cilj ovog poglavlja je dati analizu postojećeg stanja opreme u TVCG, prikazati aktuelni način administriranja operativnih sistema i otkriti sigurnosne propuste u sistemu.

2.1. STRUKTURA OPERATIVNIH SISTEMA U TVCG

Postojeće stanje računarske opreme u Radio Televiziji Crne Gore je heterogeno. Ima mnogo i starih računara kao što su Pentium III i Pentium IV sa instaliranim Windows 95, 98, XP operativnim sistemom, kao i računara novije generacije

sa I3, I5 i I7 procesorima sa instaliranim Windows 7 i Windows 8 operativnim sistemom.

Računari su podijeljeni i u ove dvije grupe. Prva grupa računara su kancelarijski računari kojih ima najviše u zgradi i koje koriste novinari, redakcije, urednici za pravljenje tekstova i za surfovanje internetom. Druga grupa računara su računari koji se koriste za pravljenje programa, kao što su računari za linearnu i nelinearnu montažu, računari za obradu audio zapisa sa raznih audio snimača, računari koji se koriste za konvertovanje video materijala sa enkodera i kamera u odgovarajućem formatu za emitovanje. U tabeli 1 se vidi prikaz broja i strukture računara u zgradi TVCG po spratovima i ukupan broj računara koji su na internetu.

TVCG	UKUPNO RAČUNARA	RAČUNARI NA INTERNETU
PRIZEMLJE	6	0
1. SPRAT	21	16
2. SPRAT	7	7
3. SPRAT	30	21
4. SPRAT	47	46
5. SPRAT	36	36
6. SPRAT	21	16
RADIO	43	37
UKUPNO	211	179
NA INTERNETU SE NALAZI 85% RAČUNARA		

Tabela 1. Statistika ukupnog broja i strukture računara

U tabeli 1 vidi se da je ukupan broj računara 211. Mjesečno se dodaju oko dva nova računara. Na internetu se nalazi trenutno 85% od ukupnog broja računara, s tim što se i ta brojka svakim novim mjesecom povećava u zavisnosti od potreba korisnika i odobrenjem tehničkog direktora za umreženjem. U tabeli 2 je prikazana struktura operativnih sistema koji se koriste u TVCG.

OPERATIVNI SISTEM	BROJ RAČUNARA
WINDOWS 95	3
WINDOWS 98	5
WINDOWS XP	139
WINDOWS 7	50
WINDOWS 8	9
LINUX DEBIAN	3
WINDOWS SERVER 2003	2

Tabela 2. Operativni sistemi

Iz tabele 2 se vidi da je najviše zastupljen i dalje Windows XP operativni sistem, zatim Windows 7, Windows 8 i vrlo mali broj Linux distribucija. U televiziji dominira heterogeni sistem u pogledu hardvera i operativnih sistema.

Što se tiče ostale opreme, treba napomenuti da TVCG raspolaže sa oko stotinu vrsta štampača.

Računari su međusobno povezani svičevima. Za svaki sprat se nalazi po jedan svič na koji su povezani računari i koji vode do glavnog sviča koji se nalazi na VI spratu. Marke i tipovi svičeva su razni (Riverstone, Dlink, itd). Topologija mreže je zvijezda.

Što se tiče interneta, u TVCG se od skoro koristi optika veće brzine. Brzine optike su date u tabeli 3 sa tim što se zadržala i iznajmljena linija koja je do 2011. godine bila i linija preko koje se obavljao čitav internet saobraćaj. Početna brzina interneta preko iznajmljenije linije je bila 2Mbps upload i 2Mbps za download, dok se sada brzina preko optike kreće od 25Mbps do 45Mbps za download i 2Mbps za upload. U tabeli 3 su date brzine interneta preko optike.

Vlakno	Brzina
Linija 0	Iznajmljena linija 4Mbp
Linija 1	35Mbps/2Mbps
Linija 2	35Mbps/2Mbps
Linija 3	45Mbps/2Mbps
Linija 4	45Mbps/2Mbps
Linija 5	25Mbps/2Mbps
Linija 6	35Mbps/2Mbps

Tabela 3. Brzina optike

2.2. AKTUELNI NAČIN ADMINISTRIRANJA OPERATIVNIH SISTEMA

Stanje operativnih sistema u TVCG je na zadovoljavajućem nivou, prvenstveno zbog jedne aplikacije pod nazivom [Deep Freeze](#). U pronalaženju i testiranju aplikacije je učestvovao IT sektor. Aplikacija je pronađena na zahtjev korisnika i njihovih čestih poziva kojima je najviše izražavana molba za intervencijom zbog virusom oštećenog sistema, a čije je pojavljivanje najčešće bila posledica djelovanja ljudskog faktora.

Kao što se iz naziva aplikacije '[deep freeze](#)' može zaključiti, radi se o programu koji kada se instalira na nekom računaru zamrzne sistem u stanju u kakvom se nalazi u tom trenutku, pod pretpostavkom da je instaliran na računar koji nije inficiran virusom. Kada se korisniku zbog bilo kojeg

razloga pojavi virus sa interneta ili sa fleške, stabilnost operativnog sistema je sačuvana i ne dovodi se u pitanje, jer posle restarta računara operativni sistem se vraća na prethodno stanje kada je sistem bio u redu.

Ova aplikacija je zaslužna za smanjenje broja problema u radu operativnih sistema, a njenom aktivacijom je ostvaren odličan rezultat, naročito zbog toga što su potrebe za intervencijama i čestim reinstaliranjima drastično smanjene .

Međutim, treba napomenuti da ovakav sistem zaštite operativnih sistema u TVCG, pored prednosti, posjeduje i svoje nedostatke. Najveći nedostatak ove aplikacije je u tome što nakon restarta operativnog sistema onemogućava korisniku ponovno nalaženje svojih podataka na potpuno zamrznutoj particiji (npr. u kreiranim fajlovima u My Documents ili Desktop-u), budući da su ti podaci trajno izgubljeni. Dakle, sa gubitkom potencijalnog virusa istovremeno bivaju izgubljeni i regularni fajlovi korisnika, u koje je uračunata i e-pošta korisnika, kao i sitna podešavanja sistema, kao što su podešavanje jezika tastature, promjena slike pozadine ekrana itd. Upravo ovi problemi postanu uzrok mnoštva negativnih reakcija od strane korisnika.

Problem je riješen tako što su od fizičkog HDD-a napravljene dvije logičke particije - jedna systemska na kojoj će biti instalirana i aplikacija za zaštitu operativnog sistema, i druga particija na koju korisnik može da smješta podatke, a da nema strah od gubitka podataka posle restarta računara.

Osim rešenja podjele particija, koristi se i aplikacija za finalno podešavanje sistema pod imenom 'Tweak UI'. Ona sadrži mnoštvo opcija za detaljno podešavanje sistema kao sto su: pomjeranje lokacije za My Documents, pomjeranje lokacije za Desktop, čuvanje korisničke pošte na drugoj particiji, čuvanje korisničkog bookmarka također na drugoj particiji itd. Ovo podešavanje se obično odradi prije instalacije aplikacije za zaštitu sistema. Naravno, aplikacija za zaštitu operativnih sistema ima opciju da se uz odgovarajuću administratorsku šifru program trenutno ili stalno isključi kako bi se dozvolilo administratoru da instalira apdejte, pačeve i zakrpe određenog operativnog sistema.

Naravno da se vodi računa o ažuriranju sistema za svaki operativni sistem posebno. Da nema redovnog ažuriranja sistema, bilo koji virus bi se lakše pojavio u sistemu i računar bi se brže zarazio. Periodično se ažuriraju i ostale aplikacije na računarima, kao sto su razni browseri, ažuriranje posljednjih definicija antivirusnog programa, itd.

Ako i pored svih mjera zaštite operativnog sistema ipak dođe do njegovog oštećenja, taj računar zahtijeva odnošenje na servis kako bi bila urađena nova instalacija sa prethodnim čuvanjem korisničkih podataka (dokumenta, pošta, imenik, bokmarks...).

Što se tiče servera, trenutno su prisutna dva, na kojima se nalazi Windows Server 2003 operativni sistem, a na njemu opet postoji nekoliko aktivnih servisa.

Na prvom serveru je podignut Web server na kojem se nalazi interni sajt televizije koji koristi tehnika. Konkretno, sajt koriste dva sektora IT sektor i Broadcast IT sektor. Sajt u

najvećoj mjeri služi za unošenje izvještaja za dnevne intervencije kolega nekog od ova dva IT sektora, da bi ostale kolege imale uvid u to šta se određenog dana dešavalo u smjeni.

Na sajtu se osim upisa izvještaja nalaze razna uputstva, kako za korisnike (narezivanje DVD materijala pomoću Nera, konfiguracija elektronske pošte, skidanje materijala sa Youtuba, itd), tako i interna uputstva IT sektora (pravljenje RJ45 konektora, razne šeme internet mreže...).

Na drugom serveru je instaliran FTP (File Transfer Protocol) servis. On služi za razmjenu fajlova između udaljenih kompjutera. Omogućava nam slanje video priloga od dopisnika. Takođe, FTP koristimo i za "upload", tj. slanje određenih fajlova na druge servere.

Na serveru je podignuta i MySQL baza. Nju koristimo za potrebe službe dokumentacije. Napravljena je sql baza sa web interfejsom u kojoj kolege iz dokumentacije unose podatke tj. evidenciju o kasetama kao što su naslov materijala, ekipa koja je radila snimanje, time code, tip kasete, broj kasete itd. Sa ovom aplikacijom je prilično olakšan posao ljudima iz dokumentacije, jer oni sada na lak način mogu doći do informacija o određenoj kaseti.

Na serveru se nalazi instaliran softverski firewall. Aplikacija se zove "[Kerio Contol](#)", a pronađena je takođe usljed mog samostalnog istraživanja. On nam je jedina barijera i tačka zaštite između spoljne mreže i lokalne unutrašnje mreže unutar zgrade.

3. PREPORUKE ZA UNAPREĐENJE BEZBJEDNOSTI

Otkrivanje sigurnosih problema u sistemu je veliki izazov za administratora računarskih sistema u bilo kojoj firmi. Blagovremenim djelovanjem može znatno da se spriječi ili uspori nastanak nekog problema. Zato je jako bitno da se postojeći sistem stalno analizira, testira, kako bi se našli problemi u sistemu i kako bi se na osnovu njih moglo blagovremeno reagovati i spriječiti negativne posledice pojavljivanja nekog problema. Stalno traganje i otkrivanje sigurnosih problema služi i da bi administrator bio u toku sa novim dešavanjima i trendovima u toj sferi.

Iako ovakav način zaštite i bezbjednosti operativnih sistema ne spada u najsavremenije, samim tim ni u najefikasnije, obaveza administratora jeste da se postojeći sistem učini što stabilnijim i dugotrajnijim, uprkos gore pomenutim nedostacima.

3.1. RAČUNARI ZAPOSLENIH

Računari zaposlenih, bez obzira da li sadrže tajne podatke ili ne, zaslužuju posebnu pažnju. Najprije je potrebno definisati programe i njihove verzije koji se smiju koristiti na tim računarima. Za upotrebu programa koji se ne nalazi na spisku, zaposleni može podnijeti zahtev nadležnom IT odeljenju ili administratoru. Na spisku nikako se nesmiye

naći program, odnosno neka od njegovih verzija za koji je poznato da ima veliki broj bezbjednosnih propusta.

Prisustvo antivirusnog programa i zaštitnog zida (Firewall) je neophodno, za šta odlično rješenje predstavlja besplatni [Comodo Internet Security](#) ili komercijalni [Kaspersky Internet Security](#). Kada se radi o besplatnim antivirusnim programima poput [Avira](#), [AVG](#), [Avast](#) i [Panda](#) oni nepredstavljaju dobro rešenje zbog trenutno niskog nivoa efikasnosti. Iako se računari zaposlenih prilikom pristupa Internetu obično nalaze iza rutera, neophodno je na njima zatvoriti sve portove koji nisu neophodni, i obavezno uspjehnost ove procedure potvrditi skeniranjem odgovarajućim programom. Opcije za automatsko ažuriranje operativnog sistema trebaju da budu uključene na svim računarima korisnika. Za uobičajeni rad sa kancelarijskim alatima za obradu teksta i tabela, u poređenju sa Windows rešenjima, bezbjednija za korišćenje i znatno jednostavnija za održavanje je neka Linux distribucija i Open Office.

Kod samostalnih radnih stanica najvažnije je da obični korisnici rade sa naložima operativnog sistema ograničenih prava i mogućnosti. Kod Windows sistema, takav nalog onemogućuje upis u Program Files i Windows direktorijume što je najčešći put širenja virusa.

Programski kontrolisana upotreba USB portova treba da ima preventivni karakter i igra značajnu ulogu u sprečavanju krađe podataka od strane zaposlenih, a preporuka je da se CD/DVD i Floppy uređaji fizički ni nepostoje. Za pristup i razmjenu datoteka između zaposlenih može se koristiti NAS (Network Access storage) ili, po potrebi, mrežno dijeljenje direktorijuma sa računara korisnika.

3.2. SERVERI

Treba nabaviti nove servere sa konkretno Windows Server 2012 RS2 operativnim sistemom, instalirati domen i iskonfigurirati grupne polise, čime bi mogućnosti administracije bile mnogo veće i bile bi centralizovane. Podešavanjem grupnih polisa bi dobili mogućnost centralizovanje kontrole nad računarima

Važnost domena se ogleda u 'Active Directory', koji bi predstavljao bazu podataka svih korisnika na mreži. DC hardverski treba da bude sa sistemom redundantnog napajanja kao i sistemom diskova u raid setu. Tom računaru treba obezbijediti kvalitetan izvor napajanja, jer on zahtijeva upotrebu jakih firmiranih UPS uređaja kao i agregata. Takav računar može da služi kao DC u okviru televizije i do par godina. Na njega uvijek treba obratiti posebnu pažnju. Prostorija u kojoj bi DC bio smješten treba da ispunjava uslove koji su najoptimalniji za boravak računara na tom mjestu, a to se prije svega odnosi na temperaturu. Zbog bezbjednosti i velike funkcije koju obavlja DC treba podići jos jedan server koji će predstavljati backup DC.

Svakom korisniku prije nego što počne koristiti resurse mreže treba obezbijediti korisničko ime i lozinku. Korisničko ime i lozinka će se dobiti otvaranjem naloga u aktivnom direktorijumu. Administrator treba da otvori nalog i smjesti korisnika u određenu već definisanu grupu. U okviru

televizije za administratora je važno da zna koliko korisnika postoji na mreži i kojom vrstom posla se bave. Pored zaposlenih lica koji imaju svoje računare i koji imaju pravo pristupa mrežnim folderima, postoje korisnici koje treba tretirati kao goste. Takvim korisnicima treba obezbijediti samo pravo pristupa internetu, dok sam pristup lokalnoj mreži i resursima bi im trebalo onemogućiti. U skladu sa tim u aktivnom direktorijumu treba postojati nalog koji će na 'Desktopu' prikazivati samo ikonicu određenog 'Browsera' za pristup internetu. Takvi korisnici će biti članovi određene grupe na koja su primjenjena pravila zabrane pristupa svemu sem interneta. Aktivni direktorijum treba biti izdijeljen tako da preslikava organizacionu strukturu televizije. Kroz aktivni direktorijum na standardizovan način se pristupa svim mrežnim objektima: konfiguracije aplikacija, usluge, računari, korisnici i procesi širom cijele lokalne mreže.

Upravljanje mrežama i novim serverom bi olakšao posao 'Group Policy'. Preko njega bi išlo upravljanje korisničkim parametrima, bezbjednosnim mjerama, domenom, parametrima radne površine, itd. Kao i organizacione jedinice aktivni direktorijum po instalaciji sadrži kreirane grupe. Svaki korisnik treba biti član neke grupe, a potom i organizacione jedinice. Postojeće grupe u okviru aktivnog direktorijuma administratoru služi da korisniku dodijeli ili oduzme prava pristupa mrežnim resursima. Pomoću polisa postojeće grupe imaju unaprijed određena prava. GP treba primijeniti na svim nivoima organizacije u aktivnom direktorijumu, od domena do organizacionih jedinica. To treba uraditi sa alatkom "Group Policy Editor". Sa GPE treba stvoriti objekte koji su u aktivnom direktorijumu pridruženi organizacionim jedinicama ili koji se pozivaju na njih.

Treba aktivirati DNS servis i izvršiti određena podešavanja koja zavise od parametara postojeće lokalne mreže, da bi DNS pravilno razrešavao imena računara. DNS servis bi bio automatski uočen na mreži. Treba takodje instalirati i DHCP servis koji bi računarima dao jedinstvenu IP adresu. Uz pomoć njega sistem bi sam automatski nadgledao računare i vodio računa da IP adrese budu pravilno raspoređene.

Aktivirati File & Print Server. File server treba biti opremljen RAID 0 ili RAID 5 nizom diskova, mehanizmom za zamjenu oštećenih diskova bez prekida rada i njihovo grupisanje, kako bi svim korisnicima fajlovi bili neprekidno dostupni. Print server bi mreži obezbijedio logičke štampace i starao se o redovima za štampanje.

Serveri moraju uvijek da budu smešteni u zaključanim centrima. Pristup tim centrima mora da bude ograničen i nadgledan. Implementacija servera za sistemsko evidentiranje, takođe može biti od pomoći.

Unapređenje sistema moglo bi se učiniti i uklanjanjem starih računara kao viška opreme, jer su takvi računari skuplji za održavanje.

3.3. RAČUNARSKE MREŽE

U televiziji bi takođe trebalo nabaviti i bolju mrežnu opremu. Poseban naglasak treba dati na nabavku hardverskog firewala, po mogućnosti Cisco-vog., da bi se spriječio neautorizovani pristup sa jedne mreže na drugu.

4. ZAKLJUČAK

Ukoliko je potrebno iznijeti procjenu trenutnog stanja bezbjednosti sistema, u ovom trenutku ona bi bila između dovoljan i dobar. Otkrivanje sigurnosih problema je veliki izazov za administratora računarskih sistema u bilo kojoj firmi. Zavisno o potrebama organizacija ili korisnika moguće je uspješno iskonfigurirati sistem, kao i lokalnu mrežu. Međutim, bolje rešenje od brze reakcije i stalnog saniranja posledica je kvalitetna preventiva.

Nijedna mreža, ma koliko bila obezbijedena, nije u potpunosti sigurna, ali što je više truda potrebno da uložiti potencijalni napadač da bi ugrozio mrežu, to je sigurnost veća. Isto se odnosi i na sigurnost operativnih sistema.

LITERATURA

- [1] http://www.vtsnis.edu.rs/Predmeti/administriranje_racunarskih_mreza/Admnistriranje_mreza_nova_predavanja_5.pdf, feb 2014
- [2] <https://www.kerio.com/control/download>, feb 2014
- [3] <http://zivot.krstarica.com/l/tehnika/rad-na-kompjuteru/sta-je-ftp/>, feb 2014
- [4] http://en.wikipedia.org/wiki/Deep_Freeze_%28software%29, feb 2014

ADAPTACIJA IZGLEDA WEB STRANICE USLOVLJENA KLIJENSKIM SPECIFIČNOSTIMA U ASP .NET MVC 4 OKRUŽENJU
THE WEB SITE ADAPTATION BASED ON CLIENT'S SPECIFICITY IN ASP .NET MVC 4 FRAMEWORK

Zoran Veličković, *Visoka tehnička škola strukovnih studija Niš*
Milojko Jevtović, *Inženjerska akademija Srbije*

Sadržaj: *Problem adaptacije Web sadržaja je postao urgentan sa porastom broja mobilnih korisnika. Razlog ovome leži u činjenici da mobilni uređaji poseduju značajne specifičnosti u odnosu na desktop računare. U ovom radu je razmatran problem adaptacije Web stranice u funkciji klijentske terminalne opreme primenom Microsoft-ove verzije MVC koncepta realizovanog na ASP .NET platformi. Na primeru šablona Internet aplikacije koja je adaptirana za desktop i mobilne platforme je pokazana primenljivost ove tehnologije. Raspoložive tehnike u ASP .NET MVC radnom okruženju u velikoj meri zadovoljavaju zahteve za adaptacijom Web stranice u funkciji klijentskih specifičnosti.*

Abstract: *The problem of adapting Web content has become urgent with the increase the number of mobile users because mobile devices have significant specificity compared to desktop computer. The adaptation of Web pages in the function of customer terminal equipment by using Microsoft's version of MVC concepts implemented in ASP. NET platform are discussed. The applicability of this technology is demonstrated on the example of pattern Internet application that has been adapted for desktop and mobile platforms. Available techniques in the ASP. NET MVC work environment largely meet the requirements for adapting Web pages to function client specifics.*

1. UVOD

Razvoj savremenih digitalnih telekomunikacija, a posebno mobilnih telekomunikacionih sistema treće i četvrte generacije (3G i 4G) doprineo je uvođenju novih bežičnih multimedijalnih usluga [1]. U primeni su multimedijalne usluge kao što su VoIP (engl. *Voice over Internet Protocol*), VoD (engl. *Video on Demand*), IPTV (engl. *Internet Protocol TeleVision*) i videokonferencing. Uvođenje potpuno novih mobilnih multimedijalnih usluga dovelo je do povećanja broja korisnika smart telefona i tablet uređaja koji poseduju izuzetnu mrežnu podršku. Direktna posledica povećanja broja mobilnih multimedijalnih korisnika je drastično povećanje obima multimedijalnog saobraćaja. Istovremeno, sa razvojem mobilnih terminalnih uređaja, razvijaju se i savremene mobilne Web aplikacije [2]. Tako, RIA (engl. *Rich Internet Application*) Web aplikacije sve češće podrazumevaju korišćenje videa i VoIP-a za prezentaciju svojih sadržaja. Zbog toga je podrška Web baziranih aplikacija postala standard savremenih digitalnih protokola i mreža [3], [4].

Najčešći pristup Web aplikacijama se ostvaruje univerzalnom klijentskom aplikacijom - Web čitačem (engl. *Browsers*). Iako Web čitači spadaju u najkorišćenije klijentske aplikacije, njih je uvek pratio problem nekompatibilnosti prikazivanja Web sadržaja. Ovaj problem potiče od samih proizvođača Web čitača koji su se utrkivali dodavanjem novih oznaka u HTML (engl. *Hyper Text Markup Language*) jezik koji je osnova za opisivanje Web stranica. Pokušaj standardizacije ovog jezika u verziji 5 je sada na delu. Česte izmene verzija Web čitača su dizajnerima Web aplikacija zadavale mnogo problema oko toga kako svim klijentima (koji poseduju heterogenu terminalnu

opremu sa različitim verzijama Web čitača) obezbediti adekvatan prikaz sadržaja [5]. Dodatni problem koji je generisan sa pojavom savremenih mobilnih uređaja je kako Web sadržaje adekvatno prikazati na mobilnim platformama [6], [7]. Dizajn Web sadržaja koji se mogu istovremeno efikasno pregledavati i na mobilnim i desktop verzijama Web čitača stvorio je novi izazov programerima. Obzirom na ograničene resurse mobilnih platformi i nešto drugačiju upotrebu samih uređaja, mobilni Web čitači imaju redukovani skup funkcija i nešto skromnije računarske mogućnosti.

U ovom radu je prikazano rešenje ovog problema bazirano na Microsoft-ovoj tehnologiji ASP .NET koje nosi prefiks MVC (engl. *Model-View-Controller*) [8]. Na primeru jednostavne Web aplikacije zasnovane na MVC 4 šablonu su pokazane mogućnosti koje pruža ovo razvojno okruženje. Struktura rada je sledeća. U drugom poglavlju je razmatrana arhitektura MVC aplikacija i dat je funkcionalni opis gradivnih blokova. Problemi prisutni u realizaciji Web sadržaja na različitim platformama su predstavljeni u trećem poglavlju, dok je u četvrtom poglavlju na primeru šablona Internet aplikacije prikazano rešenje koje pruža ASP .NET MVC 4. Pravci i perspektive daljeg razvoja iz ove oblasti su dati u završnom, petom, poglavlju.

2. ARHITEKTURA ASP .NET MVC APLIKACIJE

ASP.NET MVC je besplatno Microsoft-ovo rešenje za Web aplikacije koje koriste Model-View-Controller šablon (engl. *pattern*) [9]. Glavni cilj dizajna sa šablonima je da se izoluje poslovna logika od korisničkog interfejsa kako bi se glavna pažnja posvetila strukturi Web aplikacije. Dobro definisana struktura Web aplikacije obezbeđuje njeno lakše

održavanje i testiranje. Svaka ASP.NET MVC aplikacija ima tri osnovna dela: model (engl. *Models*), pogled - izgled (engl. *Views*) i kontroler (engl. *Controller*). Na slici 1 je prikazana komunikacija između osnovnih delova MVC aplikacije. Strelicom je predstavljen klijentski zahtev koji se ispostavlja kontroleru MVC aplikacije putem HTTP zahteva. Punim linijama su predstavljene direktne komunikacije, dok su isprekidanim linijama predstavljene indirektno komunikacije između delova ASP .NET MVC aplikacije. Značenje pojedinih oznaka su sledeće: 1. Procesiranje zahteva – direktna komunikacija; 2. Vraćanje odgovora klijentu - direktna komunikacija; 3. Slanje kao dela zahteva - direktna komunikacija; 4. Slanje za prezentaciju klijentu – indirektna komunikacija; 5. Slanje klijentskog zahteva – indirektna komunikacija.

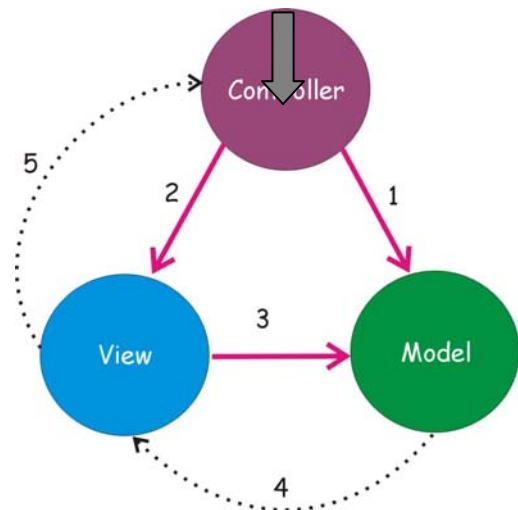
Model ASP .NET MVC aplikacije se realizuje od klasa koje obrađuju podatke i na taj način formiraju poslovnu logiku. Obradu podataka pokreću klase iz kontrolera koje su zadužene za obradu klijentskih zahteva. Po završenoj obradi podataka, pozivaju se klase koje oblikuju odgovor na klijentski zahtev u vidu HTML koda koji se prikazuje na klijentskom računaru. ASP.NET MVC 4/5 formira HTL kod koji je kompatibilan sa HTML5 standardom uz obilato korišćenje CSS3 specifikacije. Obzirom na to da HTML5 koristi semantičko označavanje, podržane su nove oznake kao što su <header>, <footer>, <nav>, <section> i pojednostavljena <!DOCTYPE html> direktiva.

MVC Kontroleri

MVC kontroleri obrađuje klijentske zahteve za Web stranicama koji su pristigli putem HTTP protokola. Koncept MVC-a podrazumeva da se svaki klijentski zahtev obrađuje zasebnim kontrolerom koji zapravo predstavlja jednu .NET klasu. Metode članice ovih klasa za obradu klijentskih zahteva se nazivaju akcionim metodama i obično vraćaju objekte tipa *ActionResult*. Dakle, za implementaciju rezultata rada akcionih klasa koristi se *ActionResult* klasa. Za specifične primene izveden je čitav niz klasa koje je nasleđuju.

MVC Rutiranje

Proces kojim se započinje obrada klijentskog zahteva se u MVC kontekstu naziva „*routing engine*“. Zapravo, on je odgovoran da se klijentski zahtev preslika u poziv na izvršenje odgovarajućoj metodi jednog od MVC kontrolera. Rezultat rada akcione metode je objekt koji treba vratiti korisniku na njemu pogodan način. Adaptacija rezultata akcionog metoda se obavlja u odeljku *Views* gde se mogu uvažiti specifičnosti klijentskog Web čitača i hardverske platforme. ASP .NET rutiranje omogućava kreiranje logičkog skupa URL-a (engl. *Uniform Resource Locator*) koji su nezavisni od strukture datoteka Web aplikacije na serveru. URL-ovi dobijeni na ovaj način su deskriptivniji i korisnički razumljiviji. Sa druge strane, ovi URL-ovi su prepoznatljiviji pretraživačima čime se ostvaruje veća vidljivost Web aplikacije. MVC definiše URL šablon (engl. *pattern*) kojim se uslovljava sadržina segmenata koji sačinjavaju URL zahtev.



Slika 1. Implementacija MVC koncepta u .NET-u, direktna (puna linija) komunikacija i indirektna (isprekidana linija) komunikacija između delova MVC aplikacije.

MVC Views

U MVC kontekstu kôd kojim se opisuje sadržaj dinamičkih Web stranica se nalazi na serverskoj strani aplikacije u vidu skriptova. Izvršavanje serverskih skriptova će proizvesti konačan HTML kod Web stranice i time odrediti konačan izgled Web stranice. Prilikom kreiranja izgleda Web stranice, treba specificirati „*view engine*“ koja će se koristiti za procesiranje serverskog koda. U ASP.NET-u stoje na raspolaganju dve tehnike (engine-a):

- *ASPX*: Originalna tehnika razvijena još u verziji ASP.NET MVC 1 koja je slična Web formama.
- *RAZOR*: Nova tehnika koja koristi novu jednostavnu fluidnu sintaksu.

Važan dodatak u ASP.NET MVC 4 su „*Display Modes*“ koji omogućavaju aplikacijama da izaberu prikaz koji je pogodniji u klijentskom pretraživaču. Na ovaj način se može uvažiti činjenica da klijenti sve češće pristupaju Web stranicama putem mobilnih uređaja. Za održavanje adekvatnog izgleda Web stranica kroz čitavu Web aplikaciju je značajna prednost koju nudi ASP.NET MVC 4 [11]. Uobičajeno je kreiranje *view* klasa za svaki akcioni metod sa istim imenom. Postoje nekoliko baznih karakteristika koje karakterišu različite *View*-e: *Strongly-typed view*, *Partial view*, i *Layout* ili *master page* (ovaj *view* je korišćen kasnije u realizovanom primeru).

MVC Modeli

Modeli u ASP. NET MVC predstavljaju skup objekata kojima se implementira funkcionalnost aplikacije. U ASP .NET MVC konceptu postoji nekoliko tipova modela: model podataka (engl. *data model*), poslovni model (engl. *business model*) i *view model*. Zajedničkim imenom skup ovih modela se naziva „*domain model*“. Za adaptaciju prikaza istog Web sadržaja na različite korisničke mogućnosti, u ovom radu nisu korišćeni Modeli, tako da o njima neće biti dalje reči.

http request

3. KLIJENTSKE SPECIFIČNOSTI I PROBLEMI U PRIKAZU WEB SADRŽAJA

U ovoj sekciji su prikazani izvori problema u prikazu istovetnog Web sadržaja na različitim platformama. Već je pomenuto da je razlika u hardveru između mobilnih i desktop računara značajna. Druge, svima vidljive, razlike se odnose na dimenzije displeja mobilnih uređaja, način njihovog napajanja, redukovani skup tastera i raspoloživost smeštajnih kapaciteta. Pored razlike u dimenzijama displeja, obzirom na mesto upotrebe, razumljiv je zahtev za povećanim kontrastom i osvetljajem displeja. Obzirom da se mobilni uređaji napajaju iz baterije, redukcija potrošnje je jedan od najznačajnijih zahteva koji je vrlo teško zadovoljiti. Mada se kapacitet i način punjenja baterija stalno unapređuje, još uvek je značajna potrošnja *Wi-Fi*, *GPS* i *Bluetooth* modula kojim je ograničeno uoprebno vreme bez punjenja baterije.

U ovom radu se razmatra način na koj bi trebalo prikazati Web sadržaj na mobilnim platformama, a da to omogući komotan rad Web aplikaciji. Jasno je da ako bi se svim uređajima u mreži slao istovetan HTML kod kojim je predstavljen Web sadržaj, to ne bi zadovoljilo sve korisnike. Veličina fonta, boja pozadine i raspored objekata na Web stranici mora se prilagoditi svakoj klasi korisnika. Tako, veličina fonta na mobilnim platformama, obzirom na veličinu displeja, mora biti veća, a da bi se povećao kontrast zbog okruženja u kome se čita displej boja pozadine treba da bude drugačija. Jedno od rešenja koje se može naći u praksi je da se razvije potpuno nova mobilna aplikacija sa zasebnim URL-om (ili najčešće na poddomenu desktop verzije). Iako su ova rešenja još uvek prisutna u praksi, ne preporučuje se kao savremeno rešenje. Ideja ovog rada je da se na jednom URL-u nalaze sve verzije aplikacije bez obzira na tip pristupnog uređaja. Nove Web i softverske tehnologije su omogućile realizaciju ovog koncepta. Osnovne tehnologije koje podržavaju ovaj koncept su CSS 3 i HTML 5 koje omogućavaju razdvajanje Web sadržaja od izgleda, što daje potencijal za različiti prikaz istih sadržaja u funkciji pristupnih uređaja. Za ove potrebe je posebno značajna CSS oznaka *media* koja se odnosi na tip pristupnog uređaja za koji se mogu definisati osnovne karakteristike. Zapravo, na osnovu ove oznake identifikuje se tip čitača kojim se pristupa Web sadržaju. Web čitačima će biti poslat HTML kod koji odgovara samo njima. Da bi se potpuno definisao izgled Web sadržaja na pojedinim platformama, u MVC-u je realizovan koncept *view override*.

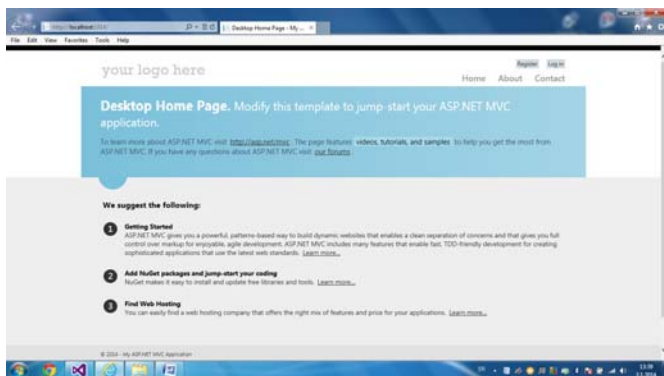
4. ADAPTACIJA MVC 4 ŠABLONA INTERNET APLIKACIJE NA KLIJENTSKE SPECIFIČNOSTI

Za prikaz mogućnosti adaptacije Web aplikacije na korisničke specifičnosti iskorišćen je šablon Internet aplikacije koji dolazi sa Visual Studiom 2012/13. Adaptacija prikaza Web stranice je zasnovana na prepoznavanju tipa pristupnog uređaja i adaptaciji Web sadržaja na veličinu displeja, podržani skup tastera i primenjeni Web čitač. Ako je cilj adaptacije Web sadržaja formiranje generičkog rešenja, onda ga treba zasnovati na Web tehnologijama koje obuhvataju standardi vezani za HTML 5, CSS 3 i JavaScript. Upravo je ovaj koncept realizovan u ASP .NET MVC 4 za razvoj savremenih Web aplikacija. U kontekstu ASP .NET

MVC 4 tehnologije treba formirati više akcionih metoda sa pripadajućim *view*-om koje treba prilagoditi različitim klijentskim specifičnostima [8]. Imena *view* stranica su jednaka imenima akcionih metoda sa ekstenzijom *.cshtml*. Na slici 2a je prikazan deo CSS koda iz fajla *Site.css* kojim se definiše opšti izgled stranice sa nepridruženim pravilima za imenovane sekcije. Na slici 2b je prikazan deo CSS koda koji se odnosi na mobilnu Web stranicu sa definisanom maksimalnom širinom ekrana mobilnog uređaja i nepridruženim pravilima. Izgled Web stranice na punoj širini prozora desktop Web čitača je prikazan na slici 3, dok je na slici 4 je prikazan izgled iste Web stranice na mobilnoj platformi u *Opera mini* Web čitaču.

```
.featured {
    background-color: #fff;
}
.featured .content-wrapper {
    background-color: #7ac0da;
    background-image: -ms-linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
    background-image: -o-linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
    background-image: -webkit-gradient(linear, left top, right top, color-stop(0, #7ac0da), color-stop(1, #a4d4e6));
    background-image: -webkit-linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
    background-image: linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
    color: #3e5667;
    padding: 20px 40px 30px 40px;
}
...
a)
/***** Mobile Styles *****/
@media only screen and (max-width: 850px) {
    .main-content,
    .featured + .main-content {
        background-position: 10px 0;
        background-image: none;
    }
    .content-wrapper {
        padding-right: 10px;
        padding-left: 10px;
    }
    .featured .content-wrapper { padding: 10px;
        background-color: #e80c4d;
        background-image: -ms-linear-gradient(left, #e80c4d 0%, #e6507d 100%);
        background-image: -o-linear-gradient(left, #e80c4d 0%, #e6507d 100%);
        background-image: -webkit-gradient(linear, left top, right top, color-stop(0, #e80c4d), color-stop(1, #e6507d));
        background-image: -webkit-linear-gradient(left, #e80c4d 0%, #e6507d 100%);
        background-image: -linear-gradient(left, #e80c4d 0%, #e6507d 100%);
        color: #fff;
    }
    ...
    b)
```

Slika 2. Fragment CSS koda iz fajla *Site.css* kojim se definiše a) opšti izgled desktop stranice b) izgled mobilne stranice za maksimalnu širinu ekrana mobilnog uređaja od 850 piksela.



Slika 3. Izgled šablona MVC Internet Web stranice u Internet exploreru.



Slika 4. Izgled šablona MVC Internet Web stranice u Opera mobilnom čitaču implemetiranom na mobilnom telefonu Samsung Galaxy II.

Različiti prikaz Web stranice je realizovan primenom CSS specifikacije za rad sa različitim tipovima medija. Za ove potrebe koristi se pravilo @media koje se odnosi samo na uređaje sa ekranom. Media queries identifikuju tip pristupnog medija i samim time skup njegovih karakteristika. Karakteristike koje značajno utiču na dizajn Web stranice se odnose na dimenzije ekrana, broj podržanih boja i podrška za tastaturu. Da bi se izgled Web stranice prilagodio širini ekrana Web čitača upotrebljeno je CSS pravilo za relativno pozicioniranje elemenata Web stranice. Veličina fonta za mobilnu verziju je znatno veća od one koja je primenjena za desktop verziju. Takođe, boja pozadine (background-color) mobilne verzije je crvene boje što jasno povećava kontrast (a time i čitljivost) u odnosu na desktop verziju.

Prikazani primer je zasnovan na MVC 4 šablonu Internet aplikacije koja se nalazi u VS 2012/13. Uz šablon MVC 4 Internet aplikacije podrazumevano se pridružuje fajl sa opisom stilova *Site.css* koji sadrži @media pravila. Prekapanjem izgleda Web stranica za desktop, odnosno mobilnu verziju, adaptiran je raspored elemenata, veličina fonta, kao i boja pozadine.

5. ZAKLJUČAK

Dijametralno različite karakteristike terminalne opreme koju klijenti koriste za pristup Web sadržajima su nametnule potrebu za adaptacijom Web sadržaja. Jednostavno je ustanoviti da je veoma mali broj (domaćih) Web stranica dizajniran tako da im se može pristupiti heterogenom terminalnom opremom. Iako su Web tehnologije koje omogućavaju adaptaciju Web sadržaja odavno u primeni, one se nisu značajno koristile u ove svrhe. Savremeno razvojno okruženje ASP .NET MVC 4 ima ugrađenu podršku za „responsive“ Web dizajn što može imati uticaja na standardizovanja ovog koncepta. Cilj ovog rada je bio da se testiraju mogućnosti adaptacije Web sadržaja koje nudi ovo razvojno okruženje.

Na osnovu dobijenih rezultata može se zaključiti sledeće: 1) ASP .NET MVC 4 u potpunosti podržava adaptaciju izgleda Web stranice korišćenjem HTML 5, CSS 3 i JQuery specifikacije; 2) Uključena je podrška za mobilne uređaje i mobilne Web čitače; 3) Podrška za identifikaciju tipa pristupnog medija omogućava implementaciju klijent-specifičnog HTML-a; 4) ASP .NET MVC 4 pojednostavljuje dizajn Web stranice za mobilne uređaje preklapanjem metoda za izgled Web stranice; 5) Razrađena je efikasna Razor tehnika za kodiranje izgleda Web stranice na strani Web servera.

ASP .NET MVC 4 je tehnologija koje pruža izuzetne mogućnosti pri adaptaciji izgleda Web stranice u funkciji tipa klijentskog čitača. Raspoložive tehnike u ASP .NET MVC 4 radnom okruženju u velikoj meri zadovoljavaju zahteve za adaptacijom Web stranice u funkciji klijentskih specifičnosti.

LITERATURA

- [1] L. Cai, X. Shen, J. W. Mark, Multimedia Services in Wireless Internet: Modeling and Analysis, Wiley, 2009.
- [2] P. Deitel, H. Deitel, AJAX, Rich Internet Applications and Web Development for Programmers, Deitel, 2008.
- [3] M. Jevtović, Z. Veličković, Komunikacioni protokoli prepletenih slojeva, Akademski misao, 2013.
- [4] Z. Veličković, M. Jevtović, V. Pavlović, „Quality of services in IP/MPLS networks“, UNITECH 2013, pp. II-107-112, Gabrovo 2013.
- [5] M. Stojković, S. Stošović, Z. Veličković, „Smart TV aplikacija zasnovana na web 2.0 tehnologiji“, YUINFO 2012, ISBN: 978-86-85525-09-4, pp. 641-646, Kopaonik 2012
- [6] Z. Veličković, A. Ilić, „Primena XML tehnologija na mobilnim platformama zasnovanim na BADI“, YUINFO 2012, ISBN: 978-86-85525-09-4, pp. 624-628, Kopaonik 2012.
- [7] B. Krstić, Z. Veličković, „Straničenje Web stranica zasnovano na CSS 3 specifikaciji“, Informacione tehnologije, pp. 52-55, Žabljak 2012.
- [8] J. Rolando, G. Paz, Beginning ASP.NET MVC 4, Apress Media, 2013.
- [9] Microsoft Corporation, Developing Modern Mobile Web Apps, patterns & practices, 2012.
- [10] <http://tfs.visualstudio.com>
- [11] <http://www.asp.net/mvc>

ALATI ZA PODRŠKU MJERENJU SINHFROFAZORA TOOLS FOR SUPPORTING THE MEASUREMENT OF SYNCHROPHASORS

Stevan Šandi, *Information Technology Systems, Podgorica*
Tomo Popović, *Texas A&M University, College Station, Texas*
Božo Krstajić, *Elektrotehnički fakultet, Podgorica*

Sadržaj: *Sinhronizovano mjerenje fazora struje i napona dobija sve veći značaj u nadzoru elektroenergetskih sistema. PMU uređaji za mjerenje fazora i PDC koncentratori izmjerenih fazora čine osnovne komponente sistema za precizno mjerenje sinhrofazora. U radu je data definicija sinhrofazora i njihove upotrebe u elektroenergetskim sistemima, kao i opis mreže za prenos i prikupljanje podataka. Konačno, rad predstavlja uvod u alate za podršku implementacije sa fokusom na PMU simulatore koji se mogu koristiti za testiranje mreže, PDC instalacije, laboratorijska istraživanja i razvoj aplikacija.*

Abstract: *There is a growing interest in the use of synchronized phasor measurements of voltage and current quantities in electric power systems. The main building blocks for the phasor measurement systems are PMU phasor measurement devices and PDC phasor data concentrators. This paper provides the definition of synchrophasors and their use in power systems, as well as a description of communication network for data collection. Additionally, this paper gives an introduction to the support tools needed for the implementation of synchrophasor systems. The focus is on PMU simulators that are used for testing PDC setup, network, research, and application development.*

1. UVOD

Uobičajene metode za procjenu stanja elektroenergetskog sistema kao što je mjerenje signala napona i struje u trafostanicama, svakih nekoliko sekundi, ne mogu prikazati dinamiku sistema. Nakon raspada elektroenergetskog sistema 1965. godine na granici između Kanade i Sjedinjenih Američkih Država uočena je potreba za preciznijim praćenjem sistema [1]. A nakon sličnih događaja 2003. godine počinje intenzivan razvoj i implementiranje sistema za nadzor, upravljanje i zaštitu tzv. WAMPAC (*Wide Area Monitoring, Protection And Control*) sistemi [2]. WAMPAC sistemi se zasnivaju na sinhronizovanom mjerenju fazora napona i struje čime se dobija precizna, dinamička slika elektroenergetskog sistema u realnom vremenu [3]. Uređaji koji vrše mjerenje nazivaju se mjerne jedinice za mjerenje fazora – PMU (*Phasor Measurement Unit*) [4]. PMU uređaji koriste vremenske reference kao što je GPS (*Global Positioning System*) za precizno označavanje trenutka mjerenja koje se ponavlja i do 120 puta u sekundi. Tako izmjereni i vremenski referencirani fazori se šalju koncentratoru fazora – PDC (*Phasor Data Concentrator*) koji prikuplja fazore iz više PMU uređaja a zatim ih sortira po vremenu i prosljeđuje narednom PDC uređaju ili ih šalje na dalju obradu i analizu. Iako se ovi uređaji danas koriste isključivo za nadzor sistema, u skorijoj budućnosti imaće značajnu ulogu u kontroli i zaštiti elektroenergetskih sistema. Iz tog razloga pouzdanost i preciznost ovih uređaja kao i mrežne infrastrukture koja ih povezuje je veoma bitna.

U ovom radu biće obrađena postojeća softverska rješenja koja se koriste za testiranje gore navedenih komponenti kao i komunikacionih kanala koji ih povezuju. U drugom poglavlju biće objašnjen pojam sinhronizovanog fazora tj.

sinhrofazora kao i IEEE standard za prenos sinhrofazora kroz komunikacionu mrežu. Treće poglavlje opisuje mrežu za prenos sinhrofazora i format IEEE C37.118.2 standarda. Četvrto poglavlje se bavi trenutno dostupnim alatima za podršku implementacije, na prvom mjestu simulacionim softverom, njihovim karakteristikama, prednostima i nedostacima. U okviru zaključka su navedene poželjne karakteristike i funkcionalnosti PMU simulatora čijom realizacijom se treba baviti u daljem radu.

2. SINHFROFAZORI

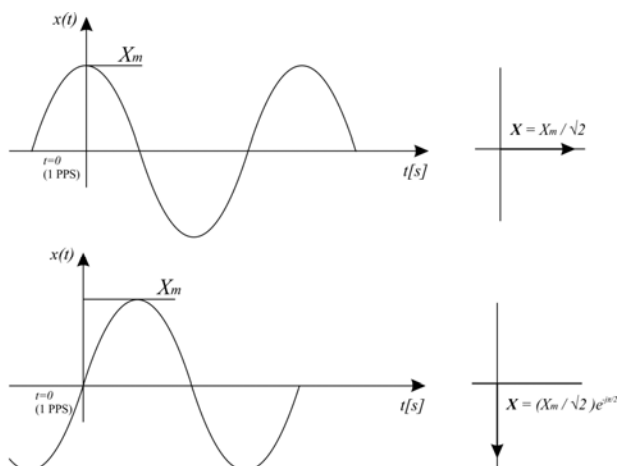
Talasni oblici napona i struja najčešće se predstavljaju pomoću fazora kako je to prikazano izrazima (1) i (2):

$$x(t) = X_m \cos(\omega t + \varphi) \quad (1)$$

$$X = (X_m / \sqrt{2})(\cos\varphi + j\sin\varphi) = X_r + jX_i \quad (2)$$

gdje je sa r i i respektivno označena realna i imaginarna komponenta fazora. Pored amplitude X_m fazor definišu kružna učestanost ω i fazni pomjeraj φ .

Sinhrofazor je reprezentacija signala $x(t)$ iz izraza (1) čija je vrijednost izražena izrazom (2) gdje je φ trenutni fazni pomjeraj u odnosu na kosinusnu funkciju nominalne frekvencije sistema sinhronizovane sa UTC-om (*Coordinated Universal Time*). Pod pojmom sinhronizovane kosinusne funkcije sa UTC-om podrazumijeva se da ona ima maksimalnu vrijednost za $t=0$ i fazni pomjeraj $\varphi=0^\circ$ u trenutku kada vremenski marker PPS (*Pulse Per Second*) označi prelazak na narednu sekundu po UTC-u ili pozitivan prolazak kroz nulu i ugao $\varphi=-90^\circ$ za sinusni oblik. Na slici 1. je ilustrovana konvencija za reprezentaciju odnosa između faznog pomjeraja sinhrofazora i UTC-a.



Slika 1. Konvencija za predstavljanje sinhrofazora

Izmjereni sinhrofazor mora imati preciznu UTC vremensku oznaku dobijenu u trenutku mjerenja. Vremenska oznaka se sastoji od tri broja i to: SOC (*Second Of Century*) broja, FRASEC (*Fraction of Second*) broja i oznaku za preciznost vremena. SOC predstavlja broj sekundi koje su protekle od ponoći (00:00:00) 1. januara 1970. godine. Ovaj broj je zapisan kao 32-bitna cjelobrona vrijednost bez predznaka. Prestupne sekunde se dodaju ili oduzimaju od ove vrijednosti da bi održala sinhronizovanost sa UTC-om. FRASEC predstavlja 24-bitni zapis djelića sekunde. Pomoću ovog broja moguće je postići maksimalnu vremensku rezoluciju od 59.6 ns. Oznaka za preciznost vremena pokazuje tačnost vremenske reference kao i eventualno uračunavanje prestupne sekunde [5]. Mjerenja sinhrofazora moraju biti usklađena sa UTC-om sa dovoljnom preciznošću tako da ne prekorače maksimalnu dozvoljenu grešku od 0.57° (0.01 radijan) prilikom mjerenja faznog pomjeraja. Greška od 0.01 radijan kod faznog pomjeraja odgovara vremenskoj grešci $\pm 26 \mu s$ za nominalnu frekvenciju sistema od 60 Hz, i $\pm 31 \mu s$ za nominalnu frekvenciju od 50 Hz. Ovakva greška faznog pomjeraja prouzrokuje 1% TVE (*Total Vector Error*) što je maksimalna dozvoljena vrijednost greške za stacionarno stanje definisana u [6]. Iz tog razloga mjerni uređaji moraju biti sinhronizovani pomoću izuzetno preciznih vremenskih referenci kao što je GPS koji je u mogućnosti da drži TVE unutar dozvoljenih granica.

Informacije o izmjenjenim sinhrofazorima moraju biti obrađene tako da mogu biti poslate, kao paketi podataka, sa učestanošću F_s . F_s je cijeli broj koji pokazuje koliko se paketa šalje u sekundi odnosno nakon koliko se sekundni šalje paket. Paketi podataka mogu sadržati informacije o više izmjenjenih sinhrofazora (razdvojenih po kanalima), analogne i digitalne riječi kao i statusne riječi definisane u [5]. Obično je učestanost slanja paketa F_s cjelobrojni djeljilac nominalne frekvencije sistema. Primjeri učestanosti slanja podataka su prikazane u tabeli 1. F je nominalna frekvencija sistema, a F_s broj paketa koji se šalje u sekundi.

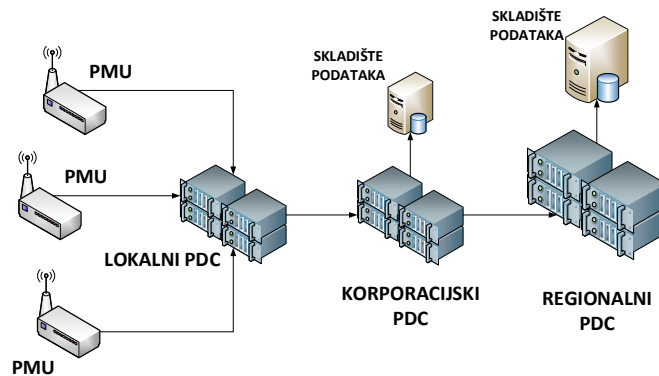
F	50 Hz				60 Hz			
F_s	10	25	50	100	15	30	60	120

Tabela 1. Zahtijevane učestanosti slanja podataka

Izbor učestanosti slanja paketa se prepušta korisniku a nerijetko su omogućene učestanosti i od 100 ili 120 paketa po sekundi ili učestanosti niže od 10 paketa u sekundi npr. 1 paket u sekundi. U opštem slučaju, kada šaljemo N paketa po sekundi to znači da se vremenski marker prvog paketa poklapa sa PPS signalom a svaki naredni je ravnomjerno raspoređen na rastojanju od $1/N$ sekundi. Tada će prvi paket imati FRASEC broj jednak 0 a posljednji $(N-1)/N$.

3. MREŽA ZA PRENOS SINHFROFAZORA

Mreža za mjerenje i prenos sinhrofazora prikazana je na slici 2. Generalno posmatrano, mrežu sačinjavaju PMU i PDC uređaji postavljeni na ključnim mjestima u elektroenergetskom sistemu. PMU uređaji mogu biti realizovani kao zasebni fizički uređaji ili kao dio nekog multifunkcijskog uređaja (najčešće zaštitnog releja). Informacije o sinhrofazorima se mogu slati kroz mrežu ili čuvati lokalno. PDC se više posmatra kao funkcionalnost nego kao zaseban uređaj jer je uvijek integrisan u druge uređaje. Posao PDC je da sortira primljene sinhrofazore po vremenu a zatim ih takve prosljedi na dalju analizu. Iz tog razloga PDC najčešće predstavlja hardversko-softverski paket. PDC se obično postavlja na centralnoj lokaciji gdje skuplja podatke od više PMU uređaja a zatim ih šalje korporativnom ili regionalnom PDC-u. Na ovaj način dobijaju se lokalizovane informacije o naponu, struji, aktivnoj ili reaktivnoj snazi koje se kasnije koriste za vizualizaciju, automatsku kontrolu, zaštitu itd.

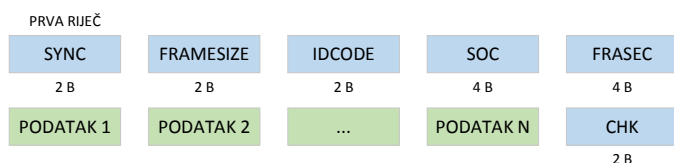


Slika 2. Mreža za mjerenje i prenos sinhrofazora

Da bi komunikacija između PMU i PDC uređaja različitih proizvođača bila moguća definisan je standard (IEEE Std C37.118) za format poruka koje se razmjenjuju. Ukoliko PMU ne šalje izmjerene podatke već ih snima lokalno onda nije potrebno implementirati standard, u suprotnom propisani zahtjevi moraju se ispoštovati. Bilo koji komunikacioni sistem ili medijum može biti korišćen za razmjenu poruka. Standard definiše 4 tipa poruka i to poruke koje sadrže: podatke, konfiguraciju, zaglavlje ili komandu. Prva tri tipa poruka prenose se od PMU/PDC uređaja prema odredištu a komande se šalju izvornom PMU/PDC uređaju. Poruke koje nose podatke sadrže vrijednosti izmjerene od strane PMU uređaja. Poruke koje nose konfiguraciju su mašinski čitljive poruke koje opisuju tipove podataka, kalibracione faktore i slične meta-podatke. Poruke zaglavlja sadrže čovjeku razumljiv opis PMU/PDC uređaja koje definiše korisnik.

Komandne poruke su mašinski čitljive i služe za konfigurisanje ili upravljanje PMU/PDC uređajima.

Sve poruke počinju dvobajtnom sinhronizacionom SYNC riječju. SYNC riječ omogućava sinhronizaciju i identifikaciju tipa poruke. Nakon SYNC riječi slijedi FRAMESIZE riječ koja definiše ukupnu dužinu poruke uključujući i CRC kontrolnu riječ. Zatim slijedi IDCODE riječ koja jednoznačno identifikuje uređaj i neophodna je za pravilno utvrđivanje toka podataka. Nakon IDCODE riječi slijedi vremenski marker definisan pomoću SOC i FRASEC riječi. Zatim slijede podaci koji su specifični za svaki tip poruke i na kraju se nalazi dvobajtna CHK riječ koja je zapravo CRC kontrolna riječ koja služi za provjeru ispravnosti poruke. Na slici 3. je prikazan uopšteni izgled poruke za razmjenu sinhrofazora.



Slika 3. Opšti format (IEEE C37.188) poruke

U standardu je predložen način prenosa sinhrofazora pomoću *Internet protokol*-a (IP). Na raspolaganju su TCP i UDP protokoli transportnog nivoa. IP paket se najčešće prenosi *Ethernet*-om. Ovako je omogućeno više vrsta prenosa, TCP nam omogućava pouzdan prenos poruka na račun povećanja neophodnog propusnog opsega linka dok UDP nudi optimalno iskorišćenje propusnog opsega bez garancije dostave poruka. Mane TCP-a dolaze do izražaja u slučaju neuspješne dostave paketa koja zahtijeva ponovno slanje istog paketa što dovodi do kašnjenja. Izgubljeni paket koji nosi poruku sa sinhrofazorima i nije toliko značajan problem jer neće mnogo uticati na analizu dok izgubljena komandna ili konfiguraciona poruka mogu prouzrokovati neispravan rad cjelokupnog sistema. Iz tog razloga pribjegava se kompromisnom rješenju a to je da se poruke koje nose konfiguraciju, zaglavlje ili komandu, zbog svoje važnosti, šalju pomoću TCP-a a poruke koje nose sinhrofazore šalju UDP-om. Na taj način kašnjenje usled retransmisije je minimizovano, bolje iskorišćen link zbog izbjegavanja slanja poruka o uspješnosti dostave a zadržana pouzdanost slanja poruka podešavanja i kontrole koje se ne šalju tako često. Posljednja revizija IEEE C37.118 standarda je napravljena 2011. godine a ostavljeno je prostora i za dalja unapređenja.

4. ALATI ZA PODRŠKU MJERENJU SINHROFAZORA

Iako su se prvi komercijalni PMU uređaji pojavili početkom devedesetih godina prošlog vijeka, još uvijek nisu u širokoj upotrebi a njihova trenutna uloga se svodi gotovo isključivo na monitoring sistema. Najzanimljivije mogućnosti koje nudi sinhronizovano mjerenje fazora su automatska kontrola i upravljanje. Današnja istraživanja su uglavnom usmjerena ka razvoju takvih sistema gdje mjerenje i obrada sinhrofazora predstavljaju veoma važnu operaciju. Kako je prosječna cijena PMU uređaja između deset hiljada i četrdeset hiljada eura testiranje i istraživanje sa stvarnim PMU uređajima je neisplativo. U tom slučaju, pribjegava se

racionalnijem rješenju kao što su softverski PMU simulatori. PMU simulatori imaju višenamjensku ulogu i često se koriste u sljedeće svrhe:

- testiranje fizičkih PDC uređaja,
- testiranje mreže i mrežnih resursa,
- testiranje sistema dok nije pod naponom u tzv. *offline* stanju,
- razna laboratorijska istraživanja i
- razvoj i testiranje aplikacija.

Testiranje PDC uređaja je moguće jedino ako su unaprijed poznati podaci koji se šalju od strane PMU uređaja. Osim toga, nekad je potrebno slati i pogrešne podatke (pogrešna konfiguraciona poruka, pogrešna komanda itd.) da bi se provjerilo ponašanje PDC uređaja ili aplikacije a to nije moguće sa stvarnim PMU uređajem jer on uvijek teži da radi ispravno. Arhitektura mrežnih resursa je izuzetno bitan faktor. Pomoću PMU simulatora se provjerava funkcionisanje, kašnjenje i stanje mreže. Na taj način moguće je detektovati smetnje u mreži koje onemogućavaju

ispravnu razmjenu poruka između PMU i PDC uređaja. Nakon svega, prije puštanja u rad, dobro bi bilo provjeriti da li sistem u cjelosti ispravno funkcioniše dok nije pod naponom pomoću PMU simulatora. Ispred PMU simulacionog softvera se postavlja nekoliko bitnih zahtjeva.

Prvi u nizu je precizno vremensko označavanje izmjerenih fazora. Kompleksnost ispunjena ovog uslova zavisi kako od hardvera tako i od operativnog sistema. Windows 7 nudi podrazumijevanu vremensku rezoluciju od 15.6 ms, a maskimalnu reda 5 ms, što je nedovoljno za potrebe PMU simulatora. Operativni sistemi bazirani na Linuxu i Unixu imaju bolju vremensku rezoluciju koja varira od 1 μ s do 50 μ s što omogućava održavanje TVE unutar dozvoljenih granica [7]. Vremenska sinhronizacija se postiže eksternim GPS uređajima i vremenskim protokolima kao što su NTP (*Network Time Protocol*) i PTP (*Precision Time Protocol*) [8][9].

Ukoliko PMU simulator treba da bude sposoban da proračunava vrijednost fazora na osnovu stvarno izmjerenih veličina napona, struje i frekvencije onda realizacija takvog softvera predstavlja još veći izazov. Kako dio koji se tiče obrade izmjerenih signala nije definisan IEEE C37.118 standardom postoji veliki broj različitih realizacija proračuna fazora [10]. Dobro poznavanje tehnika obrade signala i algoritama za proračun fazora je ključno u ovom koraku jer omogućava približavanje PMU simulatora stvarnom PMU uređaju. Trenutno dostupni PMU simulatori ne nude ovu mogućnost već samo slanje proizvoljno generisanih vrijednosti sinhrofazora ili unaprijed definisanih vrijednosti (pomoću CSV fajlova i sl.). Algoritmi za izračunavanje fazora na osnovu neobrađenih podataka se najčešće nalaze u obliku Matlab programa koji su neupotrebljivi u praksi.

Pored ključnih funkcionalnosti dobro bi bilo da se PMU simulator odlikuje nezavisnošću od operativnog sistema što je jedan od izazova zbog gore navedenog problema. Pored toga, programski kod mora optimizovan tako da koristi što manje računarskih resursa jer se na taj način može pokrenuti više instanci PMU simulatora na jednom računaru. Pogrešan izbor razvojne platforme ili neoptimalna realizacija mogu dovesti do neželjenih kašnjenja pri prenosu poruka.

Trenutno jedini dostupan nezavisni PMU simulator je dio iPDC paketa [11]. iPDC je softverski koncentrador sinhrofazora otvorenog koda. Korisnicima je ponuđena TCP i UDP konekcija kao i čuvanje primljenih podataka u MySQL bazi. Uz iPDC paket dostupan je i PMU simulator kompatibilan sa IEEE C37.188 standardom posljednje verzije. Napisan je u C programskom jeziku i nudi višenitno startovanje procesa. Ovaj PMU simulator je trenutno dostupan isključivo za Linux platforme ali kako je otvorenog koda moguće ga je adaptirati i za druge operative sisteme. Kompajliranje i konfiguriranje nije sasvim jednostavno, naročito na Linux distribucijama novijeg datuma zbog zastarjelih a neophodnih biblioteka koje nisu dostupne kroz zvanične repozitorijume. Na web stranici iPDC projekta je naznačeno da je projekat još uvijek u BETA fazi i da podrška za IEEE C37.118 – 2011 standard nije u potpunosti implementirana [11]. PMU simulator ima mogućnost za slanje podataka o sinhrofazorima iz CSV fajla kao i samostalno generisanje slučajnih vrijednosti. Pri brzini slanja od 60 paketa u sekundi potrebno je slati paket na svakih 16.666 ms. Prva verzija PMU simulatora nudila je generisanje i slanje paketa na svakih 16.666 ms pri čemu je sam proces generisanja i slanja trajao od 200 – 800 μ s što je rezultovalo kašnjenjem paketa. Ovaj problem dovodio je do toga da se paketi na strani PDC uređaja primaju brzinom od 58 – 59 paketa u sekundi što bi izazvalo nepravilnost u radu PDC uređaja koji očekuje 60 paketa u sekundi. Ukoliko se na ulazu PDC uređaja ne pojavi paket sa očekivanom vremenskom referencom on će pričekati neko unaprijed definisano vrijeme obično od 50 ms – 200 ms. Ako se ni nakon tog vremena očekivani paket ne pojavi po eventualnom prispjeću biće odbačen. Djelimično rješenje ovog problema jeste da se paket generiše i priprema za slanje za vrijeme razmaka od 16.666 ms između dva slanja. Na taj način bi paket mogao biti odmah poslat i kašnjenje bi bilo minimizovano.

Da bi se testirala ispravnost PMU simulatora koriste se slični softverski alati sa druge strane kanala. Jedna od takvih aplikacija je i PMU Connection Tester koja je dio openPDC softverskog paketa [12]. *PMU Connection Tester* nudi vizuelizaciju i snimanje podataka. *PMU Connection Tester* je dostupan samo na Windows-u jer je rađen na .NET platformi. Na Linux distribucijama se u slične namjene koristi Wireshark koji služi za generalno snimanje mrežnog saobraćaja.

Iz prethodne analize zaključujemo da je potrebno unapređenje postojećeg PMU simulatora kako u pogledu funkcionalnosti tako i u pogledu optimizacije koda i grafičkog interfejsa. Kako je PMU simulator otvorenog koda moguće je raditi i na razvijanju postojećeg.

Važno je napomenuti da postoje otvorene platforme za pravljenje niskobudžetnih PMU uređaja koji bi služili u istraživačke svrhe. Jedan od takvih projekata jeste *OpenPMU* baziran na *Labview* razvojnom okruženju i *National Instruments* hardveru za prikupljanje podataka. Na ovaj način moguće je konstruisati zamjenu komercijalnim PMU uređajima za svega 1000 američkih dolara [13]. Sličan projekat otvorenog koda je i *GridTrak* [14].

5. ZAKLJUČAK

U radu je data definicija sinhrofazora, opis mreže za prenos i prikupljanje podataka, te uvod u alate za podršku implementacije. PMU simulatori imaju značajnu ulogu u razvoju sistema za nadzor, kontrolu i upravljanje elektroenergetskim mrežama. Njihovo postojanje je neophodno zbog testiranja pouzdanosti i ispravnosti sistema kao i budućeg istraživačkog rada. Kako je trenutno na Internetu dostupan samo jedan PMU simulator korisnici gotovo da nemaju izbora. Takvo stanje otvara prostor za novi PMU simulator koji bi pokrio sve nedostatke postojećeg. Kompletno simulaciono rješenje bi moralo ponuditi pouzdan i brz algoritam za proračunavanje fazora pomoću izmjerenih podataka, nezavisnost od operativnog sistema (Windows 8.1 nudi maksimalnu vremensku rezoluciju od 1 ms), udoban grafički interfejs koji bi omogućio jednostavno upravljanje zasebnim instancama PMU simulatora i vizuelizaciju podataka.

LITERATURA

- [1] Power Failure in the Northeastern United States and the province of Ontario, *Federal Power Commission*, 1965.
- [2] Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations, *U.S.-Canada Power System Outage Task Force*, 2004.
- [3] Wide Area Monitoring, Protection, and Control Systems (WAMPAC), *National Electric Sector Cybersecurity Organization Resource*, 2012.
- [4] M. Adamiak, B. Kastenny, W. Premerlani, "Synchrophasors: Definition, measurement, and application", 59th Annual Georgia Tech Protective Relaying, Atlanta, GA, 2005.
- [5] IEEE Std C37.118.2, "IEEE Standard for Synchrophasor Data Transfer for Power Systems", *IEEE Power Engineering Society*, 2011.
- [6] IEEE Std C37.118.1, "IEEE Standard for Synchrophasor Measurements for Power Systems", *IEEE Power Engineering Society*, 2011.
- [7] Applications of PMU Simulator in PDC testing, *Philip M. Kersey*, 2012.
- [8] <http://tools.ietf.org/html/rfc958>, posljednji pristup 9.2.2014.
- [9] Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, *IEEE Instrumentation and Measurement Society*.
- [10] D. Dotta, J. Chow, L. Vanfretti, M.S. Almas, M.N. Agostini, "A Matlab-based PMU Simulator", *2013 IEEE PES General Meeting*, Vancouver, BC, Canada, 2013.
- [11] iPDC, *Nitesh Pandit, Kedar Khandeparkar*, 2012.
- [12] <http://openpdc.codeplex.com/>, posljednji pristup 9.2.2014.
- [13] *David M. Laverty, Robert J. Best, Paul Brogan, Iyad Al Khatib, D. John Morrow*, "The OpenPMU Platform for Open-Source Phasor Measurement", *IEEE Transactions on Instrumentation and Measurement*, 2013.
- [14] <http://wiki.gridtrak.com/wiki/index.php/GridTrak>, posljednji pristup 9.2.2014.

PROCENA INHARMONIČNOSTI KOPIJE ANTONIUS STRADIVARIUS VIOLINE ESTIMATION OF INHARMONICITY OF THE COPY OF ANTONIUS STRADIVARIUS VIOLIN

Zoran Milivojević, Zoran Veličković, *Visoka tehnička škola, Niš, Srbija*
Dragiša Balanesković, *Univerzitet u Nišu, Fakultet umetnosti, Niš, Srbija*

Sadržaj: U radu je prikazan algoritam za procenu inharmoničnosti žičanih muzičkih instrumenata. U drugom delu rada prikazani su rezultati primene algoritma na proceni inharmoničnosti kopije Antonius Stradivarius violine iz perioda kraj XIX i početka XX veka. Analiza je rađena za dva načina pobude žica, i to: guđenjem i b) picikato tehnikom. Rezultati su prikazani tabelarno i grafički. Na kraju je izvršena komparativna analiza rezultata.

Ključne reči: Fundamentalna frekvencija, Harmonik, Inharmonik.

Abstract: This paper presents the algorithm for estimation of inharmonicity of the string musical instruments. The second part of this paper presents the results of the application of this algorithm in estimation of inharmonicity of the copy of Antonius Stradivarius violin from the end of the 19th century or the beginning of the 20th century. The analysis was made for two ways of string arousal: a) by using a bow and b) by using the picicato technique. The results are given in tables and graphics. In the end the comparative analysis of results was made.

Key Words: Fundamental frequency, Harmonicity, Inharmonicity.

1. UVOD

Žičani muzički instrumenti proizvode ton, odnosno generišu akustički talas oscilovanjem žice. Oscilovanje žice je dobro objašnjeno u fizici i akustici, potkrepljeno odgovarajućim mehaničko-akustičkim modelom i opisano odgovarajućim diferencijalnim jednačinama. Nakon mehaničke pobude (izvođenja iz ravnotežnog položaja trzanjem, udarcem ili gudalom) žica počinje da osciluje sa tendencijom smirivanja i vraćanja u ravnotežni položaj. Žica vibrira sa osnovnom odnosno fundamentalnom frekvencijom koja zavisi od dimenzija žice (dužina i prečnik), materijala od koga je napravljena kao i od sile zatezanja. Pored osnovne frekvencije, zbog složenog oscilovanja žice (javljanje talasa na $1/2$, $1/4$, $1/8$, ... dužini žice) javljaju se akustičke komponente talasa na frekvencijama koje predstavljaju celobrojne umnoške osnovne frekvencije. U tehnici se ove spektralne komponente nazivaju harmonicima. Teorija muzike složenu strukturu oscilovanja žice i generisani ton opisuje pomoću alikvota (*lat. aliquoties*, nekoliko puta), gde alikvote predstavljaju harmonične komponente generisanog tona. Detaljna matematička analiza pokazuje da se u sastavu jednog tona preko alikvota sadrže svi tonovi (aliquote raznih tonova se međusobno preklapaju). Različiti broj prisutnih alikvota i njihova različita relativna jačina u sklopu ukupnog zvučanja, određuju boju zvuka [1]. Alikvote se nazivaju i parcijalnim tonovima, odnosno parcijalima (*engl. partials*).

Detaljnija analiza oscilovanja žice pokazuje da, usled parametara žice i sile zatezanja, frekvencije harmonika nisu sa celobrojnim umnoškom osnovne frekvencije, odnosno pojavljuje se inharmoničnost. Inharmoničnost oscilovanja žice opisuje se posredstvom koeficijenta inharmoničnosti.

Pojava inharmoničnosti nužno dovodi do narušavanja alikvotnosti. Sagledavajući prisustvo inharmoničnosti kod nekih instrumenata parcijali mogu biti sa frekvencijama koje su: a) celobrojni umnoški fundamentalne frekvencije (harmoni) i b) necelobrojni umnoški (inharmonici) [2]. Ton dobijen na inharmoničnom instrumentu nije nužno neprijatan. U [3] izneta je tvrdnja da *blago* inharmoničan ton poseduje *određenu toplinu*.

Kvalitet žičanih muzičkih instrumenata moguće je, pored ostalih parametara, izražavati i pomoću stepena inharmoničnosti. Žice su zategnute velikom silom tako da je njihova elastičnost smanjena. Posledica toga je da su frekvencijske pozicije parcijala na pozicijama necelobrojnih umnožaka fundamentalne frekvencije. Samim tim instrument sa ovakvim žicama nije harmoničan. Pored krutosti žice povećanju inharmoničnosti instrumenta doprinosi i karakter akustičke impedanse rezonatorske ploče kod klavira, odnosno rezonatorskog tela kod gitare, violine i dr. [3]. Inharmonici su izraženi kod tonova sa nižim frekvencijama. Na višim frekvencijama inharmoničnost je teže detektovati. Razlog dobrog detektovanja kod tonova sa nižom frekvencijom nalazi se u činjenici da je veliki broj parcijala u frekvencijskom opsegu gde čovek dobro čuje. Kod tonova sa višim frekvencijama svega nekoliko prvih parcijala, koji ne iskazuju veliku inharmoničnost, se nalazi u ovoj oblasti. Analizom inharmoničnosti moguće je donositi zaključke o vrsti instrumenta koji proizvodi ton. U radu opisan je algoritam za automatsku klasifikaciju nota između piana i gitare [4]. Kod gitare, kao i nekih drugih instrumenata, je jedan ton moguće odsvirati na tri različite žice. Analizom stepena inharmoničnosti moguće je odrediti odsvirani ton, žicu i polje gde je žica pritisnuta [5].

U ovom radu prikazani su rezultati analize inharmoničnosti violine koja je kopija Antonius Stradivarius violine, proizvedene u Češkoj u periodu kraj IXX i početak XX veka. U prvom delu rada opisan je algoritam koga su autori predložili u radu [6]. Algoritam se sastoji od dve celine: a) procene frekvencijske pozicije parcijala i b) procene inharmoničnosti. Procena frekvencijske pozicije bazira se na pikovanju maksimuma u spektru i primene PCC konvolucionog jezgra. Parametri ovog dela algoritma su deo rezultata autora objavljenih u [7] i [8]. Od većeg broja objavljenih formula za računanje inharmoničnosti autori su primenili formulu iz [9]. U drugom delu rada izvršena je analiza inharmoničnosti violine i to za pobudu žica: a) guđenjem i b) trzanjem (picikato tehnika). Dobijeni rezultati upoređeni su sa rezultatima inharmoničnosti koncertnog klavira Steinway D [10], pijanina August Förster (proizvedenog u Čehoslovačkoj 1971. god.) [11], i električne gitare Fender Stratokaster [6].

Organizacija rada je sledeća. U sekciji II prikazan je algoritam za procenu inharmoničnosti. U sekciji III prikazani su eksperimentalni rezultati i komparativna analiza. Sekcija IV je zaključak.

2. INHARMONIČNOSTI KOD MUZIČKIH INSTRUMENTATA SA ŽICAMA

2.1 Koeficijent inharmoničnosti žice koja vibrira

Teorija muzike kod definisanja frekvencijskog sastava tona podrazumeva harmoničnost, odnosno da su harmonici (parcijali) celobrojni umnošci fundamentalne frekvencije, što se matematički predstavlja kao:

$$f_k = k \cdot f_0, \quad k = 1, 2, \dots \quad (1)$$

gde je f_0 fundamentalna frekvencija, k redni broj parcijala a f_k frekvencija parcijala. Frekvencijsko pomeranje parcijala od frekvencijske pozicije harmonika predstavlja inharmoničnost tona. Inharmoničnost je definisana preko koeficijenta inharmoničnosti β :

$$f_k = k \cdot f_0 \sqrt{1 + \beta \cdot k^2}, \quad k = 1, 2, \dots \quad (2)$$

Koeficijent inharmoničnosti β zavisi od vrste materijala žice i može da se izračuna na osnovu:

$$\beta = \frac{\pi^3 \cdot Q \cdot d^4}{64 \cdot l^2 \cdot F} \quad (3)$$

gde je Q Jungov modul elastičnosti materijala od kojeg je napravljena žica, d prečnik žice, l dužina žice a F sila zatezanja.

2.2 Algoritam za procenu parcijala

Algoritam za procenu parcijala zasniva se na analizi spektra signala x . Najpre se primenom diskretne Furijeove transformacije izračuna spektar X . Nakon toga se metodom pikovanja spektra određuje pozicija maksimalne spektralne komponente koja predstavlja fundamentalnu frekvenciju [7]. U cilju povećanja preciznosti procene primenjuje se

parametarska kubna konvolucija i izračunava fundamentalna frekvencija f_0 [7,8]. Procena parcijala obavlja se algoritmom prikazanom na sl.1. Algoritam se sastoji od sledećih koraka:

Ulaz: Spektar X_b , frejma x_b , signala x , procenjena fundamentalna frekvencija f_0 , broj parcijala N_p .

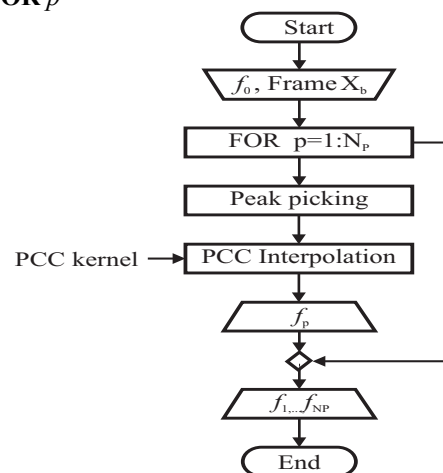
Izlaz: parcijali f_p .

FOR $p=1:N_p$

Korak 1: pikovanje p -te spektralne komponente.

Korak 2: Izračunavanje p -tog parcijala primenom PCC interpolacije.

END FOR p



Slika 1. Algoritam za procenu parcijala.

2.3 Algoritam za procenu inharmoničnosti

Algoritam za procenu inharmoničnosti sastoji se od sledećih koraka [6,9]:

Ulaz: Fundamentalna frekvencija f_0 , parcijali f_p , broj parcijala N_p .

Izlaz: koeficijent inharmoničnosti β .

Korak 1: određivanje frekvencija harmoničnih komponentata:

$$f_k = k \cdot f_0, \quad k = 1, 2, \dots, N_p \quad (4)$$

Korak 2: Izračunavanje koeficijenata inharmoničnosti prema algoritmu iz [9]:

$$\beta = \frac{\left(f_k \frac{m}{k}\right)^2 - f_m^2}{k^2 f_m^2 - m^2 \left(f_k \frac{m}{k}\right)^2}, \quad (5)$$

gde su m i k parcijali a f_m i f_k odgovarajuće frekvencije parcijala.

3. EKSPERIMENTALNI REZULTATI I ANALIZA

U nastavku rada biće analizirani koeficijenti inharmoničnosti violine kopije Antonius Stradivarius, proizvedene u Češkoj krajem IXX ili početkom XX veka (prema ekspertizi sprovedenoj u Nemačkoj 1993. god.). Ugrađene žice su tipa Dominant Set 135 B, svetski priznatog

proizvođača Thomastic Infeld Vienna (žica e^2 : chrome steel, ball end, žica a^1 : synthetic core, aluminium wound, žica d^1 : synthetic core, silver wound, žica g : synthetic core, silver wound). Analiza će podrazumevati procenu inharmoničnosti svih žica i određivanje srednje vrednosti koja će predstavljati inharmoničnost instrumenta i to za dva načina pobude žica: a) guđenjem i b) picikato tehnikom (trzanjem žice).

Za potrebe eksperimenta formirana je baza signala. Nad signalima primenjen je algoritam opisan u sekciji 2. Parametri algoritma su $T=0.66s$, $NFFT=10*218$, $m=6$, $k=10$.



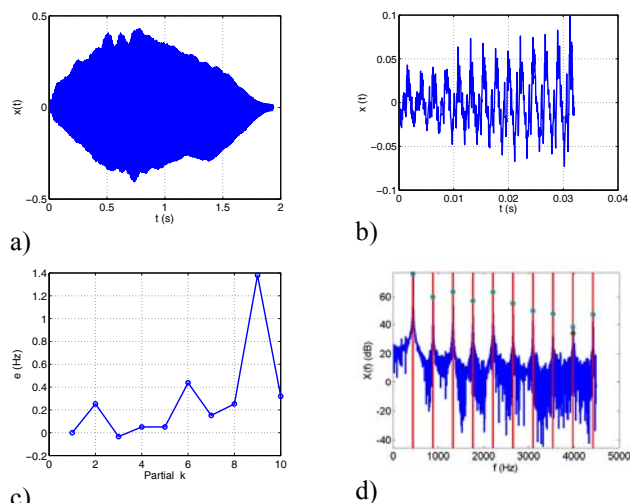
Slika 2. Testirana violina.

3.1 Baza

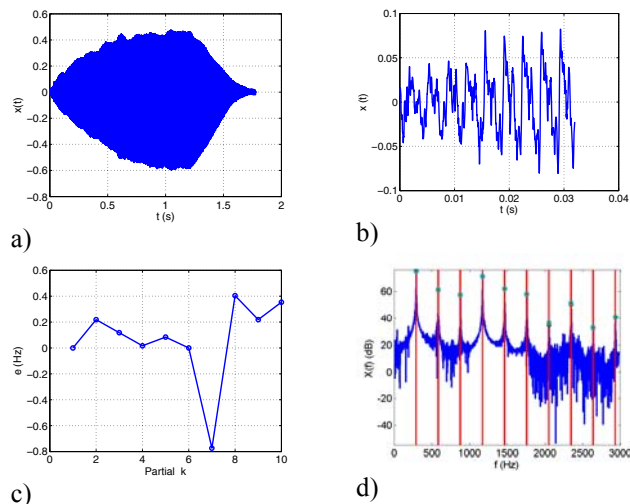
Bazu čine svi tonovi (jedna oktava) na žicama $E5$ (e^2), $A4$ (a^1), $D4$ (d^1) i $G3$ (g). Snimanje je obavljeno sa frekvencijom odmeravanja $f_s=44.1$ kHz i 16 bita po odmerku.

3.2 Rezultati

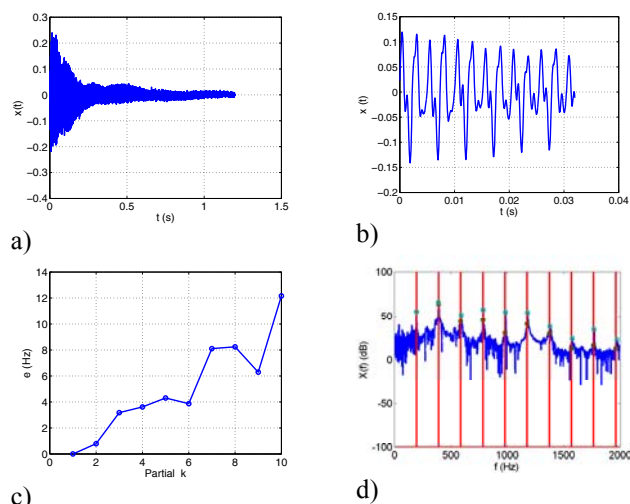
Na sl.3 i sl. 4 prikazadi su vremenski dijagrami koji se odnose na tonove dobijene guđenjem (sl.3 ($G3$), sl. 4 ($D4$)) dok su na sl. 5 do sl. 7 vremenski dijagrami tonova dobijenih picikato tehnikom (sl.5 ($G3$), sl. 6 ($D4$) i sl. 7 ($A4$)). Na svakoj slici a) predstavlja vremenski oblik kompletnog signala, b) vremenski oblik frejma trajanja 32 ms, c) razliku frekvencija između harmoničnih i inharmoničnih komponenata i d) deo amplitudske karakteristike di desetog harmonika. Vertikalna linija označava poziciju harmonijske komponente. U tabeli 1 prikazane su vrednosti inharmoničnosti izračunate primenom algoritma opisanog u sekciji II.



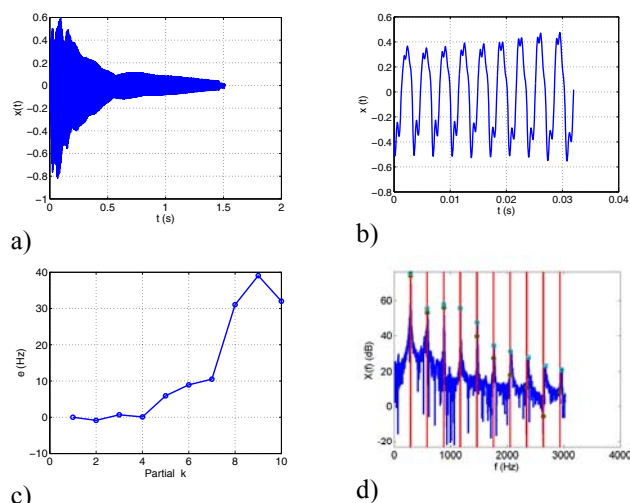
Slika 3. Signal tona $G3$ proizveden guđenjem.



Slika 4. Signal tona $D4$ proizveden guđenjem.



Slika 5. Signal tona $G3$ proizveden trzanjem žice.



Slika 6. Signal tona $D4$ proizveden trzanjem žice.

Tabela 1. Koeficijent inharmoničnosti tonova praznih žica violine.

	F0 [Hz]	Gudanje	Picikato
		$\beta(x10^{-6})$	$\beta(x10^{-4})$
G3	196	6.05	1.31
D4	293.66	1.38	2.86
A4	440	3.55	3.82
E5	659.25	2.06	1.13
		$\beta_{sr}=3.26$	$\beta_{sr}=2.28$

3.3 Analiza rezultata

Na osnovu rezultata prikazanih na sl.3 – sl.6 i u tabeli 1 može se zaključiti da:

a) koeficijenta inharmoničnosti kod tehnike picikato je $2.28 \times 10^{-4} / 3.26 \times 10^{-6} = 69.93$ puta veća u odnosu na pobudu gudalom,

b) vrednosti koeficijenta inharmoničnosti kod zvuka proizvedenog gudalom je reda 10^{-6} i može se smatrati da inharmoničnost ne postoji.

c) kada se koristi tehnika picikato inharmoničnost violine je u približno istim granicama kao pijanino August Forster $(1 \div 2) \times 10^{-4}$ [11] i električna gitara Stratokaster $(0.8 \div 2.5) \times 10^{-4}$ [6] dok je za red veličine veća od koncertnog klavira Steinway D $(0.6 \div 0.8) \times 10^{-4}$ [10].

Navedeni rezultati ukazuju na činjenicu da je kod instrumenata gde se zvuk dobija trzanjem inharmoničnost značajna zbog nepravilnosti oscilovanja zice, kada se nakon pobude prepusti slobodnom oscilovanju. Sa druge strane, kod pobude gudalom žica se ne prepusta slobodnim oscilacijama i inharmoničnost je zanemarljiva.

Dalja istraživanja biće usmerena ka testiranju većeg broja instrumenata iz grupe tržačkih i gudačkih instrumenata, u saglasno rezultatima izvlačenja globalnog zaključka.

4. ZAKLJUČAK

U radu su prikazani rezultati testiranja violine kopije Antonius Stradivarius proizvedene u Češkoj krajem XIX ili početkom XX veka, u odnosu na inharmoničnost. Testiranje je sprovedeno za dva načina proizvodjenja tonova, i to gudalom i trzanjem žice (picikato tehnika). Rezultati pokazuju da je kod upotrebe gudala inharmoničnost zanemarljiva (koeficijent inharmoničnosti reda 10^{-6}) dok je kod trzanja žice koeficijent inharmoničnosti reda 10^{-4} . U tom slučaju inharmoničnost violine je istog reda kao i kod pijanina August Forster kao i kod električne gitare Fender stratokaster.

LITERATURA

- [1] M. Tajčević, *Osnovna teorija muzike*, Prosveta, Beograd, 1962.
- [2] S. Lixin, Z. Junxing, and P. Lingbin, "Piano fundamental frequency estimation algorithm based on weighted least square method", in *Proc. Int. Forum Inf. Technol. Applicat., IFITA '09*, Vol. 2, pp. 155–158, May 2009.
- [3] L. Ortiz-Berenguer, J. Casajus-Quiros, M. Torres-Guijarro, J. Beracoechea and J. Aranda, "Simple modeling of piano inharmonicity due to soundboard impedance", In *Proc. 120th AES Convention*, Preprint, Vol. 66, pp. 1–6, 2006.
- [4] D. Fragoulis, C. Papaodysseus, M. Exarhos, G. Rousopoulos, T. Panagopoulos and Dimitrios Kamarotos, "Automated Classification of Piano-Guitar Notes", *IEEE Transactions On Audio, Speech, And Language Processing*, Vol. 14, No. 3, pp. 1040-1050, May 2006.
- [5] I. Barbancho, L. Tardon, S. Sammartino, A. Barbancho, "Inharmonicity-Based Method for the Automatic Generation of Guitar Tablature", *IEEE Transactions On Audio, Speech, And Language Processing*, Vol. 20, No. 6, pp. 1875-1868, August 2012.
- [6] Z. Milivojević, D. Brodić, "Inharmoničnost bas sekcije Fender Stratocaster električne gitare", *Informacione Tehnologije IT'13*, Zbornik str. 89-92, Žabljak, Crna Gora, 2013.
- [7] Z. Milivojević, M. Mirković, "Estimation of the fundamental frequency of the speech signal modeled by the SYMPES method", *International Journal of Electronics and Communications (AEU)*, vol. 63, no. 3, pp. 200-208, 2009.
- [8] Z. Milivojević, D. Brodić, "Estimation of the fundamental frequency of the speech signal compressed by G.723.1 algorithm applying PCC interpolation", *Journal of Electrical Engineering*, vol. 62, no. 4, pp. 181-189, 2011.
- [9] M. Hodgkinson, J. Wang, J. Timoney, V. Lazzarini, "Handling inharmonic series with median-adjustive trajectories", *12th Int. Conference on Digital Audio Effects DAFx-09*, Como, Italy, Sept. 1-4, 2009.
- [10] A. Galembo and A. Askenfelt, "Signal Representation and Estimation of Spectral Parameters by Inharmonic Comb Filters with Application to the Piano", *IEEE Transactions on Speech and Audio Processing*, Vol. 7, No. 2, pp. 197-203, March 1999.
- [11] M. Rajković, D. Milosavljević, Z. Milivojević, "Inharmoničnost bas sekcije August Forster pijanina", *INFOTEH'13*, Sekcija Računarski sistemi i softver – informacioni sistemi, str. 980-986, Jahorina, 2013.

LTspice IV KAO EDUKATIVNO SREDSTVO U NASTAVI ANALIZE ELEKTRICNIH KOLA LTspice IV AS EDUCATIONAL TOOL FOR TEACHING ELECTRICAL CIRCUIT ANALYSIS

Biljana Chitkusheva Dimitrovska, Maja Kukusheva, Vlatko Chingoski, *Faculty of Electrical Engineering, University „Goce Delcev“ – Stip*

Sadržaj: U ovom radu predlaže se uvođenje osnovnih koncepata računarskog program za analizu LTspice IV, kao neprocenjiv alat u nastavi studenata predmeta analize električnih kola. Za studente, korišćenje ovog softverskog alata pruža mogućnost da rade samostalne analize ili da provere dobivene rezultate svojih analiza i time budu sigurni da su te analize napravljene na ispravan način. Kompjuterske simulacije u LTspice IV su vrlo značajne u procesu učenja omogućavajući studentu da eksperimentiše sa promenama i da vidi efekte koje te promene imaju na električna kola. Posmatrajući rezultate svoje kompjuterske simulacije, broičano i osobito grafički, student može da stekne intuitivno shvatanje ponašanja električnih kola bez zastrašivanja od upotrebe apstraktne matematičke analize.

Abstract: This paper proposes the introduction of the basic concepts of computer analysis program LTspice IV, as an invaluable tool in teaching students subjects of electrical circuits analysis. For students, the use of this software tool provides the ability to do independent analysis or verification of the obtained results within their analysis and thus be sure that these analyzes are made in the right way. Computer simulations using LTspice IV are very important in the learning process allowing students to experiment with changes to see the effects that these changes have on an electrical circuits. Observing the results of their computer simulations, numerically and graphically in particular, the students can gain an intuitive understanding of the behavior electric circuits without intimidation from the use of abstract mathematical analysis.

1. INTRODUCTION

Having understood the fundamental laws of circuit theory (*Ohm's law* and *Kirchhoff's laws*), we are now prepared to apply these laws to develop two techniques for circuit analysis: **nodal analysis**, which is based on a systematic application of Kirchhoff's Current Law (**KCL**), and **branch analysis** which is based od Kirchhoff's Voltage Law(KVL).

With these two techniques, one can analyze any linear circuit by obtaining a set of simultaneous equations that are simultaneously solved to obtain the required values of currents and/or voltages. One method of solving simultaneous equations involves Cramer's rule or method of replacing variables, which yealds the unknown variables. Another method of solving simultaneous equations is to use **LTspice**, or any other circuit simulation software, simulate the electric circuit and solve it for its parameters.

LTspice IV is a free SPICE (*Simulation Program for Integrated Circuit Emphasis*) simulator with schematic capture from *Linear Technology Ltd*. Linear Techology (**LT**) is one of the industry leaders in production of varius analog and digital integrated circuits [2]. LT provides a complete set of SPICE models mmostly using LT components. Circuits may contain resistors, capacitors, inductors, mutual inductors, independent voltages and current sources, a few types of dependent or independent sources, transmission lines, switches, and several devices: including diodes, BJTs, JFETs, MOSFETs. Circuits with large number (*almost infinite*) number of components can be simulated. One can think of LTspice as a nodal network solver that outputs all node voltages and branch currents, or simulate their changes

for prescribed time period. One node must be named "0" (*the ground node or the reference node*) and this node is used as a reference node for calculations of all other node voltages.

2. SHEMATIC ENTRY IN LTSPICE IV

Simulation of an electric circuit using LTspice IV can be done in two steps:

- drawing (*editing*) or entering circuit using schematic capture, and
- defining the desired type of simulation and execute the simulation or performing circuit analysis.

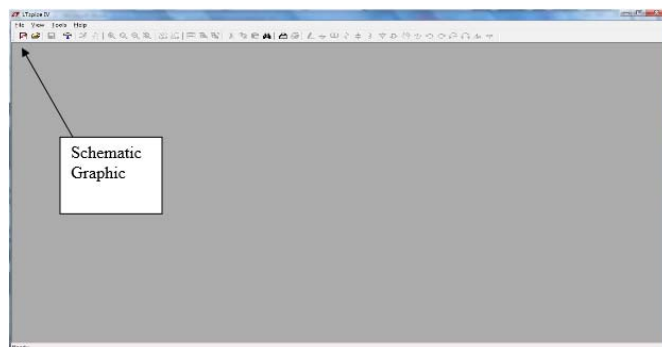


Fig. 1: Starting window in LTspice.

Drawing a circuit is easier when the grid mode is active. To activate the grid mode, click on **View** and select command **Show Grid**.

There are many buttons on the main menu for selection of basic circuit elements such as resistors, capacitors, inductors, diodes, wires, as well as a ground node (GND). The Ltspice IV's **Main toolbar** is shown on Fig. 2.

3. CIRCUIT ANALYSIS WITH LTSPICE IV

As was already mentioned, the LTSpice IV simulation software can be used as circuit analysis program which can easily determine all voltage and current values for any previously set circuit configuration and given numerical values for circuit components. We would demonstrate this, with several simple problems. These problems are usually solved by the students during their study of basic electrical circuit theory or any other basic or advance study subject that involves electric circuit analysis.

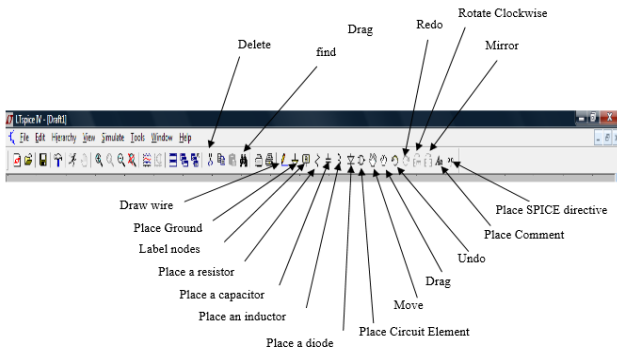


Fig. 2: Main toolbar of schematic editor for LTSpice IV.

Example1: Find the branch currents I_1 , I_2 , and I_3 for a simple electric circuit show on Fig. 3.

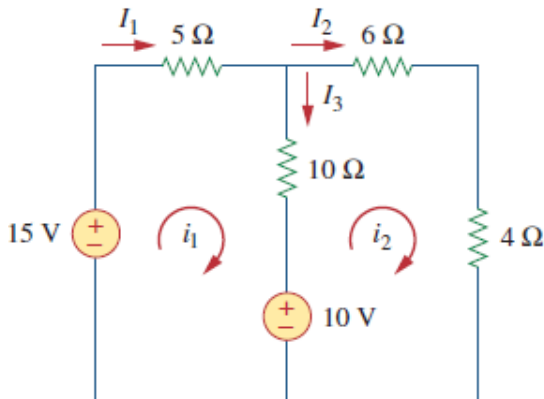


Fig. 3: Simple electric circuit as student task.

Usually students use some solving techniques such as contour currents or unknown node voltage analysis to find the unknown circuit parameters, voltages and/or currents.

Here, we could use the *Method of contour currents* to solve the above circuit for numerical values of all node voltages and branch currents. Consequently, one can obtain the following general system of equations:

$$\begin{aligned} R_{11} i_1 + R_{12} i_2 &= E_I \\ \pm R_{21} i_1 + R_{22} i_2 &= E_{II} \end{aligned}$$

Where, i_1 and i_2 are the unknown contour currents, R_{11} , R_{12} , R_{21} and R_{22} , are the equivalent contour resistances (*self and mutual contour resistance*), and E_I and E_{II} are the equivalent voltage source per contour. If we replace the numerical values given as in Fig.3, we could obtain the followings:

$$\begin{aligned} 15 i_1 - 10 i_2 &= 15 \\ -10 i_1 + 20 i_2 &= 10 \end{aligned}$$

Using the method of substitution to solve these system of two unknown currents with two equations, we get the values for the contour currents $i_1 = i_2 = 1A$. Thus, for the values of the unknown branch current we have:

$$I_1 = i_1 = 1A, \quad I_2 = i_2 = 1A, \quad I_3 = 0A$$

Next, we could show, how easy one can solve this electric circuit and get the numerical values and, if it is requested the graphic representation of all branch currents using the electric circuit solver and simulator, LTSpice IV.

First, we open the LTSpice program and start drawing (*editing*) the electrical circuit shown in Fig. 4. The analyzed electrical circuit contains two voltage generators and four resistors. After entering all circuit elements by clicking on the designated icons and setting their numerical values as parameters (*click on each component and set the desired parameters and values*), we place the circuit elements inside the program's workspace and finally connect the entire electric circuit with wires (*wiring*). The obtained electric circuit with its parameters shown inside the LTSpice IV program's workspace is shown on Fig. 4.

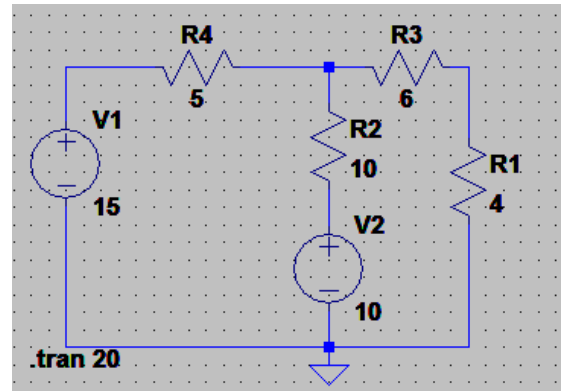


Fig. 4: Electric circuit built in LTSpice IV.

At this point, our circuit is fully defined and we could continue to the next step, to solve or simulate the circuit behavior. The only relevant simulation for this electric circuit is a simple DC analysis, known in LTSpice IV as *DC operating point*. This analysis we set by clicking on the icon *Simulate>Edit Simulation Cmd* from the top toolbar and select *DC op pnt*. This is called an *LTSpice IV directive*.

To run a circuit (*execute the solver*), we can select from the file menu icon *Simulate* and *Run*, or we can simply click on the *Run* button from the main toolbar. Next, we could see a new window appearing, containing the simulation results, or in our case solution of the electric circuit in a simple table manner as shown in Fig. 5. As can be seen, the solution is exactly the same as in previous solution method using contour current method as a circuit solution method.

We must have in mind, that this was only a simple example, only two currents to solve for. However, if we have even a little more complex electric circuit, e.g. four or five contour currents, than the advantages of using LTSpice IV for obtaining the unknown currents instead of solving by hand the same circuit by means of any other solution method are obviously large.

Next, we would present the advantages of utilizing LTSpice IV software in the education of young students by

means of another example very often used in the classwork – the solution of an electric circuit using the **Thevenin's theorem**. Thevenin's theorem provides a solution technique by which one, usually called *the fixed part of circuit*, is replaced by an equivalent circuit [1]. Our major concern using this technique is to find the **Thevenin equivalent voltage** V_{TH} and the **Thevenin equivalent resistance** R_{TH} .

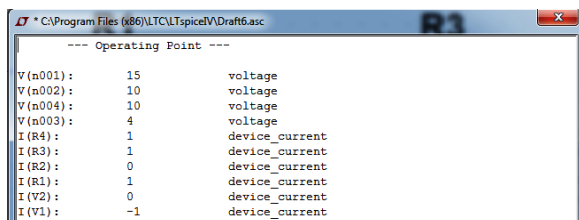


Fig. 5: Simulation (solution) results file.

Example 2: Find the equivalent Thevenin circuit of the circuit shown in Fig. 6 to the left of the terminals **a-b**. Then, find the current through the resistance $R_L = 6,16 \Omega$.

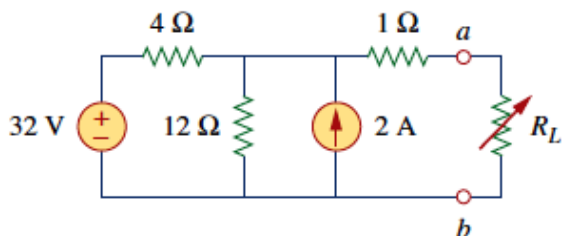


Fig. 6: Electric circuit for the Example 2.

We find equivalent resistance R_{TH} by turning off the 32V Voltage source (replacing it with the short circuit), and the Current source of 2A (replacing it with an open circuit).

Thus,

$$R_{TH} = 4 \parallel 12 + \frac{4 \cdot 12}{4 + 12} + 1 = 4\Omega$$

To find the equivalent voltage V_{TH} , we consider the electric circuit given on Fig. 7.

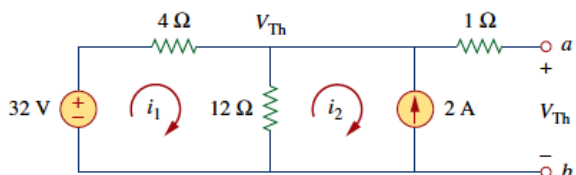


Fig. 7: Modified circuit to obtain the equivalent voltage V_{TH} .

Applying mesh analysis for the two loops, we obtain:

$$-32 + 4 i_1 + 12 (i_1 - i_2) = 0.$$

Solving for i_1 , we get $i_1 = 0,5 A$. Thus,

$$V_{Th} = 12 (0,5 + 2,0) = 30 V.$$

Respectively, the Thevenin equivalent circuit for the original electric circuit given on Fig. 6, get the configuration shown on Fig. 8.

The current through the R_L is:

$$I_L = \frac{V_{Th}}{R_{Th} + R_L} = \frac{30}{4 + R_L}$$

When $R_L = 6\Omega$, $I_L = \frac{30}{10} = 3A$. When $R_L = 16\Omega$, $I_L = 1,5A$.

If we built this electrical circuit in LTspice IV, the current through the R_L is determining in a very simple way without any calculations and complex methods.

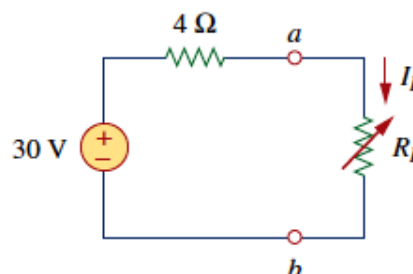


Fig. 8: Thevenin equivalent circuit for Example 2.

Using the same methodology described for the previous Example 1, one can draw the electric circuit of Example 2 in LTspice IV (see Fig. 9 – above).

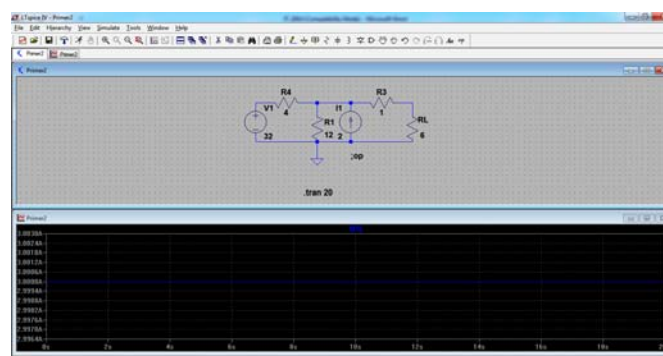


Fig. 9: Electric circuit of Example 2, built in LTspice IV (above), and the solution for the unknown current (below).

First, we could obtain the values of the current through resistor $R_L = 6\Omega$. For this value of R_L , we simulate the electric circuit shown in Fig. 6. After simulation, we get the results in output text file (.op file).

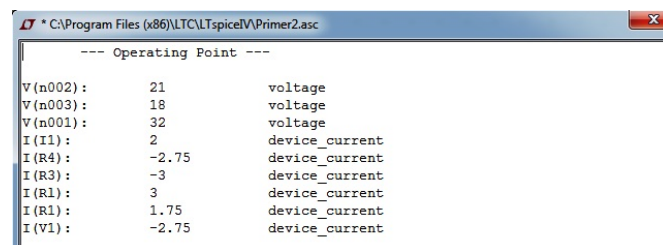


Fig. 10: Obtained results for $R_L = 6\Omega$.

LTspice has a user-friendly interface to show the results of simulations in two ways. One way is to display the results in a text file (.op file) (see Fig. 10), and the other way is using graphic display, as shown in Fig. 9 – below.

The third example of the advantages that simulation program LTspice IV provides for students and teachers in solving the subjects of electric circuit analysis, concerns about time-varying electric circuit analysis.

Example 3: Determine the time variation of the $V_0(t)$ in the circuit given on Fig. 11.

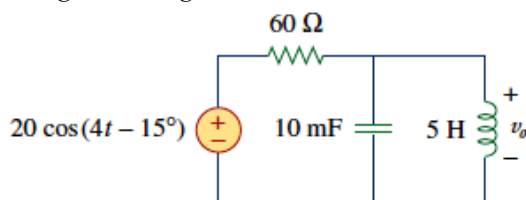


Fig. 11: Time-varying electric circuit.

To do the circuit analysis in frequency domain, we must first transform the time domain circuit, and its parameters into the phasor domain equivalent circuit. This transformation produces:

$$V_s(t) = 20 \cos(4t - 15^\circ) \Rightarrow V_s = 20 \angle -15^\circ, \omega = 4$$

$Z_1 = 60 \Omega$ is a impedance of the 60Ω resistor and $Z_2 = \frac{1}{j\omega C} \parallel j\omega L = j100 \Omega$ is a impedance of the parallel combination of the 10 mF capacitor and the 5 H inductor. Using the voltage-division principle, one can obtain:

$$V_0 = \frac{Z_2}{Z_1 + Z_2} V_s = 17.15 \angle 15.96^\circ \text{ V}$$

Now, we can convert this back to the time domain and obtain the final time-varying voltage value:

$$V_0(t) = 17.15 \cos(4t + 15.96^\circ) \text{ V.}$$

Let us now, solve the same problem using LTspice IV. First, we built the electrical circuit shown in Fig. 1. The electrical circuit contains one sinusoidal voltage source, resistor, capacitor and inductor, all of them with their respective numerical values. For the passive components (R, L, C), point at the component and do a right click. A dialogue box will appear and we could type the numerical value of the component. For the voltage supply, we do a right click on it and then click on the button **Advanced**. This enables us to set a variety of different voltage source waveforms. The one that we need for this analysis is on the left side of the dialogue box, as shown in Fig. 12.

Next, we do simulation of the circuit by clicking **Simulate > Edit Simulation Cmd** and type the **Stop time**. Finally, as a result we get the time changes of the unknown voltage $V_0(t)$ shown on Fig. 13. Later, various analysis can be performed of the obtained results such as reading of the frequency, max values, min values, attenuation factor, time constant, etc.

4. CONCLUSION

In this paper, we present how electric simulation program LTspice IV can be utilized to support teachers in education of students on the subject of electric circuit analysis.

We showed that LTspice is easy to use, very flexible and highly fast and accurate software for teaching electric circuit analysis. Three simple examples are given in order to provide

step-by-step insight how easy and resourceful could be this program for education of future electrical engineers. In addition to its solver capabilities, the program provides advanced graphic routines to visually grasp the nature of the process that happens inside simple, and especially very complex time and frequency varying electric circuits.

In such cases, use of LTspice simulation software allows for modeling of electric circuits and is an invaluable analysis tool. LTspice simulation software engages the user by integrating them into the learning experience. These kinds of interactions actively engage learners to analyze, synthesize, organize, and evaluate content and result in learners constructing their own knowledge.

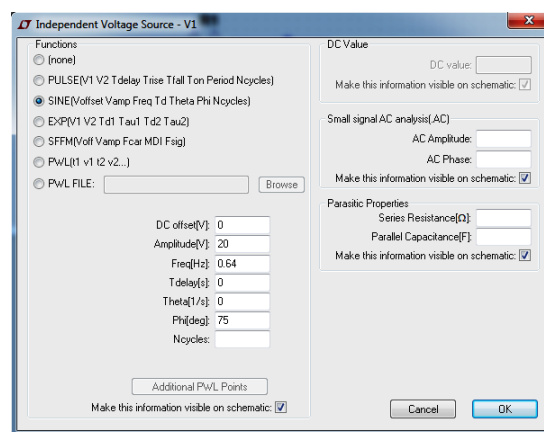


Fig. 12: Input screen for sinusoidal voltage source.

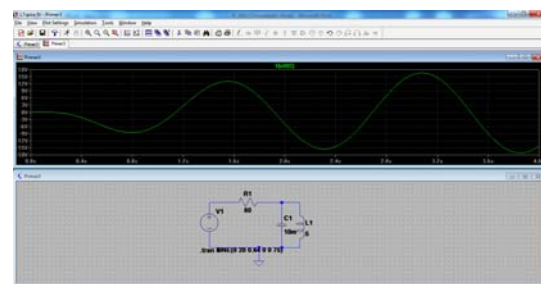


Fig. 13: Obtained time-varying function of voltage $V_0(t)$.

REFERENCES

- [1] Charles K. Alexander, Matthew N. O. Sadiku, **“Fundamentals of Electrical Circuits,”** 5th ed., The McGraw - Hill Companies, New York, 2013.
- [2] **LTspice IV tutorial,** www.linear.com/ltspice
- [3] Влатко Чингоски, **„Основи на CAD/CAM во електротехника,”** University „Goce Delcev”, Stip, 2013.
- [4] Susan A. Riedel and James W. Nilsson, **“Introduction to Pspice for electric circuits,”** Pearson Education, Inc. 2007.

POREĐENJE FUNKCIONALNOG I IMPERATIVNOG PRISTUPA PROGRAMIRANJU COMPARISON OF THE FUNCIONAL AND IMPERATIVE APPROACH TO PROGRAMMING

Petar Radunović, Tijana Vujičić, Ivan Knežević, *Fakultet za informacione tehnologije, Univerzitet Mediteran, Podgorica*

Sadržaj: Imperativni pristup koji koristi objektno-orijentisane (OO) programske jezike danas predstavlja dominantan pristup u razvoju softverskih aplikacija. U ovom radu je predstavljeno poređenje funkcionalnog i imperativnog pristupa, uz detaljniji opis funkcionalnog pristupa i relativno novog funkcionalnog programskog jezika Clojure.

Abstract: Imperative approach, these days, represents dominant approach in software development, using object-oriented programming (OOP) languages like C# and JAVA. This paper presents a comparison between imperative and functional approach in software development, with more elaborate description of functional approach, as well as of the relatively new programming language named Clojure.

Keywords: C#, Clojure, imperative, functional, object-oriented, programming

1. UVOD

Imperativni pristup razvoju softverskih aplikacija je danas tvrdo ukorijenjen i opšte prihvaćen kao glavni pristup u razvoju. Sa druge strane funkcionalni pristup je izgubio na značaju tokom ere OO programskih jezika, ali pojavom modernih funkcionalnih jezika kao što je Clojure, ponovo dobija na značaju.

Cilj ovog rada je poređenje navedenih pristupa programiranju, kako sa strane performansi softverske aplikacije koja nastaje kao krajnji rezultat, tako i kompleksnosti i čitljivosti samog koda.

Poredjenje je izvršeno na praktičnom primjeru, u toku razvoja dvije aplikacije iste namjene od koji je jedna rađena u C# a druga u Clojure programskom jeziku. [1]

2. IMPERATIVNI I FUNKCIONALNI PRISTUP RAZVOJU APLIKACIJA

U imperativnom pristupu razvoju aplikacija, programer u kodu opisuje sve korake koje računar mora preduzeti kako bi postigao željeni cilj. Ovaj način naziva se i algoritamsko programiranje.

Suprotnost ovom pristupu je funkcionalni pristup razvoju aplikacija, koji uključuje rješavanje problema kreiranjem seta funkcija koje se mogu izvršavati. Pažljivo se definiše svaka funkcija, njeni ulazi i rezultat koji ona vraća, kako bi se kombinovanjem ovih funkcija na odgovarajući način došlo do rješenja postojećeg problema.

Za programski jezik se kaže da je funkcionalan ako računске operacije tretira kao primjenu matematičkih funkcija. Kao i u matematici, funkcionalni programi nemaju stanje koje može proizvoljno da se mijenja. U tom smislu, funkcionalni programski jezici se razlikuju od drugih popularnih jezika kao što su Java ili Ruby, u tome što su ovi jezici imperativne prirode, čime se promjenljivo stanje ističe

kao sredstvo za prikazivanje rezultata izračunavanja. Imperativni programski jezici obavljaju složene zadatke izvršavajući veliki broj instrukcija koje sekvencijalno mijenjaju stanje programa sve dok se ne dobije željeni rezultat. Funkcionalni programski jezici postižu isti rezultat kroz ugnježdene funkcijske kompozicije, prosleđujući rezultat jedne funkcije kao ulazni parametar druge.

Tabela 1: Glave razlike između funkcionalnog i imperativnog pristupa

Osobine	Imperativni pristup	Funkcionalni pristup
Glavni fokus programera	Kako izvršiti određene zadatke (algoritme) i kako pratiti promjene stanja.	Koja informacije je potrebna i koja transformacija je neophodna kako bi se ona dobila.
Promjene stanja	Veoma važno.	Ne postoji
Redosljed izvršavanja	Veoma važno.	Važno samo u rijetkim slučajevima.
Glavne kontrole toka programa	Petlje, uslovi i pozivi metoda.	Pozivanje funkcija, koje mogu uključivati rekurziju.
Primarne jedinice za manipulaciju podacima	Instance struktura podataka i klasa.	Funkcije kao primarne strukture i kolekcije podataka

3. CLOJURE

Clojure je dijalekt Lisp programskog jezika kreiran od strane Riča Hikija. Spada u funkcionalne programske jezike opšte namjene i fokusira se na programiranje sa nepromjenjivim vrijednostima i eksplicitnim vremenski definisanim konstrukcijama, što ima za cilj da omogući lakši razvoj programa, a naročito onih robusnijih i sa brojnim thread-ovima.

Clojure ima tri osnovne karakteristike i to su: [2]

- ideologija Lisp programskog jezika
- funkcionalna paradigma
- JVM

3.1 Lisp

LISt Processing (Lisp) je programski jezik, koji je prvobitno dizajnirano John McCarthy 1958. godine. Dizajniran je za oblast vještačke inteligencije, posebno zbog potrebe rada sa simboličkim izrazima (umjesto sa brojevima) i da se ovi izrazi predstavljaju kao liste simbola.

Lisp je baziran na funkcijama, koje se mogu prenositi kao vrijednosti, i mogu biti sastavljene jedna od druge. Kreiran je sa "code-as-data" ideologijom, što znači da se kodom može manipulirati jednako lako kao sa podacima. [3]

Clojure sa Lisp-om dijeli code-as-data filozofiju i moćan sistem makro-a.

3.2 Funkcionalno programiranje

To što je funkcionalan jezik Clojure-u omogućava neke prilično jedinstvene karakteristike. To su nepromjenljivost, funkcije višeg reda, lijenost i odlična paralelna semantika koja omogućava Clojure programima da koriste sva raspoloživa jezgra procesora na bezbjedan način.

Clojure ima bogat set nepromjenljivih struktura podataka, ali ukoliko je potrebno promjenljivo stanje Clojure nudi softverski transakcioni memorijski sistem i Agent sistem koji osigurava čist, ispravan i multithread dizajn.

Kod napisan u Clojure-u je mnogo kraći i manje sklon greškama u poređenju sa drugim jezicima. Clojure sintaksa je jako jednostavna, funkcije se uvijek pozivaju na isti način: prvo ime funkcije, a zatim bilo koji broj argumenata.

3.3 JVM

Java virtuelna mašina (skraćeno JVM) je virtuelna mašina koja može izvršavati Java bytecode. Clojure je hostovan na JVM-u, što znači da je u suštini Java program i da se izvršava kao Java bytecode.

Danas, kako bi programski jezik bio praktičan, gotovo je neophodno da bude u interoperabilnosti sa Javom. Prilikom dizajniranja Clojure-a saradnja sa drugim Java bibliotekama

bila je jedan od osnovnih ciljeva. Rich Hickey je taj cilj ostvario izuzetno dobro i omogućio niz prednosti.

Kako bi programski jezik bio produktivan potrebno je da bude u paketu sa velikim skupom biblioteka, koje definišu jezik koliko i njegova sintaksa. Očigledna prednost JVM-a je što Clojure programeri imaju trenutni pristup hiljadama postojećih Java biblioteka, na veoma jednostavan način.

Još jedna prednost, koja dolazi od izvršavanja na JVM-u, je to što se Clojure može ugraditi u Java programe. To često vodi ka dizajnu u kom su dijelovi koda napisani u Javi, a dijelovi u Clojure-u. Clojure može da se koristi kao „lijepak“ koji drži cijeli sistem na okupu, što može biti posebno korisno kada se radi o zastarelim aplikacijama.

Konačno, prednost toga što je Clojure JVM jezik je sama JVM. Sun Microsystems (sada Oracle) i open-source zajednica su uložili veliko vrijeme i kadrove u poboljšanje JVM-a, pa je danas to jedna od najefikasnijih virtuelnih mašina. Uprkos tome što je dinamičan, Clojure kod se kompajlira u Java bytecode i izvršava se jednako, ili skoro jednako brzo kao sam Java kod. [1,2]

3.4 Clojure REPL

Mnogi programski jezici imaju REPL, a njegovu funkciju na jednostavan način opisuje skraćenica:

1. Read - kod se čita kao tekst, iz nekog input-a
2. Eval - kod se evaluira i vraća neku vrijednost
3. Print - povratna vrijednost se prikazuje na nekom izlazu
4. Loop - povratak na korak čitanja

Clojure REPL se razlikuje od drugih po tome što nije interpreter koji koristi ograničen dio koda, već se sav kod kompajlira u JVM bytecode, kao dio njegove evaluacije, sa istim rezultatima kao kada se kod učitava iz Clojure source fajla.

Nastao je iz potrebe za kreiranjem modernog Lisp dijalekta za funkcionalno programiranje, koji je sposoban da radi u simbiozi sa poznatom Java platformom.

Clojure je naročito pogodan za izradu aplikacija koje zahtijevaju konkurentnost (izvršavanje više stvari paralelno). Pristup konkurentnosti Clojure programskog jezika karakteriše koncept identiteta, koji zapravo predstavlja niz nepromjenljivih stanja u toku vremena. Kako je jedno stanje nepromjenljivo vrijednost, bilo koji broj korisnika može raditi nad određenom promjenljivom u istom trenutku, a konkurentnost postaje jednostavno upravljanje prebacivanja sa jednog stanja na drugo te iste promjenljive. Zarad ove funkcionalnosti, Clojure pruža nekoliko promjenljivih referentnih tipova, pri čemu svaki posjeduje dobro definisanu semantiku za promjenu i prelazak iz jednog u drugo stanje.

Kao i kod većine programskih jezika Lisp dijalekta, sintaksa Clojure-a je izgrađena na osnovu symbolic expression (S-expression) notacije, što znači da se source code aplikacije pretvara u odgovarajuće tipove struktura podataka prije kompajliranja. [1,2]

4. C-SHARP (C#)

C# projekat je počeo u decembru 1998. godine sa ciljem da kreira jednostavan, moderan, objektno-orijentisan i type-safe programski jezik, koji bi bio nazvan .NET platform. Od tada, C# projekat prošao je dug put.

Ovaj jezik ima svoje korijene u C porodici programskih jezika i biće odmah poznat C, C++ i Java programerima. C# je standardizovan po ECMA International standardima (ECMA-344) kao i po ISO/IEC standardima (ISO/IEC 23270). Mikrosoftov C# kompilator za .Net Framework je usaglašena implementacija oba standarda.

C# je objektno-orijentisan jezik koji uključuje podršku za komponentski orijentisano programiranje. Savremeni dizajn softvera se sve više oslanja na softverske komponente u formi samo-sadržućih i samo-opisnih funkcionalnih paketa. C# pruža jezičke konstrukte koji direktno podržavaju ove koncepte, čineći da C# bude veoma jednostavan jezik kojim se kreiraju i koriste softverske komponente.

Nekoliko C# odlika pomažu u konstrukciji robusnih i trajnih aplikacija:

- *Garbage collection* automatski oslobađa memoriju okupiranu od strane objekata koji se ne koriste;
- *Exception handling* pruža strukturiran i proširen pristup detekciji grešaka i oporavku;
- *Type safe* dizajn jezika čini nemogućim postojanje neinicijalizovanih varijabli, indeksiranje nizova van njihovih granica itd.

C# ima jedinstven tip sistema. Svi C# tipovi podataka, uključujući primitivne tipove kao int i double, nasljeđuju osobine od single root object type-a. Prema tome, svi tipovi podataka dijele set zajedničkih operacija i vrijednosti bilo koje vrste mogu biti zapamćene, premještene i upravljane na sličan način. Štaviše, C# podržava korisnički definisane referentne tipove i vrijednosne tipove, dozvoljavajući dinamičnu alokaciju objekata kao i in-line storage jednostavnijih struktura.

Da bi se osiguralo da C# programi i biblioteke mogu da napreduju tokom vremena na kompatibilan način, veliki akcenat je stavljen na versioning dizajna C# programskog jezika. Mnogi programski jezici daju malo pažnje ovom problemu, i kao rezultat toga programi pisani tim jezicima se jako često pokazuju nefunkcionalnim kada se predstavi novija verzija biblioteka koje referenciraju. Aspekti C# dizajna koji su bili pod direktnim uticajem implementacije versioning-a uključuju odvojene virtualne i override modifier-e, pravila za rezoluciju override metoda itd.[4,5]

5. C# NASUPROT CLOJURE-A

Kako se imperativni pristup veže za razvoj aplikacija korišćenjem nekih od objektno-orijentisanih programskih jezika, a u ovom radu je već bilo riječi o C# programskom jeziku, kao i o Clojure-u za koji možemo vezati funkcionalni pristup, u ovom poglavlju ćemo se osvrnuti na karakteristike osnovnih elemenata oba programska jezika, kao i njihovih prednosti i mana prikazanih na konkretnim primjerima.

Karakteristika koja je uočljiva već na prvi pogled jeste vidno kraći source code aplikacije pisane Clojure programskim jezikom, pa Clojure po svojoj prirodi podstiče korisnike da pišu kod koji je jednostavan za čitanje i otklanjanje grešaka. Eksplicitni stil omogućava veoma lako uočavanje funkcionalnosti programa, čak i bez potrebe da se uvijek razumije kako. Ovo svakako predstavlja veliku prednost u odnosu na objektno-orijentisane programe koji zbog svoje preširoke deklarativnosti i uslovljenosti postaju veoma teški za čitanje, razumijevanje i otklanjanje grešaka.

Jednostavan primjer bi bio kreiranje liste iz koje je potrebno uzeti samo jedinstvene vrijednosti i štampati ih u konzoli:

```
List<string> values = new List<string>();
values.Add("foo");
values.Add("bar");
values.Add("baz");
values.Add("foo");

IEnumerable<String> distinct = values.Distinct();

for (int i = 0; i < distinct.Count(); i++)
{
    Console.WriteLine(distinct.ElementAt(i));
}

/*Output:
foo
bar
baz
*/
```

Slika 1. Primjer kreiranja liste i prikaza njenih jedinstvenih vrijednosti u C# programskom jeziku

```
user> (distinct ["foo" "bar" "baz" "foo"])
("foo" "bar" "baz")
```

Slika 2. Primjer kreiranja liste i uzimanja njenih jedinstvenih vrijednosti.

Programerima koji su naviknuti na imperativni pristup i OO programiranje, i koji su usvojili sintaksu pisanja čitljivog koda u JAVA-i i C#-u, jedan od većih problema predstavljaju zagrade, koje se prilikom zatvaranja pišu u jednom redu.

```
private void fillDatumCombo()
{
    datumGubiciComboBox.Items.Clear();
    datum2ComboBox.Items.Clear();
    datumi = datumiTableAdapter1.datumiGubici();
    foreach (DataRow dr in datumi.Rows)
    {
        datumGubiciComboBox.Items.Add(dr["DATUM_IZVJESTAJA"]);
        datum2ComboBox.Items.Add(dr["DATUM_IZVJESTAJA"]);
    }
}
```

Slika 3. Kreiranje combobox-a za datum u C# programskom jeziku

```
(defn combo-box-options [date-value]
  [option {:value date-value} date-value])

(defn all-options-losses []
  (map #(combo-box-options (format "%1$tF" (first %)))
    (db/all-trans-losses)))

(defn combo-box-losses []
  (html [:select {:id "cmb"}
    (all-options-losses)]))
```

Slika 4. Kreiranje combobox-a za datum u Clojure-u uz pomoć Hiccup sintakse

Veoma moćna opcija Clojure-a jeste kombinovanje funkcija kako bi se dobila željena cjelina:

```
(defn data-states [datum ed]
  (if (= "SVE" ed)
    (db/find-billing-state datum)
    (db/find-biling-state-by-ed datum ed)))

(defn row-states [kol1 kol2 kol3 kol4 kol5 kol6 kol7]
  [:tr [:td kol1] [:td kol2] [:td kol3] [:td kol4] [:td kol5] [:td kol6] [:td kol7]])

(defn all-rows-states [datum ed]
  (map #(row-states (% 0) (% 1) (% 2) (% 3) (% 4) (% 5) (% 6))
    (sort-by first (data-states datum ed))))
```

Slika 5. Kreiranje svih redova html table sa podacima u ćelijama za prikaz izvještaja.

Iz svega navedenog dolazimo do zaključka da je korišćenje Clojure-a u svrhe rada sa podacima daleko jednostavnije i lakše, jer ne postoje ograničenja u vidu tipa podataka sa kojima se radi, pa na taj način je vrlo jednostavno smještati ih u mape, kao što je prikazano prethodnim primjerom, ili npr. u vektore, jer su i same mape i vektori funkcije.

6. ZAKLJUČAK

Imperativni pristup razvoju softverskih aplikacija je danas najčešće korišćen, pa samim tim postoji i velika podrška, kao i veoma veliki broj primjera, diskusija na forumima i biblioteka razvijenih za neki od OO programskih jezika koje ovaj pristup favorizuje, pa je na taj način olakšano samo učenje korištenja ovog stila i generalno navikavanje na OO programske jezike.

Nasuprot tome, prihvaćenost funkcionalnih jezika od strane programera u odnosu na prihvaćenost C# i JAVA programskih jezika nije na zavidnom nivou, iako Clojure zajednica iz dana u dan sve više raste. Ovo otežava rad, naročito početnicima na polju funkcionalnog pristupa

programiranju. Ako je još taj početnik neko sa višegodišnjim iskustvom sa OO programskim jezicima, ovakav nivo podrške i generalni nedostatak dokumentacije može uzrokovati i odustajanjem od učenja funkcionalnog pristupa.

Takođe, ostaje činjenica da je funkcionalno programiranje moćnije od OO programiranja u smislu daleko jednostavnijeg rješavanja skoro svih problema koje je moguće programski riješiti. Tokom vremena, zajednica programera koji koriste funkcionalni pristup bi trebala da raste, a samim tim i podrška, primjeri, i biblioteke koje je moguće koristiti u cilju postizanja željenog rezultata .

Trenutni pobjednik na polju razvoja softverskih aplikacija je imperativni pristup, ali samo zbog veće rasprostranjenosti i opšte prihvaćenosti. Tokom narednih godina možemo očekivati (naravno, ako trend razvoja funkcionalnog pristupa nastavi uzlaznom putanjom proporcionalno kao što je do trenutka pisanja ovog rada) postepeni pad popularnosti imperativnog pristupa, a samim tim i rast popularnosti funkcionalnog.

LITERATURA

- [1] Luke Vander Hart, Stuart Sierra, *Practical Clojure*, Apress, 2010.
- [2] Fogus, Michael; Houser, Chris, *The Joy of Clojure*, Manning. 2010.
- [3] Steve Yegge , „*Lisp is Not an Acceptable Lisp*“, 2003
- [4] Andrew Stelman, Jennifer Greene, *Head First C#, 2nd Edition*, O'Reilly, 2010.
- [5] Vidya Vrat Agarwal, *Beginning C# 5.0 Databases 2nd Edition*, Apress, 2012.

INTEGRACIJA ISTORIJSKIH PODATAKA U AHP ALGORITAM INTEGRATION OF HISTORICAL DATA IN AHP ALGORITHM

Jelena Ljucović, Ivana Ognjanović, *Fakultet Informacionih Tehnologija, Univerzitet Mediteran, Podgorica*
Ramo Šendelj, *Humanističke studije, Univerzitet Donja Gorica, Podgorica*

Sadržaj: *Nekada donošenje odluka može biti složen proces koji obuhvata širok spektar kriterijuma, koji se srijeću kako u svakodnevnom životu, tako i u raznim oblastima i domenima nauke i istraživanja, koa što su inteligentni sistemi, računarske mreže, baze podataka i. U cilju donošenja što optimalnije odluke, u ovom radu se predlažu dva inovativna pristupa integraciji istorijskih podataka kao proširenja opšte prihvaćenog i široko primjenljivog AHP (engl. Analytical Hierarchical Process) algoritma.*

Abstract: *Sometimes making decision is very difficult process which includes considerations of different criteria, that are encountered in everyday life as well as in various fields and domains of science and research, such as intelligent systems, computer networks, databases etc. In this paper, in order to make optimal decisions, we propose two innovative approaches of integrations of historical data implemented as extension of wide used and generally accepted AHP (engl. Analytical Hierarchical Process) algorithm.*

1. UVOD

Donošenje odluka nekada može biti veoma složen process koji zavisi od mnogo faktora, a u procesu donošenja odluke može biti uključeno i više učesnika. Ovaj proces se srijeće kako u svakodnevnom životu, tako i u mnogim naučnim disciplinama, kao što su baze podataka, inteligentni sistemi, mašinsko učenje, elektronsko poslovanje, računarske mreže, itd [5, 1]. Danas postoji širok izbor alata i softvera koji podržavaju proces odlučivanja, te cijele kompanije zasnovane na ideji poslovne inteligencije. Neki od najpoznatijih alata su zasnovani na kvalitativnom definisanju zahtjeva koji mogu sadržati i uslovljenosti (kao što su CP-nets [5] i TCP-nets [6]), neki od njih se odnose na kvantitativno definisane zahtjeve (npr. AHP algoritam [3], TOPSIS metod [2]) koji mogu sadržati i specifične vrste zahtjeva, kao npr. zahtjevi sa dominantnim prioritetom (npr CS-AHP [1] kao proširenje AHP algoritma), dok se u nekim oblastima posmatraju i vremenski promjenljivi zahtjevi [7]. Kako god, opšti algoritam za analizu zahtjeva ne postoji, već se odabir odgovarajućeg pristupa vrši na osnovu karakteristika domena primjene i samog problema koji se analizira [1].

Glavna srž problema koji se srijeću u oblasti analize zahtjeva se odnosi na samo predstavljanje ljudskog razmišljanja na način koji je razumljiv računaru [8]. Uzimajući u obzir da se pri donošenju odluka, čak i pri samom definisanju zahtjeva, čovjek često svjesno ili nesvjesno oslanja na iskustvene podatke, u ovom radu se analizira mogućnost integracije istorijskih podataka u sami proces odlučivanja. S obzirom da se istorijski podaci mogu analizirati primjenom osnovnih metoda deskriptivne statistike [9], pomoću kojeg se dobijaju kvantitativne vrijednosti, predlaže se njihova integracija u AHP metod, kao metod koji je u najširoj upotrebi a istovremeno se koristi za analizu kvantitativno (bez)uslovno definisanih zahtjeva. Stoga se predlažu dva inovativna pristupa integraciji istorijskih

podatka, zasnovanih na proširenju po širini (AHP – HB (*Analytical Hierarchical Process History Breadth*)) i proširenju po dubini AHP – HD (*Analytical Hierarchical Process History Depth*)

Ostatak rada je organizovan na sljedeći način: Poglavlje II uvodi osnovne koncepte AHP algoritma kao i ilustrativni primjer koji će se koristiti tokom cijelog rada, Poglavlje III opisuje mogućnosti integracije istorijskih podataka. Poglavlje IV predstavlja modifikacije AHP algoritma (AHP-BD i AHP-HD) u cilju integracije istorijskih podataka na dva konceptualno različita načina. Poglavlje V opisuje implementaciju predstavljenog rješenja, dok Poglavlje VI zaključuje rad diskusijom i pravcima za dalja istraživanja.

2. AHP ALGORITAM

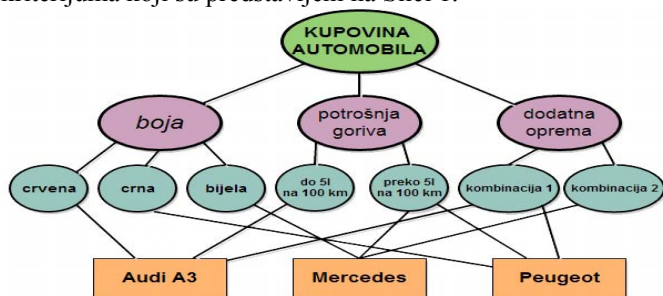
AHP (*Analytical Hierarchical Porcess*) [3] je algoritam koji je predložio Saaty, 1970.god To je fleksibilan metod za višekriterijumsko odlučivanje koji omogućava razmatranje materijalnih i nematerijalnih opcija u sami process odlučivanja, dozvoljavajući i nadgledanje stepena konzistentnosti u rasuđivanju [1], kao i mogućnosti uključenja više učesnika u sami process odlučivanja [10]. Kao opšte prihvaćen metod, AHP je naširoko korišćen u mnogim važnim oblastima donošenja odluka kao što su predviđanje, kontrola upravljanja kvalitetom, upravljanje poslovnim procesima, strateško planiranje, analizu investicija, analizu dobiti i slično [4, 11, 12].

AHP algoritam se sastoji iz četiri faze:

- izgradnja hijerarhijske strukture problema odlučivanja
- prikupljanje podataka,
- ocjenjivanje relativnih težina,
- određivanje rešenja problema.

Izgradnja hijerarhijske strukture problema se sastoji od dekomponovanja određenog složenog problema odlučivanja u seriju hijerarhija, gdje svaki nivo predstavlja

manji broj atributa (faktora) važnih za donošenje odluke. Poslednji nivo hijerarhije su alternative tj. moguće opcije datog problema koje su karakterisane u skladu sa navedenim faktorima odlučivanja. U cilju ilustracije primjene AHP algoritma posmatrajmo problem kupovine automobile po kriterijuma koji su predstavljeni na Slici 1.



b)

	Boja	Potrošnja goriva	Dodatna oprema
Boja	1	1/9	1/3
Potrošnja goriva	9	1	7
Dodatna oprema	3	1/7	1

Čitanje relacije: npr. (2-1) Potrošnja goriva ekstremno značajnija od Boje

c)

Alternativa	Težina alternative
Audi A3	0.63
Peugeot	0.41
Mercedes	0.28

Slika 1. a. Dekompozicija kod AHP algoritma, b. Matrica poređenja za definisani problem za prvi nivo, c. Kvantitativno poređenja raspoloživih opcija

Prikupljanjem podataka i njihovim mjerenjem, započinje druga faza metode AHP. Donosilac odluke dodjeljuje relativne ocjene parovima atributa jednog hijerarhijskog nivoa, i to za sve nivoe cjelokupne hijerarhije (tradicionalno se koriste vrijednosti 1, 3, 5, 7, 9 koje odgovaraju gradaciji od jednakosti do ekstremno većeg odnosa; dok se recipročne vrijednosti koriste za predstavljanje gradacije u suprotnom smjeru [4]). Po završetku ove faze, dobija se odgovarajuća matrica upoređivanja po parovima koji odgovaraju svakom nivou hijerarhije. Procjena relativnih težina je treća faza primjene metode AHP (Slika 1b). Matrica poređenja se po parovima "prevodi" u problem određivanja sopstvenih vrijednosti, radi dobijanja normalizovanih i jedinstvenih sopstvenih vektora, kao i težina $t = (t_1, t_2, \dots, t_n)$ za sve attribute na svakom nivou hijerarhije K_1, K_2, \dots, K_n .

Određivanje rješenja problema je poslednja faza metode AHP, a ona podrazumijeva nalaženje kompozitnog normalizovanog vektora. Pošto se odredi vektor redosljeda kriterijuma u modelu, u narednom krugu potrebno je, u okviru svakog posmatranog kriterijuma, odrediti redosljed važnosti alternative u modelu. Na kraju, sveukupna sinteza problema se izvodi na sljedeći način: učešće svake alternative

množi se sa težinom posmatranog kriterijuma, a potom se te vrijednosti sabere za svaku alternativu posebno. Dobijeni podatak predstavlja težinu posmatrane alternative u modelu. Na isti način se težina određuje i za sve ostale alternative, posle čega se može odrediti konačni poredak alternativa u modelu (Slika 1c).

3. ISTORIJSKI PODACI

Podaci o iskustvima, stavovima i zahtjevima korisnika se svakodnevno prikupljaju na najrazličitije načine: popunjavanjem online anketa, ispunjavanjem različitih formulara, upitnika, a mogu biti sadržani i u bazama informacionog sistema zajedno sa svim ostalim podacima o upotrebi sistema. Po našem saznanju, do sada ne postoji ni jedna metoda koja uključuje istorijske podatke u proces odlučivanja, a jasno je da oni mogu igrati značajnu ulogu kako u samom definisanju korisničkih zahtjeva (posebno ukoliko je korisnik svjestan da postoje podaci o ostalim učesnicima koji su koristili isti sistem- npr problem kupovine auta i iskustva prethodnih korisnika), ili je korisnik sam već ranije koristio sistem i već ima neka iskustva (npr kupovina proizvoda određenog brenda kada je korisnik iskusio njihove prednosti/nedostatke). Zavisno od scenarija u kojem se donosi odluka kao i od samih klijenata i njihovih preferenci biraju se istorijski podaci koji su potrebni za donošenje odluke. Možemo ih podijeliti u dvije kategorije:

- podaci koji su donešeni na osnovu subjektivnog stava ranijih korisnika (npr. ocjena o udobnosti nekog automobila, pouzdanost brenda)
- podaci koji su formirani na osnovu objektivnih iskustava (npr. najprodavanija boja automobila).

Nad ovim podacima se u zavisnosti od potrebe primjenjuju različite statističke metode [9] poput određivanja intervala povjerenja, srednjih ocjena, očekivanja, kako bi se došlo do zaključaka i informacija, a potom se vrši integracija u sisteme za odlučivanje i daje ocjena od strane korisnika za ličnu važnost prikupljenih informacija. Uzimajući u obzir različite preference korisnika sa stanovišta odnosa prema istorijskim podacima i njihovog uključenja u process odlučivanja, da li se želi uzimati kao dodatni kriterijum u odlučivanju ili se postojeći kriterijumi žele proširiti sa konceptom poređenja sa istorijskim podacima, u nastavku se predlažu dva načina za njihovu integraciju u AHP algoritam: integracija po dubini (rješenje nazvano AHP – HD (*Analytic Hierarchy Process History Depth*)) i integracija po širini (rješenje nazvano AHP – BD (*Analytic Hierarchy Process History Breadth*))

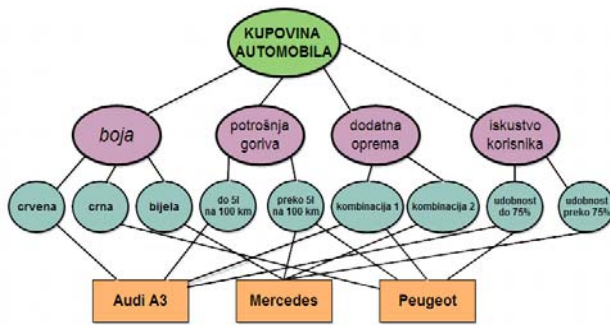
4. AHP HISTORY BREADTH

AHP – HB (*Analytic Hierarchy Process History Breadth*) je modifikacija klasičnog AHP algoritma koja integriše istorijske podatke već na prvom nivou hijerarhije uzimajući ih kao poseban kriterijum - "Iskustva korisnika" pri odlučivanju. Drugim riječima AHP – HB pomaže zainteresovanim stranama da pronađu najpogodniji skup opcija za njihov ciljni problem uključujući ranije prikupljena iskustva iz oblasti interesovanja i uzimajući ih u obzir kao kriterijum na prvom nivou razlaganja problema. Cjelokupni

proces modifikacije AHP algoritma donosi izmjene samo nekih od faza u standardnom AHP algoritmu, dok sam proces određivanja kvantitativnih vrijednosti ostaje nepromijenjen.

Faza hijerarhijskog strukturiranja podataka ima dodatnu kompleksnost koja se ogleda u što konciznijem odabiru kategorija pri razlaganju problema. Na prvom nivou postoji novi kriterijum – *Iskustva korisnika*. Proširenje se ogleda u dobrom projektovanju narednih nivoa hijerarhijskog razlaganja ovog kriterijuma i odabiru koji podaci (iskustva) među istorijskim podacima su relevantni za donošenje odluke, kao i povezivanje tih podataka sa alternativama.

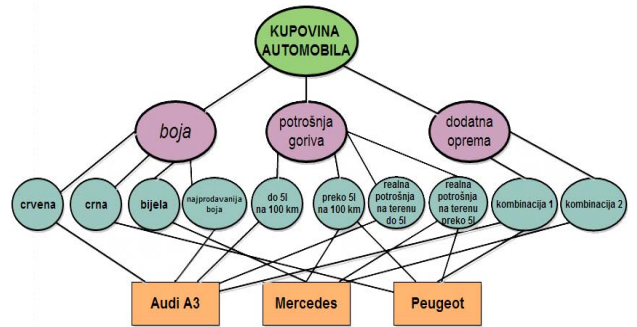
U našem primjeru odabira automobila dodatni kriterijum se može odnositi na pitanje: “Koliko je za korisnika važno mišljenje o udobnosti procijenjeno od strane postojećih vlasnika istog automobila?” kao što je predstavljeno na Slici 2. Nakon što korisnik sam odgovori na postavljeno pitanje (iznoseći svoj lični stav), raspoložive opcije (tj vrste automobila koji su na raspolaganju) treba okarakterisati po navedenom kriterijumu. Odgovor se u ovom slučaju može pronaći u rezultatima *on-line* anketa korisnika, ili sprovođenjem ankete kod osoba od poverjenja. Ostali postupci računanja i rangiranja ostaju nepromijenjeni u odnosu na standardni AHP algoritam.



Slika 2. Dekompozicija kod AHP – HB algoritma

5. AHP – HISTORY DEPTH

AHP – HD (*Analytic Hierarchy Process History Depth*) predstavlja drugi pristup integracije istorijskih podataka u AHP algoritam, pri čemu se ovi podaci integrišu tek u nižim slojevima hijerarhije, što omogućava veću fleksibilnost pri odluci da li i kada će istorijski podaci biti integrisani, kao i u sklopu kojih nadkriterijuma. Ideja ovog tipa integracije se više bazira na konkretnim činjeničnim istorijskim podacima (npr. prilikom kupovine automobile u kriterijumu boja se može dodati podkriterijum najprodavanija boja, kao što je prikazano na Slici 3), za razliku od AHP - HB koji akcenat stavlja na iskustvima korisnika, koja su često subjektivna. Kako AHP algoritam nije ograničen sa samo dva nivoa hijerarhije, tako se ovi istorijski podkriterijumi mogu naći tek u nižim slojevima hijerarhije. Razlika od standardnog AHP algoritma se sastoji kao i u prethodnom algoritmu u prvom koraku metode - dekompozicije problema i formiranju strukture. Sva izračunavanja u matricama i formiranja sopstvenih vektora, kao i proces rangiranja ostaju isti, na ma kojem nivou hijerarhije se ovi podaci nalazili.



Slika 3. Dekompozicija kod AHP – HD algoritma

Formalno, korektnost navedenih proširenja AHP algoritma je zasnovana na očuvanju osnovnih karakteristika AHP algoritma [3], kao i zadovoljenju uslova za agregiranjem relativnih prioriteta kada se uslovi mogu definisati (bez)uslovno, definisanim u prethodnim istraživanjima AHP familije metoda [1].

6. IMPLEMENTACIJA

Jedno od najvećih doprinosa predloženih proširenja AHP algoritma je njihova praktična primjena u različitim oblastima i domenima. U cilju praktične primjene predloženih AHP-HB i AHP-HD algoritama, razvijeno je okruženje pod nazivom *AHP-HistoryData* kao proširenje prethodno razvijenog alata za rad u oblasti linija softverskih proizvoda (engl. *Software Product Lines*) baziranog na CS-AHP algoritmu (tj proširenju AHP algoritma koje omogućava definisanje uslovnih zahtjeva i zahtjeva o dominantnim prioritetima) [1].

AHP-HistoryData razvijen kao proširenje (engl. *plug-in*) u Eclipse razvojnom okruženju, dok se za definisanje korisničkih zahtjeva i strukture problema koristi *Visual-feature model plug in (vis-fmp)*. Ovim alatom je omogućeno interaktivno definisanje modela karakteristika i alternativa. *Vis-fmp* omogućava:

- Formiranje stabla strukture i automatizaciju procesa konfigurisanja modela na osnovu nefunkcionalnih karakteristika i osnovnih vrsta zahtjeva.
- Podršku tehnikama vizuelizacije i interakcije koje obezbjeđuju jasan pogled na generisanu konfiguraciju

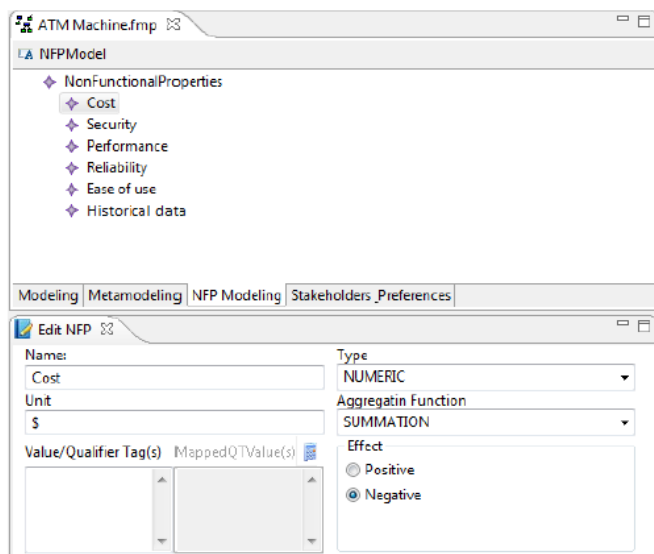
Zadatak implementacije se stoga ogleda u proširenju *vis-fmp* i omogućavanju:

- Definisane relevantnih istorijskih podataka i njihovo uključivanje u strukturu zadatka
- Rangiranje alternativa na osnovu zadate strukture i odabir odgovarajuće konfiguracije

Struktura dekompozicije problema se prikazuje 2D stablom za koje se pokazalo kao najpogodnije za predstavljanje [13]. U kreiranom stablu cvorovi predstavljaju karakteristike, a grane među njima kompozicione relacije među karakteristikama.

U razvijenom alatu, kao što je prikazano na Slici 4 korisnik može da definiše nefunkcionalne karakteristike od interesa i doda faktor istorijskih podataka – iskustva korisnika – u slučaju AHP – HB pristupa. Za svaku od njih moguće je definisati podkarakteristike, od kojih neke mogu

biti činjenični statistički obrađeni istorijski podaci – u slučaju AHP – HD pristupa. Svi istorijski podaci su prethodno obrađeni u statističkim alatima koristeći statističke metode za dobijanje informacija i u softveru se koriste samo krajnji rezultati obrade.



Slika 4. AHP-HistoryData alat

Razvijeno rješenje omogućava da više korisnika nezavisno definiše svoje zahtjeve, pri čemu se za određivanje najbolje alternative svi definisani zahtjevi razmatraju nezavisno.

7. ZAKLJUČAK

U ovom radu je analizirana mogućnost integracije istorijskih podataka u proces odlučivanja koji se zasniva na dobro poznati i opšte prihvaćeni AHP metod. Predložena dva pristupa AHP-HD i AHP-HB su zasnovana na proširenju hijerarhijske strukture kriterijuma odlučivanja, dodavanjem novog kriterijuma na najvišem nivou hijerarhije (AHP-HB) ili na nižim nivoima (AHP-HD) u zavisnosti od ličnih preferenci korisnika. Predloženim načini za proširenjem AHP algoritma ispunjavaju zahtjeve za efektivnom tehnikom rangiranja i odlučivanja koji su zasnovani na izazovima i smjericama definisanim u [14, 15], kao i mogućnosti uključivanja više učesnika u procesu odlučivanja [3, 1].

Naš budući rad će uključiti formalnu specifikaciju ova dva algoritma, kao i procjenu kolika je optimalna integracija potrebna za što preciznije odlučivanje.

LITERATURA

[1] I.Ognjanović, D.Gašević, E.Bagheri-"A Stratified Framework for Handling Conditional Preferences: an Extension of the AHP", *Expert Systems with Applications*, Vol 40, No. 4, March 2013, pp.1094-1115

[2] Turski, Z. (2008). Multi-attribute contractors ranking method by applying ordering of feasible alternatives of solutions in terms of preferability technique.

Technological and economic development, Baltic Journal on Sustainability, 14(2), 224-239.

[3] Saaty, T. L. (1980). *The Analytic Hierarchy Process*. McGraw-Hill, New York.

[4] Ognjanović, I., Gašević, D., Bagheri, E., Asadi, M. (2011), "Conditional preferences in software stakeholders' judgments," *Proc. of the 26th Annual ACM Symposium on Applied Computing (SAC 2011)*, Taiwan

[5] Brafman, R. I., & Domshlak, C. (2002). Introducing variable importance tradeoffs into CP-nets. In *The Proceedings of the Eighteenth Conference on Uncertainty in AI*, (pp. 69-76). Canada.

[6] Boutilier, C., Brafman, R. I., Domshlak, C., Hoos, H. H., & Poole, D. (2004). CP-nets: a tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of AI Research*, 21(1), 135-191.

[7] Wilson, N. (2011). Computational techniques for a simple theory of conditional preferences. *Artificial Intelligence*, 175(7-8), 1053-1091.

[8] Yu, Z., Yu, Z., Zhou, X., & Nakamu, Y. (2010). Toward an Understanding of User-Defined Conditional Preferences. In *Proc. of the 8th IEEE Conf on Dependable, Autonomic and Secure Computing*, (pp. 203-208). Washington, USA.

[9] Blaikie, N. (2003). *Analyzing Quantitative Data*. London: Sage.

[10] Roper-Low, G. C. (1990). The Analytic Hierarchy Process and its application to an information technology. *The Journal of Operational Research*, 41(1), 49-59.

[11] Büyüközkan, G., Çifçi, G., & Güleriyüz, S. (2011). Strategic analysis of healthcare service quality using fuzzy AHP methodology. *Expert Systems with Applications*, 38(8), 9407-9424.

[12] Chen, M. K., & Wang, S. (2010). The critical factors of success for information service industry in developing international market: Using AHP approach. *Expert Systems with Applications*, 37(1), 694-704.

[13] Bagheri, E., Asadi, M., Gasevic, D., & Soltani, S. (2010). Stratified AHP: prioritization and selection of software features. In *Proceedings of the 14th International conference on Software product lines*, (pp. 300-315).

[14] McManus, J. (2004). *Managing stakeholders in software development projects*. Butterworth-Heinemann, Elsevier.

[15] Sommerville, I., & Sawyer, P. (1997). Viewpoints: principles, problems and a practical approach to preferences engineering. *Annals of Software Engineering*(3), 101-130.

KOMPARATIVNA ANALIZA NOSQL I SQL BAZA PODATAKA, NA PRIMJERU DATOMIC-A I MSSQL-A

COMPARATIVE ANALYSIS OF NOSQL AND SQL DATABASE, ON THE EXAMPLE OF DATOMIC AND MSSQL

Tijana Vujičić, Petar Radunović, Ivan Knežević, *Fakultet za informacione tehnologije, Univerzitet Mediteran, Podgorica*

Sadržaj: *SQL baze podataka su opšte prihvaćene i najčešće korišćene, ali imaju niz problema koji utiču na sve češće razmišljane programera o promjeni pristupa i prelasku na ne-relacione baze podataka. U ovom radu je predstavljena komparativna analiza NoSQL i SQL baza podataka, na primjeru Datomic-a i MSSQL-a. Prilikom analize u obzir su uzete karakteristike, kompleksnost programskog koda i performanse.*

Abstract: *SQL databases are widely accepted and most commonly used, but have a number of issues that affect developers' opinion on changing their approach and transition to non-relational databases. This paper presents a comparative analysis of NoSQL and SQL databases, on example of Datomic and MSSQL. Characteristics of code complexity and performance were taken into account during the creation of this paper.*

Keywords: *SQL, NoSQL, Datomic, MSSQL, Clojure, C#, database, functional programming*

1. UVOD

U poslednjih 40-ak godina, relacione baze podataka su vladale svijetom podataka. Relacioni model se prvi put pojavio u ranim 70-im. Rane verzije SQL jezika razvijale su se i u ranim 70-im, dok se moderan SQL pojavljuje par godina kasnije, i postaje popularan od sredine 80-ih.

U nekoliko proteklih godina sve češće su rasprave na temu „SQL vs NoSQL“. NoSQL baze podataka su počele da napreduju i da se dokazuju na tržištu, tako da se borba između ova dva pristupa nastavlja.

Tehnički govoreći, SQL je jezik koji se koristi za pristup relacionim bazama podataka. Kada se govori o temi „SQL vs NoSQL“ u suštini se radi o poređenju relacionih baza podataka (kao što su Oracle, MySQL i SQL Server) i ne-relacionih baza podataka (kao što su MongoDB, CouchDB, BigTable, Datomic itd.). [1]

Poređenje ovih tehnologija najbolje je napraviti na nivou kodiranja, sa tačke gledišta programera, na praktičnim primjerima aplikacija u različitim programskim jezicima sa različitim bazama podataka. U ovom radu će biti predstavljeno poređenje dvije aplikacije iste namjene, od kojih je jedna razvijena u funkcionalnom programskom jeziku Clojure, sa Datomic NoSQL bazom podataka, a druga u C# programskom jeziku, sa MSSQL (Microsoft Server SQL) bazom podataka.

2. CLOJURE PROGRAMSKI JEZIK

Funkcionalno programiranje je programska paradigma koja tretira program kao izračunavanje matematičkih funkcija, i izbjegava stanja i promenljive podatke. Akcentat je na primjeni funkcija, u suprotnosti sa stilom imperativnog programiranja, kod koga je naglasak na promjeni stanja.

Clojure je novi funkcionalni programski jezik koji je, kao dijalekat Lisp jezika, kreirao Rich Hickey, a pojavio se 2007. godine. To je dinamički programski jezik, dizajniran da bude

jezik opšte namjene, koji kombinuje pristupačnost i interaktivni razvoj skript jezika sa efikasnom i robusnom infrastrukturom za multithread programiranje. [2][3]

Programiranje u funkcionalnim jezicima se razlikuje od programiranja u imperativnim jezicima, ali i od programiranja u proceduralnim jezicima, uprkos površnoj sličnosti procedura i funkcija.

To što je funkcionalan jezik Clojure-u omogućava neke prilično jedinstvene karakteristike. To su nepromjenljivost, funkcije višeg reda, lijenost i odlična paralelna semantika koja omogućava Clojure programima da koriste sva raspoloživa jezgra procesora na bezbjedan način.

Kod napisan u Clojure-u je mnogo kraći i manje sklon greškama u poređenju sa drugim jezicima.

3. DATOMIC BAZA PODATKA

Datomic je NoSQL baza podataka dizajnirana da omogućiti skalabilne, fleksibilne i inteligentne aplikacije koje rade na cloud arhitekturi. To je baza činjenica zasnovanih na vremenu. [4]

Datomic izbjegava kompromise i gubitke koji su karakteristika mnogih NoSQL rješenja, a nudi fleksibilnost i prednost nad tradicionalnim modelom zato što omogućava:

- hijerarhiju
- atribut sa više vrijednosti
- minimalnu šemu
- pouzdan rad na cloud-u
- ugrađivanje vremena u podatke

3.1 NoSQL

NoSQL je pokret koji uvodi potpuno novi koncept baza podataka. NoSQL termin treba da se koristi kao Not-Only-SQL (ne samo SQL), a ne kao No-to-SQL ili Never-SQL (ne SQL ili nikad SQL). Prvobitna namjera je bila moderna baza

podataka koja se lako može prilagoditi web-u. Pokret je počeo početkom 2009. i ubrzano raste. Promoviše sledeću generaciju baza podataka, koje se najčešće zasnivaju na nekoj od sledećih karakteristika: nijesu relacione, distribuirane su, otvorene su koda (open source) i horizontalno skalabilne.

Tradicionalne baza podataka zahtijevaju predvidljive, strukturirane podatke. U realnim situacijama, kad je sistem dinamičan, relacioni model nije najbolje rješenje.

Najlakši način da se razmišlja o NoSQL bazama podataka je da ne prihvataju tradicionalnu RDMS strukturu (Relational Database Management System – sistem za upravljanje relacionim bazama podataka). Nijesu zasnovane na tabelama i ne koriste SQL za manipulisanje podacima.

NoSQL baze podataka omogućavaju visoke performanse i dostupnost, bogat upitni jezik i laku skalabilnost. [1]

3.2 Razdvajanje servisa

Baze podataka u prošlosti su definisane u smislu ažuriranja (izmjene podatka), u nastojanju da sačuva prostor na disku i memoriju. Tradicionalno, baze podataka služe da omoguće sledeće funkcije: koordinaciju, konsistenciju, indeksiranje, prostor za skladištenje i upite.

Ažuriranje podatka „na mjestu“ (izmjena neke vrijednosti u memoriji) zahtjeva da ovi servisi budu zajedno, kako bi podaci bili konzistentni i pouzdani. Tradicionalni DBMS server je zadužen za sve ove funkcije, i često postaje slaba tačka arhitekture aplikacije. Neke distribuirane arhitekture dijele podatke i dozvoljavaju nezavisnim čvorovima da obavljaju ove funkcije, ali ne omogućavaju razdvajanje servisa.

Odvajanje ovih servisa je ideja koja može pružiti fleksibilniju arhitekturu i otvoriti nove mogućnosti. Datomic razdvaja tradicionalni server baze podataka tako što odvaja prostor za skladištenje, upite i transakcije i tretira ih samostalno. [4]

Prednosti odvajanja servisa:

- mogućnost premještanja podsistema
- svaki servis brine o jednom dijelu posla
- komponente sistema se pojednostavljaju

3.3 Arhitektura Datomic-a

Datomic omogućava tri nezavisne uloge: [4]

1. Peer - biblioteka koja se ugrađuje u aplikaciju, uključuje sve komponente za komunikaciju sa transaktorom i servisima za skladištenje i obezbjeđuje aplikaciji pristup podacima, keširanje i mogućnost izvršavanja upita.
2. Transaktor (Transactor) – preuzima transakcije, procesira ih serijski, potvrđuje rezultat servisima za skladištenje, prenosi promjene peer-ovima i indeksira servise za skladištenje
3. Servisi za skladištenje (Storage services) - obezbjeđuju interfejs ka visoko pouzdanom, redundantnom prostoru za skladištenje

Prednosti Datomic arhitekture: [4]

1. Čitanje i pisanje u bazu podataka su razdvojeni
2. Distribucija integrisanih podataka
3. Svaki peer dobija svoj „mozak“ (mašinu za upite i keš)
4. Elastičnost
5. Spremnost za cloud
6. Brzina memorije

4. DATOMIC NASUPROT MSSQL-A

4.1 Kreiranje baze podataka

Datomic šema definiše skup atributa koji se mogu dodijeliti entitetima. Za svaki atribut šema navodi najmanje tri podataka: ime (*:db/ident*), tip (*:db/valueType*) i kardinalnost (*:db/cardinality* - da li atribut može da ima jednu ili više vrijednosti za dati entitet). Šeme mogu definisati i dodatne karakteristike atributa, npr. da li je vrijednost jedinstvena (*:db/unique*) ili da li je indeksirana za full-text pretragu (*:db/fulltext*). [5]

Prema konvenciji, atributi koji se koriste da opišu određeni entitet nazivaju se sa zajedničkim prefiksom (npr. *:klijent/ime*, *:klijent/prezime*).

Neki atribut je referenca (*:db.type/ref*) ukoliko se njegova vrijednost odnosi na drugi entitet. (npr. *:klijent/kompanija*).

```
{:db/id #db/id[:db.part/db]
:db/ident :klijent/ime
:db/valueType :db.type/string
:db/cardinality :db.cardinality/one
:db/fulltext true
:db/doc "Ime klijenta"
:db.install/_attribute :db.part/db}

{:db/id #db/id[:db.part/db]
:db/ident :klijent/prezime
:db/valueType :db.type/string
:db/cardinality :db.cardinality/one
:db/fulltext true
:db/doc "Prezime klijenta"
:db.install/_attribute :db.part/db}

{:db/id #db/id[:db.part/db]
:db/ident :klijent/kompanija
:db/valueType :db.type/ref
:db/cardinality :db.cardinality/one
:db/doc "Kompanija u kojoj je klijent zaposlen"
:db.install/_attribute :db.part/db}
```

Slika 1. Dio šeme koji definiše attribute za entitet Klijent

Kreiranje tabela u MSSQL bazi podataka vrši se putem SQL upita.

```
CREATE TABLE Klijent (
  id int primary key,
  ime nvarchar(50) not null,
  prezime nvarchar(50) not null,
  datum_unosa timestamp not null,
  fk_kompanija int
  references Kompanija (id)
)
```

Slika 2. Kreiranje tabele Klijent

Prilikom poređenja kreiranja Datomic baze podatke i MSSQL baze neophodno je istaći sledeće:

- Kada se kreiraju veze u Datomic šemi nije potrebno navoditi preko kog atributa se entiteti povezuju, jer se čuva čitav entitet, a ne samo pokazivač na njega.
- ID je atribut koji se u Datomic-u podrazumijeva i nije ga potrebno kreirati ručno.

- Nije potrebno kreirati *timestamp* promjenjive, koje pamte trenutak unosa podatka u bazu, jer je to po default-u implementirano u Datomic-u.

4.2 Unos podatka

Unos u Datomic bazu podataka obavlja se preko transakcija, korišćenjem funkcije *transact* iz Datomic API-ja koja prima dva ulazna parametra: konekciju i podatke. Podaci se prosleđuju kao mapa koja mora da sadrži *:db/id* (ključ koji identifikuje entitet koji se dodaje) i može da sadrži bilo koji broj atribut-vrijednost parova. [4][5]

```
(.get (.transact conn
      [{:db/id 1
        :klijent/ime "Marko"
        :klijent/prezime "Markovic"
        :klijent/kompanija 1}
       {:db/id 2
        :klijent/ime "Petar"
        :klijent/prezime "Petrovic"}]))
```

Slika 3. Unos podataka za entitet Klijent

Podaci u MSSQL bazu se unose preko SQL upita. Kako bi se C# kod povezao sa bazom podataka potrebno je kreirati *DataSet* u okviru projekta. *DataSet* sadrži tabele koje su potrebne aplikaciji i *TableAdapter* za svaku tabelu. *TableAdapter* sadrži SQL upite za tabelu za koju je vezan i kreira metode koje omogućavaju da se upiti izvršavaju iz C# koda. Sve vrijednosti za jedan red koji je potrebno unijeti u bazu podataka prosleđuju kao ulazni parametri metodi iz *TableAdapter*-a koja izvršava definisani upit.

```
INSERT INTO [dbo].[Klijent] ([id], [ime], [prezime], [fk_kompanija])
VALUES (@id, @ime, @prezime, @fk_kompanija);
```

Slika 4. SQL upit za unos u tabelu Klijent

```
klijentTableAdapter1.unesiKlijenta(1, "Marko", "Markovic", 1);
klijentTableAdapter1.unesiKlijenta(2, "Petar", "Petrovic", null);
```

Slika 5. C# metoda za unos podatka o klijentima

Kod za unos podataka u Datomic bazu je znatno jednostavniji, Clojure funkcije direktno kreiraju strukturu podataka za unos u bazu i prosleđuju je transктору. Ovaj pristup omogućava programeru pregledan kod, koji se nalazi na jednom mjestu, koji je jednostavan za razumijevanje i pisanje, ali i izmjenu i testiranje.

Takođe moguće je kreirati strukturu koja sadrži sve entitete koje je potrebno unijeti u bazu i tek onda je prosljediti transктору, dok se u drugom slučaju za svaki red koji potrebno unijeti kontaktira baza i izvršava SQL kod.

4.3 Unos podatka

Datomic upitni jezik se zasniva na Datalog-u, formalnom jeziku za pronalazjenje podataka definisanjem skupa logičkih pravila i uslova. Datomic upiti se definišu kao strukture podatka.

Upit se sastoji od tri dijela:

- *:find* – definiše koje attribute upit treba da vrati
- *:in* – specifikuje izvor podataka (nije neophodan)
- *:where* – definiše jednu ili više klauzula

Lako se pristupa atributima entiteta koji su povezani preko referenci, što omogućava hijerarhijski izbor objekata.

Upiti se izvršavaju nad trenutnim stanjem baze podataka, koje se dobija pozivanjem *db* funkcije nad kreiranom konekcijom. Ova funkcija vraća referencu na *datomic.Database*, koja predstavlja stanje baze podataka u trenutku preuzimanja.

Upiti se izvršavaju korišćenjem funkcije *q*, koja kao ulazne parametre prima upit i ulazne podatke i vraća podatke iz baze u obliku *HashSet*-a (*java.util.HashSet*) u kom je svaki red preuzetih podataka predstavljen kao vektor (*clojure.lang.PersistentVector*).

Clojure funkcije kreiraju upit (koji sadrži sve potrebne parametre) i prosleđuju ih funkcijama iz Datomic API-ja koje ih izvršavaju. [4][5]

```
(defn pronadji-klijente-po-imeni [ime]
  (d/q '[:find ?i ?p ?n
        :in $ ?ime
        :where
        [?c :klijent/ime ?i ?ime]
        [?c :klijent/prezime ?p]
        [?c :klijent/kompanija ?k]
        [?k :kompanija/naziv ?n]]
    (d/db conn)
    ime))
```

Slika 6. Funkcija koja pronalazi sve klijente zadatog imena i vraća njihova imena, prezimena i nazive kompanija

Kada je u pitanju MSSQL postavljanje upita nad bazom podataka funkcioniše na isti način kao i unos. C# definiše kriterijume koje kao parametre prosleđuje *TableAdapter*-u koji izvršava definisani upit.

SQL upiti koji sadrže parametre su nepregledni i lako dolazi do grešaka u kucanju. Takođe upiti se nalaze van programskog koda (u *TableAdapter*-ima) što otežava proces izmjene, testiranja i pronalazjenja greške. Ukoliko je, na primjer, potrebno dodati novi parametar u upitu izmjena se mora izvršiti na dva mjesta – u C# kodu i u kodu SQL upita.

```
SELECT Klijent.ime, Klijent.prezime, Kompanija.naziv
FROM Klijent INNER JOIN
      Kompanija ON Klijent.fk_kompanija = Kompanija.id
WHERE (Klijent.ime = @ime)
```

Slika 7. SQL upit koja pronalazi sve klijente zadatog imena i vraća njihova imena, prezimena i nazive kompanija

```
DataTable klijentiPoImenu =
  klijentTableAdapter1.pronadjiKlijentePoImenu(ime);
```

Slika 8. Preuzimanje rezultata upita u C# kodu

Navedeni problemi se ne pojavljuju u Datomic upitima koji se kreiraju direktno u okviru Clojure funkcija.

5. ANALIZA REZULTATA

U ovom poglavlju će biti predstavljeni rezultati i zaključci nakon poređenja dvije aplikacije, uzimajući u obzir programski kod, bazu podataka i performanse.

5.1 Programski kod

Programerima koji su navikli na razvoj aplikacija u objektno-orijentisanim programskim jezicima, navikavanje na funkcionalno programiranje može predstavljati problem. Kod koji se zasnivao na objektima, promjenjivim i metodama

potrebno je zamijeniti funkcijama i prihvatiti pristup u kom promjenjive činjenice ne postoje, nema promjene stanja i vrijednosti. U početku dolazi do nesnalaženja, čestih grešaka i pokušaja da se u funkcionalnom programskom jeziku implementira objektno-orjentisani način razmišljanja. Nakon iščitavanja literature i implementacije jednostavnih primjera polako se ulazi u suštinu funkcionalnog pristupa.

Aplikacija razvijena u C# programskom jeziku koristi klasični objektno-orjentisani pristup, sadrži više fajlova sa velikim brojem linija koda.

Clojure programski jezik omogućava da se iskoriste sve prednosti funkcionalnog programiranja i da se kreira dinamičan i kompaktan programski kod. Kod aplikacije napisane u Clojure-u je znatno kraći, ali i jednostavniji, lakši za snalaženje, izmjenu i testiranje.

5.2 Baza podataka

MSSQL baza podataka, koja je korišćena za C# aplikaciju, predstavlja tipičan primjer relacione SQL baze podataka. To je tradicionalni pristup čuvanju podataka, na koji su svi programeri navikli i koji im je poznat. Baza podataka i njena šema kreiraju se odvojeno od aplikacije i svaka naknadna izmjena zahtjeva promjenu koda na više mjesta, a testiranje i izvještavanje o greškama je otežano.

Datomic baza podataka je jedna od novih NoSQL baza podataka, koja se zasniva na vremenskim činjenicama. Ima fleksibilniju šemu i omogućava kreiranje baze i šeme za podatke direktno iz programskog koda. Za preuzimanje podatka koristi se Datalog upitni jezik koji se takođe može koristiti direktno u kodu i koja preuzima podatke u Clojure strukturama koje su lake za manipulisanje.

Iz navedenih razloga može se zaključiti da Datomic baza podatka omogućava jednostavnije povezivanje sa aplikacijom, ali i unos i preuzimanje podataka. Kod koji kreira upite je pregledan, jednostavan za razumijevanje i pisanje, i nalazi se na jednom mjestu. To što se podaci iz Datomic baze podataka nikad ne brišu takođe predstavlja značajnu prednost, zbog poslovne logike i sigurnosti podataka

5.3 Performanse

Performanse aplikacija izračunate su na primjeru 6000 zapisa

Tabela 1. Poređenje performansi

Način implementacije/ Funkcionalnost	Unos podataka	Preuzimanje podataka
C# aplikacija sa MSSQL bazom podataka	44,3 sec.	0.39 sec.
Clojure aplikacija sa Datomic bazom podataka	28,1 sec.	0.35 sec.

Kao što se može vidjeti iz tabele aplikacija napisana u Clojure programskom jeziku sa Datomic bazom podatka omogućava veću brzinu kako unosa, tako i preuzimanja podataka. Na ovom primjeru razlika nije toliko velika, ali je značajna jer se količina podataka svakodnevno povećava.

6. ZAKLJUČAK

SQL baze podataka su opšte prihvaćene i najčešće korišćene, ali imaju niz problema koji utiču na sve češće razmišljane developer-a o promjeni pristupa i prelasku na nerelacione baze podataka. Kako relacione baze podataka zahtijevaju predvidljive, strukturane podatke, u realnim situacijama dinamičnih sistema, ovakav tradicionalni pristup strukturiranja podataka se nije pokazao kao najbolje rješenje. U tradicionalnoj arhitekturi baza podataka DBMS server je zadužen za ove funkcije i često predstavlja usko grlo sistema.

S druge strane, NoSQL baze podataka ne prihvataju tradicionalnu RDMS strukturu, nisu zasnovane na tabelama i ne koriste SQL za manipulisanje podacima, što im obezbjeđuje fleksibilnost u procesu obrade podataka, hijerarhijsku strukturu koja omogućava pouzdan rad u cloud-u i skalabilnost, visoke performanse i dostupnost kao i bogat upitni jezik.

Datomic je NoSQL baza podatka koja omogućava distribuirani sistem, odvaja obradu transakcija, skladištenje, keširanje i izvršavanje upita. Time se povećava skalabilnost i obezbjeđuje potrebna za redundantnost za cloud sisteme, zadržavajući performanse. Datomic premješta mogućnost manipulacije podacima, ali i same podatke, u aplikacije, i ima fleksibilan model podataka. Ovo predstavlja osnovu za sledeću generaciju elastičnih i inteligentnih aplikacija, oslobođenih granica koje postavljaju tradicionalne baze podataka.

U ovom radu je na praktičnom primjeru izvršeno poređenje Datomic-a sa relacionom bazom podataka MSSQL, sa stanovišta programskog koda i performansi.

Kako je aplikacija razvijena u funkcionalnom programskom jeziku može se zaključiti da je funkcionalno programiranje moćnije od objektno-orjentisanog, zbog daleko jednostavnijeg rješavanja problema u mnogo manje linija koda. Implementacija sistema upotrebom Datomic baze podatka dovela je do poboljšanja performansi i olakšala rad sa podacima.

LITERATURA

- [1] Jeff Cogswell, „SQL vs. NoSQL: Which Is Better?“, <http://slashdot.org>, 2012
- [2] Luke Vander Hart, Stuart Sierra, *Practical Clojure*, Apress, 2010
- [3] Chas Emerick, Brian Carper, Christophe Grand, *Clojure Programming*, O'Reilly, 2012
- [4] Datomic baza podatka: <http://www.datomic.com>
- [5] Datomic Clojure API: <http://docs.datomic.com/clojure/>

KRIPTOSISTEMI JAVNOG KLJUČA I GOLDBAHOVA PRETPOSTAVKA PUBLIC KEY CRYPTOSYSTEMS AND GOLDBACH CONJECTURE

Dragan Vidakovic, Dusko Parezanovic, Gimnazija, Ivanjica, Serbia

Sadržaj: U ovom radu ćemo predstaviti eksperimentalne rezultate koji jasnije od svake teorije sugeriraju odgovor na pitanje: Kada pri detekciji velikih (verovatno) prostih brojeva treba primeniti, resursno veoma zahtevan Miler-Rabinov algoritam. Goldbahova pretpostavka takođe može biti vrlo korisna u našoj nameri da što je brže moguće detektujemo velike proste brojeve, kao i za još neka pitanja u vezi RSA, što će takođe biti prikazano u ovom radu.

Abstract: In this paper we present the experimental results that more clearly than any theory suggest an answer to the question: when in detection of large (probably) prime numbers to apply, a very resource demanding, Miller-Rabin algorithm. Goldbach conjecture may also be helpful in intention to detect large prime numbers as quickly as we can, and for some other questions about RSA cryptosystem, which we will also show in this paper.

1. INTRODUCTION

In asymmetric schemes of protecting the confidentiality and integrity of data there is a need for large prime numbers. For some tasks required number of bits now exceeds 15000, and it is still just a passing figure in the endless game of those who protect data and those who attack them. It is therefore quite clear that the time spent on detection of large prime numbers must be as short as possible.

It would be best to check the divisibility of number n with all prime numbers less than or equal to \sqrt{n} . However, with so many bits it's not realistic. Therefore, the number which is tested to primality is previously divided by several tens of first prime numbers and then, if it is not divisible by any of these numbers, it is left to Miller-Rabin (MR) algorithm [1]-[2].

The trial division takes less time than exponentiation, but it would certainly be wrong to conclude that we should divide the number as long as possible. It is very difficult to determine the real relation between the two, since everything depends on the number we start with and odd numbers we examine so as to generate a probable prime.

It is a very difficult task to find theoretically an optimum ratio of time required for dividing the number and testing by Miller-Rabin algorithm. Perhaps a redundant task as well in terms of our needs, since in practical tasks such as ours, we have no reason to pretend that computers do not exist, that the experimentally obtained, very useful results are less valuable than the values obtained theoretically.

As a useful tool for our task (minimizing the time required for detection of prime numbers) we see the Goldbach conjecture [3]-[4], which states that every (large for us) even number is the sum of two prime numbers. It may forever remain a conjecture, or one day some talented mathematician may write a book of hundreds of pages that

will prove its truth, or some computer may find the number for which it is not valid, and with that break the conjecture.

For those of us looking for large prime numbers none of these three matters. We will, in any case, generate a random large even number $2n$, of, say, 1024 bits, and detect a much smaller random prime number of, say, 128 or 256 bits, which is negligible in terms of time, and then verify that the difference of those two numbers is a prime number. If so, we have a large prime number. If not, we will repeat the procedure or we will use this difference to generate the prime number closest to it by a combination of dividing by first prime numbers and Miller-Rabin algorithm. Experimentally we will ensure that the above procedures may also result in saving the time required to detect a large prime number [5]-[6].

2. EXPERIMENTAL RESULTS

In order to avoid unnecessary applications of Miller-Rabin algorithm to the number in question, we resort to trial division by a few initial prime numbers, since such a division takes less time.

How far we should go with such a division is the that we are trying to answer in this paper? For the theory of the matter is fully resolved. However, that in practice we do not have much use.

Therefore, we present two solutions that are probably irrelevant to theorists, but it is very useful to people who have spent many nights to produce large (probably) prime numbers using its own software.

2.1. DIVIDING BY FIRST PRIME NUMBERS

In this paragraph we show the results of detection of prime numbers of 513, 1024 and 1500 bits, namely: without

dividing by prime numbers, dividing by first 10, 20, 30,..., 100, 200 prime numbers.

Example 1

If we start with number c (513 binary digits - BD) with ones in places: 512, 255, 200, 127, 100, 50, 10, 9, 8, 7, 2, 1, 0. By dividing and testing we intend to detect first prime number with ones in places: 512, 255, 200, 127, 100, 50, 11, 9, 6, 3, 2, 1, 0. As a result we have the following Figure 1:

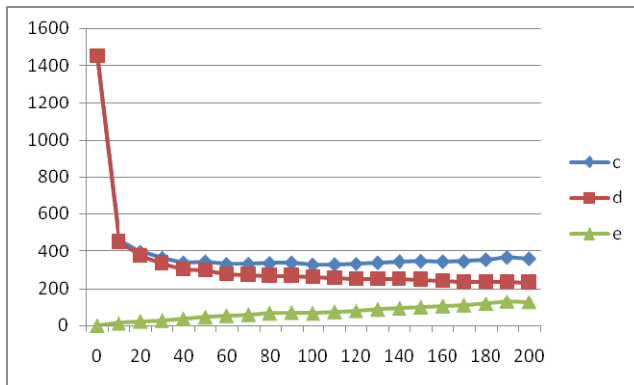


FIGURE 1. The Timing of Detection of a Prime Number

Where the following labels are valid:

- c- the total time needed for detection of a prime number
- d- time spent on the MR algorithm
- e- time spent on the division by first prime numbers
- X axis – the number of primes by which we divide
- Y axis – time in seconds

It is clear from the Figure 1 that the search for prime numbers, without dividing by first prime numbers, is not an option. This is an unnecessary waste of time. Dividing by first ten prime numbers would be a minimum. Dividing by 60, 70, ... would be a good choice (Fig. 1). The choice of tens of numbers more or less could make little savings or a small loss of time and would not significantly affect the quality of our task.

2.2. USING GOLDBACH CONJECTURE

Goldbach set the conjecture that "every even number (2n) larger than four is the sum of two prime numbers."

Our idea is to detect a prime number p less than n and then to examine whether the difference (2n-p) is a prime number. Given that we choose p to be less than n (in order to detect it faster) and 2n to be a large number, which, if 2n-p is a prime, gives us a large prime number, avoiding the search through the upper part (numbers larger than n) which is in terms of time far more demanding job than detection of prime numbers in the lower part (numbers less than n).

Of course all of this is possible if there are enough pairs with simple coordinates between all pairs of numbers (p, q), where p + q = 2n, p - prime and q - odd number.

With our own software we conducted an experiment whose aim was to find all pairs (p, q) for given number 2n, p ≤ n-1, q ≥ n + 1, where p is a prime number and q is an odd number and p + q = 2n (*₁). Then, among these pairs to find those in which the second coordinate is prime number (pairs of Goldbach conjecture (GC)) and to measure the time of finding a number of representations of number 2n which satisfy the Goldbach conjecture.

TABLE 1. Number of GC Pairs

Even number	2 ²⁵	2 ²⁶	2 ²⁷
Number of Pairs GC	83543	153881	283830
Number of pairs (* ₁)	1078257	2064123	3958400
% (GC) in (* ₁)	7.75%	7.45%	7.17%

The Table 1. shows that Goldbach conjecture can be a useful tool in our task because there is a probability, though not large, of guessing the large prime number. A possible loss of time in detecting the prime number for the first coordinate is negligible because it is number less than n, and it is particularly negligible compared to the possibility to immediately detect the other simple coordinate- a large prime number

3. SOME FURTHER OBSERVATION

If we consider the time of finding all (GC) pairs of some even number 2n, we can see that with the increase of number n, the time to find them significantly increases.

TABLE 2. Time to Find a GC Pairs

Even number	2 ²⁵	2 ²⁶	2 ²⁷
Number of pairs GC	83543	153881	283830
Time of finding the pairs GC	2h 1' 14"	3h 29' 48"	7h 18' 17"

3.1. ECC

Even if we know 2n = 2²⁷, it takes more than seven hours to find a number of its GC representations (Table 2.) (Intel Pentium CPU B950 2.10 GHz, RAM 4.00 GB). If we know the number of GC representations it is impossible to guess the number whose representations those are, which resembles a Goldbach "multiplication of point", which ends the "similarities" with ECC (Elliptic curve cryptosystems) [5], because we cannot define the addition of points.

3.2. RSA CRYPTOSYSTEM

The RSA cryptosystem is the most widely used public-key cryptosystem. It may be used to provide both secrecy and digital signatures and its security based on the

intractability of the integer factorization problem [1].

Key generation for the RSA encryption:

Each entity creates an RSA public key and a corresponding private key.

Algorithm

Each entity A should do the following:

- Generate two large distinct random primes p and q , each roughly the same size
- Compute $n = p * q$ and $\phi = (p - 1) * (q - 1)$.
- Select a random integer e , $1 < e < \phi$, $\text{gcd}(e, \phi) = 1$
- Use the extended Euclidean algorithm [1]. (Algorithm 2.107) to compute the unique integer d , $1 < d < \phi$, such that $ed \equiv 1 \pmod{\phi}$.
- A's public key is (n, e) ; A's private key is d

Now we point out the two possible connection between RSA and the Goldbach conjecture:

3.2.1. THE FIRST POSSIBILITY

Only powerful computers can calculate (GC) numbers of 1024, 2048, or a larger number of bits. We have no reasons not to believe that $(*_1)$ and (GC) are greater and greater numbers and at the same time (probably) they are unique for a given number $2n$. Even if various even numbers have the same representation it does not matter for us because we will create a table that will contain in each row for a given even number the hash value $(*_1)$ and (GC).

For large, probably prime numbers p_1 and q_1 we will calculate the number $2n = p_1 * q_1 + 1$.

For this number we will find $(*_1)$ and (GC) and their hash values $h(*_1)$ and $h(GC)$. The procedure will be repeated k times and the table of k rows (each of which contains an even number and its corresponding values $h(*_1)$ and $h(GC)$) will be, in a safe manner, delivered to users, which means that we use the known techniques for the management of the public keys and their certificates. The implementation of the table mentioned is also possible to the devices (only authorized users) which require mutual authentication, with the central device as a trusted third party (TTP), (contactless) smartcards, and others. It is clear that RSA cryptosystem set up this way can be competitive with ECC, both in wireless communications and wireless sensor networks.

Instead of a pair as a public key $(2n-1, e)$, we suggest that the first part of the public key, instead of $2n-1$, be $h(*_1)$ and $h(GC)$, based on which the user would set the number $2n$ by reading the table, and therefore the number of $2n-1$ (the RSA modulus) would be known.

It is clear that this procedure does not weaken the RSA. It just makes it difficult for those who intend to reveal the secret key, because prior to the use of algorithms for finding prime factors of the number $2n-1=p_1*q_1$, that

number should be determined primarily, which is very difficult for large numbers if we know only $h(*_1)$ and $h(GC)$.

3.2.2. THE SECOND POSSIBILITY

Another possibility would be publishing the number $2n$, which implicitly publishes (GC), too. (GC) may be (another part of the key pairs) public key for RSA (in the standard label e) if $\text{gcd}((GC), \theta) = 1$, or the first number greater than it that is relatively prime to θ , where $\theta = (p_1 - 1) * (q_1 - 1)$. This would be a semi-public key cryptosystem, as the users in addition to the secret key d , obtain in the same way, safely, the public key e , while others who have bad intentions must first find (implicitly published) public key e , which is a very demanding job in terms of time, and only then they may access the disclosure of the secret key d . It is clear that in the meantime, we can change the parameters of RSA and thus further complicate efforts to breach confidentiality and integrity of our data.

4. CONCLUSION

This work, too, is in line with our belief that it is necessary that each country protects the confidentiality and integrity of its data using its own software. Good experts are a prerequisite for this, and they cannot exist without the increased interest of young people in cryptography. We believe that there is no such an interest without more interesting approach to cryptography, and encryption of cryptographic algorithms and experimentation with own software is the best way for that. To this end we have written this paper dealing with such an important topic for practical cryptography: Minimizing the time of detection of large (probably) prime numbers.

REFERENCES.

- [1] A. Menezes, P.C. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, New York, 1997.
- [2] D. Vidakovic, *Analysis and implementation of asymmetric algorithms for data secrecy and integrity protection*, Master Thesis (mentor J. Golic), Faculty of Electrical Engineering, Belgrade, Serbia, 1999.
- [3] Goldbach, C., *Letter to L. Euler*, June 7, 1742.
- [4] Sylvester, J. J., *On the partition of an even number into two primes*, Proc. London Math. Soc., s1-4(1). 4-6. 1871.
- [5] Koblitz N., *Elliptic Curve Cryptosystems*, Mathematics of Computation, 48, pp. 203-209, 1987.
- [6] D. Vidakovic, D. Simic, *A Novel Approach to Building Secure Systems*, Second International Conference on Availability, Reliability and Security, In 1th IEEE International Workshop on Secure Software Engineering (SecSE 2007), Vienna, 2007., Austria, pp 1074-1081

POBOLJŠANA METODA VALIDACIJE $3n+1$ HIPOTEZE PUTEM TRANSFORMACIJA IMPROVED METHOD OF VALIDATION THROUGH TRANSFORMATIONS FOR $3n+1$ HYPOTHESIS

Srđan Kadić, Milenko Mosurović, Prirodno-matematički fakultet, Podgorica

Sadržaj: Pomoću jednostavnih aritmetičkih operacija $3n+1$ problem je transformisan u njemu ekvivalentan problem koji je jednostavniji za provjeru pomoću računara. Predložen je i implementiran novi algoritam koji je iskorišćen za razna numerička testiranja. Pokazano je da $3n+1$ hipoteza važi za sve stepene trojke zaključno s brojem 3^{32768} .

Abstract: By using simple arithmetic operations the $3n+1$ problem was transformed into its equivalent but such that it was more suitable for computer validation of $3n+1$ hypothesis. A new algorithm was proposed and implemented and it was used for various numerical tests. The paper proves that $3n+1$ hypothesis is valid for 3^n where $n = 1, 32768$.

1. UVOD

U matematici postoje problemi čija je formulacija jednostavna i mnogima je razumljiva, međutim još niko nije pronašao rešenje za te probleme. Jedan od takvih problema je i $3n+1$ problem. On je dugo godina usmeno prenošen, tako da nema publikacija vezanih za ovaj problem sve do početka 70-tih godina XX vijeka. Postoje izvori [4,5] koji govore da je ovaj problem od različitih predavača navođen u okviru predavanja na mnogim univerzitetima počev od 1929. godine, međutim danas se generalno taj problem pripisuje Lothar Collatz-u. Otuda je $3n+1$ problem poznat i kao Collatz-ov problem, a odgovarajuća funkcija se naziva Collatz-ovom funkcijom i definiše sa:

$$C(n) = \begin{cases} \frac{n}{2}, & n - \text{parno}, \\ 3n + 1, & n - \text{neparno}. \end{cases}$$

Vezano za ovu funkciju postavljena je sledeća hipoteza.

Hipoteza 1. Za svaki prirodan broj n postoji prirodan broj m takav da je $C^m(n)=1$.

Napomenimo da $C^m(n)$ označava uzastopnu kompoziciju funkcije $C(n)$ i definiše se induktivno sa $C^0(n) = n$, $C^{m+1}(n) = C(C^m(n))$, za $m \geq 0$.

Formulacija $3n+1$ problem glasi: Da li je prethodna hipoteza tačna. Drugim riječima, pitanje je da li će se uzastopnom primjenom Collatz-ove funkcije na proizvoljan prirodan broj uvijek dobiti jedinica. Rešenje problema je ako se dokaže da hipoteza 1. važi ili se nađe prirodan broj za koji hipoteza ne važi. Moguće je $3n+1$ problem formulirati i kao igru. Naime, Alisa zapisuje jedan prirodan broj. Bob pobeđuje ako je zapisani broj jedinica, inače Bob gleda da li je poslednji zapisani broj paran i ako jeste dijeli ga sa dva, a ako je neparan množi ga sa tri i dodaje jedan. Bob zapisuje broj koji je dobio i postupak ponavlja. Pitanje je da li Bob uvijek pobeđuje.

Primjer. Ako Alisa zapiše broj 7 onda imamo sledeći niz brojeva:

7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1. Naime, Bob je računao $3 \cdot 7 + 1 = 22$, a zatim $22/2 = 11$, itd. Na kraju je Bob zapisao jedinicu i tako pobeđio. Broj iteracija koji je Bob primjenio u ovom primjeru je 16 tj. $C^{16}(7)=1$. Takođe iz navedenog niza možemo zaključiti da je $C^{12}(17)=1$, dok je $C^4(16)=1$. Maksimalna vrijednost koju smo dobili pri ovim izračunavanjima je 52.

Mada je formulacija $3n+1$ problema u terminima igre razumljiva svakome ko zna osnovne matematičke operacije, problem prema našim saznanjima još uvijek nije riješen. Danas ima više od nekoliko stotina publikacija koje se dierktno ili indirektno, sa različitih aspekata, bave $3n+1$ problemom. Čak postoji nekoliko radova [2,3] koji tvrde da su riješili $3n+1$ problem, ali za svaki od njih se ispostavilo da sadrže grešku u dokazima. Jedan od aspekata vezanih za $3n+1$ problem je i razvoj efikasnih algoritama koji za dati broj n provjeravaju tačnost hipoteze 1 (vidi [6,7]), kao i druga svojstva vezana za $3n+1$ problem [9]. Upravo ovaj rad se bavi tim aspektom. Naime razvijen je efikasan algoritam za provjeru hipoteze 1, a zatim je iskorišćen za testiranje hipoteze na velikim brojevima specijalnog oblika. Svakako osnovni cilj rada je da se otkriju svojstva koja mogu pomoći matematici da dokaže hipotezu 1 ili pronađe kontraprimjer.

Pored uvoda rad je dalje organizovan još u 4 poglavlja. Drugo poglavlje daje kratak pregled nekih rezultata koji su povezani sa $3n+1$ problemom. U trećem poglavlju se opisuju potrebne transformacije neophodne za konstrukciju algoritma kao i opis samog algoritma. Četvrto poglavlje daje pregled rezultata i nekih statističkih podataka dobijenih testiranjem. Peto poglavlje je zaključak.

2. POZNATI REZULTATI I OZNAKE

Najčešće korišćena preformulacija $3n+1$ problema iterira drugačiju funkciju, koja se naziva $3n+1$ funkcijom i data je sa:

$$T(n) = \begin{cases} \frac{n}{2}, & n - \text{parno}, \\ \frac{3n+1}{2}, & n - \text{neparo}. \end{cases}$$

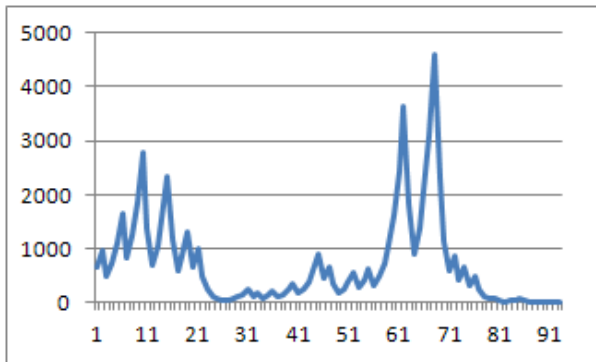
Sa tačke gledišta iteracija ove dvije funkcije su u jednostavnoj relaciji; iteracije funkcije $T(n)$ preskaču neke korake u iteracijama funkcije $C(n)$. To preskakanje se dešava kada je n –neparan broj. Naime važi:

$$T(n) = \begin{cases} C(n), & n - \text{parno}, \\ C(C(n)), & n - \text{neparo}. \end{cases}$$

Otuda se u hipotezi 1. funkcija $C(n)$ može zamjeniti sa funkcijom $T(n)$. U literaturi se uglavnom proučavaju svojstva funkcije $T(n)$. Na primjer interesantna su sledeća dva svojstva [6].

- $T^k(2^k - 1) = 3^k - 1$.
- Neka je $n_0 = n_k 2^k + m_k$, gdje je $n_k = \lfloor n_0 / 2^k \rfloor$ i $m_k = n_0 \bmod 2^k$. Opšti oblik od $T^k(n_0)$ je $T^k(n_0) = 3^{y(k, n_0)} n_k + T^k(m_k)$, gdje je $y(k, n_0)$ broj napravljenih neparanih koraka tj. broj elemenata skupa $\{j | 0 \leq j < k, T^j(m_k) \equiv 1 \pmod{2}\}$.

Prvo svojstvo pokazuje da u nekim slučajevima uzastopne iteracije funkcije $T(n)$ mogu da povećavaju vrijednost, dok je drugo svojstvo uopštenje prvog svojstva. Kako se menjaju vrijednosti pri uzastopnim iteracijama pogodno je prikazivati grafički, zato se uvodi pojam trajektorije (orbite). Trajektorija elementa x za funkciju $f(\cdot)$ je niz iteracija $(x, f(x), f^2(x), f^3(x), \dots)$. Trajektorija broja 649 za funkciju $T(\cdot)$ je prikazana na slici 1.



Slika 1. Trajektorija broja 649 za funkciju T . Maksimalna vrijednost trajektorije je 4616.

Trajektorija elementa x za funkciju f je periodična ako postoji $k \geq 1$ tako da je $f^k(x) = x$. Za sada su 1, 2 jedini poznati prirodni brojevi čije su trajektorije periodične za funkciju T . Napomenimo da postoje i drugi cijeli brojevi čije su trajektorije periodične za funkciju T . Na primjer 0, -1, -5, -17. Otuda interesantno pitanje da li su 1,2 jedine prirodni brojevi čije su trajektorije periodične za funkciju T . Vezano za ovo pitanje interesantno je spomenuti rezultate iz rada [1]. Formirajmo $n \times n$ nula-jedan matricu A_n čiji su elementi

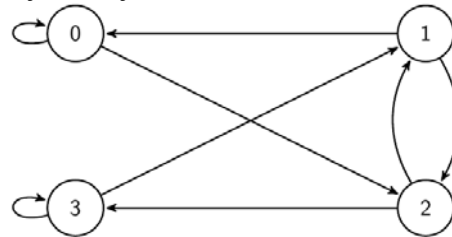
$$a_{ij} = \begin{cases} 1, & \text{za } T(i) = j, \\ 0, & \text{inače.} \end{cases}$$

Tvrđenje da je 1,2 jedina periodična trajektorija za funkciju T među prirodnim brojevima je ekvivalentno da je $\det(I - xA_n) = 1 - x^2$ za svako $n \geq 1$. Takođe je pokazano da je $\det(I - xA_n) = \det(I - xA_{n-1})$ za sve $n \neq 8 \pmod{18}$.

Zaključeno je [1] da ako postoji druga periodična trajektorija među prirodnim brojevima onda postoji $m \equiv 8 \pmod{18}$ tako da je $n = \frac{m}{2}$ periodična trajektorija. Otuda je za pronalaženje druge periodične trajektorije (ako postoji) pomoću računara dovoljno ići sa korakom 18.

Kod eksperimentalne provjere hipoteze 1. bitan je korak kada treba stopirati iteracije. Zato su uvedeni i sledeći pojmovi. Vrijeme zaustavljanja za broj n , u oznaci $\sigma(n)$ je minimalan broj k za koji važi $T^k(n) < n$, ako takav broj postoji, inače je ∞ . Totalno vrijeme zaustavljanja $\sigma_\infty(n)$ za broj n je minimalan broj k za koji važi $T^k(n) = 1$, ako takav broj postoji, inače je ∞ . Ako hipoteza 1 važi za sve prirodne brojeve manje od n , onda da bi dokazali da važi i za n dovoljno je dobiti $T^k(n) < n$. Drugim riječima treba dokazati da je $\sigma(n)$ konačan broj. Ako ne znamo informacije o prethodniku broja n onda moramo dokazati da je $\sigma_\infty(n)$ konačan broj, da bi hipoteza 1 bila tačna za broj n .

Ako pretpostavimo da je hipoteza 1 tačna pripodan je pristup da se pronađe podskup $S \subset N$ takav da ako je hipoteza 1 tačna za brojeve iz S onda je tačna i za sve prirodne brojeve. Lako se provjerava da prethodno svojstvo važi za skup $S = \{m \in N | m \equiv 3 \pmod{4}\}$. Naime za druge brojeve posle konačnog broja iteracija dobićemo element iz S ili će broj da se smanjuje svaki put kada primjenimo jednu ili dvije iteracije. To se vidi sa slike 2.



Slika 2. Dinamika iteracija funkcije T po modulu 4.

U radu [8] je pokazano da je dovoljno posmatrati skup

$$S = \left\{ m \in N \mid m \equiv 3 + \frac{10}{3} (4^k - 1) \pmod{2^{2k+2}} \right\}$$

za proizvoljan fiksiran prirodan broj k .

3. OSNOVNE TRANSFORMACIJE I ALGORITAM

Kao što je navedeno u prethodnom poglavlju obično se umjesto funkcije $C(n)$ posmatra funkcija $T(n)$. Ovo je posebno pogodno kod implementacije algoritama jer se smanjuje broj koraka. Kada su brojevi prikazani u binarnom brojnem sistemu, u računarskim sistemima, dijeljenje sa stepenom dvojke se izvodi efikasno jer se svodi na pomjeranje za odgovarajući broj bita. Otuda je osnovna ideja da se funkcija $T(n)$ transformiše na takav način da se djeljenja sa stepenom dvojke objedine.

Ako je n paran broj onda za $k = \max_{j \in N} \{j | n \equiv 0 \pmod{2^j}\}$ važi $T^k(n) = \frac{n}{2^k}$. Uočimo da je u ovom slučaju $T^k(n)$ neparan broj, pa poslje njega bi primjenjivali „neparan korak“ (tj. računali bi po formuli kada je broj neparan). S druge strane ako je n neparan broj možemo zapisati $T(n) = \frac{3n+1}{2} = \frac{3(n+1)}{2} - 1$. Očigledno, ako

je $\frac{n+1}{2}$ paran broj rezultat $T(n)$ je neparan broj. Otuda je $T^2(n) = T\left(\frac{3(n+1)}{2} - 1\right) = \frac{3^2(n+1)}{2^2} - 1$. Matematičkom indukcijom za $l \leq k = \max_{j \in \mathbb{N}} \{j | (n+1) \equiv 0 \pmod{2^j}\}$ se lako dokazuje da važi $T^l(n) = \frac{3^l(n+1)}{2^l} - 1$. Specijalno, $T^k(n) = \frac{3^k(n+1)}{2^k} - 1$ je paran broj, pa poslje njega slijedi „paran korak“. Drugim riječima sve uzastopne parne (neparne) korake možemo zamjeniti sa jednim „višestrukim“ parnim (neparnim) korakom. Tako će se parni i neparni koraci naizmjenično smjenjivati.

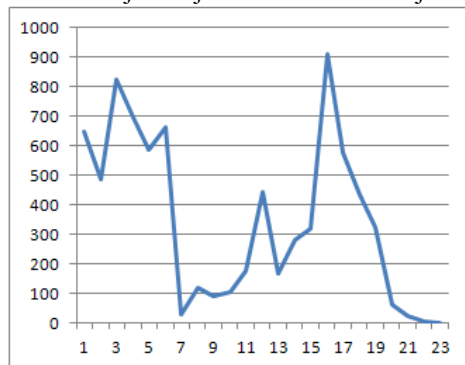
Imajući prethodno u vidu mogu se definisati dvije funkcije koje slikaju skup neparanih prirodnih brojeva u skup neparanih prirodnih brojeva. $Heven(n) = \frac{n-1}{2^p}$, gdje je $p = \max_{j \in \mathbb{N}} \{j | (n-1) \equiv 0 \pmod{2^j}\}$ i $Hodd(n) = \frac{3^k}{2^k}(n+1)$, gdje je $k = \max_{j \in \mathbb{N}} \{j | (n+1) \equiv 0 \pmod{2^j}\}$. Jednostavnim transformacijama lako je pokazati da važi sledeća lema.

Lema. Neka je n neparan prirodan broj. Naka su dalje brojevi k i p dati sa $k = \max_{j \in \mathbb{N}} \{j | (n+1) \equiv 0 \pmod{2^j}\}$, $p = \max_{j \in \mathbb{N}} \{j | (Hodd(n) - 1) \equiv 0 \pmod{2^j}\}$. Tada važi $Heven(Hodd(n)) = T^{k+p}(n)$.

Značenje prethodne leme je da za neparan broj n važi da ako postoji m takvo da je $T^m(n) = 1$, onda postoji q tako da je $(Heven \circ Hodd)^q(n) = 1$, u ovom slučaju broj q nazivamo totalno vrijeme zaustavljanja za funkciju $Heven \circ Hodd$ i označavamo sa $\sigma_n(n) = q$. Imajući u vidu da za paran broj n primjenom konačno mnogo puta funkcije T će se dobiti neparan broj ili jedinica možemo zaključiti da je hipoteza 1. ekvivalentna sledećoj hipotezi.

Hipoteza 2. Za svaki neparan prirodan broj n postoji prirodan broj q takav da je $(Heven \circ Hodd)^q(n) = 1$.

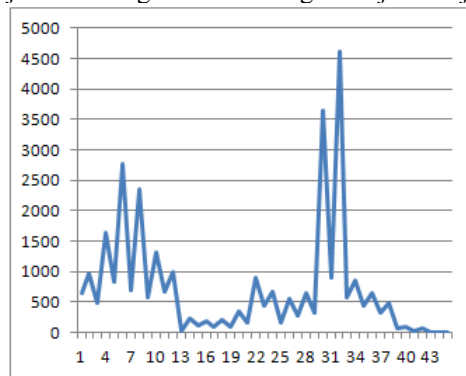
Trajektorija broja 649 za funkciju $Heven \circ Hodd$ je prikazana na slici 3. Kada uporedimo trajektorije sa slike 1 i slike 3 vidimo da trajektorija na slici 3 ima manje osilacija.



Slika 3. Trajektorija broja 649 za funkciju $Heven \circ Hodd$.

Međutim na slici 3 je skriven međukorak koji stvara funkcija Hodd. Na slici 4 je prikazana trajektorija broja 649 za funkciju $Heven \circ Hodd$ sa međukorakom. Visimo da trajektorije sa slike 1 i slike 4 imaju sličnosti, to je zato što one prolaze kroz iste ekstremne vrijednosti, samo što se

trajektorija na slici 4 generiše u mnogo manjem broju koraka.



Slika 4. Trajektorija broja 649 sa međukorakom

Sada se zadatak svodi na provjeru hipoteze 2. Algoritam za to je prikazan u tabeli 1. Velike brojeve pamtimo u osnovi $2^{16} = 65536$. Cifre smještamo u niz, stim što na nultoj poziciji pamtimo broj cifara a cifru nulte težine pamtimo na poziciji jedan. Ovakva reprezentacija velikih brojeva je izabrana jer se direktno prevodi u binarnu reprezentaciju broja pa se djeljenje sa stepenom dvojke svodi na pomjeranje cifara (tj. bita) za odgovarajući broj mjesta. Razlika predloženog algoritma u odnosu na algoritme [6,7] je što dopušta rad sa velikim brojevima, dok su algoritmi u [6,7] optimizovani za rad sa brojevima koji mogu biti smješteni u dva 64-bitna registra. Poboljšanje u odnosu na algoritam u [6,7] je i u Hodd koraku. U njemu je više uzastopnih množenja sa trojkom a zatim pomjeranje (tj. djeljenje sa dva) realizovano kao jedno množenje sa odgovarajućim stepenom trojke i jedno pomjeranje.

Poslje iteracija za jedan broj prelazi se na sledeći. Brojevi Tabela 1. Algoritam za $3n+1$ problem

Algoritam. Za provjeru $3n+1$ hipoteze.
 Ulaz: Broj od koga počinjemo B , broj koraka K i broj za koji vršimo uvećanje S .
 Izlaz: Brojevi koji zahtjevaju puno iteracija ili iteracijama postaju suviše veliki (ako takvi brojevi postoje) i broj do koga smo stigli sa provjerom hipoteze.
 Metod:
 BEGIN
 $C=B$
 FOR $j=1$ TO K DO
 $A=B$
 $i=1$
 WHILE $A \geq C$ DO
 $i=i+1$
 Hodd(A)
 Heven(A)
 If ($i > 10^5$ ili A suviše veliki broj) THEN
 Prikaži(B)
 $A=1$
 END IF
 END WHILE
 $B=B+S$
 END FOR
 Prikaži(B)
 END BEGIN.

koji zahtijevaju veliki broj iteracija ili primjenom iteracija postaju suviše veliki (npr. broj cifara je trostruko veći od broja cifara polaznog broja) se daju kao izlaz i njih treba dodatno ispitati. Naime, ovi brojevi su kandidati da imaju periodičnu trajektoriju ili im trajektorija ima podniz koji teži ka beskonačnosti.

4. NUMERIČKI REZULTATI

Prema našim saznanjima hipoteza 1 je uz pomoć računara [6,7] provjerena za sve brojeve $n \leq 20 * 2^{58}$. Ne isključujemo mogućnost da je ova granica već pomjerena, ali da rezultati nisu publikovani ili da nama publikacija nije poznata. Da bi izbjegli moguća poklapanja ovoga rada sa rezultatima testiranja drugih autora, testiranja su sprovedena na velikim brojevima specijalnog oblika. Od posebnog interesa su stepeni broja 3 koji se pojavljuju u iteracijama $3n+1$ funkcije.

Algoritam opisan u prethodnom poglavlju je poslužio kao osnova za izgradnju nekoliko procedura u programskom jeziku C za razna numerička testiranja. U ovom poglavlju navodimo rezultate nekih od sprovedenih testiranja.

Provjera da li postoji periodična trajektorija. Polazni broj je bio $5 * 2^{60} + 1$, a korak 18 (vidi [1,7]). Sprovedeno je 2^{32} ciklusa (tj. do broja $5 * 2^{60} + 9 * 2^{33} + 1$) i nije pronađena periodična trajektorija.

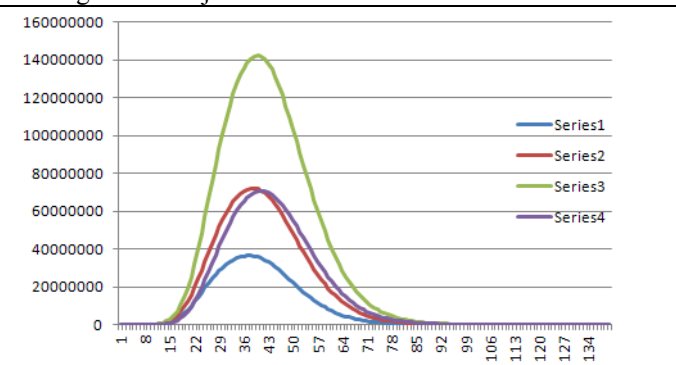
Testirana je hipoteza za brojeve oblika $3 + \frac{10}{3}(4^k - 1) \pmod{2^{2k+2}}$, za $k = 29$. Polazni broj je $3 + \frac{10}{3}(4^{29} - 1)$ a sprovedeno je 4294967296 ciklusa tj. do broja $3 + 2^{92} + \frac{10}{3}(4^{29} - 1)$. Za sve testirane brojeve važi $3n+1$ hipoteza.

Testirana je hipoteza, i ispostavilo se da važi, za stepene broja 3 zaključno sa brojem 3^{32768} . Napomenimo da se radi o velikim brojevima. Spomenuti broj ima 3247 cifre u osnovi 2^{16} što znači da taj broj imao više od 15500 dekadnih cifara. Kao jedna kontrola računa provjereno je i važi da je $3^{32768} \equiv -1 \pmod{2^{2^t} + 1}$ (Pepinov test za Fermaov broj 65537). Takođe, maksimalno totalno vrijeme zaustavljanja, pri ovom testiranju, za funkciju *Heven* o *Hodd* je iznosilo 62275.

Testirana je hipoteza, i ispostavilo se da važi, za brojeve oblika $3^k - 1$, zaključno sa brojem $3^{32768} - 1$. Pošto su brojevi oblika $3^k - 1$ parni, prije startovanja glavnog algoritma odrađen je korak *Heven*(3^k). Ovim testiranjem su obuhvaćeni i brojevi oblika $2^k - 1$ zbog svojstva $T^k(2^k - 1) = 3^k - 1$.

Testirana je i učestalost broja $\sigma_h(n)$ kada se uvećava cifra najveće težine polaznog broja n . Uočeno je da ta učestanost ima određeno zakonito ponašanje (nalik na normalnu raspodjelu), što nije slučaj sa $\sigma_\infty(n)$. Testiranja su sprovedena na brojevima oblika $3 + \frac{10}{3}(4^k - 1) \pmod{2^{2k+2}}$, za $k = 29$ (vidi sliku 5.) kao i brojevima $100 * [\pi * 10^{35}] + 1$ i $10^6 * [e * 10^{50}] + 1$. Cifre

brojeva π i e se sa jedne strane ponašaju kao slučajno generisani brojevi a sa druge strane lako se mogu upoređivati sa drugim testiranjima.



Slika 5. Učestalost broja $\sigma_h(n)$ za $2^{30}, 2^{31}, 2^{32}$ ciklusa i razlika učestalosti između 2^{32} i 2^{31} ciklusa.

5. ZAKLJUČAK

Uvedene su funkcije *Heven*, *Hodd* i predložen je proučavanje funkcije *Heven* o *Hodd* umjesto funkcije *T*, koja ima drugačija svojstva konvergencije. Svi rezultati testiranja idu u prilog da je hipoteza 1 tačna. Posebno je zanimljivo ponašanje učestalosti broja $\sigma_h(n)$ i u daljim istraživanjima treba dati matematičko objašnjenje za to.

LITERATURA

- [1] Joao F. Alves, Mario M. Graca, M. E. Sousa Dias, and Jose Sousa Ramos, *A linear algebra approach to the conjecture of Collatz*, Lin. Alg. Appl. 394 (2005), 277–289.
- [2] Paul S. Bruckman, *A proof of the Collatz conjecture*, International Journal of Mathematical Education in Science and Technology, 39, No. 3 (2008), 403–407. [Erratum: 39, No. 4 (2008), 567.]
- [3] Charles C. Cadogan, *A Solution to the $3x+1$ Problem*, Caribbean J. Math. Comp. Sci. 13 (2006), 1–11.
- [4] L. Collatz, *Letter to Michael E. Mays*, dated 17 Sept. 1980.
- [5] L. Collatz, *On the motivation and origin of the $(3n+1)$ -problem* (Chinese), J. Qufu Normal University, Natural Science Edition [Qufu shi fan da xue xue bao] 12 (1986), No. 3, 9–11.
- [6] T. Oliveira e Silva, *Maximum excursion and stopping time record-holders for the $3x+1$ problem: computational results*, Mathematics of Computation, 68(225), (1999), 371–384.
- [7] T. Oliveira e Silva, *Empirical verification of the $3x + 1$ and related conjectures*, The Ultimate Challenge: The $3x+1$ Problem (J. Lagarias, ed.), American Mathematical Society, Providence, RI, 2010, pp. 189–207.
- [8] Z.H. Yang, *An Equivalent Set for the $3x+1$ Conjecture*, Journal of South China Normal University, Natural Science Edition, no.2, (1998), 66–68.
- [9] <http://www.ericr.nl/wondrous/>

PLANIRANJE OBLASTI POKRIVENOSTI WLAN MREŽA COVERAGE PLANNING IN WLAN NETWORKS

Stevan Šćepanović, Marko Grebović, Prirodno-matematički fakultet Podgorica

Sadržaj: Tema kojom se bavi ovaj rad je jedan od najvažnijih aspekata kod planiranja WLAN mreža, a to je planiranje oblasti pokrivenosti. Biće riječi o metodološkom pristupu planiranja mreža i principu rada alata za planiranje pokrivenosti. Posebna pažnja je posvećena testovima mjerenja brzine prenosa primjenom alata Iperf. Takođe, rad se dotiče i problema odabira kanala za access point-e u okviru iste mreže. U radu se takođe govori o planiranju bezbjednosti mreže i testiranju oblasti pokrivenosti i optimizaciji lokacija access point-a i mrežnih parametara. Cijela priča je zaokružena konkretnim primjerom.

Abstract: Main topic that is addressed in this paper is one of the most important aspects in planning WLAN networks, and that is planning of the coverage area. Here will be discussed methodological approach of network planning and operating principle of coverage planning tools operating principle. Special attention is devoted to tests measuring data transfer rate using tool called Iperf. Also, the paper touches upon the problem of selecting channels of access points in the same network. The paper also discusses the planning of network security, testing of coverage area and the optimization of access points locations and network parameters. The whole story is completed with a concrete example.

1. UVOD

Sa porastom broja korisnika prenosivih računara, stvorila se potreba za izgradnjom odgovarajuće mrežne infrastrukture koja bi podržala njihovo brže i praktičnije povezivanje na lokalnu mrežu, bez korišćenja skupe i nepraktične žičane infrastrukture. Bilo je neophodno osmisliti sistem, koji će omogućiti korisnicima da se lako povežu na novu lokalnu mrežu kako u poslovne, tako i u privatne svrhe. Takav sistem je morao da isključi bilo kakva komplikovana softverska podešavanja. Tačnije, morao je da omogući korisnicima da učestala promjena lokacije i povezivanje na druge mreže ne predstavlja nikakav problem.

Međutim svi ti zahtjevi i potrebe su otvorile nova pitanja i probleme koje je bilo potrebno riješiti. Prvi je brzina prenosa. Sa većim brojem korisnika, brzina prenosa srazmjerno opada, pa je neophodno ograničiti broj korisnika koji istovremeno mogu pristupiti mreži. Drugi problem je taj što fizičke prepreke ometaju rasprostiranje signala, tako da je neophodno voditi računa o poziciji bazne stanice, tj. pristupne tačke. Takođe, kao veliki problem javlja se pitanje sigurnosti, kako privatnosti pojedinih korisnika, tako i dijeljenih resursa koji bi trebalo da budu dostupni samo određenom krugu ljudi.

Kod planiranja i projektovanja bežičnih lokalnih računarskih mreža, neophodno je povesti računa o ovim pitanjima, što je i tema ovog rada.

2. POKRIVENOST SIGNALOM

Faktori koji se moraju uzeti u obzir u fazi planiranja mreže su oblast pokrivenosti, kapacitet i troškovi. Jedini način da se obezbijede visok stepen pokrivenosti i kapacitet jeste izdvajanje većih novčanih sredstava, tj. veći troškovi.

Pritom, treba imati u vidu da što je manja snaga prenosa signala, to je manja oblast pokrivenosti. Sa druge strane, to dovodi do većeg ukupnog kapaciteta s obzirom da su access point-i bliže jedan drugom.

Ukoliko su sredstva predviđena za realizaciju WLAN mreže mala, pametno je najprije pokriti signalom sale za predavanja, konferencijske sale, hodnike i druga mjesta gdje se očekuje da korisnici pristupaju mreži preko prenosivih uređaja (mobilnih telefona, tableta, laptopova, itd.). Drugim riječima, možemo isključiti kancelarije iz oblasti koju pokrivamo signalom, jer je u njima moguće povezivanje na mrežu putem žičanih medijuma.

Takođe, treba imati u vidu kod planiranja oblasti pokrivenosti, o kom se frekvencijskom opsegu radu. Veći frekvencijski opseg donosi veću prenosnu moć, ali i veći gubitak signala sa povećanjem rastojanja i broja fizičkih prepreka.

Postoji nekoliko načina da se izvrši planiranje oblasti pokrivenosti od kojih su najznačajniji metodološki pristup i planiranje zasnovano na testiranju, odnosno mjerenju i optimizaciji snage signala u realnoj mreži. Najbolji rezultati se postižu kombinacijom ove dvije metode.

3. METODOLOŠKI PRISTUP PLANIRANJA POKRIVENOSTI

Kod metodološkog pristupa, prva faza predstavlja procjenu oblasti pokrivenosti korišćenjem softvera za izračunavanje snage signala na određenim lokacijama na osnovu parametara access point-a i organizacije prostora u građevini u koju se uvodi WLAN. Ulazni parametri za računanje uključuju brzinu prenosa podataka access point-a kao i debljinu i materijal od kog su napravljeni zidovi.

Ukoliko detaljne informacije o građevini nisu dostupne, mogu se primjenjivati druge metode kako bi se dobile grube procjene.

Alat koji se može koristiti za planiranje mreže tj. neki od postojećih Site Survey programa: Ekahau Site Survey, Fluke Networks Airmagnet Survey, VisiWave Site Survey i drugi. Prije nego što se izvrše izračunavanja, vrši se učitavanje ili ucrtavanje plana zgrade zajedno sa podacima o materijalima koji su korišćeni u izgradnji. Određuje se nekoliko potencijalnih lokacija za access point-e. Nakon toga, alat predviđa oblasti pokrivenosti i optimizuje lokacije access point-a, kako bi se kako bi se dobila maksimalna oblast pokrivenosti.

4. MJERENJE POKRIVENOSTI KORIŠĆENJEM SOFTVERSKOG ALATA IPERF

Kada govorimo o planiranju oblasti pokrivenosti u manjim zgradama, to nije tako obiman proces kao što je prethodno opisano. U slučaju da se ne može primijeniti odgovorajući softver za proračun oblasti pokrivenosti, savjetuje se da se obezbijedi nekoliko potencijalnih access point-a za testiranje. *Access point*-i bi trebalo da budu postavljeni na lokacijama koje su predviđene kao pogodne za tu namjenu: npr. hodnik. Oblast pokrivenosti *access point*-a bi trebalo izmjeriti provjeravanjem brzine prenosa, na različitim udaljenostima i u različitim situacijama, umjesto da se vodi računa samo o jačini signala. Brzina prenosa je često bolji pokazatelj kvaliteta mreže od jačine signala, iz ugla korisnika. Kako bi se bar približno dobio uvid u oblast pokrivenosti *access point*-a, trebalo bi izmjeriti brzinu prenosa u sljedećim tačkama:

- Blizu *access point*-a npr. direktno ispod njega (to je tačka A)
- Blizu *access point*-a na istom spratu
 - iza prvog skretanja u hodniku (tačka B1)
 - iza zida koji razdvaja od *access point*-a (tačka B2)
 - iza sljedećeg zida koji razdvaja od *access point*-a (tačka B3)
- Direktno iznad *access point*-a na spratu iznad (tačka C) ili direktno ispod *access point*-a na spratu ispod (tačka D).

Ukoliko je zgrada relativno homogena, ovi testovi mogu biti dovoljni da se obezbijedi slika o oblasti pokrivenosti jednog *access point*-a. Ukoliko se zgrada sastoji iz raznovrsnih struktura, potrebna su dodatna mjerenja. Na primjer, savjetuje se mjerenje uticaja protivpožarnih vrata na vrijednosti brzine prenosa.

Mjerenje brzine prenosa se može vršiti korišćenjem regularnih terminala. Ukoliko je to moguće, preporučljivo je da se mjerenje vrši sa više različitih mrežnih kartica, najmanje u jednoj tački mjerenja. Na taj način se dobija slika o kompatibilnosti *access point*-a sa različitim mrežnim karticama.

Mjerenje brzine prenosa se može izvršiti pomoću softvera Iperf. On je dostupan za *Windows*, *Linux* i *Mac* operativne sisteme. Testiranje zahtijeva dva računara: jedan od računara je povezan na isti switch kao i *access point*, dok se drugi koristi da izmjeri performanse *access point*-a.

Testiranje se vrši tako što se oba računara povežu na switch i prati se protok između njih. Najprije bi trebalo provjeriti brzinu TCP konekcije. Da bi se to uradilo, kod oba računara treba ući u direktorijum gdje je Iperf instaliran i pokrenuti sljedeće komande:

```
# Pokrenuti jedan računar kao server
>iperf -s -w 1M -i 1
```

```
# Pokrenuti drugi računar kao klijent
>iperf -c <IP adresa servera> -w 1M -l 64K -t 30 -i 1
```

Na ovaj način se provjerava konekcija pomoću TCP-a veličine prozora od 1 Mb (to bi trebalo da je dovoljno) i pakete veličine 64 Kb (maksimalna veličina TCP paketa, dostiže se u mrežama visokog kvaliteta). Mjerenja bi trebalo vršiti 30 sekundi, sa prikazivanjem rezultata na ekranu u svakoj sekundi.

Sljedeće što treba provjeriti jeste UDP konekcija:

```
# Pokrenuti jedan računar kao server
>iperf -s -u -w 1M -i 1 -l 1470
```

```
# Pokrenuti jedan računar kao klijent
>iperf -c <IP adresa servera> -u -l 1470 -b 300M -i 1 -t 30 -w 1M
```

Veličina paketa (1470 bajta) je maksimalna veličina koja može biti poslata bez dijeljenja. Opseg prenosa je npr. 300Mbps, što je teorijska maksimalna brzina prenosa koja se može dostići kod WLAN mreža.

Opisani test bi trebalo izvesti za sve prethodno pomenute tačke (A, B1, B2, B3, C ili D) sa provjeravanjem dobijenih podataka od strane *access point*-a ka klijentu i obrnuto. U oba slučajeva, jedan od računara mora biti povezan na isti switch kao i *access point*. Drugi računar mora biti povezan na WLAN mrežu preko *access point*-a. Kada se vrši testiranje od strane *access point*-a ka klijentu, računar povezan na WLAN bi trebalo pokrenuti kao server, a računar povezan na switch kao klijent. U slučaju testiranja u suprotnom smjeru uloge su zamijenjene.

Ukoliko je moguće, testove treba izvršiti primjenom oba protokola (TCP i UDP) kako bi se ustanovio uticaj protokola na brzinu prenosa, bar u tačkama gdje se predviđaju najveća i najmanja brzina prenosa. U drugim tačkama, dovoljno je izvršiti mjerenja preko UDP protokola.

5. RASPORED ACCESS POINT-A I ODABIR KANALA

Nakon izvođenja testova, može se procijeniti koliko gusto treba rasporediti *access point*-e, kako bi se obezbijedila odgovarajuća oblast pokrivenosti. Gornja granica brzine prenosa se može odrediti u zavisnosti od oblasti pokrivenosti

i raspoloživog budžeta. Međutim, rezultate ispod 10 Mbps ne bi trebalo razmatrati u fazi planiranja. Treba imati u vidu da brzina prenosa u najgorem slučaju može biti ispod tog nivoa. Ukoliko budžet to dozvoljava, u tom slučaju je preporučljiv gušći raspored access point-a.

U praksi, lokacije switch-eva, utičnica, i vidljivost access point-a treba uzeti u obzir pri raspoređivanju. Gdje je to moguće, preporučuje se sistematsko postavljanje *access point-a* dok cijela zgrada ne bude pokrivena. Kako bi interferencija između *access point-a* bila na što manjem nivou, neophodno je uzeti u obzir sljedeće faktore:

1. Ukoliko se *access point-i* nalaze jedan drugom u vidokrugu (u smislu oblasti pokrivenosti), može doći do tolikog stepena interferencije koji će dovesti smanjenja u prenosu snage u okviru mreže. Najbolje rješenje je da se *access point-i* postave unutar prostorije na regularnim razmacima duž hodnika sa jedne strane na drugu, naizmjenično.

2. Za zgrade sa više spratova, planiranje oblasti pokrivenosti se vrši na sljedeći način: kada se odrede lokacije *access point-a* na jednom spratu, lokacije *access point-a* na spratu direktno ispod i na spratu direktno iznad njega ne smiju biti na istim mjestima. Na taj način se eliminiše interferencija signala na susjednim spratovima.

Sem planiranja lokacije *access point-a*, potrebno je voditi računa o njegovom konfigurisanju, prije svega o odabiru kanala. Kod većine WLAN mreža prenos signala se vrši na frekvenciji oko 2,4 GHz. Međutim, neki elektronski uređaji rade na sličnoj frekvenciji i zbog toga može doći do interferencije i slabljenja performansi mreže. Sem, toga, iz istog razloga može da dođe do interferencije između *access point-a* koji rade na istoj frekvenciji i čije se oblasti pokrivenosti poklapaju (makar jednim dijelom). Oblast WLAN signala je podijeljen na manje jedinice - kanale. Frekventni opseg od 2,4 GHz se dijeli na 11-14 kanala u zavisnosti od kontinenta. Različite kanale odlikuju male varijacije u frekvenciji, što je dovoljno da ne dođe do interferencije između uređaja koji rade na različitim kanalima. Kod WLAN mreža, važno je da *access point-i* čije se oblasti pokrivenosti bar malo preklapaju koriste različite kanale.

6. PLANIRANJE BEZBJEDNOSTI

Osnovna ideja svake računarske mreže je razmjena informacija između korisnika na određenom rastojanju, posredstvom dijeljenog medijuma. To znači da ukoliko neko želi da pristupi dijeljenim informacijama ili resursima, ne mora da fizički pristupi serveru ili korisničkim računarima. Kod WLAN mreža ova prijetnja je posebno izražena, s obzirom na to da haker ne mora da bude na istoj fizičkoj lokaciji.

Kada govorimo o sigurnosti u WLAN mrežama, moramo obratiti pažnju na tri ključne oblasti:

- Tajnost i integritet podataka.
- Autentifikaciju i kontrolu pristupa

- Otkrivanje i sprečavanje neovlašćenog pristupa

Kako bi se obezbijedila tajnost podataka ispriječio neovlašćeni pristup, danas se gotovo svuda primjenjuje WPA/WPA2 AES standard. Ovaj standard je nastao kao rezultat poboljšanja prethodnika, WEP standarda sa RC4 enkripcijom i statički konfigurisanim ključevima. Najprije, standard WPA donosi uvođenje TKIP protokola koji omogućava redovno obnavljanje enkripcije. Kako je TKIP protokol još uvijek koristio RC4 enkripciju, uveden je standard WPA2 sa AES algoritmom enkripcije i objedinjen sa WPA u jedinstveni WPA/WPA2 AES standard.

Kada govorimo o autentifikaciji i kontroli pristupa, kod WEP standarda koristi se ACL koji dozvoljava samo poznatim MAC adresama da pristupaju mreži. Glavni problem je složenost realizacije u slučaju velike mreže. Standard WPA/WPA2 koristi 802.1x autentifikaciju koja radi na principu kapije koja sprečava klijenta koji je povezan na access point da pristupi mreži sve dok ne dobije dozvolu. To omogućava EAP protokol za dogovaranje i potvrdu ključeva.

Za rješavanje pitanja tajnosti i integriteta podataka i autentifikaciju i kontrolu pristupa, preporučuje se primjena WPA/WPA2 AES standarda sa 802.1x autentifikacijom.

Za dio bezbjednosti koji se tiče otkrivanja i prevenciju upada ne postoji standard koji se bavi tim pitanjima, već prodavci WLAN opreme nude rješenja koja su osnovi slična. Takav sistem mora da ima: daljinske senzore raspoređene kroz mreže, upravljački softver koji se naziva IDS server. Senzori za otkrivanje upada prate aktivnosti u mreži i mrežnoj konfiguraciji i obavještavaju IDS server o bilo kakvim izuzecima. IDS server analizira dobijene informacije i generiše alarme upada i zapis najskorijih događaja. Zatim, dio sistema za prevenciju upada reaguje direktno, bez potrebe za ručnom intervencijom, slanjem komandi razdvajanja klijentu čime se on prekida vezu sa otkrivenom prijetnjom.

7. TESTIRANJE MREŽE I OPTIMIZACIJA

Kada govorimo o testiranju i optimizaciji WLAN mreže, najvažniji su sljedeći koraci:

- Testiranje oblasti pokrivenosti
- Optimizacija lokacija access point-a
- Optimizacija mrežnih parametara

Korišćenjem alata za planiranje mreže, na laptopu se mogu izmjeriti jačinu signala i interferencije: na osnovu tih parametara računa se brzina prenosa. Sveukupna slika o oblasti pokrivenosti se može dobiti korišćenjem *ping* komande ili provjerom jačine signala tokom pomjeranja položaja terminalnog uređaja (npr. laptopa). Kako se WLAN mreže podešavaju u okviru granica slobodne frekvencije, interferencija može biti prilično visoka na pojedinim mjestima. Za testiranje interferencije može se koristiti npr. *Wi - Spy*.

Jedan način da se izvrši optimizacija lokacija *access point-a* je da se vrši provjera snage signala koji se odašilje. Na primjer, u slučaju niže snage kod *access point-a* koji se nalaze jedna blizu druge, najbolje je udaljiti ih međusobno.

Idealan scenario je da svi *access point*-i imaju istu snagu prenosa signala. Ukoliko se utvrdi da pokrivenost nije dovoljna ili da je neophodno povećanje kapaciteta na pojedinim mjestima, kod mreža zasnovanih na kontroli dodavanje novih *access point*-a je jednostavno, s obzirom na to da se snaga prenosa i kanal biraju u centralizovanoj kontrolnoj jedinici. Treba imati u vidu da *access point*-i moraju biti raspoređeni sa određenim razmakom. Na primjer, ukoliko je dodatni kapacitet neophodan u nekoj prostoriji, preporučuje se postavljanje dodatnih *access point*-a na suprotnom kraju prostorije, umjesto pored postojećeg *access point*-a. Možemo da zaključimo da u mrežama zasnovanim na kontroli poboljšavamo kvalitet povećanjem snage prenosa u obližnjim *access point*-ima u slučaju otkaza nekog od njih.

WLAN donosi više mogućnosti kada je u pitanju podešavanje parametara. Prema podrazumijevanim parametrima standarda, paketi se dijele ukoliko njihova veličina prelazi 2346 bajta. Drugi odašiljači su utišani primjenom RTS/CTS (*Request To Send/Clear To Send*) mehanizma rukovanja prije prenosa, ukoliko je paket veličine 2347 bajta ili veći, tj. samo ako se pređe granica za dijeljenje paketa. Takođe, mogu se podešavati neki parametri koji utiču na korišćenje resursa i utrošak energije, kao što su interval zračenja, ograničenja u broju pokušaja ponovnog povezivanja i interval oslušivanja. Iako parametri mogu biti podešeni na različite načine, savjetuje se oprez prilikom podešavanja, jer posljedice mogu biti ozbiljne. Takođe, kontroleri obično posjeduju mogućnost upravljanja resursima za emitovanje signala. Kao i kod drugih mreža, WLAN se koristi za servise koji imaju određene zahtjeve. VoIP pozivi zahtijevaju kratko odlaganje i određen nivo propusni opseg. Neki kontroleri omogućavaju da se ručno definiše brzina prenosa kao i propusni opseg. To je naročito važno za već pomenute VoIP pozive, jer se na taj način omogućava bolja mrežna podrška za taj servis. Međutim, ono što treba imati u vidu je da se na taj način ne postižu željene performanse u slučaju većeg broja korisnika, tako da treba voditi računa da se napravi balans.

8. PRIMJER PLANIRANJA OBLASTI POKRIVENOSTI U WLAN MREŽI

Primjer koji ćemo razmatrati je prostor veličine 13200 m². Za pokrivanje prostora koristimo *access point*-e *Cisco Aironet 1240AG 802.11 a/b/g/n*. Ovaj model koristimo zbog podrške glavnim 802.11 standardima. Ukoliko želimo da postignemo brzinu prenosa podataka od najmanje 24 Mbps, treba planirati da oblast pokrivenosti jednog *access point*-a ne bude veća od 350 m².

Kako se signal rasprostire u kružnom obliku, oblast pokrivenosti mora biti površina maksimalnog kvadrata upisanog u krug. To znači da poluprečnik kruga koji pokriva jedan *access point* mora biti: $(2R)^2 = 2a^2 \Rightarrow 4R^2 = 2a^2 \Rightarrow 2R^2 = a^2 \Rightarrow R^2 = \frac{a^2}{2} \Rightarrow R = \sqrt{\frac{a^2}{2}}$. Rekli smo da je željena oblast pokrivenosti (površina kvadrata) 350 m², tj. $a^2 = 350$. Tako dobijamo da je: $R = \sqrt{\frac{350}{2}} = \sqrt{175} = 13,23m$, tj. površina kruga je: $R^2\pi \approx 550 m^2$.

Aktivne komponente u WLAN mreži omogućavaju povezivanje korisnika na mrežu bežičnim putem. U ove komponente spadaju: ruter, svič i pristupne tačke. Ako je oblast koju moramo pokriti signalom 13200 m², oblast pokrivenosti jednog *access point*-a 350 m², znači da je potrebno $13200/350 \approx 38$ *access point*-a. Njihov optimalan raspored dat na slici 1.



Slika 1: Plan objekta sa ucrtanim pozicijama *access point*-a

9. ZAKLJUČAK

Danas su bežične lokalne računarske mreže dostupne svuda oko nas: kod kuće, na fakultetu, poslu, na željezničkim stanicama, aerodromima, kaficima. Međutim, uvijek je aktuelno pitanje proširenja postojećih i planiranja i projektovanja novih WLAN mreža.

U posljednjih nekoliko godina WLAN je postala dominantna među mrežama, zbog ekspanzije mobilnih uređaja, prije svega tablet i mobilnih telefona. Taj trend će se, bez sumnje, nastaviti i dalje. Zbog toga, prilikom planiranja mreže, moramo uvijek imati u vidu da ta mreža bude skalabilna tj. da postoji određena mogućnost njenog proširenja bez da konfor postojećih korisnika bude narušen.

Kada govorimo o realizaciji plana WLAN mreže, možemo uočiti da veliki budžet nije garancija da će on biti uspješan. Ukoliko se ne ispoštuju neke osnovne ideje koje se tiču rasporeda pristupnih tačaka, sigurnosti mreže, sprečavanja interferencije, ni posjedovanje najbolje WLAN opreme ne garantuje mrežu visokog kvaliteta. Sa jednakom pažnjom treba pristupati i u slučaju proširivanja mreže ili uvođenja nove infrastrukture, jer i u tom slučaju, ključnu ulogu igra planski pristup.

LITERATURA

- [1] Computer Networks, 5th Edition - Andrew S. Tanenbaum, 2010.
- [2] WLAN Network Planning and Setup - Wenche Backman, 2010.
- [3] WLAN Information Security - Wenche Backman, 2010.
- [4] WLAN Security Today: Wireless more Secured than Wired - Siemens Enterprise Communications
- [5] WLAN Security: Best practices for wireless network security - Lisa Phifer
- [6] http://www.cisco.com/en/US/prod/collateral/wireless/ps5678/ps6521/product_data_sheet0900aecd8031c844.html
- [7] <http://compnetworking.about.com/od/wifihomenetworking/qt/wifichannel.htm>

POREĐENJE ALGORITAMA ZA CS REKONSTRUKCIJU SLIKE COMPARISON OF THE ALGORITHMS FOR CS IMAGE RECONSTRUCTION

Milica Medenica, Sanja Zuković, Andjela Draganić, Irena Orović, Srdjan Stanković, *Elektrotehnički fakultet*

Sadržaj: U ovom radu je prikazano poređenje algoritama za rekonstrukciju kompresivno odabranih 2D signala. Kompresivno odabiranje je novi pristup očitavanju signala koji ima za cilj da smanji potrebu za resursima u realnim digitalnim sistemima (broj senzora, memoriju, i slično). Ovaj metod omogućava analizu signala korišćenjem nepotpunog seta slučajno uzetih odbiraka. Korišćenjem veoma kompleksnih matematičkih algoritama omogućena je rekonstrukcija kompresivno odabranih signala korišćenjem sasvim malog skupa slučajnih odbiraka, pod uslovom da signal zadovoljava neke unaprijed definisane uslove. U radu je predstavljeno poređenje performansi tri algoritma za CS rekonstrukciju slike. Kvalitet rekonstrukcije je dat odnosom signal-šum, a mjereno je i vrijeme izvršavanja opisanih algoritama.

Abstract: This paper presents comparison of algorithms for compressive sampling reconstruction of 2D signals. Compressive sensing is a new signal sensing approach aiming to decrease the requirements for resources in real digital systems (number of sensors, memory requirements, etc.). This method provides signal analysis, using small set of randomly chosen samples. Based on very complex mathematical algorithms it is possible to reconstruct the compressive sensed signals using just a small number of randomly chosen samples, under the assumption that certain conditions are satisfied. Performances of three algorithms for CS image reconstruction are presented in the paper. Reconstruction quality is measured with signal to noise ratio, and execution time of the described algorithms is given, as well.

1. UVOD

Standardni pristup obradi signala zasniva se na odabiranju signala u skladu sa Shannon-Nyquist-ovom teoremom o odabiranju. Ova teorema zahtijeva da se signal odabira sa frekvencijom koja je barem dva puta veća od najveće frekvencije u signalu, ukoliko želimo rekonstrukciju signala sa velikom tačnošću. Odabiranje signala na ovaj način zahtijeva značajne resurse za smještanje i prenos podataka, pa se javlja potreba za kompresijama signala. Posljednjih godina intenzivno se razvijaju alternativni načini za odabiranje signala, zasnovani na metodu poznatom pod nazivom Compressive Sensing (CS) [1]-[6]. Metoda CS je nova metoda snimanja i predstavljanja signala uz odabiranje koje je znatno ispod Nyquistove frekvencije. Ova metoda omogućava rekonstrukciju signala sa velikom tačnošću, uprkos smanjenju broja odbiraka, ukoliko signal zadovoljava unaprijed definisane uslove. Naime, signal mora imati rijetku predstavu u nekom domenu (Fourier-ovom, wavelet domenu, domenu diskretne kosinusne transformacije, vremenskom domenu i slično). Drugi uslov koji CS zahtijeva je da procedura mjerenja bude definisana na način na koji će se iz malog broja uzetih odbiraka signala omogućiti njegova uspješna rekonstrukcija. Da bi se signal uspješno rekonstruisao iz malog broja uzetih odbiraka, neophodno je koristiti moćne matematičke aparate (tzv. optimizacione algoritme). Opisani pristup našao je primjenu u mnogim poljima obrade signala i kodiranja.

U ovom radu analizirana je mogućnost primjene CS metoda za rekonstrukciju 2D signala [7]-[9]. Posmatrane su slike koje se javljaju u svakodnevnom životu, kao i neke medicinske slike. Važno je napomenuti da većina slika nema rijetku predstavu niti u prostornom niti u domenu frekvencija.

Kako je rijetkost signala u nekom domenu neophodan uslov da bi se CS metod mogao primijeniti, razvijeni su alternativni algoritmi koji omogućavaju CS rekonstrukciju signala koji nijesu idealno rijetki. Rekonstrukcija 2D signala koristi metod totalne varijacije [1], 0 u cilju dobijanja gradijenta koji će imati rijetku predstavu u nekom transformacionom domenu.

U radu su opisana tri algoritma za rekonstrukciju slike. Data je analiza kvaliteta rekonstrukcije u zavisnosti od broja odbiraka koji se uzimaju za rekonstrukciju, a mjereno je i vrijeme izvršavanja pojedinih algoritama.

Rad je podijeljen u 5 sekcija. U drugoj sekciji je dat teorijski osvrt na intenzivno razvijani CS metod za obradu signala, objašnjena su svojstva na koja se oslanja pomenuta metoda i opisan je postupak uzimanja odbiraka. U trećoj sekciji dat je osvrt na korišćene algoritme za rekonstrukciju slike, dok su eksperimentalni rezultati opisanih algoritama prikazani u četvrtoj sekciji. U petoj sekciji dati su zaključci koji se odnose na mogućnost primjene opisanih algoritama u zavisnosti od vrste slike, željenog kvaliteta rekonstrukcije kao i vremena izvršavanja.

2. COMPRESSIVE SENSING

Realni, konačni, jednodimenzionalni signal x moguće je prikazati kao vektor kolonu dimenzija $N \times 1$ u prostoru \mathbf{R}^N (gdje je N vektor realnih brojeva). Takav signal može se predstaviti u obliku [1]-[5]:

$$x = \sum_{i=1}^N s_i \psi_i = \Psi s \quad (1)$$

gdje je Ψ matrica transformacionog domena, čije su dimenzije $N \times N$, a s vektor transformacionih koeficijenata.

Vektori x i s su ekvivalentne reprezentacije istog signala u vremenskom (prostornom) i Ψ domenu.

Da bi se CS metod mogao primijeniti na neki signal, sljedeći uslovi treba da budu zadovoljeni: signal treba da je rijedak, što podrazumijeva da se u nekom domenu može predstaviti malim brojem nenultih koeficijenata. Upravo taj mali broj koeficijenata sadrži dovoljno informacija o signalu, pa se izbjegava potreba za kompresijom. Drugi potreban uslov za primjenu CS metoda je inkoherenca. Inkoherenca se odnosi na proceduru mjerenja, i ukoliko je zadovoljena, omogućava uspješnu rekonstrukciju signala iz malog broja uzetih odbiraka. Signali u velikom broju aplikacija zadovoljavaju ova svojstva, pa je oblast primjene CS-a široka.

Odabiranje signala (prikupljanje odbiraka signala) u CS proceduri naziva se mjerenje signala. Obično je broj mjerenja M koja su potrebna za rekonstrukciju signala mnogo manji od ukupnog broja odbiraka signala N , tj. važi $M \ll N$. Niz ovakvih mjerenja y dobija se množenjem matrice mjerenja Φ sa vektorom x , što se može opisati formulom [1], 0, 0:

$$y = \phi x. \quad (2)$$

Iz formula (1) i (2) y slijedi:

$$y = \phi x = \phi \psi s = \theta s. \quad (3)$$

Matrica $\theta = \phi \psi$ predstavlja CS matricu dimenzija $M \times N$. Od mjerenja y potrebno je rekonstruisati signal. Dok je proces uzimanja odbiraka vrlo jednostavan, proces rekonstrukcije signala uključuje složene proračune. Naime, sistem (3) je neodređeni sistem jednačina i u cilju dobijanja jedinstvenog rješenja ovog sistema, koriste se složeni optimizacioni algoritmi. Često korišćeni optimizacioni metod je l_1 minimizacija:

$$s = \min \|s\|_{l_1} \text{ odgovara } y = \theta s \quad (4)$$

3. ALGORITMI ZA REKONSTRUKCIJU SLIKE

Digitalne kamere snimaju ogromnu količinu informacije, a zatim vrše kompresiju slike. Kompresija podrazumijeva odbacivanje velikog broja već prikupljenih podataka, tj. odbiraka koji imaju nulte ili vrijednosti koje su bliske nuli. CS metod omogućava uštedu energije i memorije, tako što prilikom prikupljanja podataka snima manju količinu istih. Pomenuli smo da se za rekonstrukciju signala od prikupljenih odbiraka koriste složeni algoritmi, od kojih je najčešće korišćena l_1 minimizacija. Međutim, kako slika nema striktno rijetku reprezentaciju ni u jednom transformacionom domenu kvalitet rješenja dobijenog l_1 minimizacijom nije zadovoljavajući. Zbog toga se za rekonstrukciju 2D signala obično koristi metoda minimizacije totalne varijacije (TV) [7]-0. TV signala s predstavlja sumu amplituda gradijenta, u svakoj tački (i, j) i može se opisati formulom:

$$\|s\|_{TV} = \sum_{i,j} |(\nabla s)_{ij}|, \quad (4)$$

gdje ∇ predstavlja operator diferenciranja, tj. aproksimaciju gradijenta za piksel na poziciji (i, j) i opisan je formulom:

$$\nabla_{i,j} s = \begin{bmatrix} s(i+1, j) - s(i, j) \\ s(i, j+1) - s(i, j) \end{bmatrix} \quad (5)$$

Diskretna forma TV-a se može opisati kao:

$$TV(s) = \sum_{i,j} \sqrt{(s_{i+1,j} - s_{i,j})^2 + (s_{i,j+1} - s_{i,j})^2}, \quad (6)$$

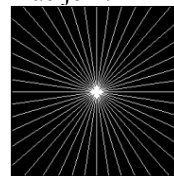
dok se minimizacioni problem korišćenjem TV-a može predstaviti u sljedećem obliku:

$$\min_s TV(s) \text{ odgovara } y = \theta s \quad (7)$$

TV minimizacija omogućava pouzdanu rekonstrukciju signala, posebno daje dobre rezultate u uslovima kada je prisutan šum. U nastavku su opisana tri algoritma, korišćena za rekonstrukciju 2D signala u radu. Sva tri opisana algoritma uzimaju mjerenja iz domena frekvencija, a rješenje traže korišćenjem TV minimizacije.

Algoritam 1:

Ovaj algoritam uzima odбирke iz domena 2D Fourier-ove transformacije, korišćenjem maske u obliku zvijezde (Slika 1) 0. U maski je moguće podesiti koliko ima linija, a samim tim, i koliko se koeficijenata 2D FT uzima u procesu mjerenja. Vektor mjerenja se dobija na osnovu linija koje pripadaju 2D FT maski, a slika se od uzetih koeficijenata rekonstruiše TV minimizacijom.



Slika 1: Maska za odabiranje

Algoritam 2:

U drugom algoritmu koji je posmatran u radu, odbirci signala se uzimaju iz domena 2D diskretne kosinusne transformacije (2D DCT), i to slučajnim putem. Slika se dijeli na blokove različitih dimenzija i iz svakog bloka se slučajnim putem uzima određeni broj koeficijenata koji će služiti kao vektor mjerenja u CS proceduri [1]. Slika se rekonstruiše blok po blok. U zavisnosti od veličine bloka kao i procenta koeficijenata koji se iz bloka uzimaju, kvalitet rekonstruisane slike će biti različit.

Algoritam 3:

U trećem algoritmu, mjerenja signala se uzimaju iz 2D DCT domena, ali tako da se uzima određeni procenat niskofrekvencijskih (NF) koeficijenata i određeni procenat srednjih (SF) ili visokih frekvencija (VF) slike [7], 0. Slika se pretvara u vektor cik-cak preuređenjem i iz tako dobijenog vektora se uzima $K1$ NF koeficijenata, i $K2$ slučajno odabranih SF ili VF koeficijenata. Vektor mjerenja je opisan relacijom $y = y_1 + y_2$, gdje y_1 predstavlja vektor NF koeficijenata, a y_2 vektor SF ili VF koeficijenata slike. Od uzetih mjerenja slika se rekonstruiše korišćenjem relacije (7), tj. TV minimizacijom.

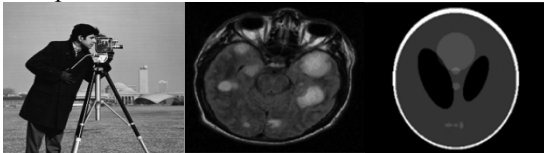
Kvalitet rekonstrukcije kod svih posmatranih algoritama se mjeri odnosom signal-šum, tj. PSNR-om (peak signal to noise ratio), koji je definisan formulom:

$$PSNR = 20 \log_{10} \frac{O_{\max}}{\sqrt{\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [S_{orig}(i, j) - S_{rek}(i, j)]^2}}, \quad (8)$$

gdje je O_{max} maksimalna vrijednost osvjetljava u slici, M i N su dimenzije slike, a S_{orig} i S_{rek} predstavljaju originalnu i rekonstruisanu sliku.

4. EKSPERIMENTALNI REZULTATI

U narednom dijelu dati su eksperimentalni rezultati rekonstrukcije prethodno navedenim algoritmima. Posmatrane su slike „Cameraman“ i medicinske slike „Brain“ i „Phantom“. Slike su veličine 256 x 256 i originalne verzije ovih slika prikazane su na Slici 2.



Slika 2: Originalne slike *Cameraman*, *MRI Brain* i *Phantom*

A. Rekonstrukcija slike Algoritmom 1

Tabela 1: Rezultati simulacija za Algoritam 1

slika	broj linija L	vrijeme (s)	PSNR (dB)
Phantom	100	163,94	121,48
Phantom	22	577,69	48.8031
Phantom	15	578,28	34.4722
Phantom	10	505,45	20.1436
Cameraman	100	1078,8	33.2119
Cameraman	40	1025,2	25.8856
Cameraman	22	1123,2	22.6849
Cameraman	10	1102,9	19,8564

Mjerenja se uzimaju iz DFT domena, na način definisan DFT maskom. Iz DFT maske (Slika 1) uzimaju se koeficijenti sa radijalnih linija. Broj linija u DFT maski je korisnički definisan. U simulacijama je vršena promjena broja linija L u maski. Na Slici 3 prikazane su slike rekonstruisane korišćenjem podataka iz Tabele 2 (za L=10, L=22, L=100, respektivno).



Slika 3: Rekonstrukcija slike *Phantom* (gore) i *Cameraman* (dolje) za različit broj linija u DFT domenu

Za rekonstrukciju slike *Cameraman* neophodno je uzeti višestruko veći broj linija DFT maski (L=100), kako bi se dobio zadovoljavajući kvalitet rekonstruisane slike (PSNR>30dB). Za slike, poput *Phantom*-a, dovoljno je uzeti mali broj linija L iz Furijeovog domena kako bi se dobio zadovoljavajući kvalitet rekonstruisane slike (L=22).

B. Rekonstrukcija slike Algoritmom 2

U narednom dijelu pokazano je da je moguće dobiti rekonstruisanu sliku zadovoljavajućeg kvaliteta, uzimanjem odbiraka iz DCT domena i rješavanjem TV minimizacionog

problema. U cilju poboljšanja kvaliteta rekonstrukcije, slika je podijeljena na blokove.

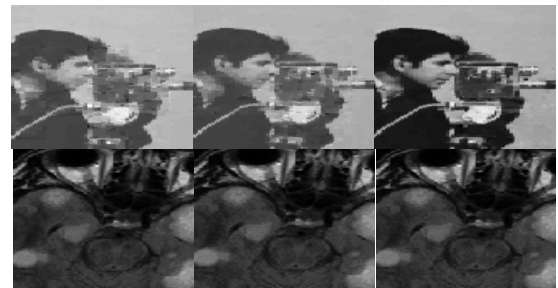
Tabela 2: Rezultati simulacija za algoritam 2, djeljenjem slike na blokove dimenzija nxn

slika	veličina bloka	mjerenja (%)	vrijeme (s)	PSNR (dB)
Cameraman	16x16	35	235.09	26.9894
	32x32		115.17	28.2634
	64x64		80.33	30.1655
	16x16	70	154.20	33.8066
	32x32		72.74	35.2401
	64x64		63.82	37.1667
MRI Brain	16x16	35	233.76	32.5143
	32x32		108.16	34.4819
	64x64		83.50	35.6922
	16x16	70	185.03	40.3012
	32x32		93.33	41.9663
	64x64		71.29	43.4082

Analizirane su veličine blokova 64x64, 32x32, 16x16. Iz svakog bloka uzima se 35% ili 70% koeficijenata a rezultati su dati u Tabeli 2. Rekonstruisane slike sa podjelom na različite veličine bloka date su na Slici 4. Rezultati su bolji za veće dimenzije bloka (za isti procentualni iznos uzetih mjerenja). Ovo se može objasniti time da duži signali (veći blokovi) imaju bolje izraženo „sparse“ svojstvo (svojstvo rijetkosti) od kraćih. Međutim, vrijeme izvršavanja je mnogo veće, što je nedostatak ovog načina rekonstrukcije jer otežava njegovu primjenu u realnom vremenu.

C. Rekonstrukcija slike Algoritmom 3

U Tabeli 3 prikazani su parametri korišćeni za rekonstrukciju slike Algoritmom 1: $K1$ - broj NF koeficijenata, $K2$ - broj VF koeficijenata i procenat uzetih mjerenja, kao i rezultati koji se rekonstrukcijom dobijaju – PSNR i vrijeme izvršavanja. Za fiksirane NF koeficijente mijenjan je broj VF koeficijenata, kao i za fiksiran broj VF koeficijenata, praćen je uticaj broja uzetih NF koeficijenata na rekonstrukciju slike. Na osnovu PSNR mjere može se ocijeniti kvalitet rekonstruisane slike. Vrijeme izvršavanja algoritma za rekonstrukciju manje je u slučaju MRI slika. Da bi se ovim algoritmom dobio zadovoljavajući kvalitet rekonstruisane slike mora se uzeti određeni broj NF koeficijenata. Pokazuje se da je ovaj broj NF koeficijenata manji u slučaju medicinskih slika.



Slika 4: Rezultati simulacija za algoritam 2, korišćenjem blokova 16x16, 32x32 i 64x64, respektivno, sa lijeva na desno. Rezultai su dati za 35% mjerenja iz bloka

Vrijeme izvršavanja algoritma predstavlja problem za primjenu ovog metoda u realnom vremenu. Analizom dobijenih rezultata, zaključuje se da se za rekonstrukciju MRI slike može uzeti manji procenat odbiraka nego za realnu

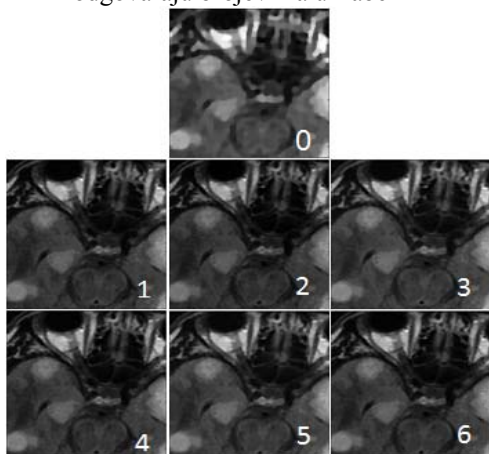
sliku, kako bi se dobio zadovoljavajući kvalitet rekonstruisane slike. Na Slikama 5 i 6 date su rekonstruisane slike, u skladu sa parametrima iz Tabele 3. Nedostatak algoritma je neophodnost postojanja NF koeficijenata slike, što bi značilo da se prilikom akvizicije signala ovi koeficijenti moraju snimiti i sačuvati.

Tabela 3: Rezultati simulacija za Algoritam 3

slika	K1 - NF	K2 - VF	mjerjenja (%)	vrijeme (s)	PSNR (dB)	slika br.
Cameraman	500	1500	12,20	31,92	24,09	0
	400	7792	50,00	49,80	32,98	1
		9430	60,00	56,31	35,21	2
	600	4398	30,51	85,57	28,37	3
		7592	50,00	48,02	32,98	4
	1000	3998	30,51	85,07	28,37	5
		7192	50,00	49,97	32,99	6
Brain	500	1500	12,20	84,12	28,84	0
	400	7792	50,00	37,97	38,87	1
		9430	60,00	40,08	40,88	2
	600	4398	30,51	71,40	34,83	3
		7592	50,00	36,57	38,87	4
	1000	3998	30,51	65,20	34,83	5
		7192	50,00	40,32	38,87	6



Slika 5: Rekonstruisana slika *Cameraman* za različite odnose NF i VF koeficijenata (zumirani region slike). Brojevi na slici odgovaraju brojevima u Tabeli 1



Slika 6: Rekonstruisana slika *MRI Brain* za različite odnose NF i VF koeficijenata. Brojevi na slici odgovaraju brojevima u Tabeli 1.

5. ZAKLJUČAK

U radu su opisana tri metoda za rekonstrukciju slike CS metodom. Kod svih opisanih algoritama mjerenja se uzimaju iz domena frekvencija, a rješenje se traži TV minimizacijom. Pokazano je da se za rekonstrukciju medicinskih slika može koristiti algoritam koji odbirke signala uzima 2D DFT

maskom. Vrijeme rekonstrukcije je malo, a kvalitet rekonstruisane slike zadovoljavajući. Rekonstrukcija slike Algoritmom 2, tj. dijeljenje slike na blokove daje zadovoljavajući kvalitet, ali je vrijeme izvršavanja veliko te nije pogodno za korišćenje u medicini. Najbolji rezultati u pogledu PSNR-a postignuti su Algoritmom 3. Međutim, ovaj algoritam koristi određeni broj NF koeficijenata, pa se podrazumijeva da sa tim koeficijentima raspolažemo, što u većini realnih aplikacija nije slučaj. Eksperimentalni rezultati pokazuju da je, generalno, za medicinske slike moguće uzeti manji set odbiraka nego u slučaju realnih, svakodnevnih slika.

LITERATURA

- [1] S. Stankovic, I. Orovic, E. Sejdic, *Multimedia Signals and Systems*, Springer-Verlag, New York, 2012.
- [2] R. Baraniuk, "Compressive sensing," *IEEE SP Magazine*, vol. 24, no. 4, 2007, pp. 118-121.
- [3] LJ. Stankovic, S. Stankovic, M. Amin, "Missing Samples Analysis in Signals for Applications to L-estimation and Compressive Sensing," *Signal Processing*, vol. 94, Jan 2014, pp. 401-408, 2013.
- [4] I. Orovic, S. Stankovic, "Improved Higher Order Robust Distributions based on Compressive Sensing Reconstruction," *IET Signal Processing*, 2014.
- [5] E. J. Candes, M. B. Wakin, "An Introduction To Compressive Sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21, 30, March 2008.
- [6] I. Orovic, S. Stankovic, T. Thayaparan, "Time-Frequency Based Instantaneous Frequency Estimation of Sparse Signals from an Incomplete Set of Samples," *IET Signal Processing, Special issue on Compressive Sensing and Robust Transforms*, 2014.
- [7] J. Romberg, "Imaging via Compressive Sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 14, 20, March 2008.
- [8] I. Orovic, S. Stankovic, "Compressive Sampling and Image Watermarking," *55th International Symposium ELMAR 2013, Zadar, Croatia, Sept. 2013*, 2013.
- [9] Lustig M, Donoho D, Pauly JM., „Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn Reson Med*. 2007 Dec;58(6):1182-95.
- [10] Potter, L.C.; Ertin, E.; Parker, J.T.; Cetin, M., "Sparsity and Compressed Sensing in Radar Imaging," *Proceedings of the IEEE*, vol.98, no.6, pp.1006,1020, June 2010.
- [11] I. Orovic, S. Park, S. Stankovic, "Compressive sensing in Video applications," *21st Telecommunications Forum TELFOR 2013, Novembar*, 2013.
- [12] I. Orovic, A. Draganic, S. Stankovic, "Compressive Sensing as a Watermarking Attack," *21st Telecommunications Forum TELFOR 2013, Novembar*, 2013.
- [13] L. Rudin, S. Osher, E. Fatemi, "Nonlinear total variation based noise removal algorithms", *Physica D*, 60:259-268, 1992.
- [14] Needell, D.; Vershynin, R., "Signal Recovery From Incomplete and Inaccurate Measurements Via Regularized Orthogonal Matching Pursuit," *Selected Topics in Signal Processing, IEEE Journal of*, vol.4, no.2, pp.310,316, April 2010.

METODA DETEKCIJE VATRE U REALNOM VREMENU NA BAZI OBRADJE SLIKE REAL-TIME FIRE-DETECTION METHOD BASED ON IMAGE PROCESSING

Marko Asanović, Radovan Stojanović, Igor Đurović, *Elektrotehnički fakultet Podgorica*

Sadržaj: U radu je prikazan jedan od pristupa detektovanja vatre u realnom vremenu snimljene nepomičnom kamerom, rezolucije 640x480 sa apsolutnom razlikom između dva frejma 1/30 sekundi. Detekcija je izvršena analizom pokreta i boje, primjenom Gausovog i RGB modela. Procesuiranje i prikaz je izvršen u Matlabu upotrebom odgovarajućih toolboxova. Dati su preliminarni rezultati testiranja pristupa na konkretnim primjerima. Predložena metodologija i rješenje biće iskorišćeni u okviru IPA ADRIATIC HOLISTIC i GEPSUS sistema za vanredne situacije.

Abstract: This paper presents an approach for fire detection in real-time by using stationary camera of 640x480 resolution with speed of 1/30 seconds. A Gaussian mixture model applied on movement and RGB color space is employed. The MATLAB with association toolboxes is used for processing and visualization. The preliminary testing results on specific examples are presented. The proposed methodology and solutions will be used within the IPA ADRIATIC HOLISTIC and GEPSUS emergency system.

1. UVOD

Sistemi za otkrivanje požara su jedni od najvažnih u sistemu nadzora praćenja objekta i okoline, kao dio mehanizma za upozorenje o mogućnosti početka požara. Kako vatra može prouzrokovati ozbiljne probleme, njena pravovremena detekcija je od izuzetne važnosti. Sa razvojem tehnologija digitalnih kamera i tehnika video obrade, postoji veliki trend zamjene uobičajnih metoda za dojavu požara. U poređenju sa nepomičnom slikom, video sekvenca daje više informacija i scenarijume koji se mijenjaju tokom vremena, pri tom pružajući veću fleksibilnost i brzinu otkrivanja. Postoji više metoda za detekciju požara pomoću obrade slike, a veoma su česti oni koji se zasnivaju na jednom ili više kolornih indeksa, kompenzacija ili formalnoj strukturi svjetlosti [1].

U radu je predstavljen algoritam koji se zasniva na obradi pokretne slike koja detektuje vatru u zatvorenom prostoru sa malim promjenama inteziteta. Da bi se smanjila mogućnost detektovanja lažnog alarma korišćene su neke od karakteristika vatre, kao što su kretanje i boja. Korišćen je Gausov model (GMM) za oduzimanje pozadine. Postignuti su zadovoljavajući rezultati na kompletnoj pozadini scene i otkrivanju vatre na osnovu boje piksela.

2. POSTUPCI SEGMENTACIJE

U mnogim sistemima gdje se vrši analiza video sadržaja vrši se segmentacija (izdvajanje) pokretnih objekata, kao jedan od prvih koraka u obradi video signala. Segmentirani objekti se potom prosljeđuju kao ulaz narednim blokovima za

obradu, čiji kvalitet rada u velikoj mjeri zavisi od kvaliteta obavljene segmentacije.

U prvim pokušajima da se riješi problem, pokretni objekti su izdvajani iz binarne slike koja predstavlja kvantovanu razliku dva uzastopna frejma sa fiksnim pragom [2]. Iako veoma brz i adaptivan, ovakav metod ima dosta nedostataka koji se pojavljuju prilikom kretanja površine uniformne teksture kada veći dio pokretne površine ostaje nedetektovan, a pogotovo je problematičan privremeni prekid kretanja objekta. Jedna od definicija termina teksture odnosi na ponavljanje osnovnih teksturnih elemenata, teksela. Jedan teksel sadrži više piksela čiji raspored može biti periodičan, kvaziperiodičan ili slučajan.

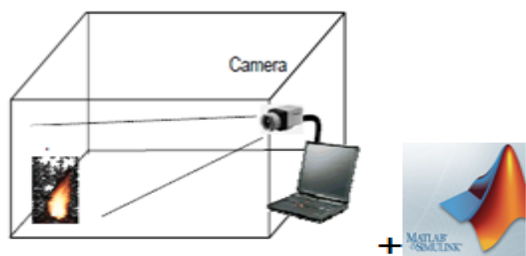
U praktičnim primjenama efikasnijim su se pokazale metode zasnovane na procjeni pozadine i njenom oduzimanju od trenutnog frejma. Ova metoda je bazirana na oduzimanju pozadine [3], a procjena pozadine se dobija "running average" filtriranjem vrijednosti intenziteta piksela u toku vremena. Ipak, ovakav način izdvajanja pozadine ne daje potpuno zadovoljavajuće rezultate s obzirom da sporo pomični i privremeno nepomični objekti vrlo brzo bivaju pridodati pozadini.

Precizan način za procjenu nepromenljive pozadine, kada statistička zastupljenost pozadine premašuje 50%, predstavlja temporalno "median" filtriranje. Glavna slabost ovakvog pristupa je nemogućnost algoritma da se adaptira na promjenljivo osvjetljenje, kao i na periodične pokrete kvazistatičkih objekata u sceni.

Da bi se algoritam za segmentaciju uspješno nosio sa nabrojanim problemima, potrebno je primijeniti komplikovanije probabilističke modele. Jedan takav model je takozvani Stauffer-Grimsonov algoritam [4], u kojem se ponašanje svakog piksela modeluje kao međusobno nezavisan model GMM. Parametri Gausove raspodjele nijesu fiksni, već se iz frejma u frejm mijenjaju, i iznova obnavljaju. Algoritam na bazi kojeg se izračunavaju nove vrijednosti imaju težinu, srednju vrijednost i varijansu svakog moda modela raspodjele, predstavlja varijantu EM (Expectation Maximization) algoritma za rad u realnom vremenu. Postupak Optičkog protoka se koristi u kompjuterskoj viziji za karakterizaciju i kvantizaciju kretanja objekta u video prikazu, često za prikaz objekta i praćenje sistema. Prednost Optičkog protoka je brzina izračunavanja a nedostatak je cjelokupni objekat koji se prati u realnom vremenu. Optički protok je distribucija prividne brzine objekta na slici. Područje pokreta pravi šum u realnoj slici i optički protok vrši procjenu koja uključuje samo lokalno računanje [5].

3. ALGORITAM

Da bi smo razvili algoritam za detekciju vatre moramo da razumijemo samu prirodu vatre. Vatra je obično crvene boje, osim toga boja vatre se mijenja u zavisnosti od temperature [6]. Aplikacije u obradi slike nam omogućavaju da koristimo visoke kontraste boja i da to iskoristimo u definisanju karakteristika vatre. Algoritam se sastoji iz sljedećih koraka: segmentacije vatre u prvi plan (foreground) korišćenjem GMM, detekcije piksela vatre pomoću boje i blob analiza. Na Slici 1 prikazan je postupak realizacije datog algoritma.



Slika 1. Postupak realizacije sistema za detektovanje vatre.

Pozadinsko modelovanje je statički opis tekuće pozadinske scene. GMM je jedan produžetak Gausove funkcije vjerovatnoće, GMM može približno aproksimirati bilo koju formu gustine raspodjele [1]. Pod pretpostavkom da se GMM sastoji od kombinacije Gausove funkcije vjerovatnoće, svaka ta funkcija ima svoju srednju vrijednost, standardnu devijaciju i težinu. Ovaj metod koristi mješavinu od K Gausovih raspodjela za modelovanje slike $\{X_1, \dots, X_t\}$ za

svaki piksel. Vjerovatnost posmatranja trenutne vrijednosti piksela je definisana sumom sveobuhvatne raspodjele [7].

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (1)$$

Gdje je K broj tih raspodjela (između 3 i 5), $\omega_{i,t}$ prosječna težina povezana sa i^{th} Gausa u vremenu t sa srednjom vrijednošću $\mu_{i,t}$ i standardna devijacija kovarijacione matrice mješavine Gausa u vremenu t, gdje je η Gausova funkcija vjerovatnoće. Ideja inicijalizacije modela po pikselu je realizovana na sljedeći način. Svaki frejm se prilagodi brzini promjene, koja je jednaka 1/ (tekućem broju frejma). Za svaki piksel, napravi se GMM koji ima K varijacija. Ako piksel pripada modu K, prilagodi parametre moda K (srednja vrijednost, varijansa i težina). Ako piksel ne pripada ni jednom modu, zamijeni rang moda sa novim gdje pikseli predstavljaju srednju vrijednost ili izračunati rang za svaki mod piksela i presortirati na temelju toga ranga. Na osnovu testiranja se odlučuje da li će pikseli biti predstavljeni u foreground. Tehnika za otkrivanje varijacija vremenski je razdvojena jedna od druge. Upoređuje se boja video frejma da se utvrdi da li su pikseli dio pozadine ili se nalaze u prvi plan. Korišćeno je pretvaranje prostorne boje RGB u Intesity. Izdvajanje djelova piksela sa izračunatim promjenama inteziteta na osnovu kojeg se izračunava pozadinska maska.

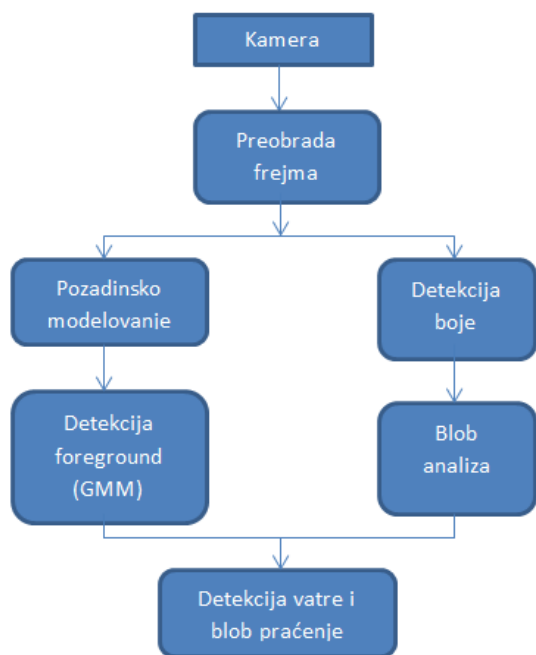
Slika se sastoji od tri osnovne boje crvene(R), zelene(G), i plave(B), dakle svaki dio piksela možemo podijeliti na ove tri vrijednosti. Svaka boja je predstavljena sa 8 bita, odnosno vrijednostima od 0 do 255 (256 vrijednosti). Kako su eksperimenti pokazali, a i čovjeku je lako uočiti, kod piksela vatre je izražen intezitet i zasićenost crvene boje. Faza odlučivanja, vrijednost thresholda, odnosno granične vrijednosti, temelji se na eksperimentima [8]. Prvo pravilo, vrijednosti piksela u crvenom kanalu su veće od threshold vrijednosti, prikazano jednačinom (2). Gdje je $R(x,y)$ intezitet piksela u vremenu t prostornih kordinata, a R_t vrijednost threshold.

$$R(x, y) > R_t \quad (2)$$

Drugo pravilo, pikseli u crvenom kanalu boja su veći od zelenog, i pikseli u zelenom kanalu boja su veći od plavog. Pikseli koji zadovoljavaju prvo i drugo pravilo smatraju se vatrom.

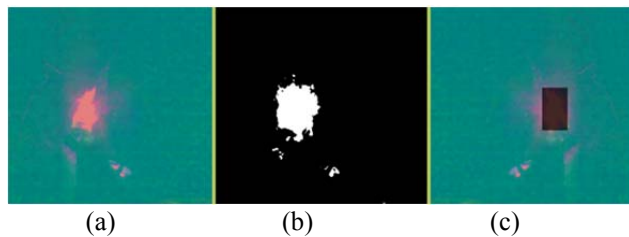
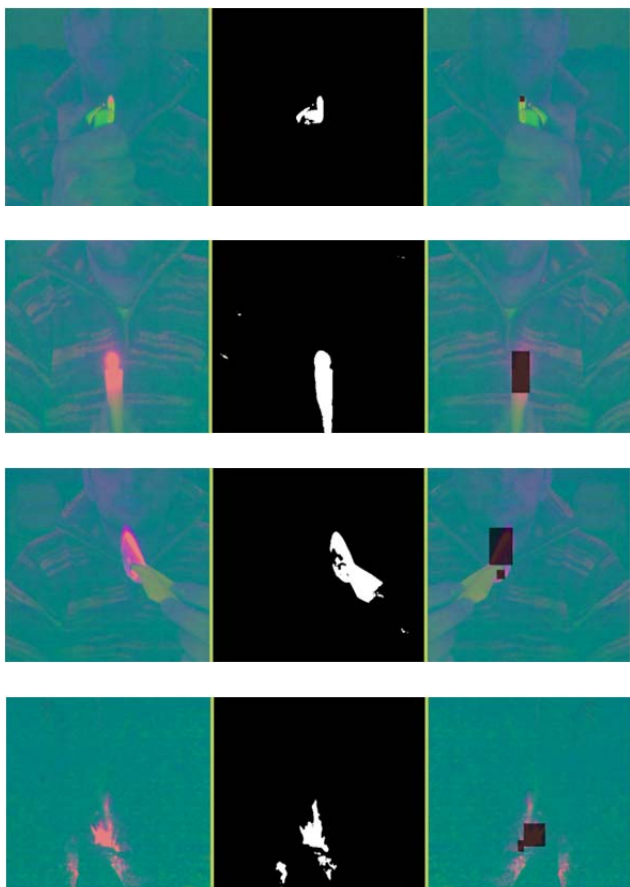
$$R(x, y) > G(x, y) > B(x, y) \quad (3)$$

U obradi slike, blob se definiše kao region povezanih piksela. Blob analiza je identifikacija i proučavanje tih regija na slici. Algoritam prepoznaje piksele na osnovu svojih vrijednosti i stavlja ih u jednu od dvije kategorije: foreground (obično imaju vrijednost različitu od nule) i background (pikseli koji imaju vrijednost nula). Ovdje koristimo blob za obilježavanje vatre u cilju dalje analize, odnosno praćenja [9]. Predloženi algoritam je prikazan na Slici 2.



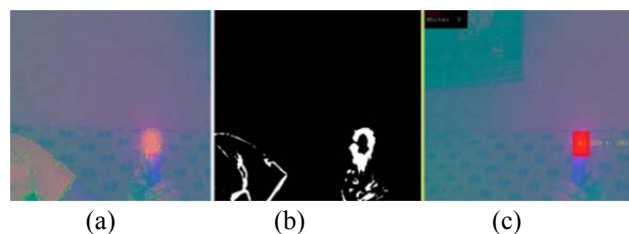
Slika 2. Algoritam za detekciju vatre

Na Slici 3 je prikazano u praksi nekoliko načina detekcije vatre u zatvorenom prostoru.



Slika 3. Prikaz simulacije i detekcija vatre primjenom GMM (a) prikaz frejma u realnom vremenu (b) GMM detekcija foreground (c) blob detekcija piksela vatre označene sa crvenom bojom.

Detekcija vatre na osnovu algoritma koji je predložen na Slici 2, upoređen je primjenom Optičkog protoka umjesto primjene GMM za detekciju vatre u foreground. Pokazano je da GMM vrši bolju segmentaciju i daje bolje rezultate. Na slici 4 je prikaz jednog takvog primjera.



Slika 4. Detekcija vatre korišćenjem Optičkog protoka (a) prikaz frejma u realnom vremenu sa prikazom piksela pomjeraja (b) detekcija foreground sa Optičkim protokom (c) detekcija vatre i blob analiza.

4. REZULTATI I DISKUSIJA

Primjetni su i neki nedostaci GMM, svaka nova vrijednost piksela koja se ne može pridružiti ni jednom od postojećih modova predstavlja začetak novog moda, što stvara probleme oko njegove inicijalizacije. Pored toga, formiranje novog moda u multimodalnoj raspodeli nepotrebno povećava kompleksnost algoritma.

Predloženi algoritam je implementiran korišćenjem Matlab 2013a. Lako je zaključiti da na uspješnost algoritma utiče mnogo faktora, i potrebno je uzeti što više karakteristika koje mogu predstavljati vatru, a najvažnije uzeti u obzir temperaturu i karakteristike materijala tokom samog procesa sagorijevanja. Vatra ima posebnu osobinu stalno mijenja svoj oblik i površinu. Cilj ovog načina pristupa koji je ovdje prikazan je bio detektovati potencijalnu vatru i izvršiti praćenje u cilju dalje analize, odnosno alarmirati čovjeka, koji zatim ima zadatak provjeriti da li je riječ o pravom ili lažnom alarmu.

LITERATURA

[1] KhosroRezaee, S. Jalal Mousavirad, Mohammad RaseghGhezelbash, JavadHaddania, "Accurate Fire Detection System for Various Enviroments using

- Gaussian Mixture Model”, Journal of Information System and Telecommunication, Vol.1, Jan-Mach 2013.
- [2] M. Sonka, V. Hlavac and R. Boyle Image Processing, Analysis, and Machine Vision PWS Publishing, 1999.
- [3] C. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland, “Pfinder: real-time tracking of the human body,” IEEE Trans. on Pattern Anal. and Machine Intell., vol. 19, no. 7, pp. 780-785, 1997.
- [4] C. Stauffer and W. Grimson, “Adaptive Background Mixture Models for Real-time Tracking”, Proceedings IEEE Conference on Computer Vision and Pattern Recognition, pp. 246-252, 1999.
- [5] A. Chauhan, P. Krishan, “Moving Object Tracking using GMM and Optical Flow”, vol 3, pp.243-246, April 2013.
- [6] T. Chen, P. Wu, and V. Chiou, "An early fire-detection method based on image processing" In: Proc. IEEE International Conf. on Image Processing, ICIP 04, 2004, pp. 1707–1710.
- [7] D. Santosh, P. Venkatesh, P. Poornesh, and L. Norayana Rao, “Tracking multiple moving Object using Gaussian Mixture Model”, ISSN: 2231-2307, May 2013.
- [8] J. Ebert and J. Shiplez, “Computer Vision based metod for fire detection in color images”, IEEE Conference on Soft. Computing in Industrial Applications, June 2005.
- [9] <http://www.mathworks.com>

ZAHVALNOST

Autori su zahvalni **IPA ADRIATIC HOLISTIC** i **NATO GEPSUS** projektima na pruženoj tematskoj i materijalnoj podršci za realizaciju predmetnog istraživanja.

ANALIZA PERFORMANSI FastICA ALGORITMA PRIMIJENJENOG NA 2D SIGNAL PERFORMANCE ANALYSIS OF THE FastICA ALGORITHM APPLIED ON A 2D SIGNAL

Nikola Besic^{1,2}, Gabriel Vasile¹, Budimir Lutovac², Srdjan Stankovic² and Dragan Filipovic²

¹Gipsa-lab, Grenoble-INP/CNRS, Grenoble, France

²Faculty of Electrical Engineering, University of Montenegro, Podgorica, Montenegro

Sadržaj: Slijepo razdvajanje izvora danas predstavlja jednu od najznačajnijih oblasti istraživanja u digitalnoj obradi signala i slike. Sama oblast zapravo objedinjava veliki broj različitih, karakterističnih metoda, među kojima je najrevolucionarnija, razmatrana u ovom radu, Analiza Nezavisnih Komponenti. FastICA method, bazirana na fixed-point iterativnom algoritmu, predstavlja najrasprostranjeniji alat u estimaciji nezavisnih komponenti. U ovom radu prezentujemo analizu performansi ovog algoritma, primijenjenog na 2D signal (sliku).

Abstract: Blind source separation is one of the major areas of research in signal and image processing today. Being a broad area, it actually comprises various distinctive methods. The revolutionary one, considered here, is Independent Component Analysis (ICA). Nowadays, the FastICA method, based on the fixed-point iterative algorithm, is probably the most widespread tool in estimating independent components. In this paper we analyse the performances of this algorithm when applied on 2D signal (image).

1. INTRODUCTION

The very purpose of the blind source separation is recovering source signals without having detailed knowledge of the mixing process. If we assume having the vector \mathbf{X} consisted of n either time or space dependent observations:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ \vdots \\ x_n(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \\ \vdots \\ s_n(t) \end{bmatrix}, \quad (1)$$

we are supposed to relate it to the vector \mathbf{s} consisted of n sources being functions of the same variable (time or space) as the corresponding observations [1]. The matrix \mathbf{A} is called a mixing matrix, and it provides information concerning the “involvement” of each of the sources for each of the observations.

If we knew coefficients of the mixing matrix (a_{ij}), the problem would be quite simple – reduced to solving a system of linear equations, but in fact, we don't have any information on the mixing process. That is why this kind of a separation is called blind. Therefore, the only thing we can do it to assume the statistical relation of the source signals and then to try estimating these signals. Exactly this kind of the statistical relation is the base for the classification of blind source separation methods, where the most distinguished ones are the Principal Component Analysis (PCA) and the Independent Component Analysis (ICA).

The PCA assumes that the sources are statistically uncorrelated. This method is limited to the analysis of the second order statistics – covariance matrix of the observations' vector (\mathbf{x}). It can be demonstrated that the resulting mixing matrix after the normalization corresponds to the unitary matrix of covariance matrix eigenvectors. The one-dimensional sources are taking role of the eigenvalues, providing the “power” of the corresponding mixing matrix column.

On the other side, the ICA is rather based on the assumption of the statistical independence of the sources. In this case, the higher order statistics is considered as well. This fact becomes essential once we deal with the Non-Gaussian observations.

There are several different approaches in estimating independent components, among which the most representative are the ones based on emphasizing the Non-Gaussianity of the sources (FastICA) [2] and the ones exploiting higher-order statistics through tensorial decompositions (FOBI, JADE) [3]. Even though the later appear to be equally attractive in the context of ICA, the subject of this paper is rather the FastICA algorithm, given its performances in case of high-dimensional data and the availability. Therefore, starting by describing the idea behind, we introduce the algorithm itself with a strong emphasis on the role of different nonlinearities which are necessarily used in estimating independent components. Finally, by applying the algorithm on two synthesized “mixed” images (2D signals), we compare the statistics of the estimated sources with the simulated ones (used in “mixing”). The comparison is made with respect to the chosen nonlinearity.

2. THE IDEA BEHIND THE FastICA

The most important restriction concerning ICA (and therefore FastICA) is the statistical distribution of the independent components. It cannot be gaussian, for if the mixing matrix actually represents orthogonal transform, the observations will also have the joint gaussian pdf, leading to the unidentifiability of mixing matrix columns directions (for the elements random vector, the joint pdf will have the shape of a circle).

This restriction represents the founding principle of the concerned approach in the IC estimation – FastICA [4]. Namely, after the Central Limit Theorem, the sum of the independent random variables always tends to follow normal distribution. If we assume that there is a vector \mathbf{y} which is related to the observation vector by means of relation:

$$\mathbf{y} = \mathbf{w}\mathbf{x}, \quad (2)$$

the goal is to find vector \mathbf{w} so that it corresponds to one of the rows of the inverse mixing matrix \mathbf{A} . As we don't know the mixing matrix, the established criteria is minimizing the gaussianity of vector \mathbf{y} . By introducing the relation $\mathbf{z} = \mathbf{A}\mathbf{w}$ we could rewrite the previous equation as:

$$\mathbf{y} = \mathbf{w}\mathbf{x} = \mathbf{w}\mathbf{A}\mathbf{s} = \mathbf{z}\mathbf{s}. \quad (3)$$

As the central theorem states the product will be the least gaussian if we have just one member of the sum, saying if the \mathbf{z} vector has just one non-zero element, which implies that the $y_i = s_i$. So we need to determine vector \mathbf{w} which maximizes the nongaussianity of \mathbf{y} . In order to do that, the measure of the gaussianity must be introduced.

The four measures of (non)gaussianity considered in the FastICA approach are the following:

- **Kurtosis** – defined in the excess form as:

$$\text{kurt}(y) = \frac{E[y^4]}{(E[y^2])^2} - 3$$

The normalized fourth moment of the gaussian variable equals 3, meaning that the excess kurtosis different from zero points out nongaussianity. Generally, if the value is positive, we have platykurtic distribution (spiky pdf) while the negative value characterize leptokurtic distributions (flat pdf).

- **Aproximated negentropy** – defined using nonlinear function G_i [6] and Gaussian variable v :

$$J(y) \approx \sum k_i [E\{G_i(y)\} - E\{G_i(v)\}]^2$$

The gaussian variable has the largest entropy among all the random variable with the same variance. Therefore, the negentropy is the quantity having maximal value for non-gaussian variable.

- Mutual information, defined in the context of the information theory.

- Maximum likelihood estimation, based on the log-likelihood function L [5], where we ought to presume the statistics of the sources:

$$L = \sum_{t=1}^T \sum_{i=1}^n \log f_i(\mathbf{w}_i^T \mathbf{x}(t)) + T \log |\det \mathbf{W}|$$

with f_i being the density function of s_i .

3. THE FastICA ALGORITHM

The algorithm used in this analysis [7] is based on maximizing the negentropy as the measure of nongaussianity. In this case, the selection of the nonlinear function appears to be critical with respect to the nature of the observation data. Therefore, having 2D signals (images) as observations, in this article we compare three mostly used functions:

$$\begin{aligned} \text{pow3: } G_1(y) &= \frac{y^4}{4} \\ \text{tanh: } G_2(y) &= \frac{1}{a} \log(\cosh(ay)) \\ \text{skew: } G_3(y) &= \frac{y^3}{3} \end{aligned} \quad (4)$$

The aim of the iterative routine is to maximize the function $F(\mathbf{w})$ given the constraint:

$$\begin{aligned} \max F(\mathbf{w}) &= \max [E\{G(\mathbf{w}^T \mathbf{x})\} - E\{G(v)\}] \\ \|\mathbf{w}\|^2 &= 1 \end{aligned} \quad (5)$$

After applying the Lagrange multiplier, the function $F(\mathbf{w})$ can be rewritten in terms of the first derivatives of the function given in the Eq. 4 (g) and the optimal value of \mathbf{w} (\mathbf{w}_0):

$$F^*(\mathbf{w}) = E\{\mathbf{x}g(\mathbf{w}^T \mathbf{x})\} - E\{\mathbf{w}_0^T \mathbf{x}g(\mathbf{w}_0^T \mathbf{x})\} \cdot \mathbf{w}. \quad (6)$$

This way, the iteration is reduced to the Newton method used in order to find a vector \mathbf{w} leading to the maximal negentropy. Given that we are actually dealing with the nonlinear system of equation, this has to be done using Jacobian matrix $J(\mathbf{w})$:

$$J(\mathbf{w}) = E\{g'(\mathbf{w}^T \mathbf{x})\} \mathbf{I} - E\{\mathbf{w}_0^T \mathbf{x}g(\mathbf{w}_0^T \mathbf{x})\} \mathbf{I}. \quad (7)$$

Finally, the principal step of the algorithm is the following:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - J^{-1}(\mathbf{w}_n) F^*(\mathbf{w}_n), \quad (9)$$

with \mathbf{w}_{n+1} being normalized after each iteration. The convergence of the algorithm is verified by calculating a dot product of \mathbf{w}_{n+1} and \mathbf{w}_n , which ought to be zero.

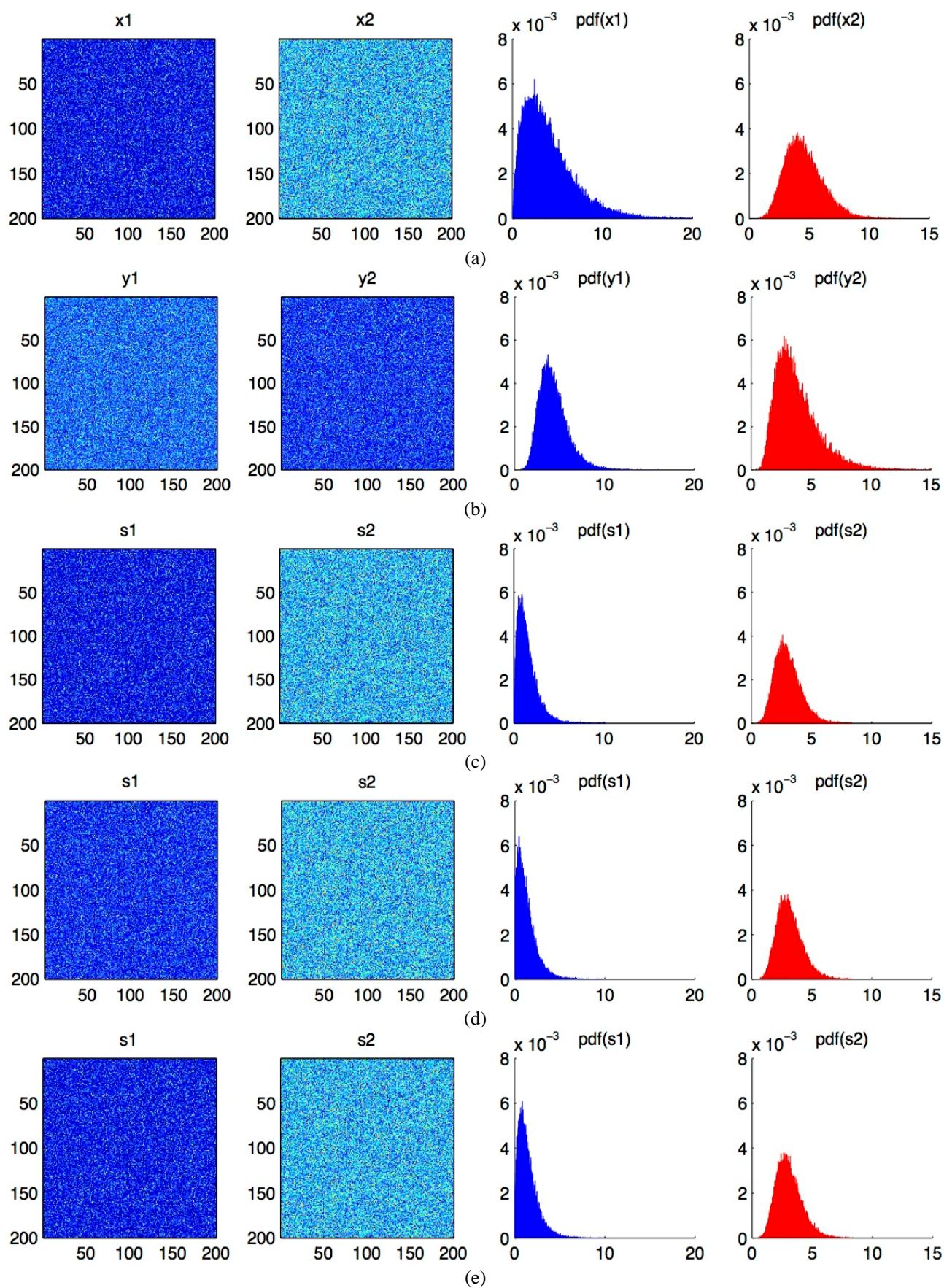


Figure 1: The comparison between the simulated (a) and the estimated sources, in terms of images and their statistics (pdf): (b) mixture, (c) first criterion – pow_3 , (d) second criterion – $tanh$, (e) third criterion – $skew$.

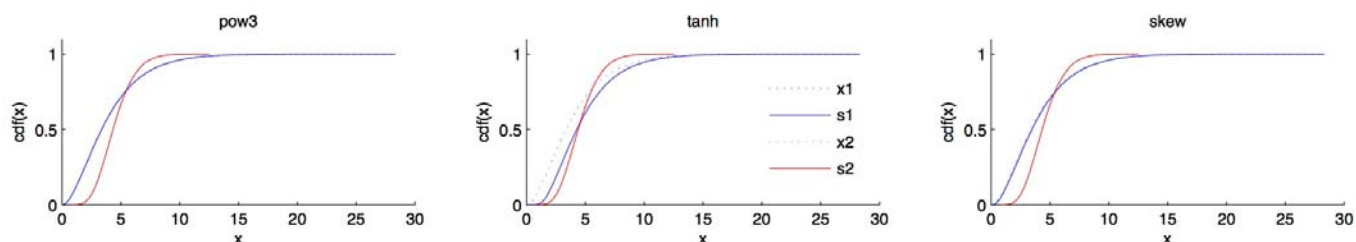


Figure 2: The comparison between the simulated and the scaled estimated sources in terms of their statistics (cdf).

4. THE COMPARATIVE ANALYSIS

In order to compare the performances of three introduced nonlinear functions in case of 2D observations, we have simulated two images characterized with the Gamma distribution:

$$\text{pdf}(x|k, \theta) = \frac{x^{k-1} e^{-\frac{x}{\theta}}}{\theta^k \Gamma(k)}. \quad (10)$$

The defining shape k and scale θ parameters are, however, different (Fig. 1a). In case of the first one $k = \theta = 2$, while for the second one $k = 9$, $\theta = 0.5$.

Following step was mixing these two images in order to synthesize the observation data (Fig. 1b). The mixing is both linear (Eq. 1) and stationary (meaning that the coefficients a_{ij} don't represent a function of space).

Finally, the introduced algorithm using all three introduced nonlinear functions (pow3, tanh and skew) is applied on the synthesized mixture (Fig. 1c, 1d, 1e). Probability density functions (pdf) of the sources are estimated by normalizing their histograms.

The evaluation of the different criteria is performed by deriving corresponding cumulative distribution functions (cdf) (Fig. 2) and calculating the mean square error (MSE) between the simulated (original) images and the estimated sources. Although the more conventional statistical tool would have been the estimation of the shape and scale parameters of the estimated sources, given the purpose (comparative analysis) and the simplicity of this method, it appears as a more suitable tool.

The obvious remark would be that the algorithm (in case of all three functions) fails to estimate the magnitude of a signal. This is, however, justified by the fact that the preprocessing and postprocessing of the observation data assume both (de)whitening and (de)centering. Therefore, before deriving the cdf, each pdf is linearly scaled in order to fit a range of values of original 2D signals. The results of the MSE comparison, provided in Table 1, unambiguously indicate the superiority of the *pow3* criterion.

Table 1: MSE between the cdf of the simulated the scaled estimated sources.

	pow3	tanh	skew
I	1.1672e-06	0.0037	5.6857e-05
II	1.6364e-08	5.4661e-05	1.5862e-07

5. CONCLUSIONS

The FastICA algorithm is the prevalent ICA method, allowing a blind estimation of the independent sources. It is the iterative fixed-point algorithm, adopting a non-gaussianity of the sources as an indicator of the independence. Depending on a measure of non-gaussianity, there are several realizations of the algorithm. The one considered as most reliable and therefore used in this article – the approximated negentropy, depends strongly on a choice of a nonlinearity allowing the approximation. Therefore, we have analyzed the performances of the FastICA algorithm depending on a choice of the nonlinear function, having a 2D signal (image) as the observation. The comparative analysis was performed by considering the MSE between the cumulative distribution functions of the original (simulated) images and the scaled estimated sources. Eventually, we conclude that the quartic function (criterion *pow3*) appears to be the most suitable in case of the considered observations.

REFERENCES

- [1] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, 2010.
- [2] A. Hyvarinen and E. Oja, "Independent Component Analysis: Algorithms and Applications," *Neural Networks*, vol. 13, pp. 411-430, 2000.
- [3] J. -F. Cardoso, "Source separation using higher-order moments," in *Proc. ICASSP*, Glasgow, May 1989, pp. 2109-2112.
- [4] A. Hyvarinen, J. Karhunen and E. Oja, *Independent Component Analysis*, John Wiley and Sons, Inc., New York, NY, USA, 2001.
- [5] D.-T. Pham, P. Garrat, and C. Jutten, "Separation of a mixture of independent sources through a maximum likelihood approach," in *Proc. EUSIPCO*, pp. 771-774, 1992.
- [6] A. Hyvärinen, "New approximations of differential entropy for independent component analysis and projection pursuit," in *Advances in Neural Information Processing Systems*, volume 10, pp. 273-279. MIT Press, 1998.
- [7] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, 10(3), 626-634. 1999.

EFIKASNOST MDB ALGORITMA KOD FILTRIRANJA SLIKA SA VODENIM ŽIGOM EFFICIENCY OF MDB ALGORITHM FOR FILTERING WATERMARKED IMAGES

Bojan Prlinčević, *Visoka tehnička škola, Zvečan, Srbija*

Zoran Milivojević, *Visoka tehnička škola, Niš, Srbija*

Darko Brodić, *Univerzitet u Beogradu, Tehnički fakultet Bor, Srbija*

Sadržaj: U prvom delu rada opisan je Decision-Based Algorithm (DB) algoritam i izvršena je njegova modifikacija u delu zamene vrednosti inficiranog piksela, Modified Decision-Based Algorithm (MDBA). U drugom delu rada izvršeno je testiranje efikasnosti MDB algoritma kod procesiranja slika sa umetnutim vodenim žigom i superponiranim impulsnim šumom. Rezultati testiranja prikazani su tabelarno i grafički. Na kraju je izvršena komparativna analiza rezultata sa PAV algoritmom i pokazana veća efikasnost MDB algoritma.

Ključne reči: Impulsne smetnje, detekcija šuma, vodeni žig.

Abstract: In the first part of this paper is described the Decision-Based Algorithm (DB) algorithm and its modification in part of the replacement of the infected pixels Modified Decision-Based Algorithm (MDBA). In the second part of this paper is tested the efficiency of the MDB algorithm for processing images with watermark inserted and superimposed impulse noise. The test results are presented in the tables and graphic. Finally is a performed a comparative analysis of the results of the PAV and demonstrated greater efficiency of MDB algorithm.

Key words: Impulse noise, noise detection, watermarking.

1. UVOD

U procesu snimanja, arhiviranja, prenosa i obrade slike neizbežno se superponiraju smetnje i šumovi. Superponirane smetnje mogu biti sa takvim intenzitetom da dovode do stepena degradacije u tolikoj meri da slika bude praktično neupotrebljiva. Kod procesiranja slika u svrhu prepoznavanja sadržaja, objekata, likova, detekcije pokreta i dr., vrlo je važno da slika bude sa zadovoljavajućim kvalitetom. Kao prva operacija kod algoritama za procesiranje slike primenjuje se popravka kvaliteta, odnosno, smanjenje efekta superponiranih šumova. Šumovi mogu da budu opisani različitim modelima. Odlikuje ih vremenska, prostorna i amplitudna neizvesnost. Najčešće se amplitude (intenzitet) šumova kod pojedinih piksela slike mogu da modeluju Gausovim zakonom raspodele. Drugi značajni oblik šumova su impulsne smetnje koje su vremenski, odnosno, prostorno slučajno raspoređene, dok je amplituda maksimalna ili minimana. Zbog toga se impulsne smetnje manifestuju u vidu belih i crnih tačaka [1] i poznate su kao so i biber (**engl. salt and pepper**). Za eliminaciju impulsnog šuma predloženi su mnogobrojni algoritmi [2-7].

Predloženi algoritmi linearnog filtriranja [6] za eliminaciju impulsnog šuma pokazali su se nedovoljno uspešnim. Postupci koji su bazirani na nelinearnom filtriranju impulsnog šuma pokazali su se efikasnijim. Proces obrade šuma u slici sastoji se iz dva dela, detekcije inficiranih tačaka i obrade inficiranih tačaka. Ispravna detekcija impulsnog šuma je od bitnog značaja za eliminaciju šuma. Takođe obrada inficiranih tačaka, tj. zamena vrednosti novom vrednošću koja treba da bude što približnija originalnoj predstavlja značajan postupak u eliminaciji impulsnog šuma [8].

Filteri dizajnirani za obradu slike treba da otklone impulsni šum bez oštećenja ivica slike [9,10]. Većina filtera funkcioniše tako što obrađuje celu sliku i na taj način modifikuje i piksele sa šumom i neoštećene piksele. Idealno bi bilo da primenjeni filter obrađuje samo piksele zaražene šumom. Filteri bazirani na odlučivanju [3,4,7] funkcionišu tako što zaraženi piksel zamenjuju srednjom vrednošću prozora analiziranih piksela, dok neoštećene piksele ostavljaju nepromenjenim. Ovakav način filtriranja je efikasan samo kod procentualno manjeg broja inficiranih piksela. Sa povećanjem procenta inficiranih piksela povećava se i broj zamenjenih i korumpiranih piksela. Kao posledica toga ivice su zamazane i dolazi do gubljenja finih detalja slike. Predlog za prevazilaženje ovog problema dali su u svom radu [11] Srinivasan i Ebenezer gde su predložili novi algoritam zasnovan na odlučivanju DBA (**engl. Decision-Based Algorithm**).

U ovom radu prikazan je algoritam za detekciju i popravku impulsnih smetnji u slici koji je baziran na algoritmu prikazanom u [11]. Modifikacija se odnosi na deo za repliciranje vrednosti detektovanog inficiranog piksela. Modifikovani algoritam (MDB), testiran je na popravci kvaliteta slika sa umetnutim vodenim žigom kojima je superponiran impulsni šum. Za utiskivanje žiga korišćen je SVD metod predložen u radu [12]. Efekat dejstva MDBA analiziran je upoređivanjem originalne slike i slike nakon eliminisanja impulsnog šuma, kao i originalnog vodenog žiga i ekstraktovanog vodenog žiga nakon popravke kvaliteta slike. Kao mera kvaliteta korišćeni su srednja apsolutna greška MAE (**engl. mean absolute error**), srednja kvadratna greška MSE (**engl. mean square error**) i vršni odnos signal-šum PSNR (**engl. peak signal-to-noise ratio**) i subjektivni

vizuelni kvalitet filtrirane slike i žiga. Numerički rezultati biće upoređeni sa rezultatima iz [8,11].

Rad je organizovan na sledeći način. U sekciji 2 opisan je algoritam za detekciju i obradu slike (algoritam DBA, algoritam MDB i algoritam za utiskivanje žiga). Sekcija 3 opisuje algoritam testiranja. U sekciji 4 se prezentuju dobijeni rezultati i vrši se njihova analiza. Zaključak je dat u sekciji 5.

2. ALGORITMI PRIMENJENI U OVOM RADU

2.1 DBA algoritam

DB algoritam za detekciju i eliminaciju impulsnog šuma realizuje se u sledećim koracima:

Korak 1: Formiranje prozora dimenzija 3x3. Analizirani piksel $W(X,Y)$ nalazi se u sredini prozora;

Korak 2: Sortiranje elemenata prozora u rastućem redosledu po vrstama, i u rastućem redosledu po kolonama.

Kao rezultat ovih aktivnosti elementi sporedne dijagonale sortirani su u rastućem redosledu (P_{min}, P_{med} i P_{max}), gde P_{min} predstavlja najmanju, P_{med} srednju a P_{max} najveću vrednost elemenata prozora.

Korak 3: Detektovanje piksela sa impulsnim šumom

IF ($P_{min} < W(X,Y)$ & $P_{min} < 0$ & $P_{max} < 255$ & $W(X,Y) < P_{max}$);
piksel je bez šuma;

ELSE

piksel je sa smetnjom;

IF ($P_{min} == W(2,2)$ & $P_{min} == 0$) | ($W(2,2) = P_{max}$ & ...
 $P_{max} == 255$);

$W(X,Y) = P_{med}$;

ELSE

$P_{med} = P_s$;

END

END.

P_s -predstavlja susedni piksel piksela P_{med} .

Korak 4: Koraci 1-3 se ponavljaju dok se ne izvrši procesiranje cele slike.

2.2 MDB algoritam

Primenjeni MDB algoritam modifikovan je, u odnosu na opisani DBA algoritam, u koraku 3. Korak 3 MDBA algoritma izvršava se na sledeći način:

IF ($P_{min} < W(X,Y)$ & $P_{min} < 0$ & $P_{max} < 255$ & $W(X,Y) < P_{max}$);
piksel je bez šuma;

ELSE

piksel je sa smetnjom;

IF ($P_{min} == W(2,2)$ & $P_{min} == 0$) | ($W(2,2) = P_{max}$ & ...
 $P_{max} == 255$);

$W(X,Y) = P_{med}$;

ELSE

$P_{med} = P_{sred}$;

END

END.

P_{sred} - predstavlja srednju vrednost procesiranog prozora. Za izračunavanje P_{sred} nisu uzimani pikseli čija je vrednost 0 i 255.

2.3 Algoritam za utiskivanje žiga

Utiskivanje žiga je izvršeno korišćenjem SVD transformacije opisane u radu [12]. Primenjeni algoritam se sastoji u sledećem:

Korak 1: Vrši se SVD transformacija nad originalnom slikom bez žiga A dimenzija $m \times n$:

$$A = U \Sigma V^T, \quad (1)$$

gde su $U_{m \times n}$, Σ , V^T matrice od kojih je Σ dijagonalna matrica čiji su elementi singularne vrednosti od A i poredani su u opadajućem nizu, a V^T je transponovana matrica.

Korak 2: Matrici Σ doda se žig $W_{m \times n}$ i dobija se nova matrica Σ_n :

$$\Sigma_n = \Sigma + \alpha W, \quad (2)$$

gde α predstavlja faktor utiskivanja žiga.

Korak 3: Nad matricom Σ_n se vrši SVD transformacija:

$$\Sigma_W = U_W \Sigma_W V_W^T, \quad (3)$$

Korak 4: Slika sa žigom se dobija kao :

$$A_W = U \Sigma_W V^T, \quad (4)$$

Algoritam izdvajanja žiga iz slike sa šumom A^*_W vrši se na sledeći način:

Korak 1: Izvrši se SVD transformacija slike sa šumom A^*_W :

$$A^*_W = U^* \Sigma^*_W V^{*T}, \quad (5)$$

Korak 2: Korišćenjem matrica U_W i V_W dobija se :

$$D^* = U_W \Sigma_W V_W, \quad (6)$$

Korak 3: Izdvojeni žig dobija se kao:

$$W_r = \left(\frac{1}{\alpha} \right) (D^* - \Sigma), \quad (7)$$

Iz prezentovanog algoritma se vidi da je za izdvajanje žiga potrebno imati originalne matrice U_W , Σ i V_W .

3. EKSPERIMENTALNI REZULTATI I ANALIZA

3.1 Eksperiment

U cilju testiranja efikasnosti algoritma za detektovanje i eliminisanje impulsnih smetnji i otpornosti vodenog žiga na algoritam za eliminisanje impulsnih smetnji izvršen je sedeći eksperiment.

U sliku A insertovan je vodeni žig W , metodom SVD koja je predložena u radu [12], sa stepenom utiskivanja α . Nakon toga slici sa vodenim žigom A_W superponiraju se impulsne smetnje sa različitim procentom p . Nad slikom A_{wp} primenjuje se algoritam MDB za detekciju i eliminisanje impulsnih smetnji. Iz slike A'_{wp} vrši se ekstrakcija vodenog žiga. Kao mera kvaliteta filtrirane slike i izdvojenog žiga primenjuju se srednja kvadratna greška MSE, srednja apsolutna greška MAE i vršni odnos signal-šum PSNR:

$$MSE = \sum_{ij} (x_{ij} - y_{ij})^2 / MN, \quad (8)$$

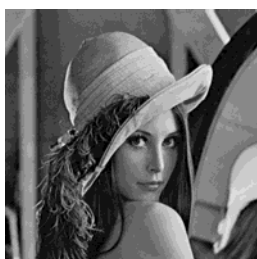
$$MAE = \sum_{ij} |x_{ij} - y_{ij}| / MN \quad (9)$$

$$PSNR = 10 \log_{10} (255^2 / MSE), \quad (10)$$

gde je x_{ij} originalna slika, y_{ij} restaurirana slika, $M \times N$ dimenzija slike.

U eksperimentu je korišćena slika Lena dimenzija 512x512 (sl.1) i žig dimenzija (512x512) (sl.2). Vrednosti

faktora utiskivanja žiga su $\alpha=0.001$, $\alpha=0.01$ i $\alpha=0.1$. Korišćene su impulsne smetnje sa $p=10-90\%$.



Slika 1. Lena (512×512)



Slika 2. Žig (512×512)

3.2 Rezultati

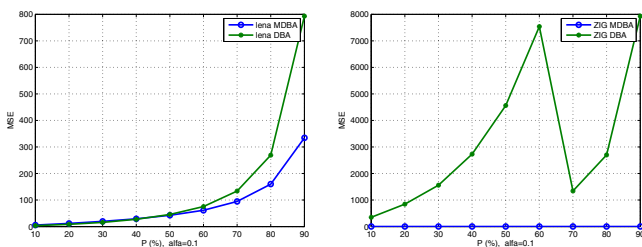
Na slikama 3-5 prikazan je odnos MSE, MAE i PSNR u zavisnosti od procenta šuma (za algoritam MDB i DB), pri faktoru utiskivanja žiga $\alpha=0.1$ za sliku Lena i za žig.

Na slikama 6 i 7 prikazani su MAE i MSE u zavisnosti od procenta šuma (za $\alpha=0.001$, $\alpha=0.01$ i $\alpha=0.1$) za sliku Lena i za žig, za primenjeni algoritam MDBA.

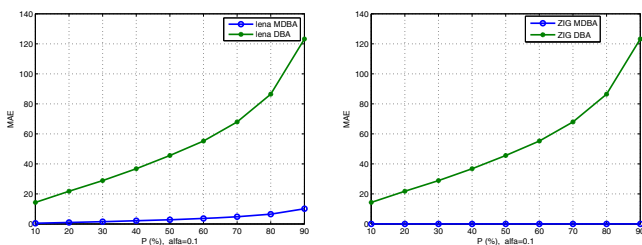
U tabeli 1. prikazana je vizualna potvrda efikasnosti primenjenog filtra u eliminaciji šuma i u čuvanju finih detalja slike.

U Tabeli 2. su prikazani parametri efikasnosti (MSE, MAE i PSNR) primenjenog filtra pri vrednosti faktora utiskivanja $\alpha=0.1$, za sliku Lena.

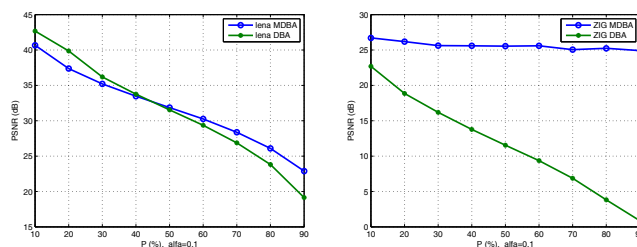
U Tabeli 3 su prikazani parametri efikasnosti (MSE, MAE i PSNR) primenjenog filtra pri vrednosti faktora utiskivanja $\alpha=0.1$, za žig.



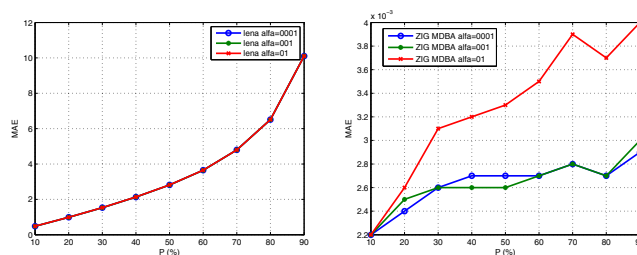
a) Lena
Slika 3. Srednja kvadratna greška u zavisnosti od procenta šuma za: a) test sliku Lena, b) žig.



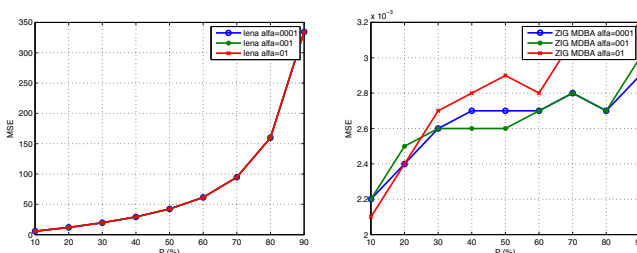
a) Lena
Slika 4. Srednja apsolutna greška u zavisnosti od procenta šuma za: a) test sliku Lena, b) žig.



a) Lena
Slika 5. Vršni odnos signal šum u zavisnosti od procenta šuma za: a) test sliku Lena, b) žig.



a) Lena
Slika 6. Srednja apsolutna greška u zavisnosti od procenta šuma, za različite vrednosti α , za: a) test sliku Lena, b) žig



a) Lena
Slika 7. Srednja kvadratna greška u zavisnosti od procenta šuma za različite vrednosti α za: a) test sliku Lena, b) žig.

Tabela 1. Vizuelni prikaz efikasnosti MDBA filtra.

Šum (%)	Sl. sa žigom i šumom	Filtrirana slika	Izdvojeni žig
10			
60			
90			

Tabela 2. MSE,PSNR i MAE za sl. Lena pri $\alpha=0.1$

Procenat šuma (%)	MSE	PSNR	MAE
10	5.5712	40.6713	0.4840
20	11.9135	37.3704	0.9889
30	19.5933	35.2097	1.5322
40	29.2194	33.4741	2.1350
50	42.3680	31.8604	2.8169
60	61.3672	30.2514	3.6501
70	94.6357	28.3703	4.7962
80	159.6946	26.0979	6.5115
90	334.2209	22.8905	10.1037

Tabela 3. MSE,PSNR i MAE za žig pri $\alpha=0.1$

Procenat šuma (%)	MSE	PSNR	MAE
10	0.0021	26.7163	0.0021
20	0.0024	26.1926	0.0024
30	0.0027	25.6205	0.0027
40	0.0028	25.5962	0.0028
50	0.0029	25.4474	0.0029
60	0.0028	25.5962	0.0028
70	0.0031	25.0452	0.0031
80	0.0030	25.2416	0.0030
90	0.0032	24.9086	0.0032

3.3 Analiza rezultata

Na osnovu rezultata prikazanih u tabelama 1-3 i grafikonima sa slika 3-7 zaključuje se da:

- primenjeni MDB algoritam pokazuje izuzetnu efikasnost kod detekcije inficiranih piksela kojima je superponiran šum, preciznost je 100% čak i pri 90% šuma,
- kvalitet rekonstruisane slike u odnosu na originalnu sliku izražen preko MSE koje je 58% manje, MAE 91% manje, dok je PSNR za 16% veće u odnosu na algoritam PAV [3],
- kvalitet rekonstruisanog žiga u odnosu na originalni žig izražen preko MSE koje je 42% manje, MAE 62 puta veće dok je PSNR 27,8% manje u odnosu na algoritam PAV,
- primenom binarizacije izdvojenog žiga mera kvaliteta MSE je $24,78 \times 10^6$ puta manje, MAE je $35,5 \times 10^3$ puta manje dok je PSNR 50,17% manje u odnosu na rezultate dobijene u radu [3],
- primenjeni MDB algoritam pokazuje bolju efikasnost u detektovanju inficiranih piksela (100% pri 90% superponiranog šuma) u odnosu na SODA algoritam [13] (99.02% pri 50% superponiranog šuma).
- kvalitet izdvojenog žiga nakon binarizacije je na veoma visokom nivou i nema velikih varijacija u zavisnosti od stepena utiskivanja žiga α .

Analiza rezultata pokazuje veliku efikasnost MDB algoritma kod eliminisanja impulsnih smetnji slika sa insertovanim vodenim žigom.

4. ZAKLJUČAK

U radu je izvršena analiza efikasnosti predloženog MDB algoritma za detekciju i otklanjanje impulsnog šuma u slici u koju je SVD transformacijom utisnut voden žig. Analiza je vršena uz variranje superponiranog šuma od $p=10-90\%$ i za vrednost faktora utiskivanja žiga $\alpha=\{0.001,0.01,0.1\}$.

Detaljna analiza parametara za meru kvaliteta slike i žiga pokazuje veliku efikasnost. Rekonstruisana slika i žig su sa

vrlo malim izobličenjem čak i pri ekstremno velikom procentu šuma $p=90\%$. (MSE=334,2209, MAE=10,1037 i PSNR=22,8905 za sliku i MSE=0,0032, MAE=0,0032 i PSNR= 24,9086 za žig). U poređenju sa PAV algoritmom opisani MDBA je pokazao bolje rezultate (MSE manje 58%, MAE 91 puta manje,PSNR veće 16% za sliku Lena i MSE manje 42% MAE manje 62 puta i PSNR manje 27,8% za žig).

LITERATURA

- [1] R.H.Chan, C-W.Ho, and M.Nikolova, „Salt and pepper noise removal by median type noise detectors and detail preserving regularization,“ *IEEE Trans.Image Process.*,vol.14, no.10,pp.1479-1485,Oct. 2005V. I. Utkin, "Variable structure control systems with sliding modes, "IEEE *Trans. Automat. Control*, AC-22, pp. 210-222, April 1977.
- [2] S.Esakkirajan, T. Veerakumar, A.N.Subramanyam, and C.H PremChand, „Removal of high density Salt and Pepper noise through modified decision based unsymetric trimmed median filter“ *IEEE Signal Processing Lettrs*, vol. 19, No5, May 2011
- [3] Nif Alajlan, Mohamed Kamel and Ed Jernigan, „Detail Preserving impulsive noise removal“, *Signal Processing: Image Communication* 19 (2004) 2017-2025
- [4] Dung Dang and WenbinLuo, „Impulse noise removal utilizing second-order difference analysis“, *Signal Processing* 87(2007) 2017-2025
- [5] Shi-Qiang Yuan and Yong-Hong Tan, „Impulsive noise removal by a global-local noise detector and adaptive median filter“,*Signal Processing* 86(2006) 2023-2028
- [6] H.G.Moreno, S.M.Bascon, M.U.Manso,P.M.Martin, „Elimination of impulsive noise in images by means of the use of support vector machines, XVI National Symposium of URSI, 2001, pp. 1-2
- [7] Y. Xu,E. M. Lae, *Restoration of images contaminated by mixed gaussian and impulse noise using a recursive minimum-maximum meth*, *Vision Image Signal Proc. IEE Proc.*145(1998) 264-270.
- [8] Z.Stevanović,Z.Milivojević, „Efikasnost nesimetričnog medijan filtra u otklanjanju impulsnih smetnji kod slika sa vodenim žigom“, *IT* 2012, Žabljak
- [9] T.Sun and Y.Neuvo, „Detail-preserving median based filters in image processing“ *Pattern Recognit. Lett.*,Vol.15,pp.341-347, 1994
- [10]C.A.Pomalaza-Racz and C.D.Mackillemm „An adaptive non linear edge preserving filter“, *IEEE Trans.Acoust.,Speech,Signal Process.*,vol.ASSP-32,pp.571-576, Jun 1974
- [11]K.S.Srinivasan and D. Ebenezer, „A New Fast Efficient Decision-Based Algorithm for Removal of High-Density Impulse Noises“, *IEEE signal Processing Letters*, vol.14, No.3, March 2007
- [12] R.liu,T.Tan, „A SVD –based watermarking scheme for protecting rightful ownership“, *IEEE Trans. Multimedia* 4 (1) (Maerch 2002) 121-128.
- [13] Z.Milivojevic,Z.Stevanovic, “Analysis on the Robustness SVD-Based Watermarking Algorithms”, *IJCIT* (ISSN:2279-0764)Vol.02-issue 04, July 2013.

UTICAJ LINEARNOG OSVETLJENJA NA NIVO DETALJA I ENTROPIJU SLIKE THE INFLUENCE OF LINEAR BRIGHTNESS ON THE LEVEL OF DETAILS AND IMAGE ENTROPY

Ratko Ivković, Mile Petrović, Petar Spalević, Dragiša Miljković, *Fakultet Tehničkih Nauka, Kosovska Mitrovica, Srbija*
Boris Gara, *Mup Republike Srbije, Beograd*

Sadržaj: U ovom radu je primenom softverskog paketa Matlab izvršena analiza nad slikama sa različitim linearnim osvetljenjem. Dobijene slike su analizirane na osnovu algoritama za određivanje nivoa detalja 2D diskretne kosinusne transformacije, wavelet transformacije i entropije. Dobijeni rezultati treba da daju uvid u promenu entropije i nivoa detalja u zavisnosti od porasta linearnog osvetljenja. Rezultati analize rada su prikazani grafički.

Abstract: In this paper, by using the software package Matlab, an analysis of images with different linear light is done. Given images are analyzed on the basis of algorithms for determining the level of details 2D Discrete Cosine Transform (DCT), Wavelet Transform and Entropy. Obtained results should provide an insight into the change of entropy and the level of details depending on the increase in linear light. The analysis results are presented graphically.

1. UVOD

Slika se sastoji od dve komponente, boje i osvetljenja. Osvetljenje kao jedna od najbitnih parametara slike, zauzima značajnu poziciju prilikom stvaranja slike. Međutim, puno korisnija i jeftinija varijanta bi bila kada bi se softverski rešio problem osvetljenja. Odnosno kada bi definisanje osvetljenja u studiju zamenio softver. Samim tim takav sistem bio bi puno jeftiniji, jednostavniji i brži.

Rezultati ovog rada treba da pomognu u kvalitetnijoj definiciji uticaja nivoa detalja i entropije na linearno osvetljenje. Ovaj rad se fokusira na promenu osvetljenja po pravilu linearne funkcije. Linearno osvetljene ima isti uticaj na sve piksele u slici koja se obrađuje. Teorijski govoreći, ako se linearno osvetljenje poveća za 25 jedinica, osvetljenje svih piksela slike poveća će se za 10%.

Veliku primenu u definisanju objekata u okviru statičke slike ima detekcija ivica. U okviru visokofrekventnog dela spektra slike najvažniju ulogu imaju ivice. Objekat i slika su potpuno prepoznatljivi samo na osnovu ivica. Kombinacijom više prepoznatljivih objekata na određenom području $M \times N$ (pri čemu su M i N vrednosti horizontale i vertikalne kod slike) definiše se slika. U kombinaciji sa detekcijom ivica slike u kome se izdvajaju objekti na digitalnim slikama, mogu se definisati različiti stepeni osvetljenja za svaki objekat ponaosob.

2. OSVETLJENJE

Osvetljenje (brightness) je jedna od najvažnijih karakteristika piksela, ona kao takva predstavlja svojstvo vizuelne percepcije u kojoj izvor deluje kao da isijava ili reflektuje određenu količinu svetlosti. Ona je neizostavni deo mnogih algoritama za obradu slika, kao što su npr. contrast ili shadow/highlight. U ovom trenutku ne postoji usvojena opšta

formula za izračunavanje osvetljenja. Stoga se može desiti da se u jednom alatu za obradu slika koristi nekoliko različitih mera osvetljenja. Ove razlike mogu varirati i do desetostruko većih vrednosti, zavisno od metode izračunavanja [1].

Uobičajeno je da se izraz Osvetljenje koristi samo za nekvantitativne reference psiholoških osećaja i opažanja svetlosti. Tako Wyszecki i Stiles određuju osvetljenje kao svojstvo vizuelnog osećaja prema kojem se dati vizuelni nadražaj (stimulus) čini manje ili više snažan, ili prema kojem se oblast u kojoj se vizuelni nadražaj javlja čini da emituje manje ili više svetlosti, i daje raspon osvetljenja od "jarko" do "zamašnjeno". Programeri algoritama za digitalnu obradu slika su primorani da nađu način da kvantitativno izraze osvetljenost. Ipak, u ovom trenutku ne postoji konvencionalni numerički opis ove karakteristike vizuelnih nadražaja [2].

U ovom radu se analiziraju neki od najpopularnijih metoda za predstavljanje osvetljenja, takođe se razmatra delotvornost tih vrednosti u algoritmima za obradu slika koji su u visokoj meri zavisni od izbora načina merenja osvetljenja [3]. Analizirane slike su osvetljene različitim stepenima pozitivne i negativne osvetljenosti korišćenjem softverskog paketa Matlab. Pozitivna osvetljenost odgovara rastućoj osvetljenosti u slici, dok negativna osvetljenost odgovara opadajućoj osvetljenosti u slici.

Osvetljenost slike je kontrolisana kroz parametar osvetljenosti - Brightness parameter (BP). Tako da povećanje osvetljenosti BP na skali od 1 do 255 (kod pozitivne osvetljenosti), i smanjenje osvetljenosti od -1 do -255 (kod negativne osvetljenosti), kao i 0, odgovaraju vrednosti izvorne slike. Promena osvetljenosti, kao i promena BP, je direktno povezana sa promenom vrednosti piksela izvorne slike. Dakle, vrednost osvetljenosti kod izvorne slike je uzeta za nultu vrednost, i u odnosu na nju se definiše pozitivno i negativno osvetljenje [4].

3. ALGORITMI ZA ODREĐIVANJE NIVOA DETALJA I ENTROPIJE

3.1. Nivo detalja

Nivo detalja je dat kao količina promene vrednosti piksela posmatrane slike. Što je verovatnoća promene veća, to je i nivo detalja viši. Stoga, veliki nivoi detalja odgovaraju velikim promenama, dok mali nivoi detalja odgovaraju malim promenama.

Logično, ovo svojstvo nije konstantno na slici, te se, zbog te varijacije, mora definisati lokalno. Izračunavanje se radi na taj način što se posmatrana slika razlaže u tri kategorije: visok nivo detalja (HL), srednji nivo detalja (ML) i nizak nivo detalja (LL).

Analiza je urađena kroz dve najpopularnije transformacije: 2D Wavelet Transformacija (DWT) uz pomoć CDF9/7 Wavelet i 2D Diskretna Kosinusna Transformacija (DCT). Dobijeni rezultati se raščaljuju u četiri kvadranta: (1) gornji levi na nižim frekvencijama duž obe dimenzije, (2) gornji desni, (3) gornji levi sa višim frekvencijama duž obe dimenzije i (4) donji desni kvadrant sa višim frekvencijama duž obe dimenzije.

Posle ovoga se pristupa izračunavanju srednje vrednosti amplitude, pri čemu se prave novi kvadranti: DCT u kvadrantu (1) (dctd); DCT u kvadrantima (2) i (3) (dctm); DWT u kvadrantu (1) (dwtd) i DWT u kvadrantima (2) i (3) (dwtm) [5].

3.2. Entropija

Iz teorije verovatnoće, a uzevši modele za matematičke izvore koje je definisao Shannon, proizilazi tretiranje podataka proizvedenih od nekog izvora kao slučajnih promenljivih. Takav informatički pojam Shannon-ove entropije je diskretna slučajna promenljiva X sa svojom verovatnoćom prostiranja (distribucije) $p_x=(p_1,p_2,\dots,p_n)$. Promenljiva X je definisana kao:

$$H(X) = H(p_x) = \sum_{i=1}^n p_i \log\left(\frac{1}{p_i}\right). \quad (1)$$

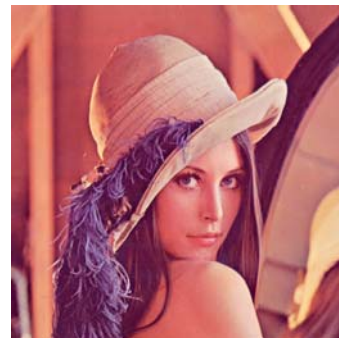
Izraz $\log 1/p_i$ izražava koliko je neizvesnosti povezano sa ishodom odgovarajućeg scenarija. Osim toga, posmatrajući pomenuti ishod, može se proučavati iznos pristiglih informacija. Stoga, može se reći da je entropija mera statističke sredine neizvesnosti, odnosno informacije [6].

Tada se može izvesti sledeće pravilo, utemeljeno na tome da funkcija p je neprekidna onda kada se ta funkcija povećava sa n . Entropija neke dve slučajne promenljive jeste zbir tih pojedinačnih entropija. Uzevši K_s za minimalnu vrednost, računanje entropije se tada vrši kao $H(X)$ i $H(X)+1$ [7].

Maksimum entropije se dostiže u slučajevima kada je verovatnoća uzoraka jednaka, tj. $H((p_1, p_2, \dots, p_n)) \leq \log n$ [8].

4. SISTEM MODEL

Za analizu u radu je korišćena originalna slika u TIFF [9] formatu, rezolucije 512x512, sa 96 dpi po horizontalnoj i vertikalnoj rezoluciji i sa 24 bita dubine, u RGB načinu zapisa, prikazna na slici 1. Format TIFF je nekompresovani TIF format tako da se najrelevantniji rezultati analize mogu dobiti obradom formata slike nad kojim nije izvršena kompresija [10].



Slika 1. Originalna slika.

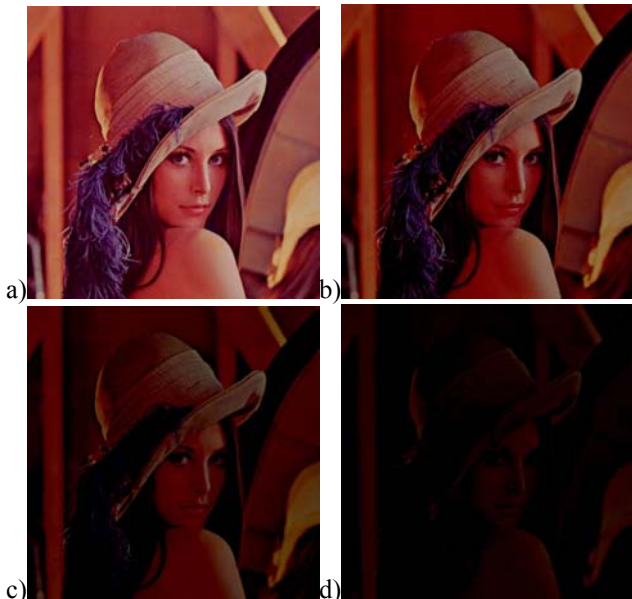


Slika 2. Dobijene slike nakon uvećanja pozitivnog osvetljenja za a)50, b)100, c)150, d)200.

Nad originalnom slikom je primenom koda u softverskom paketu Matlab linearno povećano i smanjeno osvetljenje, kako je definisano u radu [4]. Nad dobijenim slikama su primenjeni algoritmi za detekciju nivoa detalja i entropije.

Korišćeni algoritmi za detekciju nivoa detalja su 2D diskretna kosinusna transformacija i 2D wavelet transformacija.

Na slici 2 i slici 3 su prikazane dobijene slike sa različitim stepenom pozitivnog i negativnog osvetljenja, respektivno.



Slika 3. Dobijene slike nakon uvećanja negativnog osvetljenja za a)-50, b)-100, c)-150, d)-200.

5. REZULTATI

Na grafikonu 1 prikazan je odnos entropije za linearni porast osvetljenja (plavo – pozitivno, crveno - negativno). Sa grafikona se može uvideti da sa porastom vrednosti linearnog osvetljenja dolazi do pada entropije slike.

Iako se radi o linearnom porastu osvetljenja, ne može se isto reći da postoji linearan pad entropije. Već entropija sporije opada od porasti osvetljenja. Vrednosti i pozitivnog i negativnog osvetljenja se nalaze iznad vrednosti linearne funkcije.

Vrednosti entropije na posmatranom intervalu, pokazuju da vrednost linearnog osvetljenja pozitivnog i negativnog „čuvaju“ veći stepen korisne informacije od pretpostavljenog stepena.

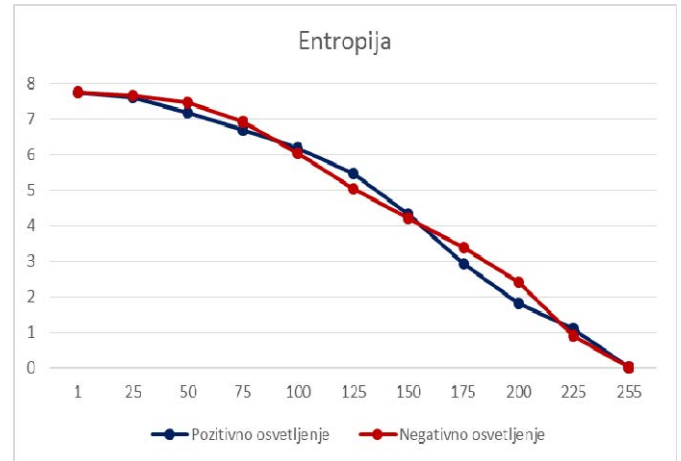
Vrednosti entropije su gotovo identične duž celokupnog posmatranog opsega vrednosti pozitivnog i negativnog osvetljenja.

Maksimalna moguća vrednost entropije je 8 bita, tako da je izabrana slika koja ima približnu vrednost entropije 8, za vrednost osvetljenja 0.

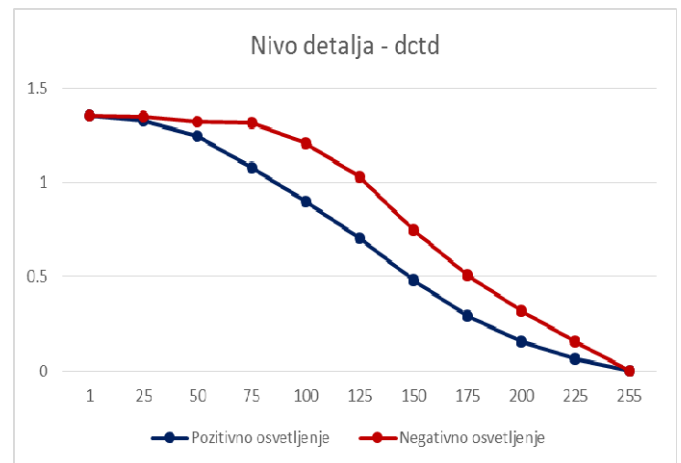
Sa grafikona 2 i 3 uočava se da vrednosti nivoa detalja, 2D kosinusne transformacije i wavelet transformacije opadaju sa porastom linearnog osvetljenja.

Pri nižim stepenima osvetljenja nivo detalja kod negativnog osvetljenja gotovo je nepromenjen, dok je kod pozitivnog osvetljenja uočljiv rapidan pad. Vrednosti nivoa detalja za dctd i wvd pri visokim linearnim osvetljenjem gotovo da su jednake nuli.

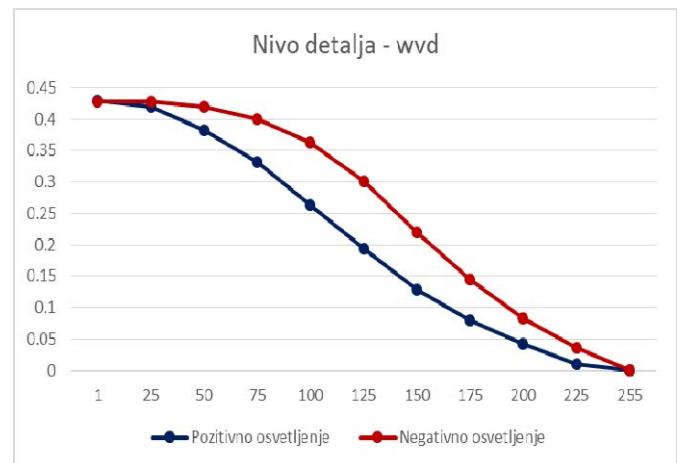
Iako dobijene slike daju iste vrednosti za entropiju, nivo detalja pokazuje da negativno linearno osvetljenje pokazuje bolje rezultate tokom analize. Izrazito bolje rezultate pokazuje 2D diskretna kosinusna transformacija. Oba algoritma transformacije daju isti oblik grafikona na posmatranom opsegu.



Grafikon 1. Odnos entropije za različite stepene linearnog osvetljenja.



Grafikon 2. Odnos nivoa detalja dctd za različite stepene linearnog osvetljenja.



Grafikon 3. Odnos nivoa detalja wvd za različite stepene linearnog osvetljenja.

6. ZAKLJUČAK

Iako se radi o povećanom linearnom osvetljenju ne dobija se potpuno linearan pad entropije, što predstavlja dobru informaciju, jer rekonstruisana slika „čuva“ više korisne informacije. Vrednosti entropije se isto ponašaju za pozitivno i negativno osvetljenje.

Međutim, kod definisanja nivoa detalja je drugačija situacija. Puno bolje rezultate pokazuje negativno osvetljenje za razliku od pozitivnog, i prilikom ove analize kako pokazuju rezultati poželjno je koristiti 2D diskretnu kosinusnu transformaciju. Iz ovoga se može izvući zaključak da pomeranje spektra ka crnoj boji (0 vrednost), odnosno ka levom delu spektra, „čuva“ više korisne informacije.

ZAHVALNOST

Ovaj rad rađen je u okviru istraživanja na projektima TR 35026 i III 47016 Ministarstva za nauku i tehnološki razvoj Srbije.

LITERATURA

- [1] S. Bezryadin, P. Bourov, D. Ilinih, Brightness Calculation in Digital Image Processing, In. Proc. *Symposium on Technologies for Digital Photo Fulfillment*, pp. 10-15(6), 2007.
- [2] S. Bezryadin, P. Bourov, Color Coordinate System for Accurate Color Image Editing Software, In. Proc. *The International Conference Printing Technology SPb 06*, pp. 145–148, 2006.
- [3] B. Jaksic, B. Gara, R. Ivkovic, M. Petrovic, M. Milošević, Analysis brightness effect on quality pictures after compression with JPEG and SPIHT compression method, *Matematicke i informaticke tehnologije*, September 2013.
- [4] R. Ivkovic, B. Jaksic, P. Spalevic, LJ. Lazic, M. Petrovic, Experimental Images Analysis with Linear Change Positive and Negative Degree of Brightness, In. Proc. *WSEAS International Conference on Image Processing and Pattern Recognition (IPPR '13)*, pp. 116-120, December 2013.
- [5] S. Ilic, M. Petrovic, B. Jaksic, P. Spalevic, Lj. Lazic, M. Milosevic, "Experimental analysis of picture quality after compression by different methods", *Przegląd elektrotechniczny*, R. 89 NR 11, pp. 190-194, 2013.
- [6] E. R. Dougherty, *Random Processes for Image and Signal Processing*, New York:Wiley-IEEE Press, 1998.
- [7] J. Beirlant, EJ. Dudewicz, L. Györfi, E. van der Meulen, "Nonparametric entropy estimation: an overview", *Intern. J. Math. Stat. Sci.*, 6(1), pp. 17–39, June 1997.
- [8] SR. Cloude, E. Pottier, "An entropy based classification scheme for land applications of polarimetric SAR", *IEEE Trans. on Geoscience and Remote Sensing*, 75, pp. 68–78, 1997.
- [9] Draft international standard ISO/DIS 12639, *Graphic technology — Prepress digital data exchange — Tag image file format for image technology (TIFF/IT)*, International Organization for Standardization, 2002.
- [10] Adobe Developers Association, *TIFF - Revision 6.0*, Adobe Systems Incorporated CA 94039-7900, pp. 13-17, June 3, 1992.

PREPOZNAVANJE DUŽICE OKA I UPOTREBA BIOCAM VISTA FA2 IRIS RECOGNITION AND THE USAGE OF THE BIOCAM VISTA FA2

Alija Dervić, Nedeljko Lekić, *Elektrotehnički fakultet Podgorica*

Sadržaj: Zbog svoje izražene individualnosti, dužica oka se sve više koristi kao biometrijski identifikator. Danas se smatra da je biometrijska tehnika identifikacije dužice u mogućnosti pružiti veći nivo pouzdanosti nego druge biometrijske identifikacione tehnike. Razvijeno je više različitih metoda i algoritama za sve faze postupka prepoznavanja dužice i na njihovom usavršavanju se i dalje intezivno radi. U cilju predstavljanja jednostavnog identifikacionog sistema za prepoznavanja dužice, u ovom radu je upotrijebljena biometrijska kamera VISTA FA2 BIOCAM i odgovarajuća PC aplikacija.

Abstract: Due to its highly distinguished individuality, the iris is increasingly used as a biometric identifier. Nowadays, it is considered that the iris biometric identification techniques are able to provide a higher level of reliability than the other biometric identification techniques. A variety of methods and algorithms for all stages of iris recognition have been developed and on their further development is still been worked intensively. For the purpose of a presentation of a simple identification system based on iris recognition, in this paper, a biometric camera VISTA FA2 BIOCAM and the corresponding PC application are used.

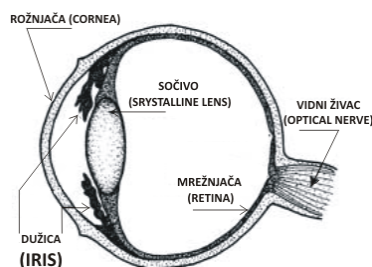
1. UVOD

Identifikacioni sistemi služe da prepoznaju korisnika i omogućе ostvarenje njegovih prava i obaveza. Danas postoji puno različitih identifikacionih sistema.

Osim tradicionalnih, koji podrazumijevaju dodjeljivanje korisniku identifikatora oblika kartice, privjeska i slično, postoje i biometrijski identifikacioni sistemi. Biometrija predstavlja automatizovan metod utvrđivanja identiteta osobe na osnovu njenih fizioloških karakteristika kao što su otisak prsta, dužica (iris) lice, glas, mrežnjača, DNK, vaslukarni obrasci, potpis, itd.

Dužica ili iris je upečatljiv dio oka. Ona sadrži pigment koji određuje boju oka. Posebnim mišićnim mehanizmom dužica reguliše količinu svjetlosti koja upada u oko i time igra ulogu dijafragme oka.

Na Slici 1, prikazan je uprošćeni poprečni presjek ljudskog oka.



Slika 1. Uprošćeni prikaz poprečnog presjeka ljudskog oka

Kao što se vidi na slici, dužica se nalazi između rožnjače i sočiva. To je unutrašnji organ čovjeka koji se vidi spolja.

Počinje se formirati oko tri mjeseca nakon začeća. Struktura linija i šara koje joj daju prepoznatljivost se formiraju do osmog mjeseca [1].

Šara dužice je individualna u svakog čovjeka. Različita je i kod jednojajčanih blizanaca. Šara dužice lijevog i desnog oka iste osobe je takođe različita.

Neke od prednosti identifikacije dužice u odnosu na druge biometrijske metode su:

- Fiziološka reakcija dužice na svjetlo. Može se iskoristiti kao prirodan test za detekciju falsifikata. Nešto slično ne posjeduje nijedna druga biometrijska identifikaciona metoda;
- Zaštićenost dužice od spoljašnje sredine. Dužica se, za razliku od otiska prsta, ne može ispljati ili izgrebati;
- Prepoznavanje dužice se može obaviti i sa rastojanja do 1m. Ovo predstavlja prednost u odnosu na tehnologiju identifikacije otiska prsta ili mrežnjače, kod kojih je neophodan kontakt sa skenerom;
- Šara dužice je nezavisna od genetskog porijekla i ne mijenja se sa protokom vremena. Ovo predstavlja prednost, u prvom redu, u odnosu na tehnologiju prepoznavanja lica.

Tehnologija prepoznavanja dužice oka suočava se i sa specifičnim poteškoćama. Neke od njih su:

- Prečnik dužice je svega oko 1cm, pa njeno skeniranje zahtijeva striktnu saradnju korisnika;
- Skeniranje dužice otežava njena pokretljivost;
- Povremeno spuštanje očnih kapaka ometa proces skeniranja.

Ideju da se dužica koristi za identifikaciju čovjeka prvi put je predložio oftomolog Frank Burch 1936. Ideja se 1980-ih godina pojavljivala samo u filmovima o Jamesu Bondu, ali i dalje se radilo o naučnoj fantastici. 1986. dva oftomologa

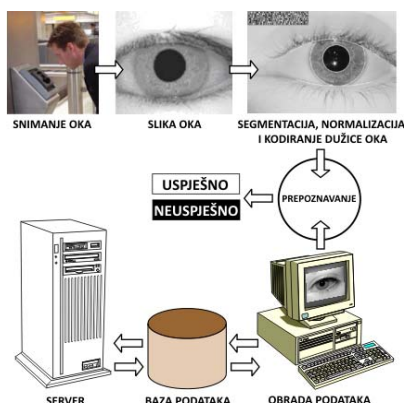
Aran Safir i Leonard Flom patentirali su tu ideju, a 1989. obratili su se Johnu Daugmanu sa Univerziteta u Cambridge-u sa predlogom da razvije algoritme za identifikaciju dužice. Algoritmi, koje je Daugman publikovao i patentirao 1993. godine, su prvi i dosad najviše korišteni u ovoj oblasti [2].

2. POSTUPAK PREPOZNAVANJA DUŽICE OKA

Postupak prepoznavanja dužice oka se može podijeliti u nekoliko faza, i to:

- Slikanje dužice,
- Segmentacija dužice oka,
- Normalizacija dužice oka,
- Dobijanje koda i
- Uspoređivanje kodova

Prva faza je slikanje oka. Tom prilikom korisnik se postavlja ispred skenera (kamere za slikanje dužice) (Slika 2). Odgovarajućom indikacijom skener pomaže korisniku da što lakše zauzme optimalni položaj.



Slika 2. Faze u postupku prepoznavanja dužice oka.

Nakon slikanja, u fazi segmentacije, iz dobijene slike izdvaja se područje dužice. Područje dužice se zatim normalizuje, svodenjem na predefinisane dimenzije. Normalizacijom se onemogućuje da na postupak prepoznavanja utiču promjene u veličini, poziciji i orijentaciji dužice. U postupku kodiranja, iz normalizovanog prikaza dužice, izdvaja se jedinstveni digitalni kod (IrisPrint). Kod se zatim uspoređuje sa ostalim kodovima iz baze podataka.

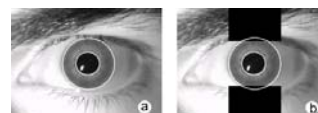
2.1. Slikanje dužice

Na početku postupka prepoznavanja, mora se slikati dužica oka. Slikanje dužice nije jednostavan postupak i zahtijeva striktnu saradnju korisnika. Jedan od razloga za to je veličina dužice. Dužica je relativno mala (u prosjeku prečnika oko 1.1cm). Da bi se skeniranje uspješno obavilo potrebno je obezbijediti odgovarajuće okolno osvjetljenje, ne prejako da bi smetalo korisniku. Poželjno je da dužica bude u centru fotografije. Koliko je to moguće, iz slike je potrebno eliminisati nepotrebne detalje. Tako na primjer, treba obezbijediti da trepavice u što manjoj mjeri zaklanjaju područje dužice. Kako se iznad dužice nalazi vlažna prozirna opna dolazi do refleksije svjetlosti. To se na slici obično vidi

kao odsjaj, odnosno bijela površina. Ta površina se ne može upotrijebiti za dobijanje koda dužice. Treba nastojati da odsjaj bude u području zjenice oka (Slika 3-a).

2.2. Segmentacija dužice oka

Nakon dobijanja slike oka, potrebno je izdvojiti region dužice. Region dužice se može aproksimirati jednim kružnim prstenom, čija unutrašnja granica je između zjenice i dužice, a spoljašnja između dužice i beonjače oka. U procesu, segmentacije osim izdvajanja dužice, uklanjaju se i smetnje, kao što su trepavice, očni kapci i odsjaj (Slika 3-b).



Slika 3. Segmentirana dužica oka (a), segmentirana dužica oka i uklanjanje smetnji (b)

Daugman-ov metod koristi poseban integralno-diferencijalni operator za lociranje dužice, kao i lukova gornjih i donjih očnih kapaka. Taj operator je definisan kao:

$$\max_{(r, x_0, y_0)} \left| G_{\sigma}(r) * \frac{\partial}{\partial r} \oint_{r, x_0, y_0} \frac{I(x, y)}{2\pi r} ds \right|$$

pri čemu je $I(x, y)$ slika oka koje se posmatra (Slika 3-a)). Operator traži, u domenu slike, maksimum parcijalnog izvoda, normalizovanog konturnog integrala od $I(x, y)$, duž kružnog luka ds poluprečnika r , sa centrom u (x_0, y_0) . Simbol $*$ označava konvoluciju. $G_{\sigma}(r)$ je Gauss-ova funkcija.

Operator traži kružnu putanju u kojoj je promjena vrijednosti piksela maksimalna, mjenjajući pri tom poluprečnik r kao i (x_0, y_0) poziciju centra kruga. Operator se primjenjuje više puta sa postepenim smanjivanjem vrijednosti parametra σ funkcije $G_{\sigma}(r)$. Smanjivanjem vrijednosti parametra σ dobija se preciznija lokacija irisa. Očni kapci se pretražuju na sličan način, s tom razlikom što putanja više nije kružna već polu-kružna [3].

2.3. Normalizacija dužice oka

Nakon uspješne segmentacije dužice, pristupa se fazi u kojoj se transformiše regija dužice, tako da ima fiksne dimenzije. Normalizacija treba da obezbijedi neosjetljivost na veličinu, položaj i orijentaciju dužice na slici, kao i na veličinu zjenice. Prilikom normalizacije, treba imati u vidu da region zjenice nije koncentričan u odnosu na dužicu, nego uvijek malo pomjeren.

Daugmanov *rubber sheet* model normalizacije transformiše sliku dužice u polarni koordinatni sistem i obezbjeđuje da na prepoznavanje nema uticaj veličina zjenice i same slike. Daugmanov model pridružuje svakoj tački dužice, bez obzira na njenu veličinu i dilataciju zjenice, par bezdimenzionalnih realnih koordinata (r, θ) . Parametar r je

na intervalu $[0,1]$ dok je θ ugaona veličina na intervalu $[0, 2\pi]$ [4], [5].

2.4. Dobijanje jedinstvenog koda dužice oka

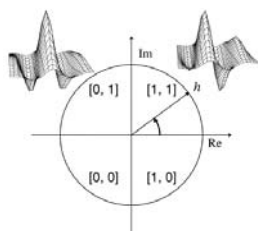
Jedinstveni kod dužice se dobija demodulacijom normalizovane slike dužice. Za ekstrakciju strukture dužice kao niza fazora (vektora u kompleksnoj ravni) koristi se banka kompleksnih 2D Gabor wavelet-a [2]. Kvantizovanjem ugla fazora postavljaju se bitovi koda dužice. Postupak kvantizacije sastoji se od identifikacije kvadranta kompleksne ravni u kojem se nalaze rezultujući fazori, dobijeni projekcijom područja dužice na kompleksne 2D Gabor wavelet-e (Slika 4). Postupak kvantizacije se opisuje sljedećom relacijom:

$$h_{\{Re, Im\}} = \text{sgn}_{\{Re, Im\}} \int_{\rho} \int_{\phi} I(\rho, \phi) e^{-j\omega(\theta_0 - \phi)} \cdot e^{-(r_0 - \rho)^2 / \alpha^2} e^{-(\theta_0 - \phi) / \beta^2} \rho \cdot d\rho \cdot d\phi$$

$h_{\{Re, Im\}}$ se može smatrati kompleksnim bitom čiji je realni i imaginarni dio 1 ili 0, zavisno od znaka 2D integrala; $I(\rho, \phi)$ je normalizovana slika dužice u bezdimenzionom polarnom koordinatnom sistemu; α i β su skalari Gauss-ove envelope 2D Gabor wavlet-a; ω je wavlet frekvencija; i (r_0, θ_0) predstavljaju polarne koordinate svakog regiona dužice za koji su izračunate fazorske koordinate $h_{\{Re, Im\}}$ [6].

Na opisan način, generiše se 2048 fazorskih bitova, odnosno 256 okteta. U novijim algoritmima, izračunava se i isti toliki broj maskirajućih bitova. Maskirajući bitovi definišu da li je određeni region dužice prekriven kopcima, zahvaćen refleksijom ili neupotrebljiv iz bilo kog drugog razloga.

U opisanom postupku je upotrijebljena samo fazna informacija jer amplitudna nije u toj mjeri diskriminišuća i zavisi od eksternih faktora kao što su kontrast slike, osvjetljenje i pojačanje kamere. Druga prednost upotrebe fazne informacije je što ona ostaje definisana bez obzira na jačinu kontrasta u slici dužice. To dalje znači da slabo fokusirane slike dužica ne mogu dovesti do konfuzije.



Slika 4. Kvantizacija ugla fazora na kvadrant kompleksne ravni i kodiranje

2.5. Upoređivanje kodova

Za upoređivanje kodova dužica *Daugman*-ov metod koristi *Haming*-ovo rastojanje. *Haming*-ovo rastojanje daje mjeru koliko je identičnih bitova u dva uzorka irisa.

Korišćenjem *Haming*-ovog rastojanja donosi se odluka o tome da li su dva uzorka dobijena iz različitih irisa ili iz jednog istog. *Haming*-ovo rastojanje je mjera različitosti i definiše se kao:

$$HD = \frac{\|(\text{code}A \otimes \text{code}B) \cap \text{mask}A \cap \text{mask}B\|}{\|(\text{mask}A \cap \text{mask}B)\|},$$

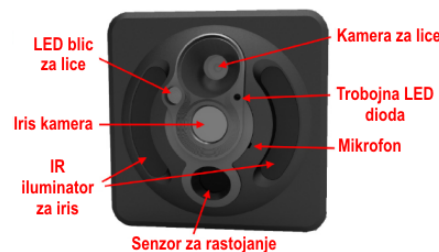
gdje su vektori kodova dužica označeni sa *codeA* i *codeB* i odgovarajući vektori maski sa *maskA* i *maskB*. Imenilac sadrži broj bitova kodnih vektora koji su od važnosti za poređenje, dok se u broiocu izračunava broj bitova od važnosti, u vektorima *codeA* i *codeB*, koji se nijesu podudarili. Rezultujuća distanca je frakciona mjera nepodudarnosti kodova dužica.

$HD=0$ označava potpuno podudaranje. Ukoliko su dva uzorka nezavisna, kao na primjer uzorki generisani od različitih irisa, *Haming*-ovo rastojanje će biti oko 0.5. Ovo stoga što nezavisnost podrazumijeva potpuno slučajnu vrijednost bita, tako da je vjerovatnoća da će dva bita iz različitih IrisPrint-ova dužica imati istu, odnosno različitu, vrijednost 50% [7].

3. VISTA FA2 BIOCAM I PC APLIKACIJA

Jednostavni identifikacioni sistem za prepoznavanje dužice sastoji se iz biometrijske kamere *VISTA FA2 BIOCAM* i PC aplikacije.

Biometrijska kamera *VISTA FA2 BIOCAM* može se koristiti u identifikacionim sistemima za prepoznavanje dužice i sistemima za prepoznavanje lica. Na Slici 5 prikazana je kamera i naznačeni njeni osnovni djelovi. Sa slike se uočava da *VISTA FA2* sadrži iris kameru, ogledalo, infracrveni (IR - InfraRed) iluminator, trobojnu LED, senzor za rastojanje, kameru za lice, LED blic za lice i mikrofona. Kamera za slikanje dužice sadrži CMOS senzor sa rezolucijom 640x480 piksela. Prilikom slikanja dužice korisnik treba da se postavi prema kameri tako da vidi svoje oko u ogledalu kamere. Na osnovu podataka senzora za detekciju rastojanja, trobojna LED pomaže korisniku prilikom pozicioniranja. Crvena boja signalizira korisniku da je preblizu, plava da je predaleko, dok zelena indicira pravi položaj. Za slikanje dužice kamera koristi infracrveno svjetlo. IR iluminator obezbjeđuje odgovarajuću osvjetljenost dužice.



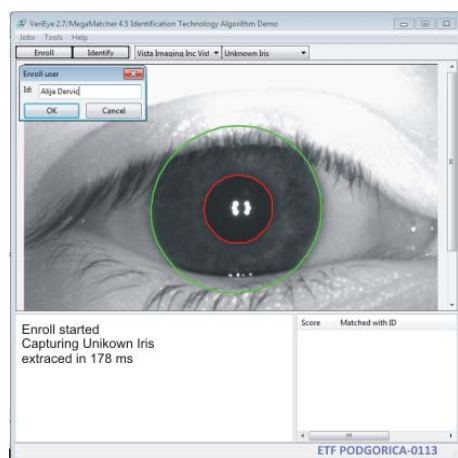
Slika 5. Biometrijska kamera Vista FA2

PC aplikacija može da radi u dva režima i to:

- režim upisivanja korisnika i
- režim identifikacije.

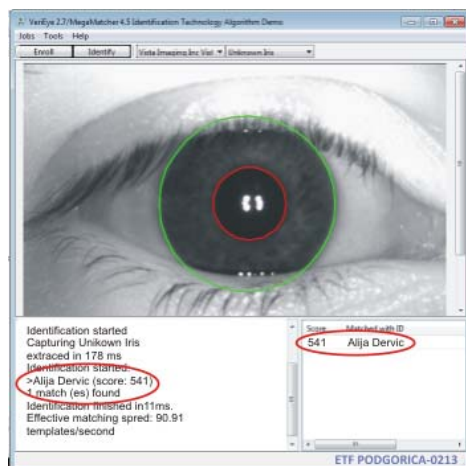
Režim rada se bira klikom na odgovarajuće komandno dugme na formi aplikacije.

Kada je potrebno da se korisnik upiše u bazu, aplikacija se prevodi u režim upisivanja. Korisnik, na već opisani način, zauzima poziciju ispred kamere, koja obavlja slikanje. Nakon slikanja, PC aplikacija obrađuje dobijenu sliku. Izdvaja dužicu iz nje (prostor između crvenog i zelenog kruga na Slici 6, normalizuje izdvojeni region i ekstrahuje jedinstveni digitalni kod dužice (IrisPrint). Dobijeni IrisPrint zatim smješta u bazu podataka, zajedno sa unesenim podacima korisnika.



Slika 6. Aplikacija u režimu upisivanja korisnika

Kada je potrebno izvršiti prepoznavanje, aplikacija se prevodi u režim identifikacije. U ovom režimu, dobijeni IrisPrint se upoređuje sa IrisPrint-evima iz baze podataka. Ukoliko je korisnik prepoznat, u donjem desnom uglu aplikacije ispisuju se njegovi podaci (Slika 7). Ukoliko korisnik nije prepoznat, prostor u donjem desnom uglu ostaje prazan.



Slika 7. Primjer uspješne identifikacija korisnika

U okviru aplikacije je moguće podešavati brzinu detekcije (sporo, srednje, brzo), kao i vrijednost FAR (False Accept Rate) u granicama 0.1% do 0.0001%.

Pouzdanost algoritma aplikacije testirana je upotrebom ICE2005 Exp1 baze slika dužice. Baza sadrži 1425 slika dužice i 124 jedinstvene desne dužice oka. Testiranja su pokazala da se FAR=0.001%, dobija FRR (False Rejesct Ratio)= 0.0942 %.

4. ZAKLJUČAK

U radu je predstavljen jednostavan identifikacioni sistem za prepoznavanje dužice oka. Dati su osnovni podaci o sastavnim djelovima sistema, biometrijskoj kameri Vista FA2 i PC aplikaciji.

U okviru rada izvršeno je upoređivanje biometrijske metode prepoznavanja dužice, sa ostalim biometrijskim metodama. Ukazano je na bitne prednosti metode prepoznavanja dužice, ali i na poteškoće sa kojima se suočava.

Izložen je i postupak prepoznavanja dužice, kroz opis svake pojedine faze postupka.

LITERATURA

- [1] P. Kronfeld, "Gross anatomy and embryology of the eye," in *The Eye*, H. Davson, Ed. London, U.K.: Academic, 1962.
- [2] Daugman J. , "How Iris Recognition Works" IEEE Transaction on Circuits and System for Video Technology , vol . 14, no. I, pp . 21- 30,2004.
- [3] Surjeet Singh, Kulbir Singh, "Segmentation Techniques for Iris Recognition System," International Journal of Scientific & Engineering Research, vol. 2, Issue 4, April-2011, ISSN 2229-5518.
- [4] J. Daugman, *Biometric personal identification system based on iris analysis*. United States Patent, Patent Number: 5,291,560, 1994.
- [5] M. R. Bendre, S. A. Shivarkar, "An improved approach of IRIS authentication system using Daugmans rubber sheet model, segmentation ,normalization and RSA security algorithm. " IJCTEE, vol.1, issue3.
- [6] J. Daugman, "Demodulation by complex-valued wavelets for stochastic pattern recognition," Journal of Wavelets Multiresolution and Information Processing Vol. 1, No. 1 (2003), 1-17.
- [7] J. G. Daugman, "High con dence visual recognition of persons by a test of statistical independence, " IEEE Trans. Pattern Anal. Machine Intel. 15 (1993), 1148-1161.

UTICAJ PUNJENJA VOZILA NA ELEKTRIČNI POGON NA KVALITET ELEKTRIČNE ENERGIJE U DISTRIBUTIVNOJ MREŽI

ELECTRIC VEHICLE CHARGERS INFLUENCE ON QUALITY OF ELECTRICAL ENERGY IN THE DISTRIBUTION GRID

Luka Čađenović, *Elektrotehnički fakultet - Podgorica*

Sadržaj: *Posljednjih nekoliko godina sve se više pažnje posvećuje ekologiji. Ozonski omotač koji je veoma bitan za ljudski opstanak znatno je oštećen i to predstavlja veliki problem. Svakako je to posljedica velikim dijelom zbog koncentracije izduvnih gasova. Drugi veoma važan faktor je da su ograničeni resursi za proizvodnju fosilnih goriva pa se automobilska industrija sve više okreće izradi električnih automobila koji bi doprinijeli očuvanju životne sredine i riješili problem energije. U ovom radu je ukratko predstavljena problematikom negativnog uticaja rada punjača električnih vozila na kvalitet električne energije u distributivnoj mreži.*

Abstract: *In the last few years, increasing attention has been paid to ecology. The ozone layer, which is very important for human survival is significantly damaged and it is a big problem. Certainly this is a result in large part due to the concentration of exhaust gases. Another very important factor is that the limited resources for the production of fossil fuels and the automotive industry is increasingly turning to making electric cars that could contribute to the preservation of the environment and solve the problem of energy. This paper briefly introduced the issue of the negative impact of electric vehicle chargers work on the quality of electrical energy in the distribution grid.*

1. UVOD

1.1. Električna vozila

Električno vozilo je automobil koji se pokreće elektromotorom, koristeći električnu energiju uskladištenu u akumulatoru, ili drugim uređajima za čuvanje energije. Električni automobili imaju nekoliko mogućih prednosti u odnosu na konvencionalne automobile sa unutrašnjim sagorijevanjem, koje uključuju značajno smanjenje zagađivanja vazduha u gradovima, jer oni ne ispuštaju zagađivače iz svojih izvora energije tokom rada, smanjene emisije gasova staklene bašte, zavisno o gorivu i tehnologiji koja se koristi za proizvodnju električne energije za punjenje akumulatora, manju zavisnost o nafti, što je u razvijenim zemljama i zemljama u razvoju uzrok zabrinutosti zbog njihove izloženosti naglim promjenama cijene i poremećaja u zalihama.

Električni automobili su svakako u apsolutnoj prednosti u odnosu na automobile koji koriste fosilna goriva. Sve više proizvođača u svoj proizvodni program uvodi model hibridnog ili potpuno električnog automobila.

Ovo su pokazatelji da će u bližoj budućnosti zavladati električna vozila, prvenstveno zbog same cijene goriva i zbog nepostojanja emisije štetnih gasova, tačnije "Zero emission" (nulte emisije).

Proces prelaska sa fosilnih goriva na električnu energiju kao osnovno pogonsko gorivo može donijeti negativan uticaj na distributivni sistem.

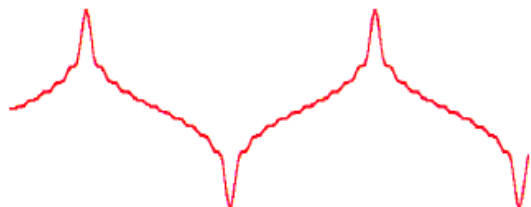
Negativan uticaj koji stvaraju punjači električnih vozila prilikom rada je evidentan, zato što spadaju u grupu nelinearnih opterećenja, koja dovode do stvaranja viših harmonika struja i napona koji negativno utiču na naponske prilike u distributivnoj mreži.

1.2. Pojam i uzroci pojave viših harmonika

Harmonici se pojavljuju kod bilo koje opreme čija struja ne prati vjerno sinusni oblik napona. Većina elektronske opreme koristi jednosmjerno napajanje, tako da koristi AC/DC ispravljač. Ako nema poseban dodatni aktivni sklop, ispravljač vuče struju u seriji kratkih impulsa.

Monofazna nelinearna opterećenja, kao npr. ispravljači, generišu neparne harmonike (3,5,7,9, itd.). Za monofazne sisteme, problematični su 3. harmonik i njegovi neparni sadržaoci (9,15, itd.). Ti se harmonici nazivaju „tripelni“ (utrostručitelji) jer su ti harmonici R,S i T faze međusobno u fazi. Tako se sabiraju, umjesto da se poništavaju, na neutralnom vodu 3-faznog 4-vodnog sistema. Ovo može dovesti do preopterećenja neutralnog voda ako nije projektovan za takvu vrstu opterećenja. Dodatno, takvi harmonici uzrokuju kružeće struje na trougaonom namotaju transformatora konfiguracije trougao-zvijezda. Kada harmonici preko neutralnog voda 3-faznog 4-vodnog sistema dođu do transformatora reflektuju se na primarni namotaj (trougao) gdje kruže. Rezultat toga je zagrijavanje transformatora slično onom kod nesimetrične 3-fazne struje. Jedan od značajja strujnih harmonika je i visoki vršni faktor.

Vršni faktor bilo kojeg talasnog oblika je odnos vršne i efektivne vrijednosti. Kod savršenog sinusnog talasa ta je vrijednost 1,414. Vrijednost različita od 1,414 pokazuje izobličenje talasnog oblika. Tipično izobličeni strujni talasni oblici imaju vršne faktore veće od 1,414 (slika 1), a izobličeni naponski oblici faktore manje od 1,414. Izobličeni naponski talasni oblici s vršnim faktorom manjim od 1,414 nazivamo „ravnog vrha“. Pravougaoni talasni oblik ima vršni faktor 1,0.

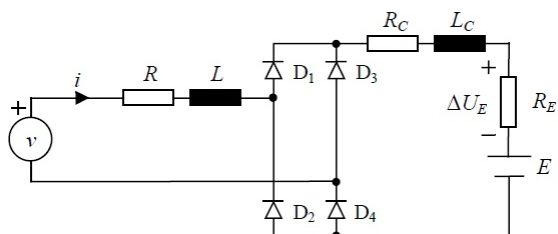


Slika 1 Tipičan talasni oblik struje za monofazno nelinearno opterećenje sa elektronskim napajanjem.

2. PUNJAČI

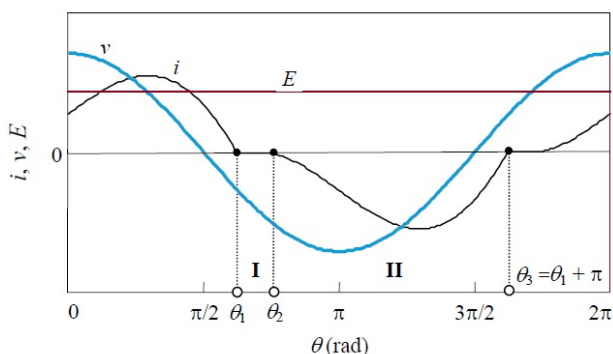
2.1. Monofazni punjači

Na slici 1 je ilustrovan prikaz karakterističnog ekvivalentnog kola monofaznih punjača. Sastoji se od monofaznog ispravljačkog diodnog mosta sa AC induktivitetom L (zajedno sa pripadajućim redno vezanim otporom R), otpora R_E , induktiviteta L_E sa strane punjenja i elektromotorne sile E (zajedno sa pripadajućim redno vezanim otporom R_E). Induktivitet L_E je obično manjih vrijednosti od induktiviteta L .



Slika 1. Ilustracija ekvivalentnog kola monofaznog punjača

Kao primjer ponašanja punjača slika 2 pokazuje izgled naizmjenične struje "i" (podrazumijevajući da je $L_E \ll L$). Naizmjenični napon "v" i napon baterije E su takođe prikazani na istoj slici.



Slika 2. Prikaz AC struje, AC napona i DC napona baterije

Ovo kolo radi na principu isprekidanog provođenja, gdje razlikujemo dva moguća stanja:

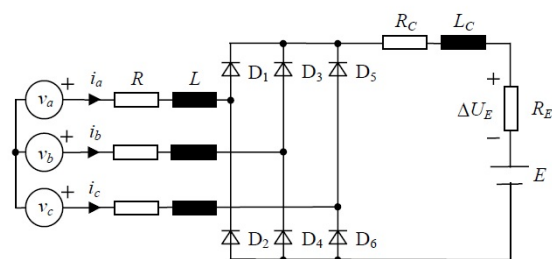
- Dioda ne provode i struja "i" je jednaka nuli (Segment I na slici 2)
- Struja "i" teče kroz ispravljački diodni most i puni bateriju (Segment II na slici 2)

Komutacioni uglovi θ_1 i θ_2 , koji definišu prelaz polutalasnog simetrije sinusoide naizmjenične struje punjača su prikazani na slici 2. Sa slike se veoma jasno vidi da je u segmentu I ($\theta_1 < \theta < \theta_2$) struja jednaka nuli, dok to ne važi za segment II ($\theta_2 < \theta < \theta_3 = \theta_1 + \pi$).

2.2. Trofazni punjači

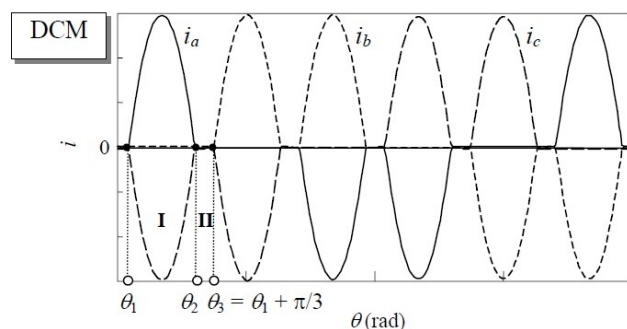
Na slici 3 je ilustrovan prikaz karakterističnog ekvivalentnog kola trofaznog punjača. Sastoji se od trofaznog ispravljačkog diodnog mosta sa tri AC induktiviteta L (zajedno sa pripadajućim redno vezanim otporima R), otpora

R_E , induktiviteta L_E sa strane punjenja i elektromotorne sile E (zajedno sa pripadajućim redno vezanim otporom R_E). Induktivitet L_E je obično manjih vrijednosti od induktiviteta L .



Slika 3. Ilustracija ekvivalentnog kola trofaznog punjača

Kao primjer ponašanja punjača u isprekidanom modu (Discontinuous conduction mode - DCM) slika 4 pokazuje izgled naizmjenične struje sve tri faze i_a, i_b i i_c (podrazumijevajući da je $L_E \ll L$).



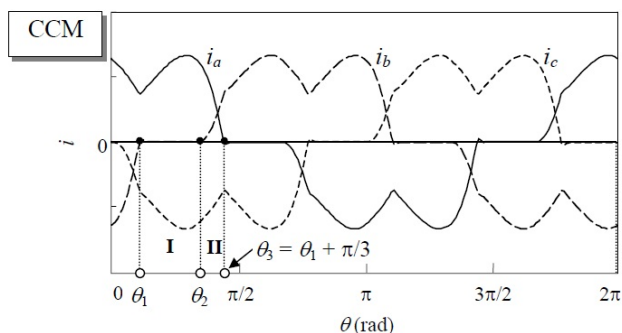
Slika 4. Prikaz struja sve tri faze u DCM modu

Dok kolo radi u DCM modu, dva segmenta se mogu razlikovati:

- Struje teku kroz ispravljački diodni most i pune bateriju (Segment I na slici 4),
- Dioda ne provode i struje su jednake nuli (Segment II na slici 4).

Komutacioni uglovi θ_2 i θ_3 , koji definišu prelaz polutalasne simetrije sinusoide naizmjenične struje punjača su prikazani na slici 4. Sa slike se veoma jasno vidi da je u segmentu II ($\theta_2 < \theta < \theta_3 = \theta_1 + \pi/3$) struja jednaka nuli, dok to ne važi za segment I ($\theta_1 < \theta < \theta_2$).

Kao primjer ponašanja punjača u neprekidnom modu (Continuous conduction mode - CCM) provođenja, slika 5 pokazuje izgled naizmjenične struje sve tri faze i_a , i_b i i_c (podrazumijevajući da je $L_c \ll L$).



Slika 5. Prikaz struja sve tri faze u CCM modu

Dok kolo radi u CCM modu, dva segmenta se mogu razlikovati:

- Struja protiče kroz jednu od neparnih dioda, puni bateriju, potom se vraća kroz jednu parnu diodu (Segment I na slici 5),
- Struja protiče kroz jednu od preostale dvije neparne diode, puni bateriju, potom se vraća kroz jednu od preostale dvije parne diode (Segment II na slici 5).

Komutacioni uglovi θ_1 , θ_2 i θ_3 , koji definišu prelaz polutalasne simetrije sinusoide naizmjenične struje punjača su prikazani na slici 5. Sa slike se veoma jasno vidi da u segmentu I ($\theta_1 < \theta < \theta_2$) imamo samo struju i_a , dok u segmentu II ($\theta_2 < \theta < \theta_3 = \theta_1 + \pi/3$) imamo dodatno i struju i_b istovremeno.

2.3. Ublažavanje struja harmonika rednim prigušnicama

Metoda za ublažavanje je bazirana na povećanju induktiviteta sa strane naizmjeničnog napajanja dodavanjem rednih prigušnica u svaku od faza. Ovaj vid ublažavanja dovodi do širenja sinusoide naizmjenične struje i smanjenja razmaka između komutacionih uglova što dovodi do smanjenja struja harmonika (male vrijednosti induktiviteta sa strane naizmjeničnog napajanja rezultuju povećanjem oštine sinusoide struje što povlači porast struje harmonika).

2.4. Uticaj punjenja nekoliko pojedinačnih vozila na stvaranje harmonijske distorzije

Da bi se verifikovala prethodna izlaganja problematike ublažavanja pomoću induktivnih prigušnica, vršila se mjerenja u realnom vremenu uz pomoć mrežnog analizatora AR5-L proizvođača CIRCUTOR (slika 6).



Slika 6. Mrežni analizator CIRCUTOR model AR5-L

Ovaj uređaj ima mogućnost snimanja mrežnih parametara (aktivnu i reaktivnu snagu, faktor snage, napon, jačinu struje, itd.) u realnom vremenu i njihovo kasnije reprodukcije u vidu grafičkih i brojnih prikaza na računar ili na sopstveni LCD ekran. Ima mogućnost priključivanja na monofazno i trofazno napajanje, a korisno je to što se može podesiti da snima duži vremenski period da bi uzorak za kasniju analizu bio kvalitetniji.

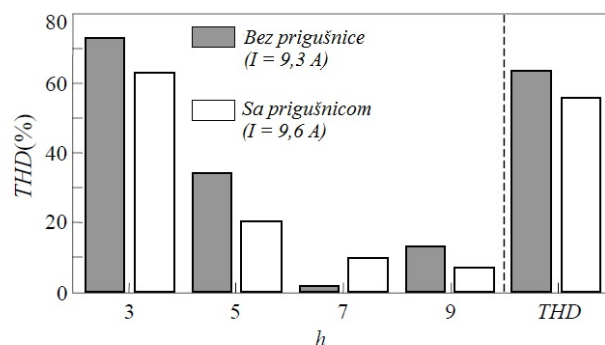
Testirana instalacija je bila garaža koja sadrži 20 monofaznih i 5 trofaznih mjesta za punjenje vozila, test vozila su bila Piaggio Porter koji se napaja sa monofazne utičnice i Fiat Ducato koji se napaja sa trofazne utičnice.[2]

Moramo napomenuti da je snaga punjenja Piaggio Portera 2,2kW, a Fiat Ducata 6,6kW.[2]

Punjači ovih vozila su testirani na dva načina:

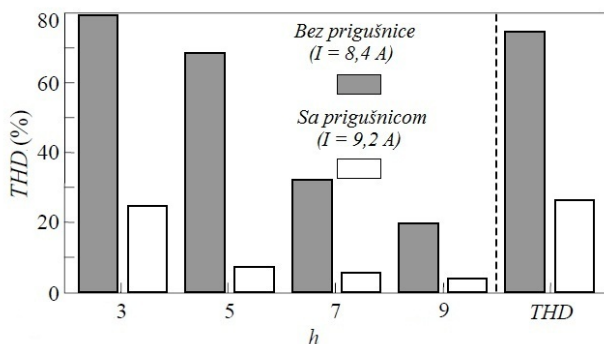
- Direktno priključeni na napajanje
- Priključeni na mrežu preko redne prigušnice induktivnosti 3mH koja se povezuje između napajanja i punjača

Rezultati mjerenja su dati kao uporedna grafička analiza THD prije i poslije ubacivanja prigušnice u kolo punjenja.



Slika 7. Mjerenja uticaja prigušnice na THDi kod monofaznog punjača

Na slici 7 se jasno vidi da je uticaj prigušnica kod monofaznih punjača neznatan, tako da stavljanje istih u svrhu ublažavanja THD nema efekta.



Slika 8. Mjerenja uticaja prigušnice na THDi kod trofaznog punjača

Na slici 9 možemo vidjeti da za razliku od monofaznih punjača ovdje ubacivanje prigušnica u kolo ima veliki uticaj na smanjenje THD što je i bio cilj.

Ova tehnika ublažavanja THD je opravdana samo u slučaju trofaznih punjača, dok je potpuno neučinkovita kod monofaznih punjača.

2.5. Uticaj na distributivni transformator

Prilikom povećanja opterećenja transformatora dolazi do njegovog zagrijavanja, međutim prilikom našeg razmatranja ne smijemo zaboraviti dodatno zagrijavanje zbog uticaja struja harmonika koje se javljaju zbog punjača. Neravnomjerna distribucija struje u namotajima i zalutali gubici od elektromagnetnog fluksa u oblastima gdje to nije bila namjera imaju za posljedicu dodatne gubitke koji se manifestuju kao dodatno zagrijavanje. Dakle životni vijek samog transformatora je ugrožen, odnosno smanjen. Da bi se smanjili uticaji ovog tipa koji dovode do smanjenja životnog vijeka distributivnog transformatora ublažavaju se smanjenjem opterećenja, poznato kao smanjenje performansi transformatora. IEEE standard C57.110¹ štiti transformator tako što propisuje faktor smanjenja performansi zavisno od vrste opterećenja i THD. Smanjenje performansi transformatora imaju razne negativne efekte koji se manifestuju kao manji stepen iskorišćenja, problem podešavanja relejne zaštite itd.

3. ZAKLJUČAK

Kroz ovaj rad je prikazano nekoliko osnovnih stvari o električnim vozilima, stanicama za punjenje i njihovom uticaju prilikom rada na kvalitet električne energije u distributivnoj mreži.

Pošto će u bližoj budućnosti električnih vozila biti sve više, ova tema se nameće kao nezaobilazna. Zbog ukupnog porasta broja automobila, doći će do povećanja uticaja na kvalitet električne energije u distributivnoj mreži, pa se mora obratiti posebna pažnja na ove negativne uticaje.

Negativni efekti koji će se odraziti na distributivnu mrežu sistema izazvani povećanjem broja električnih vozila će biti manifestovani kroz značajno povećanje opterećenja mreže i uvećanje totalne harmonijske distorzije.

Da bi se umanjio problem stvaranja totalne harmonijske distorzije koristila se tehnika dodavanja redne prigušnice, što se pokazalo kao veoma djelotvorno kad su trofazni punjači u pitanju, dok se kod monofaznih pokazalo kao potpuno beskorisno.

Treba se osvrnuti na sve pomenute uticaje i metode za ublažavanje istih, jer ih i je nemoguće ukloniti u potpunosti. Ovo znači da će ova problematika uvijek biti prisutna i aktuelna, pogotovo u bližoj budućnosti kad se broj ovih vozila poveća drastično, pa se moraju tražiti alternativna rješenja u vidu pametnih mreža, ili obnovljivih izvora energije koji bi preusmjeravali svoju proizvedenu energiju direktno na mjesta za grupno punjenje vozila ili jednostavno napajali samo ta mjesta i time oslobodili distributivnu mrežu ovih negativnih uticaja.

LITERATURA

- [1] Hessamoddin Moghaddam, *Influence of electric vehicle charging rates on transformer derating in harmonic-rich battery charger applications*, Archives of electrical engineering vol. 61(4) (2012), 483-497.
- [2] Mohd Zamri Che Wanik, Afida Ayob i Azah Mohamed, *Harmonic measurement and analysis during electric vehicle charging*, Scientific research open access Journal 5 (2013), 215-220.
- [3] Hydro-Québec, *Electric vehicle charging stations*, Vancouver (2012)
- [4] J. Balcells, *Impact of plug-in electric vehicles on the supply grid*, Pres. E.T.S.E.I.B. Technical University of Catalonia (UPC), Barcelona (2012).

¹ C57.110-1986 - IEEE Preporučena praksa za utvrđivanje sposobnosti transformatora kad snabdijeva nesinusoidalna strujna opterećenja

INDEKS AUTORA

A

Abazović Dejan	112	Dlabač Tatijana	108, 143
Angelkov Dimitrija	173	Draganić Anđela	128, 232
Apostolov Ilija	124	Drašković Borko	44
Asanović Marko	236	Duli Sidita	33

B

Balanesković Dragiša	201
Bauk Sanja	143
Bešić Nikola	240
Bogojević Pero	84
Bojičić Tamara	154
Bojović Risto	100
Božović Mirjana	158
Brodić Darko	56, 244

DŽ

Džankić Radoje	143
----------------	-----

Đ

Đordan Bojan	162
Đukanović Milena	177
Đurović Igor	5, 236

C

Chingoski Vlatko	205
Čitkuseva Dimitrovska Biljana	205

Č

Čađenović Luka	256
Čičević Svetlana	92

Ć

Ćalasan Martin	150
Ćulum Vladimir	80

D

Daković Novica	21
Damjanović Dalibor	96
Dervić Alija	252

F

Femić Balša	25
Filipović Dragan	240
Filipović Luka	29
Filipović Nemanja	48

G

Gadžović Almir	76
Gara Boris	248
Gelev Sašo	124, 181
Golubovski Roman	52, 166
Grbović Ana	162
Grebović Marko	228

H

Hristov Risto	124
Husak Ermin	177

I

Ivanović Igor	60
Ivanović N. Veselin	40
Ivković Ratko	248

J

Jevtović Miloško	193
Jovanović Ana	1, 13
Jovanovski Srđan	40

K

Kadić Srđan	60, 224
Karabegović Isak	177
Knežević Ivan	209, 217
Končar Jelena	116
Kordić Stevan	108
Kosanović Miloš	88
Kosanović Mirko	88
Kovač Draško	108
Krstajić Božo	5, 9, 29, 197
Kukusheva Maja	205

L

Lazović Luka	13
Lekić Nedjeljko	64, 76, 252
Leković Sonja	116
Lutovac Budimir	112, 240

Lj

Ljucović Jelena	213
-----------------	-----

M

Martinovska Bande Cveta	173
Medenica Milica	232
Milivojević Zoran	56, 170, 201, 244
Miljanić Igor	189
Miljković Dragiša	248
Mirković Bogdan	135, 139
Mirković Jasna	84
Mosurović Milenko	224
Mujović Saša	17, 158

N

Nešić Milkica	92
---------------	----

O

Obradović Milovan	120
Ognjanović Ivana	213
Orović Irena	128, 232
Orović Miloš	128

P

Parezanović Duško	221
Pejaković Tamara	128
Pekić Nađa	108
Pekić Željko	108
Perišić Vuko	104
Petrović Mile	248
Popović Tomo	197
Popović-Bugarin Vesna	154
Prlinčević Bojan	244

R

Radonjić Milutin	64, 68, 72
Radović Jadranka	17
Radulović Milovan	21, 150
Radunović Petar	209, 217
Radusinović Igor	64, 68, 72, 76
Raković Predrag	132, 147
Rašović Mladen	17
Ristić Aleksandar	96
Rubežić Vesna	1, 13, 150

S

Samčović Andreja	92
Savić Nataša	56
Sokolovski Aleksandar	181
Spalević Petar	248
Stanković Srđan	232, 240
Stijepović Vasilije	132
Stojanović Radovan	48, 236
Stojanović Saša	170

Š

Šabotić Rabina	72
Šandi Stevan	197
Šćepanović Stevan	25, 228
Šendelj Ramo	213
Šuh Jelena	80

T

Tomić Miladin	64
---------------	----

Tomović Slavica	68	Veličković Zoran	193, 201
Tošić Dragan	170	Veljović Zoran	36
Trifunović Aleksandar	92	Vidaković Dragan	221
		Vučeraković Aleksandar	185
		Vujičić Tijana	209, 217
U			
Urošević Uglješa	36	Z	
		Zečević Žarko	5, 9
		Zuković Sanja	232
V			
Vasile Gabriel	240		

CIP - Каталогизација у публикацији
Национална библиотека Црне Горе, Цетиње

ISBN 978-86-85775-15-4
COBISS.CG-ID 24749840

ISBN 978-86-85775-15-4



9 788685 775154 >