

IT'10
ŽABLJAK

XV

naučno-stručni skup

INFORMACIONE TEHNOLOGIJE

SADAŠNJOST I BUDUĆNOST

Urednici.
Božo Krstajić
Aleksandra Radulović

IT'10

**INFORMACIONE
TEHNOLOGIJE**

- SADAŠNJOST I BUDUĆNOST -

Urednici

Božo Krstajić

Aleksandra Radulović

*Zbornik radova sa XV naučno - stručnog skupa
INFORMACIONE TEHNOLOGIJE - sadašnjost i budućnost
održanog na Žabljaku od 24. do 27. februara 2010. godine*

Organizatori skupa

Ministarstvo za informaciono društvo CG

Ministarstvo prosvjete i nauke CG

Univerzitet Crne Gore - CIS

Održavanje skupa finansijski pomogli

- * Crnogorski Telekom
- * Microsoft Crna Gora
- * Informatika Montenegro

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 Milo Tomašević, Elektrotehnički fakultet, Beograd
Dr Srbijanka Turajlić, 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 Ramo Šendelj, FIT, Podgorica
Dr Stevan Šćepanović, PMF, Podgorica

Organizacioni odbor

Dr Vujica Lazović, Podpredsjednik Vlada CG

Dr Božo Krstajić, CIS Univerzitet CG

Dr Milovan Radulović, Elektrotehnički fakultet, Podgorica

Radovan Rutešić, Ministarstvo prosvete i nauke

Vladan Tabaš, Čikom

Zoran Sekulović, Informatika Montenegro

Sekretarijat

Aleksandra Radulović, CIS Univerzitet CG

P R E D G O V O R

Ovogodišnji, XV naučno-stručni skup INFORMACIONE TEHNOLOGIJE – sadašnjost i budućnost (IT'10), je otvoren temom „Iskustva korišćenja eLearning u visokom obrazovanju Crne Gore”. Prije 10-tak godina smo slušali naučno-stručne radove na ovu temu, a ove godine su prezentovani dugogodišnji rezultati i iskustva iz ove oblasti. Napredak se osjeća, jer se danas u Crnoj Gori upisuju studenti na fakultetima birajući studiranje elektronskim putem, nastava u Bijelom Polju i Kotoru održava se iz učionica u Podgorici, na mnogim fakultetima imamo elektronsko polaganje testova, nastavnici u školama mogu koristiti didaktičke materijale i razmjenjivati ih putem portala itd.

Razmjena pozitivnih iskustava i popularizacija evropskih projekata na Skupu su nastavljene i ove godine ali sa mnogo više učesnika. FP7 projekti EU su velika šansa za nauku i razvoj u Crnoj Gori i čitavoj Evropi. Predstavnicima projekta SEE-GRID-SCI (SEE-GRID eInfrastructure for regional eScience) prezentovali su najmoćniji računarski resurs jugoistočne Evrope u kojem učestvuje i akademska mreža Crne Gore (CIS UCG). Ovom računarskom resursu, koji je distribuiran i lociran na prostoru 14 država, mogu imati pristup svi korisnici Akademske mreže Crne Gore. Sistem se trenutno najviše koristi za prikupljanje, obradu i razmjenu informacija iz oblasti seizmologije, meteorologije i životne sredine. Treba napomenuti da je CERN učesnik ovog projekta i pruža stručnu pomoć ostalim učesnicima. “Kako uspješno učestvovati u FP7 ICT projektima” bio je naziv radionice koju su održali predstavnici Ministarstva prosvjete i nauke za buduće učesnike ovih projekata.

Ovogodišnja novina je bila prezentacija ICT rješenja i projekata koji se realizuju u Crnoj Gori. Predstavnicima Ministarstva za informaciono društvo prezentovali su projekat uvođenja digitalnog potpisa (CA), a prezentacija integralnog informacionog sistema – elektronske uprave opštine Bar je bila najposjećenija na Skupu. Crnogorski Telekom je predstavio novi infrastrukturni koncept baziran na optičkim vlaknima, dok su predstavnici firme Global Soft prikazali set eGovernment servisa.

Ipak, najmlađa, najmasovnija i najperspektivnija grupa učesnika je prezentovala svoje autorske radove koji se nalaze u zborniku pred vama. Radovi su prošli proces recenzije, a u zborniku su samo oni koji su prezentovani na Skupu. Najbolji studentski rad je nagradila kompanija Microsoft Crna Gora, koja je pored Crnogorskog Telekoma i Informatike Montenegro, bila sponzor ovogodišnjeg Skupa. Sve detalje o Skupu i informacije o narednom možete naći na poznatom sajtu www.it.ac.me.

Očekujemo vas i sledeće godine da, pod okriljem Durmitora, zajedno konstatujemo da je prošla budućnost postala sadašnjost, a novi radovi i generacije stvaraju novu budućnost.

Prof. dr Božo Krstajić

SADRŽAJ

Aleksandar Paunović, Božo Krstajić PRILOG ANALIZI KVALITETA TV SIGNALA U IP RADIODIFUZNIM SISTEMIMA PRENOSA	1
Dražen Žujović, Luka Filipović, Božo Krstajić KOMPARACIJA PERFORMANSI SEKVENCIJALNIH I PARALELNIH PROGRAMA NA NUMERIČKI ZAHTJEVNOM PRIMJERU	5
Ivan Vučinić ELEKTRONSKA DISTRIBUCIJA PROIZVODA I USLUGA – ELEKTRONSKE DOPUNE.....	9
Jezdimir - Luka Obadović PRIMJENA UČENJA NA DALJINU U OBRAZOVANJU	13
Maja Ilić-Delibašić, Božo Krstajić ZNAČAJ MODELOVANJA FIZIČKOG KANALA U SIMULACIJI BEŽIČNIH MREŽA	17
Miloš Perunović, Milovan Radulović SAVREMENI TRENDOVI U INDUSTRIJSKIM KOMUNIKACIJAMA	21
Sanja Bauk, Ruža Danilović PRIKAZ RADA NA CENTRALNOM ELEKTRONSKOM KATALOGU POMORSTVA CRNE GORE.....	25
Sanja Kotlaja RAZVOJ I IMPLEMENTACIJA E-GOVERNMENT-A U OPŠTINI PLJEVLJA	29
Slobodan Pavićević, Božo Krstajić SIGURNOSNO TESTIRANJE – PROVJERA SIGURNOSTI SISTEMA NA DJELU	33
Vladana Mrdak, Božo Krstajić PRIMJER PRIMJENE SISTEMA ZA MONITORING MREŽE NAGIOS.....	37
Aleksandar Dedić GRAFIČKO PRIKAZIVANJE VEKTORA ELEKTRIČNIH VELIČINA NA TFT-LCD DISPLEJU	41
Dušan Kostić, Branko Marović PROSTORNA NAVIGACIJA U ODNOSU NA OBJEKAT POSMATRANJA.....	45
Milovan Obradović MODEL PODATAKA ZA ODRŽAVANJE DOKUMENTACIJE SISTEMA KVALITETA U ZDRAVSTVENOJ USTANOV	48

Mirko R. Kosanović RUTIRANJE PODATAKA U BEŽIČNIM SENZORSKIM MREŽAMA SA ASIMETRIČNIM VEZAMA	52
Nebojša Nedić ALGORITAM ZA IZRAČUNAVANJE POZADINE.....	56
Nedjeljko Lekić, Zoran Mijanović, Radovan Stojanović PROSTI MIKROKONTROLESKI BLIZINSKI KAPACITIVNI SENZOR SA EKRA- NIZIRANOM SENZORSKOM POVRŠINOM.....	60
Nenad Krdžavac, Milenko Mosurović JEDNO PROŠIRENJE AKSIOMA INKLUZIJA VEZA SA REGULARNIM IZRAZIMA	64
Marina Milivojević, Zoran N. Milivojević DISONANTNI F# INTERVAL GOVORNOG SIGNALA KAO INDIKATOR HIPOKSIJE	67
Zoran N. Milivojević, Nataša Savić PROCENA FUNDAMENTALNE FREKVENCIJE GOVORNOG SIGNALA KOMPRIMOVANOG G.723.1 ALGORITMOM PCC INTERPOLACIJOM.....	71
Zoran Veličković, Miloško Jevtović PODRŠKA MREŽNIH SIMULATORA ARHITEKTURI PROTOKOLA PREPLETENIH SLOJEVA.....	75
Zoran Veličković, Miloško Jevtović CROSS-LAYER OPTIMIZACIJA PREDAJNE POLITIKE U KANALU SA REJLIJEVIM FEDINGOM	79
Dženan Strujić, Tripo Matijević MAVEN ALAT ZA WEB JAVA EE APLIKACIJE	83
Jelica Kasalica, Igor Ognjanović, Ranko Vojinović SIMULACIONI MODEL ALGORITMA RUTIRANJA DIJKSTRA SA ASIMETRIČNIM CIJENAMA LINKOVA MREŽE	87
Nataša Banović, Ramo Šendelj MOGUĆNOSTI PRIMJENE SERVIS-ORIJENTISANE ARHITEKTURE (SOA) U PRAKSI.....	90
Ramo Šendelj, Ivana Ognjanović, Mladen Bukilić KONCEPTUALNI MODEL UVOĐENJA ICT-A U OBRAZOVNI SISTEM.....	94
Snežana Šćepanović PRIMJENA WEB 2.0 TEHNOLOGIJA U UČENJU	98
Sonja Pravilović, Borislavka Golović Kažić PRIMJENA KOGNITIVIH MAPA U VISOKOM OBRAZOVANJU	101
Zoran Ćorović, Novak Radulović, Snežana Šćepanović KREIRANJE WEB CMS SISTEMA FAKULTETA ZA INFORMACIONE TEHNOLOGIJE PUTEM JOOMLA PLATFORME.....	104

Budimir Ugren, Milutin Radonjić, Igor Radusinović NETFPGA RAZVOJNA PLATFORMA ZA REALIZACIJU AKTIVNIH MREŽNIH KOMPONENTI.....	107
Dajana Tomić, Igor Radusinović UPOREDNA ANALIZA IZRAČUNAVANJA VIŠIH MOMENATA DUŽINE M/M/c REDA ČEKANJA	111
Danilo Mišović, Milutin Radonjić, Igor Radusinović IMPLEMENTACIJA SERIJSKE KOMUNIKACIJE SA SPARTAN 3E FPGA RAZVOJNOM PLATFORMOM.....	115
Duško Penezić DIGITALNI „MIKROKONTROLERSKI” SISTEM ZA REGULACIJU TEMPRATURE kod Solarnih Sistema za grijanje tople sanitarne vode i podrške sistemu grijanja.....	119
Jelena Čvorović, Igor Radusinović, Milutin Radonjić PERFORMANSE KOMUTATORA SA BAFERIMA U UKRSNIM TAČKAMA	123
Mimo Mirković, Miloš Daković, Milovan Radulović MOGUĆNOST PRIMJENE METODA DISKRETIZACIJE KONTINUALNIH SISTEMA NA ELEKTROMEHANIČKOM SISTEMU ECP 205.....	127
Nikola Ljumović, Milutin Radonjić, Igor Radusinović PRIMJER REALIZACIJE TCP/IP/ETHERNET KOMUNIKACIJE PC RAČUNARA I SPARTAN 3E FPGA PLOČICE.....	131
Sidita Duli, Božo Krstajić PRIMJER VALIDACIJE XML-a POMOĆU DTD-a	135
Aleksandar Radonjić, Budimir Lutovac, Miroslav Lutovac ANALIZA DIGITALNIH FILTARA KORIŠĆENJEM ALGEBARSKIH RAČUNARSKIH SISTEMA	139
Alen Pelinković, Igor Đurović, Đuro Stojanović SISTEM VIŠESTRUKIH PODNOSILACA BAZIRAN NA AFFINE FOURIEROVOJ TRANSFORMACIJI U VAZDUHOPLOVNIM KANALIMA	143
Ana Jovanović PRIMJENA MNK U ODREĐIVANJU DISTRIBUCIJE STRUJNOG TOKA DUŽ KRAKOVA ARHIMEDOVE SPIRALE	147
Boriša Bjelica, Miloš Daković, Ljubiša Stanković METOD EMPIRIJSKE DEKOMPOZICIJE U OBRADI RADARSKIH SIGNALA.....	150
Jurica A. Žarković, Anđelina Jokić, Radovan Stojanović, Marija Mirković PRIMJENA WBN TEHNOLOGIJE U KONTINUIRANOM PRAĆENJU MEDICINSKIH PARAMETARA.....	154
Marija Mirković, Radovan Stojanović, Milorad Kalezić, Budimir Lutovac, Nevena Vukčević DETEKCIJA QRS KOMPLEKSA SA ASPEKTA PRIMJENE DISKRETNE WAVELET TRANSFORMACIJE.....	158

Marko Simeunović, Miloš Daković, Igor Đurović PRIMJENA MPI ZA ODREĐIVANJE BIASA I VARIJANSE KUBIČNE FAZNE FUNKCIJE	162
Martin Čalasan, Vladan Vujičić ANALIZA UTICAJA ŠESTOPULSNOG I DVANAESTOPULSNOG ISPRAVLJAČA KOD HVDC SISTEMA NA HARMONIJSKA IZOBLIČENJA NAIZMJENIČNE MREŽE	166
Martin Čalasan, Milutin Ostojić GUI MODEL ZA ISPITIVANJE DEJSTVA STEP SMETNJE NA REFERENTNU VRIJEDNOST NAPONA GENERATORA	170
Miroslav Lutovac, Aleksandar Radonjić DIGITALNI HALFBAND IIR FILTRI SA PRIBLIŽNO LINEARNOM FAZOM	174

PRILOG ANALIZI KVALITETA TV SIGNALA U IP RADIODIFUZNIM SISTEMIMA PRENOSA

APPLICATION QUALITY ANALYSIS TV SIGNALS IN IP BROADCASTING TRANSMISSION SYSTEMS

Paunović Aleksandar RDC, Krstajić Božo *Elektrotehnički fakultet, Podgorica*

Sadržaj – U ovom radu je analiziran uticaj primjene sistema prenosa radiodifuzije baziranog na TCP/IP platformi na kvalitet prenošenog TV signala. Dato je objašnjenje i izvršena je analiza problema koji se javljaju u prenosu TV i radio signala kroz mrežu, a prouzrokovani su funkcionisanjem i radom IP mreže. Izvršena je analiza uticaja ovih problema na kvalitet prenošenog videa pomoću MDI-a (Media delivery index). Za analizu i praćenje korišćeni su uređaji Bridgetechnologies VB220 IP-PROBE.

Abstract – In this paper we are analyzing influence of use broadcasting transmission system based on TCP/IP platforms on quality of transmitted TV signals. We analyzed and explained problems which are met in transmission of TV and radio signals through network and are caused by work of IP network. Also, we analyzed influence of these problems on quality of transmitted video by MDI (Media delivery index).

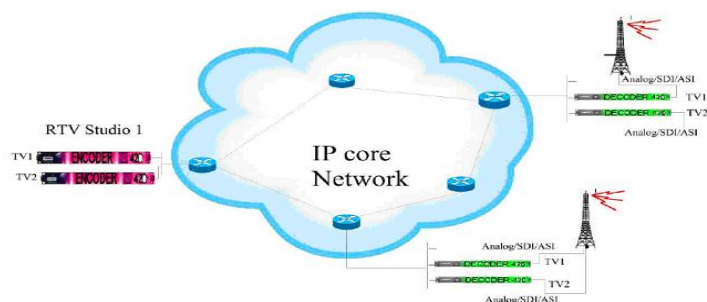
1. UVOD

Prelazak na digitalno emitovanje TV i radio programa, kao jedino moguće rješenje za prevazilaženje problema koji prate analognu televiziju, dovelo je do razvoja i implementacije novih sistema prenosa. Jedno od mogućih rješenja, a koje je primjenjeno u našoj zemlji, je izgradnja radio-relejnih veza, zasnovanih na savremenoj IP platformi, koje tako formiraju ethernet mrežu pomoću koje se signal distribuira do svakog emisionog centra. Ovakva mreža ne samo da omogućava distribuciju TV signala nego istovremeno omogućava i nadzor svih uređaja, lokacija, a takođe obezbjeđuje prenosne kapacitete drugim komercijalnim korisnicima i državnim institucijama.

Da bi ovakav sistem omogućio valjan prenos vremenski osjetljivih podataka, kao što je real-time video potrebno je garantovati potrebni propusni opseg. Velike probleme može da stvori i sama IP mreža svojim radom. Ti problemi se najčešće manifestuju preko gubitaka paketa, varijacija u vremenu dolaska paketa, dolaska paketa van svog redoslijeda. Svi ovi problemi direktno utiču na kvalitet usluge koja se pruža krajnjim korisnicima.

Mjerenjem gubitaka paketa, varijacije u dolaznom vremenu (jitter-a) moguće je dobiti vrijednost koja nam ukazuje na kvalitet prenošenog TV signala. U ovom radu za vrijednost koja karakteriše kvalitet TV signala korišćen je MDI (Media Delivery Index) index. Analiza je vršena pomoću uređaja Bridgetechnologies VB220 IP-PROBE. Za analizu je korišćen realni signal PROTV-a. Mjerenje je vršeno u tri tačke i to: na strani predaje strima, proizvoljnoj tački u sistemu i na strani emisije lokacije. Mjerenja će pokazati kakav je kvalitet videa u sve tri mjerne tačke. Iz toga ćemo moći vidjeti kakav je kvalitet signala koji se predaje, kako se mreža ponaša i kakav je kvalitet signala koji se distribuira krajnjim korisnicima.

2. MREŽA I NJENO FUNKCIONISANJE



sl.1 Blok šema prenosa TV i radio signala

Distribucija TV i radio signala vrši se korišćenjem enkodera na strani headend-a (mjesto gdje se strim TV signala ubacuje u mrežu) i dekodera na strani predajne lokacije (slika 1). Prenos podataka se može realizovati u unicast ili multicast režimu rada. Za prenos TV i radio signala do više predajnih lokacija koristi se multicast režim rada, [6], dok se za kontribuciju TV signala od mjesta snimanja do TV studia koristi unicast režim rada, [6].

Studski TV signal koji dođe u encoder komprimuje se i pakuje se u pakete MPEG Transportnog Streama od 188 ili 204 bajta [1], [2], [3]. TS sadrži sinhronizacione bite i identifikatore, paketske identifikatore PID-ove, programske informacije, video-audio streamove.

Prenos TS paketa se može vršiti na dva načina, [3]:

- Enkapsulacijom u UDP pakete pa u IP pakete
- Enkapsulacijom u RTP pakete pa onda tek u UDP i IP pakete

U prvom načinu paketi TS se direktno mapiraju u UDP pa onda u IP pakete. Najčešće se mapira sedam TS paketa što odgovara 1500 bajta Ethernet MTU(maximum transport unit).

Kada se primjenjuje drugi način prvo se TS paketi

enkapsuliraju u RTP (Real-time Transport Protocol) paket koji se kasnije enkapsulira u UDP, a onda u IP paket. Najčešće se mapira sedam TS paketa u jedan RTP paket. RTP paketi se koriste zbog bolje sinhronizacije a takođe i zbog toga što se u RTP zaglavljju nalaze 16 bitni sekvencijalni brojači koji omogućavaju jedinstvenu detekciju i brojanje odbačenih paketa što omogućava prijemniku da zna koji paket je izgubljen i da ga aproksimira pomoću interpolacije. Takođe u zaglavljju se nalazi i Timestamp polje koje definiše kad treba da se prikaže sadržaj nekog paketa.

U oba slučaja se kao transportni protokol koristi User Datagram Protokol – UDP, [6]. Osim funkcije multipleksiranja i demultipleksiranja, kao i veoma ograničene provjere grešaka UDP ništa ne dodaje IP-u. On se koristi iz više razloga a neki od njih su: nema uspostavljanja konekcije, nema praćenja stanja konekcije, malo opterećenje zaglavljja paketa, bolja kontrola na nivou aplikacije nad sadržajem i vremenom slanja. Najčešće se koristi Internet protokol verzije 4. mada je moguće koristiti i IPv6.

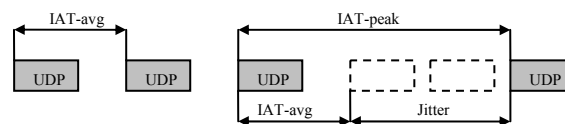
Da bi se obezbjedila redundantnost prenosa TV strama topologija mreže je napravljena u vidu prstenova tamo gdje je to bilo moguće, a gdje nije primjenjena je topologija zvijezde. Primjena topologije prstena sa sobom nosi određene probleme kao što je pojava petlji koje ugrožavaju funkcionalnost mreže. Da bi se izbjegli ti problemi na svim mrežnim uređajima diže se Spaning Tree Protocol. Kad se STP izvršava, paketi će se prosljeđivati samo na premijskim vezama koje je STP izabrao.

3. PROBLEMI KOJI NASTAJU U MREŽI

Kada se razmatra kvalitet prenosa u IP paketski baziranim mrežama vremenski osjetljivih podataka kao što je real-time video i audio, tj. televizija i radio, posebnu pažnju treba obratiti na probleme prouzrokovane funkcionisanjem i karakteristikama IP mreže. Problemi koji se javljaju su: gubici paketa, dolazak paketa van svog reda, jitter paketa, prekidi u prosljeđivanju paketa (nastali zbog stalne promjene rute ili prekida u komunikaciskim kanalima).

Prekidi u prosljeđivanju paketa [5], [7], nastaju usljed promjene rute slanja paketa ili pada linka. Prekidi u prosljeđivanju paketa se mjere veličinom koja predstavlja vrijeme koje je potrebno mreži da sama automatski obnovi servis. Promjene rute najčešće nastaju ako dođe do povremenih perioda zagušenja mreže i tada se preusmjeravanje vrši na IP sloju korišćenjem OSPF, ISIS, BGP ili nekog drugog rutinog protokola. Tada je zadovoljavajuće vrijeme trajanja prekida manje od 3 sekunde. U slučaju da do prestanka prosljeđivanja paketa dođe zbog pada linka, a u slučaju da se čvorna lokacija nalazi u prstenu, vrijeme koje je potrebno za ponovno uspostavljanje veze zavisi od STP (*Spaning Tree Protocol*). Usavršavanjem STP protokola moguće je smanjiti vrijeme reakcije na pad mreže.

Jitter IP paketa (slika 2.), [5], [7], je varijacija u IAT-*Inter arrival time* vremena u poređenju sa očekivanim IAT vremenom i izražava se u milisekundama. IAT vrijeme predstavlja vremensku razliku između dolaska dva susjedna IP paketa.



sl. 2. Jitter

Jitter događaji su najčešće prouzrokovani trenutnim odlaganjem u prosljeđivanju. Takođe veliki broj hopova može da izazove probleme, kao i smetnje i šumovi koji prouzrokuju odbacivanje paketa. Maksimalna vrijednost jittera je 50ms [5].

Problemi koje izaziva Jitter najčešće se “amortizuju“ baferisanjem podataka koji dolaze u dekodir. Zbog veoma varijabilnih vremenskih intervala nemoguće je napraviti bafer koji može da zadovolji i ispravi sve jittere, nego je moguće naći samo optimalno rješenje koje je pogodno u većini slučajeva

Problemi koji se mogu javiti kod bafera uslijed jitera su:

- Pražnjenje bafera koje nastaje zbog velikog jittera i prouzrokuje da dekodir nema šta da dekodira.
- Plavljenje bafera usljed negativnog jittera što uzrokuje odbacivanje paketa na ulazu u bafer.

Svi ovi problemi se rješavaju optimalnim izborom veličine bafera, pravilnim projektovanjem i otklanjanjem nedostataka na mreži.

Kada jitter uzrokuje velika odlaganja bafera dekodera se potpuno isprazni i dekodir nema šta da dekoduje što uzrokuje veliko „kockanje“ slike ili potpuni prestanak emitovanja programa. Pražnjenje bafera može biti prouzrokovano i velikim varijacijama u prosječnoj brzini prenosa podataka. Tako npr. Ako je MPEG video kodiran brzinom od 3,75Mb/s, a šalje se kroz mrežu brzine 3,50 Mb/s doće do pražnjenja bafera.

Slično predhodnom slučaju jitter može da uzrokuje burst (neprekidan tok) paketa što dovodi do „plavljenja“ bafera i pojave odbacivanja paketa na ulazu u bafer. Ovo je daleko ozbiljniji slučaj jer može da nastane na više mjesta na mreži a takođe da prouzrokuje odbacivanje paketa. To može da dovede do zabune jer slično se dešava i kad je propusni opseg mali. Gledaoci kao posljedicu toga vide ili sliku koja je „iskockana“ ili sliku koja nema svoj normalni tok pa izgleda kao da se pregledaju fotografije.

Reordering paketa [5], [7], predstavlja dolazak paketa van redoslijeda kojim su prvobitno poslani. Izražava se u procentualnom odnosu broja paketa koji su primljeni van svog reda u odnosu na ukupan broj primljenih paketa. Paket reordering se javlja tokom transporta paketa kroz mrežne uređaje (switcheve ili rutere) ili kad su pojedini segmenti mreže konfigurisani da retransmituju izgubljene pakete. Uticaj reorderinga paketa je veoma značajan i on zavisi od načina enkapsulacije MPEG transportnog streama. U slučaju enkapsulacije MPEG TS direktno u UDP pakete, reordering paketa će rezultovati njihovim odbacivanjem kada dođu na ulaz dekodera odnosno njegovog bafera. Odbacivanje paketa

će rezultovati smanjenjem kvaliteta slike koju vide gledaoci. Međutim ako se koristi enkapsulacija u RTP pakete pa onda tek u UTP pakete postoji mogućnost vraćanja paketa u svoje mjesto u nizu. To je sve moguće zahvaljujući sekvencionalnom brojaču koji se nalazi u zaglavlju RTP paketa. Međutim, ako paket dođe suviše kasno u bffer, tj. ako je nemoguće da se on ubaci u svoj redosljed on se neće uzimati u razmatranje i biće odbačen. Pošto svi ovi problemi koji se javljaju pri reorderingu paketa uzrokuju odbacivanje paketa kao posljedica toga javlja se „kockanje“ slike ili se javlja digitalizovani ton.

Gubitak paketa, [5], [7], je procenat IP paketa izgubljenih tokom transporta paketa od početne do krajnje tačke. Najčešći razlozi za odbacivanje IP paketa su:

- Jitter događaji koji predhode zagušenju bafera dekodera i odbacivanja paketa na ulazu u bafer.
- Šumovi i smetnje koji se javljaju na pristupnim linkovima, uzrokuju oštećenja paketa pa oni ne mogu da prođu CRC provjeru
- Greške na ruterima. Pogrešna polja checksum-a
- Ekstremno preopterećenje mreže (veoma teško da se dogodi u ovakvim sistemima, češće kod IPTV i VoD).

Gubitak paketa može da izazove jitter, a on kasnije da proizvede još veće probleme o kojima smo već govorili. S toga je veoma važno spriječiti da dođe do gubitka paketa.

4. ODREĐIVANJE KVALITETA TV I RADIO SIGNALA POMOĆU MDI INDEXA

Media Delivery Index (MDI) je numerička prezentacija dva glavna parametra koja utiču na video koji se prenosi kroz IP mrežu: odbačenih paketa i jittera paketa. MDI se mjeri tokom određenog vremenskog intervala, tipično od jedne sekunde.

MDI se prikazuje kao dvije brojne vrijednosti odvojene : (MDI (DF:MLR)), a ti brojevi su Delay Factor (DF) tj. faktor kašnjenja paketa i Media Loss Rate (MLR) tj. stopa gubitka paketa medijuma. Na primjer MDI (3,1:0) znači da je faktor kašnjenja od 3,1 milisekunde sa nula izgubljenih paketa u toku jedne sekunde.

Delay Factor (DF)-faktor kašnjenja je trenutna vrijednost izražena u milisekundama koja pokazuje koliko je vremena potrebno da se isprazni virtuelni buffer na nekom mrežnom čvoru (switch, ruter, računar, u ovom slučaju dekodera) u određenom trenutku. DF se računa tako što se prvo odredi širina virtuelnog bffera. Ona je jednaka razlici između primljenih bajtova i bajtova koji su izašli iz bffera (formula 1).

$$\Delta = \text{Priljeni bajtovi} - \text{Bajtovi poslani na obradu} \quad (1)$$

Zatim se tokom određenog vremenskog intervala (najčešće od 1 sekunde) uzima razlika između maksimalne i minimalne vrijednosti Δ . Količnik ove razlike i brzine medija (tj. njegovu bitrate) je DF, (2).

$$DF = \frac{\max(\Delta) - \min(\Delta)}{\text{media_rate}} \quad (2)$$

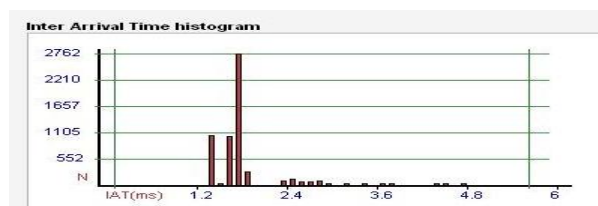
Ovakav način računanja može jedino da se primjeni na streamove konstantnog protoka. U slučaju strimova sa promjenljivim bitrateom koji ne sadrže MPEG null pakete IAT vrijeme će varirati od paketa do paketa. Zbog toga je DF ekvivalentan najvećem IAT pikou u toku mjernog perioda.

Media Loss Rate (MLR)-stopa izgubljenih paketa medija. U slučaju kad se koristi direktna enkapsulacija TS paketa u UDP računa se na taj način što se od očekivanog broja paketa medija (u ovom slučaju paketa TS) oduzme broj primljenih paketa medija u toku određenog vremenskog intervala (najčešće od 1 sekunde). U slučaju kad se koristi RTP protokol praksa je da se MLR definiše kao broj odbačenih IP paketa, tako da odbacivanje jednog IP paketa povećava MLR za 1.

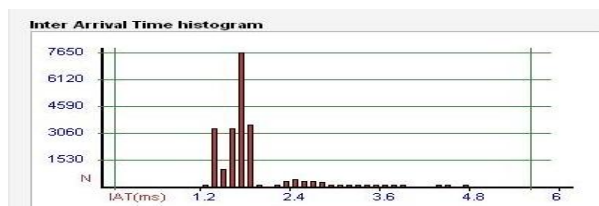
5. PRIMJER ANALIZE SIGNALA PROTV-A POMOĆU BRIDGETECHOLOGIES VB220

Pomoću uređaja Bridgetechnologies VB220 IP - Probe analiziran je kvalitet signala PRO TV-a na tri kontrolne tačke i to na strani headend-a, jednoj čvornoj tački i krajnjoj tački. TS PRO TV-a se enkapsulira u UDP/IP pakete konstantnog bitrate-a od 6Mbit/s

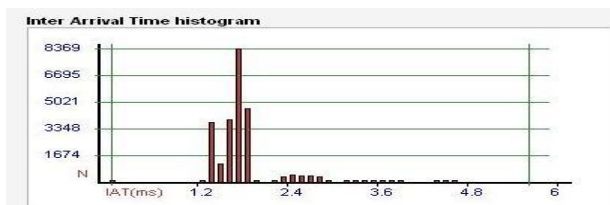
Na slikama 3, 4 i 5 vidimo histogram IAT-a. Na X osi su predstavljene vrijednosti IAT-a izražene u ms, dok je na Y osi predstavljen broj mjerenja IAT-a. Vertikalne zelene linije predstavljaju minimalnu i maksimalnu vrijednost IAT-a. Uređaj grafički prikazuje IAT pikove crvenim linijama. Dužina crvene linije direktno je proporcionalna učestalosti javljanja određene vrijednosti IAT-a, pa je tako prosječna vrijednost IAT-a najduža vertikalna crvena linija. Ostale vertikalne linije su pikovi IAT u različitim trenucima mjerenja. Oduzimanjem prosječne vrijednosti IAT-a od trenutne vrijednosti IAT-a pika moguće je dobiti vrijednost jittera u tom trenutku. Sa histograma se može zaključiti da je u svim mjernim tačkama minimalno, prosječno i maksimalno IAT vrijeme isto što nam ukazuje da je čitav sistem stabilan. Takođe vidi se da je jitter neznatan i da ne prelazi kritičnu granicu od 50ms.



sl. 3. IAT histogram na strani headend-a



sl. 4. IAT histogram na jednoj od čvornih tačaka



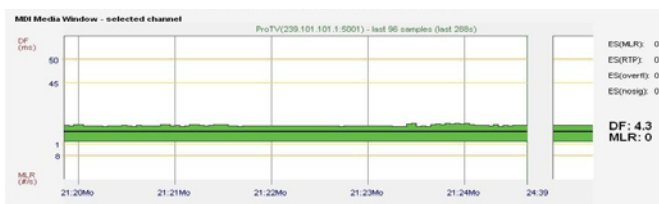
sl. 5. IAT histogram u krajnjoj tački, na predajnoj lokaciji

BRIDGE Technologies uređaji u sebi imaju software koji pored grafičke prezentacije IAT-a računa i prikazuje brojnu vrijednost IAT-a. Dobijeni rezultati su prikazani u tabeli 1, i odgovaraju prikazanim na IAT histogramima.

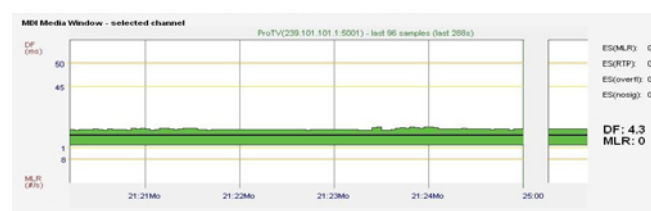
Mjerno mjesto	IAT max	IAT min	IAT avg
1	5,620ms	0,100ms	1,757ms
2	5,620ms	0,100ms	1,757ms
3	5,620ms	0,100ms	1,758ms

Tabela 1. Rezultati mjerenja IAT-a

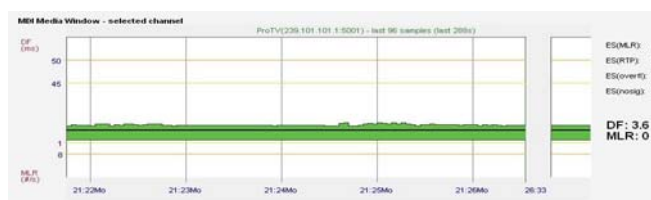
Na slikama 6, 7 i 8 prikazan su MDI dijagrami. Ovakav način prezentovanja MDI indexa koriste uređaji BRIDGE Technologies-a. Na njima su grafički reprezentovani jitter i odbačeni paketi koji se javljaju u prenosu TV i IP signala, preko DF (pozitivna Y osa) i MLR (negativna Y osa) vrijednosti. Ovakvi grafici se dobijaju tako što software vrši računsku obradu vrijednosti iz tabele 1., pomoću formula 1. i 2. Veličine koje je usvojio IETF, preko svojih RFC dokumenata, kao granice do kojih ne dolazi do promjena u kvalitetu transportovanog videa su 50ms za DF i 8 za MLR.



sl. 6. MDI dijagram na strani headend-a



Sl. 7. MDI dijagram na jednoj od čvornih tačaka



Sl. 8. MDI dijagram u krajnjoj tački, na predajnoj lokaciji

Sa MDI dijagrama vidimo da nema odbačenih paketa jer je u sve tri mjerne tačke MLR=0. Praćenjem MDI-DF dijagrama vidimo da je jitter minimalan te da njegova vrijednost ne prelazi 6ms pa samim tim ne utiče na kvalitet transportovanog videa. Kada bi grafički oduzeli MDI-DF dijagrame jedan od drugog vidjeli bismo kolika je vrijednost jittera koju je unio sam koder u sistem u odnosu na jitter koji je sama mreža indukovala. Saznanje o tome koliki jitter unosi sam koder omogućava uvid u ispravnos samog koder. Sve vrijednosti preko propisanih ukazuju da mreža slabi kvalitet video streama kojeg prenosi.

6. ZAKLJUČAK

U ovom radu analiziran je uticaj sistema prenosa signala za radio-difuziju, baziran na IP platformi, na stream TV signala. Izvršena je sistematizacija problema koji nastaju u ovakvoj IP mreži i pokazan je njihov uticaj na kvalitet video streama koji se prenosi kroz sistem. Kvalitet videa je okarakterisan MDI indexsom i pokazano je kako se on dobija. Očitavanja koja smo dobili pokazala su nam da nema izgubljenih paketa i da mrežni uređaji ne unose nikakvo veće kašnjenje IP paketa i ne izazivaju njihov gubitak. Iz svih dobijenih rezultata može se zaključiti da je čitav sistem stabilan i da nije došlo do smanjenja kvaliteta video streama PRO TV-a tokom njegovog transporta kroz mrežu od strane headend-a do emisionog centra.

LITERATURA

- [1] A. MacAulay, B. Felts, *IP Streaming of MPEG-4 native RTP vs MPEG-2 transport stream*, Envivo, October 2005
- [2] “Transmission of Professional MPEG-2 Transport Streams over IP Networks”, Pro-MPEG Forum, Release 3, July 2004
- [3] D. Hofman, “RTP Payload Format for MPEG1/MEPG2 Video”, RFC-2250, January 1998
- [4] *Digital Video Broadcasting (DVB); Transport of MPEG-2 Based DVB Services over IP Based Networks*, DVB IP Phase 1, ETSI TS 102 034, March 2005
- [5] Sergio D. Servetto, Klara Nahrstedt, *Broadcast Quality Video over IP*
- [6] S. Memill Weiss, *Video over IP*, Elsevier 2008
- [7] Broadcast Quality Video over IP/Ethernet www.harmonicinc.com
- [8] RFC – 4445 A Proposed Media Delivery Index (MDI)

KOMPARACIJA PERFORMANSI SEKVENCIJALNIH I PARALELNIH PROGRAMA NA NUMERIČKI ZAHTJEVNOM PRIMJERU

COMPARISON OF PERFORMANCES OF SEQUENTIAL AND PARALLEL ALGORITHMS ON THE NUMERICAL DEMANDING EXAMPLE

Dražen Žujović, Luka Filipović, *Centar informacionog sistema Univerziteta Crne Gore*,
Božo Krstajić, *Elektrotehnički fakultet, Podgorica*

Sadržaj – U ovom radu su predstavljene koncepti i prednosti distribuiranog računarstva, klastera i Grid infrastrukture. Na distribuiranim računarskim sistemima se izvršavaju paralelni programi. Na numerički zahtjevnom primjeru su prikazane i analizirane prednosti paralelnih u odnosu na sekvencijalne programe.

Abstract – In this paper concepts and advantages of distributed computing systems, clusters and Grid infrastructure are presented. Parallel programs executes on distributed computing systems. Advantages of parallel in compare with sequential programs are presented and analyzed.

1. UVOD

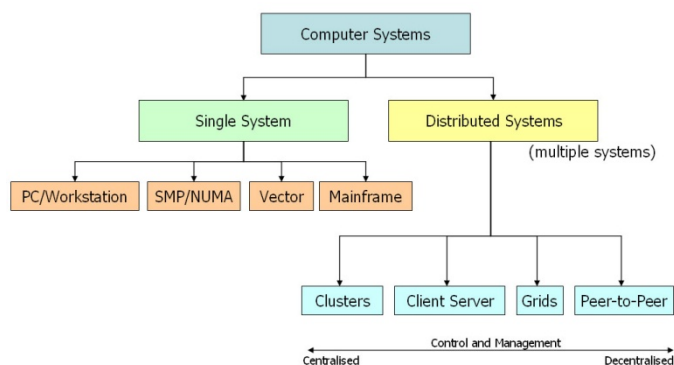
Konstantno raste potreba za povećanjem računarskih resursa. Odgovor na ove potrebe leži u distribuiranom računarstvu. Puno je projekata širom svijeta koji finansiraju izgradnju Grid sistema sa različitim namjenama. Postoje virtuelne organizacije koje ustupaju na korištenje svoje računarske resurse (klasteri, opremu specijalizovanu za određenu oblast, rezultate istraživanja...). Distribuirani računarski sistemi olakšavaju saradnju naučnika iz različitih institucija širom svijeta.

Korištenjem svih prednosti distribuiranog računarstva, istraživači će kompleksne, zahtjevne probleme rješavati brže, efikasnije uz intenzivniju međusobnu saradnju. Sa svog radnog mjesta će moći da koriste računarske resurse koji se nalaze u nekoj udaljenoj laboratoriji, a takođe i omogućiti pristup drugim istraživačima svojim računarskim resursima i rezultatima istraživanja. Neće morati da brinu o nedostatku storage prostora, manjku memorijskog kapaciteta, nedostatku specijalizovane računarske opreme i raznim drugim problemima sa kojima se susreću prilikom raznih modeliranja, simulacija, obrade velike količine podataka i sl.

U ovom radu će biti predstavljeni koncept, prednosti i primjene distribuiranih računarskih sistema, klastera i Grid-a. Navedena su pravila kojih se treba pridržavati prilikom projektovanja paralelnog algoritma i način računanja efikasnosti paralelnog algoritma. Dati su i analizirani rezultati poređenja računanja broja π na jednoprocorskom i na višeprocorskom sistemu.

2. DISTRIBUIRANI SISTEMI

Distribuirani sistemi predstavljaju kolekciju udruženih nezavisnih računarskih sistema koji se korisnicima i procesima koji se na njima odvijaju, predstavljaju kao jedinstveni koherentni sistem [1]. Zbog svog centralizovanog klijent – server dizajna informacionog sistema, distribuirani sistemi imaju brojne prednosti, kao što su: laka proširivost, velika otpornost na greške usled padova pojedinih djelova sistema, velika zbirna procesorska i memorijska moć...

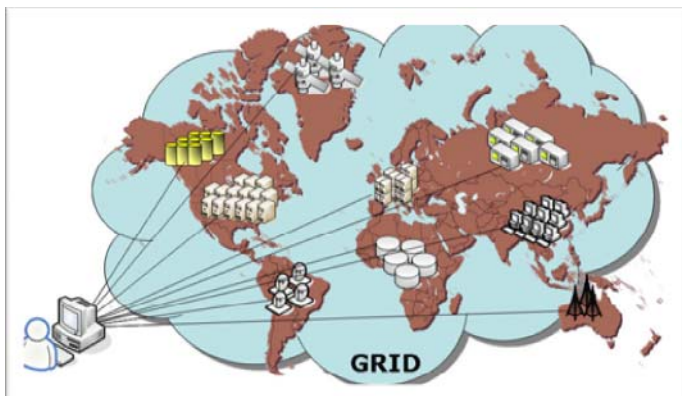


Sl. 1. Podjela računarskih sistema

Osnovu koncepta distribuiranog računanja čine distribuirani sistemi - klasteri. Računarski klaster je skup međusobno fizički nezavisnih radnih stanica, odnosno računara, integrisanih pomoću softvera i mrežnih resursa radi objedinjavanja njihovih računarskih resursa u jedan jedinstven računarski sistem. Osobine sistema sastavljenog od više komponenti da sakriva složenost i predstavlja se kao jedinstven sistem naziva se Privid Jedinstvenog Sistema (engl. Single System Image SSI), u nastavku SSI. Neke od osobina koje klaster treba da posjeduje da bi bio SSI klaster su: jedinstvena pristupna tačka, jedinstveni korisnički interfejs, jedinstveni sistem za upravljanje resursima, jedinstveni sistem za upravljanje poslovima, jedinstveni hijerarhija fajlova, jedinstvena virtuelna računarska mreža, jedinstveni memorijski prostor jedinstveni I/O prostor, jedinstveni procesni prostor, praćenje i snimanje stanja i selidba procesa...

Grid predstavlja infrastrukturu koja pruža skalabilan, zaštićen i efikasan mehanizam pronalazanja distribuiranih resursa i koordinisan pristup ovim resursima u okviru dinamičnih, multi-institucionalnih virtuelnih organizacija. Pod resursima podrazumevaju se uređaji za skladištenje podataka, uređaji za obradu podataka čije procesorsko vrijeme koristimo, kao i ostala oprema koje možemo kontrolisati, kao što su npr. teleskopi, medicinski instrumenti i mnogi drugi. Koordinisan pristup nameće potrebu da se za određene grupe korisnika definišu prava korišćenja resursa:

kojim resursima smiju da pristupe, koje od njihovih mogućnosti su im na raspolaganju, kao i koji dio ukupnog vremena rada resursa, ili njegovog skladišnog kapaciteta grupa može da koristi za svoje poslove. [2]



Sl. 2. Grid sistem

Cilj grid computing-a je rješavanje velikih problema, preobimnih za pojedinačne superračunare, dijeleći probleme na manje djelove i rješavajući ih. Drugi cilj predstavlja bolja eksploatacija dostupne računarske snage i povremeno iskorištavanje za zahtjevne računarske operacije.

Grid computing uključuje korištenje heterogenih resursa, baziranih na različitim platformama, hardware i software (middleware) arhitekturama, kompjuterskim jezicima, lociranih na različitim mjestima širom Zemlje, koristeći telekomunikacionu i računarsku mrežu koristeći priznate standarde.

Grid computing se često poredi sa cluster computing-om. Glavnu razliku čini karakteristika da cluster predstavlja grupu čvorova postavljenih na jednoj lokaciji, a Grid je sastavljen od većeg broja cluster-a i drugih resursa (računarska mreža, resursi koje možemo kontrolisati). [2,3,4]

Po načinu korišćenja, Grid možemo podijeliti u tri grupe:

- Computational Grid – fokusira se prvenstveno na aplikacije koje zahtijevaju velike proračune na velikom broju snažnih računara
- Data Grid – kontroliše i upravlja velikim količinama udaljenih storage lokacija
- Equipment Grid – fokusira se na opremi (teleskopi, medicinski uređaji, uređaji u meteorologiji...), njenim podacima i resursima koji služe za analizu tih podataka

Koncept grid-a nalazi primjenu u rešavanju veoma širokog spektra problema, kao i od strane različitih institucija, organizacija i grupa ljudi. Neke od mogućih primjena su u državnim institucijama (na primjer timovi za nacionalnu odbranu, timovi kao pomoć u slučaju nesreća, dugoročna istraživanja i planiranja...), medicinskim ustanovama (baze podataka o pacijentima i bolestima, arhive medicinskih slika, specijalizovani medicinski instrumenti...), naučne institucije (naučni proračuni, simulacije, saradnja pri analizama...) i poslovne primjene (simulacije tokova na tržištu, iznajmljivanje resursa, renderovanje slika, interaktivne igre, poslovne CAD primjene...).

3. PARALELNO PROGRAMIRANJE

Distribuirani algoritmi sastoje se od više istovremenih procesa koji saraduju u obavljanju istog zadatka. Pri tome ti procesi stupaju u međusobnu interakciju isključivo tako što razmjenjuju poruke. Za razliku od paralelnih algoritama namjenjenih specijalizovanim višeprosorskim računarima, distribuirani algoritmi izvode se na skupovima umreženih, ali inače samostalnih konvencionalnih računara.

Distribuirani algoritam teško je analizirati zato jer se njegovi procesi odvijaju na asinhron i vremenski teško predvidiv način, a pritom ni jedan proces nema potpuni uvid ni kontrolu nad onim što rade drugi procesi. Ipak, postoje brojni primjeri distribuiranih postupaka za koje se usprkos navedenim okolnostima može garantovati da rade korektno. Takođe, mnogi od njih u stanju su efikasno raspodijeliti posao na procese i tako postići značajna ubrzanja u odnosu na odgovarajuće sekvencijalne postupke [5,6].

Kod planiranja paralelnog programa treba paziti na sljedeće zamke:

- Izjednačavanje opterećenja (load balancing) - raspodjela poslova u cilju veće efikasnosti algoritma. Česta je greška da svi zadaci čekaju da jedan završi posao.
- Zrnatost (granularity) – odnos količine računanja (lokalnog reda) i količine komunikacije (nelokalnog reda). Potrebno je dobro isplanirati međusobnu komunikaciju. Sitnozrnatost (fine – grained) - mala količina računanja između uzastopnih komunikacija. Krupnozrnatost (coarse - grained) - velika količina računanja u odnosu na komunikaciju.
- Zavisnost među podacima – postoji kod višestruke upotrebe iste memorijske lokacije i čest je uzrok nemogućnosti paralelizacije. Načini da se ovaj problem riješi su raspodijeljena memorija (slanje potrebnih podataka u trenutku sinhronizacije) i zajednička memorija (sinhronizacija čitanja i pisanja memorije među procesima)
- Potpuni zastoj (deadlock) – stanje u kojem dva ili više procesa čekaju na događaj jednog od drugih procesa.

Čest je slučaj da imamo sekvencijalno urađen algoritam i da treba da ga pretvorimo u paralelni. Neki od koraka u razvoju paralelnog algoritma su:

- Pronaći djelove sekvencijalnog programa koji se mogu izvršavati istovremeno
- Rastaviti algoritam
- Izrada programa
- Ispravljanje grešaka, optimizovanje izvođenja

Efikasnost paralelnog algoritma možemo izračunati koristeći Amdahl-ov zakon.

$$Ubrzanje = \frac{1}{S + \frac{P}{N}} \quad (1)$$

Gdje su:

Ubrzanje – odnos brzina izvršavanja paralelnog i sekvencijalnog programa

S – sekvencijalni udio programa

P – paralelizovani udio programa

N – broj procesora koji izvršavaju zadatak

Da bi paralelni program bio shvaćen od strane računarskog sistema koji ga izvršava moraju da postoje neka jasno definisana pravila. Najkorišćeniji model paralelnog programiranja je komunikacija porukama (message passing). Više zadataka izvršavaju se nezavisno, podaci se razmjenjuju porukama. Ovaj model se ponekad naziva i SPMD (single program, multiple data) – jedan program se izvršava na više procesora.

Bilo koja tehnika paralelnog programiranja za svaki proces mora da obezbijedi bar sljedeće mehanizme:

- Otkriti ukupan broj procesa
- Identifikovati sopstveni proces u grupi procesa
- Poslati poruku određenom procesu
- Primiti poruku od određenog procesa

MPI (Message Passing Interface) je standard koji uključuje preko 120 funkcija za pisanje paralelnih programa, no većina funkcionalnosti se može postići mnogo manjim skupom.

4. PRIMJER POREĐENJA PARALELNOG I SEKVENCIJALNOG PROGRAMA

Kao pogodan primjer za prikaz razlika u performansi pri izvršavanju sekvencijalnog programa na jednoprocorskoj radnoj stanici i paralelnog programa na Grid sistemu, izabran je primjer računanja broja π , po sljedećem algoritmu:

$$\pi = \frac{4}{n} \sum_{i=1}^n \frac{1}{1 + \left[\frac{i-0.5}{n}\right]^2} \quad (2)$$

Za veće n se dobija rezultat veće preciznosti, što, naravno, zahtijeva i veće vrijeme računanja. Ovaj primjer je pogodan za poređenje, jer je veliki procenat algoritma moguće paralelizovati. Rađena su testiranja sekvencijalnog programa pisanog u programskom jeziku C, i upoređivani rezultati sa rezultatima dobijenim izvršavanjem paralelnog programa rađenog po istom algoritmu, takođe pisanim u programskim jezikom C. Testiranja su vršena za različito n . Paralelni program je izvršavan na SEE-GRID-SCI¹ infrastrukturi, kojoj pripada i akademski klaster Univerziteta Crne Gore.

Za izvršavanje ovog programa bilo je potrebno:

- Prekopirati program na neki od računara akademskog klastera
- Iskompajlirati program
- Prijaviti se na Grid, dobiti privremeni certifikat
- Definisane, zakazivanje izvršavanja i slanje zadatka na izvršavanje

- Provjeriti stanje zadatka
- Po završetku izvršavanja programa, iskopirati na lokalni računar rezultate sa udaljenog klastera

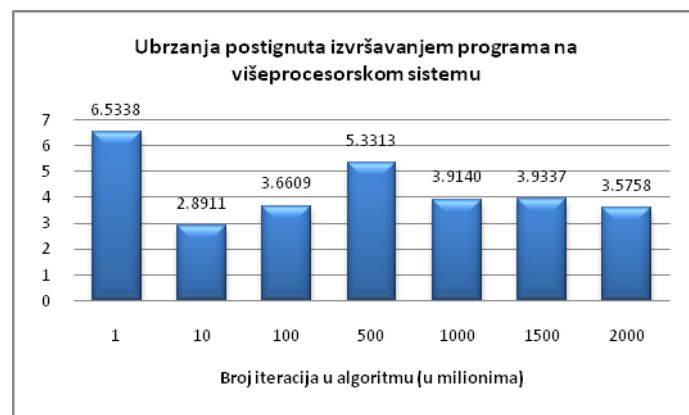
Za definisanje, zakazivanje izvršavanja i slanje zadatka na izvršavanje služi JDL (Job Description Language). Neki od JDL atributa su:

- Broj procesora potreban za izvršavanje zadatka
- Tip posla (Jobtype)
- Lista fajlova potrebnih za izvršavanje programa
- Lista fajlova koji će biti generisani
- Lista ulaznih podataka potrebnih za izvršavanje programa

Workload Management System (WMS) sadrži set Grid *middleware* komponenti odgovornih za distribuciju i upravljanje poslovima, kroz Grid infrastrukturu. WMS upravlja poslovima u korisnikovo ime, tako da korisnik ne mora da zna gdje će posao da se izvršava.

Sekvencijalni program je, u okviru testiranja, izvršavan na radnoj stanici sa Intel(R) Core(TM)2 Duo CPU, brzine procesora 2.40 GHz, 2 GB RAM memorije, sa instaliranim Windows XP Professional operativnim sistemom.

U ovom primjeru je paralelni program izvršavan na 10 jezgara. Procenat algoritma koji može da se paralelizuje je veoma visok. Razlika u brzini izvršavanja, međutim, nije 10 puta na strani paralelnog algoritma, nego nešto manja. Amdahl – ov zakon kaže, pošto je ovo algoritam koji je moguće paralelizovati u veoma visokom procentu (procenat paralelizacije $\approx 99\%$), da će odnos brzina izvršavanja paralelnog i sekvencijalnog programa biti 9,17 puta. Nije ni Amdahl – ov zakon u praksi potvrdio rezultate. To je uglavnom posledica postojanja i trošenja vremena na međuprocorsku komunikaciju.



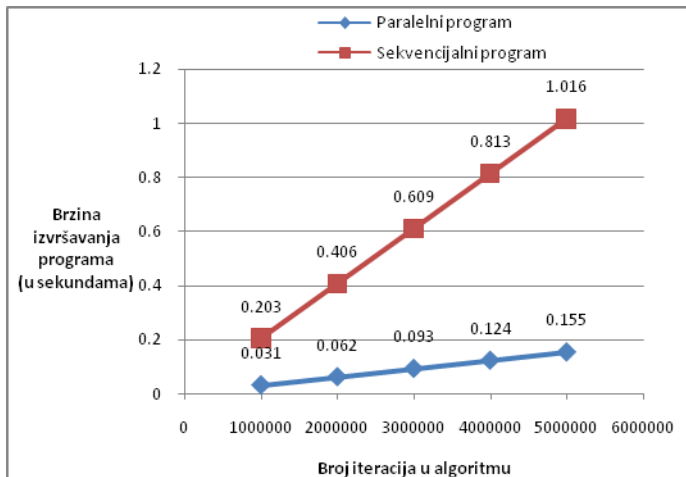
Sl. 3. Brzina izvršavanja paralelnog u odnosu na sekvencijalni program

Na Slici 3. nije moguće uočiti bilo kakav trend ubrzanja postignutog izvršavanjem programa na višeprocorskom sistemu. Obzirom na heterogenost hardvera SEE - GRID - SCI infrastrukture i na to da je WMS bio taj koji je vršio odabir klastera na kom će da se izvršava paralelni program, paralelni program nije izvršavan na radnim stanicama jedinstvene procesorske snage i memorijskog kapaciteta.

¹ The South-East European eInfrastructure

Posledica toga je i to da vrijeme rada paralelnog programa

Moguće je pri radu na SEE – GRID infrastrukturi birati klaster na kom želite da se izvršava paralelni program. Naravno, mora da je na listi ponuđenih koji zadovoljavaju uslove zadatim u JDL fajlu. To je još jedan značajan podatak, jer na taj način možete uvijek da birate klaster koji je pouzdan i za koji ste sigurni da je dobrih performansi.



Sl. 4. Poređenje dobijenih rezultata

Na grafiku (Slika 4.) je prikazano poređenje vremena izvršavanja paralelnog programa na jednoprocesorskom sistemu i na akademskom klasteru Univerziteta Crne Gore. Sekvencijalni program je izvršavan na jednom nodu iz tog klastera. Zbog homogenosti hardvera dobijamo očekivani trend u rastu vremena izvršavanja programa u zavisnosti od broja iteracija. Prednost izvršavanja programa na distribuiranom računarskom sistemu nad izvršavanjem programa na jednoprocesorskoj jedinici u uštedi vremena. Što je veći broj iteracija veća je i ušteda vremena (odnos potrošnje vremena ostaje približno isti).

5. ZAKLJUČAK

Sve je veći broj naučnika i institucija koji prepoznaju Grid kao alat za brže, jeftinije i efikasnije rješavanje zahtjevnih numeričkih zadataka. Ovaj sistem omogućava bolju saradnju među naučnicima i institucijama širom svijeta u vidu dijeljenja rezultata istraživanja i korištenja udaljenih računarskih resursa, senzora, teleskopa, laboratorijske opreme... Distribuirani računarski sistemi se brzo razvijaju, i u budućnosti će imati sve veću primjenu. Ono na šta je

nije direktno proporcionalno broju iteracija u programu.

trenutno posebna pažnja istraživača usmjerena je izgradnja boljih mehanizama za toleranciju na greške, pouzdanost sistema, sigurnost, monitoring procesa...

Kroz primjer algoritma za računanje broja π , analizirane su prednosti višeprocorskih u odnosu na jednoprocesorske sisteme. Iz rezultata dobijenih izvršavanjem ovog algoritma na jednom jezgri i izvršavanja na 10 jezgara istovremeno, dolazi se do više zaključaka. Broj iteracija nije direktno srazmjernan vremenu izvršavanja paralelnog programa. Ulogu u brzini izvršavanja paralelnog programa na Grid-u ima i vrijeme utrošeno na međuprocorsku komunikaciju. To vrijeme može da bude nezavino od broja iteracija. SEE-GRID-SCI infrastruktura ima heterogen hardver. Ako prepustimo WMS-u da bira klaster na kom će se izvršavati naš paralelni program, ne možemo znati koliko će biti vrijeme izvršavanja. Najbolje je da sami biramo klaster koji se u praksi pokazao kao pouzdan i efikasan, a koji zadovoljava uslove postavljene u JDL fajlu. Što je zahtjevniji algoritam u pitanju, što je veći broj iteracija, veća je i ušteda vremena, tj. korist od upotrebe paralelnog programa i njegovog izvršavanja na distribuiranom računarskom sistemu.

LITERATURA

- [1] Andrew S. Tanenbaum, Maarten van Steen, Distributed Systems: Principles and Paradigms. Prentice Hall (2002.)
- [2] Foster, I., Kesselman, C., Nick, J. and Tuecke, S. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. Globus Project, 2002,
- [3] Berman F, Fox G, Hey T. The grid: past, present, future. In Grid Computing: Wiley: Chichester, 2003;
- [4] Ian Foster, The Grid: A new infrastructure for 21st century science, John Wiley & Sons, Ltd, 2003.
- [5] Lynch, Nancy A (1997). Distributed Algorithms. Morgan Kaufmann.
- [6] Tel, Gerard (1994). Introduction to Distributed Algorithms. Cambridge University Press. Kaufmann.
- [7] D. Žujović (2009). Diplomski rad na temu: Komparacija performansi sekvencijalnih i paralelnih programa

ELEKTRONSKA DISTRIBUCIJA PROIZVODA I USLUGA – ELEKTRONSKE DOPUNE ELECTRONIC DISTRIBUTION OF GOODS AND SERVICES – MOBILE TOP UP

Ivan Vučinić, Crnogorska komercijalna banka AD Podgorica

Sadržaj – *Elektronska distribucija proizvoda i usluga predstavlja proces u kojem prodavac robe odnosno pružalac usluge prodaje odnosno pruža uslugu elektronskim putem kupcu robe odnosno korisniku usluge. Proces elektronske distribucije obuhvata tri koraka : definicija robe/usluge koja se prodaje/pruža, identifikacija instrumenata plaćanja i realizacija sistema elektronske distribucije. U ovom radu prikazan je i primjer realizacije sistema elektronske distribucije dopune prepaid računa mobilnih telefona.*

Abstract – *Electronic distribution of products and services is a process in which the seller of goods or service provider sells or provides the service electronically to the buyer of goods or services to the user. Electronic distribution process includes three steps: the definition of goods / services being sold / provided, the identification of payment instruments and the implementation of electronic distribution. This paper is an example of realization of electronic distribution of prepaid mobile phone accounts.*

1. UVOD

Od postanka, civilizacija nastoji da postojeće životne procese konstantno unapređuje. Jedan od najznačajnijih ljudskih procesa je razmjena dobara i usluga koja predstavlja osnov savremene ekonomije.

„Roba je prije svega spoljašnji predmet, stvar koja svojim svojstvima zadovoljava ljudske potrebe ma koje vrste.“ [1]

Roba je vremenom u novcu našla zajednički imenitelj a novac je postao jedna od natraženijih roba. Tokom godina novac je zamijenjen novčanim ekvivalentima preko mjenica, čekova, platnih kartica i konačno elektronskim novcem.

Razvoj civilizacije uvijek su pratile informacije. Informacije se definišu kroz sljedeće izraze

- "Ljudi su tvorci i nosioci informacija",
- "Informacija mora, prije svega, da ima određeni smisao, ona mora biti nosilac značenja",
- "Informacija je sve ono što daje nove podatke, ili nova obavještenja o nekoj činjenici ili nekom događaju, koji nisu bili ranije poznati",
- "Informacije su oni djelovi vijesti koji za primaoca imaju vrijednost novosti i koji mu omogućavaju da bolje izvrši svoje zadatke",
- "Informacije su saznanja koja postaju dostupna pomoću sredstava komunikacije, a posjeduju obavještajnu vrijednost"
- "...sadržaj onog što razmjenjujemo sa spoljnim svijetom dok mu se prilagođavamo i dok utičemo na njega",
- "...svjesna ili ciljna organizovanost podataka" ...[2]

Informacija je oduvijek bila i ostala najtraženija roba. Informacija samo mijenja svoj oblik da bi pronašla svoj put od izvora do ušća.

2. ELEKTRONSKA DISTRIBUCIJA PROIZVODA I USLUGA

Roba, novac i informacija su tri osnovna elementa potrebna da bi se realizovao sistem elektronske distribucije proizvoda. Ukoliko jedan od elemenata u sistemu ne postoji nije moguće realizovati sistem elektronske dopune. Proces elektronske distribucije obuhvata tri koraka : definicija robe/usluge koja se prodaje/pruža, identifikacija instrumenata plaćanja i realizacija sistema elektronske distribucije.

2.1 DEFINICIJA ROBE/USLUGA KOJA SE ELEKTRONSKI DISTRIBUIRA

Teoretski je moguće elektronski vršiti distribuciju bilo koje robe. Robu definiše skup kvalitativnih i kvantitativnih informacija koji je jednoznačno definišu. Uređaj pomoću kojeg se na osnovu informacije može stvoriti bilo koja roba je molekularni sastavljač(" **molecular assembler** " -en. prim. aut) [3-12]. U ovom trenutku tehnologija nije ni približno blizu realizacije ovog uređaja.

Robe koje se distribuiraju elektronskim putem su one robe koje su u osnovi informacije. Ove informacije su se zbog trenutnih ograničenja dostupnosti distribuirale u obliku robe. Ovakve vrste roba su svuda oko nas i svakodnevno se susrećemo sa njima. Prije svega to su informacije koje se prodaju kao roba: novine, radio, televizija. Multimedijalni sadržaji koji se mogu reprezentovati u elektronskom obliku: audio i video sadržaji. Kompjuterski sadržaji koji se mogu predstaviti u obliku skupa podataka: programi, datoteke, ključevi itd. Telekomunikacioni proizvodi i usluge koje su ekvivalent potrošenog vremena kao što su telefonski, prepaid i post-paid računi, voip računi i internet usluge. Poštanske usluge prenosa informacija fizičkim putem kao što su telegrami, uputnice, čestitke itd. Igre na sreću kao što su loto, grebalice, sportska prognoza, klađenje itd.

2.2 IDENTIFIKACIJA INSTRUMENTATA PLAĆANJA

Identifikacija instrumenata plaćanja je proces kojim se definiše skup novčanih ekvivalenata kojima se može vršiti plaćanje. Da bi korišćenje nekog instrumenta plaćanja bilo moguće potrebno je obezbijediti infrastrukturu koja bi podržala izabrani instrument plaćanja. Npr. da bi se na nekoj teritoriji uspješno koristila gotovina kao instrument plaćanja potrebno je da postoji novac, ukoliko je potrebno da se obezbijedi korišćenje kartica onda je potrebno izgraditi mrežu ATM, POS i e-Commerce lokacija a ukoliko je potrebno obezbijediti sistem elektronskog plaćanja onda je potrebno razviti sistem kojim se putem telekomunikacionih protokola povezuju kupci i prodavci. Navedeni elektronski novčani ekvivalenti se uvijek svode na razmjenu informacija o učesnicima transakcije pomoću kojeg se vrši prenos sredstava sa jednog računa na drugi.

2.3 REALIZACIJA SISTEMA ELEKTRONSKE DISTRIBUCIJE

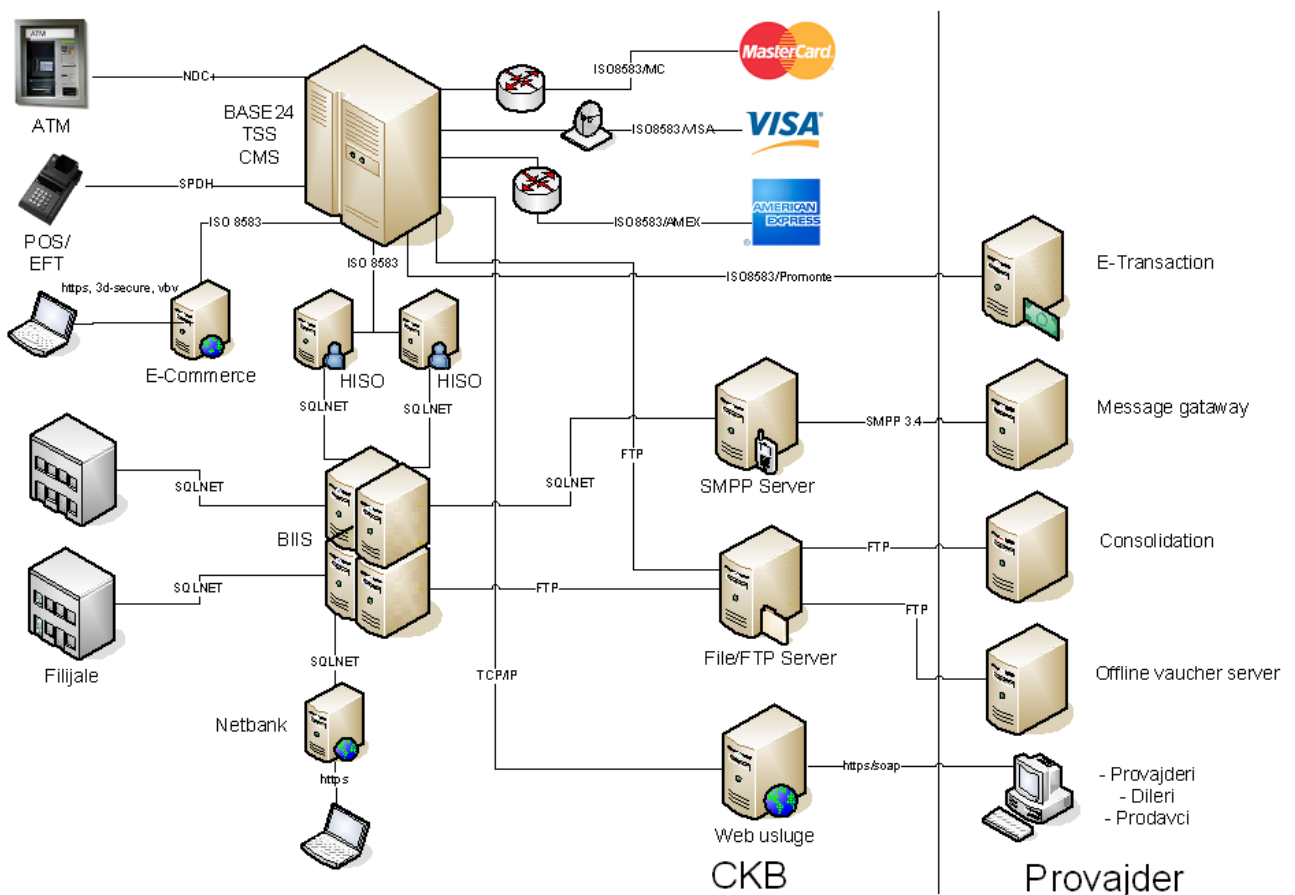
Sistem elektronske distribucije treba da obezbijedi da proizvodi i usluge koji se nude mogu biti dostavljeni kupcu/korisniku u elektronskoj formi pri čemu korisnik na lokaciji dostave vrši plaćanje predviđenim instrumentima plaćanja. Npr. korisnik umjesto da kupi novine na trafici na web stranici plati pretplatu koristeći platnu karticu ili elektronsko bankarstvo kao instrument plaćanja.

Prodavac/pružalac mora posjedovati informacioni sistem u kojem vodi evidenciju korisnika kojima pruža/prodaje robu ili uslugu. Na ovaj način elektronska distribucija predstavlja promjenu informacija vezanih za korisnika u informacionom sistemu prodavca/pružaoca usluga.

Prodavac/pružalac usluga putem sistema elektronske distribucije mora obezbijediti sistem naplate baziran na mogućim platnim sistemima na teritoriji na kojoj se nalazi.

3. PRIMJER SISTEMA ELEKTRONSKE DISTRIBUCIJE – ELEKTRONSKA DOPUNA CKB

Distribucija elektronske dopune se u klasičnom smislu vršila putem papirnih vaučera. Ovi vaučeri su se štampali u cjelobrojnom ograničenom skupu apoeni (npr. 5,10,20 EUR). Cjelokupan sistem je značio da telekomunikacione kompanije (pružaoци usluga) su morale da štampaju i distribuiraju ove vaučere preko master dileri (kompanija zaduženih za distribuciju) po prodajnim mjestima. Ovakav vid distribucije zahtijevao je ogroman napor i velike troškove.



Sl. 1

3.1 DEFINICIJA ROBE/USLUGE KOJA SE ELEKTRONSKI DISTRIBUIRA

Definicija usluge/robe koja se prodaje je stanje na pre-paid računu korisnika odnosno vrijednost dopune. Novčani ekvivalent koji korisnik plaća pretvara se u vrijednost elektronske dopune odnosno novo stanje na pre-paid računu.

3.2 IDENTIFIKACIJA INSTRUMENATA PLAĆANJA

Instrumenti plaćanja koji se koriste su gotovina, platne kartice i elektronski kanali. Da bi instrumenti plaćanja mogli da se koriste potrebno je uspostaviti infrastrukturu. Postojeće infrastrukture ATM, POS, e-Commerce i elektronskih kanala predstavljaju najoptimalnije rješenje za realizaciju ovog servisa.

3.3 REALIZACIJA SISTEMA ELEKTRONSKE DISTRIBUCIJE

Elektronska dopuna mobilnih telefona znači da je potrebno prenijeti informaciju elektronskim putem od prodajnog mjesta do telekomunikacione kompanije koliko je korisnik dopunio telefon.

U sistemu elektronskih dopuna postoje subjekti koji omogućavaju da se informacija uspješno prenese između pružaoca usluga i korisnika usluga. Ovi subjekti su : prodajno mjesto, diler i procesor.

Prodajno mjesto je lokacija na kojoj korisnik može inicirati transakciju elektronske dopune. Prodajno mjesto može biti neautomatizovano tj. pokriveno sa prodavcem (POS terminal) ili automatizovano (ATM terminal, e-Commerce Portal, e-banking portal, m-banking portal ili SMS Telefon).

Diler je subjekat koji je zadužen za širenje prodajne mreže i naplatu potraživanja.

Procesor je zadužen da obezbijedi tehničku komunikaciju između terminala prodajne mreže i centralnog sistema za upravljanje transakcijama odnosno komunikacije centralnog sistema sa informacionim sistemom pružaoca usluga.

3.4 PROCEDURA ELEKTRONSKIH DOPUNA

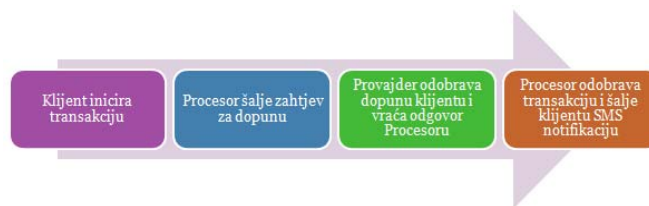
Na neautomatizovanim prodajnim mjestima ovlašćena lica unose broj telefona i iznos dopune putem POS terminala. Ukoliko je transakcija uspješna korisnik usluge dobija obavještenje putem odgovarajuće SMS poruke. Korisnik plaća elektronsku dopunu u gotovini ili putem kartice.

Procedura na automatizovanim prodajnim mjestima je da korisnik sam popunjava broj telefona i iznos dopune. Transakcije se mogu vršiti samo putem elektronskih instrumenata plaćanja kao što su platne kartice i elektronski kanali povezani na bankarske račune.

3.5 NAPLATA POTRAŽIVANJA

Naplata potraživanja u potpunosti zavisi od instrumenata plaćanja. Ukoliko je instrument plaćanja gotovina ne postoji

mogućnost automatizacije naplate. Prodajno mjesto je dužno da primljenu gotovinu na prodajnom mjestu uplati u bankarski sistem kako bi se izvršilo plaćanje prema pružaocu usluga. Nasuprot tome, korišćenje elektronskih instrumenata plaćanja obezbjeđuje automatizaciju sistema naplate jer se novac već nalazi u bankarskom sistemu.



SI. 2.

3.6 NEKE SPECIFIČNE VERZIJE SISTEMA ELEKTRONSKE DOPUNE

Postoje specifične verzije sistema elektronske dopune gdje je elektronska dopuna nije po definiciji usluga nego roba. Ovakav sistem u praksi znači da se umjesto štampe papirnih vaučera procesoru dostavlja datoteka sa vaučerima. Procesor elektronski dostavlja vaučere putem iste mreža kao kod standardnog načina dopune. Korisnik dobija na papiru/elektronskom obliku odštampan skriveni kod koji unosi u svoj mobilni telefon da bi ga aktivirao. Ovakav sistem se može realizovati ukoliko pružalac usluga ne može tehnički omogućiti direktnu on-line vezu sa procesorom.



SI. 3.

4. ZAKLJUČAK

Sistemi elektronske distribucije proizvoda omogućavaju direktne benefite i opravdavaju investicije usled mnogo brže distribucije proizvoda. Troškovi distribucije postaju značajno manji usled smanjenja troškova u štampi, prevozu i ljudskim resursima. Eliminiraju se zloupotrebe i manipulacije pri eksploataciji robe za distribuciju (prodavci prodaju stalno iste novine, pre-paid vaučeri se koriste kao novčani ekvivalenti i sl.).

Komunikacija sa korisnicima postaje personalizovana usled korišćenja elektronskih sistema za dopune. Navike potrošača se identifikuju kroz prodajna mjesta kao budući kanal za distribuciju novih proizvoda. Ogromna količina podataka omogućavaju izradu sofisticiranih "data warehouse" rješenja koja će omogućiti donošenje boljih poslovnih odluka.

LITERATURA

- [1] Karl Marks, Das Kapital: Kritik der politischen Ökonomie, 1867
- [2] Đorđe Nadrljanski, Informatika za učitelje, UF Beograd, 1996
- [3] Gall, John, (1986) Systemantics: How Systems Really Work and How They Fail, 2nd ed. Ann Arbor, MI : The General Systemantics Press.
- [4] Richard E. Smalley, "Of chemistry, love, and nanobots," Scientific American 285(September 2001):76-77.
- [5] Robert F. Service, "Is nanotechnology dangerous?" Science 290(24 November 2000):1526-1527.
- [6] Wilson Ho, Hyojune Lee, "Single bond formation and characterization with a scanning tunneling microscope," Science 286(26 November 1999):1719-1722; <http://www.physics.uci.edu/~wilsonho/stm-iets.html>
- [7] K. Eric Drexler, [Nanosystems: Molecular Machinery, Manufacturing, and Computation](#), John Wiley & Sons, New York, 1992, Chapter 8.
- [8] Ralph C. Merkle, "A proposed 'metabolism' for a hydrocarbon assembler," Nanotechnology 8(1997):149-162; <http://www.zyvex.com/nanotech/hydroCarbonMetabolism.html>
- [9] Charles B. Musgrave, Jason K. Perry, Ralph C. Merkle, William A. Goddard III, "Theoretical studies of a hydrogen abstraction tool for nanotechnology," Nanotechnology 2(1991):187-195; <http://www.zyvex.com/nanotech/Habs/Habs.html>
- [10] Michael Page, Donald W. Brenner, "Hydrogen abstraction from a diamond surface: Ab initio quantum chemical study using constrained isobutane as a model," J. Am. Chem. Soc. 113(1991):3270-3274.
- [11] Susan B. Sinnott, Richard J. Colton, Carter T. White, Donald W. Brenner, "Surface patterning by atomically-controlled chemical forces: molecular dynamics simulations," Surf. Sci. 316(1994):L1055-L1060.
- [12] D.W. Brenner, S.B. Sinnott, J.A. Harrison, O.A. Shenderova, "Simulated engineering of nanostructures," Nanotechnology 7(1996):161-167; <http://www.zyvex.com/nanotech/nano4/brennerPaper.pdf>

PRIMJENA UČENJA NA DALJINU U OBRAZOVANJU APPLICATION OF DISTANCE LEARNING EDUCATION

Jezdimir - Luka Obadović, JUSSŠ „Vukadin Vukadinović“, Medicinski fakultet, Visoka medicinska škola, Berane

Sadržaj - U radu je prikazana primjena sistema za učenje na daljinu zasnovana na korišćenju savremene informaciono-komunikacione tehnologije u gotovo svim elementima procesa učenja. Učenje u mreži računara preko Intraneta, odnosno Interneta predstavlja osnovnu ideju sistema za učenje na daljinu. Internet odnosno Intranet se koriste za ostvarivanje uslova za interakciju korisnika sa sadržajem, predavačima i ostalim učesnicima u Distance Learning modelu učenja. Mogućnost razmjene znanja, u formi multimedijalnih sadržaja, po čitavom svijetu, pospješuje brzi rast korpusa znanja što u isto vrijeme zahtijeva i definisanje novih programa i metoda obrazovanja.

Ključne riječi: učenje na daljinu, mreža računara, internet, obrazovanje.

Abstract - The paper describes the application of distance learning based on the use of modern information and communication technologies in almost all elements of the learning process. Learning to network computers via Intranet or Internet is a basic idea of distance learning. Internet and Intranet are used to achieve the conditions for user interaction with content, lecturers and other participants in the Distance Learning model of learning. The possibility of exchange of knowledge, in the form of multimedia content throughout the world, pospješuje rapid growth of the corpus of knowledge which at the same time requires the definition of new programs and methods of education.

Key words: distance learning, computer networks, internet, education.

1. UVODNA RAZMATRANJA

Škola budućnosti, ili još bliže posmatrano reformisana srednja škola, da bi ispunila svoj cilj i da bi reforma školstva u Crnoj Gori opravdala očekivanja, mora u školskoj praksi insistirati na tome da je proces školskog učenja jedan *oblik ljudske komunikacije i interakcije*.

Kada se govori o interakciji i komunikaciji, tu se prije svega misli na nekoliko stvari: da je to *dvosmjernan proces*, da se u tom procesu obavezno uzimaju u obzir stanje, potrebe i reakcije partnera u komunikaciji (*profesor-učenik*), da se smjenjuju uloge onih koji šalju i primaju poruke, da su učesnici u komunikaciji međuzavisni i da naravno postoje pravila preko kojih se odvija ovaj proces komunikacije. Ukoliko u nastavnom procesu ne bi bilo interaktivnosti, onda bi to značilo da nastava ima ozbiljan problem i da je to velika opasnost po efikasnost obrazovanja.

Obrazovanje je proces sticanja novih znanja, bez kojih nema napretka ni pojedinca ni društva. Odvija se tokom cijelog radnog vijeka, odnosno života i jedan je od uslova za sticanje prava na rad i ličnih ciljeva pojedinca. Isto tako, obrazovanje je i evolutivni proces koji se vremenom i sa društvenim promjenama i sam mijenja i usavršava, kako u strukturi i sadržaju tako i u načinu realizacije.

Osnovni principi obrazovanja moraju biti: nepristrasnost, razumljivost, fleksibilnost, raznovrsnost, odgovornost, produktivnost i dr.

Obrazovanje sve više napušta klasične i tradicionalne okvire i prostore i sve više se seli iz učionica u poslovne sisteme i domaćinstva, zahvaljujući informaciono-komunikacionoj tehnologiji i Internetu.

Razvoj informaciono-komunikacionih tehnologija (u daljem tekstu IKT) krajem prošlog vijeka stvorio je preduslov za kvalitativne promjene u organizaciji i realizaciji samog

obrazovanja. Internet, multimedija, virtuelna realnost i vještačka inteligencija sve se češće koriste kao dodatni izvor znanja.

Zahvaljujući razvoju IKT-a obrazovanje je postalo dostupno na mjestima i u vremenu koje sami korisnici obrazovnih programa odaberu.

Upotreba IKT-a uključuje: uvođenje digitalne tehnologije u učenju i prostor za učenje, obezbjeđenje visokog opsega povezanosti unutar i između obrazovnih institucija, povećani broj računarskih učionica sa većim vremenom korišćenja, kreiranje web baziranog okruženja, nabavku i razvoj resursa za podučavanje i podršku nastavnom osoblju u upotrebi tehnologija, kao i povezivanje tehnologije i učenja.

Pravila za nove načine učenja još nijesu utvrđena i potrebno je istražiti i kreirati odgovarajuće okruženje za učenje na daljinu koje se oslanja na podršku IKT-a i učenje u mreži računara preko Intraneta, odnosno Interneta.

Informatičko obrazovanje je premisa, odnosno neophodni uslov daljeg razvoja učenja na daljinu kao i ostvarivanja dvosmjerne komunikacije i razmjene podataka u vremenu koje je pred nama.

Širom svijeta danas postoji više od 90 institucija, najčešće pod nazivom *Open University*, na engleskom ili u prevodu na lokalni jezik a po uzoru na davno nastali *Otvoreni Univerzitet* u Engleskoj, koji primarno ističu značaj učenja na daljinu.

U vremenu koje je pred nama, škola kao obrazovna ustanova čiji je osnovni zadatak osposobljavanje mladih generacija za susret sa savremenim IKT-a i učenjem na daljinu, ima posebnu obavezu da se tim tokovima što prije priključi. Primjena elektronskog učenja na daljinu u savremenoj srednjoj stručnoj školi Crne Gore je nužna i neophodna.

2. KONCEPT UČENJA NA DALJINU

Učenje na daljinu se na početku svog razvoja primarno upotrebljavalo korišćenjem poštanskog sistema, pružajući mogućnost obrazovanja ljudima koji su bili spriječeni da prisustvuju nastavi u klasičnim školama. Tako je prvi stepen razvoja učenja na daljinu bio u stvari Correspondence Learning. Koristile su ga žene zbog isključenosti iz programa institucionalnog obrazovanja koje je bilo namijenjeno tada samo muškarcima, zatim, zaposleni građani koji su bili na radnim mjestima tokom održavanja nastave i oni koji su živjeli i suviše daleko od obrazovnih centara.

Učenje na daljinu je:

- omogućavanje pristupa materijalima za učenje, skriptama, multimedijalnim prezentacijama i ostalim obrazovnim resursima preko Interneta, uz kontinuirano testiranje, provjeru znanja na svim nivoima, elektronsku komunikaciju sa profesorima i drugim,

- upotreba IKT-a u svrhe učenja,
- približavanje i spajanje Interneta i učenja,
- sticanje znanja i vještina na osnovu informacija i uputstava dostavljenih primjenom različitih IKT-a i ostalih formi učenja na daljinu,

- formalizovan sistem podučavanja i učenja kreiran tako da bude funkcionalan na daljinu upotrebom elektronske komunikacije,

- obezbjeđivanje uslova učenja, za više različito udaljenih mjesta od učionice, škole ili centra u multimedijalnom obliku, kroz njihovu kombinaciju sa tradicionalnim metodama prenošenja znanja.

Škola, prateći savremene tendencije u obrazovanju, pruža učenicima mogućnost učenja na daljinu i svoju edukativnu ulogu višestruko potvrđuje i ostvaruje sljedeće beneficije, kroz:

- besplatno stvaranje tehničkih uslova (postavljanje servera, održavanje sajt i sl.),
- kreiranje nastavnih predmeta i kurseva prema školskim obrazovnim programima-zanimanjima,
- detaljan uvid u individualno napredovanje svakog učenika-ce,
- kontrolu i nadzor nad radom svakog profesora,
- tehničku podršku učenicima-cama 24h/dan.

Sistem učenja na daljinu se sastoji iz sljedećih komponenti:

- software-a za logovanje predavača, slanje podataka i video sadržaja na internet (software-a za katedru),
- web portal-a koji omogućava učenicima pristup podacima, pristup video sadržajima, statistiku ocjena, online testove, oglasne table....,
- prpratnog hardware-a, računara, kamera, bežičnog mikrofona, itd.

Sistem učenja na daljinu omogućava snimanje i slanje predavanja na internet lokaciju (web server), slanje prpratne

dokumentacije uz predavanje, organizovanje i sortiranje podataka, brzo dopisivanje profesor-učenik, forum za raspravu, on-line testove za spremanje nastavnog gradiva, on-line oglasne table, statistiku, mogućnost slanja podataka za profesore sa bilo koje lokacije (postavljanje predavanja i nastavne dokumentacije iz vlastite kuće). Sistem nudi još mnogo drugih prednosti kao što su bolji imidž škole, dostupnost predavanja za učenike, transparentna evidencija o dobijenim ocjenama itd.

Koncept učenja na daljinu premašuje granice tradicionalne nastave. Polaznici su oslobođeni dolaska u učionicu na klasičan čas. Umjesto toga koriste ostale raspoložive resurse, materijale za učenje, praktičnu vježbu i drugo putem odgovarajućih medija (Interneta, softvera za učenje itd.). Ovakav način učenja može biti i interaktivan uz mogućnost orijentacione sesije, zvaničnih časova ili odgovaranja pod nadzorom. Učenicima se otvara mogućnost obuke prema sopstvenoj dinamici i individualne konsultacije preko elektronske pošte, Chat servisa, elektronskih konferencija, na lokalnoj mreži (LAN) ili u višekorisničkom domenu. Kod ovog koncepta, predavači mogu da koriste video prenos preko Interneta koji postaje sve pristupačiji razvojem IKT-e i infrastrukture. Korišćenjem IKT-a učenici mogu učestvovati u nastavi sa bilo koje geografske lokacije.

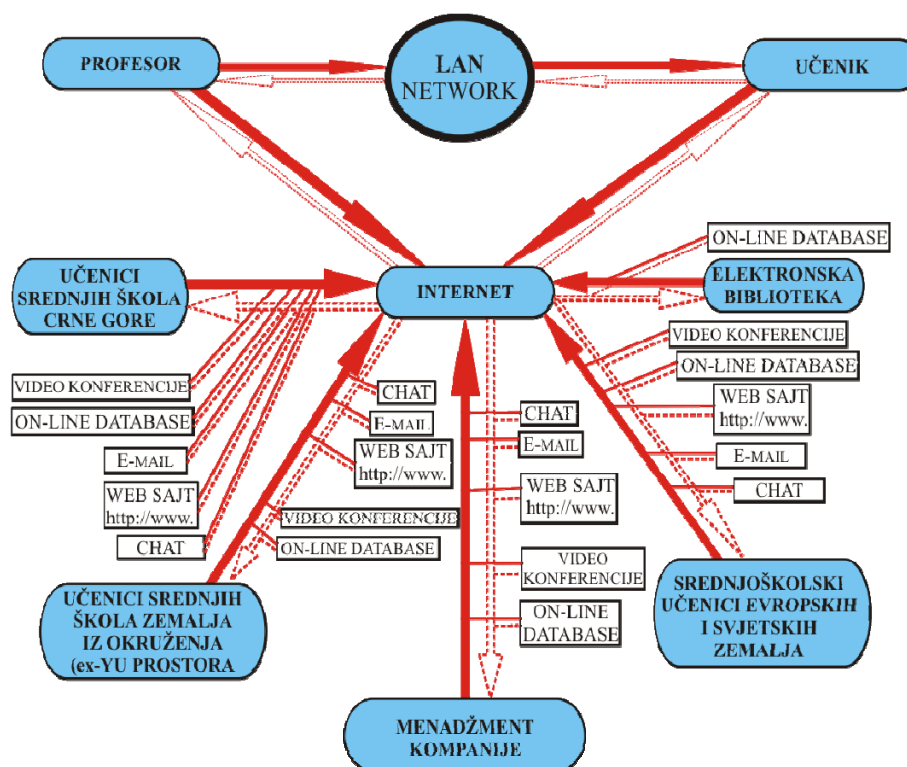
Primjena umreženih računara u oblasti obrazovanja predstavlja moćno sredstvo koje može značajno doprinijeti unapređivanju procesa obrazovanja. Mogućnost razmjene znanja, u formi multimedijalnih sadržaja, po čitavoj Crnoj Gori, ex-YU prostorima, Evropi i po čitavom svijetu, pospješuje brzi rast korpusa znanja što u isto vrijeme zahtijeva i definisanje novih elektronskih nastavnih programa i metoda obrazovanja, slika 1.

2.1 ISKUSTVA POČETAKA PRIMJENE UČENJA NA DALJINU U SREDNJOŠKOLSKOM OBRAZOVANJU

Tradicionalne uloge profesora i učenika, u učionici se mijenjaju velikom brzinom i diktirane su zahtjevima vremena i opštim razvojem crnogorskog društva.

Sistem za elektronsko učenje (slika 1), nudi učenicima: digitalne statičke materijale za učenje putem weba (digitalni udžbenik, skriptu, zbirku, katalog, uputstva za vježbu i dr.), mogućnost komunikacije *učenik ↔ učenik*, i mogućnost *on-line* komunikacije sa profesorom (e-mail, chat, facebook i dr.).

Od školske 2004/2005 do 2009/2010. godine, *Ministarstvo prosvjete i nauke Crne Gore* opremilo je više od polovine srednjoškolskih informatičkih kabineta ili računarskih učionica sa savremenim računarima, povezanim u LAN mrežu (lokalnu mrežu), sa stalnim pristupom na Internet. Tako su, stvoreni uslovi za početak primjene učenja na daljinu i u srednjoškolskom obrazovanju, **slika 1**.



Slika 1. Učenje na daljinu u odnosu na klasični „tradicionalni“ pristup učenju

Srednjoškolska iskustva u korišćenju učenja na daljinu, biće prikazana na primjeru *JU Srednje stručne škole „Vukadin Vukadinović“* - Berane, slika 1:

- informatički kabinet je opremljen sa 15 učeničkih i 1 profesorskim računarem. Svaki učenik ima svoj računar koji koristi tokom cijele školske godine. Svi računari su kablovima povezani u *Local Area Network - LAN* mrežu, tj. lokalnu računarsku mrežu u školi, sa stalnim pristupom Internetu,

- učenici u informatičkom kabinetu, preko računara:

- razmjenjuju razne obrazovne informacije među sobom (**učenik ↔ učenik**), u okviru škole,

- sa profesorom (**učenik ↔ profesor**),

- sa učenicima iz ostalih beranskih srednjih škola (**učenik ↔ učenik jedne od ostale tri beranske srednje škole**),

- sa učenicima iz ostalih 44 srednjih škola Crne Gore (**učenik ↔ učenik jedne od ostalih 44 srednjih škola**),

- sa učenicima: eX-YU prostora, Evrope i svijeta (**učenik ↔ učenik eX-YU prostora, Evrope, svijeta**),

- elektronsko učenje se ostvaruje razmjenom informacija: *e-mailom*, *chatom*, *world wide webom* (www), *facebookom*:

- *E-mailom*: svaki učenik *JU Srednje stručne škole „Vukadin Vukadinović“* ima otvorenu *e-mail adresu*, sa svojim imenom i prezimenom, kao npr. marko.markovic@gmail.com preko koje komunicira sa ostalim učenicima, profesorom i svima drugima elektronskom poštom. Učenici ovim putem razmjenjuju digitalni obrazovni materijal, razne zadatke, itd. i tako demonstriraju jedan vid elektronskog učenja,

- *Internet Relay Chat (IRC - Chat)*: dva ili više učenika, odnosno korisnika sistema za elektronsko učenje, koji su se našli zajedno na jednom mjestu, preko *chat* interaktivnog servisa Interneta međusobno komuniciraju u realnom vremenu, ćaskaju preko Interneta korišćenjem IRC-a i

programa za ćaskanje različitih portala, diskutuju i razmjenjuju različite informacije o temama koje ih zanimaju na velikim daljinama,

- *World Wide Web* (<http://www>) ili kraće samo *web sajt* predstavlja najatraktivniji servis na Internetu, sa nepreglednim izvorom informacija. U elektronskom sistemu učenja na daljinu učenici *web* koriste za brz dolazak do različitih digitalnih udžbenika, zbirki, skripti, pripremu, uputstva za vježbu, i na taj način stiču obrazovanje koje se prenosi računarem, zasnovano na Internetu (web stranicama) kao novom metodu izvođenja nastave u školama i programima cjeloživotnog obrazovanja. Na nivou škole formiran je školske 2003/2004. godine *web tim* koji u saradnji sa profesorom, na času izborne nastave iz nastavnog predmeta *Internet i elektronsko poslovanje* uređuje školski *web sajt* čija je adresa <http://www.vvukadinovic.edu.me/>. Tu su *forumi*, *wikipedija* i drugi *web* sajtovi na kojima učenici mogu svakodnevno, ili čak i svakih nekoliko minuta, naći neki novi digitalni obrazovni materijal za učenje i nove informacije. Učenici na času posjećuju *Crnogorski školski obrazovni portal* (<http://www.skole.me>) na kojem je predstavljeno 85 škola, od toga 22 srednje škole, sa vijestima. Svaki učenik je učlanjen na „*fejsbuk*“ (en: *Facebook-u*), internet stranicu koja služi kao servis za socijalnu mrežu, na kojoj učenici razmjenjuju informacije, i čine sastavni dio grupe od 350 miliona aktivnih korisnika „*fejsbuka*“ širom svijeta.

Učenje na daljinu u odnosu na klasični „tradicionalni“ pristup učenju pokazuje sljedeće prednosti:

- omogućeno stalno učenje (*lifelong learning*) i profesionalno usavršavanje,

- samostalno učeničko učenje, vlastitim tempom, na mjestu i u vremenu koje sami učenici odaberu, na raspolaganju im je veliki broj nastavnih predmeta, koji nude različite institucije ili profesori-pojedinci,

☐ vlastiti tempo učenikog prolaza kroz materijal za učenje onom brzinom i onoliko puta koliko učenici žele,

☐ odabiranje mjesta učenja u zavisnosti od medija koji se koristi za distribuiranje materijala za učenje (uči se u školi ili kod kuće...),

☐ dostupnost učenicima tema koje ne nude kursevi/programi, odnosno učeničko pronalaženje i pohađanje programa koji ih zanimaju, i ako ih ne nude srednje škole u mjestu u kojem žive i uče,

☐ učeničko učestvovanje u najkvalitetnijim ili najprestižnijim programima, gdje učenik može pohađati barem neke kurseve na kvalitetnim institucijama, odnosno na institucijama koje drže poznati stručnjaci bez promjene mjesta boravka,

☐ odabiranje svog načina učenja - aktivno ili pasivno učeničko učenje, različiti nivoi interakcije, kao što su: „klasični“ pisani materijali uz vođenje vlastitih bilješki, interaktivne simulacije, diskusija sa ostalim učenicima (*e-mail, on-line forumi, chat, mailing liste, videokonferencije*),

☐ odabiranje multimedije - grafike, animacije, zvuka, ...,

☐ praktičan rad učenika sa različitim savremenim IKT-a, gdje učenici stižu ne samo informacije o onome što se uči, nego i dodatna znanja i vještine o korišćenju različitih IKT-a.

Internet daje obrazovanju, potpuno novu dimenziju. Srednja škola može i mora da prati ove promjene.

3. SISTEM OBRAZOVANJA NA DALJINU

Obrazovanje na daljinu je kompleksan sistem koji podrazumijeva i uključuje sljedeće elemente:

▶ učenje na daljinu (*distance learning*) i predavanje na daljinu (*distance teaching*), koji su odvojeni vremenski i prostorno,

▶ nastavne materijale koji mogu da budu u raznim formama (štampani materijali, audio vizuelni,...),

▶ proces učenja koji može da bude individualni i grupni,

▶ tutorski rad kombinacijom raznovrsnih formi „*face-to-face*“ komunikacija korišćenjem medija,

▶ interaktivni rad kako bi se promovisao viši nivo učenja.

Ako posmatramo obrazovanje na daljinu sa vremenskog aspekta uočavamo četiri faze:

I) **pripremu** (organizacione pripreme, pripremu sadržaja, materijala i preduslove koje mora učenik da zadovolji),

II) **aktivnosti** učenika u toku samog obrazovnog procesa,

III) **interakciju** učenika sa nastavnim sadržajem, evaluaciju stečenog znanja i

IV) **transfer znanja**.

Sistem obrazovanja na daljinu ne može se posmatrati odvojeno od IKT-a, multimedije, Interneta, odnosno Intraneta.

4. ZAKLJUČNA RAZMATRANJA

Prelaz sa tradicionalnog načina učenja na aktivno sticanje znanja predstavlja elektronsko učenje na daljinu koje se nudi učenicima koji su na različitim mjestima, odnosno fizički udaljeni od profesora ili izvora informacija. Primjena se zasniva na interaktivnoj komunikaciji *učenik ↔ profesor* kroz sistem za elektronskog učenja (*slika 1*) koji se nudi

učenicima uz primjenu informaciono komunikacionih tehnologija i Interneta.

Učenje na daljinu je obrazovna interaktivna komunikacija između učenika i profesora koja se realizuje pomoću informaciono komunikacionih tehnologija i Interneta, ima svoju didaktičku, logičku i metodičku artikulaciju, svoju svrhu, uzročnost i posledice, svoje strategije i ciljeve, kao i resurse radi realizacije stvaralačkog maksimuma i zadovoljavanja individualnih potreba.

Učenje na daljinu može se definisati i kao oblik edukacije u kojem se nastavni sadržaji prezentiraju učenicima putem Interneta (web stranicama). Pri čemu je osnovni zahtjev koji se postavlja pred učesnike obrazovnog procesa posjedovanje mreže računara sa stalnim pristupom na Internetu.

Sistem za elektronsko učenje na daljinu nudi učenicima digitalne statičke materijale za učenje putem weba (digitalne udžbenike), i *on-line* komunikaciju sa profesorom (*e-mailom, chat-om, video konferencijom, ...*), slika 1.

Cilj rada je modernizacija obrazovnog procesa, obzirom da je rast znanja veoma brz i zahtijeva konstantno predefinisane nastavnog plana i programa i implementaciju novih metoda za efikasno učenje. Modernizaciju bi trebalo izvesti u skladu sa standardima i modelima Evropske Unije.

Učenje na daljinu mijenja navike i učenika i profesora. Uspješni učenici razvijaju upornost, organizacione sposobnosti, imaju pristup kompletnoj bazi materijala za učenje i razvijaju veću autonomnost u procesu učenja, a profesori postaju vještiji u upotrebi informaciono komunikacionih tehnologija, Interneta, PC-a i ulogu profesora pretvaraju u ulogu mentora ili trenera.

Škola kao institucija čiji je osnovni zadatak osposobljavanje mladih generacija za susret sa učenjem na daljinu, ima posebnu obaveznu da se tim tokovima što prije priključi.

LITERATURA

[1] Jezdimir - Luka Obadović, *Informational society focused on european internet penetration*, 4th International Conference for Quality Management and Environment – ICQME09 (Quality, Management, Environment, Education, Engineering), pp. 463-469, France, Paris, 2009.

[2] Jezdimir - Luka Obadović, *Koncept sistema kvaliteta na fakultetu uz podršku informaciono komunikacionih tehnologija*, Poslovna politika, Kvalitet - specijalizovani časopis za unapređenje kvaliteta, Broj 7-8, str. 78-81, Beograd, 2009.

[3] Jezdimir - Luka Obadović, *Indikatori razvoja informaciono komunikacionih tehnologija u školstvu*, V Međunarodni simpozijum - Tehnologija, informatika i obrazovanje za društvo učenja i znanja (TIO5), Fakultet tehničkih nauka - Novi Sad, Centar za razvoj i primenu nauke, tehnologije i informatike - Novi Sad, Institut za pedagoška istraživanja - Beograd, Rezime radova str. 44, Novi Sad, 2009.

[4] Učenje na daljinu, <http://www.link-elearning.com>, web sajt posjećen 10. januara 2010.

ZNAČAJ MODELOVANJA FIZIČKOG KANALA U SIMULACIJI BEŽIČNIH MREŽA IMPORTANCE OF PHYSICAL CHANNEL MODELING IN WIRELESS NETWORKS SIMULATION

Maja Ilić-Delibašić, Božo Krstajić, *Elektrotehnički fakultet, Podgorica*

Sadržaj – Većina simulatora bežičnih telekomunikacionih mreža ne raspolaže adekvatnim modelom mobilnog kanala. U ovom radu je ukazano na činjenicu da veliki uticaj na rezultate simulacije protokola viših nivoa ima model fizičkog kanala koji je primijenjen u simulatoru. Simulacije su vršene korišćenjem ns2 simulatora u koji je implementiran model Rice-ovog i Rayleigh-jevog fedinga. Postojanje fedinga izaziva značajne varijacije primljene snage signala, tako da, čak i kada je nivo srednje snage signala na prijemu znatno iznad praga prijema, dolazi do gubitaka paketa i degradacije performansi sistema.

Abstract – Most packet level network simulators lack a realistic wireless channel module which would allow having accurate simulations. This paper points out the fact that physical channel model implemented in simulator significantly affect simulation results of higher level protocols. Simulations are done using ns2, as academically most used simulator, in which Ricean and Rayleigh fading module is implemented. Fading causes significant variations of the received power, so, even in the case that average received power is above the predefined threshold, packet losses occur, that causes system performance degradation.

1. UVOD

Osnovni trend savremenih komunikacionih sistema je integracija svih telekomunikacionih servisa i omogućavanje pristupa i razmjene različitih sadržaja, uz pružanje potpune mobilnosti korisnika. Stoga je poslednjih godina evidentan intenzivan razvoj raznih vrsta bežičnih komunikacionih sistema, bilo da je riječ o mrežama lokalnog karaktera (WLAN – *Wireless Local Area Networks*), pa do onih koje pokrivaju teritorije cijelih država (npr. WiMAX – *Worldwide Interoperability for Microwave Access*, UMTS – *Universal Mobile Telecommunications System*, ...). Jasno je, stoga, da se vrše intenzivna istraživanja u ovoj oblasti, bez obzira o kojoj vrsti bežičnih mreža je riječ. Danas postoje različiti softveri za simulaciju telekomunikacionih mreža. Neki od najčešće korišćenih simulatora su OPNET [1], QualNet [2], ns2 [3], GlomoSim [4] i OMNeT++ [5]. Zahvaljujući svojoj *open-source* i *freeware* platformi ns2 je široko prihvaćen u istraživačkim projektima. Međutim, jedan od osnovnih nedostataka pomenutih simulatora, kada je u pitanju simulacija bežičnih mreža, je nedostatak adekvatnog modela fizičkog kanala. Naime, prilikom ispitivanja protokola viših nivoa, prvenstveno kreiranih za računarske mreže, koje su originalno podrazumijevale servis prenosa podataka žičnim putem, javlja se problem njihove tranzicije u bežične mreže. Problem se javlja prvenstveno zato što u bežičnim mrežama, zbog veće nepouzdanosti kanala, ne važi pretpostavka koja je validna za žične mreže, a to je da je uzrok gubitaka isključivo zagušenje, dok se greške nastale na linku zanemaruju. U uslovima fedinga, čak iako je nivo srednje snage signala značajno iznad praga prijema, fluktuacije snage mogu biti značajne, tako da trenutna snaga signala u određenim intervalima pada ispod praga prijema, uzrokujući izvjesne gubitke. Stoga je jasno da je pouzdana simulacija fizičkog nivoa u bežičnim mrežama veoma bitna za ispitivanje i optimizovanje performansi protokola viših nivoa [6], [7].

Integrisanje fizičkog sloja u pomenute mrežne simulatore je komplikovana procedura. U tabeli 1 su dati podaci o

dostupnim modelima bežičnog kanala u nekim od najčešće korišćenih simulatora.

Tabela 1. Propagacioni modeli dostupni u nekim mrežnim simulatorima

Simulator	GloMoSim	ns2	OPNET
Računanje odnosa signal-šum	Kumulativno	Poređenje dva signala	Kumulativno
Prijem signala	Na osnovu SNRT i BER	Na osnovu SNRT	Na osnovu BER
Feding	<i>Rayleigh, Rice</i>	Nije uključen	Nije uključen
Gubici usled prostiranja (<i>path loss</i>)	Slobodan prostor, <i>two ray</i>	Slobodan prostor, <i>two ray</i>	Slobodan prostor

Sa druge strane, simulacije bežičnog kanala u specijalizovanim simulatorima [8], [9], [10], [11] se obično rade nezavisno od ispitivanja performansi protokola viših nivoa. Hardverski simulatori fedinga se obično koriste za testiranje prototipova i za reprodukovanje uslova neke sredine u laboratoriji. Softverski simulatori fedinga se češće koriste za ispitivanje i modelovanje sistema.

Da bi se simulatorima, kao što su OPNET i ns2 postigli realniji rezultati pri ispitivanju protokola viših nivoa bežičnih mreža, potrebno je dodati adekvatan model kanala sa fedingom, koji će unijeti, što je moguće manju dodatnu kompleksnost. Obično se za modelovanje efekata fedinga koristi generator slučajnih brojeva, koji zadovoljava odgovarajuće statističko ponašanje. Nedostatak ovakvog pristupa je u tome što se na ovaj način ne može opisati vremenska korelacija anvelopa signala, koja je potrebna za pravilno modelovanje tzv. *burst* grešaka. Vremenska korelacija se može postići odgovarajućim filtriranjem generisanih slučajnih brojeva. Međutim, na ovaj način se povećava složenost računanja, kao i samog modela. Kompromis je odraditi određene proračune nad generisanim

podacima prije same simulacije, čime se u toku simulacije obavlja samo mali broj jednostavnih matematičkih operacija [12].

Cilj ovog rada nije kreiranje modela fizičkog nivoa i integracija u neki od pomenutih mrežnih simulatora, niti ispitivanje preformansi protokola viših nivoa u zavisnosti od modela kanala koji se koristi, već samo da se ukaže na činjenicu da izbor tipa fizičkog kanala koji se implementira ima uticaj na performanse protokola viših nivoa, što se često zanemaruje. Simulacije su vršene korišćenjem ns2 simulatora, kao najčešće korišćenog u akademske svrhe.

Rad je organizovan na sledeći način: nakon uvoda, dat je pregled osnovnih tipova fedinga u bežičnim mrežama. Potom je u poglavlju 3 opisan simulacioni model i dati su rezultati dobijeni simulacijom, i konačno, u poglavlju 4 su data zaključna razmatranja.

2. FEDING KARAKTERISTIKE

Prostiranjem signala od izvora do odredišta, bez obzira da li je riječ o žičnom ili bežičnom prenosu, signal slabi, što se obično opisuje eksponencijalnom krivom, čiji stepen zavisi od vrste kanala. Kada je riječ o bežičnom kanalu, višestruke refleksije, a uz to i mobilnost korisnika dodatno uslozjavaju situaciju. Pri opisivanju bežičnog/mobilnog kanala obično se pominju termini spori feding (*shadowing*) i brzi feding (*multipath*). Shadowing uzrokuje sam profil terena između predajnika i prijemnika, postojanje velikih prepreka, kao što su brda, visoke zgrade, drveće i sl. Varijacije anvelope primljenog signala prouzrokovane shadowing-om se obično modeluju log-normalnom raspodjelom, pa se često u literaturi još naziva i log-normalni feding. Razmatranje efekata *shadowing*-a je potrebno kako bi se postavila odgovarajuća vrijednost praga prijema.

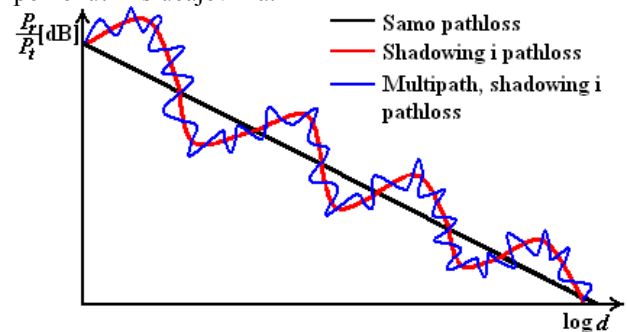
Osim ovih sporijih varijacija snage primljenog signala, višestruke refleksije signala na putu od izvora do odredišta uslovljavaju da na mjesto prijema, osim originalnog signala stigne i više njegovih replika, oslabljenih za različite vrijednosti, i pristiglih u različitim vremenskim trenucima. To će izazvati varijacije anvelope (i snage) primljenog signala koje se dešavaju na kraćim rastojanjima, reda talasne dužine signala, što je u literaturi poznato kao „brzi“, tj. *multipath* feding. Varijacije primljene snage signala mogu da budu i veoma značajne, tako da čak i u uslovima relativno velike srednje snage primljenog signala, određeno vrijeme prijem ili detekcija signala nisu mogući. U mobilnim komunikacionim sistemima, za opisivanje multipath fedinga se obično koriste sledeći modeli:

- *Two-ray ground* (TRG) – idealizovan model propagacije signala u slobodnom prostoru, tako da na mjesto prijema dolazi direktan talas, kao i jedan talas reflektovan o površinu zemlje, kao jedine prepreke.
- *Rayleigh*-jev feding – statistički model koji opisuje slučajne varijacije anvelope signala prateći *Rayleigh*-jevu raspodjelu. Ovaj model se obično koristi za modelovanje bežičnog kanala u kome se na putu od predajnika do prijemnika nalazi više prepreka, tako da na mjesto prijema dolazi više reflektovanih talasa, dok je direktni blokiran preprekom (tzv. NLOS – *Non Line of Sight*). Ovaj model se

obično koristi za opisivanje propagacije u gustim urbanim sredinama.

- *Rice*-ov feding – model koji opisuje promjene anvelope signala na prijemu koji se do prijemne strane takođe prostire po više putanja (više reflektovanih talasa), ali za razliku od *Rayleigh*-jevog modela, postoji i dominantna komponenta signala (LOS – *Line of Sight*). Tipični parameter za opisivanje *Rice*-ovog fedinga je tzv. *Rice*-ov (*K*) faktor, koji predstavlja odnos snage direktnog i svih reflektovanih talasa. Tipičan je za mikrocelularne urbane sredine, suburbane sredine, indoor okruženja, i sl.
- *Nakagami-m* feding – nešto opštiji model koji obuhvata propagaciona okruženja između *Rayleigh*-jevog i *Rice*-ovog fedinga.

Na sl. 1 je ilustrovan uticaj kanala na snagu signala u pomenutim slučajevima.



Sl.1: Pathloss, shadowing i multipath feding u funkciji rastojanja

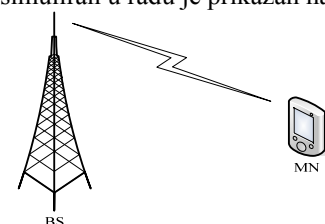
3. SIMULACIONI MODEL

Većina simulatora mrežnih protokola bežičnih mreža koristi jednostavne modele kanala, kao što je, npr. *two-ray ground* model, koji ne podrazumijeva i efekte *shadowing*-a. To znači da je na fiksnom rastojanju d od bazne stanice, nivo primljene snage P_r svuda jednak, i deterministički određen. Za dovoljno velika rastojanja određuje se prema relaciji:

$$P_r = P_t G_t G_r h_t^2 h_r^2 / d^4$$

, gdje su P_r i P_t primljena i emitovana snaga, G_r i G_t dobici prijemne i predajne antene, h_r i h_t njihove visine, a d je rastojanje između antena. Ns2 simulator posjeduje modele za propagaciju u slobodnom prostoru, *two-ray ground* i *shadowing*. U [12] je uveden i potvrđen model *Rice*-ovog i *Rayleigh*-jevog fedinga u ns2 simulator, koji se može naći u [14].

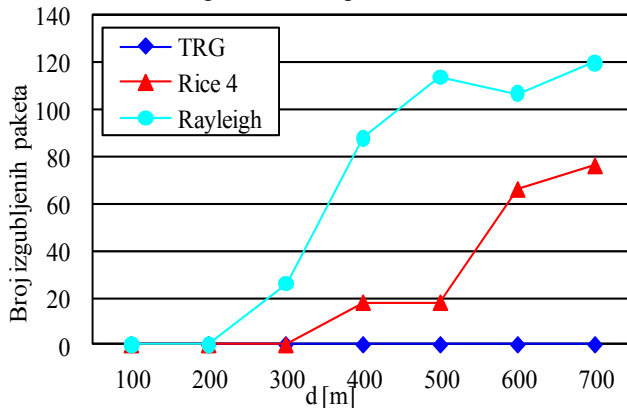
Za sve simulacije u ovom radu je korišćen ns2 simulator, verzija 2.29. Simulacije su vršene na primjeru mobilnog WiMAX-a, a kao ns2 modul je korišćena IEEE 802.16 implementacija [15] sa paketom mobilnosti koji je razvio NIST (*National Institute of Standards and Technology*). Scenario koji je simuliran u radu je prikazan na sl. 2.



Sl.2: Simulacioni scenario

Bazna stanica (BS) komunicira sa mobilnim čvorom (MN – *Mobile Node*). Izlazna snaga predajnika je 20dBm, antena je omnidirekciona, a prag prijema na strani MN iznosi -98.98 dBm. Model saobraćaja MN-a je TCP CBR (*Constant Bit Rate*). Treba napomenuti da se u radu ispituje samo uticaj fizičkog kanala na broj gubitaka i propusnost TCP konekcije, stoga je podešeno da je kapacitet linka dovoljno veliki kako ne bi došlo do zagušenja.

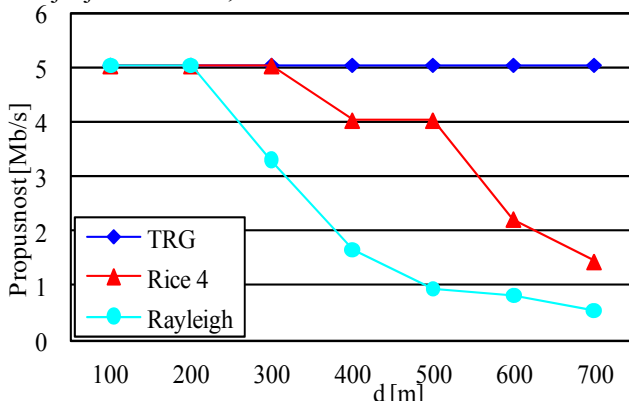
U prvom scenariju MN se ne kreće. Da bi se ispitala propusnost konekcije i broj izgubljenih paketa, mijenjano je rastojanje MN od BS, i to od 100m do 700m od BS. Razmatrani su kanal sa *Rice*-ovim fadingom, sa K faktorom 4 (6dB), i *Rayleigh*-jevom fadingom ($K=0$). Radi poređenja, dati su i rezultati sa TRG (*Two-Ray Ground*) koji je implementiran u osnovnoj verziji ns2. Saobraćaj počinje u 5. sekundi simulacije, i simulacija traje 50s. Na sl. 3 je prikazan broj izgubljenih paketa tokom trajanja simulacije za različita rastojanja MN i BS. U slučaju TRG kanala, obzirom da nema varijacija nivoa snage signala, koji je, do rastojanja obuhvaćenog simulacijom bio iznad unaprijed definisanog praga prijema, kanal ne uzrokuje gubitke paketa. U preostala dva slučaja, fading uzrokuje gubitke većeg broja paketa, čime se direktno utiče na performanse protokola viših nivoa.



Sl.3: Broj izgubljenih paketa u funkciji rastojanja MN od BS

Kako *Rice*-ov faktor može da ima vrijednost od 0 (što odgovara *Rayleigh*-jevom fadingu) do beskonačnosti (bez fadinga), jasno je da će se broj izgubljenih paketa, za različite vrijednosti K faktora, kretati između vrijednosti u slučaju kanala sa *Rayleigh*-jevom fadingom i TRG, koje su prikazane na sl. 3.

Za isti simulacioni scenario, na sl. 4 je predstavljena zavisnost propusnosti razmatrane TCP konekcije od rastojanja MN od BS, u različitim uslovima u kanalu.

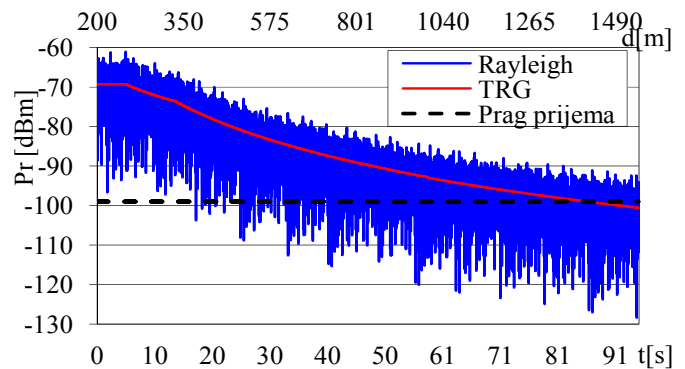


Sl.4: Propusnost konekcije u funkciji rastojanja MN od BS

U slučaju TRG modela kanala, rastojanje MN od BS nema uticaja na propusnost TCP konekcije. U slučaju modela kanala sa *Rice*-ovim i *Rayleigh*-jevom fadingom se dobija realniji scenario u kojem se propusnost TCP konekcije smanjuje pri povećanju rastojanja MN od BS, a kao posledica povećanja broja izgubljenih paketa.

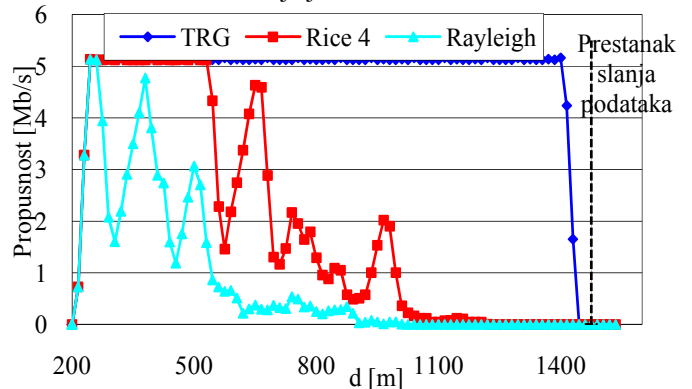
Drugi scenario podrazumijeva kretanje MN, i to brzinom 15m/s. Simulacija traje 100s, pri čemu saobraćaj i kretanje MN počinje nakon 5s trajanja simulacije. MN je na početnoj udaljenosti 200m od BS. Podaci se šalju do 90-te sekunde trajanja simulacije (što odgovara udaljenosti MN od BS 1475m). Trajanje simulacije i brzina kretanja MN su odabrani tako da MN i u najboljem slučaju (TRG) izađe iz zone pokrivanja BS (nivo primljenog signala pada ispod praga) prije završetka slanja podataka.

Na sl.5 je prikazan nivo primljene snage za slučaj TRG i *Rayleigh*. Varijacije u nivou snage u slučaju kanala sa *Rayleigh*-jevom fadingom dovode do toga da nivo snage ranije pada ispod praga prijema u odnosu na slučaj TRG što dovodi do pojave grešaka, odnosno smanjenja propusnosti kanala. Kod TRG nema varijacija u nivou signala, pa samim tim nema ni grešaka sve do momenta kada nivo snage padne ispod praga, nakon čega dolazi do potpunog prekida komunikacije.



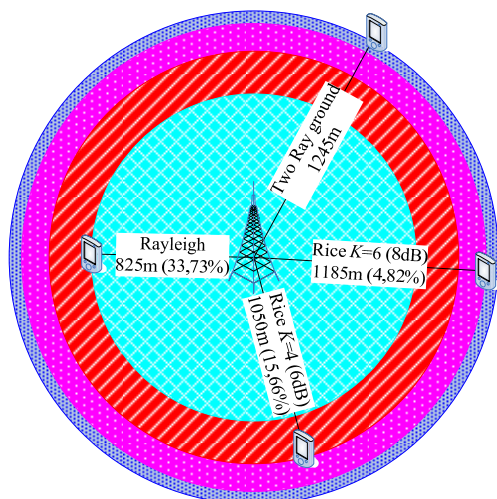
Sl.5: Nivo primljene snage u zavisnosti od vremena i rastojanja MN od BS

Zavisnost propusnosti od udaljenosti MN od BS u svakom trenutku simulacije je data na sl. 6.



Sl.6: Zavisnost propusnosti od rastojanja

Na sl. 7 je prikazana oblast pokrivenosti signalom, odnosno maksimalna dimenzija ćelije, za različita propagaciona okruženja. Takođe su date procentualne vrijednosti smanjenja oblasti pokrivanja u kanalu sa fadingom, u odnosu na kanal opisan preko TRG.



Sl.7: Oblast pokrivenosti

Sa slike se vidi da pri istim uslovima simulacije, što je feding intenzivniji, radijus pokrivanja može da se smanji i do 1/3 u odnosu na TRG.

4. ZAKLJUČAK

Prostiranjem signala od izvora do odredišta, bez obzira da li je riječ o žičnom ili bežičnom prenosu, signal slabi, što se obično opisuje eksponencijalnom krivom, čiji stepen zavisi od vrste kanala. Kada je riječ o bežičnom kanalu, višestruke refleksije, a uz to i mobilnost korisnika dodatno usložnjavaju situaciju. Tada je neophodno razmatrati efekte shadowing-a, kako bi se postavila odgovarajuća vrijednost praga prijema, ali i multipath fedinga, koji uslovljava varijacije primljene snage signala oko njene srednje vrijednosti. Te varijacije mogu da budu i veoma velike, tako da čak i u uslovima relativno velike srednje snage primljenog signala, određeno vrijeme prijema ili detekcija signala nisu mogući.

U ovom radu je ukazano na neke od osnovnih problema paketskih simulatora koji se koriste za ispitivanje preformansi protokola viših nivoa. Većina najčešće korišćenih simulatora ne sadrži adekvatan model bežičnog kanala. Obično je riječ o modelima koji opisuju prostiranje signala u slobodnom prostoru, ili u okruženjima u kojim jedinu prepreku čini površina zemlje. Pomenuti modeli gotovo nikada nisu zastupljeni u praksi, a efekti koje uzrokuje realan kanal na ponašanje protokola viših nivoa nisu zanemarljivi.

Na osnovu dobijenih rezultata se može zaključiti da kada se primijeni TRG model, nema gubitaka i nema promjene propusnosti sve dok se MN nalazi u zoni pokrivanja. Dakle, prilikom simulacije bežične mreže koristeći TRG model (što je najčešći slučaj), kanal se može smatrati praktično idealnim (bez gubitaka) sve dok nivo signala na prijemu ne padne ispod nivoa praga prijema, nakon čega su gubici 100%. Kako simulacije protokola viših nivoa u bežičnoj mreži imaju za cilj upravo to da se ispita različito ponašanje uzrokovano gubicima na bežičnom linku, onda je jasno da simulacija sa

TRG nije adekvatna. Prikazani rezultati pokazuju da, ukoliko se ne koristi realniji model kanala, dobijene rezultate treba smatrati prilično oportunističkim.

Treba napomenuti da u ovom radu nisu razmatrani neki veoma bitni efekti koje uzokuje realan bežični kanal. Shadowing nije uključen u razmatranje, iako će i njegovo postojanje dovesti do dodatnih varijacija snage signala. Razmatrana je jedna, izolovana ćelija, sa samo jednim korisnikom, i pretpostavljeno je da u sistemu ne postoji interferencija.

Imajući u vidu sve gore navedeno, jasno je da se korišćenjem propagacionog modela koji se nalazi u osnovnoj verziji većine simulatora bežičnih telekomunikacionih mreža ne dobijaju realni podaci, te da kreiranje odgovarajućeg modela fizičkog kanala, koji će obuhvatiti veliki broj parametara predstavlja izazov.

LITERATURA

- [1] OPNET Technologies, Inc., <http://www.opnet.com/>.
- [2] <http://www.scalable-networks.com/products/qualnet/>
- [3] Network Simulator - ns2, <http://www.isi.edu/nsnam/ns/>.
- [4] GloMoSim, <http://pcl.cs.ucla.edu/projects/glomosim/>.
- [5] <http://www.omnetpp.org/>
- [6] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "Performance comparison of multi-hop wireless ad hoc network routing protocols," *Proc. 4th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking*, Oct. 1998.
- [7] R. J. Punnoose, P. V. Nikitin, J. Broch, and D. D. Stancil, "Optimizing wireless network protocols using real-time predictive propagation modeling," *Proc. of IEEE Radio and Wireless Conference (RAWCON)*, pp. 39–44, August 1999.
- [8] H. R. Anderson, "A ray-tracing propagation model for digital broadcast systems in urban areas," *IEEE Trans. on Broadcasting*, vol. 39, no. 3, pp. 309–317, September 1993.
- [9] V. Fung, T. S. Rappaport, and B. Thoma, "Bit error simulation for $\pi/4$ DQPSK mobile radio communications using two-ray and measurement-based impulse response models," *IEEE Journal on Sel. Areas in Comm.*, vol. 11, no. 3, pp. 393–405, 1993.
- [10] J. Espinosa, M. Montero, J. M. Hernando, F. Perez, J. A. Milanes, I. Hidalgo, and M. Gallardo, "Coverage simulator for wireless networks," *Proc. of IEEE Int. Workshop on Factory Comm. Systems*, pp. 149–156, 1998.
- [11] M. Patzold, R. Garcia, and F. Laue, "Design of high-speed simulation models for mobile fading channels by using table look-up techniques," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 4, pp. 1178–1190, July 2000.
- [12] R. J. Punnoose, P.V. Nikitin, and D.D. Stancil, "Efficient Simulation of Ricean Fading within a Packet Simulator", *52nd Vehicular Technology Conf.*, IEEE VTS-Fall, 2000.
- [14] CMU additions to NS to handle Ricean and Rayleigh fading, <http://www.ece.cmu.edu/wireless/>
- [15] <http://w3.antd.nist.gov/seamlessandsecure/pubtool.shtml#tools>

SAVREMENI TRENDOVI U INDUSTRIJSKIM KOMUNIKACIJAMA CONTEMPORARY TRENDS IN INDUSTRIAL COMMUNICATION

Miloš Perunović, Milovan Radulović, *Elektrotehnički fakultet, Podgorica*

Sadržaj – U radu su prikazani savremeni trendovi iz oblasti industrijskih komunikacija. Ukratko je napravljeno poređenje različitih komunikacionih protokola sa njihovim prednostima i nedostacima, kao i razlozi prelaska sa analogne na digitalnu komunikaciju tehnologiju. Opisana je primjena modifikovanih standardnih računarskih mreža u industrijskim postrojenjima kao standard koji nalazi sve širu primjenu u industrijskim mrežnim komunikacijama.

Abstract – The paper presents the current trends in the field of industrial communications. Brief comparison is made of different communication protocols with their advantages and disadvantages, and reasons for the transition from analogue to digital communication technology. Described the application of the modified standard computer networks in industrial plants as a standard that finds wide application in industrial network communications.

1. UVOD

U posljednjih desetak godina različiti proizvođači su razvili i standardizovali veliki broj komunikacionih magistrala i odgovarajućih mrežnih protokola za primjenu u industriji. Ne postoji standard koji se može smatrati najboljim rješenjem, već svaka od mnoštva industrijskih magistrala, posjeduje neke specifične karakteristike koje određuju njenu oblast primjene. Danas je u upotrebi veliki broj standardizovanih komunikacionih mreža – industrijskih magistrala. Izbor odgovarajućeg standarda zavisi od zahtjeva konkretne primjene, ali i od faktora kao što su dostupnost uređaja koji podržavaju konkretni standard i njihove cijene.

Ukoliko napravimo poređenje industrijskih i standardnih računarskih mreža, možemo uvidjeti da između njih postoje mnoge zajedničke osobine. Računarske mreže omogućavaju komunikaciju između velikog broja računara, a da pri tome računari ne moraju biti direktno, fizički povezani, svaki sa svakim. Svaki računar ima samo jednu vezu sa mrežom. Računarske mreže, takođe, omogućavaju da pojedini uređaji, budu dostupni svim računarima na mreži. Slični ciljevi se postavljaju i pred industrijske mreže, s tom razlikom da sada mrežni čvorovi nisu računari već industrijski I/O uređaji i PLC kontroleri. Pored toga, komunikacioni protokoli koji se primjenjuju u industriji moraju da zadovolje tri osnovna zahtjeva: bezbjednost poruka, pouzdanost rada i maksimalni nivo determinističkog ponašanja.

2. RAZLOZI PRELASKA SA ANALOGNE NA DIGITALNU KOMUNIKACIONU TEHNOLOGIJU

Kod analogne tehnologije su korišćeni su strujni (4-20mA) ili naponski nivoi ($\pm 10V$). Za povezivanje senzora i aktuatora sa centralnim računarom korišćen je veliki broj paralelnih postavljenih kablova. U sistemima sa velikim brojem senzora, aktuatora i drugih uređaja, takvo rješenje je previše komplikovano za projektovanje i montažu, a posebni problemi nastaju pri detekciji i otklanjanju nekog kvara upravo zbog nepreglednosti takvih sistema. Digitalni prenos podataka omogućio je veću preciznost od analognog, što je bio osnovni razlog za razvoj magistrala uređaja. U suštini, industrijske magistrale zamjenjuju koncept centralizovanog upravljanja, konceptom distribuiranog upravljanja. Iz tog razloga, industrijske magistrale su mnogo više od proste

zamjene analognog standarda. Industrijske magistrale doprinose povećanju kvaliteta, smanjenju troškova i povećanju efikasnosti proizvodnje. Sve ove prednosti, u velikoj mjeri su posljedica činjenice da se prenos informacija obavlja u digitalnom obliku.

RS232 je prvi standardni interfejs za serijski prenos podataka. Glavni nedostaci su: malo rastojanje na kome se podaci mogu razmjenjivati (do 15 m), relativno mala brzina prenosa (do 20 Kb/s) i mogućnost povezivanja samo jednog predajnika i prijemnika. Najznačajniji interfejs za industrijsku primjenu u ovoj grupi je svakako RS-485 (poznat i pod nazivima EIA-485 ili TIA/EIA-485). RS-485 je standard koji definiše električne karakteristike interfejsa, i omogućava konfigurisanje jeftinih lokalnih mreža sa *multidrop* komunikacionim linkovima.

3. INDUSTRIJSKE MAGISTRALNE U KOMUNIKACIONIM MREŽAMA

Magistralne uređaja su digitalne, bidirekzione, *multidrop*, serijske komunikacione mreže koje se koriste za povezivanje izolovanih *Field* uređaja (kontroleri, transduktori, aktuatori i senzori). Za svaki *Field* uređaj je tipično da posjeduje ugrađen mikrokontroler, što ove uređaje čini „pametnim“ uređajima. Pametni uređaji su u mogućnosti da samostalno obavljaju jednostavne funkcije, kao što je dijagnostika, upravljanje i bidirekciona komunikacija. Takođe, u stanju su da automatski izvještavaju o nastalim kvarovima ili o potrebi kalibracije. Ovi uređaji ne samo da omogućavaju pristup sa daljine, već su često sposobni da komuniciraju jedni sa drugima. U oblasti upravljanja sensorima i aktuatorima, kao i akviziciji podataka primjenjuje se poseban oblik lokalne mreže – magistrala uređaja ili *Fieldbus*. *Fieldbus* se koristi za povezivanje sa decentralizovanim ulazima/izlazima i za vremenski kritičnu komunikaciju između PLC-ova. Pored *Fieldbus* postoje i mreže zasnovane na CAN protokolu koji čine osnovni CAN i viši CAN protokoli.

Komunikaciona mreža koja se primjenjuje u industriji mora da ima odgovarajuće performanse u pogledu iskorišćenosti mreže, propusnog opsega i maksimalnog kašnjenja u prenosu. Performanse i pouzdanost automatizovanih industrijskih sistema u velikoj mjeri su uslovljeni karakteristikama komunikacione mreže. U mnogim

industrijskim postrojenjima prisutni su brojni izvori električnih šumova i smetnji. Uticaj tih smetnji može rezultovati pojavom grešaka u prenosu informacija što može značajno degradirati performanse mreže. Može se reći da je komunikaciona mreža najosjetljivija komponenta automatizovanog industrijskog sistema.

Pri izboru magistrale, najbitniji faktori koje treba uzeti u obzir su: maksimalno dozvoljeno vrijeme odziva, značaj informacija za bezbjedan rad opreme, količina informacija koju treba prenositi, maksimalno rastojanje između umreženih uređaja, namjena uređaja koji se povezuju, mogućnost proširenja i sigurnost.

Dvije osnovne kategorije industrijskih komunikacionih mreža su: magistrale uređaja (eng. device bus) i magistrale procesa (eng. process bus).

Magistrale uređaja su optimizovane za razmjenu kratkih informacija o statusu i upravljačkih poruka, za razliku od klasičnih računarskih mreža gdje propusna moć pri prenosu velikih količina podataka predstavlja ključnu performansu.

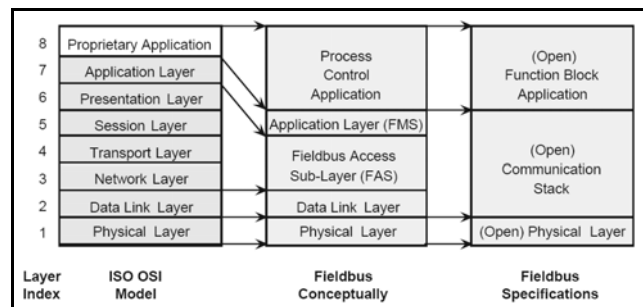
Magistrale procesa koriste se za prenos parametara rada kontrolerima procesa koji su u većini slučajeva analogni uređaji. Namijenjene su prenosu dužih poruka (paketa), dužine do nekoliko stotina bajtova, što ih čini sporijim, jer su paketi duži. Od većine analognih uređaja ne zahtijeva se brzi odziv, zato što se koriste za upravljanje procesima koji su po svojoj prirodi spori: protok fluida, koncentracija gasa, temperatura. „Pametni“ (eng. smart) uređaji su oni koji mogu samostalno da obavljaju naprednije funkcije upravljanja. *Peer* nivo se odnosi na mogućnost komunikacije između kontrolera procesa, npr. PLC kontroleri. Zatim slijede nivoi ćelije, proizvodnog pogona i nivo cjelokupne fabrike. Ne postoji magistrala koja pokriva sve nivoe. Razlog za to leži u činjenici da različiti nivoi postavljaju različite zahtjeve. Pri tome treba znati da magistrala nije samo komunikaciona linija, već i skup pravila, odnosno, protokola, kojih uređaji spregnuti magistralom moraju da se drže.

4. NAJZNAČAJNIJI FIELDBUS PROTOKOLI

Specifični zahtjevi korisnika i proizvođača usloveli su da se u prvo vrijeme na tržištu pojavi više komunikacionih sistema. Da bi se ta djelimična konfuzija prevazišla, izvršeno je integrisanje osnovnih karakteristika glavnih komunikacionih tehnika u dva standarda, IEC 61158 i IEC 61784.

Fieldbus tehnologija razvijena je osamdesetih godina prošlog vijeka sa namjerom da se tradicionalno korišćeni prenos analognim signalom (4-20mA i ±10V) zamijeni sa digitalnom tehnologijom. Osnovni *Fieldbus* protokoli su *Modbus*, *PROFINet*, *Profibus*, *Ethernet/IP*.

Modbus je serijski protokol koji predstavlja nadgradnju RS-232, sa ciljem njegove primjene kao industrijskog komunikacionog protokola. Namijenjen je za povezivanje računara, PLC-ova, operatorskih terminala, senzora, motora i drugih tipova I/O uređaja.



Sl. 1 – Poređenje *Fieldbus* i OSI modela

Osnovu ovog protokola (nivo 1 i djelimično nivo 2 OSI referentnog modela) predstavlja RS-232 kompatibilni serijski interfejs koji definiše parametre poput izlaznih pinova konektora, naponskih nivoa signala, prenosni medijum za prenos poruka, brzinu prenosa, provjeru parnosti i slično. Većina *Modbus* uređaja komunicira preko fizičkog sloja serijskog protokola EIA-485. U sljedećoj tabeli prikazane su osnovne uporedne karakteristike protokola koji se koriste u *Fieldbus* tehnologiji.

Naziv protokola	Napajanje preko bus-a	Redundantno kabl.	Maks. broj uređaja	Sinhronizacija
AS-Interface	Da	Ne	62	Ne
CANOpen	Ne	Ne	127	Da
ControlNet	Ne	Da	99	Ne
CC-link	Ne	Ne	64	Ne
DeviceNet	Da	Ne	64	Ne
EtherCAT	Ne	Opcija	65536	Da
Ethernet Powerlink	Ne	Opcija	240	Da
Ethernet/IP	Ne	Opcija	Nije ograničen	U razvoju
Interbus	Ne	Ne	511	Ne
LonWorks	Ne	Ne	32000	Ne
Modbus	Ne	Ne	246	Ne
PROFIBUS	Ne	Ne	126	Ne
PROFINET IO	Ne	Opcija	Nije ograničen	Ne
PROFINET IRT	Ne	Opcija	Nije ograničen	Da
SERCOS III	Ne	Da	511	Da
SERCOS interface	Ne	Ne	254	Da

Tabela 1 – Karakteristike *Fieldbus* tehnologije

Neka od ograničenja *Modbus*-a su posljedica toga što je on nastao krajem sedamdesetih godina, pa postoji ograničenje broja tipova podataka na one koje su „razumjeli“ PLC kontroleri tog vremena. *Modbus* isključivo tretira *master-slave* komunikaciju u kojoj jedan od uređaja (*master*) inicira komunikaciju slanjem upita (eng. query), na šta drugi uređaj (*slave*) odgovara dostavljajući tražene podatke *master*-u ili pak preduzimajući akciju koja je zahtijevana u upitu.

Ovo znači da uređaj ne može po potrebi poslati podatke *master*-u, već *master* mora periodično prikupljati podatke od uređaja u polju i nadgledati promjene. Ograničenje takođe

predstavlja i maksimalan broj od 246 uređaja na mreži koje mogu biti povezane na *master* (Tabela 1).

U posljednje vrijeme kao dominantna solucija na tržištu profesionalne opreme u oblasti industrijske i procesne automatizacije izdvaja se *Profibus*. *Profibus* je standardizovan, otvoren komunikacioni sistem za sva područja primjene u proizvodnoj i procesnoj industriji. Pogodan je za brze, vremenski zahtijevne aplikacije i složene komunikacijske zadatke. Utemeljen je na internacionalnim standardima EN50170 i IEC61158, a u upotrebi je od 1989 godine. Razmjena podataka sa distribuiranim uređajima je ciklična, dok se dodatne informacije, kao što su alarmna stanja ili dijagnostički podaci prenose neciklično (na zahtjev).

Kada je postalo očigledno da će *Profibus* postati jedan od dominantnih *Fieldbus* protokola, javila se potreba da se omogući njegova primjena u sistemima fabričke i procesne automatizacije koji zahtijevaju povišen stepen pouzdanosti i bezbjednosti. *PROFIsafe* je jednokanalni protokol koji se implementira kao nadgradnja na nivo 7 OSI referentnog modela, tako da svi ostali standardni *Profibus* elementi mogu biti nepromijenjeni. To je izuzetno povoljno jer *Profibus* može u funkcionisati u redundantnom modu rada ili u koegzistenciji sa drugim uređajima koji posjeduju samo osnovnu verziju *Profibus* protokola. *PROFIsafe* uzima u obzir sve moguće tipove grešaka koje se u procesnoj komunikaciji mogu desiti: vremensko kašnjenje, gubitak podataka, ponavljanje podatka, nekorektna sekvenca bitova u poruci, degradiran kvalitet poruke, pogrešna adresa i slično. *PROFIsafe* koristi aperiodičnu komunikaciju i može biti korišćen sa RS485, fiber-optičkim ili MBP tehnikama prenosa. Na taj način se istovremeno postiže kratko vrijeme odziva – bitno za primjenu u industrijskoj automatizaciji, i inherentna bezbjednost rada – bitno u procesnoj automatizaciji.

Kod većih sistema, PLC kontroleri i PC računari komuniciraju međusobno, a i sa IT sistemima unutar kontrolnih soba razmjenjujući velike količine podataka, koji se najčešće prenose kroz neeksplozivnu, bezbjednu sredinu, zaštićenu od jakih elektromagnetnih smetnji. Uočavajući da u takvim uslovima i sa takvim komunikacionim zahtjevima standardi kao što su TCP/IP, *Ethernet*, Intranet i Internet imaju značajne prednosti nad *Profibus*-om, NPO je razvila i standardizovala *PROFINet*, komunikacioni protokol zasnovan na *Ethernet*-u koji koristi standardne mehanizme IT komunikacije u realnom vremenu. Pod tim se podrazumijeva upotreba TCP/IP i COM/DCOM (Microsoft Component Object Model i njegovu ekstenziju za distribuirane sisteme – Distributed COM) standarda, trenutno najrasprostranjenijih PC protokola. Prednosti primjene *PROFINet*-a u odnosu na, recimo, *Ethernet* se manifestuju tek ako je komunikacija na nižem nivou, nivou distribuiranog sistema ostvarena primjenom *Profibus*-a sa kojim je *PROFINet* maksimalno kompatibilan.

5. PRIMJENA LAN MREŽA U INDUSTRIJSKOM OKRUŽENJU

Modifikacijom klasičnog *Ethernet*-a nastao je takozvani industrijski *Ethernet* koji predstavlja osnovu za realizaciju fabričkih upravljačkih mreža. Industrijski *Ethernet* (ili *Ethernet/IP*) pripada grupi mreža koje koriste zajednički

industrijski protokol (eng. Common Industrial Protocol – CIP). CIP uključuje sve vrste poruka i servisa za sve aplikacije u automatici, uključujući kontrolu, sigurnost, sinhronizaciju, konfiguraciju i informacije. CIP se može razmatrati kao aplikacioni sloj za *DeviceNet*, *CompoNet*, *ControlNet* i *EtherNet/IP*. Kao nezavisan protokol, CIP pruža korisnicima jedinstvenu arhitekturu za komunikaciju. Sa nezavisnošću medija dolazi mogućnost da se izabere najbolja CIP mreža za korišćenu aplikaciju. *EtherNet/IP* omogućava da sa određenim alatima koristimo standardnu *Ethernet* tehnologiju za industrijske aplikacije. Iako je industrijski *Ethernet* baziran na istom standardu kao i tradicionalna verzija, implementacija ova dva rješenja nije identična. Osnovna razlika između industrijskog i tradicionalnog *Ethernet*-a je u tipu hardvera koji koriste. Oprema za industrijski *Ethernet* je projektovana tako da radi i u grubim industrijskim uslovima. To znači primjenu industrijskih komponenti, specifične metode hlađenja, izlaznu relejnu signalizaciju i slično. Napajanje je standardno industrijsko 24V DC, uz obaveznu primjenu i redundantnog izvora napajanja, u cilju povećanja pouzdanosti. Industrijski *Ethernet* se koristi za komunikaciju između PLC-ova i komunikaciju PLC-ova sa nivoom nadzora.

Na osnovu strukture koju nudi OSI referentni model, može se konstatovati da je tradicionalni *Ethernet* dominantno lociran na njegovom drugom nivou. Specijalizacija koja je učinjena kod industrijskog *Ethernet*-a omogućila je da se detaljnije definišu mrežni nivo i transportni nivo OSI referentnog modela. Na mrežnom nivou industrijskog *Ethernet*-a dominantno se koristi IP protokol. Industrijski *Ethernet* na transportnom nivou uobičajeno koristi dva protokola: TCP, i rjeđe korišćen UDP. Na mrežnom i transportnom nivou takođe su implementirane i CIP poruke. Postoje i slučajevi kada se za prenos koristi UDP protokol, umjesto TCP protokola (npr. kada je blok podataka koji treba poslati mali, veličine jednog paketa).

Industrijski *Ethernet* se odlikuje i u specifičnoj realizaciji *multicast* tipa prenosa. Dok se tradicionalni *Ethernet* dominantno fokusira na što bolje iskorišćenje raspoloživog propusnog opsega, industrijski *Ethernet* prioritet stavlja na specifičan zahtjev za sinhronizovanim pristupom podacima od strane svih prijemnih čvorova. To je u skladu sa zahtjevom za što većim determinizmom u prenosu upravljačkih signala u industrijskom okruženju. Ovaj specifičan zahtjev je ispunjen uključivanjem u industrijski *Ethernet* nekih naprednih funkcija za organizaciju i određivanje prioriteta *multicast* prenosa. Industrijski *Ethernet* se često susreće i pod nazivom *Switched Ethernet*. Ime potiče od činjenice da ovakva *Ethernet* arhitektura isključivo koristi *switch* uređaje, a ne *hub*-ove u implementaciji LAN mreže. Kod industrijskog *Ethernet*-a se skoro isključivo primjenjuje *full-duplex* veza. Implementacijom specifične mrežne inteligencije na industrijski *Ethernet* moguće je realizovati fabričku komunikacionu infrastrukturu koja posjeduje fleksibilnost i zaštitu poruka karakterističnih za tradicionalne *Fieldbus* protokole, uz istovremeno daleko širi propusni opseg, otvorenu arhitekturu i standardizaciju koju nudi *Ethernet* platforma.

Prednosti koje nudi *EtherNet/IP* za aplikacije automatizacije u industriji su: kompletna administrativnost

(kontrola, konfigurisanje i skupljanje podataka sa „inteligentnih“ uređaja preko jedne mreže), i mogućnost korišćenja jedne mreže kao oslonca za višestruko distribuirane CIP mreže. Kompatibilan je sa standardnim Internet protokolima (HTTP, FTP, SNMP i DHCP), sa standardnim industrijskim protokolima (OPC), kao i sa standardnim industrijskim konektorima.

Interesantna je činjenica da je, upoređujući *Profibus* na 12Mbit/s i *Ethernet* na 10Mbit/s, prenesena količina podataka u jedinici vremena veća kod *Ethernet*-a, zahvaljujući protokolima za prenos u većim blokovima.

6. SPECIJALNE OBLASTI PRIMJENE INDUSTRIJSKIH KOMUNIKACIONIH MAGISTRALA

U mnogobrojnim industrijskim, vojnim, medicinskim, naučnim, ekološkim i drugim primjenama, zahtijeva se veoma intenzivno i opsežno prikupljanje podataka i informacija iz fizičkog okruženja, za potrebe nadzora i kontrole. Bežične senzorske mreže – WSN (eng. Wireless Sensor Networks) funkcionišu korišćenjem malih, jeftinih i potrošnih platformi koje osim senzorskih funkcija posjeduju mogućnost samostalnog formiranja *ad-hoc* bežičnih mreža u cilju međusobne komunikacije i dostavljanja prikupljenih podataka korisniku mreže. Primjena *Bluetooth* (IEEE 802.15.1) tehnologije predstavljala je korak ka korišćenju *ad-hoc* bežičnih mreža u cilju povezivanja i umrežavanja velikog broja uređaja i/ili senzora. Da bi upotreba WSN bila ekonomski isplativa, cijena uređaja mora da bude manja od cijene uređaja drugih bežičnih tehnologija, npr. *Bluetooth*. Zahtjevi u smislu potrošnje energije, procesorske snage i komunikacionih mogućnosti moraju da se prilagode tehnologijama izrade i cijeni raspoloživih rješenja.

U automobilskoj elektronici, kontrolna jedinica motora, senzori, sistemi protiv proklizavanja, i slično, povezani su korišćenjem CAN (eng. Control Area Network) mreže. CAN je serijski komunikacioni protokol koji posjeduje veoma visok nivo bezbjednosti i podržava distribuirano upravljanje u realnom vremenu. Namjenski je razvijen za primjenu u motornim vozilima. CAN je protokol zasnovan na porukama i *broadcast* komunikaciji. Pri tome garantuje integritet podataka u cjelokupnoj mreži. CAN je *multi-master* magistrala, što omogućava izgradnju složenih, redundantnih distribuiranih sistema na bazi uređaja različitog nivoa složenosti, a bez potrebe ugradnje centralizovanog, *master* kontrolera. Osnovne prednosti mreža zasnovanih na CAN protokolu su: niska cijena, sposobnost funkcionisanja u različitim okruženjima, podrška radu u realnom vremenu, izuzetno visoka pouzdanost, jednostavna upotreba.

Da bi se ostvarila sinhronizacija svih čvorova na mreži uveden je TTCAN (eng. Time Triggered communication on CAN). To je viši CAN protokol koji obezbjeđuje planiranje i realizaciju vremenski determinisanog rasporeda prenosa poruka (eng. Time Triggered - TT). Tako u vozilima tipično imamo TT poruke od sistema za kočenje i ET poruke od sistema za regulaciju temperature. Uvođenje TT poruka je omogućilo implementaciju zatvorenih upravljačkih petlji na sisteme sa CAN baziranim mrežama. Pored TTCAN u više CAN protokole spadaju *DeviceNet* i *CANopen*. Pored automobilske industrije, TTCAN nalazi primjenu i u domenu

industrijske automatizacije i medicinske opreme, naravno, u daleko manjem obimu.

7. ZAKLJUČAK

Tehnološki napredak i pad cijena u oblasti komunikacione i računarske opreme doveli su do ubrzanog razvoja i primjene računarskih mreža, odnosno, umreženih računarskih sistema.

Industrijski *Ethernet* se u posljednje vrijeme sve više koristi u automatizaciji. Pokazalo se da je to mreža koja je najjednostavnija i najisplativija za primjenu u postojećim sistemima. S obzirom da *Ethernet* postaje dominantno rješenje u realizaciji LAN mreža, mnoge industrijske kompanije su počele da ga primjenjuju, u cilju zamjene tradicionalnih *Fieldbus* arhitektura kao što su *Modbus*, *Profibus* i slično.

CAN magistrala je, zbog svojih dobrih karakteristika i jednostavne ugradnje, našla primjenu u mnogim komercijalnim proizvodima, kao i u automatizovanim industrijskim sistemima. Neke od oblasti gdje se može sresti komunikacija zasnovana na CAN protokolu su poljoprivredna mehanizacija, nautička oprema i mašine, medicinska oprema, tekstilna industrija, liftovski sistemi i oprema specijalne namjene. Proizvođači medicinske opreme su se odlučili za ovaj protokol jer ispunjava njihove veoma stroge bezbjednosne zahtjeve po pitanju pouzdanosti prenosa podataka. Evropska automobilska industrija je prihvatila CAN kao standard.

U oblasti bežičnih senzorskih mreža, dalji razvoj komunikacionih tehnologija, tehnologija izrade senzora malih dimenzija, kao i integrisanih hardversko-sofverskih platformi malih dimenzija, potrošnje energije i cijene, omogućava razvoj i primjenu bežičnih senzorskih mreža.

LITERATURA

- [1] CAN Specification, 1991, Robert Bosch GmbH, Postfach 30 02 40, D-70442 Stuttgart
- [2] K. Etschberger, C. Schlegel, CANopen-based Distributed Intelligent Automation, IXXAT Automation
- [3] The Foundation fieldbus Primer, 2001 Fieldbus Inc.
- [4] Florian Hartwich, Bernd Müller, CANNetwork with Time Triggered Communication, Robert Bosch GmbH
- [5] IEEE Std. 802.15.4a-2007, “Standard for LAN and MAN Networks: Specifications for Low-Rate Wireless Personal Area Network; Amendment A”, 2007.
- [6] Allen-Bradley, <http://www.ab.com>
- [7] Rockwell Automation, <http://www.rockwellautomation.com>
- [8] Profibus, <http://www.profibus.com/>

PRIKAZ RADA NA CENTRALNOM ELEKTRONSKOM KATALOGU POMORSTVA CRNE GORE

REVIEW OF THE WORK ON THE CENTRAL ELECTRONIC LIBRARY CATALOGUE OF MONTENEGRO

Sanja Bauk, Ruža Danilović, *Fakultet za pomorstvo, Univerzitet Crne Gore*

Sadržaj - U radu je dat pregled dosadašnjeg rada na Centralnom elektronskom katalogu pomorstva Crne Gore (CEKP_CG), kao i smjernice za predstojeći rad na istom. Takođe, ukazano je na potrebe finansijske podrške projekta i osnaživanja tima koji na njemu radi, prije svega u smislu osvježavanja znanja angažovanih u domenu informacionih tehnologija, te u domenima adekvatnog prezerviranja, prezentovanja i valorizacije dokumentarnog nasleđa koje Crna Gora ima u oblasti pomorstva.

Ključne riječi - Centralni elektronski katalog pomorstva Crne Gore (CEKP_CG), prezervacija, prezentacija, valorizacija

Abstract - The paper gives a review of the past work on the Central electronic library catalogue of Montenegro (CELC_M), as well as the directions for the forthcoming work on it. Also, the needs for the financial support of the project and for empowering the team working on it, primarily in the sense of knowledge refreshment in the domain of information technologies and in the domains of the adequate preservation, presentation and valorization of the documentary heritage that Montenegro has in the sphere of maritime affairs, have been noted.

Key words - Central electronic library catalogue of Montenegro (CELC_M), preservation, presentation, valorization

1. UVOD

Centralni elektronski katalog pomorstva Crne Gore (CEKP_CG) je zamišljen, prije svega, kao on-line dostupan bibliografski popis bibliotečkog, arhivskog i muzejskog materijala, u pisanom i štampanom obliku, a koji se svojim sadržajem odnosi na pomorstvo.

Rad na projektu je započeo 2003.godine, i iako ekstenzivan, uglavnom zbog nedostatka finansijskih sredstava, nedavno je rezultirao pilot bazom dostupnom na Web-u (www.ckpcg.ac.me). Za prvi kvartal tekuće 2010. godine planirana je prezentacija projekta, s ciljem upoznavanja naše naučne, stručne i šire javnosti sa onim što je do sada urađeno na projektu, te s ciljem što kvalitetnijeg i preciznijeg određivanja smjernica za dalji rad na njemu. Idejni nosilac projekta i glavni koordinator u radu, je nevladina organizacija «Notar» - Centar za njegovanje i prezentaciju dokumentarnog nasleđa Kotora, na čijim Web stranicama se mogu naći detaljne informacije o projektu (www.cdknotar.org).

Inače, CEKP_CG je više puta prezentovan na raznim domaćim i međunarodnim naučno-stručnim skupovima [1;3;6]. Takođe, publikovani su brojni radovi u domaćim i međunarodnim časopisima na ovu temu [2;4;5]. Ovdje se, po prvi put, CEKP_CG prezentuje na jednoj nacionalnoj konferenciji posvećenoj internet tehnologijama, što može biti od koristi u smislu dobijanja sugestija za korigovanje i poboljšanja rada na projektu, tokom predstojećih faza njegove nadgradnje. S toga se unaprijed zahvaljujemo na svim konstruktivnim primjedbama i sugestijama.

Pilot Web baza CEKP_CG, realizovana je u Access-u i ASP-u. Baza je organizovana kroz tri modula koja su na istom hijerarhijskom nivou, i to: modula za bibliotečku građu, modula za arhivsku građu i modula za muzejske artefakte. Broj dostupnih slogova je za sada jako skroman. Za očekivati je da će u dogledno vrijeme biti uvećan, budući da je materijal dostupan u odnosnim klasničnim listnim katalogima (i drugim formama) prilično obiman.

2. O PROJEKTU CEKP_CG

Projekat pod nazivom Centralni katalog pomorstva Crne Gore (CEKP_CG), u skladu s nacionalnim zakonskim propisima i preporukama Evropske zajednice o prezervaciji (očuvanju), valorizaciji i digitalnoj prezentaciji kulturnog nasleđa i naučnih informacija, omogućava da se na jednom mjestu, u obliku elektronskog (nerestriktivno on-line dostupnog) kataloga (bibliografske baze podataka), popiše i prezentira bibliotečki, arhivski i muzejski materijal pomorske provenijencije iz Boke Kotorske i čitave Crne Gore.

CEKP_CG kao elektronski katalog omogućuje: integralnu katalogizaciju i prezentaciju arhivalija, muzejalija i bibliotečkog materijala maritimne sadržine ili porijekla, kao pokretnog kulturnog spomeničnog blaga Crne Gore od izuzetnog značaja za kulturu, privredu, nauku i tradiciju Crne Gore. Posredstvom Interneta, nerestriktivno je dostupan posjetiocima, odnosno, korisnicima. Dodatno, omogućuje integrisanje CEKP_CG u slične baze podataka u Crnoj Gori i okruženju.

Ono što je od posebnog značaja, CEKP_CG sadrži Registar vlasnika pomorske građe, tj. bazu podataka o subjektima koji se bave pomorstvom i/ili posjeduju pisane i štampane građu o pomorstvu i/ili maritimne muzealije. Posebna prednost je, sa aspekta korišćenja pogodnosti Web-a, što se u Registru, u polju adrese, mogu naći odgovarajuće URL i/ili e-mail adrese, što obezbjeđuje brz i efikasan kontakt i informisanje.

Inicijator i nosilac projekta CEKP_CG je nevladina organizacija «Notar» - Centar za njegovanje i prezentaciju dokumentarnog nasljeđa Kotora, sa sjedištem u Kotoru. Zvanični podržavaoci projekta su: Fakultet za pomorstvo u Kotoru, Centralna narodna biblioteka Crne Gore, Univerzitetska biblioteka, te Udruženje bibliotekara Crne Gore. Tokom rada na projektu, javili su se brojni subjekti (kolektivi i pojedinci), koji su spremni da daju svoj doprinos. Posebnu zahvalnost zavređuje Centar informacionog sistema Univerziteta Crne Gore (CIS-u), koji je prepoznao značaj projekta i obezbijedio hosting CEKP_CG.

3. WEB PREZENTACIJA CEKP_CG

Web prezentacija CEKP_CG pruža informacije o tome što je CEKP_CG, što ima za cilj, odnosno, što se u njemu sve sadrži, ko su potencijalni korisnici, kako je organizovan, na koje se sve načine može pretraživati, koja su osnovna pravila etičkog korišćenja kataloga i sl.

U nastavku slijedi nekoliko ključnih odrednica CEKP_CG, datih Web prezentacijom (www.ckpcg.ac.me):

- CEKP_CG je kooperativni elektronski katalog za koji podatke obezbjeđuju svi vlasnici pomorske građe iz Crne Gore i dostupan je najširoj javnosti putem Interneta, besplatno;
- Cilj CEKP_CG je da omogući korisnicima jednostavno i brzo informisanje o tome da li i gdje na području Crne Gore mogu pronaći informacije o pomorstvu koje ih zanimaju;
- CEKP_CG je, kao baza bibliografskih podataka, logistička dopuna obrazovnom procesu u oblasti pomorskih nauka, prije svega na Fakultetu za pomorstvo u Kotoru i Srednjoj pomorskoj školi u Kotoru, a nezaobilazan je izvor u istraživanjima istorije bokeljskog i crnogorskog pomorstva, kao i pomorstva jadranske, odnosno, mediteranske regije;
- Najčešće kategorije potencijalnih korisnika CEKP_CG su studenti Fakulteta za pomorstvo u Kotoru, pomorci, istoričari pomorstva, privrednici, kao i svi oni kojima je pomorstvo predmet naučno-istraživačkog bavljenja, znatiželje i/ili ličnog zadovoljstva;
- CEKP_CG se u logičko-organizacionom smislu sastoji iz tri baze podataka: 1-Bibliotečki material, 2-Arhivski materijal i 3-Muzejski materijal. Radi lakšeg pretraživanja, prve dvije baze (bibliotečki i arhivski materijal) podijeljene su prema vrstama materijala, kako slijedi:

1-Bibliotečki materijal iz pomorstva

- monografske publikacije (knjige);
- serijske publikacije (časopisi, novine);

- članci i prilozi u serijskim i periodičnim publikacijama;
- neknjižni bibliotečki material;

2-Arhivski materijal iz pomorstva

- arhivska građa u arhivima, muzejima, bibliotekama i privatnim kolekcijama;
- arhivska građa u državnim i privatnim ustanovama, organima i organizacijama (registrature);

3-Muzejski materijal iz pomorstva (maritimne muzealije);

- Pretraživanje se odvija pojedinačno za svaku od pomenute tri baze (simultano pretraživanje nije moguće u ovoj fazi projekta). Korisnik se na samom početku opredjeljuje koju od tri pomenute baze će pretraživati, a zatim, u okviru konkretne baze (ukoliko se radi o bibliotečkom i arhivskom materijalu), koju vrstu materijala će pregledati. Traženje i pretraživanje omogućeni su po raznim parametrima: autoru, naslovu, ključnoj riječi (ključnim riječima), izdavaču, mjestu izdavanja, godini, lokaciji gdje je materijal smješten i sl. (zavisno od baze u kojoj se traži/pretražuje). Nakon zadatog upita, automatski se izlistavaju rezultati za onu bazu u kojoj se trenutno pretražuje (bibliotečki materijal, arhivski materijal ili muzejski materijal);
- Uvidom u pojedinačan zapis, korisnik primjećuje da on sadrži osnovne bibliografske podatke o jedinici koja ga interesuje (skraćeni bibliografski zapis), a saznaje i njegovu lokaciju (tj. u kojoj biblioteci, arhivu, muzeju ili ustanovi u Crnoj Gori se nalazi taj konkretan dokument);
- Podaci o institucijama čiji se maritimi materijal unosi u CEKP_CG (adresa, e-mail adresa, tel/fax, Web stranica, odgovorna i/ili kontakt osoba) mogu se naći u Registru vlasnika pomorske građe, što daje mogućnost kontakta i dogovora oko, eventualnog, fizičkog uvida u konkretan materijal;
- Korisnici su u obavezi da se prema informacijama dobijenim putem CEKP_CG odnose u skladu sa zakonskim propisima o pravu svojine (copyright, eng.) i drugim propisima kojima se regulišu načini i uslovi korišćenja elektronskih baza podataka i sprječava njihova eventualna zloupotreba. Nvladina organizacija «Notar» iz Kotora je vlasnik CEKP_CG, i u slučaju otkrivanja nepravilnosti i zloupotreba, ima pravo i obavezu da zaštiti autorska prava;
- Prilikom navođenja izvora iz ove baze podataka (citiranje sekundarnih izvora - bibliografskog zapisa iz CEKP_CG), na samom kraju zapisa, preporučuje se navođenje URL-a Web prezentacije CEKP_CG (www.cdknotar.org) i datuma preuzimanja reference;
- CEKP_CG je kooperativni elektronski katalog, kompatibilan sa sličnim bazama podataka i u tom pogledu, otvoren je za šire integracije, kao što su, između ostalog, INTERREG ADRIATICA, MED i sl. Trenutno, NVO CDK «Notar», kao nosilac projekta CEKP_CG, je zainteresovana za pronalaženje partnerske organizacije iz Hrvatske u programu Instruments for Pre-accession Assistance (IPA), Cross Border Croatia – Montenegro (CBCRO-MNE) - Prekogranična saradnja Hrvatske i Crne Gore (<http://www.cbccro-mne.org>);

- o Ukoliko korisnici naidu na poteškoće prilikom pretraživanja CEKP_CG, ili ukoliko žele da razjasne neku nedoumicu, postave pitanje, daju predlog ili iznesu stručno mišljenje vezano za unapređenje ove elektronske baze, mogu koristiti e-mail adrese: fzpbibl@t-com.me; ruzicad@ac.me ili regularnu poštansku adresu: Fakultet za pomorstvo u Kotoru - Pomorska biblioteka, Dobrota 36, 85330 Kotor, Crna Gora.

4. PRAVCI ZA DALJI RAD NA PROJEKTU CEKP_CG

S obzirom na složenost i obim stručnih poslova na prikupljanju i obradi materijala, na nesređeno stanje pomorske građe, na brojnost subjekata koji su uključeni u projekat, kao i njihovu prostornu dislociranost, predviđeno je da se projekat realizuje kroz četiri faze.

Prva faza podrazumijeva:

- o kontinuirano animiranje za rad na projektu i uključivanje pomorskih subjekata, vlasnika pomorske građe, kao i stručne javnosti i medija;
- o izradu Metodološkog uputstva za odabir, stručnu obradu, zaštitu i prezentaciju bibliotečkog, arhivskog i muzejskog materijala o pomorstvu;
- o dopunu Registra vlasnika pomorske građe;
- o izbor saradnika za rad na terenu;
- o uvid u stanje građe na terenu (prioritet je područje Kotora i Boke kotorske);
- o izradu softvera za katalogizaciju i pretraživanje;
- o rješavanje pitanja copyright-a sa vlasnicima pomorske građe (uz eventualno organizovanje okruglog stola na ovu temu);
- o izradu Izvještaja o stanju pomorske građe;
- o multimedijalnu izložbu pomorskog materijala;
- o poslove PR-a.

Druga faza podrazumijeva:

- o izradu projekta zaštite najugroženijeg pomorskog materijala;
- o izradu pilot baze podataka CEKP_CG i njeno testiranje;
- o obuku saradnika za katalogiziranje i traženje/pretraživanje;
- o pripremu i izradu preliminarne verzije Web prezentacije CEKP_CG;
- o poslove PR-a.

Treća faza podrazumijeva:

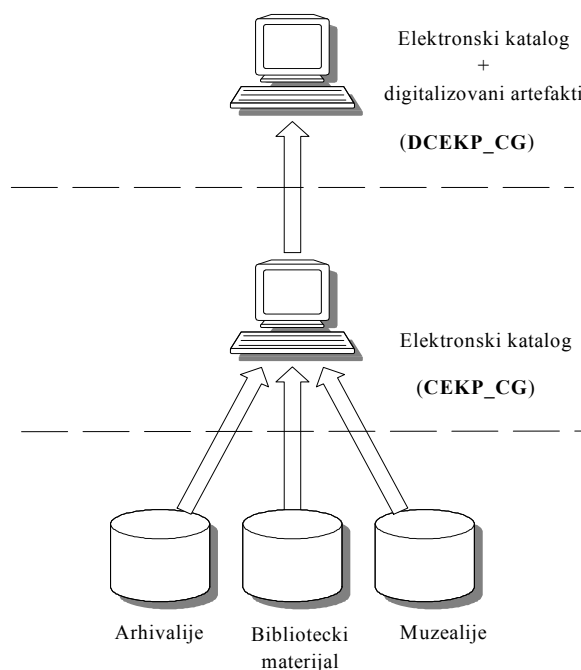
- o finalnu obradu podataka, *usability* testiranje i definisanje korisničkog interfejsa CEKP_CG;
- o izradu tradicionalnog (lispnog) i elektronskog kataloga;
- o izradu plana i programa digitalizacije najvrednijeg pomorskog arhivskog i bibliotečkog materijala;
- o pripremu CD-a o CEKP_CG;
- o poslove PR-a.

Četvrta faza podrazumijeva:

- o finalizaciju CEKP_CG i izradu digitalizovanog CEKP_CG (DCEKP_CG);
- o javnu prezentaciju finalne verzije CEKP_CG i DCEKP_CG;
- o javnu prezentaciju nove (upotpunjene) verzije CD-a o CEKP_CG i o DCEKP_CG;
- o poslove PR-a.

U radu na projektu CEKP_CG, se zapravo stiglo do treće faze. Ostalo je još puno posla, u smislu potrebe unosa velikog broja novih slogova u sve tri baze, prilagođavanja korisničkog interfejsa u smislu da vizualno bude prijemčiviji i više *user-friendly* orjentisan, zatim, trebalo bi omogućiti postavljanje *upita nad upitima*, tj. omogućiti postavljanje upita nad svima trima bazama CEKP-CG simultano, te omogućiti on-line ažuriranje elektronskog kataloga na različitim nivoima hijerarhije administriranja baze i sl.

Ono što je zapravo najvažnije, CEKP_CG bi trebao da posluži kao osnova, koju bi nadgrađivala baza sa kompletno ili fragmentarno digitalizovanim dokumentima katalogiziranim u CEKP_CG, tj. izrada digitalnog CEKP_CG ili DCEKP_CG skraćeno (slika 1). Prije svega trebalo bi digitalizovati u cjelosti reprezentativne dokumente, koji su od ključnog značaja za sticanje makar okvirnog uvida u bogastvo kulturno-istorijskog nasleđa kojim Crna Gora raspolaže u oblasti pomorstva, a koje baštini od davnina.



Slika 1. Shematski prikaz tranzicije CEKP_CG u DCEKP_CG

Među najznačajnije pomorske artefakte, koje bi neizostavno trebalo digitalizovati, mogu se svrstati: najstariji ugovori o gradnji ili kupovini brodova, o prevozu robe, dokumenti iz oblasti pomorskog osiguranja, Statut srednjovjekovne bratovštine mornara, dokumenta vezana za najstariju formaciju pomoraca - Bokeljsku mornaricu,

Balovičev pomorski priručnik sa petojezičnim pomorskim rječnikom, geografske, hidrografske i druge karte Boke Kotorske, najstarije brodske dnevničke i druge brodske knjige i sl.

Neobično je važna vizualizacija ovih najstarijih dokumenata (artefakata) iz domena pomorstva, na ovim našim litoralnim prostorima, budući da bi se na taj način najprije skrenula pažnja korisnika na nesumljivu vrijednost koju oni imaju. Takođe, pobudio bi se interes (znatiželja) korisnika za dalja istraživanja u predmetnom domenu, po raznim osnovama. Sve to urodilo bi popularizacijom i valorizacijom odnosnih artefakata, a posleđično i cjelokupnog kulturnog blaga kojim Crna Gora raspolaže. U pravcu isticanja potrebe za digitalizacijom, odnosno, vizualizacijom značajnih dokumenata iz oblasti pomorstva, u najširem smislu, slijede dva interesantna prikaza (slike 2;3).



Slika 2. Ugovor o izgradnji broda, iz XIV vijeka (1336), sklopljen između izvjesnih brodograditelja Milija i Mikoja iz Perasta i naručioca Bisto de Permuti-ja iz Kotora



Slika 3. Statut Bratovštine mornara iz XV vijeka (1463), pisan na italijanskom jeziku sa mletačkim dijalektom

ZAKLJUČAK

U radu je dat pregled dosadašnjeg rada na projektu Centralni elektronski katalog pomorstva Crne Gore - CEKP_CG, kao i smjernice za predstojeći rad na nadgradnji istog. Ukazano je na potrebu finansijske potpore projekta. Takođe, istaknute su potrebe: upotpunjavanja novim slogovima pilot verzije ovog Web kataloga, njegova bolja vizualne prezentacija, poboljšanje mehanizama traženja i/ili pretraživanja i formi prikaza rezultata, obezbjeđivanje on-line pristupa bazi u cilju njenog ažuriranja na različitim nivoima upravljanja istom, te, povezivanje baze sa sličnim bazama u regionu.

Na poslijetku, akcentirana je potreba nadgradnje kataloga fragmentarno ili u cjelosti, digitalizovanim najstarijim i najvrednijim maritimnim artefaktima sa područja današnje crnogorske litoralne zone i njenog zaleđa. Sve ovo je od neospornog značaja za adekvatno čuvanje i prezentovanje našeg kulturnog nasleđa, ne samo u naučno-istraživačke, već i u privredne svrhe, a posebno one vezane za pomorstvo i kulturni turizam.

LITERATURA

- [1] S. Bauk, R. Danilović, «Centralni elektronski i bibliotečki katalog – novi obrazovni alat Pomorske biblioteke u Kotoru», *46th IEEE International Symposium Electronics in Marine - ELMAR*, Zadar, Croatia, 2004. (pp 525-530)
- [2] S. Bauk, R. Danilović, N. Kovač, S. Pejović, «The Central Electronic Maritime Catalogue – A Segment of Montenegrin Cultural Inheritance Digitalization», *Review of the National Center for Digitalization*, (ISSN: 1820-0109), Issue 14, 2009. (pp 43-52)
- [3] S. Bauk, S. Pejović, R. Danilović, «Digitalizacija istorijskog nasljeđa Crne Gore kroz Centralni elektronski i bibliotečki katalog», *Techno-ocean '04 MTS/IEEE Conference*, Kobe, Japan, 2004. (pp 941-947)
- [4] R. Danilović, «Centralni katalog pomorstva Crne Gore u Pomorskoj biblioteci», *Zbornik Fakulteta za pomorstvo u Kotoru*, God. XIX, br. 20, 2003. (pp477-484)
- [5] R. Danilović, «Centralni katalog pomorstva Crne Gore (CKPCG) – informaciona baza u Pomorskoj biblioteci u Kotoru», *Pomorstvo*, Kotor, God. XXI, br. 22
- [6] S. Pejović, S. Bauk, R. Danilović, «Arhivska i bibliotečka građa u zajedničkoj bazi podataka na primjeru projekta Centralni elektronski katalog pomorstva Crne Gore», *Tehnički in vsebinski problemi klasičnega in elektronskega arhiviranja*, Radenci, Slovenia, 2005. (pp 512-526)

RAZVOJ I IMPLEMENTACIJA E-GOVERNMENT-A U OPŠTINI PLJEVLJA DEVELOPMENT AND IMPLEMENTATION OF E-GOVERNMENT IN THE MUNICIPALITY OF PLJEVLJA

Sanja Kotlaja, Opština Pljevlja

Sadržaj – U radu će biti prezentovana potreba uvođenja E-uprave u Opštini Pljevlja u cilju povećanja efikasnosti, transparentnosti, lakoće pristupa i mogućnosti odgovora na zahtjeve građana, intenzivnim i strateškim korišćenjem informaciono - komunikacionih tehnologija u unutrašnjem menadžmentu javnog sektora.

Abstrakt – In this work I will present the necessity of introducing E-government in the Municipality of Pljevlja in order to increase the efficiency, transparency, easy access and the possibility to answer the citizens' needs, by intensive and strategic usage of informational-communicational technologies in the inward management of public sector.

1. UVOD

Uvođenje informacionog sistema i elektronske uprave predstavlja proces čiji su ciljevi: manje kontakta sa individualnim osobama, redukcija papirologije, ubrzanje javne administracije, tačni i ažurni podaci u svim odjeljenjima javne uprave, brži, efikasniji i građanima transparentniji postupci, sprečavanje zloupotrebe (ličnih) podataka, jeftinije administrativne procedure i postupci, sigurni i stabilni sistemi, tehnologija i javno dostupni servisi, znatna ušteda novca i vremena građana i zadovoljni korisnici (građani, zaposleni u javnoj upravi).

Opština Pljevlja je sa ciljem pružanja što efikasnijih usluga krajnjim korisnicima odnosno građanima radi zadovoljenja njihovih potreba, otvorila Građanski biro u novembru 2004. godine. Posle pet godine rada, može se zaključiti da je Građanski biro zadovoljio osnovnu svrhu otvaranja, jer građani mnoge usluge mogu da dobiju na jednom mjestu. Ali to za Opštinu Pljevlja predstavlja samo jedan mali početak u razvoju i primjeni info-komunikacionih tehnologija, i tek joj predstoji stvaranje informacionog sistema, lokalnog intraneta i e-uprave.

2. ANALIZA FUNKCIONISANJA ORGANA LOKALNE UPRAVE

Usluge građanima Opština Pljevlja pruža u okviru 6 Sekretarijata, 2 Direkcije i 6 Stručnih službi. Karakteristike rada administracije na svim nivoima su sljedeći: administracija ne radi po evropskim standardima, loša preglednost rada i poslovanja, nedostaje elektronsko komuniciranje kako po horizontalnim tako i po vertikalnim nivoima, postoji plan informatizacije lokalne uprave u skorijoj budućnosti, dok postojeći sistemi djeluju izdvojeno, nepovezano i ne mogu obezbijediti građanima potrebne informacije, nedovoljno korišćenje savremenih načina komuniciranja (video konferencije, e-mail, SMS, ...), preko web stranica uprave, građani nemaju na raspolaganju mnogo usluga: osim informacija koje se redovno ažuriraju, tu su u rijetkim slučajevima i obrasci koji se mogu lokalno odštampati, itd. Suštinski, koncept eUprave nije u funkciji.

3. PODRUČJA I PRAVCI DJELOVANJA

U razvoju Opštine Pljevlja izuzetno je važno zajedničko stvaranje prepoznatljive VIZIJE moderne uprave na osnovu savremenih trendova, ali i naših vlastitih istorijskih, višenacionalnih i kulturnih specifičnosti. U javnoj upravi neophodno je stvoriti dugoročne temelje jedne nove kulture rada, koja će snažno podržati i prihvatiti iskustva kreativnih i efikasnih uprava razvijenog svijeta. Neophodno je izvršiti i sprovesti reformu i reorganizaciju strukture sistema javne uprave sa ciljem povećanja nivoa njene efikasnosti i osavremenjavanja. Na takvim osnovama uspostavljena, profesionalna i odgovorna javna uprava bi bila prevashodno servis građana i poslovnih sistema, i u potpunosti okrenuta njihovim legitimnim interesima i potrebama.

Reforma javne uprave ima za cilj da dovede do suštinskih promjena načina i metoda rada uprave, uz mijenjanje ponašanja i sistema vrijednosti prema zahtjevima vremena.

3.1 Tehnološko razvojne osnove

Informaciono - komunikacione tehnologije čine podlogu na kojoj se bazira razvoj savremene uprave kao i društva u cjelini. Pravci djelovanja treba da obuhvate organizacione, razvojne, kao i aspekte standardizacije i osiguranja kvaliteta. U organizacionom smislu napravljena je služba Informacioni centar koja je zadužena za tehnologiju, razvoj, standardizaciju i kvalitet.

Razvoj. Razvoj svake aplikacije u eUpravi treba da obezbijedi zahtijevani kvalitet proizvoda i kompatibilnost sa drugim/eksternim sistemima, prije svega iz domena eUprave, ali takođe mora da bude i efikasan, maksimalno racionalan i ekonomičan.

Standardizacija. Standardima se štite investicije, i omogućuje kompatibilnost između pojedinih komponenti rješenja, bez obzira na proizvođača, zbog čega se rješenja moraju zasnivati na otvorenim standardima.

Osiguranje kvaliteta. Osigurati kvalitet usluga znači planirati, projektovati i nuditi takve usluge, koje na najekonomičniji način i dugoročno zadovoljavaju potrebe i zahtjeve korisnika. Procesi treba da budu planirani, propisani i dokumentovani, a zatim implementirani na osnovu tih dokumenata.

3.2 Međuoperativnost organizacionih jedinica uprave

Da bi se realizovali kvalitetni servisi javne uprave, usmjereni na zadovoljavanje potreba građana i poslovnih sistema, neophodno je obezbijediti nesmetan tok informacija kroz javnu upravu. Za realizaciju tog zahtjeva, potrebno je definisati tehničke specifikacije i rješenja za ostvarivanje jedinstvenog informacionog sistema. Pored toga, potrebno je obezbijediti uniforman način komunikacije između aplikacija eUprave i eksternih, prije svega poslovnih sistema u okviru Crne Gore.

Definisanje tehničkog okvira za razmjenu podataka, i implementacija usvojenih specifikacija u aplikacijama bi omogućila efikasnu razmjenu podataka između bilo kojih aplikacija koje podržavaju date specifikacije direktno, i bez potrebe bilo kakvog specifičnog prilagođavanja aplikacija. Time bi se postigao cilj – nesmetan protok informacija kroz cijeli sistem eUprave, što bi efektivno značajno povećalo vrijednost instaliranih aplikacija. Ovakvo rješenje mora takođe da vodi računa o sigurnosti podataka prilikom razmjene, tj. podatke ne smije da dobije neko ko nema ovlaštenja za njih, niti smije postojati mogućnost neprimijećene promjene podataka za vrijeme njihovog transporta kroz sistem.

Cilj aktivnosti u ovom segmentu je da se obezbijedi nesmetan tok informacija kroz javnu upravu, kao i uniforman i standardizovan način komunikacije između eUprave i eksternih sistema: građana, poslovnih subjekata i različitih organizacija.

Kroz odgovarajuće projekte, treba da se definiše i verifikuje koncept i standard za međuoperativnost između organizacionih jedinica javne uprave (kao i međuoperativnost sa eksternim sistemima), kao i uloge i odgovornosti za trajni proces usavršavanja i implementacije usvojenog koncepta.

3.3 Centralni registri

U procesu izgradnje eUprave, jedan od prioriteta je harmonizacija i povezivanje i/ili integracija baza podataka, u cilju osiguranja njihovog integriteta i efikasnog korišćenja od strane svih subjekata koji imaju odgovarajuća ovlaštenja.

Registri koji su u nadležnosti različitih nivoa uprave (državni, lokalni) se mogu podijeliti na: civilne, bezbjednosne i privredne registre. Imajući u vidu različite nivoe uprave (državni, lokalni) formiraće se registri na određenim nivoima uprave. Činjenica da su podaci koji su generisani i raspoloživi u nekoj organizacionoj jedinici uprave potrebni i organizacionim jedinicama na drugim, horizontalnim i vertikalnim nivoima, nameće više pitanja kao npr.: tehnički način akvizicije, smještanja, razmjene i korišćenja tih podataka, nadležnosti i vlasništva nad podacima, odgovornosti za tačnost, raspoloživost i sigurnost, finansiranje itd. S jedne strane, obrada podataka na nižim nivoima jasnije razgraničava odgovornost i nadležnosti i daje mogućnost inicijative “nižim” nivoima, dok s druge strane otežava raspoloživost podataka (pogotovo kada problem međuoperativnosti nije adekvatno riješen), tako da integritet i kompletnost podataka zavisi od inicijative, volje i mogućnosti nižih nivoa. Zbog toga je potrebno da se prilikom odlučivanja o formiranju i načinu obrade ovih registara poštuju kriterijumi: funkcionalnosti, ekonomičnosti i zakonske nadležnosti.

Funkcionalnost podrazumijeva da budu zadovoljene sve potrebe u smislu vrste podataka, brzine odziva, raspoloživosti i slično, uz zadovoljenje tehničkih kriterijuma definisanih za oblast na koju se odnose.

Ekonomičnost podrazumijeva iznalaženje optimalnog balansa između tehničkih uslova i opcija, te raspoloživosti i cijene materijalnih sredstava i ljudskih potencijala u datom okviru zakonskih rješenja.

Zakonom propisane nadležnosti u suštini određuju i definisanje načina i nivoa uprave na kojem se vode registri.

Cilj je da se izvrši harmonizacija i povezivanje/integracija fundamentalnih baza podataka te izgradi i održava ažurnim sistem centralnih registara, u skladu sa zakonom, i da se obezbijedi njihovo korišćenje od strane subjekata iz javnog i privatnog sektora i građana na način propisan zakonom.

3.4 Sigurnost

eUprava mora obezbijediti okvir za siguran rad sa informacijama. Ovo podrazumijeva da se moraju preduzeti sve mjere zaštite u informacionom sistemu eUprave kako bi se minimizirala mogućnost gubitka informacija ili njihovo neovlašćeno mijenjanje/korišćenje, što bi u određenim slučajevima moglo da izazove nesagledive posljedice. Zbog toga je potrebno razviti rješenja i definisati postupke i mjere koje će obezbijediti informaciono - komunikacionu sigurnost, te stvoriti mehanizme za njihovu primjenu. Ovo je od ključnog značaja, jer se time stvaraju uslovi da svi subjekti koji koriste servise eUprave steknu povjerenje da su transakcije koje obavljaju sa upravom pouzdane, zaštićene i sa zahtjevanim stepenom povjerljivosti. No, pretjerano i neumjereno fokusiranje na sigurnost može s druge strane da rezultuje rješenjima koja su suviše rigidna, komplikovana i skupa i koja mogu da predstavljaju prepreku za razvoj i primjenu novih tehnologija. Dakle, neophodno je permanentno procjenjivati rizike s jedne strane, i sigurnost sistema, s druge strane i tražiti izbalansirano rješenje.

Paralelno, potrebno je raditi na razvoju i implementaciji jedinstvenog sistema autentifikacije i autorizacije za aplikacije u upravi, i to: Za transakcijski usmjerenu autentifikaciju preporučuje se sistem na bazi “liste passworda”; za upravljanje mrežnom infrastrukturom preporučuje se sistem sigurnosti na bazi “token-a”; za rješavanje problema digitalnog potpisa i tajnosti komunikacije između subjekata korišćenjem e-mailova preporučuje se kreiranje PKI bazirane na PGP (Pretty Good Privacy) gdje je svaki subjekat zadužen za kreiranje vlastitog para ključeva. Drugim riječima nakon što subjekat deponuje svoj javni ključ na javni server on ima opciju da taj ključ sertifikira kroz utvrđeni proces.

3.5 Specijalizovane funkcije po pojedinim organizacionim jedinicama uprave

Fundamentalni način organizovanja tj. vršenja administrativnih funkcija, odvija se u okviru najviših organa uprave tj. sekretarijata. Radna područja kroz koje se ove funkcije odvijaju su privreda, finansije, uređenje prostora, društvene djelatnosti, inspekcijski poslovi i opšta uprava. Sva radna područja uprave biće informatički podržana tj. automatizovaće se njene funkcije formiranjem informacionog sistema, za svaku oblast sa konkretnim aplikativnim rješenjima.

Aplikativna rješenja omogućit će i internim (organi lokalne samouprave) i eksternim (građani, privreda) korisnicima administrativnih usluga dobijanje kvalitetnih i brzih elektronskih usluga. Sva aplikativna rješenja uzimaju u obzir organizacionu vertikalnu (najviši - najniži nivo) svake funkcije u cilju dobijanja jednoobraznih usluga. Korišćenje elektronskih usluga-servisa od strane krajnjeg korisnika treba da bude lišeno potrebe za poznavanjem konkretnog nivoa uprave koji daje uslugu ili internih transakcija između organa lokalne samouprave koje su neophodne radi obezbjeđivanja usluge. Promjene pojedinih podataka vrše se na jednom mjestu upisom u odgovarajući registar, a prosljeđivati na sva ostala potrebna mjesta.

Cilj je da se informatizacijom svih (uključujući i specijalizovane) funkcija uprave, realizuje elektronicizovani "back office", koji će preko elektronskih servisa omogućiti: jednostavne, brze, kvalitetne i kompletne usluge krajnjim korisnicima. Kako bi specijalizovane usluge javne uprave bile elektronski podržane, neophodno je učestće svih nivoa uprave u pripremanju konkretnih akcija od lokalnog nivoa vlasti do privatnog i nevladinog sektora u realizaciji ovih aktivnosti.

Službenici u upravi takođe treba da steknu jasnu predstavu o koristi od uspostavljanja novih sistema, uz potrebu njihovog kako informatičkog opismenjavanja tako i njihovog motivisanja za aktivno učestće u stvaranju i korišćenju tih informatički podržanih sistema.

Procedura za dobijanje usluga pojedinih organizacionih jedinica uprave treba da budu maksimalno uprošćene i pojednostavljene, kako bi bile što lakše za korišćenje od strane krajnjeg korisnika. One moraju da budu fokusirane na (građane) krajnjeg korisnika a ne na upravu.

U sve automatizovane sisteme mora biti ugrađena transparentnost rada uprave. Pravila i zahtjevi neophodni za dobijanje nekog servisa moraju jasno biti navedeni kako bi se minimizirale potencijalne subjektivne akcije službenika. Sve aplikacije kroz koje se obavljaju e-servisi treba da omoguću krajnjem korisniku mogućnost uvida u trenutno stanje njegovog zahtjeva. Korišćenje dobre prakse je potrebno uvesti kao jedan od presudnih faktora u odlučivanju o konkretnim akcijama- kroz razmjenu iskustava između lokalnih i regionalnih zajednica (gradova, opština, regija, država). Pri definisanju prioriteta treba imati u obzir ekonomski aspekt – cijenu realizacije, društveni aspekt-korist za građane, itd.

Neophodno je uvesti efikasno upravljanje elektronskim zapisima. Današnje stanje je takvo da je većina arhiva papirna bez obzira na postojanje ili ne baza podataka. Za podršku svim funkcijama javne uprave potrebno je uvesti sisteme za upravljanje elektronskim zapisima i dokumentima, tako da se oni mogu održavati i da im se može pristupiti na način koji čuva autentičnost i integritet sačuvanih podataka, uz harmonizaciju elektronskih i papirnih sistema.

4. NAČIN IMPLEMENTACIJE

Osnovni akteri u procesima javne i buduće elektronske uprave, između kojih postoji snažna interakcija, su uprava (često u stranoj terminologiji Vlada/Government), građani (Citizen) i privredni subjekti (Business). Razvoj elektronske uprave je od interesa svakom od gore navedenih aktera. Ukratko, razvojem ovog sektora pospješuje se efikasnost rada

uprave, povećava kvalitet usluga uprave prema korisnicima, transparentnost rada i procesa i jača učestće građana u demokratskim procesima.

U kontekstu savremenih tehnologija uprava (Vlada/Government) svoju ulogu ostvaruje preko servisa prema građanima i privredi. Ovi servisi se realizuju na različitim nivoima uprave u zavisnosti od zakonskih rješenja koja su usvojena, odnosno od nivoa centralizacije i decentralizacije.

Da bi se realizovao koncept eServisa potrebno je pokrenuti aktivnosti u dva pravca: povećanje informatičke kulture građana i izgradnja informacione infrastrukture. Ove dvije aktivnosti su povezane jedna sa drugom, ali je u osnovi potrebno stvoriti uslove koji bi doveli do povećanja broja korisnika Interneta preko povećanja broja pristupnih tačaka i broja klijenata na globalnoj mreži. Paralelno, potrebno je raditi na obezbjeđenju infrastrukture i sigurnosnih mehanizama koji omogućavaju siguran pristup eServisima.

Cilj aktivnosti u ovom segmentu javne uprave je da se omogući građanima, poslovnim i drugim subjektima, da svoje obaveze ili zahtjeve prema upravi obavljaju na što efikasniji način, uz minimalne troškove i broj fizičkih kontakata sa organima uprave, i to putem više različitih elektronski baziranih kanala za isporuku servisa (web, mobilni telefon, digitalna televizija itd.)

U cilju obezbjeđenja jedinstvene ulazne tačke u eUpravu, za jednostavan i korisnički pogodan pristup informacijama i servisima eUprave preko interneta za građane, poslovne subjekte, zaposlene u samoj Upravi, strance – pojedince i organizacije, realizovaće se adaptivni Internet portal sa odgovarajućim subportalima. Pristup preko portala biće omogućen sa radnih mjesta, kućnih računara ili javnih informacionih kioska. Pored portala, zainteresovani subjekti će imati pristup željenim informacijama i servisima putem telefona, GSM/WAP-a, SMS-a, putem digitalne televizije ili direktnom komunikacijom sa administracijom eUprave.

Cilj aktivnosti je da se izgradi jedinstven portal ispred svih web site-ova javne uprave, putem kojeg bi se omogućilo dobijanje informacija od interesa po principu jedne destinacije (one stop shop), kao i sigurne, brze i pouzdane transakcije sa građanima i poslovnim sistemima. Organizacija portala treba da se bazira na potrebama korisnika, a ne na organizacionoj strukturi uprave, sa jedinstvenim sadržajem i izgledom, bez obzira kojem segmentu javne uprave se pristupa.

Pravci djelovanja će se odnositi na definisanje strukture portala, izgradnja javnih pristupnih tačaka, kao i načina i uslova pod kojima će subjekti pristupati resursima eUprave. Uslovi pristupa treba da se baziraju na pripadnosti korisnika određenoj grupi. Grupe korisnika su: zaposleni u Upravi, građani i poslovni subjekti. Pristup informacijama se može realizovati preko web-a ili putem, SMS, e-mail, digitalna televizija (u perspektivi).

Portal bi se sastojao od XML e-formi, identifikacijskog sistema i komunikacione veze prema infrastrukturi javne uprave.

Prije korišćenja portala eUprave korisnik bi trebao da obavi: registraciju - sastoji se od prijave nadležnom organu za kreiranje korisničkog naloga, privremene šifre kao i uloga koje će koristiti korisnik ili od autentifikacije putem PKI, u zavisnosti od razvijene infrastrukture; upis – proces upisa za servise kojima korisnik želi pristupiti; aktiviranja naloga – poslije realizacije digitalnog upisa.

Mora se sprovesti veliki broj sistemskih i podsticajnih mjera za razvoj informaciono – komunikacionih tehnologija te uskladiti zakonske i podsticajne mjere sa zemljama Evropske unije i razvijenim zemljama svijeta. Potrebna je promocija razvoja informacionog društva i uvođenje mehanizama za mjerenje i praćenje napretka u primjeni novih tehnologija. Potrebno je uspostaviti efikasne načine usvajanja međunarodnih standarda vezanih za informaciono - komunikacione tehnologije. Pošto je dinamika razvoja i uvođenje novih tehnologija povezana sa kvalitetnim kadrovima, treba podržati rad informatičke službe u Opštini Pljevlja koja promoviše i njeguje kvalitetan i profesionalan rad te propagirati stalno stručno usavršavanje. U čitavoj opštini treba podsticati stvaralaštvo, nagrađivati dobre inicijative, realizovati kvalitetne projekte koji mogu doprinijeti boljoj i bržoj usluzi. Svaku dobru inicijativu treba javno pohvaliti, a dobra rješenja permanentno promovisati. Da bi se ovaj zadatak kvalitetno uradio, neophodno će biti angažovati kao saradnike ljude iz drugih institucija i kompanija (fakulteti, Telekom, instituti i privatne firme), koji kroz svoj redovan posao proučavaju metodologije i standarde i rade na njihovoj primjeni u svojim (užim) oblastima djelatnosti. Posebno treba jačati saradnju sa fakultetima u Crnoj Gori koji školuju kadrove iz oblasti informaciono - komunikacionih tehnologija.

4.1 Zakonska regulativa

Ministarstvo za informaciono društvo Crne Gore je 2009. godine donijelo Strategiju razvoja Informacionog društva u Crnoj Gori od 2009. do 2013. godine. Nova Strategija u oblasti razvoja eGovernment-a ima za cilj definisanje prioriteta za dalju realizaciju i implementaciju rješenja za efikasno stvaranje i funkcionisanje elektronske uprave. U pogledu ostvarivanja servisa eGovernmenta, posebna pažnja će biti posvećena unaprjeđenju dostupnosti građana servisima, na lokalnom nivou upravljanja. Strategijom u ovoj oblasti do 2013. godine definisani su ciljevi za uspješan razvoj eGovernment-a.

Za razvoj eUprave neophodno je i postojanje Zakona o elektronskom potpisu. Zakon o elektronskom potpisu u Crnoj Gori je donešen 12.07.2005 i publikovan u Službenom listu ("Sl.list RCG", br: 55/03 i 31/05).

5. OČEKIVANI REZULTATI I MOGUĆI RIZICI

Rezultat aktivnosti na implementaciji koncepta eUprave treba da bude kretanje ka ostvarenju vizije i ciljeva moderne uprave. Dakle, krajnji rezultat treba da bude reformisana, efikasna i moderna uprava, zasnovana na znanju i primjeni informaciono - komunikacionih tehnologija. Takva uprava biće katalizator razvoja sveukupnog društva i njegove integracije u razvijenu Evropu.

Od transformisane i moderne uprave, očekuje se da postane prevashodno servis građana i poslovnih sistema, i da u potpunosti bude okrenuta njihovim legitimnim interesima i potrebama.

Dakle, razvojem i realizacijom eUprave, očekuje se ostvarenje ciljeva: poboljšanje kvaliteta, raspoloživosti i brzine rada i pružanja usluga javne uprave; povećanje efikasnosti razmjene informacija po vertikalnom i horizontalnim nivoima između korisnika i davaoca usluga;

poboljšanje demokratskog dijaloga s građanima osigurava njihovo šire i intenzivnije uključivanje u demokratske procese; poboljšanje transparentnosti rada javne uprave; stvaranja uslova za skladan društveno-ekonomski i prostorni razvoj kako opština, tako i države u cjelini; obezbjeđivanje zaštite zakonitosti i sigurnost građana i imovine kroz nadzor i brzu kompjutersku kontrolu; obezbjeđivanje brzog i jednostavnog pristupa do svih potrebnih informacija; korišćenje i promocija standarda, sa ciljem zaštite investicija i lakše integracije sa razvijenim svijetom.

Postoji više faktora od kojih bitno zavisi uspjeh procesa razvoja eUprave. Prije svega, uspjeh će zavisiti od istinskog opredjeljenja relevantnih subjekata koji odlučuju i donose odluke na pojedinim nivoima uprave i njihove spremnosti da maksimalno podrže ove procese. Drugi kritičan faktor je obezbjeđenje potrebnog broja i kvaliteta kadrova koji bi nosili i realizovali proces razvoja eUprave. Jedan od kritičnih faktora je i pitanje nadležnosti pojedinih subjekata uprave po pitanju informatizacije. Od velikog je značaja takođe stvaranje svijesti i širenje saznanja putem medijske promocije, o mogućnostima i prednostima informatizacije uprave kako kod zaposlenih u upravi, tako i kod korisnika usluga (poslovnih sistema i građana).

6. ZAKLJUČAK

U razvoju elektronske uprave najvažnije je shvatiti da se opština Pljevlja ne može oslanjati, osim u načelu, na iskustva drugih, jer se načini funkcionisanja lokalnih samouprava, bez obzira na istu zakonsku infrastrukturu koja u jednoj državi reguliše ovu oblast, bitno razlikuju, zavisno od veličine grada, načina organizacije i podjele zadataka po sekretarijatima.

Uostalom, najvažnije je krenuti u realizaciju naprednih servisa i zavisno od finansijskih mogućnosti realizovati jedan po jedan.

Opština Pljevlja mora, polazeći od sopstvenih specifičnosti krenuti u realizaciju projekta elektronske uprave. Nakon identifikacije korisnika informacionog sistema i njihovih potreba opština mora razviti strategiju za implementaciju najbolje prakse upravljanja u državnim organima. Najvažniji korak na tom putu jeste osnivanje Informacionog centra, što je opština Pljevlja već uradila, ali je potrebno dodatno kadrovsko osposobljavanje koje jedino može obezbijediti primjenu novih tehnologija u opštinskoj administraciji.

LITERATURA

[1] Sanja Kotlaja, *Razvoj i implementacija sistema eGovernment-a u Opštini Pljevlja*, Ekonomske fakultet Podgorica, 2008. godina

[2] Damir Baralić, *Savremena rešenja za efikasnu javnu upravu i zadovoljne građane*, ISDOD Zlatibor, 2006

[3] Lj. Marković: *Modernizacija lokalne uprave i samouprave*, Centar za promociju civilnog društva, Sarajevo, 2004

SIGURNOSNO TESTIRANJE – PROVJERA SIGURNOSTI SISTEMA NA DJELU SECURITY TESTING – CHECKING SECURITY IN PRACTICE.

Slobodan Pavićević, *Crnogorski Telekom*
Božo Krstajić, *Elektrotehnički fakultet Podgorica*

Sadržaj - *Zaštita informacionih resursa je proces koji se odvija na više nivoa. Primjena najsavremenih tehnologija je potrebna, kao i definicija i implementacija odgovarajućih sigurnosnih politika. Provjere, u kojoj mjeri su sigurnosne politike i tehnologije uspješne, rade iskusni timovi kroz proces koji se zove sigurnosno testiranje (penetration testing, security testing).*

Abstract - *Protection of information systems needs multi level approach. Implementation and usage of modern technologies is important as well as acceptance of security polices. Main checks and answers if technology and policy implementation works well, team of experts will conduit through the process called security or penetration testing.*

1. UVOD

Sigurnosno testiranje IT sistema je potreba za sve kompanije. Svaki sistem sigurnosti mora biti sistematično i sveobuhvatno postavljen. Isto tako i testiranje sigurnosti i verifikacija zaštite IT sistema mora biti metodična i dobro pripremljena. Očekuje se da se provjere sigurnosti IT sistema obavljaju redovno i kontinuirano, više puta godišnje, u zavisnosti od važnosti sistema i potreba. Na koji način se sprovodi, koji tipovi testiranja postoje, ko su učesnici u tom procesu, plan i preduslovi potrebni za njegovo izvođenje, koje preventivne i aktivne mjere treba primjenjivati u cilju zaštite sistema, koje koristi se očekuju i konačno, što je glavna svrha testiranja, teme su koje su obrađene u ovom radu.

2. SIGURNOSNO TESTIRANJE

Posjedovanje najsavremenijih tehnologija zaštite, pravilno kreirane politike zaštite i dobro osmišljen i implementiran koncept zaštite jesu neophodni ali nijesu garant da je sistem u potpunosti zaštićen. Informacioni sistemi su složeni i mnogobrojni, i svaki dio tog sistema u svom domenu može predstavljati potencijalnu opasnost. Poznat je stav da je jedan sistem zaštićen onoliko koliko je zaštićena i njegova najslabija karika. Ovo još više što su alati i tehnologije koje koriste „hackeri“ u velikoj mjeri usavršene i lako dostupne na Internet-u. Sa druge strane eksperti u ovom polju, sve više stimulirani finansijskom dobiti, predstavljaju veoma respektabilan faktor opasnosti za korporacijske resurse.

U kojoj mjeri je jedan čitav ili dio informacionog sistema zaštićen najbolji odgovor može dati sigurnosno testiranje[1,2]:

Sigurnosno testiranje se može podijeliti u tri kategorije tehnika[2]: **Review techniques** - tehnika pregleda dokumentacije, logova, politika, procedura, konfiguracija provjera integriteta; **Target Identification and Analysis Techniques** - tehnike kojim se identifikuju sistemi, portovi, servisi odnosno ranjivosti sistema (port skeneri i skeneri ranjivosti); **Target Vulnerability Validation Techniques** - tehnike kojim se vrši eksploatacija otkrivenih ranjivosti odnosno pokazuje na djelu mogućnost kompromitacije sistema (penetracioni testovi, social engineering).

Sa tačke gledišta odakle se sigurnosno testiranje vrši, razlikujemo **externo**, koje se izvodi izvan granice lokalne mreže, obično sa Interneta, daje odgovor koliko je mreža osetljiva na externe napadače i **interno**, koje se izvodi unutar granica lokalne mreže, daje odgovor koliko povjerljivi napadač može ugroziti sigurnost mreže.

Takodje, sigurnosno testiranje se može podijeliti na **Overt (white hat testing)** – kada IT osoblje učestvuje u samom procesu testiranja, što doprinosi njihovoj edukaciji i svijesti o potencijalnim slabostima i mjerama za zaštitu; i **Covert (black hat testing)** – testiranje se izvodi bez znanja IT osoblja, i ima za cilj da se utvrde sigurnosni sistemi ciljne kompanije i njihova osposobljenost na sve vrste napada.

Od svih vrsta testiranja posebno su važna, kompleksna i skupa penetraciona testiranja, koja objedinjavaju sve vrste testova. Penetraciono testiranje se može definisati kao aktivnost povjerljivog i plaćenog tima experata koja ima za cilj da kompromituje kompjuterske resurse kompanije, radi provjere i procjene nivoa sigurnosne zaštite. Penetracioni tester se služi vještinama i znanjima koje bi i sam „hacker“ primijenio prilikom napada na mrežu, računar, aplikaciju, web server, bazu podataka itd. Potreba za ovom vrstom testiranja se može opravdati u postojanju sumnje da mreža nije dovoljno zaštićena kada postoji toliko mnogo prijatnji, koje su najviše vezane za: širenje virusa i trojanaca, sigurnost bezžičnih komunikacija, kompleksnost mreža, stalno update-ovanje SW, laka dostupnost SW za napad, priroda „open source“ programa, velika zavisnost i okrenutost poslovanja kompanije prema Internet-u, prisutnost mobilnih korisnika koji obavljaju poslove udaljeno od kuće ili van nje, obaveznost poštovanje finansijskih i sigurnosnih standarda (npr: SOX) itd.

Penetracioni testovi se mogu podijeliti na tri tipa: **Black-Box** – testeri namaju ili posjeduju minimalan skup poznatih informacija o mreži i resursima koje treba skenirati. Ovim se provjerava spremnost odbrane od nepredvidljivih uticaja i napada.

White-Box – testeri imaju potpune podatke o mreži koju trebaju skenirati. Mogu imati podatke tipa dijagrama mreže, listu operativnih sistema i aplikacija. Ovo je najprecizniji tip testiranja i predstavlja najgori mogući scenario gdje napadač u potpunosti poznaje mrežu.

Grey-box ili **Crystal-Box** – tester simulira lokalnog korisnika. Testeru se daje lokalni nalog i obezbjeđuje

standardni pristup u mreži. Ovim se provjeravaju prijetnje koje mogu doći od zaposlenih u kompaniji.

Penetracioni tester u suštini otkriva, testira ranjivosti i eksploataciju istih. Ranjivost (vulnerability) se definiše kao slabost, dizajn ili implementaciona greška koja se može iskoristiti da se prekrše sigurnosne polise. Prijetnja je potencijalna mogućnost narušavanja sigurnosti koja može izazvati štete kao što su otkrivanje osjetljivih informacija, modifikacije podataka, uništenje podataka ili onemogućavanje servisa. Cilj dobre zaštite sistema je da se obezbijedi zaštita resursa od prijetnji, koje se mogu odnositi na povjerljivost (rizik da neovlašćeno lice ima pristup podacima koje su označene kao povjerljive), integritet (rizik da neovlašćeno lice promijeni podatke) i dostupnost (rizik da servis bude nedostupan korisnicima) podataka (C.I.A: Confidentiality, Integrity, Availability). Za neki sistem ćemo reći da je siguran ako vjerovatnoća da se desi nedetektovana krađa ili promjena podataka ili neka nedozvoljena radnja, na prihvatljivom nivou. Taj nivo se definiše kroz proces „cost-risk“ analize i prihvatljiv je kad je trošak zaštite podataka poredljiv sa rizikom gubljenja ili kompromitovanja podataka. Cilj penetracionog testiranja je da ukaže na ovaj rizik i umanj ga na nivo koji je prihvatljiv za upravljačke funkcije u kompaniji. Ovaj rizik se nikada ne može svesti na nulu.

3. PLANIRANJE I IZVOĐENJE TESTIRANJA

Planiranje i izvođenje testiranja podrazumjeva nekoliko važnih koraka. Prvi korak je izbor izvođača testiranja. U izboru izvođača je bitno razmotriti sledeće činjenice: reference kompanije, stručnost i sertifikiranost izvođača testiranja, pouzdanost i eventualnu krimogenu istoriju izvođača, upoznatost izvođača radova sa specifičnim standardima koji se žele zadovoljiti, koju vrstu, obim podataka i izvještaja se prilaže, mogućnost izvođača da u predviđenom roku obavi testiranje, period aktivnog bavljenja ovom vrstom posla, da li je to jedina djelatnost kompanije, da li i koje posebne metodologije rada i alate koriste u procesu testiranja itd.

Prije nego samo testiranje započne, potrebno je napraviti plan testiranja i njegov opseg: u kojem vremenu će se testiranje obaviti, da li će biti dozvoljeno DoS testiranje, koji tip testiranja će se raditi, da li će administratori sistema biti upoznati sa testom, da li je dozvoljeno brisanje logova, da li će se dozvoliti social engineering, koji serveri i aplikacije će biti uključeni u proces testiranja itd.

Budući da je aktivnost penetracionog testiranja takve prirode da bi se smatrala ilegalnom kada ne bi bila zahtijevana, od suštinske važnosti i za kompaniju koja izvodi testove kao i za naručioca testiranja, da se zadovolje legalne, pravne i etičke norme. To znači da se ugovorom, pravnim dokumentom preciziraju detalji o vrsti i obimu testiranja, informacijama koje se dostavljaju, načinu isporuke i čuvanja podataka itd

Sam čin izvođenja testiranja, nakon završenih pripremnih radnji, sastoji se od nekoliko faza:

Reconnaissance (izviđanje) – inicijalno izviđanje i dobijanje podataka o ciljnoj mreži.

Enumeration/Scanning (sakupljanje) – proces ispitivanja i skeniranja aktivnih sistema u cilju dobijanja potrebnih informacija (korisnički nalozi, dijeljeni folderi, specifične aplikacije, otvoreni portovi, ranjivosti).

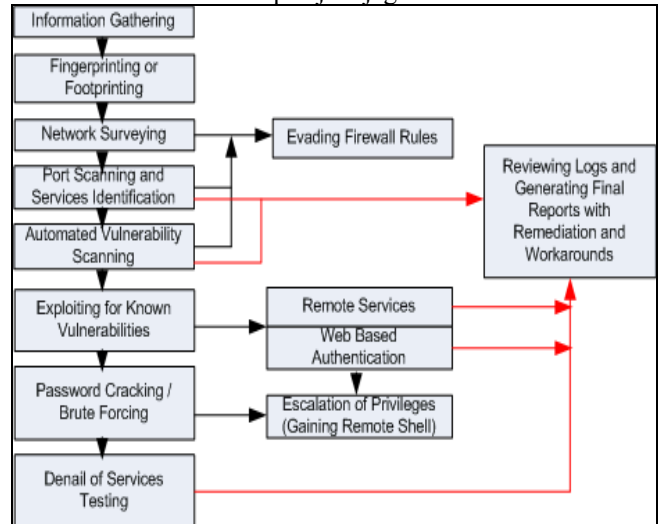
Gaining access – proces ulaska u sistem je najzujbudljiviji dio kada se koriste podaci dobijeni u prethodna dva koraka, u cilju ulaska u sistem.

Maintaining access – stvaranje pretpostavki za buduću eksploataciju sistema, na način da se instaliraju backdoor aplikacije, koje omogućavaju pen testeru ponovno vraćanje na sistem i bolju eksploataciju sistema. Ovim se ispituju i u kojoj mjeri su dobri mehanizmi detekcije backdoor aplikacija od strane vlasnika sistema.

Covering tracks – brisanje tragova o upadu u sistem.

Writing reports – prezentovanje i analiza rezultata testiranja u podobnoj formi, sa jasnim otkrićima i preporukama.

Redosled aktivnosti opisuje dijagram na Slici 1.



Slika 1. Dijagram koraka testiranja.

Očekuje se da se čitav proces testiranja obavlja u skladu sa priznatom metodologijom [3,4] ispitivanja sigurnosti sistema. Opšte priznata međunarodna metodologija je OSSTMM (Open Source Security Testing Methodology Manuel). Ona daje smjernice i uputstva za ispitivanje nivoa sigurnosti sledećih oblasti: zaštita informacija, zaštita procesa, sigurnost Internet tehnologija, zaštita komunikacija, sigurnost bezžičnih komunikacija i fizička sigurnost. Nakon obavljenog testiranja, pravi se finalni izvještaj.

4. VRSTE TESTOVA

U zavisnosti od dogovorenog obima, pen tester mogu izvesti jedan ili više vrsta testova odnosno pokušaja kompromitovanja kompanijskih resursa. Testovi koji se rade se mogu svrstati u nekoliko kategorija[1]:

Social Engineering. Ova vrsta testa je usmjerena na najslabiju kariku u lancu sigurnosti sistema – ljudski faktor. Social inženjeri koriste razne metode prevare, manipulacije, nagovaranja, uticaja kako bi postigli svoj cilj, a to je kompromitacija sistema. *Najbolji metod zaštite od ovog vida zloupotrebe je edukacija i trening.*

Footprinting (Izvođenje izviđanja i sakupljanja podataka). Cilj je otkriti sledeće informacije o ciljnim sistemima: ip adrese uređaja, otvorenost TCP i UDP portova kao i tipove i verzije operativnih sistema (*fingerprint*). Razlikuje se pasivno i aktivno istraživanje. *Najbolji metod zaštite od ove vrste skeniranja su Intrusion Host/Network Detection/Prevention sistemi, koji imaju mehanizme otkrivanja i sprečavanja ovih aktivnosti.*

Session Hijacking. Ovo je napad kojim se pokušava preuzeti već postojeća aktivna sesija (tcp) između dva hosta. Karakteriše ga osobina da je kroz preuzetu aktivnu sesiju preuzet i nivo autentifikacije koji je postignut. Razlikuju se aktivni tip, preuzima se u potpunosti aktivna sesija i pasivni, kopira se saobraćaj koji prolazi kroz sesiju. *Osnovne tehnike zaštite od ove vrste napada su korišćenje enkripcijski zaštićenih protokola npr ssh umjesto telnet-a, https umjesto http-a, VPN-a umjesto običnog udaljenog pristupa itd. IDS/IPS sistemi su sposobni da prepoznaju ove vrste napada.*

Web server Attacks. Danas, najčešći, možda i najlakši oblik kompromitacije sistema. Velika raznolikost web servera, programskih jezika i tehnologija koje se koriste (IIS, Apache, HTML, XML, PHP, ASP, Java, CGI, JavaScript, Jscript, VBScript, Perl), kao i sama arhitektura websajta i servisa doprinose većoj mogućnosti za pravljenje odnosno testiranje ovog napada. Razlikuju se dva tipa napada: napad na sami server (eksploatacija ranjivosti samog webserver u cilju postavljanje sopstvenog fajla, koda na serveru ili obaranja servera) i napad dobijanja nedozvoljene autorizacije na samom serveru (Brute Force password napadi, Web spoofing (phisihing)..: *Obrana za napade ovog tipa se radi na više nivoa – Obezbeđivanje operativnog sistema i web aplikacija: primjena sigurnosnih zahtjeva i preporuka: – Primjena sigurnog web dizajna: validacija inputa za web stranice, validacija inputa na sve podatke koje dolaze od klijenta i enkriptovanje cookies-a; – Primjena sigurne mrežne arhitekture: webserveri zbog svoje specifične funkcije moraju biti postavljeni u posebnim DMZ zonama u kojim se obezbeđuje samo potreban saobraćaj; – Primjena web aplikativnih FW: novi tip aplikacija koje registruju sve vrste napada na web aplikacije i uspješno ih sprečavaju u realnom vremenu.*

Database Attacks. Baze podataka su svuda prisutne i mogu sadržati razne tipove podataka, od običnih do strogo povjerljivih. Napadi na baze podataka su motivisani: krađom podataka, manipulacijom podataka, DoS servisa kompanije, dobijanja prolaza do drugih sistema. Glavne opasnosti za napade na baze podataka su *SQL Injection*, zloupotrebe *System store* procedura (npr xp_cmdshell), Connection Strings i Brute Force Attacks. *Dobra praksa zaštite: primjena najnovih patch-eva i update-a, dobra fizička zaštita resursa, striktna kontrola portova koji se koriste, uklanjanje nepotrebnih feature-a, fizička razdvojenost web i database servera, eliminacija svih nepotrebnih stored procedura, network biblioteka i demo bazapodataka, upotreba SSL komunikacije između klijenta i servera itd*

Password Cracking. Pristup resursima se obezbeđuje korišćenjem odgovarajućih naloga i lozinki. Ove vrste napada odnosno testova se rade uz korišćenje posebnih programskih alata koji primjenjuju: *dictionary attacks* – bazirani na upotrebu riječi iz rečnika odnosno fajla pri pogađanju lozinki, *brute force attacks* – bazirani na svim mogućim kombinacijama karaktera u pogađanju lozinke, *hybrid attacks* – bazirani na kombinaciji prethodna dva. U izvjesnim slučajevima se primjenjuju i „rainbow” tabele koje sadrže hash-ove lozinki koje su već otkrivene te se one porede sa onim na sistemima. *Zaštita se sprovodi kroz sledeće mjere: primjena kompleksnih politika za pravljenje lozinki, fizička bezbjednost resursa i edukacija zaposlenih o značaju i obaveznosti čuvanja lozinki na siguran način. Ipak, najbolja*

mjera zaštite je primjena sigurnih metoda autentifikacije baziranih na najmanje two-factor modelu.

Network Attacks. Uloga mrežnih administratora nije samo da štite serverske resurse, već i da se staraju da njihovi uređaji (Firewall, Router, Switch) ne budu zloupotrijebljeni od strane napadača. Napadi se svode na zloupotrebu posebnih servisa/protokola: routing protokola, CDP, STP, ARP itd. *Najbolja zaštita je u eliminisanju svih servisa koji nisu potrebni, obaveznom kriptovanju lokalnog password-a ili njegovoj zamjeni sa AAA autentifikacijom, eliminisanju http servisa kao ranjivog i obezbeđivanju autentifikacije u routing protokolima, pravilnom konfiguracijom portova odnosno primjenom sigurnosnih mehanizama koje posjeduju.*

Wireless Network attacks. Wireless mreže omogućavaju mobilnost i fleksibilnost zaposlenih ali takođe i opasnost za korporaciju. Ukoliko se konfiguriraju bez dodatnih mjera zaštite tj. sa standarnim protokolima za enkripciju WEP i sa broadcastom SSID-ova i njihovom standardnom postavkom, postaju laka meta za upad u sistem. Access Point mogu postati meta DoS napada, a u neobezbeđenim mrežama se mogu postaviti tkz. „rouge“ AP, koji mogu da autentifikuju prave korisnike. *Najbolja zaštita se svodi uvođenju onih mjera koje čine napade manje izvodljivim: eliminacija broadcast-a SSID-ova, uključivanje MAC filtera, izvođenje update-a firmware AP, korišćenje najsavremenijih i najačih mehanizama za enkripciju i autentifikaciju (WPA Enterprise, WPA2) smanjivanjem opsega dostupnosti AP na minimalno potrebnu itd.*

Trojans and Backdoor applications. Trojanci po svojoj prirodi su upravo nešto na što penetracioni testeri obraćaju posebnu pažnju jer mogu pokazati ranjivost mreže. Trojanci su programi koji se skriveno instaliraju na serverskim računarima da bi omogućili hakerima, najčešće, mogućnost udaljene administracije servera i pristup podacima na njemu, a u nekim slučajevima i DoS servisa. Alati koji se koriste obično imaju funkcije tipa: logovanje karaktera kucanih na tastaturi, editovanje OS registra, pristup komandnoj liniji, kontrola procesa, file transfer, udaljeni reboot i shutdown, kontrola miša i tastature, sakrivanja određenih procesa u cilju zaštite od sopstvenog otkrivanja itd. *Najbolja zaštiti od trojanaca su kvalitetni i redovno updatovani antivirus i trojan skener programi koji u svojoj bazi imaju najveći broj otkivenih trojanaca. Takođe, mrežni, a posebno host IDS/IPS sistemi su značajni u toj borbi jer posjeduju mehanizme detekcije istih. Na kraju, edukacija zaposlenih o mogućim scenarijima za napada, saznanje da svaka instalacija programa koji nije provjeren ili je preuzet sa Interneta sa nepoznatih izvora može značiti opasnost od Trojanaca.*

Elevation techniques. Kada se testiraju operativni sistema – bilo da je riječ o Unix ili Windows sistemima, cilj penetracionog testera je da dobije root odnosno administrativne privilegije. Najčešće se koriste *buffer overflow* tehnike odnosno pokreću poznati eksploiti, sa privilegijama običnog korisnika, da bi se dobile root privilegije. U cilju prikrivanja tragova o izvršenom napadu, koriste se rootkit-ovi, programi koji modifikacijom sistemskih komandi sakrivaju procese i fajlove koje koristi napadač. *Generalne preporuke kojih se treba držati u cilju zaštite OS od ove ali drugih vrsta napad sa: primjena striktno password politike, instalacija poslednjih verzija servisnih paketa (service pack) i sigurnosnih ispravki (hotfix), primjena AV programa i njihovo redovno update-ovanje,*

primjena firewall uređaja, network i host IDS/IPS sistema i fizička zaštita servera.

Buffer overflow attacks. *Buffer overflows* se definiše kao anomalija u kojoj se prilikom izvršenja određenog programa alokira više podataka u memoriji-buffer-u nego što je to programer predvidio, čime se omogućava startovanje određenih programa i upad u sistem. Osnovni razlog za ovakvo ponašanje programa je nedostatak mehanizama kojim se vrše provjere alociranja memorije (bound checking). *Sigurno programiranje, eliminacija funkcija koje nemaju kontrolu korišćenja memorije i primjena poslednjih programskih sigurnosnih zakrpa i ispravki.*

Denial of Service Attacks. Ova vrsta napada se realizuje na način da se zauzmu svi resursi mete, koja usled toga nije u mogućnosti da obavlja uobičajene radnje nesmetano ili pak sistem biva oboren. *Najbolja mjera zaštite su: upotreba Firewall i host i network IDS/IPS sistema, primjena poslednjih programskih sigurnosnih zakrpa i ispravki, eliminacija ICMP saobraćaja preko routera i firewall uređaja, uklanjanje nepotrebnih servisa i specifičnom sigurnosnom politikom u konfiguracijama mrežnih uređaja i OS (network and application hardening).*

5. PRIMJER IZVJEŠTAJA TESTIRANJA

Primjer jednog tipičnog izgleda izvještaja se sastoji iz dijela u kojem se opisuje šta je to pen testing, uvodnog dijela u kojem se definišu i opisuju stepeni ozbiljnosti otkrivenih ranjivosti, generalne informacije o kompaniji koja se testira i kompaniji koja izvodi testiranje, opseg testiranja, sumarni izvještaj otkrivenih ranjivosti, koja se metodologija koristi, opšte informacije o testiranim sistemima i nađenim slabostima, preporuke za njihovo rješavanje i zaključak. Izvodi iz takvog izvještaja prikazani su na slikama 2., 3., 4. i 5.:

- **CRITICAL VULNERABILITIES** (marked **RED**) - if exploited, these vulnerabilities allow full control of the target system or access to confidential information. Examples of such vulnerabilities include buffer overflow and SQL injection vulnerabilities.
- **HIGH RISK VULNERABILITIES** (marked **ORANGE**) - these vulnerabilities have serious security impact but do not allow full control over the target system. Examples of vulnerabilities in this category are denial of service, directory traversal or cross site scripting vulnerabilities, especially in the context of authenticated Web application components.

Slika 2. Opis ranjivosti.

CRITICAL RISK		
IP address	FQDN name	Vulnerabilities
192.168.1.1	server.company.local	1

HIGH RISK		
IP address	FQDN name	Vulnerabilities
192.168.1.1	server.company.local	0

MEDIUM RISK		
IP address	FQDN name	Vulnerabilities
192.168.1.1	server.company.local	7

Slika 3. Sumarno otkrivene ranjivosti po hostovima.

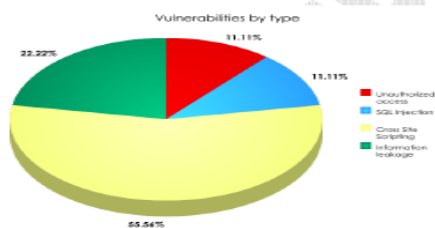


Figure 2: Vulnerabilities by type

Slika 4. Sumarno otkrivene ranjivosti po tipu.

10.1 GENERAL SERVER INFORMATION

The device with the IP address 192.168.1.1 (server.company.local) is running Linux Debian operating system with Linux kernel 2.6.20. The server is used for the majority of publicly available COMPANY services, such as web, mail, DNS, POP3 and IMAP. The table below shows general information about the server.

IP address	192.168.1.1
Operating system	Linux Debian 2.6.20
Hostname	server.company.local
Services	Mail, DNS, Webmail, Web (HTTP/HTTPS), POP3, IMAP
DNS domain	company.local

Table 3: General server information

10.2. OPEN PORTS

The following table (Table 4) lists open TCP and UDP ports on the device with the IP address 192.168.1.1.

Number	Port	Protocol	Service	Service banner
1.	25	TCP	SMTP	server.company.local ESMTP
2.	53	TCP	DNS	BIND 9.2.4
3.	80	TCP	HTTP	Apache/2.2.3 (Debian) PHP/5.2.0-8+etch7 mod_ssl/2.2.3 OpenSSL/0.9.8c
4.	443	TCP	HTTPS	Apache/2.2.3 (Debian) PHP/5.2.0-8+etch7 mod_ssl/2.2.3 OpenSSL/0.9.8c
5.	513	TCP	RLOGIN	-

Table 4: Open ports

10.3. IDENTIFIED VULNERABILITIES

Security vulnerabilities discovered on the host 192.168.1.1 (server.company.local) are given below. All vulnerabilities are categorized by their security risk level, which is calculated by multiplying two values: the likelihood that a given vulnerability will be discovered and exploited and the impact that may result from its exploitation.

Weak passwords and root account compromise

Vulnerability type: Unauthorized access
Likelihood: M
Impact: VH

Security risk: **CRITICAL**

CVSS V2 score: Basic Score: 9 (AV:N / AC:L / Au:S / C:C / I:C / A:C)
Temporal Score: 7.7 (E:POC / RL:W / RC:C)
Environmental Score: 8.8 (CDP:H / TD:ND / CR:H / IR:H / AR:M)
OVERALL SCORE: 8.8

Vulnerability description:

The security assessment identified several user accounts on the Linux server with weak passwords. These user accounts are susceptible to brute-force or dictionary attacks. During the security

assessment, "s experts were able to compromise the user account „backup“, which had the same password (backup).

It was also determined that this account has not been used for a long time; this indicates that user accounts have not been audited correctly.

Additionally, multiple „rhosts“ files containing „+“ have been identified on the system. Such files allow unauthenticated access from remote systems. An example of a user account with such „rhosts“ file is account. It is highly recommended that all „rhosts“ files are audited.

Recommendation:

All user accounts should be urgently audited. User accounts that are not needed should be removed. A security policy detailing appropriate user passwords should be created. Users should use only strong passwords with minimum length of eight characters, containing lower and upper case characters, and at least one number and special character. It is also recommended that all users change their passwords regularly, as required by the security policy (for example, every 3 to 6 months).

Slika 5. Tehnički detalji o ranjivostima.

6. ZAKLJUČAK

Sigurnosno testiranje, konačno, ima viši cilj nego da pokaže da je neki sistem ranjiv odnosno da se može kompromitovati. Nije mu ni cilj da se ukloni tipična ranjivost koja je otkrivena. Suštinski cilj je da otkrije koje su to procesne omaške dovele do mogućnosti kompromitovanja sistema, da ukaže na njih i da preporuke za njihovo rješavanje. Od značaja može biti i u dobijanju većeg budžeta za sigurnosna pitanja, koja su nekada podcijenjena. Siguran sistem se postiže primjenom savremenih tehnologija zaštite ali, prije svega definisanim i implementiranim mjerama i sigurnosnim politikama.

Sigurnost je kontinuitet procesa, napora i metoda zaštite i nikada nije apsolutna.

LITERATURA

- [1] Penetration Testing and Defence Strategy, Cisco Press
- [2] NIST Special Publication 800-115, Technical Guide to Information Security Testing and Assessment: www.nist.gov
- [3] Open Source Security Testing Methodology Manual: www.isecom.org/osstmm (OSSTMM)
- [4] The Open Web Application Security Project Testing Guide(OWASP)

PRIMJER PRIMJENE SISTEMA ZA MONITORING MREŽE NAGIOS AN IMPLEMENTATION EXAMPLE OF NETWORK MONITORING SYSTEM NAGIOS

Vladana Mrdak, Božo Krstajić, *Elektrotehnički fakultet Podgorica*

Sadržaj - Nagios je besplatan alat koji služi za nadzor i upravljanje uređajima u računarskoj mreži. U ovom radu opisane su njegove osnovne karakteristike i mogućnosti, njegova struktura i principi funkcionisanja. Predstavljen je njegov rad sa pluginovima i add-on paketima koji upotpunjuju njegovu funkcionalnost, zajedno sa osnovnim funkcijama njegovog web interfejsa. Dato je kratko poređenje Nagiosa i komercijalnog sistema za monitoring ZenOSS. U radu je predstavljen jedan primjer implementacije Nagiosa na velikom broju različitih mrežnih uređaja.

Abstract - Nagios is an Open Source enterprise-class network monitoring and management tool. In this paper, its main characteristics and capabilities are presented together with its structure and functioning principles. Integration with plug-ins and add-ons that complement its functionality are described together with basic operability of its web interface. A brief comparison between Nagios and commercial monitoring system ZenOSS has been made. Example of Nagios implementation on a large scale of network elements is presented.

1. UVOD

Postoji veliki broj besplatnih aplikacija koji se bave nadzorom rada mrežnih uređaja, međutim Nagios se izdvaja kao izuzetno kvalitetan alat koji se u zavisnosti od kreativnosti programera može iskonfigurisati da potpuno zadovolji potrebe svake kompanije. To je aplikacija koja služi za nadzor hostova, servisa i mreže, tako što šalje notifikacije administratorima kada se problem javi i kada se problem riješi, omogućavajući im da brzo pristupe njegovom rješavanju. Glavne karakteristike Nagiosa su otvoreni kod, robustnost, pouzdanost, velika konfigurabilnost, skalabilnost. Nagios ima jaku zajednicu entuzijasta koja aktivno učestvuje u njegovom razvoju. Iako je u početku razvijan na Linuxu, može se instalirati na velikom broju operativnih sistema. Proširena funkcionalnost Nagiosa koju čine Nagios plugin-ovi čini ovaj alat potpuno modularnim.

Jedna od glavnih funkcija Nagiosa je ispitivanje dostupnosti definisanog hosta pomoću ICMP paketa. On nudi čitav niz mogućnosti za nadzor, uključujući nadgledanje protokola i servisa kao što su SMTP, POP3, HTTP, NNTP, PING, itd. Nadzor resursa hostova kao što je opterećenje procesora, iskorištenje memorije, diska, praćenje procesa i logova je takođe njegova bitna funkcionalnost. Nadzor se može vršiti i nad faktorima okruženja kao što je temperatura. Postoji i mogućnost nadgledanja funkcionisanja baza podataka kao što su MySQL, Postgres, Oracle, SQL Server, itd...

Jednostavnost plugin-ova omogućava korisnicima da sami razvijaju svoje provjere servisa. Postoji čitava zajednica programera koja doprinosi Nagios projektu razvijajući različite pluginove i proširujući tako mogućnosti Nagiosa. Na jednom hostu moguće je definisati veliki broj servisa koji se paralelno provjeravaju. Definisanje mrežne hijerarhije sistema na Nagiosu je posebno korisno jer pruža potpuniju sliku mreže i obavještava koji uređaji nijesu u funkciji (down), a koji nedostupni jer uređaj između njih i Nagiosa nije u funkciji. Veoma važna funkcija Nagiosa je slanje notifikacija prilikom nastanka problema. Notifikacije se šalju administratorima putem e-maila ili SMS-a. Za svaki host i za svaki servis se može posebno definisati kontakt

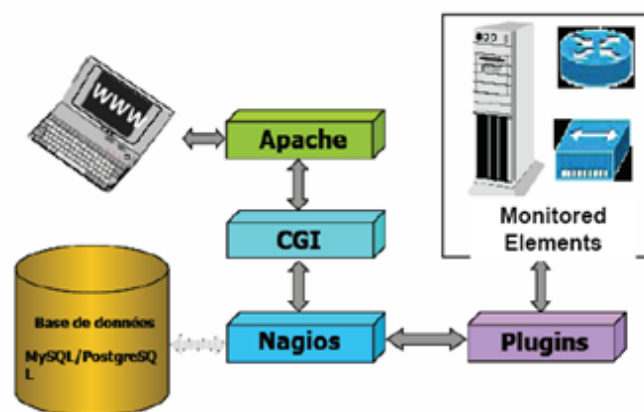
administratora. Nagios ima mogućnost definisanja "event handler-a", skripti koje se startuju po uočavanju problema od strane Nagiosa i otklanjaju problem bez intervencije administratora. Posebna pogodnost je web interface koji daje jasan pregled trenutnog stanja u mreži, istoriju problema i slanja notifikacija, logove, izvještaje, statistike itd.

2. STRUKTURA NAGIOS APLIKACIJE

Nagios funkcioniše na principu server-agenti arhitekture. On funkcioniše na hostu, a plugin-ovi na udaljenim hostovima koje treba nadgledati. Ovi plugin-ovi šalju informacije serveru, koji ih prikazuje kroz GUI (Sl. 1).

Nagios se sastoji od 3 dijela:

- Scheduler je serverski dio Nagiosa. U redovnim intervalima scheduler startuje plugin-ove i nakon toga se ponaša prema njihovim rezultatima.
- GUI je interface Nagiosa. Prikazan je pomoću Web stranica koje generiše CGI. Pristupa mu se preko web browsera kao <http://IPadresaNagiosservera>.
- Plugin-ovi provjeravaju servise na elementima koji se nadgledaju i vraćaju rezultat Nagios serveru. Njih može konfigurisati korisnik.



Slika 1 – Način funkcionisanja Nagiosa

Nagios aplikacija se uglavnom instalira u direktorijumu `/usr/local/nagios` na Unix platformi.

Nadzor i upravljanje Nagiosom se može obavljati preko komandne linije, editovanjem konfiguracionih fajlova. Takođe postoji niz konfiguracionih front-end-ova koji omogućavaju lakši rad sa Nagiosom kroz web interface. Izdvajaju se nagiosweb za 2.x verziju nagiosa i nagiosql za verziju 3.x. Instaliranjem neke od ovih aplikacija, stiće se mogućnost nadgledanja i upravljanja Nagiosom preko web interface-a. Ovi web interface-i omogućavaju korisniku dodavanje i brisanje hostova, hostgrupa, servisa, kontakata, planiranje downtime-a hostova i servisa, obradu statistike, odnosno upravljanje svim nagios procesima preko web-a. Definisanjem `http usera` može se ograničiti, odnosno omogućiti pristup informacijama Nagios procesa, pristup svim hostovima i servisima implementiranim na nagiosu, pristup konfiguracionim informacijama, može se ugasisi ili restartovati nagios proces ili mijenjati konfiguracija nagiosa.

3. ELEMENTI NAGIOSA I PRINCIP FUNKCIONISANJA

Objekti su svi elementi koji su uključeni u logiku nadgledanja i obavještanja administratora. Tipovi objekata su: servisi, servis grupe, hostovi, host grupe, kontakti, kontakt grupe, komande i vremenski periodi. Objekti se mogu definisati u jednom ili više konfiguracionih fajlova koji se specificiraju pomoću `cfg_file` direktive u glavnom konfiguracionom fajlu. Uglavnom se ovi konfiguracioni fajlovi smještaju na putanji `/usr/local/nagios/etc/objects/`.

Hostovi su centralni objekti u monitoring logici. To su obično fizički uređaji na mreži, serveri, radne stanice, ruteri, switch-evi, štampači, uređaji koji posjeduju IP ili MAC adresu. Oni mogu imati odnos roditelj/dijete sa drugim hostovima, koji predstavlja stvarne mrežne konekcije među njima kako bi se prikazala njihova hijerarhija i dostupnost na mreži. Hostovi su grupisani u host grupe radi lakšeg pregleda njihovog statusa, npr. u skladu sa sektorom koji ih administrira. Na svakom hostu posebno su implementirani različiti servisi koji su takođe centralni objekti u monitoring logici. Mogu biti atributi hosta (opterećenost procesora, iskorištenost diska, itd.), servisi koje obezbijeduje host (HTTP, POP3, FTP, SSH, itd.) ili nešto drugo u zavisnosti od uloge hosta (npr. DNS rekordi). Servis grupe čini više servisa povezanih nekom logikom kako bi omogućili lakši pregled njihovog statusa. Kontakti su ljudi uključeni u proces notifikacija, odnosno administratori sistema. Kontakt grupe su grupe od jednog ili više kontakata. Kada nagios ustanovi problem za određeni host ili servis, on šalje notifikaciju kontakt grupi koja je definisana u konfiguraciji tog hosta ili servisa. Notifikacija može biti e-mail, SMS, ili oboje, u skladu sa zahtjevima administratora. Komande se koriste kako bi obavijestile Nagios koji plugin treba izvršiti kako bi se odradile host i servis provjere, poslale notifikacije, startovali event-handler-i, itd. Vremenski periodi se koriste kako bi se definisao vremenski interval u kojem se vrši nadgledanje sistema ili u kojem se šalju notifikacije kontakt grupama.

Nagios je sposoban da nadgleda hostove i servise na dva načina: aktivno i pasivno. Češće se primjenjuju aktivne provjere (checks). Njih inicira Nagios proces, a startuju se u tačno definisanim intervalima vremena (npr. 5 min.). Kad

Nagios provjerava status određenog hosta ili servisa, izvršiće plugin koji će provjeriti stanje i vratiti Nagiosu rezultat. Nagios će procesuirati rezultat provjere i preduzeti odgovarajuće akcije (npr. poslati notifikacije ili pokrenuti event-handler). Pasivne provjere (checks) zahtijevaju implementaciju add-on-a na hostu ili integraciju sa nekom SNMP aplikacijom za menadžment i u tom slučaju host sam izvještava Nagios kada dodje do promjene stanja.

Trenutno stanje monitorovanih servisa i hostova određuju dvije komponente: status i tip. Status hosta može biti UP, DOWN ili UNREACHABLE, dok status servisa može biti OK, WARNING, UNKNOWN ili CRITICAL. Postoje dva tipa stanja u kom se servis ili host mogu naći: meko (soft) i tvrdo (hard) stanje. Soft stanje se javlja kada plugin vrati upozorenje ili grešku, međutim to stanje još uvijek nije provjereno dovoljan broj puta. Kada plugin vrati grešku dovoljan broj puta (taj broj se definiše prilikom konfiguracije hosta ili servisa), stanje prelazi u hard, i tada Nagios šalje notifikacije ili izvršava event-handler.

4. PLUGIN ARHITEKTURA

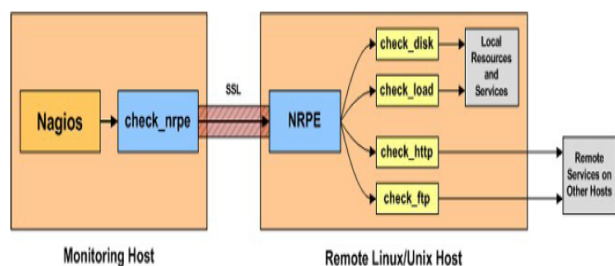
Kako bi Nagios funkcionisao, treba download-ovati i instalirati pluginove, koji se obično smještaju u direktorijumu `libexec/`, koji se nalazi u nagiosovom instalacionom direktorijumu. Plugin-ovi su binarni fajlovi ili skripte napisane u bilo kom programskom jeziku (C/shell/Perl/Php...) koji čine osnovu nagios monitoring sistema. Oni obavljaju sve provjere hostova i servisa vraćajući Nagiosu informaciju o njihovom statusu. Izlazna vrijednost plugina je 0-OK, 1 – Critical, 2 – Warning, 3 – Unknown.

Većina plugin-ova koji se nalaze u standardnom paketu Nagios plugin-ova su napisana za lokalnu mašinu koja se nalazi Unix platformi. Ako ti pluginovi treba da provjeravaju udaljene hostove, moraju se smjestiti na udaljeni Unix host i pokrenuti pomoću plugina `check_by_ssh` sa Nagiosa. Ovaj način se pokazao kao uspješan u slučajevima kada nema previše servisa koji se nadgledaju, iako opterećuje server na kojem se nalazi Nagios praveći dodatne ssh konekcije za svaki servis posebno i može rezultovati visokim opterećenjem servera. Postoji još nekoliko načina kako pokrenuti skriptu na udaljenim hostovima pod Unixom, ali svi oni zahtijevaju dodatne instalacije (add-ons) na tim hostovima. Ako Nagios nadgleda stotine ili hiljade servisa, onda vrijedi razmisliti o instalaciji odgovarajućeg add-ona na njemu.

5. DODACI (ADD-ONS) NAGIOSU

Postoji mnogo „add-on“ softverskih paketa koji funkcionišu sa Nagiosom. Koriste se kako bi proširili njegovu funkcionalnost ili integrisali Nagios sa drugim aplikacijama. Razvijeni su „add-on“ paketi sa raznim funkcionalnostima, kao što je upravljanje konfiguracionim fajlovima preko web interfejsa, slanje pasivnih provjera sa udaljenog hosta, ili pak „add-on“ paketi koji pojednostavljaju i proširuju sistem notifikacija. Važni add-onovi su NRPE i NSClient++ koji služe za nadgledanje resursa na udaljenim mašinama pod Unixom i Windowsom respektivno. Slično kao i već pomenuti plugin `check_by_ssh`, NRPE add-on omogućava izvršavanje pluginova na udaljenim Linux/Unix

hostovima (Slika 2). Plugin `check_nrpe` na monitoring hostu kontaktira NRPE daemon na udaljenom hostu, a onda on lokalno izvršava razne pluginove koji prosljeđuju rezultat daemonu, koji ih opet vraća Nagiosu.. Za razliku od plugina `check_by_ssh`, ovaj add-on ne opterećuje procesor mašine za monitoring.



Slika 2 - Grafički prikaz NRPE Add-on-a

6. UPOREĐENJE NAGIOSA I ZENOSSA

Kako pluginovi šalju izvještaj Nagiosu u obliku 0,1,2 ili 3, on nema uvida u to šta se nadgleda ni kakvi su ti parametri. Nagios samo obavještava da li je određeni servis, odnosno host funkcionalan ili ne. Ono što Nagiosu nedostaje je mogućnost praćenja određenih parametara i grafičkog prikaza tih parametara. Ta funkcionalnost je posebno korisna prilikom planiranja kapaciteta i projektovanja upgrade-a infrastrukture, kako zastareli sistemi ne bi uticali na izvršavanje aplikacija. To se može prevazići integracijom Nagios sa MRTG paketom, Cacti aplikacijom, ili nekim drugim add-on paketom oslonjenim na RRD tehniku što je prilično komplikovano. Još jedan dobar alat za nadzor mreže je ZenOSS. ZenOSS ima nekoliko prednosti u odnosu na Nagios. Jednostavniji je za implementaciju. Za razliku od Nagios on ima mogućnost nadzora performansi i njihovog grafičkog prikaza. Takođe, ZenOSS automatski pronalazi hostove, dok je Nagiosu potrebna Perl skripta `nmap2nagios` ili dodatni NACE paket. Međutim, ZenOSS ima dvije velike mane. Pored toga što je komercijalna aplikacija, opterećuje procesor tako da mu treba mnogo vremena za odgovor i zahtijeva mnogo bolji server nego Nagios.

7. PRIMJER IMPLEMENTACIJE NAGIOSA

Za potrebe Telekoma Crne Gore instalirana je aplikacija Nagios na HP serveru, snage procesora 2x2.8GHz sa 2GB RAM-a, na operativnom sistemu Fedora Core. Ta konfiguracija servera se pokazala dovoljnom za nadzor 86 hostova podijeljenih u 8 host grupa i 350 servisa sa mogućnošću proširenja. Host grupe su definisane u zavisnosti od sektora koji ih održava, odnosno u zavisnosti od njihove funkcije. Spektar hostova koji se nadgledaju je širok: serveri na Solaris platformi, na Linux platformi, na Windows platformi, firewall-i. Na svim hostovima je implementiran servis `check_ping` koji svakih 5 minuta provjerava njihovu dostupnost. Ostali servisi su implementirani u zavisnosti od zahtjeva njihovih administratora i od vrste i funkcije hosta, a uglavnom se izvršavaju na svakih 5 minuta, ako nije drugačije zahtijevano.

Za nadzor resursa na Windows platformama, koriste se posebni pluginovi koji provjeravaju iskorišćenost diska i

aktivnost određenih procesa. Na svakom udaljenom Windows serveru na kom Nagios izvršava ove pluginove je podignut `snmp` servis i postavljen `community` parametar. Između Nagios servera i hostova koji se nadgledaju mora biti propušten port 161/udp. Plugin `check_mssql.sh` provjerava konekciju sa Microsoft SQL Serverom 2000. Da bi funkcionisao, potrebno je otvoriti port 1433/tcp između Nagiosa i udaljenog hosta.

Na svim Unix hostovima postoji poseban korisnik kojeg koriste pluginovi koji svoje provjere obavljaju pomoću `ssh` konekcije. Serveru Nagios je omogućen pristup tim korisnicima bez autorizacije kopiranjem `ssh` ključa, kako bi se provjere mogle transparentno vršiti. Plugin `check_by_ssh` provjerava niz parametara konektujući se na udaljeni host i startujući plugin ili skriptu koji se nalazi na njemu. Na ovaj način se mogu dobiti informacije o dostupnosti Oracle baze preko plugina `check_oracle`, provjeriti `swap` prostor preko plugina `check_swap`, pretražiti log fajlovi pomoću plugina `check_log`, izbrojati fajlovi u direktorijumima, itd... Mogućnosti su neograničene. Na svim Unix hostovima postavljen je servis koji provjerava zauzetost hard diskova pomoću plugina `check_disk` u Perlu `check_disk.pl`.

Za nadzor funkcionalnosti protokola nije važno na kom operativnom sistemu se servisi izvršavaju. Postoji niz korisnih pluginova koji su uključeni u standardni `nagios-plugins-1.4.x.tar.gz` paket spremnih za primjenu bez modifikacije. Jedan od njih je `check_http` plugin koji provjerava dostupnost Web servera. Može testirati obične (`http`) i sigurne (`https`) servere, pratiti redirekcije, tražiti stringove i izvještavati kada sertifikatu ističe datum. Plugin `check_ftp` provjerava da li je moguće napraviti FTP konekciju na serveru. Na dva servera je postavljen servis koji uspostavlja `smtp` konekciju sa hostom pomoću plugina `check_smtp`.

Standardni paket pluginova obuhvata pluginove koji provjeravaju neke osnovne protokole i servise. Često se mora pristupiti implementaciji dodatnih pluginova koje je neko samostalno razvio i njihovoj modifikaciji. U ovom primjeru implementacije pokazali su se korisni `check_tacacs_plus.pl` koji provjerava da li `tacacs` servis funkcionise i da li se user može autorizovati. Plugin `check_sap_rfcping.pl` provjerava dostupnost i vrijeme odziva SAP sistema obraćajući se direktno određenom aplikacionom serveru. Da bi funkcionisao, na Nagiosu mora biti instaliran software RFCSDK. Plugin poziva `rfcping` koji je dio ovog software-a kako bi se povezao na aplikacioni server. Plugin `check_radius.pl` provjerava da li se Nagios može autorizovati na udaljeni Radius server. Potrebni podaci su `username` i `password` koji se implementiraju pomoću opcija `-u` i `-p` respektivno. Kako se nadgleda sistem koji pokriva veliki broj funkcija, većina servisa i pluginova je razvijana samostalno u skladu sa potrebama administratora, kao što je provjera da li je uspješno napravljena kopija podataka preko aplikacije Data Protector ili provjera da li se u određenom direktorijumu nalazi više od 100 datoteka.

Definisano je 6 vremenskih intervala u kojima se provjerava dostupnost hostova i servisa ili šalju notifikacije. Ove dvije funkcije ne moraju biti povezane, tako da se sistem može nadgledati 24 časa, a notifikacije se mogu slati samo u toku radnog vremena, u zavisnosti od kritičnosti sistema koji se nadgledaju.

Pomoću web interfejsa Nagios pruža mogućnost izrade grafičkih i statističkih izvještaja za praćene hostove i servise. Bitni izvještaji su Trend i Availability (Dostupnost). Trend je grafički prikaz stanja hosta, odnosno servisa u izabranom vremenskom periodu. Availability predstavlja statistički pregled dostupnosti u izabranom vremenskom periodu. Može se dobiti kako za host i servis tako i za grupe hostova i grupe

servisa. Na Slici 3 dat je pregled statusa nekih servisa koje Nagios prati. Servisi su grupisani po hostovima. Tu se može vidjeti status, kad je zadnji put provjeravan i koliko se dugo prati odgovarajući servis, kao i neke statusne informacije (zavisno od servisa).

Imena hostova/servisa su hiperlinkovi koji pružaju mogućnost detaljnijeg pregleda.

Host	Service	Status	Last Check	Next Check	Duration	Latency	Output
SAP Router T-Com	Connection via port 3299	OK	19-05-2008 14:09:18	0d 2h 19m 26s	1/2		Nema konekcije po portu sapdp99
SAP Router T-Mobile	Connection via port 3299	OK	19-05-2008 14:09:18	0d 2h 6m 45s	1/2		Nema konekcije po portu sapdp99
Solidus	Ping Service	OK	19-05-2008 14:07:35	45d 5h 1m 44s	1/2		PING OK - Packet loss = 0%, RTA = 13.61 ms
T-Portal	Ping Service	OK	19-05-2008 14:07:35	5d 12h 7m 53s	1/2		PING OK - Packet loss = 0%, RTA = 30.50 ms
Tara	CRNAGORA - Database	OK	19-05-2008 14:07:35	0d 8h 51m 21s	1/2		OK - dummy login connected
Taran	DISK USAGE	OK	19-05-2008 14:07:35	120d 3h 41m 19s	1/2		DISK OK
Tytan-DPS	Ping Service	OK	19-05-2008 14:07:35	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 10.87 ms
Tytan-DPS2	Ping II Interface	OK	19-05-2008 14:09:18	6d 1h 23m 8s	1/2		PING OK - Packet loss = 0%, RTA = 0.73 ms
Tytan-md	DISK USAGE	OK	19-05-2008 14:07:35	112d 4h 43m 35s	1/2		DISK OK
Tytan-md	Ping Service	OK	19-05-2008 14:07:35	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 0.74 ms
Tytan-sc1	Account po telefonu	OK	19-05-2008 13:57:35	0d 4h 40m 20s	1/2		OK: Time needed for execution of sql query is 2 seconds
Tytan-sc1	BSCC Database	OK	19-05-2008 14:07:35	28d 5h 55m 2s	1/2		OK - dummy login connected
Tytan-sc1	DISK USAGE	WARNING	19-05-2008 14:07:35	16d 5h 54m 3s	2/2		DISK WARNING [export/home 1.9G (5% free)]
Tytan-sc1	Dokumenti po accountu	OK	19-05-2008 13:57:35	1d 20h 40m 21s	1/2		OK: Time needed for execution of sql query is 4 seconds
Tytan-sc1	Ping Service	OK	19-05-2008 14:07:35	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 2.72 ms
Tytan-sc1	SSH	OK	19-05-2008 14:07:42	26d 4h 54m 53s	1/2		SSH OK - OpenSSH_4.1 (protocol 1.99)
Tytan-sc1	Ugovori po accountu	OK	19-05-2008 13:57:35	7d 0h 40m 23s	1/2		OK: Time needed for execution of sql query is 2 seconds
Tytan-sc2	DISK USAGE	OK	19-05-2008 14:07:42	53d 22h 22m 57s	1/2		DISK OK
Tytan-sc2	Ping Service	OK	19-05-2008 14:07:42	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 0.60 ms
Tytan-test	DISK USAGE	CRITICAL	19-05-2008 14:07:42	0d 0h 7m 38s	1/2		(Service Check Timed Out)
Tytan-test	Ping Service	CRITICAL	19-05-2008 14:07:42	0d 0h 7m 38s	1/2		PING CRITICAL - Packet loss = 100%
Tytan-web1	DISK USAGE	OK	19-05-2008 14:07:42	126d 8h 2m 31s	1/2		DISK OK
Tytan-web1	HTTP - Customer Care	OK	19-05-2008 14:07:42	4d 0h 35m 22s	1/2		HTTP OK HTTP/1.1 200 OK - 5590 bytes in 0.060 seconds
Tytan-web1	Ping Service	OK	19-05-2008 14:07:42	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 1.70 ms
Tytan-web1	http://bs.telekomca.com	OK	19-05-2008 14:07:42	109d 23h 37m 5s	1/2		HTTP OK HTTP/1.1 200 OK - 667 bytes in 0.003 seconds
Tytan-web2	DISK USAGE	OK	19-05-2008 14:07:42	126d 8h 2m 31s	1/2		DISK OK
Tytan-web2	HTTP - Customer Care	OK	19-05-2008 14:07:42	5d 5h 20m 22s	1/2		HTTP OK HTTP/1.1 200 OK - 5589 bytes in 0.041 seconds
Tytan-web2	Ping Service	OK	19-05-2008 14:07:42	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 1.27 ms
Tytan-web2	http://bs.telekomca.com	OK	19-05-2008 14:07:42	126d 8h 2m 31s	1/2		HTTP OK HTTP/1.1 200 OK - 666 bytes in 0.003 seconds
WEB server T-Com	HTTP	OK	19-05-2008 14:09:18	0d 2h 10m 44s	1/2		HTTP OK HTTP/1.1 200 OK - 727 bytes in 0.007 seconds
WEB server T-Com	Ping Service	OK	19-05-2008 14:07:42	45d 21h 12m 42s	1/2		PING OK - Packet loss = 0%, RTA = 3.52 ms
Web-Care	Ping Service	OK	19-05-2008 14:07:42	45d 5h 10m 6s	1/2		PING OK - Packet loss = 0%, RTA = 2.45 ms
Zaklan	HTTP	OK	19-05-2008 14:07:42	109d 4h 40m 35s	1/2		HTTP OK HTTP/1.1 200 OK - 1008 bytes in 0.016 seconds

Slika 3 - Pregled statusa servisa na Nagios web interfejsu za razmatrani primjer

8. ZAKLJUČAK

U današnje vrijeme izuzetno je bitno da računarski sistem bude dostupan u svakom trenutku kako bi pružio očekivani servis korisnicima. Na današnjem konkurentnom tržištu uspješnost preduzeća se mjeri dostupnošću servisa koje nude. Zbog toga je vrlo važno da administratori sistema u svakom trenutku budu obaviješteni o stanju u njihovoj mreži kako bi na vrijeme mogli da reaguju i spriječe eskalaciju problema.

Nagios je odlično rješenje aplikacije za nadzor mreže koje sistem administratori mogu u potpunosti prilagoditi svojim potrebama. Web GUI je brz i intuitivan, jednostavno se definiše kome se šalju e-mail i SMS notifikacije kada dodje do problema na servisima ili hostovima, a pluginovi postoje za ogroman broj kako hardvera tako i softvera. Izvještaji koji se dobijaju su izuzetno korisni prilikom analize kvaliteta sistema i servisa koje oni pružaju.

Za rad sa Nagiosom potrebno je napredno poznavanje Unixa i mrežnih protokola, a dosta često i servisa koje

treba nadgledati, što čini ovaj sistem komplikovanim za implementaciju. Često se moraju modifikovati kodovi kako bi se dobio željeni rezultat. Ipak mu treba dati prednost u odnosu na komercijalne pakete koji su jednostavni za implementaciju i održavanje i nude čitav niz sličnih funkcionalnosti, ali su zato jako skupi. Nagios je besplatan, fleksibilan, skalabilan i poslije inicijalne konfiguracije, relativno lak za održavanje.

LITERATURA

- [1] Ethan Galstad, "Nagios Documentation" <http://nagios.sourceforge.net/docs/nagios-3.pdf>, 2009
- [2] Chris Burgess, "Nagios Book – Complete Guide" <http://www.nagiosbook.org/>
- [3] Doug Dineley, James R. Borck, High Mobley, "Best of Open Source Networking Software", 2009 <http://www.infoworld.com>
- [4] <http://www.linuxworld.com/news/2007/031207-top-5-security.html?page=6>
- [5] <http://www.nagiosexchange.com>

GRAFIČKO PRIKAZIVANJE VEKTORA ELEKTRIČNIH VELIČINA NA TFT-LCD DISPLEJU GRAPHICAL PRESENTATION OF VECTORS OF ELECTRICAL QUANTITIES USING A TFT DISPLAY

Aleksandar Dedić, *Elektrotehnički fakultet Podgorica*

Sadržaj-U ovom radu je dat jedan način grafičkog prikazivanja vektora dvije električne veličine na TFT-LCD displeju dimenzija 320x240 tačkaka. Vektori su prikazani u obliku linija sa strelicama, svaka pod određenim uglom u odnosu na faznu osu. Informacije o vrijednostima veličina i vrijednostima uglova date su u obliku brojeva. Za ovo prikazivanje koristi se kombinacija vektorskog i bitmapiranog metoda prikazivanja.

Abstract - A method for graphical presentation of vectors of two electrical quantities on a TFT-LCD display with 320x240 pixels is presented in this paper. The vectors are shown in the form of the lines with arrows, each of them under the certain angle relative to the phase axis. Information on the values of the quantities and the values of the angles are given in the numerical form. A combination of the vector method and the bit-mapping method are used for this presentation.

1. UVOD

U elektrotehnici često postoji potreba da se vrše mjerenja efektivnih vrijednosti naizmjeničnih napona i struja, kao i njihovog faznog pomjeraja u odnosu na neku referentnu veličinu. Rezultati ovakvih mjerenja daju se kao brojčane vrijednosti, ali je, ponekad, zanimljivo da se električna veličina prikaže u grafičkom obliku, kao vektorski dijagram. U tom slučaju se električna veličina prikazuje kao linija sa strelicom na kraju, pri čemu je dužina linije srazmjerna izmjerenoj efektivnoj vrijednosti, a ugao, pod kojim je linija prikazana, odgovara faznom pomjeraju u odnosu na referentnu veličinu.

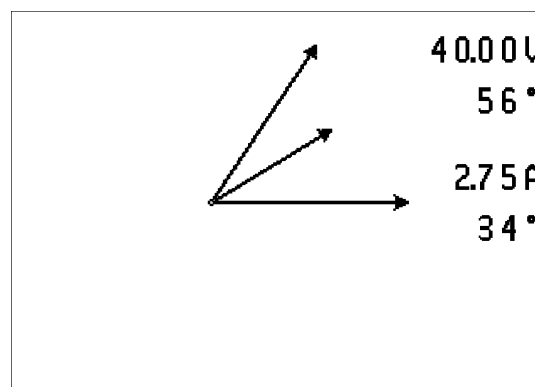
Ovaj rad je nastao kao dio praktične realizacije mjernog instrumenta za istovremeno mjerenje efektivnih vrijednosti 2 veličine: napona i struje, frekvencije 50 Hz. Kao referentna veličina za određivanje faznog pomjeraja, kako struje tako i napona, koristi se napon električne mreže. Mjerenje obje veličine vrši se korišćenjem analogno-digitalnih konvertora. Grafički prikaz sastoji se u prikazivanju dva vektora, tj. dvije linije sa strelicama. Da bi se razlikovalo koja linija kojoj veličini pripada, linija koja predstavlja napon prikazana je u crvenoj boji, a linija koja predstavlja struju, u plavoj. Pored ovoga, na displeju je prikazana i fazna osa u obliku linije crne boje. Kako se, zbog loše rezolucije, vektorski prikaz ne može koristiti za očitavanje vrijednosti električnih veličina, na desnoj strani displeja nalaze se brojevi, koji daju izmjerene vrijednosti. Brojevi su takođe prikazani u crvenoj i plavoj boji, zavisno od toga koju veličinu predstavljaju. Izgled displeja dat je na sl. 1.

2. DISPLEJ

Za realizaciju ovog rada, korišćen je displej sa 240 linija i 320 tačkaka u svakoj liniji.

Osnovna namjena pomenutog displeja je prikazivanje video informacija u minijaturnim televizorima i slično. Zbog toga je program za formiranje slike morao biti urađen prema

specifičnim zahtjevima displeja. Sam program nije vezan za specifičnosti konkretnog proizvođača, pa se može koristiti za sve srodne displeje uz, eventualno, manje izmjene.



Sl.1 Primjer prikazivanja vektora i brojeva na displeju

Osnovni problem rada displeja je što ima ugrađenu dinamičku memoriju sa lokacijama za svaku tačku, pri čemu lokacije nijesu direktno pristupačne korisniku. Podaci o osvijetljenosti moraju se unositi redno, od prve, do posljednje tačke, bez mogućnosti adresiranja. Prije početka unošenja podataka, mikrokontroler mora formirati signale za vertikalnu i horizontalnu sinhronizaciju [1], [2]. Slika, koju treba prikazati, mora se formirati u memoriji mikrokontrolera prije početka prenosa.

Uobičajeni način formiranja slike je bitmapirani prikaz. Kako displej ima 320x240=76800 tačkaka, a za svaku tačku treba minimum tri podatka za tri boje, treba imati korisničku memoriju veličine minimum 230400 bita. Standardni mikrokontroleri sa ovakvom memorijom se, za sada, ne nalaze na tržištu, pa je potrebno koristiti spoljašnju memoriju minimalne veličine 256 kB. Za potrebe mjernog instrumenta, za koji je rađen program, ovo je neracionalno jer bi se koristio veoma mali dio memorijskog prostora, a i proces formiranja slike bi se nepotrebno usporavao. Zbog toga je urađen originalni grafički program koji je kombinovao

vektorsko i bitmapirano formiranje slike. Za potrebe ovog programa potrebno je malo više od 4 kB korisničke memorije, računajući tu i memorijski prostor za obradu podataka iz A/D konvertora.

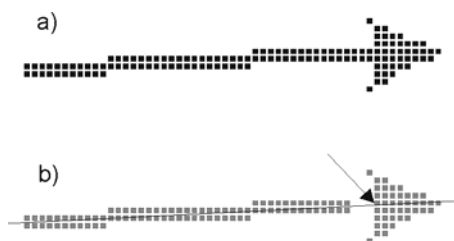
Površina displeja dijeli se na dva područja: polje vektora i polje brojeva.

3. FORMIRANJE VEKTORA

Za prikazivanje vektora koristi se područje veličine 240x240 tačaka, smješteno u lijevom dijelu displeja. Svim tačkama, u tom području, dodijeljene su brojčane vrijednosti, koje se, u daljem tekstu, nazivaju "apsolutne koordinate". Za označavanje koordinata koriste se dva bajta. Viši bajt se koristi za označavanje broja linije (ordinate) a niži bajt, za broj tačke u liniji (apscisa). U praktičnoj realizaciji koristi se 16-bitni mikrokontroler, pa se oba podatka smještaju u memoriju kao jedinstveni broj [3].

Prema TV standardu, u gornjem lijevom uglu nalazi se početna tačka, tj. tačka sa koordinatama (1, 1). Zbog toga treba, pri računanju apsolutnih koordinata, strogo voditi računa da se tačke, sa povećanjem vrijednosti ordinate, pomjeraju prema donjem dijelu displeja, što je suprotno od uobičajenog načina prikazivanja ordinate u koordinatnom sistemu.

Vektor se, kao što je rečeno u uvodu, prikazuje u obliku linije sa strelicom na kraju. Početna tačka vektora je fiksirana i ima koordinate (121, 121), a sam vektor se prikazuje kao pokretni objekat koji se obrće oko početne tačke. Primjer vektora dat je na sl. 2 (a).



Sl.2 Primjer formiranja i prikazivanja vektora.
(a) prikaz vektora na displeju.
(b) formiranje linije i strelice u odnosu na idealnu liniju.

Podaci o dužini linije i uglu u odnosu na faznu osu, dobijaju se iz A/D konvertora. Navedeni podaci predstavljaju vektor u polarnom koordinatnom sistemu i oni se moraju prevesti u dekadni koordinatni sistem. Pretvaranje podataka se vrši jednostavnim relacijama:

$$X=A\cos\varphi \quad (1)$$

$$Y=Asin\varphi \quad (2)$$

gdje je A vrijednost, odnosno dužina vektora, a φ ugao. U konkretnom programu, umjesto relacije (2) koristi se relacija:

$$Y=X\tan\varphi \quad (3)$$

Koordinate tačaka X i Y moraju biti u cjelobrojnom obliku

pa se rezultati, dobijeni računskim putem, moraju zaokružiti na bližu cjelobrojnu vrijednost.

Vektor se može prikazati samo u obliku diskretnih tačaka, a kako su dimenzije displeja ograničene, dužina vektora može biti najviše 120 tačaka. Uzeto je da dužina linije može biti najviše 110 tačaka, a dužina strelice 10. Ograničenje se postiže tako što se rezultati, dobijeni iz A/D konvertora dijele određenim koeficijentom, da bi maksimalnoj vrijednosti iz A/D konvertora odgovarala vrijednost 110.

Mikrokontroler nema mogućnost direktnog rada sa trigonometrijskim funkcijama, pa se, zbog toga, pravi tabela koeficijenata $\cos\varphi$ i $\tan\varphi$ u skokovima po jedan stepen. Kako su računске operacije sa decimalnim razlomcima komplikovane, koeficijenti se dobijaju tako, što se vrijednosti $\cos\varphi$ i $\tan\varphi$ množe sa 4096 (&1000 u heksadekadnom obliku). Ovim se decimalne vrijednosti $\cos\varphi$ i $\tan\varphi$ prevode u heksadekadne, iz razloga što se sve operacije u programu izvode sa brojevima u heksadekadnom obliku [4]. Tabela sa ovakvim koeficijentima smješta se u memoriju. Vrijednosti ovih koeficijenata imaju cjelobrojni oblik, ali suštinski predstavljaju decimalne razlomke pa se računске operacije izvode kao aritmetičke operacije sa pokretnom tačkom. Da bi se dobile tražene cjelobrojne vrijednosti za X i Y , poslije množenja vrijednosti vektora sa koeficijentima iz tabele, iz rezultata se odbacuju 12 nižih bitova jer oni predstavljaju decimalni razlomak. Kako mikrokontroler, koji se koristi u konkretnom slučaju, nema ugrađen EEPROM, svi koeficijenti, kao i ostali podaci, smještaju se u programsku memoriju [5]. Tabela za sve vrijednosti uglova 0-360° bila bi previše velika, pa se ovo područje dijeli na oktante, a tabela se pravi samo za prvi oktant, tj. za uglove 0-45°. U tom slučaju se izračunavaju relativne koordinate kao da se vektor nalazi u prvom oktantu, a zatim se one preračunavaju u apsolutne, u potrebnom oktantu.

Postupak izračunavanja koordinata sastoji se u tome što se najprije, na osnovu podatka iz A/D konvertora, odredi broj tačaka koje će biti prikazane, što predstavlja dužinu linije. Zatim se taj broj množi sa koeficijentima $\cos\varphi$ i $\tan\varphi$ iz tabele, čime se dobijaju relativne X i Y koordinate u prvom oktantu. Ovako dobijene vrijednosti preračunavaju se u apsolutne koordinate u potrebnom oktantu, prema vrijednosti ugla. U prvom oktantu, vrijednost apscise se dobija kad se X vrijednost sabere sa 121, a ordinate, kad se od 121 oduzme Y vrijednost. Za drugi oktant, X i Y vrijednosti mijenjaju svoja mjesta pa se ordinata dobija kad se od 121 oduzme X , a apscisa, kad se Y sabere sa 121. U trećem oktantu takođe X i Y mijenjaju svoja mjesta, pa se apsolutna apscisa dobija kad se od 121 oduzme Y , a ordinata, kad se od 121 odzme X . Četvrti oktant je sličan prvom, s tim, što se od 121 oduzimaju i X i Y vrijednosti. Na sličan način izračunavaju se apsolutne koordinate i u ostalim oktantima. Po završetku računanja i smještanja vrijednosti u memoriju, broj dobijen iz A/D konvertora se umanjuje za 1 (dekrementira se) i postupak se ponavlja dok se ne dobiju koordinate za sve tačke linije.

Na displeju se pomenute tačke prikazuju tako da predstavljaju pravu liniju. Međutim, zbog malih dimenzija tačaka, linija je previše tanka i ima nezadovoljavajući estetski izgled. Zbog toga se svakoj tački dodaje još jedna, dopunska, koja može da bude iznad ili ispod osnovne. Kao kriterijum

položaja dopunske tačke u odnosu na osnovnu, uzima se fiktivna linija koja bi trebala da odgovara idealnoj liniji na displeju. Ta linija je prikazana na sl. 2 (b). Ako se više od polovine osnovne tačke nalazi iznad fiktivne linije, dodaje se dopunska tačka ispod osnovne i obrnuto. Računski kriterijum za ovaj slučaj je veoma jednostavan. Kad se izračunata X vrijednost pomnoži koeficijentom za $\tan\phi$, ispituje se 12-ti bit, odnosno najviši bit iz dijela koji se odbacuje. Ako je vrijednost tog bita nula, dodaje se dopunska tačka iznad osnovne, a ako je jedinica, ispod osnovne.

Poseban problem je bilo formiranje strelice na vrhu vektora. Određivanje koordinata tačaka strelice računskim putem je prilično komplikovano i ne daje zadovoljavajuće rezultate u estetskom pogledu, pa se, zbog toga, ovdje koristi bitmapirani način prikazivanja. Strelice su predhodno nacrtane za razne uglove vektora, ali samo u prvom oktantu. Zbog loše rezolucije displeja, dovoljan je jedan crtež na svaka dva, pa čak i tri stepena. Primjer jedne strelice može se vidjeti na sl. 2 (b). Kao početna tačka strelice, uzima se tačka koja se dodiruje sa krajnjom tačkom linije. Ta tačka ima relativne koordinate (1, 1) u odnosu na krajnju tačku linije i označena je na slici. Koordinate svih ostalih tačaka određuju se sa crteža u odnosu na krajnju tačku linije, a vrijednosti se smještaju u memoriju u obliku tabele. Ove vrijednosti koordinata mogu biti i negativne. Stvarne relativne koordinate svih tačaka u prvom oktantu dobijaju se sabiranjem ovih vrijednosti sa koordinatama krajnje tačke linije. Preračunavanje relativnih koordinata tačaka strelice u apsolutne, u odgovarajućim oktantima, vrši se na isti način kao što je objašnjeno za liniju.

Fazna osa je fiksirana, pa se koordinate njenih tačaka ne izračunavaju nego se programski unose u korisničku memoriju.

Rezultati izračunavanja koordinata svih tačaka odmah se smještaju u korisničku memoriju i to u dio koji je rezervisan za te podatke. Taj dio memorije podijeljen je u tri područja i to područje za vektor napona, vektor struje i faznu osu. Ova podjela je napravljena da se koordinatama tačka ne bi dodjeljivali podaci kojom bojom treba prikazati pojedine tačke. Smještanje u prvo područje, tj. područje vektora napona, podrazumijeva da su tačke na displeju crvene boje, u drugom, tj. području vektora struje, tačke su plave boje, a u području fazne ose tačke su crne. Po završetku unosa, svi podaci se moraju sortirati po vrijednostima i to od najmanje ka najvećoj. Sortiranje se vrši za svako područje posebno. Ovo je neophodno jer bi upoređivanje tekućih koordinata tačaka displeja sa nesortiranim podacima u memoriji, tokom unošenja podataka u displej, drastično usporilo proces i izazvalo nestabilnost slike. Da se ne bi rad programa usporavao brisanjem predhodnih podataka, po završetku unosa novih podataka u sledeću memorijsku lokaciju stavlja se vrijednost nula. To je znak da dalje podataka nema.

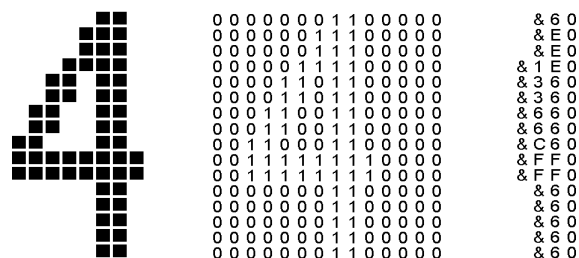
4. PRIKAZIVANJE VEKTORA

Unošenje podataka u displej vrši se na sledeći način: rezerviše se 16-bitni registar, koji služi kao brojač linija i tačaka u liniji, pri čemu se viši bajt registra koristi kao brojač linija, a niži, kao brojač tačaka. Pored toga rezervišu se i tri

lokacije kao baferi podataka i u njih se unesu prve vrijednosti iz memorije. U registar se stave koordinate početne tačke (10000001 u binarnom obliku). Vrijednost, koja se nalazi u registru, upoređuje se redom sa podacima u sva tri bafera. Ako se u nekom baferu nalazi ista vrijednost kao u registru, na ulaz za crvenu ili plavu boju displeja stavlja se jedinica, odnosno stavlja se nule (crna boja) i vrijednost u baferu se zamjenjuje sledećom iz memorije. Ako se podaci ne podudaraju, stavlja se jedinice u sve ulaze da bi tačke bile osvijetljene bijelom bojom. Kad se u nekom baferu pojavi vrijednost nula, dalja upoređivanja za tu boju se ne vrše već se na ulaz displeja stavlja jedinice, osim u slučaju kada se postavlja neka druga boja. U tom slučaju se postavljaju nule. Poslije završenog upoređivanja i postavljanja odgovarajućih vrijednosti na ulaze boja (RGB ulaze), na takt ulaz displeja dovodi se impuls, čime se vrši unos podataka. Vrijednost u registru se, poslije ovoga, inkrementira i postupak se ponavlja. Po završetku unosa 240 tačaka prve linije, inkrementira se viši bajt registra i postupak se ponavlja dok se ne unesu podaci za svih 240 linija. Ovdje treba napomenuti da svaka linija sadrži 320 tačaka, pa kad se završi unos podataka za 240 tačaka, prelazi se na program za unos brojeva u sledećih 80 tačaka.

5. FORMIRANJE BROJEVA

Polje brojeva zauzima područje 240 linija sa 80 tačaka u svakoj i nalazi se u desnom dijelu displeja. U ovom području koristi se bitmapirani način prikazivanja iz razloga što je za njega, ovdje, potrebno 2,4 kB slobodne memorije, a toliki memorijski prostor se može rezervirati u mikrokontrolerima sa 8 kB radne memorije.



Sl.3 Primjer formiranja i prikazivanja cifre "4".

Način formiranja brojeva u memoriji najlakše se može vidjeti sa primjera na sl. 3. Na toj slici je prikazana cifra "4", pri čemu je prvo dat njen izgled na displeju, a zatim su date vrijednosti koje se smještaju u memorijske lokacije i to prvo u binarnom, a zatim u heksadekadnom obliku. Cifra se nalazi u pravougaoniku dimenzija 14x16 tačaka. Horizontalne tačke čine jednu liniju i smještaju se u 16-bitnu memorijsku lokaciju kao jedinstven broj. Ukupno, za svaku cifru, potrebno je šestnaest 16-bitnih lokacija. Ovako formirani podaci smještaju se u programsku memoriju. Početna adresa lokacije u programskoj memoriji, od koje su smjestene cifre, u konkretnom slučaju, je 19200 ali može biti i druga uz uslov da se završava nulom. Prije prenošenja podataka o ciframa iz programske memorije u korisničku memoriju, potrebno je podatke iz A/D konvertora pretvoriti u BCD oblik. U konkretnom slučaju se rezultati mjerenja prikazuju sa četiri cifre, pa se one smještaju u četiri memorijske lokacije označene sa BCD1 – BCD4. Adresa lokacije sa podacima

prve linije tražene cifre u programskoj memoriji dobija se tako, što se vrijednost iz BCD lokacije pomnoži sa 16 i sabere sa početnom lokacijom od koje su smještene cifre, u konkretnom slučaju sa 19200. Sledećih 15 lokacija dobijaju se rednim inkrementiranjem adresa. Ovako dobijeni podaci prenose se u korisničku memoriju u prostor za brojeve.

Pozicioniranje cifara vrši se tako što se prve linije svih cifara, prema redosledu BCD lokacija, postavljaju u 5 lokacija prostora za brojeve rezervisanih za početnu liniju brojeva na displeju. U konkretnom slučaju, brojevi se prikazuju od 16-te linije. Da bi cifre na displeju imale isti međusobni razmak, podatke za prvu liniju svake cifre treba, u pomenutim lokacijama, pomjerati lijevo ili desno, po potrebi. Ovaj se postupak ponavlja dok se ne smjeste svih 16 linija traženih cifara. Postupak za postavljanje slova i oznaka za stepen je identičan, s tim što se za njih rezervišu posebne lokacije u programskoj memoriji.

Decimalna tačka može se prikazivati kao poseban znak, međutim, u konkretnom programu ona je urađena kao sastavni dio cifre tj. kao kvadrat veličine 3x3 tačke u donjem desnom dijelu crteža. Kod ovakvog načina formiranja, podrebnost je da se, pored osnovnog niza cifara 0 - 9, u memoriju unese i dopunski niz istih cifara sa tačkom. Početna adresa svake cifre pomjerena je za $16 \times 16 = 256$ memorijskih lokacija u odnosu na osnovnu, a zamjena potrebne cifre cifrom sa tačkom vrši se tako što se vrijednosti u potrebnoj BCD lokaciji doda 16.

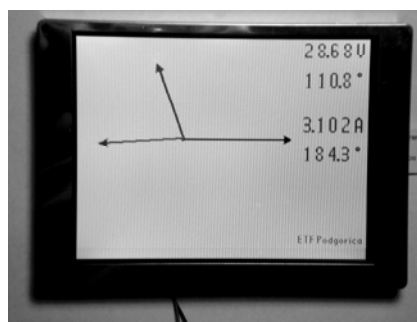
6. PRIKAZIVANJE BROJEVA

Prenošenje podataka iz memorije u displej vrši se na sledeći način: kad se u prvih 240 tačaka prve linije displeja postave podaci iz memorijskog prostora vektora, prije unošenja podataka u 241-u tačku te linije ispituje se najviši (16-ti) bit prve lokacije u memorijskom prostoru brojeva. Ako je u tom bitu nula, na RGB ulaze displeja postavljaju se jedinice, pa se ta tačka osvetljava bijelom bojom. Ali ako je tamo jedinica, postavlja se jedinica na ulaz crvene ili plave boje. Kad se ispita 16-ti bit, ispituje se sledeći niži (15-ti) na isti način kao u predhodnom slučaju, a odgovarajući podatak unosi se u 242-u tačku. Postupak se ponavlja dok se ne ispituju svih 16 bitova prve memorijske lokacije, zatim se ispituju bitovi sledeće lokacije i tako dalje, do kraja linije displeja. Kad se unesu podaci u prvu liniju displeja, počinje se sa drugom na isti način, tj. unesu se prvo 240 podataka iz memorijskog prostora vektora, a zatim od 241-e unose se podaci iz sledećih 5 lokacija u memorijskom prostoru brojeva. Kojom bojom su obojene tačke zavisi od toga da li se prikazuju podaci o naponu ili struji. U konkretnom slučaju se prvih 100 linija koriste za napon pa se prikazuju u crvenoj boji, a ostale linije u plavoj.

7. ZAKLJUČAK

U praktičnoj realizaciji korišćen je mikrokontroler dsPIC33FJ64GP802 radne frekvencije 40 MHz i TFT-LCD displej VGG322403-7UFLWA. Frekvencija takta displeja je trebala, prema fabričkoj preporuci, da bude 6,4 MHz. Kako vrijeme između dva takta nije bilo dovoljno za izvršavanje potrebnog broja instrukcija, frekvencija je morala biti snižena

na 2 MHz. Ovo je imalo za posledicu neznatno treperenje tačaka na krajevima linija i poslednje linije, ali je problem riješen dodavanjem po jedne tačke na kraju svake linije i dodavanjem još jedne prazne linije na kraju. Veći problem je bio usled kratkog vremena čuvanja podataka u dinamičkoj memoriji displeja, zbog čega je dolazilo do blagog treperenja čitave slike. Problem je riješen tako što se, u toku obrade podataka iz A/D konvertora, tri puta vršilo obnavljanje slike koristeći, pri tome, podatke iz predhodnog računanja. Ovo je malo usporilo rad programa pa se promjena prikaza na displeju mogla vršiti tek na svakih 0,2 sekunde. Međutim, za praktičnu primjenu, ovo vrijeme je sasvim dovoljno jer se brže promjene brojevanih podataka inače prate otežano. Na slici 4 data je fotografija displeja



Sl. 4 Displej

Za navedeni način grafičkog prikazivanja na displeju potrebno je malo više od 4 kB korisničke memorije pa je spoljašnja memorija, uobičajena za grafičke programe, nepotrebna. Program se može izmijeniti da zahtijeva i manje od 4 kB, ali se u tom slučaju nepotrebno komplikuje i usporava. Što se tiče programske memorije, dovoljno je imati 3 kB ali, s obzirom da je korišćen mikrokontroler sa 64 kB, iskorišćeno je znatno više, što je za posledicu imalo povećanje brzine rada.

LITERATURA

- [1] Vbest Electronic LTD, *Product specification for LCD module VGG322403-7UFLWA*, <http://www.vbest.com.tw>
- [2] Ken Green, *Interfacing LCD Panels to Microcontrollers*, Sharp Microelectronics of the Americas, www.sharpsma.com
- [3] Microchip Technology Inc. *dsPIC33 Family Reference Manual*, <http://support.microchip.com>
- [4] Sun Microsystems Inc. *Numerical Computation Guide*, Palo Alto 2001, <http://www.sun.com>
- [5] Microchip Technology Inc. *dsPIC33FJxxxGPx02x04 DataSheet*, <http://support.microchip.com>

PROSTORNA NAVIGACIJA U ODNOSU NA OBJEKAT POSMATRANJA SPATIAL NAVIGATION IN RELATION TO OBSERVED OBJECT

Dušan Kostić, *Elektronski fakultet u Nišu*
Branko Marović, *Računarski centar Univerziteta u Beogradu*

Sadržaj - Navigacija u 3D aplikacijama može biti složena za upotrebu. Skoro svaka aplikacija sa 3D prikazom ima potrebu za mehanizmom navigacije koji je intuitivan i lak za učenje. Ovaj rad je pokušaj da se korisnički interfejs jedne 3D aplikacije učini takvim, ali da ipak pokrije sve zahteve naučne vizuelizacije. Navigacija u 3D okruženju je složen problem čije rešenje zavisi od konkretne primene. Opisano rešenje daje posmatraču punu slobodu pozicioniranja, uz vezivanje kretanja za objekat koji se posmatra. Na ovaj način, korisnik ima utisak da manipuliše objektom, što mu pruža bolju orijentaciju od „leteće“ navigacije.

Abstract - Navigation in 3D applications can be very complex to use. The majority of 3D applications need a navigation that is intuitive and easy to learn. This paper is an attempt to implement such an interface for a 3D application, while satisfying scientific visualization demands. Navigation in a 3D environment is a complex problem for which the solution mainly depends on intended use. The described solution provides full freedom of positioning for the observer, tying the motion with the observed object. In such a way, the user feels as if he is manipulating the object, which provides better orientation than fly-through navigation.

1. UVOD

Problem navigacije u 3D okruženju je sve aktuelniji zbog pojave sve većeg broja aplikacija koje se služe 3D prikazom. Ovaj trend će se sigurno nastaviti i narednih godina jer je 3D prikaz podataka najprirodniji i najbliži korisniku.

Korišćenje 3D prikaza, sem očiglednih pogodnosti, uvodi i novi skup problema prilikom koncipiranja i upotrebe korisničkog interfejsa. Oni koji su upoznati sa korišćenjem 3D aplikacija su možda davno zaboravili koliko je teško savladati osnovne koncepte koji su potrebni za kontrolu pogleda ili tačke i pravca posmatranja.

Istraživanja [1] pokazuju da će mnogi korisnici odustati od korišćenja 3D aplikacije zato što je njihov prvi susret sa tom aplikacijom neprijatno i neproduktivno iskustvo, čak i pri pokušaju da se urade najosnovnije operacije u 3D navigaciji kao što je, na primer, „pogledati objekat sa leve, pa sa desne strane“. Tipovi korisnika koji često imaju problem da savladaju 3D navigaciju su: korisnik koji prenosi svoje veštine iz 2D aplikacija u 3D aplikaciju i korisnik koji ima prerano stvorene i neadekvatne ideje o tome kako bi sistem za navigaciju trebalo da radi. Još jedna česta pojava je da, kada korisnik upadne u problem sa navigacijom, on dodatno pogorša situaciju time što će tražiti rešenje problema korišćenjem komandi koje nije pravilno razumeo i koje su ga upravo i dovele u problem.

Postoje različiti pristupi rešavanju ovih problema, na primer korišćenjem specijalno dizajniranih ikonica, vizuelnih referenci, zvučnih signala, ograničavanjem prostora u kome korisnik može da se kreće, ograničavanjem kretanja koje on može da napravi, promenom komandnog interfejsa u zavisnosti od pozicije korisnika u 3D svetu (uvodi se razlika između kretanja u enterijeru i eksterijeru). Često se korisnici vezuju za jednu ravan; ograničava se mogućnost *zoom*

funkcionalnosti jer se korisnici lako zbune ako prođu kroz referentnu tačku; takođe se često sprečava prolazak kroz površine objekata (sprečava se *clipping*) korišćenjem algoritama za detekciju sudara kamere sa površinama (*collision detection*).

Korišćenje miša se pokazalo kao najbolje rešenje za upravljanje sistemom za navigaciju u većini aplikacija koje ne podrazumevaju upotrebu opreme za virtuelni stvarnost (*VR – Virtual Reality*), tako da je ona praktično postala standard u 3D aplikacijama. Vrlo često se miš koristi u kombinaciji sa tastaturom čime se postiže veća produktivnost dok je usložnjavanje interakcije minimalno.

U modernim aplikacijama se tipično koriste *pan*, *zoom* i *orbit* operacije da bi se obezbedila navigacija korisnika u 3D sceni. Savladati korišćenje ovih operacija često zahteva velike napore tako da i iskusni korisnici računara mogu naići na probleme prilikom njihovog učenja. Dobar sistem za navigaciju mora ograničavati mogućnost greške korisnika, treba da mu pomaže da se ne izgubi u prostoru i treba da obezbedi korisniku što „bezbolniji“ period navikavanja na komande. Korisnik bi uvek trebalo da zna tačno gde se nalazi, kako je orijentisan u prostoru i koje akcije treba da preduzme kako bi prešao na sledeću lokaciju od interesa.

U ovom radu opisano je rešenje sistema za navigaciju koje je implementirano u VIVE (*Volumetric Image Visualisation Environment*) klijent aplikaciju. VIVE [2] projekat predstavlja interaktivno okruženje za analizu trodimenzionalnih medicinskih slika koje olakšava dijagnostiku, planiranje hirurških intervencija, procenu uspešnosti terapija, kao i medicinski pregled na daljinu.

2. OPIS REŠENJA

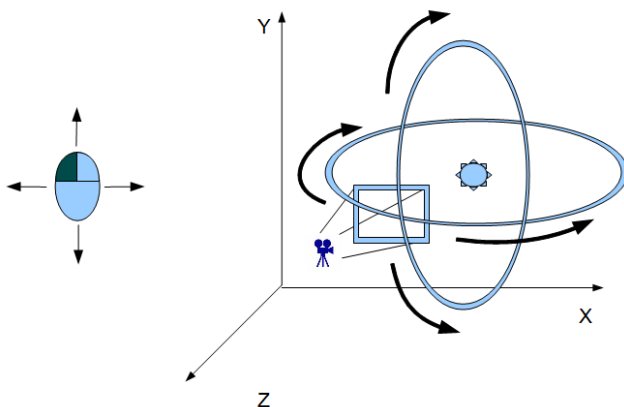
Bilo je važno napraviti sistem za navigaciju čije će ponašanje biti prilagođeno što lakšem pregledu objekta posmatranja. 3D scena se može sastojati iz više 3D objekata koji mogu biti od interesa korisniku, tako da je implementiran i mehanizam koji će korisniku dozvoliti da proizvoljno bira u odnosu na koji objekat će se vršiti navigacija. Sa tom idejom uvodimo pojam centra navigacije kao tačke u prostoru u odnosu na koju se vrši navigacija, odnosno, tačke u odnosu na koju će se korisnik orijentisati za vreme navigacije. Po pravilu, centar navigacije bi trebalo da se nalazi na objektu od interesa i da okvirno bude na pravcu gde korisnik „gleda“.

Sistem za navigaciju podeljen je na dva režima rada: režim navigacije i režim izbora centra navigacije. Režim izbora centra navigacije ima pomoćnu ulogu i služi za lakše biranje centra navigacije.

U daljem tekstu su opisane implementirane funkcionalnosti režima navigacije i režima izbora centra navigacije u VIVE aplikaciji.

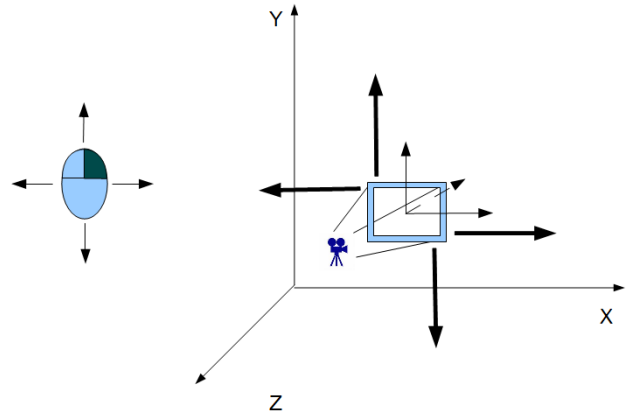
Približavanje (*zoom in*) i udaljšavanje (*zoom out*) od centra navigacije se može ostvariti na više načina. Dupli klik na levi taster miša će napraviti veliki skok unapred i približiti kameru centru navigacije dok će dupli klik na desni taster miša napraviti veliki skok unazad i udaljiti kameru od centra navigacije. Ukoliko miš ima *scroll* točkić, on se može koristiti za *zoom*-iranje tako što se točkić okreće unapred ili unazad u zavisnosti od toga da li korisnik želi da se približi ili udalji od centra navigacije. Takođe se može *zoom*-irati ako se pritisne srednje dugme miša, odnosno točkić, i miš se prevlači unapred ili unazad

Rotacija ili kretanje pogleda po orbiti oko centra navigacije (sl. 1. Rotacija pogledasl. 1) se postiže pritiskom na levi taster miša i njegovim prevlačenjem u bilo kom smeru (gore, dole, levo, desno) bez puštanja tastera. Kamera se tom prilikom kreće po zamišljenoj sferi, u čijoj se sredini nalazi centar navigacije, u smeru koji se poklapa sa smerom prevlačenja miša.



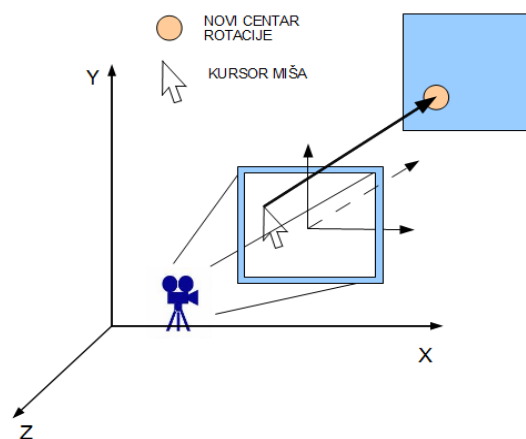
sl. 1. Rotacija pogleda

Pomeranje pogleda u ravni (sl. 2) se ostvaruje pritiskom na desni taster miša i njegovim prevlačenjem u bilo kom smeru, bez puštanja tastera. Kamera se kreće u ravni na čijoj se površini nalazi i pogled na scenu.



sl. 2. Pomeranje pogleda u ravni

Biranje novog centra navigacije (sl. 3) omogućava korisniku da preusmeri svoju pažnju na bilo koji objekat u 3D sceni ili i na određeni deo tog objekta. Novi centar navigacije se bira tako što se kursor miša postavi na željenu poziciju (tako da pokazuje na neki objekat) i zatim se klikne na levi taster miša u kombinaciji sa CTRL ili SHIFT tasterom. Ukoliko se izbor centra navigacije vrši u kombinaciji sa SHIFT tasterom, novi centar će se naći na površini objekta, tačno na mestu na koje je ukazivao kursor. U slučaju da se koristi kombinacija sa CTRL tasterom, novi centar navigacije će se postaviti u vizuelni centar izabranog objekta. Prilikom izbora novog centra navigacije vrši se repozicioniranje kamere tako da se novi centar navigacije može bolje videti. Ova funkcionalnost može dezorijentisati korisnika tako da je uvedena mogućnost izbora centra navigacije bez repozicioniranja pogleda, za šta je potrebno prilikom izbora pritisnuti taster ALT (zajedno sa CTRL ili SHIFT)



sl. 3. Izbor centra rotacije

Režim izbora centra navigacije ima samo jednu funkcionalnost, a to je biranje novog centra. Način izbora je sličan kao kod režima navigacije. Prvo se kursor miša postavi na željenu lokaciju, a zatim se klikne na levi taster miša. Ukoliko je kursor miša pokazivao na neki objekat na sceni,

novi centar navigacije će se postaviti na površinu tog objekta, i to tačno na mestu na koje je ukazivao kursor. Ukoliko je izbor centra navigacije izvršen sa pritisnutim CTRL tasterom na tastaturi, novi centar navigacije će se postaviti u vizuelni centar izabranog objekta. Ako prilikom izbora kursor miša nije pokazivao ni na jedan objekat, novi centar navigacije postavlja sam program tako što bira objekat koji je najbliži kameri i postavlja novi centar u njegovom vizuelnom centru. Nakon izbora centra navigacije sistem za navigaciju se automatski prebacuje u režim navigacije jer se podrazumeva da korisnik želi direktno da se vrati u režim navigacije.

Da bi se korisniku olakšala predstava o centru navigacije uveden je marker čija je uloga da pruža korisniku vizuelnu referencu o tome gde se trenutno isti nalazi. Kao marker centra navigacije uvedena je mala sfera čiji se centar nalazi u tački centra navigacije. Marker centra navigacije nije konstantne veličine već se prečnik sfere proporcionalno menja sa udaljenošću kamere od centra navigacije. Ovo prilikom korišćenja navigacije prouzrokuje da marker centra navigacije uvek bude prividno iste veličine.

Vidljivost markera centra navigacije je opciona u režimu navigacije dok se u režimu izbora centra navigacije marker uvek vidi. Korisniku je pružen izbor da li želi da vidi centar navigacije, jer njegova vidljivost neke korisnike može dekoncentrisati u toku rada.

3. IMPLEMENTACIJA

Rešenje navigacije je implementirano korišćenjem programskog jezika *Java* i *Java 3D API*-ja [3] (*Application Programming Interface*). Kombinacija ovih tehnologija omogućuje opisivanje 3D scena preko objekata. Cela 3D scena je uređena u graf scene (*scene graph*). Svaki objekat je opisan jedinstvenom granom sa čvorovima koji opisuju poziciju tog objekta u prostoru i zadnjim čvorom (listom grafa) koji opisuje pojavnost objekta (izgled, geometrija, specijalne osobine). Jedan od objekata 3D scene je i pogled na scenu (kamera), koja je opisana jedinstvenom granom u grafu scene.

Kao polazna tačka za izradu sistema za navigaciju uzeta je klasa *OrbitBehaviorInterim* [4], koja je već imala neke od potrebnih funkcionalnosti za navigaciju. Veliki deo rada sastojao se u prilagođavanju rada ove klase i *VIVE framework*-a, uz dodavanje novih funkcionalnosti ovoj klasi.

Navigacija se odvija u realnom vremenu tako da je i implementacija rešenja usklađena sa tim zahtevom. Navigacija se izvršava tako što se u svakom prikaznom ciklusu (*frame*) izračunavaju nova pozicija i orijentacija kamere. To se radi tako što se pre svakog prikaza prikupljaju informacije o svim događajima koji utiču na objekte u sceni, a onda se te informacije obrađuju i na osnovu njih se izračunava nova pozicija i orijentacija objekata i kamere.

Na osnovu pozicije posmatrača izračunava se udaljenost pogleda od centra navigacije. Naime, brzina navigacije zavisi od toga koliko je pogled udaljen od objekta posmatranja. Operacije *zoom* i pomeranje pogleda u ravni direktno zavise

od udaljenosti, tako da osetljivost ovih operacija na pokrete miša raste proporcionalno rastojanju. Udaljenost pogleda od centra navigacije takođe utiče na dimenzije markera centra navigacije – svaki put kada se promeni rastojanje izračunava se faktor skaliranja markera tako da je veličina markera proporcionalna rastojanju.

Prilikom izbora novog centra navigacije trenutni marker se briše (uklanja se njegova grana i korišćenjem vektora koji polazi od vrha kursora miša i ide kroz 3D scenu normalno u odnosu na ravan ekrana određuje se tačka preseka tog vektora i prvog 3D objekta kroz koji taj vektor prolazi. U toj tački se postavlja novi centar navigacije zajedno sa novim markerom centra navigacije.

U režimu za izbor centra navigacije događa se i da korisnik klikne mišem „u prazno“. U ovom slučaju program nema informaciju gde bi korisnik hteo da postavi novi centar navigacije tako da se vrši pokušaj pogađanja novog centra navigacije. Ovo se radi tako što se prostor u pravcu kamere ispituje primenom nevidljivog konusnog *Picker* objekta koji pruža *Java 3D*. Pronađeni objekat najbliži pogledu postaje objekat od interesa i novi centar navigacije se postavlja u vizuelni centar istog. Ukoliko na ovaj način nije pronađeno ni jedan objekat, centar navigacije se postavlja ispred posmatrača.

4. ZAKLJUČAK

Prostorna navigacija u odnosu na objekat posmatranja olakšava korisniku 3D aplikacije da usmeri svoju pažnju na objekat od interesa, kao i da se lakše kreće i orijentiše u prostoru. Korisnik može, po svom nahođenju, izabrati objekat na sceni koji ga najviše interesuje a sistem za navigaciju će prilagoditi svoje ponašanje tom izboru.

Ovakvo rešenje navigacije odgovara velikoj većini primena u naučnoj vizuelizaciji, kao i aplikacijama koje rade sa sporo promenljivim scenama. Primeri ovakvih primena su vizuelizacije volumetrijskih podataka sa medicinskih uređaja ili prostorno raspoređenih senzora, iz simulacija fizičkih procesa, kao i CAD (*Computer Aided Design*) aplikacije.

LITERATURA

- [1] *George Fitzmaurice, Justin Matejka, Igor Mordatch, Azam Khan, Gordon Kurtenbach*, Safe 3D Navigation. Autodesk Research, Ontario, Canada, <http://autodeskresearch.com/papers/safe3dnav/p7-fitzmaurice.pdf>
- [2] *Branko Marović*, „Distribuirano okruženje za analizu prostornih medicinskih slika“ Beograd 2001
- [3] *Daniel Selman*. Java3D programming, Manning Publications Co, 2002
- [4] *JCanvas3DNavigation Applet example*, <http://www.interactivemesh.org/testspace/orbitbehavior.htm>

MODEL PODATAKA ZA ODRŽAVANJE DOKUMENTACIJE SISTEMA KVALITETA U ZDRAVSTVENOJ USTANOVI THE DATA MODEL FOR MAINTAINING QUALITY MANAGEMENT SYSTEM DOCUMENTATION IN THE HEALTHCARE UNITS

Milovan Obradović, *Vojnomedicinska akademija, Beograd*

Sadržaj - Usvajanje sistema menadžmenta kvalitetom treba da bude strateška odluka zdravstvene ustanove. Da bi zdravstvena ustanova efektivno funkcionisala, ona mora da utvrdi i da obavlja menadžment brojnim povezanim aktivnostima. Aktivnosti koriste resurse i imaju upravljanje koje omogućava transformaciju ulaznih elemenata u izlazne. Dokumentacija sistema menadžmenta kvalitetom mora da sadrži politiku i ciljeve kvaliteta, poslovnik o kvalitetu, dokumentovane procedure i zapise i dr. koje je zdravstvena ustanova utvrdila kao neophodne da bi obezbedila efektivno planiranje, sprovođenje i upravljanje svojim procesima. Dokumentacijom koju zahteva sistem menadžmenta kvalitetom zdravstvene ustanove mora se upravljati. U ovom radu dat je model podataka za integrisani sistem kvaliteta zdravstvene ustanove kao osnove za izradu aplikativnog softvera za održavanje i upravljanje dokumentacijom uvedenog sistema kvaliteta.

Abstract - The adoption of the quality management system should be a strategic decision of the healthcare unit. For the healthcare unit to function effectively, it has to determine and manage numerous related activities. Activities use resources and their management enables the transformation of input into output elements. The quality management system documentation should include policies and quality objectives, a quality manual, documented procedures and records determined by the health units to be necessary to ensure the effective planning, operation and control of the processes. Documentation required by the quality management system of the healthcare units should be controlled. In this paper the data model for integrated quality management system of healthcare unit is presented as basis for developing application software for maintaining and managing the introduced quality management system documentation.

1. UVOD

Standard ISO 9001:2008 specificira zahteve za sistem menadžmenta kvalitetom u kome organizacija treba da pokaže svoju sposobnost da dosledno obezbedjuje proizvod ili uslugu koji ispunjava zahteve korisnika u duhu zakonskih i drugih propisa, sa ciljem da se poveća zadovoljstvo korisnika proizvodom ili uslugom i dokaže usaglašenost sa korisnikovim potrebama i odgovarajućim propisima.

Logika sistema kvaliteta bazira se na procesnom pristupu koji treba da obezbedi:

- niže troškove i kraće vreme za efektivnu realizaciju procesa
- poboljšane, konzistentne i predvidive rezultate
- identifikovane i po prioritetu složene mogućnosti za poboljšanja,

što se obezbedjuje kroz sistemski pristup menadžmentu sistema kvaliteta koji ima za cilj „identifikovanje, razumevanje i upravljanje sistemom međusobno povezanih procesa što doprinosi efikasnosti i efektivnosti organizacije u ostvarivanju svojih ciljeva“ [1].

Kada je zdravstvena ustanova (ZU) u pitanju njena obaveza je da uspostavi, dokumentuje, primenjuje i održava sistem menadžmenta kvalitetom i stalno poboljšava njegovu efektivnost u skladu sa zahtevima standarda ISO 9001. Uvodjenje i primena ovih standarda zdravstvenim ustanovama obezbedjuje:

- konkurentsku prednost u odnosu na druge zdravstvene ustanove
- sticanje poverenja drugih zdravstvenih ustanova na planu zadovoljavanja njihovih potreba
- sticanje poverenja korisnika zdravstvene zaštite da će njihovim zahtevima biti udovoljeno efikasno i efektivno.

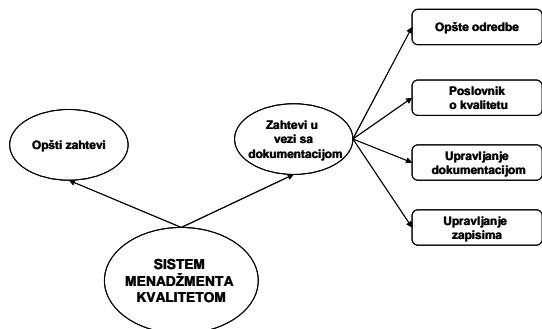
Polazeći od brzih promena u medicinskoj tehnologiji i sve većih doprinosa nauke savremenom lečenju, prirodna je potreba da zdravstvene ustanove stalno analiziraju zahteve korisnika, prate naučna i stručna dostignuća i uvode nove tehnologije koje su prihvatljive za korisnike zdravstvenih usluga. Kroz sistem menadžmenta kvalitetom se unapređuje zadovoljstvo korisnika i svi procesi u zdravstvenoj ustanovi drže pod kontrolom.

2. DEFINICIJE I HIJERARHIJA DOKUMENTA

Osnove sistema menadžmenta kvalitetom koje definišu predmet i područje primene i odgovarajuću terminologiju definisane su standardom ISO 9000:2005. U cilju precizne upotrebe osnovnih termina za dokumentaciju sistema kvaliteta dajemo kratke definicije nekih pojmova [2]:

- a. Politika kvaliteta je dokument koji predstavlja sveobuhvatne namere i način vođenja organizacije kako to zvanično izražava najviše rukovodstvo.
- b. Ciljevi kvaliteta su sve ono šta se traži ili želi postići u vezi sa kvalitetom. Ciljevi kvaliteta se, u principu, zasnivaju na politici kvaliteta i u obliku dokumenta preciziraju rezultate koje treba ostvariti u određenom vremenskom periodu (najčešće godinu dana).

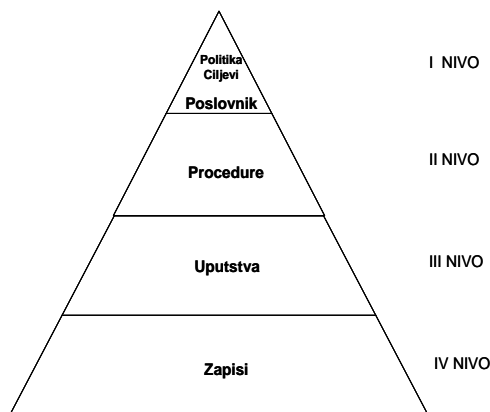
- c. Proces predstavlja skup međusobno povezanih ili delujućih aktivnosti koji pretvara ulazne elemente u izlazne.
- d. Poslovnik o kvalitetu je dokument kojim se definiše sistem menadžmenta kvalitetom organizacije.
- e. Procedura je dokument koji na utvrđeni način definiše obavljanje neke aktivnosti ili procesa.
- f. Uputstvo predstavlja dokument kojim se daju preporuke i sugestije ili detaljno opisuje neka aktivnost ili proces.
- g. Zapis je dokument kojim se iskazuju dobijeni rezultati ili daju dokazi o izvršenim aktivnostima. Istovremeno on pruža informacije o tome da li i koliko dobro funkcioniše dati sistem menadžmenta kvalitetom u organizaciji.



Slika 1. Pozicija dokumentacije u sistemu kvaliteta

Šta je i čemu služi dokumentacija sistema menadžmenta kvalitetom? Može se smatrati da je dokumentacija jedna vrsta vodiča za sve one koji je koriste i dokaz o ispunjavanju zahteva koje postavlja dati standard, čime se pruža potvrda da je organizacija sposobna da primenjuje zahteve datog sistema menadžmenta kvalitetom. Ona obezbeđuje da se unutar organizacije mogu pratiti svi bitni tokovi i preduzete mere. Ona mora biti razumljiva za sve zaposlene u organizaciji.

Pozicija dokumentacije u integrisanom sistemu kvaliteta i potreba za njenim održavanjem i upravljanjem najbolje je ilustrovana na Slici 1.



Slika 2. Hijerarhija dokumenta u sistemu kvaliteta

Obim dokumentacije sistema kvaliteta zavisi od složenosti organizacije, složenosti procesa i njihovih međusobnih uticaja, kompetentnosti zaposlenih. Između dokumenta sistema kvaliteta postoji hijerarhijski odnos, koji je predstavljen na Slici 2.

3. UVODJENJE INTEGRISANOG SISTEMA

Kada je zdravstvena ustanova u pitanju prirodno je da se uvedu svi standardi intergrisanog sistema kvaliteta (IMS - Integrated Management System), tj. standardi:

- sistem menadžmenta kvalitetom (Quality Management System - QMS), definisan serijom standarda ISO 9000,
- sistem menadžmenta zaštitom životne sredine (Environmental Management System - EMS), definisan serijom standarda ISO 14000,
- sistem menadžmenta zdravljem i bezbednošću na radu (Occupational health and safety management system – OH&S), definisan standardom OHSAS 18001,
- sistem menadžmenta bezbednošću hrane, definisan standardom ISO 22000 i analizu rizika i kritične kontrolne tačke (HACCP),
- za opšte zahteve za kompetentnost laboratorija za ispitivanje i laboratorija za etaloniranje, definisane standardom ISO/IEC 17025
- sistem menadžmenta za sigurnost informacija, definisan standardom ISO 27001.

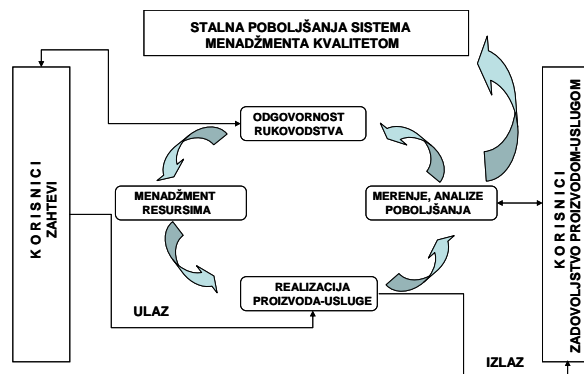
Standardom ISO 9001:2008 sugerise se primena procesnog pristupa [1] u razvoju, primeni i poboljšavanju efektivnosti sistema menadžmenta kvalitetom. Ovaj pristup insistira na:

- identifikaciji skupa procesa u sistemu
- shvatanju međuzavisnosti procesa
- povezivanju procesa sa definisanim ciljevima organizacije
- merenju rezultata u odnosu na postavljene ključne ciljeve.

Na sve identifikovane procese može se primeniti tzv. PDCA (Plan-Do-Check-Act – planirajte-izvršite-proverite-delujte) metodologija koja obuhvata aktivnosti:

- planiranja, kojima se utvrđuju ciljevi i uspostavljaju procesi za dobijanje rezultata u skladu sa zahtevima korisnika i politikom organizacije
- izvršenja, kojim se procesi odvijaju
- provere, kroz praćenje i merenje procesa i njihove komparacije sa definisanim ciljevima, politikom i zahtevima koje mora ispuniti proizvod ili usluga
- delovanja, kojima se preduzimaju akcije za poboljšanje performansi procesa.

Prema ISO 9001:2008 grafička predstava procesnog pristupa data je na Slici 3.



Slika 3. Model sistema menadžmenta kvalitetom

Primenom procesnog pristupa u ZU se mogu identifikovati tri grupe procesa u sistemu menadžmenta kvalitetom:

- 1) PROCESI MENADŽMENTA
- 2) PROCESI ZDRAVSTVENE ZAŠTITE
- 3) PROCESI PODRŠKE

Primenom procesnog pristupa takodje se utvrđuje redosled i međusobno delovanje procesa:

- strukturu sistema menadžmenta ZU i zahteve standarda na kojima počiva IMS
- opise u dokumentima IMS ZU (poslovnik o kvalitetu, poslovnik EHS, poslovnik FSMS, poslovnici o kvalitetu laboratorija, procedure i uputstva);
- opise u dokumentima potrebnim ZU da bi osiguralo efektivno planiranje i izvođenje procesa, kao i upravljanje tim procesima.

4. DOKUMENTACIJA SISTEMA KVALITETA U ZU

Dokumentacija sistema menadžmenta kvalitetom u organizaciji mora da sadrži sledeća dokumenta:

- politiku kvaliteta
- ciljeve kvaliteta
- poslovnik o kvalitetu
- dokumentovane procedure i zapise
- dokumenta i zapise koje je organizacija utvrdila kao neophodna za efektivno planiranje, sprovođenje i upravljanje procesima.

Obim dokumentacije zavisi od: tipa (delatnosti) i veličine organizacije, složenosti i međusobne povezanosti procesa, složenosti proizvoda – usluge, zahteva korisnika, propisa, osposobljenosti osoblja i minimalnih zahteva koje postavlja standard.

Zdravstvena ustanova u sistemu menadžmenta kvalitetom može imati veći ili manji broj procedura u zavisnosti od toga koliko su procesi složeni i kakvo je upravljanje za njih potrebno. Za zdravstvenu ustanovu tipa doma zdravlja, specijalističku polikliniku ili stacionarnu zdravstvenu ustanovu broj procedura može se u IMS kretati u intervalu 50 – 150. Postoje dve vrste procedura: obavezne zahtevane od standarda i ostale koje definiše ZU za efektivno planiranje, izvođenje svojih procesa i upravljanje njima.

Obavezne procedure su namenjene za: upravljanje dokumentima i zapisima, interne provere, upravljanje neusaglašenim proizvodom, korektivne mere i preventivne mere.

Koristi od dokumentacije kvaliteta u ZU:

- opisuje sistem menadžmenta i obezbeđuje povezivanje procesa i podataka
- edukuje i informiše zaposlene da bolje razumeju svoje mesto i ulogu u organizaciji
- informiše zaposlene o stavovima rukovodstva
- obezbeđuje uzajamno razumevanje između zaposlenih i rukovodstva
- određuje kako treba raditi da bi se postigao zadovoljavajući efekat

- stvara osnovu za obuku novozaposlenih i ponovnu obuku zaposlenih
- obezbeđuje uskladenost poslova
- stvara osnove za neprekidna poboljšanja
- obezbeđuje objektivne dokaze o ispunjenosti postavljenih zahteva
- stvara poverenje kod korisnika
- daje jasan okvir isporučiocima proizvoda
- pokazuje mogućnosti ZU zainteresovanim stranama.

Kod definisanja i izrade dokumentacije sistema kvaliteta u ZU treba naći meru kako sistem menadžmenta kvalitetom ne bi bio prekomerno dokumentovan a time i dosta samoograničen. Dokumentacija treba da bude „skrojena“ prema potrebama ZU. Polazna je pretpostavka da ZU ima dokazano kompetentne izvršioce za sve poslove koji se u njoj obavljaju.

5. MODEL PODATAKA ZA ODRŽAVANJE DOKUMENTACIJE IMS

Detaljnou analizom dokumenta integrisnog sistema kvaliteta identifikovana je njihova struktura koja u osnovi prati standard prema čijim se zahtevima izradjuje.

Shodno tome za poslovničke se može identifikovati sledeći sadržaj:

- identifikacioni elementi (oznaka, datum prvog izdanja, redni broj izdanja, datum važećeg izdanja)
- lica koja su ga izradila, pregledala i odobrila
- odgovornosti za uspostavljanje, primenu i održavanje procedure – nazivi radnih mesta
- definicije, oznake i skraćenice (za karakteristične pojmove i termine koji se koriste)
- dokumenta koja čine sistema kvaliteta
- ostala dokumenta (propisi, standardi)
- tekstualni opis
- podaci o promenama

Sadržaj dokumenta procedure je najkompleksniji i čine ga sledeći elementi:

- identifikacioni elementi (oznaka, datum prvog izdanja, redni broj izdanja, datum važećeg izdanja)
- lica koja su je izradila, pregledala i odobrila
- odgovornosti za uspostavljanje, primenu i održavanje
- područje primene (org. jedinice gde se primenjuje)
- definicije, oznake i skraćenice
- veze sa dokumentima sistema kvaliteta
- veze sa standardima, propisima, stručnom literaturom
- tekstualni opis (tok procesa, izvršioći, dokumenta, zapisi)
- dokumenta koja proizilaze iz procedure
- prilozi

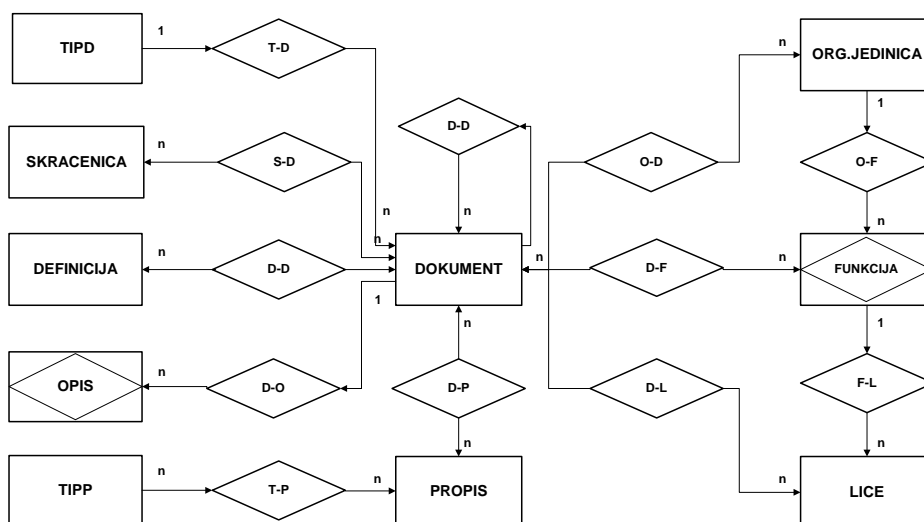
Sadržaj uputstava nije striktno definisanim standardima ali je pogodno da ima istu strukturu kao procedura.

Sadržaj zapisa koji se može javiti kao formalizovan (ima definisanu strukturu – obrazac) ili u slobodnoj formi (bez unapred definisane strukture) je sledeći:

- identifikacioni elementi (oznaka, naziv)
- period i mesto čuvanja popunjenog obrasca – zapisa
- dokument (procedure, uputstvo) iz kojeg proizilazi
- forma obrasca (samo za formalizovane zapise).

Radi lakšeg modeliranja podataka o dokumentima IMS u ZU i tumačenja stvorenog modela podataka neophodno je definisati sledeće objekte sistema:

LICE je zaposleno lice u ZU koje prema sistematizaciji poslova i zadataka izvršava obaveze iz delokruga radnog mesta. Kod uvođenja i izmena dokumenta u IMS kvaliteta može biti nosilac izrade – promena, verifikator ili lice koje je odobrilo upotrebu.



Slika 4. Model podataka za sistem kvaliteta ZU

PROPIS predstavlja zakon ili drugi normativni akt, standard, pravilnik, uputstvo proizvođača opreme, stručnu literaturu.

DEFINICIJA predstavlja precizno tumačenje pojma ili radnje radi jednoznačne primene ili razumevanja.

SKRACENICA predstavlja niz simbola za skraćeno prikazivanje određenih naziva (na matičnom ili stranom jeziku), postupaka i sl.

OPIS predstavlja otvoreni tekst kojim se definiše predmet dokumenta (po potrebi) i objašnjavaju postupci ili radnje.

TIPD predstavlja vrstu dokumenta koja može biti: poslovnik kvaliteta, politika kvaliteta, ciljevi kvaliteta, procedura, uputstvo, obrazac ili zapis.

TIPP predstavlja vrstu dokumenta stvorenog izvan integrisanog sistema kvaliteta koja može biti: zakon, propis, standard, uputstvo proizvođača, literatura i dr.

Analizom realnog sistema i polazeći od definicije objekata i informacionih zahteva može se modelirati realni sistem. Koristeći model entiteta i poveznika (Entity Relation Ship) [3,4] podsistem realnog sistema data je na Slici 4.

6. ZAKLJUČAK

Postojanje baze podataka za održavanje dokumentacije integrisanog sistema kvaliteta u zdravstvenoj ustanovi može biti celina za sebe u procesima podrške korišćenja i

FUNKCIJA predstavlja radno mesto definisano u sistematizaciji poslova i zadataka ZU.

DOKUMENT predstavlja jedan od dokumenta sistema kvaliteta koji ima svoj naziv, oznaku, status (aktivan-odobren, pasivan, u izradi, nacrt, za usvajanje), datum prvog izdanja, oznaku izdanja (redni broj) i datum važećeg izdanja ili promene.

ORG.JEDINICA je deo organizacijske strukture ZU.

održavanja. Pogodnija varijanta je da ona bude sastavni deo integralnog informacionog sistema zdravstvene ustanove jer bi se tada oslanjala na objekte sistema čije je održavanje definisano u potpunosti [5]. Radi se o objektima koji se odnose na organizacijsku strukturu, radna mesta i zaposlena lica. Treba primetiti da u izradi, uspostavljanju i održavanju dokumentacije sistema kvaliteta postoji veliki stepen sličnosti sa modeliranjem realnog sistema za potrebe uvođenja informacionog sistema za podršku njegovom funkcionisanju. Za organizacije koje su sistemski pristupale izgradnji integralnog informacionog sistema je olakšan pristup uspostavljanju sistema kvaliteta. U slučaju da ne postoji integralni informacioni sistem, dokumentacija integrisanog sistema kvaliteta predstavlja dobru i preciznu polaznu za modeliranje realnog sistema i implementaciju informacionog sistema.

LITERATURA

- [1] ISO 9001:2008, *Quality management systems – Requirements*, ISO/TC 176, London, 2008.
- [2] ISO 9000:2005, *Quality management systems – Fundamentals and vocabulary*, ISO/TC 176, London, 2005.
- [3] J.D. Ulman, *Principles of Database Systems*, Computer Science Press, 1980.
- [4] B.Lazarević i dr, *Projektovanje informacionih sistema 2. deo*. Naučna knjiga, Beograd: 1988.
- [5] J.H. Bemmell, M.A. Musen, *Handbook of Medical Informatics*, Springer, Heidelberg: 1997.

RUTIRANJE PODATAKA U BEŽIČNIM SENZORSKIM MREŽAMA SA ASIMETRIČNIM VEZAMA

ROUTING DATA IN WIRELESS SENSOR NETWORK WITH ASIMETRIC LINKS

Mirko R. Kosanović, *Visoka Tehnička škola strukovnih studija u Nišu*

Sadržaj – *Mogućnost beleženja velikog broja događaja, kao i laka primenljivost Bežičnih Senzorskih Mreža (BSM), omogućava njihovu primenu u velikom broju različitih aplikacija. Heterogena primena BSM-a omogućila je i razvoj velikog broja komunikacionih protokola za prenos podataka u njima. Međutim, zbog ograničenih resursa BSM, razvoj i primena efikasnog protokola predstavlja jako složen zadatak. Upotreba efikasnog protokola za rutiranje, koji generiše najkraći put od izvora do odredišta i obrnuto, omogućuje nam da poboljšamo ukupne karakteristike celokupne mreže štedeći mnoge resurse a naročito potrošnju električne energije. Ovaj rad predlaže jedan takav efikasan protokol rutiranja koji je primenljiv u BSM.*

Abstract – *The possibility of recording many events and easy applicability of Wireless Sensor Networks (WSN), enable their implementation in many different applications. On the other side, the heterogeneous nature of WSN enable development of many communication protocols for routing and transport data. However, because of limited resources of WSN, development and implementation of such efficient protocol is a complicated task. Using a more efficient routing algorithm, which generates shortest paths for information routing from source to destination nodes, and vice versa, we can improve the overall effectiveness of the entire network in terms of power efficiency. This paper presents an improvement on the implementation of information routing capabilities in WSNs.*

1. UVOD

Zadnjih godina svedoci smo velikog napretka na polju razvoja senzorskih tehnologija, koja je rezultirala razvojem malih a jeftinih inteligentnih senzora velike snage. Mogućnost samostalnog detektovanja nekog događaja, njegovog procesiranja kao i prosleđivanja dobijenih rezultata bežičnim putem drugim zainteresovanim subjektima daje ogromnu prednost nad svim ostalim oblicima registrovanja promena u nadgledanom regionu. Grupisanjem većeg broja ovih senzora, formiraju se *Bežične Senzorske Mreže (BSM)*, koje nam omogućuju: nadgledanje velikog prostora, primenu senzora u nepristupačnim oblastima, visoki stepen tolerancije na greške kao i mogućnost samostalnog konfigurisanja i organizovanja istih. Sve ove nabrojane prednosti omogućuju primenu BSM u gotovo neograničenom broju aplikacija kao što su: praćenje prirodnih parametara: temperatura, vlažnost vazduha i pritisak, nadgledanje objekata, praćenje određenih ciljeva, primena u inteligentnim zgradama, nadgledanje prirodnih katastrofa: tornada, vulkana, cunamija ili zemljotresa, praćenje vitalnih parametara ljudskog organizma kao i beleženje multimedijanih podataka kao što su zvuk i slika. Sa druge strane senzorski čvorovi (SČ) u BSM imaju mnoga ograničenja koja mogu da poremete i degradiraju rad i kvalitet informacija koje se prikupljaju. Neka od njih su: ograničenja u napajanju SČ, smanjeni memorijski i računarski kapaciteti, rad u otežanim uslovima, veliki broj smetnji u komunikacionom medijumu, dinamički promenljiva topologija mreže, neravnomerni raspored SČ-a kao i problemi vezani za bežičnu komunikaciju. Potrošnja električne energije, zbog limitirane količine kao i problema oko zamene energetskih izvora, sigurno predstavlja najveći problem kod BSM. Zato je kod ovih mreža primaran zadatak da se ona smanji na najmanji mogući nivo, kako bi se životni vek BSM što više produžio. Ako imamo u vidu činjenicu, da je u pogledu potrošnje električne energije, prenos jednog bita

ekvivalentan izvršavanju preko 1000 CPU naredbi [1], lako je zaključiti da je komunikacija najvažniji element na koji posebno treba obratiti pažnju kako bi ušteda električne energije bila najveća. Kako je priroda topologije BSM višeskokovita (*multihop*), postoje mnoge karakteristike ovih komunikacija koje ih bitno razlikuju od komunikacija u standardnim žičanim i bežičnim mrežama. Neke od njih su: globalna adresna šema, ograničeni resursi SČ-a, velika redundansa prikupljenih podataka, višestruka uloga svakog SČ-a, asimetrične veze SČ-a i td.

Jedan od glavnih problema, rutiranje ili usmeravanje podataka, je u žiži interesovanja koji je obrađen u ovom radu. Na početku rada, nakon uvoda u problem, dat je kratak pregled specifičnosti rutiranja podataka u BSM. U narednom poglavlju je dat pregled nekih dosadašnjih rešenja koja se nalaze u literaturi. Nakon toga dat je predlog novog rešenja koje je bazirano na dobro poznatom IP protokolu rutiranja: *Open Shortest Path First (OSPF)* [2] kao i navedene su osnovne njegove prednosti nad dosadašnjim rešenjima.

2. KARAKTERISTIKE RUTIRANJA U BSM

Osnovni zadatak protokola rutiranja je da izvrše efikasno usmeravanje prikupljenih podataka od izvorišta (SČ), do kontrolnog, odredišnog računara (*sink*) i obrnuto. Normalno, sve to treba da se izvrši u što kraćem vremenskom intervalu i uz minimalnu potrošnju elektr.energije kako bi maksimalno produžili radni vek naše mreže. To znači da moramo da vodimo računa o ravnomernoj potrošnji svih SČ, tj. globalno cele BSM, kako ne bi došlo do degradacije komunikacionih kanala a samim tim i prestanka rada cele mreže. Kako su dometi u BSM vrlo mali a gustina SČ velika, to se proces pronalaženja optimalnih ruta dosta komplikuje. Iz toga sledi da razvoj jednog takvog efikasnog protokola rutiranja koji vodi računa o potrošnji električne energije (*power aware routing protocol*) u BSM, zavisi od brojnih faktora koji u

mnogome mogu da utiču na njegovu efikasnost. U nastavku ukazaćemo na neke od njih [3] [4]:

Vrsta podataka (*Data reporting model*): Uglavnom, podaci koji se prikupljaju u BSM u mnogome zavise od vrste aplikacija koja upravlja BSM. Oni se mogu podeliti na sinhronu ili kontinualne podatke (*time-driven*), asihronu podatke (*event driven*), podaci na upit (*query driven*) i mešovite podatke.

Model prenosa podataka (*Transmission model*): Čvorovi u BSM su raspoređeni u višeskokovitoj (*multy-hop*) topologiji. Ako se tome doda da oni komuniciraju preko bežičnog medijuma jasno je da su tu prisutni mnogi problemi koji su karakteristični za ove medijume: fading, mešanje signala, veoma veliki stepen grešaka i td. Za razliku od žičanih puteva ovde je prisutan još jedan problem koji u mnogome komplikuje komunikaciju dva susedna SČ a to je asimetrična veza. To znači da su mnoge veze u ovim mrežama jednosmerne jer mi možemo da pošaljemo neki podatak prema nekom SČ, ali većina njih nije u stanju i da odgovori SČ od koga su primili taj podatak.

Dinamička topologija mreže (*Network dynamics*): Bez obzira što većina mrežnih aplikacija podrazumeva da su SČ statični, topologija BSM mreža se dinamički menja tokom vremena. Do promene dolazi zbog poremećaja u bežičnoj komunikaciji koji nastaje zbog prestanka rada nekih SČ, zbog smanjene emisione moći ili zbog nekih drugih problema u komunikaciji susednih SČ. Svi ovi problemi zahtevaju od protokola rutiranja da sasvim samostalno pronalaze nove rutiranje puteve u višeskokovitoj arhitekturi BSM a to nije ništa drugo nego da sve BSM tretiraju kao dinamičke mreže bez obzira na status SČ-a tj. na njihovu mobilnost.

Raspoređivanje SČ (*Deployment of SNs*): SČ u BSM mogu biti raspoređeni deterministički, tj. na tačno utvrđenim mestima ili , što je česti slučaj, nedeterministički tj. potpuno proizvoljno. Način na koji su SČ raspoređeni u mnogome utiče na performanse protokola rutiranja. Kako je raspored SČ u većini BSM potpuno nedeterministički, protokoli rutiranja imaju potrebu da budu potpuno samoorganizujući u pogledu pronalazanja optimalnih ruta, što znatno otežava i usložnjava njihov razvoj i primenu.

Energetska efikasnost (*Energy Efficiency*): SČ u BSM su jako mnogo ograničeni u pogledu potrošnje električne energije. Ako se uzme u obzir da oni imaju dvostruku ulogu, aktivnost većine SČ je jako velika. To se naročito odnosi na one SČ koji se nalaze u neposrednoj blizini glavnog SČ, *sink*-a. Pored osnovnog zadatka da sakupljaju i šalju podatke oni imaju i ulogu rutera jer moraju da prihvataju i prenose podatke od i ka susednim SČ. U takvim uslovima nefunkcionalnost nekog od SČ može da dovede do ozbiljnih problema u BSM, pa čak i do njihovog prestanka rada. Uloga protokola rutiranja je da bude jako efikasan u pogledu potrošnje el. energije, kako bi se izbegla nepotrebna slanja podataka a samim tim i povećanje potrošnje energije SČ.

Otpornost na greške (*Robust and Fault tolerant*): SČ predstavljaju jako male i slabo zaštićene uređaje koji se raspoređuju u prirodnim uslovima. Zato su oni podložni dejstvu prirodnih sila: sunce, kiša, vetar, velike temperaturne promene i različita zračenja, što sve može da utiče na njihov pravilan rad. Zbog svega ovoga SČ moraju da pokažu veoma visoki stepen tolerancije na moguće greške kako bi zadržali

funkcionalnost jedne BSM. Uloga protokola rutiranja je da se stalno prilagođavaju SČ u mreži kako bi maksimalno sačuvali potrošnju energije. To podrazumeva aktivno podešavanje snage prenosa podataka na pojedinim linkovima kao i pravovremeno usmeravanje podataka prema drugim SČ ili regionima koji imaju više el.energije u mreži od drugih SČ.

Skalabilnost (*Scalability*): Ukupan broj SČ u mreži jako varira i zavisi od aplikacije. On ide nekoliko stotina pa do nekoliko desetina hiljada SČ. Pored toga on jako varira tokom vremena jer mnogi SČ zbog nepovoljnih uslova u kojima oni rade prestaju se radom. Bilo koja šema rutiranja podataka mora biti sposobna da radi se različitim brojem SČ i ona ne sme da bude zavisna od njihovog broja. To znači da performanse efikasnog protokola rutiranja ne smeju da se menjaju u zavisnosti broja SČ.

Načini adresiranja (*Different Addressing Scheme*): Kod BSM razlikujemo tri načina adresiranja SČ:

1. Adresiranje prema adresi (*Address-Centring scheme*) – ovde svaki SČ ima svoju jedinstvenu adresu tj. jedinstven ID broj koji ima odlučujuću ulogu kod usmeravanja podataka. Protokoli rutiranja su kod ovakvog načina adresiranja veoma slični standardnim protokolima i poznati su još kao protokoli rutiranja koji svoje odluke zasnivaju na osnovu tabelarnih zapisa u svakom SČ [5].

2. Adresiranje prema tipu podatka (*Data-Centring scheme*) – ovde se komunikacija zasniva na *broadcast* porukama jer SČ nemaju svoje jedinstvene ID brojeve. Kod ovakvih protokola postoje dve vrste poruka: *Interest packet* koji predstavlja poruka koja se šalje kroz mrežu i nudi određenu vrstu informacije, i *Advertisement packet* koji predstavlja odgovor zainteresovanog SČ za taj tip informacije.

3. Adresiranje prema lokaciji (*Location-Centring scheme*) – ovd SČ koriste lokaciju kao primarnu informaciju za adresiranje i usmeravanje podataka. Svaki SČ ima prostornu adresu koja isključivo zavisi od fizičke lokacije samog SČ u posmatranom prostoru. Kod ovakvog adresiranja moguće je da više SČ na jednoj lokaciji imaju istu adresu.

Teško je reći koji od ova tri metoda najbolje rešava problem adresiranja SČ u BSM jer uglavnom on zavisi od tipa aplikacije koja upravlja BSM. Najbolje bi bilo da protokoli usmeravanja podržavaju sva tri načina adresiranja.

Grupisanje podataka (*Data aggregation*): Kako u BSM imamo nejednako raspoređene SČ, moguće je da veliki broj njih detektuje isti događaj. Tako se dešava da različiti SČ šalju iste podatke, što prouzrokuje nepotrebno zagušenje saobraćaja u BSM. Rešenje je pronađeno u grupisanju sličnih podataka koji su očitani sa različitih SČ. Tako smo smanjili broj transakcija jer prosleđujemo samo jedan podatak a ne više njih. Sličan postupak možemo ponoviti i u slučaju kada se pročitaju različiti podaci sa različitih SČ. Naime, svi ti podaci se sada mogu podvrgnuti nekoj statističkoj funkciji kao što je minimum, maksimum ili srednja vrednost, pa se sada dobijeni rezultat dalje prosleđuje nadređenim SČ [6].

Mali programski kod (*Simple algorithm with small program code*): SČ predstavljaju jako ograničene uređaje kako u procesorskoj snazi, tako i u količini memorije kojom raspolažu. Baš zbog toga programski kod koji podržava protokol rutiranja treba da bude što manji, kao i da zahteva veoma malo procesorskog vremena u pogledu izračunavanja efektivnih putanja usmeravanja. Sa druge strane, mali programski kod nam dozvoljava da taj protokol usmeravanja

implementiramo na različitim hardverskim platformama koje su jako jednostavne [6].

Kvalitet servisa (*Quality of service*): Stepen tačnosti informacija koje prikupljamo od SČ u mnogome zavise od vrste aplikacije. Kod tzv. *real time* aplikacija potrebno je da informaciju o nekom događaju dobijemo odmah tj. u određenom vremenskom intervalu. Međutim, kod ovih aplikacija potrošnja električne energije je jako velika, pa postoji opasnost da životni vek takvih BSM bude veoma kratak. Sa druge strane može da se desi i da je saobraćaj u BSM jako otežan, jer se javlja veliki broj ponovnih slanja-retransmisija. Zato je nekada potrebno produžiti njihov rad pa čak i na uštrb kvaliteta informacija koje se prikupljaju. Protokoli rutiranja moraju da vode računa o ovome, da se prilagođavaju trenutnoj situaciji u BSM i donose optimalna rešenja, sve sa ciljem da produže njihov životni vek.

3. USMERAVANJE PODATAKA NA OSNOVU KARAKTERISTIČNIH PODATAKA MREŽE

Kao što smo ranije napomenuli usmeravanje podataka u BSM predstavlja veoma izazovan zadatak jer se one po mnogim karakteristikama razlikuju od drugih kako žičanih tako i bežičnih mreža. Upravo zbog tih razlika razvijeni su mnogi novi algoritmi koji su na različite načine rešavali te probleme. Ono što uvek svaki mehanizam usmeravanja podataka mora da ima u vidu pri donošenju odluke to su: karakteristike SČ, tip aplikacije koja je primenjena kao i topologija BSM. Postoji veliki broj kriterijuma na osnovu koji možemo izvršiti podelu protokola rutiranja [7]. U ovom radu naša pažnja biće posvećena na protokolima rutiranja koji svoje odluke donose na osnovu stanja određenih parametara kako SČ tako i cele mreže (*stateful ad hoc routing protocols*). Informacije o ovim parametrima se prikupljaju duž cele rute-puta od izvornog čvor do odredišnog čvora i one se odnose na: električnu potrošnju svakog SČ, kvalitet prostiranja signala, opterećenost rute, domet signala i td. Postoji veliki broj protokola koji su razvijeni za tradicionalne *ad-hoc* bežične mreže kao što su: *Dynamic source routing* (DSR), *Ad hoc on-demand Distance Vector Routing* (AODV) i *Dynamic Destination-Sequenced Distance Vector Routing* (DSDV) [4]. Imajući u vidu da je broj podataka koji se šalju kroz BSM mnogo manji od broja stvarnih podataka koji se očitavaju sa neposrednog izvora podataka - senzora, mi smo se odlučili da u našem predlogu primenimo ideju DSDV protokola usmeravanja, koji smo kombinovali sa algoritmom pronalaženja najkraćeg puta (*shortest path algorithm*). Glavna prednost algoritma najkraćeg puta je ta da on na osnovu ranije definisanog kriterijuma, pronalazi optimalan put između odredišta i destinacije. Uopšteno gledano u BSM uvek postoji samo jedan optimalni put između bilo koja dva čvora u mreži za koji kažemo da je to najkraći put za prenos podataka. I suviše je govoriti o prednostima pronalaženja ovakvih ruta u BSM. Međutim, zbog prirode BSM, razvoj jednog takvog protokola predstavlja jako složen zadatak. Postoji mnogo stvari koje treba uzeti u razmatranje pre nego što pronađeni optimalni put profunkcioniše. Potrošnja električne enrgije, ograničeni memorijski resursi SČ kao i dinamičke karakteristike BSM još više otežavaju definisanje jednog efikasnog algoritma za pronalaženje optimalnih ruta. Sa druge strane treba obezbediti da to rešenje zadovolji i osnovne zahteve protokola rutiranja a to su: jednostavnost

primene, nezavisnost od hardverske platforme, skalabilnost i robustnost tj. veliku otpornost na moguće greške [8].

4. PREDLOŽENO REŠENJE

Metod rutiranja koji se predlaže u ovom radu pogodan je za aplikacije koje ne zahtevaju simetrične veze između susednih SČ-a. Naime, poznato je da je priroda BSM, kako sa gledišta topološkog razmeštaja SČ, tako i sa gledišta uspostavljanja i kvaliteta uzajamnih veza, jako heterogena. To znači da u BSM postoji jako veliki broj asimetričnih veza između SČ-a. Baš zbog toga, potrebno je da se prikažu osnovni uslovi koji su potrebni da budu ispunjeni sa gledišta BSM-a a od kojih smo mi pošli kod razvoja predloženog rešenja za rutiranje podataka.

Pretpostavka je da svaki SČ ima svoju jedinstvenu adresu kao i odgovarajuću težinu. Težina svakog SČ zavisi od mnogih parametara i ona se jedinstveno određuje na osnovu količine primljenih i poslatih bajtova, zauzetosti prolaznog bafera, preostale električne energije SČ kao i kvaliteta linka do susednih SČ-ova. Na kvalitet linka utiče intenzitet saobraćaja između SČ, kvalitet veze tj. broj retransmisija, broj skokova (broj angažovanih SČ u ostvarivanju veze) kao i širina kanala. Puno puta se pokazalo da sa gledišta uštede el.energije kod BSM, najoptimalniji put ne mora da bude onaj koji ima najmanji broj skokova. Ukoliko na tom linku imamo veliki broj retransmisija doći će do pojačanog saobraćaja a samim tim i do povećane potrošnje svakog SČ. Sledeća pretpostavka je da je BSM hijerarhijski organizovana u vidu klastera, što znači da je multi-hop karaktera. Razlikujemo dva smera podataka i to *downstream*-od master noda (MSN) ka senzorima i *upstream*-od senzora ka MSN. Razmatran je nejnepovoljniji slučaj, a to je da su ova dva pravca asimetrična tj. da postoji mogućnost da jedan SČ može da pošalje podatke drugom ali da od istog SČ ne može direktno da dobije podatke. Asimetrična veza je najčešće i realna slika odvijanja komunikacije u BSMs. Predloženi protokol koji se zasniva na metodu najkraćeg puta, sastoji se iz dva dela: inicijalizacioni deo i deo razmene podataka. Inicijalizacioni deo možemo podeliti u tri faze i to: faza inicijalizacije (jedan prema više), faza odgovora (više prema jedan) i faza definisanja prolaznih/susednih SČ(jedan prema jedan). U fazi inicijalizacije MSN šalje *broadcast* poruku (*downstream*) svim SČ.Ona se sastoji od: tipa poruke, adresnog dela SČ-ova i zbirnog težinskog dela. Svaki SČ koji primi inicijalizacionu poruku, dodaje svoju težinu u težinski deo dobijene poruke kao i svoju adresu u deo za adrese SČ i smesti je u bafer narednog slanja.Takođe aktivira tajmer pauze kako bi stopirao dalja slanja i omogućio da prihvati inicijalizacione poruke i od drugih SČ. Ukoliko primi poruku koja ima veću težinu od poruke koju je ranije primio on je odbacuje a ako primi poruku koja ima manju težinu on je zamenjuje sa prethodno upamćenom. Posle isteka tajmera pauze najoptimalnija poruka se prosleđuje dalje susednim SČ i postupak se ponavlja. Istovremeno aktivira se novi tajmer povratnog odgovora koji treba da spreči da SČ tu poruku prosledi nazad do MSN i obavesti ga o optimalnoj putanji do njega. Ta će se poruka proslediti sa povratnom porukom u *upstream* toku saobraćaja u fazi odgovora. Veličina tajmera obrnuto je proporcionalna broju skokova tj. što je broj skokova manji tajmer je duži i obrnuto. Ukoliko se neki SČ ne nađe na nijednom *upstream* putu onda će poruka biti prosleđena nakon isteka tajmera povratnog odgovora. Prosleđivanjem

downstream poruke svaki SČ dobija informaciju od kog čvora je primio najoptimalniju poruku, kao i optimalnu težinu puta kojim je ta poruka prošla. Postupak se ponavlja sve dok svi SČ ne prime poruke sa najmanjom težinom, kada prestaje aktivnost na mreži tj. nema više slanja. Svaki SČ sada u svom baferu slanja ima sliku najoptimalnijeg puta kojim je poruka od MSN stigla do njega tj. u *downstream* smeru. Nakon toga nastupa faza odgovora, gde sada svaki od SČ koji su zadnji u ostvarenom linku šalju informaciju MSN (*upstream*). Detekcija krajnjih SČ u linku omogućena je zahvaljujući isteku tajmera povratnog odgovora koji je kod ovih SČ najkraći. Postupak se sada ponavlja samo što sada kod *upstream* toka saobraćaja imamo proširenu poruku sa adresama SČ kroz koje prolazi *downstream* poruka. Ta proširena poruka se sastoji od dva dela i to: 1.) *upstream* dela poruke koji je gotovo isti kao i *downstream* poruka, samo u obrnutom smeru i sastoji se od tipa poruke, adresa SČ u *upstream* toku i njihovih ukupnih težina i 2.) *downstream* dela poruke koja predstavlja upamćenu najoptimalniju putanju u *downstream* smeru. Postupak se opet ponavlja kao i kod *downstream* poruke. Svaki SČ kada primi poruku pogleda težinski deo i ako je veći od težinskog dela u prethodno poslatoj poruci, pod uslovom da je adresa *soursa* ista, zaustavlja dalje slanje. Suprotno ako je ta vrednost manja šalje je dalje. Pošto se radi o vezi više prema jedan u *upstream* toku saobraćaja može doći do dva problema koje treba rešiti. Prvi je ako se pri povratnom saobraćaju neki od SČ ne nađu na nekom od povratnih linkova. To je rešeno putem tajmera povratnog odgovora. Kada on istekne, a nijedna poruka nije primljena, tada se taj SČ smatra kao krajnji i ponavlja se već opisana procedura. Drugi problem je ako do nekog SČ dođe poruka a u *downstream* delu poruke ne postoji adresa tog SČ. To nam govori da se radi o asimetričnoj vezi pa je potrebno da taj SČ opet generiše novu početnu poruku kao što to radi krajnji SČ u linku i na taj način obavesti o *downstream* putanji do njega. Na taj način MSN dobija informacije najoptimalnijih puteva koji se odnose i za *downstream* i za *upstream* tok saobraćaja. Nakon što dobije povratne poruke od svih SČ nastupa treća faza-faza definisanja komunikacije između susednih SČ-ova. MSN sada raspolaze sa svim potrebnim informacijama koje se odnose na broj SČ sa kojima može da komunicira, broj *layera* kao i ukupan broj SČ po *layeru*, kao i optimalne puteve do svakog SČ. Na osnovu svih sakupljenih informacija MSN svakom SČ šalje informaciju o najoptimalnijem susedu kome taj SČ treba da prosledi podatke u oba smera i *downstream* i *upstream*. Sada svaki SČ može da formira tabele rutiranja koje u zavisnosti od odredišnih čvorova, kojima je upućena poruka, ima odgovarajući susedni optimalni SČ kome se poruka prosleđuje. U tabele se upisuju samo oni SČ koji su se pojavili na rutama kod *downstream* i *upstream* inicijalizacione faze. U zavisnosti od stepena sigurnosti BSM, može se formirati prioriteta lista koja bi u sebi sadržala više susednih SČ preko kojih se može doći do odredišnog čvora. Ta informacija može da posluži i kao podrška za transportne protokole koji podržavaju prenos podataka preko višestrukih puteva. Algoritam rutiranja sada može na osnovu broja poruka koje se proslede preko određenih SČ, izvrši izbor različitih susednih SČ. Kada iskoristi sve registrovane susedne SČ, tada se šalje poruka MSN i zahteva se od njega da pokrene ponovo fazu inicijalizacije u utvrdi novo stanje težina svakog SČ u BSM. Takođe, ukoliko se desi da na izabranom optimalnom putu neki SČ otkáže, MSN obavestava

odgovarajući SČ da je potrebno da izabere alternativni susedni SČ iz prioriteta liste. U određenim vremenskim intervalima MSN proziva svaki SČ kako bi mogao da proveri njihovu aktivnost kao i da dobije njihove težine i na da osnovu tih podataka ažurira optimalne puteve, ako je to potrebno.

Najveći deo vremena SČ-ovi provode u usmeravanju podataka u fazi razmene podataka. To je faza kada SČ šalju očitane podatke MSN. Zahvaljujući formiranim tabelama u svakom SČ taj posao je znatno ubrzan i pojednostavljen. Svaki SČ u mreži tačno zna kom susednom SČ treba da pošalje/prosledi podatke i to kako u *downstream* toku tako i u *upstream* toku saobraćaja. Na taj način intenzitet saobraćaja sveden je na najminimalniju meru. Ukoliko neki SČ primi podatak koji nije njemu namenjen on ga jednostavno odbacuje i sprečava njegovo dalje širenje. U svakom prenosu podataka učestvuje optimalan broj SČ-ova, koji su unapred definisani. Na taj način minimizirali smo potrošnju elektr.energije a samim tim znatno produžili život BSM-a.

5. ZAKLJUČAK

Put do jednog efikasnog protokola usmeravanja je jako dug i složen zadatak. Jako složena priroda, kao i nepredvidljivi uslovi u kojima rade BSM, još više otežavaju i komplikuju razvoj i primenu ovakvog protokola. Očigledno je da primena standardnih protokola usmeravanja nije moguća jer se oni po mnogim karakteristikama međusobno razlikuju. Baš zbog toga, usmeravanje podataka u BSM je jako interesantno za mnoge istraživače i trenutno predstavlja jedno od najistraživanijih područja u BSM. Ovim radom dat je jedan predlog novog protokola usmeravanja koji bi trebalo da reši mnoge probleme koji su takođe navedeni ovde. Dalji rad na predloženom protokolu zahtevaće i njegovo testiranje u stvarnim realnim uslovima.

LITERATURA

- [1] H.Karl, A.Willing, "Protocols and Architectures for Wireless Sensor Networks", John Wiley & Sons, 2005
- [2] D.Medhi, K.Ramasamy "Network Routing, Algorithms, Protocols, and Architectures", by Elsevier Inc, Morgan Kaufmann Publishers, 2007
- [3] C.Yin, A.Orooji, "Routing Protocols for WSN", www.ucmss.com/books/LFS/CSREA2006/ICW3827.pdf
- [4] J.N.Al-Karaki, A.E.Kamal, "Routing Techniques in WSN: A Survey", www.ee.iastate.edu/~kamal/Docs/kk04.pdf
- [5] K.Akkaya, M.Younis, "A Survey on Routing Protocols for Wireless Sensor Networks", www.cs.umbc.edu/~kemal1/mypapers/Akkaya_Younis_JoAdHocRevised.pdf
- [6] J.Ibriq, I.Mahgoub, "Cluster-Based Routing in Wireless Sensor Networks : Issues and Challenges", <http://portal.acm.org/citation.cfm?id=1185404>
- [7] M.Kosanovic, "Specifičnosti rutiranja podataka u BSM", INFOTEH 2009, Jahorina
- [8] A.Kolioussis, J.Sventek, "Proactive vs reactive routing for wireless sensor networks", <http://www.dcs.gla.ac.uk/publications/PAPERS/9130/Kolioussis.pdf>

ALGORITAM ZA IZRAČUNAVANJE POZADINE BACKGROUND CALCULATION ALGORITHM

Nebojša Nedić, Elektrotehnički fakultet Podgorica

Sadržaj – U oblasti „Computer Vision-a“ često je potrebno izračunati pozadinu neke scene. Algoritam za izračunavanje pozadine koristi niz frejmova kao ulazni signal, dok se na izlazu dobija frejm pozadine. Algoritam sadrži određeni broj parametara koji definišu razne vrijednosti algoritma, kao što su vrijeme koje je potrebno da se pozadina izračuna, stabilnost pozadine, osjetljivost algoritma itd. Količina memorije koju algoritam koristi zavisi od podešavanja algoritma, veličine scene u pikselima i sl, ali ne i od vremena koje se odredi da bude referenca za izračunavanje pozadine.

Abstract – In Computer Vision discipline very often it is necessary to calculate background of some scene. As an input Algorithm for Background Calculation is using array of frames, while as a result it returns background frame. Algorithm have certain number of parameters that are used to define different values of algorithm, like time needed to calculate background, stability of the background, sensitivity of the algorithm etc. Quantity of memory used by the algorithm depends on parameters setup, size of the scene in pixels etc, but it is not dependant on time that is defined as a reference for background calculation.

1. UVOD

Algoritam se koristi za izračunavanje pozadine scene, a na osnovu niza frejmova, tj., ulaznog video signala. Bez obzira na scenu, algoritam izračunava frejm pozadine. Prilagođavanje rada algoritma različitim scenama vrši se pomoću niza parametara koji su definisani. Da bi se dobio bolji osjećaj kako algoritam praktično funkcioniše dat je i primjer aplikacije koja koristi ovaj algoritam.

Aplikacija je naspiana koristeći Borland Delphi 7 programsko okruženje. Bez obzira na tu činjenicu, aplikacija i algoritam su generalno primjenjivi jer ne koriste specifičnosti ovog razvojnog okruženja, osim u određenim, sporednim djelovima. Komentari i source code su napisani na engleskom jeziku. Više o ovome u nastavku teksta.

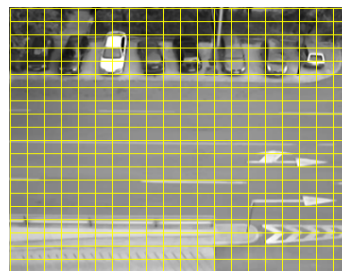
Prilikom izračunavanja frejma pozadine potrebno je obratiti pažnju na nekoliko detalja. Prije svega, potrebno je sagledati okruženje za koje se pozadina izračunava. Da li se radi o brzo promjenljivoj sceni, da li se objekti na sceni na neki način ponavljaju (primjena na pokretnoj traci u nekoj fabrici), koliko daleko je kamera postavljena od objekata koje je potrebno otkloniti i sl.

Jedan od važnijih parametara predstavlja vrijeme koje je potrebno da prođe da bi se pozadina izračunala. Da bi bilo jednostavnije ispratiti način rada algoritma, uzećemo primjer primjene u saobraćaju.

Iz razloga jednostavnijeg praćenja nastavka teksta potrebno je poznavati osnovna pravila i principe programiranja i način funkcionisanja digitalne fotografije i video signala.

2. PRINCIP RADA

Princip rada algoritma zasniva se na brojanju ponavljanja određenog dijela scene u vremenu.

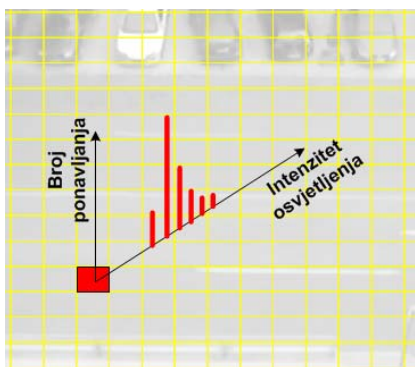


Sl. 1.

Na slici 1. dat je primjer jedne scene iz saobraćaja. Na početku, scenu je potrebno podijeliti na određen broj blokova. Veličinu bloka¹ potrebno je podesiti u zavisnosti od određenih parametara scene. Kasnije, u zavisnosti od veličine bloka, potrebno je podesiti vrijednosti i za neke druge parametre koji uglavnom upravljaju osjetljivošću algoritma.

Nakon što je scena podijeljena na blokove, potrebno je pratiti sve varijacije jednog bloka u vremenu. Postavlja se pitanje kako algoritam može da zna da li se radi o istom bloku u dva susjedna frejma. Postoji više načina kako se to može otkriti. Najjednostavniji je da se sabere intenzitet osvijetljenja svih piksela u okviru jednog bloka i da se odredi pravilo da svi blokovi istog intenziteta osvijetljenja predstavljaju i isti blok. Očigledno je da se ovdje pravi određena greška jer se ne uzima u obzir tekstura bloka. Kako se to u praksi pokazalo, ovako gruba aproksimacija ima smisla, a po potrebi, smanjenjem veličine bloka, možemo umanjiti uticaj teksture.

¹ Veličina bloka predstavlja jedan od parametara algoritma.



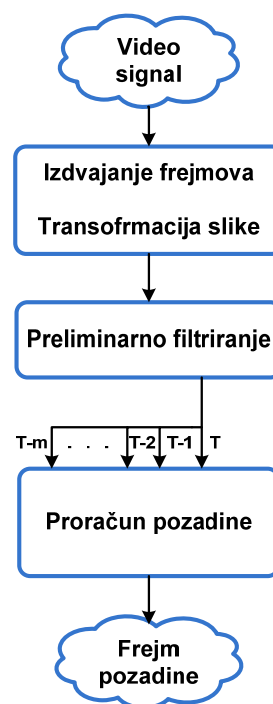
Sl. 2.

Za svaki blok sa scene potrebno je napraviti histogram koji će nam dati informaciju o broju ponavljanja u zavisnosti od intenziteta osvjetljenja. Na slici 2. grafički je predstavljeno kako bi to trebalo da izgleda za jedan odabrani blok. Intenzitet osvjetljenja koji se najviše ponavlja predstavlja najveću vjerovatnću da se radi o pozadini. Ako ovaj princip primijenimo na sve blokove na sceni, pozadinu možemo dobiti tako što ćemo odabrati blokove sa onim osvjetljenjem koje se najviše ponavljalo na sceni u vremenu.

Da li je dovoljno samo odabrati onaj blok sa najviše ponavljanja i zaključiti da je pozadina nađena? U principu nije. Da bi sa sigurnošću mogli reći da je pozadina pronađena, razlika između bloka sa najvećim brojem ponavljanja određenog intenziteta osvjetljenja i ostalih (odnosno drugog najvećeg - a samim tim i svim ostalim), mora biti dovoljno velika. Algoritam sadrži parametar koji određuje procentualnu razliku između prvog najvećeg broja ponavljanja i drugog koja mora biti ispunjena prije nego se zaključiti da je pozadina pronađena.

Dakle, pozadina je pronađena ako blok sa određenim intenzitetom ima $x\%$ više ponavljanja od drugog po redu. Na prvi pogled, dovoljan uslov. Međutim, ako posmatramo početak rada algoritma, svi blokovi određenog intenziteta će imati broj ponavljanja 1, dok će svi ostali imati vrijednost 0. To bi značilo da je pozadina odmah pronađena. Suština je da kompletan početni prelazni proces treba eksplicitno označiti i definisati da pozadina nije pronađena. Dužina ovog prelaznog procesa takođe se definiše ulaznim parametrom algoritma.

Sljedeći problem koji će se javiti u radu algoritma jeste i činjenica da se pozadina mijenja, ili bolje reći, sporo mijenja. Npr., „sporo“ kretanje sjenke koja je nastala od sunca, noć, jutro, snijeg koji pokriva okolinu i ona postaje bijela... Da bi algoritam uspješno funkcionisao, moramo uzeti u obzir i ove okolnosti. Ako bi neograničeno povećavali broj ponavljanja intenziteta osvjetljenja, došli bi u situaciju da je algoritmu potrebno previše vremena da detektuje ove „spore“ promjene. Kao rješenje ovog problema uveden je parametar koji definiše umanjene svih vrijednosti histograma za 50% onog trenutka kada najveća vrijednost dostigne određen broj ponavljanja. Na taj način, kompletan sistem će oscilovati između nekih zadatih vrijednosti. Maksimalna vrijednost ponavljanja je takođe definisana kao parametar sistema.



Sl. 3.

Na slici 3. prikazana je blok šema aplikacije koja je data kao propratni primjer. Na ulazu sistema imamo video signal koji se u prvom kolu odmah transformiše u oblik pogodan za korišćenje dalje u kodu. Nakon toga, frejm se filtrira pomoću median filtera i takava slika se proslijeđuje kolu za izračunavanje pozadine. Pozadina se računa na osnovu m ulaznih frejmova, što se podešava parametrima algoritma. Kao rezultat dobijamo frejm pozadine koji se kasnije može koristiti po potrebi.

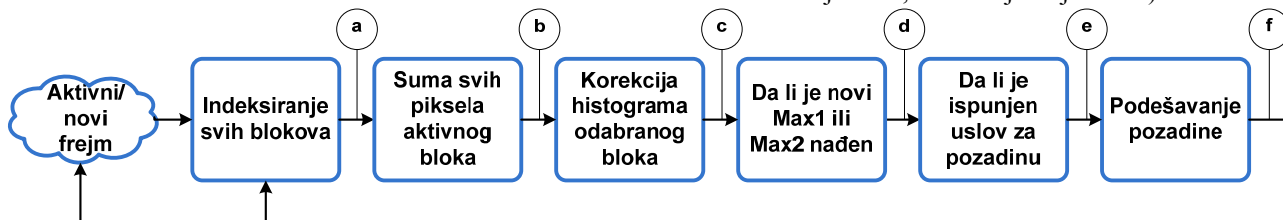
3. NAČIN FUNKCIONISANJA ALGORITMA

U daljem tekstu pogledaćemo kako algoritam funkcioniše. Da bi bilo jednostavnije shvatiti način rada, koristićemo konkretan primjer već pomenute aplikacije.

Na slici 4. data je blok šema algoritma. Na ulazu algoritma dobijamo tekući frejm koji se transformiše u oblik koji je pogodan za korišćenje. Više o ovome može se naći u izvornom kodu. Za tekući frejm već su definisani veličina i broj blokova, pa ih je sada sve pojedinačno potrebno indeksirati. Dakle, u tački „a“ dobijamo blok tekućeg frejma za koji je potrebno uraditi analizu.

Kako smo rekli ranije u tekstu, na početku je potrebno izračunati ukupan intenzitet osvjetljenja datog bloka. To se radi tako što se sumiraju sve vrijednosti piksela koji čine odabrani blok. Tako dobijen intenzitet osvjetljenja dijeli se sa vrijednošću parametra koji podešava osjetljivost algoritma (naziv promjenjive u kodu je „Sensitivity“). Za tako dobijenu vrijednost osvjetljenja uvećava se broj ponavljanja, tj., vrši se uvećanje za 1 indeksa histograma koji odgovara vrijednosti ovako dobijenog osvjetljenja. U tački „c“ na slici 4. imamo histogram koji je u skladu sa trenutnim stanjem. Nakon toga vrši se provjera da li se desila promjena u najvećem broju ponavljanja, ili u drugom najvećem broju ponavljanja.

Sad kada imamo i indexe najvećeg broja ponavljanja i drugog najvećeg broja ponavljanja, vrši se provjera da li trenutno indeksirani blok ispunjava uslov da je dio pozadine. Po potrebi, frejm pozadine se mijenja, tj., upisuje se novo detektovani blok pozadine i indeksira se idući blok. U slučaju da je cijeli proces završen za tekući frejm, uzima se idući frejm i počinje se od početka.



Sl. 4.

Sada ćemo objasniti ulogu nekoliko promjenjivih koje su potrebne da bi algoritam funkcionisao.

```

HistogramValue : array of array of array of Word;
Background1 : array of array of SmallInt;
Background2 : array of array of SmallInt;
BackgroundFoundArray : array of array of Boolean;
  
```

Promjenjiva *HistogramValue* predstavlja trodimenzionalni niz koji čuva histograme svih definisanih blokova. Prve dvije dimenzije imaju onoliko elemenata koliko postoji blokova horizontalno i vertikalno, dok treća dimenzija sadrži onoliko elemenata koliko može postojati intenziteta osvjetljenja pojedinog bloka. Nizovi *Background1* i *Background2* čuvaju indekse najvećeg broja ponavljanja i drugog najvećeg broja ponavljanja za svaki blok na sceni. Ova dva niza su potrebna da bi ubrzali rad algoritma. Niz *BackgroundFoundArray* za svaki blok prati da li predstavlja pozadinu.

Inicijalna podešavanja pomenutih nizova rade se po sljedećem pravilu. Sve vrijednosti niza *HistogramValue* postavljaju se na nulu. To praktično znači da nije došlo do bilo koje promjene. Elementi nizova *Background1* i *Background2* postavljaju se na vrijednost -1. Ovdje ne možemo koristiti vrijednost 0 jer postoji element histograma sa indeksom 0. Svi elementi niza *BackgroundFoundArray* postavljani su na vrijednost false čime označavamo sve blokove da nisu pozadina. Pored spomenutih nizova, postoje sljedeće promjenjive:

```

//When histogram reach this value, recheck is done
MaximumBckg : Word;
//Percentage between the biggest and second biggest
Max1Max2 : Word;
//Tolerance area between Max1 and Max2 hist. value
Max1Max2Tolerance : Word;
//Overall - is background found?
BackgroundFound : Boolean;
//Number of remaining blocks for backg. to be found
RemainingBlocks : Integer;
//Min. no. of frames to pass before checking is
//background found
MinNoOfFramesToPass : Integer;
//Sensitivity of algorithm
Sensitivity : Word;
  
```

Komentari koji opisuju svrhu određene promjenjive dati su u izvornom kodu. Takođe, za veći dio koda postoje komentari i objašnjenja koja ovdje nećemo pominjati da bi bili što je moguće efikasniji u pogledu količine teksta.

Primjena algoritma u aplikaciji sastoji se iz tri dijela, ne računajući ostale djelove aplikacije (median filter, inicijalna transformacija slike, učitavanje frejmova...).

Prvi dio čini procedura za proračun pozadine (*procedure Histogram*), zatim procedura koja vrši korekciju brojanja, odnosno vrijednosti histograma (*procedure BackgroundCorrection*) i na kraju procedura koja provjerava da li je pronađena pozadina (*procedure CheckIfBackgroundIsFound*). Ove tri procedure izvršavaju se upravo i ovim redoslijedom.

Procedura „Histogram“ izračunava histograme za sve definisane blokove frejmova sa ulaza. Na početku, procedura izračunava osvjetljenje bloka, zatim povećava vrijednost histograma u skladu sa izračunatom vrijednošću osvjetljenja i nakon toga provjerava da li je došlo do promjene u maksimumu 1 ili 2. U slučaju da je do promjene došlo, vrijednosti na koje pokazuju nizovi *Background1* i *Background2* odgovarajućeg bloka mijenjaju se kako bi indeksirali najveći broj ponavljanja i drugi najveći.

Nakon što se izvrši podešavanje *Background1* i *Background2* nizova vrši se kontrola da li je razlika između vrijednosti histograma na koji pokazuje element niza *Background1* i vrijednosti histograma na koji pokazuje element niza *Background2* dovoljno velika kako bi ispunila uslov potrebne procentualne razlike. U slučaju da je vrijednost histograma *Background1* veća za definisanu procentualnu vrijednost (promjenjiva *Max1Max2*) od vrijednosti *Background2* taj blok se problašava pozadinom i kopira se u frejm pozadine. U suprotnom, trenutni blok ne predstavlja pozadinu i on se označava kao takav.

Implementacija algoritma i ove procedure može se pogledati u prapratnom primjeru.

Kada procedura „Histogram“ odradi podešavanje vrijednosti histograma svih blokova (niz *HistogramValue*), po potrebi izvrši podešavanje indeksa *Background1* i *Background2*, a zatim kopira piksele tekućeg frejma odabranog bloka u frejm pozadine u slučaju da taj blok predstavlja pozadinu, startuje se procedura „BackgroundCorrection“. Svrha ove procedure je da sve vrijednosti histograma umanjuje za 50% u slučaju da je minimum jedna od njih dostigla vrijednost definisanu u promjenljivoj *MaximumBckg*. Na ovaj način moguće je podesiti koliko brzo će algoritam pratiti promjene na pozadini, kao što je ranije pomenuto u tekstu.

Treći dio algoritma predstavlja procedura za provjeru da li je pozadina detektovana (procedure `CheckIfBackgroundIsFound`). Nakon što se predhodne dvije procedure izvrše, ova procedura provjerava da li je broj blokova za koji još uvijek nemamo stabilnu pozadinu ispod definisanog limita. Kao rezultat, promjenjiva `BackgroundFound` dobiće vrijednost `true` ili `false`.

4. ZAKLJUČAK

Ovaj relativno jednostavan algoritam na uspješan način izračunava pozadinu scene čija se pozadina ne mijenja previše brzo. Da bi efikasnije sagledali rad algoritma u težim uslovima koji vladaju na sceni ili sa scenama koje imaju brže promjenjive pozadine, potrebno je izvršiti dodatna testiranja i izvesti potrebne zaključke.

Nastavak razvoja algoritma za proračun pozadine odnosio bi se više na proces optimizacije brzine rada i dodatnog testiranja, a samim tim i pronalaženju optimalnih vrijednosti za sve ulazne parametre, od kojih su neki ovdje spomenuti.

LITERATURA

- [1] O. Javed, K. Shafique and M. Shah. "A Hierarchical Approach to Robust Background Subtraction using Color and Gradient Information." Proceedings of the Workshop on Motion and Video Computing (MOTION'2002).
- [2] S. J. McKenna. "Tracking Groups of People." Computer Vision and Image Understanding, 2000.
- [3] E. Rivlin, M. Rudzskv, R. Goldenberg, U. Bogomolov, and S. Lepchev. "A Real-Time System for Classification of Moving Objects". IEEE Trans. 1051-4651/02 2002
- [4] C. Stauffer and W E. L. Crimson, "Adaptive background mixture models for real-time tracking". In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 246-252, Fort Collins, Colorado, 1999.

PROSTI MIKROKONTROLESKI BLIZINSKI KAPACITIVNI SENZOR SA EKRANIZIRANOM SENZORSKOM POVRŠINOM THE SIMPLE MICROCONTROLLER BASED PROXIMITY CAPACITIVE SENSOR WITH SCREEN BEHIND SENSOR AREA

Nedjeljko Lekić, Zoran Mijanović, Radovan Stojanović, Elektrotehnički fakultet Podgorica

Sadržaj- U ovom radu je predložen jednostavni mikrokontrolerski kapacitivni senzor. Oscilator senzora realizovan je upotrebom internog komparatora mikrokontrolera sa svega 4 eksterna otpornika. Zaštita od spoljašnjeg preopterećenja postignuta je detekcijom kolektorskog napona. Povećanje osjetljivosti i daljine detekcije postignuto je dodatkom jednog tranzistora i ekrana iza senzorske površine.

Abstract – In this paper the simple microcontroller capacitive sensor is proposed. The sense oscillator is build around internal comparator with only four external resistors. The output overload protection is achieved with collector voltage detection. By adding one transistor and screen behind sensing surface, increasing sensitivity and longer distance detection are achieved.

1. UVOD

Danas na tržištu postoji veliki broj specijalizovanih blizinskih senzorskih čipova, kapacitivnih ili induktivnih. Mnogi od su predviđeni da rade na određeni način koji se teško može izmijeniti [1], [2].

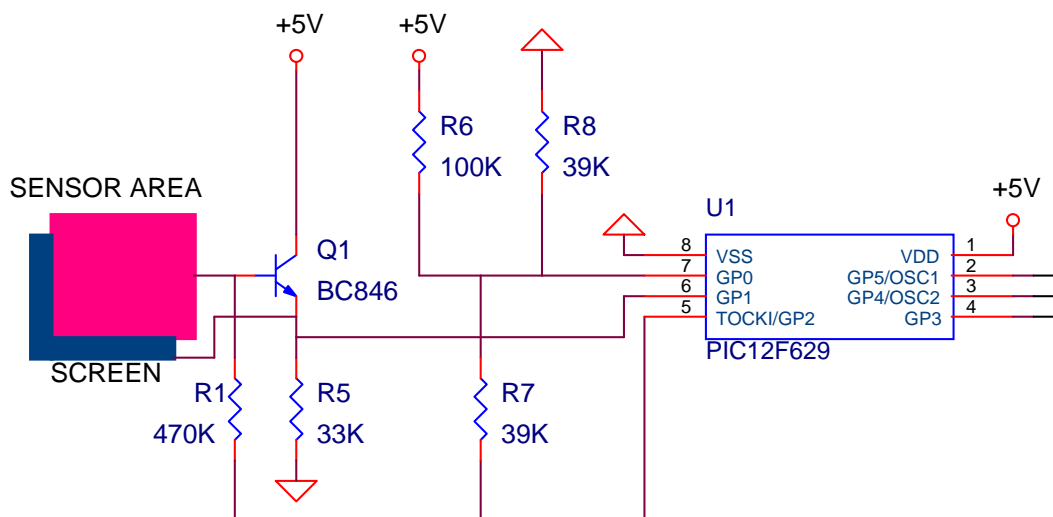
Za razliku od toga, blizinski senzori zasnovani na upotrebi mikrokontrolera su mnogo prilagodljiviji različitim aplikacijama. Na primjer, prilikom detekcije objekta mikrokontroler može aktivirati izlazni transistor, ili poslati paket podataka, ili generisati jedan ili više impulsa. Zavisno od softvera, identičan hardver se može različito ponašati. Tako, mikrokontrolerski senzor može reagovati na prisustvo nekog objekta, ili samo na približavanje objekta, ili na približavanje/udaljavanje objekta u određenom opsegu rastojanja, itd.

2. OPIS ČITAČA

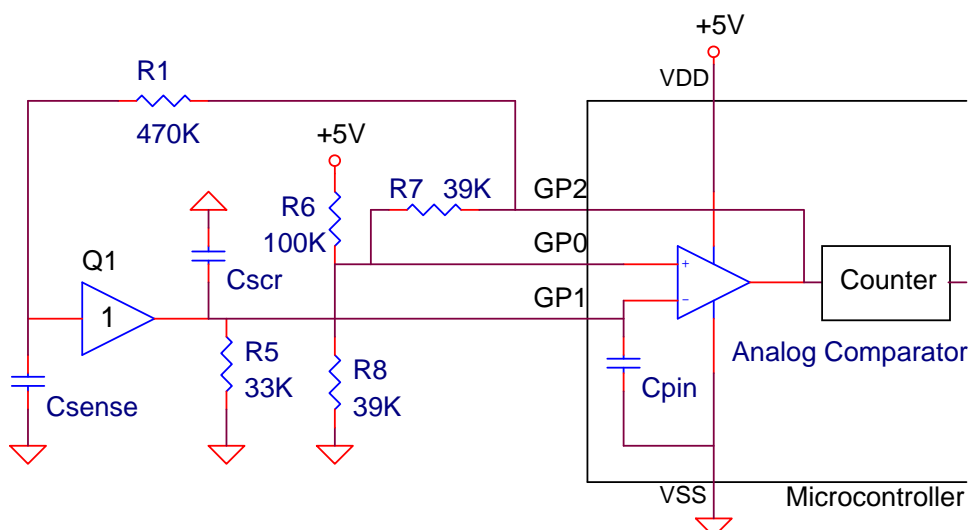
Jednostavni mikrokontrolerski kapacitivni senzor realizovan je upotrebom PIC12F629 mikrokontrolera [3].

Električna šema ulaznog dijela senzora data je na slici 1.

Portovi GP0, GP1 i GP2 povezani su a analogni komparator mikrokontrolera. Na slici 2 je prikazan ekvivalentni dijagram ulaznog dijela senzora. Sa slike se vidi da ulazno kolo formira prosti astabilni relaksacioni oscilator. C_{sense} predstavlja unutrašnju kapacitivnost senzorske površine, dok C_{pin} predstavlja kapacitivnost porta i C_{scr} kapacitivnost ekrana. Otpornici R_6 i R_8 daju početnu polarizaciju komparatora. Otpornik R_7 uvodi brzu pozitivnu povratnu spregu i histerezis. Sa R_1 i C_{sense} zatvorena je spora negativna povratna sprega.



Slika 1. Električna šema ulaznog dijela ekraniziranog kapacitivnog senzora



Slika 2 – Ekvivalentna šema ulaznog dijela ekraniziranog senzora

Frekvencija oscilovanja (f_0) ovakvog kola je funkcija C_{sense} .

$$f_0 = K / C_{sense}; \text{ gdje je } K = \text{const}$$

Detekcija objekta se vrši na osnovu mjerenja f_0 .

Tranzistorom Q_1 i ekranom iza senzorske površine, dobija se usmjerena osjetljivost senzora, tako da senzor reaguje na objekte ispred, a neosjetljiv je na objekta iza ekrana. Ovime se omogućava montaža senzora na sve vrste podloga, uljučujući i metalne konstrukcije.

Izlaz senzora je realizovan kao klasični NPN izlaz (Slika 3).

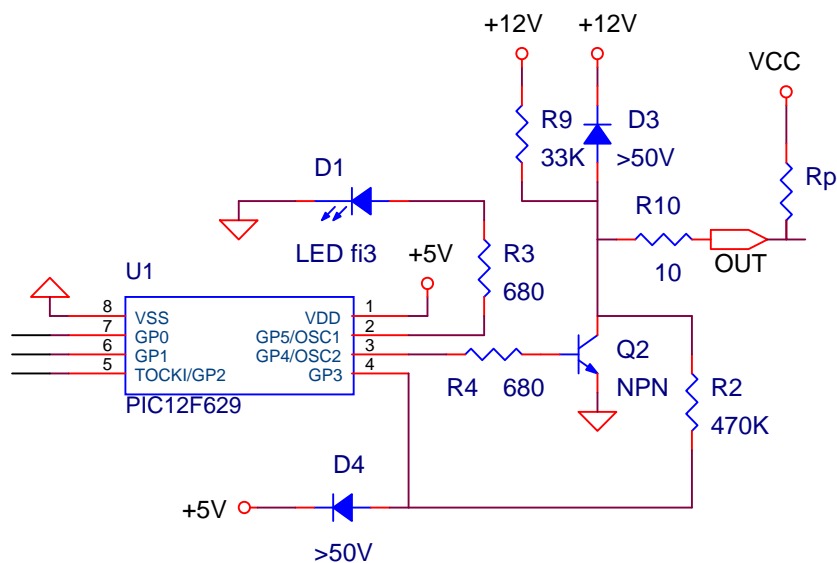
Porta GP4 upravlja radom izlaznog tranzistora Q_2 . Provođenjem, Q_2 na izlaz dovodi masu.

Port GP5 upravlja radom LED D1, kojom se signalizira prisustvo objekta u senzorskom domenu.

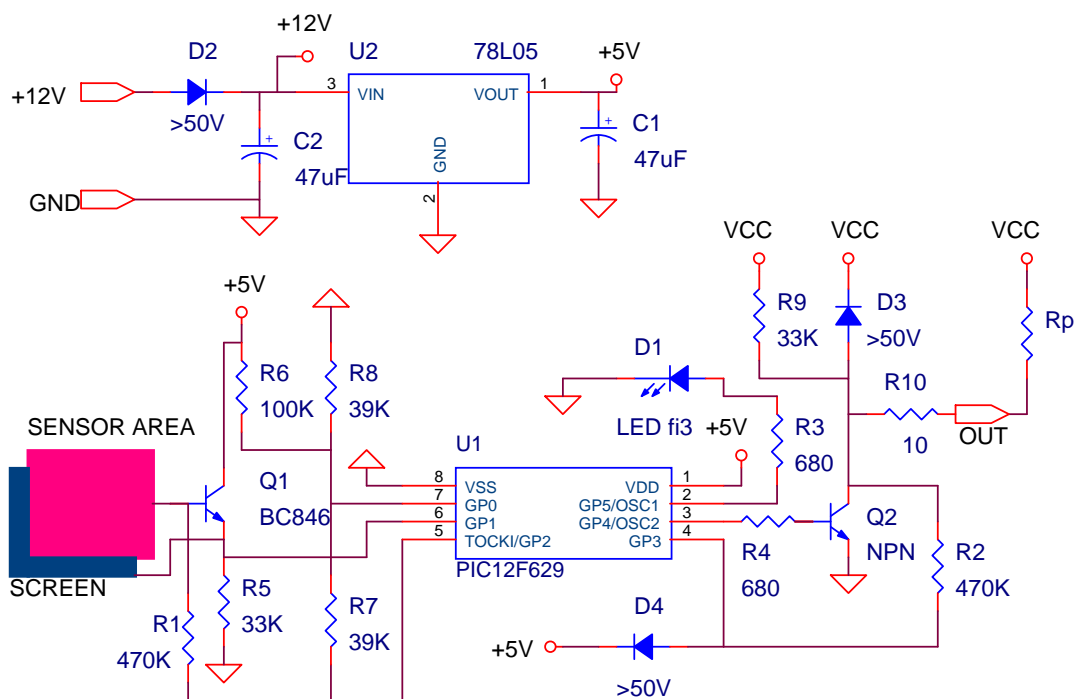
Otpornik R_2 , dioda D4 i port GP3 upotrijebljeni su kao zaštita tranzistora Q_2 od preopterećenja (npr. u slučaju da je njegov kolektor, greškom, direktno spoji na VCC). Zaštita funkcioniše na sljedeći način. Za vrijeme provođenja Q_2 preko otpornika R_2 i porta GP3 nadzire se napon na kolektoru. Ako je napon ispod logičkog praga sve je u redu, Ako je iznad nešto ne valja, tranzistor nije proveo dovoljno, i mikrokontroler ga ugasi. Zatim se u određenom vremenskom intervalu (oko 1ms) opet uključi tranzistor i provjeri se naponski nivo na GP3 portu. Ako nije u redu tranzistor se opet gasi, etc.. Na ovaj način tranzistor ne može da se preoptereći i zagrije. Ovakvo kolo bolje štiti izlazni tranzistor nego uobičajeno strujno limitiranje sa emitorskim otornikom i dodatnim tranzistorom za strujnu zaštitu.

U nekim drugim aplikacijama, osim za kontrolu napajanja potpršača, portovi GP4 i GP3 se mogu upotrijebiti za komunikaciju, kao Tx/Rx.

Na slici 4 prikazana je kompletna električna šema senzora



Slika 3. Električna šema izlaznog dijela senzora



Slika 4. Kompletna električna šema senzora

3. EKSPERIMENTALNI REZULTATI

Tabela 1 prikazuje zavisnost učestanosti f_o od C_{add} , gdje C_{add} predstavlja kapacitivnost koju smo dodali na unutrašnju kapacitivnost senzorske površine.

C_{add} [pF]	f_o [KHz]
0	173.5
0.5	159.5
1	146.4
2	140.1

Tabela 1. Zavisnost f_o od C_{add} kod ekraniziranog kapacitivnog senzora

Tabela 2 prikazuje zavisnost f_o od temperature. Mjerenje je izvedeno bez prisustva objekta.

Temperatura (°C)	f_o (KHz)
20	173.2
30	172.47
40	171.2

Tabela 2. Zavisnost f_o od temperature

Slika 5 prikazuje rezultate, koji ilustruju zavisnost f_o od rastojanja objekta (d), kada je senzor montiran na 5mm od metalne pozadine. Objekat je velika ravna metalna ploča. Izvršena su mjerenja sa ekraniziranim senzorom i senzorom bez ekrana. Rezultati pokazuju da je osjetljivost ekraniziranog senzora oko 80% veća od osjetljivosti senzora bez ekrana. Osim toga, u ovim uslovima, daljina detekcije senzora sa ekranom iznosi oko 20mm, dok je kod senzora bez ekrana oko 12mm [4].

4. ZAKLJUČAK

U ovom radu predložen je jednostavni kapacitivni senzor, zasnovan na upotrebi mikrokontrolera. Senzor je realizovan upotrebom PIC12F629 mikrokontrolera i četiri dodatna otpornika.

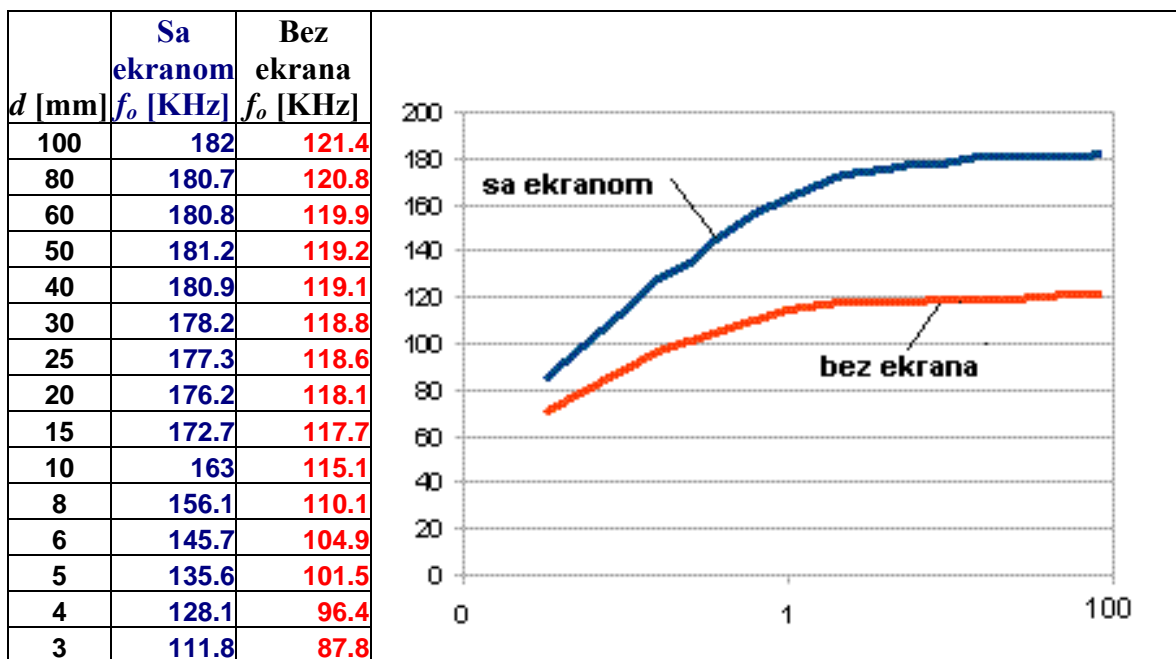
U realizaciji se nastojalo što je moguće više iskoristiti resurse mikrokontrolera. Na primje, kao generator takta upotrijebljen je interni RC oscillator, interni analogni komparator kao centralni dio oscilatora. Za mjerenje učestanosti oscilovanja upotrijebljen je interni brojač.

Upravljanje radom senzora vrši softver mikrokontrolera, što obezbjeđuje fleksibilnost za primjenu u različitim aplikacijama.

Data je popularna realizacija izlaznog stepena sa jednostavnom zaštitom od preopterećenje izlaznog stepena.

Ekranizacijom senzorske površine prošireno je područje primjene senzora i poboljšana je osjetljivost.

Eksperimentalni rezultati su pokazali da je senzor dovoljno temperaturno stabilan za detekciju objekta do rastojanja od 25m.



Slika 5. Zavisnosti f_o [KHz] od d [mm] kod ekraniziranog kapacitivnog senzora (kriva 1) i senzora bez ekrana (kriva 2), u slučaju kada postoji metalna pozadina na rastojanju 5mm.

LITERATURA

- [1] WILSON, JON S.: "Sensor Technology Handbook", Newnes, Bk&CDRom adition, December 8, 2004.
 [2] <http://sensors-transducers.globalspec.com>

- [3] www.microchip.com
 [4] Nedjeljko Lekić, Zoran Mijanović, Radovan Stojanović, "The Simple Microcontrollers Based Proximity Capacitive Sensor", Applied Electronics 2008, Pilsen 10-11 September 2008.

JEDNO PROŠIRENJE AKSIOMA INKLUZIJA VEZA SA REGULARNIM IZRAZIMA AN EXTENSION OF ROLE INCLUSION AXIOMS WITH REGULAR EXPRESSIONS

Nenad Krdžavac, *Fakultet organizacionih nauka, Univerzitet u Beogradu, Srbija*
Milenko Mosurović, *Prirodno matematički fakultet, Univerzitet Crne Gore, Crna Gora,*

Sadržaj - *Strogo parcijalno uređenje na skupu veza kod SROIQ opisne logike, kao osnova OWL 2 jezika, onemogućuje ciklično pojavljivanje veza u aksiomama inkluzije veza. Proširenjem uslova regularnosti na skupu veza, Kazakov je dokazao da je OWL 2 ontologija sa hijerarhijom veza, u kojoj veze imaju regularne jezike, odlučiva. Ovo rešenje ne obuhvata sve regularne jezike. U ovom radu se predlaže proširenje pojma hijerarhije veza sa regularnim izrazima što omogućava da se opišu odnosi između veza koje do sada nismo mogli da zapišemo. Predloženo proširenje ne narušava odlučivost ove logike.*

Abstract - *Strict partial order on the set of role names in SROIQ description logic, as logical basis for OWL 2 language, does not allow cyclic dependencies among role names in role inclusion axioms. In an extension of regularity conditions on the set of roles, Kazakov proved that OWL 2 ontology with role hierarchy, where role inclusion axioms induce regular languages, is decidable. This solution does not include all regular languages. This paper proposes an extension of role hierarchy with regular expressions which allows one to define role inclusion axioms which has not been written so far. This solution does not disturb decidability of this logic.*

1. UVOD

Opisne logike su izvedene od mreža sa strukturnim nasleđivanjem da bi se prevazišli nedostaci semantičkih mreža i sistema okvira pri opisivanju baza znanja kao i izvodjenju zaključaka nad tom bazom [1]. Jezik opisnih logika je prebrojiv skup koji se sastoji od simbola imena pojmova, zatim simbola imena veza, simbola konstanti, logičkih veznika, kao i pomoćnih simbola (leve, desne zagrade, zareza,...) [1]. Osnovu opisnih logika čine složeni pojmovi koji se grade od imena pojmova, i složene veze koje se grade od imena veza. Sa N_C, N_R, N_I, F ($F \subseteq N_R$) redom označavamo skup imena pojmova, veza, konstanti i atributa. Sa C , odnosno R , redom označavamo pojmove i veze. Simbol \perp označava pojam koji ne obuhvata nijedan pojam, dok sa \top se označava pojam koji obuhvata svaki pojam [1]. Za proveru zadovoljivosti baze znanja kod opisnih logika, razvijeno je nekoliko procedura, među kojima i adaptivna metoda tabloa [1].

Uopšteno rečeno, ontologija, čiju logičku osnovu čine neke, po dogovoru, opisne logike, se može definisati kao uređena trojka $O=(R, T, A)$ [3], gde je R jedan Rbox, zatim T je skup aksioma inkluzija klasa ili poznatiji kao Tbox, dok se sa A označavaju aksiome odnosa individua ili Abox [3].

SROIQ [4] opisna logika je osnova OWL 2 [2] jezika, kao jednog od ontoloških jezika semantičkog web-a. On predstavlja jezik za opisivanje dokumenata na web-u. Osnovu ovog jezika čine klase (*owl:Class*) i veze među klasama (*owl:ObjectProperty*) [3]. Takođe je moguće definisati i aksiome inkluzija veza (*Property chains*), što je fokus istraživanja u ovom radu. Automatiskim zaključivanjem moguće je proveriti konzistentnost ontologije, primenom tablo algoritma [4].

Sledeća sekcija daje pregled sintakse i semantike SROIQ opisnog jezika, uključujući i proširene uslove regularnosti

nad skupom veza u aksiomama inkluzija veza. Treća sekcija je glavna sekcija u ovom radu, gde se kroz primjer opisuju ideja proširena a zatim i formalno definiše sintaksa i semantika nove aksiome inkluzije veza. Poslednja sekcija opisuje rezultate dobijene u ovom istraživanju i daje smernice za buduće pravce istraživanja.

2. SROIQ OPISNA LOGIKA I SLOJEVITA ONTOLOGIJA

Prema [4], skup SROIQ pojmova je najmanji skup takav da je:

1. Svako ime pojma, uključujući i nominalne, kao i \perp , \top je SROIQ pojam.
2. Ako su C i D proizvoljni pojmovi, a R veza, dok je S prosta veza, a n je nenegativan ceo broj, tada su:
 $C \sqcap D, C \sqcup D, \neg C, \forall R.C, \exists R.C, \exists S.Self, (\geq nS.C)$
i $(\leq nS.C)$ takođe SROIQ pojmovi.

Opšta aksioma inkluzije pojmova (u daljem tekstu GCI) je izraz oblika $C \sqsubseteq D$ za dva SROIQ pojma C i D . TBox T je konačan skup GCI [4]. Ako su a i b individue ($a, b \in N_I$), C je SROIQ pojam, R je veza (moguće inverza), i S je prosta veza, onda izraze oblika $a : C, (a; b) : R, (a; b) : \neg S, a = b$ nazivamo aksiomama odnosa individua. Konačan skup aksioma odnosa individua nazivamo Abox A [4].

Složene veza (reč) je niz veza $\rho = R_1 \dots R_k$. Aksiome inkluzije veza su izrazi oblika $\rho \sqsubseteq R$. Skup aksioma inkluzije veza čini hijerarhiju veza.

RBox je sastavljen od hijerarhije veza i aksioma oblika *Sim(R), Tra(R), Ref(R), Irr(S), Fun(S), Dis(R,S)*, koji redom označavaju da su odgovarajuće veze simetrične, tranzitivne, reflektivne, irefleksivne, funkcionalne i disjunktne [4].

Semantika SROIQ pojmova se definiše interpretacijom

$I=(\Delta^1, \cdot^1)$, gde je Δ^1 domen interpretacije, dok je \cdot^1 valuacija, koja pridružuje svakom imenu pojma C podskup $C^1 \subseteq \Delta^1$, a svakom imenu veze R pridružuje podskup $R^1 \subseteq \Delta^1 \times \Delta^1$ [4]. Interpretacija se proširuje na složene pojmove i veze prema sledećim pravilima (prema definiciji 6 iz [4]):

$$\begin{aligned} \top^1 &= \Delta^1, \perp^1 = \emptyset, (\neg C)^1 = \Delta^1 \setminus C^1, \\ (C \cap D)^1 &= C^1 \cap D^1, \\ (C \cup D)^1 &= C^1 \cup D^1, \\ (\exists R.C)^1 &= \{x \mid \exists y.(x,y) \in R^1 \wedge y \in C^1\}, \\ (\exists S.Self)^1 &= \{x \mid (x,x) \in S^1\}, \\ (\forall R.C)^1 &= \{x \mid \forall y.(x,y) \in R^1 \Rightarrow y \in C^1\} \\ (\geq nS.C)^1 &= \{x \mid \#\{y : (x,y) \in S^1 \wedge y \in C^1\} \geq n\}, \\ (\leq nS.C)^1 &= \{x \mid \#\{y : (x,y) \in S^1 \wedge y \in C^1\} \leq n\} \\ (R_1R_2\dots R_k)^1 &= R_1^1R_2^1\dots R_k^1 \end{aligned}$$

Baza znanja kod SROIQ familije opisnih logika je uredjena trojka (R, T, A) gde je R jedan Rbox, zatim T je Tbox, dok je A jedan Abox. U SROIQ opisnoj logici dozvoljen skup aksioma inkluzija veza je "regularan" u odnosu na neko strogo parcijalno uredjenje (u oznaci $<$) kako bi ova klasa logika bila odlučiva [4]. Na primer skup veza

$$RS \sqsubseteq T \text{ i } RT \sqsubseteq S \dots\dots\dots(1)$$

ne zadovoljava ove uslove jer važi da je $S < T$ i $T < S$ pa je odavde, na osnovu [4], $S < S$, što dovodi do kontradikcije. Odavde sledi da ne važi strogo parcijalno uredjenje na skupu veza, pa se ne može primeniti tablo algoritam nad datim skupom aksioma veza, tj u predprocesorskom koraku nije moguće konstruisati konačne automate za definisane aksiome implikacija veza oblika (1). Jednostavnije rečeno, na ovom skupu veza nije dozvoljeno ciklično pojavljivanje veza u Rbox-u.

Da bi se prevazišao problem, regularnosti, Yevgeny Kazakov je definisano nove uslove proširenja regularnosti na skupu veza kod SROIQ opisne logike [3]. Ideja je zasnovana na činjenici da procedura za rasudjivanje u SROIQ logici bazirana na tablou [4] ne koristi sintaksne restrikcije direktno, već kao ulaz ima nedeterminističke konačne automate konstruisane na osnovu hijerarhije veza. Sintaksne restrikcije su korišćene samo za konstrukciju takvih automata. Ovo znači da se može koristiti ista procedura za bilo koji skup aksioma inkluzije veza za koji možemo konstruisati odgovarajuće konačne automate.

Prema [3], neka je $\rho \sqsubseteq_0 R$ najmanja relacija između reči veza i veza za koju važi:

- (i) $R \sqsubseteq_0 R$
 - (ii) $\rho \sqsubseteq_0 R$ i $\rho_1 R \rho_2 \sqsubseteq_0 R_1 \in O$ povlači $\rho_1 \rho_2 \sqsubseteq_0 R_1$.
- Za svaku vezu R date ontologije O definišimo jezik $L_O(R) = \{w \mid w \sqsubseteq_0 R\}$ [3].

Definicija 1. (a) Parcijalno uredjenje \leq veza je pogodno za ontologiju O ako:

- (i) $\rho_1 R \rho_2 \sqsubseteq_0 R_1 \in O$ povlači $R \leq R_1$,

(ii) $R \leq R_1$ povlači $R^{-1} \leq R_1^{-1}$.
Pisaćemo $R \approx R_1$ ako je $R \leq R_1$ u $R_1 \leq R$, takođe $R < R_1$ je skraćena za $R \leq R_1$ i ne važi $R \leq R_1$.

(b) Neka je O ontologija, a \leq pogodno parcijalno uredjenje za O . Kazaćemo da je aksioma $\rho \sqsubseteq R$ slojevita [3], u odnosu na O i \leq , ako za svako $R \approx R'$ tako da je $\rho = \rho_1 R \rho_2$ postoje R_1 i R_2 tako da je $\rho_1 R \sqsubseteq_0 R_1$, $R_1 \rho_2 \sqsubseteq_0 R'$, $R \rho_2 \sqsubseteq_0 R_2$ i $\rho_1 R_2 \sqsubseteq_0 R'$. Kažemo da je ontologija O slojevita ako je svaka aksioma oblika $\rho \sqsubseteq R$ takva da je $\rho \sqsubseteq_0 R$ slojevita [3]. U [3] je pokazano da za svaku vezu R u slojevitoj ontologiji O možemo konstruisati konačni automat za jezik $L_O(R)$. Osim toga u [3] je dokazano da je ontologija sa hijerarhijom veza, u kojoj veze imaju regularne jezike, odlučiva.

3. DALJA PROŠIRENJA SROIQ OPISNOG JEZIKA

Ako imamo aksiomu $PQ \sqsubseteq R$, semantika nameće da za svaki par $(x, y) \in (PQ)^1$ važi $(x, y) \in R^1$. Postojeća sintaksa nam ne omogućava da zapišemo da $(x, y) \in R^1$ važi za neke parove a za neke ne. Ovaj rad je, prema našim saznanjima, prvi koji dozvoljava i takve mogućnosti. Prvo ćemo kroz primer izložiti ideju a zatim ćemo dati i formalne definicije sintakse i semantike nove aksiome.

U radu [3] je dokazano da za hijerarhiju veza ontologije O kod kojih svaka veza ima regularni jezik $L_O(R)$ možemo konstruisati odgovarajuće automate i time provjeriti saglasnost ontologije.

Međutim, može se u nekom smislu postaviti i obrnuto pitanje: Ako odnos veza možemo izraziti regularnim jezikom da li to možemo zapisati pomoću hijerarhije veza? Sledeći primjer nam pokazuje da to nije moguće.

Primer 1. Prema [5], posmatrajmo ontologiju O koja sadrži sledeće aksiome

$$MotorUAutu^{-1} o AutoImaTočak \sqsubseteq MotorPokrećeTočak \dots\dots(2)$$

$$MotorUAutu o MotorPokrećeTočak \sqsubseteq AutoImaTočak \dots\dots(3)$$

Aksiome zapisuju odnose između cjeline (Auto) i deljova (Točak, Motor). Aksioma (2) nam govori da ako imamo neki motor koji pripada autu, a taj auto ima točak onda će taj motor morati da pokreće taj točak. Aksioma (3) znači da ako u autu imamo motor i taj motor pokreće točak onda će i taj točak mora pripadati tom autu [5].

Jednostavno je vidjeti da je

$$L_1 = L_O(AutoImaTočak) = (MotorUAutu \ o \ MotorUAutu^{-1})^* \ o \ (AutoImaTočak + MotorUAutu \ o \ MotorPokrećeTočak)$$

$$L_2 = L_O(MotorPokrećeTočak) = (MotorUAutu^{-1} \ o \ MotorUAutu)^* \ o \ (MotorPokrećeTočak + MotorUAutu^{-1} \ o \ AutoImaTočak)$$

U jeziku L_1 imamo i reč $w = MotorUAutu \ o \ MotorUAutu^{-1} \ o \ AutoImaTočak$

pa znači da je $w \sqsubseteq TočakUAutu$, što u modelu ne očekujemo. Naime, $(MotorUAutu \ o \ MotorUAutu^{-1})$ može dati vezu

između različitih automobila Prirodno je nametnuti uslov $Fun(MotorUAutu^{-1})$ pa je

$MotorUAutu \circ MotorUAutu^{-1} = Id.$

Međutim, ako veza $MotorUAutu^{-1}$ nije prosta ne možemo koristiti funkcijonalno ograničenje. Ono što mi očekujemo su jezici

$L_3 = (AutoImaTočak + MotorUAutu \circ MotorPokrećeTočak)$

$L_4 = (MotorPokrećeTočak + MotorUAutu^{-1} \circ AutoImaTočak)$

koji su regularni, a koje ne možemo izraziti aksiomama veza zbog međusobne zavisnosti veze $TočakUAutu$ i $MotorPokrećeTočak$. Da bi smo mogli da izrazimo i druge odnose između veza prirodno se nameće proširenje postojećeg formalizma. S druge strane ako želimo da koristimo postojeće tablo algoritme moramo se ograničiti da jezici veza budu regularni. Jednostavno rešenje je da specifične odnose između veza direktno zapišemo pomoću regularnih izraza.

Definicija 2. Neka je \leq parcijalni poredak među vezama. Za vezu R definišemo regularni izraz

$$r_R := P|r_1+r_2|(r_1) *| r_1 \circ r_2$$

gdje je P veza takva da je $P \leq R$, a r_1, r_2 regularni izrazi za R . Sa ρ_R će mo označavati regularni izraz $R|r_R$, a sa $L(\rho_R)$ jezik generisan sa ρ_R .

Definicija 3. a) Aksiome inkluzije veza su izrazi oblika

(i) $\rho \sqsubseteq R$

(ii) $R \equiv \rho_R$.

b) Skup aksioma inkluzije veza čini hijerarhiju veza.

Definicija 4. Neka je $\rho \sqsubseteq_O R$ najmanja relacija između reči veza i veza za koju važi

(i) $R \sqsubseteq_O R$

(ii) $\rho \sqsubseteq_O R$ i $\rho_1 R \rho_2 \sqsubseteq_O R_1 \in O$ povlači $\rho_1 \rho \rho_2 \sqsubseteq_O R_1$

(iii) $\rho \sqsubseteq_O R$ i $\rho_1 R \rho_2 \in L(\rho_{R_1})$ i $R < R_1$ povlači $\rho_1 \rho \rho_2 \sqsubseteq_O R_1$.

Definicija 5. Aksioma inkluzije veza $R \equiv \rho R$ je zadovoljiva u interpretaciji I ako za svaku reč $w \in L(\rho_R)$ važi $w^I \sqsubseteq R^I$.

Definicija 6. Kažemo da je ontologija O prošireno slojevita ako je svaka aksioma oblika $\rho \sqsubseteq R$ takva da je $\rho \sqsubseteq_O R$ slojevita ili je $\rho \in L(\rho_R)$.

Teorema 1. Neka je ontologija O prošireno slojevita. Za svaku vezu R u O možemo konstrisati automat za jezik $L_O(R)$, pa samim tim ontologija O je odlučiva.

Dokaz. Dokaz je direktan.

4. ZAKLJUČAK I DALJI RAD

U ovom radu smo otvorili jedan problem odnosa između veza koji je interesantan za dalja istraživanja. Uspjeli smo da djelimično rešimo jedan specijalan slučaj. Djelimično znači da je taj slučaj teorijski rješen ali u praksi regularni izrazi mogu biti veoma komplikovani pa ih neće uvijek biti lako modelirati. Drugim rječima i tu postoji prostora za dalja istraživanja na kojima intezivno radimo.

LITERATURA

- [1] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors, *The Description Logic Handbook - Theory, Implementation and Application, volume 2*, Cambridge University Press, 2007.
- [2] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. P. Schneider, and U. Sattler, „OWL 2: The next step for OWL“, *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):309-322, 2008.
- [3] Y. Kazakov, An Extensions of Regularity Conditions for Complex Role Inclusion Axioms, *In Proceedings of the 22nd International Workshop on Description Logics (DL 2009)*, Oxford. July 2009.
- [4] I. Horrocks, O. Kutz, and U. Sattler, The Even More Irresistible SROIQ, *In Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 2006)*, pp. 57-67. AAAI Press, 2006.
- [5] N. Krdžavac, C. Bock, *Reasoning in Manufacturing Part-Part Examples with OWL 2*, U.S. National Institute of Standards and Technology, Technical Report NISTIR 7535, 2008.

DISONANTNI F# INTERVAL GOVORNOG SIGNALA KAO INDIKATOR HIPOKSIJE DISSONANT F# INTERVAL OF THE SPEECH SIGNAL AS A HYPOXIA INDICATOR

Marina Milivojević, *Medicinski fakultet, Niš, Srbija.*
Zoran N. Milivojević, *Visoka tehnička škola, Niš, Srbija.*

Sadržaj – U ovom radu analiziran je efekat hipoksije na disonantni F# interval kod govornog signala. U prvom delu rada prikazani su faktori koji dovode do hipoksije, koja se manifestuje kao smanjenje koncentracije kiseonika u tkivu. Pored toga izvršena je definicija disonantnog opsega koji u odnosu na fundamentalnu frekvenciju govornog signala stoji kao ton F# u odnosu na C. U drugom delu rada grafički i tabelarno su prikazani rezultati testiranja govornog signala grupe ispitanika čiji je izgovor samoglasnika sniman na 200, 800 i 1400 m nadmorske visine. Određen je disonantni koeficijent čija vrednost ukazuje na stepen hipoksije.

Ključne reči: Govorni signal, Fundamentalna frekvencija, Disonance, Hipoksia;

Abstract – In this paper the effect of hypoxia on the dissonant F# interval of the speech signal. In the first part of this paper the factors which lead to hypoxia were shown. Hypoxia manifests itself through the reduction of the oxygen concentration in the tissue. Moreover, the definition of the dissonant frequency was performed and its relation to the fundamental frequency of the speech signal is equal to the one the F# tone has to the C tone. In the second part of the paper, by means of graphs and tabs the results of the test of the testing group were shown. The pronunciations of the group members were recorded at the altitudes of 200, 800 and 1400 meters above sea level. The dissonant coefficient whose value points to the level of hypoxia was determined.

Keywords: Speech signal, Fundamental frequency, Dissonant frequency, Hypoxia;

1. UVOD

Hipoksija (grč. *HYPo* + lat. *OXYgenium* + grč. *hAima*) predstavlja stanje smanjene koncentracije kiseonika u krvi, ćelijama i tkivima organizma [1]. Mozak je najosetljiviji organ tako da stanje hipoksije dovodi do poremećava mentalne aktivnosti (pamćenje, poremećaj vida, govora i dr.). Pored ostalih faktora koji mogu da dovedu do hipoksije je i udisanje vazduha sa smanjenom koncentracijom kiseonika. Takva situacija može da se javi u tunelima, za vreme ronjenja, letovima avionom, kao i boravkom na visokim planinama [2-4].

Sa stanovišta hipoksije može se tretirati stanje akutne i hronične hipoksije. Pod akutnom se podrazumeva situacija kada je čovek, čije je prirodno okruženje na niskim nadmorskim visinama, izložen boravku na visokim planinama. U periodu adaptacije govori se o akutnoj hipoksiji. Pod hroničnom hipoksijom podrazumeva se stanje smanjene količine kiseonika kod ljudi čije je prirodno stanište na visokim planinama. U [3,4] je pokazano da se fundamentalna frekvencija govornog signala menja usled hipoksije. U ovom radu analiziran je efekat hipoksije usled promene nadmorske visine na fundamentalnu frekvenciju govora kao i na neke disonantne spektralne opsege. Značaj disonantnih opsega na kvalitet govornog signala analiziran je zadnjih desetak godina. U radovima [5-7] pokazano je da je moguće izvršiti popravku kvaliteta govornog signala u uslovima nepovoljnog ambijentalnog šuma, kao i kvalitet promuklog glasa, filtriranjem disonantnih frekvencija. Algoritam podrazumeva procenu fundamentalne frekvencije i, na osnovu nje, definisanje i eliminisanje disonantnih spektralnih opsega.

U nastavku rada će, nakon objašnjenja hipoksije i disonantnih frekvencija, biti prikazan originalni algoritam za izračunavanje disonantnog koeficijenta. Na osnovu prikazanih rezultata testiranja biće uspostavljena veza između stepena hipoksije i disonantnog koeficijenta.

Organizacija rada je sledeća. U sekciji 2 opisana je hipoksija. U sekciji 3 prikazana je definicija disonantnih frekvencija. U sekciji pet opisan je novi algoritam za izračunavanje disonantnog koeficijenta. U sekciji 5 su prikazani rezultati testiranja i komparativna analiza. Sekcija 6 je zaključak.

2. HIPOKSIJA

Hipoksija je stanje smanjene količine kiseonika u krvi, ćelijama i tkivima, što za posledicu ima poremećaj u funkcionisanju organa, sistema i ćelija. Posledice hipoksije mogu biti za pojedine organe, kao i za čitav organizam, jer zbog nedovoljne koncentracije kiseonika izumiru ćelije, dolazi do raspada tkiva sa poremećajem funkcija mnogih organa: mozga, pluća, srca i krvnih sudova, jetre, bubrega. Od svih organa u organizmu mozak je najosetljiviji na smanjenje i nedostatak kiseonika. Zbog toga je proučavanje ovog stanja izuzetno aktuelno. Uzroci hipoksije mogu biti mnogostruki: deficit kiseonika u atmosferi (boravak na visokim planinama, u toku incidenta u rudnicima i podzemnim jamama, u vazduhoplovstvu, kosmičkim letovima, podvodnim aktivnostima itd.), bolesti pluća, poremećaji centra za disanje, bolesti krvnih sudova, povećana potreba tkiva za kiseonikom pri ekstremnom mišićnom radu i može se javiti kod sportista i fizičkih radnika.

Dejstvo hipoksije na organizam karakteriše: a) *smanjenje mentalne aktivnosti* (što se ogleda kroz slabo pamćenje, zaboravnost, usporen misaoni tok, pospanost, euforiju, glavobolju, mučninu, poremećaj vida i govora sve do pojave trzaja, konvulzija i nastanka koma), b) *smanjenje radne sposobnosti muskulature* (što se manifestuje usporenim hodom, osećajem nemoći, oslabljenim i usporenim refleksima, lošom koordinacijom motoričkih pokreta akomodacijom oka) i c) *depresija respiratornog centra* koju prati gubitak svesti, koma i smrt.

Hipoksija, kao moćan stresogeni, faktor i utiče na menjanje kompletnog funkcionalnog stanja organizma, a samim tim, i na govorni aparat kod čoveka. Vršena su ispitivanja dejstva hipoksije na govor kao posledica promene nadmorske visine [3]. Kiseonik je u atmosferi vazduha zastupljen sa približno 21% na pritisku od 1 bara (nivo mora). Sa povećanjem visine opada gustina vazduha i, samim tim, količina kiseonika. Sa povećanjem nadmorske visine čovek dobija manje kiseonika, odnosno pojačava se hipoksija. Ispitivanja su pokazala da se na 1600 mnv javljaju značajna dejstva hipoksije. Zbog toga se ova visina naziva pragom reakcije. Najveća visina na kojoj su zasnovana i opstala ljudska naselja iznosi 5500 mnv (planine Andi, Tibet, šerpasi na Himalajima) [4]. Ova visina se smatra krajnjom granicom adaptivne moći čoveka. Zdrav čovek, koji nije priviknut na ovoliko visinu, može da očuva punu radnu sposobnost, odnosno stanje pune svesti, oko 30 min. Procesom adaptacije čovek može da se prilagodi ovoj visini, i da boravi na duži rok. Sa druge strane, ispitivanja su pokazala da na visini od 6700 m nije moguć opstanak na duži rok. Bez primene odgovarajućih sredstava čovek može da izdrži najviše 10 min.

U [3] analizirane su fizičke karakteristike govornog signala usled uticaja akutne hipoksije. Prikazani su rezultati analize fundamentalne frekvencije govornog signala kod dve grupe ispitanika, i to onih koji prirodno žive na 400 mnv i 1600 mnv. Analize su vršene u cilju ispitivanja dejstva akutne i hronične hipoksije. Dejstvo akutne hipoksije analizirano je tako što je obavljeno merenje fundamentalne frekvencije nad ispitanicima koji žive na 400 mnv, i to na 400 i 1600 mnv. Hronična hipoksija analizirana je kod ispitanika koji žive na 1600 mnv. Ispitivanja su obavljena u slučajevima izdvojeno izgovorenih samoglasnika (A, E, I, O i U). Kao zaključak navedeno je da, pored ostalog, dolazi do blagog povećanja fundamentalne frekvencije usled akutne hipoksije. Dejstvo hronične hipoksije dovelo je do značajnog vovećanja fundamentalne frekvencije (od 140 na 170 Hz).

3 MUZIKOLOŠKA DEFINICIJA DISONANTNIH OPSEGA

Teorijom muzike definisane su osnovne osobine zvuka: a) trajanje, b) jačina i c) boja. Izraz *boja* se na zvuk primenjuje u prenosnom smislu, što ukazuje na složenost ove osobine zvuka. Zvučni izvor generiše zvuk sa osnovnom frekvencijom (osnovni ton) kao i prizvuke (aliquotni tonovi u odnosu na osnovni ton). Različiti broj prisutnih alikvota (**lat. aliquoties**, nekoliko puta) i njihova različita relativna jačina u sklopu ukupnog zvučanja, određuje boju zvuka [5,6].

Frekvencija muzički definisanih tonova u odnosu na osnovni ton u intervalu od jedne oktave određena je formulom:

$$F_k = F_0 \times 2^{(k/12)}, \quad k = 0, 1, \dots, 12. \quad (1)$$

gde je F_0 frekvencija osnovnog tona a F_k frekvencija k -tog polutona. Prema osnovnom tonu polutonomi formiraju intervale. Interval je definisan odnosom frekvencija polutona i osnovnog tona. Razlomci F_k/F_0 , za $k=0,1,\dots,12$, koji predstavljaju pojedine intervale (1/1, 135/128, 9/8, 6/5, 5/4, 4/3, 45/32, 3/2, 8/5, 27/16, 9/5, 15/8, 2/1) predstavljaju aproksimaciju prave vrednosti (1). Klasifikacija intervala po zvučnosti ostvaruje se na osnovu razlomka koji ga opisuje. Kada je razlomak jednostavniji interval je, kao sazvučje tonova, stabilniji, odnosno, konsonantniji. Kada je razlomak složeniji stabilnost intervala je manja, tako da je disonantnost veća.

Konsonanca i disonanca nisu oštro razgraničene već zajedno čine jednu izdiferenciranu skalu međustepena, od potpune stabilnosti na jednom, do krajnje labilnosti na drugom kraju skale. U skali se razlikuju: a) savršene (potpune) konsonance (prima (1/1), oktava (2/1), kvinta (3/2) i kvarta (4/3)), b) nesavršene (nepotpune) konsonance (velika terca (5/4), velika seksta (5/3), mala terca (6/5) i mala seksta (8/5)), c) nesavršene (nepotpune) disonance (mala septima (9/5) i velika sekunda (9/8)) i d) savršene (potpune) disonance (mala sekunda (135/138), tritonus ili prekomerna kvarta (45/32) i velika septima (15/8)). Sa stanovišta doživljaja, odnosno percepcije zvuka, muzički interval je definisan kao konsonantan ako je zvuk prijatan (**engl. pleasant**) ili miran (**engl. restful**). Muzički interval je disonantan u velikoj meri ako je zvuk neprijatan ili grub (**engl. rough - hrpav**) [7].

U odnosu na osnovni ton definišu se frekvencije polutona koji sa osnovnim tonom čine konsonance u svim oktavama iz čujnog područja:

$$F_d = F_0 \times 2^{(n+6/12)}, \quad n = 0, 1, \dots, 7; \quad (2)$$

gde je F_0 frekvencija osnovnog tona, n broj oktave i k broj polutona u pojedinim oktavama. Uzimajući ton C kao referentni, odnosno osnovni ton, tada su njemu disonantni poluton F# kao i njegovi harmonici u svim oktavama.

Digitalno procesiranje govornog signala u cilju automatskog određivanja emocionalnog stanja govornika (sreća, tuga, ljutnja, briga, dosada,...) podrazumeva, pored ostalog, i analizu fundamentalne frekvencije i njenog odnosa prema drugim frekvencijama u jednoj oktavi [8-10].

4. ALGORITAM PROCENE KOEFICIJENTA $k_{F\#}$

Cilj je odrediti odnos disonantne energije u spektru u zavisnosti od hipoksije usled promene nadmorske visine. Predloženi algoritam sastoji se od sledećih koraka:

Korak 1: Govorni signal $x(n)$, $n=1,\dots,L$ deli se, primenom prozorske funkcije w na frejmove dužine N , sa preklapanjem frejmova $N/4$.

Korak 2: Za svaki frejm određuje se fundamentalna frekvencija F_0 [11-13].

Korak 3: Određuje se energija u opsegu harmonika fundamentalne frekvencije.

Korak 4: Određuje se energija u disonantnim F# opsezima.

Korak 5: Određuje se disonantni koeficijent $k_{F\#}$.

Disonantni koeficijent $k_{F\#}$ govori o procentualnom učešću energije u disonantnim $F\#$ opsezima u odnosu na energiju u okolini fundamentalne frekvencije i njenih harmonika. Manja vrednost koeficijenta predstavlja povećanje disonantne energije.

Korak 3 predloženog algoritma realizuje se na sledeći način:

For $n=1$ **to** 7, **do**

(a) Određivanje harmonika disonantne frekvencije

$$F_{dF\#} = F_0 \times 2^{\left(\frac{n+6}{12}\right)}, \quad (3)$$

(b) Određivanje granica disonantnog opsega

$$k_{F\#_bl} = \left(F_{dF\#} - \frac{w_0}{2}\right) \frac{F_s}{NFFT},$$

$$k_{F\#_br} = \left(F_{dF\#} + \frac{w_0}{2}\right) \frac{F_s}{NFFT},$$

(c) **For** $i=k_{F\#_bl}$ **to** $k_{F\#_br}$, **do**

$$E_{F\#k}(i) = Sx(i)$$

end

end

Korak 4 predloženog algoritma realizuje se na sledeći način:

For $n=1$ **to** 20, **do**

$$F_{0n} = nF_0, \quad (4)$$

(b) Određivanje granica opsega fundamentalne frekvencije

$$k_{F0_bl} = \left(nF_{dF\#} + \frac{w_0}{2}\right) \frac{F_s}{NFFT},$$

$$k_{F0_br} = \left(nF_{dF\#} - \frac{w_0}{2}\right) \frac{F_s}{NFFT},$$

(c) **For** $i=k_{F0_bl}$ **to** k_{F0_br} , **do**

$$E_{F0k}(i) = Sx(i)$$

end

end

Korak 4 predloženog algoritma realizuje se na sledeći način:

$$E_{F0} = \sum_{k=1}^L \left(E_{F0k}(k)\right)^2, \quad (5)$$

$$E_{F\#} = \sum_{k=1}^L \left(E_{F\#k}(k)\right)^2, \quad (6)$$

$$k_{F\#} = \frac{E_{F0}}{E_{F\#}}, \quad (7)$$

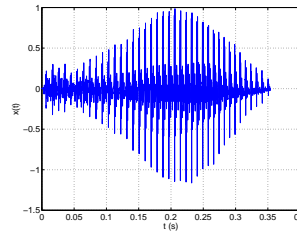
5. EKSPERIMENTALNI REZULTATI I ANALIZA

5.1 BAZA GOVORNIH SIGNALA

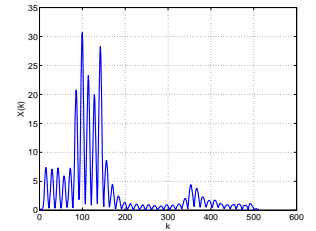
U cilju testiranja formirana je testna grupa osoba čije je prirodno prebivalište 200 mnv (grad Niš, Srbija). Testnu grupu čini 5 osoba. Merenja su obavljena na 200m, 800m i 1400 m. Svaki ispitanik je tri puta na svakoj visini, sa pauzom od 5 min između izgovaranja, izgovorio samoglasnike A, E, I, O i U. Govorni signal je u vidu wav fajla smeštan na hard disk, i time formirana baza govornih signala ($F_s=16$ kHz).

5.2 REZULTATI

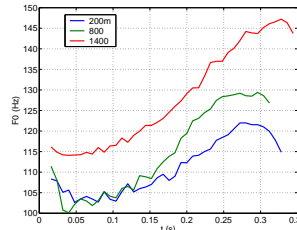
Napred opisani algoritam obrade govornog signala primenjen je sa parametrima: prozorska funkcija Hamming $N=512$, $NFFT=2048$, primenjen je nad bazom govornih signala. Rezultati procesiranja prikazani su na narednim slikama i tabeli. Vremenski oblik govornog signala kod izgovora samoglasnika A prikazan je na sl. 1. Niži delovi spektra istog signala prikazana su na sl. 2. Trajektorije fundamentalne frekvencije za različite visine prikazane su na sl.3. Srednja vrednost fundamentalne frekvencije za samoglasnike u funkciji nadmorske visine prikazana je na sl.4 i tbl.1. Srednja vrednost disonantnog koeficijenta za samoglasnike u funkciji nadmorske visine prikazane su u tbl.2 i sl.5. Srednja vrednost svih disonantnih koeficijenata za sve samoglasnike prikazana je na sl.6.



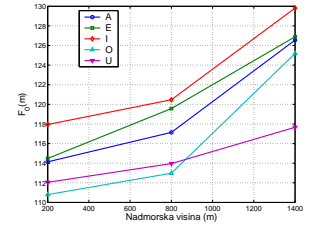
Sl. 1. Vremenski oblik govornog signala (samoglasnik A).



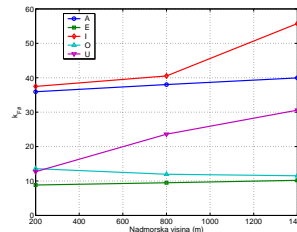
Sl. 2. NF deo spektra govornog signala (samoglasnik A).



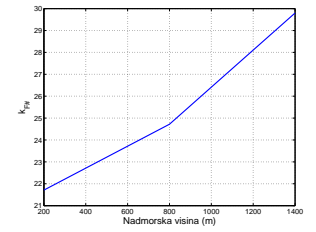
Sl. 3. Trajektorije fundamentalne frekvencije za 200, 800 i 1400 mnv (samoglasnik A).



Sl. 4. Srednja vrednost fundamentalne frekvencije za samoglasnike u funkciji nadmorske visine.



Sl. 5. Srednja vrednost $k_{F\#}$ za samoglasnike u funkciji nadmorske visine.



Sl. 6. Srednja vrednost disonantnog koeficijenta za sve samoglasnike.

Samoglasnici	Fundamentalna frekvencija [Hz]		
	Nadmorska visina [m]		
	200	800	1400
A	114.14	117.14	126.55
E	114.48	119.57	126.89
I	117.94	120.47	129.82
O	110.80	112.98	125.19
U	112.05	113.96	117.66
Srednja vrednost	113.88	116.824	125.222

Tbl. 1. Fundamentalna frekvencija u funkciji nadmorske visine za sve samoglasnike.

Samoglasnici	Disonanti koeficijent $k_{F\#}$		
	Nadmorska visina m		
	200	800	1400
A	35.94	38.02	40.96
E	8.84	9.49	10.21
I	37.48	40.53	55.75
O	13.59	11.97	11.52
U	12.71	23.59	30.58
Srenja vrednost	21.71	24.72	29.80

Tbl. 2. Koeficijent disonantne frekvencije $k_{F\#}$ u funkciji nadmorske visine za sve samoglasnike.

5.3 ANALIZA REZULTATA

Analizom rezultata prikazanih na sl.3 do sl.6 kao i numeričkih podataka prikazanih u tbl. 1 i tbl. 2 donose se sledeći zaključak:

1. Sa povećanjem nadmorske visine dolazi do povećanja Fundamentalne frekvencije (u odnosu na 200 mnv povećanje za 2% na 400 mnv i 10.06% na 1400mnv), što se u velikoj meri slaže sa rezultatima iz [3].

2. Sa povećanjem nadmorske visine povećava se vrednost disonantnog koeficijenta $k_{F\#}$ (u odnosu na 200 mnv povećanje za 13.76% na 400 mnv i 36.69% na 1400mnv) odnosno dolazi do smanjenja energije u disonantnim F# opsezima.

Analizom disonantnog koeficijenta $k_{F\#}$ moguće je doneti zaključke o stepenu hipoksije. Takav podatak je moguće iskoristiti u mnogim situacijama. Na primer, kod pilota na velikim visinama može da otkáže maska ili da dođe do dekompresije aviona. U tom slučaju moguće je, na osnovu razgovora pilota sa dispičerom otkriti znake hipoksije u inicijalnom stadijumu i preduzimanje odgovarajućih koraka u cilju smanjenja katastrofalnih posledica [4].

6. ZAKLJUČAK

U ovom radu analiziran je govorni signal sa ciljem određivanja stepena hipoksije. U radu je predložen nov metod koji se bazira na analizi disonantnih F# spektralnih opsega. Izračunat je disonantni koeficijent za samoglasnike A, E, I, O i U koje je izgovarala grupa ispitanika na različitim nadmorskim visinama (200, 800 i 1400 mnv). Sa povećanjem nadmorske visine disonantni koeficijent se povećava. Samim tim može biti korišćen kao indikator stepena hipoksije.

Rezultati prikazani u ovom radu su deo eksperimenta koji je još u toku. Dalji koraci biće u pravcu povećanja grupa ispitanika, povećanje nadmorske visine kao i analiza drugih disonantnih opsega.

LITERATURA

[1] V. Liere, J. Edward, J. Clifford, *Hypoxia*. The University of Chicago and London, 1963.

[2] Richalet J.P., R. Kacimi, A.M. Antezana. *The control of chronotropic function in hypobaric hypoxia*. Int. J. Sports Med. 13:S22-S24, 1992.

[3] J. Obrenović, *Fundamental frequency of the voice in relation to hypoxia as a stressor*. Facta Universitatis, Series: Philosophy, Sociology and Psychology, Vol. 2, No. 9, pp. 683 – 689, 2002.

[4] J. Obrenović, M. Nešić, *Kognitivna vrednost govornog signala u uslovima akutne hipoksije*, Godišnjak za psihologiju, Godina II, broj 2, Filozofski fakultet, Univerzitet u Nišu, str. 127-137, 2003.

[5] B. Joen, S. Kang, S.J. Baek, K.M. Subg, *Filtering of a Dissonant Frequency Based on Improved Fundamental Frequency Estimation for Speesh Enhancment*, IEICE Trans. Fundamentals, Vol. E86-A, No.8, pp. 2063-2064, August 2003.

[6] S. Kang, *Dissonant frequency filtering technique for improving perceptual quality of noisy speech and husky voice*, Signal Processing, Vol. 84, pp. 431-433, 2004.

[7] Z. Milivojević, D. Balanesković, *Enhancement Of The Perceptive Quality Of The Noisy Speech Signal By Using Of DFF-FBC Algorithm*, Facta Universitatis, Ser.: Elec. Energ. Vol. 22, no. 3, pp. 379-392, December 2009.

[8] B. Yang , M. Lugger, *Emotion recognition from speech signals using new harmony features*, Signal Processing, Vol. 90, pp. 1415–1423, 2010.

[9] I. Murray, C. Baber, A. South, *Towards a definition and working model of stress and its effects on speech*, Speech Communication, Vol. 20, pp. 3-12, 1996.

[10] R. Ruiz, E. Absil, B. Harmegnies, C. Legros, D. Poch, *Time and spectrum related variabilities in stressed speech under laboratory and real conditions*, Speech Communication, Vol. 20 pp. 111-129, 1996.

[11] H.S. Pang, S.J. Baek, Sung KM, *Improved Fundamental Frequency Estimation Using Parametric Cubic Convolution*, IEICE Trans. Fundamentals E83-A pp. 2747-50, 2000.

[12] Z. Milivojevic, M. Mirkovic, S. Milivojevic, *An estimate of fundamental frequency using PCC interpolation – comparative analysis*, Journal: Information Technology And Control, Vol. 35. No. 2, pp. 131-136, Jun, 2006.

[13] Z. Milivojević, M. Mirković., *Estimation of the fundamental frequency of the speech signal modeled by the SYMPES method*, AEU - International Journal of Electronics and Communications, Vol. 63, pp. 200-2008, 2009.

PROCENA FUNDAMENTALNE FREKVENCIJE GOVORNOG SIGNALA KOMPRIMOVANOG G.723.1 ALGORITMOM PCC INTERPOLACIJOM ESTIMATION OF FUNDAMENTAL FREQUENCY OF SPEECH SIGNAL COMPRESSED BY G.723.1 ALGORITHM BY MEANS OF PCC INTERPOLATION

Zoran N. Milivojević, Nataša Savić, *Visoka tehnička škola, Niš, Srbija.*

Sadržaj – U ovom radu analizirani su rezultati procene fundamentalne frekvencije govornog signala koji je komprimovan G.723.1 koderom. Procena fundamentalne frekvencije je vršena Peaking-Peaks algoritmom sa implementiranom PCC Gerville-ovim interpolacionim jezgrom. U drugom delu rada izvršena je analiza efikasnosti procene u odnosu na MSE i odabrana prozorska funkcija koja daje optimalne rezultate.

Ključne reči: Govorni signal, Fundamentalna frekvencija, Kodovanje;

Abstract – In this paper the results of the estimation of the fundamental frequency of the G.723.1 coder-compressed speech signal have been analysed. The estimation of the fundamental frequency was performed by the Peaking Peaks algorithm with the implemented PCC Gerville interpolation kernel. In the second part of the paper an analysis of the efficacy of the estimation in relation to the MSE has been performed and the window function with the optimal results selected.

Keywords: Speech signal, Fundamental frequency, Coding;

1. UVOD

Za VoIP (**engl.** *Voice over IP*) servise ITU (**engl.** *International Telecommunication Union*) je definisala standard H.324 kojim se specificiraju komponente, protokoli i procedure koje omogućavaju multimedijalne komunikacione servise u mrežama sa paketskom komunikacijom [1]. Pored ostalog, H.324 definiše audio i video kodere-dekodere (CODEC). G.723.1 je audio kodek koji se intenzivno koristi u VoIP komunikacijama [2]. Na kvalitet govora, koji se prenosi posredstvom paketske mreže, utiče kašnjenje, varijacije kašnjenja, gubici paketa i eho. U procesu kodovanja vrši se procena fundamentalne frekvencije na osnovu koje se vrši dalja obrada signala. U procesu dekodiranja obavlja se funkcija rekonstruisanja govornog signala koja podrazumeva obnavljanje fundamentalne frekvencije [3,4].

Nakon prenosa govornog signala korišćenjem VoIP servisa na prijemnoj strani se javlja potreba za procesiranjem govornog signala i, pored ostalog, određivanje fundamentalne frekvencije (smanjenje eha, popravka kvaliteta i razumljivosti govora, verifikacija govornika,...). Razvijen je veliki broj algoritama za određivanje fundamentalne frekvencije kod kojih se analiza vrši u vremenskom i frekvencijskom domenu [5-11].

Često primenjivana metoda određivanja fundamentalne frekvencije zasniva se na lociranju pikova amplitudske karakteristike u specificiranom frekvencijskom opsegu. Ovom metodom se analiziraju vrednosti signala u spektru na frekvencijama na kojima je izračunata diskretna Fourier-ova transformacija (**engl.** *Discrete Fourier Transform - DFT*). Najčešće se stvarna vrednost fundamentalne frekvencije ne nalazi na frekvencijama gde je izračunata DFT, već leži između dva odbirka. Time se prouzrokuje greška procene frekvencije. Jedan način smanjenja greške je određivanje interpolacione

funkcije i procena karakteristika spektra u intervalu između dva odbirka.

Izračunavanje interpolacione funkcije primenom parametarske kubne konvolucije (**engl.** *Parametric Cubic Convolution - PCC*) prikazano je u [8-13]. Korišćenjem srednje kvadratne greške (**engl.** *Mean Square Error - MSE*) kao mere kvaliteta algoritma pokazano je da na preciznost procene fundamentalne frekvencije utiče implementirano konvoluciono jezgro kao i prozorska funkcija.

U [13] je predložen algoritam procene fundamentalne frekvencije govornog signala komprimovanog G.723.1 algoritmom sa implementiranim Keys-ovim interpolacionim jezgrom. Testirane su neke standardne prozorske funkcije i određeni optimalni parametri jezgra. U cilju smanjenja greške procene fundamentalne frekvencije nastavljeno je istraživanje PCC jezgara i prozorskih funkcija. U ovom radu mi smo: a) saglasno definiciji Gerville-ovog kubnog interpolacionog jezgra odredili analitičke izraz za izračunavanje fundamentalne frekvencije, b) implementirali Gerville-ovo interpolaciono jezgro u algoritam za procenu fundamentalne frekvencije kod audio i govornog signala kodiranih G.723.1 algoritmom, koji je predložen u [13], c) sprovedi detaljnu analizu rezultata procene u zavisnosti od implementiranog prozora. Rezultati će biti upoređeni sa rezultatima određivanja fundamentalne frekvencije kod govornih signala nad kojima nije primenjen nikakav algoritam kompresije. Kao mera kvaliteta algoritma biće primenjena srednja kvadratna greška MSE. Na osnovu minimalnih vrednosti MSE-a biće određeni optimalni parametri jezgra za slučaj implementiranja nekih karakterističnih prozorskih funkcija.

Organizacija rada je sledeća. U sekciji 2 prikazana je algoritam procene fundamentalne frekvencije. U sekciji 2.1 opisano je Gerville-ovo jezgro, u sekciji 2.2 parametri jezgra dok su u sekciji 2.3 definisani test signali. U sekciji 3 prika-

zani su eksperimentalni rezultati i rezultat komparacije. Sekcija 4 je zaključak.

2. ALGORITAM PROCENE FUNDAMENTALNE FREKVENCIJE

Algoritam procene fundamentalne frekvencije prikazan je na sl.1. Algoritam se realizuje u sledećim koracima:

Korak 1: Test signalu $s(n)$ superponira se šum $n(n)$:

$$s_n(n) = s(n) + n(n), \quad (1)$$

Korak 2: Signal $s_n(n)$ se kodira G.723.1 koderom.

Korak 3: Kodirani signal se dekodira G.723.1 dekoderom i dobija signal $x(n)$.

Korak 4: Dekodirani signal $x(n)$ se obrađuje prozorskom funkcijom $w(n)$ dužine N .

Korak 5: Primenom DFT-a izračunava se spektar:

$$X(k) = DFT(x(n)), \quad (2)$$

Spektar je izračunat u diskretnim vrednostima indeksiranim sa $k=0, \dots, N-1$, gde N predstavlja dužinu DFT-a. Realni spektar signala $x(n)$ je kontinualan dok DFT određuje vrednosti spektra u pojedinim diskretnim tačkama.

Korak 6: Primenom Peak picking algoritma određuje se pozicija maksimuma realnog spektra koji se nalazi između k -tog i $(k+1)$ -og uzorka, pri čemu su vrednosti $X(k)$ i $X(k+1)$ najveće u specificiranom opsegu.

Korak 7: Izračunava se pozicija maksimuma u spektru primenom konvolucione interpolacije. Rekonstruisana kontinualna funkcija je:

$$X_r(f) = \sum_{i=k-L}^{k+L+1} p_i \cdot r(f-i), k \leq f \leq k+1, \quad (3)$$

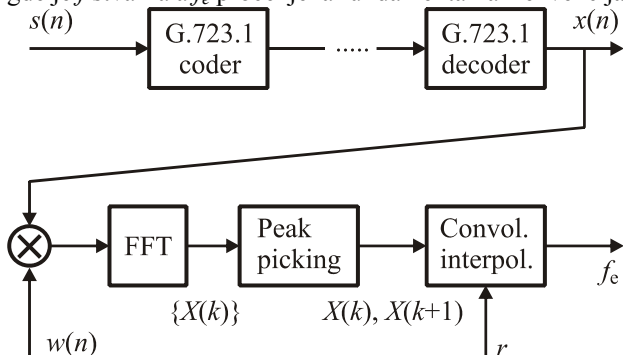
gde je $p_i = X(i)$, $r(f)$ jezgro interpolacije, dok L određuje broj uzoraka koji učestvuju u intrpolaciji.

Korak 8: Diferenciranjem $X_r(f)$ i izjednačavanjem sa nulom određuje se pozicija maksimuma koja predstavlja procenjenu fundamentalnu frekvenciju f_e .

Kvalitet algoritma procene fundamentalne frekvencije određuje se pomoću MSE:

$$MSE = \overline{(f - f_e)^2}, \quad (4)$$

gde je f stvarna a f_e procenjena fundamentalna frekvencija.



Sl. 1. Grafički prikaz algoritma za procenu fundamentalne frekvencije.

2.1 Gerville-ovo interpolaciono jezgro

U daljoj analizi korišćeno je Gerville-ovo parametarsko interpolaciono jezgro [9]:

$$r(f) = \begin{cases} \left(\alpha + \frac{3}{2}\right)|f|^3 - \left(\alpha + \frac{5}{2}\right)|f|^2 + 1; & \text{if } 0 \leq |f| \leq 1, \\ \frac{1}{2}(\alpha - 1)|f|^3 - \left(3\alpha - \frac{5}{2}\right)|f|^2 + & \\ \left(\frac{11}{2}\alpha - 4\right)|f| - (3\alpha - 2); & \text{if } 1 \leq |f| \leq 2, \\ -\frac{1}{2}\alpha|f|^3 + 4\alpha|f|^2 - \frac{21}{2}\alpha|f| + 9\alpha; & \text{if } 2 \leq |f| \leq 3, \\ 0; & \text{if } 3 \leq |f|. \end{cases}, \quad (5)$$

Za $L=6$ iz (3) i (5), izračunava se pozicija maksimuma (fundamentalna frekvencija):

$$f_{\max} = \begin{cases} k - \frac{c}{2b}, & a = 0, \\ k + \frac{-b - \sqrt{b^2 - ac}}{a}, & a \neq 0 \end{cases}, \quad (6)$$

gde je:

$$\begin{aligned} a &= -\frac{3}{2}\alpha p_{k-2} + \frac{3}{2}(\alpha - 1)p_{k-1} + 3\left(\alpha + \frac{3}{2}\right)p_k - 3\left(\alpha + \frac{3}{2}\right)p_{k+1} \\ &\quad - \frac{3}{2}(\alpha - 1)p_{k+2} + \frac{3}{2}\alpha p_{k+3}; \\ b &= -2\alpha p_{k-2} + (-3\alpha + 2)p_{k-1} - (2\alpha + 5)p_k + 4(\alpha + 1)p_{k+1} \\ &\quad - p_{k+2} - \alpha p_{k+3}; \\ c &= -\frac{1}{2}\alpha p_{k-2} + \left(\alpha - \frac{1}{2}\right)p_{k-1} - \left(\alpha - \frac{1}{2}\right)p_{k+1} + \frac{1}{2}\alpha p_{k+2} \end{aligned}, \quad (7)$$

Optimalna vrednost parametra α određuje se tako da se dobije minimalna vrednost MSE:

$$\alpha_{opt} = \arg \min_{\alpha} (MSE), \quad (8)$$

Detaljna analiza u [8] pokazuje da je minimalna vrednost MSE zavisna od implementirane prozorske funkcije kojom se vrši obrada signala $x(n)$ u vremenskom domenu.

2.2. Parametri interpolacionog jezgra

Algoritam određivanja parametra interpolacionog jezgara α realizuje se sledećim koracima:

Korak 1: Vremenski signal $x(n)$, nad kojim je predhodno izvršeno kodovanje i dekodovanje G.723.1 algoritmom, modifikuje prozorskom funkcijom $w(n)$ dužine N ,

Korak 2: primenom DFT-a određuje se spektar $X(k)$,

Korak 3: primenom PCC interpolacije vrši se rekonstrukcija kontinualne funkcije koja predstavlja spektar $X(f)$,

Korak 4: računa se MSE za razne vrednosti parametra α u zavisnosti od implementirane prozorske funkcije.

Korak 5: određuje se α_{opt} za koje se dobija minimalna vrednost MSE.

2.3 Test signali

Procena fundamentalne frekvencije biće obavljena nad:

- sinusnim test signalom i
- govornim test signalom.

Sinusni test signal za potrebe testiranja algoritama procene fundamentalne frekvencije definisanim je u [8]:

$$s(t) = \sum_{i=1}^K \sum_{g=0}^M a_i \sin \left(2\pi i \left(f_o + g \frac{f_s}{NM} \right) t + \theta_i \right), \quad (9)$$

gde je f_o fundamentalna frekvencija, θ_i i a_i faza i amplituda i -tog harmonika, K broj harmonika, M broj tačaka između dva uzoraka u spektru u kojima se vrši PCC interpolacija.

Govorni signal je dobijen snimanjem govornika u realnom akustičkom ambijentu. Matematičkim transformacijama obezbeđeno je da su fundamentalna frekvencija govornika i sinusnog signala jednake u cilju dalje komparativne analize.

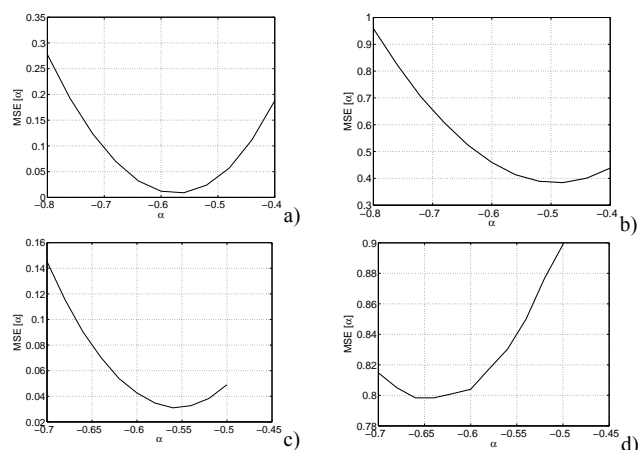
3. EKSPERIMENTALNI REZULTATI I KOMPARACIJA

3.1 Parametri testiranja

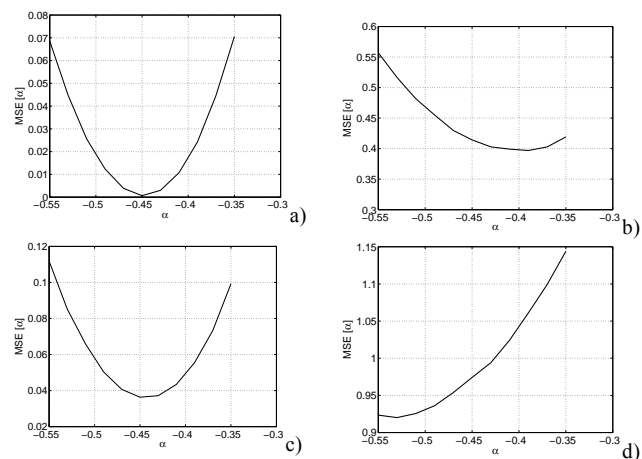
U procesu simulacije f_o i θ_i su slučajne promenljive sa uniformnom raspodelom u opsegu [G2 (97.99 Hz), G5 (783.99 Hz)] i $[0, 2\pi]$ respektivno. Frekvencija signala uzorkovanja je $f_s=8$ kHz a dužina prozorske funkcije $N=256$ čime je obezbeđena analiza podsekvenci koje traju 32 ms. Rezultati koji su prezentirani u daljem delu rada odnose se na $f_o=125$ -140.625Hz (frekvencije između osme i devete spektralne komponente). Broj frekvencija u specificiranom opsegu za koje se vrši procena je $M=100$. Sinusni test signal je sa $K=10$ harmonika. Sve dalje analize odnosiće se na a) Hamming-ov, b) Hann-ov, c) Blackman-ov, d) pravougaoni, e) Kaiser-ov i f) trougaoni prozor.

3.2 Eksperimentalni rezultati

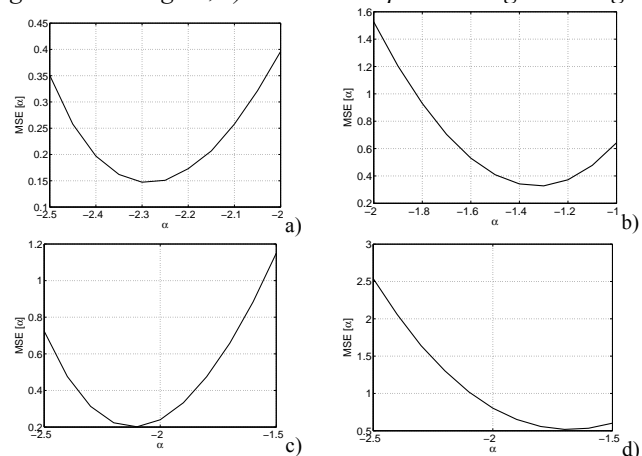
Primenjujući algoritam za određivanje parametara Greville-ovog interpolacionog jezgra nacrtani su dijagrami $MSE(\alpha)$ i određene vrednosti α_{opt} za: a) Hamming-ovu (sl.2), b) Hann-ovu (sl.3), i c) pravougaonu (sl.4) prozorsku funkciju. Vrednosti MSE i α_{opt} prikazane su u tbl.1 (sinusni test signal) i tbl.2 (govorni test signal).



Sl. 2. $MSE(\alpha)$ za Hamming-ovu prozorsku funkciju: a) nekomprimovani sinusni test signal, b) G.723.1 komprimovani sinusni test signal, c) nekomprimovani govorni test signal, d) G.723.1 komprimovani govorni signal.



Sl. 3. $MSE(\alpha)$ za Hann-ovu prozorsku funkciju: a) nekomprimovani sinusni test signal, b) G.723.1 komprimovani sinusni test signal, c) nekomprimovani govorni test signal, d) G.723.1 komprimovani govorni signal.



Sl. 4. $MSE(\alpha)$ za pravougaonu prozorsku funkciju: a) nekomprimovani sinusni test signal, b) G.723.1 komprimovani sinusni test signal, c) nekomprimovani govorni test signal, d) G.723.1 komprimovani govorni signal.

TABELA 1: MINIMALNO MSE I α_{OPT} ZA SINUSNI TEST SIGNAL

	Nekodirani signal		Signal kodiran G.723.1 algoritmom	
	α_{opt}	MSE	α_{opt}	MSE
Hamming	-0.560	0.0089	-0.4800	0.3841
Hann	-0.450	0.0006573	-0.3900	0.3969
Blackman	-0.410	0.0002441	-0.3650	0.3938
Rectangular	-2.300	0.1472	-1.3000	0.3274
Kaiser	-0.660	0.0059	-0.5400	0.3598
Triangular	-0.575	0.0007793	-0.5000	0.3659

TABELA 2: MINIMALNO MSE I α_{OPT} ZA GOVORNI TEST SIGNAL

	Nekodirani signal		Signal kodiran G.723.1 algoritmom	
	α	MSE	α	MSE
Hamming	-0.560	0.0310	-0.6400	0.7983
Hann	-0.450	0.0363	-0.5300	0.9201
Blackman	-0.410	0.0344	-0.5100	0.9209
Rectangular	-2.100	0.2016	-1.7000	0.5178
Kaiser	-0.660	0.0255	-0.7800	0.7787
Triangular	-0.575	0.0256	-0.6750	0.8011

3.3 Komparativna analiza

Na osnovu rezultata koji su prikazani u tbl.1 i tbl.2 vidi se da:

- a) kod sinusnog test signala najveća preciznost procene

fundamentalne frekvencije je kod primene Blackman-ove (MSE=0.0002441) prozorske funkcije. Kod kodiranog sinusnog test signala najveća preciznost procene je kod pravougaone (MSE=0.3274) funkcije. Primenom kodiranja preciznost procene fundamentalne frekvencije je smanjena $0.3274/0.0002441=1341.25$ puta.

b) kod govornog test signala najveća preciznost je kod Kaiser-ove prozorske funkcije (MSE=0.0255). Kod kodiranog govornog signala najveća preciznost je kod pravougaone prozorske funkcije (MSE=0.5178). Primenom kodiranja preciznost procene fundamentalne frekvencije je smanjena $0.5178/0.0255=20.3$ puta.

c) kod kodiranog govornog signala u odnosu na kodirani sinusni signal nepreciznost procene fundamentalne frekvencije veća je $0.5178/0.3274=1.58$ puta.

Kod primene Keys-ovog jezgra [13], dobijeni su sledeći rezultati: kod sinusnog test signala najveća preciznost procene je kod primene Blackman-ove (MSE=0.000619), dok je kod kodiranog sinusnog test signala kod pravougaone (MSE=0.3855) funkcije, kod govornog test signala je kod trougaone prozorske funkcije (MSE=0.0277) i kod kodiranog govornog signala je kod pravougaone prozorske funkcije (MSE=0.6752). Poređenjem rezultata sa rezultatima Gerville-ovog jezgra zaključuje se da:

a) kod sinusnog test signala je preciznost procene kod Gerville-ovog jezgra je veća $0.000619/0.0002441=2.53$ puta,

b) kod kodiranog sinusnog test signala je preciznost procene kod Gerville-ovog jezgra je veća $0.3855/0.3274=1.17$ puta,

c) kod govornog test signala je preciznost procene kod Gerville-ovog jezgra je veća $0.0277/0.0255=1.08$ puta i

d) kod kodiranog govornog test signala je preciznost procene kod Gerville-ovog jezgra je veća $0.6752/0.5178=1.3$ puta.

Sagledavajući dobijene rezultate uočava se veća preciznost procene fundamentalne frekvencije algoritmom sa implementiranim Gerville-ovim u odnosu na Keys-ov interpolaciono jezgro. Međutim, imajući u vidu greške procene kodiranog u odnosu na nekodirani signal (sinusni i govorni), izvlači se zaključak da procesiranje govornog signala nad kojim je vršeno G.723.1 kodiranje (prenos govora IP mrežama) gde se zahteva procena fundamentalne frekvencije (popravka kvaliteta govornog signala,...) vrlo neprecizno.

4. ZAKLJUČAK

U ovom radu analizirani su rezultati procene fundamentalne frekvencije govornog signala komprimovanog G.723.1 algoritma koji se intenzivno koristi kod VoIP servisa. Procena fundamentalne frekvencije vršena je Peaking-Peaks algoritmom sa implementiranim PCC interpolacijom. Eksperimenti su vršeni sa Gerville-ovim jezgrom. U cilju minimiziranja MSE-a implementirane su neke prozorske funkcije. Detaljna analiza pokazala je da je optimalan izbor pravougaona prozorska funkcija. Poređenje rezultata procene sa rezultatima Keys-ovog jezgra vidi se superiornost predložene Gerville-ovog jezgra. Poređenjem dobijenih rezultata sa rezultatima procene fundamentalne frekvencije kod govornog signala koji nije komprimovan G.723.1 algoritmom MSE-a 20.3 puta. Ovaj rezultat navodi na zaključak da, i

pored toga što su subjektivnim testovima (ACR-MOS testovi) dobijeni izuzetno dobri rezultati (3.4, 3.8), kod daljih procesiranja signala, koja se baziraju na procenjenoj fundamentalnoj frekvenciji (automatska verifikacija govornika, prepoznavanje govora,...) dobijeni rezultati ne bi bili zadovoljavajući u pogledu pouzdanosti i preciznosti.

LITERATURA

- [1] S. Karapantazis, F. Pavlidou, "VoIP: A comprehensive survey on a promising technology", Computer Networks, vol. 53, pp. 2050–2090, 2009.
- [2] ITU-T Rec. G.723.1, "Dual-rate Speech Coder For Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s", 1996.
- [3] V. Reguera, F. Paliza, W. Godoy, E. Fernandez, "On the impact of active queue management on VoIP quality of service", Computer Communications, vol. 31, pp. 73–87, 2008.
- [4] A. Zolghadr, E. Goodarzi, M. Moosavinezhad, "Real-time implementation of G.723.1 speech coder", Iranian Journal of Information Science and Technology, vol.2, No.1, pp. 3-12, 2004.
- [5] B. Joen, S. Kang, S.J. Baek, K.M. Sung, "Filtering of a Dissonant Frequency Based on Improved Fundamental Frequency Estimation for Speech Enhancement", IEICE Trans. Fundamentals, vol. E86-A, No. 8, pp. 2063-2064, August 2003.
- [6] L. Qiu, H. Yang, S.N. Koh, "Fundamental frequency determination on instantaneous frequency estimation", IEEE Signal Process., vol. 44, pp. 233-241, 1995.
- [7] Z.M. Hussain, B. Boashash, "Adaptive instantaneous frequency estimation of multicomponent signals using quadratic time-frequency distributions", IEEE Trans. Signal Process. vol. 50 (8), pp. 1866–1876, 2002.
- [8] 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.
- [9] E. Meijering, M. Unser, "A Note on Cubic Convolution Interpolation", IEEE Transactions on Image Process., vol. 12, No. 4, pp. 447-479, April 2003.
- [10] Z. Milivojević, M. Mirković, Z. Milivojević, "An Estimate of Fundamental Frequency Using PCC Interpolation – Comparative Analysis", Information technology and Control, vol. 35, No. 2, pp.131-136, 2006.
- [11] Z. Milivojević, M. Mirković, P. Rajković, "Estimating of the fundamental frequency by the using of the parametric cubic convolution interpolaton", International Scientific Conference UNITECH '04, Session: Electronics and Communication Engineering, pp. 138-141, Gabrovo, Bulgaria, 2004.
- [12] Z. Milivojević, M. Mirković, "Estimation of the fundamental frequency of the speech signal modeled by the SYMPES method", Int. J. Electron. Commun. (AEÜ), vol. 63, pp. 200 – 208, 2009.
- [13] Milivojević, Z., Brodić, D., "Procena fundamentalne frekvencije govornog signala komprimovanog G.723.1 algoritmom", TELFOR'09, Sekcija: Multimedije, MM 9.48, Beograd, 2009.

PODRŠKA MREŽNIH SIMULATORA ARHITEKTURI PROTOKOLA PREPLETENIH SLOJEVA

A NETWORK SIMULATORS SUPPORT OF A CROSS-LAYER PROTOCOL ARCHITECTURE

Zoran Veličković, *Visoka tehnička škola, Niš*
Milojko Jevtović, *Jugoslovenska inženjerska akademija*

Sadržaj - U ovom radu su prezentovane najznačajnije mrežne arhitekture koje podržavaju protokole prepletenih slojeva. Analizirana je podrška mrežnih simulatora specifičnoj arhitekturi protokola prepletenih slojeva. Na bazi razmatranih arhitektura specificirani su najznačajniji zahtevi koje mrežni simulatori treba da podrže kod protokola prepletenih slojeva. Analizirani su dostupni mrežni simulatori i procenjena je njihova podrška specifičnim zahtevima međuslojne komunikacije.

Abstract – In this paper the most important network architectures with support cross-layer protocols are presented. The support of the network simulators for specific cross-layer protocol architectures is analyzed. Based on considered architecture, the significance requirements of network simulators with cross-layer protocols support are specified. Several accessible network simulators are analyzed and the specific cross-layer support is rated.

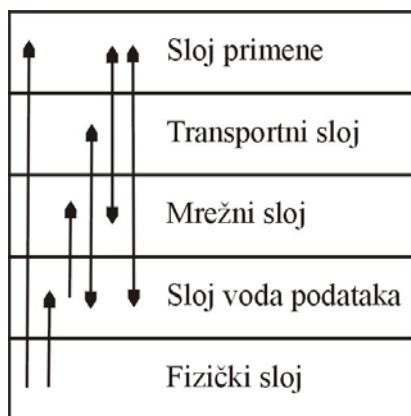
1. UVOD

Protokoli prepletenih slojeva (engl. *Cross-layer CL*) predstavljaju novu klasu komunikacionih protokola koji optimizuju mrežne parametre na više ISO/OSI slojeva. Dizajn CL protokola napušta strogu slojevitost ISO/OSI arhitekturu i razmatra mrežne parametre nezavisno od sloja na kome se vrši optimizacija [1]. Razlog napuštanja tradicionalne ISO/OSI arhitekture zapravo se nalazi u naraslim potrebama mobilnih korisnika. Zahtevi za multimedijalnim komunikacijama, a naročito u "ad hoc" mobilnim komunikacionim mrežama, ne mogu biti zadovoljeni primenom tradicionalne slojevite arhitekture protokola. U zavisnosti od tipa multimedijalne komunikacije treba obezbediti odgovarajući nivo mrežnih usluga QoS (engl. *Quality of Services*). Specifikacija zahteva za kvalitetom usluga postavlja kriterijume koje treba ispuniti u cilju kvalitetne multimedijalne komunikacije. Najčešće postavljani zahtevi multimedijalnih aplikacija su zahtevani BER (engl. *Bit Error Rate*), dozvoljeno kašnjenje u isporuci paketa, varijacija kašnjenja i zahtevani PER (engl. *Packet Error Rate*). Vrednosti pojedinih parametra QoS-a u zavisnosti od aplikacije specificirani su ITU-T preporukom G.1010. U mobilnim komunikacijama, primena standardne mrežne arhitekture ne obezbeđuje ispunjavanje postavljenih QoS zahteva. Razlog ovome leži u mobilnom mrežnom okruženju koje se karakteriše promenljivom lokacijom korisnika, pojavom fadinga u bežičnom kanalu, interferencijom zbog deljenja pristupnog medijuma i ograničenim energetskim resursima. Za zadovoljenje QoS zahteva treba obezbediti mehanizme za proizvoljne međuslojne komunikacije koje standardna arhitektura žičnih mreža ne poseduje. Međutim, potpuno redefinisane slojevite OSI mrežne arhitekture nije prihvatljivo jer bi to izazvalo velike poteškoće u primeni. Osnovni zahtev pred dizajn CL protokola je kako obezbediti međuslojnu interakciju a da se pri tome ne naruši slojevita OSI

arhitektura. Arhitektura CL protokola treba da bude transparentna sa stanovišta aplikacija i da im obezbedi odgovarajuću podršku. Mrežna razvojna okruženja su uglavnom realizovana da podrže standardno ISO/OSI okruženje. Otuda nastaju poteškoće kod dizajna i testiranja CL protokola. U ovom radu razmatrani su neki aspekti mrežnih simulacionih radnih okruženja u funkciji CL dizajna. Komparirane su mogućnosti podrške CL dizajnu ns-2 mrežnog simulacionog okruženja, OPNET-a, programskog paketa i TrueTime 1.5/2.0 simulatora baziranog na Simulink-u i Matlab-u. U delu 2 razmotrena je arhitektura CL dizajna i prikazane su tipične realizacije. U 3. delu ukratko su prikazani razmatrani programski paketi i njihova sposobnost da podrže specifičan CL dizajn. Na kraju je dat osvrt na ispunjenje zahteva CL dizajna razmatranih softverskih paketa i specificirani su neophodni koraci za standardizaciju CL dizajna.

2. STRUKTURA PROTOKOLA PREPLETENIH SLOJEVA

Klasična ISO/OSI arhitektura je dizajnirana za žično okruženje i njenom primenom u bežičnom mrežnom okruženju mogu se pojaviti mnogi novi problemi. Kod klasičnog koncepta, svakom sloju se dodeljuju određena funkcionalna zaduženja i definišu se radni mrežni parametri. Tretirajući pojedina funkcionalna zaduženja separatno, često nije moguće obezbediti zahtevane mrežne servise na traženom nivou. Bežična mrežna okruženja zahtevaju sagledavanje problema i sa aspekata kojih nije bilo prilikom definisanje klasične mrežne arhitekture. CL algoritmi podrazumevaju pristup parametrima i specifičnostima svakog ISO/OSI sloja [2]. Aplikacija ovih algoritama nije moguća u klasičnom mrežnom okruženju obzirom na slojevitost (neprepletenu) arhitekturu steka protokola. Neki od parametara koji su značajni za realizaciju algoritama prepletenih slojeva, a moraju se posmatrati združeno, su: energetska efikasnost, propusna moć mreže, vrsta i intezitet mrežnog saobraćaja, BER, PER, kapacitet kanala, spektralna efikasnost, kodiranje izvora, tip modulacije i sl.



Slika 1. Arhitektura CL dizajna sa signalizacijom između pojedinih ISO/OSI slojeva.

Implementacija CL rešenja obuhvata sve ISO/OSI slojeve počev od fizičkog sloja, pa sve do aplikativnog sloja koji obuhvata zahteve za kvalitetom servisa. Adaptacija snage predajnika, kodne tehnike i modulacione šeme na fizičkom sloju može rezultirati pružanjem boljih usluga aplikacijama. Deljenjem informacija između svih slojeva u steku protokola se obezbeđuje zajednička adaptacija mrežnih parametara prema zahtevima aplikacije odnosno QoS-a. U literaturi su opisane tri bazične arhitekture za pristup mrežnim parametrima u CL dizajnu:

- Arhitektura sa direktnom komunikacijom između slojeva [2],
- Arhitektura sa deljenom bazom podataka između slojeva [3], [4] i
- Modularna arhitektura CL dizajna sa predviđanjem odluke (engl. *Foresighted cross-layer decision*) [5].

Arhitektura sa direktnom komunikacijom između slojeva podrazumeva obezbeđenje pristupa CL optimizacionom softveru promenljivama na svim slojevima. Grafički prikaz moguće realizacije ove CL arhitekture je dat na slici 1. Ova arhitektura je primenljiva samo u slučaju kada je za potrebe realizacije CL algoritma neophodno razmeniti nekoliko poruka (parametara) između standardnih ISO/OSI slojeva. Osnovni nedostatak ovog pristupa je zahtev za poznavanjem detalja realizacije svakog sloja. Ovo je direktno u suprotnosti sa ISO/OSI principom funkcionalne nezavisnosti slojeva. Jednostavan način za aplikovanje ove mrežne arhitekture je da se za razmenu podataka između slojeva koriste zaglavlja protokola. Obzirom da svaki sloj razmatra samo podatke iz zaglavlja svog sloja, zaglavlja drugih slojeva se mogu pridružiti poruke koje nose informacije od značaja za implementaciju CL algoritma [6]. Na ovaj način podaci nesmetano prelaze postavljene granice između ISO/OSI slojeva. Ovako formiran signalizacioni kanal obezbeđuje mehanizam za realizaciju cross-layer arhitekture. Pri praktičnoj realizaciji, svakom ISO/OSI sloju se pridodaju adaptacioni softverski moduli koji prihvataju poslate poruke i prosleđuju ih CL algoritmu. Osnovni problem koji ovde treba rešiti je kompatibilnost sa standardnim

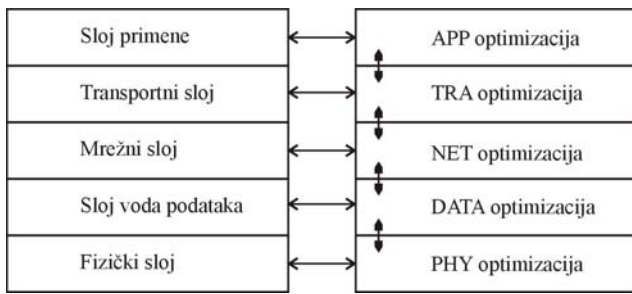


Slika 2. Arhitektura CL dizajna sa deljenom bazom podataka MobileMan [4].

stekom protokola. Potpuni redizajn standardne arhitekture nije prihvatljiv jer bi to izazvalo značajne probleme u funkcionisanju postojećih komunikacionih sistema zasnovanih na standardnoj ISO/OSI arhitekturi.

Arhitektura sa deljenom bazom podataka je zasnovana na zajedničkoj bazi svih ISO/OSI slojeva. Naime, pristup zajedničkoj bazi je omogućen svim mrežnim slojevima. Ova baza podataka se može predstaviti i kao novi komunikacioni sloj koji pruža usluge smeštanja i/ili ponovnog korišćenja podataka. CL algoritam sada može jednostavno pristupiti podacima sa svih slojeva u steku protokola i na taj način obezbediti realizaciju optimalnih protokola. Pored standardnih komunikacionih veza između susednih slojeva protokola, formiraju se nove CL veze putem razmene podataka iz deljene baze. Na slici 2 je prikazana blok šema CL arhitekture razvijene u okviru naučno-istraživačkog projekta MobileMAN [4]. Funkcionalni blok pod nazivom „Mrežni status“ je osnova ove arhitekture (sl. 2) i sadrži mrežne informacije prikupljene od svih protokola iz steka. Ova realizacija zahteva redizajn steka protokola, tako da se mrežni status čita/upisuje iz/u deljene baze.

Modularna arhitektura sa predviđanjem odluke može zadovoljiti dva osnovna zahteva CL dizajna. Prvi zahtev se odnosi na očuvanje slojevite ISO/OSI arhitekture steka protokola, dok se drugi zahtev odnosi na obezbeđivanje međuslojne komunikacija za realizaciju optimalnog CL dizajna. Blok šema ove arhitekture prikazana je na slici 3. Osnovna ideja kod ove arhitekture je da se na nivou sloja vrši autonomno odlučivanje, bez uzimanja u obzir vrednosti mrežnih parametara slojeva ispod, odnosno slojeva iznad, a da se ipak realizuje optimalni CL dizajn. Dakle, svakom mrežnom sloju je pridružen optimizacioni modul (APP, TRA, NET, DATA i PHY) koji ima zadatak da na svom domenu obavi realizaciju CL algoritma. Teorijske osnove ove arhitekture su date u radu [5]. U odnosu na prethodne mrežne arhitekture koje pretpostavljaju globalni optimizacioni algoritam, ovde se razmatra mogućnost distribuiranog CL dizajna. Delovi modularnog CL algoritma se distribuiraju svakom mrežnom sloju. Optimizujući sopstvene mrežne parametre i informišući o tome susedne slojeve, realizuje se optimalna CL arhitektura. Glavna prednost ovog pristupa nad ostalima je da ne pozleđuje slojevitost strukturu mrežne arhitekture pri realizaciji CL algoritama. Ovo je osnovni uslov efikasne primene u postojećim sistemima.



Slika 3. Arhitektura CL dizajna sa modularnom optimizacijom.

3. PODRŠKA MREŽNIH SIMULATORA ARHITEKTURI PREPELETENIH SLOJEVA

U literaturi koja se odnosi na dizajn i testiranje mrežnog okruženja mogu se naći primene nekoliko simulacionih programskih paketa. Pojedini simulacioni softverski paketi su samo komercijalno raspoloživi [7], dok su drugi besplatno dostupni [8], [9].

ns-2 mrežni simulator

U akademskim krugovima najčešće se koristi *ns-2* mrežni simulator koji predstavlja diskretni simulator događaja [8]. Ovaj simulator je našao primenu kod žičnih i bežičnih računarskih mreža tako da podržava TCP, UDP, „ruting“ i „multicast“ protokole. Mrežni simulator *ns-2* podržava i neke osnovne modele mobilnih komunikacija. Starije verzije *ns-2* simulatora su podržavale samo jednostavne modele bežičnih komunikacionih kanala. Korišćenje *ns-2* simulatora zahteva dobro poznavanje mrežne arhitekture i znanje dva programska jezika: C++ i interpretera Otcl (objektno orijentisana verzija Tcl-a). Korisnik *ns-2* simulatora kreira nove simulacione objekte kroz Otcl interpretator, a za izvođenje simulacije se mora prethodno izvršiti kompajliranje. Dakle, mrežni simulator *ns-2* koristi dva programska jezika kako bi zadovoljio postavljene zahteve pred njim [9]. Obzirom na otvoreni pristup, potencijali ovog simulacionog paketa su veliki. Međutim, pristup do njegovih resursa zahteva specijalističko znanje objektno - orijentisanog programiranja. Ovi zahtevi koji se postavljaju pred korisnikom mogu se svrstati u loše karakteristike ovog simulacionog programskog paketa. Međutim, sa druge strane omogućava se kreiranje programskih modula koji podržavaju CL dizajn. Dizajn i testiranje realizovanih modela zahteva dosta vremena i specifičnog znanja. Međutim, jednom kreirani modeli se jednostavno koriste i mogu se pridružiti biblioteci modula. Dobra strana primene ovog simulatora je što se slobodno mogu preuzeti već gotovi modeli pojedinih objekata. Dakle, može se zaključiti da *ns-2* nema specijalizovane modele neophodne za simulaciju CL dizajna, ali njegova otvorena arhitektura omogućava korisniku realizaciju svojih zamisli.

OPNET mrežni simulator

Iz klase komercijalno raspoloživih simulatora OPNET (engl. *Optimized Network Engineering Tool*) je najčešći izbor. Ovaj mrežni simulator je razvijen i za Windows i Unix platforme. Akademski verzija je raspoloživa tek od

nedavno i može se besplatno preuzeti sa adrese <http://www.opnet.com>. Akademski verzija koja je bila dostupna u vreme pisanja ovog rada ima skromne simulacione mogućnosti, a pri tome zahteva računare relativno velike snage. Osnovni nedostatak akademski verzije je odsustvo relevantne dokumentacije. Sa druge strane, profesionalna verzija sadrži izuzetno kvalitetnu dokumentaciju. OPNET se koristi za profesionalni razvoj, testiranje i određivanje mrežnih performansi. Podržava komunikacione sisteme od jednostavnih lokalnih mreža do globalnih satelitskih mreža. OPNET raspolaže velikim skupom ugrađenih simulacionih blokova. Postavljanjem i međusobnim povezivanjem simulacionih blokova na radnu površinu simulatora formira se odgovarajući simulacioni scenario. Radno okruženje je potpuno grafički orijentisano, a početni ekran akademski verzije je prikazan na slici 4. Osnovne karakteristike ovog simulacionog okruženja su:

- Hijerarhijsko građenje modela korišćenjem mrežnog editora, editora mrežnih čvorova i editora procesa;
- Izvođenje simulacije sa definisanim parametrima i alatom za otkrivanje grešaka;
- Bogata paleta alata za analizu, filtriranje i animaciju događaja.

Na fizičkom sloju implementiran je veliki broj različitih tipova bežičnih komunikacionih kanala. Omogućen je izbor modulacione tehnike, veličine propusnog opsega i paketa kao i tipa upotrebljenog kodeka. Iako se ovi parametri koriste i u CL algoritmima, direktna podrška CL dizajnu nije obezbeđena, ali je omogućena gradnja sopstvenih CL modela. Veza OPNETA i programskih jezika C/C++ dozvoljava razvoj sopstvenih simulacionih modula. Dakle, iako su potencijali ovog mrežnog simulatora veoma veliki, za CL dizajn se mora realizovati posebna biblioteka. Rad u OPNET-u je daleko konformniji od rada u okruženju *ns-2* simulatora.

TrueTime mrežni simulator

TrueTime je mrežni simulator baziran na Matlab-u [10] i Simulink-u [11]. Besplatna verzija uz registraciju ovog simulatora se može preuzeti sa URL adrese <http://control.lth.se/truetime>. Veoma važno polje primene ovog simulatora odnosi se na bežične senzorske mreže. Ove mreže su organizovane kao „ad hoc“ računarske mreže tako da ne zahtevaju mrežnu infrastrukturu [12]. Kod „ad hoc“ mreža veoma važan parametar optimizacije CL dizajna je potrošnja energije za napajanje. Funkcija izračunavanja potrošene energije je podržana ovim programskim paketom. Biblioteka raspoloživih blokova i početni ekran u Simulink-u je dat na slici 5. Korisnički definisan kernel blok, *TrueTime Kernel*, se sastoji od korisnički definisanog zadatka i prekidnih rutina. Komunikacija između mrežnih blokova je omogućena simuliranom žičnom ili bežičnom vezom. U verziji TrueTime 1.5/2.0 omogućena je upotreba standardnih mrežnih interfejsnih blokova, čime se pojednostavljuje proces mrežnog dizajna. Vreme od razvojnog do aplikativnog koda može biti veoma kratko, što ga čini veoma upotrebljivim u realnim aplikacijama. Algoritam se može kodirati Matlab kodom ili programskim jezikom C++. Ova mogućnost obezbeđuje dizajnerima izbor programskog jezika kojim se bolje služe. U osnovi, TrueTime kernel blok simulira mrežni čvor sa pripadajućim interfejsom. Ovaj blok se konfigurira inicijalizacionim skriptom koji može biti parametarizovan, čime je omogućeno da se isti skript koristi u mrežnim čvorovima različitih tipova.

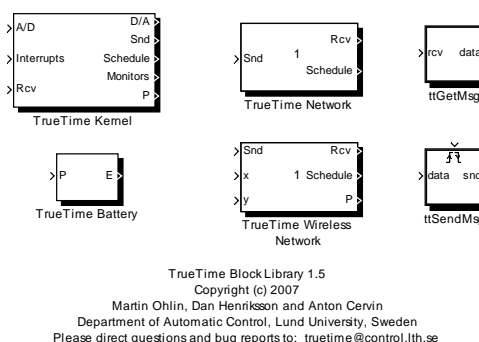


Slika 4. Početni ekran akademske verzije OPNET-a.

U inicijalizacionom skriptu korisnik može kreirati objekte poput taska, tajmera, prekidnih rutina, semafora i sl. U vreme simulacije kernel poziva kodirane funkcije taskova i/ili prekidnih rutina. TrueTime “Network blok” i TrueTime “Wireless Network” blok simuliraju fizički i sloj voda podataka ISO/OSI steka protokola [12], [13]. Podržani su različiti tipovi mreža kao što su: CSMA/CD (Ethernet), CSMA/AMP (CAN), Token Bus, FDMA, TDMA, WLAN i ZigBee. Međutim, ovim mrežnim blokovima nisu podržani protokoli višeg nivoa kao što su TCP/IP što predstavlja značajan nedostatak. Implementacija protokola mrežnog sloja se može dopisati kao aplikacija u mrežnom čvoru. Sa druge strane, konfigurisanje mrežnih blokova se obavlja preko dijalog blokova što pojednostavljuje realizaciju različitih tipova scenarija. Kod svih tipova mreža moguće je podešavati bitsku brzinu ili minimalnu veličinu frejma, dok se kod bežičnog mrežnog okruženja mogu podešavati i parametri vezani za upravljanje predajnom snagom, prag prijemnog signala, ACK tajmaut, broj retransmisija i drugo. Raspoloživi mrežni blokovi se mogu koristiti na dva osnovna načina. Prvi način podrazumeva da svi mrežni čvorovi dele isti kernel blok, dok drugi način korišćenja mrežnih blokova omogućava njihovo samostalno povezivanje, tako da ne koriste kernel blok. Međuslojna komunikacija koja je osnova CL dizajna je podržana korišćenjem mailbox-ova pridruženih svakom komunikacionom sloju. Ova činjenica je zapravo ključni faktor koji pojednostavljuje realizaciju CL dizajna i istovremeno omogućava praćenje toka međuslojnih poruka, kao i otkrivanje i eliminisanje grešaka u fazi razvoja. Iako ovaj mehanizam nije razvijen za potrebe CL dizajna, on se može uspešno primeniti za razmenu informacija između komunikacionih slojeva.

4. ZAKLJUČAK

Da bi se kreiralo kvalitetno radno okruženje za dizajn protokola prepletenih slojeva neophodno je prethodno izvršiti standardizaciju algoritama i specifikaciju transporta parametara CL dizajna. Zbog izuzetno velikog broja uticajnih faktora na CL dizajn, ni jedan od razmatranih paketa ne pruža potpuno radno okruženje za dizajn i test protokola prepletenih slojeva.



Slika 5. Biblioteka blokova TrueTime 1.5 softverskog paketa.

Orijentacija na otvorene simulacione sisteme čini se jedino mogućim rešenjem u ovom trenutku. Ovaj pristup omogućava realizaciju specifičnih CL zahteva uz dodatan napor za realizaciju specifičnih CL modula. Gotovo svi programski paketi pružaju osnovne mogućnosti za CL dizajn ali su mogućnosti veoma ograničene. Međuslojna razmena poruka putem mailbox-ova koju obezbeđuje simulacioni paket TrueTime ga izdvaja kao najprihvatljiviji izbor.

LITERATURA

- [1] V. Srivastana and M. Motani, “Cross-layer design: a survey and the road ahead,” *IEEE Commun. Mag.*, vol. 43, no. 12, pp. 112–119, Dec. 2005
- [2] L. Alonso, R. Agusti, “Optimization of wireless communication systems using cross-layer information”, *Signal Processing no. 86*, pp. 1755-1772, 2006.
- [3] B. D. Paul, J. Gomila, G. Femenias, „Cross-layer architecture design in wireless networks“, pp. 305-312, *JITEL* 2008.
- [4] MobileMan project, „Architecture, Protocols and Services“, <http://cnd.iit.cnr.it/mobile-MAN>.
- [5] F. Fu and M. Schaar, “A new theoretic foundation for cross-layer optimization,” *UCLA Technical Report*, Dec 2007.
- [6] G. Wu, Y. Bai, J. Lai, and A. Ogielski, “Interactions between TCP and RLP in wireless Internet,” in *Proc. of GLOBECOM'99*, Rio de Janeiro, Brazil, Dec. 1999, pp. 661–666
- [7] OPNET, OPNET Technologies Ltd. WWW-page, <http://www.opnet.com>.
- [8] S. McCanne and S. Floyd. ns—Network Simulator. <http://www-mash.cs.berkeley.edu/ns/>.
- [9] <http://www.isi.edu/nsnam/ns/ns-documentation>.
- [10] <http://www.mathworks.com>.
- [11] The Mathworks, „Simulink: A Program for Simulating Dynamic Systems – User’s Guide“, The MathWorks Inc., Natick, MA.
- [12] M. Jevtović, Z. Veličković, “Ad hoc mobile networks survivability and cross-layer protocol design”, *UNITECH 09*, 2009.
- [13] D. Henriksson, A. Cervin, K.-E. Årzén, “TrueTime: Real-time control system simulation with MATLAB/Simulink”, In *Proceedings of the Nordic MATLAB Conference*. Copenhagen, Denmark, 2003.
- [14] M. Ohlin, D. Henriksson, A. Cervin, K.-E. Årzén, “TrueTime 1.5/2.0 - Reference Manual, 2009.

CROSS-LAYER OPTIMIZACIJA PREDAJNE POLITIKE U KANALU SA REJLIJEVIM FEDINGOM

CROSS-LAYER TRANSMISSION POLICY OPTIMIZATION FOR RAYLEIGH FADING CHANNEL

Zoran Veličković, *Visoka tehnička škola, Niš*
Milojko Jevtović, *Jugoslovenska inženjerska akademija*

Sadržaj - U ovom radu je analizirana optimizaciona politika u funkciji tipa i nivoa fedinga u bežičnom komunikacionom kanalu. Zduženo su razmatrani adaptibilni mrežni parametri koji mogu optimizovati algoritme predajne politike u cilju maksimizacije mrežnog protoka. Kanali sa koreliranim Rejljevim fedingom i kanali sa jednako verovatnim stanjima su razmatrani na bazi FSMC modela. Dobijeni rezultati simulacije pokazuju da je nivo srednje predajne snage kod kanala sa jednako verovatnim stanjima znatno niži u odnosu na kanal sa vremenski koreliranim Rejljevim fedingom.

Abstract – In this paper the optimization policies in function of type and fading gain in wireless communication channel are analyzed. Adaptive network parameters which can optimize transmission policy algorithm are jointly considered to maximize network throughput. The channel with time correlated Rayleigh fading and channel with equiprobable states based on FSMC model are considered. The simulation results suggest that averaged transmission power of equiprobable channel states is lower than the channel with time correlated Rayleigh fading.

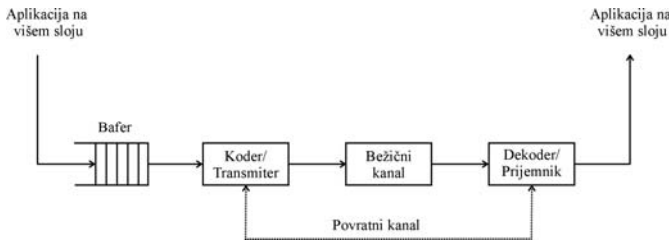
1. UVOD

Zahvaljujući svojoj fleksibilnosti i relativno niskoj ceni mrežne infrastrukture, bežične komunikacije su imale izuzetno brzi razvoj u prethodnoj deceniji. Pored pružanja standardizovanih usluga, bežične mreže su ponudile i niz novih atraktivnih servisa. Podrška mobilnim multimedijalnim aplikacijama zahteva angažovanje znatnih mrežnih resursa. Nivo kvaliteta usluga (engl. Quality of Services QoS) multimedijalnim aplikacijama je standardizovan prema tipu aplikacije. Specifikacijom G.1010, ITU-T je standardizovao *kašnjenje, varijaciju kašnjenja i gubitak informacija* kao ključne parametre koji utiču na kvalitet pruženog servisa iz perspektive korisnika [1]. Zadovoljenje postavljenog QoS-a mobilnih multimedijalnih aplikacija nije uvek jednostavan zadatak. Osnovni problemi u realizaciji zahtevanog QoS-a potiču od vremenski promenljivih karakteristika komunikacionog medijuma, interference kao posledica višekorisničkog pristupnog medijuma, fedinga, mobilnosti korisnika i dinamičke promene strukture mobilne mreže. Navedeni problemi u bežičnim mrežama nisu razmatrani pri ustanovljavanju standardnog steka ISO/OSI protokola. Ovo je razlog što pojedini protokoli iz žičnog okruženja imaju slabe performanse u bežičnom (mobilnom) okruženju. Da bi se kod mobilnih mrežnih komunikacija zadovoljili postavljeni QoS zahtevi, neophodno je izvršiti optimizaciju steka protokola za sve raspoložive mrežne resurse. Prema konceptu standardnog ISO/OSI steka, mrežni resursi su raspoređeni u pojedine slojeve i nije obezbeđena slobodna komunikacija između slojeva. Sa druge strane, optimizacija mrežnih resursa u bežičnom okruženju mora da obuhvati više mrežnih ISO/OSI slojeva. Zdužena optimizacija koja obuhvata više mrežnih ISO/OSI slojeva naziva se *cross-layer* (CL) dizajn [2].

CL algoritmi treba da reše postavljeni optimizacioni problema uz zadovoljenje niza dodatnih ograničenja. Kriterijumi pod kojima treba izvršiti optimizaciju zavise

od tipa aplikacije, ali se najčešće odnose na raspoloživu predajnu snagu mrežnih čvorova, dozvoljeno kašnjenje ili zahtevani propusni opseg. Određivanje optimalne politike slanja podataka u komunikacionom kanalu sa smetnjama je izuzetno aktuelno polje istraživanja [3]. Osnovni problemi i strukturalan pristup optimizaciji komunikacionih protokola je dat u [4]. Adaptacija brzine i snage na predajnoj strani su osnovni mehanizmi za prilagođenje komunikacionih protokola novonastalim uslovima u bežičnom kanalu. Cilj adaptivnih cross-layer optimizacionih algoritama je dostizanje Šenonovog kapaciteta komunikacionog kanala [5]. Za održavanje konstantnog BER-a (engl. Bit error rate) u uslovima fedinga je potrebno uložiti viši nivo predajne snage u odnosu na AWGN okruženje. Minimizacija predajne snage uz zadovoljavanje ostalih kriterijuma je osnovni cilj mnogih istraživanja [3], [5].

Za analizu i evaluaciju optimizacionih cross-layer algoritama neophodno je izvršiti modelovanje komunikacionog bežičnog kanala. Osnovni problem pri modelovanju bežičnog kanala je sagledavanje vremenski promenljivih karakteristika samog komunikacionog medijuma. Ovo je glavni razlog zbog čega je modelovanje bežičnih kanala veze kompleksan zadatak. Kako je već pomenuto, glavni razlozi ovome leže u zavisnosti bežičnog kanala od propagacionog modela, radijacione linije vidljivosti (engl. line of sight – LOS), refleksije od ravnih površina, difrakcije (na ivicama) i rasipanja (izazvanog objektima sa dimenzijama reda talasne dužine). Pomenuti razlozi će u realnom bežičnom okruženju prouzrokovati više prenosnih puteva između predajnika i prijemnika, što će na prijemnoj strani izazvati degradaciju prijemnog signala - feding. U zavisnosti od frekvencijskih karakteristika razlikuju se spori, brzi i „shadowing“ feding. Od predpostavljene verovatnoće raspodele amplituda prijemnog signala u kanalu sa fedingom, razlikujemo Rejljev (Rayleigh) odnosno Rician feding. Za modelovanje vremenski promenljivog komunikacionog kanala sa fedingom u ovom radu je korišćen model zasnovan na Markovljevima nizovima sa konačnim brojem stanja (engl. Finite State Markov Channel FSMC).



Slika 1. Bežični komunikacioni sistem sa jednim korisnikom i ograničenim kapacitetom bafera. Povratni kanal obezbeđuje informacije o stanju komunikacionog kanala predajnoj strani.

U ovom radu je optimizovana predana politika za vremenski korelisan i vremenski nekorelisan Rejljev kanal sa fedingom. Optimizacija predajne politike je izvršena kroz više ISO/OSI slojeva i adaptirana je tipu i nivou fedinga. Za realizaciju optimizacionog algoritma neophodno je poznavanje tipa i nivoa fedinga u komunikacionom kanalu. Nivo fedinga se može odrediti samo na prijemnoj strani, te je za dostavu informacije o trenutnom nivou fedinga potrebno obezbediti povratni komunikacioni kanal. U sekciji 2 biće dat pregled osnovnih relacija vezanih za određivanje kapaciteta komunikacionog kanala sa smetnjama. Tipovi adaptivnih modulacionih tehnika i njihove karakteristike korišćene u realizaciji optimalnih algoritama biće razmatrane u sekciji 3. Model komunikacionog kanala i rezultati optimizacionih algoritama su prezentovani u sekcijama 4 i 5. Zaključci na bazi dobijenih rezultata su sumirani u sekciji 6.

2. KAPACITET BEŽIČNOG KANALA

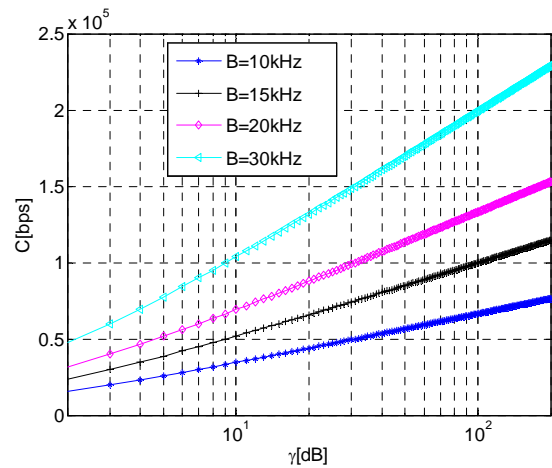
Blok šema razmatranog bežičnog komunikacionog sistema je data na slici 1. U okviru ove blok šeme, karakteristike vremenski promenljivog komunikacionog kanala sa fedingom su predstavljene u bloku „Bežični kanal“. Za kompariranje dostignutih performansi bežičnih komunikacionih kanala, neophodno je odrediti gornje granice (limite) kapaciteta ovih kanala. Limiti kapaciteta bežičnog komunikacionog kanala određuju maksimalno moguću brzinu predaje podataka uz proizvoljno malu verovatnoću greške. Ovi limiti su određeni pod pretpostavkom da nema ograničenja u kašnjenju signala ili kompleksnosti upotrebljenih algoritama i opreme. Pionirske radove iz ove oblasti je načinio Claude Shannon 1940. godine, te se ovaj limit naziva po njemu. Kapacitet diskretnog komunikacionog kanala C u prisustvu aditivnog belog Gausovog šuma AWGN (Additive White Gaussian Noise) se može odrediti na sledeći način [5]:

$$C = B \log_2(1 + \gamma) [bps], \quad (1)$$

gde je B širina propusnog opsega, dok je odnos signal šum (eng. Signal to noise SNR) na prijemu označen sa γ , a definisan je na sledeći način:

$$\gamma = \frac{P}{N_0 B}. \quad (2)$$

Snaga predanog signala je označena sa P , dok je sa N_0 označena gustina spektra snage šuma. Na osnovu nivoa fedinga, kapacitet bežičnog kanala sa fedingom $C^{(F)}$ se može odrediti na sledeći način:



Slika 2. Gornja granica kapaciteta C [bps] bežičnog komunikacionog kanala u funkciji SNR-a γ [dB].

$$C^{(F)} = \sum_{i=0}^{K-1} B \log_2(1 + \gamma_i) [bps], \quad (3)$$

gde je:

$$\gamma_i = \frac{P_i a_i p_i}{N_0 B}. \quad (4)$$

Indeks i predstavlja stanje komunikacionog kanala određeno nivoom fedinga, a_i slabljenje komunikacionog kanala, P_i angažovanu snagu predaje, a p_i verovatnoću da se kanal nađe u stanju i . Ako je definisano K stanja komunikacionog kanala, važi sledeći uslov:

$$\sum_{i=0}^{K-1} p_i = 1. \quad (5)$$

Na slici 2 prikazana je zavisnost kapaciteta bežičnog komunikacionog kanala u funkciji SNR-a na prijemnoj strani. Sa grafika se može uočiti da sa porastom odnosa signal-šum, odnosno, za bolje uslove komunikacionog kanala, raste i gornja granica kapaciteta komunikacionog kanala. Ova činjenica nalazi se u osnovi ideje o dinamičkoj adaptaciji parametara komunikacionog kanala. Naime, u zavisnosti od trenutnog SNR-a na prijemnoj strani može se adaptirati brzina prenosa u komunikacionom kanalu kako bi se dobila maksimalna propusna moć kanala. Za primenu ovih algoritama, neophodno je da predajnoj strani budu raspoloživi podaci o kvalitetu komunikacionog kanala. Podaci o stanju bežičnog komunikacionog kanala na prijemnoj strani se dostavljaju posebnim povratnim komunikacionim kanalom (sl. 1). U uslovima prisustva ostalih tipova smetnji (spori feding, brzi feding, shadowing feding), maksimalni kapacitet bežičnog komunikacionog kanala se znatno redukuje. Iz izraza (3) i (4) se može zaključiti da se kapacitet bežičnog komunikacionog kanala sa fedingom $C^{(F)}$ može dostići adaptacijom predajne snage. Sa druge strane, na dostizanje kapaciteta kanala sa fedingom može se uticati adaptacijom bitske brzine prenosa i kodne šeme. Povećanjem broja stanja komunikacionog kanala sam kanal je bolje modeliran. Međutim, sa povećanjem broja stanja kanala, kompleksnost realizacije ovih algoritama se usložnjava, tako da je često potrebno načiniti kompromis između složenosti algoritma i dostignutog kapaciteta komunikacionog kanala. Detaljna analiza kapaciteta komunikacionog kanala u uslovima različitih tipova fedinga data je u [5] i [6].

3. ADAPTIVNA MODULACIJA

Adaptivna transmisija je tehnika koja omogućava efikasnu predaju paketa u bežičnoj mreži. Adaptivnu šemu realizuje predajnik i to adaptacijom kodne strategije na bazi informacija o kvalitetu kanala sa fedingom, stanju bafera u mrežnim čvorovima i karakteristikama saobraćaja u mreži. Cilj adaptacije je da se realizuje transmisiona politika koja maksimizira brzinu pouzdanog prenosa podataka sa prihvatljivom kompromisom između kašnjenja paketa i transmisione snage. Informacije na osnovu kojih se određuje optimalna adaptivna politika se obično izračunavaju na prijemnoj strani i neophodno ih je dostaviti povratnim kanalom predajniku gde se realizuje adaptivna šema. U razmatranom komunikacionom sistemu (sl. 1) pored informacije o nivou fedinga, prenose se podaci o stanju bafera u komunikacionim čvorovima. U literaturi su opisani algoritmi koji zahtevaju od prijemnika da pošalje i informacije o uspešnosti ili neuspešnosti dekodiranja paketa, odnosno negativnu potvrdu (NAK) ako se primenjuje ARQ protokol.

Adaptivna tehnika koja optimizuje snagu predajnika i brzinu transmisije uz uvažavanje BER-a u kanalu sa fedingom koristi kvadraturnu digitalnu modulaciju QAM. Kvadraturna digitalna modulacija, QAM, je kompozitna modulaciona tehnika koja kombinuje faznu i amplitudsku modulaciju [6]. QAM definiše konstelaciju signala od $M=16$, 64 ili više tačaka, od kojih je svaka sa posebnom fazom ili amplitudom. Na ovaj način se predstavlja 4, 6 ili više bitova simbola podataka. M-QAM modulacione tehnike 16QAM ($M=2^4$) i 64QAM ($M=2^6$) se koriste u IEEE 802.11a i IEEE 802.11g standardu.

4. MODEL KOMUNIKACIONOG KANALA

U literaturi je pokazano da struktura i osobine Markovljevog niza mogu biti veoma korisne za potrebe modeliranja komunikacionog kanala sa fedingom koji aproksimira kanal konačnim brojem stanja (FSCM) [7]. Pored definisanja prostora stanja sistema, određuje se verovatnoća da se sistem nađe u tom stanju. Ova aproksimacija izvršena je diskretizacijom kontinualnog kanala sa fedingom na konačni broj stanja preko kojih se kanal može posmatrati separatno. U ovom radu se razmatra Rejljev kanal sa fedingom, uz napomenu da se samo sporo promenljiv Rejljev kanal može dobro modelovati Markovljevim nizom. Deljenjem opsega prijemnog fedinga na konačan broj nepreklopljenih opsega (K), formira se FSCM model za Rejljev kanal. Dakle, komunikacioni kanal sa fedingom je predstavljen Markovljevim nizom od K stanja, tako da svako stanje kanala predstavlja određeni opseg niva fedinga na prijemnoj strani. Pretpostavlja se da su stanja stacionarna i da je proces ergodičan. Feding u kanalu je podeljen na K nepreklopljenih sekcija korišćenjem $K+1$ nivoa:

$$0 = \gamma_0 < \gamma_1 < \dots < \gamma_{K-1} < \gamma_K \rightarrow \infty. \quad (6)$$

Komunikacioni kanal se nalazi u stanju g , $g \in \{0, 1, \dots, K-1\}$, ako je $\gamma_k \leq \gamma \leq \gamma_{k+1}$.

Za određivanje tranzicionih verovatnoća za model sa K stanja izrazi (8)-(12), treba odrediti broj preseka u jednoj sekundi (N_g) γ i γ_g [7]:

$$N_g = \sqrt{\frac{2\pi\gamma_g}{\bar{\gamma}}} f_D \exp\left(-\frac{\gamma_g}{\bar{\gamma}}\right), \quad g \in \{0, 1, \dots, K-1\}, \quad (7)$$

gde je f_D maksimum Doplerove frekvencije kanala. Nivo γ_g u odnosu na koji se određuje broj preseka, zavisi od posmatranog stanja kanala. Maksimalna Doplerova frekvencija za datu brzinu kretanja terminala v [m/s] i noseće frekvencije f je $f_D = f \cdot v/c$, gde je c [m/s] brzina svetlosti. Podrazumevajući da je feding dovoljno spor da kanal ostaje nepromenjen za vreme trajanja vremenskog frejma (T_f), a da se tranzicija stanja obavlja posle svakog frejma, i to samo između susednih stanja, verovatnoća tranzicije stanja za kanal sa K stanja se može napisati [7]:

$$p^{(G)}(g, g-1) = \frac{N_g T_f}{p^{(G)}(g)}, \quad g \in \{1, \dots, K-1\} \quad (8)$$

$$p^{(G)}(g, g+1) = \frac{N_{g+1} T_f}{p^{(G)}(g)}, \quad g \in \{0, 1, \dots, K-2\} \quad (9)$$

$$p^{(G)}(g, g) = 1 - p^{(G)}(g, g-1) - p^{(G)}(g, g+1), \quad g \in \{1, \dots, K-2\} \quad (10)$$

$$p^{(G)}(0, 0) = 1 - p^{(G)}(0, 1), \quad (11)$$

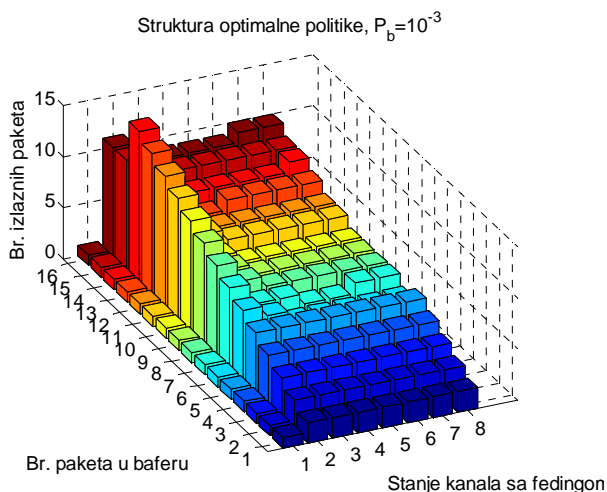
$$p^{(G)}(K-1, K-1) = 1 - p^{(G)}(K-1, K-2). \quad (12)$$

Tranziciona matrica komunikacionog kanala sa fedingom za slučaj od K stanja, P_G , se formira na osnovu izraza (8)-(12) i ima kvadratnu strukturu. Model kompletnog komunikacionog sistema (sl. 1) koji pored komunikacionog kanala obuhvata i kapacitet bafer komunikacionog čvora dat je u [8].

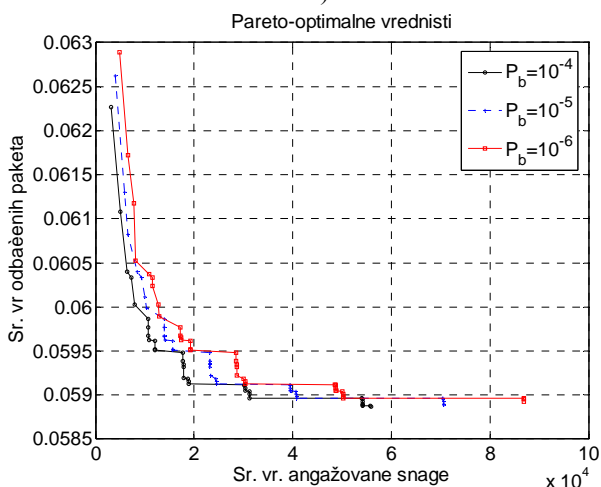
5. REZULTATI SIMULACIJE

U ovoj sekciji su prezentovani dobijeni numerički rezultati primenom optimizacionog algoritma prikazanog u [8]. Simulacioni rezultati su dobijeni za $\lambda = 3000$ paketa/s, $L = 100$ bitova u paketu, kapacitet bafer $B = 15$ paketa, $P_b = 0.001$, $W = 1000$ kHz, $N_o = 2 \cdot 10^{-5}$, vreme trajanja simbola je fiksirano na $T_s = 1/W$, a frejma je određen sa 100 simbola. Ova postavka parametra obezbeđuje da ako je konstelacija signala $M = 2^n$, bafer se prazni sa u paketa/frejmu. Kanal sa vremenski koreliranim Rejljevim fedingom je modelovan na način opisan u sekciji 4. Komunikacioni kanal je modelovan sa osam stanja koja su definisana sa nivoom fedinga na prijemniku. Tranziciona matrica P_G razmatranog komunikacionog kanala ima kvadratnu strukturu kako je to prikazano u (13). Radi komparacije transmisionih politika u funkciji tipa fedinga, razmatran je model komunikacionog kanala kod koga tranziciona matrica ima jednake tranzicione verovatnoće $p = 0.125$. Ovaj uslov obezbeđuje nekorelisanost stanja kanala sa fedingom.

$$P_G = \begin{bmatrix} 0.9359 & 0.0641 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0641 & 0.8552 & 0.0807 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0807 & 0.8334 & 0.0859 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0859 & 0.8306 & 0.0835 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0835 & 0.8420 & 0.0745 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0745 & 0.8665 & 0.0590 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0590 & 0.9048 & 0.0361 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0361 & 0.9639 \end{bmatrix} \quad (13)$$



a)



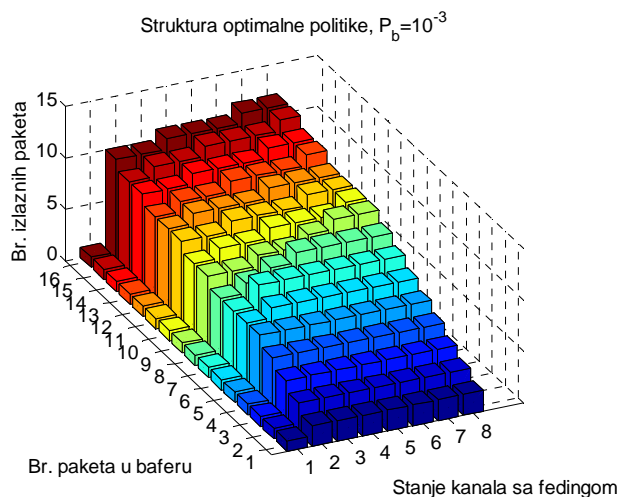
b)

Slika 3. Arhitektura CL dizajna (a) i pareto-optimalnih vrednosti za kanal sa vremenski korelisanim fedingom (b).

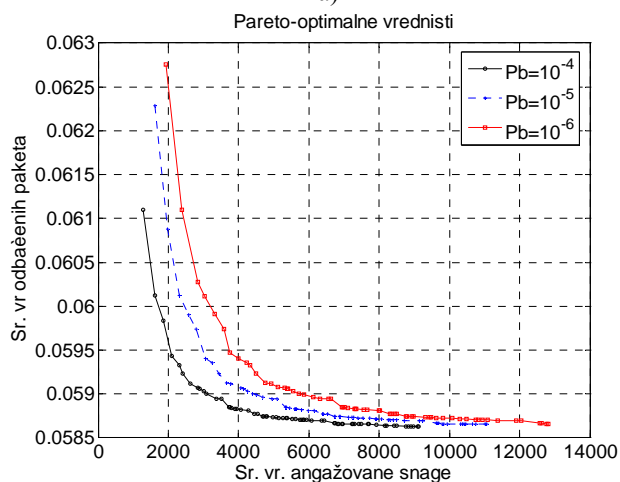
Na slikama 3a i 4a prikazani su dobijeni rezultati optimizacionog algoritama datog u [8]. Struktura optimalne politike je prikazana u funkciji broja paketa u baferu komunikacionog čvora. Na slikama 3a i 4a prikazane su strukture optimalnih transmissionih politika za kanal sa vremenski korelisanim, odnosno nekorelisanim fedingom respektivno. Na slici 3b i 4b prikazane su pareto-optimalne vrednosti broja odbačenih paketa i srednje vrednosti angažovane snage za kanal sa vremenski korelisanim, odnosno nekorelisanim fedingom respektivno za različite vrednosti BER-a. Sa ovih slika se može uočiti da za vremenski korelisan kanal sa fedingom brzina transmisije paketa ne raste uvek sa porastom kvaliteta kanala. Za razliku od kanala sa korelisanim fedingom, kanal sa nekorelisanim fedingom zahteva povećanje brzine transmisije paketa.

6. ZAKLJUČAK

Struktura optimalne transmiseone politike zavisi od tipa i nivoa fedinga u komunikacionom kanalu. U radu je pokazano da se adaptacijom transmiseone politike u funkciji tipa i nivoa fedinga mogu realizovati optimalni transmissioni algoritmi. Nivo angažovane srednje snage za vremenski nekorelisan komunikacioni kanal je znatno niži od kanala sa vremenski korelisanim kanalom sa fedingom.



a)



b)

Slika 4. Arhitektura CL dizajna (a) i pareto-optimalnih vrednosti za kanal sa vremenski nekorelisanim fedingom (b).

LITERATURA

- [1] ITU-T, "Recommendation G. 1010: End-user multimedia QoS categories", 2001.
- [2] V. Srivastana and M. Motani, "Cross-layer design: a survey and the road ahead," IEEE Commun. Mag., vol. 43, no. 12, pp. 112–119, Dec. 2005.
- [3] P. Popovski, M. A. Ingram, C. B. Peel, S. Hara, S. Toumpis, Editors, "Cross-Layer Design for the Physical, MAC, and Link Layer in Wireless Systems", EURASIP Jour. On Adv. In Sig. Proc., 2009.
- [4] M. V. Schaar, S. Shankar N, "Cross-layer wireless multimedia transmission: challenges, principles, and new paradigms", IEEE Wireless Communications, pp. 50–58, Avg. 2005.
- [5] A. Goldsmith, P. Varaiya, "Capacity of fading channels with channel side information", IEEE Trans. Inform. Theory, vol. 43, pp. 1986–1992, Nov. 1997.
- [6] A. Goldsmith, "Wireless communications", Cambridge University Press, 2005.
- [7] P. Sadehi, R. Kennedy, P. Rapajić, R. Shams "Finite-State Markov Modeling of Fading Channels", IEEE Sig. Process. Mag., vol. 25, no.5, pp. 57-80, Sep. 2008.
- [8] Z. Veličković, M. Jevtović, "Adaptivna cross-layer optimizacija zasnovana na Markovljevom procesu odlučivanja", INFOTEH 2010.

MAVEN ALAT ZA WEB JAVA EE APLIKACIJE MAVEN TOOL FOR JAVA WEB EE APPLICATIONS

Dženan Strujić, *Fakultet za informacione tehnologije, Univerzitet Mediteran, Podgorica*
Tripo Matijević, *Fakultet za informacione tehnologije, Univerzitet Mediteran, Podgorica*

Sadržaj - *Softverski sistemi i projekti vremenom postaju sve kompleksnije cjeline sa znatno većim brojem složenih zahtjeva što se istovremeno odražava na njihov razvoj i održavanje. Maven je alat za upravljanje softverskim projektima kojim se, koristeći centralizovane informacije, pomaže projektantima u razvoju i implementaciji. Maven upravlja ne samo izradom projekta već i izvještajima i dokumentacijom, i to nezavisno od platforme. U ovom radu se na praktičnom primjeru analiziraju prednosti i eventualni nedostaci upotrebe ovog alata za izradu projekta.*

Abstract - *Software systems and projects become more complex entities over time, with significantly greater number of complex requests, that simultaneously reflects on their development and maintenance. Maven is a tool for managing software projects that using centralized information helps programmers in development and implementation. Maven manages not only the creation of the project, but also manages reports and documentation, independent of the platform. This paper analyzes the practical example of the advantages and possible disadvantages of using this tool for creating the project.*

Keywords: *Maven, Project Object Model, open-source, Java*

1. UVOD

Kako su računarski sistemi, hardverski i softverski, postajali veći i kompleksniji, softverski inženjering je takođe postajao sve kompleksniji – jednostavna umjetnost „programiranja u malom“ više nije u mogućnosti da se nosi sa zadatkom. Vremenom je softver rastao sa desetina ili stotina na hiljade, sada čak milione linija koda. Razvojno okruženje je takođe pretpjelo velike promjene. Sa jednostavnih „batch“ operacija prešlo se na složene multiprogramske sisteme, distribuirana i internet okruženja. Zbog kompleksnosti sistema koji se razvijaju, i toga što proceduralno programiranje nije moglo efikasno da se nosi sa istim, bilo je neophodno uvesti nov pristup programiranju. Razvijen je objektno-orijentisani koncept (OO – Object Oriented), a radi lakše manipulacije i konfiguracije OO projekata razvijen je i objektni model projektovanja (POM – Project Object Model). [1]

Da bi se programeri lakše nosili sa razvojem aplikacija koje zahtijevaju sve veći nivo dostupnosti dijeljenja resursa, timskog rada, međuzavisnosti projekata, nezavisnog uključivanja i razmjene djelova projekata, pristupljeno je razvoju alata za automatizaciju često ponavljanih procesa – CASE alata (Computer Aided Software Engineering). Postoji više grupa CASE alata. Nekima je cilj olakšavanje projektovanja, nekima izrada projektne dokumentacije, a postoje CASE alati koji se koriste i u samom procesu programiranja.

Zbog velikog značaja posljednjih, u ovom radu ćemo opisati jedan takav alat, čija se primjena širi na sve više vrsta projekata. U pitanju je Apache-ov alat Maven v2.2. Na primjeru ćemo pokazati prednosti korišćenja Maven-a u odnosu na tradicionalni pristup projektovanju.

2. ŠTA JE MAVEN?

Apache Maven je CASE alat za upravljanje softverskim projektima. Maven je nastao izdvajanjem iz većeg Apache-ovog projekta Alexandria, avgusta 2001. godine. Zasniva se na konceptu POM modela. Uz pomoć Maven-a možemo upravljati izradom projekta, izvještajima i dokumentacijom, koristeći centralizovane informacije. Maven-ov primarni cilj je da dopusti onome ko razvija projekat uvid u kompletno stanje projekta kao i olakšanje i kraći put u njegovom razvoju. Da bi se postigao taj cilj Maven se usmjerio ka nekoliko područja interesovanja [1]:

- lako pravljenje procesa
- pružanje jedinstvenog građenja sistema
- pružanje kvalitetnih projektnih informacija
- pružanje smjernica za razvojna iskustva
- dopuštanje transparentnog migriranja na nove verzije

Kada koristimo Maven ne eliminišu se potrebe za osnovnim znanjem programiranja, ali Maven osigurava punu zaštitu od detalja. Testiranje unutar Maven-a je zasnovano na dobrom iskustvu, a Maven ga koristi kao smjernice, pa je iz tog razloga omogućeno u odvojenoj, ali paralelnoj hijerarhiji, sa Java izvornim kodom držati „test unit“ Java kod. Maven ima efikasan klijentski sistem koji pruža jednostavan način ažuriranja sopstvenih instalacija, tako da klijenti mogu automatski koristiti nove promjene koje se dešavaju u samom Maven-u. [2]

3. PRIMJENA MAVEN-A ZA RAZVOJ SOFTVERSKIH PROJEKATA

U okviru predmeta Web programiranje i Softverski inženjering, na drugoj godini osnovnih studija Fakulteta za informacione tehnologije (FIT), izučava se razvoj softverskih projekata i studenti se upoznaju sa savremenim alatima i metodologijama projektovanja. Sa ciljem da se studenti

upoznaju sa osnovnim karakteristikama i prednostima objektnog modelovanja i korišćenja CASE alata u modelovanju, programiranju, razvoju i samom upravljanju projektima, izvršen je sljedeći eksperiment – studenti su podijeljeni u dvije grupe i dobili su isti zadatak: na osnovu unaprijed prikupljenih zahtjeva i urađene analize napraviti Web orijentisanu aplikaciju koja omogućuje funkcionalnost upravljanja kupoprodajnim procesima (e-trgovina). Obije grupe su radile OO projekat, s tim da je prva grupa rješavala problem na tradicionalni način, a druga uz obaveznu upotrebu Maven-a, čime je omogućeno korišćenje novih metodologija – tzv. agilni pristup. Pri podjeli studenata po grupama, vodilo se računa o jednakom broju po grupi, kao i o jednakoj strukturi na osnovu dosadašnjeg upjeha u studiranju i pokazanog nivoa znanja.

Pošto su obije grupe dobile već prikupljene korisničke zahtjeve, glavni zadatak je bio na osnovu njih, uz određen pristupu projektovanju, proći kroz preostale faze izrade projekta:

1. dizajn
2. programiranje
3. testiranje
4. distribucija i puštanje u rad

Obije grupe su gotov projekat trebale predati najkasnije u roku od 60 dana od dana dobijanja zadatka, nakon čega se pristupilo procjeni. Svaka grupa je morala navesti poteškoće koje je eventualno imala u radu, kao i učestvovati u procjeni rada druge grupe.

Kriterijumi po kojima se vršila procjena uspješnosti izrade projekta su:

- vrijeme potrebno za prolaz kroz svaku fazu projektovanja, kao i za uspješno završavanje projekta
- mogućnost integracije djelova projekata od drugih članova tima
- stepen razmjene iskustava i znanja u okviru tima
- veličina, struktura i validnost izvornog koda
- stepen mogućnosti kasnije izmjene projekta, zavisnosti projekta
- kvalitet projektne dokumentacije
- procijenjena cijena projekta na osnovu kompleksnosti, uloženo vremena i potrebnih hardverskih resursa

3.1 POTREBNO VRIJEME

Grupa koja je radila po tradicionalnom pristupu je kompletirala projekat za 52 dana, dok je grupa koja je radila po agilnom pristupu, uz pomoć Maven-a, projekat uradila za 29 dana. U tabeli 1 je data distribucija prosječnog vremena po fazama izrade projekta.

Faza	Tradicionalni pristup (dana)	Moderni pristup (dana)
dizajn	15	9
programiranje	24	14
testiranje	10	4
distribucija	3	2
Ukupno:	52	29

Tabela 1: Potrebno prosječno vrijeme za završavanje svake faze projekta

U fazi dizajna, u odnosu na tradicionalno ručno skiciranje šema i organizacije projekta, u modernom pristupu korišćen je CASE alat StarUML, i njegov plug-in za Eclipse IDE (Integrated Development Environment – integrisano razvojno okruženje), koji olakšava sami rad generisanja interfejsa, klasa kao i projektne dokumentacije.

U fazi programiranja najbolje se mogu osjetiti prednosti Maven alata. Na tradicionalan način izrade aplikacije ručno se rade kreiranje, zavisnosti, uključenja, konfiguracije i implementacije djelova izvornog koda srodnih open-source projekata. Maven uz pomoć POM fajla omogućuje kreiranje Maven projekta i prateće dokumentacije. POM fajl je XML strukturni fajl, koji u sebi sadrži podrazumijevane vrijednosti za većinu projekata, a koje se po potrebi mogu prilagođavati konkretnom projektu. POM ima za cilj kreiranje potrebnih direktorijuma za projekat. U osnovnoj verziji kreiranja direktorijuma POM kreira izvorni direktorijum za Java klase **src/main/java**, i direktorijum za test klase u Javi **src/main/test**, dok takvo kreiranje same strukture projekta kod tradicionalnog prisupa nije moguće. Neposredno prije pokretanja projekta, Maven traži POM u datom direktorijumu, i čitanjem POM-a dobija potrebne konfiguracione podatke. U POM-u se nalaze [2]:

- projektne zavisnosti (dependencies) – spoljne biblioteke koje projekat koristi,
- šta je uključeno od proširenja razvojnog okruženja (plug-inovi),
- omogućeni načini za pokretanje projekta - ciljevi,
- izgradnja profila, i td.

Druge informacije kao što su verzija projekta, opis, programeri, mailing liste i slično, takođe mogu biti navedeni u POM-u. Ove podatke je tradicionalni pristup odvajao u posebne fajlove, koji pri tome nisu bili strukturisani prema standardnom formatu.

Faza testiranja je druga po redu na kojoj se mogu osjetiti drastične razlike u odnosu na tradicionalni pristup. U fazi testiranja grupa koja je radila na principu tradicionalnog pristupa koristila je posebne projekte u kojima su testirani djelovi aplikacije, kao i sama aplikacija. Grupa koja je koristila Maven, samo testiranje je obavila u trenutnom projektu, tačnije u direktorijumu **src/main/test**, u kojem mogu da se nalaze klase za testiranje u programskom jeziku Java. Za testiranje Java koda Maven nudi integrisano rješenje framework-a TestNG. TestNG nudi definisanje korisničkih testova koji kao ulaz mogu da očekuju predefinisane tačne korisničke podatke i obrnuto.

Faza distribucije je obuhvatala relativno isti vremenski okvir u oba slučaja, ali uz korišćenje Maven alata omogućen je automatski deploy (postavljanje na server i javno objavljivanje adrese za korisnički pristup) Web aplikacije uz korišćenje standarda po kojima trenutno većina servera funkcioniše.

3.2 MOGUĆNOST INTEGRACIJE MEĐU ČLANOVIMA TIMA

Moderni softverski projekti više nisu samo kreacije pojedinih lokalnih projektnih timova. Sa povećanjem dostupnosti, robusnosti, open-source komponenti, softverski projekti današnjice zahtijevaju dinamičnu saradnju među projektnim timovima i često zavise od mješavine globalno pravljenih i održavanih komponenti.

Grupe koje su radile po uzoru na tradicionalni pristup nisu uspjele ostvariti automatizovanu komunikaciju među članovima tima. Za kontrolu verzija softvera i revizije softvera koristili su ručno pravljenje fajl arhiva, uklapanje i kombinovanje više djelova koda tj. modula u jedan projekat. Grupa koja je radila tradicionalnim pristupom je naglasila poteškoće pri realizaciji, dok je grupa koja je imala moderni pristup mnogo lakše prihvatila izmjene u samom projektu. Uz pomoć Maven-a i POM datoteke lakše je sinhronizovati korisnike tima i dijeljenje resursa. Maven omogućava integraciju repozitorijuma („skladišta“ projekta sa svim uključenim fajlovima i zavisnostima) preko Mercurial alata. Testiranjem je ustanovljeno da tradicionalni pristup teoretski omogućava sinhronizovan rad među članovima tima, što podrazumijeva sami dogovor među članovima o podjeli zaduženja i pojedinačnom razvoju djelova projekta, ali je s praktične strane takav rad neizvodljiv. Nakon pojedinačnih razvoja djelova koda skoro je nemoguće spojiti ih u kompaktnu cjelinu bez dodatnih izmjena i dorada koje mogu da zahtijevaju podjednako vremena kao i rad pri njihovom pojedinačnom razvoju. Jasno je da ovakav rad ne samo da zahtijeva jako puno vremena, već predstavlja problem dodatnog modifikovanja već odrađenih djelova projekta i dovodi u pitanje mogućnost timske rada uopšte. Maven tenutno u svojoj drugoj generaciji verzije 2.2.1 omogućava razvoj globalnog projekta, dizajniranog po modernim shvatanjima projektovanja. Ovaj scenario timske saradnje je norma projektovanja s osnovnim komponentama koje se opslužuju putem Interneta, dok projektanti troše manje uloženog vremena, tako da mogu da zadovolje open-source ili outsource svijet.

3.3 STEPEN RAZMJENE ISKUSTAVA I ZNANJA

Razmjena iskustava i znanja u timu je izuzetno važna za usavršavanje ne samo jednog člana tima, već cijelog tima kao grupacije. Među članovima tima razmjena se odvija preko sistema za praćenje životnog ciklusa projekta, i podjednako je važna u svakoj od faza.

Kod tradicionalnog pristupa ova saradnja je moguća samo direktnom komunikacijom među članovima, dok je kod modernog pristupa, pored omogućene direktne komunikacije, zasnovana i na principu aktivnih razmjena izvornih kodova, komentara i praćenja verzija pomoću softvera za kontrolu verzija (SVC – source version control) koji u sebi sadrži i softver za praćenje životnog toka projekta.

3.4 VELIČINA, STRUKTURA I VALIDNOST IZVORNOG KODA

Veličina i sama struktura izvornog koda su veoma važni faktori u projektovanju softvera. Ukoliko izvorni kod nije

pisan po već definisanim i usvojenim anotacijama, funkcionalnost cijele aplikacije može biti ugrožena. Maven kao alat očekuje precizno definisanu anotaciju korišćenja kako bi njegova funkcionalnost bila omogućena u potpunosti. Pored koda koji je neophodan potrebno je eliminisati nepotrebne djelove koda, kako bi aplikacija postala sigurnija i otpornija na greške. Jedan od načina za to je da se koriste spoljne biblioteke klasa, čijim pozivanjem dobijamo mogućnost korišćenja unaprijed napisanog i optimizovanog koda.

Što se tiče veličine izvornog koda, moderni pristup ima sljedeći nedostatak koji se ispoljava kod manjih aplikacija: zato što koristi „dobra iskustva“ pri samom kreiranju projekta, projekat sadrži učitanne biblioteke i komponente koje se odnose na gotovo sve projekte. Međutim, kod manjih aplikacija, taj automatski dodat kod možda neće biti potreban. Pozitivna strana je što se daljim razvojem projekta veličina izvornog koda ne širi velikom brzinom, za razliku od tradicionalnog načina. Uz pomoć repozitorijuma omogućen je rad kroz revizije pa je potrebno imati samo jedan projekat na jednom mjestu, dok ovakav način nije moguće izvesti u tradicionalnom pristupu.

Repozitorijum u Maven-u se koristi za držanje razvijenih artefakta („proizvoda“ Maven-a – Java Archive i POM fajlova) i različitih tipova zavisnosti. Postoje strogo samo dvije vrste repozitorijuma: lokalni i udaljeni. Lokalni repozitorijum se odnosi na kopiju same instalacije sa udaljenog Maven repozitorijuma. Na udaljeni repozitorijum se pristupa različitim protokolima kao što je file:// i http://. Ostali udaljeni repozitorijumi mogu biti interni repozitorijumi postavljeni u posebnim datotekama ili HTTP server unutar firme, koristeći udio privatnog artefakta između timova za razvoj i za javnost. Oba repozitorijuma su strukturisana na isti način tako da se skripte mogu izvoditi na obje strane, ili se mogu sinhronizovati za offline korišćenje. U opštoj upotrebi, izgled repozitorijuma je potpuno transparentan Maven korisniku. Maven promovise ideju o korisniku lokalnog repozitorijuma u kojem su JAR-ovi, koji se mogu koristiti i za izradu drugih projekata. [3]

Stoga imamo sljedeće prednosti:

- koristimo manje prostora, jer su tipični repozitorijumi prilično veliki, a ovako se svaki JAR (Java Archive) fajl čuva na jednom mjestu i štedi prostor
- čini checkout (primjenu lokalnih izmjena na udaljenu verziju) projekta bržim
- mali stepen ažuriranja (updating)
- projekat biva brži ukoliko nema binarnih fajlova u CSV-u (Concurrent System Version).
- nema potrebe za verzijama CSV-a i drugih izvora kontrole sistema

3.5 STEPEN MOGUĆNOSTI KASNIJE IZMJENE PROJEKTA, ZAVISNOSTI PROJEKTA

Mogućnost izmjene projekta nakon implemetacije i testiranja treba omogućiti u potpunosti, a isto tako i eliminisati sve faktore koji mogu uticati na zavisnost projekta od platforme na kojoj se nalazi i sl. Jedan od načina koji Maven podstiče je i neponavljanje. Postoje okolnosti u kojima se moraju koristiti iste vrijednosti na nekoliko različitih lokacija. Maven

kroz upotrebu sopstvenih, unaprijed definisanih varijabli iz POM-a osigurava upisivanje vrijednosti samo jednom. I u tradicionalnom pristupu se može improvizovanim varijablama postići sličan cilj, ali nismo uvijek u mogućnosti da ga u potpunosti sprovedemo – ostaje primjenjiv na nivou aplikacije, ali ne i na nivou različitih lokacija.

Jedan snažan aspekt Maven-a je u rukovanju projektnim relacijama, koje uključuju zavisnosti (direktne i prelazne), nasljedstva i agregacije (kod višemodulnih projekata). Stablo zavisnosti u tim slučajevima postaje veliko i kompleksno, dešava se da verzija na jednom sistemu nije ekvivalentna verziji koja se razvija, bilo da su pogrešne verzije izdate, ili postoji protivrječnost verzije između JAR-ova koji se slično zovu. Maven rješava ovakve i slične probleme kroz zajedničke lokalne repozitorijume iz kojih se ispravno mogu povezati projekti, verzirajući ih sve. [4]

Glavnu razliku između tradicionalnog i modernog pristupa čini POM i njegovo stablo zavisnosti. Na osnovu podataka u projektu, POM zavisnosti pokušava riješiti sljedećim redoslijedom:

1. Lokalni repozitorijum je označen kao zavisnost
2. Lista udaljenih repozitorijuma je označena kao zavisnost
3. U nedostatku 1 i 2, prijavljuje se greška

U tradicionalnom pristupu, nema automatskih razrješavanja zavisnosti, već se ručno radi importovanje svih neophodnih spoljnih biblioteka. Rastom obima projekta, raste i broj direktno i indirektno zavisnih biblioteka, i veoma se usložnjava proces praćenja. Najveći problem nastaje kada se radi ažuriranje verzija nekih biblioteka, i pritom se javi nekompatibilnost novijih verzija sa već postojećim. [4]

Moderni pristup po pravilu kontaktira prve udaljene repozitorijume u drugom koraku, a nalaze se u Maven 2 centralizovanom repozitorijumu koji sadrži artefakte za najpopularnije open-source projekte i biblioteke. U slučaju in-house (zatvorenog) razvoja, može se postaviti i dodatni udaljeni repozitorijum koji sadrži artefakte iz in-house razvijenih modula.

3.6 KVALITET PROJEKTNE DOKUMENTACIJE

Kvalitet dokumentacije u današnje vrijeme nije presudan. Utisci i komentari „velikih“ programera su da dokumentaciju treba uraditi razumljivu za korisnike, ali ne i pretjerano pažnje posvetiti tome. U modernom pristupu se dobija na prvom mjestu softver koji radi, umjesto opširne dokumentacije. To znači da je dokumentacija nastala radom nekog od članova tima, ali ne svih, i većina dokumentacije je dobijena uz pomoć generatora dokumentacije. Jedan od najznačajnijih generatora dokumentacije za Java aplikacije je JavaDoc generator. Kod tradicionalnog pristupa se, nasuprot modernom, mnogo pažnje posvećuje ručnom pisanju dokumentacije.

3.7 STATISTIKA REZULTATA I PROCJENA CIJENE

Procjena svih aspekata uspješnosti projekta je obavljena tako što se za svaki kriterijum dobijao određeni broj poena. Skala je napravljena relativno, uzimajući u obzir da je projekat sa bolje ostvarenim kriterijumom dobijao 100, a projekat sa lošije ostvarenim kriterijumom dobijao proporcionalne poene. Za šest kriterijuma ocjenjivanja maksimalno je moglo da se prikupi 600 poena. Ukupan broj poena je predstavljen u tabeli 2.

Tip	Tradicionalni pristup (poena)	Moderni pristup (poena)
Vrijeme	55	100
Mogućnost integracije	10	100
Stepen razmjene iskustava	60	100
Ocjene izvornog koda	60	100
Stepen mogućnosti izmjene	20	100
Projektna dokumentacija	100	70
Ukupno:	305	570

Tabela 2: Ukupan broj poena prikupljen tokom eksperimenta

Procjena cijene projekta je urađena na principu koeficijenata, primijenjenih na stavke procjene projekta, uzevši u obzir recipročne vrijednosti dobijenih poena. Analizom rezultata po kriterijumima procjene, dobijen je odnos cijena 1.86:1, sa nižom cijenom na strani modernog pristupa.

Napomena: Radi jednostavnosti modela, u obzir je uzeto da svaki kriterijum nosi koeficijent 1.

8. ZAKLJUČAK

Maven, kao alat koji podržava objektno modelovanje i timski rad, prihvaćen je od mnogih razvojnih timova, i postao je trendom današnjice. Primjenom različitih metodologija i alata, iako su korišćeni po preporukama iz referentne literature, studenti su se i dalje susretali sa problemima u toku razvoja. Iz rezultata procjene odrađenih projekata i iskustava studenata, uvidjelo se da Maven, iako donosi velike prednosti i olakšice, još uvijek ima potrebu za usavršavanjem.

LITERATURA

- [1] Create a Customized Build Process in Maven, John Casey, August 2009
- [2] Apache Maven v2.2.1 web site: <http://maven.apache.org/>
- [3] <http://www.sonatype.com/books/maven-book/>
- [4] <http://www.maestrodev.com/better-build-maven>
- [5] <http://www.ibm.com/developerworks/edu>
- [6] <http://brettporter.wordpress.com>
- [7] <http://www.carlossanchez.eu>

SIMULACIONI MODEL ALGORITMA RUTIRANJA DIJKSTRA SA ASIMETRIČNIM CIJENAMA LINKOVA MREŽE SIMULATION MODEL OF DIJKSTRA ALGORITHM WITH ASIMETRIC NETWORK LINKS WEIGHT

Jelica Kasalica, Igor Ognjanović, Ranko Vojinović, *Fakultet za Informacione Tehnologije, Univerzitet Mediteran*

Sadržaj – U radu je dat model simulacije algoritma rutiranja Dijkstra, sa asimetričnim cijenama linkova mreže. Model je realizovan u programskom jeziku Java. Osim matematičkog modela na kojem je zasnovan simulacioni model, prikazan je i tok simulacije.

Abstract – This work presents a simulation model of Dijkstra algorithm with asymmetric network links weight. The model was implemented in the Java programming language. Besides the mathematical model, which is basic of simulation model, flow of simulation is also shown.

1. UVOD

Algoritmi rutiranja se koriste da bi se pronašao odgovarajući put paketa kroz računarsku mrežu, odnosno da bi se sačinile odgovarajuće tabele rutiranja u svakom čvoru mreže, na osnovu kojih se donosi odluka o prosleđivanju paketa. Svrha algoritma rutiranja je pronalaženje „dobre putanje“ od izvornog do određižnog rutera. Obično je dobra putanja ona koja ima najmanju cijenu.

Za formulisanje algoritama rutiranja koristi se graf, koji se sastoji od čvorova i ivica. Vezano za proces rutiranja u mreži, čvorovi grafa predstavljaju rutere - tačke u kojima se donosi odluka o prosleđivanju paketa, a ivice koje povezuju te čvorove predstavljaju fizičke linkove koji rutere povezuju. Ivica ima vrijednost koja se u literaturi označava kao cijena, i ona zavisi od parametara linka, kao što su njegova fizička dužina, brzina, propusni opseg, opterećenost, cijene iznajmljivanja i slično. Za svaki par čvorova obično postoji više putanja koje ih spajaju. Jedna od njih je putanja sa najmanjom cijenom. Problem rutiranja se svodi na određivanje te putanje.

Jedna velika klasa algoritama nazivaju se **algoritmi najkraćeg puta** (*shortest-path algorithm*). [1] Ovi algoritmi proračunavaju najmanji iznos cijene za put od polaznog do određižnog čvora u mreži, i na taj način određuju najkraći put od polaznog čvora jedne mreže, do svih ostalih čvorova u mreži. Pri tome, ukupna cijena puta definiše se kao linearna suma cijena svakog koraka (od čvora do čvora) od kojih se navedeni put sastoji.

U radu je razrađen simulacioni model, zasnovan na Dijkstrinom algoritmu [2].

U cilju objašnjenja rada ovog algoritma, uvešćemo sledeće oznake: v - naziv ili broj čvora; $D(v)$ -rastojanje od polaznog čvora, čvora v (suma cijena koje odgovaraju ivicama duž puta); $c(i,j)$ -cijena između čvora i i čvora j ; N - skup čvorova koji su uključeni u najkraći put.

Algoritam se sastoji iz dva dijela: koraka inicijalizacije i koraka koji se ponavljaju dok se algoritam ne završi.

I. *Inicijalizacija*. Na početku se usvaja da je $N = \{1\}$. Dalje, za svaki čvor v koji nije uključen u ovaj skup N a direktno je povezan sa čvorom 1 , važi da je rastojanje $D(v) = c(1, v)$. U slučaju da ne postoji direktna veza od čvora 1 do čvora v , uzima se da je $D(v) = \infty$.

II. *Koraci koji se ponavljaju*. Pronađe se čvor w koji nije uključen u skup N , za koji $D(w)$ dostiže najmanju vrijednost. Zatim se čvor w uključi u skup N . Na kraju se izračunava $D(v)$ za sve preostale čvorove koji nijesu uključeni u N , po formuli:

$$D(v) = \min[D(v), D(w) + c(w, v)] \quad (1)$$

Korak II se ponavlja sve dok se svi čvorovi ne uključe u skup N .

Cijene linkova koje povezuju čvorove mreže ne moraju biti simetrične, odnosno $c(w, v)$ ne mora biti jednako $c(v, w)$. Primjer iz prakse koji ovo ilustruje je slučaj kada cijene linkova predstavljaju intenzitet saobraćaja koji se njima prenosi. U opštem slučaju intenziteti saobraćaja između dva čvora mreže ne moraju biti jednaki u oba smjera. Ovaj slučaj usložnjava algoritam najkraćeg puta.

U literaturi se obično razrađuje Dijkstrin algoritam sa indentičnim cijenama u oba smjera linka. U ovom radu je razvijen model simulacije Dijkstrinog algoritma rutiranja, koji analizira asimetrične cijene linkova.

2. MODEL SIMULACIJE

Model je realizovan u programskom jeziku Java[3].

Simulacija se sastoji od sedam klasa. Svaka od ovih klasa je odgovorna za određeni dio simulacije, koje zajedno čine cjelinu. Klase su:

Applet.java – klasa koja kreira `JContentPane`. Ovo je osnovna podloga za kreiranje logike programa.

Documentation.java – klasa koji kreira podlogu, koja će omogućiti kreiranje dokumentacije, potrebne za pokretanje i korišćenje simulacije.

DocOptions.javas – klasa odgovorna za dodavanje padajućeg menija „Dokumentacija“. Ona sadrži nazive uputstava za korišćenje simulacije.

DocText.java – klasa naslednica TextArea klase, koja je implementirana u osnovni softverski paket programskog jezika Java. Sadrži tekstualni dio objašnjenja stavki dodatih u klasi DocOptions.java.

GraphAlgorithm.java – klasa zadužena za raspored djelova simulacije prilikom njenog pokretanja. Djelovi simulacije su opcije, dokumentacija i graf. Opcije obuhvataju dugmad za upravljanje simulacijom: *počisti*, *pokreni*, *korak*, *poništi*, *primjer*, *izlaz*. Dokumentacija objašnjava sve korake izvršavanja Dijkstra algoritma. Grafom vizualno prikazuje odvijanje svih koraka algoritma.

GraphCanvas.java – klasa koja sadrži glavnu logiku simulacije.

Options.java – klasa sadrži osnovne alate za pokretanje i zaustavljanje toka simulacije, njihov raspored i funkciju.

3. POKRETANJE TOKA SIMULACIJE

Aplikacija omogućava podešavanja parametara simulacije, pregled njnog toka i prikazivanje optimalnih ruta.

Opis rada modela se prati pokretanjem simulacije. Prije pokretanja simulacije potrebno je prikazati čvorove i veze između njih. Veze mogu biti usmjerene ili neusmjerene. Usmjerene veze povezuju jedan čvor sa drugim sa tačno definisanim pravcem kretanja, dok neusmjerene veze povezuju čvorove bez strogo definisanog smjera povezivanja. Relevantan podatak koji se odnosi na veze su njihova cijena. Prije pokretanja simulacije čvorovi se kreiraju pritiskom na lijevi taster miša na panelu za rad simulacije, dok se veze kreiraju povlačenjem miša od čvora do čvora, držeći pri tome lijevi taster miša. Na taj način kreiramo putanju i utvrđujemo njenu cijenu. Početna cijena putanje je utvrđena na vrijednost 50. Cijene nijesu statičke, već da se mogu mijenjati prije pokretanja simulacije.

Otvaranjem simulacije prikazuje se prozor, prikazan na slici 1., koji sadrži grafički korisnički interfejs kreiran u programskom jeziku Java.



Sl.1. Izgled prozora

Prozor sadrži, u desnom dijelu dugmad: *počisti*, *pokreni*, *korak*, *poništi*, *primjer* i *izlaz*. U donjem dijelu je ComboBox koji sadrži: *dodaj čvor*, *izbriši čvor*, *promeni čvr*, *početni čvor*, *uspostaviti vezu*, *promjena cijene*, *izbriši vezu*, *počisti/poništi*, *pokreni algoritam*, *korak po korak*, *primjer*, *izlaz*, *sva dokumentacija*. U centralnom dijelu prozora je prostor na kome je omogućeno kreiranje i praćenje simulacije.

Osnovna značenja dugmadi su:

Počisti – dugme koje omogućava brisanje onoga što je do tada urađeno na centralnom dijelu.

Pokreni – kada izaberemo primjer simulacije ili je sami kreiramo, pritiskom na ovo dugme obezbjeđujemo rad simulacije (ukoliko postoje veze između čvorova).

Korak – dugme koje omogućuje da simulaciju možemo pratiti po koracima. Glavna svrha je praćenje putanja (algoritma).

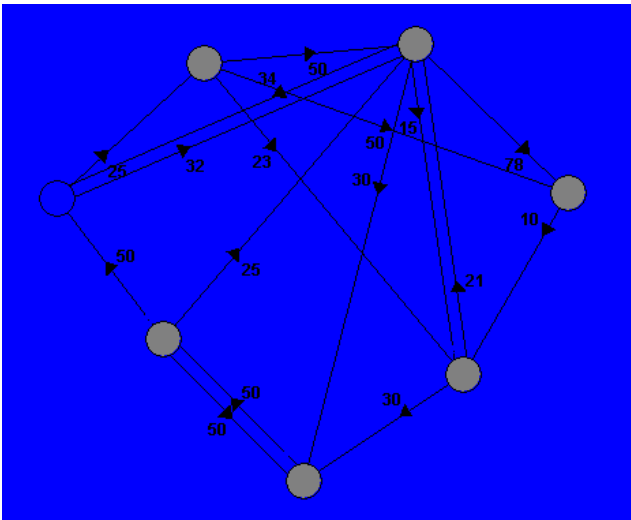
Poništi – sadrži funkciju poništavanja dotadašnjeg pokrenutog algoritma, zadržavajući kreirani graf.

Primjer- daje prikaz jednog od mogućih izgleda simulacije.

Izlaz- pritiskom na dugme zatvaramo cio prozor simulacije.

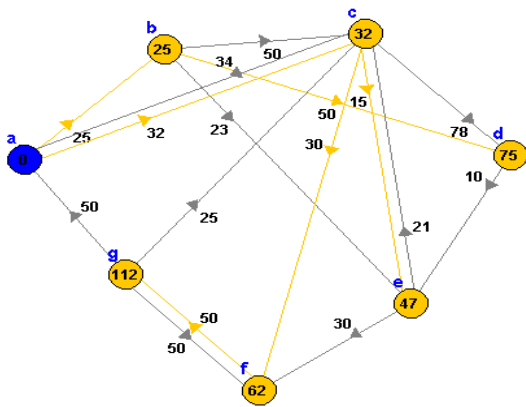
ComboBox (Dokumentacija) sadži opcije koje smo već naveli. Izborom jedne od tih opcija u TextArea prostoru prikazaće se objašnjenje načina korišćenja izabrane opcije.

Na slici 2. je prikazan izgled grafa mreže sa kreiranim čvorovima i vezama.



Sl. 2. Graf mreže

Konačni rezultat simulacije je prikazan na slici 3.



Sl. 3. Prikaz krajnjih rezultata

Plavom bojom je označen početni čvor, sive linije predstavljaju cijene putanja, dok žute označavaju optimalne putanje, dobijene simulacijom. Brojevi u čvorovima označavaju vrijednost ukupne cijene po optimalno izabranim putanjama.

Sa slike se vidi da model simulacije određuje optimalne putanje od izabranog čvora do svih ostalih čvorova u mreži.

4. ZAKLJUČAK

U literaturi [2] je napravljena simulacija modela sa najkraćom putanjom, uz pretpostavku da su veze između čvorova jednosmjerne. U ovom radu je prikazan model simulacije Dijkstrinog algoritma rutiranja sa dvosmjernim putanjama, sa asimetričnim cijenama linkova za svaki smjer. Ovakav model simulacije je bliži realnoj situaciji, jer se u računarskim mrežama inteziteti saobraćaja razlikuju po smjerovima.

Model simulacije se, za razliku od modela iz literature[2], dodatno usložnjava uvođenjem velikog broja čvorova. I na taj način se model približava realnoj situaciji. Povećanje broja čvorova, međutim, ima za posledicu povećanja obima računanja, jer je broj iteracija koje se moraju ispitati ovim algoritmom jednak $n(n+1)/2$, gdje je n broj čvorova mreže.

Cilj daljih istraživanja može biti usmjeren na smanjanje obima računanja. U tom cilju može se razviti model simulacije, zasnovan na ovom algoritmu, koji međutim, koristi njegovu složeniju implementaciju baziranu na posebnim strukturama podataka.

LITERATURA

[1] Kurose, Ros: Umrežavanje računara - Od vrha ka dnu sa internetom u fokusu, ISBN 86-7991-267-0, Računarski fakultet Beograd, CET Beograd, 2005.

[2] Robert, Lafore: Data Structures & Algorithms in Java, ISBN 0-672-32453-9, Sans Publishing 2003.

[3] H. Schildt: Java J2SE 5 kompletan priručnik, IBSN 86-7555-286-6, Mikro knjiga, Beograd 2006.

MOGUĆNOSTI PRIMJENE SERVIS-ORIJENTISANE ARHITEKTURE (SOA) U PRAKSI POSSIBILITIES OF IMPLEMENTATION OF SERVICE-ORIENTED ARCHITECTURE (SOA) IN PRACTICE

Nataša Banović, MG Soft

Ramo Šendelj, Fakultet za Informacione Tehnologije, Univerzitet 'Mediteran'

Sadržaj - Razvojem Interneta i korporativne infrastrukture, porasla su očekivanja korisnika za trenutno zadovoljenje njihovih potreba. Izvršio i klijenti podjednako zahtijevaju trenutni pristup informacijama u bilo koje vrijeme i na bilo kom mjestu. Da bi zadovoljili ova očekivanja i što prije razvili odgovarajuće softverske proizvode, organizacije, koje se bave razvojem informacionih tehnologija, morale su da osmisle šeme, koje će omogućiti ponovnu upotrebu i prilagođavanje već postojećih softverskih proizvoda. SOA je stil u arhitekturi koji se koristi za izradu preduzmljivih rješenja baziranih na servisima. SOA se bavi nezavisnim konstrukcijama poslovnih servisa, koji mogu da se kombinuju u značajne poslovne procese na visokom nivou. Značaj SOA se ogleda u kombinovanju ponovo upotrebljivih servisa i kreiranju agilnih, fleksibilnih poslovnih procesa. U ovom radu biće prikazani osnovni koncepti SOA, kao i mogući scenario njene praktične primjene za potrebe Agencije za lijekove i medicinska sredstva Crne Gore (CALIMS).

Abstract - Development of Internet and corporative infrastructure increased expectations of users in a sense of fulfilling their momentary needs. Contractors and customers equally require momentary access to information at any time and at any place. In order to fulfil those expectations and develop appropriate software products as soon as possible, organisations dealing with development of information technology had to create schemes, which will enable reuse and accommodation of already existing software products. SOA presents a style in architecture which is used for creation of efficient solutions based on services. SOA deals with independent constructions of business services which may be combined in important business processes at high level. The importance of service-oriented architecture is reflected in combination of reused services and creation of agile, flexible business processes. This paper will present the basic concepts of service-oriented architecture, as well as possible scenario of its practical application for needs of Agency for medicines and medical devices of Montenegro (CALIMS).

1. UVOD

Organizacije ulažu ogromna sredstva i troše mnogo vremena na poboljšavanje komunikacije sa svojim klijentima, ali i dalje postoje problemi u ovoj oblasti. Tehnička rješenja ne daju uvijek najbolje odgovore na tehničke probleme. Nemogućnost komunikacije doprinosi širenju problema u mnogim industrijama, pa ni farmaceutska industrija nije izuzetak. Osim toga, organizacije postaju svjesne da je fleksibilnost u poslovanju podjednako važna kao i operativna efikasnost. Naime, istraživanja Gartner-a, IDC-a i ostalih vodećih poslovnih konsultanata na svjetskom tržištu, vode do zajedničkog zaključka: kompanije troše previše novca na integraciju i održavanje postojećih poslovnih strategija, umjesto na ulaganje novca u inovacije i kreiranje novih strategija u razvoju. Zbog toga one žele da promijene nešto u poslovanju, a ove promjene su moguće uz primjenu SOA.

SOA je stil u IT arhitekturi, koji podržava transformaciju poslovnih funkcija neke organizacije u skup povezanih servisa, kojima se može pristupiti preko mreže, kad god se za tim javi potreba. Ovo može biti lokalna mreža, Internet, ili geografski i tehnološki različito iskombinovani servisi, koji se mogu nalaziti bilo gdje u svijetu. Ovi servisi omogućavaju izvršavanje specifičnih poslovnih zadataka i na taj način servisa na mreži, kakav je World Wide Web. Implementacija SOA može uključiti: razvoj aplikacija (koje

omogućavaju organizacijama da se brzo prilagode izmijenjenim uslovima i zahtjevima.

Danas je SOA široko rasprostranjena u mnogim razvijenim IT organizacijama, jer u njoj IT profesionalci vide veliki potencijal. Ona ubrzava proces razvoja softvera, omogućava razvoj aplikacija i sistema, koji se mogu prilagoditi različitim okruženjima i uslovima, i koji se mogu ponovo upotrebljavati i dograđivati, u zavisnosti od potreba krajnjih korisnika. Stoga, SOA predstavlja osnov za poboljšanja u razvoju i uvođenje inovacija u IT organizacije.

Veoma je važno posmatrati SOA kao arhitektonski model, koji je nezavisan od tehnoloških platformi. Na ovaj način, organizaciji se daje sloboda da neprestano teži ka novim stratezijskim ciljevima, koji su povezani sa servis-orijentisanim računarstvom i dostizanjem budućih tehnoloških dostignuća. Na trenutnom tržištu, Web servisi predstavljaju tehnološku platformu, koja je povezana sa realizacijom SOA.

2. OSNOVNI KONCEPTI SOA

SOA je pristup ili strategija u informacionim tehnologijama, u kojoj aplikacije koriste dostupnost koriste servise), pravljenje aplikacija (koje su dostupne kao

servisi); tako da druge aplikacije mogu da koriste njihove usluge, ili oboje.



Slika 1: SOA.

Ključni koncepti SOA, koji omogućavaju njenu primjenu su:

- Servisi
- ESB (Enterprise Service Bus)
- Labavo povezivanje (Loose coupling)
- BPM (Business process management)

2.1. SERVISI

Servis je fundamentalni koncept SOA. Neformalno, servisi se nalaze svuda oko nas i zauzimaju bitno mjesto u našem svakodnevnom životu. Svaka osoba izvršava određeni zadatak i na taj način pruža neki servis (uslugu). Slično, organizacija koja izvršava zadatke povezane sa njenim poslovanjem, takođe obezbeđuje servise. Sve dok je usluga, koja se obezbeđuje, dobro definisana i može se, relativno lako, izolovati od ostalih povezanih usluga, može se jasno klasifikovati kao servis.

U SOA svijetu i servisnoj-orijentaciji termin „servis“ nije opšti. On ima specifično značenje, koje je povezano sa jedinstvenom kombinacijom karakteristika dizajna. Kada se logika rješenja izgradi kao servis i kada se servisi dizajniraju sa ovim zajedničkim karakteristikama, tada je servisna-orijentacija uspješno realizovana u organizaciji.

U strukturi servisa moguće je odvojiti dvije ključne komponente: interfejs i implementaciju. Interfejs servisa određuje funkcije servisa, parametre koji se uzimaju kao argumenti tih funkcija i izlaze iz funkcija, i protokole, koji opisuju kako se obezbeđuju i koriste usluge servisa. Servisi obično obezbeđuju nekoliko različitih ali povezanih funkcija. Implementacija servisa opisuje načine na koje servis obezbeđuje određene usluge sadržane u interfejsu. Implementacija može biti bazirana na postojećim aplikacijama, na organizovanju servisa i kombinovanju njihovih usluga, na kodu napisanom posebno za određeni servis ili na kombinaciji svega ovog.

Veoma je važno da korisnici servisa vide samo šta servis pruža, ali ne i njegovu implementaciju. Interfejs definiše stil i detalje interakcija. Implementacija definiše način na koji proizvođač obezbeđuje određenu uslugu.

2.2. ESB (ENTERPRISE SERVICE BUS)

ESB je infrastruktura, koja omogućava upotrebu servisa na produkcionom sistemu. U zavisnosti od tehničkih i

organizacionih pristupa, ESB može biti odgovoran za: obezbeđivanje konekcije, transformaciju podataka, određivanje putanje kretanja poruka, razmatranje sigurnosti, obezbeđenje pouzdanosti, upravljanje servisima i nadgledanje rada servisa. Glavna uloga ESB-a je obezbeđivanje interoperabilnosti. Osim toga, transformacija podataka je sastavni dio ESB-a, pa ESB omogućava rješavanje problema različitosti među aplikacijama.

Tehnički i konceptualno, ESB-ovi mogu veoma da se razlikuju. Jedan od aspekata u kom se ESB tehnologije razlikuju je količina spajanja, koju obezbeđuju radi fizičkih konekcija.

Osim toga, u zavisnosti od toga da li odgovornost ESB-a potiče od provajdera (strane koja obezbeđuje servis) ili konzjumenta (strane koja traži uslugu) razlikuju se ESB-ovi usmjereni ka protokolima (protocol-driven ESB) i ESB-ovi usmjereni ka API-u. U pristupu usmjerenom ka protokolima, ESB definiše protokol, a provajder i konzjumer šalju i primaju poruke u skladu sa ovim protokolom. U pristupu usmjerenom ka API-u, ESB definiše API-ove sa specifičnom platformom, a provajderi i konzjumeri koriste API-ove za implementaciju i pozive servisa.

2.3. LABAVO POVEZIVANJE

Labavo povezivanje je ključna karakteristika SOA. Povezivanje se odnosi na nivo zavisnosti između modula, komponenti ili servisnih provajdera i konzjumenta. Labavo povezivanje servisa karakteriše dobro poznavanje zavisnosti među servisima, dok čvrsto povezani servisi posjeduju mnogo nepoznatih zavisnosti. Način povezivanja servisa utiče na cjelokupnu fleksibilnost sistema. Ukoliko je sistem čvrsto povezan, izmjene na jednom servisu zahtijevaju izmjene na ostalim servisima. Labavo povezivanje je koncept, koji se koristi kako bi se ispunili zahtjevi za skalabilnošću, fleksibilnošću sistema i tolerancijom grešaka. Glavni cilj labavog povezivanja servisa je minimizovati njihove međusobne zavisnosti.

SOA organizuje funkcionalne mogućnosti sistema u servise i ograničava povezivanje između provajdera i konzjumenta. Na ovaj način, izmjene u funkcionisanju nekog procesa utiču samo na konzjumere povezane sa servisom, koji obezbeđuje tu funkciju. Ostali konzjumeri, koji međusobno djeluju sa istim provajderom, ali preko različitih servisnih interfejsa, nisu obuhvaćeni ovim izmjenama.

2.4. BPM (BUSINESS PROCESS MANAGEMENT)

Upravljanje poslovnim procesima obično se odnosi na tehnološka sredstva, koja organizaciji omogućavaju posmatranje i upravljanje složenim procesima, koji obuhvataju širok opseg sistema i ljudi u okviru jedne ili više organizacija. Poslovni proces opisuje šta treba uraditi. On može uključivati manuelne aktivnosti i može koristiti bilo koju vrstu resursa. Upravljanje poslovnim procesima uključuje razne aspekte, kao što su: analiza poslovanja (uključujući potrebe i mogućnosti), implementacija i integracija poslovnih strategija, posmatranje i optimizacija

poslovnih procesa, uvođenje posebnih alata i kulture i usklađivanje poslovanja sa informacionim tehnologijama.

Kada govorimo BPM-u vezano za SOA, jasno je da su servisi aktivnosti dekomponovanih procesa na najnižem nivou. Postoje dva različita pristupa za dijeljenje sistema ili funkcionalnosti na servise i njihovo dizajniranje, kako bi se koristili u raznim situacijama. To su top-down pristup i bottom-up pristup. U top-down pristupu vrši se dekomponovanje problema, sistema ili procesa na manje djelove, sve dok se ne dostigne nivo osnovnih servisa. U bottom-up pristupu, poslovni proces se izgrađuje tako što se servisi sastavljaju u uopštenije djelove. U praksi se koristi kombinacija ova dva pristupa.

Oblikovanje poslovnih procesa je tehnika, koja se koristi za formalizaciju koraka poslovnih procesa i podataka vezanih za svaki od ovih koraka. Uz SOA, ovi modeli igraju važnu ulogu, zato što standardi uključuju modelovanje procesa u različitim alatima.

BPEL (Business Process Execution Language) igra veoma važnu ulogu u procesu modelovanja i projektovanja poslovnih procesa. BPEL je XML jezik za opisivanje tokova poslovanja, koji su sami po sebi servisi. Elementi jezika omogućavaju povezivanje servisa i rade sa procesnim promjenljivim, strukturama kontrola i greškama.

3. PRIMJER NA KOME SE MOŽE PRIMIJENITI SOA

Agencija za lijekove i medicinska sredstva je organizacija koja izdaje, mijenja, dopunjava i obnavlja dozvole za stavljanje lijeka u promet, izdaje dozvole za: proizvodnju lijekova, promet lijekova na veliko i promet na malo veterinarskih lijekova, izdaje odobrenje za klinička ispitivanja lijekova koji nemaju dozvolu za stavljanje lijeka u promet, evidentira klinička ispitivanja lijekova koji imaju dozvolu za stavljanje lijeka u promet i vrši kontrolu sprovođenja kliničkih ispitivanja, vrši kontrolu kvaliteta lijeka i izdaje potvrdu o kvalitetu lijeka, kao i druge aktivnosti u skladu sa Zakonom. Obavljanje ovih aktivnosti u postojećem sistemu je otežano jer podnosilac zahtjeva podnosi prijave u papirnoj formi, nakon čega referent za prijem obrazaca podatke unosi ručno. Ovakav način rada otežava posao i referentu službe za prijem i podnosiocu zahtjeva. Kako bi poboljšala poslovanje, ova organizacija se odlučuje za uvođenje novih servisa i njihovu integraciju sa postojećim. Za postizanje ovih ciljeva pogodan je SOA pristup. Kako bi smanjili složenost i obim primjera, obuhvatićemo samo mali dio aktivnosti, tj. dio poslovnih funkcija CALIMS-a. U nastavku navodimo funkcionalnost osnovnih modula:

Parametri predstavljaju modul u kom se unose, ažuriraju i brišu osnovni parametri, neophoni za funkcionisanje čitavog sistema.

Prijem je modul u kom referent za prijem dokumentacije i zahtjeva za dodatnom dokumentacijom vrši prijem obrazaca i dokumentacije i izdaje potvrdu o prijemu.

ObradaPredmeta je modul u kom referent za obradu predmeta formalno i suštinski obrađuje predmet, nakon čega donosi odluku o prihvatanju zahtjeva ili njegovom odbijanju.

Registri predstavljaju modul koji se koristi za prikaz podataka vezanih za registrovane lijekove, proizvođače, nosioce dozvola i druge podatke, koje je Agencija, po Zakonu, dužna da javno objavi.

OracleBazaPodataka je baza podataka koja sadrži informacije o čitavom informacionom sistemu.

Nakon poslovne analize i servis-orientisane dekompozicije, definišu se servisi koji će podržati glavni dio poslovnih aktivnosti CALIMS-a.

U ovom primjeru ćemo definisati servis RegistracijaLijeka, koji bi sadržao sljedeće funkcije:

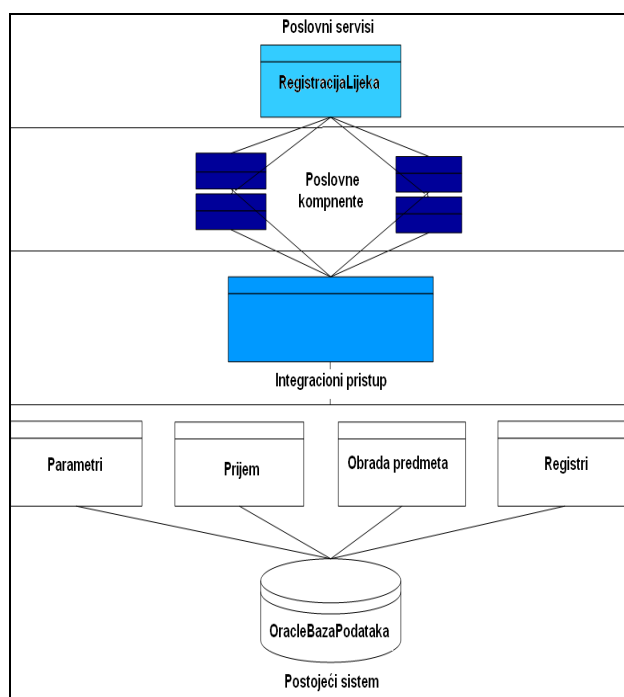
UnosProizvođača – Ova funkcija kreira novi zapis u sistemu slogova i vraća ID proizvođača.

UnosAktivneSupstance – Ova funkcija kreira novi zapis u sistemu slogova i vraća ID supstance.

UnosLijeka – Ova funkcija kreira zapis u sistemu slogova, status lijeka postavlja na Zaveden i vraća šifru i verziju unesenog lijeka.

UnosSastavaLijeka – Ova funkcija za zadati ID aktivne supstance provjerava da li ona postoji u sistemu slogova. Ako ne postoji, poziva funkciju UnosAktivneSupstance, inače unosi aktivnu supstancu u sastav lijeka i količinu, a vraća ID sastojka lijeka.

Nakon funkcionalne analize, poređenja postojeće aplikacije i novih poslovnih procesa, razmatra se mapiranje i integracija, na visokom nivou, između postojećeg i novog sistema, koja je prikazana na sl. slici:



Slika 2: Integraciona arhitektura.

Integraciona arhitektura je bazirana na povezivanju postojećeg sistema i novih poslovnih servisa. Za integraciju sa postojećim sistemom predlaže se upotreba web servisa. Web servis koristi XML tehnologije, koje su nezavisne od dobavljača, platforme i jezika, i, široko rasprostranjeni HTTP za transport podataka, što, aplikacijama koje ga koriste, omogućava nesmetanu komunikaciju sa bilo kojom drugom aplikacijom.

4. ZAKLJUČAK

SOA ne predstavlja novi koncept. To je paradigma, koja povezuje postojeće koncepte i praksu, kako bi se zadovoljio poseban skup zahtjeva.

Uvođenje SOA u organizaciju zahtijeva razmatranje mnogih aspekata. Proces započinje razumijevanjem ovog arhitektonskog stila, realnim očekivanjima, uvođenjem pravih ljudi u proces, kao i aktiviranjem i formulisanjem odgovarajućih tehnologija i procesa. SOA je mnogo više od infrastrukture. Veoma je važna dobra komunikacija među članova tima, u organizaciji u kojoj se SOA uvodi. Menadžment treba da pruži odgovarajuću podršku u ovom procesu, kao i da donese prave i odgovarajuće odluke, kada se za tim javi potreba.

Kao rješenje za distribuirane procese, SOA predstavlja odgovarajuću arhitekturu za bilo koji vid komunikacije između različitih sistema. Međutim, SOA ima i par

nedostataka. Replikacija baze podataka, masovno procesiranje podataka i lokalni klijenti ne predstavljaju snagu SOA. Prijemnom ovog arhitektonskog stila na koncepte i rješenja, organizacija može imati mnogo koristi, ali treba imati u vidu da SOA, kao i ostale tehnologije ili paradigme, nije idealna.

LITERATURA

- [1] Thomas Erl, SOA Principles of Service Design, Prentice Hall, 2008
- [2] Michael Rosen, Boris Lublinsky, Kevin T. Smith, Marc J. Balcer, Applied SOA : Service-Oriented Architecture And Design Strategies, MISL-WLY, 2008
- [3] Nicolai M. Josuttis, SOA in Practice, O'Reilly Media, 2007
- [4] <http://www.soamag.com/132/0909-3.asp>
- [5] <http://service-architecture.blogspot.com/2007/03/soa-vendor-ratings-q1-2007.html>
- [6] <http://soa.sys-con.com/node/284541#>
- [7] <http://xml.sys-con.com/node/434465#>
- [8] <http://www.soamethodology.com>

KONCEPTUALNI MODEL UVOĐENJA ICT-A U OBRAZOVNI SISTEM CONCEPTUAL FRAMEWORK FOR INTEGRATION ICT IN EDUCATION SYSTEM

Ramo Šendelj, Ivana Ognjanović, Mladen Bukilić, *Fakultet za Informacione Tehnologije, Univerzitet 'Mediterran'*

Sadržaj - *Savremeno tržište nameće potrebu za "kvalifikovanim" radnicima koji imaju sposobnosti za inovativno razmišljanje, efikasnu komunikaciju i koji su adekvatno obučeni da odgovore zahtjevima digitalnog doba. U takvim uslovima obrazovne institucije ne mogu opstati u svom tradicionalnom obliku, već se moraju prilagoditi zahtjevima globalnog tržišta i osavremeniti svoju ulogu obezbjeđujući učenicima-studentima potrebne vještine. Integracija ICT-a u obrazovni proces je ključni faktor uspješnog sprovođenja reformi obrazovnog sistema. Prikazani model uvođenja ICT-a u obrazovni sistem definiše ključne oblasti postupka integracije na makro i mikro nivou.*

Abstract - *Modern market imposes the need for "qualified" workers who have the capacity for innovative thinking, effective communication and are adequately trained to respond to demands of the digital age. In such conditions, educational institutions can not survive in its traditional form, but must be adapted to the requirements of global markets and modernize its role of providing students the necessary skills. Integration of ICT in the educational process is a key factor in successful reform of the education system. Presented model of introducing ICT in the education system defines key areas of the process of integration at the macro and micro level.*

1. UVOD

Informaciono-komunikacione tehnologije (ICT) smatraju se jednim od ključnih faktora i glavnih pokretačkih snaga promjena u savremenom globalizovanom svijetu. S jedne strane, ICT otvaraju nove perspektive za vlade, kompanije i pojedince time što omogućavaju nesmetani informacioni tok na svim nivoima, veliku transparentnost podataka, informacija, znanja i značajnu dostupnost svih djelova tržišta, kako velikim tako i onim srednjim i malim učesnicima. S druge strane, veća ili manja dostupnost i raspoloživost ICT potencijala vodi novim linijama razdvajanja između onih koji su shvatili značaj i adekvatno počeli da primjenjuju nove ICT alate i tehnologije, i onih koji to nijesu sagledali i nijesu pružili dovoljan stepen važnosti transformaciji prema društvima zasnovanim na znanju i korišćenju ICT potencijala.

2. PRIMJENA INFORMACIONO-KOMUNIKACIONIH TEHNOLOGIJA U OBRAZOVANJU

Pojava globalne ekonomije ima veliki uticaj na prirodu i namjenu obrazovnih institucija. Savremeno tržište nameće potrebu za "kvalitetnim" informacijama, a brz pristup informacijama pruža stratešku prednost. U takvim uslovima obrazovne institucije ne mogu opstati u svom tradicionalnom obliku kao puki "prenosioci propisanih skupova informacija i znanja od nastavnika do učenika na određeni period vremena". Umjesto toga, obrazovne institucije moraju osavremeniti svoju ulogu i obezbijediti vještine u cilju da se učenici-studenti nauče kako da što lakše i svrsishodnije usvoje potrebno znanje. Dakle, cilj budućeg obrazovnog sistema je "prenijeti znanje i vještine koje će omogućiti kontinuirano učenje tokom života". Futurist, Alvin Tofler [8] kaže "nepismeni u 21. vijeku neće biti samo oni koji ne znaju čitati i pisati, već i oni koji se ne mogu naučiti, koji će zaboraviti naučeno i koji ne mogu obnoviti naučeno".

Značaj ICT-a za privredu razvijenih zemalja Evrope i svijeta, značajno je uticao da sedamdesetih godina dvadesetog vijeka, obrazovni sistemi svih razvijenih zemalja uključe izučavanje ICT-a kao obavezan predmet na srednjoškolskom i visoko-školskom nivou obrazovanja. Danas je kompjuterska pismenost osnovni zahtjev za skoro svako radno mjesto, iz tog razloga su obrazovni sistemi svih zemalja svijeta uključili izučavanje ICT-a na svim nivoima obrazovanja, obezbjeđujući mladim generacijama dovoljno znanja da mogu da prate tehnološki napredak i uspješno koriste informaciono-komunikacione uređaje. Osim toga ICT su prisutne u školama i univerzitetima i kao nastavno sredstvo koje može da omogućiti nove načine za učenje, pruži radoznalost, razvija sposobnosti rješavanja problema, pruži informacije i olakša komunikaciju.

Tehnološka pismenost, odnosno sposobnost da se ICT koriste kvalitetno i efikasno, do skoro je posmatrana kao konkurentna prednost na globalnom tržištu. Danas se tehnološka pismenost ne ostvaruje samo angažovanjem ICT eksperata, već se od svih zaposlenih zahtijeva poznavanje osnovnog nivoa korišćenja ICT-a čiji kvantitet definiše savremeno globalno tržište. EnGauge iz *North Central Regional Educational Laboratory* (SAD) je definisala „Kompetencije radnika 21. vijeka“, koje uključuju: digitalnu pismenost (sastoji se od funkcionalne, vizuelne, naučne, tehnološke, informacione i kulturne pismenosti i globalne svijesti), inovativno i napredno razmišljanje, efektivnu komunikaciju i visoku produktivnost. [7]. U tabeli 1. je ukratko objašnjena svaka od vještina.

Potencijal ICT-a ogleda se u tome da one značajno pospješuju sticanje tih vještina, i to korišćenjem kao alata za podizanje kvaliteta obrazovanja i obrazovnog sistema u cjelini.

Tabela 1: Kompetentnost radnika 21. vijeka

PISMENOST DIGITALNOG DOBA	
<i>Funkcionalna pismenost</i>	Sposobnost predstavljanja informacija i znanja pomoću raznih vrsta medija: upotreba slike, grafike, video zapisa, grafičkih i vizuelnih prikaza
<i>Naučna pismenost</i>	Shvatanje teorijskih osnova i praktične primjene matematike i ostalih nauka
<i>Tehnološka pismenost</i>	Kompetencija u upotrebi ICT-a
<i>Informaciona pismenost</i>	Sposobnost pronalaženja, razvoja i adekvatnog korišćenja informacija, uključujući i upotrebu ICT-a
<i>Kulturna pismenost</i>	Poznavanje raznolikosti kultura
<i>Globalna svijest</i>	Poznavanje odnosa i međusobne sprege društva, privrede i velikih korporacija
INOVATIVNO RAZMIŠLJANJE	
<i>Adaptivnost</i>	Sposobnost adaptacije i upravljanja u svijetu kojeg karakteriše međuzavisnost
<i>Radoznalost</i>	Želja za znanjem
<i>Kreativnost</i>	Sposobnost upotrebe mašte za kreiranje novih stvari i ideja
<i>Riskiranje</i>	Spremnost podnošenja rizika
NAPREDNO RAZMIŠLJANJE	
<i>Razmišljanje višeg reda</i>	Kreativno rješavanje problema i logičko razmišljanje
EFEKTIVNA KOMUNIKACIJA	
<i>Timski rad</i>	Sposobnost rada u timu
<i>Zajednički rad</i>	Sposobnost saradnje i produktivnog rada sa ostalim članovima tima
<i>Lična i društvena odgovornost</i>	Odgovorna upotreba ICTa i obuka ostalih u cilju napretka društva i okoline
<i>Interaktivna komunikacija</i>	Odgovornost u prenosu, prijemu i tumačenju informacija
<i>Visok nivo produktivnosti</i>	Sposobnost prioritarnog upravljanja i planiranja programima i planovima u cilju postizanja postavljenih rezultata. Sposobnost primjene stečenih teorijskih znanja na realnim problemima i kreiranje odgovarajućih proizvoda visokog kvaliteta

3. RAZVOJNI KONCEPT UVOĐENJA ICT-A U OBRAZOVANJU

Obezbjedenje pristupa i efektivna upotreba ICT-a u obrazovanju predstavlja jednu od važnih oblasti u okviru otvaranja obrazovnog sistema prema svijetu i društvu znanja u kojem sposobnost nalaženja, selekcije, obrade i vrednovanja informacija postaje sve značajnija u svim oblastima učenja i primjene znanja.

Iz dosadašnje praktične primjene ICT-a u obrazovne sisteme širom svijeta, nameće se zaključak da ne postoji jedan model za određivanje optimalnog stepena integracije ICT –a u obrazovni sistem [9]. Na slici 1. prikazan je konceptualni model uvođenja ICT-a u obrazovni sistem, koji predviđa da na nacionalnom nivou postoji koncenzus o potrebi integracije ICT-a u obrazovni sistem. Na ovaj način definiše se strategije implementacije u skladu sa definisanim ciljevima nacionalnog obrazovnog sistema i preporukama Evropske Unije, na makro i mikro novou.

Za kvalitetnu integraciju ICT-a u obrazovni sistem, neophodno je obezbijediti potrebne resurse u skladu sa postojećom ICT infrastrukturom i definisanim ciljevima usvojene strategije.

Paralelno ovim procesima, neophodno je izvršiti adaptaciju postojećih kurikuluma koji će obezbijediti da učenici-studenti prvenstveno ovladaju potrebnim ICT vještinama, a zatim da ih uspješno koriste u toku učenja, za

istraživanje, prezentovanje i komunikaciju.

Za uspješnu primjenu definisanih postupaka integracije ICT-a u obrazovni proces, neophodno je uspostaviti kvalitetnu i funkcionalnu organizacionu strukturu, čija je osnovna namjena da uz tehničku podršku obrazovnim institucijama pruži okvir oko kojeg će se cjelokupni obrazovni sistem graditi.

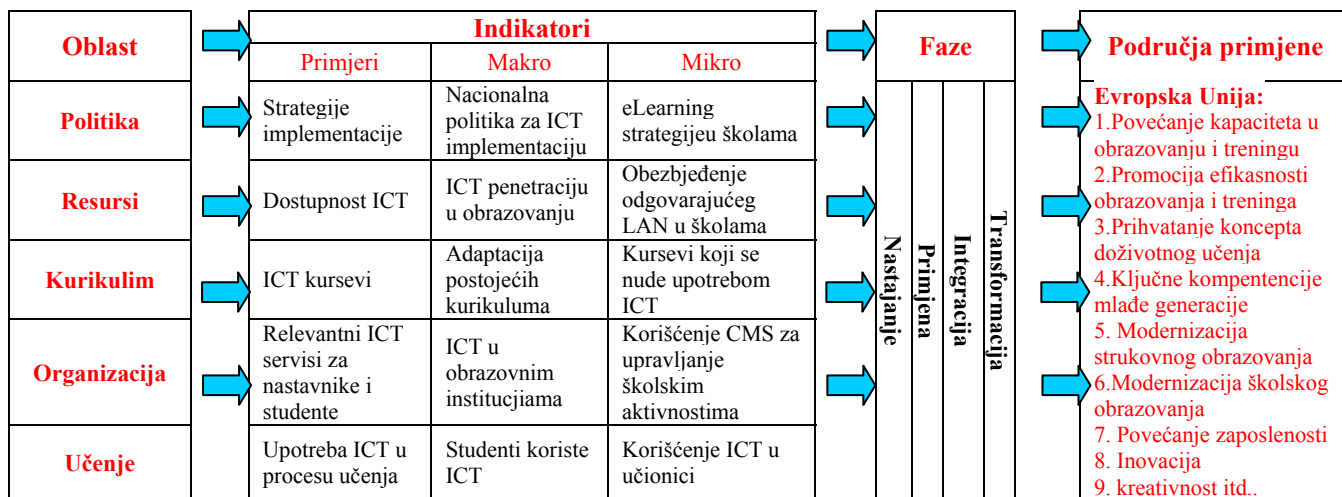
Proces učenja obuhvata aktivnu i pasivnu primjenu ICT-a. Pasivna primjena ICT-a u procesu učenja, podrazumijeva da se ICT koriste kao alat za pristup nastavnom materijalu i multimedijalni prikaz informacija, dok se aktivnim korišćenjem ICT-a podrazumijeva pronalaženje i analiza podataka, uz korišćenje specijalizovanih softverskih rješenja u cilju kvalitetnijeg rada.

Pri uvođenju ICT-a u obrazovanju detaljno treba razmotriti mogućnosti i sagledati efekte uvođenja pojedinačno u osnovnim, srednjim školama i fakultetima i na osnovu toga jasno definisati koji su ciljevi i smjernice daljeg rada i investiranja.

3.1. Osnovni nivo obrazovanja

Potencijal ICT-a za promociju kreativnosti djece i vještine rješavanja raznih problema je izuzetno razvijen ali nedovoljno korišćen. Djeca u ranom uzrastu računar koriste isključivo kao sredstvo za igru i zabavu pa u tom smjeru treba iskoristiti ICT u početnim razredima osnovne škole. Nastavnik mora biti svjestan važnosti upotrebe ICT-a ne samo kao podrške nastavi već kao podršku za unapređenje i stimulaciju interesovanja djece za učenje.

Grafika i razni efekti djeci izuzetno privlače pažnju, pa organizovanjem nastave na taj način, moguće povećati nivo interesovanja, motivisati ih za rad i održati njihovu aktivnost tokom cijelog časa. Takođe je neophodno da nastavnik koristi svoje organizacione i upravljačke vještine kako bi omogućio da svako dijete stekne kompletnost u korišćenju ICT hardvera i softvera kao i da kod svakog djeteta razvije zadovoljavajući spektar ICT vještina [3].



Slika 1. Konceptualni model uvođenja ICT u obrazovni sistem [9]

Za uspješno uvođenje i upotrebu ICT-a u nastavi neophodno je povećati nivo povjerenja nastavnika u smislu opšte nadležnosti i upotrebe ICT –a za podršku učenju i nastavi. Takođe treba obratiti pažnju i na sledeće [5]:

- Efektivno i strateški razmišljati i planirati integraciju ICT-a u školi, obezbijediti da se potencijal ICT-a na pravi način podrži, poboljša učenje i nastava realizuje u potpunosti
- U svakoj školi je neophodno sistemsko praćenje i vrednovanje razvoja ICT-a
- Posebnu pažnju treba posvetiti prilagođavanju nastavnog plana i programa i upotrebi ICT –a u redovnoj nastavi.
- Mnogi nastavnici i dalje moraju da ulažu u svoj profesionalni razvoj da bi mogli da koriste široku paletu raspoloživih softverskih alata za efikasniju podršku i poboljšanje nastave i učenja.
- Neophodno je naglasiti djeci da su im ICT vještine veoma važne za srednjoškolsko i visoko obrazovanje

3.2. Srednji nivo obrazovanja

Za razliku od osnovaca, učenici srednjih škola su već ovladali osnovnim ICT vještinama i u narednom školovanju njihovo znanje i sposobnosti samo treba još više nadograđivati i povećavati. Svaka srednja škola ima izuzetno težak zadatak za koji sama mora naći strategiju za rješavanje a odnosi se na savladavanje činjenice da učenici iz različitih osnovnih škola dolaze sa različitim stepenom prehodnog znanja i korišćenja ICT-a kako u nastavi tako i za samostalan rad i učenje.

Nijedna škola ne bi smijela krenuti od samoga početka i dio vremena i resursa usmjeriti ka osposobljavanju učenika za upotrebu ICT-a već se eventualni nedostaci i razlike u stepenu znanja i osposobljenosti moraju efektnije i u hodu rješavati.

Tempo učenja i nastavnih aktivnosti se moraju povećati, a u okviru svakog predmeta učeniku se moraju postavljati zadaci koji zahtijevaju njegovo angažovanje u smislu pronalaženja potrebnih informacija po Internetu, tumačenja i obrade tih podataka, samostalnog pisanja eseja i pripreme projekata i na kraju njihovo prezentovanje. Na taj način učenik u znatnoj mjeri proširuje svoja znanja i koristi razne

savremene softverske alate za obradu i pripremu projekata. Nastavnik svojim sugestijama i pomoći pri izradi treba kod učenika već da stvori sliku o profesionalnom odnosu ka radu i profesionalnom načinu rada i da nakon završenog projekta učenik upoređujući svoje djelo sa sličnim radovima profesionalaca može da bude ponosan na sebe svojim postignućem, da nametne kritičan stav i da sebi stvori jasnu sliku o daljem napretku i usavršavanju [4].

Pri uvođenju ICT-a u srednjim školama takođe treba posebno obratiti pažnju na:

- Efektivno i strateški razmišljati i planirati integraciju ICT-a, sistemski pratiti i vrednovati razvoj ICT-a
- Stručne srednje škole moraju prepoznati i razviti upotrebu ICT-a u svojoj oblasti i na taj način razviti NPP
- Insistirati na usavršavanju nastavnog osoblja u korištenju ICT-a
- Omogućiti učenicima on-line praćenje pojedinih nastavnih predmeta.
- Usavršavanje stranih jezika korišćenjem svaremenih softverskih rješenja i on-line resursa.

3.3 Visoki nivo obrazovanja

ICT se najviše primenjuju u visokom obrazovanju. Danas praktično svi univerziteti imaju svoje web-sajtove i aktivno koriste sisteme za upravljanje učenjem. Ipak, izuzetno je mali stepen korišćenja prednosti ICT-a u cilju preuređenja postojećih planova i programa.

Prvi pokazatelji ukazuju na neradikalne promjene, pri čemu su se ICT koristile kao podrška tradicionalnim pristupima učenju. Do 2004. godine, oko 70% univerziteta u EU imali su iskustvo sa velikim porastom u upotrebi ICT-a u nastavi. Do 2005.godine, pojedini moduli, a u nekim slučajevima cijeli programi, bili su nuđeni on-line, sa sporim pomjeranjem ka nastavnim metodama koje su više bazirane na kolaboraciji, problemima i projektima.

Danas, univerziteti širom Evrope nude širok spektar e-learning programa, a broj saradničkih projekata za projekciju i promociju inovativnih e-learning vježbi je u porastu [5]. ICT podstiče razvoj internacionalizacije visokog obrazovanja, obezbjeđujući lakšu razmjenu studenata,

nastavnika i studijskih programa.

MIT (*Massachusetts Institute of Technology*) je prvi probio led, a otvoreni obrazovni resursi (*Open Educational Resources – OER* – digitalizovani materijali koji se nude predavačima, studentima i samostalnim učenicima za upotrebu i korištenje pri predavanju, učenju i istraživanju) sada uključuju sadržaj za učenje, alate i resurse za implementaciju, uključujući prava na intelektualnu svojinu. OER se jako razvija u Evropi, pokazujući okvir za nove poslovne modele za obrazovanje i obuku[1].

Sistemi za provjeru znanja su suštinski važni za efektivno obrazovanje. Oni treba da odrede uticaj ICT-a u učenju, i da najbolje upotrebe ICT-a za provjeru znanja.

Univerziteti imaju posebnu ulogu, a i posebnu odgovornost, u razvoju baze znanja koja je potrebna za uspješnu implementaciju organizacionih inovacija u obrazovanju i obučavanju, uključujući i inteligentnu i inovativnu upotrebu ICT-a za učenje za cijeli život. Evropska Komisija takođe podržava razvoj standarda e-learninga za različita okruženja finansirajući mnoge razvojne projekte u cilju što kvalitetnije integracije ICT-a u obrazovne sisteme.

4. ZAKLJUČAK

ICT ne samo da unapređuju učenje, već imaju potencijal da transformišu procese nastave i učenja i sposobnost da ponude nove oblike obrazovanja i obučavanja uporedo sa tradicionalnim sistemom obrazovanja. Uticaj upotrebe ICT-a na učenike je usko povezan sa potencijalom ICT-a da unaprijede pristupe u nastavi i učenju. Mnoge studije su pokazale da usmjerenost ka učeniku, grupni rad i istraživački projekti rezultiraju boljim vještinama i znanjima i da interaktivni oblici elektronskog učenja mogu dovesti do reflektivnijeg, dubljeg i participativnijeg učenja.

Učenje uz rad, istraživačko učenje, rješavanje problema i kreativnost, imaju presudnu ulogu kompetentnosti za inovacije, a njihov uticaj se može bitno poboljšati i obogatiti korištenjem e-learninga.

Povećanjem uticaja ICT –a u obrazovanju i obučavanju, škole se razvijaju ka centrima za otvoreno učenje, univerziteti ka pružaocima servisa za učenje, kompanije ka organizacijama za učenje, gradovi i regioni ka sredinama koje podržavaju učenje. Promene u pedagogiji će doći sa porastom e-zrelosti.

Postavljeni ciljevi i potrebe obrazovnog sistema moraju nametnuti kvalitetno korišćenje savremenih ICT -a pa čak i

podstaći njihov razvoj. Samo na taj način obrazovne institucije zemalja u razvoju mogu odgovoriti na nove izazove i zahtjeve globalnog tržišta. Od zemalja u razvoju se ne očekuje da budu lideri u razvoju novih ICT-a koje će se primjenjivati u obrazovnom procesu ili nekih novih modela obrazovnog sistema, već da uspješno implementiraju neke od modela obrazovnog sistema razvijenih zemalja, kako bi što bolje postigli svoje ciljeve i što prije prebrodili izraženu razliku u kvalitetu obrazovnog procesa sa razvijenim zemljama.

Pred relevantnim institucijama i subjektima u Crnoj Gori je izuzetno važan zadatak: adekvatnim korišćenjem ICT-a proširiti pristup obrazovanju, značajno podignuti nivo kvaliteta obrazovanja prilagođavanjem procesa nastave i učenja potrebama stvarnog života, čime će se napraviti krupan korak prema svijetu i društvu znanja [10].

LITERATURA

- [1] The use of ICT to support innovation and lifelong learning for all - A report on progress, SEC(2008) 2629 final Commission staff working document
- [2] Using ICT in Learning and Teaching, HMIE 2004.
- [3] Measurement of the Impact of ICT on Children's Education, SITC, Moray House School of Education, University of Edinburgh, <http://www.miice.org.uk/>
- [4] Improving Scottish Education, A report by HMIE on inspection and review 2002-2005, HMIE, February 2006
- [5] Teaching Scotland's Children – A report on progress in implementing “A Teaching Profession for the 21st Century”, HMIE 2007
- [6] Framework for the evaluation of ICT in education, Peer2Peer programme, European Commission, eLearning Programme, 2006, http://insight.eun.org/shared/data/insight/documents/P2Pi_inspectorateframework.pdf
- [7] Collaboration in Teaching and Learning, Millea, J., Galatis, H. Strategic. ICT advisory service 2008.
- [8] Teacher Professional Learning: Planning for Change, Galatis, H. Williams, A. Strategic ICT advisory service 2009.
- [9] Conceptual Framework for Studying the Effects of ICT in Education, *Friedrich Scheuermann*, European Commission, Joint Research Centre Centre for Research on Lifelong Learning (CRELL) Ispra, Italy, International Expert Meeting on ICT in Education Indicators, Busan, South-Korea, 7-9 July 2009
- [10] Strategija razvoja Informacionog društva Crne Gore od 2009 do 2013, Ministarstvo za Informacionodruštvo, 2009.

PRIMJENA WEB 2.0 TEHNOLOGIJA U UČENJU USING WEB 2.0 TECHNOLOGIES IN LEARNING

Snežana Šćepanović, Fakultet za Informacione Tehnologije, Univerzitet Mediteran

Sadržaj - U radu je predstavljen novi model za kreiranje okruženja za elektronsko učenje koji se zasniva na integraciji Web 2.0 aplikacija i postojećih sistema za elektronsko učenje koji se koriste u formalnom obrazovanju na univerzitetima. Analizirane su rezultati primjene okruženja za učenje 2.0 u okviru kursa Internet tehnologije na Fakultetu za informacione tehnologije u Podgorici.

Abstract - The work presents new model for creating learning environment based on integration Web 2.0 application with existing systems used for e-learning in universities. Environment for e-learning 2.0 was tested during the course Internet technologies on the Faculty for information technology, Podgorica.

Keywords: e-learning, Web 2.0, social software, user-generated content, student-centered learning, knowledge creation metaphor of learning, higher education, blogs, wikis, social networking, learning community, pedagogy

1. UVOD

Posljednjih godina prisutna je sve veća upotreba Web aplikacija pomoću kojih korisnici kreiraju i distribuiraju multimedijalni sadržaj na Internetu. Ove Web aplikacije zajednički nazvane Web 2.0 ili druga generacija Web aplikacija [1], mogu se posmatrati kao sredstva za masovnu kolaboraciju Internet korisnika, jer omogućavaju da sami Internet korisnici međusobno sarađuju i dinamički kreiraju sadržaja Web sajta.

U tehnološkom smislu, Web 2.0 možemo posmatrati kao skup Web aplikacija za kolaborativno kreiranje i razmjenu znanja. Popularizacija i razvoj Web 2.0 aplikacija pokrenuli su nove trendove u načinu učenja i podstakli su na ponovno razmišljanje o samoj suštini učenja i njegovim osnovama. Danas, dizajniranje okruženja kao i nastavnih materijala za elektronsko učenje zahtijeva upotrebu novih tehnologija i kombinaciju različitih modela učenja sa ciljem da se omogući kvalitetno sticanje znanja prilagođeno novim načinima učenja.

2. ELEKTRONSKO UČENJE 2.0

Veliki uticaj na razvoj modernih pedagoških modela elektronske nastave i instrukcioni dizajn imale su postojeće teorije učenja (biheviorizam, kognitivizam, konstruktivizam). Postojeće teorije učenja razvijene su u vrijeme kada nisu postojale današnje Web 2.0 tehnologije, pa stoga ne objašnjavaju na zadovoljavajući način učenje u okruženju novih tehnologija. Nov pristup modelu učenja nazvan *konektivizam* predložio je Džordž Simens [4] i objavio ga 2004. godine na websajtu <http://www.connectivism.ca>. Model *učenja 2.0* opisan novom teorijom konektivizma podrazumijeva učenje kroz konstantnu izgradnu društava (network) pri čemu je kapacitet da se uči važniji od trenutnog znanja, početna tačka učenja nije sadržaj već kontakt sa osobama, grupama, nodovima za učenje, znanje posjeduje grupa a ne pojedinac. Učenje 2.0 je proces baziran više na instrukcijama nego na specifičnim metodologijama ili

modalitetima dok je naglasak na fleksibilnom, pristupačnom i ležernom radnom okruženju.

Jadin i Wageneder [3] definišu e-učenja 2.0 na sledeći način: "E-učenje 2.0 podrazumijeva upotrebu Web 2.0 aplikacija kao što su Wiki, Blog ili RSS u nastavnim aktivnostima sa ciljem da studenti doprinesu kolaborativnom radu samostalnim kreiranjem materijala za učenje koje će koristiti za postizanje ciljeva učenja. Prethodna definicija jasno ističe osnovnu karakteristiku e-učenja 2.0 – učenik postaje autonoman u sticanju novih znanja.

Projekat Evropske komisije HELIOS [5] u segmentu »E-learning za inovacije« navodi kao potencijale elektronskog učenja 2.0 sledeće:

- Neograničene mogućnosti kreiranja, uređivanja, publikovanja, razmjene, komentiranja sadržaja društvenom kolaboracijom na Internetu.
- Aplikacije i servisi bazirani na Web 2.0 tehnologijama znatno umanjuju napor za sastavljanje rješenja za elektronsko učenje.
- Trend ka nastavnom planu i programu usmjerenom na studenta i odstupanje od »hijerarhijskog načina učenja«.
- Uvrštavanje paradigme konektivizma, u kojem elektronsko učenje postaje »kreiranje mreže«.

Nova paradigma učenja ukazuje na niz promjena i postojanje inovativnih trendova u teoriji i praksi kreiranja okruženja za učenje. Gdje se paradigma učenja 2.0 primjenjuje danas? Prema činjeničkom stanju, proces učenja 2.0 primjenom Web 2.0 tehnologija više se spominje u akademskim raspravama nego što se sprovodi u praksi. Primjena učenja 2.0 je još uvijek na eksperimentalnom nivou, pri čemu se mora istaći da su dosadašnja iskustva vrlo uspješna (MIT Open CourseWare, Stanford on iTunes,

HarwardLaw School blog katalog, Universitat Oberta de Catalunya blogovi).

3. KREIRANJE OKRUŽENJA ZA UČENJE 2.0

Prema Simensu [4] ključni aspekt učenja 2.0 je učenje kroz konstantnu izgradnju mreža distribuiranih baza znanja koje kreiraju grupe (learning communities). Oslanjajući se na model učenja 2.0 koji opisuje teorija konektivizma Kerres [6] definiše osnovne zahtjeve koje mora da zadovolji okruženje za učenje 2.0. To su:

- **Otvorenost**
Okruženje za učenje mora biti otvoreno u smislu da omogućava konekcije i razmjenu znanja između članova grupe i između grupa.
- **Participacija**
Učenje 2.0 naglašava ulogu kolaborativnog rada u kreiranju čitavog iskustva učenja. Učenik više nije pasivni «primatelj» informacija. On dobija aktivnu ulogu u procesu učenja, jer se od njega očekuje da doprinese stvaranju i otkrivanju onih elemenata koji se trebaju naučiti. Učeniku je takođe data odgovornost da bira iz okruženja elemente koje treba uključiti u učenje, predlaže nastavniku dodatne elemente, traži elemente van datog okruženja, i da ih donese natrag u grupu.
- **Motivacija**
Kod učenja 2.0 od nastavnika se više ne očekuje da se ponaša kao ekspert, ili u nekim slučajevima isključivi dizajner iskustva učenja, već da nadgleda učenike i da im pruži i pomogne u izgradnji radnog okruženja u kojem se učenje odvija. Kada se iskustvo učenja završi zadatak nastavnika je da održava okruženje i pripremi ga za novu grupu učenika, ili za učenike iz prethodnih grupa.
- **Nadgledanje i evaluacija**
Nastavnik prati, pomaže, usmjerava i ocjenjuje rad učenika u okruženju za učenje.

Slijedeći preporuke teorije konektivizma i uzimajući u obzir zahtjeve za okruženje za učenje 2.0 koje definiše Kerres kreirali smo četiri funkcionalne cjeline novog modela okruženja za učenje 2.0.

1. Centar okruženja za učenje

Centar okruženja za učenje predstavlja početnu tačku za učenje. Njega čine instrukcije, aktivnosti i zadaci u okviru kursa koje kreira nastavnik. Uloga centra okruženja za učenje je da omogući nastavniku da upozna studente sa elementima okruženja za učenje, da ih usmjerava i motiviše dok oni uvode svoje elemente u okruženje za učenje i da ocjenjuje njihov rad.

2. Baza znanja

Baza znanja predstavlja skup svih resursa (text, audio, video) koje kreiraju studenti pri čemu je svaki resurs opisan metapodacima odnosno indeksiran pomoću tagova. Na taj način postaje dostupan alternativni način traganja za informacijama u odnosu na klasične Web pretraživače.

3. Dnevnik (weblog)

Dnevnik omogućava studentima da objavljuju svoja zapažanja i iskustva u radu, komentarišu ili da predlažu nove resurse za učenje (tekst, link, audio, video) uz mogućnost indeksiranja pomoću tagova.

4. Servis za obavješavanje

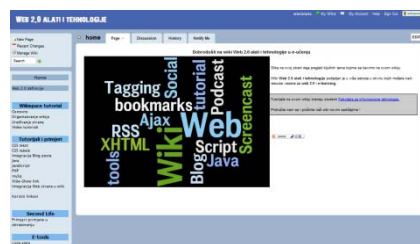
Servis za obavješavanje informiše studente o svim izmjenama u okruženju za učenje a ujedno služi i kao način za distribuciju obavještenja iz centra za učenje. Uobičajen način za implementaciju servisa za obavješavanje je RSS.

Okruženje za učenje 2.0 može biti implementirano primjenom različitih Web 2.0 aplikacija. Način implementacije prvenstveno zavisi od: cilja učenja odnosno sadržaja kursa, zatim individualnih karakteristika studenata i prethodno stečenih vještina i znanja, kao i načina na koji se koristi okruženje za učenje - kao mjesto za distribuciju informacija, kao skladište podataka, kao interaktivno okruženje, kao okruženje u kojem se dijeli znanje između većeg broja studenata.

4. IMPLEMENTACIJA OKRUŽENJA ZA UČENJE 2.0

Model okruženja za učenje 2.0 koji smo kreirali koristili smo u okviru laboratorijskih vježbi kursa „Internet tehnologije“ na Fakultetu za informacione tehnologije u Podgorici. Kurs „Internet tehnologije“ kreiran je za studente druge godine osnovnih studija. Kurs je implementiran u sistem za elektronsko učenje Fakulteta za informacione tehnologije koji se zasniva na Moodle platformi. Organizovan je u tematskom obliku, a u okviru svake teme nalaze se resursi za učenje u tekstualnom i audiovizuelnom obliku, testovi za provjeru znanja i zadaci za samostalan rad. U okviru laboratorijskih vježbi za ovaj kurs predviđeno je da se studenti upoznaju sa web 2.0 aplikacijama počev od osnovnih kao što su Blog, Wiki, RSS i Mashup, Tagging i Podcasting, pa do čitavog niza novih aplikacija i servisa koji kombinuju ili objedinju osnovne Web 2.0 tehnologije. Cilj primjene modela za učenje 2.0 bio je da studenti kroz samostalan i kolaborativan rad uz instrukcije nastavnika kreiraju bazu znanja za oblast koju proučavaju (web 2.0 aplikacije).

Centar okruženja za učenje predstavlja Moodle sistem, odnosno kurs „Internet tehnologije“ koji je implementiran u Moodle sistem. U okviru kursa za studente su kreirana uputstva za vježbe pomoću kojih su studenti upoznati sa elementima Web 2.0 okruženja za učenje. Studentima je predodčen cilj vježbi (kreiranje baze znanja) a zatim su dobili mogućnost da samostalno biraju neku od ponuđenih tema za istraživanje.



Slika 1. Baza znanja - <http://web2tools-technologies.wikispaces.com/>

Rezultate istraživanja studenti su unosili u bazu znanja. Za kreiranje baze znanja izabrana je wiki platforma, u ovom slučaju Wikispace. Wikispace softver omogućava kolaborativan rad studenata na zajedničkom projektu, definisanje tagova za resurse koje se kreiraju, diskusije i RSS.

Za kreiranje dnevnika korišćen je Blogger servis. Studenti su na svojim blogovima objavljivali iskustva u radu, analizirali Web 2.0 alate i predlagali nove teme za istraživanje. Svi elementi okruženja za učenje 2.0 koje smo koristili sadrže RSS servis što je omogućilo studentima i nastavniku da dobijaju obavještenja o svim izmjenama u okruženju za učenje.

5. ANALIZA PRIMJENE OKRUŽENJA ZA UČENJE 2.0

Fakultet za informacione tehnologije u Podgorici koristi kombinovano (blended learning) učenje koje podrazumijeva izvođenje nastave djelimično online primjenom Moodle platforme, a djelimično na tradicionalan način u f2f (*face-to-face*) okruženju ili okruženju u živo. Primjena virtuelnog okruženja za učenje u okviru kursa „Internet tehnologije” pokazala je da su studenti sa lakoćom prihvatili nov način učenja u Web 2.0 okruženju. Upotreba Wiki softvera kao platforme za kolaborativno kreiranje znanja pokazala je da su studenti visoko motivisani za samostalno učenje i istraživački rad i da uspješnije završavaju zadatke u neformalnom virtuelnom okruženju nego u učionici. Sa druge strane primjećena je slaba motivacija za upotrebu blog servisa za virtuelne asinhronne diskusije u okviru kursa. Studenti su za komunikaciju uglavnom koristili servise kao što su Skype, Facebook ili Google chat čime su sami uveli nove elemente u okruženje za učenje. Takođe, studenti su se u cilju rješavanja zadataka samostalno povezali sa novim zajednicama na mreži i razmjenjivali znanja.

U našem slučaju pokazalo se da Web 2.0 tehnologije mogu uspješno da se primjene u formalnom obrazovanju kao način za podsticanje istraživačkog rada i kreiranje inovativnih okruženja za učenje. Studenti su kolaborativnim radom kreirali indeksiranu bazu znanja koja može poslužiti sledećim generacijama za učenje i dalji razvoj. U čitavom iskustvu primjene okruženja za učenje 2.0 u formalnom obrazovanju neophodno je istaći da nastavnik ima ključnu ulogu u kreiranju i održavanju okruženja za učenje. Kako je u formalnom obrazovanju neophodno obezbijediti prisutnost učenja u virtuelnom okruženju, nastavnik iako ima ulogu moderatora okruženja za učenje, mora usmjeravati studente, davati podršku i podsticati studente da slobodno izražavaju svoje ideje u kolaborativnom kontekstu. Pokazalo se da su za uspjeh učenja u virtuelnom okruženju veoma važni predznanje i motivisanost ciljne grupe studenata za upotrebu web tehnologija u učenju, što u velikoj mjeri može da umanji angažovanost nastavnika u kreiranju inicijalne baze korisnih sadržaja i upućivanju studenata na to kako da koriste date Web resurse.

6. ZAKLJUČAK

Danas se univerziteti kao institucije visokog obrazovanja suočavaju sa brojnim izazovima u načinu primjene *novih Web 2.0 tehnologija*, servisa i aplikacija u procesu elektronskog učenja. Uprkos proklamaciji za pedagošku revoluciju, još uvijek je malo istraživanja o evaluaciji elektronskog učenja, pa tako i iskustva sa učenjem 2.0. Naprotiv, osjeća se strah ili inercija formalnog obrazovanja da u nastavne planove i programe ugradi učenje 2.0, dijelom zbog činjenice da ovaj oblik učenja nije u potpunosti shvaćen od nastavnika i instruktora. Kao i u slučaju bilo kojeg neformalnog učenja, sertifikacija i priznavanje kompetencija stečenih kroz učenje 2.0 još uvijek je otvoreno pitanje. Ključni stavovi koje treba razmotriti su kredibilitet socijalno generisanog znanja i rizik od manipulacije i ubjeđivanja lidera mlade generacije. Učenje 2.0 ima ozbiljne probleme u sferi naučne validacije samostalno kreiranog i upravljanog učenja. Štaviše, visok je rizik da »nastavni plan i program centriran na studenta«, podržan učenjem 2.0 vodi u jedan alternativni, ali još uvijek »hijerarhijski način učenja«, u kojem mali broj lidera nove generacije upravlja procesima online socijalizacije i učenja. Na kraju, nije manje značajno da tradicionalno učenje ne može (i neće) nestati iz procesa obrazovanja. Budućnost nije u tradicionalnom učenju, niti samo u elektronskom učenju, već u njihovoj kombinaciji, prilagođenoj različitim potrebama, okolnostima i učenicima [5].

LITERATURA

- [1] O'Reilly, T.: "What Is Web 2.0-Design Patterns and Business Models for the Next Generation of Software", <http://www.oreillyn.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>, 2006.
- [2] Downes, S. (2005). E-learning 2.0. preuzeto sa: <http://www.elearnmag.org/subpage.cfm?section=articles&article=29-1>, 2006.
- [3] Jadin T., Wagender G. : E-learning 2.0 – New teaching and learning culture with social software, preuzeto sa: <http://wageneder.net/artikel/finma-13.html>, 2007.
- [4] Siemens, G.: Connectivism: A Learning Theory for the Digital Age, preuzeto sa: <http://www.elearnspace.org/Articles/connectivism.htm>, 2004.
- [5] HELIOS: E-learning for Innovation. Yearly Report, preuzeto sa: <http://education-observatories.net/helios>, 2007.
- [6] Kerres, M.: Microlearning as a challenge to instructional design, preuzeto sa <http://mediendidaktik.uni-duisburg-essen.de/system/files/Microlearning-kerres.pdf>, 2009

PRIMJENA KOGNITIVIH MAPA U VISOKOM OBRAZOVANJU APPLICATION OF COGNITIVE MAPS IN HIGH EDUCATION

Sonja Pravilović, Borislavka Golović Kažić, *Fakultet za informacione tehnologije, Univerzitet „Mediteran“*

Sadržaj - Kognitivne karte (kognitivne mape, koncept karte, mape uma, mentalne mape) su grafički alati za organizovanje i predstavljanje znanja. Oni uključuju koncepte, obično zatvorene u krug ili neku vrstu kutija znanja određene vrste, gdje su odnosi između pojmova označeni spajanjem linija koja povezuje dva pojma. Kognitivne mape predstavlja grafički prikaz gradiva kojeg se želi naučiti ili predstaviti, a crtaju se u obliku krošnje drveta, u raznim bojama i slikama. Praveći mape otkrivaju se bitna svojstva riječi i pojmova, a njima se služe profesori i studenti pri učenju o međusobnom odnosu tih pojmova, stvarajući nove asocijacije na njihova značenja i time omogućava dugotrajno pamćenje.

Ključne riječi: kognitivne mape, visokoobrazovanje

Abstract - Cognitive cards (cognitive maps, concept of card, mind map, mental maps) are graphical tools for organizing and representing of knowledge. They include concepts, usually closed in a circle or certain boxes of knowledge of a specific kind, where relationships between terms are marked by connection of lines that link two terms. Cognitive maps are graphical overview of the program that one wants to learn or present, and it is often drawn as tree branches, in different colors and pictures. By making these maps, important properties of words and terms are revealed. Professors and students are using these maps for learning about relationships of these same terms, creating new associations to their meanings and by that enabling long-term memory.

Key words: cognitive maps, high education

1. UVOD

Kognitivne mape (mape uma, umne karte) spadaju u vještine organizacije teksta, slika, pojmova, formula, web strana, file-ova na računaru, i dio su vještina samostalnog učenja. Kognitivne mape ili popularno zvane mentalne mape (engl. mind mapping) su tehnike osmišljene za lakše organizovanje čitanja, učenja ili rada. Pripadaju grupi mnemotehnika – koje služe zadržavanju materijala koji bi inače najvjerojatnije zaboravili. Mentalne – kognitivne mape su pogodne za pomoć i studentima i profesorima u samoorganizaciji vremena, kao i pri učenju u organizaciji pojedinih dijelova ili sveukupnog gradiva.

Mape uma direktno utiču na znanje, tj. unaprjeđuju ga na najbolji mogući način. Mape kao poseban način organizovanja stotina podataka i informacija dio su vještina samostalnog učenja. Spadaju u kategoriju vještina kojima se organizuje i elaborira udžbenički tekst, znanje koje je sačuvano na web-u, file-ovima računara i sl. Nazivaju se još i mapama uma ili kognitivnim mapama. Spominju se danas u mnogim knjigama kao put do uspješnijeg učenja.

Tehnika se bazira na naučnim saznanjima o tome kako radi mozak, na koji način čuva i obrađuje informacije, te kako se u pamćenju informacije i podaci zadržavaju i na koji način ih se kasnije čovjek prisjeća.

Kognitivna mapa je, prije svega, tehnika učenja kojom se na najjednostavniji način u mozak unose informacije, a omogućuje i da se kasnije vrlo uspješno student i učenici koriste tim informacijama. Kognitivne mape djeluju na prirodan način rada ljudskog mozga koji po svojoj prirodi nije linearan, svaka ideja koja se javi u ljudskom mozgu odvija se u slikama, a svaka slika ima stotine veza u pamćenju. Mozak pamti ključne riječi i slike, a ne rečenice.

Kako kognitivne mape više vizuelno prikazuju veze između ključnih riječi, njih se mnogo lakše prisjetiti nego linearnih bilješki. Kognitivne mape su izraz radijalnog načina razmišljanja (sve polazi iz centra) i to je prirodni način funkcionisanja ljudskog mozga. Uz pomoć kognitivnih mapa svaka nova ideja *zakači* se za sve one koje već postoje. Što postoji više tih *udica* to je lakše uloviti svaku informaciju koja korisniku treba.

Novi pojmovi dakle, bivaju zapamćeni u asocijativnom slijedu slika, riječi i značenja koje je profesor ili student sam stvorio praveći jednu takvu mapu.

Na ovaj način stvara se struktura gradiva kao šeme, ponekad i prave beskrajne baze znanja čije su konture sami osmislili. Naučeno gradivo postaje aktivni dio znanja, a ne samo hrpa informacija pohranjena u pamćenju koje će se aktivirati samo na nivou površnog prepoznavanja.

Kognitivne mape prirodni su sistem organizacije informacija. One su sredstvo koje integriše sve moždane funkcije – osjećaj za riječi, slike, brojeve, logiku, ritam, boju i prostorne odnose – čime čovjeku omogućuju stvarno korišćenje beskonačnih kapaciteta svoga mozga.

Ovdje je važno spomenuti da to nije nikakva nova „izmišljotina“. Svi koristimo te principe na sebi svojstven način, u raznim situacijama, u većoj ili manjoj mjeri. Ono što ovdje jeste novo je svjestan i planski pristup takvom načinu rada na fakultetima i visokoškolskim ustanovama.

2. KOGNITIVNE MAPE – EFIKASNO UČENJE

Tajnu profesorskog uspjeha na planu motivisanja studenata ne treba tražiti samo u njegovoj predavačkoj

vještini, iako je i ona bitan element edukacije, nego u modelima socijalne promocije studenata. Studenti su najviše zainteresovani za nastavu kada se stvore uslovi da se oni međusobno dokazuju na programu koji se zajedno prolazi.

Mape su jedan od, zaista, pametnih načina da se razvije samostalnost u učenju, a indirektno i uspjeh u učenju. Organizirajući informacije na ovaj način studenti stižu dragocjena iskustva o važnosti specifičnog vizealnog predstavljanja gradiva koje žele naučiti. Sem toga, na ovaj način studenti postaju svjesni svog stila učenja što im donosi više razumijevanja i više uvida u ono čemu služe informacije koje trebaju zapamtiti. Mape mogu kreirati profesori, i studenti.

Osnovni princip efikasnog pamćenja su informacije u dugoročnom pamćenju koje moraju biti kvalitetno i jasno organizovane i strukturirane, jer inače postoji poteškoća u nalaženju željenih sadržaja. Dobro organizovano znači: smisleno, logički i hijerarhijski organizirano. Kognitivne mape su korisne, zbog:

1. kvalitetne organizacije informacija:
2. integracije znanja

Kvalitet organizacije informacija

Svaka kognitivna mapa služi kao mreža ili kostur oko kojeg se informacije mogu sakupljati i logički strukturirati na sljedeći način:

- hijerarhijski (nadređeni elementi obuhvataju pod-elemente),
- vremenski, tj. paralelno (elementi se pojavljuju, djeluju, su pod uticajem - istovremeno),
- sukcesivno (pojedini elementi slijede jedni za drugima),
- logički (npr. više elemenata istovremeno utiče na treći element, a i međusobno su neki elementi u interaktivnom odnosu),
- ili sve zajedno.

To osigurava smislenu organizaciju znanja u dugoročnom pamćenju. Smisljena organizacija omogućava uvid u međusobne odnose informacija, a time pospješuje kvalitetnu organizaciju pamćenja.

Integracije znanja

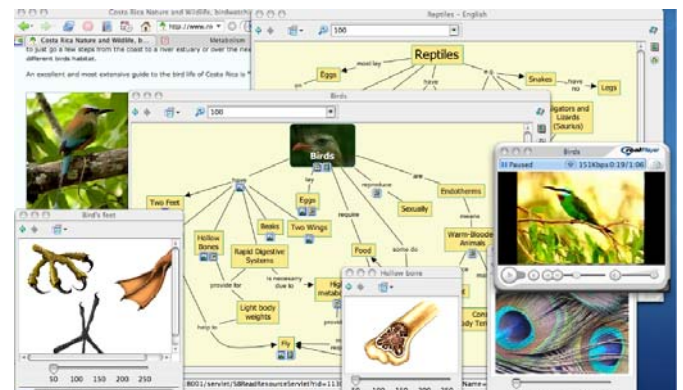
Više kognitivnih mapa se lako mogu smisleno spojiti u jednu jedinstvenu kognitivnu mapu te postoji mogućnost stvaranja veza među različitim područjima znanja i informacija u jednu veliku smislenu cjelinu.

Multimodalno kodiranje informacija - Izradom kognitivne mape stvara se dvostruki trag pamćenja: verbalni i vizuelni. Time se aktiviraju obje hemisfere mozga što povećava sposobnost pamćenja i kreativnog mišljenja.

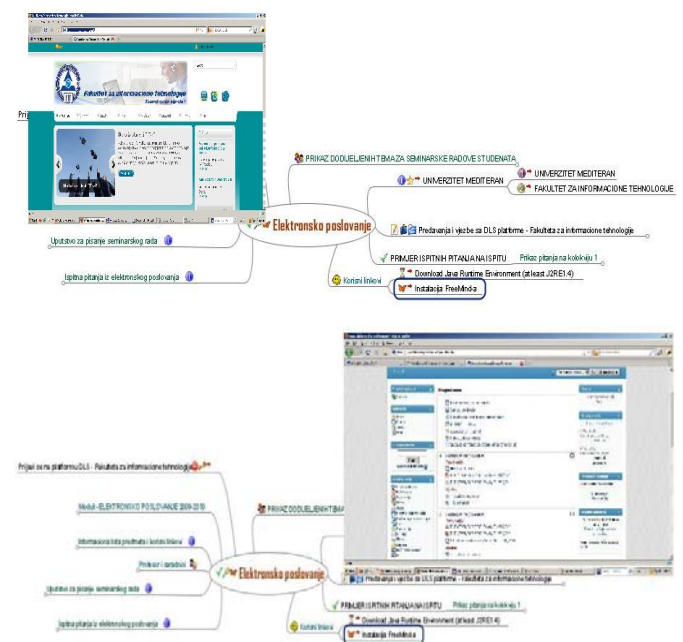
Verbalna informacija su ključne riječi koje se navode u kognitivnoj mapi, a vizuelna je grafički prikaz kognitivne mape. Kognitivne mape su posebno korisne kod vrlo apstraktnih pojmova i informacija jer se za apstraktne pojmove u pravilu stvara samo verbalni trag, pa je manja vjerojatnoća zadržavanja i pronalaženja tih pojmova.

Izrada kognitivne mape zahtijeva *dublju kognitivnu analizu informacija*, tj. usmjerava osobu na razumijevanje:

- logičke povezanosti pojedinih kognitivnih elemenata i čestica znanja i informacija,
- hijerarhijske odnose među elementima,
- sem toga, kognitivne mape podstiču *elaborirano kodiranje*, tj. stvaranje asocijacija, između atributa ili svojstava novih čestica, koje tek treba zapamtiti i između atributa ili svojstava novih i starih, već sačuvanih čestica.



Slika 1. Primjer kognitivne mape



Slika 2. Primjer kognitivne mape za predmet Elektronsko poslovanje – na Fakultetu za informacione tehnologije

3. OSNOVNI PRINCIPI IZRADE KOGNITIVNIH MAPA

Osnovni principi kojih se treba pridržavati pri izradi kognitivnih mapa su sljedeći:

1. hijerarhija i
2. numerički redoslijed.

Koristeći hijerarhijsku organizaciju, kreira se uvid u međusobne odnose i povezanost pojmova. Time se olakšava

pamćenje, jer su stvari organizovane logično. Ako je potrebno, mogu se koristiti brojevi kako bi se povezali podaci po redu – hronološki, po važnosti i ili po nekom specifičnom poretku.

Principi i tehnike izrade kognitivnih mapa

1. **isticanje** - isticanje je glavni faktor za poboljšanje memorije i povećavanje dostupnosti željenih informacija. Sve tehnike preporučene za isticanje mogu se koristiti i za asociiranje i obratno. Navedene tehnike služe za postizanje optimalnog isticanja u napravljenim mapama;
2. **centralni lik** - slika u centru predstavlja fokus za mozak i automatski privlači pažnju. Ako se u sredini nalazi riječ, bilo bi poželjno koristiti iluziju trodimenzionalnosti kako bi se istakla;
3. **slike** - slike su, vrijedne više od hiljadu riječi. Uz to, one su privlačne i mogu spriječiti dosadu pri učenju;
4. **boje** - boje stimuliraju memoriju i kreativnost, a daju život i atraktivnost slici;
5. **3d** - trodimenzionalni likovi iskaču i čine stvari upamtljivijima, a koristite se za glavne dijelove mape;
6. **sinestezija** - u mapama riječi i slike koje predstavljaju kombinacije osjetila (vid, sluh, njuh, dodir, kinestezija, ukus). Tako, umjesto da se na mapi o biljkama napiše riječ *cvijeće*, lakše je ubaciti sliku (cvijeća u jpg ili nekom drugom laganom formatu). To višestruko povećava mogućnosti pamćenja;
7. **veličina** - variranjem veličine slova, likova i grana kodira se relativni značaj pojmova u hijerarhiji;
8. **razmak** - ravnomjerni razmaci povećavaju preglednost i jasnoću slike, a ostavljajući dovoljno prostora oko pojmova osigurava se preglednost mape. Uz to, mapa će ostati *otvorena* za eventualne dopune ili izmjene koju mogu napraviti studenti i/ili profesori.

Asocijacije pomažu da se produbi razumijevanje bilo kojeg gradiva ili problema. Povezujući stvari međusobno, mozak daje smisao iskustvima. Npr. *Strelice* automatski vode ljudsko oko da poveže dijelove. Mogu biti jednosmjerne, dvosmjerne, 2D, 3D, različitih veličina, boja i oblika. *Boje* stimuliraju mozak, jer se određenim bojama može kodirati međusobno povezana područja. To će dovesti do lakše uočljivosti, boljeg pamćenja i povezivanja. *Kodovi* i *skraćenic*e pomažu da se brzo uočite pojmovi koji se obrađuju. Kognitivne mape se mogu koristiti u bilješkama za označavanje stvari koje se često ponavljaju. Što su bilješke jasnije, to će lakše biti pamćenje i snalaženje u njima. *Jedna riječ na liniji* – riječi, na sebe vežu puno asocijacija. Stavljajući samo jednu riječ na jednu liniju koja povezuje pojmove ostavlja se slobodan prostor za nove ideje i preglednost, a pri tome osnovna ideja ili fraza nije izgubljena. Štampana slova jasnije su oblikovana, pa ih mozak lakše *snima*. Pri tome značajna je ušteda pri čitanju i pamćenju. Takođe, štampana slova podstiču na kraće i jednostavnije izražavanje.

Jednaka dužina crta i riječi i *povezivanje više grana* pomažu povezivanju misli, a podebljana središnja linija omogućava isticanje pojedinih djelova mape. *Uokvirivanje* - pomaže da se poveže više informacija, jer se na takav način odmah imaju dostupne sve informacije *s iste grupe*. *Jasnim slikama* – podstiče se jasnoća misli kreatora i korisnika kognitivne mape, a *prikaz papira* *vodoravno* daje više prostora za crtanje određene mape. Postoji izuzetno mnogo *open source programa* za realizovanje kognitivnih mapa. Jedan od njih je i **FreeMind** softver za kreiranje kognitivnih mapa, definisanje ideje ili koncepta.

Softver kao što je FreeMind može poslužiti za jako puno stvari, osim što može vizuelno predstaviti ideju, može se npr. napraviti pregled poslovnog plana, prikazati predavanja određenog profesora, pregled najvažnijih datoteka na hard disku, pregled najčešće korištenih linkova na internet, pregled marketing strategije preduzeća ili branda, itd. Mogućnosti su velike, skoro neograničene, ali možda ono što je najbitnije, FreeMind ima potencijal da navede kreatora ili korisnika kognitivne mape da promijeni način na koji smo do sada obavljali određene stvari na računaru. Kao i sav softver koje preduzeće Aranea mreže d.o.o. zagovara (a i isključivo koristi), i FreeMind je FOSS, što znači da se može instalirati kao potpuno besplatan i legalan softver na računarima fakulteta, visokoškolskih ustanova ili personalnih računara studenata i profesora.

4. ZAKLJUČAK

Kognitivne mape su prije svega namijenjene vizuelnom organizovanju ideje, plana, koncepta, strategije, taktike, a koriste ih menadžeri u marketing sektoru, profesori na fakultetu, student pri učenju, vaspitači u vrtiću, hirurzi u operacionim salama, arhitekta, administratori računarskih mreža, programeri i dr. Kognitivne mape su zaista jedan od pametnih načina da se razvije samostalnost u učenju, a indirektno postigne i uspjeh u učenju. Organizujući informacije na ovaj način studenti stižu iskustva o važnosti specifičnog vizuelnog predstavljanja gradiva koje trebaju naučiti. Sem toga, ovo je samo još jedan način da oni postaju svjesni svog stila učenja koje im donosi više razumijevanja i više uvida u ono čemu služe informacije koje memoraju zapamtiti. Praveći mape otkrivaju se bitna svojstva riječi i pojmova kojima se služe, uči se o njihovim međusobnim odnosima, realizuje intenzivno i samostalno razmišljanje. FreeMind je samo jedan od niza softvera koji se koristi za mapiranje, kojim se olakšava učenje svih mogućih tema koje se mogu prikazati linearno, kružno, hijerarhijski.

LITERATURA

- [1] K. M. Ford, A. J. Cañas, J. Jones, H. Stahl, J. D. Novak, Adams -Webber, J. Iconkat: An integrated constructivist knowledge acquisition tool. *Knowledge Acquisition*, 3, pp. 215-236, 2009.
- [2] O.R. Anderson, Some interrelationships between constructivist models of learning and current neurobiological theory, with implications for science education. *Journal of Research in Science Teaching*, 19(10), pp. 1037-1058, 2007.

KREIRANJE WEB CMS SISTEMA FAKULTETA ZA INFORMACIONE TEHNOLOGIJE PUTEM JOOMLA PLATFORME

THE CREATION OF WEB CMS SYSTEM FOR FACULTY OF INFORMATION TECHNOLOGY BY USING JOOMLA PLATFORM

Zoran Ćorović, Novak Radulović, Snežana Šćepanović, Fakultet za Informacione Tehnologije, Univerzitet Mediteran

Sadržaj - Razvoj web sajtova se sve više bazira na modernim, dinamičkim rešenjima. Razvoj open source web sistema za upravljanje sadržajem web strana omogućava fleksibilnost i laku dinamičkih web aplikacija. U radu se razmatra koncept Joomla web sistema za upravljanje sadržajem i praktična primjena za razvoj web sajta Fakulteta za informacione tehnologije.

Abstract - Web site development is more and more based on modern, dynamic solutions. Development of open source web content management systems allows flexibility and ease of development of dynamic web applications. The paper discusses the concept of Joomla web content management system and practical application for the development of the web site for the Faculty of Information Technology.

Keywords: CMS, Web development, Open source software, Joomla

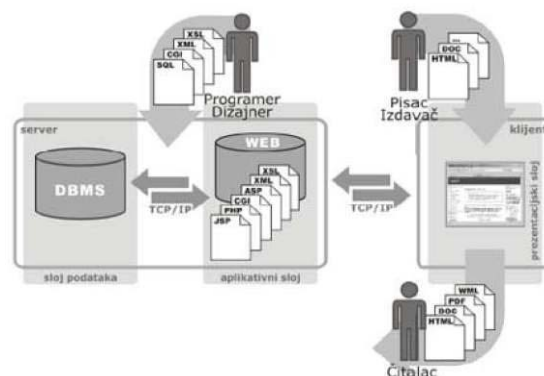
1. UVOD

Sistemi za upravljanje sadržajem (eng. CMS – *Content Management Systems*) su softverske aplikacije koji se koriste za kreiranje, skladištenje i objavljivanje sadržaja kao što su dokumenti, binarni fajlovi, slike, audio i video zapisi. Ukoliko je glavna namjena CMS-a kreiranje i održavanje web strana, tada se on naziva Web sistem za upravljanje sadržajem (Web CMS). Podvrste CMS-a su i sistemi za upravljanje dokumentima (DMS – *Document Management Systems*), sistemi za upravljanje digitalnim resursima (DAMS – *Digital Assets Management Systems*) i sistemi za upravljanje edukativnim sadržajem (LCMS – *Learning Content Management System*).

2. KONCEPT WEB SISTEMA ZA UPRAVLJANJE SADRŽAJEM

U pitanju je dinamička web aplikacija sa troslojnom arhitekturom. Na serverskoj strani nalazi se aplikativni sloj, koji se sastoji od web i aplikativnog servera (npr. Tomcat, Apache, IIS) zaduženog da procesira aplikaciju kodiranu u nekom od programskih ili script jezika (npr. JSP, PHP, CGI, ASP, Python, Perl). Na istoj strani nalazi se i sloj za rukovanje bazama podataka (DBMS) koji se oslanja na funkcionalnost nekog od sistema za rukovanje relacionim bazama podataka (npr. MySQL, SQL Server, Oracle). Na drugoj, klijentskoj strani, prezentacijski sloj CMS sistema sastoji se iz klijentskih aplikacija, koje se izvršavaju na korisnikovom operativnom sistemom. Osnovni koncept sistema za upravljanje sadržajem prikazan je na slici 1.

Komunikacija između tri navedena sloja ostvaruje se ili posredstvom lokalne mreže, ili putem Interneta, a u oba slučaja uz korišćenje transportnog protokola TCP/IP



Slika 1: Troslojna arhitektura CMS-a

Javnom dijelu CMS sistema - stranicama sajta, može pristupiti svako. Administrativni dio ima zaseban panel i može mu pristupiti administrator sistema, kao i ostali korisnici sa jasno definisanim pravima i nivoom pristupa. Administrativni dio se velikim dijelom koristi kako za dizajn sajta, tako i za njegovo ažuriranje, postavljanje novosti na sajt i slično. Administrativnom dijelu može pristupiti osoba zadužena za ažuriranje web portala i kroz panel za unos sadržaja dodati ili editovati sadržaj. Promjene dizajna može izvršiti samo administrator, programer ili dizajner.

Glavna odlika web sistema za upravljanje sadržajem je mogućnost uređivanja online – pomoću web browsera. Sistem je postavljen na web server i mogu mu pristupiti ovlašteni korisnici, za razliku od statičkih web prezentacija gdje se na lokalu vrši uređenje stranica i gdje je potrebno svaki put nove stranice kopirati na server, povezivati sa ostatkom sadržaja, paziti na dizajn nove stranice, što je previše komplikovano za održavanje većih web sajtova koji zahtjevaju redovno ažuriranje.

3. JOOMLA CMS

Joomla (<http://www.joomla.org>) predstavlja besplatan, Open Source sistem za organizaciju, obradu i objavljivanje web sadržaja. CMS sistem Joomla je kao projekat proistekao iz Mambo sistema za upravljanje sadržajem, kada je određeni broj programera riješio da pokrene zaseban web sistem. Joomla web zajednica ima aktivnu ulogu u razvoju aplikacije, koja se ogleda u zajedničkom testiranju, pisanju ekstenzija, smišljanju ideja za dalji razvoj, razmjenu iskustava i međusobnom pomaganju pri rješavanju problema u radu sa platformom. Joomla se nalazi pod GNU licencom.

Joomla se može koristiti u više domena: kod izrade web i korporativnih portala, intranet i ekstranet rješenja, online magazina, e-commerce rješenja, rješenja za mali biznis, community portala, portala obrazovnih ustanova, kao i ličnih prezentacija.

Koncept Joomla sistema je tako kreiran da krajnji korisnici mogu uz minimalnu obuku sami održavati web sajtove. Joomla može odgovoriti raznim zahtjevima zbog ogromnog broja dodataka koji se mogu naći na internetu. Takođe, postoji onlajn direktorijum dodataka za Joomla CMS, koji se mogu implementirati kroz administratorski panel, i prikazati na sajtu. Napominjemo da nisu svi dodaci open source, već su neki i komercijalne prirode.

Joomla se trenutno distribuira u 2 pravca. Verzija 1.0.x predstavlja Joomla rješenje zasnovano na razvoju predhodnika Mambo. Drugu distribuciju predstavlja Joomla 1.5 koja je aktuelna realizacija Joomla razvojnog tima. Trenutno veću podršku u smislu dodataka ima Joomla 1.0.x, dok je sve ostalo na strani Joomla 1.5 verzije, prije svega potencijal i dalji razvitak. Međutim, već neko vrijeme se najavljuje nova realizacija Joomla CMS-a, to je verzija 1.6 koja novi veliki broj novina. Sama logika funkcionisanja sistema je donekle izmijenjena, a radilo se i na ubrzanju performansi jer je poznato da Joomla baš i ne štedi resurse. Uglavnom, nova verzija je tek u "alpha" razvojnoj fazi, samo za testiranje.

2. SIGURNOST JOOMLA SISTEMA

Postoje brojne polemike o sigurnosti Joomla sistema. Sama činjenica da je u pitanju Open Source mnoge korisnike postavlja u skeptično stanje. Ipak, uz redovno ažuriranje sistema i uz nekoliko dodatnih metoda Joomla postaje više nego siguran sistem. Ja ću se osvrnuti na akcije koje bi administrator Joomla CMS-a trebao da obavlja kako bi svoj sistem držao na relativno sigurnom nivou, i tako što bolje zaštićenim od napada sa interneta.

Prvenstveno, Joomla razvojni tim konstantno dobija povratne informacije od korisnika Joomla CMS-a, u cilju otkrivanja grešaka u sistemu, gdje se naglasak najviše stavlja na sigurnosne propuste sistema. Filozofija je jednostavna, i ona u suštini predstavlja standardno testiranje softvera na greške. Otkrivanjem propusta, on se pokušava nadoknaditi kroz kod, i na taj način ispraviti greška. Ovaj logičan slijed događaja, prati korak - postavka (sigurnosne) zakrpe na fajl server, kako bi bio dostupan svim korisnicima. Sve što treba da uradi administrator CMS-a je da tu sigurnosnu zakrpu

primijeni na svoj sistem. Ovo je, razumljivo, iterativan proces.

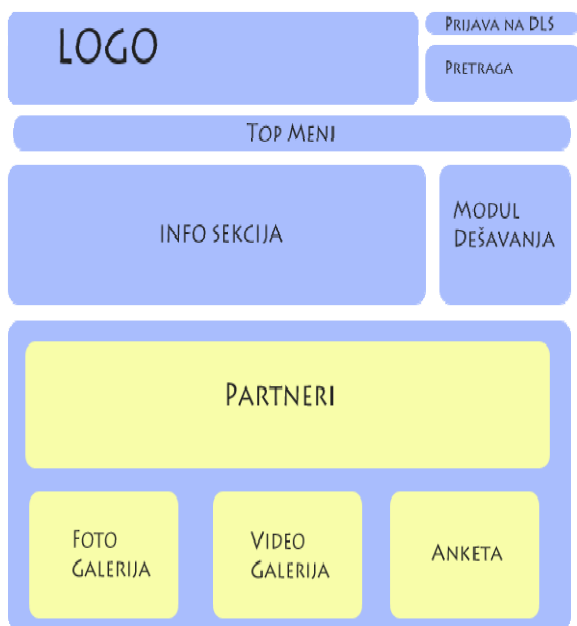
Međutim, postoje i druge metode koje se tiču samih postavki web servera, kao i postavljanja dostupnosti kontrolnog panela CM sistema. U svakom sistemu postoje kritični fajlovi koji sadrže parametre neophodne za pravilan rad sistema. Slično je i sa Džumlom. Stoga je neophodno primijeniti određene restriktivne dozvole koje se mogu izvršavati nad tim fajlovima. Obično se ovakvi fajlovi postavljaju u stanje "read-only" (samo za čitanje), kako bi se na bilo koji način spriječila njihova izmjena. To se radi primjenom komande CHMOD. Primjer ovakve komande nad glavnim konfiguracionim fajlom Joomla sistema bi bio: *CHMOD 444 configuration.php*.

Osim ovoga, opšte je poznato da se Joomla administratorskom panelu može pristupiti preko adrese *http://adresawebsajta/administrator*. Ono što dočeka korisnika na toj adresi je forma za unos korisničkog imena i lozinke. Dobra stvar u vezi lozinke je njeno kodiranje u samoj Mysql bazi. Kodiranje lozinke se vrši MD5 algoritmom (Message Digest algorithm 5) koji koristi 128-bitnu zaštitu. Do danas nije pronađen način za probijanje ove vrste zaštite, što je jasno mjerilo sigurnosti lozinke. Međutim, korištenjem raznih alata za "nagađanje" korisničkog imena i lozinke, moguće je doći do istih. Stoga dodatna mjera opreza od napada sa interneta može biti "sakrivanje" administracionog panela od javnog pristupa. Postoji dodatak (besplatan, open source) koji se može preuzeti iz Džumline onlajn baze dodataka, a čija je osnovna funkcija da zaštiti administracioni panel. U konfiguraciji ovog dodatka se postavlja dodatna sigurnosna lozinka, koju je takođe neophodno unijeti u polje za web adresu kako bi se pristupilo panelu. Na ovaj način se sama forma za unos korisničkog imena i lozinke štiti od raznih vrsta napada.

3. PRIMJENA JOOMLA SISTEMA

Joomla CMS koristili smo za izradu nove verzije web sajta Fakulteta za informacione tehnologije u Podgorici. (<http://fit.unimediteran.net>). Ono što predstavlja važan korak pri izradi bilo kojeg web sajta je smisleno i logičko definisanje navigacione strukture i dizajna naslovne strane, pritom prateći zahtjeve korisnika, i postavljajući one ključne u prvi plan..

Idejno rješenje naslovne strane web sajta za Fakultet informacionih tehnologija, predstavljeno je na slici 3. Sa dizajna početne strane jasno su uočljivi ključni moduli. Prateći trendove razvoja weba, moduli su zamišljeni da budu dinamični, stoga su većinom rađeni u ajax scripting tehnologiji.



Slika 3: Dizajn naslovne strane Fakulteta za informacione tehnologije

Sadržaj svakog od modula se može mijenjati putem administratorskog panela, a to čak i ne mora raditi administrator sajta, već neko sa nižim nivoom pristupa, npr. urednik vijesti na sajtu, i sl. U daljem tekstu detaljnije ćemo opisati funkcionalnosti modula na početnoj strani web sajta Fakulteta za informacione tehnologije..

Modul "Info sekcija" pruža osnovne informacije budućim studentima o radu fakulteta, uslovima na fakultetu, izučavane materije, predmetima i slično. Rađen je na principu rotatora fotografija sa opisom. Sadržaj se mijenja dinamički, što je postignuto upotrebom scripting jezika - javascript. Modul je lako uočljiv, zauzima jedan od tri bitna "containera" web sajta, odmah ispod glavnog menija.

Modul "Dešavanja" pruža osnovne informacije o trenutnim dešavanjima, kao i informacije o trenutnim informatičkim kursovima na fakultetu, i slično. Informacije o dešavanjima na fakultetu se protežu na više strana unutar samog modula, i takođe se rotiraju dinamički, bez potrebe za ponovnim učitavanjem strane. Radi se na doradi modula, kako bi se omogućilo posjetiocima sajta da slobodno komentarišu najnovije vijesti. Pozicija modula je odmah desno od info sekcije, kako bi se postigla skladnost informacija na web sajtu. Modul je rađen u AJAX tehnologiji.

Modul "Partneri" pruža informacije o kompanijama koje su sklopile saradnju sa Fakultetom za informacione

tehnologije, i čiji se programi kurseva izučavaju na određenim predmetima nastavnog plana i programa Fakulteta, a takođe održavaju zasebno van nastave. Tu se u prvi plan ističu kompanije koje imaju primat u IT tehnologiji, kao što je to "Oracle" ili "Cisco". Rađen je na sličan način kao i info sekcija, a njegova pozicija je nešto ispod nje. Modul svoju dinamičnost postiže javascript tehnologijom.

Modul "Anketa" pruža određenu vrstu "feedback"-a administratoru, jer je dizajniran da prikuplja informacije od strane posjetilaca, jednostavnim odgovorom na postavljeno pitanje. Rađen je po standardu u AJAX tehnologiji, stoga ima opciju za glasanje i za pregled rezultata. Ugrađena je opcija za rotiranje više anketa, ukoliko ih se nalazi više od jedne u bazi.

Modul "Prijava na DLS" ima za cilj da integriše sajt Fakulteta za informacione tehnologije sa drugim sistemom fakulteta – Moodle platformom za e-learning. Pravljen je na bazi osnovnog modula za prijavljivanje u Joomla CMS sistem, a doraden je izmjenom php koda, koji daje funkcionalnost modulu, odnosno omogućava pristup na moodle distance learning sistem Fakulteta za informacione tehnologije. Dinamičnost modulu daje javascript (mootools open source web 2.0 framework), dok je dizajn modula urađen uglavnom koristeći CSS. Modul se doraduje na način da se omogući povezivanje i usklađivanje vijesti sa dvije platform – Moodle i Joomla što se trenutno postiže RSS-om.

Ovako modifikovan i uređen modul prerađen je kao dodatak, i dodat u oficijalnu bazu Joomla dodataka, i tako je dat određeni doprinos open source filozofiji kreiranja softvera. Trenutno Joomla baza dodataka broji oko 5000 dodataka. Oni su razvrstani po kategorijama, u zavisnosti od funkcionalnosti koje pružaju.

4. ZAKLJUČAK

Joomla CMS je softverski sistem dizajniran za organizovanje, kreiranje i menadžment web sadržaja. Web sistemi za upravljanje sadržajem u znatnoj mjeri olakšavaju proces održavanja i ažuriranja web sajta jer omogućavaju kolaborativan rad više korisnika u isto vrijeme putem web browser-a. Modularna priroda CMS sistema omogućava njegovo funkcionalno proširivanje različitim sa vrstama web aplikacija. Poseban značaj za razvoj web aplikacija ima činjenica da je izvorni kod Joomla sistema open source, što omogućava web dizajnerima da kreiraju i implementiraju nove module u postojeći sistem.

LITERATURA:

[1] Joomla community: <http://www.joomla.org>.

NETFPGA RAZVOJNA PLATFORMA ZA REALIZACIJU AKTIVNIH MREŽNIH KOMPONENTI

NETFPGA PLATFORM FOR ACTIVE NETWORK DEVICE IMPLEMENTATION

Budimir Ugren, *Crnogorski Telekom*,
Milutin Radonjić, Igor Radusinović, *Elektrotehnički fakultet Podgorica*

Sadržaj — U ovom radu je predstavljena NetFPGA razvojna platforma koja predstavlja polaznu osnovu za razvoj i realizaciju različitih aktivnih mrežnih komponenti. NetFPGA je otvorena platforma pa je implementaciju novih projekata moguće zasnivati na već razvijenim i provjerenim modulima. Prikazana je principijelna šema platforme i opisan način njenog povezivanja sa razvojnim okruženjem. Dat je i osvrt na primjer jedne moguće implementacije, u vidu generatora paketskog saobraćaja.

Abstract — In this paper, NetFPGA platform which is basis for development of various active network components, has been introduced. NetFPGA is open platform and many modules from previous projects can be reused. General NetFPGA configuration is presented and procedure for connecting to development environment is described. One possible example of implementation, through network packet traffic generator, is observed.

1. UVOD

NetFPGA je rekonfigurabilna hardverska platforma, optimizovana za realizaciju mrežnih uređaja u računarskim mrežama velikih brzina. Razvijena je na Stanford univerzitetu i ima kompletnu podršku za implementaciju mnogih mrežnih uređaja, kao što su: četvoroportni mrežni adapter, ruter, komutator, generator paketskog saobraćaja i slično. NetFPGA uključuje sve što je potrebno za kompletnu realizaciju ovih uređaja. To znači da uključuje logičke resurse, memorijske jedinice, kao i četiri Gigabit Ethernet interfejsa. Budući da je cijeli modul za prenos podataka implementiran u hardveru, sistem može podržati punu linijsku brzinu po svakom portu na nivou od 1Gb/s.

Brzi rast Interneta omogućili su upravo mrežni ruteri i komutatori sa hardverskom implementacijom prosleđivanja saobraćaja. Oni se koriste unutar lokalnih mreža (Local Area Networks - LAN) za raspoređivanje mrežnog saobraćaja, kao i za rutiranje Internet Protocol (IP) paketa kroz WAN (Wide Area Networks) mreže. Da bi se u današnje vrijeme napravio stvaran sistem koji odgovara ovakvim zahtjevima neophodno je obezbijediti mogućnost procesiranja paketa u hardveru. U današnje vrijeme naučna zajednica najčešće koristi Field Programmable Gate Arrays (FPGAs) ili Application Specific Integrated Circuits (ASICs).

U ovom radu biće prikazana principijelna šema NetFPGA razvojne platforme i detaljnije objašnjene njene najvažnije komponente. Biće prikazane njihove karakteristike, aspekti logičke realizacije kao i princip funkcionisanja, na kojima je zasnovan potencijal ove platforme. Dat je i osvrt na jednu moguću realizaciju, u vidu generatora paketskog saobraćaja. Detaljniji proces njegove realizacije sa pojedinostima o načinu korišćenja i praktičnim primjerima dat je u [1]. Istorijat i glavne karakteristike NetFPGA platforme opisane su i pojašnjene u poglavlju 2, dok su specifičnosti sa detaljima za jednu od realizacija date u nastavku – poglavlja 3, 4 i 5. Način implementacije sa smjernicama o korišćenju

jedne od primjena, koja je u ovom slučaju generator paketskog saobraćaja, tema je preostalih segmenata ovog rada – poglavlja 6 i 7.

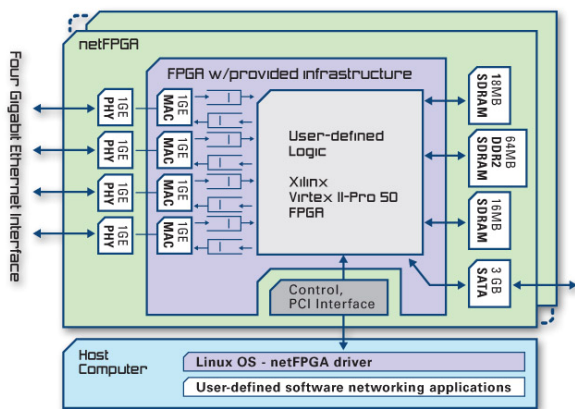
2. NETFPGA PLATFORMA

Prva verzija je započeta 2001. tako da je prvi prototip bio u upotrebi već krajem 2003. godine. Predviđeno je bilo da se NetFPGA kartici pristupa udaljeno i za potrebe njenog konfigurisanja kao i za otkrivanje i ispravljanje grešaka, jer uređaj nije bio fizički dostupan.

NetFPGA-v1 kartica bila je dimenzija 6x9 inča i imala je tri Altera EP20k400 APEX uređaja sa programabilnom logikom, osmoportni Ethernet kontroler, tri SRAM modula po 1MB kao i pomoćnu logiku. Nije bilo CPU-a na ploči. Jedan od tri FPGA, nazvan Control FPGA (CFPGA), bio je unaprijed programiran i preko Ethernet kontrolera povezan sa dva User FPGA (UFPGA). Radi olakšavanja razvojnog procesa bile su obezbijedene odgovarajuće skripte i biblioteke. Kada bi sinteza i simulacija određenog dizajna bile završene, učitavanje FPGA konfiguracionog fajla na NetFPGA-v1 karticu radilo se preko Web interfejsa. Nakon svega toga, vršilo se međupovezivanje u lokalnu mrežu tako da je uređaj sa projektovanim dizajnom učestvovao u realnom saobraćaju. (Za mapiranje NetFPGA v1 portova u lokalnom okruženju korišćen je lokalno razvijen alat, takođe na Stanfordu, pod imenom Virtual Network System [2]).

U septembru 2005. godine pojavljuje se prvi prototip NetFPGA-v2, koji je PCI kartica i u potpunosti je 32-bitan sistem, koji radi na 33MHz. Na slici 1. je prikazana blok šema NetFPGA-V2 platforme i njena veza za host računarom. Imala je Xilinx Spartan uređaj, a posjedovala je i Xilinx Virtex2P30 i 512Kx36 SRAM. Glavna promjena bio je prelazak na PCI format. Ovo je imalo dvije prednosti: prvo, format je veoma uobičajen za potencijalne korisnike, a drugo, veća propusnost otvara nove mogućnosti za sistem. Još u verziji v1 ploča je napajana preko PCI magistrale, a ovdje je

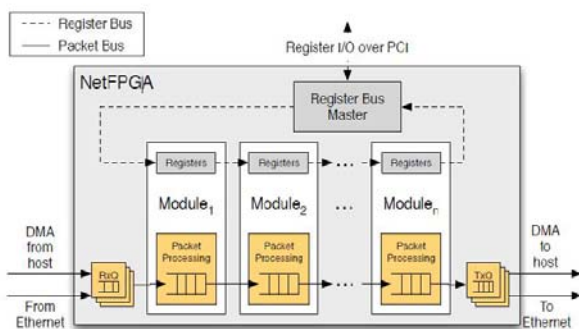
ovdje je iskorišćena brzina ove magistrale i za programiranje Virtex-a. Prilikom inicijalizacije Spartan se sam konfigurira iz flash EPROM-a i „prijava“ se PCI magistrali.



Slika 1. NetFPGA šema

Softver na host-računaru može, nakon toga, da konfigurira Virtex kroz PCI interfejs. Konfiguracija traje oko jedne sekunde. Time je postignuto da je korisnik u mogućnosti da rekonfigurira Virtex bez potrebe za restartovanjem host-računara.

Funkcionalnost NetFPGA sistema je obezbijedena kroz tri komponente. Prva je kernel modul, koji se koristi da komunicira sa NetFPGA hardverskom komponentom. Kernel modul daje mogućnost da bitfajl za FPGA bude učitao preko PCI magistrale sa host-računara pod Linux-om. Uz to daje mogućnost softveru da komunicira sa NetFPGA kroz registar interfejs implementiran korišćenjem dijeljene memorije. Softver čita statistiku brojača iz registara i upisuje kontrolne podatke. Drugi dio su uslužni programi za komunikaciju sa NetFPGA pločom. Oni uključuju (ali nijesu ograničeni samo na) *bitfile download* i programe za čitanje i upisivanje u registar. Treći dio su referentni tokovi. Slika 2 prikazuje referentne tokove kroz module. Mogu se učitati paketska magistrala i registar magistrala.



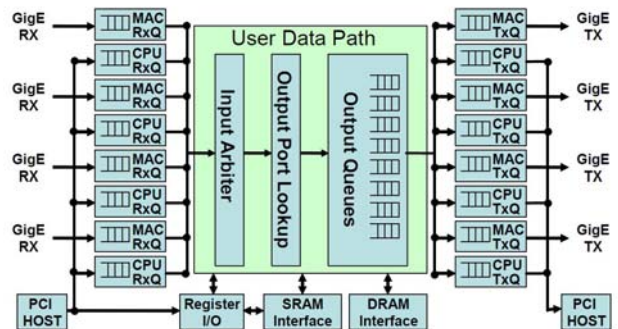
Slika 2. Referentni tokovi kroz module

Putem paketske magistrale distribuiraju se podaci između modula koji obrađuju pakete. Nizovi 64-bitnih riječi, koje su sačinjene od zaglavlja i podataka, obrađuju se kao rijeka podataka kroz svaki modul. Ukupno kašnjenje sistema lako se može odrediti brojanjem taktova koji su potrebni da paket prođe kroz svaki modul u referentnom toku.[3]

Standardni registar interfejs daje softveru uvid u hardverske registre, brojače i tabele i dozvoljava mu da ih

modifikuje. Softver memorijski mapirane registre vidi kao Ulazno/Izlazne registre i može im pristupiti koristeći U/I kontrole. Jedan modul u lancu inicira zahtjev ili pak odgovara na zahtjev koji stiže kao PCI zahtjev od strane softvera.

Funkcionalnosti modula mogu biti različite i specifične za određeni dizajn, ali postoje moduli za koje se može reći da su univerzalni, kao što se može vidjeti na Slici 3. Ulazi i izlazi su međusobno povezani preko *User Data Path-a*, koga čine moduli *Input Arbiter* i *Output Queues*. To su glavni moduli i prisutni su u skoro svim NetFPGA dizajnim, kao i *Output port lookup* koji se brine o tome na koji port treba prosljediti pakete. Na izlazu se nalazi 8 redova za slanje, a na ulazu 8 redova za prijem paketa. Redovi mogu biti tipa MAC ili CPU. Svakom od četiri mrežna porta na NetFPGA kartici pripada po jedan MAC red za slanje i jedan za primanje paketa. Paralelno sa njima nalaze se CPU redovi koji služe za komunikaciju sa centralnim procesorom. Za Ethernet komutaciju, odluka je zasnovana na MAC adresi na početku Ethernet frejma dok se za IP rutiranje odluka bazira na IP adresi odredišta iz za to predviđenog polja u zaglavlju IP paketa. Druge politike, koje koriste kombinacije polja, mogu se koristiti modifikovanjem logike u lookup algoritmu. Izvorni kod za sve module napisan je u Verilogu i dostupan je u okviru tzv. open-source projekta.



Slika 3. NetFPGA referentni tokovi

3. PRIMJENE PLATFORME

Kroz implementaciju generatora paketskog saobraćaja na NetFPGA platformi verzije 2.1 biće demonstrirana jedna moguća primjena ove platforme.

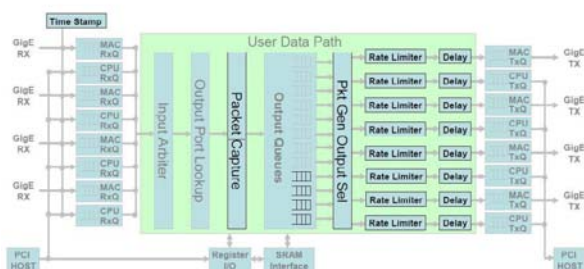
NetFPGA v2.1 platforma sadrži Xilinx Virtex-II Pro FPGA čip koji je programiran logikom definisanom od strane korisnika i Xilinx Spartan FPGA čip koji programira Virtex i implementira PCI interfejs ka procesoru hosta. Platforma uključuje dva statička RAM (SRAM) modula koji rade sinhrono sa FPGA i dva Double Data Rate (DDR2) SDRAM-a druge generacije koji rade asinhrono sa FPGA. Četiri fizička porta (tj. medijuma na fizičkom nivou) koji su ugrađeni, daju mogućnost platformi da šalje i prima pakete preko četiri standardna Ethernet konektora. Dva Serial-ATA (SATA) konektora na platformi omogućavaju povezivanje više NetFPGA ploča unutar sistema. NetFPGA logička biblioteka sadrži kosturni Verilog dizajn koji sadrži četiri Gigabit Ethernet Media Access Controller-a (GMACs) i interfejs logiku SRAM i DDR2 memorije. Interni modul

interfejsa koristi standardni First-In-First-Out (FIFO) algoritam.

NetFPGA koristi standardni Computer Aided Design (CAD) alat da implementira kola koja rade na FPGA [4]. Funkcija logike je simulirana koristeći Mentor Graphics ModelSim alat. Verilog source code je sintetizovan u netlist-i koja definiše konfiguraciju i topologiju hardverskih resursa. Logika je postavljena i rutirana koristeći Xilinx ISE alat. Bitfile-ovi se upisuju u Virtex FPGA kroz PCI interfejs, tako da download kabl nije potreban da bi rekonfigurisali logiku. Paketi stižu na ulazni modul NetFPGA sa četiri mrežna Gigabit Ethernet interfejsa do CPU-a hosta preko PCI bus-a.

4. ARHITEKTURA GENERATORA PAKETSKOG SAOBRAĆAJA

Glavne funkcionalnosti generatora paketskog saobraćaja su realizovane u okviru *User Data Path-a*. Slika 4. prikazuje šemu referentnih tokova sa naglašenim modulima za realizaciju paket generatora. Samo neznatne modifikacije trebalo je uraditi izvan *User Data Path-a*, kao što je dodavanje *Time Stamp* modula. Ovo je omogućilo dolaznim paketima da budu obilježeni vremenskim pečatom kad budu primljeni od strane hardvera. Ovaj podatak se kasnije koristi pri kreiranju *pcap* fajla sa vremenskim obilježjima koja su mnogo preciznija nego što je slučaj kada se evidentira trenutak kada kernel primi paket od hardvera. Moduli tipični za generator paketskog saobraćaja su *Packet Capture*, *Packet Generator Output Select*, *Rate Limiter* i *Delay* [5].



Slika 4. Šema referentnih tokova paket generatora

Packet Capture modul je zadužen da pravi zbirne statistike za pokrenute funkcije generisanja ili primanja paketa kao što je, na primjer, broj primljenih paketa i ukupno vrijeme primanja po sesiji i po portu, kao i da uklanja vremenska obilježja sa paketa kada je funkcija generisanja ili primanja paketa onemogućena.

Packet Generator Output Select modul je povezan na *Output Queues* i određuje koji od 12 odlaznih redova bi trebao biti povezan na koji od 8 izlaznih redova odgovarajućeg referentnog toka. Četiri neraspoređena reda ostavljena su za funkciju prihvatanja. Na ovaj način se obezbjeđuje da imaju 2 prenosna reda po MAC-portu, jedan koji bi bio korišten za slanje dok bi se na drugom, ako treba, prihvatili paketi.

Svaki od 8 referentnih odlaznih redova mogu biti povezani sa *Rate Limiter* i *Delay* modulima, tako da se za svaki red nezavisno može odrediti brzina predaje i kašnjenje među paketima.

5. SOFTVER GENERATORA PAKETSKOG SAOBRAĆAJA

Perl skripta koja se zove *paket_generator.pl* i koja se nalazi u direktorijumu *sw* unutar projekta *paket_generator* predstavlja softversku komponentu generatora paketskog saobraćaja. Generator paketskog saobraćaja može generisati sobračaj na jednom ili na više NetFPGA portova, istovremeno. Paketi saobraćaja se iz *pcap* datoteke prebacuju u memoriju na NetFPGA kartici. Ukoliko je datoteka veća od memorije koja je na raspolaganju, onda se učitava dio po dio. Ako je uključeno primanje paketa, oni će biti pohranjeni u SRAM, zatim prebačeni na host računar i upisani u datoteku korištenjem standardnog *pcap* formata.

6. REALIZACIJA PRIMJENE PLATFORME NA PRIMJERU GENERATORA

A. Instalacija

Nakon što se NetFPGA ploča fizički doda na neki od PCI slotova na host-u, neophodno je obezbijediti Gateway/Software komplet za instalaciju, koji se nalazi na Internetu [5]. Naravno, podrazumijeva se da je host računar preinstaliran i da se na njemu nalazi na Linux-u bazirani operativni sistem CentOS, verzije 4.x ili 5.x, iz razloga što samo za navedeni operativni sistem trenutno postoji kompletna podrška. Potrebno je instalirati i *yum* repository za NetFPGA, jer je *yum* program za instalaciju software-a na CentOS-u. Ova dva repository-a su različita za dvije navedene verzije operativnog sistema. Sledeća komanda će instalirati NetFPGA osnovni paket:

```
yum install netfpga-base
```

Može se dogoditi da postoje još neki programski paketi koje je neophodno instalirati da bi željeni program radio i takve pakete operativni sistem prepoznaje i upozorava na to, pa se moraju i oni instalirati. Instalacija NetFPGA osnovnog paketa priprema sistem za rad.

Na ovako pripremljenom hostu kreira se NF2 direktorijum u svom korisničkom home direktorijumu. Skripta koja kopira čitav NF2 na željenu lokaciju je:

```
user_account_setup.pl
```

a nalazi se u direktorijumu:

```
/usr/local/NF2/lib/scripts/user_account_setup
```

Ova skripta takođe dodaje i promijenljive okruženja u **bashrc_addon** fajl.

Jako je važno restartovati računar nakon ovoga.

Zatim se prekompajlira kernel i instalira se drajver za NetFPGA. Ovo radi root korisnik ili superuser u *~/NF2* direktorijumu komandama:

```
make  
make install
```

respektivno. Ovdje još jednom treba restartovati računar. Drajver će se nalaziti na adresi

```
/lib/modules/'uname-r'/kernel/drivers/
```

kao fajl **nf2.ko**, gdje **'uname-r'** figurira kao oznaka tipa i verzije kernela.

B. Verifikacija software-a i hardware-a

Najpre treba provjeriti da li je drajver učitano, i to komandom:

```
lsmmod | grep nf2
```

i na terminalu treba da se dobije :

```
nf2                22540  0.
```

Sljedeći korak je provjera da li su NetFPGA interfejsi podignuti, na sledeći način:

```
ifconfig -a | grep nf2
```

i ako jesu izlaz će izgledati ovako:

```
nf2c0    Link encap:Ethernet  HWaddr 00:4E:46:32:43:00
nf2c1    Link encap:Ethernet  HWaddr 00:4E:46:32:43:01
nf2c2    Link encap:Ethernet  HWaddr 00:4E:46:32:43:02
nf2c3    Link encap:Ethernet  HWaddr 00:4E:46:32:43:03
```

gdje treba primijetiti da naziv imena počinje sa nf2c što bi značilo NetFpgaversion2Card sa indeksima 0, 1, 2 i 3. U slučaju da nijesu podignuti interfejsi akciju treba uraditi ručno:

```
for i in `seq 0 3`
```

```
do ifconfig nf2c$i up
```

```
done
```

Ovim bi bila završena verifikacija software-a i hardware-a.

C. Reprogramiranje CPCI (CompactPCI)

CPCI je industrijski standard za modularne računarske sisteme koji su u osnovi PCI uređaji. Prilikom svakog startovanja host računara trebalo bi se uvjeriti da je na NetFPGA ploči u radnom režimu najnovija verzija firmware-a. Da bi se to obezbijedilo potrebno je poslednji CPCI image fajl, koji se nalazi na lokaciji **/usr/local/NF2/bitfiles**, upisati na NetFPGA karticu. Za to je predviđena skripta:

```
/usr/local/sbin/cpci_reprogram.pl
```

Ako na host računaru ima instalirano više NetFPGA kartica onda na kraju treba dodati:

```
--all
```

da bi odjednom učitali najnoviji firmware na sve kartice. Ukoliko je potrebno da nakon svakog resetovanja host računara bude reprogramiran CPCI, najlakše je u fajl **/etc/rc.local** dodati sledeću liniju:

```
/usr/local/NF2/lib/scripts/cpci_reprogram/cpci_reprogram.pl --all
```

D. Provjera ispravnosti NetFPGA kartice

NetFPGA Self-test potvrđuje da su sve komponente na platformi u potpunosti funkcionalne. Self-test se sastoji od FPGA bitfile-a koji sadrži logiku i interfejs za vanjske komponente kao i softvera koji prikazuje rezultate. Self-test tretira sav hardver paralelno i nastavlja se izvoditi više puta, sve dok ne bude prekinut od strane korisnika. Self-test je rađen i u fabrici odmah nakon što su NetFPGA kartice proizvedene. Kartice se ne distribuiraju dok u potpunosti ne prođu sve funkcije koje predviđa self-test process [1].

E. Regression testovi

Komplet regression testova je skup provjera za razne funkcionalnosti realizovanog gateway-a i software-a. Ne preporučuje se ove testove prolaziti ako prethodno nije prošao Self-test. Postupak je sledeći:

- Povezati eth1 sa nf2c0, a eth2 sa nf2c1
- Prijaviti se kao root ili superuser
- Učitati referentni bitfile na NetFPGA

- Pokrene se skripta

nf21_regress_test.pl –project packet-generator

Kao što je već pomenuto, ne smije biti poruka o greškama tokom testiranja. Dužina trajanja testova je oko deset minuta, ukupno [1].

7. KORIŠĆENJE GENERATORA

Pokretanjem skripte **packet_generator.pl**, sa željenim parametrima za svaki port ponaosob, startuje se proces generisanja i nakon zavšetka daje se statistički pregled izvršenih akcija. Mogući parametri su:

```
-qX <file.pcap> -dX <y> -rX <z> -iX <w> -cX
```

gdje **X** predstavlja index porta (0-3), **-q** predstavlja funkcionalnost generisanja paketa, **-d** kašnjenje među paketima, **-r** brzinu predaje, **-i** broj iteracija i **-c** funkcionalnost primanja paketa. Vrijednosti uokvirene u zagrade su karakteristične za funkcionalnost uz koju stoje. Tako je **file.pcap** naziv fajla iz kojeg se čitaju paketi koji se šalju sa određenog porta, a **y**, **z** i **w** su broježane vrijednosti koje odgovaraju kašnjenju, brzini predaje i broju iteracija[1].

8. ZAKLJUČAK

NetFPGA razvojna platforma predstavlja hardversku osnovu koja daje mogućnost da se računar opšte namjene pretvori u neku od aktivnih mrežnih komponenti, ako se ispoštuju odgovarajuće instrukcije i sugestije. NetFPGA hardverski obrađuje sve tokove podataka, ostavljajući softveru jedino funkciju kontrolne jedinice, tako da su stvoreni uslovi da implementirani mrežni uređaj bude ranga opreme koju karakterišu brzine do jednog gigabita u sekundi. Ključni doprinos ovog rada je sistematizacija procedura i instrukcija, koje je potrebno primijeniti kako bi se izbjegli problemi pri implementaciji, u ovom slučaju prikazanih na primjeru generatora paketskog saobraćaja.

LITERATURA

- [1] B. Ugren, M. Radonjić, I. Radusinović, Implementacija generatora paketskog saobraćaja na NetFPGA platformi, TELFOR, Beograd, Srbija, 2009.
- [2] M. Casado, N. McKeown, The Virtual Network System, ACM SIGCSE Bulletin, Volume 37, Pages 76-80, 2005.
- [3] G. A. Covington, G. Gibb, J. Naous, J. Lockwood, and N. McKeown, Methodology to contribute netfpga modules, In International Conference on Microelectronic Systems Education (submitted to), 2009.
- [4] J. W. Lockwood, N. McKeown, G. Watson, G. Gibb, P. Hartke, J. Naous, R. Raghuraman, and J. Luo, Netfpga - an open platform for gigabitrate network switching and routing, In International Conference on Microelectronic Systems Education, San Diego, CA, 2007.
- [5] G. A. Covington, G. Gibb, J. Naous, J. Lockwood, and N. McKeown, A Packet Generator on the NetFPGA Platform, NetFPGA website <http://netfpga.org/>.

UPOREDNA ANALIZA IZRAČUNAVANJA VIŠIH MOMENATA DUŽINE M/M/c REDA ČEKANJA

COMPARASION ANALYSIS OF METHODS FOR HIGHER MOMENTS CALCULATION OF M/M/c QUEUE LENGHT

Dajana Tomić, Crnogorski Telekom, Igor Radusinović, Elektrotehnički fakultet, Podgorica

Sadržaj – Problemi koji uključuju momente slučajnih promjenljivih prisutni su u mnogim oblastima nauke. Primjena viših momenata je povezana sa rešavanjem problema optimizacije. Cilj ovog rada je da predstavi uporednu analizu dva različita pristupa određivanju momenata dužine reda čekanja kod M/M/c reda. Počevši od osnovnih tranzitnih jednačina reda, može se dobiti rekurzivna formula za momente dužine reda čekanja. Ovaj metod predstavlja alternativu izvođenju viših momenata korišćenjem klasičnog pristupa koji uključuje traženja izvoda funkcije generisanja momenata. Poređenje rezultata dobijenih upravo na osnovu ova dva načina biće predstavljeno u radu.

Abstract - Problems involving moments of random variables arise in many areas of science. An application of the higher moments of queue length distribution is related to optimization problem. The aim of this paper is to present comparasion analysis of two diffrent approaches to determination of moments of queue length distribution for the M/M/c queue. Starting from the basic state transition equations of the queue, the recursive formula for the moments of queue length distribution could be obtained. This method provides an alternative to derivation of higher moments of queue length by using classic method which includes taking the derivatives of Moment Generating Function. Comparasion of results obtained by using this two methods will be presented in this paper.

Ključne riječi – M/M/c red čekanja, momenti dužine reda čekanja, rekurzivna formula, funkcija generisanja momenata, uporedna analiza

1. UVOD

Viši momenti dužine reda čekanja imaju važnu ulogu prilikom rešavanja problema optimizacije. Korišćenje viših momenata je relevantno prilikom određivanja ograničenja performansi ili prilikom analize tranzitnog ponašanja sistema [1]. Problem određivanja granica vjerovatnoće da određena slučajna promjenljiva pripada zadatom opsegu može biti riješen upotrebom date informacije o momentima slučajne promjenljive [2]. Momenti distribucije veličine reda, na primjer, mogu odrediti granice za srednje vrijeme čekanja. Međutim, aproksimacije zasnovane samo na prva dva momenta veličine reda ne mogu biti dovoljno precizne za sve tipove distribucije. Dokazano je u literaturi da treći moment, kao i ostali viši momenti veličine reda imaju značajan uticaj na srednje vrijeme čekanja [3].

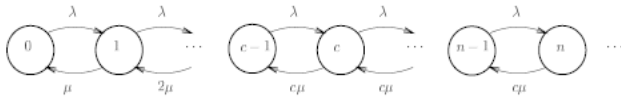
Momenti dužine reda čekanja se obično izračunavaju upotrebom funkcije generisanja momenta (FGM). Alternativni metod opisan je u radu [5]. Kao osnova tog rada poslužila su prethodna istraživanja za M/M/1 i M/M/1/B redove čekanja prikazana u literaturi [1]. Suština ovog pristupa je da se, počevši od osnovnih balansnih jednačina reda, izvedu rekurzivne formule za momente dužine reda čekanja. Pokazano je da ovaj metod može biti primijenjen na više različitih sistema redova čekanja. Rekurzivne formule su izvedene i dokazane za M/M/c, M/M/c/0 i M/M/∞ redove, kao i kompleksnije sisteme kao što su M/E_r/1 i E_r/M/1

redovi, gdje E_r označava Erlangov server sa r nivoa. Radi verifikacije prethodno izvedenih rekurzivnih formula, u okviru ovog rada biće izvršeno poređenje dobijenih rezultata sa FGM. Prema tome, ovaj rad pretstavlja logičan nastavak istraživanja započetog u radu [5].

Rekurzivna formula dužine reda čekanja za M/M/c će biti opisana u drugom poglavlju. Kratak osvrt na primjenu FGM u slučaju M/M/c reda biće prikazan u trećem dijelu. U četvrtom dijelu je dato poređenje rezultata dobijenih korišćenjem FGM i rekurzivne formule. Konačno, u poslednjoj sekciji biće izneseni zaključci.

2. REKURZIVNA FORMULA ZA MOMENTE M/M/c REDA ČEKANJA

M/M/c je dobro poznata Kendalova notacija reda čekanja sa c identičnih servera, gdje paketi dolaze u skladu sa Poisson-ovom raspodjelom srednjom dolaznom brzinom λ, a vrijeme posluživanja ima eksponencijalnu raspodjelu srednje vrijednosti 1/μ [6]. Dijagram stanja M/M/c reda čekanja prikazan je na Slici 1. Stanje sistema je u potpunosti okarakterisano brojem paketa koji se nalaze u redu u datom trenutku. Ukoliko je broj paketa u sistemu n < c, paket se direktno poslužuje sledećim dostupnim serverom, u suprotnom, paket se smješta u red [7].



Slika 1: Dijagram stanja za M/M/c red čekanja

Intenzitet saobraćaja je $\rho = \frac{\lambda}{c\mu}$. Neka je P_n ($n = 0, 1, 2, \dots$)

definisano kao vjerovatnoća da red u ravnotežnom stanju sadrži n paketa (uključujući c paketa koji se servisiraju). Ove vjerovatnoće zadovoljavaju balansne jednačine sistema u ravnotežnom stanju:

$$\begin{cases} \mu P_1 = \lambda P_0 \\ (n+1)\mu P_{n+1} = (\lambda + n\mu)P_n - \lambda P_{n-1}, 1 \leq n \leq c-1 \\ c\mu P_{n+1} = (\lambda + c\mu)P_n - \lambda P_{n-1}, n \geq c \end{cases} \quad (1)$$

Na osnovu (1) se može dobiti funkcija raspodjele paketa kao:

$$P_n = \begin{cases} \frac{(c\rho)^n}{n!} P_0, n = 0, 1, \dots, c-1 \\ \frac{c^c \rho^n}{c!} P_0, n = c, c+1, \dots \end{cases} \quad (2)$$

Gdje je P_0 vjerovatnoća da je sistem prazan.

Moment k -tog reda se može odrediti na sledeći način:

$$M_k = \sum_{n=0}^{\infty} n^k P_n \quad (3)$$

Teorema 1:

U M/M/c redu čekanja sa srednjom dolaznom brzinom λ i srednjim vremenom posluživanja $1/\mu$, kada je opterećenje reda $\rho = \lambda/c\mu < 1$, k -ti moment dužine reda M_k ($k \geq 1$) zadovoljava izraz:

$$M_k = \frac{\sum_{l=2}^{k+1} \binom{k+1}{l} [(-1)^l + \rho] M_{k+1-l} - (-1)^{k+1} P_0}{(1-\rho)(k+1)} - \frac{\sum_{n=1}^{c-1} n^{k+1} P_0 \frac{(\rho c)^n}{n!} \left(\rho \left(\frac{c}{n+1} - 1 \right) - \left(1 - \frac{n}{c} \right) \right)}{(1-\rho)(k+1)} \quad (4)$$

Dokaz prethodne formule detaljno je opisan u radu [5].

3. PRIMJENA FUNKCIJE GENERISANJA MOMENATA

Funkcija generisanja slučajne promjenljive je očekivana vrijednost određene transformacije date promjenljive. Sve

funkcije generisanja imaju veoma bitne karakteristike prilikom određivanja svojstva promjenljive. Ukoliko su zadovoljeni određeni uslovi, funkcija u potpunosti određuje raspodjelu promjenljive [4]. Bitna je činjenica da momenti slučajne promjenljive mogu biti određeni korišćenjem izvoda funkcije generisanja [8]. FGM za diskretne promjenljive se u opštem slučaju definiše na sledeći način:

$$M_X(t) = E(e^{tX}) = \sum_{k=0}^{\infty} \frac{\mu_k t^k}{k!} = E\left(\sum_{k=0}^{\infty} \frac{X^k t^k}{k!}\right) = \sum_{j=1}^{\infty} e^{tX_j} p(x_j) \quad (5)$$

$$M_{(Z)} = \sum_{k=0}^{\infty} Z^k p(k), \text{ pri čemu je } Z = e^t \quad (6)$$

gdje je p funkcija gustine raspodjele vjerovatnoće.

Momenti se mogu definisati kao set numeričkih vrijednosti koji neosporno određuju distribuciju vjerovatnoće slučajne promjenljive [4]. Poznavanje prva dva momenta omogućava određivanje srednje vrijednosti ($E[X] = \mu$) i varijanse ($V(X) = \sigma^2$). Međutim, ove dvije numeričke vrijednosti ne sadrže sve potrebne informacije o funkciji raspodjele. Radi jednoznačnog određivanja funkcije raspodjele slučajne promjenljive neophodno je poznavati i više momente. Utvrđeno je da različite distribucije ne mogu imati identične momente. Moglo bi se očekivati da se vrijednosti μ i σ^2 , kao i vrijednosti viših momenata, uvijek mogu odrediti na osnovu funkcije gustine raspodjele. U praksi se, ipak, pokazalo da su ova izračunavanja često vrlo komplikovana.

Za slučajnu promjenljivu k -ti moment se može odrediti pomoću izvoda FGM, koji se mogu dobiti na osnovu Z -transformacije:

$$E[Z^k] = M_z^{(k)}(0) = \left. \frac{d^k}{dt^k} M_{(Z)} \right|_{Z=1}, k = 1, 2, 3... \quad (7)$$

Prvi momenat za red M/M/c se izračunava kao:

$$M_1 = \left. \frac{d}{dt} M_{(Z)} \right|_{Z=1} = E[N] = c\rho + \rho \frac{(c\rho)^c}{c!} \frac{P_0}{(1-\rho)^2} \quad (8)$$

Daljim diferenciranjem FGM i korištenjem relacija koje važe među momentima, mogu se izvesti opšti oblici za drugi i teći moment.

4. UPOREDNA ANALIZA DOBIJENIH REZULTATA

U ovom dijelu rada je data uporedna analiza rezultata dobijenih korišćenjem rekurzivne formule i rezultata dobijenih upotrebom FGM. Osnovni zadatak analize jeste provjera validnosti rekurzivne formule. Posmatrane su vrijednosti prva tri momenta u zavisnosti od opterećenja M/M/c reda (koje varira od 0 do vrijednosti bliske 1) i broja servera u sistemu. Numeričke vrijednosti, kao i grafički prikaz dobijeni su upotrebom programskog paketa MATLAB.

Na Slici 2 se nalaze grafici na kojima je predstavljeno poređenje rezultata kada se u sistemu nalazi samo jedan server, tj. $c=1$. Za ovaj granični slučaj formula (4) prelazi u:

$$M_k = \frac{\sum_{l=2}^{k+1} \binom{k+1}{l} [(-1)^l + \rho] M_{k+1-l} - (-1)^{k+1} [1-\rho]}{(k+1)(1-\rho)} \quad (9)$$

Funkcija generisanja momenata ima oblik:

$$M_{(z)} = \frac{1-\rho}{1-\rho z} \quad (10)$$

Odgovarajući izrazi za prva tri momenta dobijena na osnovu prethodne formule su:

$$M_1 = \frac{\rho}{1-\rho} \quad (11)$$

$$M_2 = \frac{\rho(1+\rho)}{(1-\rho)^2} \quad (12)$$

$$M_3 = \frac{\rho(1+4\rho+\rho^2)}{(1-\rho)^3} \quad (13)$$

Naredni posmatrani slučaj je kada broj servera u sistemu teži ∞ . U tom slučaju formula (4) se se transformiše u:

$$M_{k+1} = \frac{\sum_{l=2}^{k+1} \binom{k+1}{l} [(-1)^l \mu M_{k+2-l} + \lambda M_{k+1-l}] + \lambda(k+1)M_k}{\mu(k+1)} \quad (14)$$

Za ovaj red čekanja FGM ima oblik:

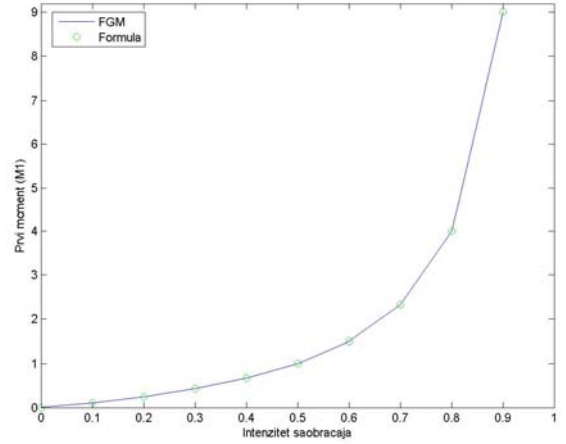
$$M_{(z)} = e^{\rho(z-1)} \quad (15)$$

Na osnovu prethodne jednačine, dobijaju se izrazi:

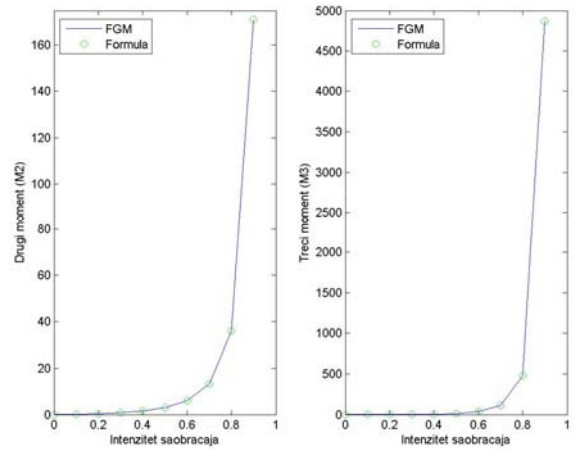
$$M_1 = \rho \quad (16)$$

$$M_2 = \rho(1+\rho) \quad (17)$$

$$M_3 = \rho(1+3\rho+\rho^2) \quad (18)$$



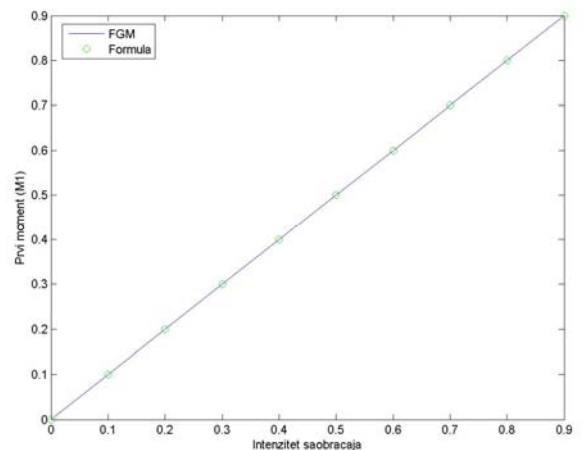
a)



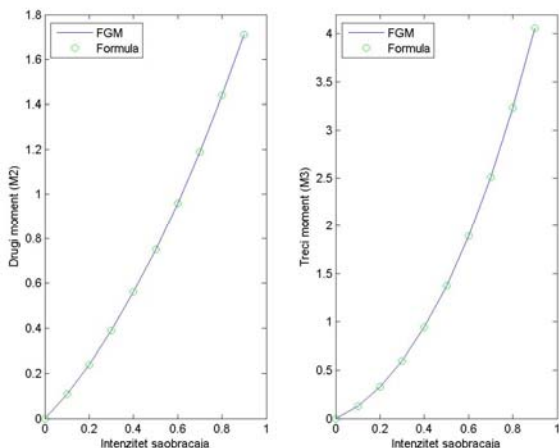
b)

Slika 2: Uporedna analiza rezultata za $c=1$: a) prvi moment, b) drugi i treći moment

Slika 3 prikazuje poređenje rezultata za gore navedeni slučaj.



a)



b)

Slika 3: Uporedna analiza rezultata za $c \rightarrow \infty$: a) prvi moment, b) drugi i treći moment

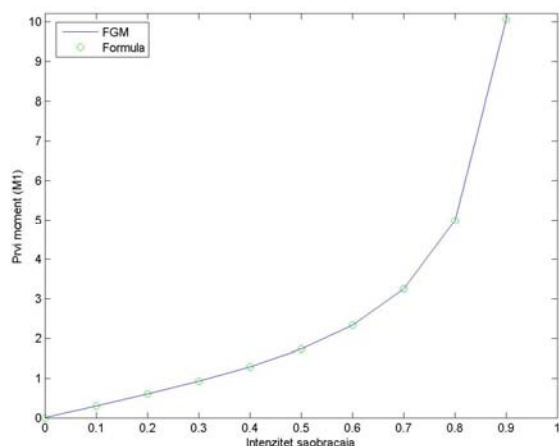
Ukoliko se posmatra slučaj kada je broj servera u sistemu konačan broj različit od 1, recimo $c=3$, za izračunavanje momenata prema novom metodu koristi se formula (4). Kada je FGM u pitanju, na osnovu (8), nakon što se uvrsti da je $c=3$, mogu se dobiti sledeći izrazi:

$$M_1 = 3\rho + \frac{4.5\rho^4 P_0}{(1-\rho)^2} \quad (19)$$

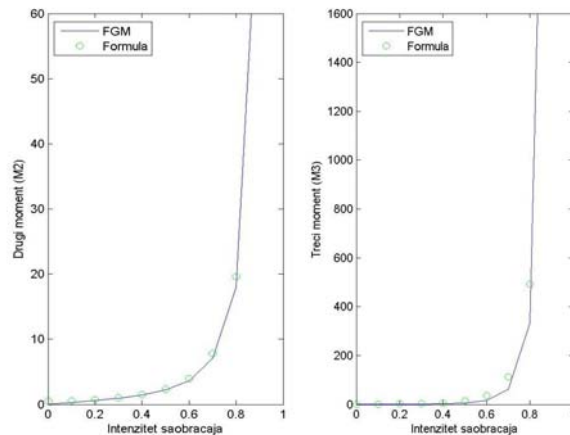
$$M_2 = 3\rho + \frac{4.5\rho^4 P_0}{(1-\rho)^2} + \frac{4.5\rho^4 P_0}{(1-\rho)^3} \quad (20)$$

$$M_3 = 3\rho + \frac{4.5\rho^4 P_0}{(1-\rho)^2} + \frac{13.5\rho^4 P_0}{(1-\rho)^3} + \frac{9\rho^4 (\rho+1)P_0}{(1-\rho)^4} \quad (21)$$

Na Slici 4 predstavljen je grafički prikaz poređenja rezultata kada je $c=3$.



a)



b)

Slika 4: Uporedna analiza rezultata za $c=3$: a) prvi moment, b) drugi i treći moment

5. ZAKLJUČAK

Predstavljeno je poređenje dva različita pristupa određivanju viših momenata dužine M/M/c reda čekanja. Metod koji uključuje rekurzivnu formulu je nedavno razvijen. Princip na kom je ovaj metod zasnovan, kao i njegova primjena na različite sisteme reda čekanja, je detaljno elaboriran. Svrha ovog rada je nastavak istraživanja kojim bi se verifikovao novi metod na konkretnim primjerima uz grafički prikaz poređenja sa klasičnim metodom, što je i pokazano. Klasični metod zasnovan je na korišćenju izvoda funkcije generisanja momenata. Osnovna prednost novog pristupa nad FGM je jednostavnost izračunavanja.

LITERATURA

- [1] J. Liu., H. Jiang, and H. Susumu, "Recursive formula for the moments of queue length in the M/M/1 queue", IEEE communications letters, vol. 12, No. 9, September 2008
- [2] D. Bertsimas, I. Popescu and J. Sethuraman, "Moment Problems and Semidefinite Optimization", Sloan School of Management and Operations Research Center, Cambridge, April 2000
- [3] V. Gupta, J.Dai, M. Harchol-Balter and B. Zwart, "The effect of higher moments of job size distribution on the performance of an M=G=K queueing system", School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, February 2008
- [4] S. Spataru, "Applying the Moment Generating Functions to the Study of Probability Distributions", Academy of Economic Studies, Bucharest, 2007
- [5] D. Tomic, I. Radusinovic, "Recursive Formula for the Moments of Queue Length in the M/M/c and Erlangian Queues", Telfor, Beograd, Novembar 2009
- [6] J. R. Artalejo and M. Pozo, "Numerical Calculation of the Stationary Distribution of the Main Multiserver Retrial Queue", Springer, vol. 116, pp 41-56, October 2002
- [7] I. Adan and J. Resing, "Queueing Theory", Eindhoven University of Tehnology, February 2001
- [8] Generating Functions, Chapter 10, Dartmouth College

IMPLEMENTACIJA SERIJSKE KOMUNIKACIJE SA SPARTAN 3E FPGA RAZVOJNOM PLATFORMOM

IMPLEMENTATION OF SERIAL COMMUNICATION WITH SPARTAN 3E FPGA DEVELOPMENT PLATFORM

Danilo Mišović, Milutin Radonjić, Igor Radusinović, *Elektrotehnički fakultet Podgorica*

Sadržaj- U ovom radu opisana je implementacija serijske komunikacije između personalnog računara i Spartan 3E FPGA razvojne platforme. Cilj komunikacije je da se kroz odgovarajući set komandi kontroliše rad periferija koje se nalaze na razvojnoj pločici. S obzirom na značaj u ostvarivanju serijske komunikacije i kontrole periferija, posebno su izdvojeni realizacija UART-a i PicoBlaze mikrokontrolera kao i njihova međusobna interakcija. Opisan je način funkcionisanja pojedinih elemenata komunikacionog kontrolera i date su instrukcije za njihovu uspješnu realizaciju.

Abstract- In this paper implementation of serial communication between personal computer and Spartan 3E FPGA development platform is described. The aim of communication is control function of peripherals which are situated on development board by using a suitable set of commands. Considering importance in realizing the serial communication and controlling peripherals, realizations of UART and PicoBlaze microcontroller are specially highlighted as their mutually interaction. It is also described a way of functioning particular elements of communication's controller and instructions for their successful realization are given.

1. UVOD

Razvoj programabilnih logičkih kola donio je značajne promjene u postupku projektovanja digitalnih sistema. Od pojave prvih programabilnih logičkih kola koja su se mogla programirati samo jednom, do današnjih izuzetno moćnih razvojnih platformi, prošlo je jako malo vremena. Njihova primjena je danas toliko velika da se rješavanje ogromnog broja problema teško može zamisliti bez njihove upotrebe. Performanse koje ih karakterišu ali i mogućnosti koje nude, posebno izdvajaju FPGA (*Field Programmable Gate Arrays*) čipove. FPGA integriše mnoštvo komponenti srednjeg stepena integracije. Primjenom FPGA čipova moguće je značajno smanjiti složenost, veličinu, pa i cijenu uređaja, jer u tipičnim realizacijama oni zamjenjuju desetine do stotine MSI (*Medium Scale Integration*) sklopova. Velika prednost ovih čipova je u njihovoj reprogramabilnosti, tj. konfiguraciju i funkcionalnost je moguće mijenjati i unutar gotovog sklopa, pa čak i tokom rada. FPGA kolo se sastoji od velikog broja identičnih logičkih ćelija koje se naknadno povezuju kako bi se ostvarila željena funkcija [1].

U realizaciji velikog broja aplikacija, pored FPGA čipa, neophodno je primjeniti i mikrokontrolere. Prvenstveno zbog svoje cijene i fleksibilnosti, mikrokontroleri nalaze sve veću primjenu u mnogim oblastima, prilikom izrade softverski kontrolisanih uređaja i sistema. Velika prednost mikrokontrolera jeste činjenica da se mogu programirati u višim programskim jezicima. Mogućnost realizacije jezgra mikrokontrolera na jednom FPGA čipu programabilne pločice značajno olakšava upravljanje svim njenim djelovima. Prije pojave ugrađenih procesora, među kojima je i PicoBlaze, mikrokontroler se nalazio van FPGA čipa što ograničava mogućnost povezivanja sa drugim elementima čipa i smanjuje ukupne performanse sistema. Nasuprot tome, potpuno ugrađeni mikrokontroleri unutar FPGA se karakterišu fleksibilnom, proširivom *on-chip* konektivnošću

ka drugim resursima čipa. Signal ostaje unutar FPGA poboljšavajući ukupne performanse i smanjujući cijenu sistema.

Značaj što bržeg i pouzdanijeg prenosa informacija doveo je do pojave različitih tehnika prenosa podataka. Jedna od najranijih i najvažnijih je serijska komunikacija. Njena jednostavnost implementacije i mogućnost prenosa koristeći samo jednu dvožičnu liniju doveo je do toga da je skoro svaki računar imao ugrađen serijski port [2].

Spartan 3E familija FPGA kola je posebno dizajnirana da udovolji potrebama velikog broja projekata koji zahtijevaju povoljan odnos cijena/performance. Ova familija nadograđuje uspjeh ranijih Spartan 3 familija povećavajući količinu logike, pritom značajno smanjujući troškove po logičkoj ćeliji. Xilinx-ova Spartan 3E razvojna pločica obezbjeđuje moćnu i veoma naprednu razvojnu platformu za dizajn složenih uređaja. Poboljšanja koja uvodi Spartan 3E omogućavaju više funkcionalnosti i propusnog opsega nego što je bilo moguće ranije, po istim cijenama, postavljajući nove standarde u industriji programabilne logike [3].

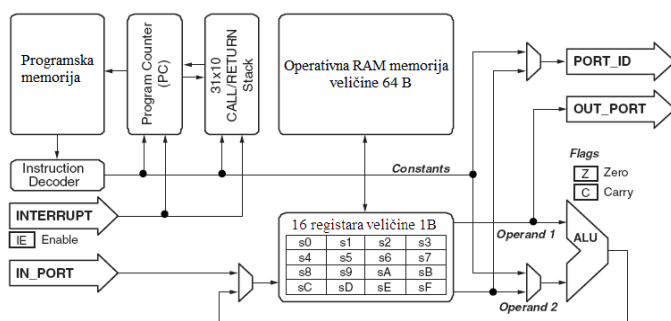
O PicoBlaze mikrokontroleru, osobinama i signalima koji ga karakterišu se govori u poglavlju 2. Implementacija UART predajnika i prijemnika, kao ključne komponente koja omogućava serijsku komunikaciju, je obrađena u poglavlju 3. U poglavlju 4 je opisana interakcija Spartan 3E pločice sa računarom kroz razmjenu informacija koje se tiču pojedinih periferija pločice. Zaključne konstatacije su navedene u poglavlju 5.

2. PICOBLAZE MIKROKONTROLER

PicoBlaze mikrokontroler je kompaktan, moćan i ekonomičan 8-bitni mikrokontroler optimizovan za Spartan-3, Virtex-II i Virtex-II Pro FPGA familije. Izuzetno efikasno

koristi resurse tako da izvodi zavidnih 44 do 100 miliona instrukcija u sekundi u zavisnosti od FPGA familije i njenih vremenskih zahtjeva. Jezgro PicoBlaze mikrokontrolera je u potpunosti ugrađeno u FPGA i ne zahtijeva spoljašnje resurse. Izuzetno je fleksibilan i osnovne funkcionalnosti se jednostavno proširuju i nadograđuju povezivanjem dodatne FPGA logike na ulazne i izlazne portove mikrokontrolera. Zbog toga što je implementiran kao VHDL program, PicoBlaze mikrokontroler je imun na zastarijevanje i kao mikrokontroler može biti ponovo biran za buduće generacije Xilinx FPGA čipova.

Na slici 1. je prikazana arhitektura PicoBlaze mikrokontrolera. Ovdje se radi o KCPSM3 verziji koja uvodi operativnu RAM memoriju veličine 64B, ali smanjuje broj registara sa 32 na 16. Programska memorija omogućava smještanje do 1024 instrukcije dok CALL/RETURN *stack* čuva do 31 adresu i na taj način omogućava gniježđenje do 31 CALL sekvence [4].



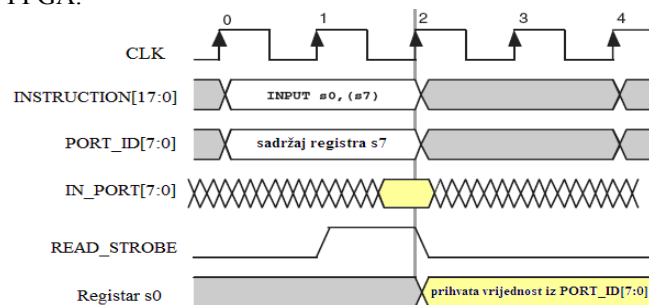
Slika 1. Arhitektura PicoBlaze mikrokontrolera

PicoBlaze mikrokontroler podržava i do 256 ulaznih portova i 256 izlaznih portova ili kombinaciju ulazno/izlaznih portova. *Input* operacija smješta podatke dostavljene na ulazni port u bilo koji od 16 internih registara. PORT_ID izlazni port, definisan ili sadržajem nekog od registara ili kao 8-bitna konstanta, bira željeni ulaz. Ulazi se, generalno, biraju preko multipleksera, koristeći raspored bita iz PORT_ID izlaznog porta. Veličina multipleksera proporcionalna je broju mogućih ulaza, što direktno utiče na performanse. INPUT operacija zahtijeva pridruženi READ_STROBE izlazni signal na drugom od dva ciklusa INPUT operacije kao što je prikazano na slici 2. On označava da je PicoBlaze mikrokontroler primio podatke.

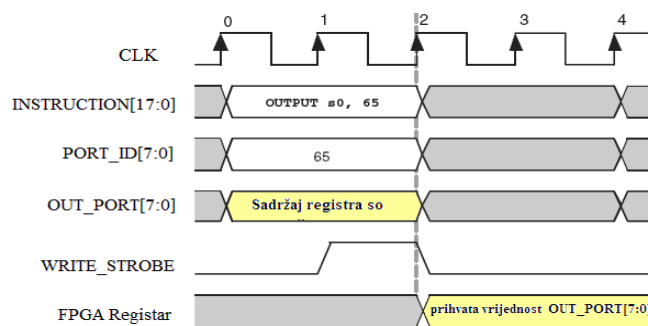
Output operacija sadržaj bilo kojeg od 16 registara prenosi na OUT_PORT izlazni port. PORT_ID zadaje željenu izlaznu destinaciju. WRITE_STROBE signal pokazuje da su podaci na OUT_PORT portu validni i spremni za prihvatanje kao što je prikazano na slici 3.

Zahtjevi za resursima kod PicoBlaze mikrokontrolera ostaju konstantni uprkos povećanju složenosti. Ipak sekvencijalno izvršavanje instrukcija značajno degradira performanse sistema. Kako se povećava složenost, broj instrukcija zahtijevanih za implementaciju aplikacije raste i performanse sistema se time smanjuju. Zbog toga se mikrokontroler koristi za implementaciju složenih kontrolnih funkcija koje nijesu vremenski kritične. Takođe, kao nedostatak je značajno izdvojiti da zahtjevi za memorijom rastu sa povećanjem složenosti aplikacije. S druge strane,

FPGA karakterišu značajno bolje performanse i mogućnost paralelnog izvršavanja instrukcija, ali i značajno veća složenost programiranja aplikacija. Zbog toga se kao najbolje rješenje nameće realizacija PicoBlaze mikrokontrolera unutar FPGA.



Slika 2. Vremenski dijagram za INPUT operaciju



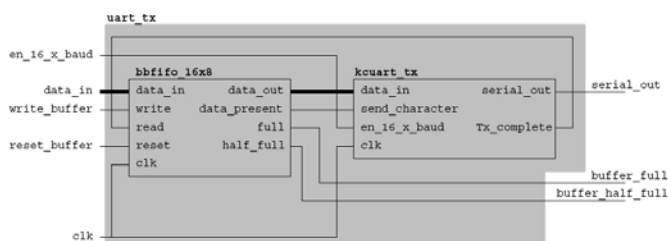
Slika 3. Vremenski dijagram za OUTPUT operaciju

3. IMPLEMENTACIJA UNIVERZALNOG ASINHRONOG PREDAJNIKA/PRIJEMNIKA

Univerzalni asinhroni predajnik/prijemnik (UART) je ključna komponenta serijske komunikacije. On ima zadatak da prihvati bajtove podataka i sekvencijalno prenosi bit po bit. Na prijemu, drugi UART na osnovu primljenih bita formira bajt tako što vrši konverziju podataka iz serijske u paralelnu formu. Zbog toga svaki prijemni UART za ovu konverziju posjeduje pomjerački registar [5]. Serijski prenos digitalne informacije preko odgovarajućeg medijuma koji zahtijeva samo jednu signalnu liniju ima mnogo nižu cijenu nego paralelni prenos koji zahtijeva više linija. Upravo ta činjenica serijski prenos čini i danas aktuelnim pa samim tim i primjenu UART-a.

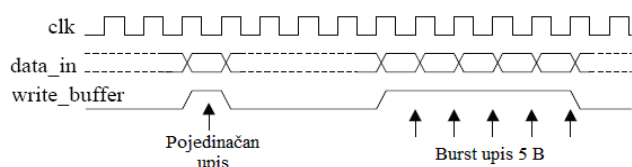
Na slici 4. su prikazani signali UART predajnika realizovanog unutar FPGA čipa. UART predajnik, predstavljen *uart_tx.v* fajlom, nastaje kombinovanjem FIFO bafera (*bbfifo_16x8.v* fajl), i *kcuart_tx.v* modula koji treba da izvrši zahtijevanu konverziju. Na *data_in* portu se dovode podaci koje treba poslati. Ti podaci se upisuju u bafer veličine 16B onda kada je *write_buffer* signal logička jedinica, na uzlaznoj liniji takta. Ukoliko postane pun, FIFO bafer će odbaciti podatke. Ako je *reset_buffer* logička jedinica, unutrašnji bafer veličine 16B će biti resetovan i svi podaci koji se trenutno nalaze u baferu se gube. Signal *en_16_x_baud* omogućava utvrđivanje vremenske reference za serijski prenos podataka. Ovaj signal bi trebao biti logička jedinica u toku trajanja samo jednog takt ciklusa, i trebao bi imati brzinu koja je (približno) 16 puta veća od brzine pri kojoj se obavlja serijski prenos. Alternativno, ovaj signal bi mogao biti konstantno postavljen na logičku jedinicu tako da

se serijski prenos podataka vrši na maksimalnoj brzini od $clk/16$ bita u sekundi. Signali *buffer_full* i *buffer_half_full* su logičke jedinice onda kada je bafer pun odnosno kada se u baferu nalazi 8 ili više bajta podataka. Signal *buffer_half_full* je izuzetno koristan pokazatelj sistemu da FIFO bafer postaje pun, te da bi bilo pametno smanjiti brzinu kojom se podaci upisuju u njega. Konvertovani podaci se pojavljuju na *serial_out* portu i serijski se prenose do prijemnika.



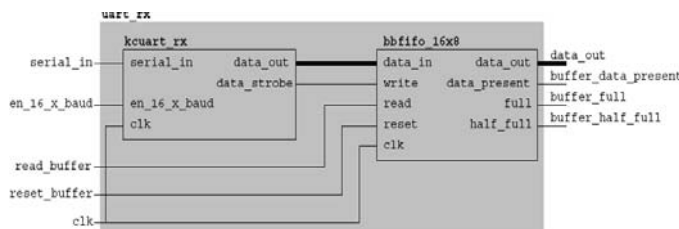
Slika 4. UART predajnik

Vremenski dijagram upisivanja podataka u bafer je prikazan na slici 5. Kao što se vidi, u zavisnosti od intenziteta prenosa ali i slobodnog prostora u baferu, podaci mogu biti zapisani odvojeno ili u grupama od nekoliko bajta. Sadržaj bafera se automatski čita od strane *kcuart_tx* kola, koji ima zadatak da prosljedi podatke na serijsku liniju.



Slika 5. Upis podataka u predajni bafer

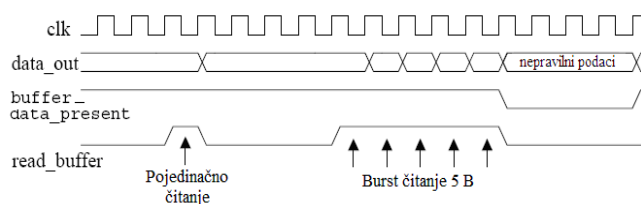
Na slici 6. su prikazani signali UART prijemnika realizovanog unutar FPGA čipa. UART prijemnik, predstavljen *uart_rx.v* fajlom, nastaje kombinovanjem *kcuart_rx.v* modula, koji ima zadatak da izvrši konverziju iz serijskog u paralelni oblik, i FIFO bafera (*bbfifo_16x8.v* fajl). Logička jedinica *read_buffer* signala pokazuje da su podaci koji se nalaze na *data_out* portu pročitani (ili će biti pročitani u narednoj uzlaznoj ivici takta). Bilo koji pokušaj čitanja podataka kada *buffer_data_present* nije aktivan neće imati efekta i ove nedozvoljene pokušaje treba izbjeći. Kada su podaci prisutni (*buffer_data_present=1*), pojedinačno ili čitanje većeg broja bajta iz FIFO bafera može biti realizovano kao što je prikazano na slici 7.



Slika 6. UART prijemnik

Ono što se uočava kao ključna razlika u načinu funkcionisanja UART predajnika i prijemnika je činjenica da kod UART prijemnika ne postoji kontrola upisa podataka u bafer. To znači da prijemnik ima na raspolaganju vremenski interval u trajanju od deset bita koji se serijski prenose da pročita makar jedan bajt podataka iz bafera. U suprotnom novi

podaci će se upisati preko podataka koji su prvi pristigli. Ovo dovodi do greške koja se naziva *Overrun error*.



Slika 7. Čitanje podataka iz bafera

4. KONTROLA PERIFERIJA NA SPARTAN 3E RAZVOJNOJ PLOČICI

Povezivanje računara i Spartan 3E razvojne pločice korišćenjem serijskog porta omogućava razmjenu poruka između njih. Da bi računar mogao da šalje i prima poruke potrebno je koristiti neki od dostupnih terminalskih programa. Terminal je taj koji omogućava unos komande koju čine ASCII karakteri. U ovom slučaju komande, koje se unose na terminalu, su identifikatori na osnovu kojeg se donosi odluka koju od aktivnosti treba realizovati i predstavljaju prvi korak u interakciji sa pločicom. Spartan 3E pločica sadrži nekoliko perifera koje se često koriste u realizaciji različitih aplikacija. Na primjer, LCD je praktičan način za prikazivanje različitih informacija koristeći standardne ASCII i prilagođene karaktere. LED diodama se često signalizira aktivnost određenih signala. *Rotary encoder* se može koristiti za podešavanje parametara u skladu sa postojećim uslovima, inkrementiranjem ili dekrementiranjem. Prekidačima se pokreće ili zaustavlja određena operacija. Poruke koje šalje računar usmjerene su upravo na gore navedene periferije i imaju za cilj da kontrolišu njihov rad. Konkretno, odgovarajućim komandama, se realizuje čitanje trenutnog statusa prekidača, uključivanje ili isključivanje LED dioda, čitanja trenutne pozicije *rotary encoder*-a i ispisivanje poruka sa terminala na LCD displej. Hardverski svaka od navedenih perifera mora se preko multipleksera povezati sa mikrokontrolerom. Multiplekser na osnovu vrijednosti PORT_ID porta prenosi odgovarajuće informacije do mikrokontrolera. Softverski dio programa se odnosi na programiranje mikrokontrolera koji svaku informaciju koju primi treba na odgovarajući način da obradi. Nakon slanja komande (u vidu ASCII karaktera), mikrokontroler ima zadatak da najprije provjeri njenu ispravnost. Tek nakon toga se prelazi na željenu periferiju.

Hardverski, od *rotary encoder*-a polaze dvije linije. Jedna linija je direktno povezana sa INTERRUPT ulazom PicoBlaze-a i prenosi signal svaki put kada se *rotary* pomjeri u bilo koju stranu. Drugom linijom se signal prenosi samo ako se *rotary* pomjeri u lijevu stranu. S obzirom da ne postoji univerzalni sistem obilježavanja pozicija *rotary encoder*-a, kao početna pozicija se uzima ona u kojoj se *rotary* nalazio u momentu uključivanja uređaja. Pošto je neophodno ažurirati stanje svaki put kada *rotary* promijeni poziciju, najlakši način da se isprati svaka promjena je pomoću prekida. Dakle, svaka promjena pozicije *rotary*-ja generiše signal i dovodi do prekida. Prekid prisiljava PicoBlaze procesor da izvrši instrukciju CALL 3FF, odmah nakon završetka instrukcije koja se trenutno izvršava. CALL 3FF je poziv potprograma koji se nalazi na posljednjoj memorijskoj lokaciji. Instrukcija

na posljednjoj lokaciji definiše kako aplikacija treba da rukuje sa prekidom. Najčešća instrukcija na lokaciji 3FF je skok na ISR (*Interrupt Service Routine*). U ISR-u se nalazi softverska realizacija koja ima zadatak da ažurira trenutnu poziciju. Ova realizacija podrazumijeva poređenje binarnih informacija koje se prenose dvijema linijama koje polaze od *rotary*-ja. U slučaju da su ove dvije informacije identične, došlo je do rotacije u lijevu stranu i trenutnu poziciju je potrebno smanjiti za jedan. U suprotnom trenutna pozicija se povećava za jedan. Takođe, potrebno je koristiti poseban registar u kojem će se čuvati informacija o trenutnom statusu. Kako *rotary* može biti u jednoj od dvadeset različitih pozicija u programu označenih kao 00-19, dvije situacije je neophodno posebno izdvojiti. To su situacije prelaska iz posljednje pozicije (19) u poziciju 00 i obratno. Nakon određivanja nove pozicije, mikrokontroler šalje poruku u vidu ASCII karaktera računaru koja se ispisuje na terminalu.

Čitanje trenutnog statusa prekidača i uključivanje ili isključivanje LED dioda komandama sa terminala, hardverski, zahtijeva realizaciju 8-bitnih magistrala koje povezuju mikrokontroler i navedene periferije. Softverski dio kod LED dioda se sastoji u slanju binarnih kombinacija shodno unesenim brojevima, a kod prekidača u prepoznavanju binarne kombinacije pristigle sa te periferije. Korišćenje LCD-a je mnogo složenije jer podrazumijeva jako veliki broj podešavanja koje treba izvršiti prije upotrebe. Ta podešavanja se tiču inicijalizacije displeja, njegove konfiguracije i pozicioniranja kursora [6].

Svaki od signala na FPGA čipu se može preciznije opisati korišćenjem UCF (*User Constraint File*) fajla. UCF datoteka sadrži ograničenja logičkog dizajna koja utiču na to kako se on implementira u konkretni FPGA čip. U ovu datoteku mogu se unijeti vremenska ograničenja, ograničenja u razmještaju i rasporedu pinova kao i naponski nivoi signala. UCF ograničenja za DTE i DCE RS – 232 portove, uključujući oznake za ulazno izlazne pinove i standarne korišćene za nivo signala su prikazana u tabeli 1. Kao što se vidi radi se o LVTTTL (*Low Voltage TTL*) standardu.

Tabela 1. UCF ograničenja za DTE i DCE RS – 232 serijski port

NET	LOC	IOSTANDARD	DRIVE	SLEW
"RS232 DTE RXD"	"U8"	LVTTL		
"RS232 DTE TXD"	"M13"	LVTTL	8	SLOW
"RS232 DCE RXD"	"R7"	LVTTL		
"RS232 DCE TXD"	"M14"	LVTTL	8	SLOW

Prilikom realizacije serijskog prenosa između računara i FPGA ploče neophodno je UCF ograničenja definisati upravo onako kako je prikazano u tabeli 1. UCF ograničenja preporučena od strane Xilinx-a izostavljaju definisanje ulazno izlaznog standarda. Koristeći se tako definisanim UCF fajlom javio se problem. Naime, naponski nivoi signala koje generišu računar i pločica su različiti. Zbog toga se može desiti da poruke koje šalje jedna strana ona druga ne prepozna kao validne, što dovodi do nemogućnosti realizacije serijske komunikacije. Adekvatnim definisanjem ulazno izlaznog standarda, u okviru preporučenih UCF ograničenja, prevaziđen je problem naponske neusaglašenosti.

Koliko moćnu i naprednu razvojnu platformu obezbjeđuje Spartan 3E pločica najbolje pokazuje tabela 2. U ovoj tabeli se nalaze podaci o iskorišćenosti dostupnih resursa u realizaciji ovog projekta. Kao što se vidi, realizacija PicoBlaze mikrokontrolera, UART predajnika i prijemnika, kao i memorije za smještanje programa i podataka, zauzela je svega 3% raspoloživih resursa.

Tabela 2. Iskorišćenost dostupnih resursa

Device Utilization Summary				ⓘ
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	158	9,312	1%	
Number of 4 input LUTs	280	9,312	3%	
Logic Distribution				
Number of occupied Slices	182	4,656	3%	
Number of Slices containing only related logic	182	182	100%	
Number of Slices containing unrelated logic	0	182	0%	

5. ZAKLJUČAK

Razvoj mikrokontrolera i mikroprocesora uz pojavu prvih komercijalno dostupnih programabilnih platformi učinio je da logički dizajn bude mnogo fleksibilniji. Ipak, realizacija mikrokontrolera van FPGA čipa značajno smanjuje ukupne performanse sistema. Zbog toga je u ovom radu izabrana realizacija PicoBlaze mikrokontrolera u okviru FPGA čipa. Zbog važnosti u realizaciji serijskog prenosa izdvojena i opisana uloga UART kontrolera. Predstavljen je jedan od načina kontrole periferija a posebno je naglašena važnost potpunog definisanja signala unutar UCF fajla. Izdvojen je problem koji se javio korišćenjem Xilinx-ovog UCF fajla kao i način na koji je taj problem prevaziđen.

Značaj ovog rada je u tome što se analizirani principi mogu koristiti i na drugim razvojnim platformama kao i kod mnogo komplikovanijih uređaja i sistema prisutnih u praksi. Dalji rad bi se mogao bazirati na implemetaciji USB komunikacionog interfejsa kao i svih novijih komunikacionih protokola baziranih na serijskoj komunikaciji koji omogućavaju jako velike brzine prenosa podataka.

6. LITERATURA

- [1] R. Munden, "ASIC and FPGA Verification: A Guide to Component Modelling", Morgan Kaufmann 2004
- [2] R.W.D Nickalls, R. Ramasubramanian, "Interfacing the IBM-PC to medical equipment: the art of the serial communication", June 1994
- [3] Xilinx, "Spartan-3E Starter Kit Board User Guide", March 2006
- [4] Xilinx, "PicoBlaze 8-bit Embedded Microcontroller User Guide for Spartan-3, Virtex II, and Virtex II Pro FPGA", June 2004
- [5] http://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter
- [6] D. Mišović, "Primjer realizacije serijske komunikacije računara i Spartan 3E FPGA pločice", specijalistički rad Oktobar 2009
- [7] K. Chapman, "UART Transmitter and Receiver Macros", Xilinx January 2003
- [8] K. Chapman, "Initial Design for Spartan-3E Starter Kit (LCD Display Control)", February 2006

DIGITALNI „MIKROKONTROLERSKI” SISTEM ZA REGULACIJU TEMPERATURE kod Solarnih Sistema za grijanje tople sanitarne vode i podrške sistemu grijanja DIGITAL "MICROCONTROLLER" SYSTEM FOR TEMPERATURE CONTROL of Solar Systems for heating sanitary hot water and heating support system

Duško Penezić, *Termo-Solar, Bar*

Sadržaj – U radu su prikazani osnovni principi realizacije jednog univerzalnog digitalnog „inteligentnog“ regulatora temperature kod Solarnih Sistema za grijanje tople vode i podrške grijanju stambenih prostorija, korišćenjem mikrokontrolera Microchip PIC18F452 u sprezi sa ostalim elektronskim komponentama sistema. Rad se osim izloženog opisa rada i organizacije sistema u kome digitalni uređaj radi, bavi i opisom prednosti upotrebe ovakvog vida automatske regulacije nad klasičnom „analognom“ regulacijom.

Abstract - This paper presents the basic principles of realization of a universal digital "intelligent" temperature regulator for Solar Systems for heating hot water and heating support residential premises, using the Microchip PIC18F452 microcontroller in conjunction with other electronic components of the system. The paper work than the above description and organization of the system in which digital devices work, and deals with a description of the advantages of using this form of automatic regulation of the classic "analog" regulation.

Ključne riječi – Mikrokontroler, RS232 standard, AD konvertor, digitalni senzori, programator, UART, MAX232, SHIFT REGISTAR, LM705, toplotna pumpa, električni grijač, programski kod

1. UVOD

Današnji trend u oblasti regulacione tehnike je da se analogni regulatori zamjene digitalnim (mikrokontroleri, PLC idrugi digitalni uređaji koji se mogu programirati). Primjena mikrokontrolera na polju tehnike, omogućila je razvoj digitalnog upravljanja procesima, gdje kvalitet upravljanja zavisi od kvaliteta programiranja.

Automatsko upravljanje procesima i mašinama, do pojave mikroracunara bilo je zasnovano na analognoj regulaciji. Digitalno upravljanje otvorilo je mogućnost primjene savremenijih postupaka u radu, koji nisu mogući u radu sa analognim regulatorima. Velika prednost je i fleksibilnost u pogledu upravljanja pomoću softvera, koji se može po potrebi mijenjati i dograđivati u slučaju izmjene funkcije i namjene sistema upravljanja.

Rad sistema za prenos informacija, regulaciju i upravljanje, zasniva se na obradi i prenosu signala. Sama konstrukcija i funkcije sistema zavise od postupka obrade i prenosa signala. Ovi postupci se mogu svrstati u dvije kategorije i to: analogne (kontinualne) i diskretne. U analognom postupku se obrađuje i prenosi signal u cjelini, tj. sve trenutne vrijednosti signala.

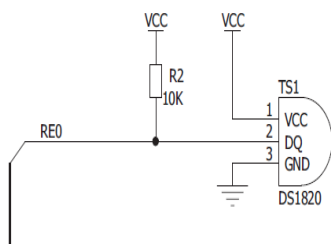
Digitalni sistemi rade u diskretnom postupku, gdje se mjerni signal prvo diskretizuje ili kvantuje po nivou i po vremenu, nakon čega se obrađuju i prenose samo kvantovane vrijednosti. To se postiže amplitudno-kodovanim modulatorom ili digitalnim A/D konvertorom. Na svom izlazu A/D konvertor daje brojne vrijednosti odbiraka u binarnom kodu, odnosno povorci digitalnih riječi dužine izlaznog registra konvertora.

Mjerenje i regulacija temperature spada u red sporih tehnoloških procesa, pa samim time brzina promjene upravljane preomjenljive dozvoljava da perioda odabiranja bude dovoljno velika, tako da mikroprocesor ima na raspolaganju relativno dovoljno vremena, da izvrši sve funkcije neposredno posle trenutka odabiranja, unutar periode odabiranja T. Pri tome, odbirci su jednaki vrijednostima ulaznog signala u trenutcima odabiranja. U preostalom dijelu raspoloživog realnog vremena do sledećeg trenutka mirovanja, procesor može mirovati ili izvršavati druge operacije, vezane za uspješnu realizaciju primjenjenog algoritma upravljanja.

Stalno usavršavanje mikroelektronske tehnologije i algoritama upravljanja, doveli su do velikog interesovanja za digitalnim sistemima upravljanja i sve većoj primjeni u inženjerskoj praksi. Jedna od tih mogućnosti uspješne primjene inteligentnog mikrokontrolerskog regulatora, biće u ovom radu opisana i realizovana kod Solarnih Sistema za grijanje vode.

2. PROJEKTNO RJEŠENJE

Projekat treba da zadovolji funkcionalne zahtjeve koji se pred njega postavljaju: automatsku korekciju temperature u zadatim granicama, da omogući korisniku da ima uvid u trenutno stanje temperature i mogućnost korekcije parametara, za šta treba osmisлити odgovarajući ručni terminal. Treba naravno napraviti program za PC računara, preko koga korisnik može da kontroliše stanje sistema. Prema navedenim zahtjevima osmišljena su sledeća rješenja:



Sl. 3 Blok šema digitalnog termosenzora DS 1820

3. SOFTVER

Pisanje softvera za mikrokontrolere se dosta razlikuje od pisanja softvera za PC računare. Neophodno je poznavanje većeg stepena hardvera i dosta pažljivosti kod inicijalizacije mikrokontrolera, da bi on pravilno radio. Za razvoj softvera u ovom radu korišćen je mikro C kompajler firme Mikro-Elektronika, koji sadrži veliki broj gotovih funkcija za rad sa PIC mikrokontrolerima i umnogome olakšava rad.

Na samom startu (uključenjem napona) ili posle reseta, potrebno je inicijalizovati periferije. Nakon inicijalizacije učitavaju se parametri sistema iz EEPROM-a i na displeju se pojavljuje izvještaj da li je inicijalizacija bila uspješna. Ako jeste program nastavlja sa daljim izvršavanjem.

Program ulazi u beskonačnu petlju, iz koje ne izlazi dok se ne resetuje sistem ili isključi napajanje.

4. REALIZACIJA REGULACIJE

Glavni dio uređaja dakle predstavlja mikrokontroler. Mikrokontroler PIC 18F452 očitava senzore periodično. Kada ustanovi da je mjerena temperatura van zadatog opsega, mora da pokrene proces njenog vraćanja u zadate granice. Proces startuje tako što PIC 18F452 pošalje zahtjev preko UART-a drugom mikrokontroleru istog tipa, koji kontrolira rad izvršnih organa, da aktivira ili isključi električne grijače i motore toplotnih pumpi.

Za ručni terminal u radu je predviđen još jedan mikrokontroler, koji može biti iz serije skromnijih performansi. Odlučio sam se za PIC 16F648 (sl. 4).

4.1 MJERNO-UPRAVLJAČKI BLOK

Mjerno-upravljački blok upravlja procesima u izvršnom organu, provjerava temperaturne senzore, komunicira sa ručnim terminalom i PC računarom. Sastoji se od mikrokontrolera PIC 18F452, stabilizatora napona LM 7805 na +5 V, MAX kola 232 (3 kom) i analognog prekidača 74HC4051. MAX 232 kola služe za komunikaciju sa uređajima tipa PC računara i ima ih tri. Dva izlaza preko ovih kola služe za vezu sa ručnim terminalom i PC računarom a treći je predviđen za nadogradnju sistema (ugradnja modema za GSM protokol).

Obzirom da mikrokontroler raspolaže sa samo jednim UART-om a vrši komunikaciju sa tri uređaja (terminalom,

PC računarom i izvršnim organom), bilo je potrebno ugraditi i jedan analogni prekidač 74HC4051. Njegovim uvođenjem obezbjeđeno je da mikrokontroler dodjeljuje UART liniju perifernim uređajima proizvoljno, o čemu odlučuje mjerno upravljački blok.

4.2 FIRMER

Firmer je osmišljen da radi na sledeći način: Obzirom da PIC18F452 vrši komunikaciju sa tri uređaja i očitavanje senzora, to je odlučeno da postoje dva interapta na jednu sekundu. Prvi interapt se pokreće kad treba da se pokrene sekvenca očitavanja senzora.

Pri svakom interaptu se pokreće konverzija sa drugim sensorom. Razlog za ovo je što je minimalno vrijeme za očitavanje senzora 750 ms. Drugi interapt se dešava, kada treba da se prebaci linija na analognom prekidaču. Dakle, za vrijeme od jedne sekunde, mjerno upravljački blok vrši konverziju podataka samo sa jednim uređajem.

4.3 ELEKTRONIKA U IZVRŠNOM ORGANU

Ima zadatak da upravlja radom grijača i motora toplotnih pumpi.

Utvrđeno je da je potrebno 2 pina za komunikaciju, 3 pina za kontrolu grijača i 3 pina za kontrolu motora toplotnih pumpi. Dakle ukupno 8 pinova. Obzirom da serija 16F raspolaže sa 2 pina za komunikaciju i 13 pinova za ostale komunikacije, to je odlučeno da se ugradi PIC16F648 koji potpuno zadovoljava funkcionalne zahtjeve.

4.4 NAPAJANJE IZVRŠNOG ORGANA

Moguće je iskoristiti napajanje iz PC računara, a takođe i nezavisno napajanje koje posjeduje sam uređaj. Zato je u projektu i predviđen trafo 220/12 (objašnjeno na početku).

U radu je odlučeno da napajanje bude uzeto sa PC-a, onda je opet trebalo ugraditi stabilizator 78L05 koji daje stabilisani napon +5 V za potrebe rada mikrokontrolera

4.5. RUČNI TERMINAL

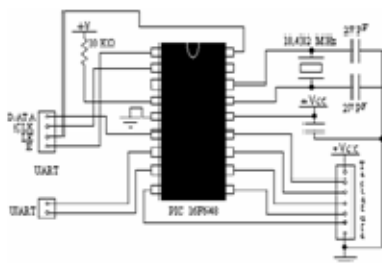
Ručni terminal je zamišljen kao lokalni interfejs, preko koga korisnik pristupa uređaju, provjerava vrijednosti i zadaje nove parametre po želji. Ideja je da se rad prati sa pet tastera. Rad se takođe može kontrolisati i preko PC računara, koji za razliku od ručnog terminala, ne mora uopšte biti blizu solarnog sistema, što je i bio cilj rada. Bitno je u ovom slučaju samo obezbediti vezu preko UART-a između PC-a i projektovanog uređaja.

Terminal se sastoji iz: mikrokontrolera, LCD displeja sa 2x16 karaktera, tastaturom, kolom za napajanje i integralnim kolom MAX232 (sl.4). Mikrokontroler vrši skeniranje tastature, omogućava prikaz na displeju, komunikaciju sa mjerno-upravljačkim blokom, itd. Da bi adekvatno bio odabran mikrokontroler za ovu priliku, bilo je potrebno odrediti tačan broj potrebnih pinova za komunikaciju i kontrolu perifernih uređaja. Analizom je utvrđeno da je potrebno:

- za skeniranje tastature 5 pinova (1 pin po tasteru),
- za komunikaciju preko UART-a dva pina (Rx i Tx),

-LCD je uzet sa 2x16 karaktera. Kontrola i pristup se ostvaruje sa 6 ili sa 10 pinova. Prvi način koristi 4 pina za upis karaktera (od D7....D4), dok drugi koristi 8 pinova (od D7.....DO).

Prema navedenim podacima potrebno je 13 do 17 pinova za kontrolu perifernih uređaja i komunikaciju. Odlučeno je da to opet bude PIC16F648 (sl.4) jer posjeduje dobre karakteristike.

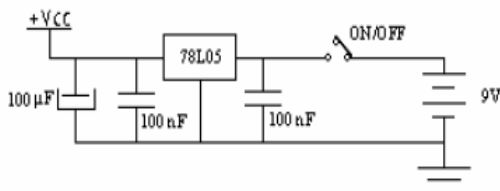


Sl. 4 Mikrokontroler PIC 16F648 u sprezi sa periferijom

4.6. NAPAJANJE TERMINALA

Kolo za napajanje koristi bateriju od 9 V (sl. 5), pa je i ovde potrebno lokalno kolo za stabilizaciju LM7805 na +5V. Prekidač je predviđen zbog manje potrošnje baterijskog napajanja.

Kolo MAX 232 je predviđeno je zbog potrebebe komunikacije terminala sa PC računaram, koji nema mogućnost komunikacije na TTL nivou



Sl. 5 Kolo za napajanje ručnog terminala

5. MOGUĆA POBOLJŠANJA

Moguća poboljšanja ogledaju se u implementaciji memorijskih kartica, USB protokola, ugradnjom modema za GSM protok SMS poruka, itd...

6. ZAKLJUČAK

Nakon testiranja uređaja odlučeno je da se za svaki mikrokontroler ugradi po jedno kolo MAX 232, jer se time dobija na univerzalnosti uređaja, samom mogućnošću da uređaj može da komunicira i sa PC računaram i sa mikrokontrolerima. Pinovi 2,3 i 5 konektora SUB 9 su iskorišćeni za komunikaciju preko UART-a. Prilikom testiranja uređaja u praktičnim uslovima uređaj se pokazao kao dobro rješenje. Nije se dešavalo da zakoči u bilo kom trenutku.

Ručni terminal je pravio manje probleme, pa je kasnije izvršena zamjena mikrokontrolera. Ugrađen je PIC 18F452, posle čega više nije bilo problema.

Upotreba softvera na PC računaru se pokazala kao dobro rješenje i nije postajalo nikakvih problema sa njim u toku ispitivanja uređaja.

Uređaj bi uz dogradnju pomenutih sklopova , mogao predstavljati ozbiljnu komercijalnu konkurenciju, sličnim uređajima iz inostranstva.

LITERATURA

- [1] M. Stojić, *Digitalni sistemi upravljanja*, Beograd, Naučna knjiga, 1989.
- [2] *Matlab*, ver. 5.0, 1996
- [3] Keil Elektronik GmbH, *C-Compiler-51 - User's guide 10.90*, München.
- [4] www.microlink.co.uk
- [5] www.temperatures.com

PERFORMANSE KOMUTATORA SA BAFERIMA U UKRSNIM TAČKAMA CROSSPOINT-QUEUED SWITCH PERFORMANCE

Jelena Čvorović, *Crnogorski Telekom*
Igor Radusinović, Milutin Radonjić, *Elektrotehnički fakultet*

Sadržaj - U ovom radu predstavljen je jedan od načina analize performansi crossbar komutatora sa N^2 bafera u ukrsnim tačkama i slučajnim izborom bafera u koliziji, za uniformni Bernulijev dolazni saobraćaj. U cilju analize performansi, napravljen je analitički iterativni model koji na bazi balansnih jednačina omogućava određivanje propusnosti komutatora i vjerovatnoće gubitka paketa. Poređenjem rezultata dobijenih primjenom analitičkog modela sa simulacionim rezultatima pokazuje se visok stepen približnosti.

Abstract - In this paper we presented performance analysis of the crossbar switch with N^2 crosspoint buffers and random scheduling algorithm under the Bernoulli i.i.d. incoming traffic. We made analytical iterative model based on method of balance equations, which enables us calculating switch throughput and packet loss probability. Comparing our analytic results and simulation results, it appears that they are very close.

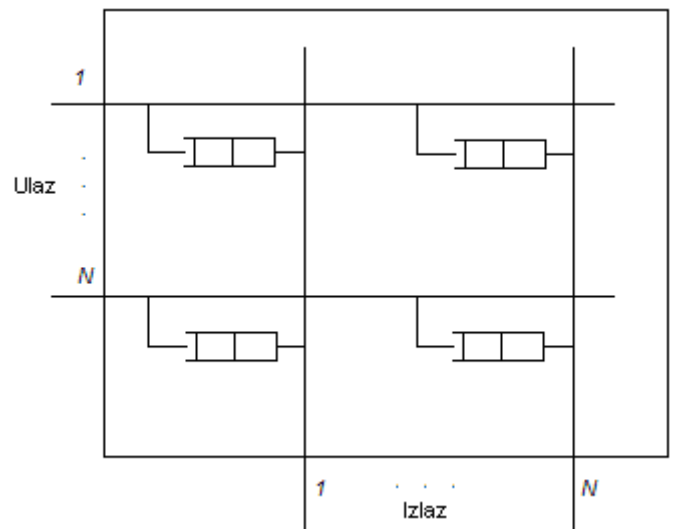
Ključne riječi - balansne jednačine, crossbar komutator, propusnost, vjerovatnoća gubitka

1. UVOD

U ovom radu je predstavljena analiza performansi komutatora sa baferima u ukrsnim tačkama. CPQ (*crosspoint-queued*) komutacioni uređaj je baferovani *crossbar*, koji ima stukturu prikazanu na Slici 1. Paketi fiksne dužine sa ulaza dolaze direktno u baferu u ukrsnim tačkama tj. paketi se ne baferuju na ulazu i ne postoji bilo kakav transfer kontrolnih informacija između ulaza i bafera u ukrsnim tačkama. To znači da nije potreban kompleksni ulazni algoritam raspoređivanja kao kod *crossbar*-a sa Virtual Output Queuing (VOQ) [1] ili kontrola komunikacije između linijskih kartica i algoritma raspoređivanja kao u Combined Input and Crosspoint Queued (CICQ) komutatoru [2]-[3], što predstavlja jednu od najvećih prednosti ovog rješenja. Hardverska implementacija je veoma jednostavna i moguća sa današnjom tehnologijom jer se baferi i komutacioni uređaj mogu implementirati na jednom čipu.

U CPQ arhitekturi prikazanoj na Slici 1, paket koji dolazi sa određenog ulaza namijenjen za određeni izlaz čuva se u ukrsnom baferu. U svakom vremenskom slotu, raspoređivač bira jedan od ukrasnih bafera koji nije prazan i prosleđuje njegov *head of line* (HOL) paket ka izlazu. Ova selekcija se može zasnivati na različitim algoritmima. Posebno su interesantni *work-conserving* algoritmi [4] (OQ, LQF, RR, FRRM, RAND) kod kojih svaki izlaz uvijek opslužuje bafer kada jedan od bafera ima paket za slanje na taj izlaz. Kod komutatora koji se zasniva na LQF (Longest Queue First) algoritmu [5], svaki izlaz opslužuje najzauzetiji bafer na izlaznoj magistrali. Round Robin algoritam [6] poslužuje baferu ciklično bez obzira na prioritet saobraćaja. FBRR (Frame Based Round Robin) [4] algoritam razmatra sve baferu u ukrsnim tačkama u *round-robin* poretku i kada pronade bafer koji nije prazan, poslužuje ga dok ga ne isprazni. FBRR je koristan u slučaju neuniformnog dolaznog saobraćaja, kada ERR (Exhaustive Round Robin) algoritam [7] može izazvati zapostavljanje bafera na koje dolazi nizak nivo dolaznog saobraćaja. U CPQ komutatoru sa random

algoritmom (RAND), na svakoj izlaznoj magistrali se slučajno bira i poslužuje bafer koji nije prazan.



Sl. 1 Komutator sa baferima u ukrsnim tačkama

Neka je N broj ulaza/izlaza komutatora i B veličina bafera u ukrsnim tačkama (Sl. 1). Pretpostavka je da je vrijeme podijeljeno u vremenske slotove fiksne dužine dovoljne za prenos jednog paketa kroz komutator, i da je svaki vremenski slot podijeljen u dvije faze: fazu odlaska i fazu dolaska [8]. Važno je naglasiti da je ovdje razmatran slučaj kada slot počinje fazom odlaska, a zatim nastavlja fazom dolaska paketa.

Performanse ovog komutatora kada je bafer jedinične dužine su već analizirane u radovima [8] i [9]. U radu [8] su izvedeni izrazi za propusnost, vjerovatnoću gubitka i srednje kašnjenje, korišćenjem Z transformacija. U radu [9] je izlazna magistrala komutatora modelovana jednim virtuelnim

baferom veličine N , čije se performanse mogu dobiti razmatranjem diskretnog $M^N / M / 1 / N$ Markovljevog lanca sa $N+1$ stanja.

U ovom radu je prikazan iterativni model za izračunavanje performansi komutatora sa slučajnim izborom bafera u koliziji pri uniformnom dolaznom saobraćaju kada veličina bafera može biti veća od jednog paketa. Ovaj model je baziran na balansnim jednačinama koje opisuju stanje jednog, posmatranog bafera. Kako bi se obezbijedila jednostavnost izračunavanja pretpostavljeno je da stanje posmatranog bafera zavisi od stanja u kome se bafer nalazio u prethodnom slotu, a ne zavisi od stanja u kojima se nalaze ostali baferi na istoj magistrali. Performanse dobijene korišćenjem programa napravljenog u MATLAB-u, su poređene sa simulacionim rezultatima i pokazano je da su rezultati veoma približni. Takođe, rezultati dobijeni na bazi novog modela i rezultati predstavljeni u radovima [8] i [9] za komutator jedinične veličine bafera su veoma bliski. Osnovna prednost novog modela je u jednostavnosti koja ga čini primjenljivim u analizi komutatora sa baferima većim od jedan. Koncepti predstavljeni u radovima [8] i [9] su za slučaj bafera dužine veće od jednog paketa gotovo neupotrebljivi zbog svoje složenosti.

Strukturu ovog rada čini nekoliko poglavlja. U poglavlju koje slijedi nakon uvoda, predstavljen je novi iterativni matematički model za izračunavanje performansi CPQ komutatora sa baferom koji može biti veći od jediničnog. U trećem poglavlju predstavljeno je poređenje analitičkih i simulacionih rezultata. Na kraju, u četvrtom dijelu su dati zaključci.

2. MATEMATIČKI MODEL

A. Balansne jednačine

Pretpostavlja se da se na ulaz CPQ komutatora sa N ulaza i N izlaza dovodi *Bernoulli* i.i.d (*independent and identically distributed*) dolazni saobraćaj. *Bernoulli* i.i.d saobraćaj podrazumijeva da je vrijeme dolaska novog paketa nezavisno i jednako distribuirano na sve izlaze, pri čemu je saobraćaj na svakom ulazu nezavistan od saobraćaja na ostalim ulazima. Vjerovatnoća dolaska paketa na ulaz komutatora je p odnosno vjerovatnoća dolaska paketa na jedan ukrasni bafer iznosi p/N . Vjerovatnoća odlaska zavisi od popunjenosti ostalih bafera na određenom izlazu.

U ovom radu će biti predstavljeni rezultati za komutator koji ima dva ulaza i dva izlaza ($N=2$), pri čemu u svaki ukrasni bafer može biti smješteno do B paketa. Neka je sa $P_{ij}(n)$ označena vjerovatnoća da u n -tom trenutku, bafer i na posmatranoj izlaznoj magistrali ima j paketa. Sa d će biti označena vjerovatnoća odlaska paketa. Tada, za bafer u prvoj ukrasnoj tački, važi:

$$P_{10}(n) = P_{10}(n-1)(1-(p/2)) + P_{11}(n-1)d(1-(p/2)) \quad (1)$$

$$P_{11}(n) = P_{10}(n-1)(p/2) + P_{11}(n-1)[(p/2)d + (1-(p/2))(1-d)] + P_{12}d(1-(p/2)) \quad (2)$$

Za slučaj kada je $2 \leq j \leq B-1$:

$$P_{1j}(n) = P_{1j-1}(n-1)(p/2)(1-d) + P_{1j}(n-1)[(p/2)d + (1-(p/2))(1-d)] + P_{1j+1}(n-1)d(1-(p/2)) \quad (3)$$

Vjerovatnoća da prvi bafer u n -tom trenutku ima B paketa:

$$P_{1B}(n) = P_{1B-1}(n-1)(p/2)(1-d) + P_{1B}(n-1)[(p/2) + (1-(p/2))(1-d)] \quad (4)$$

Kao što se vidi iz prethodnih jednačina, stanje bafera u n -tom slotu zavisi isključivo od stanja u kome se nalazio u prethodnom slotu.

Pretpostavljajući da se zbog uniformnosti saobraćaja može smatrati da oba bafera imaju identičnu raspodjelu stanja, umjesto modelovanja drugog bafera, uzima se da je njegova raspodjela stanja identična posmatranom baferu. Prema tome, vjerovatnoća odlaska d se izračunava po formuli:

$$d(n) = \frac{1}{2}(1 + P_{20}(n-1)) = \frac{1}{2}(1 + P_{10}(n-1)) \quad (5)$$

Kako se na izlazu paket ne pojavljuje samo onda kada su oba bafera prazna propusnost se može izračunati kao

$$T_h = 1 - P_{10}(n)P_{20}(n) = 1 - (P_{10}(n))^2 \quad (6)$$

Kada n teži beskonačnosti propusnost postaje konstantna veličina jer sistem prelazi u ravnotežno stanje.

Vjerovatnoće gubitka se može izračunati po formuli:

$$P_b = \frac{p - T_h}{p} \quad (7)$$

B. MATLAB program

Kako je MATLAB pogodan za matrični račun, balansne jednačine za slučaj kada je $N=2$ predstavljene su na sljedeći način:

$$\begin{pmatrix} P_{10}(n) \\ P_{11}(n) \\ P_{12}(n) \\ \vdots \\ P_{1B}(n) \end{pmatrix} = \begin{pmatrix} 1-\frac{p}{2} & d\left(1-\frac{p}{2}\right) & 0 & \dots & 0 \\ \frac{p}{2} & \frac{p}{2}d + \left(1-\frac{p}{2}\right)(1-d) & d\left(1-\frac{p}{2}\right) & \dots & 0 \\ 0 & \frac{p}{2}(1-d) & \frac{p}{2}d + \left(1-\frac{p}{2}\right)(1-d) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{p}{2} + \left(1-\frac{p}{2}\right)(1-d) \end{pmatrix} \begin{pmatrix} P_{10}(n-1) \\ P_{11}(n-1) \\ P_{12}(n-1) \\ \vdots \\ P_{1B}(n-1) \end{pmatrix} \quad (8)$$

Ako se sa $\mathbf{s}(n)$ označi vektor stanja posmatranog bafera, onda se prethodna jednačine transformiše u

$$\mathbf{s}(n) = \mathbf{P}\mathbf{s}(n-1) \quad (9)$$

Pretpostavlja se da su baferi u početnom trenutku prazni, odnosno:

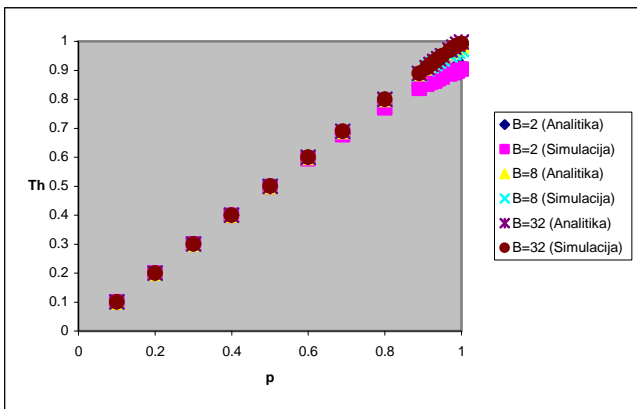
$$P_{10}(0) = P_{20}(0) = 1 \quad (10)$$

$$P_{11}(0) = P_{12}(0) = \dots = P_{1B}(0) = P_{21}(0) = P_{22}(0) = \dots = P_{2B}(0) = 0 \quad (11)$$

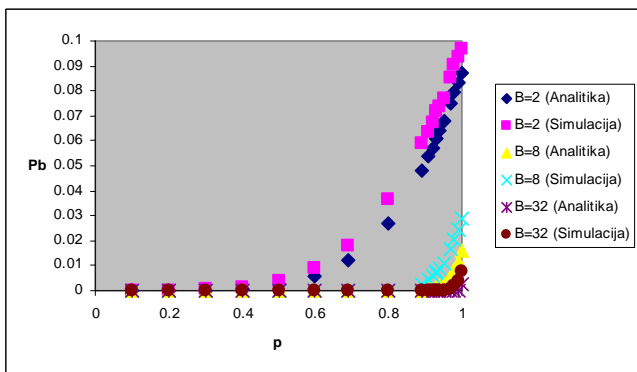
3. ANALIZA REZULTATA

Simulacioni rezultati su dobijeni korišćenjem programa napisanog u programskom jeziku C++. Simulacija je vršena na deset miliona vremenskih slotova.

Na slici 1 je prikazana propusnost u funkciji ulaznog opterećenja kada je $B=2$, $B=8$, odnosno $B=32$. Vjerovatnoća gubitka u funkciji ulaznog opterećenja kada je $B=2$, $B=8$, odnosno $B=32$ je prikazana na slici 2. Kao što je očigledno sa grafika, rezultati dobijeni predstavljenim modelom su veoma bliski rezultatima simulacije. Pri niskim nivoima dolaznog saobraćaja propusnost je jednaka ulaznom opterećenju tako da je vjerovatnoća gubitka ćelija praktično zanemarljiva. Povećanjem ulaznog saobraćaja i vjerovatnoća gubitka raste.

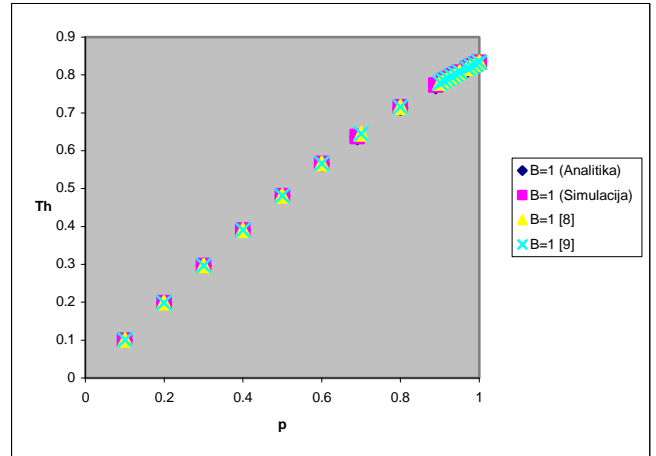


Slika 1: Propusnost komutatora u funkciji ulaznog opterećenja

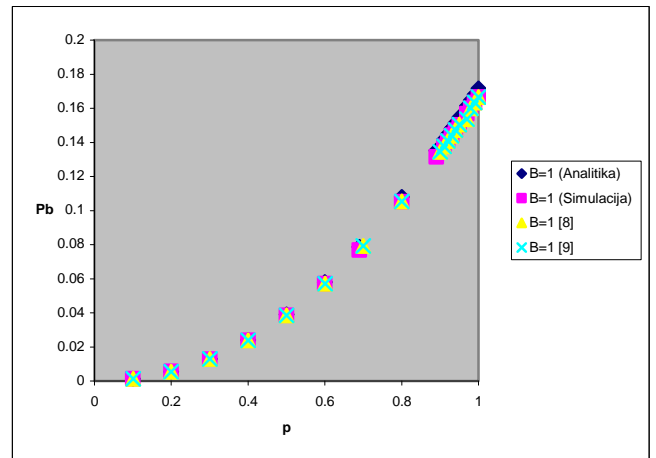


Slika 2: Vjerovatnoća gubitka u funkciji ulaznog opterećenja

Na slikama 3 i 4 upoređeni su rezultati za propusnost i vjerovatnoću gubitka u funkciji ulaznog opterećenja za slučaj kada su u ukršnim tačkama jedinični baferi. Jasno se uočava da su rezultati dobijeni novim analitičkim modelom i simulacijama veoma bliski rezultatima dobijenim u [8] i [9]. U cilju što boljeg poređenja, u Tabeli 1 su prikazane vrijednosti propusnosti za pojedine vrijednosti ulaznog opterećenja.



Slika 3: Propusnost komutatora u funkciji ulaznog opterećenja za slučaj kada je $B=1$



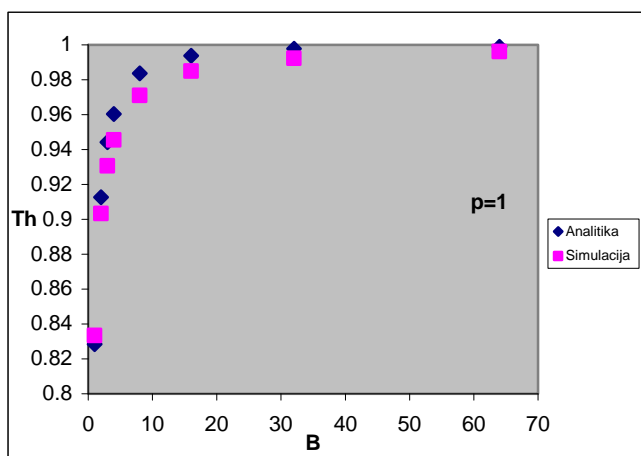
Slika 4: Vjerovatnoća gubitka u funkciji ulaznog opterećenja za slučaj kada je $B=1$

Tabela 1.

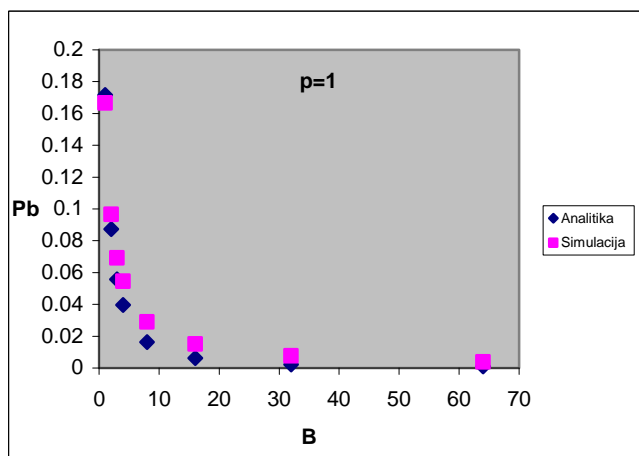
p	0.1	0.5	0.8	0.9	0.95	1
Analitika	0,099	0,48	0,71	0,77	0,803	0,828
Simulacija	0,099	0,48	0,71	0,77	0,807	0,833
[8]	0,1	0,49	0,72	0,78	0,807	0,833
[9]	0,1	0,49	0,72	0,78	0,807	0,833

Na slikama 5 i 6 su prikazana poređenja rezultata dobijenih korišćenjem analitičkog modela i simulacija za propusnost, odnosno vjerovatnoću gubitka, pri opterećenju $p=1$, za različite veličine bafera. Očigledno je da su dobijeni rezultati veoma bliski rezultatima simulacije. Ako se uzme u

obzir jednostavnost ovog modela onda je razlika koja se pojavljuje zanemarljiva.



Slika 5: Propusnost komutatora za $p=1$ u funkciji veličine ukrasnog bafera



Slika 6: Vjerovatnoća gubitka paketa za $p=1$ u funkciji veličine ukrasnog bafera

4. ZAKLJUČAK

U ovom radu predstavljen je novi iterativni model za analizu performansi 2×2 crossbar komutatora sa baferima u ukrasnim tačkama koji mogu biti veći od jediničnih. Razmatran je Bernoulli i.i.d. dolazni saobraćaj i komutator sa slučajnim izborom bafera u koliziji. Predstavljeni model je zasnovan na balansnim jednačinama koje opisuju stanje bafera koje ne zavisi od stanja bafera sa iste ulazne, odnosno izlazne magistrale. Obzirom na analizirani saobraćaj, pretpostavljeno je da baferi sa iste magistrale imaju identične raspodjele stanja. Pokazano je da su dobijeni rezultati veoma

bliski rezultatima simulacije. Glavne prednosti ovog pristupa u odnosu na analitičke rezultate prisutne u literaturi su jednostavnost i ekstenzibilnost na bafere veće od jednog paketa. Polazeći od ovog modela za komutator 2×2 , moguće je napraviti opšti oblik za veće komutatora, što će i biti tema naših sledećih radova.

LITERATURA

- [1] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch (extended version)," *IEEE Transactions on Communications*, vol. 47, Avgust 1999.
- [2] M. Nabeshima, "Performance evaluation of a combined input-and crosspoint-queued switch," *IEICE Transactions on Communications*, vol. E83-B, Mart 2000.
- [3] R. Rojas-Cessa, E. Oki, H. Chao, "On the combined input-crosspoint buffered switch with round-robin arbitration", *IEEE Transactions on Communications*, vol. 53, No. 11, Novembar 2005.
- [4] D. Banovic, I. Radusinovic, "Scheduling algorithm for VOQ switches". *Int. Journal of Electronics and Communications* 2008;62:455-8.
- [5] A. Mekkittikul, "Scheduling non-uniform traffic in high speed packet switches and routers", PhD. Thesis, Stanford University, 1998.
- [6] E. Shin, V. Mooney, G. Relay, "Round-robin arbiter design and generation", *Proceedings of the 15th international symposium on System Synthesis, Kyoto*, 2002. p. 243-8.
- [7] Y. Li, S. Panwar, H. Chao, "The dual round-robin matching switch with Exhaustive Service", *Proceedings of IEEE HPSR, Kobe*, 2002. p. 58-63.
- [8] Y. Kanizo, D. Hay, I. Keslassy, "The Crosspoint-Queued Switch" Technical report TR08-04, Comnet, Technion, Israel
- [9] J.Čvorović, I. Radusinović, M. Radonjić, "Buffering in Crosspoint-Queued Switch", *Telfor* 2009, Beograd, Novembar 2009.
- [10] F. Gebali, "Analysis of Computer and Communication networks", Springer Science+Business Media, LLC, New York, 2008.

MOGUĆNOST PRIMJENE METODA DISKRETIZACIJE KONTINUALNIH SISTEMA NA ELEKTROMEHAČKOM SISTEMU ECP 205

POSSIBILITY OF APPLYING METHODS FOR CONTINUOUS-TIME SYSTEMS DISCRETIZATION ON ELECTROMECHANICAL SYSTEM ECP 205

Mimo Mirković, Miloš Daković, Milovan Radulović *Elektrotehnički fakultet, Podgorica*

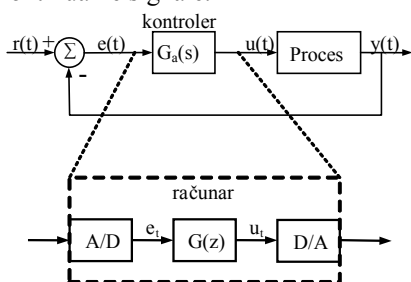
Sadržaj – U radu su prikazane metode diskretizacije kontinualnih sistema sa primjenom na standardni PID algoritam. Dobijeni diskretni PID algoritmi su implementirani na elektromehaničkom sistemu ECP 205. Dobijeni laboratorijski rezultati pokazuju da se odzivi sistema neznatno razlikuju pri korišćenju PID algoritama koji su dobijeni različitim metodama diskretizacije, pri čemu neki od njih uzrokuju nestabilnost sistema u zatvorenoj sprezi.

Abstract – In this paper we present methods for continuous-time linear time-invariant systems discretization applied to the PID controller. Discrete-time systems are implemented on ECP 205 educational control system. Obtained results shows that system response approximately the same for applied discretization method, while some of them causes unstable feedback system.

1. UVOD

Diskretni sistemi imaju niz prednosti koje ih stavljaju ispred analognih i omogućavaju im da sve više dobijaju na značaju. Diskretni sistem se veoma lako realizuje na računaru, karakteristike su mu veoma fleksibilne i adaptivne, sistem se mijenja jednostavno sa mijenjanjem nekog od parametara u programu. Ukoliko se diskretni sistem realizuje softverski onda jedan računar može da obavlja računanja za više sistema istovremeno. Primjenom računara eliminisana je pojava šuma u sistemu, a osim toga moguće je izvršiti i neke transformacije koje se ne mogu izvršiti analognim elementima.

Na slici 1 je prikazana struktura računarski upravljano sistema koji ima osobinu da se zbog prirode računara koji se nalazi u konturi upravljanja pojavljuju signali različitih priroda. Naime, signali sa kojima operiše računar su po svojoj prirodi diskretni, dok kontinualni proces na svom ulazu i izlazu ima kontinualne signale.



Sl. 1. Računarska realizacija kontinualnog sistema

Realizacija u diskretnom domenu podrazumijeva A/D konverziju, a zatim obradu dobijenog digitalnog signala prema algoritmu koji će odgovarati proračunatoj funkciji prenosa u analognom domenu, i na kraju D/A konverziju.

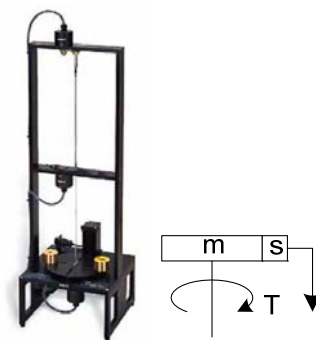
Postavljeni zadatak diskretizacije može se formulisati na sledeći način. Pretpostavlja se da je poznat model kontinualnog kontrolera za sistem u povratnoj sprezi (jed. 1). Traži se algoritam digitalnog kontrolera (jed. 2) tako da se digitalni kontroler koji zamjenjuje kontinualni i koji je

realizovan u računaru ponaša u što je moguće većoj mjeri sličnije kontinualnom sistemu.

$$G_a(s) = \frac{a_0 + a_1s + \dots + a_Ns^N}{b_0 + b_1s + \dots + b_Ms^M} \quad (1)$$

$$G(z) = \frac{A_0 + A_1z + \dots + A_nz^{-n}}{B_0 + B_1z + \dots + B_mz^{-m}} \quad (2)$$

Elektromehanički sistem nad kojim su vršeni eksperimenti je Model ECP 205. Korišćen je sistem sa jednim diskom koji ima mogućnost rotacije oko svoje ose uz minimalno trenje. Izgled i šema sistema su prikazani na Sl. 2.



Sl. 2. Elektromehanički sistem sa jednim diskom (lijevo) i njegova uprošćena šema (desno)

Osovina je pogonjena brushless DC servo motorom sa kojim je povezana preko krutog kaiša i remenice redukcionog odnosa 3:1. Na donjem kraju osovine direktno je vezan enkoder koji registruje ugaoni pomjeraj prvog diska.

PID regulatori su najčešće primjenjivani regulatori za upravljanje u industrijskim procesima. Ranije su se izvodili u analognoj tehnici, ali u današnje vrijeme uglavnom se primjenjuju digitalni PID regulatori, bilo kao samostalni mikroprocesorski uređaji ili kao standardni programski moduli u programabilnim logički kontrolerima ili u procesnim računarima. Diskretni PID regulator se dobija diskretizacijom analognog PID regulatora. Naravno, cilj

diskretizacije je da dobijeni diskretizovani regulator dobro oponaša kontinualni PID.

2. METODI DISKRETIZACIJE ANALOGNIH SISTEMA U DISKRETNE

Razvijen je čitav niz metoda za direktno projektovanje digitalnih kontrolera. Jedan broj ovih metoda razvijen je po analogiji sa metodama za sintezu kontinualnih kontrolera, dok su druge metode razvijene posebno za diskretne sisteme korišćenjem osobenosti diskretnih funkcija prenosa. Međutim, uprkos analogiji koja postoji u sintezi diskretnih i kontinualnih sistema činjenica je da je, posebno kod grafo-analitičkih metoda sinteza kontinualnih kontrolera daleko jednostavnija. Pored toga, tokom više od pola vijeka razvoja teorije i prakse projektovanja kontinualnih sistema upravljanja sakupljeno je mnogo iskustva. Postupci izbora parametara su u velikoj mjeri algoritimizirani i sagledane su sve moguće posledice promjene njihovih vrijednosti. Iskustvo pokazuje da se ovo praktično znanje ne može direktno primijeniti kod projektovanja digitalnih kontrolera.

Svi navedeni razlozi doveli su do toga da se proces projektovanja digitalnih kontrolera često odvija u dva koraka. U prvom koraku se za kontinualni proces razvija kontinualni kontroler. Zatim se vrši diskretizacija kontinualnog kontrolera sa zadatkom da se dobije što bolje slaganje rezultujućeg digitalnog sistema sa kontinualnim. Pri tome se, pored ostvarene tačnosti u postupku diskretizacije vodi računa i o računarskim resursima koji će biti potrebni za realizaciju digitalnog algoritma, kao i o vremenu potrebnom za izračunavanje jednog algoritamskog koraka.

Postoji više metoda pomoću kojih je moguće izvršiti diskretizaciju kontinualnih sistema, a to su:

- ⇒ Metod impulsne invarijantnosti,
- ⇒ Metod step invarijantnosti,
- ⇒ Metod diferenciranja unaprijed (udesno),
- ⇒ Metod diferenciranja unazad (ulijevo ili Backward Difference method)
- ⇒ Bilinearna ili Tustinova transformacija

U narednom dijelu su ukratko prikazani metodi diskretizacije kontinualnih sistema. Svaki od ovih metoda pokazuje izvjesne prednosti i nedostatke, a to je i razlog zašto nije razvijen jedinstveni metod već se izbor vrši u svakom pojedinačnom slučaju i u skladu sa zadatim kriterijumima.

2.1. METOD IMPULSNE INVARIJANTNOSTI

Transformacija analognog u diskretni sistem se, u ovom slučaju zasniva na tome da impulsni odziv diskretnog sistema $h(n)$ bude jednak odbircima impulsnog odziva analognog sistema $h_a(t)$ koji su pomnoženi korakom odabiranja, jednačina

$$h(n) = Th_a(t)|_{t=nT} \quad (3)$$

U [1] je pokazano da se Z-transformacija impulsnog odziva diskretnog sistema može napisati kao:

$$G(z) = \sum_{i=1}^N \frac{Tk_i}{1 - e^{s_i T} z^{-1}} \quad (4)$$

gdje su s_i polovi analogne funkcije prenosa.

U opštem slučaju, ako postoje polovi većeg stepena ($m > 1$), potrebno ih je transformisati u oblik:

$$\frac{k}{(s - s_i)^m} \rightarrow \frac{1}{(m-1)!} \left[\frac{d^{m-1}}{da^{m-1}} \frac{Tk}{1 - e^{aT} z^{-1}} \right]_{a=s_i} \quad (5)$$

2.2. METOD STEP INVARIJANTNOSTI

Metod step invarijantnosti ima za cilj da prenosnu funkciju regulatora $G(s)$ prevede u z domen, a da im odzivi na odskočnu (step) funkciju budu isti (što bliži). Pretpostavi se da se na ulaz $G(s)$ dovede jedinični odskočni siglal $x(t)$, čija je Laplasova transformacija $X(s) = 1/s$. Prenosna funkcija regulatora u z domenu je:

$$G(z) = \frac{z-1}{z} Z \left\{ L^{-1} \left[G(s) \left(\frac{1}{s} \right) \right] \right\} \quad (6)$$

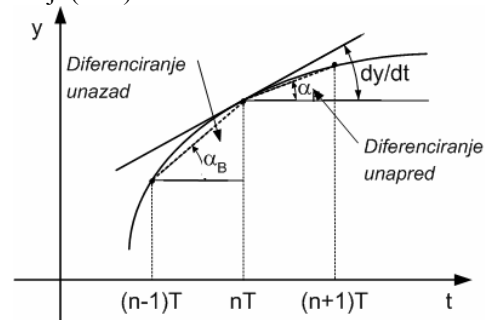
2.3. METOD DIFERENCIRANJA UNAPRIJED

Naredni metodi za diskretizaciju kontinualnih sistema pripadaju grupi numeričkih metoda. Osnovna ideja leži u činjenici da operator s Laplasove transformacije predstavlja operaciju diferenciranja, a s^{-1} operaciju integraljenja, tj.

$$L \left\{ \frac{dy(t)}{dt} \right\} = sY(s); \quad L \left\{ \int_0^t y(\tau) d\tau \right\} = \frac{1}{s} Y(s) \quad (7)$$

Problem diskretizacije se prevodi u problem numeričke aproksimacije prvog izvoda funkcije.

Metod diferenciranja unazad se zasniva na sledećoj aproksimaciji (sl. 3)



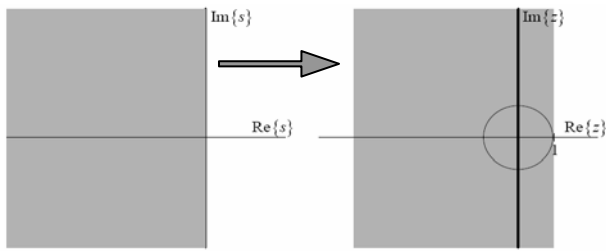
Sl. 3. Aproksimacija metodom diferenciranja unazad i unaprijed

$$\left. \frac{dy}{dt} \right|_{t=nT} \approx \frac{y(nT+T) - y(nT)}{T} \quad (8)$$

$$\begin{aligned} L & \qquad \qquad \qquad Z \\ \downarrow & \qquad \qquad \qquad \downarrow \\ sY(s) & \qquad \frac{z-1}{T} Y(z) \\ \Rightarrow s = \frac{z-1}{T}; & \quad z = 1 + Ts \end{aligned} \quad (9)$$

U cilju razmatranja primjenljivosti ovog metoda treba analizirati stabilnost sistema. Potrebno je uočiti u šta se preslikava granica stabilnosti. Na osnovu [5], slijedi da se lijeva poluravan iz „ s “ ravni preslikava u poluravan u „ z “ ravni koja se nalazi lijevo od prave čiji je realni dio jednak 1, slika 4. Proizilazi da sistem koji je kao kontinualan bio

stabilan, može lako postati nestabilan nakon diskretizacije pomoću ovog metoda.



Sl. 4. Preslikavanje oblasti stabilnosti iz „s“ u „z“ ravan

2.4. METOD DIFERENCIRANJA UNAZAD

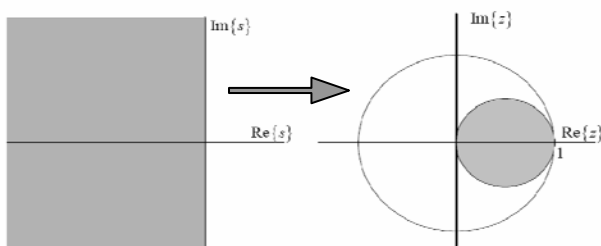
Koristeći ideju iz prethodno obrađenog metoda, moguće je dobiti još jedan metod na osnovu sledeće aproksimacije.

$$\left. \frac{dy}{dt} \right|_{t=nT} \approx \frac{y(nT) - y(nT - T)}{T} \quad (10)$$

$$\begin{array}{ccc} L & & Z \\ \downarrow & & \downarrow \\ sY(s) & \frac{1-z^{-1}}{T} Y(z) & \\ \Rightarrow s = \frac{1-z^{-1}}{T}; & z = \frac{1}{1-Ts} & \end{array} \quad (11)$$

Ovaj metod se često i u domaćoj literaturi naziva Backward Difference method.

Stabilnosti sistema prilikom ove vrste diskretizacije, cijela lijeva poluravan „s“ ravni preslikava u krug poluprečnika 0.5 čiji je centar u tački (0.5, j0), slika 5. Izvodi se zaključak da će sistem koji je bio stabilan u kontinualnom domenu, ostati stabilan i nakon diskretizacije, ali i da neki sistemi koji nijesu bili stabilni prije diskretizacije, mogu postati stabilni kao diskretni ekvivalenti, [5].



Sl. 5. Preslikavanje oblasti stabilnosti iz „s“ u „z“ ravan

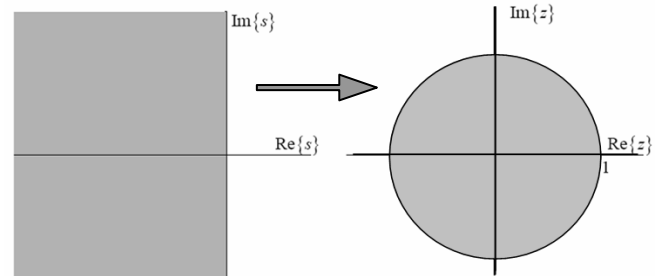
Osnovna prednost obje ove metode sastoji se u jednostavnosti njihove primjene i jednostavnoj strukturi rezultujuće diskretne funkcije prenosa. Pri tome se s obzirom na moguće probleme oko stabilnosti diferenciranje unaprijed rjeđe koristi, [5].

2.5. BILINEARNA (TUSTINOVA) TRANSFORMACIJA

Sledeći od numeričkih metoda kojim se kontinualni sistemi mogu diskretizovati jeste i bilinearna ili Tustinova transformacija:

$$s = \frac{2}{T} \frac{z-1}{z+1} \quad (12)$$

U [1] je pokazano da se lijeva „s“ poluravan preslikava u unutrašnjost jediničnog kruga „z“ ravni. Zaključak koji se može izvesti je da se stabilan kontinualan sistem putem ove transformacije preslikava u stabilan diskretan sistem. Isto tako, sistem koji je prije diskretizacije bio nestabilan, i nakon diskretizacije ostaće nestabilan, slika 6.



Sl. 6. Preslikavanje oblasti stabilnosti iz „s“ u „z“ ravan

3. EKSPERIMENTALNI REZULTATI

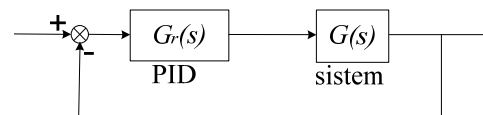
Za laboratorijsku analizu je korišćen jednostavan sistem sa jednim diskom koji na sebi ima dva teža mase po 500g. Cilj je pozicioniranje diska na zadatu vrijednost ugla.

Za upravljanje sistemom je korišćen klasični PID regulator koji se nalazi u direktnoj grani (Sl. 7), čija je funkcija prenosa:

$$G_r(s) = k_p + k_d s + \frac{k_i}{s} \quad (13)$$

Koeficijenti PID algoritma su dobijeni eksperimentalno i usvojene su sledeće vrijednosti:

$$k_p = 0.27; \quad k_d = 0.022; \quad k_i = 1.72$$



Sl. 7. Šema sistema upravljanog PID-om u jediničnoj negativnoj povratnoj sprezi

Pri upravljanju pomenutim sistemom, generalno se radi sa diskretnim vremenom u realnom vremenu. Projektovani kontinualni regulator je potrebno diskretizovati. Potrebno je odrediti period odabiranja, kao i nekom od pomenutih metoda dobiti diskretnu funkciju prenosa PID regulatora.

Za ovaj sistem čija je karakteristična frekvencija manja od 10 Hz uzeto je vrijeme odabiranja $T_s = 0.00442s$ što odgovara frekvenciji od oko 260 Hz.

Za diskretizaciju prenosne funkcije PID kontrolera korišćene su dvije metode: metoda diferenciranja unazad i Tustinova transformacija. U literaturi postoji veliki broj načina diskretizacije PID regulatora, [5].

Prvi metod kojim je dobijena diskretna prenosna funkcija regulatora, $G_1(z)$, je diferenciranje unazad:

$$G_1(z) \approx \frac{5.25498 - 10.224z^{-1} + 4.9774z^{-2}}{1 - z^{-1}} \quad (14)$$

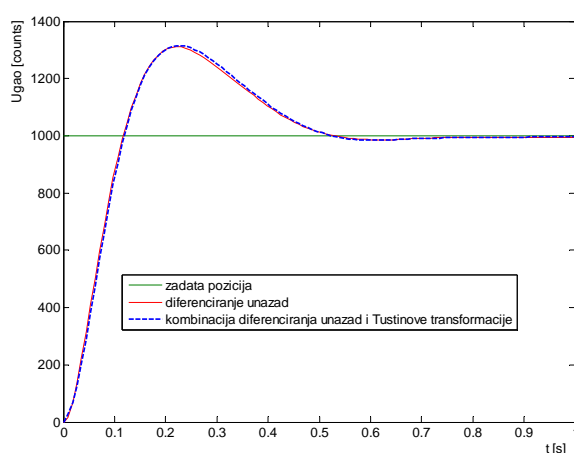
Metod diferenciranja unaprijed ne može da se primijeni na laboratorijskoj opremi zbog zatvorenosti forme preko koje

se unose koeficijenti prenosne funkcije. Međutim, treba napomenuti da se ovaj metod rijetko i koristi za diskretizaciju PID regulatora, jer tako dobijen regulator obično dovodi do nestabilnosti sistema u zatvorenoj sprezi.

Pri pokušaju implementacije diskretne prenosne funkcije regulatora dobijene pomoću Tustinove transformacije sistem je postao nestabilan. Nakon toga je izvršena diskretizacija diferencijalnog dejstva pomoću metode diferenciranja unazad, a integralnog dejstva pomoću Tustinove transformacije, [5]. Ovim načinom kombinacije metoda dobijena je diskretna funkcija prenosa koja je data jed. (15)

$$G_2(z) \approx \frac{5.25118 - 10.22095z^{-1} + 4.9774z^{-2}}{1 - z^{-1}} \quad (15)$$

Na slici 8 su prikazana dva odziva sistema u zatvorenoj sprezi na odskočnu funkciju. Primjećuje se da su odzivi sistema veoma slični.



Sl. 8. Odstivi sistema u zatvorenoj sprezi sa PID regulatorima $G_1(z)$ i $G_2(z)$

U tabeli 1. su prikazane karakteristike u vremenskom domenu za dva odziva sa Sl. 8. Zadata pozicija diska je 1000 counts (1000 counts $\approx 22^\circ$).

Tabela 1. Karakteristike sistema u vremenskom domenu

	Dif. unazad	kombinacija
Stac.stanje	995	997
Preskok [%]	31.66	31.79
vr. uspona [s]	0.081	0.082
vr. smirenja [s]	0.45	0.445
vr. kašnjenja [s]	0.062	0.064

Pored pomenutih metoda, izvršena je i diskretizacija diferencijalnog dejstva pomoću Tustinove transformacije, a integralnog dejstva pomoću metode diferenciranja unazad, međutim i ovako dobijen regulator čini sistem u zatvorenoj sprezi nestabilnim.

4. ZAKLJUČAK

Na osnovu prikazanih eksperimenata zaključuje se da sva tri metoda različito utiču na sistem u zatvorenoj sprezi i na njegovu stabilnost.

Prva dva opisana metoda diskretizacije nijesu primjenljiva kod idealnog PID regulatora. Metod diferenciranja unaprijed utiče na stabilnost sistema u zatvorenoj sprezi, tako da njegov odziv nije bilo moguće snimiti. Metod diferenciranja unazad daje zadovoljavajuće rezultate. Realno je za očekivati da metod Tustinove transformacije očuva stabilnost sistema, međutim, to se nije dogodilo pri odskočnom odzivu ovog sistema. Na kraju se pokušalo sa kombinacijama pomenutih metoda diskretizacije, posebno za diferencijalno i za integralno dejstvo regulatora. Dobijeni su zadovoljavajući rezultati i u slučaju kada je diferencijalno dejstvo diskretizovano pomoću metode diferenciranja unazad, a integralnog dejstva pomoću Tustinove transformacije.

Laboratorijska edukaciona oprema koja se nalazi u Laboratoriji za Automatiku na Elektrotehničkom fakultetu može poslužiti studentima za veliki broj eksperimenata kojima bi provjerili i utvrdili stečeno teorijsko znanje.

Autor će nastaviti istraživanja u vezi diskretizacije i drugih, naprednijih algoritama upravljanja (PID + Notch filter, LQR regulator) i njihovoj primjeni na realnim sistemima. Takođe, cilj je utvrditi zašto je sistem sa PID regulatorom koji je diskretizovan pomoću Tustinove aproksimacije nestabilan u zatvorenoj sprezi.

LITERATURA

- [1] Dr ing. Ljubiša Stanković, „Digitalna obrada signala“, Naučna knjiga, Beograd, 1990.
- [2] Alan V. Oppenheim, Alan S. Willsky with Ian T. Young, „Signals and Systems“, Prentice-Hall signal processing series, Alan V. Oppenheim Series editor, New Jersey, 1983.
- [3] Thomas R. Parks, „Torsional Control System (Instructor's Edition) – Manual for Model 205/205a“, ECP, Bell Canyon, CA, www.ecpsystems.com.
- [4] Benjamin C. Kuo, Farid Golnaraghi, „Automatic Control Systems“, 8th Edition(2003), John Wiley & Sons, INC.
- [5] S. Turajlić, „Upravljanje procesima pomoću računara“, materijali za predavanja, <http://automatika.etf.bg.ac.yu>, Beograd, 2008.
- [6] M. Mirković, „Neželjeni efekti pri realnom izvršavanju kontrole nad elektromehaničkim sistemom“, XII konferencija IT, Žabljak, 2007.
- [7] Đurović Željko M.; Kovačević Branko D., „Sistemi automatskog upravljanja“, Akademska misao, 2006.

PRIMJER REALIZACIJE TCP/IP/ETHERNET KOMUNIKACIJE PC RAČUNARA I SPARTAN 3E FPGA PLOČICE

IMPLEMENTATION EXAMPLE OF THE TCP/IP/ETHERNET COMMUNICATION BETWEEN PC AND SPARTAN 3E FPGA BOARD

Nikola Ljumović, Milutin Radonjić, Igor Radusinović, *Elektrotehnički fakultet Podgorica*

Sadržaj – U ovom radu je prikazan jedan način realizacije TCP/IP/Ethernet komunikacije između personalnog računara i Spartan 3E FPGA razvojne pločice. Korišćenjem ovog vida komunikacije ostvarena je interakcija sa periferijama koje se nalaze na razvojnoj pločici. Predstavljene su najvažnije karakteristike korišćenog procesora (MicroBlaze) i objašnjen njegov hardverski dizajn. Na kraju, prikazani su i objašnjeni najvažniji dijelovi kreirane aplikacije koja upravlja radom procesora.

Abstract – In this paper, one realization of TCP/IP/Ethernet communication between PC and Spartan 3E FPGA development board is presented. Using this form of communication, interaction with peripherals on the board is achieved. Most important features of the used processor (MicroBlaze) and its hardware design are explained. Finally, most important parts of the created application that controls the operation of processor are presented and explained.

1. UVOD

FPGA (*Field-Programmable Gate Array*) je integrisano kolo koje pruža korisnicima mogućnost brzog konfigurisanja i programiranja i kao takvo omogućava veliku fleksibilnost i efikasnost u radu. Razvoj i proizvodnja FPGA uređaja doživjeli su ekspanziju tokom devedesetih godina prošlog vijeka. Danas ovakav model razvojne platforme sve više predstavlja nezaobilazan element u dizajniranju i projektovanju digitalnih sistema koji treba da obavljaju veliki broj različitih funkcija. Prednosti ovakve razvojne platforme dovele su do značajnog skraćenja perioda razvoja hardverskih proizvoda.

Umrežavanje bazirano na Ethernet protokolu je tehnologija koju karakteriše brz razvoj i veoma velika zastupljenost u današnjim sistemima. Popularnost Ethernet mreža je posledica njihovog velikog potencijala, niske cijene i lakoće implementacije. FPGA razvojno okruženje sa implementiranim Ethernet interfejsom, u tom smislu predstavlja vrlo moćnu razvojnu platformu.

Jedna od takvih razvojnih platformi je i Spartan 3E FPGA pločica [1] proizvedena od strane kompanije Xilinx. Ova platforma posjeduje različite interfejse, aktivne i pasivne komponente, što je čini odličnim izborom pri dizajniranju uređaja umjerene složenosti. Upravo zbog te svoje osobine i relativno niske cijene, Spartan 3E FPGA predstavlja interesantno rješenje, prihvatljivo velikom broju korisnika.

Periferije (LED diode, tasteri, prekidači i LCD displej) i interfejsi implementirani na Spartan 3E pločici služe za modeliranje raznih realnih sistema koji se pojavljuju prilikom projektovanja uređaja za konkretnu implementaciju u praksi. Stoga interakcija sa njima, vršena Ethernet komunikacijom, predstavlja dobru osnovu za razumijevanje koncepta FPGA programiranja i potencijala kojeg posjeduje implementacija Ethernet-a na FPGA razvojnim platformama.

Ovaj rad predstavlja rezultat projekta realizovanog u cilju ostvarivanja interakcije sa pomenutim periferijama podstredstvom Ethernet komunikacije [2]. Interakcija podrazumijeva kontrolu periferija zadavanjem komandi pomoću personalnog računara, kao i stalan nadzor njihovog stanja. Projekat je realizovan korišćenjem procesora (mikrokontrolera) pod nazivom MicroBlaze. Ovakav pristup je odabran jer se upotrebom mikrokontrolera značajno pojednostavljuje i čini efikasnijim proces FPGA dizajniranja, zbog čega ona danas predstavlja važan trend u ovoj oblasti.

Ovaj projekat podrazumijeva prvenstveno definisanje *MicroBlaze* procesora a zatim kreiranje aplikacije koja će upravljati radom tako definisanog procesora. Aplikaciju je moguće pisati u višim programskim jezicima, kao što su C i C++. U ovom slučaju ona je napisana u programskom jeziku C tako da omogućava uspostavljanje TCP konekcije i zadavanje komandi u formi TCP segmenata korišćenjem TCP/IP klijentskog programa na računaru.

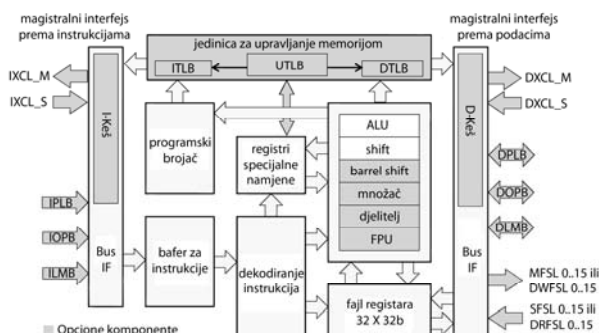
U drugom poglavlju rada opisana je arhitektura i osobine *MicroBlaze* procesora. Način definisanja njegovog hardvera dat je u trećem poglavlju. U četvrtom poglavlju se nalazi opis kreirane aplikacije pomoću koje se vrši interakcija sa periferijama na pločici. Na kraju su, u petom poglavlju, date zaključne konstatacije i zapažanja.

2. MICROBLAZE PROCESOR

Mikroprocesori dostupni za korišćenje u Xilinx-ovim FPGA pločicama se generalno mogu podijeliti na *soft core* i *hard core* procesore. MicroBlaze [3] je *soft core* procesor kojeg je moguće u potpunosti softverski implementirati koristeći programabilnu logiku prisutnu na pločici, za razliku od *hard core* procesora koji su fabrički ugrađeni. Ovakav virtuelni procesor se generiše kombinacijom blokova HDL koda zvanih *jezgra*. Velika prednost ovog pristupa se ogleda u fleksibilnosti procesora, mogućnosti vršenja različitih podešavanja i prilagođavanja potrebama projekta koji se želi realizovati. Izborom željenih jezgara moguće je odabrati i

implementirati potrebne interfejske i periferije kao što su fleš memorija, Ethernet interfejs, ulazne/izlazne periferije itd.

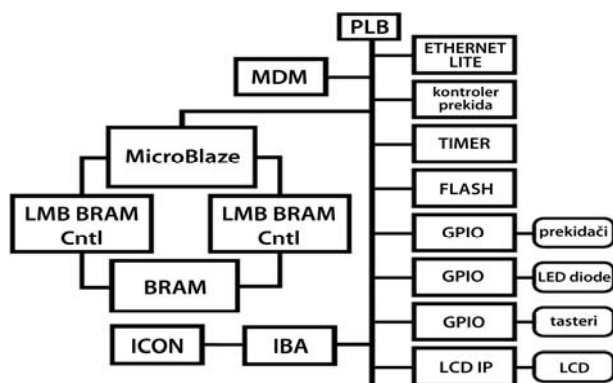
Na slici 1. je prikazana arhitektura MicroBlaze procesora. Komponente u bijeloj boji čine osnovu arhitekture dok su sive komponente opcione tj. koriste se u zavisnosti od potreba aplikacije. Kako je MicroBlaze *soft core* procesor, opcione komponente koje se ne koriste se ne implementiraju i time nepotrebno ne zauzimaju resurse na čipu. Kao što se može vidjeti sa slike, MicroBlaze se odlikuje 32-bitnom arhitekturom sa odvojenim 32-bitnim magistralama za instrukcije i podatke. Osnovu ove arhitekture čine 32 registra opšte namjene, aritmetičko logička jedinica (ALU), pomjeračka jedinica i dva nivoa prekida. Ovakav bazični dizajn je dalje moguće nadograditi izborom i konfigurisanjem dodatnih, opcionih komponenti. Ova fleksibilnost omogućava korisniku da vrši balans između zahtijevanih performansi, kompleksnosti i cijene dizajna.



Slika 1. Arhitektura MicroBlaze procesora

3. DEFINISANJE HARDVERA PROCESORA

Definisanje i konfiguracija MicroBlaze procesora se vrše u razvojnom okruženju *Xilinx Platform Studio*. U ovom razvojnom okruženju moguće je jednostavno odabrati i konfigurirati potrebne komponente procesora i na taj način njegov hardver prilagoditi aplikaciji koja se želi kreirati. Hardverski dizajn procesora se jednostavno implementira na pločici korišćenjem programabilne logike FPGA čipa.



Slika 2. Blok šema hardverskog dizajna

Na slici 2. se mogu vidjeti osnovne komponente i periferije hardverskog dizajna kreiranog za potrebe ovog projekta. Osnovu dizajna čini MicroBlaze procesor sa radnom frekvencijom od 50MHz i 16 kB lokalne (BRAM) memorije. Dodate periferije (u zagradama su dati njihovi nazivi u *Xilinx Platform Studio*-u) su: 64 MB DDR SDRAM memorije (*DDR_SDRAM*), dva LMB BRAM kontrolera (*lmb_bram_if_cntlr*), 16 MB fleš memorije (*xps_mch_emc*),

Ethernet Lite MAC (*xps_ethernetlite*), tajmer (*xps_timer*), I/O interfejs za led diode (*xps_gpio*), 4 tastera (*xps_gpio*) i 4 prekidača (*xps_gpio*). Ethernet MAC i tajmer koriste prekide.

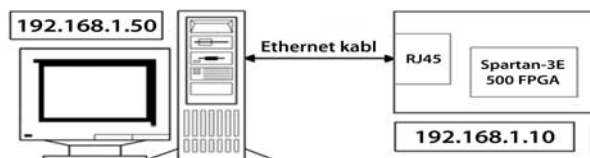
Ovaj projekat podrazumijeva i interakciju korisnika sa LCD displejem na Spartan 3E pločici. Međutim, zbog njegove specifičnosti, prilikom generisanja novog projekta u *Xilinx Platform Studio*-u nije ga moguće jednostavno odabrati (kao što je to bio slučaj sa ostalim, gore navedenim, periferijama). Detaljno uputstvo za dodavanje LCD displeja u vidu korisnički definisane periferije dato je u [4].

4. SOFTVERSKI DIZAJN

Nakon definisanja hardvera procesora u *Xilinx Platform Studio*-u, tako definisan hardver se, u obliku *bit* fajla (format koji služi za programiranje FPGA čipa), unosi u razvojno okruženje *Software Development Kit*. U ovom razvojnom okruženju je, u cilju realizacije ovog projekta, napisana aplikacija u programskom jeziku C kojom se programira rad prethodno definisanog procesora. Nakon pisanja koda aplikacije u ovom razvojnom okruženju vrši se programiranje FPGA čipa a zatim i pokretanje aplikacije.

4.1. Struktura i osobine aplikacije

Napisana aplikacija ima za cilj da demonstrira interakciju sa periferijama na Spartan 3E FPGA pločici i to sa tasterima, kliznim prekidačima, LED diodama i LCD displejem. Komunikacija između PC računara i pločice se obavlja korišćenjem TCP protokola preko Ethernet interfejsa (RJ45 konektor) na pločici i mrežnoj kartici računara. Ethernet komunikacija se obavlja korišćenjem Ethernet Lite MAC kontrolera [5], odabranog prilikom definisanja hardvera procesora. U kodu aplikacije potrebno je dodijeliti pločici željenu IP i MAC adresu. U ovom projektu odabrana je, i u kodu aplikacije pločici dodijeljena, IP adresa: 192.168.1.10 sa *subnet* maskom: 255.255.255.0 i MAC adresa: 00-0a-35-00-01-02. Nakon startovanja aplikacije pločica osluškuje zahtjeve za uspostavljanjem TCP konekcije na portu 1024, takođe definisanom u samom kodu aplikacije. Nakon stizanja zahtjeva od strane računara, TCP komunikacija se uspostavlja i tada je moguće vršiti prenos podataka. Pri tome, da bi se TCP konekcija mogla uspostaviti potrebno je da pločica i računar budu u istoj mreži, dok je zahtjeve za konekcijom potrebno slati na gore navedeni broj porta. Konfiguracija razvojne pločice i PC računara prikazana je na slici 3.



Slika 3. Konfiguracija razvojne pločice i PC računara

Aplikacija je napisana tako da omogućava interakciju sa pomenutim periferijama i to kroz:

- mogućnost uključivanja/isključivanja bilo koje kombinacije od 8 dostupnih LED dioda;
- ispisivanje teksta na prvoj ili drugoj liniji LCD displeja;
- rotiranje sadržaja displeja (u slučaju da je potrebno prikazati više od 16 karaktera po liniji koji inicijalno mogu biti prikazani);

- očitavanje statusa kliznih prekidača prilikom svake promjene pozicije nekog od njih;
- očitavanje pritiska bilo kog od tastera (osim "južnog" tastera [1] koji se koristi za resetovanje aplikacije).

Za aktiviranje funkcija uključivanja/isključivanja LED dioda, ispisivanja na prvoj ili drugoj liniji LCD displeja kao i rotiranja njegovog sadržaja, koriste se komande koje se zadaju na PC računaru. Komande se šalju u obliku TCP segmenata nakon uspostavljanja TCP konekcije. U ovom projektu je u tu svrhu korišćen TCP/IP klijentski program koji omogućava formiranje čitavog TCP segmenta prije njegovog slanja. Naime, pojedini programi automatski vrše slanje TCP segmenata nakon svakog unijetog karaktera tj. ne dozvoljavaju formiranje kompletne komande i stoga se, u ovom slučaju, ne mogu koristiti.

Za razliku od ovih, funkcije detektovanja pomjeranja položaja prekidača i pritiska tastera se automatski obavljaju. Naime, odmah nakon detektovanja promjene (pomjeranja prekidača ili pritiska tastera na pločici), formira se TCP segment sa odgovarajućom porukom kojeg pločica automatski šalje računaru. TCP/IP klijentski program na računaru prima tu poruku i ispisuje njen sadržaj.

4.2. Dizajn aplikacije u programskom jeziku C

Aplikacija je organizovana kroz više fajlova napisanih u programskom jeziku C:

- fajl *platform.c* definiše softversku platformu i kreira se automatski od strane razvojnog okruženja;
- fajl *ethernet.c* je zadužen za definisanje TCP/IP komunikacije i kreiran je na osnovu koda demo aplikacije pod nazivom *echo server* (uz neophodne modifikacije tog koda) ugrađene u razvojno okruženje;
- fajl *main.c* čini osnovu aplikacije. Kod ovog fajla je napisan u okviru ovog projekta, tako da u potpunosti definiše rad MicroBlaze procesora pozivajući, kada je potrebno, funkcije iz ostalih fajlova.

Softverska platforma se kreira u obliku posebnog fajla (*platform.c*) kojeg generiše razvojno okruženje i predstavlja osnovu za dalje programiranje. U njenim podešavanjima neophodno je dodati lwIP biblioteku. Ova biblioteka pruža podršku za realizaciju TCP/IP funkcionalnosti sa brojnim ugrađenim funkcijama u tom pravcu. U [6] je dat detaljan opis lwIP biblioteke dok se u [7] mogu naći primjeri aplikacija kreiranih korišćenjem ove biblioteke.

Nakon kreiranja softverske platforme i dodavanja lwIP biblioteke moguće je pristupiti kreiranju fajla (*ethernet.c*) u kome će se definisati TCP/IP komunikacija, korišćenjem funkcija lwIP biblioteke. Ovaj fajl se poziva prilikom uspostavljanja i raskida TCP konekcije, prijema i slanja podataka, provjere stanja bafera itd.

Nakon toga, kreira se glavni fajl (*main.c*) u kojem se programira rad procesora. U kodu ovog fajla je definisan veći broj funkcija neophodnih za funkcionisanje aplikacije. Tu spada niz funkcija za kontrolu LCD displeja koje se zasnivaju na upisivanju različitih vrijednosti (kombinacija bita) u njegovu memoriju (detaljno u opisu LCD displeja u [1]). Osim njih, ovdje su definisane i funkcije za kontrolu i

realizaciju zadate komande koje vrše provjeru validnosti komande a zatim i ostvarivanje njom definisane radnje. Glavni (*main*) dio koda ovog fajla se sastoji od inicijalizacije korišćenih periferija i glavne petlje programa. Najprije se vrši inicijalizacija prekidača, LED dioda, tastera i LCD displeja [2]. Nakon toga se inicijalizuje Ethernet komunikacija:

```
struct netif *netif, server_netif;
struct ip_addr ipaddr, netmask, gw;
unsigned char mac_ethernet_address[] = {0x00, 0x0a,
0x35, 0x00, 0x01, 0x02}; //MAC adresa ploce
netif = &server_netif;
//Inicijalizacija platforme
init_platform();
//Podesavanje prekida
platform_setup_interrupts();
//Inicijalizacija IP adresa koje ce biti koriscene
IP4_ADDR(&ipaddr, 192, 168, 1, 10);
IP4_ADDR(&netmask, 255, 255, 255, 0);
IP4_ADDR(&gw, 192, 168, 1, 1);
lwip_init();
//Dodavanje mreznog interfejsa na netif_list
if (!xemac_add(netif, &ipaddr, &netmask, &gw,
mac_ethernet_address, PLATFORM_ETH_MAC_BASEADDR)){
    xil_printf("Greska prilikom dodavanja N/W
interfejsa\nr");
    return -1;}
netif_set_default(netif);
//Omogucavanje prekida
platform_enable_interrupts();
//Specificiranje da je mreza podignuta i
oglasavanje ploce u mrezi
netif_set_up(netif);
start_application(); //Startovanje aplikacije
```

Ovdje je neophodno istaći važnost pravilne inicijalizacije Ethernet komunikacije. Naime, kao što se može vidjeti, za ostvarivanje ove komunikacije neophodan je rad sa prekidima. Već kod definisanja hardvera MicroBlaze procesora, prilikom izbora potrebnih periferija podešeno je da tajmer i Ethernet Lite MAC koriste prekide. Međutim, rad sa prekidima je potrebno pravilno definisati i u glavnom kodu aplikacije. Prilikom realizacije projekta javio se problem u radu sa prekidima koji je uzrokovao nemogućnost uspostavljanja TCP konekcije. Naime, u demo aplikaciji *echo server* ugrađenoj u razvojno okruženje *Software Development Kit v11.1*, koja je u jednom dijelu poslužila kao osnova za kreiranje aplikacije u okviru ovog projekta, prekidi nijesu na pravilan način definisani. Kao što se može vidjeti iz datog koda, za definisanje prekida su zadužene funkcije *platform_setup_interrupts()* i *platform_enable_interrupts()*, koje vrše podešavanje prekida i njihovo izvršavanje. Problem je prevaziđen pozivom funkcije *platform_setup_interrupts()* koja je izostavljena u pomenutoj demo aplikaciji. Dakle, izostavljanje poziva neke od ove dvije funkcije dovodi do nemogućnosti uspostavljanja komunikacije.

Kada je završena inicijalizacija svih periferija vrši se registrovanje početnih stanja tastera i prekidača nakon čega program ulazi u glavnu petlju. Prilikom svakog prolaska kroz ovu petlju program provjerava da li je stigao TCP segment poslat od strane računara. Ako je segment stigao i ako je prijem komande omogućen, program preuzima podatke iz pristiglog frejma i provjerava komandu. Prijem komande je omogućen automatski nakon pristizanja frejma i onemogućava se odmah nakon provjere i eventualnog izvršenja komande. Ovim se sprečava da program, prilikom svakog prolaska kroz petlju, ponovo uzima i obrađuje već obrađenu komandu.

Nakon toga, program provjerava da li je došlo do promjene stanja prekidača i tastera. Ako je do promjene došlo, vrši se ažuriranje stanja i kreiranje adekvatne poruke koja se šalje računaru u obliku TCP segmenta. Nakon toga, program ponovo prolazi kroz glavnu petlju ponavljajući opisani proces. Kompletan kod aplikacije dat je u [2]. Opisana glavna petlja je realizovana na sledeći način:

```
while (1) {
    //Konstantan prijem paketa
    xemacif input(netif);
    //Primi Ethernet komandu ako je prijem omogucen
    if(ethKomandaEnable==1)
        ehtKomanda(framePayload, komanda, payloadLen);
    //Provjera komande ako postoji
    if(strlen(komanda)!=0){
        //Ispis na prvoj liniji LCD-a?
        if (ProvjeriLCD(komanda)==1){
            XromLCDclear();
            LCDstampaj(komanda);}
        //Ispis na drugoj liniji LCD-a?
        else if (ProvjeriLCD(komanda)==2){
            XromLCDclear();
            XromLCDSetLine(2);
            LCDstampaj(komanda);}
        //Rotiranje LCD ekrana?
        else if(ProvjeriLCDRT(komanda)==1){
            LCDrotiraj();}
        //Komanda za uključivanje dioda?
        else if (ProvjeriLED(komanda)==1){
            if(LEDformiranje(komanda, diode)==500){
                PogresnaKomanda();}
            else { //Upisi novu kombinaciju dioda
                diode=LEDformiranje(komanda, diode);
                XGpio DiscreteWrite(&gpLED, 1, diode);}}
        //U suprotnom - pogresna komanda
        else PogresnaKomanda();
        //Očisti komandu i onemogući njen ponovni prijem
        komanda[0]='\0';
        ethKomandaEnable=0;
        //Registruj stanje tastera i prekidača
        psb_check = XGpio_DiscreteRead(&push, 1);
        dip_check = XGpio_DiscreteRead(&dip, 1);
        //Da li je doslo do promjene stanja tastera?
        if (t!=psb_check){
            for(i=0;i<=2;i++)//Ako jeste ispisi promjenu
                tasterPrethodno[i]=taster[i];
            AzurirajTastere(psb_check, taster);
            TasteriPromjena(taster, tasterPrethodno);}
        //Sacuvaj novo stanje
        t=psb_check;
        //Da li je doslo do promjene stanja prekidača?
        if (p!=dip_check){
            //Ako jeste ispisi novo stanje
            for(i=0;i<=3;i++)
                prekidacPrethodno[i]=prekidac[i];
            AzurirajPrekidace(dip_check, prekidac);
            PrekidaciStatus(prekidac);}
        //sacuvaj novo stanje
        p=dip_check;}
```

Na kraju je neophodno naglasiti važnost pravilnog alociranja memorije. Naime, prilikom realizacije ovog projekta javile su se teškoće u funkcionisanju aplikacije uzrokovane podrazumijevanim podešavanjima za alokaciju memorije u razvojnom okruženju *Software Development Kit*. Teškoće se manifestuju kroz povremeno blokiranje rada aplikacije i nemogućnost uspostavljanja komunikacije. Stoga je neophodno promijeniti pomenuta podešavanja. Prije svega, neophodno je za sve sekcije koda, sekcije podataka kao i za *heap* i stek (*stack*) dodijeliti DDR SDRAM memoriju na korišćenje. To je moguće uraditi u meniju *Tools* (opcija *Generate Linker Script*) u okviru *Software Development Kit*-a. Nakon toga je potrebno promijeniti dodijeljene vrijednosti

memorije za *heap* i stek tj. početne vrijednosti od 0x400 bajtova povećati na 0x8000 bajtova. Ukoliko bi ove vrijednosti ostale nepromijenjene, aplikacija zbog nedostatka memorije, ne bi mogla da funkcioniše korektno.

5. ZAKLJUČAK

U ovom radu je opisana interakcija sa Spartan 3E razvojnom platformom korišćenjem Ethernet interfejsa. Ovaj vid komunikacije je odabran upravo zbog svoje velike važnosti u vremenu u kom su sve više prisutne IP komunikacije. Osim toga, interakcija sa periferijama na razvojnoj platformi predstavlja dobru osnovu za rad na FPGA projektima, jer periferije predstavljaju modele raznih sistema koji se mogu pojaviti u rješavanju praktičnih problema.

Komunikacija između personalnog računara i razvojne pločice ostvarena je uspostavljanjem TCP konekcije tj. slanjem i prijemom TCP segmenata. Vodeći se opisanim principima, moguće je ostvariti i UDP komunikaciju. Sama priroda UDP komunikacije i činjenica da za njeno ostvarivanje nije potrebno prethodno uspostavljati konekciju, čini je jednostavnijom za implementiranje. Ostvarivanje ovakvog, sofisticiranog vida IP komunikacije, korišćenjem TCP i UDP protokola, predstavlja dobru osnovu za rad na brojnim projektima počevši od implementiranja prostih *web* servera, do rješavanja složenijih problema kroz kontrolu i manipulaciju različitim uređajima u praksi. Osim toga, principi izloženi u ovom radu mogu poslužiti prilikom projektovanja mrežnih uređaja na FPGA pločicama koje imaju više Ethernet interfejsa. Dalji rad u tom pravcu bi dao dobru osnovu za istraživanja u pogledu unaprijeđenja efikasnosti i funkcionalnosti aktivnih mrežnih uređaja kroz implementaciju raznih algoritama funkcionisanja.

LITERATURA

- [1] Xilinx, “Spartan-3E Starter Kit Board User Guide”, March 2006, url: http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf
- [2] N. Ljumović, “Primjer realizacije TCP/IP/Ethernet komunikacije PC računara i Spartan 3E FPGA pločice”, Specijalistički rad, Elektrotehnički fakultet Podgorica, Oktobar 2009.
- [3] Xilinx “MicroBlaze Processor Reference Guide”, January 2008, url: http://www.xilinx.com/support/documentation/sw_manuals/mb_ref_guide.pdf
- [4] Xilinx university program “Adding Custom IP to an Embedded System”, url: <http://users.utcluj.ro/~baruch/ssce/labor/EDK-L3-e.pdf>
- [5] Xilinx “XPS Ethernet Lite Media Access Controller”, June 2009, url: http://www.xilinx.com/support/documentation/ip_documentation/xps_ethernetlite.pdf
- [6] Xilinx “lwIP Library (v2.00.a)”, January 2007, url: http://www.xilinx.com/ise/embedded/edk91i_docs/lwip_v2_00_a.pdf
- [7] Xilinx “LightWeight IP (lwIP) Application Examples”, June 2009, url: http://www.xilinx.com/support/documentation/application_notes/xapp1026.pdf

PRIMJER VALIDACIJE XML-a POMOĆU DTD-a AN EXAMPLE OF XML VALIDATION WITH DTD

Sidita Duli, *Faculty of Natural Sciences, University of Shkoder, Albania*
Božo Krstajić, *Department of Electrical Engineering, University of Montenegro*

Sadržaj - XML je ključni standard za kodiranje poruka kod savremenih Web servisa. On pruža mnoge prednosti, a jedna od njih je odvojenost podataka od prezentacije. XML je unio revoluciju u IT svijet, kako kod podataka tako i u arhitekturi i softveru. U ovom radu je prezentovana potreba validacije XML koda i prikazan primjer iste pomoću DTD-a.

Abstract - Content XML is a key standard of coding messages in a Web Service. It offers many advantages; one of them is the separation of the data from the presentation. XML have brought revolutions in the IT world: in data, in architecture and in software. This paper will presented need of XML validation and an example of that with DTD.

1. INTRODUCTION

Web service is at once a technology, a process, and a phenomenon. As a technology it is a set of protocols that builds on the global connectivity made possible by SOAP (Simple Object Access Protocol) and the synergies of XML (Extensible Markup Language) and HTTP (Hypertext Transport Protocol). As a process, it is an approach to software discovery and connection over the Web. As a phenomenon, it's an industry-wide realization that the decentralized, loosely coupled, synergistic Web can't be ignored. [1]

Build on XML, a standard that is supported and accepted by thousands of vendor worldwide, Web services first focus on interoperability. XML is the syntax of messages, and HTTP, the underlying protocol, is how applications send XML messages to Web services in order to communicate. [2]

In this paper will be explained the role of DTD (Document Type Definition) and how it differs in content with the XML document. It is shown the format for both DTD and XML document. It is given two other schemes for XML validation, showing the key differences between them and DTD schema.

2. THE XML ADVANTAGES AND INTEGRATION

XML has requirement to coding the data. HTTP and SMTP (Simple Mail Transfer Protocol) both can be used to exchange the data between the client and the server, but none of them specifies in which way can be encoded the message in the body of the message sent. This message should be encoded in a standardized and in a platform independent way. XML is a natural choice for this aim. It can be used in Internet protocols. DCOM (Distributed Component Object Model), CORBA and Java RMI are encoding schemes that offer compatibility between hardware platforms. XML uses a text-based encoding scheme using character sets, and not binary encoding. SMTP and some other transport protocols transfers messages coded in texts. XML is easier in handling than some other schemes that use binary encoding. Many parsers make simpler the creation of the XML message on every platform. It has many supporting tools which enables the communication of every client in any platform with a web service.

The advantages of XML are listed below:

- It is easy like HTML
 - It is compatible with applications in Java for example, and with any other application which can process an XML document in that platform where is used.
 - It is very portable language. It can be used in wide networks like the Internet, working with many platforms, but on the other hand, it can also be used on palmtops or PDA systems.
 - It is an extensible language. Anyone can add other tags, like everyone can use the existing tags and there is no limit in the number of tags everybody can add. These new tags can be defined in the document type definition and the XML document can be successfully validated. This document type definition can be used for many XML documents that use the same structure of data.
 - It is platform independent. An application must only work with the XML; meanwhile XML has no problems of working with other platforms.
 - It is independent from the vendor and from the system, so all the data that is transferred between two systems will be sent without losing, even in the case when these two systems use different formats.
 - It is easy to search the data in a XML document. All the search engines can navigate through the XML tag-s and locate the required data.
 - The data in a XML document is formatted in a tree structure. The codes in XML can be read easily by humans because it is in a simple plain text.
 - Mostly relational databases can read and generate XML data.
 - It makes easier the data exchange. A third party tool can transform XML data to any other form of data.
 - XML can pass through firewalls. It is text based, and firewalls are opened for text based documents. XML can contain remote procedure calls, and in this way it can call remote methods on servers. Applications of different platforms and technologies can be integrated together.
- But XML has its own disadvantages. One of them is the fact that XML requires an application where it can be processed. In contrast with a HTML file, which can be written and anyone in the Web can see it through a browser, a document in XML can't be preceded in this way. None of actual browser can view the XML document. A solution

might be of converting the XML document in a HTML document.

XML web services are the method of delivering data and application interfaces. The data is transferred back and forth without changing the business rules and the validation intact. XML web services comprises the entire platform and with a defined structure for getting to and updating the data.

XML web services are the platform for application integration because they use open standards. Applications are constructed using multiple XML Web services from various sources that work together regardless of where they reside or how they were implemented.

All the XML web service has these in common:

-A web protocol which is used to expose the XML Web Service in the Web world for all users. Mostly the protocol used is SOAP.

-XML Web Service needs a way to be described its interface. Only by this detailed description a user can create a client application. The format of this description is a XML document too, called Web Services Description Language (WSDL) document.

-XML Web services must be registered in order to be found by the users. It's called the Universal Discovery Description and Integration (UDDI).

A main advantage of XML Web Service is that enables the communications of programs in different languages on different platforms.

The other significant advantage that XML Web services have over previous efforts is that they work with standard Web protocols—XML, HTTP and TCP/IP.

A significant number of companies already have a Web infrastructure, and people with knowledge and experience in maintaining it, so again, the cost of entry for XML Web services is significantly less than previous technologies.

Many businesses have the Web infrastructure, stuff that can maintain these technologies, so the cost of adding the XML Web Service technology is not so high.

As the definition specifies, XML Web Service is a service published in the Web, using SOAP protocol, with a description file WSDL and registered in the UDDI register. But what is the functionality of it? The aim of the early XML Web Services was to enable such information like weather forecast, to be incorporate into Web applications. This information could be analyzed and presented in different other ways.

XML Web services have advantages over the other Web service build over other technologies, for example on RPC. All the XML web services are platform independent, whereas RPC are platform dependent. This is because the information passed to the service provider is formatted in XML, using the protocol HTTP, which is supported by every browser.

XML did radical changes and offered many solutions in these below fields of computer sciences:

Ebusiness : In this field XML made changes in the communication business to business, the transactions between them, the messages transmitted.

Content management systems: As XML can divide the content from the presentation, in the system where the content is managed, XML plays a decisive role, separating content and using the styles in XSLT (XSL Transformations) to rebuild the presentation.

Data integration in syndications: Syndication is reusing the content of a site and changed the presentation to a different format. Generally, such applications are called portals, where for some users the data of a web page is costumed in format to be watched in a different style. XSLT make possible this, to change the presentation of such content, of syndication content or even of a web service. The most knows syndication, called Resource description framework site summary, RSS, is build using XML.

Data mining: even the fact that object oriented programming languages offered a new type of databases named OODBMS, XML has its impact in relational databases. It has created a new form of XML databases called native XML databases, where the information is XML data. All the database systems are integrating the new type of data, XML data as native data type.

3. THE XML EXCHANGE PROCESS

XML processing is mostly used in the server side. The reason is because all the developers know the tools available on the server, giving more control to the content than to the presentation. These tools will not change, so the server side XML processing will be written only for one configuration. The client side XML processing, on the other hand, includes such code to control if it has access from the browser, and after this, to process the XML document. A system that uses client-side XML processing must have the capability to control which browser versions are used.

Another way to create different outputs of XML data without leaving the client side is through the use of style sheets. In this way, the viewable content can change in response to user interaction by applying an XSL style sheet to the XML DOM (Document Object Model) representation. [3]

As a result, the option of client-side XML processing is often overlooked. The most common reason for this has to do with the wide variety of tools and implementations that exist across browsers and versions. Although this can be a strong factor in avoiding client-side processing, there are situations when these obstacles should be overcome and the client browser should be used to process XML.

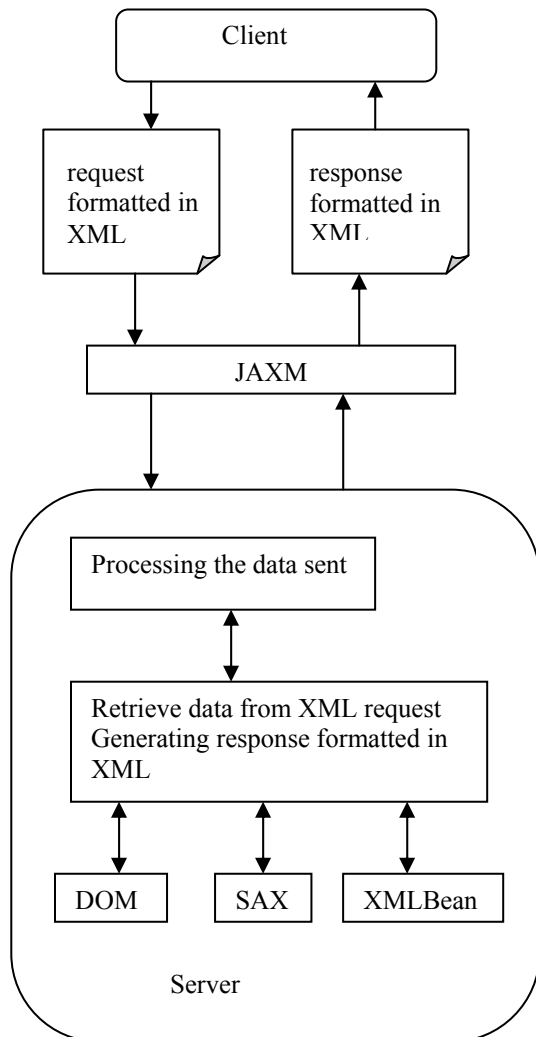
A XML document can be exchanged between a server and a client in an application that wants to process XML data. The client sends a request to the server. This request can be a document formatted in XML. The request done can be either a request calling a method to the remote server, or containing data that the server has to use and proceed. This request can also be as a request in a Web Service architecture, where is also user XML data documents. But when all this happens in a web service technology, the application that requests data must be registered as a web service. In this case the request sent from the application, the web service, is a SOAP request. SOAP requests are formatted in XML. This SOAP request includes all the information about the method which will be called to the remote server, and also includes information about the type of the message sent. This information is necessary to know to the server all about the format of the response that will be sent to the client. This SOAP request may also include the data that will be processed by the remote application in the server. Java API and its version Java API for XML Messaging JAXM is the technology

included to enable this message, though the standard of web service does not define mandatory the usage of JAXM.

The server, after has received the XML document, will process it, using one of these techniques: DOM, SAX(Simple API for XML) or XMLBeans. After it has finished processing, a new XML document can be created by the server, and this will be the response for the client.

Again will be used JAXM technique to pass the response to the client. In the case of a web service, these XML document will be according the SOAP protocol.

In the picture 1 it is shown a schema of a request response using XML documents. The two components, the client and the server, are shown as two main blocks, including all what they process.



Pic. 1. A client-server communication

4. XML VALIDATION WITH DTD

An XML document must be validated for a DTD (Document Type Definition). Below it is described the difference between an XML document and a DTD. The DTD document shows all the structure of a an XML document, its elements and attributes.

```

<!ELEMENT Product_info (Product)*>
<!ELEMENT Product (Name, Quantity, Price, Stock)>
<!ELEMENT Name (#PCDATA)>
  
```

```

<!ELEMENT Quantity (#PCDATA)>
<!ELEMENT Price (#PCDATA)>
<!ELEMENT Stock (#PCDATA)>
<!ATTLIST Product Product_Number CDATA
#REQUIRED>
  
```

This is an implementation of an XML document. It contains only data, and no information about the presentation. Here there are tags created from the user, containing information called metadata. The difference between this code and an HTML script is that, here no tag is for formatting, and we can easily handle only the data.

The XML schema below is written in XML language, but the DTD above is in a different language, not in XML.

```

<?xml version="1.0"?>
<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema" >
  <xs:element name="Product_Info"
type="ProductInfoType" />
  <xs:complexType name="ProductInfoType">
    <xs:sequence>
      <xs:element ref="Product" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="Product" type="ProductType" />
  <xs:complexType name="ProductType">
    <xs:sequence >
      <xs:element ref="Name" />
      <xs:element ref="Quantity" />
      <xs:element ref="Price" />
      <xs:element ref="Stock" />
    </xs:sequence>
    <xs:attribute name="Product_Number"
type="xs:int" use="required"/>
  </xs:complexType>
  <xs:element name="Name" type="xs:string" />
  <xs:element name="Quantity" type="xs:long" />
  <xs:element name="Price" type="xs:float" />
  <xs:element name="Stock" type="xs:long" />
</xs:schema>
  
```

XSD (XML Schema definition) is a successive of DTD and soon can even replace it. Some reasons for this, are the facts that XML Schemas are richer in tags, more powerful than DTD and are written im XML format. Unlike DTD, XML Schemas has support for data types and namespaces.

RELAX NG is another schema language for XML It specifies a pattern for the structure and content of an XML document. RELAX NG provides a choice of an XML syntax or a compact non-XML syntax to represent schemas. And this is different from other XML DTD-s. It does not allow entities and notations to be specified.

5. CONCLUSION

Using XML in a Web application offers the possibility of reformatting the content of page in different ways, searching only for data or writing metadata. It is the solution for many problems in integrating different application in different environments, and widely used in Web Service architecture.

XML document contains data about the data; meanwhile DTD consists of a structure on which a XML document is validated. DTD defines the building blocks of the XML document. Both, XML and DTD documents have different goals and different data included. While XML document is used as request or response document, DOM is used in the server side.

This paper shows the role of the DTD schema in an XML validation. It gives a concrete example that emphasizes the difference in structure between the XML document and the DTD document.

REFERENCES :

- [1] F. P. Coyle “ XML, Web Services and the data revolution “ Addison Wesley, March 2002
- [2] M. C. Daconta, L. J. Obrst , K. T. Smith “The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management ” John Wiley & Sons, 2003
- [3] C. Kochmer, E. Frandsen “JSP™ and XML Integrating XML and Web Services in Your JSP™ Application” Addison Wesley, March 2002

ANALIZA DIGITALNIH FILTARA KORIŠĆENJEM ALGEBARSKIH RAČUNARSKIH SISTEMA ANALYSIS OF DIGITAL FILTERS USING COMPUTER ALGEBRA SYSTEMS

Aleksandar Radonjić, *Crnogorski telekom, Podgorica, Crna Gora*
Budimir Lutovac, *ETF, Podgorica, Crna Gora*
Miroslav Lutovac, *Državni Univerzitet u Novom Pazaru, Srbija*

Sadržaj – *Prezentovana je analiza digitalnih filtara korišćenjem algebarskog računarskog sistema. Analiza započinje crtanjem šeme filtra. Funkcija prenosa i implementacioni kod se automatski generišu na osnovu nacrtane šeme filtra. Procedura je automatizovana korišćenjem grafičkog korisničkog interfejsa. Analiza kvantizacije koeficijenata i kvantizacije proizvoda se jednostavno rade korišćenjem automatski generisanog implementacionog koda.*

Abstract – *The analysis of digital filters using computer algebra system is presented. The analysis starts by drawing a schematic of the filter. The transfer function and implementation code is derived from the schematic. The whole procedure is automated using graphical user interface. Analysis of coefficient quantization and product quantization can be simply done using the automatically derived implementation code.*

1. UVOD

Digitalni filtri se uobičajeno implementiraju korišćenjem numeričkih računarskih sistemima zato što, u osnovi, digitalno filtriranje radi sa sekvencama koje su numeričke. Najpoznatiji alat za numeričko procesiranje je Matlab [1]. U ovom alatu postoje brojne komande koje se koriste za automatizovanu analizu digitalnih filtara. Na primer, Filter Design Toolbox u Matlabu omogućava da se sve analize urade korišćenjem grafičkog korisničkog interfejsa (GUI – graphical user interface). Rezultati analiza su numeričke ili grafičke. Ceo postupak je automatizovan počev od zadavanja specifikacija koje filter treba da zadovolji, do prikaza uticaja kvantizacije koeficijenata filtra ili kvantizacije proizvoda.

Literatura koja opisuje rad digitalnih filtara je zasnovana na složenoj teoriji za koju se podrazumeva da je korisnici znaju. Pa ipak, u suštini, ceo postupak projektovanja i analize rada digitalnog filtra može se uraditi i bez detaljnog izučavanja teorije i dovoljno je samo pratiti uputstva za korišćenje softverskih alata. Sve lepo funkcioniše do trenutka kada se pojave problemi, na primer kada se dobijaju rezultati koji se ne očekuju. To se često dešava kada se koristi predstavljanje koeficijenata filtra i vrednosti digitalnih signala u aritmetici sa fiksnim zarezom (aritmetika sa fiksnom tačkom u engleskoj notaciji).

Po nekada se dešava da filtri ne rade onako kako se očekuje i tada treba pronaći grešku. Greške mogu da nastanu zato što se pretpostavlja da filter radi na jedan način, na primer kao digitalni filter implementiran u direktnoj formi, a procesiranje se izvršava u transponovanoj formi. U tim slučajevima je teško naći grešku zato što se ne zna kako su nastali rezultati koji postoje samo kao brojevi.

U ovom radu je pokazano da se za analizu rada digitalnih filtara mogu koristiti softverski alati koji pripadaju klasi algebarskih računarskih sistema (*Mathematica* [2]), kao i aplikacioni paket koji radi u ovom sistemu (*SchematicSolver* [3]).

Korišćeni alati iz klase algebarskih računarskih sistema već su pokazali da je moguće pronaći osobine analiziranih sistema i relacije koje daju vezu parametara kontinualnih,

vremenski diskretnih i nelinearnih sistema u zatvorenoj formi [4], [5], [6].

U sledećem poglavlju biće opisano kako se koriste *SchematicSolver* i *Mathematica* za analizu digitalnih filtara. Najvažnija osobina ovih alata jeste da se koristi šema digitalnog sistema kao generalni zapis koji se koristi i za crtanje sistema, ali i za automatizovano određivanje funkcije prenosa i generisanje implementacionog koda. Implementacionom kodu se mogu jednostavno dodati posebno definisane funkcije, na primer za analizu uticaja kvantizacije proizvoda pri obradi nekog signala u aritmetici sa fiksnim zarezom.

2. CRTANJE ŠEME DIGITALNOG FILTRA

Uobičajeno je da se za crtanje šema sistema koriste posebni alati koji su specijalno namenjeni za crtanje. Međutim, takvi alati nisu povezani sa ostalim analizama pa se može dogoditi da slika prikazuje šemu jednog sistema, a da analiza odgovara nekoj sasvim drugačijoj šemi. Osnovna ideja alata *SchematicSolver* jeste da sve analize koje slede odgovaraju šemi koja je nacrtana. Tako na primer, ako su koeficijenti množača u povratnoj grani nacrtani i označeni na šemi simbolom a , tada i funkcija prenosa koja se izvodi na osnovu te šeme takođe treba da ima koeficijente označene simbolom a u rekurzivnoj grani. Često se dešava da projektanti na šemi označe koeficijente jednim simbolom, a kada koriste matematičku interpretaciju koriste drugi simbol preuzet iz nekog publikovanog rada. Ako se greškom zamene simboli u direktnoj i rekurzivnoj grani, može da se dogodi da filter ima drugačiju funkciju prenosa od one koja odgovara nacrtanoj šemi filtra.

Tipična greška koja se dešava jeste i da se obrne redosled koeficijenata, pa polovi umesto da su unutar jediničnog kruga, tada se nalaze izvan jediničnog kruga. Kada se nacrtaju funkcija prenosa, ova greška ne može da se uoči, ali zato

umesto da odziv filtra na impulsnu pobudu konvergira ka nuli, odziv filtra raste i teži ka beskonačnosti.

Za analizu u ovom radu korišćen je softverski paket *Mathematica*. Nakon toga je potrebno da se učitava znanje iz aplikacionog paketa *SchematicSolver*. Znanje se učitava komandom **Needs**

```
Needs["SchematicSolver`"]
```

Za crtanje šeme digitalnog filtra može se koristiti paleta za crtanje kao vrsta GUI-a. Klikom na odgovarajuće dugme se učitava znanje, definišu standardni parametri za crtanje i otvara ćelija sa slikom koja služi kao tabla za crtanje. Elementi filtra se unose tako što se klikne na dugme sa slikom elementa, a zatim na tabli označi gde treba da se nalazi element u šemi filtra. Istovremenom sa crtanjem se generiše lista koja sadrži sve nacrtane elemente. Lista sa tekstualnim opisom filtra, ili specifikacija šeme kako se još naziva, se nalazu u posebnoj ćeliji. Kada je generisana specifikacija šeme, palete za crtanje se mogu zatvoriti da ne ometaju dalji rad. Dodatne korekcije i izmene se mogu raditi direktno u tekstualnom zapisu šeme. Sledi primer jednog tekstualnog opisa šeme digitalnog filtra, ili specifikacije šeme jednog filtra koja je dodeljena promenljivoj **mySchematic**

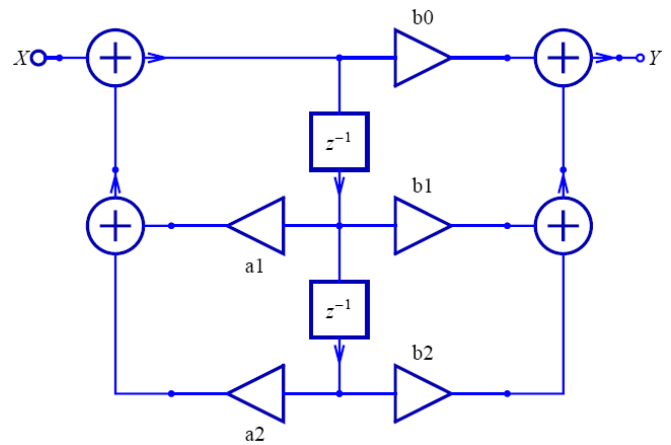
```
mySchematic = {
  {"Input", {2, 10}, X, "", TextOffset -> {1, 0}},
  {"Adder", {{2, 10}, {3, 8}, {7, 10}, {3, 11}}, {1, 1, 2, 0}},
  {"Adder", {{2, 7}, {4, 4}, {4, 7}, {3, 8}}, {0, 1, 1, 2}},
  {"Multiplier", {{7, 7}, {4, 7}}, a1},
  {"Delay", {{7, 10}, {7, 7}}, 1},
  {"Delay", {{7, 7}, {7, 4}}, 1},
  {"Multiplier", {{7, 4}, {4, 4}}, a2},
  {"Multiplier", {{7, 10}, {10, 10}}, b0},
  {"Multiplier", {{7, 7}, {10, 7}}, b1},
  {"Multiplier", {{7, 4}, {10, 4}}, b2},
  {"Adder", {{10, 10}, {11, 8}, {12, 10}, {11, 11}}, {1, 1, 2, 0}},
  {"Adder", {{10, 7}, {10, 4}, {12, 7}, {11, 8}}, {1, 1, 0, 2}},
  {"Output", {12, 10}, Y, "", TextOffset -> {-1, 0}};
```

Tekstualni opis se sada može koristiti za sve ostale potrebe, za crtanje šeme, za izračunavanje funkcije prenosa ili za generisanje implementacionog koda. Za crtanje se koristi komanda **ShowSchematic**, koja za argument ima promenljivu **mySchematic**.

```
ShowSchematic[mySchematic, Frame -> False, GridLines -> None]
```

Mogu se definisati opcije za crtanje. Na primer, dok se crta šema, poželjno je da se vide koordinate po ordinati i apscisi, kao i mreža koja olakšava vizuelno pozicioniranje elemenata filtra. Kada je šema nacrtana, poželjno je da se vidi samo šema bez ostalih nepotrebnih oznaka, i to se radi korišćenjem opcija koje isključuju mrežu za crtanje i koordinate. Sabirači su vizuelno označeni krugovima a rezultat sabiranja je označen strelicom. Množači su označeni trouglom koji pokazuje smer kretanja signala. Uz trougao je simbolički definisana vrednost konstante množenja. Memorijski element je označen kvadratom u kome eksponent označava kašnjenje u odbircima.

Na slici 1 je nacrtana šema koju generiše komanda **ShowSchematic**, a za filtersku specifikaciju koja je dodeljena promenljivoj **mySchematic**. Radi se o digitalnom IIR filtru drugog reda koja odgovara direktnoj formi.



Slika 1. Primer digitalnog IIR filtra – direktna forma.

Na slici su označene simboličke vrednosti koeficijenata množenja kao i ulaz i izlaz iz filtra.

2. FUNKCIJA PRENOSA

Za izračunavanje funkcije prenosa se koristi komanda **DiscreteSystemTransferFunction**

Argument ove funkcije je isti kao i kod komande za crtanje šeme filtra, **mySchematic**

```
myTransferFunction = DiscreteSystemTransferFunction[mySchematic];
myTF = myTransferFunction[[1]][[1, 1]]
```

Rezultat izračunavanja je matrica funkcija prenosa, pošto je alat predviđen da ima jedan ili više ulaza, kao i jedan ili više izlaza. Matrica funkcija prenosa se dodeljuje promenljivoj **myTransferFunction**. Da bi se izdvojila samo jedna funkcija prenosa, potrebno je iz matrice uzeti odgovarajući element. Sama funkcija prenosa je dodeljena promenljivoj **myTF**.

Ceo postupak određivanja funkcije prenosa je automatizovan i odvija se klikom na jedno od dugmadi palete za crtanje šema. U ovom primeru je izdvojena samo komanda da bi se lakše uočile karakteristike softvera.

Rezultat automatski generisane funkcije prenosa je sledeći

$$-\frac{b_2 + b_1 z + b_0 z^2}{a_2 + a_1 z - z^2}$$

Uobičajeno je da se u funkcijama prenosa koeficijenti grupišu uz $1/z$. Ovakav rezultat se dobija komandom čiji argument je funkcija prenosa **myTF**

DiscreteSystemDisplayForm

Rezultat je sledeći

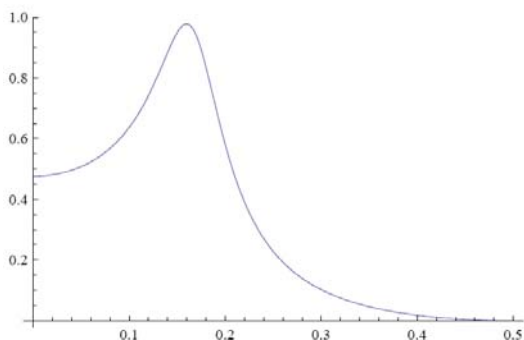
$$\frac{-b_0 - b_1 z^{-1} - b_2 z^{-2}}{-1 + a_1 z^{-1} + a_2 z^{-2}}$$

Znak uz slobodan član u imeniocu treba da bude 1, pa se lako pokazuje da se dobija željena funkcija prenosa

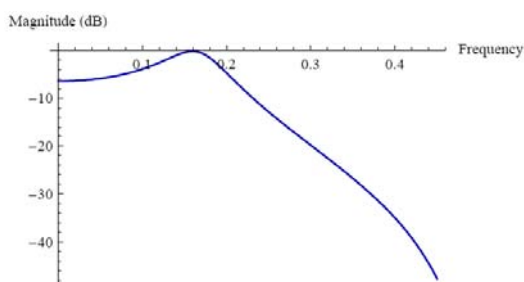
$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} - a_2 z^{-2}} \quad (1)$$

Ovde uočavamo da za ovako nacrtani filter, funkcija prenosa ima negativne vrednosti uz koeficijente u imeniocu, dok se u literaturi često mogu videti izrazi u kojima je znak + uz ove članove. Razlika u znaku uz koeficijente u imeniocu je inače razlog čestih grešaka koje se mogu naći u praksi, a to

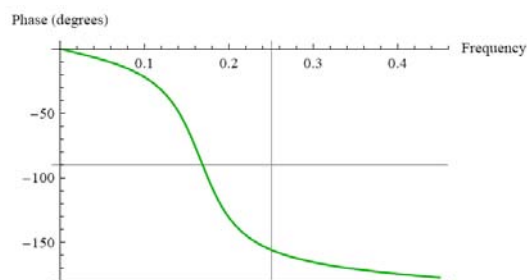
je da bi na osnovu šeme trebalo koristiti negativne vrednosti koeficijenata, a na osnovu teorijske funkcije prenosa se očekuju pozitivne vrednosti uz koeficijente.



Slika 2. Amplitudska karakteristika izračunata po definiciji.



Slika 3. Amplitudska karakteristika – pojačanje u dB.



Slika 4. Fazna karakteristika.

4. FREKVENCIJSKA KARAKTERISTIKA

Funkcija prenosa je određena u simboličkoj formi. Da bi nacrtali amplitudsku ili faznu karakteristiku, moraju se simboličke vrednosti zameniti numeričkim. Zato se najpre definišu zamene simbolički definisanih promenljivih sa numeričkim vrednostima koje se koriste samo onda kada je to potrebno. Promenljiva **values** sadrži smene, a nova funkcija **myTFn** se dobija iz opšteg izraza kada se izvrše smene simboličkih vrednosti sa numeričkim

```
values = {a1 → 0.8, a2 → -0.64, b0 → 0.1, b1 → 0.2, b2 → 0.1};
myTFn = myTF /. values;
% // DiscreteSystemDisplayForm
```

Dobija se funkcija prenosa sa numeričkim vrednostima koeficijenata

$$\frac{-0.1 - 0.2 z^{-1} - 0.1 z^{-2}}{-1. + 0.8 z^{-1} - 0.64 z^{-2}}$$

Sada se može nacrtati i amplitudska karakteristika. Jedan način je da se iskoristi formula po definiciji

```
Plot[Abs[myTFn /. z → ei 2 π f], {f, 0, 0.5}]
```

Na slici 2 je nacrtana amplitudska karakteristika u funkciji učestanosti, odnosno kada je nezavisno promenljiva z zamenjena eksponencijalnim izrazom po f .

Može se koristiti i komanda koja automatski crta amplitudsku karakteristiku (u dB), datoj na slici 3, i faznu karakteristiku na slici 4.

5. GENERISANJE IMPLEMENTACIONOG KODA

Generisanje implementacionog koda je automatizovano korišćenjem jednog dugmeta na paleti koja izvršava ceo niz komandi. Radi jednostavnosti, ovde će biti prikazane samo najvažnije komande. Najpre se zadaje ime funkcije za implementacioni kod, **dfiltDF2**, a zatim se koristi komanda **DiscreteSystemImplementation** koja tekstualni opis šeme pretvara u set jednačina, a zatim generiše funkciju koja se može koristiti i kao nova komanda sa imenom implementacionog koda **dfiltDF2**

```
procedureName = dfiltDF2;
DiscreteSystemImplementation[mySchematic, ToString[procedureName]]
```

S obzirom da su svi parametri na šemi bili simbolički zadati, to i implementacioni kod očekuje da su njegovi argumenti promenljive sa istim nazivima. Zato se definišu argumenti funkcije koja će kasnije izvršiti procesiranje sa implementacionim kodom

```
initialConditions = {0, 0};
systemParameters = {a1, a2, b0, b1, b2};
```

Početna stanja registara i lista sistemskih parametara se automatski generišu, a ovde su prikazani rezultati koji se dobijaju klikom na odgovarajuće dugme palete.

Za procesiranje vremenski-diskretnim sistemom, ili u ovom slučaju filtriranje digitalnim filtrom, potrebno je definisati ulazne sekvence, u ovom slučaju impulsnu pobudu, i pozvati funkciju za procesiranje

DiscreteSystemImplementationProcessing

```
inputSequenceD = UnitImpulseSequence[];
{outputSequence, finalConditions} =
  DiscreteSystemImplementationProcessing[
    inputSequenceD, initialConditions,
    systemParameters, procedureName];
outputSequence[[1 ;; 6]] // Simplify // Expand
```

Kako su parametri sistema bili zadati simbolima, to je odziv na impulsnu pobudu u funkciji simboličkih parametara

$$\{ \{b_0\}, \{a_1 b_0 + b_1\}, \{a_1^2 b_0 + a_2 b_0 + a_1 b_1 + b_2\}, \{a_1^3 b_0 + 2 a_1 a_2 b_0 + a_1^2 b_1 + a_2 b_1 + a_1 b_2\}, \{a_1^4 b_0 + 3 a_1^2 a_2 b_0 + a_2^2 b_0 + a_1^3 b_1 + 2 a_1 a_2 b_1 + a_1^2 b_2 + a_2 b_2\}, \{a_1^5 b_0 + 4 a_1^3 a_2 b_0 + 3 a_1 a_2^2 b_0 + a_1^4 b_1 + 3 a_1^2 a_2 b_1 + a_2^2 b_1 + a_1^3 b_2 + 2 a_1 a_2 b_2\} \}$$

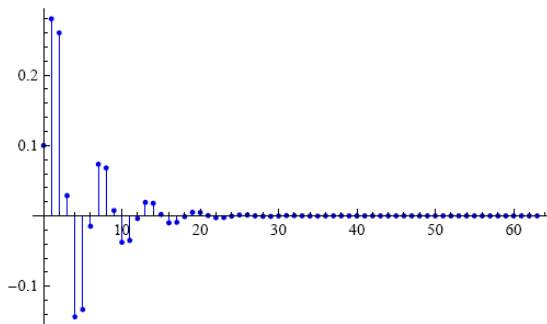
Ovu vrstu obrade nazivamo simboličko procesiranje zato što je svaki odbirak izlaznog signala u funkciji simboličkih vrednosti parametara sistema. Na ovaj način možemo da uočimo kako svaki od parametara sistema utiče na vrednost svakog odbirka izlaznog signala.

6. NUMERIČKA OBRADA SIGNALA

U bilo kom trenutku se simboličke vrednosti mogu zameniti numeričkim. Tako na primer, ako treba nacrtati

odziv filtra na impulsnu pobudu, tada sve vrednosti izlazne sekvence moraju biti numeričke. Najjednostavnije je da se pre procesiranja izvrši zamena simbolički zadatih parametara odgovarajućim numeričkim vrednostima. Na primer, korišćenjem promenljive `values` dobija se odziv na impulsnu pobudu prikazan na slici 5

```
inputSequence = UnitImpulseSequence[64];
{outputSequence, finalConditions} = DiscreteSystemImplementati
  inputSequence, initialConditions, systemParameters /. values,
outputSequence[[1 ;; 6]]
```

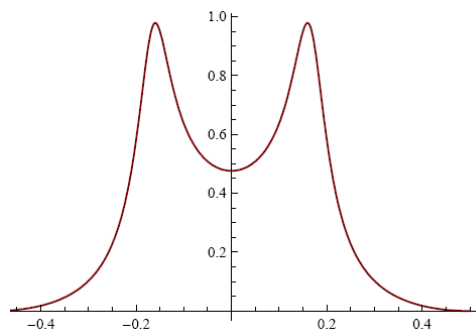


Slika 5. Odziv na impulsnu pobudu.

7. SPEKTAR SIGNALA

Za izračunavanje spektra signala može se koristiti

```
SequenceFourierTransformMagnitudePlot[outputSequence,
  NormalizedSpectrum -> False];
```



Slika 6. Spektar signala.

Spektar signala je nacrtan na slici 6. Može se uočiti da je spektar signala odziva na impulsnu pobudu identičan amplitudskoj karakteristici sa slike 2. Ovime smo dokazali da procesiranje radi ispravno.

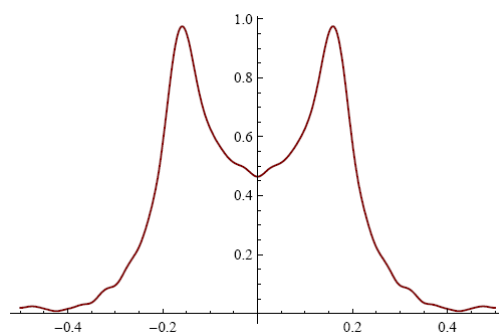
8. ANALIZA EFEKATA KVANTIZACIJE

Za potrebe analize uticaja kvantizacije, za procesiranje u aritmetici sa fiksnim zarezom, može se definisati funkcija kvantizacije, na primer,

```
Quantize[x_, b_ : 8, q_ : 0] := Module[{},
  If[q == 0
  , y = Round[x * 2^b] / 2^b;
  , y = Floor[2^nm * x] / 2^nm];
  y];
```

Kvantizacijom koeficijenata filtra novom funkcijom `Quantize` pojaviće se nule funkcije prenosa koje nisu postojale u idealnom slučaju, a ako se svaka operacija

množenja zameni sa `Quantize`, tada se mogu uočiti deformacije u spektru koji nije više gladak, videti sliku 7.



Slika 7. Spektar signala sa kvantizacijom proizvoda.

9. ZAKLJUČAK

U ovom radu je pokazano kako se koristi softverski alat koji pripada klasi algebarskih računarskih sistema za analizu, generisanje implementacionog koda i procesiranje digitalnih IIR filtara. Ceo postupak se zasniva na generisanju opisa filtra, koji se dobija crtanjem šeme filtra. Tekstualni opis se dodeljuje promenljivoj koja se zatim koristi kao argument funkcija za crtanje šeme, izračunavanje funkcije prenosa, generisanje implementacionog koda i procesiranje. Simboličke vrednosti se zamenjuju samo onda kada je potrebno, na primer da bi se nacrtala amplitudska karakteristika, ili odziv na impulsnu pobudu ili spektar signala na izlazu iz filtra.

ZAHVALNICA

Ovaj rad je delimično finansiralo Ministarstvo nauke Republike Srbije, Projekat TR11002.

LITERATURA

- [1] MATLAB Version 7, MathWorks, Inc., Natick, MA, 2005
- [2] S. Wolfram, *The Mathematica Book*, Cambridge: Cambridge, University Press, Wolfram Media, 2003.
- [3] M. D. Lutovac and D. V. Tosić, *SchematicSolver* Version 2, www.schematicsolver.com, 2009.
- [4] M. D. Lutovac and D. V. Tosić, "Symbolic Analysis and Design of Control Systems using Mathematica," *International Journal of Control*, Special Issue on Symbolic Computing in Control, vol.79, no.11, pp.1368–1381, 2006.
- [5] M. D. Lutovac, J. Čertić, and Lj. Milić, "Digital Filter Design using Computer Algebra Systems," *Circuits, Syst. and Signal Process*, Birkhäuser Boston, vol. 29: pp. 51–64. 2010.
- [6] V.M. Poučki, A. Žemva, M. D. Lutovac, T. Karčnik, "Elliptic IIR filter sharpening implemented on FPGA," *Digital Signal Processing*, vol. 20, pp. 13–22, 2010.

SISTEM VIŠESTRUKIH PODNOSILACA BAZIRAN NA AFFINE FOURIEROVOJ TRANSFORMACIJI U VAZDUHOPLOVNIM KANALIMA A MULTICARRIER SYSTEM BASED ON THE AFFINE FOURIER TRANSFORM IN AERONAUTICAL CHANNELS

Alen Pelinković, Igor Đurović, *Elektrotehnički fakultet Podgorica,*
Đuro Stojanović, *Crnogorski Telekom*

Sadržaj – *Affine Fourierova transformacija (AFT), opšta formulacija chirp transformacija, je nedavno predložena za upotrebu u komunikacijama sa višestrukim nosiocima. AFT je uopštenje Fourierove i frakcione Fourierove transformacije (FrFT). Predstavljamo aplikaciju AFT sistema baziranog na višestrukim nosiocima (AFT-MC multicarrier) u vazduhoplovnim vremenski-promjenljivim kanalima. Opisana je mogućnost AFT-MC sistema u suzbijanju interferencije u vazduhoplovnim kanalima. Data je gornja i donja granica interferencije, praćena aproksimacijom snage interferencije koja značajno pojednostavljuje analizu interferencije. Osim toga, data je detaljna analiza interferencije i optimalni parametri za različite scenarije kanala: na putu, dolazak i poletanje. Pokazano je da se AFT-MC može prilagoditi promjenama tokom različitih stanja leta i da efikasno suzbija interferenciju izazvanu Doplerovim širenjem.*

Abstract – *The AFT, a general formulation of chirp transforms, has been recently proposed for use in multicarrier communications. The AFT represents a generalization of the Fourier and fractional Fourier transform. We present an application of AFT based multicarrier system in aeronautical time-varying multipath channels. An interference suppression ability of AFT-MC system in aeronautical channels is described. Upper and lower bound on interference power are given, following by interference power approximation that significantly simplifies interference analysis. Moreover, detailed interference analysis and optimal parameters are given for different channel scenarios: en-route, arrival and takeoff. It is shown that AFT-MC is able to match changes during the different states of the flight and efficiently suppress interference caused by the Doppler spread.*

1. UVOD

Konvencionalni komunikacioni sistemi u vazduhoplovstvu koriste amplitudsku modulaciju AM (Amplitude Modulations) u VHF (Very High Frequency) opsegu. U cilju poboljšanja efikasnosti i sigurnosti potrebno je uvesti nove tehnike digitalnog prenosa [1]. Digitalni sistemi sa višestrukim nosiocima predstavljaju najboljeg kandidata za buduće komunikacione sisteme u vazduhoplovstvu. Sistemi sa višestrukim nosiocima su već pronašli svoje mjesto u mnogobrojnim aplikacijama u komunikacionim sistemima, gdje je tipično razmatran ortogonalni frekvencijski multipleks (OFDM), baziran na Fourierovoj transformaciji. OFDM predstavlja jednu od najvažnijih klasa modulacionih tehnika sa jednako razmaknutim podnosiocima i preklapanjem spektra [2]. Implementiran je u fizičkom sloju mnogih bežičnih komunikacionih standarda. OFDM se pokazao kao efikasna tehnika za prenos podataka visokim protokom. Međutim, njegova velika osjetljivost na Doplerovo širenje ozbilno narušava njegovu efikasnost u vazduhoplovnim kanalima, naročito tokom leta.

U fokusu aktuelnih istraživanja je pogodna modulaciona tehnika za vezu dugog dometa, manevrisanje terminala i letova na relacijama [1]. Iako je OFDM naizgled prvi izbor kao najpopularnija tehnika sa višestrukim nosiocima, njegova Fourierova baza nije optimalna za transmisiju u vazduhoplovnim kanalima. Sistem sa višestrukim nosiocima baziran na AFT-MC, je od skora predložen kao obećavajuća

tehnika za prenos u bežičnim kanalima. AFT-MC je odabran kao pogodan u komunikacijama sa višestrukim nosiocima kao što su vazduhoplovne i satelitske komunikacije [3].

U ovom radu, ćemo proširiti analizu performansi AFT-MC sistema do opštih vremenski-promjenljivih kanala sa fokusom na dva najznačajnija vazduhoplovna kanala: na putu (relaciji), dolazak i poletanje. Dobijena je gornja i donja granica interferencije. Kako AFT-MC predstavlja uopštenje OFDM-a, on će se u najgorem ponašati kao OFDM. U svim ostalim situacijama kada je dominantna LOS komponenta i kada je uzak snop rasijanih komponenti AFT-MC ima bolje karakteristike od OFDM-a.

Ovaj rad je organizovan na sljedeći način. Performanse signaliziranja su predstavljene u dijelu 2. Analiza momenata i optimalni parametri su dati u dijelu 3. Primjeri praktičnih implementacija u vazduhoplovnim kanalima su dati u dijelu 4. Konačno, na kraju je dat zaključak.

2. ANALIZA PERFORMANSI

AFT-MC signal u osnovnom opsegu je definisan kao:

$$s(t) = \sum_{n=-\infty}^{\infty} \sum_{k=0}^{M-1} c_{n,k} g_{n,k}(t), \quad (1)$$

gdje je M broj podnosilaca, $\{c_{n,k}\}$ simboli, n i k su i intervali odabiranja i redni brojevi podnosilaca, respektivno, $g_{n,k}$

predstavlja vremenski pomjerenu i modulisanu (chirp bazom) verziju osnovnog impulsa $g(t)$

$$g_{n,k}(t) = g(t-nT)e^{j2\pi(c_1(t-nT)^2 + c_2k^2 + \frac{k}{T}(t-nT))}, \quad (2)$$

gdje je T period simbola, c_1 i c_2 su AFT parametri. Pretpostavljeno je da su $\{c_{n,k}\}$ statistički nezavisne, identički distribuirane promjenljive, sa nultom srednjom vrijednošću i varijansom 1. Dvostruko disperzivni kanal se može modelovati multipath fading linearnim operatorom \mathbf{H} . Signal na prijemu je dat kao [4]

$$r(t) = (\mathbf{H}s)(t) + n(t), \quad (3)$$

gdje $n(t)$ predstavlja aditivni bijeli Gausov šum (AWGN- Additive White Gaussian Noise), sa jednostranom spektralnom gustinom snage šuma N_0 . Rekonstrukcija signala na prijemu se obavlja projekcijom primljenog signala na set signala $\{g_{n,k}(t)\}$. Primljeni simboli se mogu predstaviti kao [6]

$$\begin{aligned} \hat{c}_{n,k} &= \langle \mathbf{H}s, g_{n,k} \rangle + \langle n, g_{n,k} \rangle \\ &= \sum_{n'=-\infty}^{\infty} \sum_{k'=0}^{M-1} c_{n',k'} \langle \mathbf{H}g_{n',k'}, g_{n,k} \rangle + \langle n, g_{n,k} \rangle. \end{aligned} \quad (4)$$

Slično kao klasična ambiquity funkcija [6], $A(\tau_p, \nu_p)$ se može definisati kao linearno transformisana ambiquity funkcija [3]

$$A(\tau_p, \nu_p) = \int_{-\infty}^{\infty} g(t)g^*(t-\tau_p)e^{-j2\pi\nu_p t} dt, \quad (5)$$

gdje je

$$\begin{aligned} \tau_p &= (n'-n)T + \tau, \\ \nu_p &= \frac{1}{T}(k'-k) + \nu - c_0 - 2c_1((n'-n)T + \tau). \end{aligned} \quad (6)$$

Za kanale koji su stacionarni u širem smislu i kod kojih su slabljenje i fazni pomjeraj međusobno nekorelisani (WSSUS- Wide Sense Stationary Uncorrelated Scattering) [4], scattering funkcija $S(\tau, \nu)$ koja u potpunosti opisuje WSSUS kanal je

$$E[h(\tau, \nu)h^*(\tau_1, \nu_1)] = S(\tau, \nu)\delta(\tau - \tau_1)\delta(\nu - \nu_1). \quad (8)$$

Korisna snaga je dobijena kao [3]

$$P_U = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(\tau, \nu) |A(\tau_p, \nu_p)|_{n=n', k=k'}^2 d\tau d\nu. \quad (9)$$

Snaga interferencije P_I u praktičnim bežičnim kanalima, gdje vremensko i frekvencijsko širenje ima konačnu vrijednost, $\tau \in [0, \tau_{\max}]$ i $\nu \in [-\nu_d, \nu_d]$ se može izraziti kao [3]

$$P_I = 1 - \int_{-\nu_d}^{\nu_d} \int_0^{\tau_{\max}} S(\tau, \nu) |A(\tau_p, \nu_p)|_{n=n', k=k'}^2 d\tau d\nu. \quad (10)$$

AFT-MC sistem se svodi respektivno na FrFT i OFDM baziran sistem za $c_1 = \cot\alpha/(4\pi)$ i $c_1 = 0$ u (6) [3].

Razmotrimo uopšten slučaj, gdje imamo AFT-MC sistem sa pravougaonim impulsima $g(t) = 1/\sqrt{T}$, $-T/2 \leq t \leq T/2$ bez zaštitnog intervala GI (Guard Interval). Ovo je uopšten slučaj za razliku od [3] gdje je ispitivan samo AFT-MC sa GI. U ovom slučaju se ambiguity funkcija $|A(\tau_p, \nu_p)|_{n=n', k=k'}^2$ za $n=n'$ i $k=k'$ može izraziti kao

$$|A(\tau_p, \nu_p)|_{n=n', k=k'}^2 = \frac{\sin^2 \pi(\nu - c_0 - 2c_1\tau)(T - \tau)}{\pi^2(\nu - c_0 - 2c_1\tau)^2 T^2}. \quad (11)$$

Pogodno je između simbola dodati GI trajanja T_G ($T_G > \tau_{\max}$) u cilju eliminisanja kašnjenja prouzrokovanog višestrukom propagacijom. Interferencija se može ograničiti sa gornje strane uvrštavanjem (11) u (10) i korišćenjem osobina Tejlorovog reda $1 - \cos \theta \geq \frac{1}{2}\theta^2 - \frac{1}{24}\theta^4$ kao

$$P_{IUB} \leq \frac{1}{3}m_{20}(c_0, c_1)\pi^2 T^2, \quad (12)$$

gdje $m_{ij}(c_0, c_1)$ za $i, j \in N$ predstavlja momente scattering funkcije za AFT-MC

$$\begin{aligned} m_{ij}(c_0, c_1) &= \int_{-\nu_d}^{\nu_d} \int_0^{\tau_{\max}} S(\tau, \nu) \\ &\quad \times (\nu - c_0 - 2c_1\tau)^i \tau^j d\tau d\nu. \end{aligned} \quad (13)$$

Slično se može izraziti donja granica interferencije korišćenjem $1 - \cos \theta \leq \frac{1}{2}\theta^2 - \frac{1}{24}\theta^4 - \frac{1}{720}\theta^6$, (10) i (11) kao

$$P_{ILB} \geq \frac{1}{3}m_{20}(c_0, c_1)\pi^2 T^2 - \frac{2}{45}m_{40}(c_0, c_1)\pi^4 T^4.$$

(14) OFDM momenti $m_{ij}(0,0)$ se mogu dobiti za $c_0 = 0$ i $c_1 = 1$. AFT-MC momenti $m_{ij}(c_0, c_1)$ se mogu izračunati korišćenjem $m_{ij}(0,0)$ za OFDM kao :

$$\begin{aligned} m_{ij}(0,0) &= \sum_{k=0}^i \sum_{l=0}^{i-k} (-1)^{l+k} \binom{i}{k} \binom{i-k}{l} \\ &\quad \times c_0^l (2c_1)^k m_{i-k-l, k+j}(0,0). \end{aligned} \quad (15)$$

Bitno je naglasiti da (12) i (14) predstavlja AFT-MC ekvivalent za gornju i donju granicu za prethodno izveden OFDM u [5].

Za $(v + |c_0| + 2|c_1|\tau_{\max})T > 1$ (trajanje simbola i brzina su veliki) i ako je dodat dovoljan GI, aproksimacija interferencije se može napisati kao [3]

$$P_I = \frac{\left(\frac{1}{3}\right)\left(\frac{1}{K+1}\right)m_{20}(c_0, c_1)\pi^2 T^2}{\left(\frac{1}{K+1}\right) + \left(\frac{1}{3}\right)m_{20}(c_0, c_1)\pi^2 T^2} \quad (16)$$

3. ANALIZA MOMENATA

Momenti $m_{20}(0,0)$ i $m_{02}(0,0)$ imaju važno fizičko značenje, s obzirom da predstavljaju Doplerovo širenje v_m i kašnjenje τ_m kanala u OFDM sistemu, respektivno. Optimalni koeficijenti c_{0opt} i c_{1opt} se mogu izračunati minimizacijom v_m (c_0, c_1) kao [3] i [7]

$$c_{0opt} = \frac{m_{02}(0,0)m_{10}(0,0) - m_{01}(0,0)m_{11}(0,0)}{m_{02}(0,0) - m_{01}^2(0,0)},$$

$$c_{1opt} = \frac{m_{11}(0,0) - m_{01}(0,0)m_{10}(0,0)}{2(m_{02}(0,0) - m_{01}^2(0,0))}. \quad (17)$$

Koeficijenti $m_{10}(0,0)$ i $m_{01}(0,0)$ predstavljaju srednje Doplerovo širenje i srednje kašnjenje, respektivno. U narednom dijelu će biti izračunati optimalni parametri za različita okruženja kanala.

4. ANALIZA INTERFERENCIJE U VAZDUHOPLOVNIM KANALIMA

Vazduhoplovni kanal predstavlja izazov za implementaciju MC sistema. Četiri različita scenarija kanala se mogu definisati: na putu, polijetanje i slijetanje, kretanje po pisti i parking [8]. Ovi scenariji su okarakterisani različitim vrijednostima fading-a, Doplerovog spread-a i kašnjenja. Zbog osjetljivosti na Doplerov spread OFDM nije optimalan za vazduhoplovni kanal. Nasuprot, AFT-MC je u mogućnosti da se prilagodi promjenama tokom različitih faza leta i suzbije interferenciju prouzrokovanu Doplerovim spread-om sa istom spektralnom efikasnošću kao OFDM. U ovom radu će biti predstavljena analiza prva dva (najznačajnija) scenarija leta. U svim scenarijima, smo uzeli frekvenciju nosioca $f_c = 1.55$ GHz i različita trajanja simbola $T = \{112\mu s, 224\mu s, 448\mu s, 896\mu s\}$. Maksimalni Doplerov pomjeraj se mijenja u zavisnosti od brzine letilice $v_d = (v/c)f_c$. Ostali parametri kanala su uzeti iz [8]. Sve snage interferencije su proračunate korišćenjem (16).

A: Scenario na-putu

Scenario na-putu opisuje zemlja-vazduh ili vazduh-vazduh komunikaciju. To je je model kanala sa LOS putanjom i grupom rasutih putanja. U ovom slučaju, $S(\tau, \nu)$ ima oblik

$$S(\tau, \nu) = \frac{K}{K+1} \delta(\tau) \delta(\nu - \nu_{LOS}) + \frac{1}{K+1} P_{diff}(\nu) \delta(\tau - \tau_{diff}), \quad (18)$$

gdje $P_{diff}(\nu)$ predstavlja Doplerov profil snage (DPP) i τ_{diff} označava vremensko kašnjenje rasijanih komponenti. Tipična maksimalna brzina za zemlja-vazduh komunikaciju je $v_{\max} = 440$ m/s i $v_{\max} = 620$ m/s za vazduh-vazduh komunikaciju. U ovom scenariju rasijane komponente nisu uniformno raspoređene u intervalu $[0, 2\pi)$. Širina snopa rasijanih komponenti je uzeta oko 3.5° [8]. $P_{diff}(\nu)$ se može modelovati pomoću ograničenog Jake-ovog modela [9]

$$P_{diff}(\nu) = \psi \frac{1}{v_d \sqrt{1 - \left(\frac{\nu}{v_d}\right)^2}}, \quad \nu_1 \leq \nu \leq \nu_2 \quad (19)$$

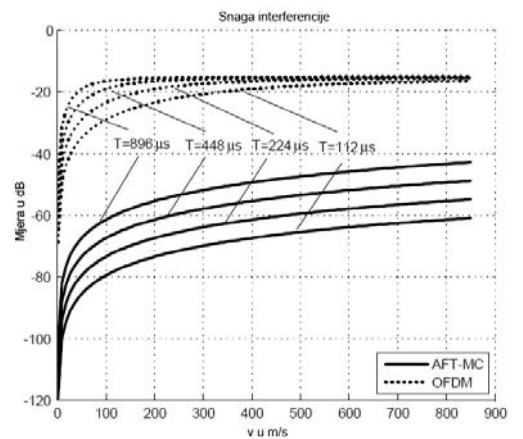
gdje $\psi = 1/(\arcsin(\nu_2/v_d) - \arcsin(\nu_1/v_d))$, označava faktor uveden da normalizuje DPP. Za ovaj kanal optimalni parametri su

$$c_{0opt} = \nu_{LOS},$$

$$c_{1opt} = \frac{1}{2\tau_{diff}} \psi \left(\sqrt{\nu_d^2 - \nu_1^2} - \sqrt{\nu_d^2 - \nu_2^2} \right),$$

$$-\frac{1}{2\tau_{diff}} \nu_{LOS}. \quad (20)$$

U najgorem slučaju LOS komponenta dolazi direktno sa prednje strane letilice i rasijana komponenta dolazi sa zadnje strane. Ovdje je $\nu_1 = -\nu_d$ i $\nu_2 = -\nu_d(1 - \Delta\phi_B/\pi)$, gdje $\Delta\phi_B$ predstavlja širinu snopa rasijanih komponenti simetrično raspoređenih oko $\phi = \pi$. Slika 1 prikazuje poređenje AFT-MC i OFDM sistema sa GI za scenario na-putu za različito T . Jasno se vidi da je AFT-MC značajno bolji od OFDM-a u suzbijanju interferencije. Sa slike 1 se vidi da AFT-MC efikasno suzbija interferenciju čak i za veliki Doplerov pomjeraj. Parametri za ovaj scenario su $\Delta\phi_B = 3.5^\circ$, $\tau_{diff} = 66 \mu s$ i Rice-ov factor $K = 15$ dB [8].



Slika 1. Poređenje snage interferencije u scenariju na putu za AFT-MC i OFDM sa zaštitnim intervalom, za različito T

B : Scenario stizanje i poletanje

Ovaj scenario podrazumjeva model komunikacije kada avion polijeće ili je blizu zemljine površine. Pretpostavljeno je LOS i rasijane komponente stižu direktno sa prednje strane aviona i da je širina snopa rasijanih komponenti od prepreka na aerodromu 180°. Maksimalna brzina aviona je 150 m/s i Rice-ov faktor je $K=15$ dB [8]. U ovom kanalu $S(\tau, \nu)$ je separabilna i predstavljena sa

$$S(\tau, \nu) = Q_{diff}(\tau)P_{diff}(\nu), \quad (21)$$

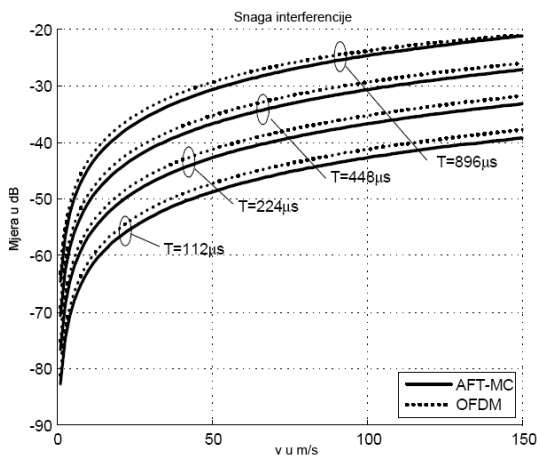
gdje je $Q_{diff}(\tau)$ profil snage kašnjenja (PDP), $P_{diff}(\nu)$ se može modelovati pomoću ograničeno Jake-ovog modela (19), sa $\nu_1=0$ i $\nu_2= \nu_d$. PDP se može modelovati kao eksponencijalna funkcija slično ruralnom nebrdovitom COST 207 modelu [10]

$$Q_{diff}(\tau) = \begin{cases} c_n e^{-\tau/\tau_s}, & 0 \leq \tau \leq \tau_{diff}, \\ 0, & \text{ostalo,} \end{cases} \quad (22)$$

gdje je τ_{diff} maksimalno prekomjerno kašnjenje, τ_s karakteriše nagib funkcije i $c_n = 1/(\tau_s(1 - e^{-\tau_{diff}/\tau_s}))$ predstavlja normalizacioni faktor. Za ruralan nebrdoviti model $\tau_{diff} = 0.7 \mu s$ i $\tau_s = 1/9.2 \mu s$ [10].

Parametri $m_{10}(0,0)$, $m_{20}(0,0)$, $m_{01}(0,0)$, $m_{02}(0,0)$ i $m_{11}(0,0)$ se mogu dobiti kao

$$\begin{aligned} m_{10}(0,0) &= \frac{K}{K+1} \nu_{LOS} + \frac{1}{K+1} \frac{2\nu_d}{\pi}, \\ m_{20}(0,0) &= \frac{K}{K+1} \nu_{LOS}^2 + \frac{1}{K+1} \frac{\nu_d^2}{2}, \\ m_{01}(0,0) &= \frac{1}{K+1} c_n \tau_s \left(\tau_s - e^{-\tau_{diff}/\tau_s} (\tau_{diff} + \tau_s) \right), \\ m_{02}(0,0) &= 2m_{01}(0,0)\tau_s - \frac{1}{K+1} c_n \tau_s e^{-\tau_{diff}/\tau_s} \tau_{diff}^2, \\ m_{11}(0,0) &= \tau_{diff} \left(m_{10}(0,0) - \frac{K}{K+1} \nu_{LOS} \right). \end{aligned} \quad (23)$$



Slika 2. Poređenje snage interferencije u scenariju stizanje i poletanje za AFT-MC i OFDM sa zaštitnim intervalom, za različito T

Na slici 2 su prikazana poređenja snage interferencije u OFDM i AFT-MC sa zaštitnim intervalom. AFT-MC i dalje nadmašuje OFDM, pošto je širina snopa zračenja 180°. Slični se rezultati dobijaju i za scenario kretanja po pisti. U scenariju parking se dobijaju isti rezultati za OFDM i AFT-MC s obzirom da nije prisutna LOS komponenta.

5. ZAKLJUČAK

U ovom radu je predstavljen AFT-MC sistem u vazduhoplovnim vremenski promjenljivim kanalima sa višestrukom propagacijom. Data je gornja i donja granica snage interferencije, praćena aproksimacijom snage interferencije. Optimalni parametri su dobijeni u zatvorenoj formi i praktični primjeri za njihov proračun su dati. Takođe, smo razmatrali aplikaciju u dva različita scenarija: na-putu, dolazak i poletanje.

AFT-MC sistem u poređenju sa OFDM-om značajno poboljšava suzbijanje interferencije u kanalima gdje je prisutna LOS komponenta i uzak snop rasijanih komponenti, koji je tipičan scenarijo u vazduhoplovnim i satelitskim kanalima.

LITERATURA

- [1] T. Gilbert, J. Jin, J. Berger, and S. Henriksen, *NASA/CR-2008-215144: Future Aeronautical Communication Infrastructure Technology Investigation*, Apr. 2008.
- [2] J. A. C. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no.5, pp. 5-14, May 1990.
- [3] Dj. Stojanović, I. Djurović, and B. R. Vojcic, "Interference analysis of multicarrier systems based on affine Fourier transform," *IEEE Trans. Wireless Comm.*, vol. 8, no. 6, pp. 2877-2880, June 2009. [5] P. A. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun. Syst.*, vol. 11, no. 4, pp. 360-393, Dec.1963.
- [4] P. A. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun. Syst.*, vol. 11, no. 4, pp. 360-393, Dec.1963.
- [5] Y. Li and L. Cimini, "Bounds on the interchannel interference of OFDM in time-varying impairments," *IEEE Trans. Commun.*, vol.49, no. 3, pp. 401-404, March 2001.
- [6] W. Kozek and A. F. Molisch, "Nonorthogonal pulses for multicarrier communications in doubly dispersive channels," *IEEE J. Select. Areas In Commun.*, vol. 16, no. 8, pp. 1579-1589, Oct. 1998.
- [7] S. Barbarossa and R. Torti, "Chirped-OFDM for transmissions over time-varying channels with linear delay/Doppler spreading," in *Proc. IEEE ICASSP'01 Salt Lake City*, vol. 4, pp. 2377-2380, May 2001.
- [8] E. Haas, "Aeronautical Channel Modeling," *IEEE Trans. Veh. Technol.*, vol. 51, no. 2, pp. 254-264, March 2002.
- [9] M. Paetzold, *Mobile Fading Channels*, Wiley, New York, 2002.
- [10] M. Falli, Ed., *Digital land mobile radio communications-COST 207: Final report*, Luxembourg: Commission of European Communities, 1989.

PRIMJENA MNK U ODREĐIVANJU DISTRIBUCIJE STRUJNOG TOKA DUŽ KRAKOVA ARHIMEDOVE SPIRALE

APPLICATION OF THE LEAST SQUARE BOUNDARY METHOD IN FINDING OF THE CURRENT DENSITY DISTRIBUTION ALONG THE ARCHIMEDEAN SPIRAL ARMS

Ana Jovanović, Elektrotehnički fakultet u Podgorici

Sadržaj – U ovom radu je demonstrirana primjenljivost MNK u nalaženju raspodjele strujnog toka duž krakova antenske strukture oblika simetrične i ekscentrične Arhimedove spirale. Pokazalo se da se sa malim brojem baznih funkcija veoma dobro zadovoljava granični uslov, što dobijene rezultate čini pouzdanim.

Abstract – This paper demonstrates the applicability of the Least square boundary method (LSBM) in finding of the current density distribution along the Archimedean spiral arms. It appiered that the boundary condition is highly fulfilled by a small number of eigenfunctions, which makes the obtained results very reliable.

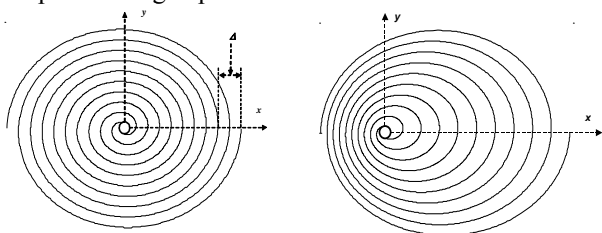
1. UVOD

Za analizu antenskih formi najčešće se koristi metod momenata, čija suština jeste u tome da se antena dijeli na veliki broj malih segmenata sa vrijednostima struje u njima kao nepoznatim veličinama (vidi na pr. [1],[2],[3]) što dovodi do rada sa matricama vrlo visokog reda. U posljednje vrijeme sve više se koristi metod konačnih razlika u vremenskom domenu (vidi na pr. [4],[5],[6]) u kome je jedan od osnovnih problema određivanje vještačke granice na kojoj se završava numerička procedura. Predstavljanje polja antene sistemom sopstvenih funkcija ([7]) dovodi kod analize polja u bliskoj zoni, čak i kod najjednostavnijih struktura do vrlo složene procedure.

U ovom radu će biti demonstrirana mogućnost primjene MNK u analizi antenske strukture oblika simetrične i ekscentrične Arhimedove spirale. Potencijal polja se predstavlja kao linearna kombinacija baznih funkcija talasne jednačine, a nepoznati koeficijenti razvoja se traže zadovoljenjem graničnog uslova na samoj anteni. Ovaj metod, u izvornoj ili modifikovanoj formi, pokazao se veoma prost i tačan na nizu različitih problema (vidi na pr. [8],[9],[10],[11],[12],[13],[14],[15]).

2. OPIS PROBLEMA I PRIMIJENJENOG METODA

Neka je zadata antenska struktura (slika 1.) koja leži u ravni xoy ($\theta = \frac{\pi}{2}$). Antena se napaja izvorom prostoperiodičnog napona U .



Slika 1. Simetrična i ekscentrična Arhimedova spirala

Antenska struktura oblika Arhimedove spirale ima dva kraka čija geometrija je opisana sledećim jednačinama u sfernom koordinatnom sistemu:

$$r = a\varphi\sqrt{2K\cos\varphi + 1 + k^2} + b \quad 0 \leq \varphi \leq 2m\pi \quad \text{gornji krak} \quad (1)$$

$$r' = a(\varphi - \pi)\sqrt{-2K\cos\varphi + 1 + k^2} + b \quad \pi \leq \varphi \leq (2m+1)\pi \quad \text{donji krak}$$

gdje je $a = \frac{\Delta}{2\pi}$ konstanta spirale, b je koordinata početne tačke, k je ekscentričnost spirale.

U najopštijem slučaju polje zračenja, bilo u bliskoj ili dalekoj zoni, ne pripada niti TE ni TM talasu. Da bi našli komponente polja zračenja uvešćemo oba vektorska potencijala, tj magnetni i električni vektor potencijal:

$$A_r = \sum_m \sum_n C_{mn} B_n(kr) P_n^m(\cos\theta) e^{jm\varphi} \quad (2)$$

$$F_r = \sum_m \sum_n D_{mn} B_n(kr) P_n^m(\cos\theta) e^{jm\varphi}$$

gdje su C_{mn} i D_{mn} nepoznati koeficijenti razvoja, $B_n(kr)$ Besselove f-je i $P_n^m(\cos\theta)$ Ležandrovi polinomi.

Komponente polja date su sledećim izrazima [16]:

$$\begin{aligned} E_r &= \frac{1}{j\omega\epsilon} \left(\frac{\partial^2}{\partial r^2} + k^2 \right) A_r & H_r &= \frac{1}{j\omega\mu} \left(\frac{\partial^2}{\partial r^2} + k^2 \right) F_r \\ E_\theta &= \frac{-1}{r\sin\theta} \frac{\partial F_r}{\partial \varphi} + \frac{1}{j\omega\epsilon r} \frac{\partial^2 A_r}{\partial r \partial \theta} & H_\theta &= \frac{1}{r\sin\theta} \frac{\partial A_r}{\partial \varphi} + \frac{1}{j\omega\mu r} \frac{\partial^2 F_r}{\partial r \partial \theta} \\ E_\varphi &= \frac{1}{r} \frac{\partial F_r}{\partial \theta} + \frac{1}{j\omega\epsilon r \sin\theta} \frac{\partial^2 A_r}{\partial r \partial \varphi} & H_\varphi &= -\frac{1}{r} \frac{\partial A_r}{\partial \theta} + \frac{1}{j\omega\mu r \sin\theta} \frac{\partial^2 F_r}{\partial r \partial \varphi} \end{aligned} \quad (2.a)$$

Granični uslov jeste anuliranje tangencijalnih komponenti električnog polja na metalnoj površini krakova antene. Postupkom opisanim u [14] dobijamo nepoznate koeficijente razvoja C_{mn} i D_{mn} , a samim tim i sve komponente polja u bliskoj zoni antene. Opšti izraz za gustinu struje je:

$$\vec{J}_s = \vec{n} \times \vec{H}$$

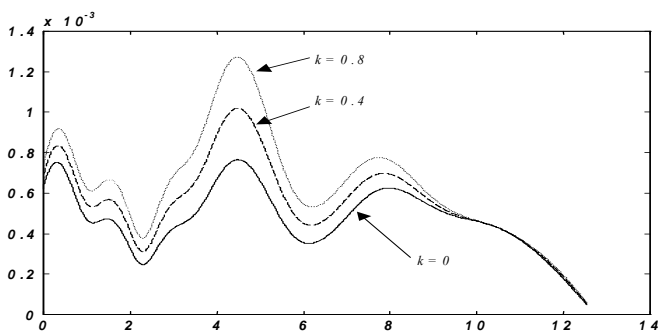
U slučaju opisane antenske strukture izraz za gustinu struje se svodi na sledeći oblik:

$$J = (H_\theta \sin \alpha, H_r \sin \alpha - H_\phi \cos \alpha, H_\theta \cos \alpha)$$

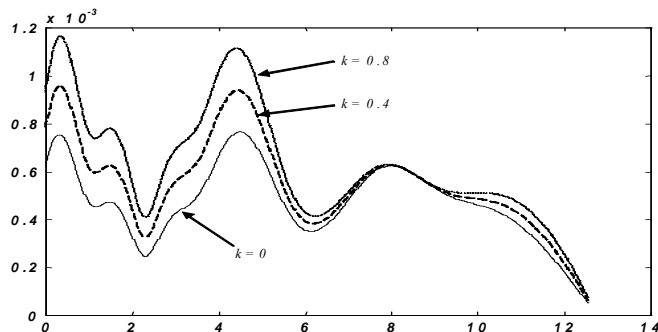
$$\left. \begin{aligned} J_{rt} &= J_r \sin \alpha = H_\theta \sin^2 \alpha \\ J_{\phi t} &= J_\phi \cos \alpha = H_\theta \cos^2 \alpha \end{aligned} \right\} J_t = H_\theta$$

3. NUMERIČKI REZULTATI

Opisanim numeričkim postupkom dobijene su raspodjele struje duž krakova antene. Na slikama (2) i (3) date su dobijene raspodjele struje za tri različite vrijednosti ekscentriciteta spirale i dvije vrijednosti konstante spirale. Električna dužina kraka spirale određena uglom $0 \leq \varphi \leq 4\pi$.

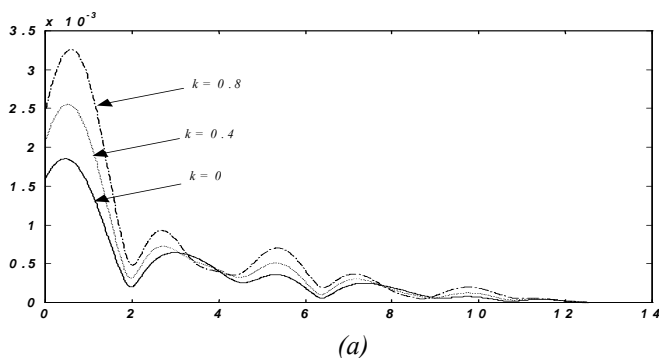


(a)

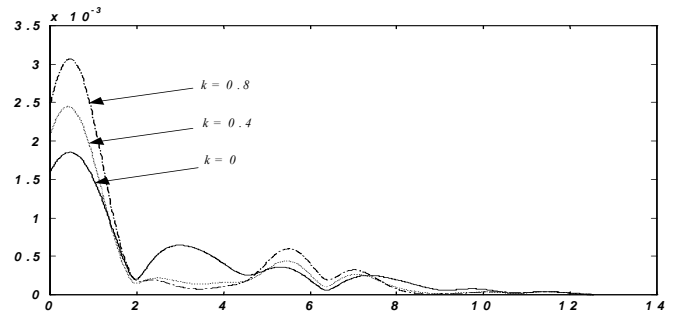


(b)

Slika 2. Raspodjela struje duž (a) gornjeg i (b) donjeg kraka antene ($\Delta = 0.25\lambda$)



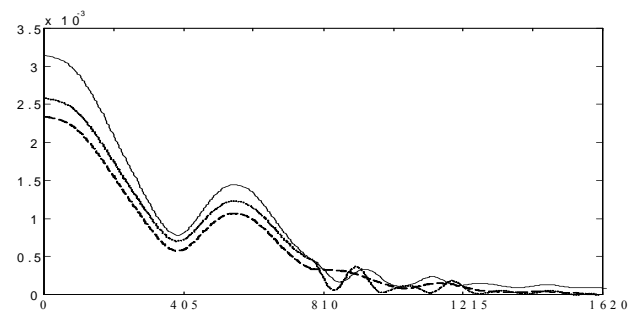
(a)



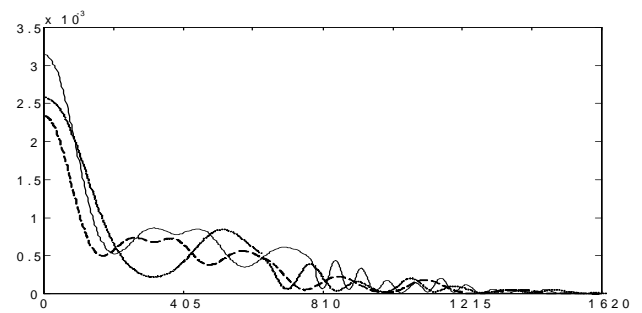
(b)

Slika 3. Raspodjela struje duž (a) gornjeg i (b) donjeg kraka antene ($\Delta = 1.25\lambda$)

Na slici (4) data je, postupkom opisanim u ovom radu, dobijena raspodjela struje duž krakova antene tačno određene geometrije iz [1].



(a)



(b)

Slika 4. Raspodjela struje duž krakova antene, (a) gornji krak, (b) donji krak

(— za $k = 0.8$, za $k = 0.4$, - - - za $k = 0$)

Upoređujući dobijene dijagrame sa dijagramima raspodjele struje iz [1] uočava se dobro slaganje rezultata.

4. ZAKLJUČAK

Demonstrirana je mogućnost upotrebe metoda najmanje kvadratne greške u analizi i dobijanju distribucije strujnog toka duž krakova antenske strukture. Dobijene su raspodjele struje za tri različite vrijednosti ekscentriciteta spirale i dvije vrijednosti konstante spirale. Provjera metoda izvršena je upoređivanjem dobijene distribucije antenske strukture tačno određene geometrije sa dijagramima raspodjele struje iz [1] i uočeno je dobro slaganje rezultata. Veoma dobro

zadovoljenje graničnog uslova na svim djelovima antene postignuto je sa dvadeset članova razvoja.

Na osnovu izloženoga metod se pokazao vrlo tačnim i relativno jednostavnim za korišćenje, što ga preporučuje za analizu antena još složenijeg oblika.

LITERATURA

- [1] R.T.Gloutak, Jr. and N.G.Alexopoulos, "Two-Arm Eccentric Spiral Antenna", *IEEE Trans. on Antennas and Propag.*, vol.45, no.4, april 1997.
- [2] S.K.Khamas and G.G.Cook, "Moment-Method Analysis of Printed Wire Spirals Using Curved Piecewise Sinusoidal Subdomain Basis and Testing Functions", *IEEE Trans. on Antennas and Propag.*, vol.45, no.6, jun 1997.
- [3] H.Nakano, Y.Shinma and J.Yamauchi, "A Monofilar Spiral Antenna and Its Array Above a Ground Plane-Formation of a Circularly Polarized Tilted Fan Beam", *IEEE Trans. on Antennas and Propag.*, vol.45, no.10, october 1997.
- [4] O.M.Ramahi, "Near-and Far-Field Calculations in FDTD Simulations Using Kirchoff Surface Integral Representation", *IEEE Trans. on Antennas and Propag.*, vol.45, no.5, may 1997.
- [5] A.Shlivinski, E.Heyman and R.Kastner, "Antenna Characterization in the Time Domain", *IEEE Trans. on Antennas and Propag.*, vol.45, no.7, july 1997.
- [6] O.M.Ramahi, "The Complementary Operators Method in FDTD Simulations", *IEEE Trans. on Antennas and Propag. magazine*, vol.39, no. 6, december 1997.
- [7] L.W.Li, M.S.Leong, P.S.Kooi and T.S.Yeo, "Exact Solutions of Electromagnetic Fields in Both Near and Far Zones Radiated by Thin Circular-Loop Antennas: A General Representation", *IEEE Trans. on Antennas and Propag.*, vol.45, no.12, december 1997.
- [8] L.J.Stanković, S.Jovičević, "Modified least squares method with application to diffraction and eigenvalue problems", *IEEE Proceedings*, vol.135, Pt.H, no.5, october 1988.
- [9] L.J.Stanković, S.Jovičević, "A round-ridge waveguide", *Ann.Telecommun.*, 43, n^o 9-10, 1988.
- [10] L.J.Stanković, S.Jovičević, "Boundary Condition Expansion of Basis Functions Method Implemented by Fast Fourier Transform Algorithms", *IEEE Trans. on Microwave Theory and Techniques*, vol.38, no.3, march 1990.
- [11] A.Jovanović, S.Jovičević, "A general Solution of the Thin Circular Loop Radiation", *Electromagnetics*, Vol. 23, no.1, january 2003.
- [12] S.Jovičević, A.Jovanović, "The analysis of the biconical antenna by the Least-Squares Boundary Residual Method", *Int. J. Electron. Commun. (AEU)* 57 (2003), no. 6, 415-419.
- [13] A.Jovanović, S.Jovičević, "Analiza žičanih antena metodom najmanjih kvadrata", XLII ETRAN sveska II, Vrnjačka Banja 1998.
- [14] A.Jovanović, S.Jovičević, "Analiza polja simetrične Arhimedove spirale metodom najmanjih kvadrata", 49 konferencija za ETRAN sveska II, Budva jun 2005.
- [15] A.Jovanović, S.Jovičević, "Field Analysis of the Eccentric Archimedean Spiral Antenna by the Least Square Boundary Residual Method", ICEST 2005, Niš jun 2005.
- [16] R.F.Harrington, *Time-Harmonic Electromagnetic Fields*, Mc Graw-Hill Book Company, New York 1961.

METOD EMPIRIJSKE DEKOMPOZICIJE U OBRADI RADARSKIH SIGNALA EMPIRICAL DECOMPOSITION METHOD IN RADAR SIGNAL PROCESSING

Boriša Bjelica, Miloš Daković, Ljubiša Stanković, *Elektrotehnički fakultet – Podgorica*

Sadržaj – U radu su objašnjene osnove algoritma za empirijsku dekompoziciju signala (EMD) i sifting proces. Nakon toga smo analizirali uticaj ivičnih efekata na sifting proces i predložili modifikaciju EMD algoritma u cilju smanjenja njihovog uticaja. Prikazani metod je zatim primijenjen na analizu realnih radarskih signala. Pokazano je da EMD metod može izdvojiti signal mete u slučajevima kada su mu superponirane jake smetnje.

Abstract – In this paper we explained basic Empirical Mode Decomposition (EMD) algorithm and associated sifting process. In the sequel we analyze border effects in the EMD implementation and propose a simple modification for minimizing border effects influence to the sifting process. Presented method is then applied to the analysis of experimental radar signals. It is shown that EMD can extract target signal in heavy clutter environment.

1. UVOD

Empirijska dekompozicija signala je predložena u [1] kao prvi dio procesa za spektralnu analizu nelinearnih i nestacionarnih signala. U prvom dijelu navedenog procesa signal se EMD dekompozicijom razlaže na prirodne sastavne funkcije (IMF) i ostatak, a zatim u drugom dijelu slijedi izračunavanje Hilbertove transformacije IMF-ova. Ovako se dobija energijsko-frekvencijsko-vremenska raspodjela, odnosno Hilbertov spektar. Ovaj dvodijelni proces je nazvan Hilbert Huang-ova transformacija (HHT) i sve više se koristi u raznim oblastima: medicini, klimatologiji, hidrologiji, radarskim i sonarskim sistemima i sl.

Motiv za uvođenje HHT, odnosno EMD bili su nedostaci tradicionalnih oblika spektralne analize, kao što su Fourier-ova i Wavelet, kod analize nestacionarnih i nelinearnih signala. Na primjer pretpostavka Fourier-ove spektralne analize je da se signal može razložiti na niz linearnih, stacionarnih i harmoničnih komponenti. Međutim, kako se povećava stepen nelinearnosti i nestacionarnosti u signalu, tako raste i broj potrebnih harmonika za njegovo predstavljanje. Jer, kao što znamo Fourier-ova spektralna analiza koristi linearnu superpoziciju trigonometrijskih funkcija. Stoga su joj potrebni dodatni harmonici da simuliraju deformisane talasne profile. Ovakve deformacije su direktne posledice nelinearnih efekata. Kad god oblik signala odstupa od potpuno sinusne ili kosinusne funkcije, Fourier-ov spektar će sadržati harmonike. Kao što je objašnjeno, nestacionarnost i nelinearnost mogu proizvesti „lažne“ harmonijske komponente koje prouzrokuju „razlivanje“ energije i nemaju fizički smisao. Posledica ovoga je nepravilna energijsko-frekvencijska distribucija nelinearnih i nestacionarnih signala.

U proučavanju realnih signala HHT metoda se vrlo dobro pokazala jer ne zahtijeva da se unaprijed zna da li je signal linearan ili stacionaran (što je obično nepoznato). Dalje, za HHT nam ne treba ni informacija o izgledu sastavnih funkcija signala, kako bismo mogli odabrati odgovarajuću jer se EMD algoritam adaptira lokalnim uslovima i promjenama unutar posmatranog signala.

HHT je testirana i potvrđena, ali samo empirijski. U mnogim slučajevima dala je jasnije rezultate od ostalih tradicionalnih metoda za vremensko-frekvencijsko-energijski prikaz signala. Iako moćan, ovaj metod je u potpunosti empirijski. Da bi metod bio još bolji i precizniji, potrebno je riješiti mnoge praktične probleme. Problem ivičnih efekata je analiziran u ovom radu.

U ovom radu više pažnje posvećeno je prvom koraku Hilbert Huang-ove transformacije - EMD-u. Kao što smo ranije rekli empirijska metoda dekompozicije je relativno nova metoda za obradu signala.

Dekompozicija se bazira na ekstrakciji energije sa različitim vremenskim skalama, tj. najvažnijim parametrima sistema, izraženim u prirodnim sastavnim funkcijama (IMF), nad kojim kasnije primjenjujemo Hilbertovu transformaciju. Signal se razlaže na IMF komponente jer se iz njih mogu izračunati trenutne frekvencije. Ako znamo trenutnu frekvenciju možemo lokalizovati bilo koji događaj na vremenskoj isto kao i na frekvencijskoj osi. Lokalna energija i trenutna frekvencija izvedena iz prirodnih sastavnih funkcija kroz Hilbertovu transformaciju daju nam punu i preciznu energijsko-frekvencijsko-vremensku raspodjelu signala. Ovo je ključna prednost EMD u odnosu na ostale metode za analizu spektra signala, jer je omogućila analizu nelinearnih i nestacionarnih signala.

2. EMD

Novu nelinearnu tehniku za obradu signala, *Empirical Mode Decomposition* (EMD), predstavio je N.E Huang *et al.*[1] za adaptivno prikazivanje nestacionarnih signala kao sumu AM-FM komponenti sa nultom srednjom vrijednošću. AM-FM komponente su označene kao *Intrinsic Mode Functions* (IMF) odnosno prirodne sastavne funkcije. One moraju zadovoljiti dva uslova:

- (i) svaki IMF mora imati ili isti broj ekstremuma i nula ili se taj broj razlikuje najviše za jedan
- (ii) IMF funkcija mora u bilo kojoj tački biti simetrična u odnosu na nultu srednju vrijednost.

Ovi uslovi moraju biti zadovoljeni kako bi se iz IMF-a kasnije mogla odrediti trenutna frekvencija.

Uvođenje prirodnih sastavnih funkcija je osnovna konceptualna inovacija kod empirijske dekompozicije signala. Sama procedura njihovog dobijanja iz signala naziva se *sifting* („prosijavanje“) i možemo je opisati na sledeći način:

- (1) Naći sve maksimume i minimume posmatranog signala $x(t)$.
- (2) *Cubic spline* interpolacijom odrediti gornju $e_{up}(t)$ i donju anvelopu $e_{low}(t)$.
- (3) Izračunati srednju vrijednost anvelopa signala u svakoj tački $m(t) = (e_{up}(t) + e_{low}(t))/2$.
- (4) Izvući tzv. detalj $d(t) = x(t) - m(t)$.
- (5) Ispitati da li je detalj IMF;
 - (5.1) ako je $d(t)$ IMF, izdvojiti IMF i $x(t)$ zamijeniti ostatkom $r(t) = x(t) - d(t)$,
 - (5.2) ako $d(t)$ nije IMF, treba nastaviti *sifting* proces, i zamijeniti $x(t)$ sa $d(t)$.
- (6) Ponavljati korake (1) – (5) dok signal ne zadovolji neki od kriterijuma zaustavljanja.

Sifting proces ima dva efekta: eliminiše sitna talasanja i poravnava neujednačene amplitude. Eliminacija sitnih talasanja neophodna je da bi trenutna frekvencija imala smisla. Drugi uslov je takođe neophodan u slučaju da postoji velika razlika između susjednih amplituda. Na žalost, ovaj drugi efekat može uništiti fizički smisao amplitudskih fluktuacija. Stoga bi sifting proces trebao biti primjenjivan sa oprezom, jer provođenjem procesa u ekstremno velikom broju koraka može rezultujući IMF da učini čistim frekvencijski modulisanim signalom konstantne amplitude. Da bismo bili sigurni da će IMF zadržati fizički smisao i kod amplitudske i frekvencijske modulacije signala, moramo utvrditi kriterijum za zaustavljanje sifting procesa.

Jedan od načina je ograničavanje veličine standardne devijacije SD koja se izračunava iz dva uzastopna sifting rezultata po formuli:

$$SD = \sum_{t=0}^T \left[\frac{|imf_{(k-1)}(t) - imf_k(t)|^2}{imf_{(k-1)}^2(t)} \right] \quad (1)$$

Proces se zaustavlja kada vrijednost SD padne u opseg između 0.2 i 0.3. Sifting proces može biti zaustavljen i bilo kojim od sledećih kriterijuma: kad komponenta, imf_N , ili ostatak, r_N postane manji od neke ranije definisane vrijednosti ili kada ostatak r_N postane monotona funkcija iz koje se više ne može dobiti IMF, kao i nakon određivanja unaprijed poznatog broja IMF-ova.

Nakon dekompozicije signal možemo predstaviti kao:

$$x(t) = \sum_{i=1}^N imf_i + r_N \quad (2)$$

gdje je N ukupan broj IMF-ova, a r_N ostatak.

3. IVIČNI EFEKTI

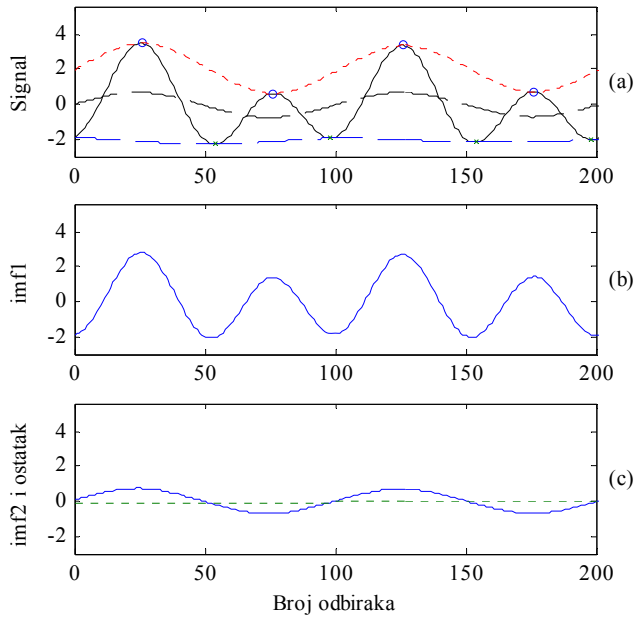
Pored brojnih prednosti EMD ima i nedostataka. Većinom su oni vezani za određivanje anvelopa koje se dobijaju interpolacijom kroz maksimume (minimume) signala. Najbolja, do sada poznata metoda interpolacije je *cubic spline*. Problem je što i kod nje imamo situacije da će anvelopa biti iznad maksimuma, odnosno ispod minimuma, signala što ne bi smjelo da se desi. To izaziva greške prilikom dekompozicije pa ćemo više govoriti o problemu divergencije anvelopa na krajevima signala.

Divergencija se pojavljuje na oba kraja signala, međutim postepeno utiče i na unutrašnje dijelove kroz *sifting* proces, tako da tačnost krajnjih rezultata može biti narušena. Ovi efekti poznati su pod nazivom „ivični efekti“.

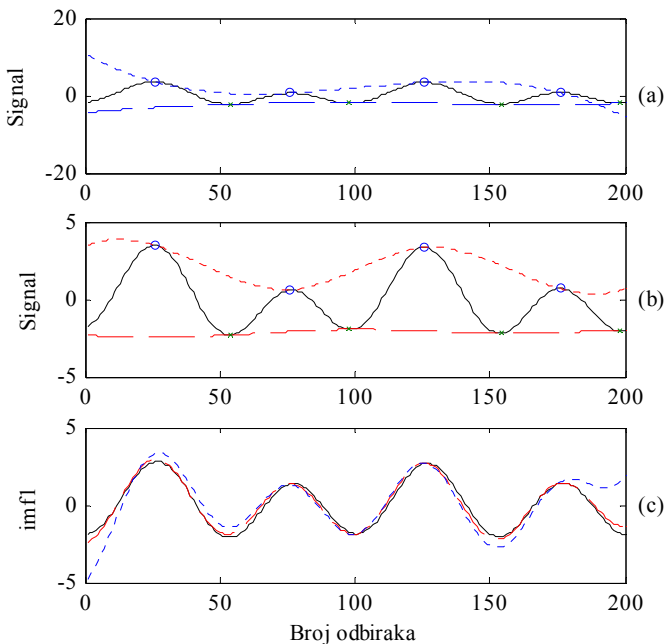
Problem ivičnih efekata počeo je da rješava Huang. On je već iznio efekatan metod produžavanja, koji je patentiran u SAD-u dok je predstavljao EMD metodu, i potrebno je kupiti softver za EMD dekompoziciju za rad sa kratkim signalima, u suprotnom, EMD dekompozicija nema smisla. Huang je naveo da je za produžavanje originalnog signala koristio metodu karakterističnog talasa, ali ne i kako izračunati odgovarajući karakteristični talas.

Za duži signal ovaj se problem jednostavno rešava. Najjednostavniji pristup je da zanemarimo određeni broj odbiraka na početku i na kraju signala, do prvog maksimuma (minimuma) čime izbjegavamo pojavu ivičnih efekata. Signal će biti skraćen, ali uz pretpostavku da podaci na krajevima ne nose neku značajnu informaciju dobijamo tačnu dekompoziciju ostatka signala. Na slici 1 je prikazana EMD signala koji se sastoji od dvije komponente. Posmatrali smo signal koji je analitički te se jednostavno može produžiti van razmatranog intervala tako da se ivični efekti ne pojavljuju.

Ivični efekti su demonstrirani na slici 2(a). Jasno se vidi da imamo četiri maksimuma i tri minimuma, tako da se nekoliko distorzija javlja u gornjoj i donjoj anvelopi signala, naročito na krajevima gornje anvelope. Pošto je signal kratak distorzije sa krajeva anvelope utiču na čitav signal, tako da će rezultat dekompozicije biti nekorektan. Prvi IMF ovog signala označen je na slici 2(c) tačkastom linijom. Vidimo da su odstupanja u amplitudi na krajevima signala velika u odnosu na teorijski dobijeni IMF (koji je na grafiku prikazan punom linijom). U središnjem dijelu razmatranog vremenskog intervala odstupanja su manja, ali i dalje postoje.



Slika 1. Ilustracija empirijske dekompozicije signala: (a) originalni signal (crna puna linija), gornja anvelopa (crvena linija tačka-tačka), donja anvelopa (plava linija tačka-tačka) i srednja vrijednost (crna isprekidana linija); (b) IMF1 (c) IMF2 i ostatak



Slika 2. Ilustracija ivičnih efekata empirijske dekompozicije signala: (a) signal sa anvelopama bez ikakve korekcije na krajevima (b) signal sa anvelopama koje imaju korekciju na krajevima (c) prve imf funkcije za navedene oblike anvelopa i tačna vrijednost prve imf funkcije.

Pojava izobličenja je u ovom slučaju i očekivana jer *cubic spline* funkcija interpolaciju vrši kroz tri susjedna maksimuma (minimuma) posmatrane tačke. Lijevo i desno od krajeva signala ne znamo koji su maksimumi pa se zbog toga javlja ovolika distorzija gornje anvelope. Jedan od načina da se ovaj problem riješi je da se dodaju dva

maksimuma i dva minimuma sa obje strane signala. Time izbjegavamo problem ekstrapolacije podataka, odnosno smanjujemo ivične efekte uzrokovane divergencijom. Problem je kako odrediti prave vrijednosti ovih, na vještački način dodatih ekstremuma?

Posmatrani signal teško je proširiti koristeći metodu karakterističnog talasa, jer unutar njega imamo dva talasna oblika, gdje zavisni oblik varira sa početnom i krajnjom tačkom. Unutar realnih signala često se javlja više talasnih oblika, pa je ovaj metod u takvim slučajevima neefikasan.

Na slici 2(b) vidimo rezultat jednostavne metode produžavanja signala. Princip je sledeći: početna (odnosno krajnja) tačka će biti proglašena maksimumom ako u njoj signal ima veću amplitudu od najbližeg stvarnog maksimuma u signalu, a ako je navedena amplituda manja onda će joj se dodijeliti vrijednost amplitude prvog stvarnog maksimuma u signalu. Slično važi i za donju anvelopu. Predloženi metod nam neće dati potpuno tačan rezultat, ali će ivični efekti biti značajno ublaženi. Signal zajedno sa izračunatim anvelopama je prikazan na slici 2(b) dok je odgovarajući IMF prikazan isprekidanom linijom na slici 2(c).

Ova jednostavna metoda naročito je korisna kada se radi o dužem signalu sa izraženom slučajnošću; rezultat neće biti potpuno tačan, ali ćemo na ovaj način ublažiti veliku distorziju sa krajeva signala i spriječiti da se ona prenese na srednji dio signala koji obično nosi željenu informaciju.

Pored predložene metode, u literaturi su navedeni pristupi zasnovani na produženju signala korištenjem fazi identifikacije ili identifikacije pomoću neuralne mreže. Obje metode mogu dati dobre rezultate za signale koji imaju jasne periode, dok je za signale koji ih nemaju (kao što je npr. površinsko uzdizanje morskih talasa) efikasnije primijeniti metodu sa neuralnom mrežom [2].

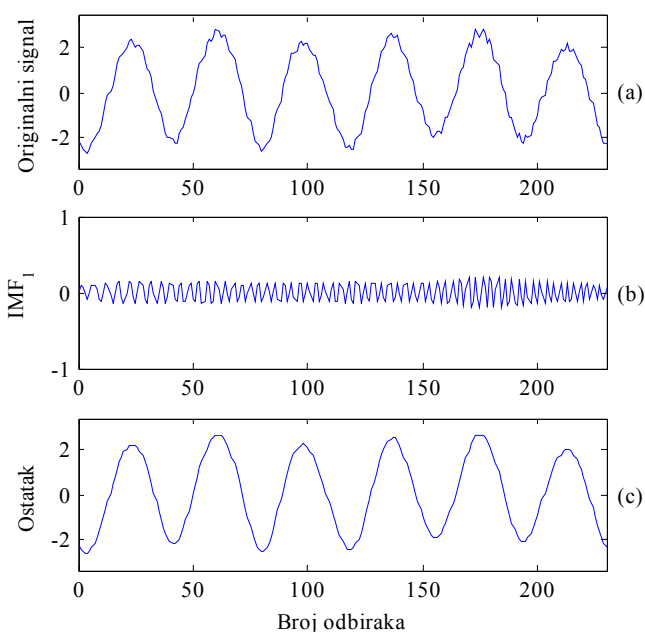
Napomenimo da u najjednostavnijim slučajevima kada posjedujemo dovoljno dugačak signal sama priroda EMD olakšava problem ivičnih efekata: nije potrebno predviđati originalni signal, već samo IMF, koja ima puno uži frekvencijski opseg.

4. EMD U OBRADI RADARSKIH SIGNALA

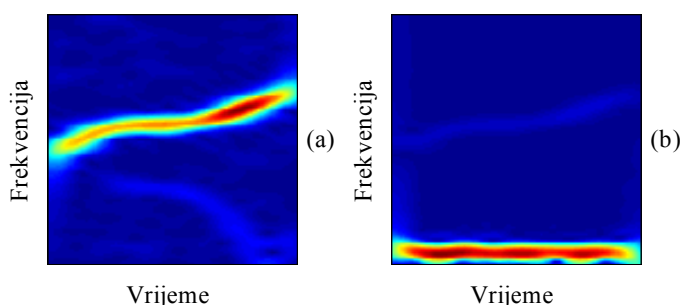
Radarski signali predstavljaju izrazito nestacionarne signale. Rezultati dobijeni klasičnim metodama analize signala, bilo u vremenskom ili frekvencijskom domenu, su u najvećem broju slučajeva nezadovoljavajući ili se mogu značajno poboljšati.

Glavni problem kod analize radarskih signala je detekcija i izdvajanje korisnog signala (signala reflektovanog od mete) u prisustvu jakih smetnji. U [6] i [7] je izložen pristup dekompoziciji signala zasnovan na sopstvenim vektorima vremensko frekvencijske reprezentacije radarskog signala. Jake smetnje (clutter) se javljaju zbog toga što na prijemni dio radara dolaze, pored korisnog signala i signali reflektovani od okolnih objekata. Dakle, osnovni zadatak u obradi radarskih signala je odvajanje korisnog signala od signala smetnji.

Na realnom radarskom signalu primijenit ćemo EMD i pokazati njene mogućnosti u detekciji mete. Posmatra se avion koji izvodi manevre tokom kojih je praćen visokofrekvencijskim radarom sa površinskim talasom (HFSWR), pri čemu je korišćen sistem od deset prijemnih antena u linearnom nizu. Osnovna frekvencija radara je 5,672 MHz a frekvencija ponavljanja impulsa 9,17762 Hz. Dobijeni radarski signal je kompleksan, ima imaginarnu i realnu komponentu. Iako postoji ekstenzija EMD za analizu kompleksnih signala [5], ona je u osnovi zamišljena za rad sa realnim signalima. Zato ćemo je primijenjivati samo na realni dio radarskog signala. Korisni signal dobijamo kao prvu IMF. Na slici 3 prikazan je realni dio analiziranog signala, prva IMF i ostatak dekompozicije.



Slika 3. Empirijska dekompozicija radarskog signala: (a) radarski signal, (b) IMF1, (c) ostatak



Slika 4. (a) VFR prvog IMF-a, (b) VFR radarskog signala

Zbog prisustva šuma ne možemo primijeniti Hilbertovu transformaciju kako bismo saznali više informacija o kretanju aviona (procjena trenutne frekvencija Hilbertovom transformacijom je jako osjetljiva na šum). Umjesto toga, koristit ćemo spektrogram kao vremensko frekvencijsku reprezentaciju signala prikazanu na slici 4.

5. ZAKLJUČAK

Empirijski model dekompozicije signala je efikasan metod u analizi nestacionarnih signala. Pošto je dekompozicija bazirana na lokalnoj karakterističnoj vremenskoj skali signala, može se primjeniti na nelinearnim i nestacionarnim procesima.

U radu smo diskutovali o ivičnim efektima kao jednom od problema u implementaciji EMD analize. Iako oni još uvijek nisu u potpunosti prevaziđeni, EMD je vrlo koristan alat u obradi signala.

Empirijsku dekompoziciju signala smo primijenili na eksperimentalne radarske signale i dobili rezultate koji opravdavaju nastavak istraživanja mogućnosti EMD algoritma.

Osnovni problem u detaljnoj analizi EMD algoritma je što EMD nema dovoljno jaku teorijsku podlogu, već je većina postupaka zasnovana na eksperimentalnim (empirijskim) rezultatima.

LITERATURA

- [1] N. E. Huang, Z. Shen, S. R. Long, et al., „The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis,“ *Proceedings of the Royal Society of A*, vol. 454, no. 1971, pp 903-995, 1998.
- [2] Deng Yongjun, Wang Wei, Quian Chengchun, Wang Zhong and Dai Dejun, „Boundary-processing-technique in EMD method and Hilbert transform“, *Chinese Science Bulletin*, Vol. 46, No. 1, 2001
- [3] N. E. Huang, Samuel S. P. Shen, „Hilbert-Huang Transform and Its Applications“, *Interdisciplinary Mathematical Sciences*, vol.5, 2005.
- [4] Quin Pinle, Lin Yan and Chen Ming, „Empirical Mode Decomposition Method Based on Wavelet with Translation Invariance“, *EURASIP Journal on Advances in Signal Processing*, Volume 2008, Article ID 526038
- [5] T. Tanaka and D. P. Mandić, „Complex Empirical Mode Decomposition,“ *Signal Processing Letters, IEEE*, vol. 14, no. 2, Feb. 2007, pp. 101-104
- [6] LJ. Stankovic, T. Thayaparan, M. Dakovic, "Signal Decomposition by Using the S-Method with Application to the Analysis of HF Radar Signals in Sea-Clutter," *IEEE Trans. on Signal Processing*, vol 54, no 11, Nov. 2006, pp 4332-4342
- [7] LJ. Stankovic, T. Thayaparan, M. Dakovic, "Algorithm for Signal Decomposition by Using The S-Method", *EUSIPCO 2005, Antalya, Turkey, Sept. 2005*

PRIMJENA WBN TEHNOLOGIJE U KONTINUIRANOM PRAĆENJU MEDICINSKIH PARAMETARA

APPLICATION OF THE WBN TECHNOLOGY IN CONTINUOUS MONITORING OF MEDICAL PARAMETERS

Jurica A. Žarković, Anđelina Jokić, Radovan Stojanović i Marija Mirković
Elektrotehnički fakultet, Univerzitet Crne Gore

Sadržaj - U radu se opisuje način upotrebe WBN (Wireless Body Network) sistema za kontinuirano praćenje zdravstvenih parametara, uz primjenu bazične tehnologije koja će nam pomoći u uvođenju proaktivne i pristupačne zdravstvene zaštite. Ona omogućava pojedincu da pažljivo prati promjene svoga zdravstvenog stanja, vitalne funkcije izražene kroz vitalne parametre i pri tome pruži povratne informacije koje će pomoći pri održavanju optimalnog zdravstvenog stanja. Ukoliko su senzori integrisani u telemedicinski sistem, onda takvi sistemi mogu čak i upozoriti medicinsko osoblje ukoliko promjene na pacijentu pokažu ugrožavanje života. WBN sistem sastoji se od minijaturnih senzora, strateški postavljenih duž tijela pacijenta-korisnika, Personalnog servera (PS) koji prikuplja podatke, obrađuje ih i šalje putem WLAN ili GSM mreže i Medicinskog servera (MS) čiji je posao prikupiti obrađene podatke, uskladištiti ih, formirati korisničku bazu podataka, te upozoriti ostale subjekte zdravstvenog sistema (ljekare, medicinsko osoblje, hitnu pomoć).

Abstract - The paper describes a usage of WBN system for continuous monitoring of health parameters, where the applied key technology helps us in the introduction of proactive and accessible health care. It allows individuals to carefully monitor changes in themselves, their vital functions, signs and thereby provide feedback that will help maintain optimal health. If the sensors are integrated in the Telemedicine system, then such systems can even alert medical staff in case the changes are endangering patient's life. WBN system consists of tiny sensors, strategically placed along the body of the patient-user, Personal Server (PS), which collects data, processes them and sends them via WLAN or via the GSM network and of the third part, Medicine Server (MS) whose job is to collect processed data, store them, create user database, and alarm health care system (doctors, medical staff, ambulance).

1. UVOD

Wireless Body Network (WBN) ili bežični mrežni sistem za praćenje zdravstvenih parametara predstavlja novu oblast istraživanja koja je našla svoju primjenu u telemedicini, omogućavajući nam spriječavanje ozbiljnih posljedica pri detekciji abnormalnih dešavanja u ljudskom tijelu, slika 1.

Optimalni uslovi koje pruža WBN tehnologija pri kontinuiranom praćenju životnih funkcija mogu pomoći mnogim pacijentima prilikom praćenja istih nakon hiruške intervencije, oporavka ili pri praćenju toka hronične bolesti.

Značajni nedostaci postojećih sistema za praćenje zdravstvenih parametara su:

- Neprikladni žičani provodnici između senzora i jedinice za obradu;
- Nedostatak sistemske integracije individualnih senzora;
- Interferencija sa ostalim wireless multi-uređajima, frekvencijskim kanalima;
- Nepostojeća podrška za masivno prikupljanje podataka i pristup istim.

Tradicionalno, postojeći sistem za medicinski monitoring tjelesnih funkcija, kao što je Holter monitor, pruža tek kasniju obradu prikupljenih informacija, dakle „off-line processing”. Sistemi sa višestrukim sensorima za tjelesnu rehabilitaciju,

posjeduju veliki broj žičanih provodnika između senzora i jedinice za praćenje. Ove žice mogu uticati na ograničenje pacijentovih aktivnosti, na komfornost i naravno, mogu imati negativan uticaj na izmjerene vrijednosti.

WBN sistem je „nosivi“ sistem, koji se koristi na samom tijelu, doslovce neprimjetan, lagan i potpuno neškodljiv. Može čak biti integrisan i u odjeću ili obuću pacijenta. Koristi Personal Area Network (PAN) mrežu, ili Body Area Network (BAN) mrežu. Novi tehnološki napredak u oblasti fabrikovanja, mikro vlakana, bežičnih komunikacija, integrisanih mikro senzora, omogućio nam je izgradnju WBN senzora - čipa prilagodljivog mnogim aplikacijama i protokolima.

Postojeći telemetrijski uređaji koriste specijalne komunikacione frekvencije za prenos ogromnog - sirovog materijala podataka ili koriste standardne visoko rangirane bežične protokole kao što je Bluetooth, koji su suviše kompleksni, zahtijevaju veliku količinu energije i koji su skloni interferenciji sa drugim bežičnim uređajima koji rade na istim ili sličnim frekvencijama. Karakteristike ovakvih uređaja onemogućavaju njihovo nošenje tokom dužeg vremena. Jednostavan i precizan uređaj, koji bi pratio dnevne aktivnosti jednog čovjeka van laboratorije sa dosadašnjim telemetrijskim sistemom, ne postoji. Pomoću aktuelnih tjelesnih mjerenja, koje sadašnji sistem podržava, otkucaji srca (puls), zasićenost kiseonikom (SpO2), video nadzor pacijenta, pedometar, mjerač brzine trčanja, mogu dati samo estimiranu sliku pacijenta. I na kraju, individualni podaci

pacijenata su veoma rijetko integrisani u istraživačku bazu podataka, koja bi omogućila dalja istraživanja, otkrića relevantnih specifičnih stanja, uslova i kategorisanje pacijenata.

Današnji sistemi za obradu podataka su izrazito sofisticirani, pa kao takvi mogu s lakoćom prihvatiti niz podataka u *live* modu, tj. u realnom vremenu i obraditi ih, koristeći pri tome povratni sistem informisanja. Koristeći ovakav metod, kao rezultat imamo momentalnu reakciju sistema, koja nam pomaže u sprječavanju neželjenih posljedica.

WBN sistem je otišao i korak dalje, pa je kao alternativni izvor napajanja senzora upotrijebljen prirodni izvor energije, koji se temelji na generisanju napona pri prirodnim promjenama temperature, hemo-elektrici, dinamici i slično. Ovakva tehnika je privukla veliku pažnju među istraživačima u oblasti fizikalne medicine i tele-rehabilitacije.

Intenzivni praktični pregledi, definisani po rasporedu, pokazali su se korisnim za oporavak motornih funkcija. Nažalost, nije realno očekivati pristup ovakvoj vrsti rehabilitacije, koja uključuje stalan nadzor fizioterapeuta, u okviru savremenog zdravstvenog sistema u kojem su bolesnici najčešće pregledani dva puta nedjeljno po oko 30-40 minuta.



Slika 1: Primjer biofeedback-a.

„Nosive“ tehnologije, dakle WBN senzori koji se nose na tijelu, kao i *biofeedback*¹ tehnologija (slika 1), daju nam validnu alternativu aktuelnom zdravstvenom sistemu, jer prilično smanjuju vrijeme pripreme pacijenta i zahtijevaju jako malo vremena pri angažovanju ljekara ili terapeuta. Štaviše, WBN tehnologija omogućava postavljanje senzora na duže vrijeme i tako eliminiše potrebu za postavljanjem aparature pri svakom pregledu, liječenju i oporavku. Tako na primjer, lični server, kao što je PDA uređaj, može gotovo odmah da pokrene novu sesiju treninga kad god je pacijent spreman i voljan da joj se podvrgne.

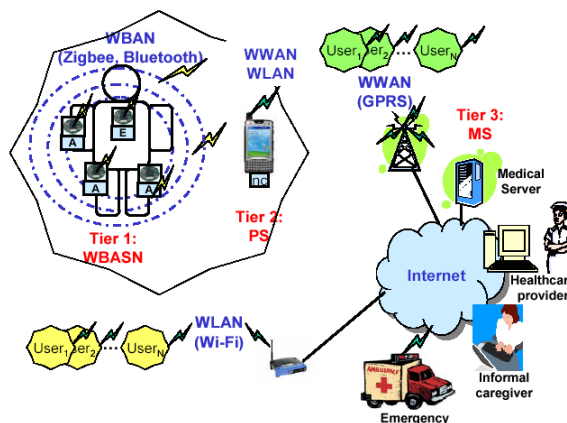
Osim u kućnom liječenju, tehnologija WBN-a takođe može biti korisna u kliničkom liječenju, kako bi se što više sačuvalo dragocijeno vrijeme ljekara i terapeuta. Sistem tako može izdati blagovremeno upozorenje ili alarmirati pacijenta, odnosno specijalizovane medicinske službe u slučaju značajnih odstupanja od norme, tj. potrebe za hitnom medicinskom intervencijom. Međutim, kao i za sve sisteme, redovno, rutinsko održavanje je neophodno.

¹ Biofeedback – informacije koje se prikupljaju na licu mjesta o fiziološkim procesima u tijelu neke osobe.

Tipični primjeri mogućih primjera su rehabilitacija nakon moždanog udara, fizikalna terapija nakon operacije koljena ili kuka, rehabilitacija poslije infarkta miokarda i rehabilitacija nakon traumatske povrede mozga. Procjena efikasnosti rehabilitacionih postupaka je do sada bila ograničena na laboratorijske uslove. Relativno malo se zna o korišćenju WBN sistema u rehabilitaciji u realnim životnim situacijama. Minijaturna, bežična i „nosiva“ tehnologija nudi bezbrojne mogućnosti u okviru ove oblasti.

2. ARHITEKTURA SAVREMENOG WBN SISTEMA

Savremeni prijedlog arhitekture *Wireless Body Network-a* za nadgledanje i kontrolu zdravstvenih parametara pacijenata, za implementaciju u širi sistem telemedicine, ilustrovan je na slici 2. Arhitektura koja je primijenjena u ovom sistemu je jedna od mogućih savremenih rješenja WBN sistema. Sistem se sastoji od više senzorskih čvorova, koji prate kretanje tijela, srčanu aktivnost, temperaturu i slično, kao i od ličnog servera (PS) instaliranog na prenosivom PDA uređaju ili ličnom računaru-notebook-u.



Slika 2: Arhitektura WBN sistema.

Telemedicinski sistem obuhvata mrežu sastavljenu od pojedinačnih zdravstvenih sistema monitoringa, koji se povezuju preko Interneta na medicinski server koji se nalazi na vrhu ove hijerarhije.

Na slici 2: Treći nivo – Tier3 predstavlja centralizovani medicinski server optimizovan da opsluži stotine i hiljade pojedinačnih korisnika, a obuhvata složene mreže međusobno povezanih usluga, medicinsko osoblje i zdravstvene radnike generalno. Svaki korisnik nosi određen broj senzorskih čvorova koji su taktički postavljeni duž tijela. Osnovne funkcije ovih senzora su da nenametljivo prikupe vitalne znake i prenesu relevantne podatke do ličnog servera preko male bežične mreže upotrebom implementiranog protokola *ZigBee* (802.15.4) ili *Bluetooth* (802.15.1).

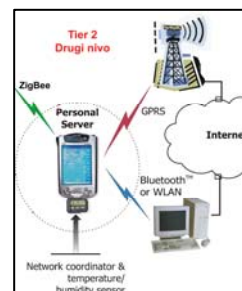
Mrežni protokol *ZigBee* je nastao kada se javila potreba za umrežavanjem velikog broja uređaja između kojih se prenosi mala količina podataka, sa aplikacijama koje zahtijevaju veliku energetska autonomiju uređaja, te samim time i malu potrošnju. Tipični primjeri aplikacija sa ovakvim zahtjevima su WBN senzori, WSN (*Wireless sensor network*) kontrolne mreže, prikupljanje medicinskih podataka i slično. *ZigBee* uređaji bi trebali biti maleni, jeftini i pouzdani. *ZigBee* definiše tri pojasa frekvencija: prvi na 868 MHz koji sadrži kanal 0 i podržava prenos podataka od 20 kbps, drugi

na 915 MHz sa 10 kanala (1-10) i brzinom prenosa od 40 kbps te treći na 2,4 GHz sa 16 kanala (11 – 26) sa brzinom prenosa od 250 kbps i širinom kanala od 5 MHz. Na prva dva pojasa koristi se binarna PSK (*Phase Shift Keying*) modulacija, dok se na trećem pojasu koristi QPSK (*Quadrature Phase Shift Keying*). U frekventijskom opsegu od 2,4 GHz nominalni domet u otvorenom prostoru iznosi 100m, dok u zatvorenom iznosi 30m pri snazi predaje od 1 mW. Radi poboljšanja odnosa signal/šum te povećanja selektivnosti kanala upotrebljava se tehnika modulisanja raspršenja spektra direktnim postupkom (DSSS - *Direct Sequence Spread Spectrum*).

Lični server - (PS), implementiran na personalnom digitalnom asistentu (PDA) ili mobilnom telefonu. On podešava i kontroliše WBN mrežu, pruža grafički ili audio interfejs za korisnika, prenosi informacije o zdravstvenom stanju na medicinski server preko interneta ili preko mobilnih telefonskih mreža (npr. GPRS i 3G). Lični server sadrži autentične podatke pacijenta i konfigurisan je IP adresom medicinskog servera (MS) u cilju komunikacije sa njim. Ako je na raspolaganju kanal komunikacije sa medicinskim serverom, lični server uspostavlja sigurnu komunikaciju i šalje izvještaje koji se upisuju u medicinsku evidenciju korisnika, odnosno pacijenta. Međutim, ukoliko je veza između PS i MS onemogućena ili MS nije dostupan, PS bi trebalo da bude u mogućnosti da sačuva podatke na lokalnom nivou i da inicira slanje podataka pri prvoj novoj konekciji sa MS, tj. čim veza bude uspostavljena. Ovakva organizacija omogućava punu mobilnost korisnika sa sigurnim i u realnom vremenu poslatim i obrađenim zdravstvenim informacijama (*upload*).

Medicinski server – (MS) je skup elektronskih dosijea registrovanih korisnika koji pruža razne usluge korisnicima, medicinskom i pomoćnom osoblju. Odgovornost medicinskog servera je da utvrdi autentičnost korisnika, prihvati podatke zdravstvenog stanja, formira ih i ubaci u odgovarajuću medicinsku evidenciju, analizira uzorke podataka, prepozna ozbiljne zdravstvene anomalije u cilju kontaktiranja donatora, i prosljedi nove instrukcije korisnicima, kao što ljekar propisuje vježbe ili lijekove. Pacijentovim podacima ljekar može da pristupi iz kancelarije putem interneta i ispita stanje pacijenta pomoću telemetrike (da provjeri puls, krvni pritisak, aktivnost...), obezbijedi reagovanje pacijenta na dati tretman, odnosno datu terapiju. Odgovarajući softver instaliran na serveru u vidu serverskog agenta može sam ispitati primljene podatke i kreirati obavještenje u slučaju potencijalnih zdravstvenih problema. Velika količina podataka prikupljena preko ovog sistema može se koristiti za rano otkrivanje akutnih bolesti. Integracija podataka prikupljenih u istraživanju baza podataka i kvantitativne analize uslova i modela, mogla bi biti dragocjena za istraživačke pokušaje da se povežu simptomi i dijagnoze sa istorijskim promjenama u zdravstvenom stanju, fiziološkim podacima ili drugim parametrima (npr. pol, starost, težina). Na sličan način ova infrastruktura može značajno da doprinese praćenju i proučavanju terapijskih efekata lijekova.

Drugi nivo – *Tier2*, predstavlja lični server (PDA) sa WBN interfejsom za senzorske čvorove, koji nudi grafički interfejs i komunicira sa uslugama iz trećeg nivoa (*Tier3*). Lični server se obično svodi na PDA uređaj ili mobilni telefon, mada alternativno može da se pokrene i instalira na PC-u. To je posebno pogodno za praćenje osoba u domu starih. Lični server vrši interfersiranje WBN senzorskih čvorova kroz mrežu kordinatora (NC) i implementira *ZigBee* ili *Bluetooth* povezivanje. Da bi komunicirali sa medicinskim serverom (MS), lični server upotrijebljava mobilnu telefonsku mrežu (2G, GPRS, 3G) ili WLAN kako bi pristupio internetu.



Slika 3: Tier 2

Centralni dio višeslojnog telemedicinskog sistema je zapravo **Prvi nivo** – *Tier1* – bežična, telemetrijska WBN mreža. Ona obuhvata skup inteligentnih čvorova, od kojih je svaki sposoban za detekciju, uzimanje uzorka, komunikaciju i obradu fizioloških signala. Na primjer, EKG senzori se mogu koristiti za praćenje aktivnosti srca, EMG za praćenje aktivnosti mišića, EEG za praćenje električnih moždanih aktivnosti, senzor krvnog pritiska, senzor nagiba, pozicije trupa i senzor disanja za praćenje respiracije, dok se senzori kretanja mogu koristiti za razlikovanje stanja korisnika i procjenu njenog ili njegovog nivoa aktivnosti.

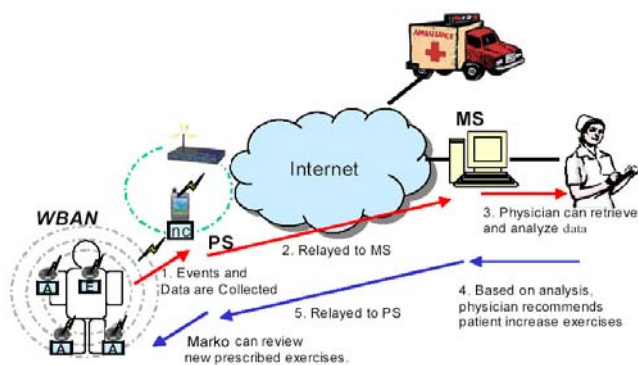
3. STUDIJA SLUČAJA

U ovom dijelu ćemo predstaviti studiju hipotetičkog slučaja kao ilustraciju upotrijebljivosti predloženog WBN sistema. Predstavljeni pacijent je fiktivni, ali i predstavnik zajedničkih problema koji bi zadesili pojedince tokom oporavka od srčanog udara. Razmišljali smo o problemima i opisali WBN sistem kao prevenciju i lijek, te kakve prednosti pruža u odnosu na tipična, današnja rješenja.

Marko Marković se oporavlja od srčanog udara. Nakon otpuštanja iz bolnice, prošao je nadziranu fizičku rehabilitaciju u trajanju od nekoliko nedjelja. Njegovi ljekari propisali su režim vježbanja kod kuće. Tokom fizikalne rehabilitacije bilo je lako pratiti i provjeriti Marka kako je izvodio vježbe. Nažalost, kada je ostavljen da se sam samodisciplinuje i samostalno vježba, on nije rigorozno pratio vježbe kako je to propisano. On vježba, ali ne dovoljno koliko to intezitet i trajanje vježbe iziskuju. Kao rezultat toga, Markov oporavak je sporiji od očekivanog i kao takav podiže zabrinutost za njegovo zdravlje, prognozu, i još gore, njegov ljekar nema kvantitativnu verifikaciju Markovog pridržavanja programa.

Naš WBN zdravstveni sistem monitoringa pruža rješenje za Marka. Opremljen malenim WBN sensorima koji pružaju kontinuirano praćenje vitalne statistike, procjenu energetske gubitaka prilikom vježbanja, pomoći će Marku u praćenju vježbi. Maleni elektronski inercijski senzori mjere pokret,

dok elektrode na prsima mjere Markovu srčanu aktivnost. Vrijeme, trajanje i nivo intenziteta vježbe, mogu se odrediti obradom senzorskih informacija. Putem Interneta, njegov ljekar može prikupiti i pregledati podatke, provjeriti Markovo redovno vježbanje, izdati nove vježbe, prilagoditi podatke i definisati limite vježbi i zakazati novi pregled. Markov ljekar se ne treba oslanjati na Markovo subjektivno mišljenje, već može kontrolisati nivo i trajanje vježbe. Osim toga, Markovi parametri varijabilnosti srčanog ritma, pružaju objektivnu sliku njegove fiziološke reakcije na vježbe, koje služe kao kućni stres test. Daljinskim stres testiranjima i prikupljanjem podataka putem WBN sistema, Markov ljekar smanjuje broj ambulantskih posjeta. To smanjuje zdravstvene troškove i omogućava bolje korištenje ljekarskog vremena. U hitnim slučajevima, lični server može direktno alarmirati hitnu medicinsku pomoć (EMS), ako se pacijent pretplati na ovu uslugu. Slika 4 prikazuje mogući protok podataka:



Slika 4: Studija slucaja - tok podataka

4. ZAKLJUČAK

WBN sistemi koji prate vitalne funkcije, obećavaju sveprisutan ali ipak ekonomsko povoljan zdravstveni monitoring. Vjerujemo da će WBN sistem omogućiti drastičnu promjenu načina na koji ljudi misle o svom zdravlju i brinu o njemu - na isti način na koji je Internet promijenio način komunikacije među ljudima i razmjenu informacija. Ova promjena prema proaktivnijoj preventivnoj zdravstvenoj zaštiti ne samo da će poboljšati kvalitet života, već će takođe smanjiti zdravstvene troškove.

Razvoj bežičnih uređaja i najnovija dostignuća u minijaturizaciji senzora dokazuju tehničku fleksibilnost sveprisutnih zdravstvenih monitoring sistema. Međutim, WBN projektanti se suočavaju sa brojnim izazovima u pokušaju da poboljšaju korisničke zahtjeve koji se odnose na jednostavnost upotrebe, veličinu, pouzdanost i sigurnost. U cilju rješavanja nekih od ovih izazovnih zadataka osmišljen je WBN prototip koji sadrži senzore pokreta sa akcelerometrom, EKG senzor, kao i Pocket PC koji predstavlja lični server-PS. Ova nova tehnologija ima potencijal da ponudi širok spektar povoljnosti za bolesnika, medicinsko osoblje i društvo kroz kontinuirano praćenje u ambulantom liječenju, rano otkrivanje abnormalnih uslova, nadzor rehabilitacije i potencijalna saznanja kroz ekstrakciju podataka prikupljenih informacija.

U ovom radu je opisana opšta WBN arhitektura, važna implementaciona pitanja i WBN prototip, koji se temelji na *off-the-shelf* platformama bežičnih senzora i na posebno dizajniranim EKG senzorima i senzorima pokreta. Naša softverska arhitektura je zasnovana na *TinyOS* sistemu, široko korišćenom *open-source* operativnom sistemu za ugrađene senzorske mreže. Potrebni su dalji napori za poboljšanje QoS (*Quality of Service*) bežične komunikacije, pouzdanosti senzor-čvorova, sigurnosti i standardizacije interfejsa i interoperabilnosti. Osim toga, potrebna su dalja istraživanja različitih medicinskih stanja u kliničkim i ambulantskim uslovima, radi utvrđivanja specifičnih ograničenja i mogućih novih primjena ove tehnologije. Trenutno je u razvoju verzija prototipa za računarsko upravljanu fizičku rehabilitaciju i monitoring zdravstvenog statusa.

ZAHVALNICA

Ovim putem zahvaljujemo se profesoru doktoru Radovanu Stojanoviću, Univerzitet Crne Gore, Elektrotehnički fakultet i doktoru Aleksandru Milenkoviću, docentu na University of Alabama u Huntsville-u, Department of Electrical and Computer Engineering, za njihovu pomoć u realizaciji ovog rada.

Takođe, dio rada opisuje aktivnosti obuhvaćene projektom „Novi koncept u razvoju WHC sistema“ finansiranom od strane Ministarstva za obrazovanje i nauku republike Crne Gore.

LITERATURA

- [1] <http://www.nari.ee.ethz.ch/wireless/>
- [2] <http://faculty.uml.edu/yluo/Teaching/AdvCompArch/reports/WeinbergZhangFinalReport.pdf>
- [3] Bouten, C.V.C., Koekkoek, K.T.M., Verduin, M., Kodde, R., and Janssen, J.D., A Triaxial Accelerometer and Portable Data Processing Unit for the Assessment of Daily Physical Activity. in IEEE Transactions On Biomedical Engineering, 44 (3). March 1997. 136-147.
- [4] CardioLabs, <http://www.cardiolabs.com>, last accessed December 2005.
- [5] CardioNet, <http://www.cardionet.com>, last accessed December 2005.
- [6] Cox, D., Jovanov, E., and Milenkovic, A., Time Synchronization for ZigBee Networks. in
- [7] U.S. Census Bureau, U.S. Interim Projections by Age, Sex, Race, and Hispanic Origin, <http://www.census.gov/ipc/www/usinterimproj/>, accessed in September 2005.
- [8] Warren, S., Lebak, J., Yao, J., Creekmore, J., Milenkovic, A., and Jovanov, E., Interoperability and Security in Wireless Body Area Network Infrastructures. in Proceedings of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, (Shanghai, China, 2005).
- [9] Wheelock, B., Autonomous Real-Time Detection of Silent Ischemia. M.S. thesis, University of Alabama in Huntsville, 1999.
- [10] ZigBee Alliance, <http://www.zigbee.org>, last accessed December 2005.

DETEKCIJA QRS KOMPLEKSA SA ASPEKTA PRIMJENE DISKRETNE WAVELET TRANSFORMACIJE

DETECTION OF QRS COMPLEXES WITH ASPECTS OF DISCRETE WAVELET TRANSFORMATION

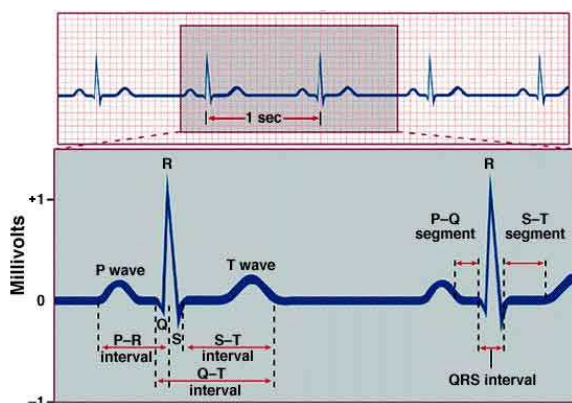
Marija Mirković, Radovan Stojanović, Milorad Kalezić, Budimir Lutovac, Nevena Vukčević
 Elektrotehnički fakultet, Univerzitet Crne Gore, Podgorica

Sadržaj – Cilj rada je provjera tačnosti algoritma za prepoznavanja QRS kompleksa i to sa aspekta primjene Diskretne Wavelet Transformacije (DWT). Precizna identifikacija QRS kompleksa je od velike važnosti za automatsku analizu EKG signala pomoću koje dolazimo do velikog broja informacija o trenutnom zdravstvenom stanju srca. U posljednjih 30-tak godina razvijen je veliki broj algoritama, od koji svaki ima različite prednosti i mane. U ovom radu biće prikazani rezultati do kojih smo došli primjenom algoritma za detekciju baziranog na DWT, koja se pokazala kao pogodan metod za procesiranje EKG signala. Algoritam je testiran na EKG signalima iz standardne MIT-BIH baze.

Abstract – The aim of this work is checking the accuracy of the algorithm for recognition of QRS complexes and from the point of application of Discrete Wavelet Transformation (DWT). Precise identification of QRS complexes is of great importance for the automatic analysis of ECG signals, whose appearance may give us much information about the current state of the heart. Many algorithms have been developed in the last 30 years, each of which has different advantages and disadvantages. This paper presents the results we came upon application detection algorithm based on the DWT, which proved an excellent method for processing ECG signals. The algorithm was tested on ECG signals from MIT-BIH standard database.

1.UVOD

EKG (Elektrokardiogram) predstavlja elektro-fiziološki signal kojim se grafički reprezentuje električna aktivnost srca. To je ponovljiv složenoperiodični signal kod kojeg razlikujemo nekoliko karakterističnih oblika kao što su: P talas, QRS kompleks i T talas. Na slici 1. je prikazan tipični EKG signal.



Slika 1. Tipičan izgled EKG signala

QRS kompleks je najvažniji talasni oblik u okviru EKG signala. Vrijeme njegovog pojavljivanja kao i njegov izgled nam daju mnogo informacija o trenutnom stanju srca. Ostali karakteristični talasi, kao što su P i T talas, su ponekad premali da bi se detektovani, a vrlo često imaju karakteristike koje su slične šumu koji se javlja pri detekciji. Iz tog razloga, upravo QRS kompleks predstavlja osnovu za automatsko određivanje frekvencije rada srca, polaznu tačku za klasifikaciju srčanog ritma kao i polaznu tačku u algoritmima za kompresiju EKG podataka.

Međutim, detekcija QRS kompleksa u EKG signalu je složen problem iz više razloga:

- spektralne karakteristike komponenti signala zavise od fizioloških promjena i od pacijenta,
- ponekad P i T talas mogu imati karakteristike slične karakteristikama QRS kompleksa i
- postoje mnogi izvori šumova u kliničkom okruženju koji degradiraju EKG signal (interferencija mrežnog napajanja, šum usljed kontrakcije mišića - EMG šum, loš kontakt elektroda sa tijelom, pomjeranje elektroda, pokretanje pacijenta, pomjeranje osnovne linije...).

Za uspješno izdvajanje QRS kompleksa razvijen je veliki broj detektora. Postoje mnoge podjele ovih detektora: hardverski (analogni) i softverski (digitalni), online i offline neadaptivni (derivacije, filtri, ...) i adaptivni (adaptivni pragovi, whitening filteri, matched filteri, neuronske mreže, ...) netransformirajući (rade na originalnom signalu) i transformirajući (rade na transformiranom signalu, npr. Wavelet transformacija, Furijeova transformacija, ...) [1].

Većina algoritama za detekciju ima sličnu algoritamsku strukturu koja se može podijeliti u dvije faze: faza predprocesiranja i procesiranja. Faza predprocesiranja obuhvata linearno i nelinearno filtriranje. Druga faza obuhvata detekciju vrha i logiku za odlučivanje da li se radi o QRS kompleksu ili ne. Osim samog izbora algoritma, od velike važnosti su osobine samih klasifikatora. U slučaju EKG signala, te osobine se mogu direktno dobiti iz odabira EKG signala, morfoloških deskriptora, vremensko-frekvencijskih deskriptora ili njihove kombinacije. Morfološke osobine su obično osobine dobijene ekstrakcijom iz segmentisanih EKG pikova, na primjer dužina ST intervala, dok vremensko-frekvencijske osobine mogu biti

koeficijenti Diskretne Wavelet Transformacije (DWT) EKG signala.

2. WAVELET TRANSFORMACIJA

Zbog prirode EKG signala izabrana je Wavelet transformacija kao efikasna metoda za digitalnu obradu signala omogućujući izdvajanje korisnog signala iz zašumljenog i njegovu analizu, kao i njegovo lokalizovanje u frekvencijskom i vremenskom domenu. Osim za detekciju QRS talasa, Wavelet transformacija se primjenjuje i za detekciju P i T talasa, za detekciju i analizu zakasnilih potencijala, proučavanje aritmija i njihovu klasifikaciju, za izdvajanje raznih tipova poremećaja od korisnog signala, za proučavanje varijabilnosti srčanog ritma [4].

Furijeova transformacija daje samo amplitudno-frekvencijska svojstva datog signala, dok je informacija o vremenu izgubljena, zato je ona neprimjenjiva u situacijama kada istovremeno postoji zahtjev za vremenskim i frekvencijskim informacijama, kao što je to slučaj kod detekcije QRS talasa. Da bi se prevladao ovaj problem, uvedena je STFT (Short Time Fourier Transform). Međutim, osnovno ograničenje STFT transformacije je konstantna širina propusnog opsega filtra koja nije primjerena analizi QRS kompleksa, čije se frekvencijske komponente mijenjaju. Takođe, konstantna širina prozora ne odgovara promjenama trajanja QRS kompleksa. Zato wavelet-i predstavljaju idealan alat za analizu EKG signala [1].

Wavelet transformacija je spektralna estimacijska metoda u kojoj se bilo koja funkcija može izraziti kao beskončan niz talasića (engl. wavelet). Signal izražavamo kao linearnu kombinaciju određenog skupa funkcija dobijenih skaliranjem i translacijom osnovne funkcije (engl. mother function). Osnovni (mother) wavelet je funkcija koja ima karakterističan oblik, određenu širinu, odnosno trajanje u vremenskom domenu i odgovarajuću širinu spektra. Iz njega se izvode ostali wavelet-i koji imaju različite širine spektra i različite širine u vremenskom domenu (skalirani su sa odgovarajućim faktorom). U zavisnosti od faktora skaliranja osnovni wavelet se širi ili sabija. Dekompozicijom signala dobijamo wavelet koeficijente. Signal možemo rekonstruirati kao linearnu kombinaciju wavelet funkcija određenih skala. Skale definišu wavelet koeficijenti. Osnovna prednost je vremensko-frekvencijska lokalizacija što znači da je većina energije waveleta ograničena na konačni vremenski interval. Osim toga, wavelet transformacija daje dobru vremensku i slabiju frekvencijsku rezoluciju na visokim frekvencijama, a dobru frekvencijsku i slabiju vremensku rezoluciju na niskim frekvencijama [3]. Ova osobina waveleta-a je jako pogodna za procesiranje EKG signala, kog kojeg su komponente na visokim frekvencijama kratkog trajanja, a na niskim dugog trajanja.

Generalno gledano, postoji nekoliko opštih koraka u pronalaženju wavelet transformacije datog signala:

1. Na samom početku signal se korelacijom poredi sa osnovnom funkcijom.
2. Wavelet se šiftuje u desno i opet se poredi sa osnovnom funkcijom. Taj proces se nastavlja se sve dok čitav signal ne bude pokriven.
3. Skaliranje (sabijanje ili širenje osnovne funkcije) i ponavljanje koraka 1. i 2.

4. Ponavljanje koraka od 1. do 3. za onoliko skala koliko je potrebno.

3. DISKRETNA WAVELET TRANSFORMACIJA I ODABIR OSNOVNE FUNKCIJE

Wavelet transformacija može biti kontinualna (CWT) ili diskretna (DWT), isto kao i Furijeova transformacija. Osnovna formula za CWT je predstavljena formulom:

$$CWT_f(b, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} f(t) \psi\left(\frac{t-b}{a}\right) dt \quad (1)$$

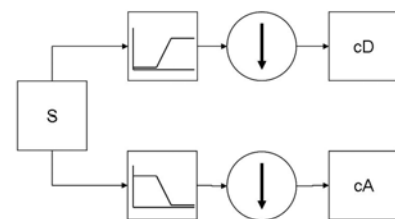
Gdje je a -faktor skaliranja, a b -parametar translacije osnovne (mother) funkcije $\psi(t)$. Mijenjanjem skalirajućeg faktora a pokriveni su različiti frekvencijski opsezi, dok se mijenjanjem faktora šiftovanja b dobija lokalizacija u vremenu. Međutim, osnovni nedostatak CWT je složenost algoritma za izračunavanje. Analiza će biti mnogo efikasnija i jednako tačna ako se diskretizuju parametri a ili b , ili oba. U koliko diskretizujemo samo parametar a i to tako da on uzima vrijednosti 2^j , $j=0,1,2,\dots$ onda CWT postaje DyWT (Dyadic Wavelet Transformacija). Ako diskretizujemo i a i b CWT postaje DWT (Diskretna Wavelet Transformacija):

$$DWT_f(m, n) = \frac{1}{\sqrt{|a_0^m|}} \int_{-\infty}^{\infty} f(t) \psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) dt \quad (2)$$

Gdje sada a_0^m predstavlja skalirajući faktor, a $nb_0 a_0^m$ šiftovanje po vremeskoj osi, $a_0 > 1$, $b_0 > 0$, a m i n su cijeli brojevi. Podrazumijeva se da obje jednačine zadovoljavaju uslov :

$$\int \psi(t) dt = 0 \quad (3)$$

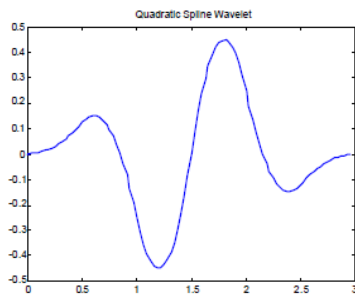
Diskretna wavelet transformacija prevodi signal u niz koeficijenata. Signal se istovremeno provodi kroz niskopropusni i visokopropusni filter. Niskopropusni filter nam daje aproksimacijske koeficijente signala, a visokopropusni detaljne koeficijente signala., slika 2.



Slika 2. Detaljni i aproksimacijski koeficijenti

DWT se može modelirati korišćenjem FIR (Finite Impulse Response) filtera, čiji koeficijenti se mogu izračunati na osnovu izabrane osnovne funkcije. Pravilan izbor osnovne funkcije je jako bitan za detekciju QRS talasa. A jedan od glavnih kriterijuma za odabir je sposobnost mother funkcije da u potpunosti rekonstruiše signal iz wavelet dekompozicije. Zato je u ovom radu izabran *quadratic spline wavelet* (slika 3.) jer se pokazao kao dobar izbor za detekciju QRS talasa. Definisani su funkcijom :

$$\psi(\omega) = j\omega \left(\frac{\sin(\omega/4)}{\omega/4} \right)^4 \quad (4)$$



Slika 3. Quadratic spline wavelet

Filtri koje smo koristili za implementaciju DWT baziranu na quadratic spline wavelet funkciji su $H(\omega)$ i $G(\omega)$ čiji frekventni odzivi su definisani na sljedeći način :

$$H(\omega) = e^{j\omega/2} \left(\cos \frac{\omega}{2} \right)^3 \quad (5)$$

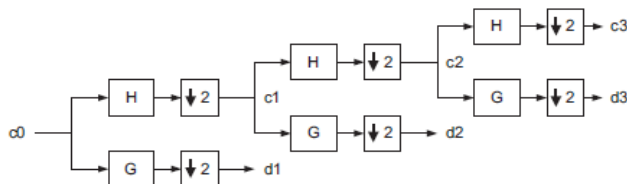
$$G(\omega) = 4ie^{j\omega/2} \left(\sin \frac{\omega}{2} \right) \quad (6)$$

Funkcije prenosa ovih filtara su :

$$H(z) = \frac{1}{8}z^2 + \frac{3}{8}z + \frac{1}{8}z^{-1} \quad (7)$$

$$G(z) = 2z - 2 \quad (8)$$

Jednačine (7) i (8) daju koeficijente za dva FIR filtra koja su potrebna za implementaciju DWT. U cilju dobijanja različitih skala DWT potrebno je ostvariti kaskadnu vezu ovih filtara kao što je predstavljeno na slici 4.

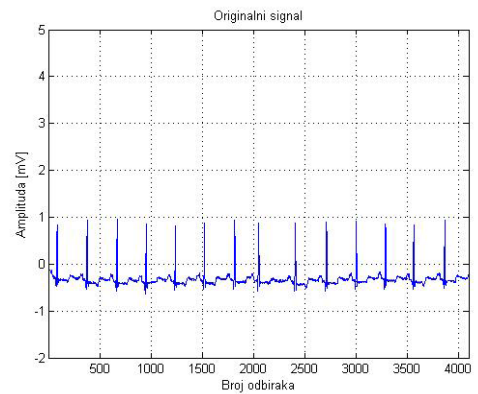


Slika 4. Generalna šema praktične realizacije DWT.

4. REZULTATI ANALIZE

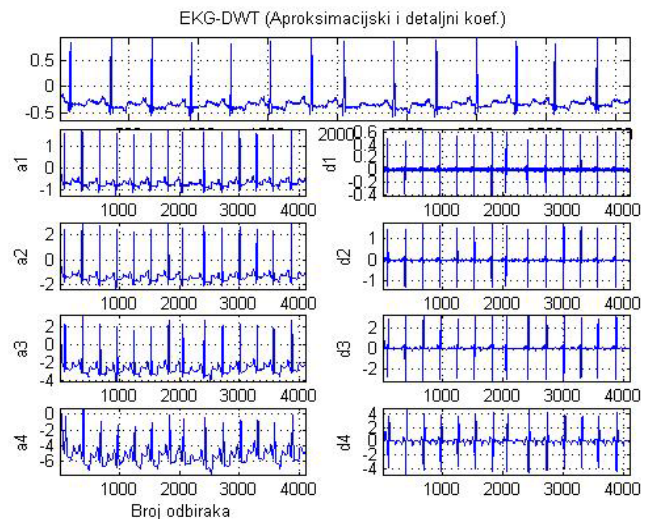
EKG signal koji smo koristili za analizu uzet je iz MIT-BIH standardne baze [6]. MIT-BIH baza raspolaže sa 10 baza EKG signala za različita testiranja. Najčešće se koristi MIT-BIH baza za aritmiju koja se sastoji od 48 polučasovnih označenih EKG signala. Ovo znači da su lokacije QRS kompleksa označene i da se tačno može vidjeti gdje se nalaze. Za provjeru našeg algoritma koristili smo signale prikazane u tabeli 1. Na slikama 6, 7, 8 prikazani rezultati za EKG signal Rec. No 100. Frekvencija odabiranja korišćena u analizi je 360Hz. Amplitude svih signala su u opsegu od ± 5 mV i one su predstavljene sa 11 bita.

Kompletan program je napisan u MATLAB-u. Prikazani su rezultati za jedan segment EKG signala u trajanju od 1,37s (4096 odbiraka).



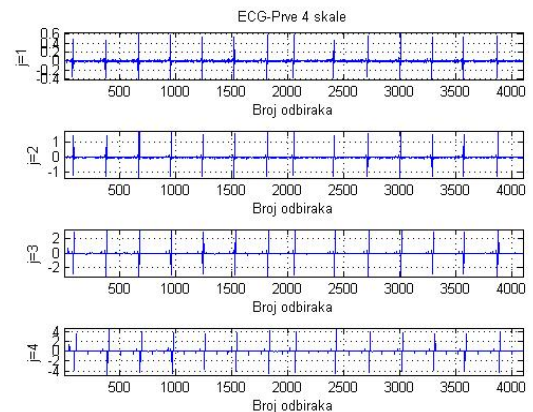
Slika 5. Originalni EKG signal

Na EKG signal sa slike 5. primijenili smo DWT za 4 skale. Dekompozicija signala je prikazana slici 6.



Slika 6. Dekompozicija EKG signala korišćenjem quadratic spline wavelet-a.

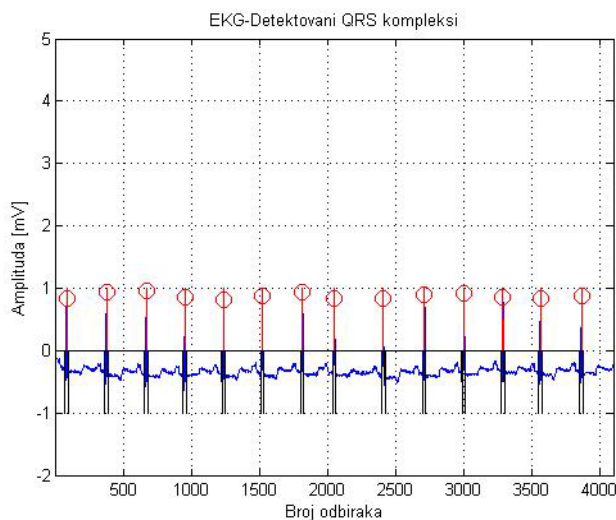
Dekompozicijom signala wavelet transformacijom dobili smo detalje signala različitih skala (na slici 6. označeni kao d1, d2, d3, d4), za skale $2^1, 2^2, 2^3, 2^4$ respektivno. Izdvojeni detalji su prikazani na slici 7. Može se primijetiti da su smetnje (kao što su interferencija mrežnog napajanja, šum usljed kontrakcije mišića) uglavnom na skalama d1 i d2 (slika 7.). Tako da je sva energija QRS kompleksa uglavnom na skalama d3 i d4, pa da smo samo njih odabrali za detekciju R pika.



Slika 7. Prve četiri skale DWT.

Nakon uklanjanja šuma i izračunavanja DWT koeficijenta vrši se ekstrakcija samog QRS kompleksa. Prolazak kroz nulu (zero crossing) Wavelet transformacije omogućuje lociranje oštrih promjena u signalu. Početak i kraj QRS kompleksa se određuju detekcijom para modula maksimuma pojedinih skala Wavelet transformacije. Prolazak kroz nulu između dva modula maksimuma odgovara R pik. R pikovi imaju najveću amplitudu, što veoma olakšava detekciju. Njegova detekcija izvršena je poređenjem koeficijenta DWT za određene skale sa fiksnim thresholdom. U principu, traženje vrha svodi se na pronalaženje lokalnog maksimuma s' tim da postoje određena pravila kada je lokalni maksimum vrh. Novi vrh je moguće detektovati samo onda kada nivo signala padne na polovinu prethodnog vrha. Kada smo pronašli vrh, svi lokalni maksimumi na koje naiđemo nisu vrhovi jer signal nije pao ispod polovine vrijednosti prethodnog vrha [5].

Postoji još jedno olakšanje prilikom same detekcije pika, a to je da nakon što se detektuje QRS kompleks, mora proći 200ms dok se sljedeći QRS kompleks može detektovati. Ovo ograničenje dolazi od samog srca, jer to vrijeme predstavlja najmanji mogući razmak između dva otkucaja. Stanje kada srce ima između 140 i 250 otkucaja u minuti, odnosno između 2,3 i 4,1 otkucaja u sekundi se zove supraventrikularna tahikardija. Fiziološki gledano, slučajevi kada srce kuca 4,1 put u sekundi se veoma rijetki tako da slučajevi kada srce ima 5 otkucaja u sekundi su nemogući iz čega možemo zaključiti da nakon što detektujemo QRS kompleks u sljedećih 200 ms sigurno neće biti novih QRS kompleksa. Na ovaj način sprječavamo lažnu detekciju. Na slici 8. prikazani su detektovani R talasi.



Slika 8. Detektovani QRS kompleksi

U Tabeli 1. je data statistika za uspješnost detekcije, tj. broj lažno pozitivnih (FP) i lažno negativnih (FN) QRS kompleksa na dužini signala od 30 minuta kada je frekvencija odabiranja 360Hz, odnosno širina prozora 54 odbirka. Za QRS kompleks kažemo da je lažno pozitivan kada je detektovan QRS kompleks na mjestu gdje ga nema, a lažno negativan kada QRS kompleks nije detektovan, a trebao bi biti. Pokazalo se da algoritam odlično detektuje QRS komplekse kod signala koji nemaju brze promjene srčanog ritma, anomalija u obliku QRS kompleksa kao ni nagle promjene njegove amplitude. Primjer signala koji sadrže

ovakve anomalije su signali 203 i 207 gdje nije moguće napraviti ispravnu detekciju svih QRS kompleksa.

Rec. No	Ukupan br. QRS kom.	FP	FN	Uspješnost detekcije(%)
100	2261	0	0	100
101	1865	2	1	99.84
102	2181	0	0	100
103	2082	0	4	99.80
104	2300	9	11	99.13
203	2980	25	57	97.25
207	2372	38	26	97.30

Tabela 1. Rezultati detekcije QRS kompleksa korišćenjem diskretne wavelet transformacije na signalima iz standardne MIT-BIH baze.

5. ZAKLJUČAK

U radu je predstavljen algoritam za detekciju QRS talasa baziran na diskretnoj wavelet transformaciji. Ova metoda je izabrana jer ima mogućnost lokalizovanja karakterističnih promjena nestacionarnih signala bez gubljenja informacije o vremenu i frekvenciji. Kompletan program je napisan u programskom paketu MATLAB i testiran na EKG signalima iz standardne MIT-BIH baze. Rezultati su pokazali da je greška u detekciji QRS kompleksa primjenom diskretne wavelet transformacije jako mala, odnosno da se uspio dostići visok procenat uspješne detekcije. Međutim, artefakti i patološki signali još uvijek predstavljaju problem.

ZAHVALNOST: Dio istraživanja u ovom radu je realizovan uz pomoć projekta „Novi koncept u razvoju WHC sistema“, finansiranom od strane Ministarstva prosvjete i nauke Crne Gore.

LITERATURA

- [1] A. Samachisa, "Investigating the Effects of an On Chip Pre-Classifier on Wireless ECG Monitoring", *MS Thesis, Rochester Institute of Technology Rochester, New York*, August 2007.
- [2] R. Stojanovic, Bogdan Asanin and Dejan Karadaglic, "An effective teaching tool for biomedical engineering education", *2nd International Conference on e-Health and Bioengineering - EHB 2009*, Iasi-Constanta, Romania, 2009.
- [3] I. Daubechies. "Ten lectures on wavelets", Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
- [4] B-U. Kohler, C. Hennig, and R. Orglmeister, "The Principles of Software QRS Detection," *IEEE Engineering in Medicine and Biology Magazine*, vol. 21, no. 1, pp. 42-57, January 2002
- [5] P. S. Hamilton, "Open Source ECG Analysis Software Documentation", E. P. Limited, Somerville, Mass, USA, 2002.
- [6] MIT-BIH Arrhythmia Database, www.physionet.org.

PRIMJENA MPI ZA ODREĐIVANJE BIASA I VARIJANSE KUBIČNE FAZNE FUNKCIJE

MPI APPLICATION TO EVALUATION OF BIAS AND VARIANCE OF CUBIC PHASE FUNCTION

Marko Simeunović, Miloš Daković i Igor Đurović, *Univerzitet Crne Gore, Elektrotehnički fakultet Podgorica*

Sadržaj – U ovom radu predložen je algoritam za realizaciju paralelnog programa za određivanje biasa i varijanse kubične fazne funkcije. Korišćen je MPI set funkcija za komunikaciju između procesa. Rezultati dobijeni paralelnim programom potvrđuju već dobijene teorijske rezultate. Ilustrovana je zavisnost brzine izvršavanja programa od broja procesa. Program je izvršavan na lokalnom klasteru.

Abstract – In this paper we proposed an algorithm for bias and variance parallel computing of Cubic Phase Function. Communication between processes is done by MPI set of functions. The output of the program verifies already obtained results. Dependences of the computational time to number of processes is also given. The program is run on the local cluster.

1. UVOD

Intezivan razvoj elektronike posljednjih nekoliko decenija doprinjeo je pojavi relativno jeftinih računara velike procesorske moći. Međutim, sa napretkom tehnike povećali su se i prohtijevi korisnika, tako da postoje poslovi koje sadašnji PC ne mogu izvršiti u predviđenom vremenskom roku. Kupovina “super računara” još uvijek zahtijeva ogromna novčana sredstva, koja mogu izdvojiti samo veliki računarski centri. Umjesto toga, pribjegli su umrežavanju postojećih PC u tzv. klastere [1], koji po svojim osobinama ni u čemu ne zaostaju u odnosu na pomenute. Cijena “moćnijih” klastera je i po nekoliko desetina puta manja u odnosu na “super računare”, čineći ih na taj način dostupnim i nešto “siromašnijim” korisnicima. Kako se od aplikacija u takvim sistemima zahtijeva istovremeno izvršavanje na više procesora, logika u pisanju takvih programa se razlikuje u odnosu na logiku pisanja programa za jednoprocesorske sisteme. Da bi podržali zahtijevne višeprocorskih aplikacija (aplikacija koje se izvršavaju na više procesora istovremeno), programeri su predložili različite načine komunikacije aplikacije sa procesorima. Kao rezultat toga, na tržištu su se pojavili različiti tipovi interfejsa za tu namijenu. Još uvijek nijedan od njih nije standardizovan, ali se kao kandidat ispred svih izdvaja Message Passing Interface (MPI) [2]. Tako da MPI danas predstavlja takoreći defakto standard u oblasti paralelnog programiranja.

Naučnici često imaju potrebu da dobijene, teorijske rezultate potvrde u praksi. Kako je u najvećem broju slučajeva praktična realizacija predloženih metoda vrlo teška, a i ekonomski neisplativa, pribjegava se računarskim simulacijama. One, pored niske cijene, nude i mnogo veću fleksibilnost u odnosu na fizičku realizaciju, jer bilo kakva izmjena nad realizovanim sistemom se svodi na mijenjanje programskog koda. Međutim, u cilju vjerodostojnije potvrde, pojedine metode zahtijevaju veliki broj računskih operacija čije izvršavanje na jednom računaru može dugo potrajati. Zbog relativno niske cijene današnjih računara, podjela posla između više računara se nameće kao logično rješenje. Tu na scenu stupa MPI koji treba da omoguću komunikaciju između procesora.

U ovom radu pažnja će biti usmjerena na primjeni MPI u cilju ubrzavanja postupka za određivanje biasa i varijanse kubične fazne funkcije. Pokazaćemo na koji način se vrlo efikasno mogu potvrditi dobijeni teorijski rezultati. Radi ilustracije prednosti višeprocorskih aplikacija prikazaćemo odnos vremena izvršavanja programa za različite brojeve procesa.

Rad je organizovan na sljedeći način. U Sekciji II dati su osnovni principi MPI. Sekcija III govori o kubičnoj faznoj funkciji (CPF), dok je u Sekciji IV predložen način na koji se može realizovati paralelni program za određivanje biasa i varijanse CPF. Zaključak sumira dobijene rezultate.

2. OSNOVE MPI

MPI predstavlja grupu funkcija objedinjenih u okviru programske biblioteke koje omogućavaju komunikaciju između procesa, bilo u okviru jednog računara ili više njih. Nastao je kao rezultat rada grupe od preko 80 ljudi iz vodećih svjetskih hardverskih i softverskih kuća. U suštini ovaj standard nije ništa drugo nego pokušaj objedinjavanja svih dobrih osobina velikog broja interfejsa za podršku paralelnom programiranju koji su postojali prije 1994. godine, kao i njihovog poboljšavanja.

Osnovna ideja ovoga interfejsa je da programer piše samo jedan program i da se u zavisnosti od identifikacionog broja procesa (cijeli broj koji jednoznačno definiše proces u grupi) izvršava određeni dio koda na tom procesu. Radi lakše komunikacije između sebe, procesi su organizovani u određene grupe i međusobno komuniciraju razmjennom poruka. Sadržaj tih poruka obično predstavljaju podaci do kojih se došlo izvršavanjem dijela koda na procesu koji ih šalje, a koji su potrebni procesu koji ih prima. Poruke u okviru jednog procesa razdvojene su preko odgovarajućih tagova, tako da je adresa prijemne strane određena identifikacionim brojem procesa primaoca i tagom.

Za komunikacione svrhe, MPI posjeduje veliki broj funkcija. Neke od njih imaju i mogućnost kombinovanja

primljenih podataka na razne načine, tj. mogućnost njihove manipulacije. U osnovi, od ogromnog skupa implementiranih naredbi, programeri često koriste samo jedan mali dio. To je obično iz razloga što je pamćenje ogromnog seta instrukcija vrlo naporno. Stoga, u MPI najznačajnije naredbe su one koje omogućavaju direktnu komunikaciju, bez manipulacije sa primljenim podacima. Te naredbe se mogu podijeliti na one koje omogućavaju tzv. blokirajuću komunikaciju i one koje omogućavaju neblokirajuću komunikaciju.

Za blokirajuću komunikaciju je karakteristično da, kada program dođe da koraka kada je potrebno poslati podatke drugome procesu, njegovo izvršavanje se prekida sve dok se podaci koji se šalju ne prebace iz predajnog u sistemski bafer. Ti podaci će iz sistemskog bafera biti prosledeni u prijemni bafer drugog procesa tek onda kada se stvore nesmetani uslovi za njihov prenos. Isto tako, izvršavanje programa na prijemnoj strani je blokirano sve dok se u prijemni bafer ne smjeste podaci.

Kod neblokirajuće komunikacije, naredbama za slanje podataka se samo signalizira da predajni bafer sadrži podatke za slanje, ali se ne čeka njihovo premještanje u sistemski bafer. To znači da je sistem spreman za kopiranje podataka iz predajnog bafera. Za vrijeme trajanja slanja ni na koji način se ne smije pristupiti predajnom baferu, jer će to ugroziti ispravnost podataka koji se šalju. S druge strane, kada počne da se izvršava naredba prijema podataka, proces koji prima podatke ne smije pristupiti prijemnom baferu sve dok se prijem ne završi.

Kao prijedlog za standard, MPI je implementiran za C, C++ i Fortran programske jezike [4], [5]. Ova činjenica omogućuje visok stepen prenosivosti između različitih operativnih sistema takvih programa, jer program napisan u ovim programskim jezicima sa lakoćom se može kompajlirati na različitim mašinama. Takođe se pokušalo i sa implementacijom MPI u Matlab-u [6] i Octave-i [7], tako da danas postoji veliki broj različitih MPI toolbox-ova za ove softverske pakete. U budućnosti, prednost za razvoj MPI toolbox-a treba dati Octave-i, jer predstavlja potpuno besplatno softversko rješenje koje podržava veliki broj klastera.

3. VARIJANSA KUBIČNE FAZNE FUNKCIJE

Posmatrajmo polinomijalno fazni signal (PPS) $f(n)$ zahvaćen aditivnim bijelim Gausovim šumom srednje vrijednosti 0 i varijanse σ^2 :

$$\begin{aligned} x(n) &= f(n) + v(n) = A \exp(j\phi(n)) + v(n) \\ &= A \exp(j \sum_{i=0}^p a_i n^i) + v(n), \quad n = -L, \dots, L, \end{aligned} \quad (1)$$

gdje je A amplituda, P red i a_i koeficijenti polinoma faze PPS.

Zbog činjenice da maksimum svoje vrijednosti dostiže u drugom izvodu faze PPS, kubična fazna funkcija [3],

$$C_x(n, \Omega) = \sum_{n=-L}^L x(n + n_1 T) x(n - n_1 T) \exp(-j\Omega(n_1 T)^2), \quad (2)$$

se može iskoristiti za estimaciju parametara faze signala $f(t)$. Kako prisustvo šuma pomjera tačku maksimuma ove funkcije, veoma važno je procijeniti njeno ponašanje u tom slučaju. Ova procijena se može izvršiti sagledavanjem biasa i varijanse CPF:

$$\text{bias} = E\{C_x(n, \Omega)\} - E\{C_f(n, \Omega)\}, \quad (3)$$

$$\text{var} = E\{(C_y(n, \Omega))^2\} - E^2\{C_y(n, \Omega)\}. \quad (4)$$

U radu [8] izvršili smo izvođenje ovih izraza i dobili sljedeće rezultate

$$\text{bias} = 4\sigma^2 A^2 (2L+1) + 2\sigma^4 (2L+1), \quad (5)$$

$$\begin{aligned} \text{var} &= 4\sigma^2 |CD_f(n, \Omega)|^2 [2A^2 N + \sigma^2 N + \\ &8\sigma^2] + 4\sigma^4 N [4A^4 N + \sigma^2 (N+4)(4A^2 + \sigma^2)]. \end{aligned} \quad (6)$$

Kako se radilo o vrlo komplikovanom izvođenju u kome se javljaju sume sa velikim brojem članova, vjerovanoća pojave greške je bila velika. Stoga je neophodno spovesti simulacije kojima bi se potvrdila tačnost dobijenih izraza.

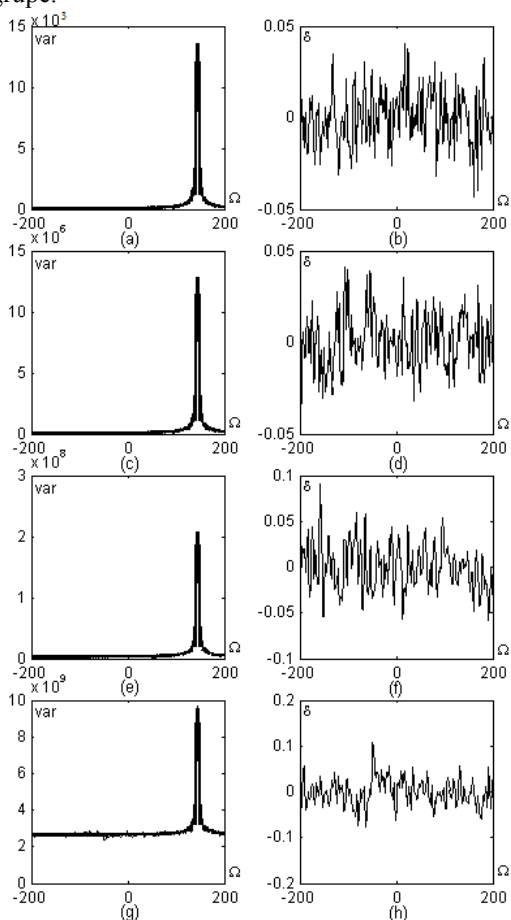
4. PRIMJENA MPI

Da bi konstatovali tačnost izraza (5) i (6) neophodno je sprovesti simulacije nad signalima sa fazom različitog oblika, za različite vrijednosti odnosa signal-šum (SNR) i u različitim vremenskim trenucima. Pored toga, kako je šum slučajne prirode da bi smo mogli uporediti dobijene rezultate sa onima iz (5) i (6), neophodno je cijeli postupak ponoviti veliki broj puta. Tek tada je srednja vrijednost izraza dobijenih u svakoj iteraciji uporediva sa teorijskim rezultatima.

Iz prethodno rečenoga očigledno je da potvrda izraza za bias i varijansu zahtijeva veliki broj računskih operacija koje oduzimaju značajno vrijeme ukoliko bi se sprovodile na računaru sa jednim procesorom. Stoga se, kao rješenje za ubrzanje posla, nameće realizacija simulacije u vidu paralelnog programa koji bi se izvršavao na klasteru.

U ovom radu, mi predlažemo sljedeći način paralelizacije programskog koda. Ukupan broj procesa dodijeljenih našoj aplikaciji potrebno je podijeliti u onoliko grupa koliko ima različitih vrijednosti SNR. Svaka grupa će biti zadužena za određivanje biasa i varijanse CPF za slučaj jedne vrijednosti SNR. Ako sa T_r i N_{SNR} obilježimo broj realizacija procesa sa šumom za jednu vrijednost SNR i broj procesa u grupi, respektivno, svaki proces u okviru grupe treba da izvrši T_r/N_{SNR} iteracija i da usrednjene rezultate pošalje procesu sa najvećom vrijednošću identifikacionog broja (ID) u grupi. Sada, pored izvršavanja svog dijela iteracija, zadatak procesa sa najvećim ID je da prihvati podatke od ostalih procesa u grupi, na odgovarajući način ih usrednji i pošalje konačan izraz za bias i varijansu CPF za sve vremenske trenutke

procesu sa nultim ID. Proces sa nultim ID ne učestvuje u iteracijama, već samo treba da prihvata dobijene rezultate od svake grupe.



Slika 1 - Izračunate i stvarne vrijednosti varijanse CPF: (a) Izračunata (debela linija) i stvarna (tanka linija) vrijednost varijanse za SNR=40dB; (b) Relativna greška za slučaj SNR=40dB; (c) Izračunata (debela linija) i stvarna (tanka linija) vrijednost varijanse za SNR=10dB; (d) Relativna greška za slučaj SNR=10dB; (e) Izračunata (debela linija) i stvarna (tanka linija) vrijednost varijanse za SNR=0dB; (f) Relativna greška za slučaj SNR=0dB; (g) Izračunata (debela linija) i stvarna (tanka linija) vrijednost varijanse za SNR=-10dB; (h) Relativna greška za slučaj SNR=-10dB.

Tačnost izraza (5) i (6) provjerili smo na primjeru PPS drugoga reda:

$$f(t) = \exp(j23\pi t^2), \quad (7)$$

gdje $t \in [-1, 1]$. Korak odabirnja iznosi $\Delta t = 0.0078$, što daje ukupno $N=257$ odbiraka digitalne verzije ovoga signala. Signalu je superponiran aditivni bijeli Gausov šum srednje vrijednosti 0. U obzir smo uzeli 21 različitu vrijednost SNR u rasponu od -20 do 20 dB sa korakom od 2dB. Varijansa za svaku vrijednost SNR računata je pomoću ukupno 10000 iteracija. Simulacija je izvršavana paralelno na više računara. Za paralelni pristup, korišćen je prethodno opisani algoritam. Razmijena poruka između procesa obavljena je MPI. Korišćen je programski jezik C++. Na Slici 1. upoređene su vrijednosti izraza za varijansu CPF ovoga signala za četiri različite vrijednosti SNR i prikazane odgovarajuće relativne greške.

Rezultati prikazani na Slici 1. potvrđuju tačnost izraza (5) i (6). Maksimalna relativna greška za velike vrijednosti SNR

(10dB i 40dB) je oko 5%, dok za male vrijednosti odnosa signal šum (-10dB i 0dB) iznosi oko 9%. Ova greška je posljedica konačnog broja realizacija procesa sa šumom.

Radi ilustracije prednosti paralelnog programiranja u odnosu na standardne jednoprocesorske aplikacije u pogledu brzine izvršavanja, realizovni program smo izvršavali na različitom broju računara i sa različitim brojem procesa. Program je puštan u rad na jedan računar i na lokalnom klasteru. Zbog jednostavnog načina povezivanja računara u klaster, korišćen je Pelikan operativni sistem [9], paralelna verzija Linux-a [10]. U klaster je uključeno ukupno 10 PC sa jednim jezgrom. U Tabeli 1. su prikazani dobijeni rezultati.

r. br.	OS	broj procesora	br. procesa	vrijeme (min)
1	WinXP	1	1	112
2	WinXP	1	22	103
3	Pelikan	2	64	22.45
4	Vista	2	64	37.44
5	Pelikan	5	64	9.52
6	Pelikan	10	64	5.46

Tabela 1 – Vrijeme izvršavanja programa na različitim operativnim sistemima.

Podaci prikazani u Tabeli 1. nedvosmisleno ilustruju prednost paralelnog programa.

5. ZAKLJUČAK

U radu je predložen način za efikasno određivanje biasa i varijanse kubične fazne funkcije. Algoritam je realizovan paralelnim programiranjem. Komunikacija između procesa realizovana je upotrebom message passing interfejsa. Takođe je prikazana zavisnost vremena izvršavanja od broja procesa/procesora.

LITERATURA

- [1] [http://en.wikipedia.org/wiki/Cluster_\(computing\)](http://en.wikipedia.org/wiki/Cluster_(computing)).
- [2] M. Snir, S. Otto, S. Huss-Lederman, D. Walker and J. Dongarra, *MPI: The Complete Reference*, The MIT Press, Cambridge, 1998.
- [3] P. O'Shea, "A fast algorithm for estimating the parameters of a quadratic FM signal", *IEEE Tran. Sig. Proc.*, Vol. 52, No. 2, Feb. 2004, pp. 385-393.
- [4] I. Đurović, S. Đukanović i V. Popović, *Programski jezik C sa zbirkom urađenih zadataka*, ETF Podgorica, 2006.
- [5] N. Larry, *Fortran 77 For Engineers And Scientists With An Introduction To Fortran 90*, Prantice Hall, New York, 1996.
- [6] J. King, *Engineer's Toolkit: MATLAB 5.0 for Engineers*, Addison-Wesley, 1998.

- [7] J. W. Eaton, D. Bateman and S. Hauberg, *Gnu Octave Manual Version 3*, Network Theory, 2008.
- [8] I. Djurović i M. Simeunović, "Izvođenje izraza za bias i varijansu kubične fazne funkcije," *Etran*, Vrnjačka Banja, jun 2009.
- [9] <http://pareto.uab.es/mcreel/ParallelKnoppix/>.
- [10] <http://www.linux.org/>.

ANALIZA UTICAJA ŠESTOPULSNOG I DVANAESTOPULSNOG ISPRAVLJAČA KOD HVDC SISTEMA NA HARMONIJSKA IZOBLIČENJA NAIZMJENIČNE MREŽE ANALYSIS OF THE IMPACT OF SIX-PULSE AND TWELVE-PULSE RECTIFIERS FOR HVDC SYSTEMS TO HARMONIC DISTORSION IN A POWER GRID

Martin Čalasan, Vladan Vujičić, Elektrotehnički fakultet Podgorica

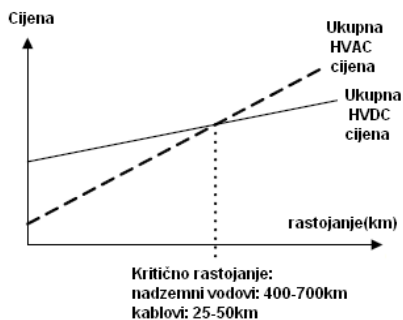
Sadržaj - U ovom radu se analizira uticaj šestopulsnog i dvanaestopulsnog ispravljača kod HVDC sistema na THD napona i struje mreže. Analiza je urađena koristeći razvijene Simulink modele pojedinih sistema. Dobijeni rezultati odgovaraju izvedenim zaključcima iz studije o postavljanju podmorskog kabla između Crne Gore i Italije.

Abstract - This paper deals with the impact of 6-pulse and 12-pulse HVDC rectifier systems to voltage and current THD in the electric power grid. The analysis was performed using the developed Simulink models of individual systems. The results correspond to conclusion derived from studies on setting up underwater cable between Montenegro and Italy.

Keywords: Rectifier - HVDC – Underwater cable – Simulink analysis

1. UVOD

HVDC (High Voltage Direct Current) sistem predstavlja energetske prenosne sisteme jednosmjernog napona. Prenos i distribucija električne energije jednosmjernim naponom počela je krajem 19-og vijeka. Međutim, naizmjenična struja je pružala mnogo efikasniji i jednostavniji način za prenos energije, prije svega zbog mogućnosti veoma lakog podešavanja naponskog nivoa, upotrebom transformatora. U situacijama kada se prenosi energija na velike udaljenosti, ispod vodenih površina, kao i pri spajanju asinhronih sistema, naizmjenična struja se pokazala neefikasna. U takvim slučajevima, HVDC sistem ima veliku prednost. Na slici 1.1 prikazan je dijagram zavisnosti cijene prenosnog sistema od rastojanja, na koje se prenosi energija, kod HVDC sistema i kod naizmjeničnih sistema (HVAC).

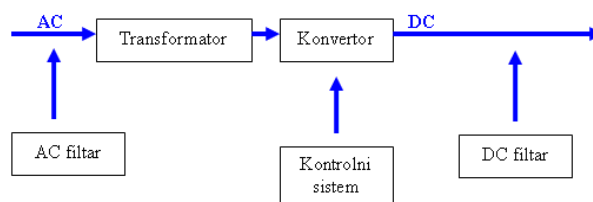


Sl. 1.1 HVAC i HVDC prenos energije

O negativnim stranama naizmjeničnog prenosa energije (gubici, fazni pomjeraji, itd) govori se u referenci [1]. Glavni razlozi upotrebe HVDC sistema i njihove prednosti u odnosu na AC sisteme, prikazane su u referencama [2] i [3].

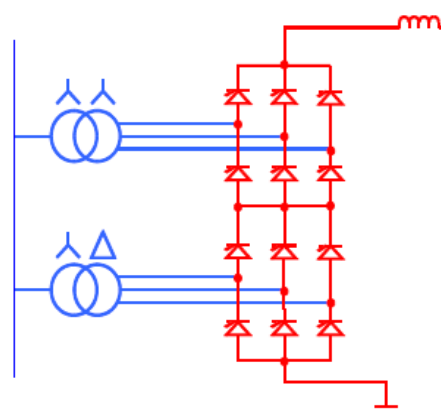
Komponente HVDC sistema prikazane su na slici 1.2. Glavni elementi konvertorske stanice su ispravljači, koji konvertuju AC u DC napon. Transformator ima zadatak da prilagodi naponski nivo AC linije konvertorskoj stanici. Blok DC filtera vrši zaštitu svih elemenata HVDC sistema od naglih i velikih promjena struje i smanjuje struje kratkih spojeva koji mogu nastati u mreži. Naizmjenični filtri imaju zadatak

da apsorbiraju harmonike formirane od strane HVDC sistema i da primaju reaktivnu energiju. Kontrolni sistem ima funkciju zaštite, prije svega konvertorske stanice, ali i funkciju kontrole naponskog nivoa DC sistema.



Sl. 1.2 Najvažniji elementi HVDC sistema

U svijetu je danas konstantan rast prenosa energije preko HVDC sistema, a kakav će biti njihov dizajn zavisi od mnogo parametara, kao što su snaga koja se prenosi, naponski nivo, tip mreže, zahtjevi po pitanju sigurnosti i stabilnosti i slično. U opštem slučaju, HVDC sistemi se dijele na one čija je konvertorska stanica tiristorska i na one čija je konvertorska stanica sa GTO tiristorima, ili sa IGBT tranzistorima [4].



Sl. 1.3 Dvanaestopulsni ispravljač

Šestopulsni tiristorski most vrši ispravljanje naizmjeničnog napona u jednosmjerni. Međutim, taj DC napon nije idealno ispravljen i u njemu su dominantni viši harmonici reda $6n$, dok su na naizmjeničnoj strani posebno izraženi harmonici reda $6n \pm 1$, gdje je $n=1,2,3,\dots$, [5-6]. Zbog

toga je veoma važno, kako na naizmjeničnoj, tako i na DC strani, imati filtre. Danas se najviše koriste dvanaestopulsni ispravljači (1.3), pa su najizraženiji harmonici reda $12n \pm 1$, ali je njihovo ukupno harmonijsko izobličenje (THD – total harmonic distortion) napona i struje značajno manji nego u slučaju upotrebe šestopulsnog ispravljača. Upotreba dvadesetčetiri-pulsnog ispravljača nije praktična, zbog velike cijene konvertorske stanice.

2. HVDC IZMEĐU CRNE GORE i ITALIJE

U bliskoj budućnosti između Crne Gore i Italije planira se postavljanje podmorskog kabla, koji će omogućiti prenos energije između Balkanskog i Apeninskog poluostrva, u slučaju deficita na nekoj od strana. Prema ovom planu, elektroenergetski sistem (EES) Crne Gore biće tranzitni sistem za tokove snaga između okolnih zemalja, prije svega Srbije, Rumunije, Bugarske i Bosne i Hercegovine, sa Italijom. Priključna stanica tog kabla u Crnoj Gori biće u Tivtu.



Sl. 2.1 Planirana HVDC veza između EES-a Crne Gore i Italije

EES Crne Gore je mali, a time i relativno slab sistem, pa pojava viših harmonika, izgradnjom HVDC sistema, može izazvati velike probleme. Pitanju viših harmonika posvećena je posebna pažnja u urađenoj studiji o uticajima postavljanja podmorskog kabla između Crne Gore i Italije. U zaključku ove studije, koja je obuhvatala i zemlje u okruženju, ukazuje se na neophodnost ugradnje aktivnih ili pasivnih filtara, uz napomenu da se posebna pažnja mora posvetiti pravilnom odabiru konvertorskih stanica, [7].

3. UKUPNA HARMONIJSKA IZOBILIČENJA (TOTAL HARMONIC DISTORTION - THD)

Ispravljači služe da isprave naizmjenični napon u jednosmjerni. Međutim, oni vrše negativan uticaj na naizmjeničnu mrežu u smislu izazivanja viših harmonika kako u naizmjeničnoj struji tako i u naponu.

Kao mjera odstupanja struje (napona) od prostoperiodičnog oblika definiše se THD (Total Harmonic Distortion) faktor struje (napona).

$$THD = \sqrt{\frac{I_s^2 - I_{s1}^2}{I_{s1}^2}} = \sqrt{\frac{I_s^2}{I_{s1}^2} - 1} = \sqrt{\frac{\sum_{n=0}^{\infty} I_{sn}^2}{I_{s1}^2} - 1} \quad (1)$$

gdje je:

I_{s1} - efektivna vrijednost osnovnog harmonika

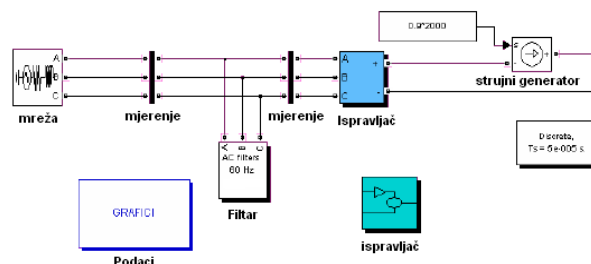
I_{sn} - efektivna vrijednost n-tog harmonika

I_s - efektivna vrijednost struje (napona)

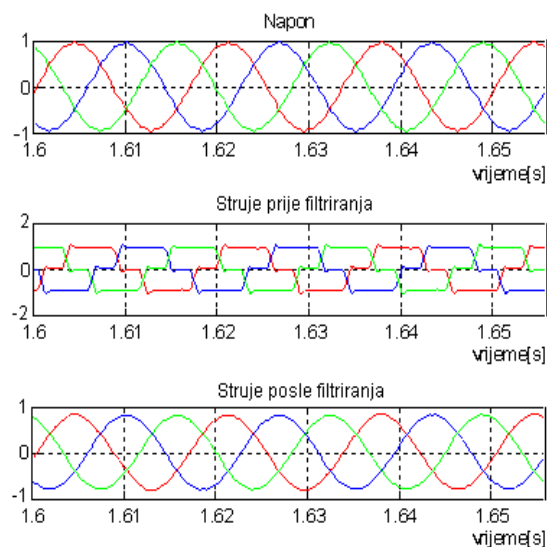
U referenci [8] dati su standardi (IEEE Standard 519-1992) vezani za maksimalno dozvoljene vrijednosti THD-a napona i struje naizmjenične mreže kod HVDC sistema. Prema ovom standardu THD napona za DC sistem naponskog nivoa $>161kV$ ne smije biti veći od 1.5%, dok THD struje za isti naponski nivo ne smije biti veći od 2.5%.

4. ANALIZA UTICAJA HVDC SISTEMA, NA NAIZMJENIČNU MREŽU, KOD KOGA JE DC STRUJA IDEALNO ISPEGLANA

Slika 4.1 predstavlja realizovani Simulink model jednog dijela HVDC sistema, kod koga je opterećenje strujni generator (analogno idealno ispeglanoj struji DC sistema).



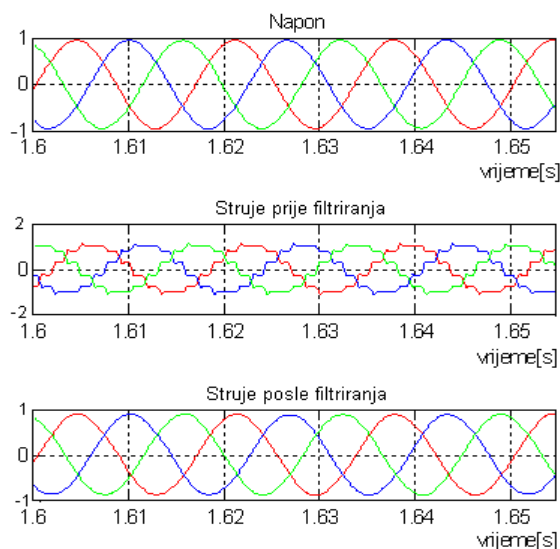
Sl. 4.1 Simulink model ispravljačkog dijela HVDC sistema sa strujnim generatorom



Sl. 4.2 Talasani oblici napona mreže i struje (prije i posle filtriranja) kod 6-to pulsno ispravljača

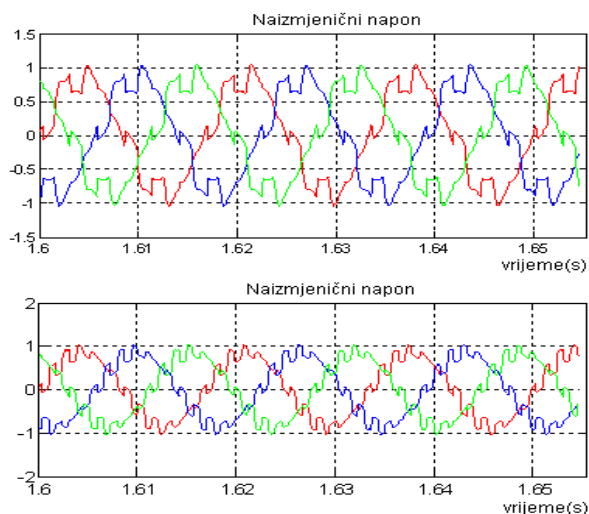
Na slici 4.2 prikazani su normalizovani oblici napona i struje mreže (prije i nakon filtriranja) kod 6-to pulsno ispravljača. Sasvim je jasno da struja prije filtra mora biti

pravougaonog oblika jer je opterećenje samog ispravljača čisto strujni generator. Nakon filtriranja najznačajnih harmonika (5, 7, 11 i 13), dobija se oblik koji je veoma blizak sinusoidi.



Sl. 4.3 Talasni oblici napona mreže i struje (prije i posle filtra) kod 12-to pulsno ispravljača

Normalizovani napona i struje mreže (prije i nakon filtra) kod 12-to pulsno ispravljača, prikazani su na slici 4.3. Ovdje je jasno da struja prije filtra mora biti stepenastog oblika, a nakon filtriranja istih harmonika, kao i kod šestopulsno ispravljača, dobija se oblik struje koji je takođe značajno blizak sinusoidi.

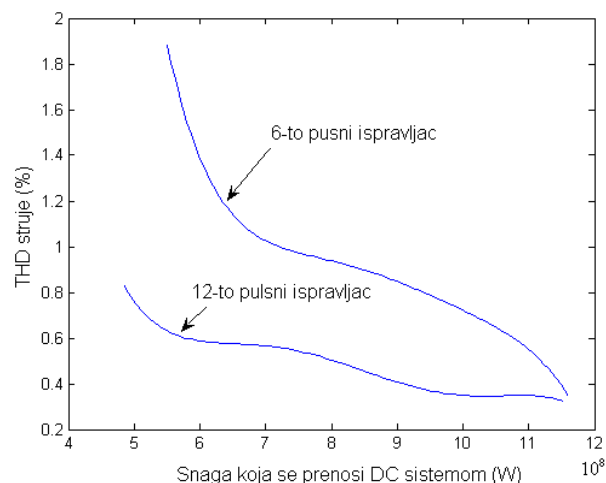


Sl. 4.4 Talasni oblik napona mreže u slučaju da nema filtra, kod 6-to pulsno i kod 12-to pulsno ispravljača, respektivno

Značaj postavljanja filtra može se uočiti sa slike 4.4, gdje se tačno mogu uočiti odgovarajući žljebovi (urezi ili udubljenja) na naponu mreže, koji nastaju kao posledica komutacije napona tiristora, [5].

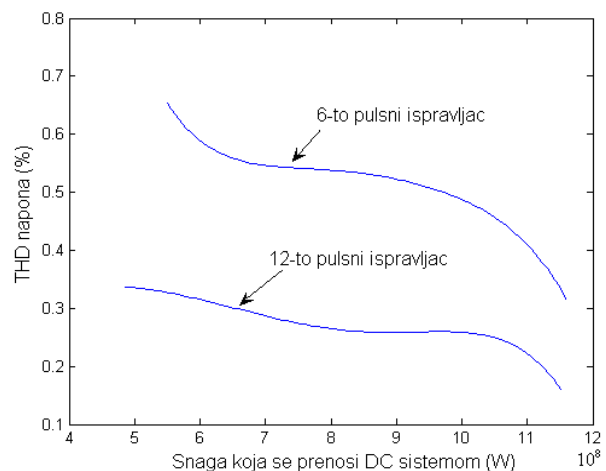
Sa slika 4.5 i 4.6, na kojima su prikazane interpolirane vrijednosti THD-a, može se zaključiti da je THD napona i struje kod dvanaestopulsno ispravljača manji od THD-a kod šestopulsno ispravljača. Takođe, jasno je da se sa porastom snage, koja se prenosi DC sistemom, smanjuje THD napona i struje.

Za snage reda 1000 MW, THD struje (a i napona) dvanaestopulsno ispravljača je manji od 0.4 %, dok je THD struje (i napona) kod šestopulsno ispravljača manji od 0.8%.



Sl. 4.5 Zavisnost THD-a struje mreže od snage koja se prenosi DC sistemom (pri struji od 1800A)

U ovim simulacijama, struja opterećenja je držana na vrijednosti od 1800A, dok je mijenjana vrijednost ugla paljenja tiristora. Za manji ugao paljenja tiristora, napon DC sistema je veći, pa je veća i snaga koja se prenosi DC sistemom, a THD struje (napona) je manji. Sa povećanjem ugla provođenja tiristora, napon DC sistema se smanjuje, a sa njim i snaga koja se prenosi, dok THD struje (napona) raste.



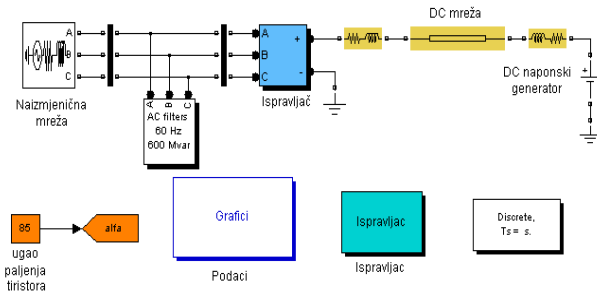
Sl. 4.6 Zavisnost THD-a napona od snage koja se prenosi DC sistemom (pri struji od 1800A)

5. ANALIZA UTICAJA HVDC SISTEMA NA NAIZMJENIČNU MREŽU, PRI KONSTANTNOM NAPONU DC MREŽE

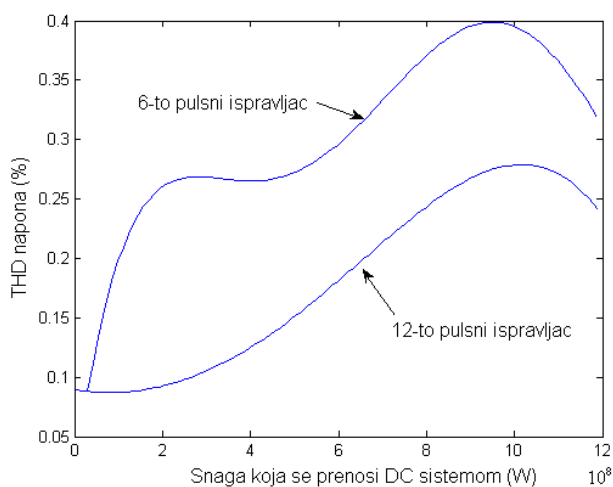
Za analizu uticaja HVDC sistema, na naizmjeničnu mrežu, kod koga je napon mreže konstantan, realizovan je Simulink model prikazan na slici 5.1.

Koristeći realizovani Simulink model, dobijene su zavisnosti prikazane na slikama 5.2 i 5.3. Na osnovu ovih rezultata može se zaključiti da je THD u slučaju upotrebe dvanaestopulsno ispravljača značajno manji nego u slučaju upotreba šestopulsno ispravljača. THD struje i napona (slike 5.2 i 5.3) kod dvanaestopulsno ispravljača ima vrijednost koja je manja od 0.4%, što je u skladu i sa rezultatima

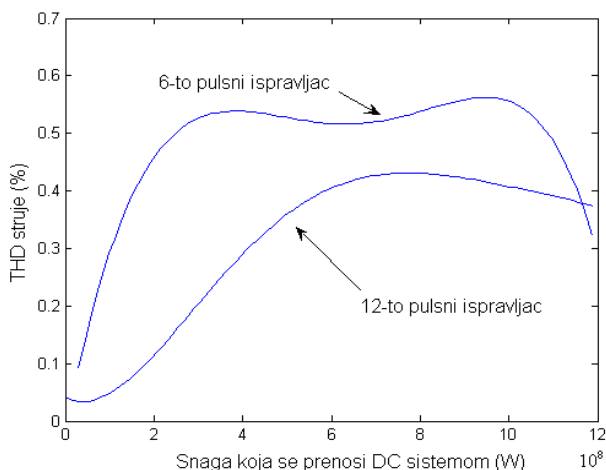
prikazanim na slikama 4.5 i 4.6. Za razliku od njega, THD napona i struje kod šestopulsnog ispravljača je znatno veći, ali, u poređenju sa rezultatima prikazanim na slikama 4.5 i 4.6, značajnije je manji.



Sl.5.1 Simulink model dijela DC sistema sa naponskim generatorom i DC mrežom, kao opterećenjem (napon potrošača je $U_{dc}=500kV$)



Sl. 5.2 Zavisnost THD-a napona kod 6-to i 12-to pulsno ispravljača od snage koja se prenosi DC sistemom (napon potrošača je 500 kV)



Sl. 5.3 Zavisnost THD-a struje kod 6-to i 12-to pulsno ispravljača od snage koja se prenosi DC sistemom (napon potrošača je 500 kV)

6. ZAKLJUČAK

U ovom radu prikazan je uticaj šestopulsnih i dvanaestopulsnih ispravljača kod HVDC sistema na naizmjeničnu mrežu. Takođe, prikazan je značaj postavljanja filtra za poništavanje najizraženijih harmonika.

Cilj ovoga rada bio je da se pokaže koliki uticaj ima odabir tiristorskih mostova kod HVDC sistema na struje i napone naizmjenične mreže, jer se u Crnoj Gori planira postavljanje podmorskog kabla do Italije. Pošto je zaključak studije o postavljanju podmorskog kabla da je EES Crne Gore slab sistem i da će na njega veliki uticaj imati pojava viših harmonika, ovaj rad je posvećen analizi THD-a napona i struje naizmjenične mreže. Rezultati izvršene analize pokazuju da dvanaestopulsni ispravljač ima bolje karakteristike u pogledu THD-a, u slučaju kada se koriste isti filteri. Dobijena vrijednost THD-a napona i struje, u ovoj simulaciji, u skladu je sa standardom IEEE Standard 519-1992.

LITERATURA

- [1] N.Hingorani: *High-voltage DC Transmission*, 1993. Power Electronics Workhorse
- [2] D. M. Larruskain1, I. Zamora1, A. J. Mazón1, O. Abarrategui1, J. Monasterio2 : "Transmission and Distribution Networks: AC versus DC", *University of the Basque Country*
- [3] Michael P. Bahrman : *Overview of HVDC transmission*, ABB Inc. ©2006 IEEE
- [4] V.G.Agelidis, G. D. Demetriades, N. Flourentzou: *Recent Advances in High-Voltage Direct-Current Power Transmission Systems*, John Wiley & Sons, 2007.
- [5] N. Mohan, T. M. Undeland, W. P. Robbins: *Power Electronics, Converters, Applications and Design*, New York, John Wiley 2003.
- [6] J. Arrillaga, Y. H. Liu, N. R. Watson: *Flexible Power Transmission (The HVDC Options)*, 2007.
- [7] Short Circuit, Voltage and Dinamic Stability Analyses Reliability and Harmonic impact for new under-sea HVDC interconnection between Italy and Montenegro, Electricity Coordinating Center, Belgrade, July 2008.
- [8] Muhammad H. Rashid: *Power Electronics Handbook*, Academic press, 2001.

GUI MODEL ZA ISPITIVANJE DEJSTVA STEP SMETNJE NA REFERENTNU VRIJEDNOST NAPONA GENERATORA GUI MODEL FOR TESTING STEP INTERFERENCE EFFECTS ON GENERATOR VOLTAGE REFERENCE VALUE

Martin Čalasan, Milutin Ostojić, *Elektrotehnički fakultet Podgorica*

Sadržaj - U ovom radu je prikazan realizovani GUI model jednog generatora iz HE „Perućica“, koji služi za proučavanje dejstva step smetnje na referentnu vrijednost napona generatora kod pobudnog sistema Thyricon. Dobijeni simulirani odzivi imaju eksperimentalnu potvrdu.

Abstract - This paper presents a realized model of a GUI generator from HPP Perucica, which is used to study effects of step disturbances to the reference value of the voltage generators at static excitation system Thyricon. The experimental and computed responses show very good agreement.

Keywords: *excitation system, step interference, GUI model*

1. UVOD

Savremeni uređaji u industriji, telekomunikacijama, vojsci, pa i u domaćinstvima zahtijevaju neprekidno napajanje električnom energijom, sa strogim zahtjevima po pitanju njenog kvaliteta. Pošto su osnovni pokazatelji kvaliteta električne energije, napon i frekvencija, onda se u procesu proizvodnje električne energije vrši prije svega regulacija tih parametara, a posredno preko njih i regulacija aktivne i reaktivne snage sinhronih generatora. Održavanje stalne vrijednosti napona na krajevima mašine se ostvaruje preko sistema pobude, odnosno automatskog regulatora napona, promjenom pobudne struje sinhronih generatora.

U radu [1] naveden je značaj pobudnih sistema, dok je u radu [2] navedena podjela pobudnih sistema, njihove karakteristike, osnovni djelovi i slično. Klasifikacija pobudnih sistema koji su primijenjeni kod generatora u HE „Perućica“ prikazana je u radu [3], dok je u radu [4] predstavljen Simulink model jednog generatora iz HE „Perućica“, sa pobudnim sistemom Thyricon. Opis pobudnog sistema Thyricon objašnjavaju reference [5] i [6]. Reference [7-11] prikazuju načine modelovanja djelova EES-a.

Ovaj rad prikazuje GUI model jednog generatora iz HE „Perućica“, sa pobudnim sistemom Thyricon. Poznavanje modela ovog sistema je veoma značajno jer programski paketi koji služe za analizu tokova snaga, kratkih spojeva, dinamičke stabilnosti i slično, kao što su PSS/E i ETAP, zahtijevaju i tačan model pobudnog sistema. GUI je pogodan jer vizuelno pruža mogućnost lake promjene parametara, odabira grafika za crtanje ... Na kraju samog rada biće prikazani simulirani odzivi ovoga pobudnog sistema na dejstvo step smetnje na referentnu vrijednost napona generatora, kao i odzivi koje je dobio VOITH SIEMENS mjerenjem na generatoru od 40MVA u HE „Perućica“.

2. POBUDNI SISTEMI

Sistem pobude predstavlja izvor struje pobude potrebne za održavanje napona na krajevima mašine. Ovaj sistem uključuje mašine i aparate potrebne za proizvodnju struje pobude (pobudnica), uređaje za regulaciju pobude (regulator

pobude) i elemente ručnog upravljanja, mjerenja, zaštite i automatike. Sistem za regulaciju pobude razlikuje se od sistema pobude jer uključuje i sam objekat regulacije, odnosno sinhronu mašinu [2]. U zavisnosti od toga kako se obezbjeđuje jednosmjerna pobudna struja sinhronih mašina, razlikuju se 3 tipa pobudnih sistema:

- *Jednosmjerni (DC) sistemi*
- *Nezavisni naizmjenični (AC) sistemi pobude*
- *Statički naizmjenični (ST) sistemi pobude*

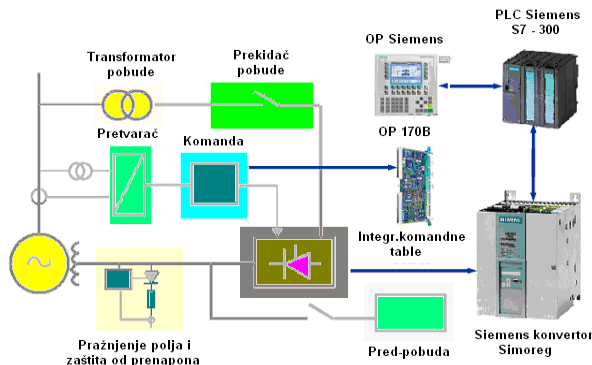
Kod *jednosmjernih (DC) sistema pobude*, kao izvor struje pobude sinhronog generatora koriste se jednosmjerni generatori. Ovaj generator može da se nalazi na istoj osovini zajedno sa sinhronim generatorom, dok može i da ga pokreće neki motor napajan iz mreže sopstvene potrošnje elektrane. Kod *nezavisnih naizmjeničnih (AC) sistema pobude*, pobudna struja se dobija korišćenjem pobudnog sinhronog generatora i ispravljača. Ispravljači mogu biti nekontrolisani (upotrebom dioda) ili kontrolisani (upotrebom tiristora).

Statički naizmjenični sistemi pobude uzimaju energiju za pobudu sa krajeva samog generatora kojeg pobuđuju. Osim transformatora, koji može biti ili obični energetski, ili kompaundni (sa strujnom ili faznom kompaundacijom), u ovom slučaju koriste se i ispravljači, kako kontrolisani tako i nekontrolisani. Za ovaj sistem pobude je karakteristično da su sve komponente stacionarne, a pošto se kod njega koristi napajanje sa samog generatora koji se pobuđuje, ovaj sistem pobude spada u *sisteme samopobude*. Da bi generator mogao započeti indukovanje napona na svojim krajevima, ovi sistemi moraju biti snabdjeveni nekim izvorom početne energije (najčešće su to akumulatorske baterije sopstvene potrošnje elektrane). Nedostatak pobudnih sistema bez kompaundacije je taj što se pri kvarovima (kratkim spojevima) u mreži, na krajevima generatora pojavljuje nizak napon, pa je odmah niska i vrijednost pobudne struje, jer se pobuda napaja sa krajeva generatora preko transformatora i ispravljača. U tim situacijama zahtijevaju se velike vrijednosti pobudne struje, da bi se zadovoljili uslovi dinamičke stabilnosti generatora. Ovaj nedostatak se koriguje na taj način što se transformator izvodi kao kompaundna jedinica: *Sekundarni napon takvog transformatora zavisi ne*

samo od napona primara već i od struje, pa se kod njega pad napona na primaru kompenzuje velikim strujama kvara.

3. STATIČKI POBUDNI SISTEM THYRICON

Thyricon spada među najmodernije pobudne sisteme sinhronih generatora. Ovi statički pobudni sistemi našli su veliku primjenu u brojnim elektranama širom Evrope.

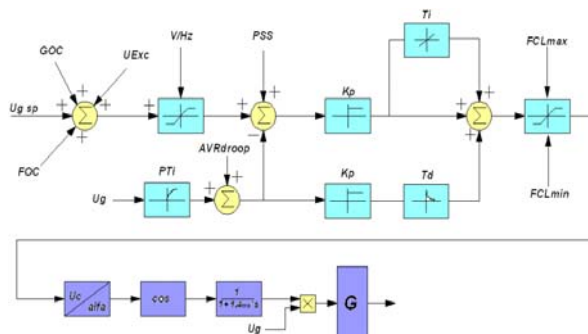


Slika 3.1 Glavne komponente Thyricon-a

Pobudni sistem Thyricon podijeljen je na dva dijela: *komandni i energetska*, a njegove glavne komponente prikazane su na slici 3.1. *Komandni dio* čine elektronski moduli, kao što su: programabilni logički kontroleri (PLC), interfejsi čovjek-mašina (operacioni panel), komandni paneli, pretvarači (konvertori), releji, komunikacione table i slično. Thyricon koristi standardni konvertor Siemens AG (*Simoreg 6RA70*), koji radi na principu integrisanog komandnog sistema, što znači da se komanda vrši u panelima *T400* (odgovoran je za kontrolu komandne petlje upravljanja), *CUDI* (vrši paljenje tiristora u ispravljačkim mostovima), *CUDI2* (dodatak kartice *CUDI*), koje su smještene u konvertorskoj jedinici. U *energetski dio* spadaju kola i komponente koje su robusnije od komandnog dijela, kao što su: mostovi ispravljača, kola za detekciju prenapona, kolo field flashing (kolo koje obezbjeđuje početnu energiju pobudnom namotaju), kolo za pražnjenje prenapona i slično [5], [6].

Thyricon je dizajniran tako da nudi kompletan set automatske i ručne regulacije napona sinhrona mašine. Svrha *automatskih modova* jeste automatsko regulisanje napona generatora (AVR), reaktivne snage i faktora snage generatora. Regulator reaktivne snage i faktora snage koristi AVR kao podređenu strukturu, mijenjajući podešenu vrijednost napona radi dobijanja željene vrijednosti reaktivne snage ili samog faktora snage. Funkcija AVR-a je kontrola izlaznog napona generatora upotrebom PID regulatora. Razlika između podešene vrijednosti napona i stvarnog napona generatora predstavlja ulaz u PID regulator. Izlaz iz PID regulatora kontroliše ugao okidanja tiristorskog mosta, pa na taj način AVR podiže ili spušta napon pobude i tako reguliše izlazni napon generatora. Na slici 3.2 prikazan je blok dijagram automatskog regulatora napona (AVR mod) Thyricona. Kod *ručnog moda* Thyricona vrši se ručna kontrola izlaznog napona generatora, na taj način što se djeluje na struju pobude. Podešavajući pobudnu struju, vrši se indirektna kontrola izlaznog napona. Ručni mod regulatora napona radi kao podrška automatskom modu. No, ručni mod

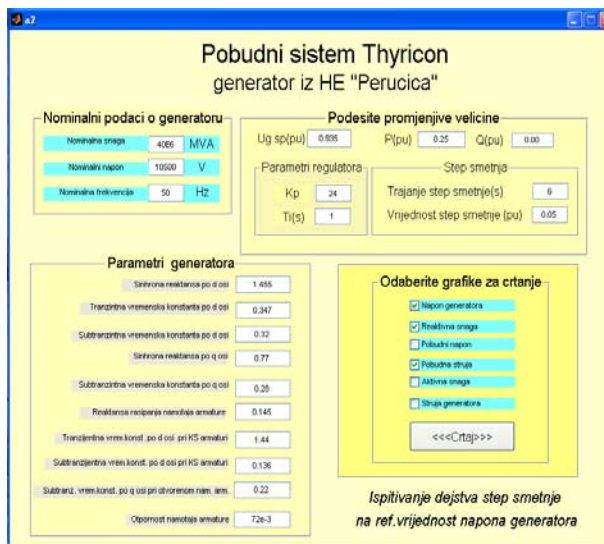
ima i dodatne funkcije, kao što su upotreba kod ispitivanja u praznom hodu i u kratkom spoju generatora. Postoji *FCR* – regulator struje pobude i *EFCR* – pomoćni regulator struje pobude.



Slika 3.2 Blok dijagram automatskog regulatora napona

U Thyricon-u su implementirani i brojni limiteri koji obezbjeđuju siguran i pouzdan rad mašine i njenih dijelova. To su: *FOC* – limiter pobudne struje, *GOC* – limiter struje generatora, *V/Hz* – limiter fluksa u mašini, *FCL* – brzi limiteri pobudne struje, *UExc* – limiter podpobude. Osim limitera rada, kod Thyricona postoje i stabilizator EES-a (*PSS*), kao i kompenzatori pada reaktivne i aktivne snage u generatoru i blok transformatoru (*AVRdroop*). Sa slike 3.2 se može vidjeti na koji način oni vrše uticaj na glavni PID regulator AVR-a kod pobudnog sistema Thyricon.

4. GUI MODEL JEDNOG GENERATORA IZ HE „PERUĆICA“ SA POBUDNIM SISTEMOM THYRICON

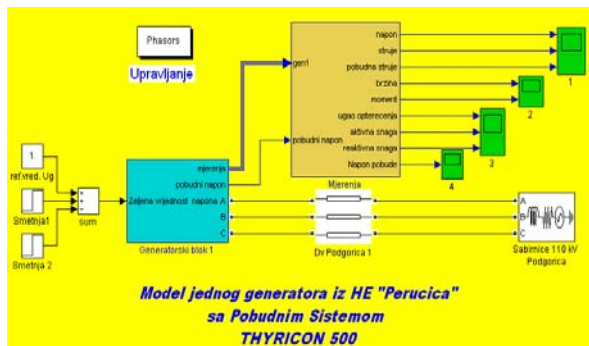


Slika 4.1 GUI model jednog generatora iz HE „Perućica“ sa pobudnim sistemom Thyricon

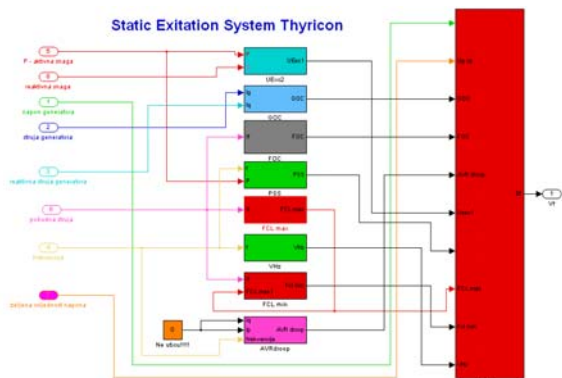
HE „Perućica“ ima sedam sinhronih generatora. Na slici 4.1 prikazan je GUI model jednog generatora od 40MVA, sa dalekovodom 110kV do TS Podgorica 2. Ovaj GUI model pruža mogućnost lake kontrole parametara regulatora, trajanja vrijednosti step smetnje, kao i lak odabir grafika za crtanje.

GUI model je realizovan tako da aktivira Simulink model jednog generatora iz HE „Perućica“ sa priključnim dalekovodom, koji je prikazan na slici 4.2. Na slici 4.2

Generatorski blok čine generator, sopstvena potrošnja elektrane, transformator, pobudni sistem, turbinski regulator i blokovi za mjerenja. Za regulaciju protoka fluida iskorišćen je turbinski regulator koji postoji u biblioteci *SimPowerSystem* u Simulinku, dok pobudni sistem odgovara stvarnom statičkom pobudnom sistemu Thyron, čije je modelovanje prikazano na slici 4.3.



Slika 4.2 Simulink model jednog generatora sa pobudnim sistemom Thyron iz HE „Perućica“



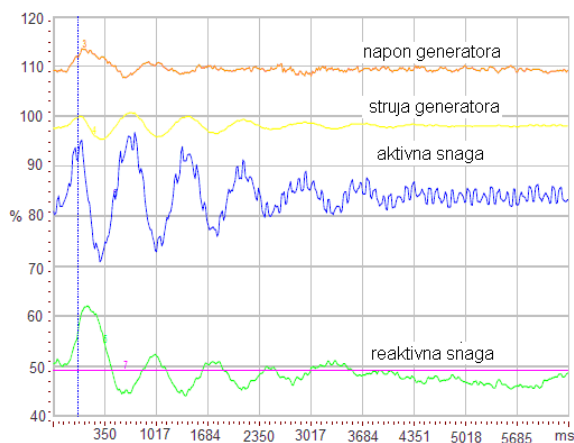
Slika 4.3 Simulink model pobudnog sistema Thyron

5. ISPITIVANJE DEJSTVA STEP SMETNJE NA REFERENTNU VRIJEDNOST NAPONA GENERATORA

Na slici 5.1 prikazani su odzivi karakterističnih veličina na dejstvo step smetnje, u trajanju od 200ms, na referentnu vrijednost napona generatora, koja je vršio VOITH SIEMENS, dok su na slici 5.2 prikazani ekvivalentni odzivi dobijeni upotrebom realizovanog GUI modela jednog generatora iz HE „Perućica“. Slike 5.3 i 5.4 prikazuju odzive na step smetnju, čije je trajanje 6 sekundi.

Step smetnja utiče na povećanje vrijednosti greške na ulazu u PI regulator AVR-a. Kao posljedica toga, napon pobude se naglo povećava, a samim tim i struja pobude. Forsirajući pobudnu struju mašine povećava se iznos reaktivne snage. Pošto step smetnja ne utiče na snagu turbine koja pokreće rotor mašine, aktivna snaga se nije promijenila, mada je malo zaoscilovala. Kako se povećao iznos reaktivne snage koju mašina daje mreži, a aktivna snaga je ostala konstantna, promijenio se i faktor snage. Takođe, forsiranjem pobude dolazi i do smanjenja ugla opterećenja [1].

Tabela I prikazuje tumačenje „y“ ose snimljenih veličina za odzive prikazane na slikama 5.1 i 5.2, a Tabela II tumačenje za odzive prikazane na slikama 5.3 i 5.4.



Slika 5.1 Odziv karakterističnih veličina na step smetnju, u trajanju od 200ms, koja je vršio VOITH SIEMENS u HE „Perućica“



Slika 5.2 Odziv karakterističnih veličina na step smetnju u trajanju od 200ms, koja se dobijaju upotrebom GUI modela jednog generatora iz HE „Perućica“

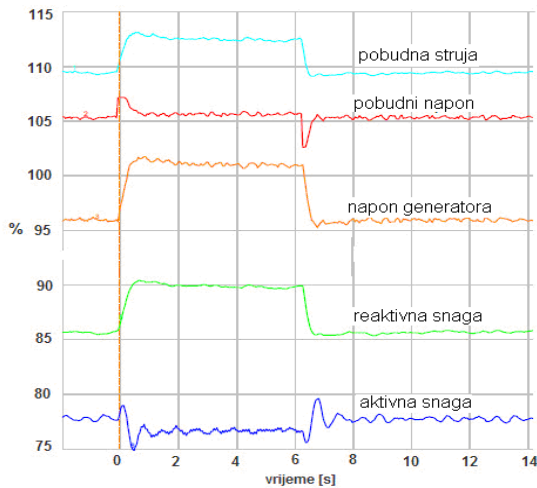
Tabela I: Skala „y“ ose za odzive prikazane na slici 5.1 i 5.2

VELIČINA	SKALA
Struja generatora	$(-200\% \text{ do } 200\%) \cdot I_n \text{ (A)}$
Napon generatora	$(60\% \text{ do } 100\%) \cdot U_n \text{ (V)}$
Reaktivna snaga	$(-10\% \text{ do } 70\%) \cdot S_n \text{ (MVA)}$
Aktivna snaga	$(40\% \text{ do } 120\%) \cdot S_n \text{ (MVA)}$

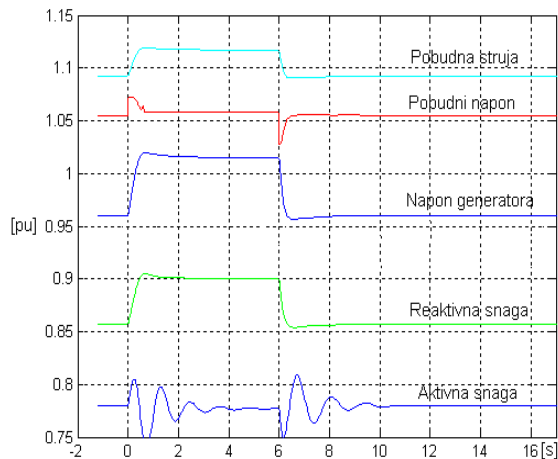
Tabela II: Skala „y“ ose za odzive prikazane na slici 5.3 i 5.4

VELIČINA	SKALA
Pobudna struja	$(-120\% \text{ do } 40\%) \cdot 1600 \text{ (A)}$
Pobudni napon	$(-1200\% \text{ do } 400\%) \cdot 2U_{fn} \text{ (V)}$
Napon generatora	$(75\% \text{ do } 115\%) \cdot U_n \text{ (kV)}$
Reaktivna snaga	$(-100\% \text{ do } 300\%) \cdot S_n \text{ (MVA)}$
Aktivna snaga	$(20\% \text{ do } 100\%) \cdot S_n \text{ (MVA)}$

Ako je skala (od -200% do 200%), to znači da na grafiku oznaci 40% (ili 0.4) odgovara vrijednost $-2 \cdot I_n$, a vrijednosti 120% (ili 1.2) odgovara $2 \cdot I_n$.



Slika 5.3 Odziv karakterističnih veličina na step smetnju, u trajanju od 6s, koja je vršio VOITH SIEMENS u HE "Perućica"



Slika 5.4 Odziv karakterističnih veličina na step smetnju u trajanju od 6s, koji se dobijaju upotrebom GUI modela jednog generatora iz HE "Perućica"

6. ZAKLJUČAK

U ovom radu je prikazan GUI model jednog generatora iz HE „Perućica“ sa statičkim pobudnim sistemom Thyricon, koji služi za ispitivanje dejstva step smetnje na referentnu vrijednost napona generatora. Osim toga, u radu je prikazan i Simulink model jednog generatora od 40 MVA, kao i model pobudnog sistema Thyricon.

Analizom dobijenih odziva u simulaciji i odziva koje je snimio VOITH SIEMENS u HE „Perućica“ uočava se dobro poklapanje rezultata. Sa druge strane GUI model pruža laku mogućnost modifikovanja parametara regulatora, ili pak samog generatora. Takođe, on pruža mogućnost lakog odabira grafika za crtanje.

Zbog toga, realizovani GUI model jednog generatora sa pobudnim sistemom Thyricon može biti modifikovan i prilagođen za brojna druga ispitivanja i analize, a ne samo za analizu step smetnji.

LITERATURA

- [1] M. Ostojić: „Sinhronne Mašine“, Nikšić, 1994 godine
- [2] M. Čalović: „Regulacija elektroenergetskih sistema“, Tom 2, (Elektrotehnički fakultet u Beogradu, Beograd, 1997)
- [3] M. Čalasan, R. Vukotić: „Pobudni sistemi sinhronih generatora koji su u funkciji u HE „Perućica“, CIGRE, oktobar 2009.
- [4] M. Čalasan, M. Ostojić: „Ispitivanje dejstva step smetnje na referentnu vrijednost napona generatora sa pobudnim sistemom Thyricon“, *ETF Journal of ELECTRICAL ENGINEERING*, Podgorica, Novembar 2009.
- [5] Tehnička dokumentacija sistema pobude generatora br. 1, 2, 3 i 4, HE "Perućica"
- [6] Tehnička dokumentacija sistema pobude Thyricon, HE "Piva"
- [7] „Computer Models for Representation of Digital-Based Excitation Systems“, *Digital Excitation Task Force of the Equipment Working Group*,
- [8] Jun Ho, Kenneth Mun: "Implementation of Controls to a Synchronous Machine (Simulation using MATLAB)", University of Queensland, October 2003
- [9] Z. Chen, F. Blaabjerg, F. Iov: "A Study of Synchronous Machine Model Implementations in Matlab/Simulink" *Simulations for New and Renewable Energy Systems*
- [10] A. S. Bozin: "Electrical power systems modeling and simulation using Simulink", 1998, *The Institution of Electrical Engineers*, London
- [11] Simulink, "Dynamic system simulation software", *The Mathworks Inc*, 2001

DIGITALNI HALFBAND IIR FILTRI SA PRIBLIŽNO LINEARNOM FAZOM DIGITAL HALFBAND IIR FILTERS WITH APPROXIMATELY LINEAR PHASE

Miroslav Lutovac, *Državni Univerzitet u Novom Pazaru, Srbija*
Aleksandar Radonjić, *Crnogorski telekom, Podgorica, Crna Gora*

Sadržaj – Projektovanje digitalnih IIR filtara linearne faze je poznato za slučaj maksimalno ravne i Čebiševljeve amplitudske karakteristike. U ovom radu je predložen metod za projektovanje IIR filtara linearne faze za poznate nule funkcije prenosa. Projektovanje se zasniva na primeni algebarskog računarskog sistema.

Abstract – The design of digital IIR linear phase filters is known for approximation in the sense of maximal flatness and in the sense of Chebyshev. In this paper is presented the method for the design of IIR linear phase filters for known transfer function zeros. The design is based on computer algebra system.

1. UVOD

U radu [1] koji je posvećen profesoru Alfredu Fettweis-u povodom njegovog 75. rođendana, opisani su postupci za projektovanje rekurzivnih halfband filtara. U ovom radu [1] može se naći i spisak radova i knjiga koji detaljno obrađuju ovu materiju. Brojne primene halfband filtara su opisane u monografiji [2], u kojoj se mogu naći i programi za projektovanje digitalnih filtara kao i linkovi na Internetu gde se nalaze izvorni kodovi i dodatna literatura. Jedna od važnih primena rekurzivnih digitalnih filtara je za dvo-kanalne filterske banke zasnovane na tehnici frekvencijskog maskiranja (Frequency-Response Masking Approach) [3].

Uobičajeni problemi koji se javljaju u praksi su da linkovi sa izvornim kodom budu dostupni neko vreme, a da se potom više ne mogu naći. Tada je potrebno da se provere publikovani rezultati i na osnovu njih napišu sopstveni programi. Ovaj rad je nastao analizom osobina digitalnih IIR filtara linearne faze i pronalaženjem osobina ovih filtara. Korišćen je softverski alat koji pripada klasi algebarskih računarskih sistema (*Mathematica* [4]), kao i aplikacioni paket koji radi u ovom sistemu (*SchematicSolver* [5]).

Korišćeni alati iz klase algebarskih računarskih sistema već su pokazali da je moguće pronaći osobine analiziranih sistema i relacije koje daju vezu parametara kontinualnih, vremenski diskretnih i nelinearnih sistema u zatvorenoj formi [6], [7], [8]. U sledećem poglavlju biće opisani digitalni halfband IIR filtri, i prikazane poznate osobine ovih filtara.

U narednim poglavljima će biti pokazano kako se koristi algebarski računarski sistem za nalaženje osobina koje nisu bile poznate. Na kraju će biti pokazano kako se generalizuju rezultati i proverava tačnost upoređivanjem sa rezultatima dobijenih numeričkim postupcima.

Kao polazni filter koristi se allpass filter, a rezultujući halfband filter ima približno linearnu fazu. U ovom radu korišćemo termin linearne faze radi jednostavnijeg praćenja. Sa IIR filtrima nije moguće da se ostvari idealna linearna fazna karakteristika, ali se često može smatrati da je odstupanje od linearne faze toliko malo da se mogu zanemariti ova odstupanja [9].

2. ALLPASS FILTER KAO OSNOVNI FILTER

Allpass IIR filtri n -tog reda imaju osobinu da su im koeficijenti u brojiocu i imeniocu identični takvi da je koeficijent a_i pridružen uz z^{-i} u imeniocu, a uz z^{-n+i} u brojiocu

$$H_{AP1}(z) = \frac{a_n + a_{n-1}z^{-1} + \dots + a_1z^{-n+1} + z^{-n}}{1 + a_1z^{-1} + \dots + a_{n-1}z^{-n+1} + a_nz^{-n}} \quad (1)$$

U slučaju halfband filtra, koeficijenti uz neparne stepene su jednaki nuli, pa zbog toga red filtra mora da bude paran. Na primer, za slučaj filtra četvrtog reda se dobija

$$H_{AP1}(z) = \frac{a_4 + a_2z^{-2} + z^{-4}}{1 + a_2z^{-2} + a_4z^{-4}} \quad (2)$$

Da bi realizovali halfband filter propusnik niskih učestanosti i njegov komplementarni filter propusnik visokih učestanosti, treba realizovati allpass filter neparnog reda. Na primer, filter trećeg reda koji je jednak kašnjenju z^{-3}

$$H_{AP2}(z) = z^{-3} = \frac{z^{-3} + a_2z^{-5} + a_4z^{-7}}{1 + a_2z^{-2} + a_4z^{-4}} \quad (3)$$

Funkcija prenosa filtra propusnika niskih učestanosti je jednaka zbiru ove dve funkcije prenosa, a funkcija prenosa filtra propusnika visokih učestanosti je jednaka razlici ove dve funkcije prenosa. Potrebno je dodatno pomnožiti konstantom $1/2$ da bi funkcija prenosa niskih učestanosti imala jediničnu vrednost za $z=1$. Filter propusnik niskih učestanosti ima funkciju prenosa

$$H_{LP}(z) = \frac{H_{AP1}(z) + H_{AP2}(z)}{2} \quad (4)$$

Filter propusnik visokih učestanosti ima funkciju prenosa

$$H_{HP}(z) = \frac{H_{AP1}(z) - H_{AP2}(z)}{2} \quad (5)$$

Konačno se dobija da je funkcija prenosa niskih učestanosti za slučaj allpass filtra četvrtog reda $H_{LP}(z)$

$$H_{LP} = \frac{\frac{a_4}{2} + \frac{a_2 z^{-2}}{2} + \frac{z^{-3}}{2} + \frac{z^{-4}}{2} + \frac{a_2 z^{-5}}{2} + \frac{a_4 z^{-7}}{2}}{1 + a_2 z^{-2} + a_4 z^{-4}} \quad (6)$$

Allpass filter ima uvek jedan par polova simetrično na imaginarnoj osi u odnosu na $z = 0$. Ako je red filtra $n = 4k$, $k = 1, 2, 3, \dots$, tada jedan par polova mora da bude na realnoj osi simetrično u odnosu na $z = 0$. Ako je red filtra $n \neq 4k$, $k = 2, 3, \dots$, jedan par polova je na imaginarnoj osi a ostali polovi se pojavljuju u kvadrupletu svi simetrično u odnosu na $z = 0$. Svi polovi moraju biti unutar jediničnog kruga.

Cilj svakog filtra propusnika niskih učestanosti jeste da ostvari što je moguće veće slabljenje u nepropusnom opsegu, pa je stoga najbolje da su sve nule funkcije prenosa na jediničnom krugu u prvom i četvrtom kvadrantu z ravni. To znači da koreni polinoma po z u imeniocu funkcije prenosa filtra propusnika niskih učestanosti datih relacijom (6) treba da budu unutar jediničnog kruga i to tako da je jedan par polova uvek na imaginarnoj osi, da jedan par polova može biti na realnoj osi, a da svi ostali polovi, ako ih ima, se pojavljuju u kvadrupletima. Istovremeno, koreni polinoma po z u brojiocu funkcije prenosa definisanih relacijom (6), a to su nule funkcije prenosa, treba da budu na imaginarnoj osi. Kako postoje samo dva koeficijenta u funkciji prenosa allpass filtra četvrtog reda, a istovremeno je moguće realizovati samo dve nule funkcije prenosa, to sledi da postoji relacija na osnovu koje mogu (1) da se odrede nule funkcije prenosa ako su poznati koeficijenti, ili (2) da se odrede koeficijenti filtra ako su poznate nule funkcije prenosa. Cilj ovog rada je upravo da se pronađe ova relacija između koeficijenata filtra i nula funkcije prenosa.

3. AMPLITUDSKA KARAKTERISTIKA

Postupak koji sledi izveden je korišćenjem softverskog alata koji pripada klasi algebarskih računarskih sistema (*Mathematica* [4]), kao i aplikacionog paketa koji radi u ovom sistemu (*SchematicSolver* [5]). Najpre je definisan imenilac allpass funkcije prenosa četvrtog reda

$$\text{den} = 1 + A_2 z^{-2} + A_4 z^{-4}$$

Brojioc funkcije prenosa filtra propusnika niskih učestanosti je

$$\text{num1} = \frac{A_4}{2} + \frac{A_2}{2} z^{-2} + \frac{1}{2} z^{-3} + \frac{1}{2} z^{-4} + \left(\frac{A_2}{2} z^{-5} + \frac{A_4}{2} z^{-7} \right)$$

Nalaženje apsolutne vrednosti nije pogodno za manipulaciju, pa je stoga određen kvadrat amplitudske funkcije

$$H_m(z) = H_{LP}(z) H_{LP}\left(\frac{1}{z}\right) \quad (7)$$

Ovo se može dobiti na sledeći način

$$\text{myTF1b} = \frac{\text{num1} (\text{num1} /. z \rightarrow 1/z)}{\text{den} (\text{den} /. z \rightarrow 1/z)}$$

Konačno, amplitudska funkcija se dobija za vrednosti kompleksne promenljive na jediničnom krugu

$$\text{myTff} = (\text{myTF1b} /. z \rightarrow \text{Cos}[2\pi f] + I \text{Sin}[2\pi f]) // \text{ComplexExpand}$$

Imaginarni deo mora biti jednak nulu, pa se posle uprošćavanja dobija izraz u kome se pojavljuju isključivo

koeficijenti filtra, i učestanost koja može imati vrednost u opsegu od 0 do 0.5

$$\frac{\text{Cos}[f\pi]^2 (-1 + A_2 + A_4 - 2(A_2 + A_4) \text{Cos}[2f\pi] + 2A_4 \text{Cos}[4f\pi] - 2A_4 \text{Cos}[6f\pi])^2}{1 + A_2^2 + A_4^2 + 2A_2(1 + A_4) \text{Cos}[4f\pi] + 2A_4 \text{Cos}[8f\pi]}$$

Može se odrediti i prvi izvod ove funkcije, i iz njega se nalaze nule amplitudske funkcije ali i učestanosti svih ekstremuma. Izraz može biti veoma složen. Softver omogućava da se uradi faktorizacija po polinomima i zatim je moguće izdvojiti polinom koji sadrži nule funkcije prenosa ili učestanosti maksimuma u propusnom opsegu.

4. APROKSIMACIJA POLINOMIMA ZA ALLPASS FILTER ČETVRTOG REDA

Na osnovu dobijenih izraza nije bilo moguće utvrditi neko pravilo, pa se pribeglo uprošćavanju. Najpre su izvršene zamene cosinusnih funkcija korišćenjem trigonometrijskih pravila tako da se pojavljuje samo $\cos(\pi f)$, a zatim je uvedena smena

$$\text{Cos}[f\pi] \rightarrow w$$

Grupisanjem polinoma po w uz odgovarajuće koeficijente, dobijeni su polinomi kod kojih je prepoznato pravilo da bi polinomi mogli da budu Čebiševljevi. Tako na primer, polinom koji odgovara nulama funkcije prenosa filtra propusnika visokih učestanosti jednak je proizvodu koeficijenata i odgovarajućeg čebiševljevog polinoma prve vrste

$$\text{polHP} - A_4 * \text{ChebyshevT}[7, w] / w - A_2 * \text{ChebyshevT}[3, w] / w - 1$$

0

Prvi red u prethodnom izrazu je komanda, a drugi red je rezultat uprošćavanja, čime se dokazuje tačnost izraza za proizvoljne vrednosti koeficijenata i w koje odgovara učestanosti.

Polinom koji odgovara nulama funkcije prenosa filtra propusnika niskih učestanosti jednak je proizvodu koeficijenata i odgovarajućeg čebiševljevog polinoma druge vrste

$$\text{polLP} - A_4 * \text{ChebyshevU}[6, w] - A_2 * \text{ChebyshevU}[2, w] + 1$$

0

5. APROKSIMACIJE POLINOMIMA N-TOG REDA

Nakon izvršenih numeričkih provera za filtre 6. i 10. reda, rezultat je generalizovan tako da je polinom koji sadrži nule funkcije prenosa visokih učestanosti funkcija $w = \cos(\pi f)$ i čebiševljevog polinoma prve vrste k -tog reda $T_k(w)$

$$P_H(w) = 1 + \frac{+a_2 T_3(w) + a_4 T_7(w) + \dots + a_{2k} T_{4k-1}(w)}{w} \quad (8)$$

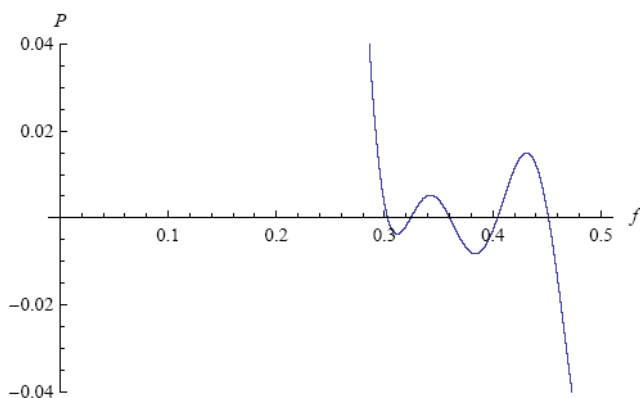
Polinom koji sadrži nule filtra propusnika niskih učestanosti može se izraziti preko čebiševljevih polinoma druge vrste k -tog reda $U_k(w)$

$$P_L(w) = 1 + a_2 U_2(w) + a_4 U_6(w) + \dots + a_{2k} U_{4k-2}(w) \quad (9)$$

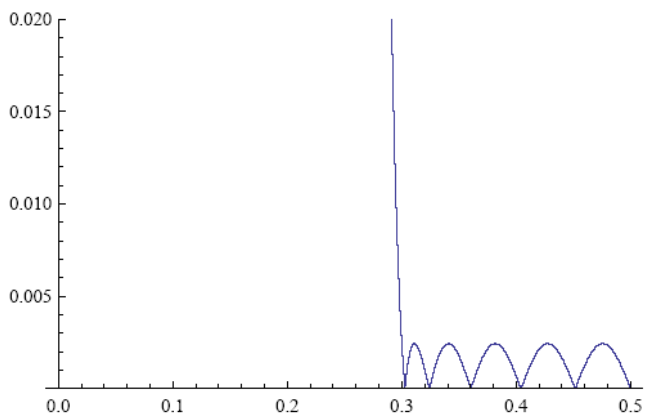
Na sledećim slikama su prikazane vrednosti polinoma i amplitudskih karakteristika. Na slici 1 je nacrtana polinomska karakteristika za filter propusnik niskih učestanosti. Na slici 2 je nacrtana amplitudska karakteristika filtra propusnika niskih učestanosti.

Svi primeri su urađeni za sledeće vrednosti koeficijenata:

$a_2 = 0.487783$,
 $a_4 = -0.10749$,
 $a_6 = 0.0423204$,
 $a_8 = -0.0179627$,
 $a_{10} = 0.00861601$.



Slika 1. Primer za $P_L(w)$ za $w = \cos(\pi f)$.



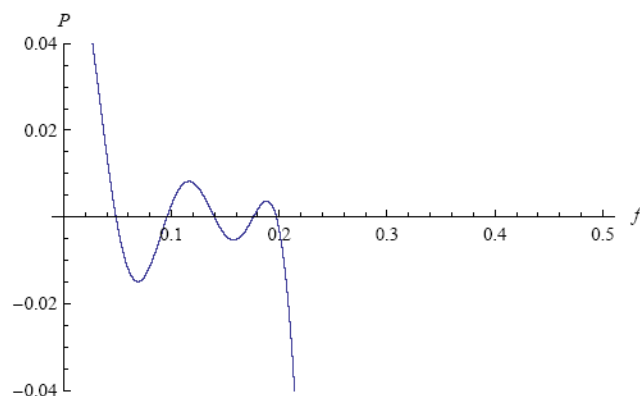
Slika 2. Primer za $M = |H_{LP}(e^{j2\pi f})|$.

Na slici 1 i 2 se vidi da su nule za iste vrednosti učestanosti.

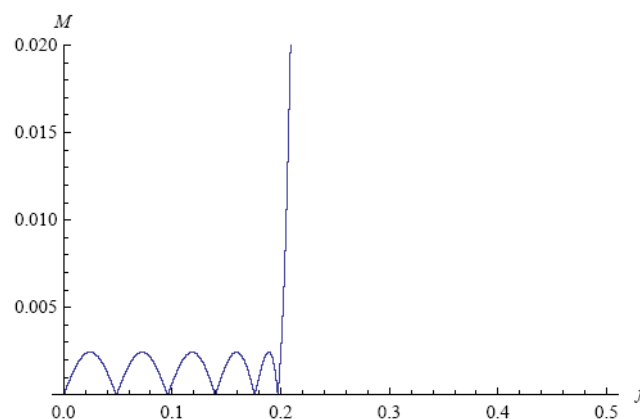
Na slici 3 je nacrtana polinomska karakteristika za filter propusnik visokih učestanosti. Na slici 4 je nacrtana amplitudska karakteristika filtra propusnika visokih učestanosti. Filter propusnik visokih učestanosti ima iste koeficijente kao i filter propusnik niskih učestanosti, osim što su koeficijenti uz neparne stepene po kompleksnoj promenljivoj negativne u odnosu na koeficijente filtra propusnika niskih učestanosti.

Na slikama 1 i 3, kao i na slikama 2 i 4, može se uočiti da su simetrične u odnosu na učestanost 0.25. To znači da svakoj nuli filtra propusnika niskih učestanosti f odgovara učestanost $0.5 - f$. To znači da se dobije isti rezultat bez obzira da li se koristi izraz (8) ili (9), s tim da se učestanost komplementarnog filtra izračunavaju korišćenjem sledeće formule.

$$f_{HP} = 0.5 - f_{LP} \quad (10)$$



Slika 3. Primer za $P_H(w)$ za $w = \cos(\pi f)$.



Slika 4. Primer za $M = |H_{HP}(e^{j2\pi f})|$.

Dobijeni rezultati su provereni u Matlabu [10].

6. PROJEKTOVANJE U MATLABU

Nekoliko čebiševljevih polinoma prve vrste neparnog reda, tačnije $T_k(w)/w$, koji su potrebni za ova izračunavanja mogu se zadati na sledeći način

```

T3w = flipplr([-3, 0, 4]);
T7w = flipplr([-7, 0, 56, 0, -112, 0, 64]);
T11w = flipplr([-11, 0, 220, 0, -1232, 0, 2816, ...
0, -2816, 0, 1024]);
T15w = flipplr([-15, 0, 560, 0, -6048, 0, 28800, ...
0, -70400, 0, 92160, 0, -61440, 0, 16384]);
T19w = flipplr([-19, 0, 1140, 0, -20064, ...
0, 160512, 0, -695552, 0, 1770496, 0, ...
-2723840, 0, 2490368, 0, -1245184, 0, 262144]);
T23w = flipplr([-23, 0, 2024, 0, -52624, ...
0, 631488, 0, -4209920, 0, 17145856, 0, ...
-44843008, 0, 76873728, 0, -85917696, ...
0, 60293120, 0, -24117248, 0, 4194304]);
T27w = flipplr([-27, 0, 3276, 0, -117936, ...
0, 1976832, 0, -18670080, 0, 109983744, ...
0, -428654592, 0, 1143078912, ...
0, -2118057984, 0, 2724986880, 0, ...
-2387607552, 0, 1358954496, ...
0, -452984832, 0, 67108864]);
T31w = flipplr([-31, 0, 4960, 0, -236096, ...
0, 5261568, 0, -66646528, 0, 533172224, ...
0, -2870927360, 0, 10827497472, ...
0, -29297934336, 0, 57567870976, 0, ...
-82239815680, 0, 84515225600, 0, -60850962432, ...
0, 29125246976, 0, -8321499136, 0, 1073741824]);
  
```

Komanda `flipplr` je upotrebljena radi jednostavnijeg izračunava zbira više polinoma.

Nekoliko čebiševljevih polinoma druge vrste parnog reda koji su potrebni za ova izračunavanja mogu se zadati na sledeći način

```
U2 = fliplr([-1, 0, 4]);
U6 = fliplr([-1, 0, 24, 0, -80, 0, 64]);
U10 = fliplr([-1, 0, 60, 0, -560, 0, 1792, ...
0, -2304, 0, 1024]);
U14 = fliplr([-1, 0, 112, 0, -2016, 0, 13440, ...
0, -42240, 0, 67584, 0, -53248, 0, 16384]);
U18 = fliplr([-1, 0, 180, 0, -5280, 0, 59136, ...
0, -329472, 0, 1025024, 0, -1863680, ...
0, 1966080, 0, -1114112, 0, 262144]);
U22 = fliplr([-1, 0, 264, 0, -11440, 0, 192192, ...
0, -1647360, 0, 8200192, 0, ...
-25346048, 0, 50135040, 0, -63504384, ...
0, 49807360, 0, -22020096, 0, 4194304]);
U26 = fliplr([-1, 0, 364, 0, -21840, 0, 512512, ...
0, -6223360, 0, 44808192, 0, ...
-206389248, 0, 635043840, 0, -1333592064, ...
0, 1917583360, 0, -1857028096, 0, 1157627904, ...
0, -419430400, 0, 67108864]);
U30 = fliplr([-1, 0, 480, 0, -38080, ...
0, 1188096, 0, -19348992, 0, 189190144, 0, ...
-1203937280, 0, 5239111680, 0, -16066609152, ...
0, 35283533824, 0, ...
-55710842880, 0, 62704844800, 0, -49073356800, ...
0, 25367150592, 0, -7784628224, 0, 1073741824]);
```

Polinom za izračunavanje nula funkcije prenosa može se izračunati korišćenjem sledećeg koda

```
P = a10*fliplr(U18);
P(1:length(U14))=P(1:length(U14)) + a8*fliplr(U14);
P(1:length(U10))=P(1:length(U10)) + a6*fliplr(U10);
P(1:length(U6))=P(1:length(U6)) + a4*fliplr(U6);
P(1:length(U2))=P(1:length(U2)) + a2*fliplr(U2);
P(1) = koef2nule(1) - 1;
P = fliplr(P);
```

Za poznate vrednosti koeficijena filtra, na osnovu prethodnih izraza se izračunava polinom. Komanda `roots` se može iskoristiti da se odrede nule funkcije prenosa.

Ako se projektovanje započinje na osnovu poznatih nula funkcija prenosa, najpre se generiše matrica sa vrednostima polinoma gde su u vrstama vrednosti za jednu učestanost. U tom slučaju se problem izračunavanja koeficijena filtra svodi na sistem linearnih jednačina, što se u Matlabu realizuje operacijom deljenja matrice vrste koja ima jedinične vrednosti i matrice sa vrednostima polinoma za odgovarajuće učestanosti.

7. ZAKLJUČAK

U ovom radu je pokazano kako se koristi softverski alat koji pripada klasi algebarskih računarskih sistema za određivanje osobina digitalnih halfband IIR filtara sa približno linearnom fazom. Projektovanje filtra može da se uradi polazeći od željenih učestanosti na kojima funkcija prenosa ima nulte vrednosti, i tada se koeficijenti izračunavaju rešavanjem sistema linearnih jednačina. U

slučaju kada se polazi od poznatih vrednosti koeficijena allpass filtra (na primer, dobijenih iterativnim postupkom za filtre koji imaju ekviripl karakteristiku, a nakon kvantizacije), za određivanje nula se mogu koristiti standardni programi za nalaženje nula polinoma.

ZAHVALNICA

Ovaj rad je delimično finansiralo Ministarstvo nauke Republike Srbije, Projekat TR13012.

LITERATURA

- [1] H. W. Schüßler and P. Steffen, "Recursive Halfband-Filters," *Int. J. Electron. Commun., (AEU)*, Vol 55, No. 6, pp. 377–388, 2001.
- [2] L. Milic, *Multirate Filtering for digital Signal processing: MATLAB Applications*, Information Science Reference, Hershey PA, 2009.
- [3] Lj. D. Milić and J. D. Čertić, "Recursive Digital Filters and Two-Channel Filter Banks: Frequency-Response Masking Approach," *Telsiks 2009*, pp. 177-184, Oct. 2009.
- [4] S. Wolfram, *The Mathematica Book*, Cambridge: Cambridge, University Press, Wolfram Media, 2003.
- [5] M. D. Lutovac and D. V. Tomic, *SchematicSolver Version 2*, www.schematicsolver.com, 2009.
- [6] M. D. Lutovac and D. V. Tomic, "Symbolic Analysis and Design of Control Systems using Mathematica," *International Journal of Control*, Special Issue on Symbolic Computing in Control, vol.79, no.11, pp.1368–1381, 2006.
- [7] M. D. Lutovac, J. Čertić, and Lj. Milić, "Digital Filter Design using Computer Algebra Systems," *Circuits, Syst. and Signal Process*, Birkhäuser Boston, vol. 29: pp. 51–64. 2010.
- [8] V.M. Poučki, A. Žemva, M. D. Lutovac, T. Karčnik, "Elliptic IIR filter sharpening implemented on FPGA," *Digital Signal Processing*, vol. 20, pp. 13–22, 2010.
- [9] M. D. Lutovac and Lj. D. Milic, Approximate linear phase multiplierless IIR halfband filter, *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 52-53, Mar. 2000.
- [10] MATLAB Version 7, MathWorks, Inc., Natick, MA, 2005.

INDEKS AUTORA

B		J	
Banović Nataša	90	Jevtović Milojko	75, 79
Bauk Sanja	25	Jokić Anđelina	154
Bjelica Boriša	150	Jovanović Ana	147
Bukilić Mladen	94		
Č		K	
Čvorović Jelena	123	Kalezić Milorad	158
		Kasalica Jelica	87
		Kosanović Mirko R.	52
		Kostić Dušan	45
		Kotlaja Sanja	29
		Krdžavac Nenad	64
		Krstajić Božo	1, 5, 17, 33, 37, 135
Ć		L	
Ćalasan Martina	166, 170	Lekić Nedjeljko	60
Ćorović Zoran	104	Lutovac Budimir	139, 158
		Lutovac Miroslav	139, 174
D		LJ	
Daković Miloš	127, 150, 162	Ljumović Nikola	131
Danilović Ruža	25		
Dedić Aleksandar	41	M	
Duli Sidita	135	Marović Branko	45
		Matijević Tripo	83
		Mijanović Zoran	60
		Milivojević Marina	67
		Milivojević Zoran N.	67, 71
		Mirković Marija	154, 158
		Mirković Mimo	127
		Mišović Danilo	115
		Mosurović Milenko	64
		Mrdak Vladana	37
Đ		N	
Đurović Igor	143, 162	Nedić Nebojša	56
F			
Filipović Luka	5		
G			
Golović Kažić Borislavka	101		
I			
Ilić-Delibašić Maja	17		

O		Stojanović Radovan	60, 154, 158
		Strujić Dženan	83
Obadović Jezdimir - Luka	13		
Obradović Milovan	48		
Ognjanović Igor	87	Š	
Ognjanović Ivana	94	Šendelj Ramo	90, 94
Ostojić Milutin	170	Šćepanović Snežana	98, 104
P		T	
Paunović Aleksandar	1		
Pavićević Slobodan	33	Tomić Dajana	111
Pelinković Alen	143		
Penezić Duško	119	U	
Perunović Miloš	21	Ugren Budimir	107
Pravilović Sonja	101		
R		V	
Radonjić Aleksandar	139, 174	Veličković Zoran	75, 79
Radonjić Milutin	107, 115, 123, 131	Vojinović Ranko	87
Radulović Milovan	21, 127	Vučinić Ivan	9
Radulović Novak	104	Vujičić Vladan	166
Radusinović Igor	107, 111, 115, 123, 131	Vukčević Nevena	158
S		Ž	
Savić Nataša	71	Žarković Jurica A.	154
Simeunović Marko	162	Žujović Dražen	5
Stanković Ljubiša	150		
Stojanović Đuro	143		