



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



ЗБОРНИК РАДОВА ФАКУЛТЕТА ТЕХНИЧКИХ НАУКА

Едиција: Техничке науке - зборници

Година: XXXIII

Број: 4/2018

Нови Сад

Едиција: „Техничке науке – Зборници“

Година: XXXIII

Свеска: 4

Издавач: Факултет техничких наука Нови Сад

Главни и одговорни уредник: проф. др Раде Дорословачки, декан Факултета техничких Наука у Новом Саду

Уредништво:

Проф. др Раде Дорословачки

Проф. др Драгиша Вилотић

Проф. др Срђан Колаковић

Проф. др Владимир Катић

Проф. др Драган Шешилија

Проф. др Миодраг Хаџистевић

Проф. др Растислав Шостаков

Доц. др Мирослав Кљајић

Доц. др Бојан Лалић

Доц. др Дејан Убавин

Проф. др Никола Јорговановић

Доц. др Борис Думнић

Проф. др Дарко Реба

Проф. др Ђорђе Лађиновић

Проф. др Драган Јовановић

Проф. др Мила Стојаковић

Проф. др Драган Спасић

Проф. др Драгољуб Новаковић

Редакција:

Проф. др Владимир Катић, главни
уредник

Проф. др Жељен Трповски, технички
уредник

Проф. др Драган Шешилија

Проф. др Драгољуб Новаковић

Др Иван Пинђер

Бисерка Милетић

Језичка редакција:

Бисерка Милетић, лектор

Софија Рацков, коректор

Марина Катић, преводилац

Издавачки савет:

Савет за библиотечку и издавачку делатност ФТН,
проф. др Радош Радивојевић, председник.

Штампа: ФТН – Графички центар ГРИД, Трг Доситеја Обрадовића 6, Нови Сад

CIP-Каталогизација у публикацији
Библиотека Матице српске, Нови Сад

378.9(497.113)(082)
62

ЗБОРНИК радова Факултета техничких наука / главни и одговорни уредник
Раде Дорословачки. – Год. 7, бр. 9 (1974)-1990/1991, бр.21/22 ; Год. 23, бр 1 (2008)-. – Нови Сад :
Факултет техничких наука, 1974-1991; 2008-. – илустр. ; 30 цм. –(Едиција: Техничке науке –
зборници)

Месечно

ISSN 0350-428X

COBISS.SR-ID 58627591

ПРЕДГОВОР

Поштовани читаоци,

Пред вама је четврта овогодишња свеска часописа „Зборник радова Факултета техничких наука“.

Часопис је покренут давне 1960. године, одмах по оснивању Машинског факултета у Новом Саду, као „Зборник радова Машинског факултета“, а први број је одштампан 1965. године. Након осам публикованих бројева у шест година, пратећи прерастање Машинског факултета у Факултет техничких наука, часопис мења назив у „Зборник радова Факултета техничких наука“ и 1974. године излази као број 9 (VII година). У том периоду у часопису се објављују научни и стручни радови, резултати истраживања професора, сарадника и студената ФТН-а, али и аутора ван ФТН-а, тако да часопис постаје значајно место презентације најновијих научних резултата и достигнућа. Од броја 17 (1986. год.), часопис почиње да излази искључиво на енглеском језику и добија поднаслов «Publications of the School of Engineering». Једна од последица нарастања материјалних проблема и несрећних догађаја на нашим просторима јесте и привремени прекид континуитета објављивања часописа двобројем/двогодишњаком 21/22, 1990/1991. год.

Друштво у коме живимо базирано је на знању. Оно претпоставља реорганизацију наставног процеса и увођење читавог низа нових струка, као и квалитетну организацију научног рада. Значајне промене у структури високог образовања, везане за имплементацију Болоњске декларације, усвајање нове и активне улоге студената у процесу образовања и њихово све шире укључивање у стручне и истраживачке пројекте, као и покретање нових мастер и докторских студија, доносе потребу да ови, веома значајни и вредни резултати, постану доступни академској и широј јавности. Оживљавање „Зборника радова Факултета техничких наука“, као јединственог форума за презентацију научних и стручних достигнућа, пре свега студената, обезбеђује услове за доступност ових резултата.

Због тога је Наставно-научно веће ФТН-а одлучило да, од новембра 2008. год. у облику пилот пројекта, а од фебруара 2009. год. као сталну активност, уведе презентацију најважнијих резултата свих мастер радова студената ФТН-а у облику кратког рада у „Зборнику радова Факултета техничких наука“.

Поред студената мастер студија, часопис је отворен и за студенте докторских студија, као и за прилоге аутора са ФТН или ван ФТН-а.

Зборник излази у два облика – електронском на веб сајту ФТН-а (www.ftn.uns.ac.rs) и штампаном, који је пред вама. Обе верзије публикују се сваки месец, у оквиру промоције дипломираних мастера.

У овом броју штампани су радови студената мастер студија, сада већ мастера, који су радове бранили у периоду од 22.09.2017. до 31.10.2017. год., а који се промовишу 22.03.2018. год. То су оригинални прилози студената са главним резултатима њихових мастер радова.

Известан број кандидата објавили су радове на некој од домаћих научних конференција или у неком од часописа. Њихови радови нису штампани у Зборнику радова.

Велик број дипломираних инжењера–мастера у овом периоду био је разлог што су радови поводом ове промоције подељени у три свеске.

У овој свесци, са редним бројем 4., објављени су радови из области:

- машинства,
- електротехнике и рачунарства и
- мехатронике.

У свесци са редним бројем 5. објављени су радови из области:

- грађевинарства,
- саобраћаја,
- графичког инжењерства и дизајна,
- архитектуре,
- инжењерства заштите животне средине и
- инжењерства информационих система.

У свесци са редним бројем 6. објављени су радови из области:

- инжењерског менаџмента,
- геодезије и геоматике,
- регионалне политике и развоја и
- инжењерства третмана и заштите вода (TEMPUS).

Уредништво се нада да ће и професори и сарадници ФТН-а и других институција наћи интерес да публикују своје резултате истраживања у облику регуларних радова у овом часопису. Ти радови ће бити објављивани на енглеском језику због пуне међународне видљивости и проходности презентованих резултата.

У плану је да часопис, својим редовним изласком и високим квалитетом, привуче пажњу и постане довољно препознатљив и цитиран да може да стане раме-уз-раме са водећим часописима и заслужи своје место на СЦИ листи, чиме ће значајно допринети да се оствари мото Факултета техничких наука:

„Високо место у друштву најбољих“

Уредништво

SADRŽAJ

STRANA

Radovi iz oblasti: Mašinstvo

1. Mirjana Ristić, UTICAJ TEMPERATURE RADNE PLOČE 3D ŠTAMPAČA NA PROCENAT KRISTALNOSTI PLA UZORAKA IZRAĐENIH FDM TEHNOLOGIJOM	477-480
2. Nikola Moroz, Sebastian Baloš, RUČNO – ELEKTROLUČNO ZAVARIVANJE RUTILNIM ELEKTRODAMA SA MODIFIKOVANOM OBLOGOM	481-484
3. Obrad Mirković, Zvonko Rakarić, PASIVNI VIBROIZOLATOR ZA SEIZMIČKI OPTEREĆENU KONSTRUKCIJU SA PREDNAPREGNUTIM ELEMENTOM	485-488
4. Aleksandar Petković, Mladomir Milutinović, EKSPERIMENTALNA PROVERA UTICAJA KOEFICIJENTA TRENJA NA PARAMETRE PROCESA DUBOKOG IZVLAČENJA	489-492
5. Roland Kuti, Borislav Savković, Nenad Kulundžić, SIMULACIJA OBRADE GLODANJEM POMOĆU METODA KONAČNIH ELEMENATA	493-496
6. Ilija Vlahović, DISKRETIZACIJA NAVIJE-STOKSOVE JEDNAČINE PRIMENOM METODE KONAČNIH ZAPREMINA	497-500
7. Danilo Dujović, Pavel Kovač, Nenad Kulundžić, PRIMENA VISOKOBRZINSKOG GLODANJA U IZRADI ALATA ZA BRIZGANJE	501-504
8. Nemanja Spasojević, Sebastian Baloš, ODREĐIVANJE VELIČINE GREŠKE TIPA TUNELA PRI PRERADI TRENJEM SA MIJEŠANJEM	505-508

Radovi iz oblasti: Elektrotehnika i računarstvo

1. David Vuletić, KONTROLA SLOŽENOSTI LOZINKI ZA WINDOWS OPERATIVNE SISTEME	509-512
2. Lazar Prodanović, RAZVOJ GENERATORA ZA OPTIMALNO KONFIGURISANJE MODELA DISTRIBUTIVNE MREŽE .	513-516
3. Mihailo Vasiljević, PROGRAMSKI JEZIK R	517-520
4. Mirko Ivanić, ODREĐIVANJE UDALJENOSTI IZMEĐU ČVOROVA BEŽIČNE SENZORSKE MREŽE NA BAZI INDIKATORA SNAGE PRIMLJENOG SIGNALA (RSSI)	521-524

	STRANA
5. Zorana Babić, PRIMENA HOMOMORFNE KRIPTOGRAFIJE U INDUSTRIJSKIM KONTROLNIM SISTEMIMA .	525-528
6. Kristina Vujkov, Veran Vasić, Đura Oros, PRORAČUN KAVEZNOG ASINHRONOG MOTORA	529-532
7. Немања Миладиновић, ПРОНАЛАЖЕЊЕ РАЊИВОСТИ У СОФТВЕРУ НА ОСНОВУ ДИЈАГРАМА ТОКА ПОДАТАКА	533-536
8. Nino Ignjić, PRIMENA FOTONAPONSKIH ELEKTRANA U NEMAČKOJ	537-540
9. Sreten Petrović, Veran Vasić, Đura Oros, ELEKTROMOTORNI POGONI I NAČINI STARTOVANJA MOTORA U FABRICI SOLI	541-544
10. Tara Delari, MODELOVANJE NADZEMNOG VODA, KABLA I RL SEKCIJE	545-548
11. Stevan Cvijanović, MPEG-DASH STANDARD ZA DINAMIČKI PRILAGODLJIVI PRENOS VIDEO SADRŽAJA PREKO HTTP-A	549-553
12. Stefan Mićović, Željen Trpovski, PREGLED TRŽIŠTA MOBILNOG PRENOSA PODATAKA U SRBIJI I OKRUŽENJU	554-557
13. Igor Halkić, Veran Vasić, Đura Oros, PREDLOG TERMIČKOG MODELA ASINHRONOG MOTORA	558-561
14. Dragan Šeatović, Željen Trpovski, AUTONOMNA VOZILA	562-565
15. Milica Vulić, ŠIFARSKI SISTEM ZA RAZMENU SMS PORUKA	566-569
16. Danijel Popović, ANDROIDANNOTATIONS OKVIR ZA RAZVOJ MOBILNIH APLIKACIJA	570-573
17. Владимир Поповић, РАЧУНАРСКИ ПОДРЖАНО УПРАВЉАЊЕ ПРОЦЕСОМ УПИСА КАНДИДАТА НА СТУДИЈСКЕ ПРОГРАМЕ	574-577
18. Predrag Krža, REALIZACIJA MODBUS I MODBUS SUNSPEC SERVERA NA XILINX ZYNQ PLATFORMI	578-581
19. Jovan Knežević, Vladimir Katić, TERMOVIZIJSKA DIJAGNOSTIKA ELEMENATA U RAZVODNIM ORMANIMA	582-585
20. Ана Лукић, ПРОЈЕКТОВАЊЕ МОБИЛНЕ АПЛИКАЦИЈЕ ЗА ЕВИДЕНЦИЈУ СТАЊА У АЗИЛИМА ПАСА У ПРОГРАМСКОМ ЈЕЗИКУ JAVA	586-589
21. Njegoš Janković, DINAMIČKA ANALIZA RADA MIKROMREŽE	590-593
22. Vladan Stevanović, Vladimir Katić, VERIFIKACIJA POTENCIJALA VETRA U VOJVODINI	594-597
23. Dajana Mitrović, MODELOVANJE TOPLOTNE PUMPE ZA RAD U REŽIMU GRIJANJA	598-601
24. Biljana Durlević, OPTIMALNA REKONFIGURACIJA TOPOLOGIJE KAO METODA ZA UPRAVLJANJE ZAGUŠENJIMA U PRENOSNOJ MREŽI	602-605
25. Rajko Ilić, VEB APLIKACIJA ZA ORGANIZOVANJE FUDBALSKOG TAKMIČENJA	606-609
26. Radisav Ristić, Vladimir Katić, STANICE ZA PUNJENJE ELEKTRIČNIH AUTOMOBILA NA TRASI SUBOTICA-BAR	610-613
27. Aleksandra Mitrović, RAZVOJ MOBILNE IGRE ZA REHABILITACIJU RUČNOG ZGLOBA	614-617

	STRANA
28. Dušan Đumić, APLIKACIJA ZA UPRAVLJANJE AUTONOMNIM VOZILOM ZASNOVANA NA ANDROID I ARDUINO PLATFORMAMA	618-621
29. Bojan Baltić, JEDNO REŠENJE PRIMENE VEŠTAČKIH NEURONSKIH MREŽA U SISTEMIMA INTERNET STVARI	622-625
30. Anđelka Dražić, Đura Oros, ZAŠTITA VISOKONAPONSKOG TRANSFORMATORA	626-629
31. Stevan Filipović, BRZA SELEKCIJA KODNIH JEDINICA ZA INTRA MOD SCC EKSTENZIJE HEVC KODERA	630-633
32. Marko Mandić, Živko Bojović, KREIRANJE VEB APLIKACIJE ZA BUDŽETIRANJE TROŠKOVA U . NET TEHNOLOGIJI KORIŠĆENJEM C# JEZIKA	634-637
33. Ilija Pantić, PRIMJENA NETWORK PROTECTORA U UPETLJANIM DISTRIBUTIVNIM MREŽAMA I MREŽAMA SA DISTRIBUTIVNIM GENERATORIMA	638-641
34. Dragoljub Ilić, GENERATOR KODA ZA SISTEME ZA DINAMIČKO UPRAVLJANJE SADRŽAJEM	642-645
35. Đorđe Bogdanović, UPRAVLJANJE I NADZOR RADA GENERATORA U DSTRIBUTIVNIM MREŽAMA	646-649
36. Ivan Bogosavljević, INTEGRISANI IZVOR REFERENTNOG NAPONA VREDNOSTI 0.6 V U 55nm CMOS TEHNOLOGIJI	650-653

Radovi iz oblasti: Mehatronika

1. Саво Ђокић, УПРАВЉАЊЕ КОНТРОЛОМ ПРИСТУПА НА ЗАТВОРЕНИМ ПАРКИНЗИМА	654-657
2. Marko Markov, PREDLOG REŠENJA POLUAUTONOMNOG UPRAVLJANJA NA ADAS PLATFORMI	658-661
3. Nikola Pucar, PROJEKTOVANJE HIDRAULIČNE POLUPLATFORME ZA KIPOVANJE DRUMSKIH TRANSPORTNIH SREDSTAVA SA RASTRESITIM TERETOM	662-665
4. Nemanja Đekić, RAZVOJ DESKTOP APLIKACIJA KOJE MOGU DA SE IZVRŠAVAJU NA RAZLIČITIM OPERATIVNIM SISTEMIMA	666-669
5. Jelena Raca, APLIKACIJA ZA UNOS I KNJIŽENJE BANKARSKIH SPECIFIKACIJA	670-674
6. Atila Smieško, APLIKACIJA ZA NAPLATU PARKIRANJA NA OTVORENIM PARKING PROSTORIMA.....	675-678

UTICAJ TEMPERATURE RADNE PLOČE 3D ŠTAMPAČA NA PROCENAT KRISTALNOSTI PLA UZORAKA IZRAĐENIH FDM TEHNOLOGIJOM**INFLUENCE OF HEATED BED TEMPERATURE ON CRYSTALLINITY IN FDM - BUILT PLA SPECIMENS**

Mirjana Ristić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – PROIZVODNO MAŠINSTVO

Kratak sadržaj – U okviru eksperimentalnog dela rada izvršena je analiza uticaja tehnoloških parametara na procenat kristalnosti uzoraka izrađenih od PLA (Polylactic Acid) materijala FDM tehnologijom (Fused Deposition Modelling). Za trinaest uzoraka je, na osnovu tabele plana eksperimenta, DSC analizom utvrđen procenat kristalnosti. Statistička analiza glavnih uticaja i interakcija omogućena je primenom DSD (Definitive Screening Design) faktornog eksperimenta. Rezultati su pokazali dominantan uticaj debljine sloja i temperature ekstrudiranja na procenat kristalnosti.

Ključne reči: FDM, temperatura ploče, temperatura ekstrudiranja, brzina ekstrudiranja, debljina sloja, kristalnost.

Abstract – In this paper, the influence of layer thickness, heated bed temperature, extrusion temperature and extrusion speed on the percentage of crystallinity in APLA (Advanced Polylactic Acid) specimens built by FDM, was experimentally investigated. Samples were analyzed using DSC thermal analysis method. Statistical analysis was conducted using DSD (Definite Screening Design). Experimental results showed dominant influence of layer thickness and extrusion temperature on crystallinity.

Key words: FDM, layer thickness, heated bed temperature, extrusion temperature, extrusion speed, crystallinity.

1. UVOD

Tehnologija ekstrudiranja materijala, u koju spada FDM, od početka dvehiljaditih preuzela je primat u pogledu zastupljenosti među brojnim tehnologijama aditivne proizvodnje. Prema podacima Wohlers asocijacije [1], u 2003. godini, kompanija Stratasys je objavila da je broj prodatih FDM štampača bio gotovo jednak ukupnom broju prodatih tipova mašina za sve ostale aditivne tehnologije. Međutim, pravu popularnost, FDM tehnologija je počela da beleži od 2007. godine, kada je započeo RepRap projekat gradnje open-sors FDM štampača koji su po ceni bivali sve pristupačniji širokom broju entuzijasta, ali i profesionalaca iz domena inženjerstva, umetnosti, dizajna, itd.

Tehnološki parametri, pre svega temperature ekstrudiranja i uslovi hlađenja, imaju direktan uticaj na kristalnost materijala, a samim tim i na njegove mehaničko-fizičke karakteristike. Iz tog razloga je važno ostvariti kontrolu

nad samim tehnološkim procesom, pri čemu toplotna analiza ponašanja polimernih materijala predstavlja jedan od ključnih alata. U nastavku je dat kratak pregled literature koja se odnosi na predmet istraživanja ovog rada.

Wang i Gramlich [2] su izučavali uticaj tehnoloških parametara štampe (temperatura radne ploče i debljina sloja) na čvrstoću štampanih PLA uzoraka. Analiziran je procenat kristalnosti štampanih PLA uzoraka (DSC, XRD, POM) i uticaj na morfološke, hemijske osobine (SEM, SEC). Rezultati FDM ekstrudiranja su poređeni sa konvencionalnim brizganjem uzoraka od istog materijala. Rezultati su pokazali da temperature radne ploče koja se nalazi znatno iznad temperature stakljenja PLA plastike, dovodi do najveće čvrstoće, i da utiče na povećanje procenta kristalnosti.

Tabi i ostali [3] su ispitivali uticaj termičkog postprocesiranja injekciono livene PLA, na kristalnu strukturu. Za potrebe eksperimenta korišćene su izlivenne ploče od PLA plastike, debljine 2 mm, a potom je izvršeno postprocesiranje, tj. rekristalizacija u konvencionalnim pećima, na temperaturama od 60 do 140°C, i vremenskim intervalima od 10 do 60 minuta, kako bi se postigli različiti procenati kristalnosti. Nakon toga su PLA ploče ispitane diferencijalnom skenirajućom kalorimetrijom (DSC), dinamičko mehaničkom analizom (DMA), i širokougaonom rendgenskom difrakcijom (WAXD).

2. KRISTALNOST POLIMERA**2.1 Definicija**

Jedna od veoma važnih karakteristika polimernih materijala jeste stepen kristalnosti, koji predstavlja procentualni udeo kristalnih područja u odnosu na amorfnu fazu. Udeo kristalne strukture predstavlja jednu od važnijih karakteristika svakog polukristalnog termoplasta, budući da ostvaruje značajan uticaj na mehaničke i fizičke osobine. Polimeri prema svojoj uređenosti mogu biti amorfni i kristalni, te tako i stepen kristalnosti može da varira od potpuno amorfne, do 95% kristalne strukture. Procenat kristalnosti se izračunava na osnovu sledeće jednačine (1)

$$C(\%) = \frac{(\Delta H_m - \Delta H_c)}{\Delta H_{mo}} \times 100 \quad (1)$$

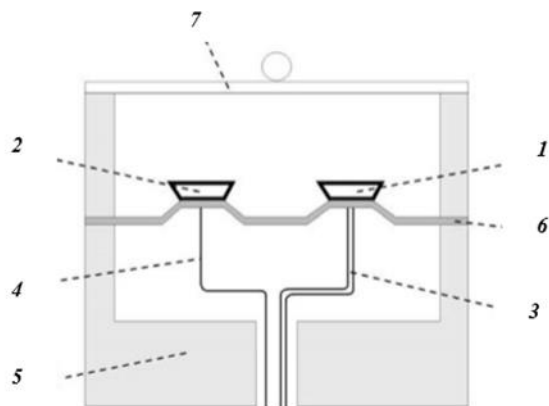
2.2 Merenje procenta kristalnosti polimera

Diferencijalna skenirajuća kalorimetrija (DSC - *Differential Scanning Calorimetry*) je tehnika koja registruje energetske fluks potreban za održavanje nulte temperaturne razlike između uzorka i referentnog materijala, pri unapred definisanoj brzini grejanja (hlađenja), uz pretpostavku da se oba materijala nalaze pod istim uslovima. DSC se koristi

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Ognjan Lužanin, vanr. prof.

kao kvantitativna merna metoda koja pruža mogućnost određivanja stepena kristalnosti, temperature stakljenja temperature topljenja i ključanja, promene entalpije i entropije, i druge efekte usled kojih dolazi do promena toplotnog kapaciteta ili latentne toplote [4]. Metoda se zasniva na merenju temperaturne razlike: ona određuje intenzitet razmene toplote između termoelektričnog diska i uzorka - referentnog dela. Dobijeni toplotni protok je proporcionalan razlici temperature [5]. Tipična konstrukcija DSC uređaja data je na slici 1.



Slika 1. Šema uređaja za DSC analizu
1- posuda sa uzorkom, 2- posuda sa referentnim materijalom, 3- termopar za uzorak, 4- termopar za referentni materijal, 5- grejno telo, 6- termoelektrični disk, 7- poklopac.

3. PLAN I REALIZACIJA EKSPERIMENTA

3.1 Dizajn eksperimenta

Eksperimentalni rad je organizovan prema principima inženjerskog faktornog eksperimenta. S tim u vezi, korišćen je DSD (*Definitive Screening Design*) plan faktornog eksperimenta. Prema randomizovanom redosledu, uzorci su odštampani na 3D štampaču. Izabrani faktori i odgovarajuće vrednosti nivoa dati su u Tabeli 1.

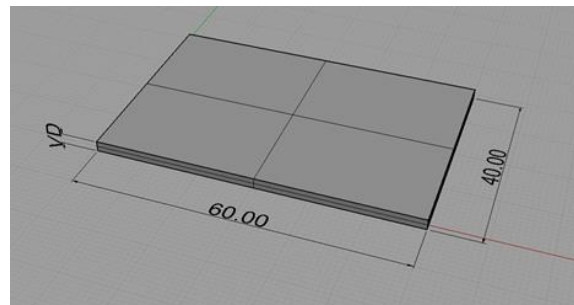
Tabela 1. Eksperimentalni faktori, nivoi i eksperimentalno izmerene vrednosti procenta kristalnosti

	Temp.ploče	Temp. ekstrudiranja	Brzina ekstrudiranja	Debljina sloja	Procentat kristalnosti	Uzorak
1	65	210	30	0.1	34.1	#4
2	80	230	90	0.1	35.7	#5
3	50	210	30	0.3	36.4	#6
4	50	230	30	0.1	33.5	#8
5	65	230	90	0.3	36.8	#9
6	80	230	30	0.2	32.5	#1
7	50	230	60	0.3	33.8	#2
8	50	220	90	0.1	34.3	#3
9	50	210	90	0.2	33.7	#10
10	80	210	60	0.1	33.2	#12
11	80	210	90	0.3	36.1	#11
12	65	220	60	0.2	33.8	#7
13	80	220	30	0.3	40.3	#13

3.2 Geometrija uzoraka

Za realizaciju eksperimenta modelirana su dva uzorka prizmatičnog oblika, koji su po dimenzijama osnove bili identični, 60x40 mm, samo su im se razlikovale debljine, označene sa VD, na slici 2. Kod uzorka A, debljina je

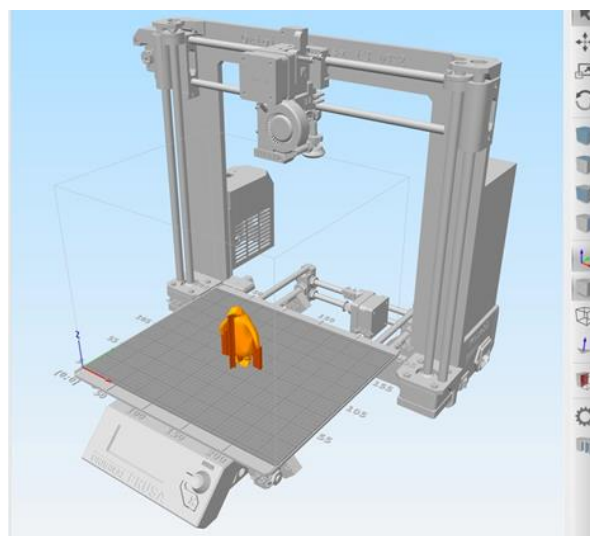
iznosila $VD=0.9$ mm i on je korišćen u eksperimentima se debljinama slojeva od 0.1 ili 0.3 mm. Kod uzorka B, debljina je iznosila $VD=1$ mm, i ovaj model je bio namenjen eksperimentima sa debljinom sloja 0.2 mm. Time je omogućeno da u svim eksperimentima, bez obzira na debljinu sloja, količnik debljine uzorka i debljine sloja bio ceo broj.



Slika 2. CAD model osnovne geometrije korišćene za FDM uzorke koji su izrađeni u eksperimentu

3.3 3D štampa uzoraka

U eksperimentu je, za izradu uzoraka od PLA (*Poly-Lactic Acid*) na osnovu tabele eksperimenta, korišćen 3D štampač Prusa i3Mk2. Ovaj štampač spada u 3D štampače personalne klase i raspolaže naprednim karakteristikama. CAD model štampača je prikazan na slici 3. Radi se o Dekartovom tipu konstrukcije, sa upravljanjem duž tri koordinatne ose, koje su na slici x obeležene bojama (X osa - crvena, Y osa - zelena i Z osa - plava).



Slika 3. CAD model 3D štampača Prusa i3 MK2 koji je korišćen u eksperimentu

Maksimalna zapremina radnog prostora ovog štampača jeste 250x210x200 mm. Maksimalna temperatura ekstrudera iznosi 280 °C. Zahvaljujući grejanju radne ploče, čija temperatura može ići do 110 °C, ovaj štampač je u mogućnosti da radi sa širim dijapazonom materijala, u koje spadaju:

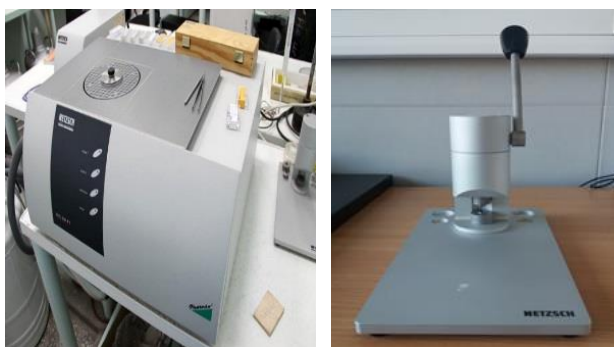
- ABS
- PLA
- Poliamid (najlon)
- PET
- FLEX

- Razne vrste kompozitnih materijala, kopolimera sa udelom staklenih, ugljeničnih vlakana, itd.

Štampač takođe raspolaže procedurom za autokalibraciju ravnosti radne ploče, tako da, za razliku od većine drugih štampača u ovoj klasi, ne zahteva ručno poravnavanje ploče pre početka rada.

4. DSC ANALIZA PLA UZORAKA

Za ispitivanje toplotnih svojstava trinaest odštampanih PLA uzoraka, korišćena je diferencijalno skenirajuća kalorimetrija (DSC). Za DSC analizu je korišćen uređaj NETSCH, model 204F1 Phoenix (sl.4). Takođe su korišćene specijalne aluminijumske posudice, koje obezbeđuju optimalnu toplotnu provodljivost i minimalnu reakciju sa uzorcima. Mase trinaest pripremljenih uzoraka su izmerene na analitičkoj vagi, a nakon toga su posudice hermetički zatvorene korišćenjem specijalne prese (slika 4).



Slika 4. Levo - DSC uređaj NETSCH 204F1 Phoenix i Desno - presa za hermetičko zatvaranje

Pripremljeni DSC uzorci su prikazani na slici 5. Toplotna analiza DSC uzoraka od termoplastičnog polimera izvedena je u temperaturnom intervalu od 25 °C do 200 °C. Merenje je vršeno pri linearnom povećanju temperature koje se odvijalo brzinom od 10 °C u minuti.



Slika 5. Uzorci za DSC analizu koji su izvađeni iz središnje sekcije FDM uzorka pod odgovarajućim brojem

Pomoću indijuma je izvršena standardna kalibracija, znajući da je temperatura topljenja ovog materijala 157 °C. Prvo dinamičko zagrevanje, koje se odvija ravnomernom brzinom sve do temperature od 200°C, sprovodi se u cilju uklanjanja termalne istorije; potom sledi proces hlađenja, brzinom od 20°C u minuti, do temperature od 25 °C, proces se potom održava na konstantnoj temperaturi 5 minuta i zagreva se po drugi put. Tokom hlađenja, uzorak ima dovoljno vremena da kristališe, što se na DSC dijagramu može jasno uočiti, vidljivim pikom kristali-

zacije (sl. 6). U ovom radu uzorci su prošli samo prvo dinamičko zagrevanje, budući da je cilj bio da se registruje toplotna istorija koja je nastala FDM rocesiranjem. Na slici 7 je prikazan superponirani dijagram na kojem se vide DSC krive za trinaest FDM uzoraka koji su procesirani na osnovu tabele eksperimenta (Tab. 1).

Karakteristične toplotne tranzicije koje su detektovane DSC analizom, prikazane su na DSC dijagramima. Radi se o sledećim tranzicijama:

- Prelaz u staklasto stanje koje karakteriše temperatura stakljenja (T_g);
- Kristalizacija koju karakteriše temperatura kristalizacije (T_k);
- Entalpija kristalizacije (ΔH_{cc}); i topljenje koje karakteriše temperatura topljenja i entalpija topljenja (ΔH_m).

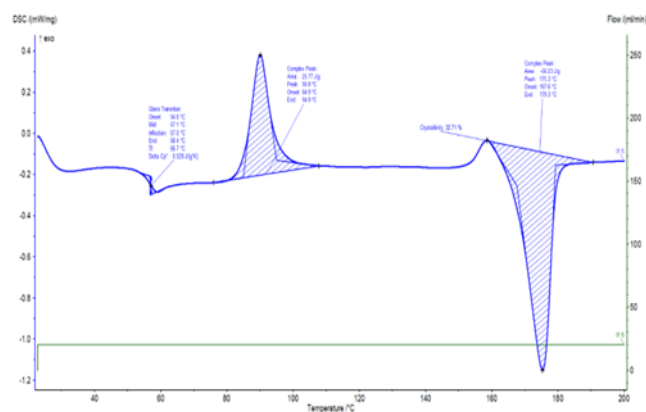
Procenti kristalnosti su prikazani u tabeli eksperimenta, u prethodnoj koloni (Tab. 1).

5. ANALIZA REZULTATA

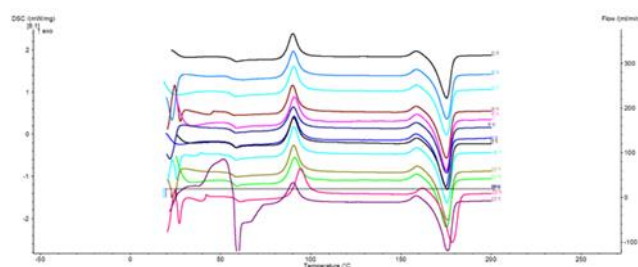
Na osnovu eksperimentalno dobijenih rezultata, formiran je regresioni model koji uspostavlja zavisnost između procenta kristalnosti i vrednosti ulaznih tehnoloških parametara 3D štampe. Vrednosti aproksimiranih parametara modela date su u tabeli 2, dok su rezultati analize varijanse prikazani u tabeli 3.

U okviru odabranog regresionog modela (Tab.2), faktor brzine ekstrudiranja ne figuriše, dok faktori temperatura ploče i temperatura ekstrudiranja nisu statistički signifikantni. Slede statistički značajni koeficijenti regresione jednačine, a to su: debljina sloja ($p=0.0150$), kvadratni član debljine sloja ($p=0.0108$) i kvadratni član temperature ekstrudiranja ($p=0.0451$) (Tab. 2).

Iz ANOVA tabele (Tab. 3) se vidi da je usvojeni regresioni model statistički značajan ($p=0.0314$).



Slika 6. DSC dijagram dobijen za uzorak #1



Slika 7. Superponirani dijagram sa trinaest DSC krivih

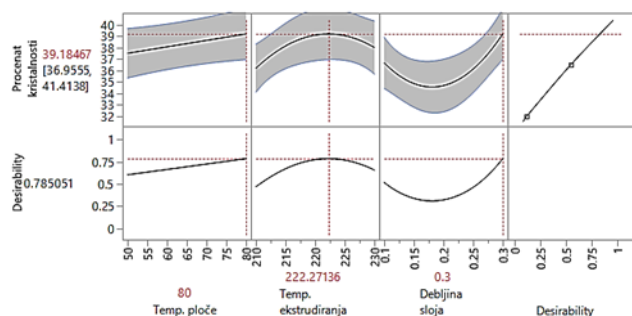
Na slici 8 prikazan je dijagram optimizacije, koji je generisan na osnovu analize metodom odzivne površi (*response surface method*). Sa dijagrama se vidi da je teoretski optimirani procenat kristalnosti približno 39.2%, pri čemu se date i optimirane vrednosti ulaznih tehnoloških parametara.

Tabela 2. Izračunati koeficijenti regresije i njihova statistička značajnost

Term	Estimate	Std Error	t Ratio	Prob> t
Intercept	33.975472	0.903369	37.61	<.0001*
Temp. ploče(50,80)	0.61	0.373527	1.63	0.1536
Temp. ekstrudiranja(210,230)	-0.12	0.373527	-0.32	0.7589
Debljina sloja(0.1,0.3)	1.26	0.373527	3.37	0.0150*
Temp. ploče*Temp. ekstrudiranja	1.0216981	0.473036	2.16	0.0741
Temp. ekstrudiranja*Temp. ekstrudiranja	-1.984906	0.786536	-2.52	0.0451*
Debljina sloja*Debljina sloja	3.2367925	0.888679	3.64	0.0108*

Tabela 3. ANOVA tabela

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	6	44.179411	7.36324	5.2774
Error	6	8.371358	1.39523	Prob > F
C. Total	12	52.550769		0.0314



Slika 8. Dijagram optimizacije sa teoretskim maksimumom koji iznosi 39.185%

6. ZAKLJUČCI

U ovom radu je realizovano istraživanje uticaja temperature zagrevanja radne ploče štampača, brzine i temperature ekstrudiranja i debljine sloja, na procenat kristalnosti uzoraka koji su izrađeni u FDM tehnologiji. Izvedeno je ukupno trinaest eksperimenata koji su omogućili sistematsko ispitivanje uticaja parametara i njihove interakcije, na ponašanje zavisne promenljive, procenta kristalnosti.

Rezultati koji su dobijeni na osnovu eksperimenta pokazuju da, pri korišćenom izboru tehnoloških parametara, debljina sloja ($p=0.015$) i njen kvadratni član ($p=0.0108$) imaju dominantan, statistički značajan uticaj na vrednost kristalnosti. Pored toga, kvadratni član temperature ekstrudiranja takođe je statistički značajan ($p=0.041$).

Optimizacijom na osnovu analize odzivne površi, pri odabranim graničnim uslovima, tj. intervalima vrednosti za sva četiri tehnološka parametra, prikazano je da se može dobiti maksimalna vrednost kristalnosti od približno 39%. Kada se ova vrednost upoređi sa procentom kristalnosti filameta od 26.56%, pre procesiranja na 3D štampaču, zaključuje se da dobijeno povećanje kristalnosti iznosi približno 32%.

Značajno je primetiti da je maksimalna vrednost procenta kristalnosti dobijena pri kombinaciji vrednosti tehnoloških parametara koji nisu prethodno definisani u tabeli eksperimenata (temperatura ekstrudiranja 222 °C). Zbog toga su neophodni dopunski eksperimenti za verifikaciju optimuma. Potrebno je izvesti eksperimente sa teoretski dobijenim, optimalnim vrednostima ulaznih parametara, pri čemu potreban broj uzoraka treba utvrditi na osnovu zahtevane moći testa

7. LITERATURA

- [1] <https://wohlersassociates.com/blog/2016/01/-popularity-of-fdm/>
- [2] Tábi, T., Sajó, I. E., Szabó, F., Luyt, A. S. and Kovács, J. G. (2010) 'Crystalline structure of annealed polylactic acid and its relation to processing', *Express Polymer Letters*, 4(10), pp. 659–668. doi: 10.3144/expresspolymlett.2010.80.
- [3] Wang L, Gramlich WM, Gardner DJ, Improving the impact strength of Poly(lactic acid) (PLA) in fused layer modeling (FLM), *Polymer* (2017), doi: 10.1016/j.polymer.2017.03.011
- [4] V.A.Lihachev, AE Volkov, V. Shudegov, Continuum theory of defects, LGU, 1986
- [5] Materials Measurement Methods, Ed: Czichos, H., Saito, T., Smith, L., Springer, 2006; pp. 411–413

Adrese autora za kontakt:

MSc Mirjana Ristić – mira051193@gmail.com

**RUČNO – ELEKTROLUČNO ZAVARIVANJE RUTILNIM ELEKTRODAMA SA
MODIFIKOVANOM OBLOGOM****SHIELDED METAL ARC WELDING WITH RUTILE ELECTRODES WITH MODIFIED
COATING**Nikola Moroz, Sebastian Baloš, *Fakultet tehničkih nauka, Novi Sad***Oblast – PROIZVODNO MAŠINSTVO**

Kratak sadržaj – Cilj ovog rada jeste da prikaže uticaj nanočestica TiO_2 na karakteristike zavarenog spoja, ostvarenog ručnim-elektrolučnim zavarivanjem niskougljeničnog čelika. Izvršeno je ispitivanje mehaničkih osobina, mikrostrukture i hemijskog sastava. Pri procesu zavarivanja korišćene su obložene rutilne elektrode sa infiltriranim nanočesticama TiO_2 . Rezultati su upoređivani sa kontrolnim uzorkom koji je zavaren elektrodom bez nanočestica. Na osnovu rezultata utvrđeno je da prisustvo nanočestica pri optimalnim parametrima povoljno utiče na mehaničke osobine i mikrostrukturu zavarenog spoja.

Abstract – The point of this study is to show influence of TiO_2 nano particles on characteristics of joints, welded with shielded metal arc welding process of low carbon steel. Mechanical properties, microstructure and chemical composition were determined. For the process of welding, rutile electrodes were used, infiltrated with TiO_2 nano particles. Results were compared with control sample, welded with non infiltrated rutile electrode. Results have shown that the presence of TiO_2 nano particles influence on increase in mechanical properties and refines the microstructure of welded joint with optimal infiltration parameters.

Ključne reči: Ručno-elektrolučno zavarivanje, obložene rutilne elektrode, nanočestice TiO_2 , mikrostruktura, mehaničke osobine.

1. UVOD

Ručno-elektrolučno zavarivanje predstavlja postupak zavarivanja topljenjem uz prisustvo dodatog materijala u vidu topljive obložene elektrode. Izvor toplote predstavlja električni luk koji se obrazuje između osnovnog materijala i elektrode. Obložene elektrode se sastoje iz metalnog jezgra i obloge koje mogu imati različiti hemijski sastav u zavisnosti od upotrebe [1]. Zavarivanjem niskougljeničnih čelika unutar metala šava se stvara mikrostruktura od alotriomorfog, Widmanstätten-ovog i igličastog ferita. Pored ovoga, može da se pojavi i martenzit, zaostali austenit i perlit. Na poboljšanje mehaničkih osobina metala šava se utiče putem izbora elektrode, odnosno sastava njene obloge. Odabirom odgovarajuće obloge može se u velikoj meri uticati na pojavu odgovarajuće

mikrostrukture, pretežno od iličastog ferita, čime se poboljšavaju mehaničke osobine metala šava i omogućava dostizanje optimalnog odnosa čvrstoće i žilavosti. Suštinski mehanizam ojačavanja koji se postiže igličastim feritom, u odnosu na Vidmanštetenov ili alotriomorfni, jeste ojačavanje granicama zrna [2]. Uticaj na formiranje igličastog ferita, može imati dodatak oksida titana (TiO , Ti_2O_3 i TiO_2), koji imaju uticaj na efikasnu inokulaciju i formiranje većeg broja zrna tokom kristalizacije. Ti se oksidi mogu uneti preko obloge elektrode [3].

Cilj ovog istraživanja je ispitivanje uticaja infiltriranih nano čestica TiO_2 na poboljšanje mikrostrukture i mehaničkih osobina metala šava ručnim-elektrolučnim zavarivanjem obloženom rutilnom elektrodom.

2. EKSPERIMENTALNI RAD

U ovom radu, korišćen je osnovni materijal u vidu konstrukcionog čelika hemijskog sastava prikazanog u tabeli 1. Debljina lima je bila 4 mm. Kao dodatni materijal, korišćene su rutilne obložene elektrode Elvacor EL58C prečnika jezgra $d_e=2,5$ mm. Za infiltraciju obloge elektrode korišćene su hidrofilne nanočestice Aerioxide TiO_2 P25. U tabeli 2 su prikazane mehaničke osobine osnovnog materijala.

Tabela 1. Hemijski sastav osnovnog materijala u %.

C	Si	Mn	S	Cr	P
0,11	0,16	0,52	0,039	<0,002	0,011
Al	Cu	Mo	Ni	Ti	Fe
0,006	0,35	0,012	0,083	<0,002	ostatak

Tabela 2. Mehaničke osobine osnovnog materijala.

Napon tečenja R_{eH} [MPa]	Zatezna čvrstoća R_m [MPa]	Izduženje A [%]
339	457	28

Eksperiment je izveden sa četiri grupe elektroda. Tri grupe elektroda su infiltrirane nanočesticama TiO_2 , dok je jedna grupa kontrolna, bez infiltracije:

I grupa – bez infiltracije (R0).

II grupa – vreme infiltracije 1 min (R1);

III grupa – vreme infiltracije 5 min (R5);

IV grupa – vreme infiltracije 10 min (R10);

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Sebastian Baloš.

Pre infiltracije nanočestica u oblogu elektrode, bila je neophodna priprema nanočestica. Nanočestice su prvo stavljene u staklenu posudu, u količini od 5 mg, a zatim je

dodato 95 ml destilovane vode. Nakon toga je izvršeno njihovo mešanje u ultrazvučnoj kadi Emmi-5 Emag u odgovarajućem trajanju. U tako pripremljenu homogenu disperziju su potapane elektrode i infiltrirane u zavisnosti od predviđenog vremena. Pripremljene elektrode su nakon infiltracije stavljene u peć za sušenje Instrumentaria ST-05, gde su sušene na temperaturi od 100°C u trajanju od 60 min. Nakon sušenja, elektrode su ostavljene u peći, kako bi se postepeno hladile do sobne temperature.

Pripremljeno je ukupno osam pločica osnovnog materijala dimenzija 150x40x4 mm, od toga šest za zavarivanje modifikovanim elektrodama a dve za zavarivanje kontrolnom elektrodom. Uzorci su obeleženi u zavisnosti od toga kojom grupom elektroda su zavarivani, odnosno od vremena infiltracije.

Zavarivanje je izvršeno postupkom ručnog-elektrolučnog zavarivanja obloženom elektrodom u dva prolaza. Spoj je izveden kao sučeon sa pripremom ivica u vidu I – šava sa razmakom između limbova od 2,5 mm što odgovara prečniku jezgra elektrode. Položaj zavarivanja je horizontalan. Za zavarivanje je korišćen aparat za zavarivanje Iskra Avtomatika E10, a parametri zavarivanja su bili isti za sve uzorke i dati su u tabeli 3.

Tabela 3. Parametri režima zavarivanja.

Prečnik elektrode d_e [mm]	Napon luka U [V]	Jačina struje I [A]	Brzina zavarivanja V_z [mm/s]	Vrsta struje	Položaj zavarivanja
2,5	22	80	1,68	DC (+)	PH

Nakon zavarivanja prešlo se na pripremu uzoraka za mehanička, metalografska i ispitivanja hemijskog sastava.

Za potrebe mehaničkih ispitivanja, napravljeni su uzorci u vidu standardnih epruveta, sa poprečnim presekom 4x3 mm i početne merne dužine 20 mm, isečene iz metala šava, kako bi se dobile zatezne karakteristike samog metala šava, što bi se uporedilo sa osnovnim materijalom. Pre isecanja epruveta, izvršeno je skidanje nadvišenja šava. Ispitivanje zatezanjem je izvršeno na mehaničkoj kidalici VEB ZDM 5/91, na sobnoj temperaturi.

Ispitivanje tvrdoće je izvršeno na uzorcima za metalografska ispitivanja, isečenim poprečno u odnosu na šav, nakon analize mikrostrukture. Ispitivanje je izvršeno na poprečnom preseku zavarenog spoja po metodi Vickers, sa opterećenjem od 5 kg, na dubini od 2 mm ispod površine uzorka, linijski, sa rastojanjem između otisaka od 0,5 mm. Korišćen je uređaj VEB HPO-250.

Za metalografska ispitivanja primenjena je standardna priprema uzoraka što podrazumeva: isecanje, montiranje, brušenje, poliranje i nagrivanje. Uzorci su isečeni na mašini za sečenje uzoraka Struers Discotom. Nakon isecanja uzorci su postavljeni u dvokomponentni autopolimerizujući akrilat Simgal-R. Tako pripremljeni uzorci su brušeni vodobrusnim papirima različitih granulacija i to P150; P220; P360; P500; P600; P800; P1000; P1500; P2000 i P2500, na mašini za brušenje Struers Knuth Rotor. Poliranje uzoraka je izvršeno na mašini Struers DP-U2 korišćenjem tkanina za poliranje uz dodavanje dijamantske suspenzije Meta DITM Buehler, sa različitim granulacijama od 6, 3, 1 i 1/4 μ m. Nakon obrade, uzorci su oprani vodom i alkoholom, zatim obrisani i osušeni strujom toplog vazduha. Kada su uzorci

osušeni, izvršeno je njihovo nagrivanje Nitalom (3% HNO₃ i 97% etanol). Analiza mikrostrukture na tako pripremljenim uzorcima je izvršena pomoću svetlosnog mikroskopa Leitz Orthoplan. Na uzorcima u poliranom stanju, izvršena je analiza slike, sa ciljem dobijanja broja i veličine nemetalnih uključaka, softverom ImageJ.

Analiza hemijskog sastava osnovnog materijala i metala šava je izvršena na optičko-emisionom spektrometru ARL-2460.

3. REZULTATI

3.1. Hemijski sastav

Pri analizi hemijskog sastava metala šava, utvrđeno je da se povećanjem vremena infiltracije elektrode smanjuje sadržaj C, Si i Mn dok sadržaj Ti konstantno raste, tab 4.

Tabela 4. Hemijski sastav metala šava.

	%C	%Si	%Mn	%S
R0	0,08	0,28	0,402	0,027
R1	0,08	0,26	0,380	0,028
R5	0,07	0,25	0,367	0,026
R10	0,07	0,23	0,334	0,033
	%Cr	%P	%Al	%Ti
R0	0,055	0,014	0,005	0,001
R1	0,044	0,012	0,005	0,005
R5	0,047	0,014	0,005	0,009
R10	0,042	0,013	0,005	0,009

3.2. Zatezne osobine

U tabeli 5 su date vrednosti zatezne čvrstoće (R_m), napona tečenja ($R_{p0,2}$) i izduženja (A) osnovnog materijala i metala šava zavarenih uzoraka.

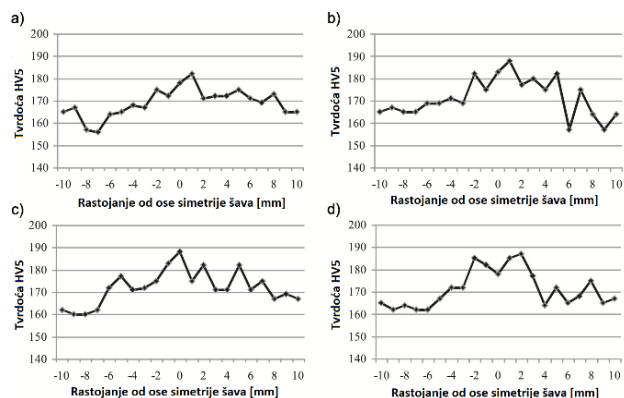
Tabela 5. Vrednosti zateznih karakteristika uzoraka.

Uzorak	$R_{p0,2}$ (MPa)	R_m (MPa)	A(%)
Osnovni materijal	339	457	28
R0	362	472	26
R1	372	482	26
R5	378	492	28
R10	350	440	20

U tabeli 5 se može videti da metal šava uzorka R0 (zavaren nemodifikovanim elektrodama) ima veći napon tečenja i zateznu čvrstoću, odnosno nešto niže izduženje u odnosu na osnovni materijal. Modifikacijom elektroda dolazi do poboljšanja mehaničkih osobina materijala, posebno na uzorku R5. Međutim, na uzorku R10 sa najdužim vremenom infiltracije, dolazi do pada mehaničkih osobina u odnosu na uzorak R0.

3.3. Tvrdća

Na slici 1 su prikazani dijagrami promene tvrdoće u odnosu na rastojanje ispitivanog mesta od ose simetrije šava.

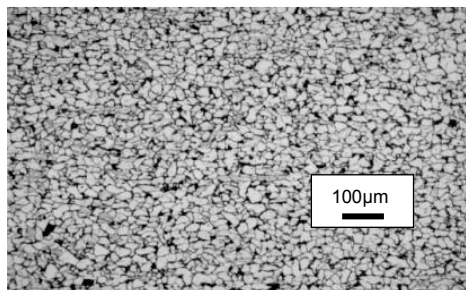


Slika 1. Dijagrami promene tvrdoće u odnosu na udaljenost od ose simetrije šava, za svaki uzorak, gde su: a) R0; b) R1; c) R5; d) R10.

Najmanji rast tvrdoće u metalu šava u odnosu na osnovni materijal je primećen kod uzorka R0. Kod uzoraka R1 i R5 se vidi najveći skok u tvrdoći a naročito kod uzorka R1, gde je veća tvrdoća i u zoni uticaja toplote (ZUT).

3.4. Mikrostruktura

Mikrostruktura osnovnog materijala je prikazana na slici 2, na kojoj se vidi mikrostruktura ferita i perlita što je u skladu sa hemijskim sastavom.



Slika 2. Mikrostruktura osnovnog materijala.

Rezultati analize slike mikrostrukture metala šava, izvršene u programu za analizu ImageJ, su prikazani u tabeli 6.

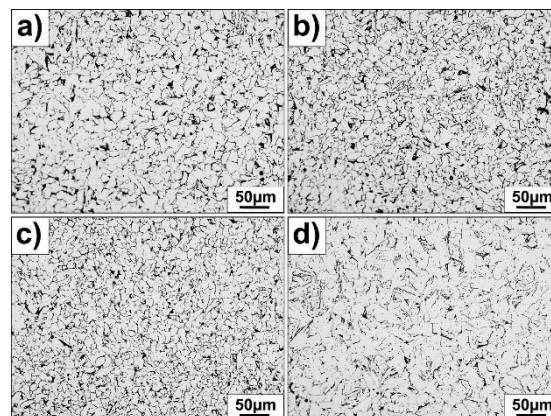
U tabeli 6 se može uočiti povećanje količine nemetalnih uključaka, ali i njihove veličine, osim kod uzorka R10 gde se smanjuje broj uključaka, što je rezultat njihovog rasta i samim tim olakšane migracije u šljaku.

S obzirom da se zavarivanje izvodilo u dva prolaza, prvi prolaz (koreni) je normalizovan i njegova mikrostruktura je data na slici 3.

Može se primetiti da je feritna mikrostruktura uzoraka R1 i R5 slične krupnoće, dok je kod uzorka R10 mikrostruktura krupnija. Takođe je kod uzorka R10 primećena pojava beinita, što je posledica izotermalne transformacije kao posledica unosa toplote tokom drugog prolaza.

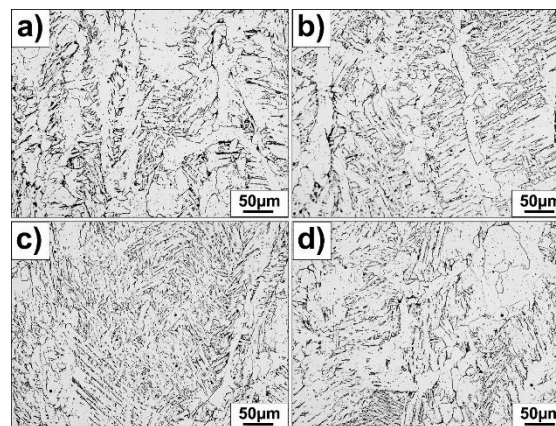
Tabela 6. Rezultati analize slike.

Uzorak	Broj nemetalnih uključaka	Prečnik nemetalnih uključaka	Prosečni prečnik nemetalnih uključaka	Procenat nemetalnih uključaka na posmatranoj površini
R0	657	0,097 – 15,24	0,32	1,29
R1	692	0,106 – 15,68	0,34	1,32
R5	712	0,118 – 15,68	0,38	1,38
R10	487	0,106 – 25,55	0,60	1,81



Slika 3. Izgled mikrostrukture prvog prolaza, nakon normalizacije (usled drugog prolaza), gde je: a) R0; b) R1; c) R5; d) R10.

Na slici 4 je prikazana mikrostruktura drugog prolaza koji nije normalizovan, već ima, za metal šava, stubastu mikrostrukturu.



Slika 4. Mikrostruktura drugog prolaza, gde je: a) R0; b) R1; c) R5; d) R10.

Uzorak zavaren elektrodom R0 pokazuje izraženi krupnozrni alotriomorfni ferit sa malom količinom Vidmanštenovog i vrlo malo igličastog ferita. Dodavanje nanočestica u oblogu elektrode dovodi do značajne rafinacije feritnih zrna, što se naročito vidi kod uzorka R5. Kod uzorka R1 se primećuje fino zrna granica zrna između alotriomorfnog i igličastog ferita, i mala količina Vidmanštenovog ferita. Na uzorku R10 se primećuju krupna zrna alotriomorfnog ferita, uz određenu količinu Vidmanštenovog i vrlo malo igličasto ferita.

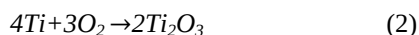
4. DISKUSIJA

U ovo radu je obrađena tehnika infiltracije nanočestica TiO_2 u oblogu elektrode i u sastav metala šava. Takođe je ispitan i njihov uticaj na mehaničke osobine samog šava.

Tokom formiranja električnog luka, stvara se veoma visoka temperatura koja utiče na disocijaciju infiltriranog TiO_2 u Ti i O_2 :



Pri hlađenju, dolazi do oksidacije elemenata. Kako i Si i Mn i Ti stvaraju okside, prvi će nastati oni koji se obrazuju na najvišoj temperaturi: Ti_2O_3 na 2130°C ; TiO 1700°C ; TiO_2 1843°C ; MnO 1945°C , SiO_2 1713°C . Prvo će nastati Ti_2O_3 , s obzirom na najvišu temperaturu topljenja/očvršćavanja. Proces dobijanja ovog oksida je:



Ovi oksidi predstavljaju inokulante za stvaranje MnO a samim tim i stvaranje zona osiromašenih Mn oko nemetalnih uključaka usled čega dolazi do formiranja Si oksida oko tih kompleksnih Mn-Ti oksida.

Brzina hlađenja je smanjena zbog povećane jonizacije atmosfere električnog luka (koja poboljšava njegovu stabilnost i penetraciju) i deblje šljake koja je rezultat povećane migracije nemetalnih uključaka u nju. Kod rezultata hemijske analize se ova pojava može jasno uočiti na taj način što se vidi opadanje količine Si i Mn u metalu šava sa povećanjem vremena infiltracije TiO_2 , što je posledica stvaranja velikih nemetalnih uključaka koji lakše od manjih napuštaju metal šava i odlaze u šljaku, povećavajući njenu debljinu i na taj način obezbeđujući smanjenu brzinu hlađenja metala šava.

Najveća zrna ferita su u uzorku R10 što je uslovljeno manjom brzinom hlađenja, ali i manjom količinom nemetalnih uključaka koji su migrirali iz metala šava u šljaku. Uzrok smanjenoj brzini hlađenja je pojava beinita u zoni koja je ponovo zagrevana (prvi prolaz). Uzorak R10, pored najvećih nemetalnih uključaka, sadrži i najmanji njihov broj, tako da je broj centara kristalizacije ujedno najmanji, što rezultuje najkrupnijom mikrostrukturom i najnižim mehaničkim osobinama. S druge strane, kod uzorka R5 koji ima najviše zatezne karakteristike, mikrostruktura sadrži najviše igličastog ferita, što je rezultat najvećeg broja centara kristalizacije, u vidu relativno sitnih nemetalnih uključaka.

5. ZAKLJUČAK

Infiltracija TiO_2 u metal šava zavarenog rutilnom elektrodom ima više efekata koji se mogu sumirati na sledeći način:

1. Dejstvom električnog luka dolazi do disocijacije nanočestica TiO_2 . Pri očvršćavanju, slobodni Ti reaguje sa kiseonikom i stvara se Ti_2O_3 koji predstavlja centar stvaranja Mn i Si oksida. Tako stvoreni nemetalni uključci predstavljaju centre kristalizacije feritnih zrna.

2. Veći broj nemetalnih uključaka uzrokuje povećanje količine igličastog ferita, koji nakon normalizacije rezultuju sitnijim poligonalnim feritom, koji utiču da zavarnei spoj ima veće mehaničke karakteristike.
3. Povećana količina nanočestica uzrokuje pojavu krupnijih nemetalnih uključaka, koji lakše migriraju u šljaku, te usporavaju brzinu hlađenja i uz smanjen broj uključaka, uzrokuju pojavu krupnozrne mikrostrukture. Takva mikrostruktura, koja se sastoji od povećanog udela alotriomornog ferita, nakon normalizacije takođe ostaje krupnija, uz smanjene zatezne karakteristike zavarenog spoja.

6. ZAHVALNOST

Rezultati prezentovani u ovom radu su realizovani u okviru projekta „Istraživanje i primena savremenih metoda karakterizacije materijala i zavarenih spojeva u proizvodnom mašinstvu“, Departmana za proizvodno mašinstvo, FTN Novi Sad.

7. LITERATURA

- [1] Palić V., "Zavarivanje I", Fakultet tehničkih nauka, Novi Sad, 1976.
- [2] Bhadeshia H. K. D. H., "Case Study: Weld Microstructure", Department of Material Science & Metallurgy, University of Cambridge, izvod sa predavanja
- [3] Fattahi M., Nabhani N., Vaezi M. R., Rahimi E., "Improvement of impact toughness of AWS E 6010 weld metal by adding TiO_2 nanoparticles to the electrode coating", Materials Science and Engineering A 528, 2011., str. 8031-8039

Kratka biografija:



Nikola Moroz je rođen u Sremskoj Mitrovici 1991. god. Master rad na Fakultetu tehničkih nauka iz oblasti Proizvodnog mašinstva odbranio je 2017.god.



Prof. dr. Sebastian Baloš rođen je u Somboru 1974. god. Doktorirao je na Fakultetu tehničkih nauka 2010. god., a 2016. god. je biran u zvanje vanrednog profesora iz užje oblasti Materijali i tehnologija spajanja.

PASIVNI VIBROIZOLATOR ZA SEIZMIČKI OPTEREĆENU KONSTRUKCIJU SA PREDNAPREGNUTIM ELEMENTOM

A PASIVE VIBROISOLATOR WITH A PRELOADED ELEMENT FOR SEIZMIC EXCITED STRUCTURE

Obiad Mirković, Zvonko Rakarić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – U radu je izvršena analiza primene nelinearnog dinamičkog amortizera na smanjenje seizmičkog uticaja na konstrukciju. Model konstrukcije je trospratna zgrada čije su mase dominantno koncentrisane u međuspratnim pločama. U radu je izvršen detaljan numerički eksperiment koji uključuje modalnu i dinamičku analizu osnovnog sistema bez amortizera, kao i ponašanje strukture nakon što je uveden dinamički amortizer. U zavisnosti od prednaprežanja elementa u amortizeru, moguće je ostvariti tri potpuno različita tipa restitucione sile i ponašanja amortizera. Parametri korišćeni u numeričkom eksperimentu su usvojeni na osnovu eksperimentalnih rezultata iz drugih izvora. Model seizmičke pobude je kretanje fundamenta strukture usvojen kao multifrekventna pobuda koja sadrži deset harmonika na frekventnom opsegu prvog lateralnog moda oscilovanja posmatrane strukture.

Abstract – In this paper, the use of a nonlinear tuned mass damper (TMD) for decreasing a seismic influence on a certain construction is analysed. The model is a three-story building with masses concentrated on the floors. A detailed numerical experiment is performed, which includes a modal and dynamical analysis with and without the TMD. Depending on preloading of a certain element in the TMD, three different types of the restoring force and the associated behaviour can be achieved. The parameters used are experimentally obtained and gained from other sources. A multifrequency excitation with ten harmonics is used as a model of an earthquake on a frequency range of the first lateral mode of the structure under consideration.

Ključne reči: Pasivni vibroizolator, seizmičko dejstvo, nelinearna restituciona sila, modalna analiza, strukturalna dinamika

1. UVOD

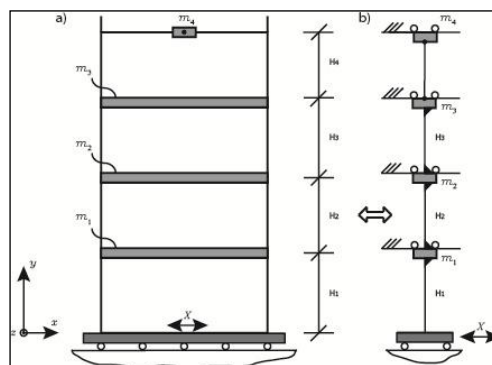
U radu [1] izvršena je eksperimentalna analiza modela koji je izrađen u cilju simulacije seizmičkog uticaja na realnu strukturu, u ovom slučaju na trospratnu zgradu. Simulacija seizmičkog dejstva je ostvarena sa kinematskom pobudom osnove pomoću uređaja, tzv. „shaker-a“, pomoću kojeg je u ovom slučaju ostvarena multifrekvencijska pobuda. Motivisani eksperimentalno dobijenim podacima, u ovom radu je sprovedena detaljna analiza modela i parametara sistema putem numeričkog eksperimenta originalno urađenog u

programskom paketu MATHEMATICA WOLFRAM [2]. Jedan od ciljeva rada je bio da se izvrši poređenje rezultata modalne analize učinjenih ovde sa rezultatima koji su u [1] sprovedeni u korisničkom paketu ABRABIBE [3] na osnovu eksperimentalno dobijenih vrednosti. U cilju umanjenja efekta oscilatornog kretanja osnove na dinamičko ponašanje osnovnog sistema sa tri stepena slobode kretanja, uvodi se dinamički amortizer. U ovom radu se uvodi dinamički absorber na poslednjem nivou (spratu) usvojenog modela, u formi teškog klizača čijim kretanjem se deformiše zategnuta žica a koja je fiksirana čvrsto na krajevima poslednjeg sprata. Deformacijom zategnute žice amortizera se generiše, u opštem slučaju nelinearna restituciona sila, kao neophodan uslov oscilatornog kretanja amortizera. U radu [4] je eksperimentalno pokazano da ovako generisana restituciona sila zavisi od prethodnog prednaprežanja žice. U zavisnosti od prethodnog prednaprežanja je pokazano da se može ostvariti dominantno linearna restituciona sila, dominantno nelinearna i restituciona sila mešovitog tipa koji u sebi uključuje i linearni i nelinearni član. Imajući to u vidu, u ovom master radu je ispitivan uticaj sva tri tipa restitucione sile na ponašanje strukture, koja je određena pažljivim izborom parametara amortizera u skladu sa realnim vrednostima eksperimentalnog modela. Izvršena je analiza dobijenih rezultata, kao i izvođenje zaključaka u smislu izbora parametara koji će umanjiti međusobno relativno kretanje između međuspratnih ploča, kao i njihove maksimalno ostvarene vrednosti ubrzanja.

2. FORMIRANJE MEHANIČKOG I MATEMATIČKOG MODELA

2.1. Formiranje mehaničkog modela

Eksperimentalni model prikazan je na Slici 1 a).



Slika 1. a) Eksperimentalni model; b) Mehanički model

Model je sačinjen od ravnih pravougaonih površina od aluminijumskog lima, i to šest vertikalnih i tri horizon-

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Zvonko Rakarić, vanr.prof.

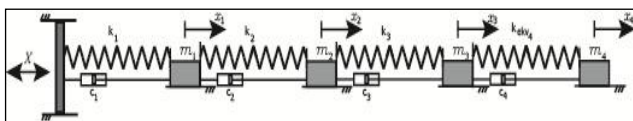
talna dela. Pri tome su na horizontalne elemente postavljene dodatne mase sa čim je ostvareno da je na svakom od tri horizontalna nivoa ostvarena masa od 1.031 kg. Imajući u vidu usvojenu geometriju, kao i specifičnu masu aluminijuma, pojedinačne mase vertikalnih elemenata, koje igraju ulogu zidova, iznose 0.2025 kg. Kao što je uobičajeno prilikom formiranja mehaničkog modela, usvajaju se određene pretpostavke. Ove pretpostavke su sledeće: 1. Kretanje međuspratnih ploča je malo u odnosu na visinu zidova i usvaja se da je samo u horizontalnom pravcu; 2. Uticaj masa zidova na dinamičko ponašanje konstrukcije je mnogo manje od uticaja masa horizontalnih delova; 3. Veza između zidova i horizontalnih elemenata je ukleštenje.

Na osnovu pretpostavke 1. sledi da se horizontalne mase kreću translatorsno pravolinijski, te imajući u vidu pretpostavku 2. biće usvojen diskretni mehanički model sa tri koncentrisane mase, kako je prikazano na Slici 1b). Usvojeni mehanički model ima tri stepena slobode kretanja tako da je matrica vektor generalisanih koordinata

$$X = \{x_1, x_2, x_3\}^T \quad (1)$$

2.2. Formiranje matematičkog modela

U cilju formiranja matematičkog modela, najpre je mehanički model diskretnog oscilatornog sistema sa elastičnim konzolama (Slika 1b) kao međusobnim vezama, predstavljen u obliku diskretnog sistema mase-opruge-prigušivači, gde se sve mase kreću po istom pravcu kako je prikazano na Slici 2.



Slika 2. Mehanički model: mase-opruge-prigušivači

Diferencijalne jednačine kretanja ovog sistema su određene dekompozicijom, te primenom drugog Njutnovog zakona za svaku masu pojedinačno. Dolazi se do sistema od tri spregnute diferencijalne jednačine u sledećem obliku:

$$m_1 \ddot{x}_1 = -k_1 x_1 - c_1 \dot{x}_1 - k_2 x_1 - c_2 \dot{x}_1 + k_2 x_2 + c_2 \dot{x}_2$$

$$m_2 \ddot{x}_2 = k_2 x_1 + c_2 \dot{x}_1 - k_2 x_2 - c_2 \dot{x}_2 - k_3 x_2 - c_3 \dot{x}_2 + k_3 x_3 + c_3 \dot{x}_3$$

$$m_3 \ddot{x}_3 = k_3 x_2 + c_3 \dot{x}_2 - k_3 x_3 - c_3 \dot{x}_3$$

(2 a,b,c)

Sistem jednačina može biti predstavljen u matricnoj formi u obliku:

$$[M]\{\ddot{x}\} + [C]\{\dot{x}\} + [K]\{x\} = \{F\} \quad (3)$$

Gde su sa $[M]$, $[C]$ i $[K]$ obeležene masena matrica, matrica prigušenja i matrica krutosti, respektivno. Sa $\{\ddot{x}\}$, $\{\dot{x}\}$ i $\{x\}$ su označeni vektori ubrzanja, brzine i pomeranja, respektivno. Sa $\{F\}$ je označen vektor spoljašnje pobude.

3. MODALNA ANALIZA

Modalna analiza se sprovodi korišćenjem matrice jednačine (3), ali pretpostavljajući da se radi o slobodnim i neprigušenim oscilacijama. U tom cilju se u matricnoj jednačini usvaja da su matrice prigušenja i vektor spoljašnje pobude jednaki nuli. Sistem predstavljen jednačinom (3) postaje:

$$[M]\{\ddot{x}\} + [K]\{x\} = \{0\} \quad (4)$$

Oscilatorni sistem definisan sa (4) ima tri stepena slobode i u skladu sa tim ima tri osnovna moda oscilovanja, kao i tri sopstvene frekvencije gde svaka odgovara jednom modu [4]. Modove oscilovanja određujemo tako što pretpostavljamo da svaka masa osciluje sinhrono sa ostalim masama i to po harmonijskom zakonu, ali sa različitim amplitudama oscilovanja. U skladu sa tim se usvajaju rešenja za kretanje masa u obliku $\{x\} = \{A\}\cos(\omega t - \phi)$, te njegovom dvostrukom integracijom se uvodi u matricnu jednačinu (4). Množenjem takve matrice jednačine sa leve strane sa inverznom masenom matricom $[M]^{-1}$ formira se standardan problem sopstvenih vrednosti i sopstvenih vektora u formi

$$[B]\{A\}^{(i)} = \omega_i^2 \{A\}^{(i)}, \quad (4)$$

Gde $(i), i=1,2,3$ označava odgovarajući mod, a $[B]=[M]^{-1} \cdot [K]$. Budući da u ovom slučaju postoje tri vektor kolone amplituda, i one onda mogu biti predstavljene u matricnoj formi, tako da izraz za rešenja kretanja pojedinačnih masa su:

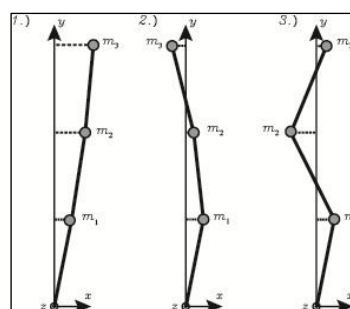
$$\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{bmatrix} (A_1)^{(1)} \\ A_2 \\ A_3 \end{bmatrix} \begin{bmatrix} (A_1)^{(2)} \\ A_2 \\ A_3 \end{bmatrix} \begin{bmatrix} (A_1)^{(3)} \\ A_2 \\ A_3 \end{bmatrix} \cdot \cos(\omega_i t - \phi_i), \quad (5)$$

Znači, matrica u izrazu (5) predstavlja matricu sopstvenih vektora, gde svaki vektor prikazan kolonom definiše modalne amplitude. Imajući ovo u vidu, te korišćenjem programa Mathematica su dobijene vrednosti za sopstvene vrednosti (sopstvene kružne frekvencije svakog moda)

$$\omega^2 = \{1145.26, 8991.31, 18775.2\}^T, \quad (6)$$

kao i matricu sopstvenih vektora:

$$A = \begin{bmatrix} 0.327985 & 0.736976 & 0.591009 \\ 0.591009 & 0.327895 & -0.73697 \\ 0.736976 & -0.59100 & 0.327985 \end{bmatrix} \quad (7)$$



Slika 3. Modovi oscilovanja osnovnog sistema

Korišćenjem amplituda modova iz (5) grafički su na Slici 3 predstavljeni oblici sva tri moda oscilovanja.

Treba naglasiti da su mase elemenata sistema, kao i ekvivalentne krutosti, parametri koji se na određeni način mogu direktno meriti ili izračunati. Za razliku od njih, mehanizam prigušenja u ovakvim sistemima je po pravilu unapred nepoznat. U ovom slučaju će se iskoristiti eksperimentalno određene veličine urađene u [1] na osnovu kojih je, primenom softvera ABRVIBE dobijene vrednosti za odnose prigušenja u formi $\zeta = [\zeta_1, \zeta_2, \zeta_3]^T = [1.4\%, 0.7\%, 0.6\%]^T$. Vrednost za $\zeta_1 = 0.014$ predstavlja odnos prigušenja u prvom modu oscilovanja. Da bi se dobio njemu odgovarajući koeficijent viskoznog prigušenja za eksperimentalni model, a koji će dalje biti iskorišćen i za mehanički model, pristupa se prevođenju matrične jednačine osnovnog sistema datog sa (3) za $\{F\} = 0$ u matričnu jednačinu po modalnim koordinatama q , koristeći teoremu modalnog razvoja [4]. To se vrši tako što se pretpostavlja rešenje sistema (3) sa $\{x\} = [A]\{q\}$, te množenjem jednačine (3) sa leve strane sa matricom $[A]^{-1}$, tako da se dobija:

$$\underline{M}\{\ddot{q}\} + \underline{C}\{\dot{q}\} + \underline{K}\{q\} = \{Q\}, \quad (8)$$

gde su: $\underline{M} = [A]^{-1} \cdot [M] \cdot [A]$, $\underline{C} = [A]^{-1} \cdot [C] \cdot [A]$, $\underline{K} = [A]^{-1} \cdot [K] \cdot [A]$ i $\underline{Q} = [A]^{-1} \{F\}$. Prikazanim postupkom se od sistema sa tri stepena slobode opisan sa (3), dolazi do tri jhedačine po modalnim koordinatama, gde se sada svaka jednačina može posmatrati kao mehanički system sa jednim stepenom slobode kretanja. Imajući sve to u vidu, kao i da je $\zeta_1 = c_1 / 2m\omega_1$, i nakon sprovedenog izračunavanja, dolazi se do vrednosti za koeficijent viskoznog prigušenja osnovnog sistema, i on iznosi $c = 4.93251 \text{ Ns/m}$.

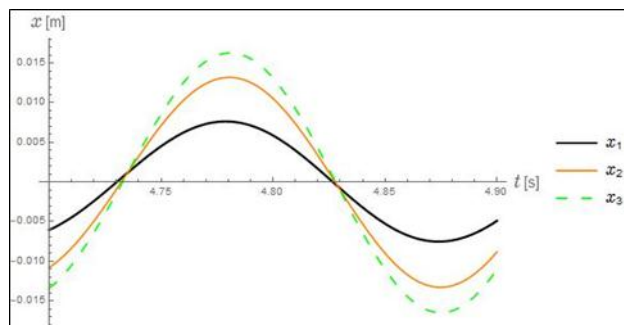
4. ODZIV OSNOVNOG SISTEMA NA MULTIFREKVENCIJSKU POBUDU

Sada se osnovni model izlaže multifrekvencijskoj pobudi u obliku:

$$\{F\} = \left\{ \sum_{j=0}^{10} \left(k f_0 \sin\left(\frac{\Omega_{j+1} + \Delta\Omega}{j+1} t\right) + c f_0 \Omega_{j+1} \cos\left(\frac{\Omega_{j+1} + \Delta\Omega}{j+1} t\right) \right) \right\}, \quad (9)$$

gde su k i c ekvivalentna krutost i prigušenje zidova prvog nivoa, respektivno. Prilikom izbora multifrekvencijske pobude se nastojalo izabrati takav frekvencijski opseg da bi se pokrio određeni frekventni opseg u okolini prvog moda oscilovanja osnovnog sistema. Naime, prilikom seizmičkog dejstva na neku konstrukciju, uobičajeno je da se vrši analiza najpre sa najnižom sopstvenom frekvencijom. U ovom master radu se nije ispitivao odziv sitema u slučaju pobuđivanja drugog i trećeg moda. Sa Ω_1 je označena donja granica izabranog frekvencijskog spektra, Sa f_0 su označene amplitude harmonica, I one iznose $f_0 = X_0 / 10$, gde X_0 predstavlja maksimalno pomeranje

podloge. Procedure urađene u Mathematici i koje su predstavljene u ovom master radu, omogućuju da se zadanjem vrednosti za veličine Ω_1 i X_0 zadaje željeno kretanje podloge, a zatim vrši odgovarajuća analiza. Na Slici 4 su predstavljeni dijagrami kretanja međuspratnih ploča izazvani cikličnim kretanjem podloge.

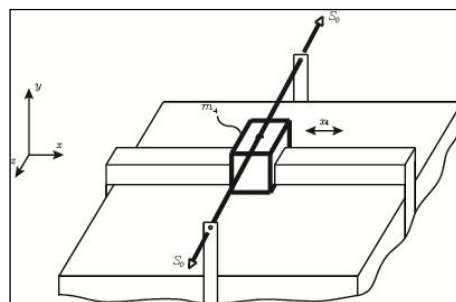


Slika 4. Kretanje međuspratnih ploča pod dejstvom multifrekvencijske pobude

4. ODZIV OSNOVNOG SISTEMA SA DINAMIČKIM AMORTIZEROM NA MULTIFREKVENCIJSKU POBUDU

4.1. Nelinearni dinamički amortizer

Nelinearni oscilator čini teški klizač koji se može kretati translatorno po vodiči (Slika 4). Kroz klizač je provučena žica, a čiji su krajevi fiksirani za konstrukciju. Vodič se održava u podmazanom stanju, da bi se ostvarilo da sila trenja bude proporcionalna brzini kretanja klizača.



Slika 5. Skica nelinearnog dinamičkog amortizera

U slučaju kretanja konstrukcije izazvane seizmičkim dejstvom (ili vetrom), u stanje kretanja dolazi i klizač koji onda svojim kretanjem izaziva deformaciju žice. Kao rezultat te deformacije generiše se restituciona sila na klizaču, uvek usmerena ka njegovom ravnotežnom položaju i na taj način obezbeđujući njegovo ciklično kretanje. Oscilacije žice kao elastičnog kontinuuma neće biti posebno razmatrano. Analitička forma restitucione sile može biti učinjena čisto teorijski ili eksperimentalno. Eksperiment predstavljen u [4] pokazuje da se restituciona sila ovakvog sistema masa+žica može predstaviti u formi:

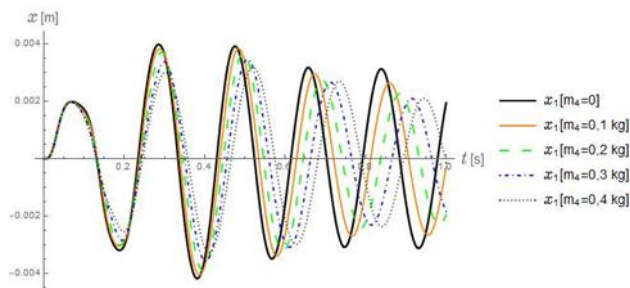
$$F_r = k_{lin} x + k_{nl} |x|^\alpha \text{Sgn}(x). \quad (10)$$

U izrazu (5) koeficijenti k_{lin} i k_{nl} , kao i stepen α , se menjaju u zavisnosti prednaprezanja žice. Povećanjem prednaprezanja se povećava vrednost linearnog koeficijenta, a smanjuju vrednosti za nelinearni koeficijent i za stepen α . Takođe, treba napomenuti i da je u radu [4], usvojen i viskozni model prigušenja, gde je pokazano da

se povećanjem sile prednaprezanja koeficijent viskoznog trenja takođe povećava.

4.2. Odziv sistema u slučaju jako prednapregnutog elementa dinamičkog amortizera *dominantno linearna restituciona sila

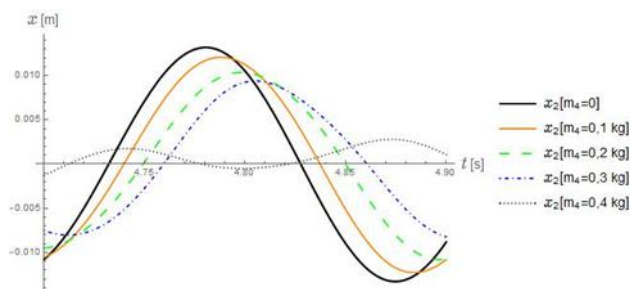
U slučaju značajnog prednaprezanja žice, (gde su konkretne vrednosti prikazane u master radu u izrazu za restitucionu silu (10) je dominantan linearni član. Na Slici 6 je prikazano pomeranje prve međuspratne ploče.



Slika 6. Odziv sistema za slučaj velikog prednaprezanja; dominantno linearna restituciona sila

4.3. Odziv sistema u slučaju slabo prednapregnutog elementa dinamičkog amortizera-čisto nelinearna restituciona sila

U slučaju da je žica slabo prednapregnuta (oko četiri puta manje nego u prethodnom slučaju), dominantan je nelinearni član, stepen α je blizak vrednosti 3, i ovde je slučaj čisto kubne restitucione sile.



Slika 7. Skica nelinearnog dinamičkog amortizera na pomeranje prvog sprata za slučaj čisto nelinearne restitucione sile

4.4. Odziv sistema u slučaju umerenog prednapregnutog elementa dinamičkog amortizera

U slučaju umerenog prednaprezanja elastičnog elementa restituciona sila se može predstaviti u obliku Dufingove restitucione sile. Detaljna analiza, kao i dijagrami pomeranja i ubrzanja za ovaj slučaj su prikazani u master radu.

5. ZAKLJUČAK

U radu je analiziran uticaj parametara nelinearnog dinamičkog amortizera na ponašanje modela trospratne zgrade pod uticajem seizmičkog opterećenja. Imajući u vidu dinamičke karakteristike osnovnog modela bez dinamičkog amortizera urađena je detaljna modalna analiza da bi se model seizmičke pobude izabrao tako da izazove pobuđivanje prvog lateralnog moda, tj. da frekventni spektar pobude sigurno obuhvati frekvenciju prvog moda i izazove fenomen rezonancije, odnosno podrhtavanja.

Paarametri sistema, izabrani na osnovu dostupnih eksperimentalnih rezultata su pokazali ove pomenute fenomene u numeričkom eksperimentu. Uvođenjem dinamičkog amortizera osnovni sistem dobio je još jedan stepen slobode kretanja. Ponavljanjem postupka, kao u slučaju osnovnog sistema bez amortizera, su se birali takvi parametri nelinearnog oscilatora koji bi bili realno prihvatljivi sa jedne strane, kao i da smanjuju maksimalne amplitude pomeranja i intenzitete ubrzanja međuspratnih ploča. Specifičnost dinamičkog amortizera opisanog tipa je u tome što se promenom samo jedne veličine, tj. prednaprezanja, utiče na promenu više dinamičkih parametara, kao što je zavisnost restitucione sile od pomeranja i na unutrašnje trenje koje je ovde usvojeno u obliku sile viskoznog trenja. Numerička analiza je pokazala da se izborom nelinearne restitucione sile dinamičkog amortizera može ostvariti značajno poboljšanje u odnosu na čisto linearan slučaj. Kao rezultat obimne numeričke analize je aplikacija urađena u programu Mathematica koja korisniku omogućuje simultanu promenu parametara, kako osnovnog sistema, tako i parametara dinamičkog amortizera i formira rešenja za pomeranja i ubrzanja na jedan pregledan način i uz istovremenu animaciju kretanja u realnom vremenu. Demonstracija aplikacije se može videti u [5].

6. LITERATURA

- [1] W. Coudron, "Comparision between linear and nonlinear vibration absorbers for seismic activity", *Master of Science in Civil engineering*, Univerzitet u Gentu, Belgija, 2016.
- [2] Wolfram Research, Inc. Mathematica, Version 11, Champaign, IL (2017)
- [3] Brandt, A. *ABRAVIBE toolbox for MATLAB*. [http://www.abravibe.com/\(2016\)](http://www.abravibe.com/(2016)).
- [4] D.M.McFarland i drugi, „*Characterization of a Strongly Nonlinear Laboratory Benchmark System*“, Proceedings of the IMAC-XXVIII, Society for experimental Mechanics inc, USA, 2010
- [5] <https://www.youtube.com/watch?v=5JRhD4452WI>

Kratka biografija:



Obrad Mirković rođen je u Vrbasu 1990. god. Master rad na Fakultetu tehničkih nauka iz oblasti Mašinstva – Tehnička mehanika i dizajn u tehnici odbranio je 2017.god.



Zvonko Rakarić rođen je u Novom Sadu 1973. Doktorirao je na Fakultetu tehničkih nauka 2011. god., a od 2017 je zvanju vanrednog profesora. Oblast interesovanja su nelinearne oscilacije.

**EKSPERIMENTALNA PROVERA UTICAJA KOEFICIJENTA TRENJA NA
PARAMETRE PROCESA DUBOKOG IZVLAČENJA****EXPERIMENTAL VERIFICATION OF THE INFLUENCE OF FRICTION COEFFICIENT
ON THE PARAMETERS OF DEEP DRAWING PROCESS**

Aleksandar Petković, Mladomir Milutinović *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – U radu su prikazani eksperimentalni rezultati istraživanja uticaja različitih kontaktnih na parametre procesa dubokog izvlačenja. Da bi se analizirao tok materijala i međusobno uporedile deformacije, na svim pripremcima izvršeno je elektro-hemijsko nanošenje merne mreže u vidu krugova jednakih prečnika.

Ključne reči: duboko izvlačenje, trenje,

Abstract - The paper presents the experimental results of the study of the influence of different friction conditions on the parameters of deep drawing process. In order to analyze the flow of material and compare the deformations, the electrochemical deposition of the measuring mesh in the form of circles of equal diameters was carried out on surface of all billets.

Key words: deep drawing, friction

1. UVOD

Cilj ovog rada je eksperimentalna provera uticaja koeficijenta trenja na naponsko- deformaciono stanje posmatranog materijala i parametre procesa dubokog izvlačenja.

Mašina na kojoj je vršen eksperiment je hidraulična presa trostrukog dejstva "Erichsen" nominalne sile 0,4 MN, namenjena je ispitivanju dubokim izvlačenjem, ali u slučaju ugradnje dodatne opreme može funkcionisati i kao kidalica, to jest kao mašina za ispitivanje zateznih karakteristika lima.

Lim korišćen pri ispitivanju je debljine 0,8 mm, u prvom koraku se radi prosecanje iz table lima, drugi korak je samo duboko izvlačenje. Materijal je čelik Č0146 (Re=280 MPa, Rm=270-410 MPa [1]).

Pre procesa dubokog izvlačenja izvršeno je elektro-hemijsko obeležavanje lima nanošenjem mreže mreže krugova prečnika 5 mm kako bi se mogla pratiti veličina deformacije za različite tribološke uslove tokom procesa dubokog izvlačenja.

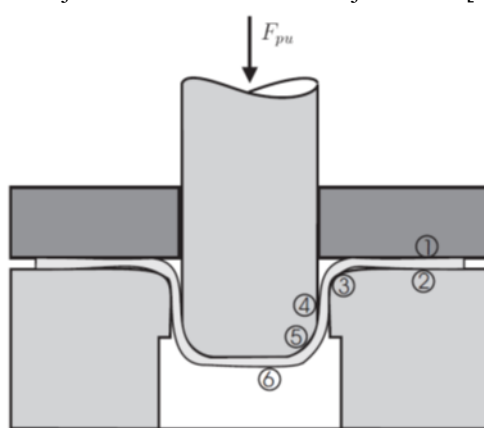
2. TRENJE U PROCESU DUBOKOG IZVLAČENJA

Kod dubokog izvlačenja do trenja dolazi u zonama kontakta alata i obratka. Razlikuje se šest kontaktnih regija (zona) (sl.1).

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Mladomir Milutinović.

Zone jedan predstavlja kontaktnu površinu između držača lima i lima, brojem dva je označena zona kontakta matrice i lima. U ovoj zoni javlja se radialno izvlačenje i naprezanja u obratku (vencu) su relativno mala. Zona 3 predstavlja kontaktnu površinu između radijusa matrice i lima. U ovoj zoni dolazi do savijanja a i potom ispravljanja lima. Naponi u materijalu su znatni i sila istezanje igra važnu ulogu. Kontakt između bočne površine žiga i lima je označen brojem četiri. U kontaktnoj regiji 5 dolazi do kontakta između radijusa žiga i lima. Kao u regiji 3 napon u materijalu u ovoj regiji je veoma visok. U regiji 6 u kontaktu su čelo žiga i lim. Što se tiče trenja literature ne govori mnogo o zonama četiri i šest zato što trenje u ovim zonama nema veliki uticaj na proces dubokog izvlačenja. Trenje u ostalim regijama (1,2,3 i 5) bitno utiče na mehanizam dubokog izvlačenja na različite načine. Trenje u zoni pet mora biti dovoljno visoko kako bi se osiguralo da lim prati kretanje žiga. Trenje u regijama jedan, dva i tri ne sme imati visoke vrednosti jer visoke vrednosti trenja dovode do povećanja sile izvlačenja što dalje rezultuje lakšem nastanku oštećenja obratka [3].



Slika 3. Kontaktne zone pri dubokom izvlačenju

**3. EKSPERIMENTALNA PROVERA UTICAJA
KOEFICIJENTA TRENJA NA PARAMETRE
PROCESA DUBOKOG IZVLAČENJA**

Eksperimentom su obuhvaćena tri različita slučaja podmazivanja: prvi slučaj podrazumeva proces dubokog izvlačenja bez mazivnog sredstva, dok drugi i treći slučaj podrazumevaju proces dubokog izvlačenja uz primenu mazivnog sredstva, mineralnog ulja i molibden disulfida. Korišćenjem različitih mazivnih sredstava dobijamo različite koeficijente trenja u zonama dodira obratka i alata. Ostali parametri procesa tokom procesa dubokog izvlačenja nisu varirani.

3.1. Proračun sile držača lima

Proračun sile držača lima je rađen za pripremak prečnika $D_0 = 64$ mm, debljine $s = 0,8$ mm i obradak unutrašnjeg prečnika $d_u = 33,2$ mm. Sila držača lima dobijena je kao proizvod površine kontakta držača lima i specifičnog pritiska držača koji iznosi 2,5 MPa [2].

$$d_1 = d_u + 2s = 33,2 + 1,6 = 34,8 \text{ mm} = 0,0348 \text{ m}$$

$$F_D = \frac{D_0^2 - d_1^2}{4} \cdot \pi \cdot q = \frac{0,064^2 - 0,0348^2}{4} \cdot 3,14 \cdot 2500000 = 7212,4 \text{ N} \approx 7,2 \text{ kN}$$

3.2. Duboko izvlačenje bez podmazivanja

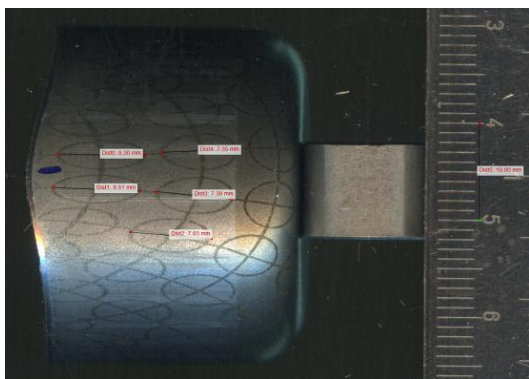
U prvom slučaju proces dubokog izvlačenja realizovan je bez mazivnog sredstva.

Izgled obratka nakon operacije dubokog izvlačenja i merenje dimenzija deformisane mreže su prikazani na slikama 2 i 3.

Sila držača lima je bila podešena na 7,2 kN, a registrovana maksimalna sila izvlačenja je iznosila $F_i = 34,2$ kN.



Slika 2. Obradak nakon procesa dubokog izvlačenja bez korišćenja mazivnog sredstva



Slika 3. Merenje deformacije mreže u procesu dubokog izvlačenja bez korišćenja mazivnog sredstva

3.3. Podmazivanje sa mineralnim uljem

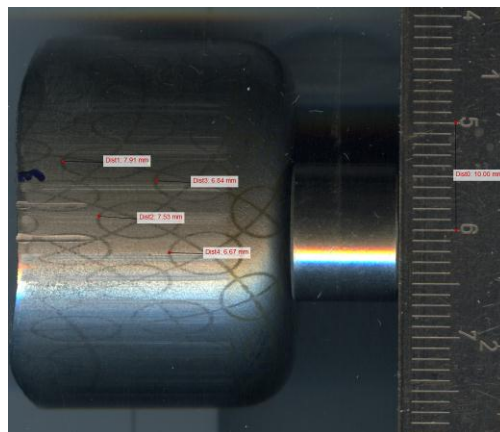
Slučaj dva podrazumeva proces dubokog izvlačenja u kome je kao mazivno sredstvo korišćeno mineralno (hidrauličko) ulje. Izgled obratka i procedura premeravanja deformacije su prikazni na slikama 4 i 5, sila držača lima je takođe iznosila 7,2 kN, a izmerena sila izvlačenja $F_i = 31,2$ kN.

3.4. Podmazivanje molibden disulfidom

Slučaj tri odnosi se na proces dubokog izvlačenja u kome je kao mazivno sredstvo korišćen molibden disulfid. Izgled obratka i merenje deformacija merne mreže su prikazani na slikama 6 i 7. Sila izvlačenja je iznosila $F_i = 33$ kN.



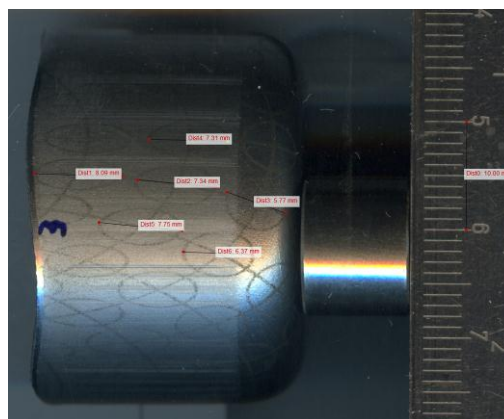
Slika 4. Obradak nakon procesu dubokog izvlačenja pri podmazivanju sa hidrauličnim uljem



Slika 5. Merenje deformacije mreže u procesu dubokog izvlačenja - mazivno sredstvo hidraulično ulje



Slika 6. Obradak u procesu dubokog izvlačenja pri podmazivanju sa MoS2



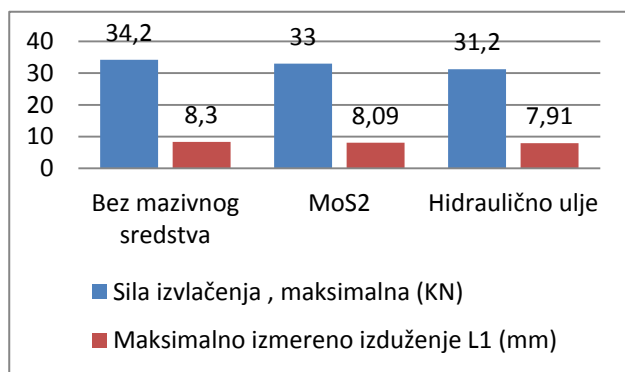
Slika 7. Merenje deformacije mreže u procesu dubokog izvlačenja - mazivno sredstvo MoS2

3.5. Analiza rezultata

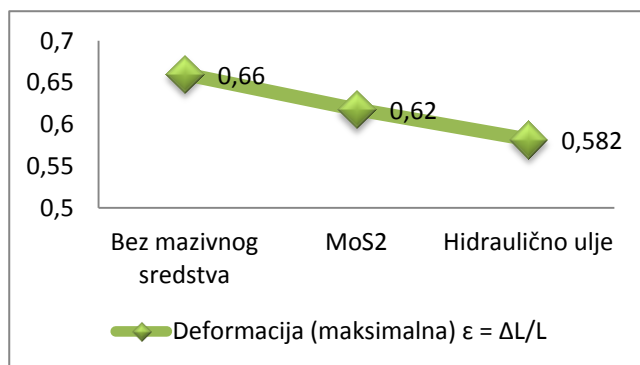
Analizom rezultata su sumirani i poređeni rezultati za sva tri slučaja. Prilikom merenja konstatovano je da su vrednosti izduženja u pravcu dejstva sile izvlačenja veća od deformacija u poprečnom pravcu te su ona uzeta u razmatranje. U tabeli 1 dati su vrednosti izmerenih deformacionih sila i deformacija deformacione, a na slikama 8 do 10 dobijeni rezultati grafički su prezentovani.

Tabela 3. Parametri procesa za različita mazivna sredstva i konstantnu silu držača lima

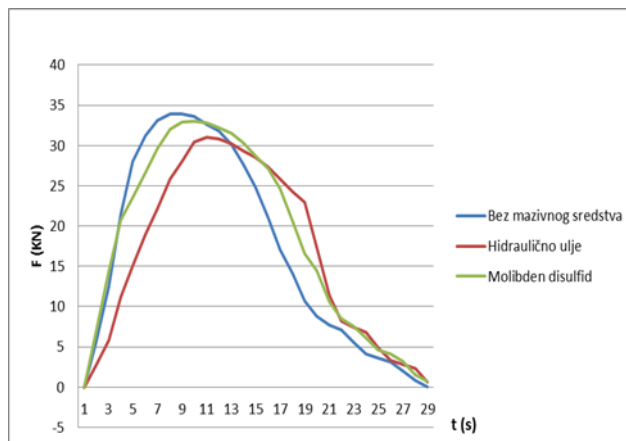
Mazivno sredstvo	Bez mazivnog sredstva	MoS ₂	Ulje
Sila držača lima (kN)	7,2	7,2	7,2
Sila izvlačenja, maksimalna (kN)	34,2	33	31,2
Prečnik kruga pre deformisanja L (mm)	5	5	5
Maksimalno izmereno izduženje L1 (mm)	8,3	8,09	7,91
Izduženje $\Delta L = L1 - L$ (mm)	3,3	3,09	2,91
Deformacija (maksimalna) $\epsilon = \Delta L / L$	0,66	0,62	0,582



Slika 8. Promena sile izvlačenja i maksimalnog izduženja u zavisnosti od korišćenog mazivnog sredstva



Slika 9. Promena deformacije u zavisnosti od korišćenog mazivnog sredstva



Slika 10. Dijagrami sile izvlačenja dobijeni digirtalizacijom dijagrama koji je iscrtala mašina.

Promenom mazivnih sredstava dovodi do promena sile izvlačenja i deformacija samog obratka. Može se zaključiti da hidraulično ulje ima bolja mazivna svojstva od MoS₂ jer su sila izvlačenja i vrednost deformacije manje u slučaju kada je kao mazivno sredstvo korišćeno hidraulično ulje. Vrednosti sile izvlačenja i deformacije su očekivano najveće u slučaju kada mazivno sredstvo nije korišćeno. Vrednost trenja je za ovaj slučaj najveća.

4. ZAKLJUČAK

Cilj ovog rada je eksperimentalna provera uticaja koeficijenta trenja na silu izvlačenja i deformaciono stanje pri dubokom izvlačenju cilindričnih obradaka. Korišćenjem različitih mazivnih sredstava kreirani su različiti kontakti uslovi u zonama kontakta obratka i alata.

Poznato je da što su mazivna svojstva sredstva za podmazivanje bolja to je i manji koeficijent trenja u zoni deformisanja. Manji koeficijent trenja rezultuje manjim otporima relativnom kretanju alata u odnosu na obradak. Pri manjim otporima manja je sila potrebna za duboko izvlačenje. Može se zaključiti da je koeficijent trenja najmanji u slučaju kada se kao mazivno sredstvo koristi hidraulično ulje dok nanošenje MoS₂ rezultuje nešto većim koeficijentom trenja u zoni deformisanja. Najveći koeficijent trenja imamo u slučaju kada se proces izvodi bez mazivnog sredstva.

Eksperimentalnim izvođenjem procesa dubokog izvlačenja potvrđeno je da se povećanjem koeficijenta trenja povećavaju vrednosti sile izvlačenja i deformacije. Sve navedeno govori da se povećanjem koeficijenta trenja u procesu plastičnog deformisanja povećava verovatnoća pojave defekata pri procesu dubokog izvlačenja.

Rezultate ovog rada je moguće primeniti u optimizaciji procesa dubokog izvlačenja i prevenciji pojave defekata korišćenjem adekvatnih mazivnih sredstava.

5. LITERATURA

- [1] www.scribd.com/doc/83045862/duboko-vučenje
(novembar 2012.).
- [2] www.cfile203.uf.daum.net/attach/124E591B499B45171A64CE (novembar 2012.)
- [3] <https://ris.utwente.nl/ws/portalfiles/portal/6074771>
(oktobar 2017.)

Kratka biografija:



Aleksandar Petković rođen je u Sremskoj Mitrovici 1988. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Mašinstva – Savremene tehnologije oblikovanja materijala odbranio je 2012.god. Od 2013. godine zaposlen u NIS – Gasprom njeft na poziciji mašinskog inženjera.

SIMULACIJA OBRADJE GLODANJEM POMOĆU METODA KONAČNIH ELEMENATA

FINITE ELEMENT METHOD SIMULATION OF MILLING PROCES

Roland Kuti, Borislav Savković, Nenad Kulundžić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – U ovom radu je izvršena simulacija formiranja strugotine i pojave temperature pri procesu glodanjem primenom programskog paketa Third Wave Systems. Za materijal alata korišćen je TM, dok za pripremak AL 7075-T6. Izvršena je simulacija obrade glodanjem, kao i nastanak strugotine i temperature u toku rada. Alat je predstavljen u obliku rezne pločice, a izbor materijala je povučen iz biblioteke alata programskog paketa Third Wave AdvantEdge, kao i pripremak. Definisanjem mreže konačnih elemenata, brzine rezanja, broj obrtaja i karakteristike materijala, omogućena je analiza formiranja strugotine, deformacije, kao i izgled samog procesa rezanja.

Abstract - This thesis presents a modern approach of study of chip forming, temperature and deformation in the milling process using Third Wave Systems software package. For the tool material is taken Hard Steel, while the work piece aluminium 7075-T6. Conducted a simulation of processing milling, as well as the formation of chips and temperature during operation. The tool is presented in the form of cutting plates and the selection of material is taken from the Third Wave AdvantEdge software library, as well as the preparation. By defining the finite elements, cutting speed and performance of materials, provided an analysis of chip forming, deformation and appearance of the cutting process.

Ključne reči: Glodanje, simulacija, Third Wave Systems, temperatura, metoda konačnih elemenata

1. UVOD

Tehnologija obrade skidanjem materijala, a naročito obrade rezanjem, danas zauzima veoma značajno mesto u proizvodnji. Osnove obrade, ili teorija rezanja su se postepeno konstituisali u toku druge polovine 20 veka. Predstavljaju znanja trajne vrednosti, potvrđena od strane velikog broja stručnjaka, verifikovana pri obradi različitim alatima i obradcima, sadržana su u velikom broju knjiga, naučnim časopisima, magistarskim radovima i doktorskim disertacijama.

Godine 1798. je B. Rumford ispred Royal Society of England u delu istraživanje vezano za izvor toplote, koja je izazvana trenjem predstavio svoje eksperimentalne rezultate u vezi toplotnih pojava pri bušenju topovskih cevi. Ovo delo se smatra prvim pokušajem teoretskog istraživanja procesa obrade.

NAPOMENA:

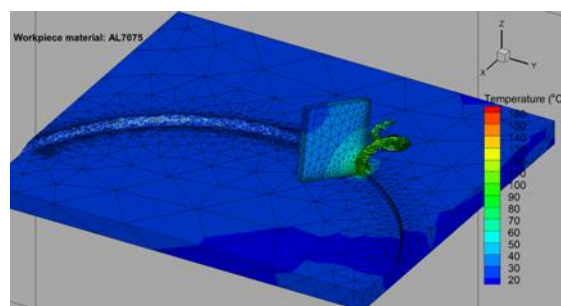
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Borislav Savković.

Sredinom pedesetih godina usledila je sve češća primena nove metode u avioindustriji, pri čemu se beleže radovi timova iz Boeing-a u SAD (M.J. Turner, Ray W. Clough, Harold Clifford Martin i LeRoy J. Topp) i Velikoj Britaniji (J.H. Argyris i Sydney Kelsey). Na predlog američkog istraživača Clough-a, ovaj metod je 1956. godine dobio svoj današnji naziv metod konačnih elemenata MKE (eng. *Finite Elements Method*). Analiza pomoću MKE bila je predmet interesovanja isključivo specijalista iz domena analize napona koji su, uz to, po pravilu radili za velike državne kompanije [1].

2. CILJEVI RADA

Polazeći od brojnih mogućnosti koje pruža programski paket Third Wave Systems i njegov modul AdvantEdge, konkretno na polju primene metode konačnih elemenata, kao takav je izabran za ovo istraživanje.

U ovom radu je izvršena simulacija formiranja strugotine i temperature pri procesu glodanjem. Cilj ovog rada je da se računarski analiziraju deformacije, temperature kod procesa obrade glodanjem. Na slici 1. prikazana je simulacija čeonog glodanja, pri čemu glavno kretanje vrši alat.



Slika 1. Simulacija čeonog glodanja u programskom paketu Third Wave Systems

Ako je potrebno brzo i lako postavljanje simulacije procesa obrade, za to su poželjni programski paketi DEFORM i AdvantEdge. Ovi programski paketi omogućavaju brzo postavljanje simulacije, pošto imaju ugrađene module materijala, geometrije alata i obratka i parametre procesa. Između ova dva paketa, programski paket DEFORM nudi veću kontrolu procesa simulacije, dok je programski paket AdvantEdge lakši za podešavanje [2].

3. METODA KONAČNIH ELEMENATA

Poseban značaj u razvoju metode konačnih elemenata (MKE) imali su varijacioni principi mehanike kontinuuma koji su primenjeni na formulaciju MKE, pa je dobio opšti pristup. Budući da je MKE numerička metoda, sva dobijena rešenja su približna, odnosno realnim vrednostima se može samo približiti uz pravilan odabir proračunskog

modela i uz pravilno odabrane tipove konačnih elemenata. Zbog toga je naročito važno poznavati fizikalno ponašanje konstrukcije, koja se analizira te poznavati teorijske osnove konačnih elemenata. Ova metoda temelji se na diskretizaciji kontinuuma, odnosno područje kontinuuma se deli na konačan broj potpodručja koja se nazivaju konačni elementi. Posmatrani kontinuum postaje mreža konačnih elemenata. Elementi su međusobno povezani tačkama koje nazivamo čvorovima, a stanje u svakom elementu (pomaci, deformacije, temperatura, itd.) opisano je interpolacijskim funkcijama, koje moraju zadovoljavati određene uslove [3].

Prema načinu na koji se formulišu i izvode osnovne jednačine MKE, izvedena je podela na:

- direktnu metodu,
- varijacionu metodu,
- metodu reziduuma i
- metodu balansa energije.

Osnovu za analizu konstrukcije predstavlja poddomen, deo domena (strukture) koji se zove konačni element. Konačni element nije diferencijalno malih dimenzija, nego ima konačne dimenzije zbog čega se zove konačni element. Zbog toga su i jednačine koje opisuju stanje u pojedinim konačnim elementima algebarske jednačine. Pomoću njih se definiše i stanje domena u celini. To znači da se razmatrani domen koji ima beskonačno mnogo stepeni slobode može podeliti na konačan broj elemenata sa konačnim brojem stepeni slobode [4]. Za ovu aktivnost nema pravila. Izbor najboljeg diskretnog modela zavisi od intuicije, inženjerske prakse i poznavanje suštine razmatranog problema [5].

4. PROGRAMSKI PAKET *THIRD WAVE SYSTEMS*

Third Wave Systems razvija i distribuira softvere za modeliranje i vrši usluge rešavanja problema svih vrsta obrade rezanjem. Proizvodi AdvantEdge i Production Modul, su izgrađeni i dizajnirani u osnovi da poboljša postojeći proces obrade. Oba proizvoda koriste napredne kompanije za optimizaciju svojih specifičnih poslova, dajući inženjerima pristup većem broju informacija, od testova na osnovu pokušaja – greške, što im omogućava da donesu bolje odluke.

Prednost izabranog programskog paketa koji je baziran na MKE je da razume "načine" učinka alata, obezbeđujući virtualno testiranje okruženja za ocenjivanje definisane konstrukcije alata. AdvantEdge nudi korisnicima jednostavno i efikasno prepoznavanje prototipova i vrhunske performanse, uz istovremeno smanjenje iteracije dizajna za generisanje više podataka bez testiranja pokušaja i greške u 2D i 3D modeliranju.

Osnovne funkcije:

- definisati geometriju alata ili uvesti iz CAD programa,
- izabrati alat i materijal obratka,
- podesiti parametre rezanja,
- pokrenuti simulaciju,
- uporediti rezultate simulacije sa optimalnim parametrima rezanja ili geometrijom alata,
- analiza temperature i napona u odnosu na habanje alata,

- korišćenjem formirane strugotine predvideti bolju evakuaciju strugotine,
- proceniti površinsku silu radi smanjenja sile rezanja i potrošnje energije [6, 7].

5. INŽENJERSKA ANALIZA KONKRETNOG DELA U PROGRAMSKOM PAKETU *THIRD WAVE ADVANTEDGE*

U doktorskoj disertaciji [8] izvršena je eksperimentalna analiza procesa obrade čeonim glodanjem i izvršena je simulacija procesa u programskom paketu DEFORM, koja je zasnovana na bazi MKE. Izvršeno je upoređivanje tačnosti programskog paketa sa realnim eksperimentom.

Eksperimentalna ispitivanja su izvedena na obratku od legure aluminijuma 7075, a širina i dužina svih obradaka je bila 100 mm, a debljina 15 mm.

Mašina alatka na kojoj su izvršeni zahvati glodanja je vertikalna glodalica "PRVOMAJSKA" FSSGVK-3 (slika 2) [8].

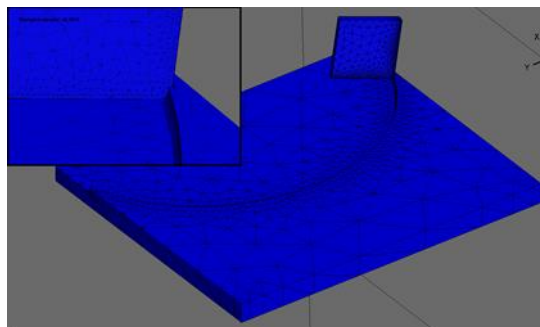


Slika 2. Vertikalna glodalica "PRVOMAJSKA" FSSGVK-3

U radu je predstavljen identični eksperiment u drugom programskom paketu Third Wave Systems i izvršeno je upoređivanje sa rezultatima dobijenim eksperimentalnim putem i dobijenim u programskom paketu DEFORM.

Koristi se pravougaoni oblik pripremk, širine 8 mm, dužine 5 mm i debljine 0,8 mm. Definisanje materijala se vrlo lako određuje odabirom iz biblioteke alata.

Definisanje MKE na obradku se vrši prilikom izbora materijala obradka u *Advanced Options*. Program nudi određivanje minimalne i maksimalne veličine mreže i sam definiše sitniju mrežu na mestu gde se vrši kontakt sa alatom, što je velika prednost u odnosu na ostale programe. Prilikom definisanja mreže konačnih elemenata, takođe važi pravilo kao i kod alata da krupnoća mreže i tip mreže mnogo zavisi od tačnosti rezultata. Što je sitnija mreža, znači da je tačnost veća.



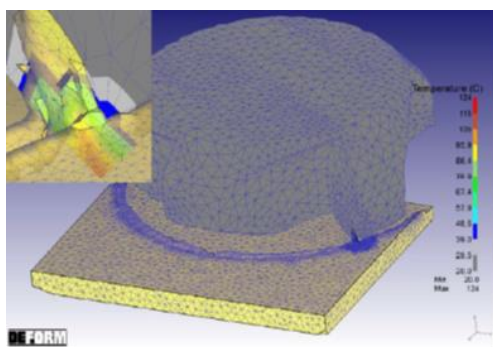
Slika 3. Izgled mrežaste strukture obradka i alata

6. REZULTATI SIMULACIJE

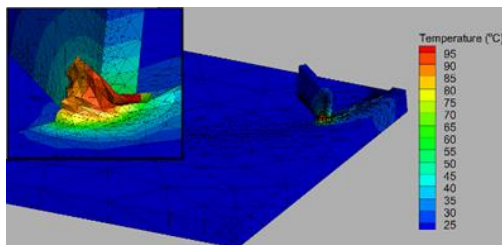
Rezultat simulacije je prikazana u nekoliko slika, koje su izvučene u odgovarajućim trenucima. Nijanse boja su prikazane počevši od plave boje, gde je minimalna temperatura pri rezanju, pa do crvene, gde je maksimalna temperatura. Za materijal alata korišćen je TM, dok za pripremak legura aluminijuma oznake 7075-T6.

U simulaciji su uzeti sledeći parametri rezanja: broj obrtaja od 1120 o/min, translatorno kretanje obratka od 4,16 mm/s ($s=250$ mm/min), dok je dubina glodanja iznosila 0,114 mm. Dužina obratka je 8 mm, širina 5 mm, dok visina kreiranog modela iznosila 0,8 mm. Treba napomenuti da su definisane dimenzije korišćenog modela umanjene za 10 puta u odnosu na veličine korišćene u eksperimentu. Simulaciju je računar obrađivao više od 40 dana.

Na slici 4 i 5 prikazan je položaj alata u samom procesu početka stvaranja strugotine u programskom paketu DEFORM i Third Wave AdvantEdge.

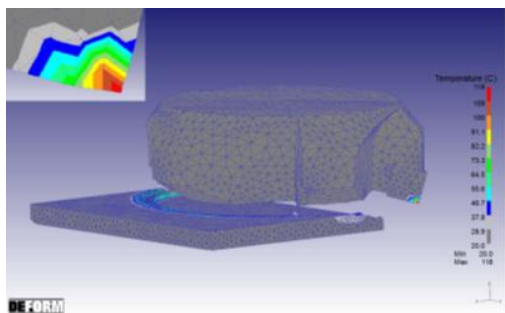


Slika 4. Položaj alata u samom početku procesa stvaranja strugotine u programskom paketu DEFORM [8]



Slika 5. Položaj alata u samom početku procesa stvaranja strugotine u programskom paketu Third Wave AdvantEdge

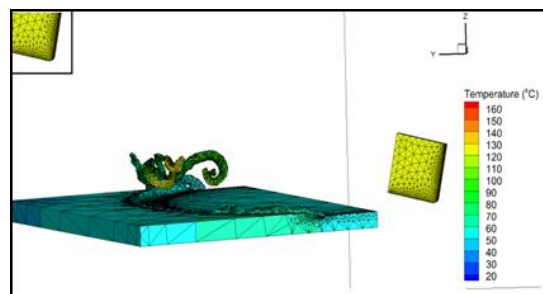
Na slici 6 i 7 prikazan je položaj alata pre samog ulaska u zahvat u programskom paketu DEFORM i Third Wave AdvantEdge nakon što je alat zauzeo najveću moguću širinu glodanja.



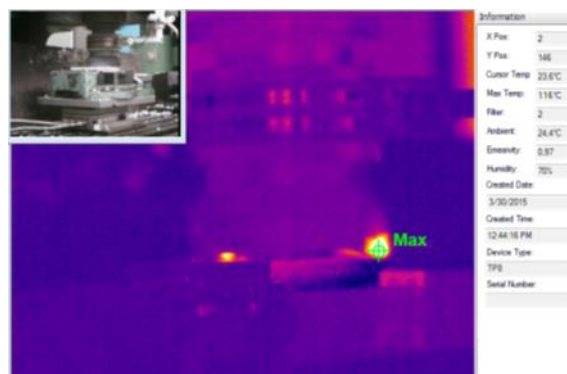
Slika 6. Položaj alata pre ulaska u zahvat u programskom paketu DEFORM [8]

Na slici 8 prikazano je eksperimentalno merenje temperature posredstvom termo-kamere pre samog ulaska u

zahvat nakon prethodno postignute konstantne temperature rezanja.

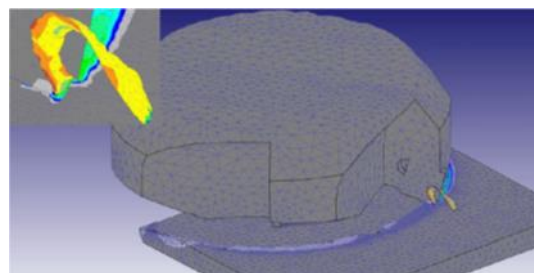


Slika 7. Položaj alata pre ulaska u zahvat u programskom paketu Third Wave AdvantEdge

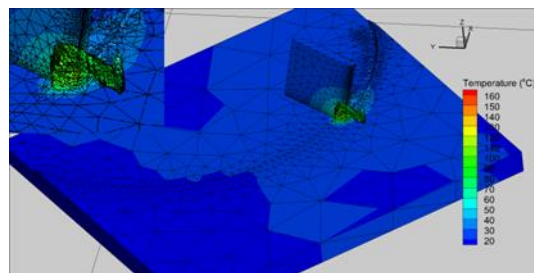


Slika 8. Eksperimentalno merenje temperature posredstvom termo-kamere [8]

Na slici 9 i 10 prikazane su simulacije položaja alata gde se formira strugotina u programskim paketima Deform i Third Wave AdvantEdge. Ovaj primer služi da se bolje sagleda sam proces rezanja, stvaranje strugotine pri istim režimima obrade dobijeni eksperimentalnim putem, slika 11. Ako se uporedi oblik strugotine dobijen eksperimentalnim putem i posredstvom simulacije, može se konstatovati da su približno identični.



Slika 9. Oblik strugotine dobijene posredstvom simulacije u programskom paketu DEFORM [8]

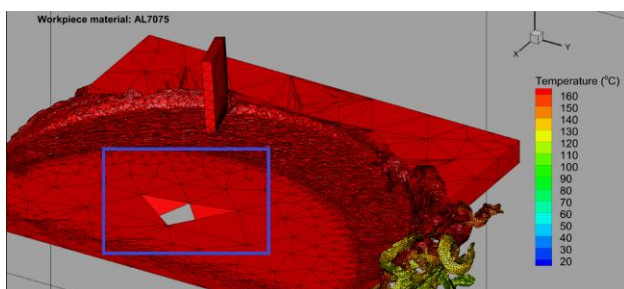


Slika 10. Oblik strugotine dobijene posredstvom simulacije



Slika. 11. Oblik strugotine dobijene eksperimentalnim putem [8]

U toku ove simulacije alat je izvršio 20 prolaza. Kako se broj prolaza povećavao došlo je do nastajanja greške u vidu rupe na obratku što se vidi na slici 12.



Slika 12. Greška nastala nakon 14 prolaza

7. ZAKLJUČAK

Upoređivanjem dva programska paketa koji su bazirani na MKE, može se zaključiti kako programski paket DEFORM, tako i programski paket Third Wave AdvantEdge prikazuju približno identične rezultate. Programski paket DEFORM nudi veću kontrolu procesa simulacije, dok je programski paket AdvantEdge lakši za podešavanje. Ovi programski paketi omogućavaju brzo postavljanje simulacije, pošto imaju ugrađene module materijala, geometrije alata i obratka, parametre procesa i na taj način omogućavaju da brzo i jednostavno dobijemo željene rezultate.

Na osnovu analize urađene u programskom paketu Third Wave Systems, utvrđeno je da se temperature, kao i nastajanje strugotine mogu ispitivati na ovaj način.

Primenom Third Wave Systems programa skraćuje se vreme pripreme tehnologije obrade i životnog ciklusa proizvoda, poboljšava se postojeći proces obrade, inženjerima potpomaže uz pristup većem broju informacija, na osnovu testova tipa "pokušaj – greška", i omogućava da donesu bolje odluke.

Za dobijanje rezultata koji su veoma blizu realnim rezultatima, potrebno je alat predstaviti u što realnijem obliku, povećati broj mrežne strukture, što će opteretiti sam proračun i ujedno produžiti vreme simulacije, navesti sve karakteristike kako materijala obratka, tako i materijala alata, u sam proces obrade uvesti neki vid sredstva sa hlađenje i podmazivanje. Kao što je prethodno navedeno simulacija je vremenski trajala više od 40 dana i kako se povećavao broj prolaza dolazilo je do vidljivih grešaka prilikom izvođenja simulacije. To ukazuje na to da se čak i bez uvođenja svih karakteristika procesa u model simulacije ne može do kraja izvesti kompletan proces obrade, ako se ne poseduje računar sa izuzetnim performansama što će omogućiti izvođenje kompleksnijih i dužih zahvata.

8. LITERATURA

- [1] Milikić D., Gostimirović M., Sekulić M.: Osnove tehnologije obrade rezanjem, Univerzitet u Novom Sadu, Fakultet Tehničkih nauka, Novi Sad 2008.
- [2] Vijayaraghavan, A., Gardner, J.D.: Comparative study for finite element simulation software, University of California, Berkeley, 2015.
- [3] Lemeš, doc. Dr. S.: Modeliranje konstrukcija primjenom računara, Univerzitet u Zenici, Politehnički Fakultet, Zenica 2015.
- [4] Kovač, P., Savković, B., Kulundžić, N., Gostimirović, M., Holešovský, F., Sekulić, M.: "Simulation of temperature distribution on the 3D moving cutting tool", Development in Machining Technology, Scientific Research Reports, Cracow University of Technology, 2015, pp. 20-26, ISBN:978-83-7242-844-8.
- [5] Kulundžić, N: Simulacija obrade rezanja u programskom paketu ANSYS, Master rad, Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka, Novi Sad 2013.
- [6] Third Wave Systems, <http://www.thirdwavesys.com/>. Preuzeto: 25.02.2017.
- [7] Kovač, P., Kulundžić, N., Mitrović, A., Savković, B., Gostimirović, M.: "FEM milling tools temperature measurement, modelling and simulation", PSU-UNS International Conference on Engineering and Technology – ICET, Faculty of Technical Sciences, Novi Sad 2017, Vol. 8, No. 1, pp. 1-4, ISBN 978-86-7892-934-2.
- [8] Savković, B. : Modeliranje funkcija obradivosti pri procesu obrade glodanjem, Doktorska disertacija, Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka, Novi Sad 2015.

Kratka biografija:



Roland Kuti rođen je u Kikindi 1988. god. Diplomski-master rad na Fakultetu tehničkih nauka iz oblasti Proizvodnog mašinstva iz predmeta Visokoproduktivne obrade odbranio je 2017.god.



Borislav Savković rođen je u Novom Sadu 1982. god. Diplomski-master rad odbranio je 2007. god., a doktorirao je na Fakultetu tehničkih nauka 2015. god. Autor je 1 univerzitetskog udžbenika i preko 80 naučnih radova. Oblast interesovanja su procesi obrade skidanjem materijala i simulacije.



Nenad Kulundžić rođen je u Kraljevu 1988. god. Diplomski-master rad na Fakultetu tehničkih nauka iz oblasti Proizvodnog mašinstva iz predmeta Visokoproduktivne obrade odbranio je 2013.god. Od 2015. je u zvanju istraživača pripravnika.

DISKRETIZACIJA NAVIJE-STOKSOVE JEDNAČINE PRIMENOM METODE KONAČNIH ZAPREMINA**DISCRETISATION OF NAVIE-STOKES EQUATION USING A FINITE VOLUME METHOD***Ilija Vlahović, Fakultet tehničkih nauka, Novi Sad***Oblast – MAŠINSTVO**

Kratak sadržaj – U okviru ovog rada razmatrana je diskretizacija Navije-Stoksove jednačine primenom metode konačnih zapremina. Cilj rada je da se teorijski prikaže postupak diskretizacije Navije-Stoksove jednačine primenom metode konačnih zapremina. Zatim da se u komercijalnom softveru uradi numerička simulacija strujanja vazduha kroz filter boks kojom će se ilustrovati postupak diskretizacije Navije-Stoksove jednačine primenom metode konačnih zapremina. Numeričke simulacije su urađene za sledeće brzine strujanja vazduha: 2 m/s, 4 m/s, 6 m/s, 8 m/s i 10 m/s, sa vazduhom na temperaturi 15 °C i pritisku 101325 Pa. Analiziran je uticaj promene dimenzija postojećeg filter boksa na pad pritiska i intenzitet buke koju generiše strujanje vazduha. Varirane su: dužine pravih deonica, uglovi ivica, prečnici ulazne i izlazne cevi plenuma, širina i visina plenuma. Predložene su optimalne dimenzije postojećeg filter boksa koje će dati minimalan pad pritiska i minimalni intenzitet buke koju generiše strujanje vazduha kroz plenum.

Abstract – Within this paper, the discretization of the Navi-Stokes equation is considered using a finite-volume method. The aim of the paper is to present the theory of the discretisation of the Navi-Stokes equation using a finite-volume method. Then, in commercial software, a numerical simulation of air flow through a filter box will be performed to illustrate the method of discretization of the Navy-Stokes equation using the method of finite volumes. Numerical simulations were made for the following air velocities: 2 m / s, 4 m / s, 6 m / s, 8 m / s and 10 m / s, with air at 15 °C and pressure 101325 Pa. The effect of changing the dimensions of the existing filter box to the pressure drop and the noise intensity generated by the air flow is analyzed. The length of the right sections, the angles of the edges, the diameter of the inlet and outlet pipes of the plenum, the width and height of the plenum are varied. The optimal dimensions of an existing filter box are proposed, which will give a minimum pressure drop and minimum noise intensity generated by air flow through the plenum.

Ključne reči: Računarska dinamika fluida, diskretizacija Navije-Stoksove jednačine, plenum boks, plenum

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Siniša Bikić.

1. UVOD

Cilj rada jeste da se teorijski prikaže postupak diskretizacije Navije-Stoksove jednačine primenom metode konačnih zapremina. Zatim da se u komercijalnom softveru uradi numerička simulacija strujanja fluida kojom će se ilustrovati postupak diskretizacije Navije-Stoksove jednačine primenom metode konačnih zapremina.

Kao primer ilustracije strujanja uzeće se strujanje vazduha kroz postojeći plenum boks. Plenum (*pleh-num*, od latinske reči- puno) je element sistema za grejanje, ventilaciju i klimatizaciju (Heating Ventilation and Airconditioning - HVAC). Plenum boks služi za distribuciju vazduha u domaćinstvima, zgradama, industriji itd. Uz pomoć plenuma vazduh se iz jednog izvora deli na više izlaza iz plenuma i na taj način razvodi do prostora u kom je potreban. U zavisnosti od potrebe postoje razne izvedbe plenuma, raznih dimenzija i karakteristika. Često se u plenumima sreću ispune koje služe za smanjivanje buke koje proizvodi strujanje vazduha kroz njih.

Ima mnogo tipova plenuma koji se sreću u grejanju, ventilaciji i klimatizaciji, ali svi oni se sastoje od zatvorene zapremine pod pozitivnim ili negativnim pritiskom i najmanje dva otvora kako bi se omogućio protok vazduha. Plenum u kome se nalazi ventilator naziva se plenum sa ventilatorom. Sve druge vrste plenuma se nazivaju protočni plenumi. Plenumi su uglavnom pravougaonog oblika, konstantnog poprečnog preseka sa višestrukim vezama manje veličine [1].

U okviru master rada razmatran je plenum boks jednostavne konstrukcije sa jednim ulaznim i jednim izlaznim otvorom.

2. PREGLED STANJA U OBLASTI

Plenumi se koriste u različite svrhe, ali pre svega kao aerodinamični uređaji koji služe za distribuciju vazduha kroz više kanala koji se nalaze u ograničenom prostoru. Veoma malo je napisano o plenumima u tekstovima koji se bave projektovanjem kanala za distribuciju vazduha, a autori nisu svesni prethodnih studija o aerodinamičkim performansama HVAC plenuma.

Mašinskih inženjeri prvenstveno predviđaju protočne plenuma. U poslednje vreme su sve više popularni ventilatorski, gde je unutar plenuma smešten ventilator. Plenumima se smanjuje intenzitet buke niske frekvencije [2]. Akustični plenum je vrsta pasivnog prigušivača koji je najefikasniji u nižim frekvencijama.

Ranija istraživanja akustičnih plenuma pokazala su da postoji značajan opseg greške inteziteta buke (± 10 dB) pri predviđanju akustičnih performansi. Nedavni radovi na akustičnim plenumima su značajno smanjili granice greške inteziteta buke [3].

Skorašnji istraživački projekat sponzorisan od strane ASHRAE, RP-1026, razmatrao je protočne plenuma (bez unutrašnjih ventilatora) i kvantifikovao njihove akustičke performanse, energetske gubitke i njihove aerodinamičke performanse u ukupnom padu pritiska [3].

3. TEORIJSKI OSNOVE DISKRETIZACIJE NAVIJE-STOKSOVE JEDNAČINE PRIMENOM METODE KONAČNIH ZAPREMINA

Pri rešavanju problema konvekcije i difuzije pretpostavlja se da je polje brzine poznato. Koristeći polje brzine pronalaze se rešenja ostalih skalarnih polja. Polje brzine se pronalazi iz jednačine kontinuiteta i Navije-Stoksove jednačine, koje se rešavaju zajedno.

Jednačina kontinuiteta za dvodimenzijско strujanje glasi:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0,$$

gde su:

- u – brzina strujanja u pravcu x ose [m/s];
- v – brzina strujanja u pravcu y ose [m/s];
- x – pravac x ose [m] i
- y – pravac y ose [m].

Navije-Stoksove jednačine za dvodimenzijско strujanje glase:

x -pravac:

$$\rho \left[\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right] = -\frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right),$$

y -pravac:

$$\rho \left[\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right] = -\frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right),$$

gde su:

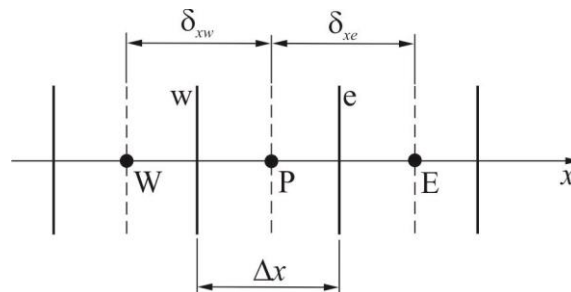
- p – pritisak fluida [Pa];
- ρ – gustina fluida [kg/m^3];
- μ – koeficijent dinamičke viskoznosti [Pa s] i
- t – vreme [s].

Radi se o nelinearnim jednačinama koje se mogu rešiti iterativnim postupkom. Problem se usložnjava usled prisustva gradijenta pritiska u Navije-Stoksovim jednačinama, jer ne postoji posebna jednačina iz koje bi se odredio gradijent pritiska. Za dobijanje realnog rešenja primenjivog na trodimenzionalni problem treba zadržati pritisak u jednačini.

Urađiće se diskretizacija gradijenta pritiska. Koristiće se opšti principi diskretizacije (slika 1):

- zapremina fluida podeli se na veliki ali konačan broj zapremina;

- diferencijalna jednačina se integriše po uočenoj zapremini, gde se vrednost fizičke veličine dodeljuje centralnoj tački (težištu) zapremine, mrežnoj tački i
- pretpostavi se profil fizičke veličine između mrežnih tačaka.



Slika 1. Uočena zapremina

Integraljenjem člana gradijenta pritiska od granice e do granice w (slika 1) dobija se:

$$\int_e^w -\frac{\partial p}{\partial x} dx = p_w - p_e.$$

Sada će se pritisci na granicama uočene zapremine izraziti u funkciji pritiska susednih mrežnih tačaka, uradiće se interpolacija pritiska. Pretpostaviće se da je pritisak na granici uočene zapremine jednak aritmetičkoj sredini pritisaka susednih mrežnih tačaka:

$$p_w = \frac{p_w + p_p}{2},$$

$$p_e = \frac{p_p + p_e}{2},$$

pa je rešenje integrala:

$$\int_w^e -\frac{\partial p}{\partial x} dx = \frac{p_w + p_p}{2} - \frac{p_p + p_e}{2} = \frac{p_w - p_e}{2}.$$

Protok kroz uočenu kontrolnu zapreminu koja se nalazi između graničnih površina w i e definisan je razlikom pritisaka u tačkama W i E .

Iz dobijene jednačine može da se zaključi da ako je pritisak u svim mrežnim tačkama jednak gradijent pritiska će biti jednak nuli, ali isto tako ako je pritisak u tačkama W i E jednak, a pritisak u tački P različit, gradijent pritiska će opet biti jednak nuli. U prvom slučaju ne postoji gradijent pritiska, a u drugom postoji gradijent pritiska međutim jednačina ne obuhvata taj gradijent.

Problem nastaje u tome što se integriše po površini uočene kontrolne zapremine za čije granice w i e nisu definisane vrednosti. Vrednosti su definisane samo za mrežne tačke P, W i E . Vrednosti na granicama kontrolne zapremine w i e su tražene uz pomoć interpolacije, gde je način interpolacije napravio problem. Problem se rešava korišćenjem SIMPLE ili SIMPLER algoritma [6].

4. PLAN ISTRAŽIVANJA

Plenumi (plenum boksevi) se koriste u grejanju, ventilaciji i klimatizaciji otkad su ventilatori povezani sa cevovodima. Ima mnogo tipova plenuma koji se sreću u grejanju, ventilaciji i klimatizaciji, ali svi oni se sastoje od zatvorene zapremine pod pozitivnim ili negativnim pritiskom i najmanje dva otvora kako bi se omogućio protok vazduha.

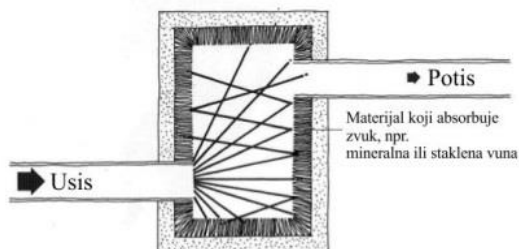
Plenum u kome se nalazi ventilator naziva se plenum sa ventilatorom slika 2. Sve druge vrste plenuma se nazivaju protočni plenumi. Plenumi su uglavnom pravougaonog oblika, konstantnog poprečnog preseka sa višestrukim vezama manje veličine.

Uglavnom se izrađuju od ploča metala galvanizovanog ili nerđajućeg čelika, debljine i veličine propisanim standardom.

Zidovi plenuma se često prekrivaju akustičnim materijalom na unutrašnjoj površini kako bi se smanjio prenos buke nizstrujno kroz cevovod ili kako bi se obezbedila toplotna izolacija plenuma slika 3.



Slika 2. Plenum sa ventilatorom [4]

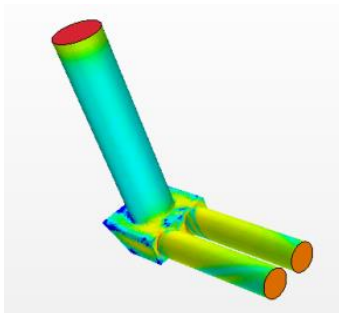


Slika 3. Kompenzacija buke u plenumu [5]

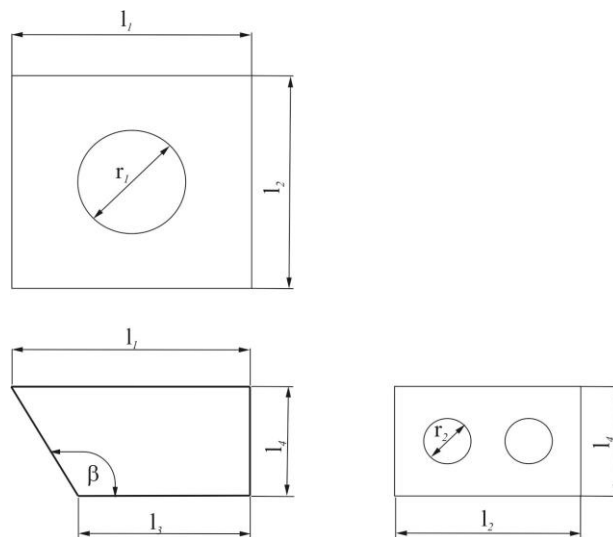
Na slici 4. prikazan je model jednostavnog plenuma.

Na Slici 5. prikazane su dimenzije plenuma jednostavne konstrukcije koje će biti faktori optimizacije dimenzija plenum boksa, gde će svaki faktor imati određen broj nivoa.

U Tabeli 1 prikazani su faktori i njihovi nivoi.



Slika 4. Plenum jednostavne konstrukcije



Slika 5. Skica plenuma jednostavne konstrukcije

Tabela 1. Faktori i nivoi pri optimizaciji dimenzija plenum boksa

Faktor	Nivo
β [°]	120
	130
	140
	150
r_1 [m]	0,04
	0,05
r_2 [m]	0,03
	0,02
l_1 [m] l_2 [m] l_3 [m] l_4 [m]	0,25
	0,25
	0,178
	0,133
	0,133
l_1 [m] l_2 [m] l_3 [m] l_4 [m]	0,308
	0,2
	0,228
	0,183
l_1 [m] l_2 [m] l_3 [m] l_4 [m]	0,358
	0,35
	0,278
	0,233
l_1 [m] l_2 [m] l_3 [m] l_4 [m]	0,408
	0,4
	0,328
	0,283

Faktor istraživanja bila je i brzina strujanja vazduha. Svi prikazani nivoi u tabeli 1 urađeni su za sledeće nivoe brzine strujanja: 2, 4, 6, 8 i 10 m/s.

5. PREGLED REZULTATA

Numeričke simulacije su izvršene tako da su varirani nivoi samo jednog faktora u odnosu na postojeći plenum boks i to za različite nivoe brzine strujanja vazduha. Zatim je postupak ponovljen pojedinačno za svaki razmatrani faktor.

Nakon izvršenih numeričkih simulacija analizirani su pad pritiska i intezitet buke koju generiše strujanje vazduha kroz plenum.

Pojedinačno gledajući faktor poluprečnik r_1 ulazne cevi plenuma i njegov nivo 0,04 m imali su najveći uticaj na smanjenje pada pritiska i inteziteta buke koju generiše strujanje vazduha kroz plenum.

U sledećoj tabeli 2 prikazani su faktori i nivoi koji su pojedinačno dali najmanje vrednosti pada pritiska i inteziteta buke. Takođe su prikazane vrednosti pada pritiska i inteziteta buke na originalnom plenum boks.

Zatim su urađene numeričke simulacije sa kombinacijom faktora i odgovarajućih nivoa koji su pojedinačno dali najbolje rezultate u pogledu smanjenja pada pritiska i inteziteta buke. Numeričke simulacije su izvršene za definisane nivoe brzine strujanja vazduha.

Tabela 2. Faktori i nivoi koji su pojedinačno dali najmanje vrednosti pada pritiska i inteziteta buke

Plenum jednostavne konstrukcije	v [m/s]	2	4	6	8	10
Original	Δp [Pa]	6,0	23,3	51,7	90,8	143,7
	Li [dB]	37,3	59,6	77,6	81,3	84,7
$\beta=120^\circ$	Δp [Pa]	5,8	22,2	49,4	87,3	136,5
	Li [dB]	38,5	57,6	73,4	80,7	85,9
$r_1=0,04$ m	Δp [Pa]	2,4	7,7	20,0	40,9	70,4
	Li [dB]	28,5	46,8	64,5	69,4	74,3
$r_2=r_3=0,03$ m	Δp [Pa]	18,5	72,9	161,7	284,7	442,2
	Li [dB]	48,5	76,8	85,2	92,1	96,7
$l_1=0,408$ m $l_2=0,4$ m $l_3=0,328$ m $l_4=0,283$ m	Δp [Pa]	7,5	29,4	65,9	120,7	187,2
	Li [dB]	40,9	61,3	79,0	86,0	92,5

Najmanji pad pritiska i intezitet buke koju generiše strujanje vazduha kroz plenum dobijeni su za sledeće faktore i nivoe: poluprečnik ulazne cevi plenuma $r_1=0,04$ presek plenuma $l_1=0,408$ m, $l_2=0,4$ m, $l_3=0,328$ m, $l_4=0,283$ m.

6. ZAKLJUČAK

Diskretizacijom Navije-Stoksove jednačine metodom konačnih zapremina, upotrebom komercijalnog softvera postignuta je optimizacija dimenzija jednostavnog plenum boks sa jednim ulaznim i izlaznim otvorom.

Preporučene su dimenzije za postojeći plenum boks koje će dati najmanji pad pritiska i intenzitet buke koju generiše vazduh koji struji kroz boks.

5. REFERENCE

- [1] Karl Peterman, P.E. , Emanuel Mouratidis, P.Eng. ,Pressure Drop and Acoustical Application Guidelines for HVAC Plenums, NA-04-3-2 (RP-1026), CONFERENCE PROCEEDING by ASHRAE, 2004
- [2] Wells, R.J. 1958 Acoustical Plenum Chambers. Noise Control, 4, 945.
- [3] Mouratidis, E. 2003. The Aero-Acoustic Properties of Common HVAC Plena, TRP-1026. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.
- [4] <http://www.wildeboer.de/en/products/air-distribution/ceiling-diffusers/plenum-boxes/> , (21.09.2017).
- [5] <https://www.slideshare.net/narasimhaswamy/noise-control-55355735> , (21.09.2017).
- [6] Vlahović, I., (2017). Diskretizacija Navije-Stoksove jednačine primenom metode konačnih zapremina, master rad, FTN, Novi Sad.

Kratka biografija:



Ilija Vlahović rođen je u Novom Sadu 1989. god, gde je završio srednju mašinsku školu. Diplomirao je 2014.god na Fakultetu tehničkih nauka, studijski program Mašinstvo, smer Energetika i procesna tehnika. Zaposlen je u kompaniji "Naftna Industrija Srbije".

PRIMENA VISOKOBRZINSKOG GLODANJA U IZRADI ALATA ZA BRIZGANJE APPLICATION OF HIGH SPEED MILLING IN INJECTION MOLDS MACHINING

Danilo Dujović, Pavel Kovač, Nenad Kulundžić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – Rad se bavi ulogom CAM sistema i Visokobrzinskih obrada, kao i njihovoj međusobnoj povezanosti u tehnologiji izrade alata za brizganje plastike. Takođe u radu će ukratko biti opisana simulacija glodanja kalupne forme u istom programskom paketu SolidCAM. Ukratko rečeno, u ovom radu će biti opisani postupci od usvajanja priprema izabranog dela do generisanja upravljačkog G koda za mašinu gde bi se obrađivao.

Abstract – This paper deals with role of the CAM and HSM, as well as their interconnection with the technology of plastic injection molding tools. There will be as well briefly description of the simulation of mold milling in the SolidCAM software package. In short, this paper will describe the procedures from adopting and preparation for the selected part G control code for the machining process.

Ključne reči: injeksiono brizganje, VBO, CAD, CAM, glodanje

1. UVOD

Materijali zauzimaju veliki udeo u kvalitetu i ceni samog proizvoda. Svaki materijal ima ulogu da se uz njegovu minimalnu cenu i primenu što jednostavnijih tehnologija obrade dobije proizvod zadovoljavajućeg kvaliteta. Polimeri su takozvani materijal budućnosti, jer poseduju sledeće karakteristike:

- Nisku cenu,
- Jednostavnu preradu,
- Veliku slobodu u pogledu dizajna i boje proizvoda,
- Zadovoljavajući kvalitet proizvoda i
- Mogućnost recikliranja.

Polimeri su organska ili neorganska jedinjenja izgrađena od molekula velikih molekulskih masa (makromolekula) u kojima su atomske grupe, tzv. "strukturne jedinice", povezane hemijskim vezama čineći tako polimerni (makromolekulski) lanac [1].

Brizganje plastike spada u ciklične postupke oblikovanja termoplasta i ujedno je najčešći i najsloženiji proces dobijanja plastičnih proizvoda [2].

Ovaj postupak je ujedno i najusavršeniji od svih postupaka oblikovanja plastičnih komponenti i spada u primarnu preradu polimera. Razlog tome je što pripremak nema određenu formu i najčešće je u obliku granulata, mada su prisutni i prah, pelet i komadići.

NAPOMENA:

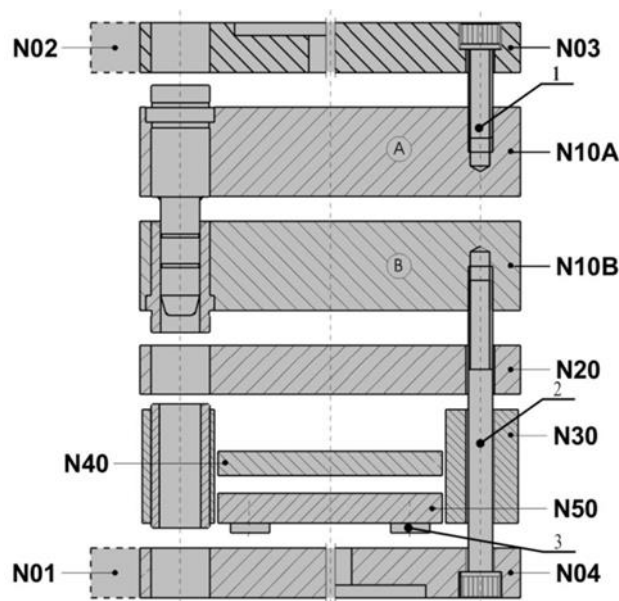
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Pavel Kovač.

Alat ili kalup je namenski deo sistema injeksionog presovanja polimera i po pravilu služi za proizvodnju jednog proizvoda. Osnovni zadaci alata su prihvatanje rastopa iz mlaznice jedinice za ubrizgavanje i njegovo hlađenje do postizanja forme otpreska, potiskivanje otpreska iz kalupne šupljine i ciklusni rad sistema injeksionog presovanja.

Sve te postavljene zadatke alat ne može da izvrši ako nema sledeće elemente:

- Ulivni sistem,
- Kalupnu šupljinu,
- Sistem za izbacivanje otpreska,
- Sistem za vođenje kalupa,
- Kućište i
- Elemente za temperiranje kalupa.

Pored ovih elemenata alat može imati i druge elemente koji se koriste samo u posebnim konstrukcijama alata (slika 1).



Slika 1. Konstrukcija alata za injeksiono presovanje plastike sa elementima za vezu [3]

Prema slici 1 standardni nazivi i oznake (prema D-M-E standardu) su: Pokretna stezna ploča N01 i N04,

- Nepokretna stezna ploča N02 i N03,
- Nepokretna kalupna ploča N10A,
- Pokretna kalupna ploča N10V,
- Među-ploča (ili osnovna ploča) N20,
- Distantna letva (ili odstojni segment) N30,
- Izbacivačka ploča N40 i
- Nosač izbacivačke ploče N50.

2. CAM SISTEMI

CAM (*Computer Aided Manufacturing*) sistemi predstavljaju programske alate koji podržavaju intenzivnu upotrebu računara za planiranje i projektovanje proizvodnih i tehnoloških procesa, operacija i upravljanja proizvodnjom, odnosno proizvodnim procesima.

Osnovne funkcije CAM sistema vezane za planiranje proizvodno – tehnoloških procesa su:

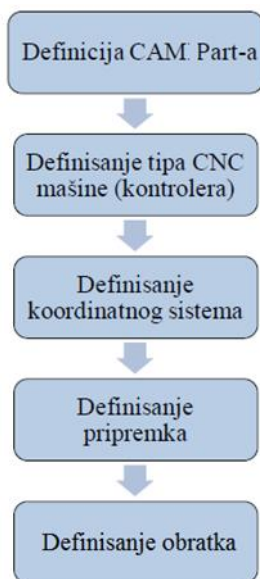
- Generisanje priprema,
- Generisanje i optimizacija putanja alata,
- Kreiranje i korišćenje baza podataka, kataloga režima i alata,
- Proračun vremena izrade,
- Generisanje NC programa (numerički upravljačkog),
- Simulacija i vizuelizacija procesa izrade,
- Generisanje proizvodne dokumentacije,
- Brza izrada prototipova (*Rapid Prototyping*).

Geometrijski model proizvoda predstavlja finalni oblik koji treba postići nakon obrade, a pripremak od koga nastaje pojedinačni mašinski element je drugačijeg oblika. Model obrade predstavlja skup priprema i obratka. CAM sistemi imaju mogućnost za automatsko generisanje priprema na osnovu geometrijskog modela proizvoda.

Formiranje modela obrade (slika 2) definiše se u nekoliko faza definisanjem:

- tipa CNC mašine na kojoj će se izabrani deo obrađivati (CNC mašina sa 3, 4, 5 – upravljačkih osa)
- koordinatnog sistema radnog predmeta
- geometrije priprema
- geometrije obratka

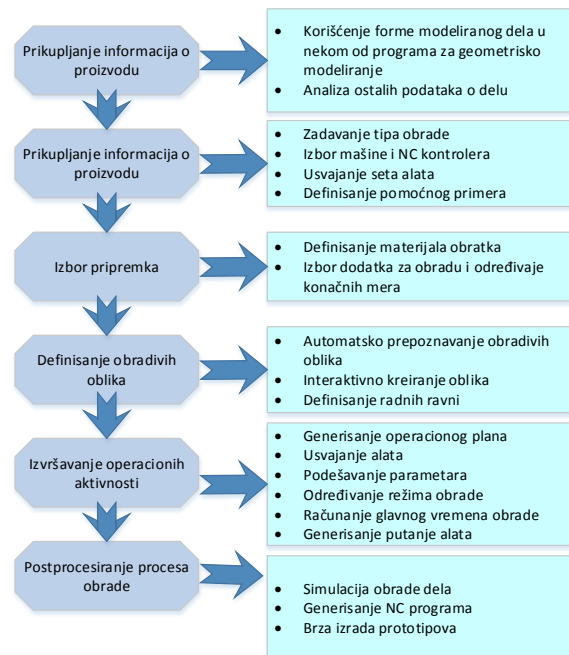
Potrebno je naglasiti da je formiranje modela obrade preduslov za definisanje tehnoloških operacija (zahvata) obrade obratka.



Slika 2. Struktura definisanja modela obrade

Generisanje NC programa kojim se vrši računarsko upravljanje proizvodnom opremom je u potpunosti automatizovano. To se izvodi posebnim funkcijama CAM sistema, na osnovu geometrijskog i tehnološkog modela.

Redosled i sadržaj osnovnih funkcija CAM sistema su prikazane na slici 3.



Slika 3. Redosled i sadržaj osnovnih funkcija CAM sistema

3. VISOKOBRZINSKE OBRADNE

Visokobrzinska obrada (*High Speed Machining*) nema jednoznačnu definiciju već ih različiti autori različito definišu na osnovu polaznih kriterijuma, koji mogu biti:

- Brzina rezanja
- Broj obrtaja glavnog vretena
- DN broj (proizvod broja obrtaja i prečnika glavnog vretena)
- Dinamičko ponašanje i
- Materijal obratka i postupak obrade

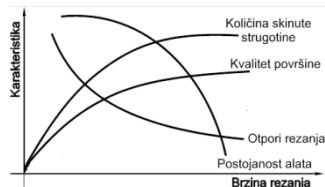
Ako je polazni kriterijum brzina rezanja, onda se po Schneider-u može napraviti čvrsta podela na tri područja obrade:

- Klasična obrada ; $v_c < 500$ m/min,
- Visokobrzinska obrada ; $v_c = 500 - 10\,000$ m/min,
- Ultravisokobrzinska obrada ; $v_c > 10\,000$ m/min.

Za polazni kriterijum broja obrtaja glavnog vretena, visokobrzinskim obradama se smatraju brzine preko 8.000 o/min, dok se ultravisokobrzinskim obradama podrazumevaju obrade sa preko 60.000 o/min. Ovaj kriterijum je varljiv, jer ne uzima u obzir prečnik glodala. Glodala sa malim prečnicima mogu lakše postići veliki broj obrtaja, za razliku od većih glodala. Baš zbog ove problematike je uveden pojam DN broja. Vrednost DN broja predstavlja proizvod prečnika glavnog vretena (mm) i broja obrtaja (o/min). Vrednost DM broja kod visokobrzinskih obrada se kreće od 500.000 do 1.000.000. Postoji veliki broj komercionalnih mašina sa DM brojem većim od 1.500.000, a nekoliko laboratorija poseduju i mašine sa DM brojem od 2.000.000 [4].

Najveća prednost visokobrzinskih obrada je povećanje produktivnosti koja se ogleda u povećanoj količini skinutog materijala u jedinici vremena. Jako važna karakteristika visokobrzinskih obrada je mogućnost rada sa velikim pomacima (povećanje i do 10 puta u odnosu na konvek-

cionalne). Povećanjem brzine rezanja poboljšava se kvalitet obrađene površine i smanjuje se hrapavost obrađene površine. Kod visokobrzinske obrade smanjuju se sile rezanja (manji utrošak energije). Manje sile uzrokuju manje deformacije obratka, stoga je moguća preciznija obrada, kao i obrada tankozidnih obradaka. Veliki problem visokobrzinskih obrada je veliki pad postojanosti alata (slika 4).



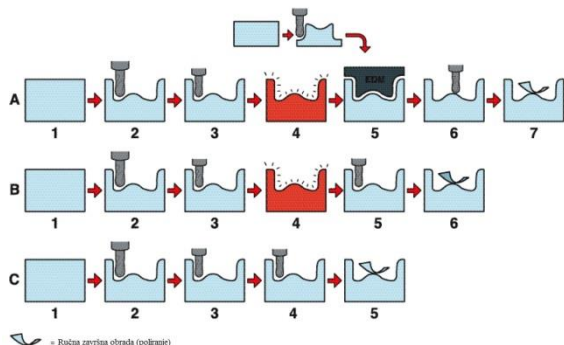
Slika 4. Karakteristika procesa u zavisnosti od brzine rezanja [5]

Za vrednosti tvrdoće od 46 do 63 HRC, visokobrzinsko glodanje može čak zameniti skupu tehnologiju kao što je EDM izborom pogodnih glodala i odgovarajućih tehnoloških parametara. Moguće je procesom glodanja izraditi matrice i žigove za duboko izvlačenje, što značajno utiče na smanjenje vremena obrade (Slika 5).

Na slici 5-A prikazan je tradicionalni proces obrade. Pripremak je neokaljeni čelik (1), gruba obrada (2), poluzavršna obrada (3), kaljenje čelika na potrebnu tvrdoću (4), EDM (elektroerozivna obrada), proces izrada EDM elektroda i EDM obrada kalupa (5), završna obrada delova dobre pristupačnosti (6), poliranje (7).

Na slici 5-B je sličan postupak kao pod 5-A samo što je EDM proces zamenjen visokobrzinskom obradom i rezultat je smanjenje jednog zahvata obrade.

Na slici 5-C materijal je okaljen odmah na početku obrade (1). Prvo ide gruba obrada (2), zatim poluzavršna (3), i na kraju završna obrada (4). Visokobrzinska obrada je primenjena u gotovo svim operacijama (posebno s malim alatima). Proces je skraćen za 3 operacije (izrada elektrode, EDM proces i termička obrada). Normalno smanjenje vremena u poređenju sa procesom (A) je za oko 30-50 %.



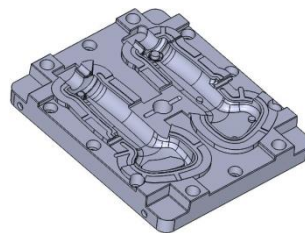
Slika 5. Prikaz poboljšanja produktivnosti primenom VBO [6]

4. PRIMER VEZE IZMEĐU VISOKOBRZINSKIH OBRADA I CAM SISTEMA PRI IZRADI ALATA ZA BRIZGANJE PLASTIKE

Pri tehnologiji izrade alata za brizganje plastike pomoću visokobrzinske obrade su u stalnoj vezi sa CAM sistemima. Rezultat toga je dobijanje povećane produktivnosti samih obrada i simulacioni pristup obradama preko računara. U ovom primeru CAM sistem je pogodan radi pra-

ćenja putanja alata i neutralisanje pojave loma alata o obradak (kolizije).

Za obradak je usvojena forma koja se nalazi u nepokretnoj kalupnoj ploči alata za brizganje ručke za ručnu kosilicu. Forma je prikazana na slici 6 i sadrži dve šupljine koje su međusobno slične i u velikoj meri imaju iste regione.

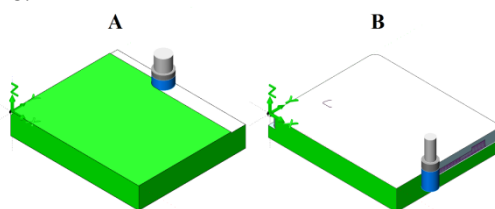


Slika 6. Forma nepokretne kalupne ploče

Za CAM programski paket je izabran SolidCAM kako iz razloga lakog upravljanja, tako i iz velike zastupljenosti ovog programa u ovoj delatnosti.

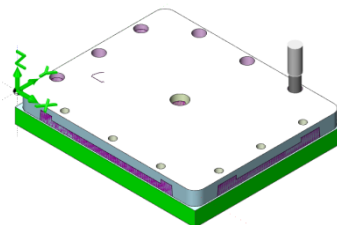
U ovom primeru je pretpostavljeno da se obradak u vidu ploče steže za neobrađenu površinu. U izradi kalupa primenjeno je visokobrzinsko glodanje sa standardnim oblicima glodala. Obrade koje idu nakon ovih obrada (brušenje i elektrohemijska obrada) neće biti opisane. Režimi obrade su birani po preporukama proizvođača reznog alata.

Prva operacija je tradicionalno čeonno glodanje (sl. 7-A). U ovoj operaciji se poziva alat T10 (oznaka u SolidCAM) koji je ustvari čeonno glodalo. Zatim sledi profilno glodanje (slika 7-B) i poziva se alat T20, odnosno vretenasto glodalo.



Slika 7. Čeonno (A) i profilno glodanje (B)

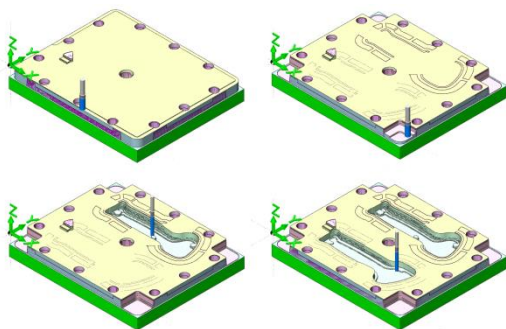
Nakon pomenutih operacija sledi izrada otvora (slika 8). U ovom primeru izrada otvora se sastoji od zabušivanja (alat T30 – zabušivač), bušenja na predmetu (alati T40 i T50 – burgije), razvrtanja na konačnu meru (alati T60 i T70 – razvrtači) i cilindrično upuštanje (alat T80 – upuštač).



Slika 8. Izrada otvora

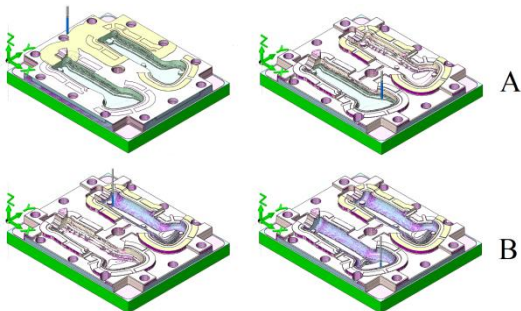
Izradom otvora završava se obrada konvencionalnim dvoiploosnim (skr.: 2,5D) glodanjem. Zatim sledi visokobrzinsko troosno (skr.: 3D) glodanje koje se u ovom slučaju može podeliti na grubo visokobrzinsko glodanje, poluzavršno visokobrzinsko glodanje i završno visokobrzinsko glodanje.

Grubo visokobrzinsko glodanje se sastoji nekoliko faza, koje su prikazane na slici 9. Pozivaju se alati T90 a zatim T100 (vretenasta glodala).



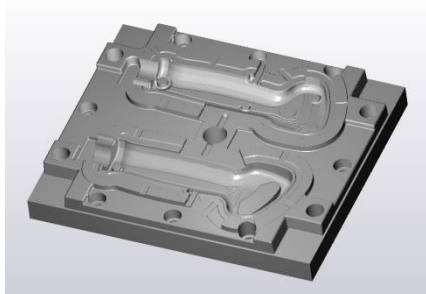
Slika 9. Gruba 3D visokobrzinska obrada

Poluzavršna obrada je prikazana na slici 10-A i poziva se alat T110 (torusno glodalo). Završno visokobrzinsko glodanje (slika 10-B) se vrši u dve faze i poziva se alat T120 (loptasto glodalo).



Slika 10. Poluzavršna visokobrzinska obrada (A) i završna visokobrzinska obrada (B)

Nakon završne obrade visokobrzinskog glodanja, model obrade ima izgled kao na slici 11. Nakon toga na alatu se vrši obrada brušenja i elektrohemiska obrada.



Slika 11. Izgled obratka nakon završne obrade glodanja

Na slici 12 može se videti slika gotovog alata kao i plastičnog dela koji se dobija pomoću injektionog brizganja polimera u izrađeni kalup.



Slika 12. Izgled kalupa i ručke nakon brizganja

6. ZAKLJUČAK

U radu je opisan postupak izrade kalupa za injektiono brizganje ručke uz pomoć visokobrzinske obrade. Simulacija obrade je izvedena pomoću SolidCAM programskom paketa koji je implementiran kao modul u SolidWorks CAD-u.

Primarni cilj ovog rada je bio da se predstave prednosti visokobrzinske obrade pri izradi kalupa za brizganje plastike pomoću integrisanih programskih rešenja, koja pružaju efikasno, brzo i pouzdano projektovanje tehnoloških operacija obrade uz naglasak na generisanje savremenih putanja alata.

S obzirom na sve veću prisutnost proizvoda od plastičnih masa u svakodnevnom životu, može se reći da je potreba za ovim proizvodima povećana, a samim tim i za kalupima za injektiono presovanje. Zbog toga svaka inovacija u polju injektionog presovanja je dobrodošla, a svako novo rešenje za obradu kalupa omogućava dalje unapređenje ovog postupka.

7. LITERATURA

- [1] Čatić I.: „Proizvodnja polimernih tvorevina,“ Biblioteka Polimerstvo, zelena serija, Zagreb 2006.
- [2] Vilotić D.: „Mašine za injektiono presovanje,“ Fakultet tehničkih nauka, Novi Sad, 2013
- [3] <http://www.dme.net> Preuzeto: 21.08.2017.
- [4] Milikić, D., Gostimirović, M., Sekulić, M.: Osnove tehnologije obrade rezanjem, Fakultet tehničkih nauka, Novi Sad, 2008.
- [5] Kovac, P., 2014. Teorija obradnih procesa, Univerzitetski udžbenik, Univerzitet u Novom Sadu, Fakultet tehničkih nauka, Novi Sad.
- [6] Metalworkin World, München, Deutschland: The application of high speed machining

Kratka biografija:



Danilo Dujović, rođen je u Novom Sadu 1 oktobra 1987. godine. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Tehnologije obrade rezanjem je odbranio 2013. godine.



Pavel Kovač, rođen je u Bačkom Petrovcu 1950. Magistarski rad je odbranio 1980. god, a doktorirao je na Fakultetu tehničkih nauka 1987. god.. Od 1998. je u zvanju redovnog profesora. Autor je 5 univerzitetskih udžbenika i 2 monografije. Oblast interesovanja su procesi obrade skidanjem materijala.



Nenad Kulundžić, rođen je u Kraljevu 1988. god. Diplomski-master rad na Fakultetu tehničkih nauka iz oblasti Proizvodnog mašinstva iz predmeta Visokoproduktivne obrade odbranio je 2013.god. Od 2015. je u zvanju istraživača pripravnika.

ODREĐIVANJE VELIČINE GREŠKE TIPA TUNELA PRI PRERADI TRENJEM SA MIJEŠANJEM**DETERMINATION OF TUNNELLING DEFECT SIZE IN FRICTION STIR PROCESSING**

Nemanja Spasojević, Sebastian Baloš, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – *Prerada trenjem sa miješanjem je postupak kojim se vrši poboljšanje površinske i podpovršinske strukture materijala i uklanjanje defekata nastalih procesom prethodnih obrada. Procesiran materijal usljed neadekvatnih parametara procesiranja često za posljedicu ima pojavu tunela, te ga je iz tog razloga potrebno ispitati. U ovom radu je izvršeno ispitivanje uzoraka koji su procesirani različitim alatima i režimima (brzina obrtanja i translacije), a dobijeni uzorci su ispitani ultrazvučnom metodom. Rezultati su pokazali da najveći uticaj na eliminaciju pojave tunela ima geometrija alata, odnosno primena nagiba alata, a od sekundarnog značaja su kinematski parametri poput brzine obrtanja i translatorne brzine.*

Abstract: *Friction stir processing is a procedure that results in an improved surface and subsurface properties as well as the removal of defects from previous manufacturing processes. The processed material can also contain different defects, most notably tunnelling, due to unadequate parameters, so it is necessary to test it. In this work, specimens obtained with different tools and parameters (rotation and translation speed) were tested by ultrasound technique. It was shown that tool geometry and tool tilt has the most significant influence on the elimination of tunneling, while kinematic parameters such as rotation and translation speed have a secondary effect.*

Ključne reči: *procesiranje trenjem sa mešanjem, ultrazvučno ispitivanje, tuneli*

1. UVOD

Istraživanjem i usavršavanjem različitih tehnika zavarivanja došlo se novih ideja koje su u znatnoj mjeri unaprijedile procese spajanja materijala. Pravi primjer za prethodno navedeno jeste zavarivanje trenjem sa miješanjem. Različitim variranjem parametara i postupaka ove metode došlo se do konstatacije da se ova metoda može koristiti i u druge svrhe, (ne samo spajanje materijala), kao što su ojačavanje površinskog i podpovršinskog sloja materijala i ispravljanje greški nastalih livenjem i kao takva nazvana je Procesiranje trenjem sa miješanjem. Postupak procesiranja trenjem sa miješanjem se izvodi identično kao i zavarivanje trenjem sa miješanjem.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Sebastian Baloš.

Alat se postavlja u steznu glavu glodalice, dok se materijal koji se procesira pozicionira i steže na radni sto. Obrtnim kretanjem alata kroz materijal vrši se izjednačavanje površinske i podpovršinske strukture materijala. Ovom metodom se pored ispravljanja površinskih defekata vrši i ojačavanje površinskog sloja materijala [1].

Postoji i mogućnost dodavanja keramičkih čestica u obrtno kretanje alata, čime se vrši ojačavanje materijala, odnosno, postoji mogućnost dobijanja sloja koji je po sastavu zapravo kompozitni materijal sa metalnom osnovom. Na taj način se vrši povećavanje tvrdoće, čvrstoće, otpornosti na habanje, itd. Pored navedenog, uticajni faktori na kvalitet procesiranja mogu biti: vrsta alata, broj obraza alata, brzina pomijeranja alata, ugao zakošenja alata itd. Odabirom najoptimalnijih parametara dobijaju se najbolje karakteristike procesiranog materijala bez pojave greški. Izbor neadekvatnih parametara procesiranja najčešće dovodi do pojave tunela u materijalu [2]. Iz tog razloga je neophodno izvršiti kontrolu procesiranih uzoraka nekom od metoda ispitivanja bez razaranja materijala (najčešće ultrazvučnom ili radiografskom kontrolom). Prednost ultrazvučne kontrole je što joj debljina materijala koji se ispituje ne predstavlja prepreku.

Ultrazvučni uređaj (Slika 1) preko emitera šalje signal kroz materijal i taj isti signal prihvata na detektoru. Na osnovu brzine kretanja ultrazvučnog talasa i vremena određuje se debljina materijala. Pri nailasku na grešku u materijalu (tunel) ultrazvučni talas se odbija i dobija se dimenzija do tunela [3].

Kombinovanjem podataka dimenzija s od površine do greške sa svake strane i ukupne debljine, dobija se dimenzija greške.



Slika 1: Ultrazvučno ispitivanje

U ovom radu je prikazan uticaj variranja parametara procesiranja trenjem sa miješanjem i makroskopsko i ultrazvučno ispitivanje dobijenih materijala.

2. EKSPERIMENTALNI DIO

Esperimentalni dio se može podijeliti na dvije faze:

1. Procesiranje trenjem sa miješanjem
2. Makroskopsko i ultrazvučno ispitivanje

2.1. Procesiranje trenjem sa miješanjem

Procesiranje je vršeno na leguri aluminijuma AA5754-H111 debljine 8 mm. Hemijski sastav, legure aluminijuma, prikazan u tabeli 1, određen je sprektomatrom ARL 3580.

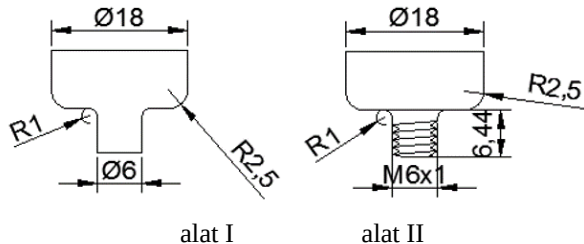
Tabela 1. Hemijski sastav legure aluminijuma

Cu	Mn	Mg	Si	Fe	Zn	Ti	Cr	Al
0,05	0,33	3,10	0,33	0,37	0,06	0,02	0,04	Ost.

Izvršeno je sedam proba procesiranja pri čemu su varirani nauticajni parametri.

Korištena su dva alata, slika 2 (alat I bez, alat II sa navojem) pri čemu su brzina obrtanja alata i pomak varirani.

Hemijski sastav alata za procesiranje određen je spektrometrom ARL 2460. Procesiranje je vršeno na Fakultetu tehničkih nauka u Novom Sadu na glodalici Prvomajska UHG 520, slika 3.



Slika 2: Šematski prikaz korištenih alata

Tabela 2. Hemijski sastav alata za procesiranje

C	Si	Mn	P	S	Cr	Mo	V	Fe
0,37	1,01	0,38	0,02	0,01	4,85	1,23	0,32	Ost.



Slika 3: Glodalica prvomajska UHG 520

Izvršeno je sedam proba, pri čemu su prve dvije alatom sa glatkim trnom, dok je ostalih pet proba izvršeno alatom koji na trnu ima urezan navoj.

U tabeli br. 3 dat je prikaz variranja parametara.

Tabela 3. Parametri koji su korišteni pri procesiranju

	Broj obrtaja [min ⁻¹]	Brzina procesiranja [mm/min]	Ugao zakretanja alata [°]	Tip alata
1	1400	20	0	I
2	1120	31,5	0	I
3	1400	20	0	II
4	900	20	0	II
5	1120	31,5	0	II
6	450	63	3	II
7	355	40	3	II

Nakon završenog procesiranja trenjem sa miješanjem izvršeno je makroskopsko i ultrazvučno ispitivanje duž zone miješanja. Makroskopsko ispitivanje je izvršeno isijecanjem centralnog dijela uzorka na osnovu koga se vidi poprečni presjek zone miješanja, slika 4. Nakon isecanja, izvršena je standardna metalografska priprema brusnim papirima (oznake P320, 400, 600, 800 i 1000) radi utvrđivanja prisustva tunela.



Slika 4: Izgled procesiranog materijala nakon isijecanja poprečnog presjeka za makro ispitivanje

Ultrazvučno ispitivanje izvršenih proba vršeno je u livnici preciznih odlivaka - LPO u Adi, na ultrazvučnom uređuju "Nova Scope 6000" novije generacije. Korišteni uređaj za ispitivanje dat je na slici 5.



Slika 5: Ultrazvučni uređaj Nova Scope 6000

Uređaj se sastoji iz dva dijela. Lijevi dio služi za slanje ultrazvučnih talasa kroz ispitivani material, dok desni dio služi za prihvatanje podataka, njihovu obradu i prikaz dobijenih rezultata. Između emitera zvučnih talasa i materijala se mora ostvariti dobar kontakt i zbog toga se između njih nalazi i destilovana voda u svrhu sprečavanja prekida ultrazvučnog talasa. Ispitivanje je vršeno u 4 tačke na svakom uzorku (40, 100, 140 i 200 mm od ulaska alata), slika 6.



Slika 6: Prikaz mijernih mjesta na uzorku (mjerjenje se vrši s obje strane)

Povratni talas se šalje na obradu u desni dio uređaja u kome se na osnovu brzine prostiranja ultrazvučnog talasa i vremena određuje pređeni put talasa tj. dimenzija do greške ako postoji ili do drugog kraja ispitivanog materijala ako greška ne postoji [4]. Prikaz dobijenog rezultata dat je na slici 7.



Slika 7: Prikaz dobijenih rezultata ultrazvučnim mjerenjem

3. REZULTATI I DISKUSIJA

3.1. Makro ispitivanje poprečnih presjeka

Na slici 9 prikazani su poprečni presjeci na kojima se vide tuneli, kao posljedica prolaska obrtnog alata uz neadekvatnu geometriju alata i parametre procesiranja. Uočljivo je da su tuneli prisutni na svim uzorcima, osim na posljednjem.



Proba 1 Proba 2 Proba 3 Proba 4 Proba 5 Proba 6 Proba 7

Slika 9: Poprečni presjeci procesiranih uzoraka (vizuelni prikaz tunela)

3.2. Ultrazvučno ispitivanje

Rezultati ultrazvučnog ispitivanja su prikazani u tabelama 4 – 10. Ukupna debljina svih ispitivanih uzoraka iznosila je 8 mm. U tabelama date su dimenzije izmjerene ultrazvukom, sa gornje i donje strane uzorka, 40, 100, 140 i 200 mm od ulaska alata. Pored toga, data je ukupna greška, koja predstavlja dimenziju dobijenog tunela.

Analizom dobijenih rezultata ispitivanja prikazanih u tabeli 4 vidi se da u materijalu postoji tunel cijelom dužinom. Debljina materijala u prolazu alata za procesiranje pored tunela iznosila je 7,92 mm. Oduzimanjem zbira dobijenih rezultata izmjerenih u tačkama s obje strane, od ukupne debljine materijala u prolazu dobila se veličina tunela koja je u probi br. 1 iznosila oko 1mm.

Ukupna debljina materijala u zoni prolaska alata u probi 2 (tabela 5) za procesiranje pored tunela iznosila je 7,89 mm. Kao što se iz dobijenih rezultata može vidjeti veličina tunela je porasla cijelom dužinom za približno 0,5 mm u odnosu na probu br. 1. Ukupna debljina materijala u zoni prolaska alata u probi 3 (tabela 6) pored tunela

iznosila je 7,96 mm. Tunel se i dalje provlačio cijelom dužinom kroz material ali je dimenzija tunela smanjena približno 2,5 puta u odnosu na prethodnu probu.

Tabela 4: Prikaz dobijenih rezultata probe br. 1 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,9		
40 mm od ulaska alata	2,29	4,69	0,94
100 mm od ulaska alata	2,23	4,76	0,93
140 mm od ulaska alata	2,10	4,60	1,22
200 mm od ulaska alata	2,15	4,48	1,29

Tabela 5: Prikaz dobijenih rezultata probe br. 2 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,92		
40 mm od ulaska alata	1,87	4,29	1,73
100 mm od ulaska alata	1,98	4,27	1,64
140 mm od ulaska alata	1,93	4,53	1,43
200 mm od ulaska alata	2,03	4,61	1,25

Tabela 6: Prikaz dobijenih rezultata probe br. 3 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,96		
40 mm od ulaska alata	2,01	5,61	0,34
100 mm od ulaska alata	1,95	5,56	0,45
140 mm od ulaska alata	1,98	5,30	0,68
200 mm od ulaska alata	1,85	4,94	1,14

Na osnovu dobijenih rezultata prikazanih u tabeli 7, može se uočiti da se veličina tunela u probi br. 4 smanjila na približno 0,5 mm ali da i dalje postoji cijelom dužinom prolaska alata. Ukupna debljina materijala u prolazu alata za procesiranje pored tunela iznosila je 7,95 mm.

Analizom dobijenih rezultata iz probe br. 5 (tabela 8) može se zaključiti da promjena parametara procesiranja nije odrađena u pravom smjeru te da se tunel povećao u odnosu na probu br. 4 za približno 0,2 - 0,3 mm. Debljina materijala u zoni prolaska alata pored tunela iznosila je 7,90 mm, dok je veličina tunela bila približno 0,75mm kroz cijelu dužinu prolaska alata.

Na osnovu dobijenih rezultata probe br. 6 prikazanih u tabeli 9 može se konstatovati da je debljina materijala u prolazu pored tunela iznosila 7,56 mm i manja je u odnosu na prethodne probe zbog zakošenja alata za 3°.

Tabela 7: Prikaz dobijenih rezultata probe br. 4 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,95		
40 mm od ulaska alata	1,87	5,49	0,59
100 mm od ulaska alata	1,82	5,63	0,50
140 mm od ulaska alata	1,75	5,55	0,65
200 mm od ulaska alata	1,95	5,39	0,61

Tabela 8: Prikaz dobijenih rezultata probe br. 5 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,89		
40 mm od ulaska alata	1,71	5,48	0,71
100 mm od ulaska alata	1,67	5,45	0,78
140 mm od ulaska alata	1,73	5,47	0,70
200 mm od ulaska alata	1,80	5,29	0,81

Tabela 9: Prikaz dobijenih rezultata probe br. 6 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,56		
40 mm od ulaska alata	1,70	5,75	0,11
100 mm od ulaska alata	1,76	5,77	0,03
140 mm od ulaska alata	1,70	5,72	0,14
200 mm od ulaska alata	1,67	5,68	0,21

Analizom date tabele može se uočiti da su za dati materijal pronađeni optimalni parametri procesiranja pri kojima nema pojave tunela. Kao što se iz tabele vidi, ultrazvučnim ispitivanjem nije pronađena greška u procesiranom uzorku. Debljina materijala u zoni prolaska lata iznosila je 7,45mm.

Tabela 10: Prikaz dobijenih rezultata probe br. 7 ultrazvučnim ispitivanjem

Mjesto mjerenja	Izmjerene vrijednosti [mm]		Ukupna greška [mm]
Pored tunela	7,45		
40 mm od ulaska alata	7,45	7,45	0
100 mm od ulaska alata	7,45	7,45	0
140 mm od ulaska alata	7,45	7,45	0
200 mm od ulaska alata	7,45	7,45	0

4. ZAKLJUČAK

Kod zavarivanja a i kod procesiranja trenjem sa miješanjem, posebnu pažnju treba posvetiti pojavi karakteristične greške tipa tunela. Za ispitivanje pojave tunela, pored makro ispitivanja, vrlo korisno je ultrazvučno ispitivanje, koje može dati kontinualne rezultate po cijeloj dužini prerađenog sloja. Iz dobijenih rezultata ultrazvučnog ispitivanja i na osnovu vršenih varijacija procesiranja trenjem sa miješanjem, može se zaključiti da su najuticajniji parametri geometrija alata, njegov nagib, kao i broj obrtaja alata i brzina translatorsnog kretanja. U ovom slučaju alatom sa navojem je postignuto najbolje miješanje i najveći pritisak materijala te je s toga izbjegnuta greška. Tuneli u većini slučajeva imaju tendenciju povećanja, što je u vezi sa smanjivanjem pritiska na samom kraju obratka zbog prisustva slobodne površine uzorka.

5. ZAHVALNOST

Rezultati prezentovani u ovom radu su realizovani u okviru projekta „Istraživanje i primena savremenih metoda karakterizacije materijala i zavarenih spojeva u proizvodnom mašinstvu“, Departmana za proizvodno mašinstvo, FTN Novi Sad.

6. LITERATURA

- [1] M.W. Mahoney, Science Friction, Welding and Joining, Jan/Feb 1997, 18–20;
- [2] D. Labus, S. Balos, M. Dramicanin, P. Janjatovic, L. Sidjanin, Optimization of tool geometry and friction stir processing parameters in AA5754 H111 alloy, Proceedings of PSU-UNS ICET-2017, T11-1.4
- [3] S. Rešković, Predavanja iz predmeta Ispitivanje materijala, Metalurški fakultet Sveučilišta u Zagrebu, 2009 g.
- [4] V. Mihaljević, Detekcija zavarova u metalnim šavovima, Diplomski rad, Fakultet strojstva i brodogradnje, Sveučilište u Zagrebu, 2007 g.

Kratka biografija:



Nemanja Spasojević rođen je u Bijeljini 1992. god. Master rad na Fakultetu tehničkih nauka iz oblasti Mašinstvo – Savremene tehnologije oblikovanja materijala odbranio je 2017. god.



Prof. dr Sebastian Baloš rođen je u Somboru 1974. god. Doktorirao je na Fakultetu tehničkih nauka 2010. god., iz oblasti Materijali i tehnologije spajanja. 2016. god. stekao je zvanje vanrednog profesora.

KONTROLA SLOŽENOSTI LOZINKI ZA WINDOWS OPERATIVNE SISTEME PASSWORDS COMPLEXITY CONTROL FOR WINDOWS OPERATING SYSTEMS

David Vuletić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je analizirana kontrola složenosti lozinke u okviru Windows operativnog sistema. Prikazan je WinPasswordFilter alat namenjen kreiranju polisa za lozinke u okviru Windows operativnog sistema. Prikazana je i opisana specifikacija i implementacija alata.

Abstract – This paper analyzes password complexity control for Windows operating systems. It presents the WinPasswordFilter tool for creating password policies for the Windows operating system. The description and specification of the implemented tool are given.

Ključne reči: Lozinka, Bezbednost, Filter, Polisa, Windows

1. UVOD

Lozinke su jedan od najrasprostranjenijih vidova autentifikacije korisnika u svetu računara i računarskih mreža, što ih čini izuzetno značajnim za zaštitu podataka i zaustavljanje neovlašćenog pristupa zaštićenim resursima [1]. Ukoliko neko sa malicioznim namerama uspe da pristupi korisničkom nalogu koji je zaštićen lozinkom, može napraviti veliku štetu korisniku ili organizaciji kojoj korisnik pripada.

Da bi korisnici zaštitili svoje naloge, nije dovoljno samo da uspešno čuvaju svoje lozinke, nego i da naprave dovoljno sigurne lozinke da hakeri ne mogu lako da ih otkriju. Korisnici su skloni da, radi lakšeg pamćenja, kao lozinke koriste podatke o sebi, uzastopne brojeve, uzastopne karaktere na tastaturi, reči kao što su *password* ili *welcome* i slično. Takođe, lozinke su im često kratke, i često koriste jednu lozinku za sve naloge. Ovakve lozinke su lake mete za napade.

Kako bi osigurale svoje sisteme, kompanije moraju na neki način da primoraju korisnike da prave kompleksne (jake) lozinke. Ovo čine pravljjenjem polisa za lozinke (*Password Policy*). Polisa za lozinke je skup pravila kojima se lozinke korisnika jednog sistema moraju povinovati. Mogu se odnositi na sve korisnike, ili samo na određenu grupu korisnika sistema. Pomoću ovakvih polisa se korisnicima postavljaju ograničenja pri kreiranju lozinke i na taj način se vrši kontrola njihove složenosti. Zadatak ovog rada je da objasni na koji način je moguće napraviti dobru polisu za lozinke u okviru Windows operativnog sistema, odnosno da opiše jedan alat specijalno kreiran u ovu svrhu, koji otklanja neke od mana koje poseduje ugrađeni alat Windows-a.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Goran Sladić, vanr. prof.

1.1. Windows polisa za lozinke

U svim novijim verzijama, Windows dolazi sa alatom za pravljenje polise za lozinke. Ovaj alat se nalazi u okviru bezbednosnih podešavanja operativnog sistema i nudi opcije za podešavanje polise kao što su: pamćenje i zabrana ponovnog korišćenja istih lozinke, minimalno i maksimalno vreme korišćenja jedne lozinke, minimalan broj karaktera u okviru lozinke, šifrovano skladištenje lozinke i opciju uključivanja zahteva za kompleksnim lozinkama. Kompleksnim lozinkama se smatraju lozinke koje sadrže bar tri od ukupno pet vrsta karaktera u okviru sebe [2].

Međutim, ovaj Windows-ov alat za podešavanje polise za lozinke ima nekoliko bitnih nedostataka. Pamćenje prethodnih korisnikovih lozinke sprečava samo ponovno korišćenje identičnih lozinke, tako da njihove blage varijacije mogu da budu prihvaćene. Na primer, korisnik može samo svaki put da inkrementira broj na kraju lozinke, a da njenu osnovu uopšte ne menja. Nakon *mypassword1*, može da stavi *mypassword2* i na taj način skoro potpuno obesmisli navedeni zahtev. Zatim, čak i sa uključenim zahtevom za kompleksnim lozinkama, korisnik može da koristi veoma slabe lozinke kao što su *Password123*, *LetMeIn2017*, *W3lcom3*, *Passw0rd*, *January1* i slične, što znači da nivo zaštite nije na odgovarajućem nivou [3].

1.2. Filter lozinke

Mnoge organizacije zahtevaju značajno veći nivo zaštite sistema i za to im je neophodan ozbiljniji alat za implementaciju polisa za lozinke od onog koji Windows nudi. Rešenje za ovaj problem je filter lozinke (*Password Filter*).

Filter lozinke je program koji administraciji nudi način da od korisnika zahteva da poštuje određena pravila pri kreiranju lozinke, odnosno omogućava pravljenje ozbiljne polise za lozinke upotrebljive u realnom sistemu kojem je neophodan visok nivo zaštite [4]. Implementira se pisanjem dinamičke biblioteke (*dll*). Velika prednost filtera za lozinke je njegova prilagodljivost, s obzirom da daje apsolutnu slobodu administratoru pri kreiranju polise.

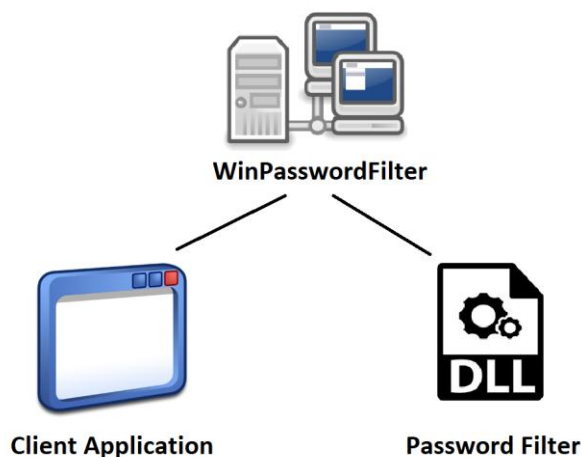
2. STRUKTURA I OPIS ALATA

Da bi se prevazišli propusti ugrađenih alata za održavanje kvaliteta korisničkih lozinke u okviru Windows operativnog sistema, osmišljen je WinPasswordFilter alat. Ovaj alat omogućava administratorima da, na jednostavan način, u sistemu implementiraju složenije polise za lozinke. Glavni prednosti ovog alata u odnosu na Windows Password Policy su mogućnost kreiranja i održavanja rečnika zabranjenih reči, kao i detaljnija

specifikacija neophodnih tipova karaktera u okviru lozinke.

2.1. Struktura alata

WinPasswordFilter se sastoji iz dve ključne komponente: dinamičke biblioteke - *WinPasswordFilter.dll*, i klijentske aplikacije sa grafičkim korisničkim interfejsom – *WinPasswordFilterClient*. Na slici 1 je ilustrovana struktura alata.



Slika 1. Struktura alata

Biblioteka *WinPasswordFilter.dll*, u daljem tekstu oslovljena samo kao filter, predstavlja srž ovog alata. To je dinamička biblioteka čije funkcije operativni sistem poziva radi evaluacije lozinke. U okviru filtera je implementirana logika za ispitivanje usklađenosti lozinke po svim stavkama polise.

Klijentska aplikacija je prpratna komponenta ovog alata, bez koje ne bi bilo moguće napraviti konfigurabilnu polisu za lozinke. U okviru nje administratori sistema vrše specifikaciju polise, na osnovu koje potom filter može da evaluiira lozinke pri njihovom postavljanju.

Komunikacija između klijentske aplikacije i filtera se vrši putem datoteka u fajl sistemu računara. Klijentska aplikacija piše podatke o polisi, koje potom filter čita. Pristup direktorijumu, u kojem se nalaze datoteke pomoću kojih komuniciraju filter i klijentska aplikacija, omogućen je samo administratorima.

Ovo je izuzetno bitno sa stanovišta bezbednosti jer sprečava obične korisnike da, slučajno ili namerno, naprave neovlašćene promene u polisi za lozinke ili rečnicima, ili onesposobe filter.

2.2. Opis funkcija alata

Administrator sistema, pri kreiranju polise za lozinke pomoću *WinPasswordFilter* alata, može da podešava sledeće opcije:

- Minimalan broj karaktera
- Maksimalan broj karaktera
- Minimalan broj malih slova
- Minimalan broj velikih slova
- Minimalan broj cifara
- Minimalan broj specijalnih karaktera
- Zabrana korisničkih podataka u okviru lozinke
- Zabrana reči iz rečnika parcijalnog poklapanja
- Zabrana reči iz rečnika kompletnog poklapanja

- Dodavanje i brisanje reči iz rečnika parcijalnog poklapanja
- Dodavanje i brisanje reči iz rečnika kompletnog poklapanja

Podešavanje minimalnog broja karaktera je postavljanje broja koji označava koliko lozinka mora najmanje da ima karaktera da bi bila prihvaćena. Što više karaktera lozinka ima, to ju je teže otkriti, i primoravanjem korisnika na duže lozinke povećava se bezbednost sistema [5]. *WinPasswordFilter* dozvoljava podešavanje minimalnog broja karaktera od 0 do 100.

Podešavanjem maksimalnog broja karaktera postavlja se gornja granica za dužinu korisničkih lozinki. Ukoliko je ova stavka postavljena na nulu, onda ne postoji ograničenje maksimalne dužine, odnosno ova provera je isključena. Ograničavanje dužine lozinke može da bude korisno u sprečavanju korisnika da koriste predugačke lozinke koje će zaboraviti i na taj način opterećivati administratora sistema, koji će biti zadužen za otključavanje naloga.

Moguće je podešavati i minimalan broj neophodnih malih slova, velikih slova, cifara i specijalnih znakova. Specijalni znaci su svi karakteri koji nisu malo slovo, veliko slovo ili cifra. Dakle, to su znakovi interpunkcije, svi tipovi zagrada, oznake valuta, matematički operatori, taraba, znak procenta, donja crta, tilda i mnogi drugi. Ovim opcijama se korisnik primorava na korišćenje kompleksnijih lozinki, sa karakteristikama različitih tipova.

Što se tiče rečnika, postoje dve vrste: rečnik parcijalnog poklapanja i rečnik kompletnog poklapanja. U rečnike se upisuju reči, odnosno nizovi znakova koji će biti zabranjeni u okviru lozinke. Jedina razlika između ova dva rečnika je u načinu poređenja njihovih stavki sa lozinkom. Iz rečnika se mogu i brisati reči, i oni se mogu isključivati i uključivati.

Kod rečnika sa kompletnim poklapanjem, proverava se da li je čitava lozinka jedna od stavki u tom rečniku. Ukoliko je lozinka identična kao neka od stavki u rečniku, onda se ta lozinka ne može koristiti. Dakle, ukoliko korisnik pokuša da podesi lozinku 1234, a rečnik sa kompletnim poklapanjem sadrži reč 1234, lozinka neće biti prihvaćena. Ova reč iz rečnika neće zabraniti upotrebu lozinke 12345.

Kod rečnika sa parcijalnim poklapanjem, gleda se da li lozinka u sebi sadrži neku od stavki iz rečnika. Kao i u prethodnom primeru, ako rečnik sadrži reč 1234, lozinka 1234 neće biti prihvaćena. Ali, u ovom slučaju filter neće prihvatiti ni lozinku 12345, jer ona sadrži u sebi navedenu reč 1234 iz rečnika.

Poređenje lozinke sa rečima iz rečnika je implementirano tako da ne pravi razliku između velikih i malih slova (*case insensitive*). Ovim se znatno smanjuje broj neophodnih reči u rečniku, i učinjeno je radi optimizacije alata.

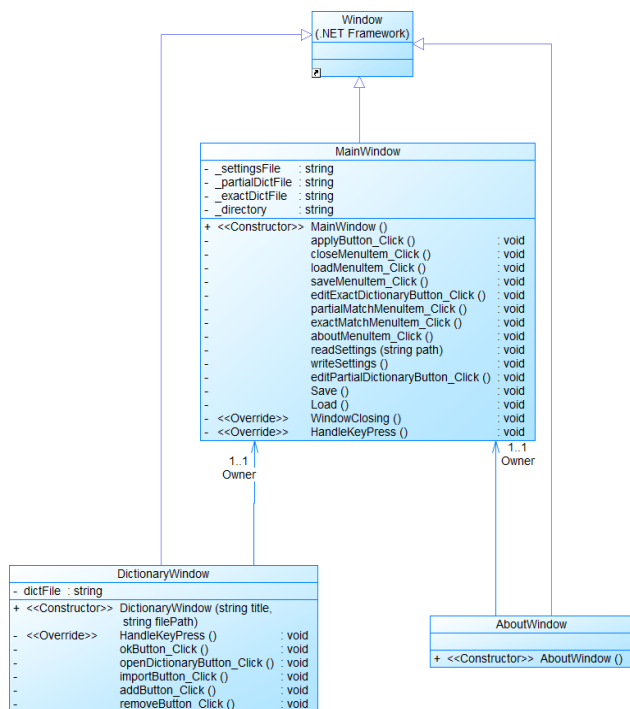
Osim reči iz rečnika, moguće je zabraniti korisniku korišćenje podataka o sebi u okviru lozinke. Konkretno, u okviru *Windows* operativnog sistema, to su korisničko ime (*Account Name*) i puno ime korisnika (*Full Name*). Filter puno ime korisnika podeli na tokene po razmacima i brani korišćenje bilo kog od tih tokena. Na ovaj način korisnik ne može da koristi ni ime, ni prezime u okviru lozinke, kao ni korisničko ime.

Osim funkcija vezanih za kreiranje polise, *WinPasswordFilter* nudi još neke mogućnosti koje pomažu u radu alata. Te funkcije su: čuvanje podešavanja polise, učitavanje podešavanja polise, uvoz rečnika i izvoz rečnika.

2.3. Model klijentske aplikacije

Na slici 2 je prikazan dijagram klasa klijentske aplikacije. Osnovna klasa od koje se polazi je *Window* klasa *.NET Framework*-a. Ovu klasu nasleđuju svi prozori u klijentskoj aplikaciji *WinPasswordFilter*-a: *MainWindow*, *DictionaryWindow* i *AboutWindow*.

MainWindow klasa predstavlja glavni prozor grafičkog korisničkog interfejsa aplikacije. Ona sadrži raspored glavnih kontrola za podešavanje polise za lozinke, a u njenim metodama je implementirana većina osnovnih funkcija alata. Klasa *DictionaryWindow* se odnosi na modalni potprozor glavnog prozora. Zadužena je za funkcije kao što su rukovanje stavkama rečnika, i izvoz i uvoz rečnika, koje su implementirane u njenim metodama. *AboutWindow* je jednostavna klasa koja sadrži samo informacije o rasporedu tekstova sa osnovnim podacima o alatu u okviru prozora.



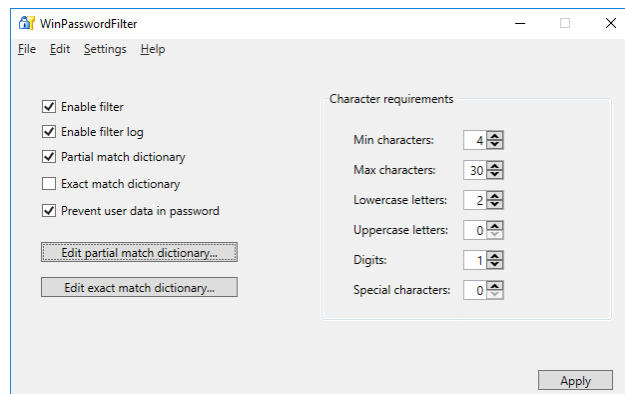
Slika 2. Model klasa klijentske aplikacije

3. IMPLEMENTACIJA REŠENJA

Filter je pisan u programskom jeziku C++ [6], pošto *Microsoft* ne savetuje pisanje procesa operativnih sistema u jezicima visokog nivoa [7]. Klijentska aplikacija je pisana u programskom jeziku C# [8], tačnije korišten je WPF (*Windows Presentation Foundation* [9]). Za kreiranje instalera korišćeni su *Visual Studio Installer* projekti [10]. Čitav alat je implementiran kroz više različitih projekata u okviru jednog *Solution*-a u razvojnom okruženju *Microsoft Visual Studio 2015*.

Na slici 3 je prikazan glavni prozor klijentske aplikacije, u okviru kojeg se nalaze kontrole za podešavanje svih parametara polise za lozinke, isključujući dodavanje i

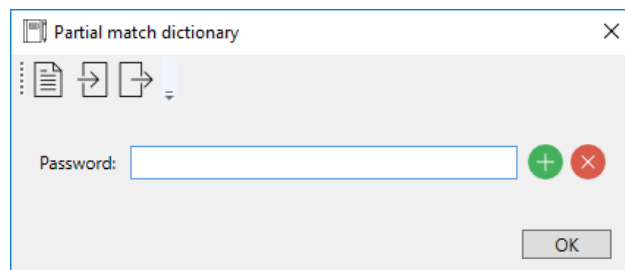
brisanje reči u i iz rečnika. Na levoj strani se nalazi niz polja za potvrdu koja služe za uključivanje i isključivanje određenih opcija polise, kao i uključivanje i isključivanje same polise. Ispod njih se nalaze dva dugmeta, kojima se otvaraju modalni prozori za manipulaciju rečnicima. S desne strane su numerička polja za podešavanje neophodnih karaktera u okviru lozinke. U ova polja je moguće unositi samo brojeve, a njihovo postavljanje je moguće direktno u polju ili putem strelica. U donjem desnom uglu glavnog prozora aplikacije nalazi se dugme za primenu aktuelnih podešavanja.



Slika 3. Glavni prozor klijentske aplikacije

Na slici 4 je prikazan prozor za manipulaciju rečnikom. Za oba rečnika se koriste isti prozori, koji se jedino razlikuju po nazivu i slici u naslovnoj traci. Ispod naslovne trake se nalazi paleta sa alatima, koja sadrži tri dugmeta. Prvo dugme služi za otvaranje tekstualne datoteke koja sadrži reči aktuelnog rečnika. Drugo dugme je za uvoz eksternog rečnika, i ono otvara novi prozor za izbor datoteke iz koje se uvozi rečnik. Treće dugme služi za izvoz aktuelnog rečnika.

U centralnom delu prozora se nalazi tekstualno polje za unos lozinke. Lozinka unesena u polje se može dodati u rečnik pritiskom tastera *Enter* ili klikom na zeleni plus koji se nalazi pored polja. Brisanje lozinke se vrši na atipičan način: lozinka koja se želi obrisati iz rečnika se unese u tekstualno polje i potom se pritisne crveni „iks“. Ovakav koncept brisanja je implementiran zbog potencijalne obimnosti rečnika. U slučaju desetina ili stotina hiljada stavki u rečniku, što bi u realnom sistemu trebalo da je normalna pojava, nepraktično bi bilo prikazati sve stavke i vršiti brisanje odabirom željene reči.



Slika 4. Prozor za manipulaciju rečnikom

4. ZAKLJUČAK

Većina računarskih sistema koristi lozinke za autentifikaciju svojih korisnika. Da bi se postigao adekvatan nivo bezbednosti u ovakvim sistemima,

neophodno je primorati korisnike na korišćenje jakih lozinki, jer čak i jedna slaba lozinka može omogućiti napadaču upad u sistem. Ovaj problem se rešava kreiranjem polise za lozinke.

Windows kao operativni sistem poseduje sopstveni alat za kreiranje ovakvih polisa. Međutim, ovaj alat administratorima sistema ne daje dovoljno mogućnosti pri kreiranju polisa, tako da sistem koji koristi *Windows Domain* ostavlja ranjivim. Ipak, Microsoft nudi način da se ovaj problem prevaziđe, a to se čini pisanjem sopstvene dinamičke biblioteke – filtera za lozinke.

Upravo na taj način je i kreiran alat *WinPasswordFilter*, koji je i opisan u ovom radu. Zadatak ovog alata je da nadomesti nedostatke ugrađenog Windows alata i omogući kreiranje kvalitetne polise za lozinke u okviru *Windows Domain*-a ili pojedinačnog računara sa Windows operativnim sistemom.

Predstavljeno rešenje bi se moglo poboljšati omogućavanjem kreiranja granuliranih polisa, odnosno da se različitim korisnicima ili ulogama u sistemu mogu postaviti različite polise za lozinke. Osim toga, mogla bi se promeniti poruka koju operativni sistem prikaže pri odbijanju lozinke, s obzirom da trenutna ne specificira zbog čega je ona neadekvatna.

Ovaj problem bi se mogao ublažiti i generisanjem teksta o zahtevima polise, koji bi se prosledio svim korisnicima sistema.

Što se same polise tiče, mogla bi se dodati ograničenja kao što su: maksimalan broj instanci jednog znaka, maksimalan broj uzastopnih karaktera, minimalan broj različitih znakova, pamćenje i istek lozinke.

Sa tehničke strane, moglo bi se ubrzati poređenje lozinke sa rečima u rečniku. Alat trenutno radi dobro sa deset hiljada reči u rečniku, prihvatljivo sa sto hiljada, a sa milion stavki već ima ozbiljnih problema.

Ovaj problem bi se mogao rešiti implementacijom naprednije pretrage u tekstualnim datotekama koje sadrže reči iz rečnika, ili korišćenjem neke manje baze podataka (na primer: *SQLite*) umesto tekstualne datoteke.

5. LITERATURA

- [1] R. Duncan, *An Overview of Different Authentication Methods and Protocols*, Sans Institute, 2002.
- [2] Microsoft, [https://technet.microsoft.com/en-us/library/hh994572\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/hh994572(v=ws.11).aspx)
- [3] nFrontSecurity, <https://www.slideshare.net/nFrontSecurity/the-windows-password-policy-is-not-enough>
- [4] nFrontSecurity, <https://www.slideshare.net/nFrontSecurity/how-do-password-filters-work>
- [5] InfoSecurity Magazine, <https://www.infosecurity-magazine.com/news/longer-password-harder-to-crack/>
- [6] CPlusPlus, <http://www.cplusplus.com/info/description/>
- [7] Microsoft, <https://support.microsoft.com/en-us/help/841927/do-not-use-high-level-languages-libraries-frameworks-virtual-machines>
- [8] Microsoft, <https://docs.microsoft.com/en-us/dotnet/csharp>
- [9] Microsoft, <https://docs.microsoft.com/en-us/dotnet/framework/wpf/>
- [10] Microsoft, [https://msdn.microsoft.com/en-us/library/aa243946\(v=vs.60\).aspx](https://msdn.microsoft.com/en-us/library/aa243946(v=vs.60).aspx)

Kratka biografija:



David Vuletić rođen je u Subotici 1993. godine. Upisuje se na Fakultet tehničkih nauka, odsek računarstvo i automatika, 2012. godine. Školske 2014/2015. godine upisuje smer Primenjene računarske nauke i informatika. Osnovne studije završava u septembru 2016. godine.

Master studije na Fakultetu tehničkih nauka, odsek računarstvo i automatika, usmerenje elektronsko poslovanje, upisuje 2016. godine, a završava ih u oktobru 2017.

**RAZVOJ GENERATORA ZA OPTIMALNO KONFIGURISANJE MODELA
DISTRIBUTIVNE MREŽE****DEVELOPMENT OF GENERATOR FOR OPTIMAL DISTRIBUTION NETWORK
MODEL CONFIGURING**

Lazar Prodanović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu je opisan algoritam za određivanje optimalnog broja i rasporeda distributivnih podstanica i izvoda. Analizirani su i razvijeni algoritmi za klasterizaciju distributivnih izvoda i podstanica po unapred definisanim parametrima, kao i algoritam za određivanje broja generisanih kontejnera opreme.

Abstract – In this paper an algorithm for determining the optimal number and layout of distribution substations and feeders is described. Algorithms for clustering feeders and substations are analyzed and developed, as well as an algorithm for determining the number of generated equipment containers.

Ključne reči: Model podataka distributivne mreže, CIM, GDA, klasterizacija, optimizacioni algoritmi

1. UVOD

Sa povećavanjem populacije ljudi na Zemlji, raste i potreba za strujom. Mreže vremenom postaju sve veće i kompleksnije, a DMS sistemi zahtevaju sve više hardverskih resursa za rad. Kako bismo odredili neophodne fizičke specifikacije sistema, potrebno je na neki način opteretiti sistem. U ovom radu opisan je način generisanja modela distributivne mreže proizvoljne veličine, koja u tome pomaže.

U okviru ovog rada biće prikazane performanse algoritma za određivanje činilaca modela distributivne mreže, koji bi mogao pomoći u testiranju specifikacija DMS sistema.

**2. DISTRIBUTIVNA ELEKTROENERGETSKA
MREŽA**

Distributivni sistemi predstavljaju poslednji stepen u isporuci električne energije i obično se sastoje iz tri komponente: distributivne podstanice, primarnog i sekundarnog distributivnog izvoda. Na nivou distributivnog sistema, naponski nivo se snižava i energija se u manjim količinama distribuira do potrošača. Jedna podstanica će napajati energijom veliki broj potrošača. Distributivni sistem napaja se kroz distributivnu podstanicu.

Dizajn podstanica zavisi od velikog broja parametara i postoji gotovo beskonačno mnogo mogućnosti za njegovu realizaciju. Za tipičnu podstanicu, napon na visokonaponskoj strani može biti bilo gde između 34.5 kV, pa skroz do 345 kV. Prosečan napon na visokoj strani je između 115 i 138 kV. Na niskonaponskoj strani se obično nalaze dva ili više izvoda, spojenih preko prekidača (Tie switch) [1].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Darko Čapko.

Primarni distributivni izvodi prenose električnu energiju na srednjem naponu do distributivnog transformatora, bliže mestu neposredne potrošnje. Svrha ovog transformatora jeste da smanji primarni napon na nivo koji može biti upotrebljen od strane potrošača. Ta električna energija niskog napona se potom, posredstvom sekundarnog distributivnog izvoda, prenosi do krajnjih korisnika. Dužina izvoda je usko povezana sa opterećenjem mreže na datoj lokaciji. Potrošači koji traže veću snagu se mogu povezati direktno na primarni distributivni izvod ili potprenosnu mrežu [1].

3. MODEL PODATAKA DISTRIBUTIVNE MREŽE

U poslednje dve decenije postoji intenzivno interesovanje za standardizovano modelovanje podataka distributivne mreže. CIM [2][3] je apstraktni informacioni model koji se koristi za modelovanje električne mreže i različite opreme, upotrebene unutar mreže. Korišćenjem zajedničkog (eng. common) modela podataka, omogućena je razmena podataka između aplikacija različitih proizvođača, čime kompanije i proizvođači smanjuju cenu integracije tih aplikacija unutar sistema, što dozvoljava da se više sredstava uloži u povećanu funkcionalnost upravljanja i optimizovanja električnog sistema. Pored toga, definisanjem CIM standarda, proširuje se svest modelara o različitim aspektima, uređajima i njihovim karakteristikama, koji se potencijalno greškom ne bi razmatrali niti uključili u model.

Informacioni model je apstraktna i formalna reprezentacija objekata, njihovih atributa, njihove povezanosti sa ostalim objektima, i ponašanja i operacija koje mogu biti izvedene nad njima. Modelovani objekti mogu biti fizički objekti, kao što su uređaji u elektroenergetskoj mreži, ili i oni sami mogu apstraktni. Primarna svrha koncepta informacionog modela jeste da formalno opiše domen problema, bez ograničenja kako će taj opis biti mapiran na stvarnu softversku implementaciju. Informacioni model je obično i sam po sebi formalno opisan na poseban, dobro definisan način, tipično strogim jezikom ili pomoću tehnologije dijagrama. U mnogim slučajevima informacioni model može biti predstavljen na različite načine korišćenjem formalnih prevodioca. [2]

IEC standard 61970-301 [3] je informacioni model koji opisuje komponente elektroenergetskog sistema na električnom nivou, i opisuje veze između komponenti. IEC 61968-11 [3] proširuje taj model kako bi pokrio ostale aspekte razmene podataka softvera energetskog sistema, kao što su praćenje imovine, planiranje rada i naplata usluga. Ova dva standarda (61970-301 i 61968-11) su zajedno poznati kao Common Information Model (CIM) za elektroenergetske sisteme i trenutno imaju dve primarne

uloge: da olakšaju razmenu podataka elektroenergetske mreže između kompanija, i da dozvole razmenu podataka između aplikacija unutar same kompanije [3]. Formalna definicija ovog modela dobija se korišćenjem UML-a [2].

CIM je izgrađen na nekoliko osnovnih tehnologija:

- Unified Modeling Language (UML)
- eXtensible Markup Language (XML)
- Resource Description Framework (RDF)

4. GDA

GDA [4] standard omogućava pristup podacima bez ikakvog znanja o logičkoj šemi koja se koristi za unutrašnje skladištenje podataka - implementaciji. Dovoljno je poznavanje definisanog modela, a u ovom slučaju je to CIM.

GDA specifikacija opisuje dve forme GDA upita. Prvi, prvobitno standardizovan u okviru OMG-a (Object Management Group) kao Utility Management System (UMS) Data Access Facility (DAF) [5], pruža klijentima osnovnu mogućnost postavljanja upita nad modelom. Drugi proširuje UMS DAF kako bi klijentima obezbedio naprednije mogućnosti poput filtriranja i spajanja upita. S obzirom da se u ovom radu koriste samo osnovni upiti, obradimo samo prvu formu.

DAF standard formuliše upite i odgovore u vidu resursa, osobina i njihovih vrednosti. Odgovori se zajedničkim imenom nazivaju opis resursa.

Upiti nad podacima se postavljaju posredstvom Resource Query servisa, na osnovu čijih operacija dobijamo opise resursa, kao odgovor. Ovaj interfejs obezbeđuje tri osnovne operacije, kao i jednu naprednu. Osnovne operacije su: `get_values()`, `get_extent_values()`, i `get_related_values()`. Četvrta operacija, `get_descendent_values()`, je generalizacija prethodne tri i omogućava veću optimizaciju upita [4].

Svaka operacija ovog interfejsa izvršava samo jedan upit. Svaki opis resursa koji se vrati sadrži vrednosti osobina za podskup osobina koji je prosleđen operaciji. Vrednosti osobina se vraćaju u istom redosledu u kom su prosleđene. Ukoliko vrednost neke osobine nije dostupna za dati resurs, ili je prosleđeni identifikator osobine pogrešan, ona se preskače [4].

get_values() upit zahteva opis resursa za jedan resurs, na osnovu prosleđenog identifikatora resursa [4].

get_extent_values() upit zahteva opis za svaki resurs date klase. Klasa se prosleđuje preko identifikatora klase, koji predstavlja identifikator resursa [4].

get_related_values() upit zahteva opis za svaki resurs koji je u određenoj vezi sa prosleđenim izvornim resursom. Izvorni resurs se specifikira pomoću identifikatora resursa, a asocijacija pomoću određenih parametara asocijacije. Provajder informacija ispituje asocijacije izvornog resursa, na osnovu čega se vraća nula ili više resursa kao rezultat. Za svaki rezultujući resurs, provajder ispituje prosleđene osobine i stvara opis resursa, koji se vraća putem iteratora [4].

get_descendent_values() upit je generalizacija prethodnih i osmišljen je kako bi klijenti mogli da postave upit na

opšti način. Takođe, sa strane provajdera stvara bolju priliku za optimizaciju [4].

ResourceUpdateService interfejs nudi operacije za ažuriranje podataka koji su dostupni putem GDA upita [4].

apply_updates() operacija zahteva ažuriranje modela na osnovu DifferenceModel strukture. DifferenceModel struktura sadrži četiri stavke:

1. zaglavlje – sadrži informacije o promenama, kao što su autor, datum i svrha
2. promene unapred – sadrži niz opisa resursa koji se nalaze u modelu nakon uspešnog ažuriranja, ali ne pre toga
3. promene unazad – sadrži niz opisa resursa koji se nalaze u modelu pre primene ažuriranja, ali ne i posla toga
4. preduslove za primenu ažuriranja

Svaka od ovih stavki može biti prazna [4].

Prilikom poziva operacije `apply_updates()` vrši se provera preduslova i ažuriranje, takvo da odražava ono što se nalazi unutar nizova promena unapred i unazad [4].

U našem modelu, željene izmene se u obliku opisa resursa čuvaju unutar objekta tipa Delta. Delta objekat sadrži 3 tipa operacija: dodavanje, ažuriranje i brisanje. Dete, tj. izmene, se mogu trajno sačuvati u obliku XML fajla.

5. MATEMATIČKI OPIS ZADATKA

Za zadati niz objekata veličine n , gde svaki objekat ima m parametara, potrebno je odrediti skup S , sačinjen od tih objekata, takav da suma po svakom parametru u skupu S bude približno jednaka unapred zadatoj vrednosti. Svaki objekat se u skupu S može ponavljati neograničen broj puta. Svi parametri objekata su celobrojne vrednosti.

Cilj je minimizovati:

$$\sum_{j=1}^m \left(\frac{P_j - \sum_{i=1}^n X_i p_{ij}}{P_j} \right)^2$$

gde je:

- $O_1, O_2, \dots, O_n$ – zadati niz objekata
- $p_1, p_2, \dots, p_m$ – vrednosti parametara objekta
- $P_1, P_2, \dots, P_m$ – zadate vrednosti parametara skupa S
- $X_1, X_2, \dots, X_n$ – broj ponavljanja svakog objekta u skupu S

Zadati niz objekata $O_1, O_2, \dots, O_n$ fizički predstavlja skup distributivnih podstanica i distributivnih izvoda. Svaki od elemenata skupa ima 4 parametra od značaja. Ti parametri su, redom:

1. Broj transformatora
2. Broj signala
3. Broj SCADA tačaka
4. Broj potrošača

Kako distributivne podstanice nemaju direktnih potrošača, smatra se da je u tom slučaju (za objekte tipa distributivna podstanica) vrednost ovog parametra jednaka nuli.

Potrebno je, za unapred zadate vrednosti parametara distributivne mreže, odrediti skup podstanica i izvoda, koji najbolje zadovoljava prethodno navedeni kriterijum optimalnosti.

Zadati objekti (podstanice i izvodi) i njihovi parametri se izvlače iz CIM-baziranog modela, posredstvom GDA upita.

S obzirom da ima puno zadatih objekata, vrši se k-means klasterizacija ili grupisanje podataka, kako bi se izbacili objekti sa sličnim vrednostima parametara i ubrzalo izvršavanje algoritma.

Klasterizacija podataka predstavlja proces svrstavanja nizova podataka u grupe, tako da se slični nizovi nalaze u istoj grupi(klasteru). Kako je naš cilj da smanjimo skup podataka nad kojim radi program, klasterizacija sama po sebi nam neće puno pomoći. Nakon što su svi objekti razvrstani po klasterima, potrebno je iz svakog klastera izvući predstavnika. To ćemo uraditi tako što ćemo za svaki objekat iz klastera izračunati njegovu prostornu udaljenost od centroida. Kao predstavnika biramo onaj objekat čija je udaljenost najmanja. Na taj način ćemo od prvobitnog skupa elemenata dobiti skup veličine zadatog broja klastera.

6. IMPLEMENTACIJA

U ovom poglavlju je opisan sam algoritam za određivanje optimalne konfiguracije modela distributivne mreže, koji radi nad podacima opisanim u prethodnom poglavlju.

Priloženi algoritam radi na principu "pohlepnog" algoritma, što znači da u svakom trenutku donosi odluke na osnovu trenutnog stanja i zarad trenutnog boljitka. Međutim, u ovaj konkretni algoritam ubačena je mala doza "pameti", kako bi rešenje na kraju ipak bilo što bliže globalno najboljem rešenju.

Ono što još izdvaja ovaj algoritam, jeste postojanje više iteracija, gde se u svakoj iteraciji traži zasebno rešenje problema, ali sa drugačijim skupom podataka, iste veličine, dobijenim primenom prethodno pomenute modifikovane klasterizacije. Drugačiji skupovi podataka dobijaju se zahvaljujući generatoru slučajnih brojeva, definisanom pre početka svih iteracija. Ti skupovi podataka, iako drugačiji međusobno, mogu se ponavljati za svako naredno pokretanje programa, postavkom istog semena(eng. seed) za generator slučajnih brojeva. U ovom slučaju, algoritam je deterministički, jer će konačno rešenje uvek biti isto za iste početne uslove. U suprotnom, svakim pokretanjem programa biće generisani novi skupovi podataka, što će dovesti do različitih rešenja, pa stoga i stohastičke prirode algoritma.

Kako postoji više iteracija i više rešenja problema, za konačno rešenje biće izabrano ono koje najbolje zadovoljava zadati kriterijum optimalnosti.

Pre početka algoritma, neophodno je definisati željeni broj iteracija, kao i željeni broj klastera, odnosno veličinu skupa podataka nad kojim se traži rešenje.

U nastavku, dat je pseudo kod algoritma:

- Prolazi kroz petlju dok ne odradiš zadati broj iteracija

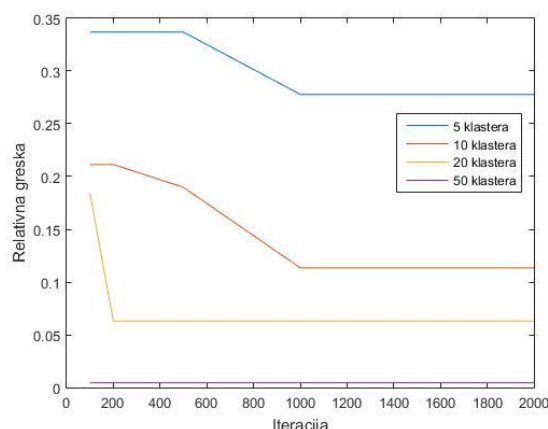
- Prikupi podatke i izvuci predstavnike klastera
- Parametre predstavnika upiši u matricu
- Izračunaj početnu sumu relativnog kvadratnog odstupanja po parametrima
- Prolazi kroz petlju dok god je prethodna suma relativnog kvadratnog odstupanja veća od trenutne
 - Definiši matricu profita* po parametrima za sve objekte
 - Odredi srednju vrednost profita po parametrima za svaki objekat
 - Pronađi objekat sa najmanjim odstupanjem profita od srednje vrednosti
 - Dodaj izabrani objekat u mrežu i preračunaj sumu relativnog kvadratnog odstupanja po parametrima
- Za dati prolaz upiši preostala odstupanja po svakom parametru u matricu
- Izračunaj sumu relativnog kvadratnog odstupanja za svaki objekat i odredi konačno rešenje sa najmanjom sumom
- Kraj

* profit nekog objekta predstavlja relativno odstupanje od zadatih parametara distributivne mreže, ukoliko se u mrežu doda taj objekat

Nakon određivanja optimalnog broja ponavljanja svakog objekta u modelu distributivne mreže sa zadatim parametrima, potrebno je objekte, odnosno resurse, dodati u prazan model, posredstvom Delta objekta i GDA interfejsa, kako bi mreža bila izgenerisana.

7. REZULTATI

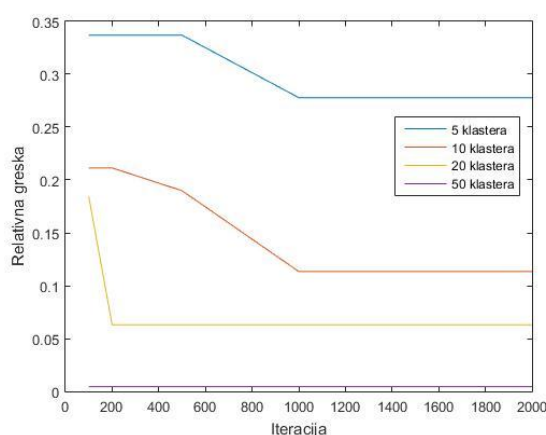
Algoritam opisan u prethodnom odeljku pokretan je nad test skupom, koji se sastoji iz 8 nizova željenih parametara distributivne mreže. Za svaki niz željenih parametara algoritam je pokretan više puta, sa različitim zadatim brojem iteracija, kao i brojem klastera. U



su prikazani dobijeni rezultati za najbolju kombinaciju broja iteracija i broja klastera u datom testu. U belim poljima nalaze se željeni parametri mreže, a u sivim poljima ispod, odstupanja nakon završetka algoritma. Na Slici 1. dat je grafički prikaz dobijenih rezultata za jedan od testova.

Tabela 1. Dobijeni rezultati

Broj transformatora	Broj signala	Broj SCADA tačaka	Broj potrošača
10000	20000	750	8000
5	-57	0	14
15000	20000	1000	8000
103	-153	-2	-22
5000	15000	750	5000
-7	-37	0	-2
10000	10000	750	5000
1072	-1899	12	550
5000	7500	500	8000
3	-50	3	31
10000	15000	1000	10000
-30	-54	-5	-27
10000	20000	500	5000
-7	-27	-1	8
1000	1000	200	1000
157	-320	-7	247



Slika 1. Rezultati algoritma

8. ZAKLJUČAK

U radu su obrađeni osnovni standardi i specifikacije za modelovanje distributivne mreže, kao i za rad sa podacima iz modela. Pomoću ovih standarda i specifikacija, omogućeno nam je da iz postojećeg modela mreže izvučemo podatke potrebne za generisanje novog modela. Na osnovu tih podataka i obrađenog algoritma za optimalni izbor činilaca modela mreže, stvara se novi model mreže, takav da što bolje zadovoljava prethodno zadate parametre.

Na osnovu dobijenih rezultata mogu se izvući neki zaključci:

- Za isti broj klastera, rešenje je uvek bolje ili jednako ukoliko se zada veći broj iteracija

- Rešenje je gotovo uvek bolje ukoliko je zadat veći broj klastera, odnosno ukoliko je veći izbor objekata prilikom izvršavanja algoritma
- Algoritam je dosta robustan i daje prihvatljive rezultate (do 32% odstupanja) čak i za nerealne vrednosti zadatih parametara distributivne mreže
- Za prihvatljive specifikacije modela distributivne mreže, odstupanje po prametrima ne prelazi 1%

Povećavanjem kako broja klastera, tako i broja iteracija, povećava se i vreme izvršavanja algoritma, o čemu treba voditi računa prilikom biranja ovih inicijalnih ulaznih parametara.

Ono što je mana ovog algoritma jeste mala šansa za određivanjed globalnog optimuma, i na tom polju bi trebalo raditi, kako bi algoritam postao tačniji i upotrebljiviji.

9. LITERATURA

- [1] **Cugnet, Pierre.** 1997. "Confidence Interval Estimation for Distribution Systems Power Consumption by using the Bootstrap Method" - Master rad. Virginia Polytechnic Institute and State University
- [2] The Common Information Model for Distribution - An Introduction to the CIM for Integrating Distribution Applications and Systems, 1016058. Electric Power Research Institute – EPRI. 2008.
- [3] **McMoran, Alan.** 2007. An Introduction to IEC 61970-301 & 61968-11: The Common Information Model. University of Strathclyde Glasgow UK
- [4] Draft IEC 61970: Energy Management System Application Program Interface (EMS-API) – Part 403: Generic Data Access, Draft revision 7. International Electrotechnical Commission – IEC. 2005.
- [5] Utility Management System (UMS) Data Access Facility Specification, Version 2.0.1. Object Management Group – OMG. 2005.
- [6] **McCaffrey, James.** 2014. Machine Learning Using C# Succinctly. Syncfusion Inc.

Kratka biografija:



Lazar Prodanović rođen je 1994. godine u Novom Sadu gde i dalje živi. Školske 2012/2013 upisuje Fakultet tehničkih nauka, smer Računarstvo i automatika. Diplomski rad brani 2016. godine i iste godine upisuje master akademske studije na smeru Automatika i upravljanje sistemima.

PROGRAMSKI JEZIK R

R PROGRAMMING LANGUAGE

Mihailo Vasiljević, *Fakultet tehničkih nauka, Novi Sad*

Oblast – PRIMENJENO SOFTVERSKO INŽENJERSTVO

Kratak sadržaj – Ovaj rad opisuje osnovne elemente programskog jezika R kao i primer njihove upotrebe.

Abstract – This paper describes fundamentals of R programming language and their applications.

Ključne reči: R, programiranje, jezik, statistika

1. UVOD

R je interpretirani programski jezik i okruženje otvorenog koda za statističku obradu podataka, njihovu analizu i vizualizaciju. Sadrži sve elemente karakteristične za jedan programski jezik poput promenljivih, objekata, grananja, petlji i korisnički definisanih funkcija. Ovo je programski jezik opšte namene i iako ga je moguće koristiti za pravljenje softvera, on je svoju primenu pronašao pre svega u analizi i vizualizaciji podataka.

2. OSNOVNI ELEMENTI

U ovom poglavlju biće opisane najvažnije strukture podataka programskog jezika R kao i osnovni tipovi podataka. Strukture podataka u programskom jeziku R se razvrstavaju po dimenzionalnosti (1 dimenzija, 2 dimenzije, ... n dimenzija) kao i po tome da li su homogone ili nisu (da li mogu da sadrže samo elemente istog tipa).

Tabela 1. *Strukture podataka*

	Homogene	Heterogene
1-dimenzionalne	Atomični vektori	Liste
2-dimenzionalne	Matrice	Okviri podataka
n -dimenzionalne	Nizovi	

Skoro svi kompleksniji objekti su izgrađeni na osnovu prikazanih struktura podataka.

2.1. Vektori

Vektori se mogu smatrati nizom susednih lokacija u kojima se nalaze neki podaci, a u programskom jeziku R svakoj od tih lokacija se pristupa putem indeksa. Glavna struktura podataka u programskom jeziku R je vektor. Vektori postoje u dva oblika: atomični vektori i liste. Oni imaju tri zajednička svojstva:

- Tip, `typeof()` – šta je to,
- Dužina, `length()` – koliko elemenata sadrži,
- Atributi, `attributes()` – dodatni proizvoljni metapodaci.

Kao što Tabela 1 prikazuje, razlika između ove dve vrste vektora je u tome što svi elementi atomičnog vektora moraju biti istog tipa dok to nije slučaj sa listom.

2.2. Tip promenljive

Pošto je R dinamički tipiziran jezik prilikom deklarisanja promenljive nije potrebno navesti njen tip već će on biti zaključen na osnovu R objekta koji je dodeljen toj promenljivoj. Takođe, možemo da menjamo tip promenljive iznova i iznova prilikom upotrebe programa.

2.3. Atomični vektori

Za razliku od jezika kao što su C i Java, promenljive u R-u se ne deklariraju kao određeni tip podataka, odnosno R ne omogućava direktan pristup memoriji računara već pruža niz specijalizovanih struktura podataka koje se označavaju kao objekti.

Ovi objekti se referenciraju kao simboli ili promenljive.

Ovi objekti mogu da imaju svoje klase koji liče na osnovne tipove koje je moguće videti u ostalim programskim jezicima.

Tabela 2. *Atomični vektori*

Tip	Primer
Celobrojni tip - <i>Integer</i>	<code>2L, 3L</code>
Decimalni tip - <i>Numeric</i>	<code>1.6, 189, 34500</code>
Logički tip - <i>Logical</i>	<code>TRUE, FALSE</code>
String - <i>Character</i>	<code>'A', 'B', "Big Data"</code>
Kompleksni tip - <i>Complex</i>	
Sirovi tip - <i>Raw</i>	

Kao što je moguće primetiti ovi atomični vektori podsećaju na osnovne tipove podataka u jezicima kao što su C ili Java ali treba voditi računa da su to vektori, pa tako broj 5.8 ili string *"big data"* predstavljaju vektore dužine jedan; nema više osnovnih tipova.

Za kreiranje atomičnih vektora, dužine veće od jedan, može se koristiti funkcija `c()`, naziv funkcije je prvo slovo engleske reči *combine*.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Aleksandar Kupusinać, vanr.prof.

Tabela 3. Primer kreiranja atomičnih vektora pomoću funkcije `c()`

Primer	Rezultat izvršavanja
<code>int_vector <- c(1L, 100L, 1000L)</code> <code>print(int_vector)</code>	<code>[1] 1 100 1000</code>
<code>str_vector <- c('big', 'data', "in R.")</code> <code>print(str_vector)</code>	<code>[1] "big" "data" "in R."</code>
<code>flat_atom_vector <- c(1.5, c(1.5, 6.8))</code> <code>print(flat_atom_vector)</code>	<code>[1] 1.5 1.5 6.8</code>

U posljednjem primeru se može videti da su atomični vektori uvek jednodimenzionalni iako je prilikom njihovog pravljenja moguće ugnježdavanje `c()` funkcija.

2.4. Liste

Liste ili generički vektori, su još jedna struktura podataka u programskom jeziku R. Lista je heterogena struktura podataka, što znači da njen element može da bude bilo koji R- objekat, poput vektora, funkcija pa čak i drugih listi. Elementima liste se pristupa pomoću tri različite operacije indeksiranja. Liste se prave pomoću funkcije `list()`.

Tabela 4. Primer upotrebe funkcije `list()`

Primer	Rezultat izvršavanja
<code>list1 <- list("big data", 25L, 1.5, c(3L, 5L))</code> <code>print(list1)</code>	<code>[[1]]</code> <code>[1] "big data"</code> <code>[[2]]</code> <code>[1] 25</code> <code>[[3]]</code> <code>[1] 1.5</code> <code>[[4]]</code> <code>[1] 3 5</code>

Moguće je napraviti vektor pomoću funkcije `c()`, čiji su elementi liste. U slučaju da je jedan od elemenata lista, a ostali elementi atomični vektori, izvršiće se implicitna konverzija atomičnih vektora u listu i napraviti vektor listi.

Korisne funkcije za rad sa listama:

- `typeof()` - vraća tip, što je u slučaju liste lista – list,
- `is.list()` - da li je objekat lista,
- `as.list()` - eksplicitna konverzija u listu,
- `unlist()` - pretvaranje liste u atomični vektor.

2.5. Nizovi i matrice

Nizovi i matrice su vektori sa atributom `dim` i opcionim atributom `dimnames`. Atribut `dim` omogućava atomičnom vektoru da se ponaša kao višedimenzionalni niz.

Specijalan slučaj niza je matrica koja predstavlja dvodimenzionalni niz. Nizovi i matrice se kreiraju funkcijama `array()` i `matrix()` ili pomoću funkcije `dim()`.

Tabela 5. Primer upotrebe funkcija `array()` i `matrix()`

Primer	Rezultat izvršavanja
<code>array1 <- array(1:6, dim = c(1,6,1))</code> <code>print(array1)</code>	<code>[,1] [,2]</code> <code>[,3] [,4] [,5]</code> <code>[,6]</code> <code>[1,] 1 2</code> <code>3 4 5 6</code>
<code>matrix1 <- matrix(c('x', 'y', 'y', 'z', 'x', 'z'), nrow=2, ncol=3)</code> <code>print(matrix1)</code>	<code>[,1] [,2]</code> <code>[,3]</code> <code>[1,] "x" "z" "y"</code> <code>[2,] "y" "x" "z"</code>

Za utvrđivanje broja vrsta i kolona matrice, odnosno dimenzije niza, mogu se koristiti funkcije `nrow()`, `ncol()` i `dim()`, dok se za utvrđivanje imena vrsta, odnosno kolona, kao i za njihovo zadavanje mogu koristiti funkcije `rownames()` i `colnames()`, odnosno kod nizova `dimnames()`.

Tabela 5. Primer upotrebe funkcija za rad sa nizovima i matricama

Primer	Rezultat izvršavanja
<code>array1 <- array(1:9, dim = c(1,9,1))</code> <code>matrix1 <- matrix(c('x', 'y', 'y', 'z', 'x', 'z'), nrow=2, ncol=3)</code> <code>nrow(matrix1)</code> <code>ncol(matrix1)</code> <code>dim(array1)</code>	<code>[1] 2</code> <code>[1] 3</code> <code>[1] 1 9 1</code>
<code>matrix1 <- matrix(c('x', 'y', 'y', 'z', 'x', 'z'), nrow=2, ncol=3)</code> <code>rownames(matrix1) <- c("big", "data")</code> <code>colnames(matrix1) <- c("r", "big", "data")</code> <code>print(matrix1)</code>	<code>big r big data</code> <code>big "x" "z" "y"</code> <code>data "y" "x" "z"</code>

2.6. Okviri podataka (Dana frames)

Okviri podataka su R struktura koja deli osobine liste i matrica. U suštini, to je lista vektora, faktora, i/ ili matrica iste dužine. Za razliku od matrica, okviri podataka se sastoje od kolona koje mogu da imaju različite vrste podataka. Okviri podataka, dakle, imaju sledeće funkcije:

`names()`, `colnames()`, `rownames()` pri čemu prve dve vraćaju isti rezultat, kao i `length()`, `ncol()`, `nrow()` pri čemu prve dve vraćaju isti rezultat. Za kreiranje okvira podataka koristi se funkcija `data.frame()`.

Tabela 7. Primer kreiranja okvira podataka

Primer	Rezultat izvršavanja
<pre>dataFrame <- data.frame(a = 1:2, b=c("x", "y")) print(dataFrame)</pre>	<pre> a b 1 1 x 2 2 y</pre>

Da bi se ustanovilo da li je neka promenljiva okvir podataka može se koristiti funkcija `class()` ili funkcija `is.data.frame()`, funkcija `typeof()` vraća tip vektora upotrebljen za pravljenje okvira podataka, tj. listu.

Ukoliko postoji objekat koji je potrebno konvertovati u okvir podataka, eksplicitna konverzija se radi pomoću funkcije `as.data.frame()` pri čemu:

- od vektora nastaje okvir podataka sa jednom kolonom,
- od liste nastaje okvir podataka sa po jednom kolonom za svaki element, pri čemu će korisnik biti obavešten o grešci ukoliko elementi nisu iste dužine,
- od matrice će nastati okvir podataka sa istim brojem vrsta i kolona.

Moguće je kombinovati okvire podataka i tada se koriste funkcije `cbind()` i `rbind()`. Kada se vrši kombinacija po kolonama, broj redova mora da se poklapa, ali imena redova se ignorišu. Kada se vrši kombinacija po redovima i broj i imena kolona moraju da se poklapaju.

Korisnici su u zabludi da mogu da naprave okvir podataka koristeći funkciju `cbind()` nad vektorima. Ovo će raditi samo u slučaju da je jedan od argumenata funkcije već okvir podataka, u suprotnom nastaje matrica.

Pošto je okvir podataka, u suštini, lista vektora, moguće je da okvir podataka ima kolonu koja je lista. Kada se lista prosledi kao parametar funkciji `data.frame()`, ova funkcija će pokušati da stavi svaki element liste u zasebnu kolonu i to neće uspeti. Rešenje je da se koristi funkcija `I()` kojom se kaže funkciji `data.frame()` da tretira listu kao jednu jedinicu posmatranja.

2.7. Faktori

Faktori se koriste da opišu jedinice koje mogu imati konačan broj unapred definisanih vrednosti (pol, društveni status, itd.) i koriste se za čuvanje kategoričkih podataka.

Tu do izražaja dolaze atributi, čija jedna važna upotreba jeste kreiranje faktora. Faktori imaju `levels` atribut koji predstavlja skup dozvoljenih vrednosti i klasu `"factor"` koja im omogućava da se ponašaju drugačije od običnih celobrojnih vektora na osnovu kojih su izgrađeni.

Tabela 8. Primeri faktora

Primer	Rezultat izvršavanja
<pre>a <- factor(c("x", "y", "z", "z")) class(a)</pre>	<pre>[1] x y z z Levels: x y z</pre> <pre>[1] "factor"</pre>
<pre>levels(a)</pre>	<pre>[1] "x" "y" "z"</pre>
<pre>a[2] <- k #Ne možete da koristite vrednosti koje nisu u levels</pre>	<pre>#> Warning: invalid factor level, NA generated</pre>

Faktori su izuzetno korisni kada znamo koje su potencijalne vrednosti koje neka promenljiva može da uzme, čak iako ne vidimo sve te vrednosti u okviru nekog skupa podataka koji imamo. Upotrebom faktora umesto vektora karaktera možemo jasnije da predstavimo kada neka grupa nema observacija.

Faktori se često koriste za šifrovanje kategoričkih podataka, odnosno transformisanje kategoričkih promenljivih u numeričke kategorije kako bi se mogli vršiti dalji proračuni. Na primer: Data je promenljiva `dataset` i u njoj kolona `Country` u kojoj se ponavljaju sledeće zemlje: Srbija, Engleska, Nemačka. Transformaciju ovih kategoričkih promenljivih u numeričke je moguće uraditi na sledeći način:

```
dataset$Country <- factor(dataset$Country,
                           levels=c("Srbija","Engleska","Nemacka"),
                           labels = c(1, 2, 3))
```

Pomoću `labels` je rečeno koje numeričke vrednosti se koriste u zameni.

2.8. Atributi

Svi objekti, osim `NULL`, mogu da imaju jedan ili više povezanih atributa. Atributi se koriste za opisivanje metapodataka o objektima. Atributi se čuvaju kao parovi ime-vrednost. Lista atributa se može preuzeti preko promenljive `attributes`, a postavlja se preko `attributes<-`. Individualnim atributima se pristupa preko `attr`, a postavljaju se preko `attr<-`.

Tabela 9: Primeri atributa

Primer	Rezultat izvršavanja
<pre>a <- 1:100 attr(a, "description") <- "Vector range from 1 to 100" attr(a, "description")</pre>	<pre>[1] "Vector range from 1 to 100"</pre>

Bitno je voditi računa o tome da će modifikacija vektora dovesti do gubljenja skoro svih atributa osim tri najvažnija:

1. ime - vektor karaktera koji svakom elementu daje ime,
2. dimenzije - koristi se da bi se vektor pretvorio u matrice i nizove,
3. klasa - vezana za objektni deo programskog jezika R

Atribut *names*, kada postoji, označava individualni element vektora ili liste. Kada se neki objekat ispiše komandom `print` na primer, ako ovaj atribut postoji, on će biti upotrebljen da se označe elementi. Ovaj atribut može da se iskoristi i u svrhe indeksiranja, na primer: `quantile(x)["25%"]`.

Imena ne moraju da budu jedinstvena, ali pravljenje podskupa karaktera je najvažniji razlog za upotrebu imena, a najkorisnije je kada su imena jedinstvena. Ne moraju svi elementi vektora da imaju ime. Ako neko ime nedostaje, funkcija *names()* će vratiti prazan string za ime tog elementa. Ako nedostaju sva imena, funkcija će vratiti *NULL*. Moguće je kreirati i vektor bez imena upotrebom funkcije *unname(x)* ili se imena mogu naknadno ukloniti pomoću *names(x) <- NULL*.

Atribut *dim* se koristi da se implementira niz. Sadržaj niza se čuva u okviru vektora u glavnom redosledu kolona a *dim* atribut je vektor celih brojeva koji određuje odgovarajuća proširenja niza. Vektor nije isti kao jednodimenzionalni niz jer on ima *dim* atribut dužine jedan, dok prvi nema atribut *dim*.

2.9. Nedostajuće vrednosti i konverzija tipova

U statističkom smislu, nedostajuće vrednosti su promenljive čije vrednosti nisu poznate.

Vrednosti koje nedostaju označavaju se logičkim vektorom *NA* čija je dužina jedan. *NA* će uvek biti odgovarajućeg tipa ukoliko se koristi funkcija *c()* zato što se tada prave atomični vektori čiji elementi moraju biti istog tipa, pa će biti izvršena implicitna konverzija tipova. U suprotnom moguće je eksplicitno navesti tip nedostajuće vrednosti, pa tako imamo *NA_real_* (decimalni vektor), *NA_integer_* i *NA_character_*.

Za proveru tipa datog vektora može se koristiti funkcija *typeof()*, a da se proverí da li je posmatrani vektor nekog tipa može se koristiti odgovarajuća *"is"* funkcija, na primer, *is.integer()*.

Ukoliko dođe do kombinovanja elemenata različitih tipova, najčešće automatski, svi elementi jednog vektora će biti konvertovani u isti tip. To je implicitna konverzija tipova. Na primer, kombinacijom elementa čiji je tip *string* i elementa čiji je tip *celobrojni* tip, dobijamo vektor čiji su svi elementi tipa *stringa* ili kombinacijom elemenata logičkog tipa i elementa decimalnog ili celobrojnog tipa dobijamo vektor čiji su svi elementi decimalnog ili celobrojnog tipa. U slučaju da prilikom kombinovanja može doći do gubitka informacija interpreter će ispisati obaveštenje.

Pored implicitne konverzije, postoji i eksplicitna konverzija pomoću funkcija: *as.integer()*, *as.double()*, *as.logical()* i *as.character()*.

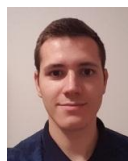
3. ZAKLJUČAK

Programski jezik R, iako opšte namene, danas praktično predstavlja standard za analizu podataka i statističke proračune. Iako ima svojih mana, prednosti programskog jezika R u navedenim oblastima ih nadomeštaju i prevazilaze. Velika količina ugrađenih struktura podataka i kompleksnih objekata, među kojima se pored onih koje je moguće videti u različitim programskim jezicima, ističu okviri podataka i faktori, kao i velika količina postojećih paketa sa implementiranim najvažnijim algoritimima iz različitih oblasti kao što su statistika, mašinsko učenje, analiza i obrada podataka, čine ovaj jezik posebno pogodnim za primenu u ovim oblastima.

4. LITERATURA

- [1] Zvanična veb stranica R projekta, <https://www.r-project.org/about.html>.
- [2] Zvanična dokumentacija R programskog jezika - <https://cran.r-project.org/doc/manuals/r-release/R-lang.html>.
- [3] Hadley Wickham, "Advanced R", Chapman & Hall, 2015.
- [4] Kun Ren, "Learning R Programming", Packt, 2016.

Kratka biografija:



Mihailo Vasiljević rođen je u Novom Sadu 1993. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Računarstvo i automatika odbranio je 20016.god.

ODREĐIVANJE UDALJENOSTI IZMEĐU ČVOROVA BEŽIČNE SENZORSKE MREŽE NA BAZI INDIKATORA SNAGE PRIMLJENOG SIGNALA (RSSI)

RSSI BASED DISTANCE MEASUREMENT BETWEEN NODES IN WIRELESS SENSOR NETWORK

Mirko Ivanić, Fakultet tehničkih nauka, Novi Sad

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Mogućnost lokalizacije čvorova kod aplikacija zasnovanih na bežičnim senzorskim mrežama (eng. WSN) sve je poželjnija. Metoda lokalizacije na osnovu snage prijemnog signala (eng. RSSI) predstavlja jedan od pristupa koji mogu da zadovolje te potrebe. Ovaj rad predstavlja empirijski eksperiment u kome su preduzeta merenja zavisnosti snage prijemnog signala od udaljenosti i orijentacije u odnosu na izvor signala. Na osnovu dobijenih rezultata merenja izveden je model zavisnosti RSSI od udaljenosti. Za eksperiment su korišćeni IRIS MOTE bežični senzorski čvorovi.

Ključne reči: WSN, RSSI, IRIS MOTE,

Abstract – The possibility of localization of nodes in applications based on wireless sensor network (WSN) is desirable. Localization methods based on the dependence of received signal strength (RSSI) from distance is one approach which can meet localization requirements. This paper represent an experiment in which are performed empirical measurements of dependence of RSSI from distance and orientation. RSSI dependency from distance is modeled on measurement results. IRIS MOTE wireless sensor nodes are used for this experiment

Key words: WSN, RSSI, IRIS MOTE

1. UVOD

U aplikacijama zasnovanim na bežičnim senzorskim mrežama sve više se javlja potreba za poznavanjem pozicije bežičnih senzorskih čvorova unutar određenog prostora.

Pritom sama lokalizacija ne sme da utiče na osnovne koncepte zahvaljujući kojima su bežične senzorske mreže postale toliko rasprostranjene: mala cena konstruisanja, mala potrošnja energije.

Imajući to u vidu, korišćenje naprednijih tehnologija i daleko preciznijih metoda lokalizacije, kao što je GPS, gubi smisao naročito ako postoji potreba za lokalizacijom čvorova u zatvorenom prostoru kao što su industrijske hale i slično.

Stoga, daleko jeftinija varijanta lokalizacije može da bude zasnovana na poznavanju zavisnosti snage prijemnog signala (eng. RSSI) od distance [1].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Ivan Mezei, docent.

Ovaj rad predstavlja jedan empirijski eksperiment u kome su preduzeta merenja zavisnosti RSSI od distance i orijentacije u odnosu na izvor signala, kako bi se dobio model na osnovu kojeg bi se odredila udaljenost između čvorova bežične senzorske mreže.

Za merenja su korišćeni IRIS MOTE bežični senzori.

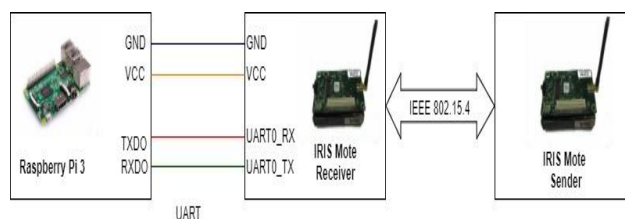
2. HARDVERSKI SISTEM

Da bismo izvršili merenja RSSI vrednosti potrebno je imati na raspolaganju minimalno dva bežična senzorska uređaja koji će na određeni način formirati bežičnu senzorsku mrežu i uspostaviti komunikacione linkove. Za te potrebe korišćeni su IRIS 2.4GHz bežični senzorski uređaji. IRIS Mote bežični senzorski modul predstavlja proizvod kompanije Crossbow Technology namenjen za izgradnju bežičnih senzorskih mreža koje odlikuju mala potrošnja energije, jednostavnost i niska cena ugradnje [3].

Antena koju koristi IRIS modul je $\frac{1}{4}$ talasna dipol antena čija je karakteristika zračenja prilično sferičana, ali zračenje nije ujednačeno u svim pravcima [4].

Pored uređaja koji direktno učestvuju u bežičnoj senzorskoj mreži potreban nam je i uređaj koji će nam omogućiti prikaz izmerenih RSSI vrednosti.

Za tu potrebu koristimo Raspberi Pi 3 razvojni sistem. Na slici 1 je prikazan hardverski sistem korišćen pri merenju.



Slika 1. Hardverski sistem korišćen pri merenju

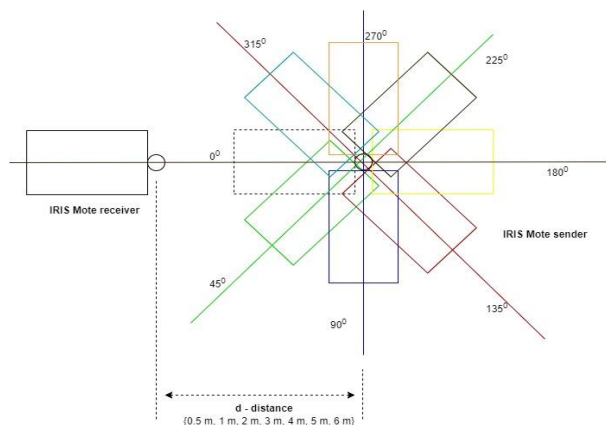
3. MERENJE RSSI VREDNOSTI

Realizovana merenja mogu se podeliti na:

1. Merenja u zatvorenom prostoru
2. Merenja u otvorenom prostoru

Na slici 2 je predstavljen vizuelni prikaz položaja IRIS modula prilikom merenja. IRIS MOTE prijemnik zauzima fiksni položaj u toku merenja, dok se položaj IRIS MOTE predajnika menja.

U ovom eksperimentu za svaki položaj IRIS Mote predajnika izmereno je po 10 RSSI vrednosti.



Slika 2. Vizuelni prikaz orijentacije IRIS modula za svaku distance d

4. REZULTATI MERENJA I ANALIZA

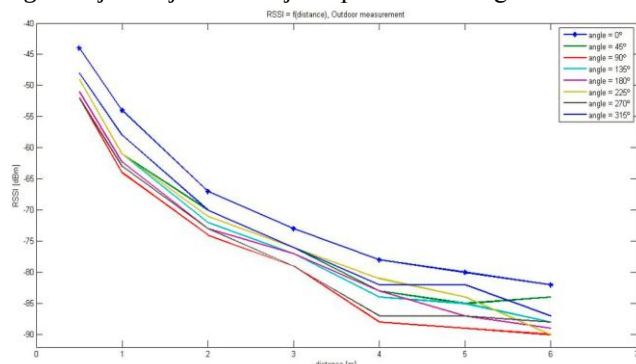
4.1. Rezultati merenja u otvorenom prostoru

U tabeli 1 prikazane su srednje vrednosti RSSI-a [dBm] dobijene na osnovu merenja u otvorenom prostoru.

Tabela 1. Srednje vrednosti RSSI-a [dBm] za sve distance i uglove orijentacije (otvoreni prostor)

	0°	45°	90°	135°	180°	225°	270°	315°
0.5 m	-44	-49	-52	-49	-51	-49	-52	-48
1 m	-54	-61	-64	-61	-62.3	-61	-63	-58
2 m	-67	-70	-74	-72	-73	-71	-73	-70
3 m	-73	-76	-79	-77	-77	-76	-79	-76
4 m	-78	-83	-88	-84	-83	-81	-87	-82
5 m	-80	-85	-89	-85	-87	-84	-87	-82
6 m	-82	-84	-90	-88	-89	-90	-88	-87

Na grafikonu 1 uočavamo da snaga primljenog signala slabi kako se udaljenost od predajnika povećava, tj. obrnuto je proporcionalna kvadratu udaljenosti od predajnika. Najveće izmerene RSSI vrednosti dobijaju za ugao orijentacije 0° i linije su prilično sličnog oblika.



Grafikon 1. Objedinjeni prikaz svih linija koje prikazuju zavisnost RSSI od distance (otvoreni prostor) za različite uglove orijentacije

4.1. Rezultati merenja u zatvorenom prostoru

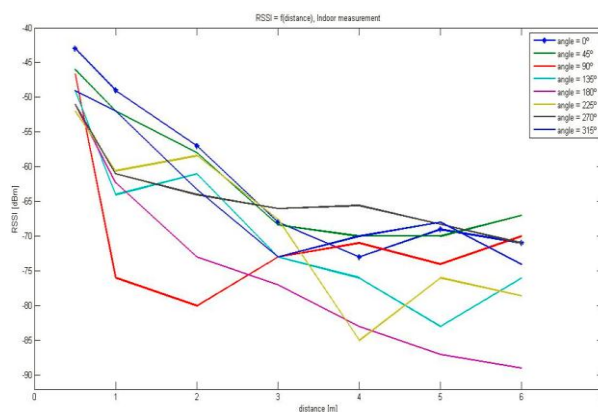
U tabeli 2 prikazane su srednje vrednosti RSSI-a [dBm], dobijene na osnovu merenja u zatvorenom prostoru.

Tabela 2. Srednje vrednosti RSSI-a [dBm] za sve distance i uglove orijentacije (zatvoreni prostor)

	0°	45°	90°	135°	180°	225°	270°	315°
0.5 m	-43	-46	46.6	-49	-52	-52	-51	-49
1 m	-49	-52	-76	-64	-60.6	-61	-61	-52
2 m	-57	-58	-80	-61	-58.4	-58	-64	-63.3
3 m	-68	-68.4	-73	-73	-67.6	-67	-66	-73
4 m	-73	-70	-71	-76	-85	-72	-65.6	-70
5 m	-69	-70	-74	-83	-76	-69	-68.3	-68
6 m	-71	-67	-70	-76	-78.6	-82	-71	-74

Vrednosti RSSI-a dobijene merenjem unutar nekog zatvorenog prostora izrazito su zavisne od faktora kao što su interferencija, prelamanje i rasipanje signala. Signal unutar zatvorenih prostora često nailazi na mnoge prepreke sačinjene od različitih materijala što u krajnjem slučaju ima uticaj na vrednost RSSI-a.

Na grafikonu 2 vidimo nešto što je očekivano za merenja u zatvorenom prostoru.



Grafikon 2. Objedinjeni prikaz svih linija koje prikazuju zavisnost RSSI od distance (zatvoreni prostor) za različite uglove orijentacije

5. MODELOVANJE ZAVISNOSTI RSSI VREDNOSTI OD DISTANCE

Merenjem RSSI vrednosti saznajemo snagu signala na prijemniku, i ako znamo snagu predajnika signala možemo izračunati slabljenje usled propagacije signala. Teoretski i empirijski modeli se koriste kako bi se procenila udaljenost od izvora signala usled slabljenja prilikom propagacije [2]. Ova metoda se koristi uglavnom za RF signale.

Veza između vrednosti snage primljenog signala i distance se može izvesti na dole prikazan način.

$$P \propto 1 / d^n \quad (1)$$

$$RSS \propto 10 \times \log(1 / d)^n \quad (2)$$

$$RSS = -10 \times n \times \log(d) + C \quad (3)$$

Gde zapravo,

d – predstavlja udaljenost od izvora signala.

n – je indeks gubitka gustine snage usled propagacije signala.

C – predstavlja konstantu.

Iz jednačine (3) se može uvideti linearna zavisnost snage prijemnog signala od logaritamski predstavljene distance.

Tako se snaga primljenog signala može predstaviti na mnogo kompaktniji način jednačinom:

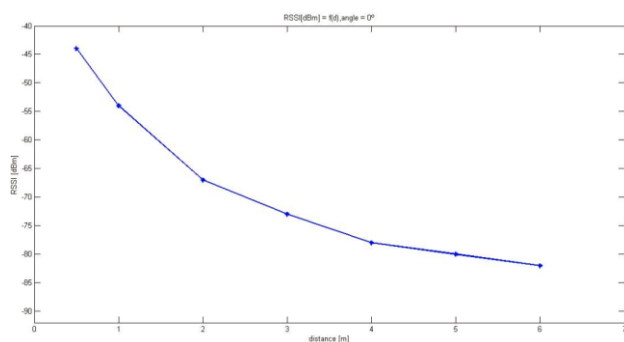
$$RSS = -m \times \log(d) + C \quad (4)$$

Gde m predstavlja nagib u linearnoj jednačini zavisnosti RSS od $\log(D)$. Indeks slabljenja gustine snage usled propagacije može se predstaviti kao:

$$n = m / 10 \quad (5)$$

5.1. Modelovanje za otvoreni prostor

Kao početnu tačku modelovanja zavisnosti RSSI-a od distance uzeli smo merne rezultate za onu orijentaciju IRIS Mote predajnika za koju smo dobijali najveće vrednosti RSSI-a (Grafikon 3).

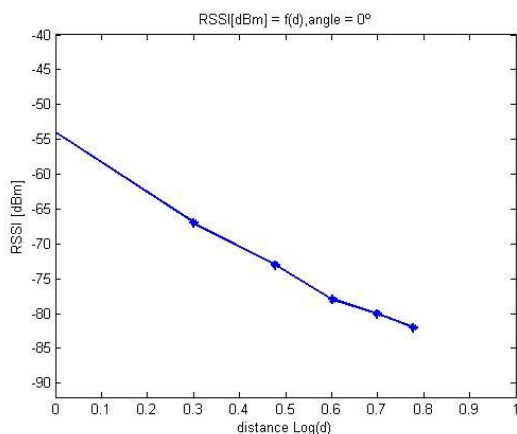


Grafikon 3. Zavisnost RSSI-a od distance za ugao orijentacije 0° (otvoreni prostor)

Pomoću jednačina (1), (2) i (3) dobijamo jednačinu (4):

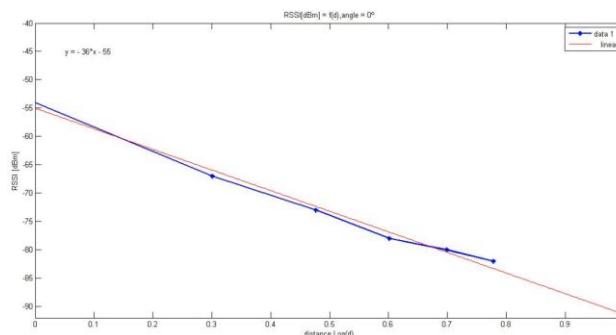
$$RSS = -m \times \log(d) + C$$

Na osnovu jednačine (4) dolazimo do zaključka da je snaga primljenog signala linearno zavisna od logaritamski izražene distance i da je samo potrebno odrediti nagib m i vrednost konstante C . Na grafikonu 4 je prikazana zavisnost RSSI = $f(\log(d))$.



Grafikon 4. Zavisnost RSSI-a od logaritamski izražene distance za ugao orijentacije 0° (otvoreni prostor)

Da bi smo saznali koje su vrednosti nagiba m i konstante C , koristeći MATLAB primenjujemo tehniku linearnog fitovanja zavisnosti RSSI-a od logaritamski izražene distance.



Grafikon 5. Fitovanje zavisnosti RSSI-a od distance izražene u logaritamskom opsegu za ugao orijentacije 0° (otvoreni prostor)

Nakon fitovanja (Grafikon 5) dobili smo sledeće vrednosti za nagib i konstantu:

$$m = 36$$

$$C = -55$$

Tako da sada jednačina (4) sa dobijenim vrednostima modeluje zavisnost RSSI-a od distance u uslovima otvorenog prostora:

$$RSS = -36 \times \log(d) - 55$$

Distance na osnovu snage RSSI može biti određena kao:

$$d = 10^{((RSSI + 55) / -36)}$$

U tabeli 3 uporedno su predstavljene procenjene i stvarne vrednosti distance za izmerene vrednosti RSSI-a.

Tabela 3. Poređenje procenjene distance i stvarne distance

RSSI [dBm]	Stvarna distance [m]	Procenjena distance [m]	Odstupanje
-44	0.5	0.4948	1%
-54	1	0.938	6%
-67	2	2.1544	-8%
-73	3	3.1623	-5%
-78	4	4.3540	-9%
-80	5	4.9482	1%
-82	6	5.6234	6%

Na osnovu dobijenih rezultata zaključujemo da se srednja vrednost odstupanja u proceni distance kreće u granicama od -5% do +5% u odnosu na stvarnu vrednost distance na rastojanju do 6 m od izvora.

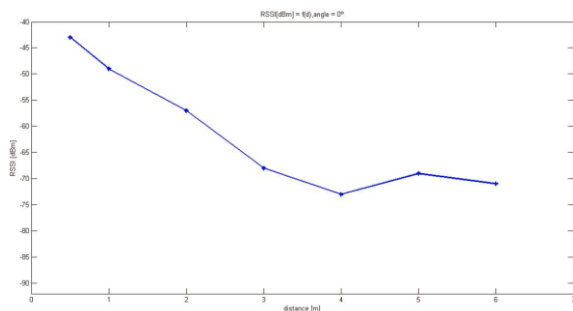
5.2. Modelovanje za zatvoreni prostor

Proračuni koji će biti preduzeti za slučaj zatvorenog prostora su identični kao za slučaj otvorenog prostora, samo očekujemo da će odstupanja biti veća. Kao početnu tačku modelovanja zavisnosti RSSI-a od distance uzeli smo merne rezultate za onu orijentaciju IRIS Mote predajnika za koju smo dobijali najveće vrednosti RSSI-a (Grafikon 6).

Pomoću jednačina (1), (2) i (3) dobijamo jednačinu (4):

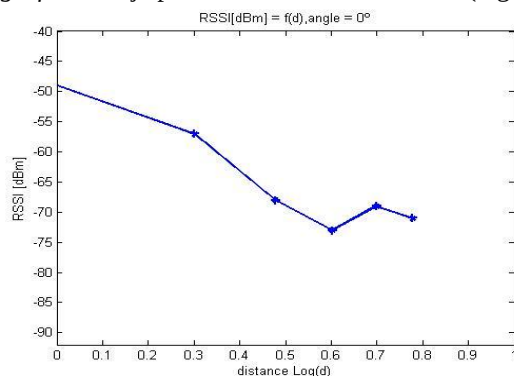
$$RSS = -m \times \log(d) + C$$

Na osnovu jednačine (4) snaga primljenog signala je linearno zavisna od logaritamski izražene distance i da je samo potrebno odrediti nagib m i vrednost konstante C .



Grafikon 6. Zavisnost RSSI-a od distance za ugao orijentacije 0° (zatvoreni prostor)

Na grafikonu 7 je prikazana zavisnost $RSSI = f(\log(d))$.



Grafikon 7. Zavisnost RSSI-a od logaritamski izražene distance za ugao orijentacije 0° (zatvoreni prostor)

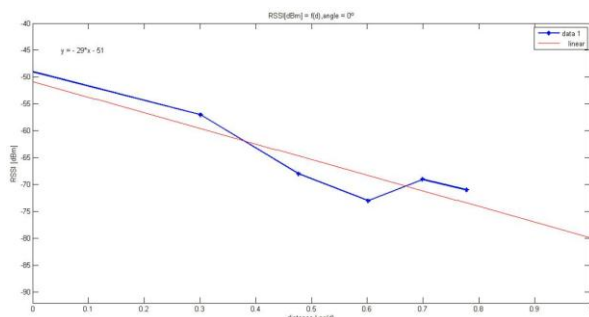
Da bismo saznali koje su vrednosti nagiba m i konstante C , koristeći MATLAB primenićemo tehniku linearnog fitovanja zavisnosti RSSI-a od logaritamski izražene distance. Nakon fitovanja (Grafikon 8) dobili smo sledeće vrednosti za nagib i konstantu:

$$m = 29$$

$$C = -51$$

Samim tim jednačina (4) sa dobijenim vrednostima modeluje zavisnost RSSI-a od distance u uslovima otvorenog prostora:

$$RSS = -29 \times \log(d) - 51$$



Grafikon 8. Fitovanje zavisnosti RSSI-a od distance izražene u logaritamskom opsegu za ugao orijentacije 0° (zatvoren prostor)

Tako da distancu na osnovu snage RSSI može biti određena kao:

$$d = 10^{((RSSI + 51) / -29)}$$

U tabeli 4. uporedno su predstavljene procenjene i stvarne vrednosti distance na osnovu izmerenih RSSI vrednosti.

Tabela 4. Poređenje procenjene distance i stvarne distance

RSSI [dBm]	Stvarna distanca [m]	Procenjena distanca [m]	Odstupanje
-49	0.5	0.5298	-6%
-54	1	0.8532	15%
-57	2	1.6103	19%
-68	3	3.8566	-29%
-73	4	5.7362	-43%
-69	5	4.1753	16%
-71	6	4.8939	18%

Iz tabele najveće odstupanje od stvarne distance je oko 1.73m što je prilično veliko za opseg distanci za koji smo izveli merenja RSSI signala. Srednja vrednost odstupanja izražena u procentima iznosi od -21% do +21%.

6. ZAKLJUČAK

Na osnovu dobijenih rezultata možemo zaključiti da je određivanje udaljenosti bazirano na RSSI vrednosti prilično nepouzđano. Naročito je to izraženo kad se senzorska mreža nalazi unutar nekog zatvorenog prostora, gde na propagaciju signala najviše utiče okruženje koje je neminovno puno prepreka i prisustva drugih izvora RF zračenja. Za slučaj da je bežična senzorska mreža raspodeljena na nekom otvorenom prostoru, bez nekih prepreka propagaciji signala, sa većom pouzdanosću se može odrediti udaljenost između čvorova mreže.

7. LITERATURA

- [1] Praveen Kumar, Lohith Reddy, Shirshu Varma, *Distance Measurement and Error Estimation Scheme for RSSI Based Localization in Wireless Sensor Networks*, Indian Institute of Information Technology, Allahabad, INDIA, 2009.
- [2] Omotayo Adewumi, Karim Djouani, and Anish Kurien, *Performance Evaluation of RSSI Based Distance Measurement for Localization in Wireless Sensor Networks*, K. Jonas, I.A. Rai, and M. Tchunte (Eds.): AFRICOMM 2012, LNICST 119, pp. 74–83, 2013.
- [3] *MPR-MIB Users Manual*, Crossbow Technology, June 2007
- [4] *Wireless Measurement System IRIS*, MEMSIC

Kratka biografija:



Mirko Ivanić rođen je 08.06.1986. u Sanskom Mostu, BiH. Osnovne akademske studije završio na Fakultetu tehničkih nauka 2014. god. Diplomski-master rad iz oblasti Elektrotehnike i računarstva – Embedded sistemi i algoritmi, odbranio je 2017. god.

PRIMENA HOMOMORFNE KRIPTOGRAFIJE U INDUSTRIJSKIM KONTROLNIM SISTEMIMA**APPLICATION OF HOMOMORPHIC CRYPTOGRAPHY IN INDUSTRIAL CONTROL SYSTEMS**

Zorana Babić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – PRIMENJENO SOFTVERSKO INŽENJERSTVO

Kratak sadržaj – U ovom radu su obrađivani homomorfni kriptografski algoritmi i analizirane mogućnosti njihove primene u industrijskim kontrolnim sistemima. Detaljno je prikazan princip rada i klasifikacija homomorfnih algoritama. Dat je pregled četiri odabrana algoritma, po dva iz grupe potpunih i parcijalno homomorfni. Ispitana je mogućnost primene tih algoritma sa ciljem povećanja bezbednosti podataka razmenjenih u sklopu operacija u nadzoru i kontroli procesnih postrojenja.

Ključne reči: homomorfizam, enkripcija, kriptografski sistem

Abstract – This paper contains an overview of homomorphic cryptography and its applicability in industrial control systems. A detailed overview of homomorphic algorithm operation and classification is given. Four homomorphic algorithms are discussed, two from both the full and partial categories. An applicability analysis of these algorithms in protecting data exchanged within industrial control systems was performed.

Key words: homomorphism, encryption, cryptosystem

1. UVOD

Sistem za nadzor i kontrolu industrijskih procesa (eng. *Supervisory Control and Data Acquisition*- SCADA) prikupljaju sa mernih uređaja statuse aktuatora i šalju komande upravljačkoj opremi u procesnom postrojenju. Tradicionalni SCADA sistemi su za ovu komunikaciju koristili protokole koji su imali minimalan broj mera za obezbeđivanje tajnosti i integriteta podataka.

Homomorfna kriptografija za razliku od drugih drugih tipova kriptografskih tehnika omogućava da se nad kodiranim podacima izvršavaju matematičke operacije, bez potrebe za dekodiranjem podataka.

U ovom radu će biti prikazana analiza homomorfni kriptografskih algoritama sa stanovišta mogućnosti njihove primene u SCADA sistemima. Analizirane su najčešće operacije nad podacima npr. provera opsega analognih vrednosti, provera diskretnih vrednosti, računanje proseka.

NAPOMENA:

Ovaj rad proistekao je iz master rada, čiji mentor je bio prof.dr Imre Lendak.

2. HOMOMORFNI KRIPTOGRAFSKI SISTEMI**2.1 Homomorfna kriptografija**

Reč homomorfizam nosi poreklo iz grčkog jezika, sastoji se od dve reči *homos* što znači isti i *morph* što znači forma ili oblik. Ono što ovakav kriptografski sistem čini posebnim jeste to što dozvoljava da se operacije vrše nad kodiranim podacima. To znači da klijent može da kodira podatke i da ih pošalje serveru. Server nad tim kodiranim podacima izvršava određene operacije i rezultat tih operacija vraća klijentu. Klijent potom dekodira podatke i dobija rezultat koji mu je neophodan. Ono što je veoma značajno jeste da server nije imao uvid u (dekodirane) podatke klijenta i samim tim nije razotkrio značenje podataka.

2.2 Istorija homomorfnih algoritama

Koncepte homomorfne kriptografije su 1978. godine predložili Ronalda Rivest, Leonard Adleman i Michael Dertouzos. Godine 1982. Shafi Goldwasser i Silvio Micali predstavljaju rešenje koje dostiže visok nivo sigurnosti, spadao je u grupu aditivnih homomorfnih algoritama, ali mu je mana bila što je mogao da kodira samo jedan bit. Takođe još jedna vrsta aditivne homomorfne enkripcije je predložena od strane Pascal Pailier-a 1999. godine. U ovih 20-ak godina od kada je predložen prvi koncept homomorfne enkripcije veoma je malo napretka bilo u ovoj oblasti. Craig Gentry 2009. godine kao istraživač IBM-a u okviru svoje doktorske teze [1] predlaže prvi kriptografski sistem koji je potpuno homomorfan, ovaj sistem dozvoljava proizvoljan broj aditivnih i multiplikativnih operacija i time računa bilo koji tip funkcija nad kodiranim podacima. Nakon Gentryjevog rada počinje sve veće interesovanje u ovoj oblasti. Ova oblast je počela sve više da se razvija tek u prethodnih nekoliko godina, iako je prvobitna ideja nastala pre nekih 40 godina [2].

2.3 Princip rada

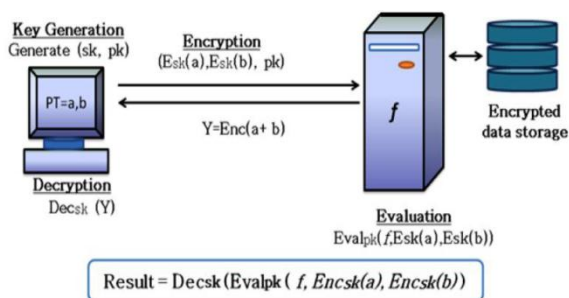
Homomorfna enkripciona šema H predstavlja skup od četiri funkcije (Slika 1)

$H = \{\text{Generisanje ključa, Enkripcija, Dekripcija, Evaluacija}\}$

1) Generisanje ključa (Key Generation): Klijent će generisati par ključeva, privatni ključ sk i javni ključ pk za kriptovanje teksta.

2) Enkripcija (Encryption): Uz pomoć privatnog ključa sk klijent kodira tekst PT i generiše $Esk(PT)$ zajedno sa javnim ključem pk ovaj kodirani tekst biće poslat serveru

- 3) **Evaluacija (Evaluation):** Server ima funkciju f koja vrši evaluaciju kodiranog teksta (tzv šifrata) i izvodi odgovarajuću funkciju uz pomoć javnog ključa pk .
- 4) **Dekripcija (Decryption):** Generisana $Eval(f(PT))$ biće dekodirana od strane klijenta uz pomoć privatnog ključa pk i klijent će dobiti originalni rezultat.



Slika 1. Funkcije homomorfne enkripcije [3]

2.4 Klasifikacija homomornih kriptografskih sistema

Osnovna podela homomornih kriptografskih sistema jeste na parcijalne i potpune. Unutar svake podele se mogu naći kriptosistemi prve i druge generacije.

Parcijalni homomorni algoritmi pokazuju ili aditivni ili multiplikativni homomorfizam, tj. omogućavaju izvršavanje aditivnih ili multiplikativnih operacija nad kodiranim podacima.

Sistem koji podržava proizvoljne proračune nad šifriranim kodiranim podacima jeste potpuna (eng. Fully Homomorphic Encryption - FHE). Potpuno homomorni kriptografski algoritmi imaju mogućnost da vrše i aditivne i multiplikativne operacije.

U parcijalne homomorfne kriptosisteme spadaju RSA bez padding-a, ElGamalov i Goldwasser-Micali kriptosistem, Benaloh i Paillier kriptosistem, predstavnici ranije generacije kriptosistema. Okamoto-Uchiyama, Boneh-God-Nissimi, Naccache-Stern kriptosistemi spadaju u kriptosisteme novije generacije.

U ranije generacije potpuno homomornih sistema spadaju Gentry-jev kriptosistem i Kriptosistem preko celih brojeva, dok u drugu generaciju spadaju Brakerski-Gentry-Vaikuntanathan kriptosistem, Kriptosistem zasnovan na NURU, Gentry-Shai-Waters kriptosistem, pošto je ova tema trenutno veoma popularizovana može se naići na još neke primene. [1]

2.5 Primena homomorfne enkripcije

Vremenom primena homomornih kriptografskih algoritama je sve veća i veća. Jedna od bitnijih primena je u okviru **cloud computing-a**. Korisnici sve češće svoje podatke skladište na cloud, ono što je veoma bitno jeste zaštita ovih podataka. Postoji nekoliko problema kada je u pitanju bezbednost cloud computing-a, to su problemi sa podacima, pitanje privatnosti, inficirane aplikacije, sigurnosni problem i problem kome verovati. Homomorni kriptosistem ima veoma važnu ulogu za razrešavanje ovih pitanja [4].

Još jedna od primena može i da se nađe kod medicinskih aplikacija. Zamisao je da se na cloud servis za vođenje

elektronskih medicinskih kartona šalju podaci o korisniku koji se tiču vitalnih informacija, kao što su pritisak, otkucaji srca, nivo šećera u krvi, itd. Nakon toga server može da izvrši određenu analizu poslatih podataka i na osnovu te statistike može da se prati stanje pacijenta, ili da se propisuju određeni lekovi. Najbolje rešenje za ove podatke upravo jeste cloud jer je broj ovih podataka veliki i bilo bi veoma teško čuvati ih lokalno. Naravno da bi se podaci zaštitili šalju se kriptovani na cloud, gde se onda uz pomoć homomornih algoritama može izvršiti računanje statistike, bez da se podaci izlože nekoj drugoj strani [5].

3. MOGUĆNOSTI PRIMENE HOMOMORFNE KRIPTOGRAFIJE U INDUSTRIJSKIM KONTROLNIM SISTEMIMA

3.1. Prikaz odabranih algoritama

Prva dva razmatrana algoritma su Paillier algoritam i RSA algoritam bez padding-a, koji spadaju u grupu parcijalno homomornih algoritama. Implementacija ova dva algoritma je izvršena samostalno u okviru projekta. Druga dva algoritma su iz grupe potpunohomomornih algoritama i to su Brakerski-Gentry-Vaikuntanathan (BGV) i Gentry-Sahai-Waters (GSW). Zbog složenosti matematičkih operacija koje se koriste u ova dva algoritma korišćene su biblioteke koje sadrže njihov open source implementacije. HELib biblioteka koja predstavlja implementaciju BGV algoritma i TFHE biblioteka koja predstavlja implementaciju GSW kriptosistema, obe ove biblioteke su implementirane u C++ programskom jeziku.

RSA algoritam je danas u širokoj upotrebi, objavljen je 1978 godine i iste te godine od strane autora RSA algoritma objavljen je rad koji predstavlja prvu ideju vezanu za homomorfnu enkripciju. Posebna vrsta RSA je homomorfna i to je RSA bez padding-a, ono što ovaj algoritam čini manje sigurnim je to što kriptovanje nekih poruka sa bilo kojim ključem daje uvek isti rezultat. Ovaj algoritam koristi asimetrične ključeve i spada u algoritme koji mogu da izvedu homomorfne multiplikativne operacije [6].

Paillier-ov algoritam spada u probabilističke algoritme i kao i RSA koristi asimetrične ključeve. Ovaj algoritam ima aditivno svojstvo, tj nad kodiranim podacima se može izvoditi samo operacija sabiranja.

Brakerski-Gentry-Vaikuntanathan (BGV) algoritam spada u algoritme novije generacije potpuno homomornih algoritama. Kada se koristi ovaj algoritam postoji mogućnost biranja šeme po kojoj će raditi. Obe šeme su zasnovane na problemu učenja sa greškama (eng. Learning with Errors - LWE). Prva je zasnovana na običnom učenju, dok se druga zasniva na kružnom učenju. U ovom radu ćemo koristiti šemu koja je zasnovana na problemu kružnog učenja sa greškama [7].

Gentry-Sahai-Waters (GSW) algoritam takođe spada u potpune homomorfne algoritme novije generacije, on se (takođe) zasniva na učenju sa greškama. Ono što ovu šemu čini zanimljivom jeste da su tvorci ove šeme izbegli postojanje posebnog ključa za evaluaciju, tako da korisnik koji ima javne parametre može da izvrši evaluaciju, što nije slučaj kod ostalih šema [8].

4. REZULTATI

4.1 Arhitektura aplikacije

Aplikacija je implementirana u radnom oruženju Visual Studio 2015, u programskom jeziku C#. Klijent-server arhitektura je implementirana uz pomoć Windows Communication Foundation (WCF) radnog okvira koji služi za servisno orijentisanu arhitekturu. Glavna karakteristika ovog radnog okvira je u razmeni poruka. Uz pomoć klijent-server arhitekture biće simulirana komunikacija između uređaja u industrijskom sistemu, koji će predstavljati klijente i SCADA sistema, koji će biti server. SCADA od strane uređaja prima izmerene vrednosti, prima dve vrste ulaza digitalne, odnosno statusne bitove, i analogne ulaze, npr. merenja sa mernih uređaja. Pored SCADA servera i klijenata, tu se nalazi i klijent koji predstavlja sistem za izdavanje naređenja uređajima u procesnom postrojenju i to posredno preko servera. U ovu grupu uređaja spada npr. prekidačka oprema sistema u elektrtoenergetskom sistemu, koja može da ima nekoliko statusa, da je prekidač otvoren ili zatvoren, takođe može i da se desi da je prekidač poluotvoren ili da je status nepoznat usled komunikacione ili druge greške u sistemu. Kada SCADA hoće da izda neku komandu prekidačkom uređaju to mora da se izvrši preko upravljačkog uređaja. Sva komunikacija koja se obavlja između ovih uređaja je kodirana.

4.2. Test scenarija

U okviru aplikacije testirano je nekoliko scenarija, koji bi mogli biti primenljivi u relnom sistemu, i oni su opisani u ovom poglavlju

4.2.1 Test granica vrednosti

Kada je u pitanju SCADA sistem neophodno je da pratimo promenu merenja koja se dešavaju na uređajima. Svaki uređaj, u zavisnosti od toga šta meri (napon, struju, itd.) ima postavljene granice merenja. Očitana merenja moraju biti u određenim granicama. Ukoliko izmerena vrednost izađe van granica, bilo da je to iznad gornje ili ispod donje granice treba da se generiše alarm.

Tok podataka u ovom eksperimentu je sledeći: uređaj šalje svoja merenja na SCADA sistem. SCADA nad pristiglim kodiranim podacima treba da izvrši poređenje sa unapred definisanim granicama, koje su takođe poslate u kodiranom obliku. Rezultati scenarija su prikazani u tabeli 1.

Tabela 1. Prikaz rezultata scenarija za proveravanje granica vrednosti

Ime algoritma	Da li je moguće izvršavanje operacije poređenja?
RSA bez padding-a	NE
Paillier algoritam	NE
BGV algoritam	DA
GSW algoritam	DA

4.2.2. Test proseka analognih vrednosti

SCADA sistem je zadužen za praćenje podataka u sistemu i kao takav ima mogućnost kreiranja raznih analiza i izveštaja. Ovim scenariom simulirano je računanje pro-

seka analognih vrednosti u toku jednog dana, odnosno dnevni prosek. Uređaji u toku jednog dana šalju u određenim vremenskim intervalima analogna očitavanja. SCADA sistem te podatke beleži u svom sistemu. Ovaj scenario je simuliran tako što upravljački uređaj može da zatraži u svakom momentu prosek vrednosti za taj dan. Prosečna vrednost će biće u kodiranom obliku i kao takva može da se dekodira samo od strane uređaja koji poseduje tajni ključ. Rezultati scenarija su prikazani u Tabeli 2.

Tabela 2. Prikaz rezultata scenarija računanja prosečnih vrednosti

Ime algoritma	Da li je moguće računanje prosečnih vrednosti?
RSA bez padding-a	NE
Paillier algoritam	NE
BGV algoritam	DA
GSW algoritam	DA

4.2.3 Test statusnih bitova

Kao što smo prethodno objasnili SCADA ima mogućnost upisa vrednosti statusa opreme u procesnom postrojenju. U ovom scenariju će biti provereno da li su vrednosti koje je upravljački sistem poslao na uređaj iste sa vrednostima koje taj uređaj očitava, odnosno da li je izvršeno dobro podešavanje statusa opreme. Rezultati scenarija su prikazani u tabeli 3.

Tabela 3. Prikaz rezultata scenarija proveravanje bitova status

Ime algoritma	Podržan operator jednakosti?
RSA bez padding-a	DA
Paillier algoritam	DA
BGV algoritam	DA
GSW algoritam	DA

4.3 Diskusija dobijenih rezultata

Prolazeći kroz ove scenarije došli smo do zaključka da algoritmi koji spadaju u grupu parcijalnih homomorfnih algoritama bi mogli da se upotrebe samo kada je u pitanju poređenje vrednosti (tj. operator =). Ova karakteristika nije ključna jer bi poređenje šifrata radilo i sa većinom drugih kriptografskih algoritama bez homomorfnih karakteristika. S toga RSA bez padding-a i Paillier kao takvi trenutno nisu pogodni za primenu u SCADA sistemima. RSA bez padding-a je algoritam koji može da vrši isključivo multiplikaciju podataka dok je Paillier algoritam aditivni algoritam, odnosno može da izvršava sabiranje dva kriptovana podatka.

Za razliku od parcijalnih homomorfnih algoritama potpuni homomorfni algoritmi bi mogli da pronađu upotrebu kod SCADA sistema. S obzirom na svoje karakteristike oni su u mogućnosti da računaju složene matematičke izraze. Istovremeno mogu da obavljaju multiplikaciju i sabiranje pa je zbog toga moguće da vrše komplikovaniye proračune kao što je na primer srednja vrednost.

5. ZAKLJUČAK

U ovom radu je prikazan pregled homomorfne kriptografije, identifikovane su vodeći algoritmi u oblasti i analizirana je mogućnost njihove primene u industrijskim kontrolnim sistemima sa ciljem povećanja nivoa informacione bezbednosti. Jedna od velikih prednosti ovih algoritama je da je strani od nižeg poverenja moguće delegirati rad nad podacima bez poznavanje dekodiranih vrednosti podataka

Izvršenim eksperimentima je ustanovljeno da parcijalni homomorfni algoritmi su ograničeni time da mogu da izvršavaju samo jednu vrstu operacija i time mogu biti zastupljeni u aplikacijama koje koriste ograničen skup matematičkih operacija. Takođe još jedan od nedostataka jeste taj da ovi algoritmi mogu da funkcionisu samo sa veoma malim porukama. Potpuni homomorfni algoritmi mogu da nađu višestruku primenu zbog svojih karakteristika da mogu da izvršavaju različite matematičke operacije nad kodiranim podacima.

Ova oblast je u poslednjoj deceniji postala sve popularnija i kao takva se sve više razvija. Operacije za koje je uspeo da se dobije homomorfizam su operacije dodavanja i operacije množenja, naravno bilo bi od velikog značaja ako bi se operacija deljenja uspešno izvršila nad kodiranim podacima i tako se proširio spektar mogućnosti ovih algoritama. Zamerke za trenutne algoritme se uglavnom tiču performansi, tako da bi se već postojeći kriptografski sistemi mogli proširivati u cilju smanjenja brzine kriptovanja, jer za sisteme koje rade u realnom vremenu brzina je jedan od ključnih faktora.

6. LITERATURA

- [1] Gentry, C.: A fully homomorphic encryption scheme. PhD thesis, Stanford University (2009)
- [2] Moore, C., O'Neill, M., O'Sullivan, E., Doroz, Y., & Sunar, B., (2014, June). Practical homomorphic encryption: A survey. 2014 IEEE International Symposium on Circuits and System (ISCAS), pp. 2792-2795
- [3] Parmar, P., Padhar, S., Patel, S., Bhatt, N., & Jhaveri, R., (2014, April). Survey of Various Homomorphic Encryption algorithms and Schemes, International Journal of Computer Applications (IJCA), pp. 26-32
- [4] Winkler, V., (2011) Securing the Cloud, Cloud Computer Security
- [5] Bosch, C., Andreas, P., Hartel, P., & Jonker W., (2014, February) SOFIR: securely outsourced forensic image recognition. IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP. Pp 2694-2698
- [6] Rivest, R., Adelman, L., & Dertouzos, M., (1987, October) On Data Banks And Privacy Homomorphisms. Journal Foundations of secure computation, pp 169-180
- [7] Brakerski, Z., Gentry, C., & Vaikuntanathan, V., (2012, January) Fully Homomorphic Encryption without Bootstrapping, Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, pp 309-325
- [8] Gentry, C., Sahai, A., Waters, B., (2013). Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based, Advances in Cryptology – CRYPTO 2013, pp 75-92

Kratka biografija:



Zorana Babić rođena je u Novom Sadu 1993. god. Osnovne studije na Fakultetu tehničkih nauka, smer Računarstvo i automatika završava 2016. godine i stiče zvanje diplomiranog inženjera elektrotehnike.

PRORAČUN KAVEZNOG ASINHRONOG MOTORA CALCULATION OF SQUIRREL CAGE INDUCTION MOTOR

Kristina Vujkov, Veran Vasić, Đura Oros, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA

Kratak sadržaj – U radu je predstavljen analitički model za proračunavanje kaveznih asinhronih motora. Analitički model je omogućio proračun visokonaponskog kaveznog asinhronog motora 530kW, koji je konstruisan i proizveden u kompaniji ATB Sever. U cilju dokazivanja validnosti ovih proračuna, oni su upoređeni sa rezultatima softvera korišćenog prilikom konstruisanja motora u ATB Sever-u i rezultatima izmerenim u ispitnoj stanici.

Abstract – The paper will present analytical model for calculation of squirrel cage induction motors. Analytical model made it possible to calculate high voltage squirrel cage induction motor 530kW, which is designed and manufactured in company ATB Sever. With a goal to verify this calculations, they are compared with results of software which is used during design of this motor in ATB Sever and with results of measured values in testing station.

Ključne reči: asinhroni motor, kavezni, momenat, napon, struja, snaga, Fildov sačinilac, Karterov sačinilac, gubici, stepen korisnog dejstva.

1. UVOD

Električne mašine imaju široku primenu u industriji gde predstavljaju osnovni izvor mehaničkog rada. Takođe su široko rasprostranjene u svim sferama ljudskog delovanja u uređejima koji se upotrebljavaju u svakodnevnom životu.

Oko polovine ukupne električne energije proizvedene na globalnom nivou koristi se u elektromotornim pogonima, a ovaj udeo se povećava sve širom primenom upravljanih motornih pogona. Primenom različitih tehnika upravljanja može se efikasno upravljati momentom i brzinom što može kao rezultat dati znatno efikasniji proces koji se odvija pod dejstvom rada elektromotora. Ovako upravljana električna mašina može ostvariti znatne uštede energije. Pored toga različitim tehnikama upravljanja se mogu smanjiti gubici same mašine, što povećava ukupnu efikasnost sistema. Zbog toga je važno dobro isprojektovati električnu mašinu. Ako su električne mašine loše projektovane ili neefikasno korišćene, mere za povećanje efikasnosti su uzaludne.

U radu su iznete teorijske osnove potrebne za razvijanje analitičkog proračuna kaveznog asinhronog motora. Precizno proračunavanje karakteristika električnog motora pri projektovanju i proizvodnji je neophodno zbog ugovornih obaveza između kupca i proizvođača.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio prof. dr Đura Oros.

Ugovorom se kupcu garantuje minimalan broj karakterističnih veličina: nazivna snaga elektro motora, nazivni napon, nazivna struja, stepen korisnog dejstva, faktor snage, brzina obrtanja, polazni momenat, maksimalni momenat i polazna struja. Odstupanje od navedenih vrednosti u najblažem slučaju dovodi do odbijanja porudžbine, a često može dovesti do dodatnih finansijskih izdataka kompanije koja nije ispunila ugovoreno.

Ovaj rad se zasniva na određivanju svih parametara trofaznog visokonaponskog asinhronog kaveznog motora potrebnih za što tačnije određivanje karakteristike motora. Da bi se pokazala validnost proračuna prikazanih u radu, proračun je urađen za motor koji je konstruisan i koji je kao takav proizveden i ispitan u ispitnoj stanici kompanije ATB Sever. Radi utvrđivanja tačnosti prikazanog proračuna, za poredjenje, korišćen je i proračun dobijen internim Severovim softverom KONPRO koji se u kompaniji koristi još od 80ih godina.

2. PRORAČUN MAGNETNOG KOLA

Magnetno kolo električnih mašina se sastoji od feromagnetnih materijala i vazdušnog procepa. U asinhronim mašinama, svi namotaji učestvuju u magnećenju mašine. Takođe se mora primetiti da za mašinu sa više polova magnetno kolo ima nekoliko magnetnih putanja. Električna mašina ima onoliko magnetnih putanja koliko ima polova. U dvopolnom sistemu, magnetni krug je simetrično podijeljen na dve putanje. Ova simetričnost se koristi da se u dizajnu kompletnog magnetnog polja analiza sprovodi na magnetnom kolu koje pripada jednom polu.

Proračun magnetnog kola zasnovan je na analizi magnetne indukcije B i jačine magnetnog polja H u različitim delovima mašine. Proračun magnetnog kola je određen Amperovim zakonom. Prvo, se određuje odgovarajuća prihvatljiva vrednost magnetne indukcije vazdušnog procepa B_g . Zatim se izračunava vrednost magnetne indukcije u zubima i u jarmu i odgovarajuće vrednosti jačine polja H u različitim delovima mašine. Magnetomotorna sila (mms) kola Θ je jednaka linijskom integralu jačine polja duž zatvorene linije obuhvatne putanje, računa se preko obrasca:

$$\oint H_l dl = \Theta \quad (1)$$

Za proračunavanje magnetnog kola električnih mašina, integral se zamenjuje sa sumom proizvoda pojedinih delova putanje l_x i jačine polja $H_1, H_2, \dots$, gde se delovi putanje l_x biraju tako da je jačina polja H_x u tim delovima konstantna. Sada se jednačina (1) može zapisati kao:

$$H_1 l_1 + H_2 l_2 + \dots + H_n l_n = \Theta \quad (2)$$

Leva strana ove jednačine predstavlja zbir magnetnih napona, koji se mere kao i mms Θ u amperima (A), kada se stavi H u A/cm a i l_n u cm.

Proračun pojedinačnih magnetnih napona sastoji se iz [1]:

- 2.1. Magnetnog napona vazdušnog procepa
- 2.2. Magnetnog napon zuba
- 2.3. Magnetni napon jarma

Umesto stvarne dimenzije u proračunima za širinu vazdušnog procepa δ uvodi se njegov ekvivalent, odnosno efektivna vrednost vazdušnog procepa δ_c , koja je srazmerna veličini k_c koja se naziva Karterov sačinilac.

$$\delta_c = \delta \cdot k_c \quad (3)$$

Sa Karterovim sačinioem uzima se u obzir uticaj žljebova na vazdušni procep kod poluzatvorenih a, naročito kod otvorenih, bilo na statoru ili na rotoru, ili i na samo jednom od njih.

U slučaju ovakvih žljebova, magnetni pad napona u vazdušnom procepu veći je od onoga koji bi se dobio kada bi se uračunala samo stvarna dimenzija magnetnog procepa δ , odnosno njegova fizička vrednost.

Fizička vrednost vazdušnog procepa raste sa Karterovim sačinioem k_c [2].

Karter je pokazao da vrednost sačinioa k_c zavisi od odnosa otvora žljeba i dužine stvarnog vazdušnog procepa i dao potpuni obrazac kao i ovaj uprošćeni pomoću koga se k_c računa sa dovoljnom tačnošću [3]:

$$k_c = \frac{\tau_n}{b} \quad (4)$$

$$\tau_n = \frac{\delta}{5 + \frac{b}{\delta}}$$

3. PRORAČUN OSNOVNIH PARAMETARA ELEKTROMOTORA

Otpornosti i induktivnosti predstavljaju osnovne parametre elektromotora i kao takvi definišu karakteristike električne mašine.

3.1. Proračun otpornosti

Sa stanovišta efikasnosti, električni otpori namotaja su važni parametri električnih mašina.

U mnogim slučajevima, gubici na otpornostima formiraju dominantne gubitke u mašini.

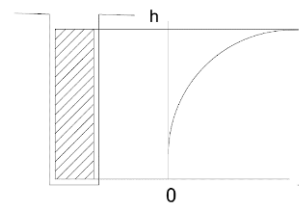
U električnoj mašini, provodnici su obično okruženi feromagnetnim materijalom, koji usmerava fluks da protiče kroz namotaje.

Ovo, opet, može izazvati probleme u vidu velikih potiskivanja struje ukoliko namotaji nisu pravilno projektovani.

Raspodelu gustine struje po visini provodnika za namotaj sa samo jednim slojem prikazuje slika 1.

Uticaj potiskivanja struje na otpornost uvažava se pomoću Fildovog sačinioa.

U donjem delu provodnika pri njegovoj dovoljnoj visini nema ni struje ni polja.

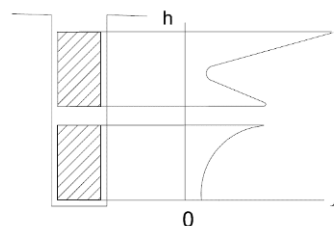


Slika 1. Raspodela gustine struje u jednoslojnom provodniku

h -visina provodnika

J -gustina struje u provodniku

Raspodela gustine struje po visini provodnika prikazana je na slici 2 za slučaj dvoslojnog namotaja. Sa slike se zaključuje je da je raspodela struje u gornjem sloju sasvim drugačija od gustine struje u donjem sloju. Ovo je posledica toga da je za potiskivanje struje u gornjem sloju zaslužno ne samo sopstveno poprečno polje već i poprečno polje u žljebu donjeg sloja.



Slika 2. Raspodela gustine struje u dvoslojnom provodniku

Metoda za proračun strujnog potiskivanja zasniva se na Maksvelovim jednačinama, u daljem izlaganju korišćeni su rezultati računa koje su izvršili razni autori [1].

Razmotra se slučaj prikazan na slici 3, gde se u žleb postavlja $z_t \times z_a$ provodnika. Svi provodnici u žljebu su redno povezani. Visina provodnika je h_0 a širina b_0 . Efektivna (smanjena) visina provodnika izračunava se pomoću jednačine:

$$\xi = \alpha h_0 = h_0 \sqrt{\frac{1}{2} \omega \mu_0 \gamma_{cu} \frac{b_0}{b}} \quad (5)$$

Gde su:

ξ - efektivna visina provodnika

α - korekcionni faktor usled potiskivanja struje

h_0 - visina bakra

b_0 - ukupna širina bakra

γ_{cu} - specifična provodnost bakra

b - širina žljeba

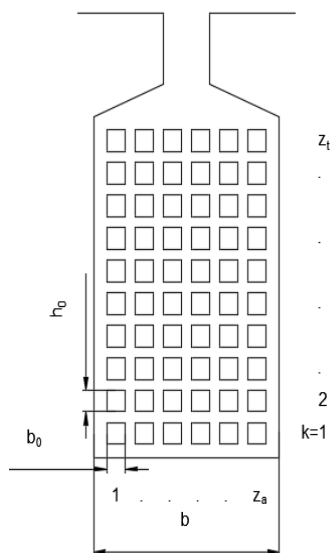
Faktor povećanja otpornosti k-tog sloja usled potiskivanja struje, sa slike 3, odnosno Fildov sačinilac:

$$k_{Rk} = \varphi(\xi) + k(k-1)\psi(\xi) \quad (6)$$

Gde su funkcije:

$$\varphi(\xi) = \xi \frac{\sinh 2\xi + \sin 2\xi}{\cosh 2\xi - \cos 2\xi} \quad (7)$$

$$\psi(\xi) = 2\xi \frac{\sinh \xi - \sin \xi}{\cosh \xi + \cos \xi} \quad (8)$$



Slika 3. Uz pojašnjenje proračuna za Fildov sačinilac

Jednačina (6) pokazuje da je Fildov sačinilac najmanji na donjem sloju, dok je najveći na gornjem sloju. Ovo znači da u slučaju redno povezanih provodnika, provodnici u dnu žljeba manje doprinose gubicima otpornosti od provodnika koji su prostorno pozicionirani na većim visinama u žljebu. Prosečna vrednost Fildovog sačinioca duž žljeba je:

$$k_R = \varphi(\xi) + \frac{z_t^2 - 1}{3} \psi(\xi) \quad (9)$$

Gde je z_t broj provodnika sa slike 3. Za kavezni rotor ovaj broj provodnika jednak jedinici. Sada neka je faktor potiskivanja za žleb označen sa k_{Ru} i računa se prema obrascu (9), dok je faktor potiskivanja za bočne veze označen sa k_{Rw} . Pod pretpostavkom da je temperatura duž provodnika ista, otpori su proporcionalni dužinama provodnika. Označavajući efektivnu dužinu bočne veze kao l_m i ekvivalentna dužina gvožđa kao l , Fildov sačinilac je [3]:

$$k_R = k_{Ru} \frac{l}{l_m} + k_{Rw} \frac{l_m - l}{l_m} \quad (10)$$

Obično je efekat potiskivanja struje na krajnjim namotajima zanemarljiv, stoga $k_{Rw} = 1$ i

$$k_R = 1 + (k_{Ru} - 1) \frac{l}{l_m} \quad (11)$$

3.2. Proračun induktivnih otpornosti

Uspostavljeni magnetni fluks u električnim mašinama ne učestvuje u celosti u elektromagnetnoj konverziji energije. Deo fluksa koji učestvuje u konverziji energije naziva se glavni fluks mašine, a komponente fluksa koje ne učestvuju u ovom procesu su definisani kao rasipni fluks Φ_σ . Kod rotacionih mašina glavni fluks (fluks vazdušnog procepa) Φ_δ mora da prođe kroz vazdušni procep i čini magnetsku spregu između statora i rotora. Rasipni fluksevi se javljaju i u statoru i u rotoru [1], a njihove putanje uglavnom ne prelaze vazdušni procep. Zbog rasipnih flukseva potrebno je više feromagnetnog

materijala, ili srazmerno ovoj pojavi, potrebno je više struje magnećenja.

Proračun rasipnog fluksa je komplikovan zbog uglavnom složene geometrije električne mašine. Međutim, u praksi se upotrebljavaju uprošćene metode koje u suštini ne polažu pravo na tačnost, ali su zato jednostavne i precizne. Njihova praktična upotrebljivost je vremenom potvrđena [4].

Fluks rasipanja se deli na delove shodno geometrijskom prostoru u kojem se prostire. Dakle, linije rasipanja imaju pet različitih putanja na osnovu kojih se proračunavaju različiti magnetski otpori. Zbog toga razlikujemo:

- Rasipanje u vazdušnom procepu,
- Rasipanje u žljebu,
- Rasipanje na glavi zuba i
- Rasipanje na bočnim vezama.

Detalji ovog proračuna prevazilaze okvire ovog rada.

4. GUBICI ASINHRONOG ELEKTROMOTORA

Važna karakteristika motora je stepen korisnog dejstva koji je određen gubicima snage. Pored ukupnih gubitaka motora za konstruktora je veoma važno poznavanje iznosa gubitaka u pojedinim delovima mašine. Na taj način eventualno može konstrukciono da se utiče na iznos pojedinih gubitaka pa time i na stepen korisnog dejstva. U asinhronom motoru gubici se dele na stalne i promenljive.

Stalni gubici su:

- Gubici na trenje i ventilaciju Q_{fv}
- Gubici u gvožđu Q_{fe}

Promenljivi gubici su:

- Gubitak u bakru statora Q_{cu1}
- Gubici u bakru rotora Q_{cu2}
- Dodatni gubici Q_{dod}

Poznavajući parcijalne gubitke, ukupni gubici ΣQ dobijaju se kao zbir parcijalnih gubitaka:

$$\Sigma Q = Q_{fv} + Q_{fe} + Q_{dod} + Q_{cu1} + Q_{cu2} \quad (12)$$

Na ovaj način daje se mogućnost sračunavanja primljene snage dodavanjem gubitaka ΣQ korisnoj snazi:

$$P_1 = P + \Sigma Q \quad (13)$$

Stepen korisnog dejstva dobija se kao odnos korisne i primljene snage:

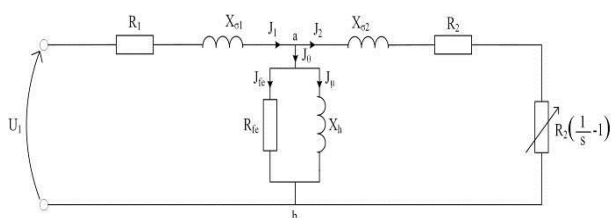
$$\eta = \frac{P}{P_1}, \quad (14)$$

ili koristeći zbir gubitaka:

$$\eta = 1 - \frac{\Sigma Q}{P_1}. \quad (15)$$

5. ODREĐIVANJE KARAKTERISTIKA ELEKTROMOTORA

Uobičajena šematska predstava asinhrone mašine je T ekvivalentna šema prikazana na slici 4. Ovakva ekvivalentna šema predstavlja prikaz matematičkog modela sveden na jednu fazu. Pošto su elementi šeme dati preko parametara mašine, ona može da posluži za određivanje radnih karakteristika.



Slika 4. Ekvivalentna šema asinhronog trofaznog motora

Poreklo ekvivalentne šeme se bazira na teoriji obnog magnetnog polja. Iz ekvivalentne šeme se pored određivanja uticaja motora na mrežu koja ga napaja, mogu odrediti i karakteristike samog motora. Posmatranje se može svesti na posmatranju jedne faze s obzirom da jedinu razliku između faza čini karakteristični vremensko-fazni ugao uravnoteženog višefaznog sistema. Svi naponi, struje i impedanse predstavljaju vrednosti za jednu fazu mašine. U ovom slučaju je u pitanju motor sa spregom namotaja statora u zvezdu. Posledica sprege je da je fazna struja iz ekvivalentne šeme u suštini i linijska struja motora, a napon iz ekvivalentne šeme fazni napon motora. Parametre ekvivalentne šeme konstruktor izračunava u toku proračuna motora. Ukoliko su poznati elementi ekvivalentne šeme moguće je doći do svih izlaznih veličina i karakteristika elektromotora.

6. VERIFIKACIJA PRORAČUNA

Da bi se proverila valjanost postupka proračuna asinhronog motora izvršeno je izračunavanje za visokonaponski motor snage 530kW. Za ovaj isti motor izvršen je proračun softverskim paketom KONPRO. Ovaj softverski paket je vlasništvo ATB Sever i standardno služi za projektovanje motora koji se proizvode u kompaniji. Karakteristične veličine projektovanog motora su prikazane u Tabeli 1. Ove veličine se dobijaju kao rezultat izračunatih parametara i primene ekvivalentne šeme.

Tabela 1. Poređenje proračunatih vrednosti, softverski dobijenih vrednosti i izmerenih u ispitnoj stanici

Parametri	Jedinica	KONPRO	Račun	Ispitni protokol
I_0	A	19,57	19,5	19,1
P_0	W	9622,27	9572,9	10582
Q_{fe}	W	5985,28	5968,3	6984
Q_{dod}	W	2798,18	2650	2676
S	%	0,899	0,948	0,0081
n	ob/min	1486	1485,8	1487,9
Q_1	W	6742,8	6741,6	7193
Q_2	W	4860,1	5125,8	4304
J_k/J_n	r.j.	5,65	5,65	5,74
M_k/M_n	r.j.	0,84	0,82	0,89
M_{max}/M_n	r.j.	2,56	2,56	2,4
I_n	A	65,54	65,54	65,1
$\cos \varphi$		0,887	0,887	0,894
η	%	95,78	95,76	95,61

Drugu proveru iznetih razmatranja predstavljaju rezultati dobijeni ispitivanjem proizvodnog motora u ispitnoj stanici. Karakteristične vrednosti su tabelarno prikazane u Tabeli 1.

7. ZAKLJUČAK

Osnovni zadatak konstruktora, elektro inženjera, električnih mašina je što preciznije određivanje izlaznih karakteristika mašine. Za što tačnije određivanje ovakvih karakteristika potrebno je ogromno iskustvo u radu, dobro poznavanje rada električnih mašina u različitim uslovima, kao i tehnološke mogućnosti fabrike koja proizvod fizički realizuje.

Koristeći iskustva raznih autora, u ovom radu je pokazana validnost iznetih proračuna zasnovanih na opšte dostupnoj literaturi i znanju iz oblasti električnih mašina. Rad je usredsređen na razvijanju modela za proračunavanje asinhronog kaveznog motora, međutim uz minimalne promene prikazani postupak se može sprovesti za proračunavanje kliznokolutnog asinhronog motora.

Verifikacija proračuna iznetog u radu je potvrđena tabelarnim poređenjem rezultata dobijenih računom u radu, softverom KONPRO i na kraju ispitanim protokolom proizvedenog motora 1.KO 5450-4 za koji je proračun rađen.

Iz prikazanih rezultata se može zaključiti da se rezultati analitičkog proračuna sa vrednostima dobijenih softverski skoro poklapaju ali neznatno odstupaju od izmerenih vrednosti na proizvedenom motoru. Ova odstupanja su neminovna pošto proračun, zbog svoje složenosti, zahteva uprošćavanje analize stvarnih fizičkih pojava u motoru.

Tendencija razvoja asinhronih motora na svetskom tržištu je usmerena ka smanjivanju gabaritnih mera i povećanju stepena korisnog dejstva. Da bi se išlo u korak sa izazovima koje zadaje jaka konkurencija, potrebno je neprestano inoviranje znanja o fizičkim dešavanjima u motoru i implementaciji tog znanja u proračun asinhronih motora.

8. LITERATURA

- [1] Dr. inž. M. Livšic. (1950). Električne mašine, Knjiga III, Proračunavanje i određivanje mera. Beograd: Naučna knjiga.
- [2] Dr. inž. M. Livšic. (1950). Električne mašine, Knjiga I, Opšti osnovi. Beograd: Naučna knjiga.
- [3] Juha Pyrhonen, Tapani Jokinen, Valeria Hrabovcova. (2008). John Wiley & Sons, Ltd.
- [4] Vladimir V. Petrović. (1959). Uput u proračun asinhronog motora. Beograd: Naučna knjiga.

Kratka biografija:



Kristina Vujkov je rođena u Novom Sadu 1992. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti elektroenergetike odbranila je 2015. god. U ATB Sever-u radi od 2016. god. kao konstruktor električnih mašina.

ПРОНАЛАЖЕЊЕ РАЊИВОСТИ У СОФТВЕРУ НА ОСНОВУ ДИЈАГРАМА ТОКА ПОДАТАКА**DETECTION OF SOFTWARE VULNERABILITIES USING DATA FLOW DIAGRAMS**

Немања Миладиновић, Факултет техничких наука, Нови Сад

ПРИМЕЊЕНЕ РАЧУНАРСКЕ НАУКЕ И ИНФОРМАТИКА – Информациона безбедност

Кратак садржај – У раду су описани процес безбедног развоја софтвера и његов значај, са акцентом на фазу моделовања претњи и алате креиране у ту сврху. Представљен је систем за проналажење напада, који проналази потенцијалне нападе на дијаграму тока података. Описане су његове функционалности и компоненте.

Abstract – This paper describes the process of software development and its significance, with emphasis on the phase of threat modeling and tools created for this purpose. The exploits detection system, which finds potential exploits on a data flow diagram is presented. Its functionalities and components are described.

Кључне речи: Моделовање претњи, дијаграми тока података, база знања, Java, XML.

1. УВОД

Безбедност софтверског система би требало да буде веома битан фактор приликом његове израде, што није увек случај.

Последице неадекватне заштите система су напади који могу имати крајње различите мотиве и последице. *Morris worm* [1] је био експериментални напад који за циљ није имао наношење штете. *Stuxnet* [2] је представљао напад САД-а и Израела на Иран у циљу уништења иранског нуклеарног програма. Напад на компанију *Ashley Madison* [3] покренут је са жељом да се осуди неморално понашање промовисано од стране ове компаније. *WannaCry* [4] је класичан пример напада који за циљ имају искључиво финансијску добит нападача.

Нападаци користе најразличитије рањивости система како би извршили свој напад, а заштита система од напада није нимало једноставна операција. Процес безбедног развоја софтвера нуди савете и кораке за имплементацију безбедности у сам процес развоја софтвера.

Један од корака овог процеса јесте и моделовање претњи којим се потенцијалне претње проналазе и рангирају према њиховом негативном утицају на систем. Моделовање претњи врше стручњаци при чему као помоћ могу користити специјализовани софтвер.

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био др Горан Сладић, ванр. проф.

2. ПРОЦЕС БЕЗБЕДНОГ РАЗВОЈА СОФТВЕРА

Примери софтвера који откривају сигурносне пропусте за жељени софтверски систем су *Threat Modeling Tool (TMT)* и Систем за проналажење напада (енг. *Exploits Detection System*) (СПН), који је тема овог рада.

Процес безбедног развоја софтвера (енг. *Security Development Lifecycle – SDL*) осмислила је софтверска компанија Мајкрософт (енг. *Microsoft*). *SDL* представља скуп доказано најбољих пракси и алата који су успешно коришћени при развоју софтвера. Овакав приступ развоју софтвера омогућио је откривање и уклањање његових рањивости у најранијим фазама његове израде, чиме је значајно унапређен квалитет крајњег производа. *SDL* се састоји од седам фаза [5]:

- *Тренинг* подразумева обуку и креирање свести код свих учесника у изради софтвера о значају креирања безбедног софтвера, чиме се образује запослени кадар који ће независно од своје позиције бити у стању да допринесе безбедности софтвера [5].
- *Прикупљање захтева* представља почетак рада на пројекту, где се разматрају основни захтеви које софтвер треба да испуњава као и његова безбедност, а потом се врши њихово усклађивање са пословним потребама и трошковима [5].
- *Дизајн* обухвата креирање функционалне и структурне спецификације софтвера коришћењем најбољих програмерских пракси. Спровode се и анализе којима се откривају безбедносне претње и рањивости софтвера – моделовање претњи [5].
- *Имплементација* је процес креирања и документовања софтверског производа. Како би софтвер био безбедан потребно је држати се следећих препорука: користити одобрене алате, одбацити небезбедне функције и извршавати анализе програмског кода [5].
- *Верификација* је фаза процеса безбедног развоја софтвера, у којој је потребно утврдити да ли је софтвер у сагласности са безбедносним принципима који су установљени у претходним фазама [5].
- *Пуштање у рад* означава пуштање производа у јавну употребу, а подразумева да је софтвер успешно прошао све тестове и да су учесници у његовом развоју спремни за све изазове који их могу очекивати када корисници почну да конзумирају њихов софтвер [5].
- *Одржавање* почиње оног тренутка када се софтвер пусти у продукцију, траје све до тренутка његовог повлачења из употребе и подразумева реаговање произвођача софтвера у случају откривања сигурно-

сног пропуста над софтвером. Реакција може бити у виду упутства које компанија шаље корисницима како би се заштитили или у виду ажурирања софтвера како би се отклонила рањивост [5].

3. СИСТЕМИ ЗА МОДЕЛОВАЊЕ ПРЕТЊИ

Моделовање претњи је кључни корак у процесу безбедног развоја софтвера. Циљ моделовања претњи је проналажење, рангирање и разумевање претњи у раним фазама развоја софтвера. Процес моделовања претњи може се поделити у три основне фазе [6]:

- *Дekomпоновањем система* реалан систем се репрезентује као ДТП¹. Уочавају се архитектура система, начин на који компоненте међусобно комуницирају и идентификују крајње тачке система. Анализирањем ресурса који се налазе у систему и правцима њиховог кретања унутар њега могу се утврдити компоненте система од посебне важности. Све ове информације обједињено се чувају у ДТП-у.
- *Откривање и категорисање претњи* врши се анализирањем ДТП-а. Претње се могу категоризовати STRIDE категоризацијом [7].
- *Одређивање противмера и ублажавање претњи* се врши за оне претње које су оцењене као потенцијално најозбиљније и најштетније, на основу њиховог ризика. Једна од метода којима се рачуна ризик појединачне претње је DREAD [8].

3.1. Threat Modeling Tool

Threat Modelling Tool (TMT) је софтверски алат за моделовање претњи који је направила софтверска компанија Microsoft. У циљу моделовања претњи TMT омогућава извршавање наредних активности [9]:

- *Креирање ДТП-а* помоћу графичког едитора где се бирају елементи, постављају на дијаграм, међусобно повезују и додељују им се додатне информације.
- *Анализа ДТП-а и генерисање скупа потенцијалних претњи* као резултат даје табеларни приказ претњи откривених на анализираном ДТП-у, које су категорисане и рангиране. Могуће је погледати детаљне информације о свакој откривеној претњи.
- *Предлагање противмера* од стране корисника софтвера, уносом њиховог описа у одговарајуће поље.
- *Креирањем извештаја* откривене потенцијалне претње и предложене противмера се обједињују у јединствен HTML документ.

TMT је могуће прилагодити потребама анализе одређеног система креирањем њему одговарајућег шаблона. За сваки шаблон могуће је креирати и описати посебне врсте елемената и претњи [9].

4. МОДЕЛ СИСТЕМА

Систем за проналажење напада (енг. Exploits Detection System) је софтверски алат који се користи за моделовање претњи. СПН је у највећој мери настао по угледу на TMT који је описан у претходном поглављу.

¹ Дијаграм тока података (ДТП) састоји се од чвор-елемената који су међусобно повезани ток-елементима (везама). Елементи су додатно описани својом врстом и атрибутима који тој врсти одговарају.

4.1. Структура система

Систем за проналажење напада састоји се од седам засебних компоненти, чијом колаборацијом се омогућава правилно функционисање система. Систем поседује седам компоненти, које су са улазним и излазним документом приказане на слици 1:

- *СПН Апликација* је основна компонента система и задужена је за комуникацију система са корисником, прихватање улазних података и спровођење процеса анализе ДТП-а. Руковање осталим компонентама система, преузимање њиховог садржаја и кораци у обради улазног дијаграма врше се иницијативом ове компоненте.



Слика 1. Структура СПН-а

- *База знања* је најбитнија компонента система за проналажење рањивости на дијаграму, јер се оно врши на основу информација које се овде налазе.
- *XML шема ДТП-а* дефинише врсте елемената и њихове атрибуте који се могу појавити на ДТП-у. Улога XML шеме ДТП-а је да систему достави опис дијаграма који се могу наћи на улазу у систем.
- *Дефиниције ресурса* који се користе на ДТП-у и при раду система дате су у XML документу. Ресурсе је неопходно дефинисати како би се могло приступити њиховим атрибутима и како би се кориснику посредством извештаја доставиле потребне информације.
- *Дефиниције напада* који су систему на располагању морају бити наведене у XML документу и користе се како би кориснику преко крајњег извештаја било представљено довољно информација о сваком пронађеном потенцијалном нападу.
- *Алат за мапирање XML шеме ДТП-а на класе* креира објекте који одговарају елементима дефинисаним XML шемом, а потом ове елементе пакује у посебну библиотечку, коју ставља на располагање систему.
- *Алат за претварање ДТП-ова* претвара ДТП-ове креиране помоћу TMT-а у ДТП-ове погодне за овај систем. Алат пролази кроз садржај TMT документа, проналази елементе и преписује их на одговарајуће елементе СПН ДТП-а.

4.2. Фазе у процесу анализе ДТП-а

Како би пронашао потенцијалне нападе за улазни ДТП, систем извршава следеће секвенцијалне фазе:

- *Провера и учитавање улазних докумената* подразумева синтаксну и семантичку проверу следећих XML докумената: дефиниције ресурса, дефиниције напада и ДТП-а који ће бити анализиран. Након успешне провере, они се учитавају.

- *Увезивање дефиниција ресурса* мења ознаке ресурса које су записане на елементима ДТП-а њиховим дефиницијама, јер оне садрже податке који су потребни у каснијим корацима.
- *Декомпоновање ДТП-а* на мање целине (шаблоне) врши се јер је ДТП исувише комплексан да буде анализиран као целина. Шаблон представља ток података од једног елемента (почетног) ка другом (крајњем) или само један елемент (крајњи) без тока података. Сваки шаблон описан је: крајњим и почетним елементом, повезујућим елементима, и листама дефиниција ресурса који се налазе на крајњем, односно почетном елементу.
- *Анализа шаблона* примењује правила дефинисана у бази знања над сваким шаблоном појединачно. Правила анализирају шаблоне и проналазе њихове рањивости, а када пронађу рањивост шаблону додељују ознаку напада који се потенцијално појављује експлоатацијом ове рањивости и ознаку ресурса који би био угрожен овим нападом.
- *Увезивање дефиниција напада* врши се заменом ознака напада и ресурса њиховим дефиницијама у свим шаблонима.
- *Креирање извештаја* врши се претварањем анализираних шаблона у извештај-шаблоне, преписивањем потребних информација. Извештај-шаблон садржи: почетни елемент шаблона, крајњи елемент шаблона и лису потенцијалних напада. Од листе извештај-шаблона креира се извештај, а записују се: датум и време његовог креирања и назив анализираних ДТП.

4.3. Употреба система

Корисник може интераговати са СПН-ом у циљу обављања неке од наредних пет функционалности:

- *Анализирање ДТП-а* је основна функционалност система. Пре почетка овог процеса неопходно је припремити XML документ са ДТП-ом који описује систем који корисник жели да анализира. Овај документ се преко корисничког интерфејса проследи апликацији као улаз. Потом корисник покреће анализу ДТП-а, а систем према дефинисаним корацима анализира ДТП. На крају успешног процеса систем кориснику предаје извештај са пронађеним потенцијалним нападима у виду XML датотеке.
- *Ажурирањем дефинисаних ресурса* мењају се постојећи или дефинишу нови ресурси, што повлачи измену садржаја XML датотеке са дефиниција ресурса. Измена ознаке ресурса захтева одговарајућу измену и у бази знања, а додавање новог ресурса повлачи креирање новог документа базе знања са правилима која ће одговарати овом ресурсу.
- *Ажурирање дефинисаних напада* постиже се изменама XML датотеке са дефиницијама напада. Додавање и измена дефиниције напада почињу мењањем садржаја ове датотеке. При измени ознаке напада потребно је ову измену испратити и у бази знања, а уколико се дефинише нови напад потребно је у бази знања додати правила која одговарају овом нападу.
- *Ажурирање дефинисаних типова елемената ДТП-а* врши се изменом шеме ДТП-а. Додавање нових врста елемената или измена атрибута постојећих

захтевају измене и у бази знања уколико нови елемент или нови атрибут има улогу при проналажењу рањивости. Након измене шеме ДТП-а, потребно је покренути помоћни алат за примену шеме ДТП-а, како би се систем прилагодио изменама.

- *Ажурирање базе знања* врши се када су резултати рада апликације нетачни и када постоји потреба за изменама неких од правила. Такође, база знања се може ажурирати и као последица извршавања неке од претходне три функционалности.

5. ИМПЛЕМЕНТАЦИЈА СИСТЕМА

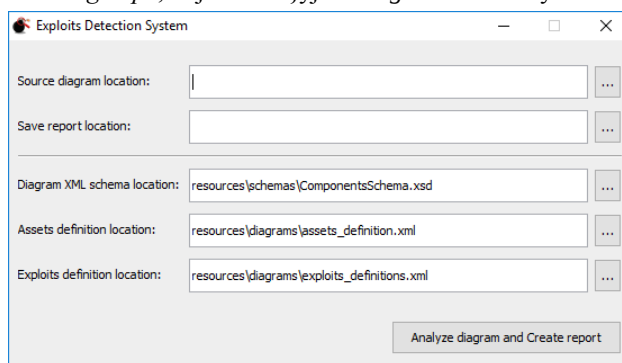
СПН се састоји од седам засебних компоненти:

1. СПН апликација;
2. База знања;
3. XML шема ДТП-а;
4. Дефиниције ресурса;
5. Дефиниције напада;
6. Алат за мапирање XML шеме ДТП-а на класе;
7. Алат за претварање ДТП-ова.

5.1. СПН апликација

СПН апликација имплементирана је коришћењем програмског језика Java 8 [10]. Библиотеке које се користе за рад са базом знања су *org.drools.drools-compiler* [11] и *org.kie.kie-api* [12]. Користи се и библиотека *model_lib* где је описана структура улазног ДТП-а, а коју креира алат за примену XML шеме.

Апликација поседује скроман интерфејс који је имплементиран Java-ином Swing библиотеком, и приказан је на слици 2. Акције интерфејса дефинисане су класама које наслеђују *AbstractAction* класу. Кораци којима се анализира улазни дијаграм описани су у методи *doInBackground* класе *WorkingSteps*, која наслеђује *SwingWorker* класу.



Слика 2. Интерфејс СПН апликације

Функционалности апликације дефинисане су у различитим класама.

Класа *Decomposer* декомпонује дијаграм на шаблоне. Класа *DiagramMerge* замењује ознаке ресурса и ознаке напад-ресурс њиховим дефиницијама.

KieRulesBase класа је задужена за креирање извршне базе знања.

Класа *Validator* врши синтаксну и семантичку проверу садржаја улазних XML докумената.

Класа *XMLLinker* врши учитавање и упис XML докумената.

Претварање шаблона у извештај-шаблоне врши се у једној од функција *WorkingSteps* класе.

5.2. База знања

База знања имплементирана је коришћењем *Drools* [13] базе знања и састоји се од више докумената у којима су дефинисана правила одлучивања. Сваки документ одговара једном од дефинисаних ресурса.

Правила базе знања састоје се од назива (*rule*), услова (*when*) и последице (*then*) која ће бити извршена уколико је услов испуњен. Сва правила у документу осим првог поседују и назив групе којој припадају (*agenda-group*), на основу ког је извршено њихово груписање. Правила која припадају истој групи налазе се у истом документу и активирају се истовремено.

Искључиво прво правило из докумената не припада ни једној групи правила, и једино ће ово правило бити активирано по почетку анализе шаблона. Уколико је услов првог правила испуњен активира се наредна група правила (која је предвиђена последицом), а која може узроковати активацију даљих група правила.

5.3. XML шема ДТП-а

ДТП је описан *XML* шемом, која дефинише врсте елемената који су могу појавити на ДТП-у, као и њихове атрибуте и њихов распоред у овом *XML* документу. ДТП који се користи као улаз у систем мора бити у складу са овом шемом како би могао бити анализиран.

5.4. Дефиниције ресурса

Ресурси су дефинисани у *XML* документу. Формат овог документа тачно је специфициран и прописује елементе који се могу појавити, њихове атрибуте и њихов распоред.

5.5. Дефиниције напада

Дефиниције напада наведене су у *XML* документу, чији формат је строго дефинисан.

5.6. Алат за мапирање XML шеме ДТП-а на класе

Овај алат је имплементиран као *batch* датотека (.bat) којој се као параметар прослеђује локација *XML* шеме за коју се креирају одговарајуће класе. Како би креирао датотеку са дефиницијама класа, овај алат позива *XJC* алат да креира *Java* класе које одговарају елементима описаним у *XML* шеми. Алат потом ове *Java* класе компајлира употребом *JAVAC* алата и на крају их *JAR* алатом пакује у посебну датотеку.

5.7. Алат за претварање ДТП-ова

Алат за претварање ДТП-ова имплементиран је као конзолна *Java 8* [10] апликација којој се као улазни параметар прослеђује локација .tm7 датотеке са описом ДТП-а који ће бити претворен. Алат из оригиналног ДТП-а проналази и учитава потребне информације, дефиниције: чвор-елемената, области, ток-елемената, граница и додатне описе ток-елемената. На основу ових дефиниција алат креира њима одговарајуће елементе који су погодни за СПН. Ове елементе потом чува у нову *XML* датотеку.

6. ЗАКЉУЧАК

Систем за проналажење напада је адаптиван и једноставан алат за моделовање претњи. Адаптивност је постигнута могућношћу измена његових компоненти у циљу прилагођавања алата потребама система који треба анализирати. Једноставност алата добијена је

коришћењем једноставних формата за имплементацију његових компоненти. Садржај улазних датотека може се формирати претварањем других ДТП-ова у одговарајући облик, што је показано имплементацијом Помоћног алата за претварање ДТП-ова. Начин на који систем декомпонује улазни ДТП на шаблоне омогућава детаљну и прецизну анализу дијаграма у циљу откривања његових рањивости.

Постојећи систем могуће је унапредити његовом оптимизацијом са становишта брзине и коришћене меморије. Додатни механизми за проналажење рањивости могу се увести имплементацијом компоненте за анализу рањивости технологија којима су компоненте анализираних система имплементиране – за сваки елемент ДТП-а дефинише се технологија којом је имплементиран, а потом се коришћењем базе рањивости за технологије (нпр. *NVD*) пронађу рањивости те технологије. Унапређење се може постићи и проширењем самог система додавањем нових компоненти, попут графичког едитора посредством ког би корисник креирао ДТП-ове. Друга компонента могла би да омогући креирање прегледних извештаја у *PDF* формату. Нове компоненте требало би објединити у складну целину креирањем базе компоненте.

7. РЕФЕРЕНЦЕ

- [1] Hilarie Orman. *The Morris worm: A fifteen-year perspective* <http://ieeexplore.ieee.org/abstract/document/1236233/>
- [2] Jon R. Lindsay. *Stuxnet and the Limits of Cyber Warfare* http://erikgartzke.com/assets/lindsay2013_stuxnet.pdf
- [3] Ben Light. *The rise of speculative devices: Hooking up with the bots of Ashley Madison* <http://firstmonday.org/ojs/index.php/fm/article/view/6426/5525>
- [4] EY. "WannaCry" ransomware attack - Technical intelligence analysis. [http://www.ey.com/Publication/vwLUAssets/ey-wannacry-ransomware-attack/\\$File/ey-wannacry-ransomware-attack.pdf](http://www.ey.com/Publication/vwLUAssets/ey-wannacry-ransomware-attack/$File/ey-wannacry-ransomware-attack.pdf)
- [5] Microsoft. *Simplified Implementation of the Microsoft SDL* <https://www.ida.liu.se/~TDDC90/literature/papers/sdl.pdf>
- [6] OWASP. *Application Threat Modeling* https://www.owasp.org/index.php/Application_Threat_Modeling
- [7] OWASP. *Threat Risk Modeling – STRIDE* https://www.owasp.org/index.php/Threat_Risk_Modeling#STRIDE
- [8] OWASP. *Threat Risk Modeling – DREAD* https://www.owasp.org/index.php/Threat_Risk_Modeling#DREAD
- [9] Microsoft. *Microsoft Threat Modeling Tool 2016 User Guide* <https://www.scribd.com/document/319642345/Threat-Modeling-Tool-2016-User-Guide>
- [10] Oracle. *Java Platform Standard Edition 8 Documentation* <https://docs.oracle.com/javase/8/docs/>
- [11] The JBoss Drools team. *Drools Documentation – Dependencies and JARs* https://docs.jboss.org/drools/release/6.0.1.Final/drools-docs/html_single/#d0e147
- [12] The JBoss Drools team. *Drools Documentation – Part II KIE* https://docs.jboss.org/drools/release/6.0.1.Final/drools-docs/html_single/#KIEChapter
- [13] Red Hat projects, JBoss. *Drools* <https://www.drools.org/>

Кратка биографија:



Немања Миладиновић рођен је у Новом Саду 1993. год. Године 2012. завршио је Гимназију Јован Јовановић Змај у Новом Саду. Дипломирао је 2016. год на Факултету техничких наука, одсек Електротехника и рачунарство, смер Примењене рачунарске науке и информатика.

PRIMENA FOTONAPONSKIH ELEKTRANA U NEMAČKOJ APPLICATION OF PHOTOVOLTAIC POWER PLANTS IN GERMANY

Nino Ignjić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu dat je kratak pregled i istorijat fotonaponskih (FN) ćelija, a zatim je prikazan solarni potencijal Nemačke, upoređivanje solarne energetike u Nemačkoj i Srbiji, kao i motivacija za uspešno implementiranje i veoma uspešan razvoj solarne energije, na kraju rada prikazana je proizvodnja jedne od preko 1,5 miliona FN elektrana u Nemačkoj.

Abstract – In this paper a brief overview and history of photovoltaic (PV) cells is given, followed by the solar potential of Germany, comparison of solar energy in Germany and Serbia, as well as the motivation for successful implementation and the very successful development of solar energy. At the end of the thesis, production of one of more than 1.5 million PV power plants in Germany is shown.

Ključne reči: Fotonaponske elektrane, solarni paneli, Nemačka, proizvodnja solarne energije

1. UVOD

Kada se govori o prirodnim, odnosno primarnim oblicima energije važno je za istaći da mogu da se podele na obnovljive i neobnovljive. Podela je izvršena s obzirom na vremensku mogućnost njihovog iscrpljivanja. Epitet obnovljivi, ovi izvori, duguju činjenici da se energija troši u iznosu koji ne premašuje brzinu kojom se stvara u prirodi.

Tek kada je čovečanstvo, pre tridesetak godina, postalo svesno ozbiljne opasnosti od promene klime koju je samo prouzrokovalo i kada je evidentno značajno smanjenje rezervi fosilnih goriva dosetilo se da se pogleda ka nebu. Sunčevo zračenje, odnosno energija Sunčevog zračenja je najveći i pri tome potpuno čist izvor energije.

Iskorištavanje energije Sunca već nekoliko godina beleži konstantan rast u gotovo celom svetu, te počinje dobijati važnu ulogu u elektroenergetskom miks u većini zemalja [8].

Tehnologije fotonaponskih ćelija i koncentrisane Sunčeve energije ubrzano se razvijaju, a ulaganja investitora postaju sve veća. Za većinu zemalja najveće prepreke za veće iskorištavanje energije Sunca predstavljaju zamršeno zakonodavstvo i nedovoljni podsticaji od strane vlada.

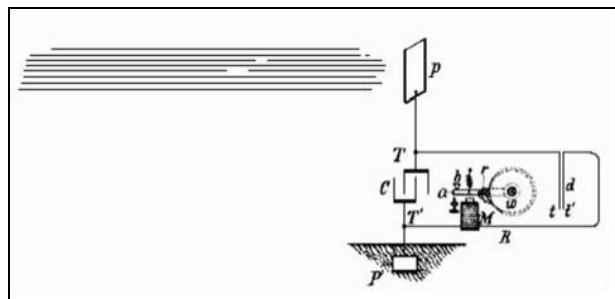
2. ISTORIJAT

Prvom originalnom solarnom ćelijom smatra se ona izrađena 1883 god. (Charles Fritts) od selen kao poluprovodnika s vrlo tankim slojem zlata [7]. Ova prva ćelija

imala je efikasnost pretvaranja energije manju od 1%. Nakon toga 1888 – 1897. god. objavljeno je nekoliko patenata pod naslovom "Solar cell". Godine 1960. Hoffman Electronics je razvio solarne ćelije sa efikasnošću od 14% i time je nagovešten nadolazeći tehnološki razvoj u solarnoj industriji [7]. U zemaljskim uslovima solarne ćelije počinju, doduše skromnu, primenu na usamljenim objektima, svetionicima, mernim i istraživačkim mobilnim jedinicama, stambenim i industrijskim objektima.

Da je solarna energija pozitivno prihvaćena govori i činjenica da je 1990. god. prvi put postavljen solarni sistem na krovu katedrale u Magdeburg-u u Istočnoj Nemačkoj.

Interesantno je, i veoma važno za istaći, da je prvu solarnu mašinu osmislio Nikola Tesla 1901 godine (patent US 685957) [6]. Izgled ove mašine dat je na slici 1.



Slika 1. Teslin patent za korišćenje solarne energije [6].

3. UTICAJ VREMENSKIH PRILIKA NA PERFORMANSE SOLARNIH PANELA

Za svaku fotonaponsku elektranu je izuzetno važno izvršiti analizu njene efikasnosti, kao i analizu svih parametara koji na nju mogu da utiču. Električna energija FN sistema, koja se svakodnevno predaje elektroenergetskom sistemu, određuje se preko srednje dnevne insolacije, površine panela i srednje vrednosti efikasnosti. Kao vremenski period može se posmatrati jedan dan, mesec ili godina.

3.1. Uticaj insolacije na U-I karakteristiku

Na slici 2 (levo) predstavljena je funkcija rada jedne FN ćelije, odnosno FN panela u toku sunčanog i oblačnog dana, na slici 2 (desno) odgovarajuća U-I karakteristika. Na osnovu analize karakteristike sa slike 2. može da se primeti da kako insolacija opada, struja kratkog spoja takođe opada u direktnoj proporciji. Polovljenjem insolacije, naprimer, struja kratkog spoja opada za pola.

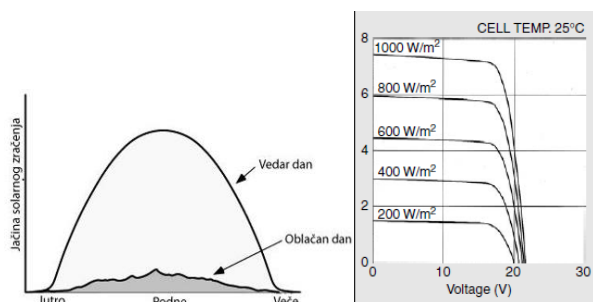
3.2. Uticaj temperature na efikasnost solarnog panela

Na slici 3 dat je uticaj temperature na U-I karakteristike FN panela. Na osnovu analize karakteristika sa slike 3, može se doneti zaključak da kako raste temperatura, napon praznog hoda se smanjuje, a struja kratkog spoja se

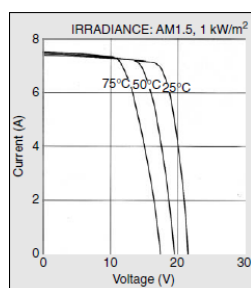
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Vladimir Katić.

povećava, ali neznatno. Iz toga proizilazi da je proizvodnja fotonaponskih ćelija i modula je veća pri hladnijim, ali vedrim danima, nego pri visokoj temperaturi. Zbog ovakvih uticaja na performanse FN modula, temperatura treba da bude uvažena u svim proračunima.



Slika 2. Prikaz moguće proizvodnje panela u slučaju vedrog i oblačnog dana[1]



Slika 3. Prikaz uticaja temperature na efikasnost solarnog panela [1]

4. SOLARNA ENERGIJA U NEMAČKOJ

Nemačka u proseku zbog prostrane teritorije ima skoro isti solarni potencijal kao Aljaska, koja ima 3,08 sunčanih sati po danu, što je dosta skromno. Radi poređenja, potencijal Srbije je 30-40% veći od onog u Nemačkoj.

Uprkos veoma lošem solarnom potencijalu, u prethodnih desetak godina, Nemačka prednjači po broju instalisanih FN panela u celom svetu. Zaključno sa krajem 2016. god. ukupna instalisana snaga FN sistema iznosila je oko 42 GW, a broj elektrana popeo se na preko 1,5 milion [5].

Državni cilj je da se konstantnim unapređenjem i proširivanjem kapaciteta obnovljivih izvora energije (OIE) sve više i više energije proizvodi na ovaj način. Teži se tome da do 2020. godine 20% celokupne energije bude proizvedeno iz OIE, do kraja 2030. godine da taj iznos bude 30%, a do kraja 2050. godine taj iznos bude čak 80% [1,5].

Sa najvećim kapacitetom solarne energije od bilo koje države u Evropi, Nemačka je počela da skladišti solarnu energiju kako bi poboljšala potrošnju.

Prošle godine, 41% novih instalisanih solarnih elektrana su opremljene baterijama za skladištenje energije [5, 8]. Time je Nemačka postala i zemlja sa najviše baterija za skladištenje energije.

Trenutno postojeće FN elektrane imaju kapacitet 4 puta veći od nuklearnih elektrana, za koje je odlučeno da se pogase do 2022. godine, i to predstavlja 16% od ukupnog broja instalisanih FN elektrana u svetu [5].

5. PODSTICAJNE MERE U NEMAČKOJ (FEED-IN TARIFA)

Radi podsticanja investiranja i korišćenja OIE, a posebno sunca, vetra, biomase, hidro energije i dr. u Nemačkoj su uvedene podsticajne mere u vidu feed-in tarifa. Ova vrsta podsticaja predviđa da proizvedena energija iz OIE ima prioritet kod tržišnog plasmana i da se plaća po državno određenim cenama. Na ovaj način vrednost izlazne energije zavisi od garantovane cene i proizvedene energije, pri čemu je svaka elektroprivreda u obavezi da otkupi proizvedenu električnu energiju iz OI energije po toj fiksnoj ceni, nezavisno od tržišnih uslova i trenutnih potreba.

U Nemačkoj, 2007. godine, za energiju dobijenu iz novih instaliranih fotonaponskih elektrana, distribucija je plaćala između 0.38 €/kWh i 0.54€/kWh.

U Nemačkoj, podsticajna otkupna cena za električnu energiju proizvedenu putem krovnih FN sistema iznosi 12.31 €/kWh. Visina naknade za slobodnostojeće FN sisteme zavisi od ugovora.

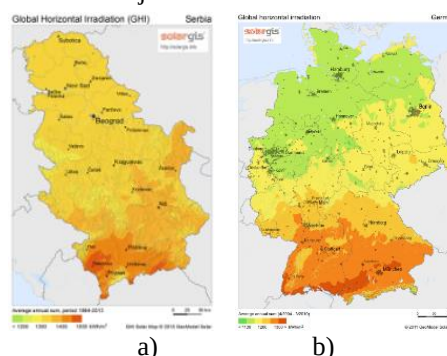
Slobodnostojeći FN sistemi male snage (<10 MW) imali su srednju vrenost *Feed-In* tarife(FIT) od 7.41 €/kWh. Slobodnostojeći FN sistemi, preko 10 MW, se više ne stimulišu preko podsticajnih tarifa.

Danas, *FIT* za male krovne FN sisteme iznose 12.88 €/kWh, dok za velike FN sisteme kao što su solarni parkovi iznose 8.92 €/kWh. Ove tarife opadaju svake godine, i opadaju brže nego za bilo koji drugi OI energije.

6. POREĐENJE SOLARNE ENERGETIKE U NEMAČKOJ I SRBIJI

Na većini teritorije Srbije broj sunčanih dana je znatno veći nego u mnogim evropskim zemljama i kreće se između 1.500 i 2.200 časova godišnje. Potencijal solarne energije u Srbiji iznosi 0,64 miliona toe godišnje tj. oko 16,7% ukupnog potencijala OI. Najveći potencijal za korišćenje solarne energije imaju gradovi u južnom delu Srbije, i to Niš, Kuršumlja i Vranje.

Prosečan intenzitet sunčevog zračenja na teritoriji Srbije kreće se od 1,1 kWh/m²/dan na severu do 1,7 kWh/m²/dan na jugu tokom januara, a od 5,9 do 6,6 kWh/m²/dan tokom jula.



Slika 4. a) prikaz solarnog potencijala Srbije, b) prikaz solarnog potencijala Nemačke [5]

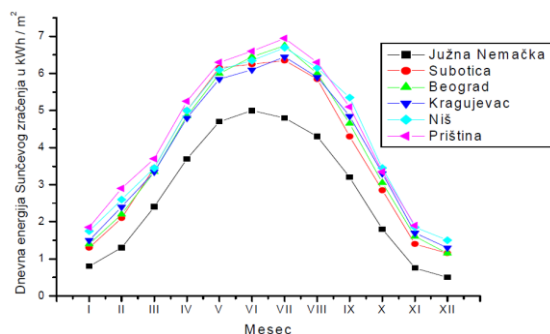
U Nemačkoj je totalno drugačija situacija, i prosečno godišnje globalno zračenje se kreće u intervalu od 951 do 1.257 kWh/m², dok prosečno godišnje trajanje sijanja Sunca iznosi od 1.300 do 1.900 sati. Preko 1,5 miliona FN sistema instaliranih širom Nemačke generišu oko 35,1 milijardi kWh električne energije godišnje, što predstavlja 6% ukupne potrošnje električne energije u Nemačkoj.

Slika 5. pokazuje da je osunčanost Srbije znatno veća nego osunčanost južne Nemačke i to skoro 40% veća. U toku vegetacionog perioda, od Aprila do Septembra, količina dobijene energije od Sunca se kreće od 4,9 kWh/m² na zapadu, do 5,7 kWh/m² na jugoistoku Srbije.

Najniže vrednosti u Srbiji podudaraju se sa najvećim vrednostima u Austriji i Nemačkoj, koje intenzivno koriste Sunčevu energiju.

Prilikom vršenja ispitivanja i analiza, došlo se do zaključka da su vrednosti ove energije u brdskim krajevima zapadne Srbije niske.

To je posledica povećane dnevne oblačnosti u tim krajevima u letnjem periodu.



Slika 5. Prikaz dnevne osunčanosti nekih gradova u Srbiji i južnog dela Nemačke [3]

U Srbiji se takođe primenjuju feed-in tarife kao vid podsticaja za proizvođače električne energije i to od 2009. god. Međutim, nagle promene na tržištu opreme za FN elektrane, a posebno drastično pojeftinjenje solarnih panela, dovela su i do izmena nivoa podsticaja. Za samo četiri godine visina podsticajne otkupne cene, u Srbiji, je sa 23 €/kWh u 2009. god. pala na vrednost između 16,25 i 20,66 €/kWh u zavisnosti od instalisane snage i načina instalacije u 2013. god.

Dalja pomeranja na tržištu, još više su umanjila vrednost i potrebu za podsticajima. Tako je nova podsticajna regulativa iz 2016. godine još više oborila feed.in tarifu, naročito za solarne elektrane na tlu, koja sada iznosi samo 9 €/kWh, što je niže od podsticaja za vetar (9,2 €/kWh). Time su praktično investicije u velike FN elektrane dobile negativan podsticaj i mala je verovatnoća da će ova oblast energetike u Srbiji imati tako dinamičan razvoj kao u Nemačkoj.

7. ANALIZA PROIZVODNJE JEDNE FN ELEKTRANE U NEMAČKOJ

Da bi se ilustrovala efikasnost korišćenja solarne energije u Nemačkoj, u nastavku će biti predstavljen primer jedne krovne FN elektrane.

Solarna elektrana u Botropu, na krovu njihovog zatvorenog skijaškog centra, predstavljala je jednu od najvećih i najizazovnijih solarnih elektrana za izgradnju. Osposobljena je 11.08.2011. godine i predstavlja elektranu snage oko 1,4 MWp. Sastoji se od 18.500 solarnih panela. Pored toga što joj je primarna funkcija proizvodnja energije, mora da obezbedi i smanjenje gubitaka onutar objekta koja se tiču temperature.

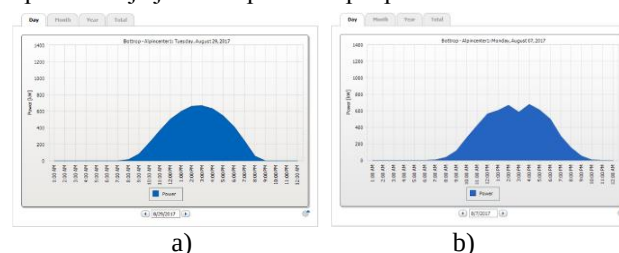
Snaga ove solarne elektrane je 1.392,4 kWp i godišnje proizvede oko 1.044.300 kWh (750 kWh/kWp). Na godišnjem nivou smanji emisiju CO₂ za oko 731 tonu.

7.1. Poređenje proizvodnje pri vedrom i oblačnom danu

Ako se uzme u obzir da se ceo princip pretvaranja solarne energije u električnu svodi na obasjanost FN ćelija i panela, to znači da se proizvodnje pri sunčanim i oblačnim danima razlikuju u velikoj meri.

Na slici 6.a) može da se vidi da je ceo dan bilo sunčano, i da nije bilo nikakvih smetnji u proizvodnji električne energije. Od izlaska sunca, kako je intenzitet zračenja bio sve intenzivniji, tako je rasla i proizvodnja, sve do 15 časova kada je intenzitet počeo da slabi, pa samim time i proizvodnja.

Na slici 6.b) može da se primeti naglo opadanje u proizvodnji između 14 i 15 časova, u tom periodu sunčevi zraci su bili sprečeni u dopiranju do panela (došlo je do iznenadnog naoblačenja). Između 15 i 16 časova nebo se razvedrilo i proizvodnja je u tom periodu opet počela da raste.



Slika 6. a) prikaz proizvodnje tokom sunčanog dana, b) prikaz proizvodnje tokom sunčanog dana sa naoblačenjem [5]

7.2. Poređenje proizvodnje u različitim godišnjim dobima

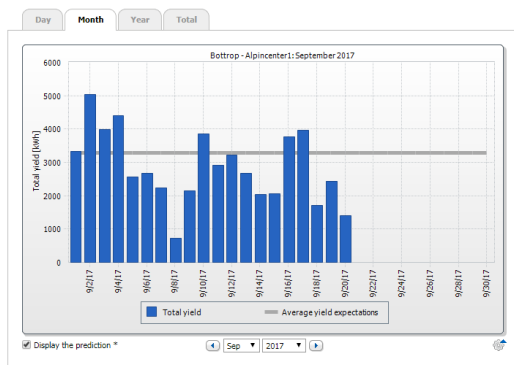
Kao što je već rečeno, proizvodnja jedne solarne elektrane veoma zavisi od intenziteta i količine zračenja koje dospe do panela. To znači da proizvodnja u različitim dobima godine neće biti ista, naprimer ne može da se poredi proizvedena energija u nekom mesecu tokom zime i nekom mesecu tokom leta. Na slici 7 dat je prikaz proizvodnje za dva meseca sa min/max proizvodnjom.



Slika 7. a) prikaz proizvodnje solarne elektrane tokom januara 2017. god., b) prikaz proizvodnje tokom jula 2017. god. [5]

Prilikom posmatranja i analize proizvodnje u januaru, što je prikazano na slici 7.a), utvrđeno je da je najviše proizvedeno 26.01.2017., nešto više od 1.400 kWh, što u poređenju sa proizvodnjom, naprimer, 18.07.2017. koja se vidi na slika 7b. i iznosi skoro 6.000 kWh, nije neka velika proizvodnja.

Na slici 7.b) može da se vidi da proizvodnja u julu 2017. godine nije bila baš na zavidnom nivou, jer je bila ispod prosečne proizvodnje za to doba godine 26 dana u mesecu, a samo 5 dana u mesecu proizvodnja je premašila prosečnu predviđenu potrošnju za taj mesec.



* The expected average yield results from a manual specification. Depending on the orientation and the irradiation fluctuations between the individual years, there might be considerably larger deviations in contrast to the expected average yield.

Slika 8. Prikaz mesečne proizvodnje za septembar 2017. god. [5]

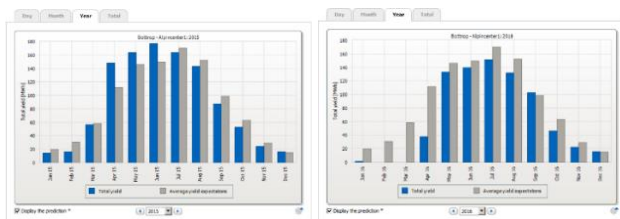
Na slici 8 može se videti proizvodnja tokom septembra 2017. god. Proizvodnja je na relativno dobrom nivou ako se u obzir uzme da je septembar mesec početka jeseni i da su moguće česte kiše. Najveća proizvodnja je ostvarena 02.09.2017. i iznosila je oko 5.000 kWh.

7.3. Poređenje proizvodnje elektrane na godišnjem nivou

Kako planeta stari, tako je temperatura na njoj sve veća jer je ozonski omotač svake godine sve više oštećen, ali to ne mora da znači da će svake sledeće godine, u istom mesecu, biti sve viša proizvodnja električne energije.

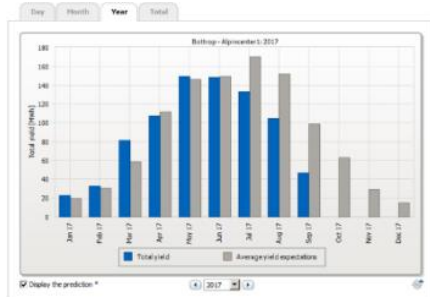
Ovo se tačno vidi ako se uporede meseci u kojima bi trebalo da bude najviše proizvedene energije, to su april, maj, jun, jul i avgust. Na slici 9 može da se vidi da je 2015. god. u tim mesecima bilo mnogo više proizvedene energije nego 2016. i 2017. god.

To, još jednom, dokazuje da proizvodnja energije zavisi samo od vremenskih prilika, pošto je očigledno da je 2016. i 2017. god. bilo mnogo lošije vreme u istim mesecima u odnosu na vreme 2015. god.



a)

b)



c)

Slika 9. Prikaz proizvodnje a) 2015. god., b) 2016. god. i c) 2017. god. [5]

8. ZAKLJUČAK

Solarna energija tj. energija dobijena iz sunčevog zračenja kao jedna od energija koja se dobija iz OIE je veoma bitna i predstavlja jednu od energija na kojima će se zasnivati budućnost. Ona ne samo da štedi novac time što koristi nešto što je već prisutno, i ukoliko se ne iskoristi odlazi u nepovrat, već i njenom proizvodnjom može da se smanji proizvodnja energije iz drugih izvora. Time ne samo da se čuvaju ti energenti već se čuva i životna sredina, jer joj način iskorišćenja solarne energije ne šteti i ne dovodi do ugrožavanja okoline.

Energija dobijena iz OIE mogla bi da podmiruje većinu energetske potrebe sveta do 2050. god., ali samo ako države dramatično povećaju investiranje i davanje političke podrške tehnologijama koje koriste snagu vetra i Sunca za dobijanje energije. Pokazalo se da nije problem dostupnost OI energije, već politika koja će odlučivati da li će industrija obnovljive energije rasti ili će se suzdržavati tokom narednih godina. Ključno pitanje koje treba postaviti je koliko će vremena biti potrebno da se podigne svest o neophodnosti razvoja i korišćenja obnovljivih izvora energije i koliko političke i ekonomske štete, kao i štete po životnu sredinu, je prihvatljivo u međuvremenu.

9. LITERATURA

- [1] G.M. Masters, „Renewable and Efficient Electric Power Systems“, John Wiley & Sons, Hoboken, NJ-USA, 2004.
- [2] O. Lalatović, „Fotonaponski i fototerminički uređaji“, Diplomski rad, Elektrotehnički fakultet, Beograd, 2014.
- [3] B. Radičević, Dušan Mikić i Đukan Vukić, „Energetski potencijal sunca u Srbiji i primena energije sunca u poljoprivredi“, Poljoprivredna tehnika, God. XXXIV, Br.4, Beograd, 2009, pp.53-62.
- [4] I.M. Babić, „Modelovanje uticaja vremenskog profila solarnog zračenja na efekte rada fotonaponskih sistema u elektroenergetskom sistemu“, Doktorska disertacija, Elektrotehnički fakultet, Beograd, 2016.
- [5] H. Wirth, „Recent Facts about Photovoltaics in Germany“, Fraunhofer ISE, Freiburg, 2017.
- [6] <https://www.sunnyportal.com/Templates/PublicPageOverview.aspx?page=7482deb4-024d-496f-8a50-1cf28c23b7c1&plant=e6a3f909-68fd-4512-8654-078c20b8c30d&splang=en-US>
- [7] <http://www.solarnipaneli.org/2010/12/istorija-solarne-celije-i-deo/>
- [8] <http://energypost.eu/germany-sets-new-solar-storage-record/>

Kratka biografija:



Nino Ignjić, rođen je u Brčkom 1991. god. Diplomski rad na Fakultetu tehničkih nauka u Novom Sadu iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine odbranio je u julu 2015. god. sa ocenom 10, a master rad na istom fakultetu i smeru u oktobru 2017. godine.

**ELEKTROMOTORNI POGONI I NAČINI STARTOVANJA
MOTORA U FABRICI SOLI****ELECTRIC DRIVES AND MODELS OF ENGINE STARTING
IN SALT FACTORY**

Sreten Petrović, Veran Vasić, Đura Oros, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U okviru ovog rada predstavljeni su tipovi rešenja za puštanje u rad asinhronih motora, koji su korišćeni prilikom projektovanja i izgradnje fabrike soli. Analizirane su njihove prednosti i mane i predstavljeni su rezultati merenja i njihov uticaj na elektroenergetsku mrežu.

Abstract – As part of this work presents the types of solutions for commissioning asynchronous motor, which were used during the design and construction of the salt factory. Their advantages and disadvantages were analyzed and the measurement results were presented along with their impact on the electrical grid.

Ključne reči: Asinhroni motor, puštanje u rad, elektromotorni pogon

1. UVOD

Pojavom elektromotora počinje razvoj elektromotornih pogona kakve mi danas poznajemo. Prvi elektromotorni pogoni bili su vrlo jednostavni, kao i sami električni motori. Pojava naizmjeničnih sistema je omogućila porast snage energetske izvora nekoliko puta, prenos energije na velike daljine i smanjuje cenu električne energije, a pojava asinhronog motora smanjuje cenu motora iste snage za nekoliko puta, čime asinhroni motor potiskuje motor jednosmerne struje u postojećim pogonima.

Razvojem energetske elektronike, energetske pretvarača i primenom mikroprocesora, omogućeno je da se u širokom opsegu upravlja parametrima asinhronih motora, efikasno i sa malo gubitaka.

Jedan od problema koji se javlja kod asinhronih motora je njihov start odnosno puštanje u rad. Pokretanje asinhronog motora je proces koji započinje u trenutku u kojem je rotor u stanju mirovanja, a završava se kada se pri odgovarajućoj brzini obrtanja, izjednače razvijeni momenat motora i otporni momenat radne mašine. Cilj ovog rada je upoznavanje sa elektromotornim pogonima i načinom startovanja motora u fabrici soli.

2. TEHNOLOGIJA PROCESA PROIZVODNJE

Mehaničko povezivanje opreme u jednu tehnološku celinu nije podržano od strane upravljačkog sistema, tako da svaka od njih funkcioniše kao nezavisna celina.

NAPOMENA:

Ovaj rad je proistekao iz master rada čiji je mentor bio dr Đura Oros.

Pogon za prerađu morske soli funkcioniše kao skup više nezavisnih tehnoloških celina, povezanih transportnim putevima u jednu tehnološku celinu.

Tehnološki postupak prerade sirove morske soli obuhvata niz tehnološki nezavisnih procesa: usipni koš (početak tehnološkog procesa), proces pranja soli, sistem za vodo-slabdevanje mašine za pranje, sistem za hlađenje hidrauličnog ulja – čiler, proces sušenja soli, separacija soli (odvajanje u frakcije), proces jodiranja soli, transportni putevi, sistem za slabdevanje komprimovanim vazduhom, mlevenje soli i odlaganje u silose.

4. ELEKTROMOTORNI POGONI U FABRICI**4.1. Asinhroni motor**

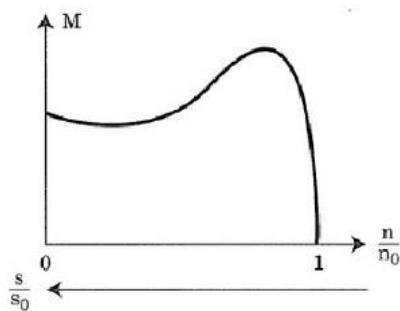
Asinhronne mašine su elektromagnetni uređaji za pretvaranje električne energije u mehaničku i obratno. Kad rade u motorskom režimu rada pretvaraju električnu energiju u mehaničku, a mehanička energija se pretvara u električnu kad rade kao generatori. Prelazak iz motorskog u generatorski režim rada se kod asinhronih motora ostvaruje dodavanjem izvora mehaničke snage umesto tereta. Naravno, potrebni su i odgovarajući uređaji za regulaciju i upravljanje.

Asinhroni motori su najčešće korišćena vrsta električnih mašina, i predstavljaju pogonsku snagu industrije. Za manje snage se koriste jednofazni asinhroni motori koji se priključuju na napon električne mreže od 230V. Proizvode se u velikim serijama i imaju široku upotrebu u različitim vrstama kućnih aparata. Za područje srednjih i velikih snaga koriste se trofazni asinhroni motori, i priključuju se na mrežu trofaznog napona. Imaju široku primenu u industriji, pogotovo u pogonima koji ne zahtevaju izuzetno precizno i brzo upravljanje, gde su jednosmerni motori i dalje dosta zastupljeni.

4.2. Start asinhronog motora

U praktičnoj primeni puštanje u rad asinhronog motora predstavlja veoma važan proces. Vrednost polaznog momenta i struje su od bitnog značaja pri puštanju asinhronog motora u rad.

Ove struje mogu da izazovu visoka zagrevanja namotaja motora kao i velike padove napona, što može negativno da utiče na druge prijemnike u mreži. Da bi rotor motora pri puštanju u rad mogao da pređe u obrtno kretanje, polazni momenat koji motor razvija mora da bude veći od otpornog momenta koji na vratilu proizvodi radna mašina. Sa druge strane vrednost polazne struje za datu mrežu ne sme da prevazilazi određene granice koje zavise od snage mreže.

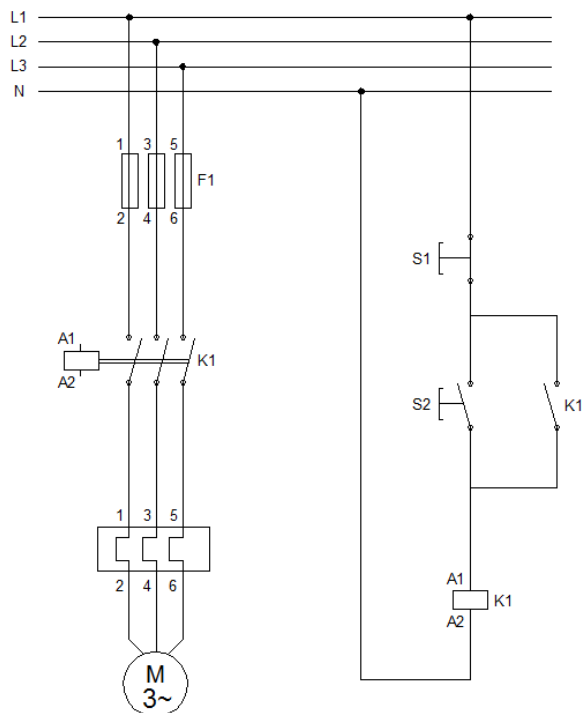


Slika 1. Izgled mehaničke karakteristike asinhronog motora

4.2.1 Direktni start motora

Najjednostavniji i najprostiji oblik starta asinhronog motora je direktno priključenje motora na mrežu. Prilikom direktnog povezivanja asinhronog motora na mrežu delovi koji se koriste su osigurač, prekidač, kontaktor i zaštitni relej.

Kontaktori se kontrolišu zasebnim start-stop tasterima koji su namenjeni za pokretanje i zaustavljanje motora, a pomoćni kontakt na kontaktoru kojim se upravlja takođe preko tastera za start motora, koristi se kao kontakt za držanje stanja. Kontaktor je električno zatvoren dok motor radi.

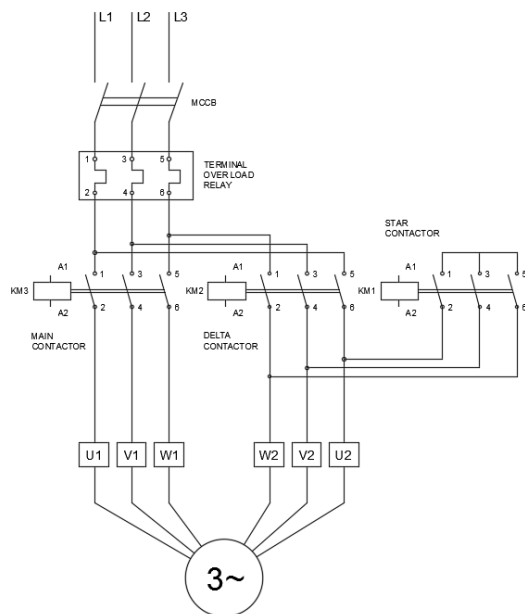


Slika 2. Direktni start asinhronog motora

4.2.2 Start motora prebacivačem zvezda trougao

Ovo je metoda startovanja motora pomoću redukovanja napona, što se postiže fizičkom rekonfiguracijom navojaka motora.

Tokom startovanja navojci motora su povezani u zvezdu što smanjuje napon na sva tri navojka, ovo takođe smanjuje i polazni obrtni momenat motora. Posle određenog vremena navojci se prevezuju u trougao i motor nastavlja da radi normalno. Ovi starteri spadaju u najkorišćenije metode startovanja motora i koriste kako bi smanjili polaznu struju koju motor povlači prilikom starta.



Slika 3. Prebacivač zvezda-trougao

4.2.3 Start i regulacija brzine obrtanja asinhronog motora pomoću frekventnog pretarača

Frekventni pretvarači su elektronski uređaji koji omogućavaju upravljanje brzinom trofaznih asinhronih motora, pretvarajući ulazni napon fiksne amplitude i frekvencije u napon promenjive amplitude i frekvencije.

Pored osnovne funkcije upravljanja brzinom AC motora, frekventni pretvarači integrišu i brojne druge funkcionalnosti kao što su: zaštita motora, alarmiranje, procesno upravljanje u zatvorenoj petlji, mogućnosti podešavanja brzine i kontrola rada putem raznih interfejsa.

Frekventni pretvarač kontroliše zajedno izlaznu frekvenciju i napon, održavajući konstantan odnos napon/frekvencija (U/f). Moment koji se stvara je direktno srazmeran ovom odnosu, što znači da se može postići da je na svim brzinama (do nominalne brzine) moment konstantan i jednak je nominalnom momentu. Ovo znači da motor na svim brzinama može da isporuči pun moment.

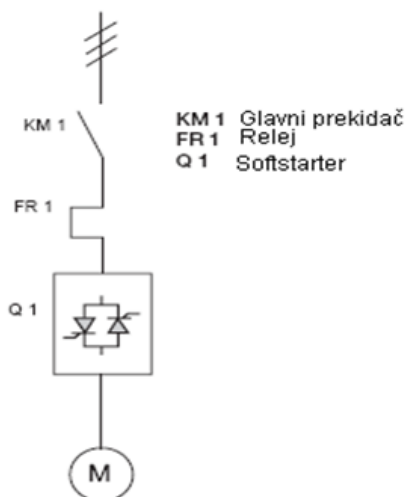
4.2.4 Start asinhronog motora pomoću soft-startera

Soft-start asinhronog motora u principu svodi se na postupak kontrolisanog povećanja statorskog napona, čime se praktično kontroliše statorska struja, a time i obrtni moment. Soft-starter upravlja uglom vođenja tiristora, pa na taj način menja efektivnu vrednost izlaznog napona.

Za rešavanje tehničkog problema kontrolisanog povećanja statorskog napona pri puštanju motora najčešće se koriste tiristorski podešavači napona. Uz pomoć odgovarajuće upravljačke elektronike moguće je vrlo jednostavno ograničiti polaznu struju, odnosno moment na apsolutni minimum vrednosti koja obezbeđuje potrebno ubrzanje, sve do granice termičke preopterećenosti motora.

5. REZULTATI MERENJA

Za verifikaciju rezultata merenja koje je izvršeno prilikom analize mreže je korišćen instrument MI 2492 PowerQ proizvođača METREL. Ovo je ručni multifunkcionalni instrument za analizu kvaliteta i merenje energetske efikasnosti mreže. Power View v2.0 je prateći RS softver koji omogućava najlakši način za preuzimanje, pregled, analizu i štampu merenih rezultata.



Slika 4. Soft-starter



Slika 5. METREL MI 2492 PowerQ

5.1. Analiza starta motora prilikom direktnog puštanja u rad

Snimanje parametara motora i njegov uticaj na mrežu prilikom direktnog starta motora, urađeni su na primeru motora koji pokreće pužni transporter koji doprema so iz mašine za pranje soli u mašinu za sušenje soli. Prema podacima proizvođača nominalna snaga ovog je 3kW, nominalna struja 7A i nominalni napon 400V.

Na slici 6 prikazan je dijagram napona i struje koju povlači neopterećen asinhroni motor. Vidi se da direktan start motora ne utiče previše na mrežu i ne uzrokuje veliki propad napona. Sa dijagrama se može videti, da prilikom starta propad napona iznosi svega 0,5 V što iznosi oko 0,12% nominalnog napona napajanja motora. Takođe sa dijagrama se vidi da u toku rada pogona, dolazi do većeg propada napona nego prilikom startovanja motora, što je uzrokovano radom nekih drugih potrošača u fabrici.



Slika 6. Napon i struja prilikom direktnog starta motora

Sa slike se može zaključiti da udarna stuja pri pokretanju motora iznosi oko 36A što je 5,14 puta veće od nominalne struje motora. Dobijena vrednost polazne struje je u granicama uobičajene vrednosti polazne struje asinhronih kaveznih motora. Nakon zaleta motor radi u režimu bliskom praznom hodu, jer u trenutku snimanja karakteristike nije bilo sirovine u pužnom transporteru. Vrednost struje koju motor povlači iz mreže u toku rada je manja od nominalne struje motora i iznosi oko 5,9A.

5.2. Analiza starta motora prilikom puštanja u rad pomoću frekventnog pretvarača

Frekventni pretvarač koji je korišćen u ovom slučaju osim za start koristi se i za regulaciju brzine obrtanja motora. Prema podacima proizvođača nominalna snaga ovog motora je 22kW, nominalna struja je 39,2A i nominalni napon je 400V. Prilikom upotrebe, pumpe se prilično često koriste bez kontrole brzine. U tom slučaju protok se reguliše sa ventilima ili prigušivanjem na druge načine, usled čega dolazi do bespotrebnog trošenja energije.

Snimanje parametara motora i njegov uticaj na mrežu prilikom starta pomoću frekventnog pretvarača, urađeni su na primeru motora koji pokreće centrifugalnu pumpu mašine za pranje soli.



Slika 7. Napon i struja prilikom puštanja u rad pomoću frekventnog pretvarača

Na slici 7. prikazan je dijagram napona i struje koju povlači asinhroni motor kojim se upravlja pomoću frekventnog pretvarača. Sa slike se vidi da je struja kontrolisana rampom i da je prevarač podešen tako da zalet motora traje 30 sekundi, dok je zaustavljanje motora programirano na 150 sekundi.

Maksimalna struja koju motor povlači prilikom starta je 51,9A što je oko 1,32 puta veće od nominalne struje motora, nakon čega se struja koju motor povlači iz mreže u toku normalnog rada ustalila na vrednosti oko 24A.

Sa slike takođe može da se zaključi da vrednost napona prilikom rada motora nije ustaljena već da malo osciluje blizu nominalne vrednosti. Primećujemo da prilikom pokretanja motora dolazi do malog propada napona od svega 1,3V koji je skoro i neprimetan u poređenju sa smetnjama u mreži koje nastaju usled rada drugih potrošača u fabrici.

5.3. Analiza starta motora prilikom puštanja u rad pomoću soft-startera

Snimanje parametara motora i njegov uticaj na mrežu prilikom starta pomoću soft-startera urađeni su na primeru motora koji pokreće centrifugalni ventilator u delu za sušenje soli.



Slika 8. Napon i struja prilikom puštanja u rad pomoću soft-startera

Prema podacima proizvođača nominalna snaga ovog motora je 22kW, nominalna struja je 39,2A i nominalni napon je 400V. Na slici 8. prikazan je dijagram napona i struje koju povlači asinhroni motor koji se pušta u rad pomoću soft-startera. Maksimalna struja koju motor povlači prilikom starta je 117A što je oko 3 puta veće od nominalne struje motora, nakon čega se struja koju motor povlači iz mreže u toku normalnog rada ustalila na vrednosti od oko 26A.

Sa dijagrama napona se primećuje da ovakvo pokretanje ne utiče značajno na mrežu. Dolazi do malog propada napona od svega 0,5V što je tek oko 0,12% nominalnog napona napajanja motora, što znači da ne nanosi velike smetnje mreži.

6. ZAKLJUČAK

Upotrebom energetske pretvarače i savremenih metoda upravljanja, postalo je moguće upravljati parametrima asinhronih motora u širokom opsegu, brzo i precizno kao DC motorima. Razvojem energetske elektronike upravljanje strujom, momentom i brzinom asinhronog motora je sada moguće, kako kod njihovog normalnog rada, tako i prilikom startovanja motora.

U ovom radu predstavljeni su tipovi rešenja za start asinhronih motora, koja su korišćena prilikom projektovanja i izgradnje fabrike soli. Analizom i merenjima je utvrđeno koliko opterećenja trpe pojedini asinhroni motori prilikom starta i koliko taj start utiče na električnu mrežu i rad samog pogona fabrike.

Na osnovu merenja koje je sprovedeno i rezultata zaključili smo da asinhroni motori, koji su izabrani za merenje ne nanose velike smetnje mreži, niti imaju negativan uticaj na rad ostalih potrošača u fabrici kao i na rad celokupnog pogona.

7. LITERATURA

- [1] Tripković Mihajilo, „Tipovi motora u savremenim električnim pogonima“, Diplomski rad, Visoka škola elektrotehnike i računarstva, Beograd, 2010
- [2] „Dinamička stanja asinhronog motora upravljanog energetskim pretvaračem napona i frekvencije ABB ACS-800“, Diplomski rad, Elektrotehnički fakultet Sarajevo, Sarajevo, 2009
- [3] <http://electrical-engineering-portal.com/direct-on-line-dol-motor-starter>
- [4] <http://electrical-engineering-portal.com/star-delta-motor-starter>
- [5] <http://www.momentum-automation.com/>
- [6] Madeško Mirsad, „Meki start asinhronih mašina“, Elektrotehnički fakultet Sarajevo, Sarajevo, 2008

Kratka biografija:

Sreten Petrović rođen je 1989. godine u Užicu, Republika Srbija. Upisao Fakultet Tehničkih Nauka u Novom Sadu 2008. godine i diplomirao 2014.godine

Đura Oros rođen je 1957. godine u Ruskom Krsturu, opština Kula, Republika Srbija. Doktorirao je na Fakultetu tehničkih nauka 2008. godine. Oblast interesovanja su električne mašine

MODELOVANJE NADZEMNOG VODA, KABLA I RL SEKCIJE**MODELING OF TRANSMISSION LINE, UNDERGROUND CABLE AND RL SECTION**

Tara Delari, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu su izvedeni matematički modeli nadzemnog voda, kabla i RL sekcije. Izvedeni modeli su korišćeni u implementaciji istoimenih komponenti u programskom jeziku Python. Novonastale komponente su postale deo standardne biblioteke komponenti Typhoon HIL Control Center-a, kompanije Typhoon HIL. Rad takođe obuhvata verifikaciju izvedenih modela i implementiranih komponenti, gde su referentne vrednosti za poređenje preuzete iz softverskog paketa MATLAB.

Ključne reči: nadzemni vod, kabl, RL sekcija, modelovanje

Abstract – In this paper, the mathematical models of transmission line, underground cable and RL section are derived. The derived models are used to implement appropriate components in Python programming language. The new components have become part of the standard library of components of Typhoon HIL Control Center, from Typhoon HIL company. The paper also includes the verification of the derived models and implemented components, using values from MATLAB software pack as reference.

Key words: transmission line, underground cable, RL section, modeling

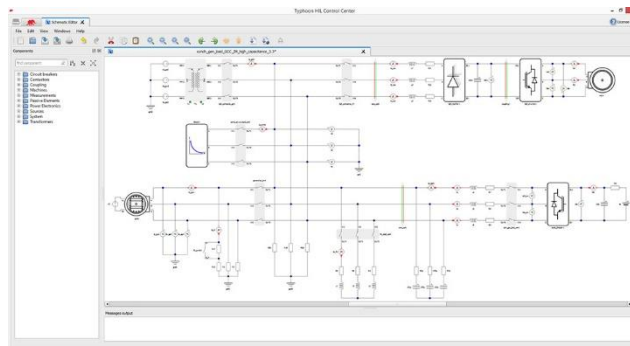
1. UVOD

Pod prenosom električne energije smatra se transport električne energije između proizvođača (elektrana ili distribuirani generator) i centara potrošnje (elektrodistribucije, industrijski potrošači, domaćinstva itd.).

Električna energija se može prenositi nadzemnim vodovima ili podzemnim kablovima. U realnom elektroenergetskom sistemu, prenos električne energije na visokim naponima se obavlja nadzemnim vodovima dok se na nivou distribucije prenos obavlja podzemnim kablovima. Ovo je posledica činjenice da je cena izgradnje nadzemnog voda jeftinija od odgovarajućeg podzemnog kabla. Podzemni kablovi se dakle koriste tamo gde estetski nije prihvatljivo izgraditi nadzemne vodove (naselja, gradovi itd.). Cilj ovog rada jeste da prikaže modele i implementaciju nadzemnog voda i podzemnog kabla koji se primenjuje u simulatoru Typhoon HIL.

Pod ovim se podrazumeva kreiranje matematičkog modela nadzemnog voda, podzemnog kabla i RL sekcije, koja uz odgovarajuće pretpostavke predstavlja pojednostavljeni model vodova i kablova. Pomoću ovih matematičkih

modela se dalje vrši implementacija modela u programskom jeziku Python u integrisanom razvojnom okruženju PyCharm. Finalni proizvod su tri komponente koje se nalaze u biblioteci komponenta Schematic Editor-a Typhoon HIL Control Center-a. Korisnici Typhoon HIL simulatora mogu da koriste ove komponente prilikom pravljenja modela elektroenergetskog sistema u Schematic Editor-u, za koji se testira kontroler koji će biti integrisan u pravi elektroenergetski sistem. Prikaz Schematic Editor-a Typhoon HIL Control Center-a dat je na slici 1.1.

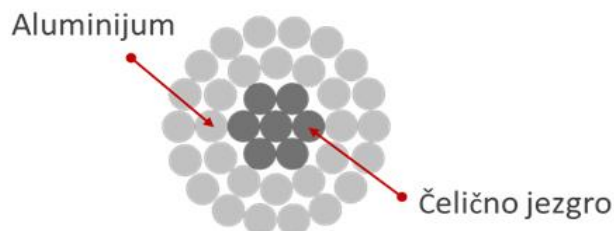


Slika 1.1: Schematic Editor Typhoon HIL Control Center-a

2. ANALIZA ULAZNIH PODATA MODELA

Uz podatke o prostornoj raspodeli provodnika (geometrija sistema), kataloški podaci će dalje biti korišćeni kao ulazni podaci za modelovanje novih komponenti, te ih je potrebno objasniti.

Najčešći tip provodnika koji se upotrebljava je „alučel provodnik“ (engl. ACSR). Izgled poprečnog preseka alučel provodnika dat je na slici 2.1.



Slika 2.1: Poprečni presek alučel provodnika

U tabeli 2.1 prikazan je deo kataloga podataka provodnika, koji je preuzet iz [1], sa informacijama o provodnicima koji se koriste za nadzemne vodove. Katalog podataka se dobija od proizvođača provodnika i on sadrži sve bitne informacije o karakteristikama provodnika. Iz prikazane tabele će se koristiti prečnik, podužna otpornost i GMR kao ulazni podaci za modelovanje, gde je sa GMR obeležen srednji geometrijski prečnik provodnika. GMR po definiciji određuje sopstvenu induktivnost provodnika i on

NAPOMENA:

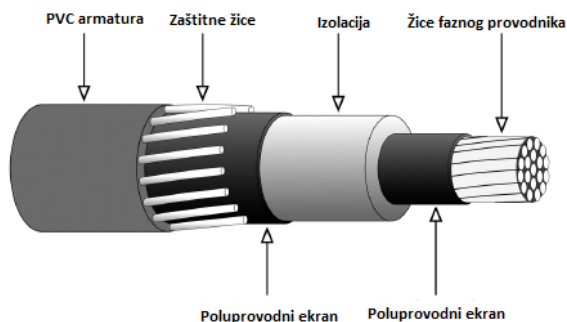
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Grabić, docent.

predstavlja radijus fiktivnog provodnika koji ima istu vrednost sopstvene induktanse kao i pravi provodnik poluprečnika r .

Tabela 2.1: *Primer kataloga sa podacima o provodnicima*

Veličina	Upetljanost	Materijal	Prečnik (in.)	GMR (stopa)	Podužna otpornost (Ω/milji)	Kapacitet (A)
2	6/1	ACSR	0.316	0.00418	1.69	180
10	SLD	Bakar	0.1019	0.00330	5.9026	75

U slučaju podzemnih kablova, modelovan tip je koncentrični ili „nn” tip provodnika kod kojih je nulti, odnosno zaštitni provodnik, postavljen koncentrično preko izolacije kod jedno-žičnih kablova ili preko jezgra kod više-žičnih kablova. Na slici 2.2 dat je presek dela konstrukcije razmatranog visokonaponskog kabla. Tabela kataloških podataka neće biti prikazana jer će biti korišćene samo informacije o broju i veličini bakarnih žica kao i spoljašnji prečnik provodnika. Podaci o faznom provodniku i zaštitnim žicama se dobijaju iz tabele 2.1, kao u slučaju nadzemnih vodova.



Slika 2.2: *Presek dela konstrukcije visokonaponskog kabla*

3. MODELOVANJE NADZEMNIH VODOVA, KABLOVA I RL SEKCIJE

Određivanje redne impedanse i otočne admitanse kablova i nadzemnih vodova je ključan korak pre bilo kakve analize prenosnog ili distributivnog sistema. Rednu impedansu čine rezistansa provodnika i sopstvena i međusobna induktivna reaktansa provodnika koja je rezultat magnetskog polja provodnika, dok otočnu admitansu čini kapacitivna susceptansa, koja je rezultat razlike potencijala provodnika.

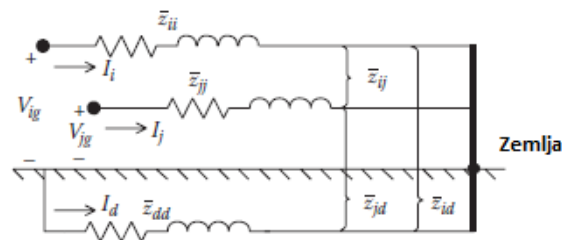
Sve jednačine i primeri iz ove glave su preuzete iz [1]. Induktansa provodnika i se sastoji od sopstvene induktivnosti provodnika i i međusobne induktivnosti provodnika i i ostalih provodnika. Jednačine za proračun sopstvene induktivnosti i međusobne induktivnosti dati su izrazima 3.1 i 3.2, respektivno.

$$L_{ii} = \frac{\lambda_{ii}}{I_i} = 2 \cdot 10^{-7} \cdot \ln \frac{1}{GMR_i} \text{ H/m} \quad (3.1)$$

$$L_{in} = \frac{\lambda_{in}}{I_n} = 2 \cdot 10^{-7} \cdot \ln \frac{1}{D_{in}} \text{ H/m} \quad (3.2)$$

Džon Karson je 1926. godine objavio rad u kojem se prikazuje postupak računanja reaktanse nadzemnog voda uzimajući u obzir povratni put kroz zemlju [2]. On je predstavio nadzemni vod provodnicima koji su sa jedne strane priključeni na naponski izvor, a sa druge strane su uzemljeni. Ovo je ilustrovano na slici 3.1 na kojoj se može videti nadzemni vod sačinjen od dva provodnika (i i j), kroz

koje prolaze struje (I_i i I_j), sa uzemljenim krajevima. Povratni put kroz zemlju je prikazan fiktivnim provodnikom kroz koji prolazi struja I_d .



Slika 3.1: *Dva provodnika sa povratnim putem kroz zemlju*

Primenom Kirhofovih zakona na šemu sa slike 3.1 dobija se jednačina:

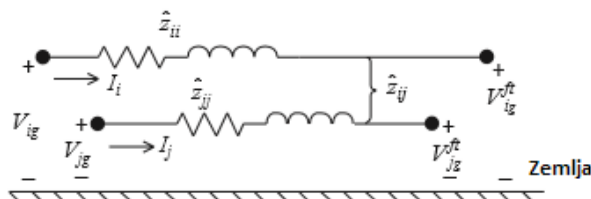
$$V_{ig} = \hat{z}_{ii} \cdot I_i + \hat{z}_{ij} \cdot I_j, \quad (3.3)$$

gde je

$$\hat{z}_{ii} = \bar{z}_{ii} + \bar{z}_{dd} - \bar{z}_{di} - \bar{z}_{id} \quad (3.4)$$

$$\hat{z}_{ij} = \bar{z}_{ij} + \bar{z}_{dd} - \bar{z}_{dj} - \bar{z}_{id} \quad (3.5)$$

Na slici 3.2 je dato ekvivalentno primitivno kolo, kola sa slike 3.1, koje je opisano jednačinama 3.4 i 3.5. U ovim jednačinama je efekat zemlje iskazan pomoću takozvane primitivne sopstvene i međusobne impedanse voda.



Slika 4.2: *Ekvivalentno primitivno kolo*

Ako se jednačina 3.4 proširi dobija se jednačina:

$$\hat{z}_{ii} = r_i + jx_{ii} + r_d + jx_{da} - jx_{dn} - jx_{na} \quad (3.6)$$

Uvrštavanjem jednačina 3.1 i 3.2 u jednačinu 3.6 dobija se:

$$\hat{z}_{ii} = r_d + r_i + j0.002 \cdot f \cdot \left(\ln \frac{1}{GMR_i} + \ln \frac{D_{id} \cdot D_{di}}{GMR_d} \right) \quad (3.7)$$

Izraz za $\hat{z}_{ij}$ se izvodi analognim postupkom. Očigledan problem pri korišćenju jednačina 3.7 jeste što se ne zna vrednost podužne otpornosti zemlje (r_d), srednji geometrijski prečnik zemlje (GMR_d) i udaljenost provodnika od zemlje (D_{id}, D_{di}). Ove vrednosti se mogu dobiti primenom Karsonovih obrazaca. Karson je u svom radu pretpostavio da je zemlja beskonačan provodnik sa konstantnom vrednošću podužne otpornosti. Koristio je teoriju slika, za svaki provodnik iznad zemlje postoji slika provodnika na istoj udaljenosti ispod zemlje. Primenom modifikovanih Karsonovih obrazaca dobijaju se konačne jednačine koje se koriste u implementaciji komponenti:

$$\hat{z}_{ii} = r_i + 0.095 + j0.121 \left(\ln \frac{1}{GMR_i} + 7.934 \right) \quad (3.8)$$

$$\hat{z}_{ij} = 0.095 + j0.121 \left(\ln \frac{1}{D_{ij}} + 7.934 \right) \quad (3.9)$$

U jednačinama 3.8 i 3.9 se može primetiti da vrednosti podužne otpornosti zemlje, srednjeg geometrijskog prečnika zemlje i udaljenost provodnika od zemlje nisu više nepoznate. Modifikovani Karsonovi obrasci se koriste za modelovanje podzemnih kablova kao što su primenjeni za nadzemne vodove. Pošto je tip kabla koji se posmatra u ovom radu tip sa koncentričnim neutralnim provodnikom, koji ima fazni provodnik i provodnik zaštite, razlikovaće se samo dimenzije primitivne matrice. Otočna admitansa se sastoji od konduktanse i kapacitivne susceptanse. Konduktansa se najčešće zanemaruje pošto je mala u poređenju sa susceptansom. Kapacitivnost voda i kabla je rezultat razlike potencijala između provodnika. Prilikom proračuna otočne kapacitivnosti voda takođe se koristi teorija slika. Jednačina 3.10 daje pad napona između provodnika i i njegove slike. Pad napona između provodnika i i zemlje će tada biti polovina proračunatog napona u jednačini 3.10:

$$V_{ig} = \frac{1}{2\pi\epsilon} \left(q_i \ln \frac{S_{ii}}{RD_i} + q_j \ln \frac{S_{ij}}{D_{ij}} \right) \quad (3.10)$$

Jednačina 3.10 može biti napisana i u sledećoj formi, gde su $\hat{P}_{ii}$ i $\hat{P}_{ij}$ sopstveni i međusobni potencijalni koeficijenti:

$$V_{ig} = \hat{P}_{ii} \cdot q_i + \hat{P}_{ij} \cdot q_j \quad (3.11)$$

Za nadzemni vod od n provodnika, konstruiše se primitivna matrica koeficijenata potencijala koja je dimenzije $n \times n$ i oblika je:

$$\hat{\mathbf{P}}_{\text{primitivna}} = \begin{bmatrix} [\hat{\mathbf{P}}_{ij}] & [\hat{\mathbf{P}}_{in}] \\ [\hat{\mathbf{P}}_{nj}] & [\hat{\mathbf{P}}_{nn}] \end{bmatrix} \quad (3.12)$$

Pošto je neutralni provodnik uzemljen može se koristiti metoda Kronove redukcije [3] da se dobije matrica koeficijenata potencija u faznom domenu, čija je inverzna matrica matrica kapacitivnosti u faznom domenu:

$$\mathbf{P}_{abc} = \hat{\mathbf{P}}_{ij} - \hat{\mathbf{P}}_{in} \cdot \hat{\mathbf{P}}_{nn}^{-1} \cdot \hat{\mathbf{P}}_{nj} \quad (3.13)$$

$$\mathbf{C}_{abc} = \mathbf{P}_{abc}^{-1} \quad (3.14)$$

Zanemarujući otočnu konduktansu, matrica otočne admitanse u faznom domenu jednaka je:

$$\mathbf{Y}_{abc} = 0 + j \cdot \omega \cdot \mathbf{C}_{abc} \quad (3.15)$$

U slučaju kabla, koncentrična neutralna zaštita je uzemljena, tako da su sve žice zaštite na istom potencijalu. Zbog petljanja žica, pretpostavlja se da je električno polje koje stvara naelektrisanje faznog provodnika ograničeno na neutralan provodnik. Pošto su sve žice neutralnog provodnika na istom potencijalu, potrebno je odrediti samo razliku potencijala između faznog provodnika p i jedne žice. Detaljan postupak izvođenja jednačine otočne admitanse kabla dat je u [1], ovde će samo biti napisana krajnja jednačina:

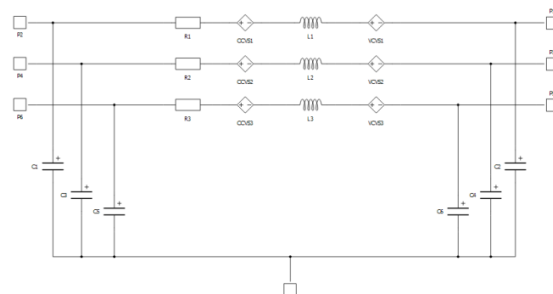
$$y_{ag} = j\omega \frac{2\pi\epsilon_0 \cdot 2.3}{\ln \frac{R_b}{RD_c} - \frac{1}{k} \ln \frac{k \cdot RD_s}{R_b}} \quad (3.10)$$

Ovim su u potpunosti određeni parametri PI šeme koja se koristi za modelovanje kabla i nadzemnog voda. Prilikom modelovanja RL sekcije uvedena je pretpostavka da faze nisu spregnute, to jest da struja u jednoj fazi ne utiče na bilo koji način na druge faze komponente i da ne postoji kapacitivna sprega između komponente i zemlje. Pored toga, uvedena je pretpostavka da sve faze imaju iste karakteristike, što znači da su im jednake podužne otpornosti i podužne induktivnosti. Ove dve podužne vrednosti, pored dužine sekcije, predstavljaju ulazne podatke za modelovanje.

4. IMPLEMENTACIJA KOMPONENTI

Implementacija komponenti RL sekcija, nadzemni vod i podzemni kabel rađena je u programskom jeziku Python, u integrisanom razvojnom okruženju PyCharm kompanije JetBrains. Sve komponente u biblioteci Typhoon HIL Schematic Editor-a su ili atomske ili kompozitne. Atomske komponente predstavljaju osnovne komponente koje nisu sačinjene ni od jedne druge komponente, dok su kompozitne komponente sačinjene od jedne ili više atomskih komponenta. Sve implementirane komponente su u ovom radu modelovane kao kompozitne komponente.

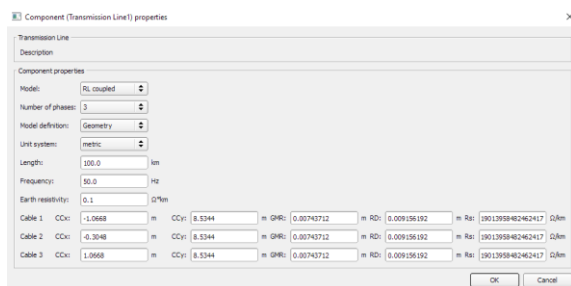
Na slici 4.1 se može videti model trofaznog nadzemnog voda sa koji je implementiran, ukoliko je odabran tip modela PI. Na slici su strujno kontrolisani naponski izvori obeleženi sa CCVSi, a naponski kontrolisani naponski izvori sa VCVSi, gde i predstavlja broj grane. Korisnik može da bira između 2, 3 ili 4 faze, na osnovu kojih se dinamički menja struktura modela, to jest broj grana modela, a samim tim ukupnog broja otpornika, kalema i kontrolisanih naponskih izvora.



Slika 4.1: PI šema trofaznog nadzemnog voda

Na slici 4.2 se može videti izgled prozora prilikom geometrijskog definisanja komponente nadzemni vod. Interna struktura komponenti se dinamički menja na promenu odgovarajućih ulaznih parametara. Da bi ovo bilo moguće napisane su funkcije koje menjaju model. Korisnik tako ima mogućnost da menja broj faza, tip ulaznih parametara, da li su parametri zadati u metričkom ili anglosaksonskom sistemu mernih jedinica itd.

Interna struktura kabla je ista kao i za slučaj nadzemnog voda, dok je struktura komponenta RL sekcija jednostavna i sastoji od na red vezanih otpornosti i induktivnosti (ne postoje zavisni izvori pošto je pretpostavljeno da nema sprega između faza).



Slika 4.2: Prikaz prozora koji se pojavljuje prilikom definisanja komponente geometrijskim parametrima

5. PROVERA RADA MODELA

Za proveru rada modela nadzemnog voda i kabla su se koristile Matlabove ugrađene funkcije `power_lineparam` [4] i `power_cableparam` [5], gde se prva koristi za proveru modela nadzemnog voda, a druga modela podzemnog kabla. Tačnije, za iste unete podatke u odgovarajuću funkciju i komponentu upoređivaće se matrice R , L i C u faznom domenu. U oba slučaja se posmatrao trofazni sistem modelovan PI šemom. U tabeli 5.1 data su izračunata odstupanja za slučaj nadzemnog voda, dok su u tabeli 5.2 data odstupanja modela kabla.

Tabela 5.1: Odstupanje izračunatih vrednosti elemenata matrica od vrednosti izračunatih pomoću funkcije `power_lineparam`

MATRICA OTPORNOSTI					
ΔR_{11}	$= -0,0013$	ΔR_{12}	$= -0,0013$	ΔR_{13}	$= -0,0013$
ΔR_{21}	$= -0,0013$	ΔR_{22}	$= -0,0013$	ΔR_{23}	$= -0,0013$
ΔR_{31}	$= -0,0013$	ΔR_{32}	$= -0,0013$	ΔR_{33}	$= -0,0013$
MATRICA INDUKTIVNOSTI					
ΔL_{11}	$= 2,3372e-4$	ΔL_{12}	$= 2,3381e-4$	ΔL_{13}	$= 2,3374e-4$
ΔL_{21}	$= 2,3381e-4$	ΔL_{22}	$= 2,3372e-4$	ΔL_{23}	$= 2,3377e-4$
ΔL_{31}	$= 2,3374e-4$	ΔL_{32}	$= 2,3377e-4$	ΔL_{33}	$= 2,3372e-4$
MATRICA KAPACITIVNOSTI					
ΔC_{11}	$= -1,0542e-9$	ΔC_{12}	$= -6,1555e-10$	ΔC_{13}	$= -1,9907e-10$
ΔC_{21}	$= -6,1555e-10$	ΔC_{22}	$= -1,1542e-9$	ΔC_{23}	$= -3,8980e-10$
ΔC_{31}	$= -1,9907e-10$	ΔC_{32}	$= -3,8980e-10$	ΔC_{33}	$= -8,9054e-10$

Tabela 5.2: Odstupanje izračunatih vrednosti elemenata matrica od vrednosti izračunatih pomoću funkcije `power_cableparam`

MATRICA OTPORNOSTI	
$\Delta R_{11} = 0,000004$	$\Delta R_{12} = 0,0000003$
$\Delta R_{21} = 0,0000003$	$\Delta R_{22} = 0,0000008$
MATRICA INDUKTIVNOSTI	
$\Delta L_{11} = 0,00013$	$\Delta L_{12} = -6,67e-6$
$\Delta L_{21} = -6,67e-6$	$\Delta L_{22} = -0,000016$
$\Delta L_{12} = -7,32e-7$	$\Delta L_{23} = -7,32e-7$
	$\Delta L_{33} = 0,00013$
MATRICA KAPACITIVNOSTI	
$\Delta C_{11} = -9,28e-8$	

Najveće proračunato odstupanje u tabelama 5.1 i 5.2 je 0,00013, što predstavlja veoma malo odstupanje. Nakon poređenja odstupanja komponenti odgovarajućih funkcija Matlaba dolazi se do zaključka da su modeli i implementirane komponente ispravni.

6. ZAKLJUČAK

U ovom radu je predstavljen proces modelovanja, implementacije i testiranja razvijenih komponenti nadzemni vod, kabel i RL sekcija. Odstupanje koje je primećeno prilikom testiranja je veoma malo, te se može konstatovati da su sve komponente uspešno implementirane. One su nakon dodatnih testiranja u kompaniji ušle u standardnu biblioteku komponenti i otvorene korisnicima za korišćenje.

Implementacija komponenti nadzemni vod i kabel koji su modelovani sa distribuiranim parametrima predstavljaju predloge autora za poboljšanja.

7. LITERATURA

- [1] William H. Kersting, “*Distribution Systems Modeling and Analysis*”, CRC Press, 2012.
- [2] John R. Carson, “Wave Propagation in Overhead Wires with Ground Return”, *The Bell System Technical Journal*, Volume 5, Issue 4, Okt. 1926.
- [3] Gabriel Kron, “Tensorial Analysis of Integrated Transmission Systems Part I. The Six Basic Reference Frames”, *Transactions of the American Institute of Electrical Engineers*, Volume 70, Issue 2, pp. 1239-1248, Jul 1951.
- [4] https://www.mathworks.com/help/physmod/sps/power_sys/ref/power_lineparam.html
- [5] https://www.mathworks.com/help/physmod/sps/power_sys/ref/power_cableparam.html

Kratka biografija:



Tara Delari rođena je 14.5.1991. u Novom Sadu, Srbija. Osnovnu i srednju školu završila je u Novom Sadu. Osnovne akademske studije završila je na Fakultetu tehničkih nauka 2014. god. i 2015. god. upisala master akademske studije na istom fakultetu.

**MPEG-DASH STANDARD ZA DINAMIČKI PRILAGODLJIVI PRENOS VIDEO
SADRŽAJA PREKO HTTP-A****MPEG-DASH STANDARD FOR DYNAMIC ADAPTIVE STREAMING OF VIDEO
CONTENT OVER HTTP**

Stevan Cvijanović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu opisan je MPEG-DASH standard, jedino prilagodljivo streaming rešenje koje je postalo međunarodni standard. Akcenat je stavljen na samu strukturu MPEG-DASH-a i njegovog MPD (Media Presentation Description) dokumenta koji sadrži sve potrebne informacije koje će omogućiti kvalitetno iskustvo klijenta u gledanju video sadržaja. Takođe ćemo govoriti o glavnim prednostima MPEG-DASH standarda naspram ostalih sličnih rešenja i prikazaćemo primer njegove serverske i klijentske realizacije.

Abstract – In this paper, we will discuss MPEG-DASH standard, the only adaptive streaming solution that became an international standard. The emphasis is placed on the very structure of MPEG-DASH and its MPD (Media Presentation Description) document that contains all the necessary information that will enable the client's quality experience in viewing video content. We will also talk about the main advantages of the MPEG-DASH standard against other similar solutions and we will show the example of its server and client implementation.

Ključne reči: MPEG-DASH, streaming, video, MPD, HTTP, medija, kodiranje

1. UVOD

Još tokom Olimpijskih igara 2008. godine, NBC je prijavio promet od 3,4 petabajta video sadržaja preko interneta. Potencijal streaming-a multimedijalnog sadržaja preko interneta je samo rastao od tada.

Jedan od problema koji je stajao na putu još većoj ekspanziji ovakvog prometa jeste nedostatak kompatibilnosti između uređaja i različitih proizvođača, jer je svaka komercijalna platforma bila zatvoren sistem i imala svoj vlastiti manifest, format i protokol.

Rešenje takvog problema bi bilo uvođenje standarda koji obezbeđuje kompatibilnost između raznih servera i uređaja, i koje će doprineti rastu tržišta i omogućiti sadržaju i servisima da snabevaju širok spektar uređaja kao što su računari, televizori, laptopovi, konzole, tableti, mobilni telefoni. To rešenje je upravo bio MPEG-DASH [1].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dejan Vukobratović.

2. DASH

MPEG-DASH, poznat i kao *DASH* (Dynamic Adaptive Streaming over HTTP), jeste tehnika prilagodljivog prenosa multimedijalnog sadržaja visokog kvaliteta putem interneta preko HTTP veb servera. MPEG-DASH radi tako što deli sadržaj u manje segmente. Svaki segment sadrži kratak interval reprodukcije sadržaja koji može da bude trajanja od više sati, kao što su filmovi ili prenos sportskih manifestacija. Sadržaj je dostupan u više različitih brzina prenosa, tako da su i manji segmenti dostupni u alternativnim brzinama. Dok se sadržaj reprodukuje na MPEG-DASH klijentu, klijent automatski bira odgovarajući segment koji će da preuzme na osnovu trenutnog stanja mreže (internet protoka). Klijent bira segment sa najvećom mogućom brzinom prenosa koji se može preuzeti bez prouzrokovanja zadržavanja i baferovanja (engl. *buffering*), tako da se MPEG-DASH klijent može neprimetno prilagoditi promenama stanja mreže i može da pruži visoku kvalitetnu reprodukciju sa što manje zadržavanja i baferovanja [2].

Streaming audio i video sadržaja danas zauzima preko 70% od ukupnog internet saobraćaja u periodima najvećeg opterećenja. Preko 50% internet saobraćaja zauzimaju samo Netflix (približno 35%) i Youtube (približno 17%) servisi. Oba ova servisa koriste MPEG-DASH, i jedni su od glavnih zaslužnih za ubrzanje razvoja ovog standarda. Sa ova dva velika izvora internet saobraćaja, preko 50% on ukupnog internet saobraćaja je MPEG-DASH [3].

2.1. Standardizacija

MPEG-DASH tehnologija je razvijena od strane MPEG-a (Moving Picture Experts Group). Rad na DASH-u je počeo 2010. godine, postao je DIS (Draft International Standard) u januaru 2011. godine i međunarodni standard u novembru 2011. god. DASH je nezavisan od platforme i tipa kodeka koji platforma podržava. DASH Industry Forum (DASH-IF) je imao značajnu ulogu u usvajanju MPEG-DASH-a. Sastoji se od velikih medijskih kompanija kao što su Microsoft, Netflix, Google, Ericsson, Samsung, Adobe itd. i kreira smernice kako da se DASH koristi u različitim slučajevima. MPEG-DASH je zvanično potvrđen 2012. godine [4].

2.2. Druge prednosti

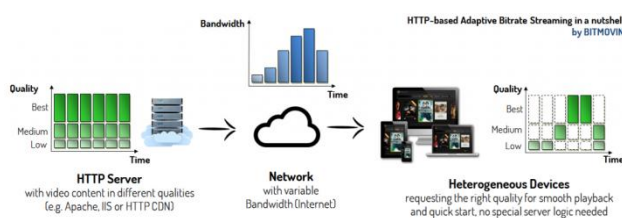
Ključne stvari i prednosti MPEG-DASH-a [4]:

- Smanjenje kašnjenja pri startovanju i zastajanja (baferovanja) tokom video prenosa.
- Kontinualna prilagodljivost stanju mreže (protoku) klijenta.
- Klijentski orijentisana streaming logika koja omogućava visoku skalabilnost i fleksibilnost.

- Korišćenje postojećih i isplativih HTTP baziranih CDN-ova, proksija i keša.
- Efikasno zaobilazanje preslikavanja mrežne adrese (engl. Network address translation; NAT) i zaštitnog zida korištenjem HTTP-a.
- Jednostavno spajanje i umetanje reklama.
- Podrška za efikasan trik mod (eng. trick mode ili trick play) - mogućnost premotavanja ili ubrzavanja videa.
- I mnogo toga drugog.

2.3. MPEG-DASH ukratko

Osnovna ideja MPEG-DASH je sledeća - isekcati medija datoteku u segmente koji mogu biti kodirani sa različitim brzinom prenosa ili rezolucijom. Segmenti su dostupni na veb serveru i mogu da se preuzmu preko HTTP GET zahteva gde HTTP server obezbeđuje tri različita kvaliteta u segmentima jednakih dužina kao što je prikazano na slici 1. Prilagođavanje brzini prenosa ili rezoluciji se vrši na klijentskoj strani za svaki segment [4].

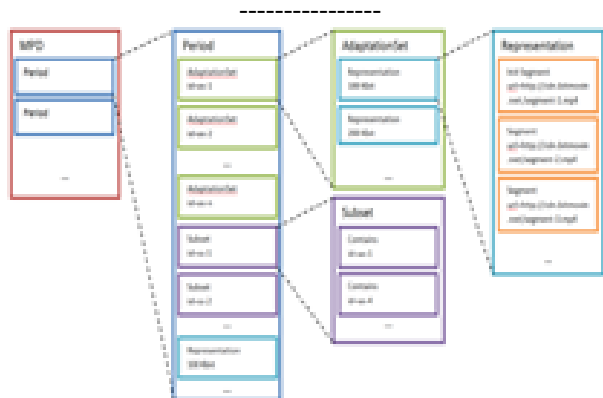


lika 1. Prikaz komunikacije servera i klijenta u razmeni sadržaja različitog kvaliteta

Da bismo opisali vremensku i strukturalnu vezu između segmenata, MPEG-DASH je predstavio MPD dokument. MPD je XML (Extensible Markup Language) dokument koji predstavlja različite kvalitete medija sadržaja i pojedinačne segmente različitih kvaliteta sa HTTP URL-om (Uniform Resource Locators). Ovakva struktura povezuje segmente različitih brzina prenosa ili rezolucija. Svaki klijent će prvo zahtevati MPD koji sadrži informacije o medija sadržaju i na osnovu tih informacija će zahtevati određeni segment koji najbolje odgovara zahtevima [4].

3. MEDIA PRESENTATION DESCRIPTION (MPD)

MPEG-DASH MPD je XML dokument koji sadrži informacije o medija segmentima, njihovim vezama i druge metapodatke koji su potrebni klijentu. U ovom delu su opisani najvažniji delovi MPD dokumenta, počevši od najvišeg nivoa (Periodi) do najnižeg (Segmenti), kao na slici 2 [6].



Slika 2. Prikaz strukture MPD dokumenta

3.1. Periodi (<Period>)

Svaki MPD može da sadrži jedan ili više period. Period, najviši MPD element, opisuje deo sadržaja sa startnim vremenom i trajanjem. Svaki od tih perioda sadrži medijske komponente kao što su video komponente za na primer drugačije uglove gledanja ili drugačije kodeke, audio komponente sa različitim jezicima ili različitim vrstama informacija, prevod, dodatni tekst. Ovakve komponente imaju određene karakteristike kao što je brzina prenosa, broj slika u sekundi, audio kanali itd., koji se ne menjaju tokom jednog perioda. Period može razdvojiti sadržaj, npr. za umetanje reklama, promenu ugla kamere u utakmicama uživo itd. Pre i posle ovog perioda, postojali bi drugi periodi koji bi sadržavali pravi sadržaj (npr. film) u više brzina prenosa i rezolucija od standardne do visoke [4].

3.2. Adaptacioni setovi (<AdaptationSet>)

Tipično, medija sadržaj kao što je video, zvuk, prevod itd. su organizovani u adaptacione setove. Svaki period može da sadrži jedan ili više adaptacionih setova koji omogućavaju grupisanje različitih multimedijalnih komponenti koji logički pripadaju zajedno [4].

U najjednostavnijem slučaju, Period može imati jedan AdaptationSet koji sadrži čitav audio i video sadržaj, ali da bi se smanjio protok, svaki tok može biti podeljen u različite adaptacione setove. Čest slučaj je da imamo jedan video adaptacioni set i više audio adaptacionih setova (svaki za različiti jezik) [6].

3.3. Reprezentacije (<Representation>)

Adaptacioni set se sastoji od setova reprezentacija koje sadrže izmenjive verzije odgovarajućeg sadržaja, npr. različite rezolucije, brzine prenosa itd. Reprezentacije dozvoljavaju da adaptacioni setovi imaju isti sadržaj ali različito kodiran. Više reprezentacija daje klijentu mogućnost da prilagodi prenos trenutnom stanju mreže i protoku i tako zagaruje usklađen prikaz. Ovo omogućuje korisniku da zatraži najkvalitetniji sadržaj koji može gledati bez ikakvog zastajanja (baferovanja) i bez trošenja protoka na nepotrebne. Reprezentacije takođe mogu biti različito kodirane kako bi omogućile podršku za klijente koji podržavaju neku drugu vrstu kodeka. Reprezentacije se uglavnom biraju automatski, ali neki programi dozvoljavaju korisniku da pregazi te izbore (naročito rezoluciju). Korisnik može da izabere reprezentaciju po želji ako ne želi da troši protok na određeni video (možda mu je stalo samo do zvuka), ili želi da pauzira video i baferuje u zamenu za veći kvalitet videa [6].

3.4. Segmenti

Reprezentacije su isekcane u segmente. Medija segmenti predstavljaju stvarne medijske datoteke koje DASH klijent reprodukuje, puštajući ih jednu za drugom kao da se radi o jednoj datoteci. Stvarna lokacija tih datoteka može biti opisana putem BaseURL elementa za jedno-segmentnu reprezentaciju, liste segmenata sa SegmentList elementom ili putem šablona sa SegmentTemplate elementom. Informacije koje se odnose na sve segmente mogu se pronaći u SegmentBase elementu. Početak i trajanje segmenata mogu biti opisani SegmentTimeline elementom. Segmenti takođe mogu biti u posebnim datotekama (često kod prenosa uživo) ili mogu biti zapisani u rasponima bajtova unutar jedne datoteke [6].

U kontaktu sa drugim sistemima, MPEG-DASH ne ograničava dužinu segmenata niti predlaže optimalnu dužinu. To se odlučuje na osnovu datog slučaja, npr. duži segmenti omogućavaju efikasniju kompresiju i manje mrežno preopterećenje, jer će se svaki segment zahtevati preko HTTP-a i sa svakim zahtevom određena količina HTTP preopterećenja će biti prisutna. Suprotno tome, kraći segmenti se koriste za slučaje uživo prenosa kao i za visoko promenljivo stanje protoka kao kod mobilnih mreža i tu se omogućava brže i fleksibilnije prebacivanje između pojedinačnih segmenata [4].

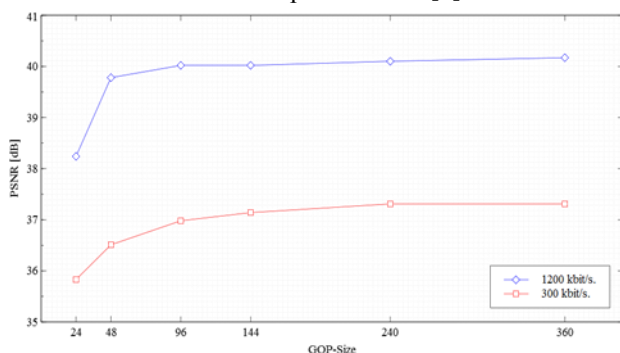
4. OPTIMALNA DUŽINA MPEG-DASH SEGMENTA

Pitanje optimalne dužine segmenta nije lako odgovoriti i zavisi od okruženja, sadržaja i veb servera (trajne konekcije, engl. persistent connections) [7].

4.1. Procena dužine segmenata na osnovu efikasnosti kodiranja i kvaliteta

Da bi se omogućilo neometano prebacivanje između različitih reprezentacija kvaliteta DASH formata, I-frejmi (eng. I-frame, keyframe) interval se mora ravnomerno podeliti po dužini segmenta kako bi osigurao I-frejmi na početku svakog segmenta. Što je duži segment, duži je i potencijalni I-frejmi interval. Duži I-frejmi interval obično daje bolji video kvalitet jer I-frejmovi koji nisu referencirani na bilo koji drugi frejmi su najmanje efikasni frejmovi. Izbor kraćeg segmenta ima manu zbog velikog konačnog broja segmenata jer će biti potrebno više I-frejmovi da bi bilo zagarantovano prebacivanje između reprezentacija. Ovo dovodi do niže efikasnosti kodiranja [7].

Kraći segmenti mogu uticati na ukupan kvalitet sadržaja sa do 1,5 dB PSNR vrednosti. Ipak, uticaj ovog efekta je znatno smanjen sa povećanjem dužine segmenta. Kao što je prikazano na sledećoj slici, segmenti kraći od dve sekunde utiču veoma loše na performanse [7].

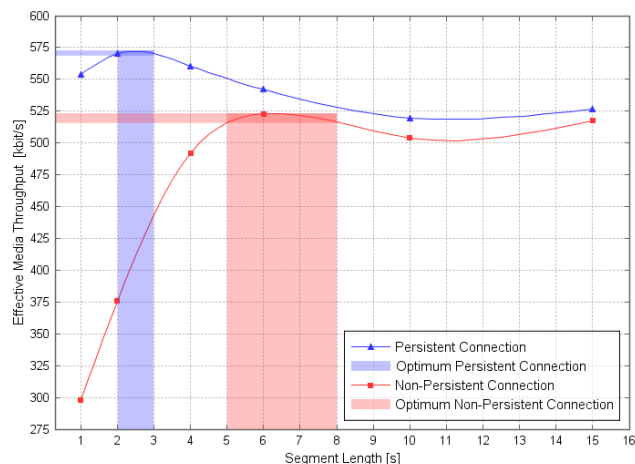


Slika 3. Grafički prikaz uticaja dužine segmenata na kvalitet sadržaja.

4.2. Procena dužine segmenata na osnovu striming preformansi i konfiguracije veb servera

Sa perspektive mreže i interneta, takođe postoji dosta uticajnih faktora koji moraju da se uzmu u obzir. Na primer, duži segmenti mogu da izazovu zastajanje kod bežičnih internet konekcija sa čestim promenama protoka, ali kratki segmenti mogu izazvati loše striming preformanse zbog preopterećenja izazvanih čestim zahtevima ka serveru i uticajem kašnjenja mreže. Prikazaćemo istraživanje Bitmovin organizacije. Korišten

je standardni HTTP veb server za omogućavanje trajne HTTP 1.1 konekcije i ne-trajne HTTP 1.0 konekcije. Analizirani su segmenti dužina 1, 2, 4, 6, 10 i 15 sekundi i interpolirani su u grafikon koji prikazuje njihove performanse u smislu efektivnog medijskog protoka. Kako je prikazano na sledećoj slici, optimalna dužina segmenta za ove postavke mreže bi bila između 2 i 3 sekunde ako se koriste HTTP 1.1 trajne konekcije ili između 5 i 8 sekundi ako se koristi HTTP 1.0. Efektivni medijski protok se ne poboljšava kada se povećava dužina segmenta zbog toga što se dostupni protok menja kroz vreme [7].



Slika 4. Grafički prikaz efektivnog medijskog protoka za segmente različite dužine

5. PRIMER SERVERSKOG REŠENJA ZA KODIRANJE VIDEO SADRŽAJA – BITMOVIN

Kao primer jednog rešenja kodiranja video sadržaja uzeli smo Bitmovin. U nastavku ćemo prikazati PHP rešenje za kodiranje jednog video sadržaja u MPEG-DASH format [8].

5.1. Ulaz

Ulaz predstavlja skup informacija koje su potrebne da bi se pristupilo određenoj vrsti skladišta. U ovom primeri ćemo kreirati S3 ulaz sa sledećim linijama koda:

```
$videoInputPath = "path/to/your/input/file.mp4";
$s3InputAccessKey = 'YOUR-AWS-S3-ACCESS-KEY';
$s3InputSecretKey = 'YOUR-AWS-S3-SECRET-KEY';
$s3InputBucketName = "YOUR-AWS-S3-BUCKETNAME";
$s3Input = new S3Input($s3InputBucketName,
    $s3InputAccessKey, $s3InputSecretKey);
$input = $apiClient->inputs()->s3()->create($s3Input);
```

5.2. Izlaz

Izlaz predstavlja skup informacija koje su potrebne da pristupimo određenoj vrsti skladišta da bi kodirani sadržaj bio dostupan za reprodukciju. Za ovaj primer ćemo kreirati S3 izlaz sa sledećim linijama koda:

```
$s3OutputAccessKey = 'YOUR-AWS-S3-ACCESS-KEY';
$s3OutputSecretKey = 'YOUR-AWS-S3-SECRET-KEY';
$s3OutputBucketName = "YOUR-AWS-S3-BUCKETNAME";
$s3Output = new S3Output($s3OutputBucketName,
    $s3OutputAccessKey, $s3OutputSecretKey);
```

5.3. Podešavanje kodeka

Podešavanje kodeka definiše koji kodek i koja postavke kodiranja ćemo koristiti da bismo kreirali jednu audio ili video interpretaciju iz ulazne datoteke.

Primer podešavanja H264 video kodeka:

```
$codecConfigVideo1080p = createH264VideoCodec-  
Configuration($apiClient, 'StreamDemo1080p',  
H264Profile::HIGH, 4800000, null, 1080);
```

Primer podešavanja AAC audio kodeka:

```
$codecConfigAudio128kbit = createAACAudioCodec-  
Configuration($apiClient, 'StreamDemoAAC128k',  
128000);
```

Ulazi i izlazi su resursi koji se mogu lako ponovno iskoristiti za druga kodiranja. Zbog toga, većinom, sve što trebamo da uradimo je da kreiramo jedan ulaz i izlaz i koristimo ih za ostala kodiranja koja koriste druge ulazne i izlazne putanje. Isto važi i za podešavanje kodeka.

5.4. Kreiranje kodiranja

Kodiranje je jednostavno kolekcija resursa koji su mapirani jedni na druge.

```
$encoding = new Encoding('A Name for your encoding');  
$encoding->setCloudRegion  
(CloudRegion::GOOGLE_EUROPE_WEST_1);  
//optional  
$encoding = $apiClient->encodings()-  
>create($encoding);
```

5.5. Početak kodiranja

Kako smo završili sa konfiguracionim delom, možemo preći na samo kodiranje:

```
$apiClient->encodings()->start($encoding);
```

Da bi bili u mogućnosti da reprodukujemo sadržaja, potreban je manifest koji sadrži sve informacije gde i kako će klijent pronaći sadržaj za reprodukciju.

5.6. Kreiranje MPD manifesta

Počnimo sa kreiranjem MPD manifesta uz pomoć Bitmovin API-a. Da bismo to uradili moramo podesiti izlaznu konfiguraciju (izlaz + informacije o izlaznoj putanji), ime manifesta i tip manifesta:

```
//MANIFEST OUTPUT DESTINATION  
$manifestOutput = new EncodingOutput($s3Output);  
$manifestOutput->setOutputPath($outputPath);  
$acl = new Acl(AclPermission::ACL_PUBLIC_READ);  
$manifestOutput->setAcl([$acl]);  
//DASH MANIFEST  
$manifestName = "stream.mpd";  
$manifestType = DashMuxingType::TYPE_TEMPLATE;  
// CREATE DASH MANIFEST  
$manifest = new DashManifest();  
$manifest->setOutputs(array($manifestOutput));  
$manifest->setManifestName($manifestName);  
$manifest = $apiClient->manifests()->dash()-  
>create($manifest);
```

Dodavanje perioda manifestu:

```
$period = new Period();  
$manifestPeriod = $apiClient->manifests()->dash()-  
>createPeriod($manifest, $period);
```

Dodavanje video i audio adaptacionog seta periodu:

```
// CREATE VIDEO ADAPTATION SET  
$videoAdaptionSet = new VideoAdaptationSet();  
$videoAdaptionSet = $apiClient->manifests()->dash()-  
>addVideoAdaptionSetToPeriod($manifest,  
$manifestPeriod, $videoAdaptionSet);  
// CREATE AUDIO ADAPTATION SET FOR EACH  
LANGUAGE  
$audioAdaptionSet = new AudioAdaptationSet();  
$audioAdaptionSet->setLang("en");  
$audioAdaptionSet = $apiClient->manifests()->dash()-  
>addAudioAdaptionSetToPeriod($manifest,  
$manifestPeriod, $audioAdaptionSet);
```

Kreiranje MPD manifesta:

Kao i kod kodiranja, kada se završi sa konfiguracijom možemo početi sa kreiranjem sadržaja. Kada je kreiranje završeno, rezultat možemo pronaći na izlaznoj lokaciji koju smo naznačili.

```
$response = $apiClient->manifests()->dash()-  
>start($manifest);
```

Sada kada smo kreirali MPD manifest, možemo početi sa zahtevanjem sadržaja i prikaza na klijentskoj strani.

6. PRIMER KLIJENTSKOG REŠENJA ZA PRIKAZ VIDEO SADRŽAJA – DASH.JS

Dash.js poseduje bogatu API dokumentaciju kojom možemo kontrolisati razne željene ishode u manipulaciji sa segmentima pri prikazivanju video sadržaja. U nastavku ćemo prikazati samo osnovna podešavanja, neophodna za pokretanje Dash.js programa [9].

6.1. Standardno podešavanje

Prvo kreiramo video element bilo gde unutar „body“ elementa html dokumenta. Takođe, „id“ atribut video elementa nam je bitan kako bismo kasnije u programu indentifikovali određeni element u slučaju da ih postoji više u html dokumentu.

```
<video id="videoPlayer" controls></video>
```

Zatim preko „script“ elementa, dodamo „dash.all.min“ datoteku na kraj „body“ elementa.

```
<body>  
...  
<script  
src="yourPathToDash/dash.all.min.js"></script>  
</body>
```

Nakon toga prelazimo na glavni deo, kreiranje i inicijalizaciju medija programa, kao i učitavanje samog audio ili video sadržaja koji je u mpd formatu.

```
var url = "https://my.site.net/manifest.mpd";  
var player = dashjs.MediaPlayer().create();  
player.initialize(document.querySelector("#videoPlayer")  
, url, true);
```

I sad možemo da počnemo sa reprodukcijom kodiranog sadržaja na klijentskoj strani.

7. MPEG-DASH PODRŠKA PRETRAŽIVAČA I KOMPATIBILNOSTI UREĐAJA

Zbog velike podrške pretraživača i kompatibilnosti uređaja, MPEG-DASH je brzo postao prioritetni standard za prilagodljivi streaming. Od 2012. godine, kada je MPEG-DASH potvrđen od strane ISO/IEC MPEG, sve više i više pretraživača, televizora, smartfona, drugih uređaja i proizvođača se prilagodilo standardu. Korist od međunarodnog, otvorenog standarda koji je nezavisan od proizvoda u poređenju sa vlasničkim rešenjima kao što je Apple-ov HTTP Live Streaming (vođen od strane samo jedne kompanije) je očigledna, i procenat programera, kako softverskih tako i hardverskih, koji prihvataju MPEG-DASH standard će samo da raste [10].

8. MPEG-DASH, APPLE HLS, MICROSOFT SMOOTH STREAMING I ADOBE HDS

Sve postojeće prilagodljive HTTP streaming tehnologije, kao što su vlasnički Adobe HTTP Dynamic Streaming, Apple HTTP Live Streaming, Microsoft Smooth Streaming, i jedino međunarodno standardizovano rešenje MPEG-DASH prate skoro iste principe [11].

8.1. Prilagodljivi streaming - poređenje karakteristika

Navesćemo nekoliko razlika između najpopularnijih standarda [11]:

- MPEG-DASH kao i Apple-ov HLS mogu da se koriste sa običnim HTTP serverima kao što su Apache, Nginx, IIS, itd. Adobe, kao i Microsoft koriste mehanizme na serverskoj strani koji zahtevaju dodatnu logiku na serveru.
- Prebacivanje između više audio kanala je posebno važno kod sadržaja sa više jezika. MPEG-DASH podržava ovu mogućnost kao i Apple HLS i Microsoft Smooth Streaming.
- MPEG-DASH može efektivno koristiti HTML5 Media Source Extensions (MSE) i omogućava "prirodan" prenos u pretraživaču. iOS na Mac-u ne podržava MSE, ali adaptivni streaming može da se reprodukuje na HTML5 koristeći HLS.
- MPEG-DASH je zvanični streaming format u HbbTV 1.5 i 2.0.
- MPEG-DASH je nezavisan od video i audio kodeka i može da se koristi sa više video kodeka kao što su H.264/AVC, H.265/HEVC, MPEG-2 video, VP8, VP9, itd. i više audio kodeka kao što su MP3, AAC, itd. Takođe, od juna 2017. godine, Apple je najavio podršku za H.265/ HEVC za macOS High Sierra i iOS11.

9. ZAKLJUČAK

Uzimajući u obzir da Youtube i Netflix koriste MPEG-DASH standard, moglo bi se reći da su dva najveća izvora prometa na internetu upravo izabrali MPEG-DASH kao glavnu striming tehnologiju. Videti ta dva imena uz MPEG-DASH mogu dati dosta sigurnosti u tu tehnologiju, ali postoji tu puno toga više od samo velikih imena. Postoji mnogo prednosti u prihvatanju MPEG-DASH standarda kako sada tako i u budućnosti.

10. LITERATURA

- [1] Iraj Sodagar. MPEG-DASH: The Standard for Multimedia Streaming Over Internet. Microsoft Corporation, 1 Microsoft Way, Redmond, WA, USA 98052. URL http://www.bogotobogo.com/VideoStreaming/images/mpeg_dash/DASH-IEEE-multimedia-preprint.pdf
- [2] Wikipedia. Dynamic Adaptive Streaming over HTTP. URL https://en.wikipedia.org/wiki/Dynamic_Adaptive_Streaming_over_HTTP
- [3] Christian Timmerer. MPEG-DASH: Overview, State-of-the-Art, and Future Roadmap. July 2016.
- [4] Christopher Mueller. MPEG-DASH (Dynamic Adaptive Streaming over HTTP). April 2015. URL <https://bitmovin.com/dynamic-adaptive-streaming-http-mpeg-dash/>
- [5] Reinhard Grandl. Watch HEVC MPEG-DASH Streams in HTML5 using Bitmovin. October 2015. URL <https://bitmovin.com/html5-hevc-mpeg-dash-streams-using-bitmovin/>
- [6] Brendan Long. The Structure of an MPEG-DASH MPD. March 2015. URL <https://www.brendanlong.com/the-structure-of-an-mpeg-dash-mpd.html>
- [7] Stefan Lederer. Optimal Adaptive Streaming Formats MPEG-DASH & HLS Segment Length. April 2015. URL <https://bitmovin.com/mpeg-dash-hls-segment-length/>
- [8] Wikipedia. Bitmovin. <https://en.wikipedia.org/wiki/Bitmovin/>
- [9] <http://cdn.dashjs.org/latest/jsdoc/index.html>
- [10] Stefan Lederer. MPEG-DASH Browser Support and Device Compatibility. . March 2015. URL <https://bitmovin.com/mpeg-dash-browser-support-device-compatibility/>
- [11] Christopher Mueller. MPEG-DASH vs. Apple HLS vs. Microsoft Smooth Streaming vs. Adobe HDS. March 2015. URL <https://bitmovin.com/mpeg-dash-vs-apple-hls-vs-microsoft-smooth-streaming-vs-adobe-hds/>

Kratka biografija:



Stevan Cvijanović rođen je u Doboju 1992. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Teorija informacija i komunikacija odbranio je 2017.god.

PREGLED TRŽIŠTA MOBILNOG PRENOSA PODATAKA U SRBIJI I OKRUŽENJU

MOBILE DATA MARKET OVERVIEW FOR SERBIA AND SURROUNDING COUNTRIES

Stefan Mićović, Željen Trpovski, *Fakultet tehničkih nauka, Novi Sad*

Oblast - ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu akcenat je stavljen na pregled prenosa mobilnih podataka u Srbiji, i kao zemlje u okruženju, Crnoj Gori za 2016. godinu. Kao uvod u samu analizu prenosa mobilnih podataka, čitalac se uvodi u rad teorijskim osnovama mobilnih tehnologija, gde su hronološki obrađene tehnologije mobilnih mreža i principi prenosa podataka kroz mobilne mreže. Nakon toga sledi detaljna analiza.

Abstract – In this paper, emphasis is placed on the review of mobile data transfer in Serbia, and as countries in the region, Montenegro for 2016. As an introduction to the analysis of mobile data transfer, the reader is introduced into the work of theoretical foundations of mobile technologies, where chronologically processed mobile network technologies and principles of data transmission through mobility networks. After that, a detailed analysis follows.

Ključne reči: Mobilne telekomunikacije, mobilni operateri, tržište mobilnih telekomunikacija.

1. UVOD

Držeći korak sa novim tehnologijama, postali smo zavisni od interneta na mobilnim uređajima.

Podaci na osnovu kojih je prikazano stanje na tržištu telekomunikacija u Republici Srbiji odnose se uglavnom za teritoriju Republike Srbije bez Kosova i Metohije.

Pod mobilnim tehnologijama podrazumevaju se prenosivi uređaji koji obuhvataju hardver (fizički uređaj), softver (operativni sistem i aplikacije) i komunikaciju (mrežni servisi).

Sveprisutnost i brz razvoj mobilnih i bežičnih tehnologija omogućio je razvoj novih okruženja za elektronsko [4] učenje.

Tema ovog rada jeste pregled tržišta mobilnog prenosa podataka prvenstveno u Srbiji, čije je tržište detaljno analizirano kroz najveći deo rada uz korišćenje raznih metoda i parametara radi dobijanja jasne i koncizne slike tržišta i upoređivanja kako sa državom u regionu Crnom Gorom, tako i sa ostalim državama u Evropi i ostatku sveta.

2. PREGLED MOBILNIH OPERATIVNIH SISTEMA

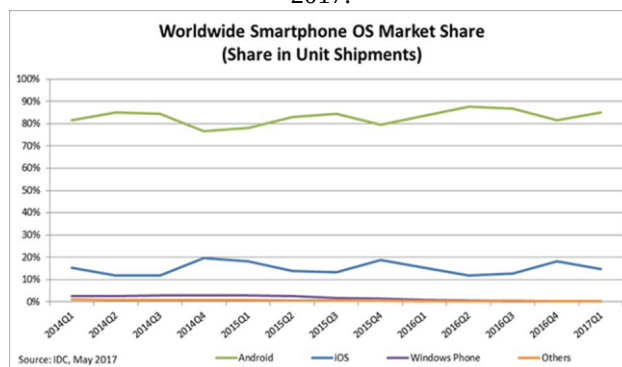
Poput računarskih operativnih sistema, operativni sistemi za mobilne uređaje predstavljaju platformu koja omogućava drugim programima da rade. Jednostavniji su od računarskih OS i imaju veći akcenat na bežičnim širokopojasnim i lokalnim vezama, mobilnim multimedijalnim formatima i različitim metodima unosa.

Prema podacima IDC Quarterly Mobile phone tracker mobilne kompanije su isporučile ukupno 344.3 miliona pametnih telefona širom sveta u prvom kvartalu 2017.

Korisnici nastavljaju da potražuju pametne telefone čini se više nego ikad. Isporuca pametnih telefona širom sveta porasla je 3.4% u prvom kvartalu 2017 godine što je nešto niže od prethodne prognoze IDC-a od 3.6%. Data je tabela 1., kao i grafikon 1.

Period	Android	iOS	Windows Phone	Others
2016Q1	83,4%	15,4%	0,8%	0,4%
2016Q2	87,6%	11,7%	0,4%	0,3%
2016Q3	86,8%	12,5%	0,3%	0,4%
2016Q4	81,4%	18,2%	0,2%	0,2%
2017Q1	85,0%	14,7%	0,1%	0,1%

Tabela1. IDS izvor iz maja 2017.



Grafikon 1. Svetski tržišni udeo

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Željen Trpovski, vanr. prof.

3. LTE

S LTE (Long Term Evolution) je projekat koji je nastao 2004. godine od strane telekomunikacionog regulatornog tela - 3GPP (Third Generation Partnership Project). LTE predstavlja standard za bežični prenos podataka velikog protoka. Ova mrežna tehnologija je evoluirala od ranijeg projekta 3GPP-a UMTS-a koji je evoluirao od prethodnog GSM-a. Standardizacija LTE tehnologije je razvijena u okviru 3GPP specifikacije Release 8.

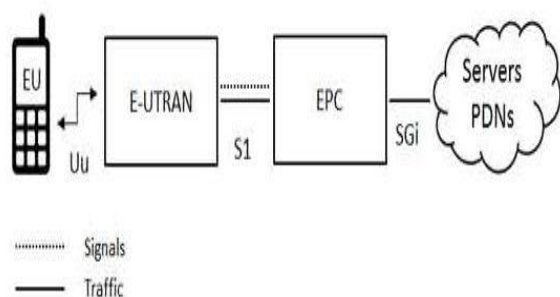
U slučaju ove mrežne tehnologije ispunjena je kompatibilnost unazad sa GSM/EDGE i UMTS/HSDPA mrežama. LTE se oslanja na već postojeću infrastrukturu uz određene dodatke. LTE je dizajniran da obezbedi korisniku veliku brzinu protoka podataka i ispuni zahteve korisnika za novim servisima. Protok koji je moguće ostvariti na downlink-u je 100Mb/s, a u laboratorijskim uslovima ostvaren je protok od neverovatnih 1Gb/s, dok je na uplink-u protok do 50Mb/s. U slučaju HSDPA tehnologije najveći protok na downlink-u bio je svega 14Mb/s, a na uplink-u 5,7 Mb/s.

Očigledno je da LTE omogućava korisnicima daleko veći protok. Cilj prelaska na LTE mrežnu tehnologiju je obezbeđivanje većeg protoka korisnicima, poboljšanje spektralne efikasnosti, realizovanje znatno efikasnije paketske komutacije i prevođenje mobilne mreže na isključivo paketsku mrežu.

EPS (Evolved Packet System) je podeljen na dva dela. LTE deo koji se bavi tehnologijom koja je u vezi sa radio pristupom E-UTRAN (Evolved UTRAN) i drugi deo koji predstavlja EPC (Evolved Packet Core) koji se bavi tehnologijom koja se odnosi na jezgro mreže (core network). LTE mreža je E2E IP (End to end IP) što znači da svi saobraćajni tokovi, sav prenos podataka od UE (User Equipment) do PDN (Public Data Network) koji povezuje entitete, prenose se zasnovano na IP protokolu u okviru EPS-a.

EPC pruža pristup ka spoljnim mrežama za prenos podataka (Internet), upravlja funkcijama sigurnosti (autentifikacija, dodela sigurnosnih ključeva), vodi računa o pretplatničkim informacijama, tarifiranju i naplati kao i mobilnost prema drugim pristupnim mrežama. E-UTRAN obavlja sve funkcije koje su u vezi sa radio pristupom za aktivne terminale. Sastoji se od baznih stanica (eNB – e-Node B).

Pojednostavljena arhitektura LTE mreže je na slici 1 na kojoj se mogu videti glavne komponente ove arhitekture.



Slika 1. Uprošćena arhitektura LTE mreže

4. 5G MOBILNA TEHNOLOGIJA

Pošto 4G telekomunikacioni sistemi sada počinju da se primenjuju, pažnja se okreće u pravcu razvoja 5. generacije, ili 5G tehnologija i usluga. Mada je za uvođenje bilo kojeg ćelijskog sistema potrebno mnogo godina, već se ispituje razvoj tehnologije 5G sistema. Da bi se obezbedilo pravovremeno i pouzdano uvođenje, nove 5G tehnologije će morati da se odaberu, razviju i usavrše. Nova peta generacija, 5G tehnologija za mobilne sisteme verovatno će početi da se ostvaruje oko 2020. godine [3] dok će uvođenje slediti kasnije.

Trenutno stanje 5G tehnologije za mobilne sisteme nalazi se u veoma ranim fazama razvoja. Nekoliko kompanija ispituje tehnologije koje bi mogle da se koriste kao delovi sistema. Pored toga, organizacije za standarde, pogotovo 3GPP, svesne su tog razvijanja ali još uvek aktivno ne planiraju 5G sisteme.

Mnoge tehnologije koje će se koristiti za 5G počće da se pojavljuju u sistemima koji se koriste za 4G, a zatim, kako novi 5G mobilni sistem bude počeo da se formuliše na konkretniji način, one će se ugraditi u novi 5G mobilni sistem.

Kako su se razvijale različite generacije mobilnih telekomunikacija, svaka od njih je donosila svoja poboljšanja. Isto će biti i sa 5G tehnologijom.

Svaka nova 5G mobilna tehnologija pete generacije mora da obezbedi značajan porast u odnosu na prethodne sisteme kako bi mobilnim operaterima nudila adekvatan poslovni podstrek da investiraju u bilo koji nov sistem. Mogućnost koja bi mogla da se pojavi sa 5G tehnologijom bio bi daleko bolji nivo povezanosti i pokrivanja. Za to je izmišljen izraz World Wide Wireless Web, ili WWW. Da bi 5G tehnologija bila u stanju da to postigne, biće potrebni novi metodi povezivanja pošto su među glavnim manama prethodnih generacija bili nedostatak pokrivanja, prekidanje poziva i loše performanse na rubovima ćelija. 5G tehnologija će morati da se pozabavi time.

Mada organizacije za standarde još nisu definisale parametre potrebne da bi se zadovoljile 5G performanse, druge organizacije su postavile vlastite ciljeve, koji bi mogli na kraju da utiču na konačne specifikacije.

Tipični parametri za 5G standard bi mogli da obuhvate navedeno u Tabeli broj 2.

Predložene 5G bežične performanse	
Parametar	Predložene performanse
Kapacitet mreže (Network capacity)	10 000 puta trenurna mreža
Maksimalna brzina prenosa podataka (speed data rate)	10 Gbs/s
Brzina prenosa podataka na rubu ćelije (Cell edge data rate)	100 Mb/s
Kašnjenje (Latency)	< 1 ms

Tabela 2. Predložene 5G bežične performanse

5. MOBILNI PRENOS PODATAKA U SRBIJI U 2016.

Ukupan prihod ostvaren na tržištu elektronskih komunikacija Republike Srbije u 2016. godini iznosi oko 189,4 milijardi dinara, što je neznatno više (oko 1%) u odnosu na prethodnu godinu. Prihodi izraženi u evrima iznose 1,54 milijarde evra i neznatno su manji u odnosu na prethodnu godinu, a ova razlika u odnosu na rast posmatran u dinarima je posledica višeg prosečnog srednjeg kursa dinara prema evru u 2016. godini u odnosu na 2015. godinu. U bruto društvenom proizvodu Srbije prihodi od elektronskih komunikacija su u 2016. godini imali udeo od 4.51%.

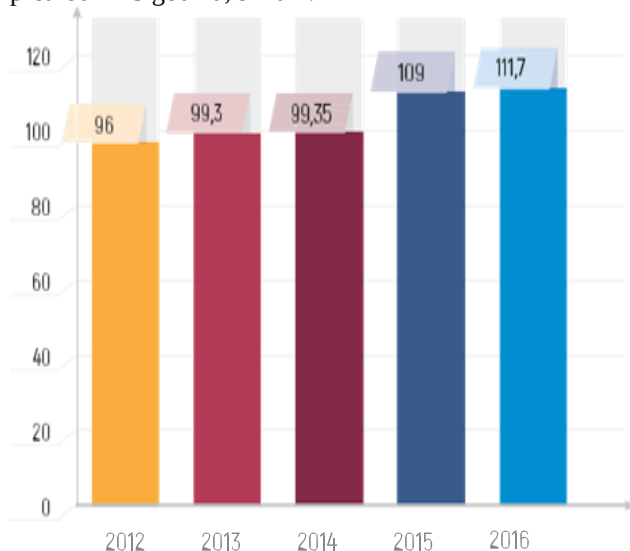
Posmatrajući udeo svake od usluga u ukupnim prihodima na tržištu elektronskih komunikacija u Srbiji [1,3] u 2016. godini, najveći udeo u ukupnim prihodima na tržištu elektronskih komunikacija ostvaren je, kao i prethodnih godina, od pružanja usluge mobilne telefonije i čini 59% ukupnih prihoda.

Tokom 2016. godine ostvareni su prihodi od mobilne telefonije u iznosu od oko 111,7 milijardi dinara, odnosno 907,5 miliona evra. Izraženi u dinarima, prihodi u 2016. godini su zabeležili rast od oko 2,7% u poređenju sa prethodnom godinom, kada su iznosili oko 109 milijardi dinara.

Prihodi izraženi u evrima su viši za manje od 1%, a ova razlika u odnosu na rast posmatran u dinarima je posledica višeg prosečnog srednjeg kursa dinara prema evru u 2016. godini u odnosu na 2015. godinu.

Investicije u ovom segmentu tržišta [3] elektronskih komunikacija su smanjene za oko 14% u poređenju sa prethodnom godinom i iznose 11,1 milijardu dinara.

U nastavku biće prikazana slika na kojoj se može sagledati ukupan prihod od mobilne telefonije za period u prethodnih 5 godina, slika 2.

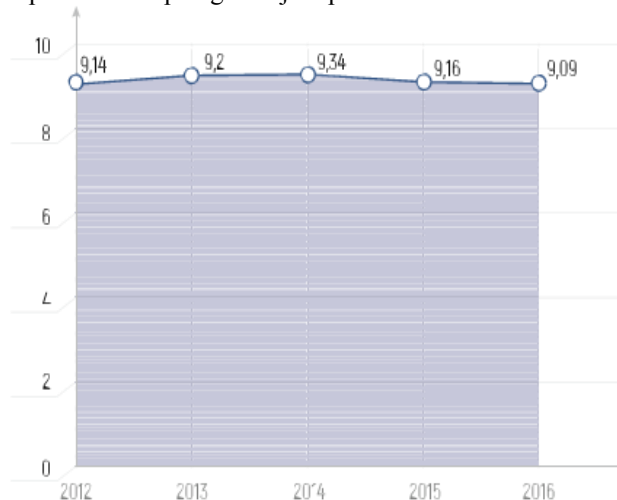


Slika 2. Ukupni prihodi od mobilne telefonije (u milijardama dinara) [1]

Ukupan broj korisnika mobilne telefonije je na kraju 2016. godine iznosio 9.094.447, što je za oko 0,7% manje u odnosu na 2015. godinu. Smanjenje ukupnog broja korisnika u poslednje dve godine je posledica smanjenja broja pripejd korisnika, ali penetracija je i pored toga i

dalje visoka i zabeležen je rast odlaznog govornog saobraćaja [1] u minutima.

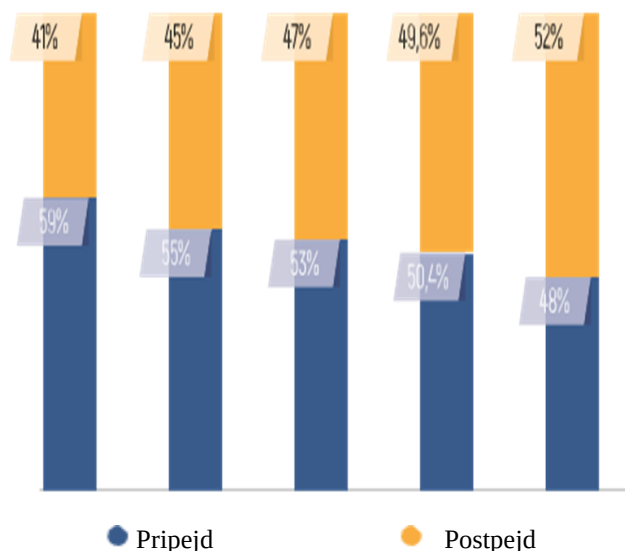
Na Slici 3. prikazano je kretanje ukupnog broja korisnika u prethodnom petogodišnjem periodu.



Slika 3. Ukupan broj aktivnih korisnika mobilne telefonije u milionima [1]

Ukupan broj korisnika čine postpejd korisnici i pripejd korisnici aktivni u poslednja tri meseca posmatrane godine. Raspodela korisnika na pripejd i postpejd je prikazana na slici 4.

U prethodnom periodu broj pripejd korisnika je uvek bio veći od broja postpejd korisnika. Međutim, u 2016. godini po prvi put broj postpejd korisnika prevazilazi broj pripejd korisnika i dostiže učešće od 52%.



Slika 4. Raspodela pripejd/postpejd korisnika [1]

6. MOBILNI PRENOS PODATAKA U CRNOJ GORI U 2016.

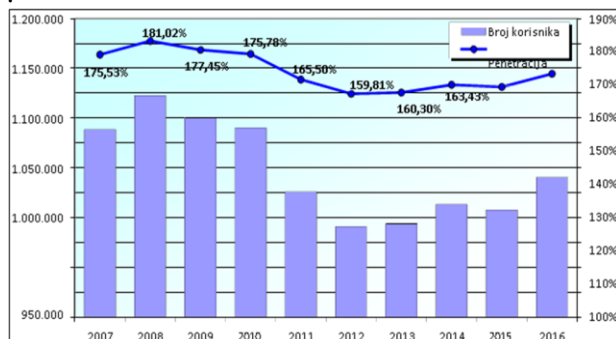
Na crnogorskom tržištu mobilne telefonije usluge korisnicima pružaju tri telekomunikaciona operatora i to: Telenor d.o.o, Crnogorski Telekom a.d. i Mtel d.o.o.

Na kraju 2016. godine broj korisnika mobilne telefonije u Crnoj Gori je iznosio 1.040.747 što odgovara penetraciji od 167,85 %.

Od ovog broja je 578.874 (55,62 %) bilo prepaid korisnika, dok su preostali 461.873 (44,38 %) bili korisnici u kategoriji postpaid. U odnosu na kraj 2015. godine, broj korisnika je veći za 3,26 %.

Najveća penetracija tokom 2016. godine na tržištu mobilne telefonije Crne Gore zabeležena je u septembru, kada je ista iznosila 181,79 %, a najniža u februaru kada je iznosila 158,03 %.

Na slici 5. prikazano je kretanje penetracije mobilne telefonije u periodu 2007-2016. godine



Slika 4. Kretanje penetracije mobilne telefonije u desetogodišnjem periodu, 2007.-2016. [1]

Od ukupnog broja 1.040.747 korisnika mobilne telefonije u Crnoj Gori na kraju 2016. godine, mobilni operator Telenor d.o.o. je imao 377.701 korisnika, Crnogorski Telekom a.d. 361.149 korisnika i Mtel d.o.o. 301.897 ili procentualno: Telenor d.o.o. 36,29 %, Crnogorski Telekom a.d. 34,70 % i Mtel d.o.o. 29,01 %.

U toku 2016. godine sva tri mobilna operatora u Crnoj Gori ostvarila su 1.801.419.845 minuta odlaznog saobraćaja. Ukupno ostvareni odlazni saobraćaj iz mreže Telenora d.o.o. je imao učešće od 39,53 %, odlazni saobraćaj ostvaren iz mreže Crnogorskog Telekoma a.d. je iznosio 37,96 % dok je udeo odlaznog saobraćaja iz mreže Mtele d.o.o. iznosio 22,51 %.

Imajući u vidu broj korisnika, stepen konkurencije, raznovrsnost usluga i primenjene tehnologije može se smatrati da mobilne komunikacione mreže i usluge [2] predstavljaju najrazvijeniji segment tržišta elektronskih komunikacija u Crnoj Gori.

7. ZAKLJUČAK

Ovim radom napravljen je pregled prenosa mobilnih podataka u Srbiji, kao i u Crnoj Gori. Opisane su osnovne komponente mobilnih tehnologija, problematike i realizacije mreža, opisane razne njene prednosti, kao i neki nedostaci.

Zasebnom i uporednom analizom operatera koji pružaju usluge mobilne telefonije stvorena je slika tržišta telekomunikacija, stepen i dalja predviđanja razvoja.

Ubrzani razvoj mobilnih mreža četvrte generacije je u najvećoj meri obeležio telekomunikaciono tržište Srbije u 2016. godini.

Na kraju godine su sva tri operatora mobilne telefonije pokrivala više od polovine stanovništva LTE signalom, a u 2017. se očekuje da preko 90% populacije dobije mogućnost da koristi sve prednosti ove tehnologije. Kao što je i logično, širenje pristupačnosti mobilnog interneta velikih brzina donelo je značajno povećanje količine prenetih podataka. U 2016. je potrošeno preko 50% gigabajta više u odnosu na prethodnu godinu.

Verovatno najbolji pokazatelj koliko koristimo pametne telefone za komunikaciju putem interneta jeste penetracija društvenih medija na mobilnim uređajima (broj korisnika koji društvenim medijima pristupaju putem mobilnih uređaja u odnosu na ukupan broj stanovnika). Crna gora sa 46% (290 hiljada korisnika društvenih medija na mobilnim uređajima), Makedonija sa 44% su iznad evropskog proseka, dok je na začelju liste u regionu Srbija 31% (2,70 miliona) što predstavlja ozbiljan zaostatak u odnosu na evropski prosek.

Na osnovu sveukupnih prikazanih podataka možemo zaključiti da Srbija, Crna Gora, kao i ostale države iz regiona, u principu, nalaze ispod svetskog proseka kada je u pitanju stepen upotrebe digitalnih tehnologija. Najveća pouka ovog rada jeste da digitalne tehnologije nisu naša budućnost već naša sadašnjost. Ipak, sve ovo što sada vidimo nije ni delić onoga što će nam nove tehnologije omogućiti u narednih 20 godina. Sve to željno i verujem spremno očekujemo.

8. LITERATURA

[1] www.ratel.rs

[2] www.digitalizuj.me

[3] www.telekomunikacije.rs

[4] N. Miloradović, integracija mobilnih obrazovnih servisa u sisteme elektronskog obrazovanja

Kratka biografija:



Stefan Mićović rođen je u Smederevu 1988. godine. Master rad odbranio je 2017. godine na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Telekomunikacioni sistemi



Željen Trpovski rođen je u Rijeci 1957. godine. Doktorirao je na Fakultetu tehničkih nauka 1998. god. Oblast interesovanja su telekomunikacije i obrada signala.

PREDLOG TERMIČKOG MODELA ASINHRONOG MOTORA PROPOSAL OF THERMAL MODEL OF INDUCTION MOTOR

Igor Halkić, Veran Vasić, Đura Oros, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu je prezentovan pojednostavljen termički model asinhronog motora i eksperimentalno verifikovan. Termički model je baziran na metodi koncentrisanih parametara. U radu su predložene analitičke jednačine potrebne za modelovanje termičkog kola. Predloženi model omogućava predviđanje temperature u glavnim delovima motora na osnovu merenih ili procenjenih gubitaka u motoru. Model dozvoljava proračun raspodele temperature u prevashodno radialnom pravcu motora, od vratila unutar motora do spoljne površine kućišta. Na kraju, model je primenjen i uspešno verifikovan prema merenjima izvršenim na asinhronom motoru od 450 kW u praznom hodu.

Abstract – In this paper a simplified thermal model of an induction motor is presented and verified. Thermal model is based on a method of lumped parameters. Paper proposes a set of analytical equations needed for thermal network modelin. The proposed model allows to predict over temperature in the main parts of the motor; based on a measured or estimated losses in the machine. Model allows calculation of temperature distribution in radial direction, from the shaft to the frame outer surface. Eventually, model is applied and succesfully verified with measurements on induction motor of 450 kW conducted in no-load test.

Ključne reči: Termička analiza, metod koncentrisanih parametara, asinhroni motor, termički model

1. UVOD

Glavni ograničavajući faktor opretnosti bilo koje električne mašine predstavlja temperatura. U svrhu predviđanja temperatura unutar mašine uvode se termički modeli. Metod koncentrisanih parametara daje prosečnu temperaturu elemenata unutar mašine. Njegova prednost leži u jednostavnoj matematičkoj formi i implementaciji. Ovaj rad se bavi termičkom analizom kaveznog asinhronog motora sa ventilatorom na sopstvenom vratilu (TEFC) putem metode koncentrisanih parametara.

Cilj je predstaviti odgovarajući termički model koji se može lako implementirati i čije je parametre relativno lako proračunati i čija upotreba nije ograničena samo na primenu u ustaljenom stanju, već i u tranzijentnim režimima. Model se sastoji iz 7 čvorova i 13 termičkih otpornosti. Performanse modela su potvrđene na osnovu eksperimentalno dobijenih vrednosti temperatura asinhronog motora od 450 kW u praznom hodu.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Veran Vasić, red.prof.

2. TEORIJA PRENOSA TOPLOTE U ASINHRONOM MOTORU

Prenos toplote je toplotna energija koja je u tranzitu zahvaljujući prostornoj temperaturnoj razlici. Projektovanje termičkih performansi asinhronog modela predstavlja veći izazov od projektovanja elektromagnetskih svojstava. Najbitniji faktor u termičkom dizajnu mašine predstavlja temperatura fluida oko mašine, jer ona definiše temperaturu. Zagrevanje unutar mašine se dešava usled snage gubitaka tokom prenosa snage od električnog izvora do mehaničkog opterećenja. Drugim rečima, gubici u električnoj mašini predstavljaju izvore toplote.

Toplota se odvodi kondukcijom, konvekcijom i radijacijom. Prisilna konvekcija je najefikasniji metod hlađenja ukoliko se izostavi direktno hlađenje putem vode. Značajne količine toplote se mogu preneti i kondukcijom preko stopa i montažne prirubnice. Prenos toplote radijacijom je zanemaren.

Analogno sa električnom otpornošću [1], koja se definiše kao odnos razlike potencijala i struje, definiše se konduktivna termička otpornost R_{th} kao odnos razlike temperatura i toka toplote Φ_{th} ,

$$R_{th} = \frac{\theta}{\Phi_{th}} = \frac{l}{\lambda S}. \quad (1)$$

Toplota se uvek prenosi istovremeno kondukcijom i konvekcijom. Konvekcija se definiše kao prenos toplote između polja visoke temperature i polja niske temperature. Poznavanje graničnih slojeva je ključno u analizi prenosa toplote i mase između čvrste površine i rashladnog fluida koji teče na površini čvrstog tela. Postoje dve vrste konvekcije, prirodna i prisilna konvekcija, koje se razlikuju prema izvoru kretanja.

Njutnov zakon hlađenja definiše gustinu toplotnog toka q_{th} koji je stvoren konvekcijom

$$q_{th} = \frac{\Phi_{th}}{S} = \alpha_{th} \theta, \quad (2)$$

kao posledica se ima termička otpornost konvekcije

$$R_{th} = \frac{\theta}{\Phi_{th}} = \frac{1}{\alpha_{th} S}, \quad (3)$$

gde α_{th} predstavlja koeficijent prenosa toplote [1]. Do ovog koeficijenta je teško doći s obzirom na to da zavisi od čitavog niza parametara.

3. TERMIČKO MODELOVANJE ASINHRONOG MOTORA

U poređenju sa drugim metodama, metod koncentrisanih parametara je jednostavan i atraktivan i u stanju je da pruži tačnu predstavu termičkih dešavanja unutar mašine. Model [2] se uspešno koristi za proračun porasta temperature u električnim mašinama, a jedna od osnovnih prednosti ovog modela je analogija sa električnim kolima. Model je zasnovan na tezi da se sa termičke tačke gledišta motor može podeliti na određen broj celina. Svaka celina ima svoju termičku zapreminu, izvor toplote i međuveze sa susednim komponentama. Vrednosti parametara se dobijaju skoro u potpunosti na osnovu dimenzija motora, termičkih svojstava materijala i na osnovu toplotnih koeficijenata. Model se sastoji iz termičkih otpornosti, termičkih kapacitivnosti i gubitaka unutar mašine.

3.1 Termičke otpornosti

U ovom radu se primenjuju dve vrste otpornosti u zavisnosti od tipa razmene toplotne energije između površina koncentričnih cilindara koje te površine predstavljaju. Konduktivne termičke otpornosti u radijalnom pravcu za šuplji cilindar se mogu računati kao

$$R_{kond} = \frac{\ln(r_o/r_i)}{2\pi kL} \quad (4)$$

Gde je:

r_o - predstavlja spoljašnji prečnik cilindra, r_i - unutrašnji, L - dužinu cilindra i

k - predstavlja termičku provodljivost materijala.

Konvektivne otpornosti se definišu na sledeći način:

$$R_{konv} = \frac{1}{hA} \quad (5)$$

Veličina h se naziva koeficijent konvektionog prenosa toplote, a A predstavlja površinu objekta koji predaje toplotu.

3.2 Termičke kapacitivnosti

U vremenski zavisnim problemima pojava uskladištene energije u sistemu se modeluje putem termičkih kapacitivnosti. Svakom čvoru je dodeljena termička kapacitivnost između čvora i ambijenta. Termička kapacitivnost elementa se dobija iz geometrije elementa i konstruktivnih materijala

$$C_t = m_i c_i \quad (6)$$

Gde je:

m_i predstavlja masu, a

c_i specifičnu toplotnu kapacitivnost i -tog elementa [3].

3.3 Termički model asinhronog motora

Termički model [2] je od svog predstavljanja služio kao referentni model za istraživače koji rade na termičkom problemu asinhronog motora. Model je vrlo tačan zato što uzima u obzir i aksijalne i radijalne tokove toplote, kao i nekoliko kompleksnih prenosa toplote putem konvekcije. Ipak, zbog kompleksne geometrije i tokova fluida u motoru, vrlo je zahtevno doći do svih parametara modela, te se u ovom radu predlaže nešto jednostavniji model zasnovan na modelu, [4]. Sledeće pretpostavke su donesene:

- Pretpostavljena je simetričnost motora oko vratila i simetričnost u odnosu na radijalnu ravan kroz centar mašine.
- Uticaj asimetričnih raspodela temperatura koje postoje unutar motora sa ventilatorom montiranim na jednom kraju vratila je zanemaren.
- Svaki cilindar je termički simetričan u radijalnom pravcu;
- Unutrašnji izvori toplote su uniformno raspoređeni.
- Tok toplote se razmatra samo u radijalnom pravcu. Ovom pretpostavkom se termičke radijalne otpornosti mogu modelovati upotrebom jednačina šupljeg cilindra.

Direktne posledice ovih pretpostavki su:

- model sa smanjenim brojem termičkih otpornosti;
- sve termičke otpornosti su jednostavne za proračun.

Radi jednostavnosti, mehanički gubici i dodatni gubici su integrisani u gubitke u bakru rotora. Gubici u gvožđu statora se primenjuju na prosečni prečnik jarma statora.

Sledeće vrednosti termičkog modela se mogu izračunati sa dovoljnom tačnošću upotrebom analitičkih jednačina:

R_{sj1} – Polovina radijalne kondukcione termičke otpornosti jarma statora (unutrašnji deo)

$$R_{sj1} = \frac{1}{2\pi k_{Fe} L_s} \ln \left(\frac{r_m}{r_{iy}} \right) \quad (7)$$

R_{sj2} – Polovina radijalne kondukcione termičke otpornosti jarma statora (spoljašnji deo)

$$R_{sj2} = \frac{1}{2\pi k_{ir} L_s} \ln \left(\frac{r_{oy}}{r_m} \right) \quad (8)$$

R_{svz} – Konvekciona termička otpornost između statora i vazdušnog zazora

$$R_{svz} = \frac{1}{A_{ust} h_{vz}} \quad (9)$$

$A_{ust} = 2\pi r_{is} L_s$ – unutrašnja površina statora

$$h_{vz} = \frac{N_{Nu} k_{vz}}{2l_{ag}} \quad (10)$$

Nuseltov broj zavisi od Rejnoldsovog kritičnog broja, i za predmetno razmatranje iznosi $N_{Nu} = 2$.

R_{rvz} – Konvekciona termička otpornost između rotora i vazdušnog zazora

$$R_{rvz} = \frac{1}{A_{srt} h_{vz}} \quad (11)$$

$A_{srt} = 2\pi r_{or} L_s$ – spoljašnja površina rotora

R_{sz} – Radijalna kondukciona termička otpornost zubaca statora

Otpornost uzima u obzir toplotni tok od zubaca statora do jarma statora. Ovaj deo motora se ekvivalentira cilindrom.

$$R_{st} = \frac{1}{2\pi k_{Fe} L_s p_{Fe}} \ln \left(\frac{r_{iy}}{r_{is}} \right) \quad (12)$$

p_{Fe} – procentualna vrednost zapremina zubaca statora u odnosu na ukupnu zapreminu zubaca i žlebova

R_{os} – Radijalna kondukciona termička otpornost osovine

$$R_{os} = \frac{1}{2\pi k_{Fe} L_s} \ln \left(\frac{r_{ory}}{r_{iry}} \right) \quad (13)$$

R_{nsk} – Kondukciona termička otpornost između bočnih namotaja statora i kućišta

$$R_{nsk} = \frac{1}{2\pi k_{Fe} (L_{ec} - L_s)} \ln \left(\frac{r_{oy}}{r_{oy} - \alpha t_{sj}} \right) \quad (14)$$

Termička otpornost uzima u obzir toplotni tok između bočnih namotaja statora i kućišta. Kako je brzina strujanja vazduha u ovom delu mašine zanemarljiva, razmatra se samo fenomen konduktivnosti. Koeficijent α dozvoljava razmatranje ekvivalentne udaljenosti između bočnih namotaja i kućišta. Vrednosti α se kreću u rasponu 0,4-0,7.

R_{bsv} – Konvekciona termička otpornost između bočnih namotaja statora i vazduha

$$R_{bsv} = \frac{1}{A_{bs} h_{bs}} \quad (15)$$

$A_{bs} = (L_{ec} - L_s) 2\pi r_{is}$ – površina bočnih strana u kontaktu sa vazduhom koji struji unutar mašine. Prema [5], konvekcioni koeficijent h_{bs} se može izračunati sledećom formulom

$$h_{bs} = 41,4 + 6,22 \omega \frac{r_{or}}{2} \quad (16)$$

r_{or} – spoljašnji prečnik rotora

ω – ugaona brzina rotora

R_{vp} – Konvekciona termička otpornost između vazduha unutar mašine i poklopaca mašine

$$R_{vp} = \frac{1}{A_{vp} h_{vp}} \quad (17)$$

$A_{vp} = 2\pi(r_{oy} + t_k)^2$ – spoljašnja površina dva poklopca mašine. Pretpostavlja se da je konvekcioni koeficijent h_{bs} jednak koeficijentu h_{vp} i se poklopci nalaze na identičnim temperaturama kućišta. Površina poklopaca se računa kao prosta površina kruga.

Sledeće otpornosti zahtevaju sprovođenje ogleda putem DC napajanja [4].

R_{sz} – Kondukciona termička otpornost između bakra statora i žleba

$$R_{sz} = \frac{t_e}{k_{sz} A_z} \quad (18)$$

$A_z = l_{oz} L_s$ – površina unutrašnjosti žleba

$$t_e = \frac{S_z - S_{cu}}{l_{sb}} \quad (19)$$

R_{sjk} – Kondukciona otpornost između jezgra statora i kućišta

$$R_{sjk} = \frac{l_{kp}}{2k_{vz} \pi r_{oy} L_s} \quad (20)$$

R_{nka} – Prirodna konvekciona termička otpornost između kućišta i vazduha ambijenta

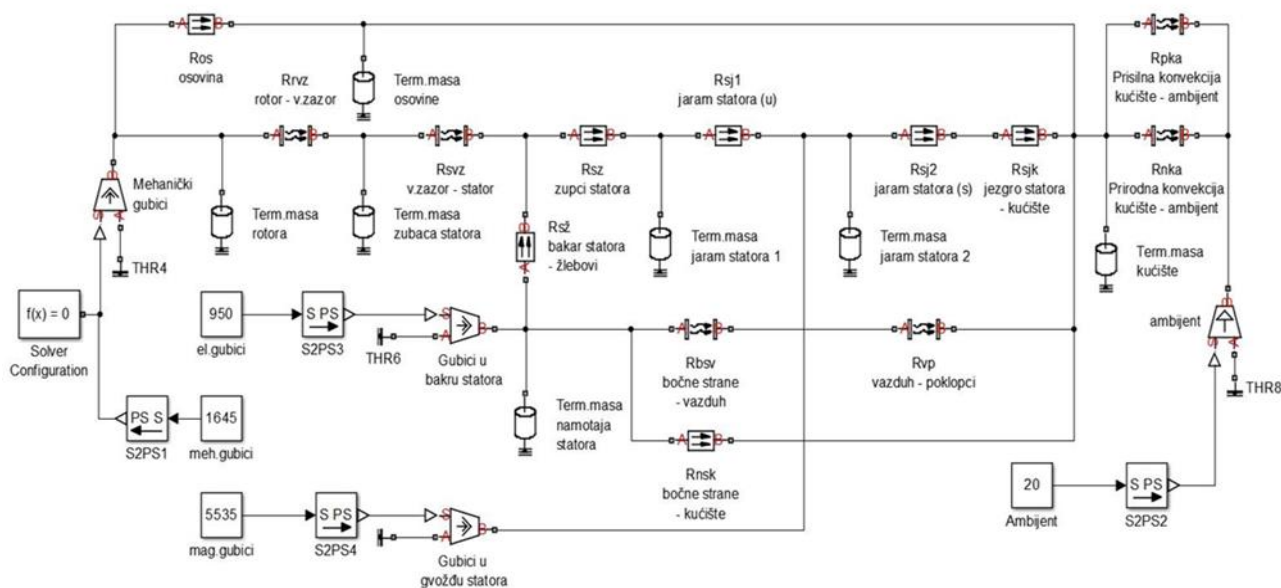
$$R_0 = \frac{T_k - T_a}{P_{DC}} \quad (21)$$

R_{pka} – Prisilna konvekciona termička otpornost između kućišta i vazduha ambijenta

$$R_{pka} = \frac{1}{A_k h_{pk}} \quad (22)$$

$$h_{pk} = \frac{N_{Nu} k_{vz}}{L_{ec}} \quad (23)$$

A_k – ukupna spoljašnja površina kućišta



Slika 1. Ekvivalentno termičko kolo asinhronog motora

Lista upotrebljenih simbola

r_m	– srednja vrednost prečnika jarma statora
r_{iy}	– vrednost unutrašnjeg prečnika jarma statora
r_{oy}	– vrednost spoljašnjeg prečnika jarma statora
k_{Fe}	– termički koeficijent konduktivnosti gvožđa
k_{vz}	– termička provodnost vazduha
$k_{sž}$	– konduktivni koeficijent vazduha i izolacionog materijala u žlebovima, određuje se iz eksperimenta
L_s	– dužina magnetnog jezgra statora
r_{is}	– unutrašnji prečnik statora
l_{vz}	– širina vazdušnog zazora
r_{or}	– spoljašnji prečnik rotora
r_{is}	– unutrašnji prečnik statora
r_{ory}	– spoljašnji prečnik jarma rotora
r_{iry}	– unutrašnji prečnik jarma rotora
L_{ec}	– dužina kućišta
α	– koeficijent redukcije (<1)
t_{sj}	– širina jarma ($t_{sy} = r_{oy} - r_{iy}$)
t_k	– debljina kućišta
t_e	– ekvivalentna debljina vazduha i izolacije u žlebovima statora
l_{kp}	– dužina kontaktne površine, procenjuje se na osnovu eksperimentalnih testova DC napajanjem
S_s	– površina žleba statora
S_{cu}	– površina bakra u žlebu
$l_{ož}$	– ukupni obim žlebova statora
T_k	– temperatura kućišta tokom DC ogleda
T_a	– temperatura ambijenta tokom DC ogleda
P_{DC}	– Gubici tokom DC ogleda
h_{vz}	– koeficijent konvekcije vazdušnog zazora
h_{bs}	– koeficijent konvekcije za vazduh između bočnih strana
h_{pk}	– koeficijent prisilne konvekcije
h_{vp}	– konvekcioni koeficijent između vazduha i poklopaca

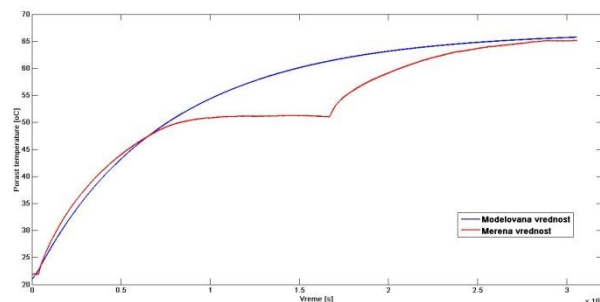
Parametri ($k_{sž}$, l_{kp} i R_0) se mogu dobiti samo putem ogleda DC napajanja. Tokom DC testa, postoje samo gubici u bakru statora.

U ustaljenom režimu se mere temperature namotaja statora, jezgra statora, kućišta statora i ambijenta.

Otpornost R_{nka} se računaju prema datoj formuli, dok se za druga dva parametara rade iterativna rešenja ekvivalentnog kola do postizanja željenog odziva.

4. ZAKLJUČAK

Predloženi model je napravljen u Simulinku upotrebom Simscape termičke biblioteke, slika 1. Model je poslužio za predviđanje temperatura u praznom hodu asinhronog motora od 450 kW. Merenje otpora namotaja obavljeno je mernim uređajem GWINSTEK GOM-802, dok je merenje ambijenta vršeno u tri tačke digitalnim i živim termometrom. Na slici 2, upoređene su krive merenih vrednosti (crvena) i krive predviđenih vrednosti (plava). Najveće neslaganje između temperatura iznosi $\pm 9^\circ\text{C}$ u prelaznom režimu, dok je greška u ustaljenom stanju manja od $\pm 2^\circ\text{C}$. Ovim je potvrđena validnost predloženog termičkog modela.



Slika 2. Rezultati simulacije

5. LITERATURA

- [1] Juha Pyrhonen, Tapani Jokinen, Val'eria Hrabovcova „Design of rotating electrical machines“
- [2] P. Mellor, D. Roberts D. Turner, “Lumped parameter thermal model for electrical machines of TEFC design”, IEE Proceedings – B, Vol.138, NO 5, September 1991.
- [3] G.Dajaku, D.Gerlin, „An Improved Lumped Parameter Thermal Model for Electrical Machines“
- [4] A.Boglietti, A.Cavagnino, M.Lazzari, M.Pastorelli, “A Simplified Thermal Model for Variable-Speed Self-Cooled Industrial Induction Motor”. IEEE Transactions on Industry Application, 2003.
- [5] A.Boglietti, A.Cavagnino, „Analysis of the endwinding cooling effects in TEFC induction motors“, IEEE Transactions on Industry Applications, Vol.43, No.5,Sept-Oct. 2007

Kratka biografija:



Igor Halkić rođen je u Bačkoj Topoli 1991. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine odbranio je 2014.god.

AUTONOMNA VOZILA AUTONOMOUS VEHICLES

Dragan Šeatović, Željko Trpovski, *Fakultet tehničkih nauka, Novi Sad*

Oblast - ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu objašnjena su autonomna vozila, data je klasifikacija autonomije vožnje i osnovna arhitektura ovakvih vozila. Opisani su senzorski sloj autonomnih vozila i komunikacioni standardi.

Abstract – In this paper we have described autonomous vehicles. Classification of driverless capabilities and basic architecture of autonomous vehicles has been presented as well. Sensors layer of these cars and communication standards have been described.

Ključne reči: Autonomna vozila, robotizovana vozila, senzori autonomnih vozila, lidari, internet vozila.

1. UVOD

Tendencije tehnološkog razvoja, kako u svim ostalim granama industrije tako i u automobilskoj, jesu razvoj veštačke inteligencije i robotizacija procesa.

Automobilska industrija se nalazi u fazi početka komercijalizacije autonomnih vozila. Prvi podsticaj razvoju ovakvih vozila dala je, još osamdesetih godina prošlog veka, naravno vojna industrija, dok početkom ovog veka primat preuzimaju kompanije iz automobilske industrije. Kako se tehnološki napredna zapadna autoindustrija trenutno bori i sa rastućom konkurencijom sa dalekog istoka i pronalaženjem alternative korišćenju nafte, trenutno u prvi plan razvoja autonomije vozila dolaze kompanije iz sveta informatike, pre svega Google i Uber. Porast kretanja ljudi u celom svetu dovodi do sve većih gužvi na saobraćajnicama i sve dužeg vremenu utrošenog na upravljanje automobilom.

Porast broja vozila u saobraćaju dovodi i do porasta broja saobraćajnih nezgoda.

Autonomna vozila donose poboljšanja u svim ovim aspektima. Robotizacija vozila doneće promene i u sferu osiguranja, prava i zakona, veštačenja saobraćajnih nezgoda, obuke vozača, smanjiće potrebe za vozačima kamiona, autobusa, taksi vozila...

Masovnije upotrebe autonomnih vozila neće biti u ovoj deceniji, a ostaje da se vidi koliku će primenu doživeti u sledećoj. Ovakva vozila su još uvek preskupa. Zakonska regulativa ih uglavnom ne prepoznaje, izuzev u Americi, Singapuru i nekim zemljama Evropske Unije. Putna infrastruktura i signalizacija nisu dovoljno standardizovane i često su nepotpune, a i svest ostalih učesnika u saobraćaju mora biti prilagođena, posebno pešaka.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Željko Trpovski, vanr. prof.

2. KLASIFIKACIJA AUTONOMNIH VOZILA

Još 2013. godine u SAD-u Agencija za bezbednost saobraćaja klasifikovala je kategorije autonomije motornih vozila. Iako je ova klasifikacija namenjena za tržište SAD-a, danas se primenjuje u celom svetu i postala je globalni relevantni faktor za klasifikaciju autonomije pojedinih vozila. Tako se autonomija vožnje svakog vozila može rasporediti u jednu od pet kategorija [1]:

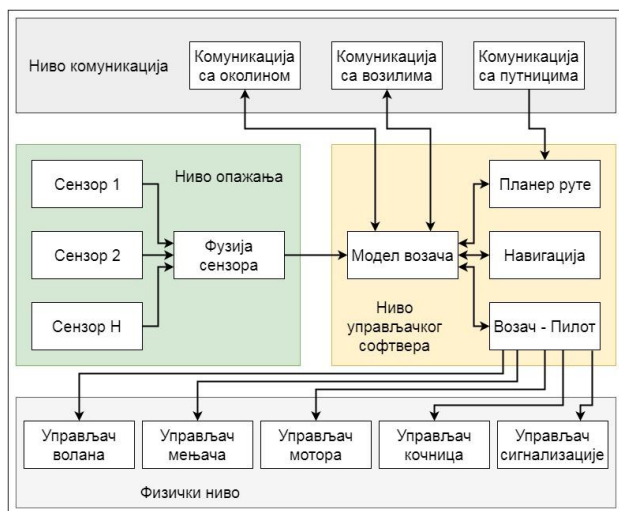
- Kategorija 0 (Neautonomna vozila): Vozač je jedini i u potpunoj kontroli nad primarnim upravljačkim jedinicama vozila: kočnica, upravljača, gasa i pogonske snage u svakom trenutku.
- Kategorija 1 (Autonomija specifičnih funkcija): Autonomija u ovoj klasi uključuje jednu ili više specifičnih upravljačkih funkcija. Takvi primeri uključuju elektronsku kontrolu stabilnosti ili elektronski pomoćni sistem kočenja za uspostavu kontrole nad vozilom.
- Kategorija 2 (Autonomija kombinovanih funkcija): Ova klasa uključuje autonomiju za barem dve primarne funkcije vozila, koje rade kooperativno kako bi vozače oslobodile od upravljanja tim funkcijama. Najčešći primeri su tempomat ili adaptivno prolagođenje brzine vožnje u kombinaciji sa sistemom praćenja (ne prelaska) bele kolovozne linije.
- Kategorija 3 (Kategorija ograničene autonomije): Vozila ovog stepena autonomije sposobna su od vozača preuzeti potpunu kontrolu nad svim sigurnosnim funkcijama vozila u određenim kritičnim trenucima, u takvim situacijama od vozača se očekuje gotovo potpuno oslanjanje na "polu-autonomno" vozilo. Od vozača se međutim očekuje i da bude u mogućnosti za povremenu kontrolu vozila ali s dovoljno vremena za siguran prelaz između modova vožnje.
- Kategorija 4 (Potpuno autonomno vozilo): Vozilo je dizajnirano da sprovodi sve kritične operacije i prati uslove u saobraćaju tokom čitave vožnje. Ovakav oblik vožnje zahteva od vozača unos željene rute ili destinacije, ali ne zahteva od njega nikakve druge sposobnosti za upravljanje vozilom niti u jednom trenutku vožnje. Ovakva vozila sposobna su za saobraćaj i bez prisustva čoveka.

Danas gotovo svi proizvođači automobila u svoja vozila ugrađuju sisteme s najmanje prvom kategorijom autonomije. Vodeće autokompanije isporučuju vozila, posebno ona iz viših kategorija, sa sistemima iz druge kategorije. Dok je ceo svet automobilske industrije fokusiran na izbacivanje na tržište vozila sa bar trećim ali i četvrtim stepenom autonomije vožnje.

3. ARHITEKTURA

Arhitektura autonomnog vozila deli sistem na četiri nivoa:

- Nivo opažanja,
- Nivo upravljačkog softvera,
- Fizički nivo vozila i
- Komunikacioni nivo.



Slika 1. Šema arhitekture autonomnih vozila [2]

Nivo opažanja čine različite vrste senzora koji prikupljaju informacije iz spoljašnjeg sveta, takozvanom fuzijom senzora pomoću odgovarajućeg algoritma. On sledećim informacijama snabdeva upravljački softver [2]:

- Geografski položaj, pravac i smer kretanja,
- Brzinu kretanja,
- Položaj vozila na kolovozu,
- Informacije sa saobraćajne signalizacije (vertikalne i horizontalne),
- Podaci o statičkim fizičkim preprakama (građevine, bandere, ivičnjaci, drveće...),
- Podaci o ostalim učesnicima u saobraćaju (kretanju vozila, pešaka...),
- Podaci o uslovima saobraćaja na izabranoj ruti (gužve, rampe, granice, putarine),
- Podaci o vremenskim uslovima i stanju kolovoza i
- Podaci o stanju vozila (ispravnost, količina goriva...).

Nivo upravljačkog softvera vrši interakciju sa putnicima čime dobija informaciju o željenoj destinaciji. Planira put, obrađuje informacije dobijene od senzora, sklapa sveobuhvatnu sliku okruženja vozila, odlučuje o akcijama koje treba izvršavati i upravlja komponentama fizičkog nivoa.

Hardverski senzori (nivo opažanja) predstavljaju ulaze softvera.

Komunikacioni uređaj u vozilu predstavlja ulazno-izlaznu komponentu softvera.

Izlazne komponente su hardver za upravljanje vozilom (fizički nivo vozila) kao i uređaj za skladištenje podataka i uređaji za komunikaciju sa ostatkom sveta.

Fizički nivo vozila izvršava komande upravljačkog softvera, poznat je i kao reaktivni sloj. U sebi sadrži kontrole upravljanja vozilom, i to sledeće:

- Kontrolu motora
- Kontrolu upravljača
- Kontrolu kočnica
- Kontrolu transmisije
- Kontrolu signalizacije

Komunikacioni nivo služi za razmenu podataka sa putnicima, lokalno čuvanje podataka o vožnji, razmenu informacija sa drugim autonomnim vozilima u blizini, kao i razmenu informacija sa putnom informacionom infrastrukturom.

4. SENZORI

Razvoj autonomnih vozila zavisi kako od razvoja upravljačkog softvera tako i od razvoja hardvera senzora. Senzori predstavljaju spregu između vozila i okruženja, pomoću kojih vozilo postaje svesno okruženja u kom se nalazi.

Postoje pasivni i aktivni senzori. Pasivni senzori su posmatrači okruženja, oni hvataju signale koje generišu drugi izvori u okruženju. Aktivni senzori emituju energiju u okruženje i oslanjaju se na činjenicu da se ta energija reflektuje od okruženja i vraća nazad ka senzoru.

Bilo da su aktivni ili pasivni, senzori autonomnih vozila dele se na tri vrste u zavisnosti šta opažaju [2]:

- Okruženje
- Lokaciju
- Kretanje vozila

Za opažanje okruženja koriste se daljinomeri, senzori koji mere udaljenost vozila od statičkih i pokretnih objekata u okruženju. Prve verzije daljinomera su bili sonarni senzori, oni emituju zvučne talase usmerene ka objektima u okolini, koji se reflektuju i vraćaju do senzora. Kod autonomnih vozila najčešće se koriste ultrazvučni sonari koji imaju mali domet i frekvenciju iznad gornje granice čujnosti za normalno ljudsko uho.

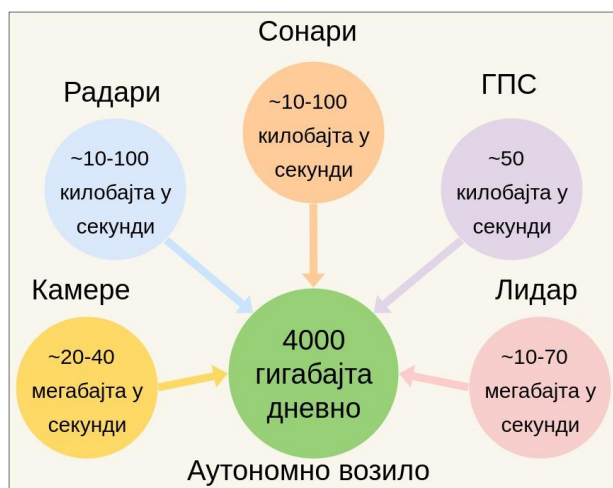
Vreme i jačina povratnog signala određuju udaljenost objekata. Druga vrsta su radari, koji koriste radio talase, oni mogu da mere rastojanje i do više kilometara. Radari se aktivno koriste u autonomnim vozilima, a najveća prednost im je što rade podjednako dobro u svim vremenskim uslovima.

Najnovija i najskuplja vrsta daljinomera su lidari, koji za merenje udaljenosti koriste laserske (svetlosne) zrake, koje se usmeravaju pomoću složeno podešenih ogledala i rotirajućih elemenata. Lidari emituju infracrvenu svetlost koja nije vidljiva ljudskom oku.

Opažanje lokacije se vrši merenjem udaljenosti od određenim poznatih tačaka, kako na zemlji tako i u orbiti. Globalni pozicioni sistem, poznatiji kao GPS, našao je svoju primenu i u autonomnim vozilima. Sateliti emituju pulsirajuće signale a GPS prijemnik analizom pomeraja faze meri udaljenost od satelita.

Triangulacijom signala sa više satelita, prijemnici utvrđuju svoju apsolutnu lokaciju na zemlji, sa

preciznošću od nekoliko metara. Da bi se dobila preciznija lokacija koristi se drugi zemaljski GPS prijemnik sa poznatom lokacijom, pa se pod idealnim uslovima dobija milimetarska preciznost. Autonomna vozila su opremljena i elektronskim žiroskopima, koji se koriste za određivanje lokacije kada GPS nije dostupan. Proprioceptivni senzori opažaju kretanje samog vozila. Najbitniji senzor ove klase je odometar. Brojeći obrtaje točkova, odometar se koristi za merenje pređenog puta. Broj obrtaja pomnožen sa obimom pneumatika daje pređeni put. Pređeni put podeljen sa utrošenim vremenom daje brzinu kretanja.



Slika 2. Dotok podataka sa senzora [3]

5. LIDARI

Lidar (eng: *Light Detection And Ranging*) kao uređaj za opažanje okruženja koristi se na skoro svim modelima autonomnih vozila treće i četvrte kategorije autonomije vožnje. Postoje dve osnovne varijante:

- Rotirajući sistem sa jednim fizičkim senzorom koji je postavljen na krovu automobila tako da ima pogled na sve strane. Glava senzora se rotira i opažajući 360° horizontalno stvara 3D sliku okruženja.
- Statički sistem sa više fizičkih senzora postavljenih na svim stranama automobila. Ovi senzori zajedno opažajući horizontalno 360° stvaraju 3D sliku okruženja.

Pri pogodnim vremenskim uslovima lidari imaju veliku preciznost i veliki domet, a prednost je i što detektuju sve vrste pokretnih i statičkih prepreka, automobile, bicikle, ljude, bandere, drveće, žbunje... Ali im se preciznost značajno smanjuje pri otežanim vremenskim uslovima poput snega, kiše, magle...

Neki primeri lidara sa navedenim karakteristikama:

- HDL-64E kompanije *Velodyne* je rotirajući lidar, koji na rotirajućoj glavi ima vertikalno postavljene 64 emitujuće laserske glave i 64 prijemne ploče. Stvara horizontalni vidokrug od 360° i vertikalni od 26.9° (od +2° do -24.9°). Snima oko 2 miliona tačaka u sekundi, ima domet od 120 metara i preciznost od 2 centimetra. Troši 60W i kači se konektorima RS-232 i RJ-45. Uređaj je prilično veliki, 30x30x30 centimetara i masu od 13 kilograma. Veoma je skup, košta oko 9 miliona dinara.

- VLP-16 kompanije *Velodyne* je noviji rotirajući model lidara, najveći napredak napravljen je u veličini i ceni, dimezija 10x10x10 centimetara i ima manje od jednog kilograma. Košta samo 900 hiljada dinara. Pored toga novi model je naizgled statičan, tačnije nema vidljivi rotirajući deo, rotiranje glave se obavlja unutar uređaja. Opremljen je sa 16 emitujućih i 16 prijemnih stakala, vertikalno hvata do 30° (od +15° do -15°) a horizontalno naravno 360°. Ima snagu od 8 vati i kači se preko konektora M12. U poređenju sa starijim modelom hvata manje tačaka, oko 300 hiljada u sekundi, i ima manju preciznost, 3 centimetra i domet od 100 metara. Iako su performanse umanjene u odnosu na HDL-64E, VLP-16 i dalje može da obezbedi dovoljno informacija da se kreira zadovoljavajuća 3D slika okruženja.



Slika 3. Lidari kompanije *Velodyne*

- *Velarray* kompanije *Velodyne* je najnoviji statički lidar sa fiksno pozicioniranim laserskim prijemnicima i predajnicima, snima horizontalno 120°, dimenzije 12,5x5x5 centimetara. Kada uđe u masovnu proizvodnju koštaće nekoliko desetina hiljada dinara. Po jedan ovakav lidar sa svake strane vozila bio bi dovoljan za stvaranje kompletne 3D slike okruženja.
- *Ibeo Lux Fusion* sistem se sastoji od 6 laserskih skenera koji su povezani preko centralne jedinice. Tri skenera se postavljaju na prednji branik a tri na zadnji. Svaki skener zasebno emituje svetlosne signale i povratne informacije šalje centralnoj jedinici, gde se stvara 3D slika okruženja od svih 360° oko vozila. Domet svako senzora je 100 metara i pouzdano radi u skoro svim vremenskim uslovima. Cena svakog pojedinačnog senzora je oko 50 hiljada dinara.



Slika 4. Jedan od senzora *Ibeo Lux Fusion* sistema

6. KOMUNIKACIJE

Autonomna vozila moraju da vrše nekoliko oblika komunikacije. Primaju direktna uputstva od putnika ili vlasnika o željenoj destinaciji, komuniciraju sa drugim vozilima u okruženju i razmenjuju informacije sa naprednim sistemima saobraćajne signalizacije.

Komunikacija sa putnicima ne zahteva specifične tehnologije, ekrani osetljivi na dodir, mikrofoni i zvučnici u paru sa analizom i sintezom govora dovoljni su za odličnu komunikaciju sa putnicima. Sve ove tehnologije su dovoljno razvijene da mogu da uđu u masovnu proizvodnju.

S druge strane komunikacija dva autonomna vozila je u svom začetku. V2V (eng: *Vehicle-to-Vehicle*) komunikacija je u aktivnom razvoju kod više vodećih proizvođača automobila. Danas postoji i konzorcijum proizvođača pod nazivom *Car-to-Car*. Komunikacija između klasičnih vozila se obavlja uglavnom preko svetlosnih signala, kao što su stop svetla i pokazivači pravca, ili pomoću zvučne sirene. Od V2V komunikacije se očekuje da svetlosne signale pretvori u digitalne informacije koje će se bežičnom vezom razmenjivati među vozilima. Za ovo je potrebna komunikacija kratkog dometa, visoke fleksibilnosti, brzog uspostavljanja i kratkog trajanja.

Isti princip potreban je i za komunikaciju vozila sa infrastrukturom, poznatu i kao V2I (eng: *Vehicle-to-Infrastructure*). Od nje se očekuje da dopuni i zameni jednosmernu komunikaciju saobraćajne signalizacije prema vozačima automobila. Rezultat integracije svih vozila u veliku mrežu je VANET (eng: *Vehicular Ad-hoc Network*) i Internet vozila (eng: *IoV - Internet of Vehicles*).

Za V2V i V2I komunikaciju definisan je set standarda WAVE (eng: *Wireless Access in a Vehicular Environment*). Familija standarda IEEE 1609 za WAVE, definiše bezbednu mobilnu bežičnu radio komunikaciju, kratkog dometa a visoke brzine i malog kašnjenja. Frekvencija od 5.9 GHz, širina kanala 10 MHz i brzina od 3 do 27 Mb u sekundi. Fizički sloj ovog standarda baziran je na IEEE 802.11 (što je standard za bežični internet kratkog dometa), ali je zbog specifičnosti okruženja u kom se odbija komunikacija napravljen amandman IEEE 802.11p. Osnovna izmena je što se zbog kratkotrajnosti veze, uređaju koji se tek kači na mrežu dozvoljava da šalje podatke i bez prethodne autentifikacije i autorizacije. Time se odbacuju načela privatnosti i autentifikacije iz IEEE 802.11, a ti problemi se rešavaju na višim slojevima komunikacije. Sloj veze je definisan sa IEEE 1609.4, dok su mrežni i transportni sloj definisani u IEEE 1609.3. Bezbednost se definiše u IEEE 1609.2 [3].

Koristeći WAVE u idealnim uslovima sa najvećom brzinom, vozilo može dnevno da isporuči 300 GB podataka, dok samo senzori vozila mogu da proizvedu oko 4000 GB, od kojih najviše lidari i kamere.

Bežične komunikacije pete generacije 5G treba da reše ovaj problem, smanje kašnjenje i povećaju protok. Na konferenciji 2015. godine za V2V i V2I komunikaciju određena je frekvencija od 28 GHz, ali sa tri podkanala. Donji kanal koristi opseg od 694 do 790 MHz, ima veliki domet i pouzdanost rada u svim uslovima okruženja. Srednji kanal ima frekvenciju od 3.4 do 3.87 GHz. Ima slične performanse kao i WiFi, a i frekventijski je pozicioniran između varijanti bežičnog interneta na 2.8

GHz i 5 GHz. Poslednji visokofrekventijski kanal od 25 do 27.5 GHz treba da obezbedi brzinu prenosa do 10 GB u sekundi i veliki broj istovremenih konekcija, što je vrlo bitno za bazne stanice u blizini saobraćajnih gužvi.

7. ZAKLJUČAK

Autonomna vozila postaju realnost, prvo u vidovima teretnog i javnog saobraćaja, pošto preduzeća uvek imaju najviše interesa da ulažu u moderne tehnologije. A odmah zatim stižu i kao privatna vozila.

Neadekvatna i neusaglašena zakonska regulativa kao i putna infrastruktura usporavaju masovnije testiranje i realnu primenu autonomnih vozila.

Cena inovativnih komponenti koje se koriste kao komponente pametnih vozila, posebno lidara, poslednjih godina je drastično snižena.

Kada te komponente uđu u masovnu proizvodnju cena će im biti još niža, čime će autonomna vozila postati dostupna većem broju kupaca.

Neizbežan je gubitak vozača kao radnih mesta u sektoru saobraćaja, s druge strane sigurno će se otvoriti dodatna administrativna i nadzorna radna mesta, ali nedovoljno da pokriju smanjenje potrebnog broja vozača. Broj potrebnih vozača će se postepeno smanjivati što će ostaviti vremena za prekvalifikacije.

Pored smanjenja vremena koje će ljudi trošiti na upravljanje automobilima, autonomna vozila će najviše doprineti smanjenju broja saobraćajnih nezgoda kao i žrtava u njima, a omogućiće i bolju samostalnu mobilnost ljudi sa invaliditetom.

8. LITERATURA

- [1] Alen Pavlič, *Taksonomija motornih vozila*, 2016. godine
- [2] Miloš Kajtez, *Autonomna vozila: početak i primena robotizovanih vozila u saobraćaju*, 2014. godine
- [3] Shereen A. M. Ahmed, *Overview of Wireless Access in Vehicular Environment (WAVE) Protocols and Standards*, 2013. godine

Kratka biografija:



Dragan Šeatović rođen je u Sarajevu 1988. godine. Diplomski-master rad odbranio je 2017. godine na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Telekomunikacije



Željko Trpovski rođen je u Rijeci 1957. godine. Doktorirao je na Fakultetu tehničkih nauka 1998. god. Oblast interesovanja su telekomunikacije i obrada signala.

ŠIFARSKI SISTEM ZA RAZMENU SMS PORUKA SMS ENCRYPTION SYSTEM

Milica Vulić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu analizirane su mobilne aplikacije za razmenu šifrovanih poruka. Na osnovu analize su specificirani zahtevi i dizajn mobilne aplikacije za razmenu otvorenih i šifrovanih SMS poruka. Specificirana aplikacije implementirana je i testirana na Android i Google App Engine platformama. Prilikom dizajna i implementacije aplikacije korišćen je asimetrični šifarski sistem.

Abstract – This paper analyses mobile apps used for sending and receiving encrypted messages. The analysis was used to specify requirements and design of a mobile app for sending and receiving open and encrypted SMS messages. The specified app was implemented and tested using Android and Google App Engine platforms. Asymmetrical encryption system was used during the design and the implementation of the app.

Cljučne reči: mobilne aplikacije, SMS, kriptozastita

1. UVOD

Danas je teško zamisliti život bez modernih informaciono-komunikacionih sredstava. Do vesti sve ređe dolazimo putem novina, a sve češće putem internet stranica. Transfer novca je brži, lakši i jeftiniji na telefonu nego na šalteru banke. Pisma polako ulaze u istoriju, a SMS i druge poruke su preuzele njihovu ulogu.

Iako jednostavnije, dostupnije i brže od klasičnog načina komunikacije, moderne komunikacione tehnologije nisu savršene. Njihova velika mana je nedostatak bezbednosti koji sa sobom donosi razne izazove i pretnje. To može prouzrokovati nepoverenje korisnika u ove tehnologije i biti velika prepreka za njegovu upotrebu.

Tokom svog rada u kriptozastiti primetila sam da je dosta toga urađeno na polju zaštite tekstualnih poruka koje se šalju putem web-a, a da faktički ne postoje aplikacije koje bi pokrile zaštitu tekstualnih poruka koje se mogu razmeniti tamo gde internet nije dostupan i gde se mogu koristiti samo SMS poruke. Pored toga, jedan od glavnih problema u kriptozastiti jeste razmena ključeva. Ovaj problem se ublažava asimetričnom kriptografijom i pojavom javnog ključa koji može svima bude dostupan.

2. SRODNA REŠENJA

Analiza prednosti i mana postojećih softverskih rešenja je poslužila za analizu i specifikaciju zahteva sistema razvijenog u ovom radu. Postoji dosta rešenja koja vrše kriptovanje poruka, ali su uglavnom vezani za web servise, a to ne pokriva slučajevne kada je internet nedostupan.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio prof. dr Stevan Gostojić.

Kroz praktičan pristup istraživanju opisano je nekoliko rešenja kao što su *Confide*, *Telegram*, *Signal*, *Wickr* i *WhatsApp* i navedene su njihove prednosti i mane.

2.1 Telegram

Telegram je aplikacija koja je bazirana na cloud-u i dozvoljava pristupanje sa više uređaja. Mogu se slati poruke i fajlovi bilo kog tipa (.doc, .zip, .mp3, itd.). Omogućava kreiranje grupa za do 5.000 korisnika, kao i kanala za objavljivanje neograničenom auditorijumu. Mogu se kontaktirati postojeći kontakti iz imenika koji imaju instaliranu aplikaciju, kao i korisnici čije je korisničko ime poznato.

2.2 WhatsApp

WhatsApp je trenutno najpopularnija aplikacija za razmenu poruka na svetu. Posедуje opciju za kreiranje grupa i slanje multimedijalnog sadržaja. Moguće je obavljati pozive, kako audio, tako i video tipa. Komunikacija je moguća samo sa onim korisnicima koji poseduju instaliranu aplikaciju.

Od aprila 2016. godine ima integrisan *Signal* protokol. Enkriptuje sav sadržaj (poruke, multimedijalni sadržaj i pozive). Prikuplja podatke iz imenika ako korisnik to odobri pri instalaciji. Ne čuva korisnički ključ na servisu, tako da na zahtev organa državne bezbednosti u bilo kojoj zemlji ne može predati otvoren sadržaj već samo enkriptovane podatke.

2.3 Signal

Signal podržava rad sa standardnim SMS-ovima i MMS-ovima. Pruža korisnicima mogućnost pozivanja, gde su pozivi takođe enkriptovani. Kako bi se poslao kriptovan podatak, potrebno je biti povezan na internet. Pored ovih opcija, podržava i rad sa grupama. Ono u čemu se posebno ističe ova aplikacija jeste to što je otvorenog koda. To znači da programeri širom sveta mogu doprinosti razvoju i unapređivanju aplikacije. Onemogućen je *screenshot*, čime se postiže dodatna zaštita druge strane od kompromitovanja.

Signal upotrebljava *end-to-end* enkripciju i ne čuva metapodatke, što je još jedan nivo bitan za bezbednost korisnika na mreži. Kriptograf i osnivač *Open Whisper System*-a, *Moxie Marlinspike*, tvrdi da je jedino što server čuva o korisniku podatak kad je korisnik poslednji put bio konektovan na server, a preciznost ove informacije je svedena sa tačnog vremena samo na dan.

2.4 Wickr

Wickr podržava kako standardno slanje poruka, tako i kreiranje grupa i grupno časkanje u kome broj učesnika može doseći 10. Svako poruci može se podesiti «rok trajanja» posle koga se ona automatski briše iz memorije uređaja.

Enkripcije koje koristi su: *AES-256*, *RSA 4096*, *ECDH512* i *TLS*. Tvrdi da njihove zaštićene komunikacije ispunjavaju standard koji je preporučila *NSA* (*National Security Agency*) za komunikacije na nivou državne tajne (*top secret*). *Wickr* briše sve metapodatke sa svog servera, a njegov *secure file shredder* uklanja podatke iz *RAM* memorije nakon otvaranja slike ili poruke.

2.5 Confide

Confide omogućava slanje enkriptovanog sadržaja između krajnjih korisnika koji imaju instaliranu aplikaciju i pristup internetu. U slučaju da primalac nema instaliranu aplikaciju, *Confide* će mu poslati poziv za instaliranje putem *SMS*-a ili *mail*-a.

Tek pošto je instalira, primalac će biti u mogućnosti da dekriptuje poruku. Primalac može da vidi poruku samo jednom. Pošto je pročitao, automatski se briše, dok na strani pošiljaoca ostaje ako je tako specificirano u podešavanjima. Specifično je što se poruka ili slika otkriva prevlačenjem prsta preko ekrana. Podržava razmenu fotografija, dokumenata i govornih poruka.

3. KORIŠĆENE TEHNOLOGIJE

3.1. Android

Android je operativni sistem kompanije *Google* prevashodno namenjen za uređaje koji imaju ekrane osjetljive na dodir. *Android* operativni sistem je otvorenog koda i dostupan je pod *Apache* licencom. Time je data sloboda programerima, proizvođačima uređaja i telekomunikacionim operaterima za slobodnu izmenu i distribuciju softvera.

Tako je nastala velika zajednica programera, a istraživanja su pokazala da je *Android* najpopularnija platforma za razvoj softvera za mobilne uređaje, sa zastupljenošću od preko 80 procenata.

3.2 GreenDAO

GreenDAO je object-relational mapping okvir za *SQLite* bazu podataka. Kreiran je kako bi programere oslobodio rada sa *low-level* zahtevima baze i time skratio vreme neophodno za razvoj aplikacija.

SQLite baza podataka je ugrađena u *Android* operativni sistem. Kako je pisanje *SQL* upita i parsiranje rezultata uglavnom jednoličan posao i najčešće oduzima mnogo vremena, *GreenDAO* se nametnuo kao dobar izbor za automatizaciju ovog procesa. *GreenDAO* mapira *Java* objekte na vrste u tabelama u bazi podataka koristeći pritom tehniku *object-relational mapping*.

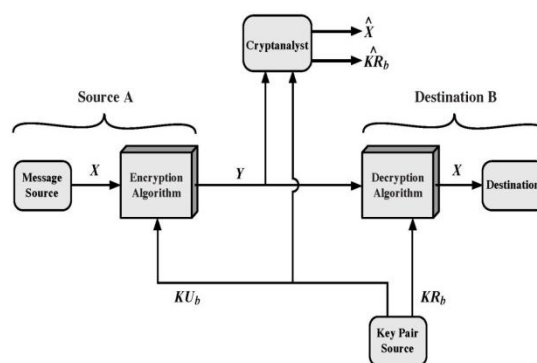
Ovom tehnikom se *Java* objekti ažuriraju, brišu, čuvaju ili kreiraju. Takođe se mogu pisati upiti koristeći objektno orijentisani *API*.

3.3 Google App Engine

Google App Engine sastoji se od seta *API*-ja koji su razvijeni od strane *Google*-a i obezbeđuju funkcionalnosti kao što su analitika, mašinsko učenje i pristup korisničkim podacima (u slučaju da je data dozvola za čitanje podataka). Pripadaju grupi servisa u oblaku, tačnije *PaaS* (*Platform as a Service*). Ovakav vid servisa korisniku nudi platformu za razvoj koja oslobađa korisnika uspostavljanja i održavanja servera i ostale infrastrukture.

3.4. Šifarski sistemi sa javnim kručevima

U asimetričnim šifarskim sistemima koriste se različiti ključevi za šifrovanje i dešifrovanje, tzv. javni i tajni ključ, tako da ključ za šifrovanje može imati svako, a samo posednik ključa za dešifrovanje može dešifrovati poruku. Međutim, visoka računarska zahtevnost ovih algoritama utiče na performanse sistema u kojima se primenjuju, tako da se ne preporučuje primena za zaštitu tajnosti informacija u sistemima sa velikim protokom informacija. Ovo naravno ne isključuje automatski ove algoritme, jer način na koji je, uz korišćenje ovakvih algoritama, moguće ostvariti funkcije integriteta, autentičnosti i neporicanja, ima nesumnjivu prednost nad tradicionalnim tehnikama. *RSA* algoritam [1], jedan od algoritama za šifrovanje i dešifrovanje javnim ključem, već više od dvadeset godina odoleva svim teorijskim i tehnološkim napadima. Opis i način upotrebe ovog algoritma propisani su u standardu *PKCS#1* verzija 2 [2].



Slika 1. Sistem kriptozastite sa javnim ključem

Postupak kriptovanja i dekriptovanja ilustrovan je na slici 1. U objašnjenju je korišćenja sledeća notacija :

- KUb označava javni ključ korisnika B,
- KRb označava tajni ključ korisnika B,
- $EKUb$ označava postupak šifrovanja javnim ključem korisnika B,
- $EKRb$ označava postupak šifrovanja tajnim ključem korisnika B,
- $DKUb$ označava postupak dešifrovanja javnim ključem korisnika B,
- $DKRb$ označava postupak dešifrovanja tajnim ključem korisnika B.

Ako se poštuje gorenavedena notacija, a sa X označimo otvoreni tekst, možemo da kažemo da važe sledeće relacije:

$$DKRb(EKUb(X)) = EKUb(DKRb(X)) = X$$

$$DKUb(EKRb(X)) = EKRb(DKUb(X)) = X,$$

Primalac kome je namenjena poruka, ukoliko raspolaže odgovarajućim parom ključeva, u mogućnosti je da invertuje datu transformaciju. Na prijemnoj strani, strana B algoritmom dešifrovanja poruke i primenom svog tajnog ključa dešifruje dobijenu poruku i dobija otvoreni tekst koji je poslala strana A.

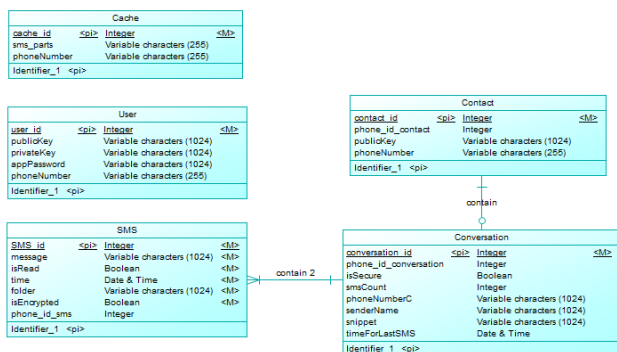
4. SPECIFIKACIJA ZAHTEVA

Na dijagramu slučajeva korišćenja (slika 2) su opisane akcije koje može da preduzme korisnik. Budući da se radi o aplikaciji za razmenu SMS-ova, većina akcija je vezana za SMS poruke. Pri pokretanju aplikacije, ali i po povratku u nju, korisnik obavezno mora da se prijavi. Nakon što se prijavio, dobija dozvolu za pristup drugim akcijama. Na dijagramu je prikazano da korisnik može napisati i pročitati regularnu SMS poruku i da su ovi slučajevi korišćenja prošireni ako korisnik želi da pošalje kriptovanu SMS poruku. Korisnik takođe može da obriše poruku i celu konverzaciju.



Slika 2. Dijagram slučajeva korišćenja

5. SPECIFIKACIJA DIZAJNA



Slika 3. Model podataka

Na dijagramu entiteta i veza (slika 3) prikazani su entiteti koji postoje u bazi podataka. Tri entiteta koja već postoje u matičnoj bazi telefona (*Contact*, *SMS* i *Conversation*) postoje i u ovoj aplikaciji ali u izmenjenom obliku. Dodata su još dva entiteta: *Cache* i *User*.

Entiteti slični onima koji već postoje u bazi telefona imaju dodato polje koje počinje sa *phone_id*, a koje služi za izmenu podataka koji su učitani iz baze telefona kako u ovoj bazi podataka, tako i u bazi podataka telefona.

Entitet *Contact* ima polje *contact_id* za jednodzračnu identifikaciju, polje *phone_id_contact* za komunikaciju sa matičnom bazom telefona, *phoneNumber* polje koje čuva broj telefona korisnika i polje *publicKey* gde se nalazi javni ključ za odgovarajući broj telefona.

Entitet *SMS* sadrži identifikujuće polje *SMS_id*, zatim polje *message* za sadržinu poruke i *isRead* polje koje daje podatak o tome da li je taj SMS pročitao. Folder kojem pripada SMS je opisan poljem *folder*, podatak o tome da li je SMS enkriptovan ili ne nalazi se u polju *isEncrypted*, dok se za komunikaciju sa postojećom bazom koristi *phone_id_sms*.

Entitet *Conversation* ima identifikujuće polje *conversation_id*, *phone_id_conversation* za vezu na postojeće konverzacije iz baze podataka telefona, polje *isSecure* za proveru tipa konverzacije, *smsCount* za prikaz broja poruka koje postoje u konverzaciji i *phoneNumberC* koji je redundantno polje, ali je tu stavljeno radi bržeg odziva glavne aktivnosti (smanjenje broja spajanja tabela kako ne bi došlo do prekida UI niti). Pored ovih, nalaze se i polja *senderName* za predstavljanje imena sagovornika, *snippet* za skraćeni prikaz poslednje pristigle/poslate poruke i *timeForLastSMS* koje daje podatak o tačnom vremenu i datumu poslednje poruke u konverzaciji.

Entitet *Cache* je osmišljen kao rešenje za problem prihvata pristiglih kriptovanih SMS poruka. Naime, one su mnogo duže od standardnog SMS-a. Potrebno je da se kriptovana poruka prikupi u delovima, spoji u jednu poruku, dekriptuje i zatim prikaže korisniku. Ova tabela sadrži polje *cache_id* koje je jedinstveni identifikator, polje *sms_parts* za delove poruke i polje *phoneNumber* za broj pošiljaoca koji je poslao kriptovanu poruku.

Entitet *User* opisuje korisnika aplikacije, i ima identifikujuće polje *user_id*, polje za javni ključ *publicKey*, za privatni ključ *privateKey*, polje za čuvanje lozinke za prijavu na sistem *appPassword* i *phoneNumber* za čuvanje sopstvenog broja telefona.

6. IMPLEMENTACIJA APLIKACIJE

U ovom radu je najviše pažnje posvećeno kriptozastiti SMS poruka, pa je u ovom odeljku opisano kako se prihvataju zaštićene SMS poruke kao jedna od karakterističnih metoda aplikacije (pored metoda enkripcije, dekripcije i razmene ključeva).

```
if (bundle != null) {
    // Retrieve the Binary SMS data
    Object[] pdus = (Object[]) bundle.get("pdus");
    msgs = new SmsMessage[pdus.length];
    // As many pdus are there, as much SMS
    messages will be created

    // For every SMS message received (although
    multipart is not supported with binary)
    for (int i=0; i<msgs.length; i++) {
        byte[] data = null;
        msgs[i] = SmsMessage.createFromPdu((byte[])
        pdus[i]);

        str += "Binary SMS from " +
        msgs[i].getOriginatingAddress() + " :";
        phoneNumber =
        msgs[i].getOriginatingAddress();

        str += "\nBINARY MESSAGE: ";

        // Return the User Data section minus the
        // User Data Header (UDH) (if there is any
        UDH at all)

        data = msgs[i].getUserData();
        for (int index=0; index < data.length;
        index++) {
```



```

        str += Byte.toString(data[index]);
    }

    str += "\nTEXT MESSAGE (FROM BINARY): ";
    for (int index=0; index < data.length;
    index++) {
        str += Character.toString((char)
    data[index]);
        smsContent += Character.toString((char)
    data[index]);
    }
    str += "\n";
}

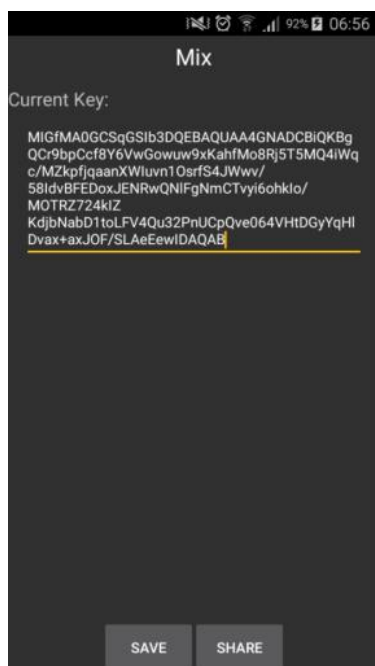
```

Listing 1 – Prijem zaštićene poruke u SMSReceiver2

U listingu 1 prikazan je prijemnik poruka SMSReceiver2 koji se upotrebljava za prihvatanje zaštićenih poruka koje se kasnije obrađuju. Prijemnik poruka postoji samo u toku izvršavanja *onReceive*. Tekst zaštićenog SMS-a takođe stiže u obliku niza *PDU*, koje se izdvajaju iz originalne poruke i konvertuju u tip podatka String. Osim teksta, sa porukom se dobijaju i detalji pošiljaoca (kao što je broj telefona). Ta dva podatka se stavljaju u *intent*, a zatim se polje *isSecure* podešava na vrednost *true* kako bi se kasnije znalo da se radi o SMS-u čijoj obradi se mora pristupiti na poseban način.

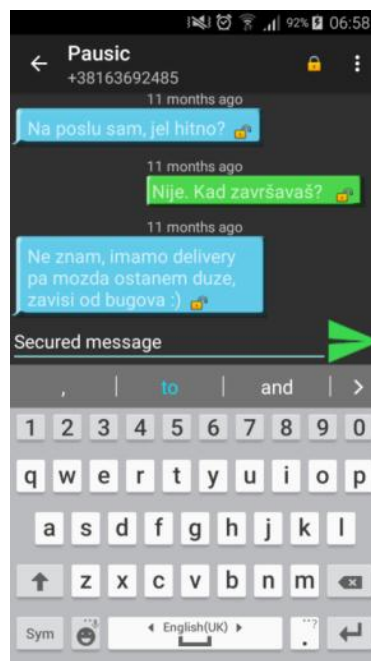
7. DEMONSTRACIJA

Unutar rada su demonstrirane glavne funkcije implemen-
tiranog softverskog rešenja poput slanja i primanja krypto-
vanih i otvorenih SMS poruka i razmena ključeva sa servisa.



Slika 4. Prikaz aktivnosti Public Key

Na ovom ekranu je prikazan javni ključ kontakta. On se može ručno izmeniti ulaskom u način rada za izmenu, a takođe se može i podeliti pomoću aplikacija koje podržavaju deljenje teksta. Ovo je naročito korisno kada ne postoji internet konekcija i ne izvrši se automatsko osvežavanje baze javnih ključeva ili kada korisnik ne želi da koristi online bazu ključeva (slika 4).



Slika 5. Prikaz kriptovane konverzacije u Conversation Activity

Na slici 5 prikazana je zaštićena konverzacija koja se može prebaciti u nezaštićeni režim rada pritiskom na katanac u gornjem desnom uglu. Korisnik sa ovog ekrana pritiskom na tri tačkice u gornjem desnom uglu može ući u podešavanja aplikacije. Korisniku je takođe omogućeno da primenom dugog pritiska na poruku odabere akciju brisanja, prosleđivanja, kopiranja ili prikaza detalja.

8. ZAKLJUČAK

Sa obzirom na to da rešenje prikazano u ovom radu omogućava bezbednu komunikaciju SMS porukama, akcenat je na stepen sigurnosti koji se može ostvariti. Sa ovog aspekta nije moguće ustanoviti da li je došlo do tzv. *Man-In-The-Middle* [3] napada koji bi se mogao dogoditi prilikom sinhronizacije ključeva. Jedno od potencijalnih rešenja bi bilo uvođenje sertifikacionog tela koje bi moglo da izdaje i verifikuje ključeve. U krajnjem slučaju, moguće ih je i ručno razmeniti unutar aplikacije.

9. LITERATURA

- [1] The RSA Cryptosystem – Stanford University, <https://crypto.stanford.edu>.
- [2] Diffie-Hellman key exchange and pairing – Princeton, <https://www.cs.princeton.edu>
- [3] Man-in-the-middle attack – OWASP, https://www.owasp.org/index.php/Man-in-the-middle_attack

Kratka biografija:

Milica Vulić rođena je u Novom Sadu 1991. godine. Master rad na Fakultetu tehničkih nauka iz oblasti elektrotehnika i računarstva – softversko inženjerstvo odbranila je 2017. Godine.

ANDROIDANNOTATIONS OKVIR ZA RAZVOJ MOBILNIH APLIKACIJA ANDROIDANNOTATION MOBILE APP DEVELOPMENT FRAMEWORK

Danijel Popović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu analiziran je AndroidAnnotations okvir za razvoj Android aplikacija. Implementirane su dve Android aplikacije po jednoj specifikaciji zahteva i dizajna: jedna korišćenjem AndroidAnnotations okvira i druga bez korišćenja AndroidAnnotations okvira. Komparativnom analizom razvoja ovih aplikacija ustanovljene su prednosti i mane AndroidAnnotations okvira za razvoj Android aplikacija.

Abstract – This paper analyzes AndroidAnnotations framework for Android app development. Two Android apps were developed according to one requirements and design specification: an app that uses AndroidAnnotations framework and an app that doesn't use AndroidAnnotations framework. Comparative analysis of development of those apps was used to assess pros and cons of AndroidAnnotations framework for Android app development.

Ključne reči: mobilne aplikacije, Android, razvojni okviri, AndroidAnnotations

1. UVOD

Potreba tržišta za sve većim brojem mobilnih aplikacija stvorila je potrebu za unapređenjem procesa razvoja mobilnih aplikacija. Jedan način za povećanje efikasnosti ovog procesa je korišćenje okvira za razvoj mobilnih aplikacija koji omogućavaju programerima da rade na višem nivou apstrakcije i da izbegnu pisanje boilerplate koda.

U ovom radu je analiziran AndroidAnnotations [1] okvir. MovieMaster aplikacija je implementirana na Android platformi bez korišćenja AndroidAnnotations okvira i sa korišćenjem AndroidAnnotations okvira. Aplikacije su poslužile kao studija slučaja na osnovu koje su analizirane prednosti i mane AndroidAnnotations okvira.

2. ANDROIDANNOTATIONS

AndroidAnnotations je razvojni okvir otvorenog koda koji ubrzava razvoj Android aplikacija tako što omogućava programerima da izbegnu pisanje boilerplate koda i da se skoncentrišu na implementaciju poslovne logike aplikacije. Radi tako što uvodi još jedan korak u prevođenju Android [2] aplikacija u kome se na osnovu standardnih Java [3] anotacija za svaku anotiranu klasu generiše po jedna klasa u kojoj se automatski generiše boilerplate kod. Koristi tehniku ubrizgavanja zavisnosti, što kod čini preglednijim, bolje organizovanim i lakšim za održavanje.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Gostojić, vanr. prof.

Korišćenjem ove tehnike u ovom okviru mogu se ubrizgati pogledi, događaji i resursi uz korišćenje anotacija @ViewById, @AfterViews, @AfterInject, @Click, @LongClick, @Touch, @EActivity, @EBean, @RootContext, @EView i @EFragment.

3. ANALIZA ZAHTEVA

Kao studija slučaja za upotrebu AndroidAnnotations okvira implementirana je mobilna aplikacija za pregled filmova. Slika 3.1 prikazuje dijagram slučajeva korišćenja aplikacije.

4. DIZAJN APLIKACIJE

Na osnovu specificiranih zahteva je dizajnirana aplikacija. Specifikacija dizajna opisana je dijagramom klasa, i dijagramom rasporeda. Dijagram klasa i dijagram rasporeda opisuju statički model sistema.

4.1 Dijagram klasa

Model podataka (slika 4.1) sastoji se od klasa: Movie (film), MovieDetails, Category (kategorija), Comment (komentar), Crew (ekipa), Sync, User (korisnik), MovieCategory i MovieCrew. Film može da ima više komentara. Jedan komentar može da ima samo jedan film. Jedan film može da bude u više kategorija i podkategorija. Jedna kategorija ili podkategorija može da ima više filmova. Jedan film može da ima više glumaca, a jedan glumac može da glumi u više filmova. Svaka od ovih klasa ima svoj jedinstveni identifikator i skup potrebnih atributa.

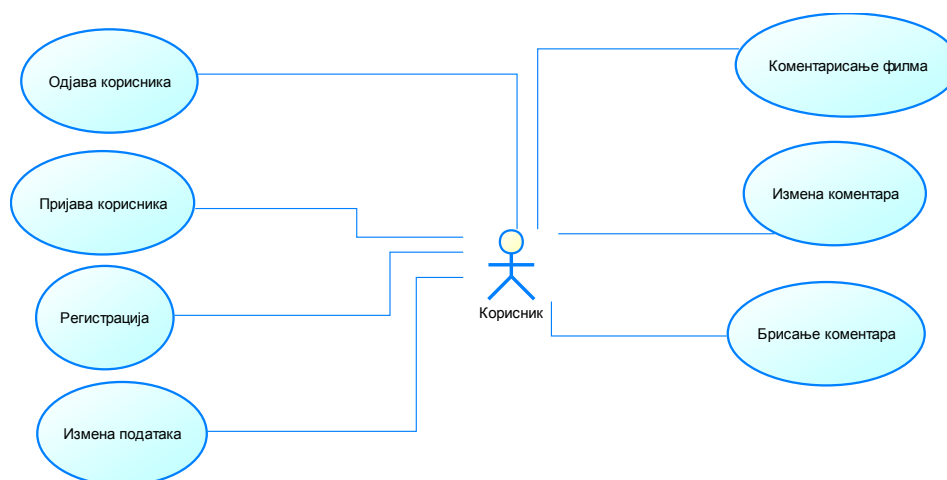
4.2 Dijagram rasporeda

Dijagram rasporeda (slika 4.2) prikazuje raspored komponenti aplikacije po čvorovima. Glavni čvor na dijagramu je mobilni uređaj. Taj čvor sadrži čvor operativni sistem, a čvor operativni sistem sadrži čvor mobilna aplikacija. Čvor mobilna aplikacija se sastoji od više komponenti. Na slici 4.2 na čvoru mobilna aplikacija raspoređene su komponente: LoginActivity, UserRepository, ContentProvider, DatabaseHelper i ServiceManager.

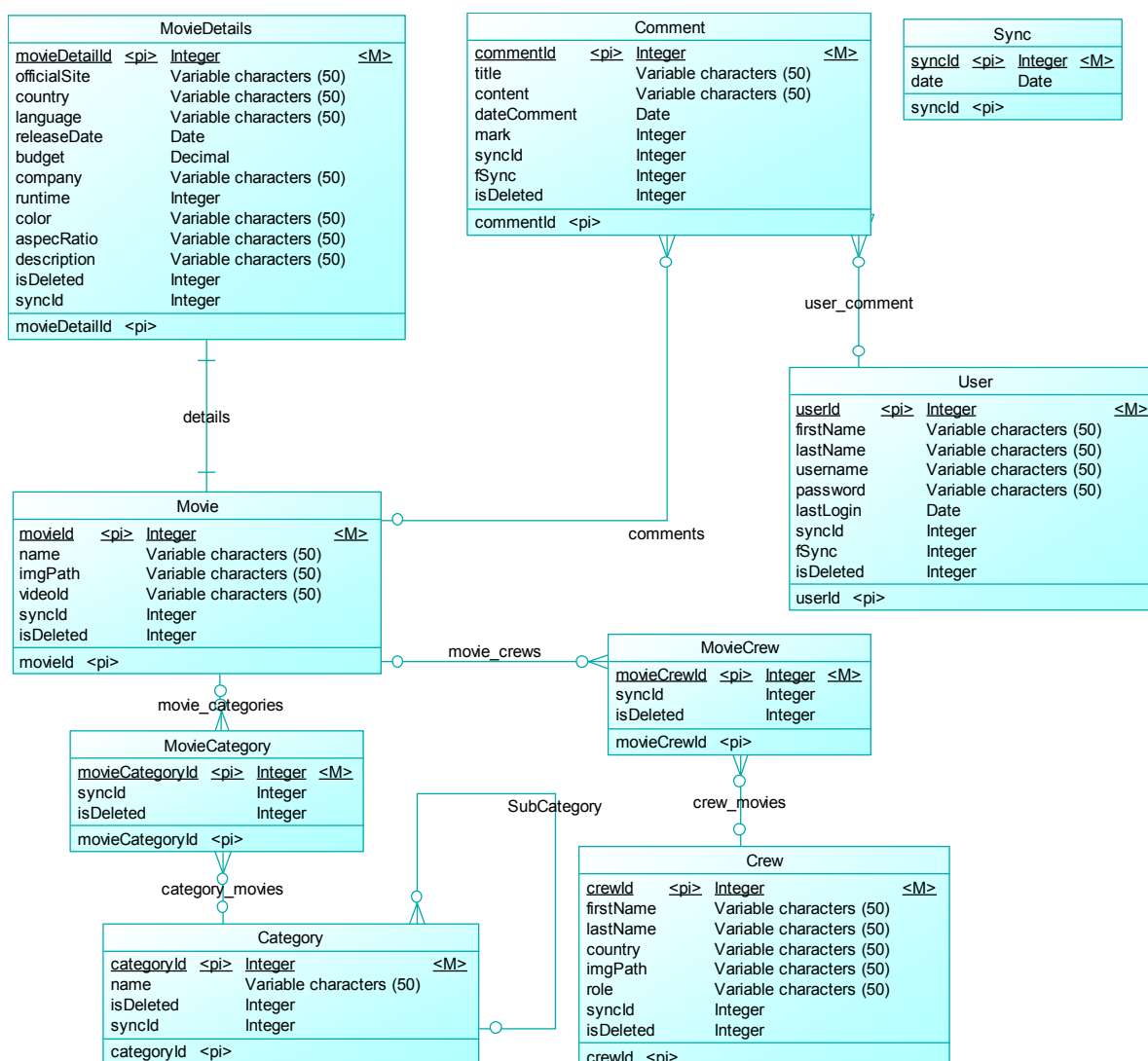
Na čvoru Android OS, pored mobilne aplikacije, raspoređene su komponente baza podataka i fajl sistem.

Na čvoru Server OS raspoređena je komponenta serverska aplikacija i baza podataka, a na čvoru serverska aplikacija raspoređene su komponente ValueController i DbContext.

LoginActivity je aktivnost, UserRepository je sladište, a ServiceManager je servis za komunikaciju sa serverskom aplikacijom. Aktivnosti komuniciraju sa skladištima. Skladišta komuniciraju sa dobavljačima sadržaja. Dobavljači sadržaja komuniciraju sa komponentom DatabaseHelper. Komponenta DatabaseHelper komunicira sa bazom podataka.



Slika 3.1. Dijagram slučajeва korišćenja

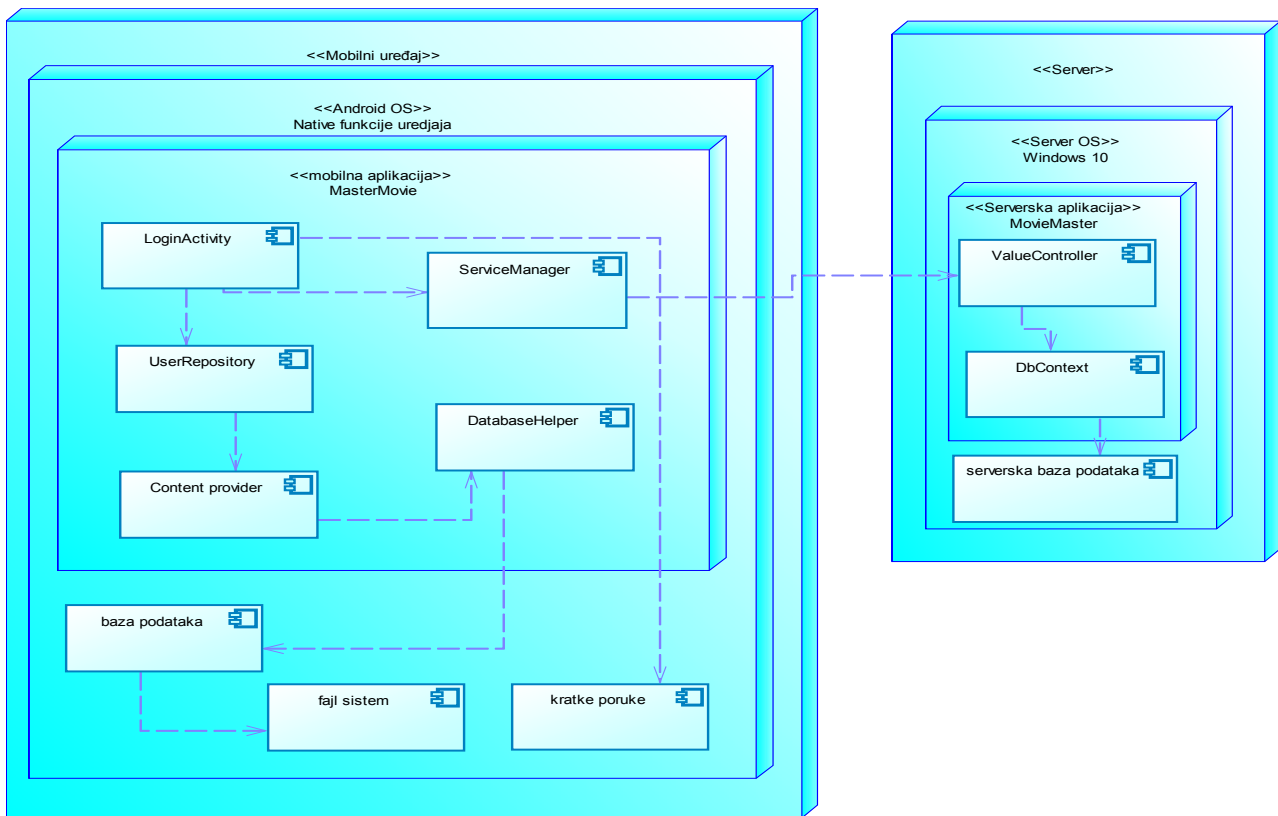


Slika 4.1 Model podataka

5. IMPLEMENTACIJA

U okviru ovoga rada, implementirano je softversko rešenje upotrebom AndroidAnnotations okvira specifikacije zahteva i dizajna sistema. Softversko rešenje predstavlja Android aplikacija. Za implementaciju korišćeni su sledeći alati i biblioteke: Android Studio 2.3 (razvojno okruženje) [4], AndroidAnnotations 4.3.0 (razvojni okvir), SQLite (baza

podataka) [5], .NET (razvojno okruženje za serversku aplikaciju) i MSSQL (baza podataka za serversku aplikaciju). Implementirano rešenje sastoji se od dva projekta. Jedan projekat je Android aplikacija implementirana sa Android-Annotations okvirom. Drugi projekat je serverska aplikacija implementirana na .NET platformi. Android aplikacija sinhronizuje podatke sa .NET aplikacijom.



Slika 4.2. Dijagram rasporeda

5.1 Model podataka

Model podataka definisan je u paketu db. Java klase koje implementiraju model podataka imaju naziv u obliku naziva entiteta sa sufiksom Table. Svaka klasa ima attribute koji definišu nazive kolona u bazi podataka, listu svih naziva kolona i iskaz za kreiranje tabele u bazi podataka.

5.2 Objekti prenosa podataka

Objekti prenosa podataka (eng. *data transfer object*- DTO) su objekti koji prenose podatke između dva procesa. U aplikaciji su DTO korišćeni u dva slučaja. Prvi slučaj je prenos podataka iz baze podataka prema grafičkom prikazu i obrnuto od grafičkog prikaza prema bazi podataka ukoliko se vrše akcije upisa, izmene i brisanja podataka. Drugi slučaj je prenos podataka od serverske aplikacije prema bazi podataka na mobilnoj aplikaciji.

5.3 Skladišta

```
@EBean
public class CrewRepository {
    @RootContext
    Context context;
    private ContentResolver resolver;
    @RootContext
    ...
    public long insert(CrewDto crew) {
        ...
    }
    ...
}
```

Listing 5.1. Implementacija skladišta u AndroidAnnotations okviru

Za svaki entitet napravljeno je posebno skladište. U skladištima je smeštena kompletna biznis logika aplikacije. Operacija za dodavanje, izmenu, brisanje podataka kao i

pripremanje prezentujućih objekata nalazi se u skladištu. Metode za dodavanje i izmenu podataka kao ulazne parametre primaju transfer objekte, zatim se ti objekti mapiraju u listu ContentValues i proslede ContentProvider klasi. U listingu 5.1 prikazan je primer skladišta implementiranog u AndroidAnnotations okviru.

5.4 Grafički korisnički interfejs

Grafički korisnički interfejs deklarisan je u XML dokumentima (po jedan dokument za svaku aktivnost i fragment). Svaki dokument ima strukturu stabla čiji elementi su pogledi i rasporedi. Raspored pogleda na ekranu je određen tipovima rasporeda. U aplikaciji je korišćen linearni raspored. Iskorišćena je mogućnost kombinovanja horizontalnog linearnog rasporeda i vertikalnog linearnog rasporeda. Za navigaciju je korišćena fioka za navigaciju (Navigation-Drawer). Svi stringovi su eksternalizovani u strings.xml dokumentu. Aplikacija je višejezična, pa je ovaj dokument preveden na srpski i engleski jezik.

5.5 Aktivnosti i fragmenti

Aktivnosti su Java klase koje nasleđuju klasu Activity i u njima se implementira grafički korisnički interfejs. Fragmenti su podaktivnosti grupisane tako da formiraju složenu aktivnost i kreiraju se nasleđivanjem Fragment klase. Fragmenti se implementiraju isto kao i aktivnosti. Klasi se dodeli anotacija @EFragment sa identifikatorom dokumenta u koje je deklarisan grafički korisnički interfejs.

MainActivity (listing 5.2) aktivnost sadrži metodu display-SelectedScreen. Ova metoda je zadužena za instanciranje izabranog fragmenata. Inicijalno, pri uspešnoj prijavi, prikazuje se fragment Movies. U meniju u fioci za navigaciju je moguće izabrati i druge fragmente.

```

@EActivity(R.layout.activity_main)
@OptionsMenu(R.menu.activity_main_drawer)
public class MainActivity extends
AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {
    ...
    @Bean
    public UserRepository userRepository;
    @ViewById(R.id.nav_view)
    NavigationView navigationView;
    ...
}

```

Listing 5.2. Implementacija aktivnosti u AndroidAnnotations okviru

5.6 Adapteri

Adapteri povezuju pogled i podatke. Aplikacija sadrži tri adaptera. To su adapteri za kategorije, komentare i filmsku ekipu. Svaki adapter sadrži anotaciju @EBean i nasleđuje klasu BaseAdapter.

5.7 Sinhronizacija

Postupak sinhronizacije se izvršava u oba smera od aplikacije prema serverskoj aplikaciji i obrnuto. U Android-Annotations okviru metoda za sinhronizaciju je deklarirana korišćenjem anotacija. Sa anotacijom @UiThread (delay = 3600000) podešeno je da se ova metoda izvršava u glavnoj niti na svakih dat vremena. U ovoj metodi poziva se metoda syncMasterMovieBackgroundWork koja se izvršava u pozadini. To je deklarirano tako što je metoda syncMasterMovieBackgroundWork anotirana sa @Background anotacijom. U ovoj metodi se pozivaju sync metode skladišta.

5.8 Serverska aplikacija

Serverska aplikacija implementirana je u .NET tehnologiji, a za skladištenje podataka koristi se MSSQL sistem za upravljanje bazom podataka. Podaci svakog registrovanog korisnika se sinhronizuju sa serverskom aplikacijom.

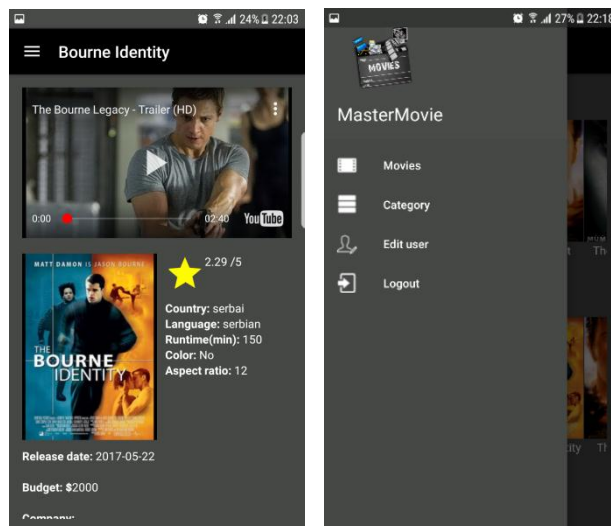
6. DEMONSTRACIJA

Aplikacija se sastoji od sledećih ekrana: ekran za login, ekran za registraciju, ekran za pregled filmova, ekran za pregled kategorija, ekran za dodavanje i izmenu komentara i ekran za izmenu podatka o korisniku.

Login ekran se sastoji od dva polja u koja je potrebno uneti korisničko ime i lozinku. Na polja je postavljena validacija.

Ukoliko je validacija zadovoljena otvara se početni ekran, a to je pregled filmova u sekcijama: top 10 i novi filmovi. Filmovi su raspoređeni u horizontalne trake. Svaki film ima sliku i naziv ispod slike. Kada se klikne na neki od filmova prikazu se detalji o filmu, kratak video filma i komentari korisnika.

Komentari su prikazani u hronološkom redosledu. Sa ekrana detalji o filmu (slika 6.1 levo) postoji mogućnost komentarisanja i ocenjivanja filma. Korisnik može da vrši izmene komentara. U svakom momentu korisnik može da pristupi meni ikonici koja se nalazi u gornjem levom uglu. Izborom kategorije u meniju (slika 6.1 desno) otvara se stranica gde su prikazani filmovi sortirani po kategorijama.



Slika 6.1 Detalji o filmu(levo), navigacioni meni(desno)

Na ovom ekranu filmovi su prikazani po kategorijama u horizontalnim trakama kao i na početnom ekranu. Na ekranu za izmenu podataka o korisniku, postoji mogućnost menjanja lozinke, imena i prezimena korisnika. Sva polja za unos imaju validaciju.

7. ZAKLJUČAK

Zadatak ovog rada bila je analiza AndroidAnnotations okvira za razvoj mobilnih aplikacija i njegova primena u implementaciji mobilne aplikacije koja je izabrana kao studija slučaja. Ovaj okvir korišćen je za razvoj aplikacije koja je vizuelno i funkcionalno identična aplikaciji koja je implementirana bez AndroidAnnotations okvira. Korišćenjem AndroidAnnotations okvira za implementaciju Android aplikacija, u značajnoj meri se smanjuje količina izvornog koda koja je potrebna za njihovu implementaciju. Razvoj aplikacija je brži, izvorni kod je čitljiviji, a održavanje aplikacija olakšano. Takođe je pojednostavljen pristup komponentama i resursima. Međutim, aplikacija razvijena u AndroidAnnotations okviru radi sporije u odnosu na aplikaciju razvijenu bez njega. Jedna od mana ovog okvira je i to da se u pozadini generiše suboptimalan kod. Optimizacijom procesa generisanja koda moglo bi se dobiti na performansama aplikacije.

8. LITERATURA

- [1] AndroidAnnotations, <http://androidannotations.org/> [pristupljeno 15.09.2017.]
- [2] Android, <https://www.android.com/> [pristupljeno 15.09.2017.]
- [3] Java, <https://java.com/en/> [pristupljeno 15.09.2017.]
- [4] AndroidStudio, <https://developer.android.com> [pristupljeno 15.09.2017.]
- [5] SQLite, <https://www.sqlite.org/> [pristupljeno 15.09.2017.]

Kratka biografija:



Danijel Popović rođen je u Vrbasu 19.09.1989. godine. Završio je srednju tehničku školu Mihajlo Pupin u Kuli 2008. godine. Osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu završio je 2015. godine. Master akademske studije upisao je 2015. godine. Položio je sve ispite predviđene studentskim programom.

РАЧУНАРСКИ ПОДРЖАНО УПРАВЉАЊЕ ПРОЦЕСОМ УПИСА КАНДИДАТА НА СТУДИЈСКЕ ПРОГРАМЕ**COMPUTER SUPPORTED MANAGEMENT OF THE PROCESS OF CANDIDATES' ENROLLMENT TO STUDY PROGRAMS**

Владимир Поповић, Факултет техничких наука, Нови Сад

Област – ПРИМЕЊЕНЕ РАЧУНАРСКЕ НАУКЕ И ИНФОРМАТИКА

Кратак садржај – Подручје истраживања овог рада представља управљање пословним процесима и њихова практична примена у процесу пријемног испита на Медицинском факултету у Новом Саду. Наиме, проблематика која се јавила у виду неадекватних софтверских решења захтевала је другачији приступ проблему, те је покушано превазићи је развојем апликације која се заснива на систему за управљање пословним процесима. Успешном имплементацијом задатог проблема показана је употребна вредност управљања пословним процесима и на који начин се њима може убрзати и контролисати систем. Такође, приказан је и сам приступ решавању задатог проблема, који не служи само за спецификацију захтева, већ често може идентификовати грешке у процесима или унапредити већ постојеће процесе.

Abstract – The field of research of this paper is the management of business processes and their practical application in the process of enrollment at the Faculty of Medicine in Novi Sad. Namely, the problems that occurred in the form of inadequate software solutions required a different approach to the problem, and it was attempted to be overcome by developing an application that is based on the business process management system. Successful implementation of the given problem shows the usable value of business process management and how can they speed up and control the process. Also, the approach to solving the given problem is shown, which not only serves for the specification of requests, but can often identify process errors or improve existing processes.

Кључне речи: Управљање пословним процесима, Business Process Model and Notice, BPMN

1. УВОД

Модерно пословање је постало незамисливо, и у крајњој линији неизводљиво, без ослањања на рачунаре и интернет. Бенефити су јасно видљиви, географска удаљеност је постала ирелевантна а пропагација и синхронизација информација моментална.

Као и свака појава, и информатичке технологије носе изазове. Тако је на једном пословном форуму, Бил Гејтс (Bill Gates), суоснивач компаније Microsoft,

констатовао: „Прво правило везано за употребу технологије у бизнису је да ће аутоматизација примењена на ефикасне процесе увећати ефикасност. Друго је да ће аутоматизација примењена на неефикасне процесе увећати неефикасност“. Таквим приступом се у први план доводи потреба за детаљним планирањем, усклађивањем и имплементацијом софтверских решења у складу са стварним потребама и захтевима. Додатни изазов је како одговорити променама тржишних трендова и промена пословања, јер они захтевају и прилагођавање софтверских решења.

Један приступ се показао као решење за оба изазова. Пословни процеси (или пословне методе) су повезане и уређене активности или задаци које реализују одређену услугу или производ. Управљање пословним процесима за циљ има дефинисање и обједињавање у смислену целину коначног броја пословних процеса једног предузећа. Оваквим приступом омогућава се сагледавање и верификовање постојећих процеса, као и брзо прилагођавање променама, јер их је лако идентификовати у моделу, те моделе прилагодити.

Циљ рада је практична примена оваког приступа у процесу пријемног испита на Медицинском факултету у Новом Саду.

2. СТАНДАРД BPMN 2.0

Стандар одржава и унапређује међународна организација *OMG* (Object Management Group), непрофитни конзорцијум за технолошке стандарде који је отвореног типа. У његовом раду, још од 1989. године, учествују произвођачи, корисници, академске институције и разне федералне агенције. Усмерени су ка развоју стандарда за интеграцију из области технологије и индустрије [1]. Стандард *BPMN* је у актуелној верзији 2.0 од јануара 2011. године, тј. у ревизији 2.0.2 од децембра 2013. године.

2.1. Основне идеје и принципи

Идеје којима се *OMG* водила заснивају се на концепту широке применљивости. Наиме, с једне стране стандард треба да идентификује најбоље праксе и да их обједини у једну спецификацију. С друге стране, нотација треба да буде јасна и доступна свим учесницима пословних процеса. И с треће стране, стандард покушава да квантитативно измери усклађеност постојећих развојних окружења, те дефинише три класе усклађености:

- Усаглашеност моделовања процеса,
- Усаглашеност извршавања процеса и
- Усаглашеност моделовања кореографија.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Мирослав Зарић, доцент.

2.1. Погледи научне заједнице

Како је управљање пословним процесима актуелна тема на тржишту, тако је и у академској заједници. Стога је у овој секцији обрађен академски аспект истих, а кроз призму чланака у стручним часописима.

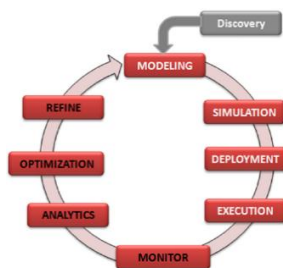
Врло занимљиво истраживање спроведено је на Универзитету у Севиљи под називом „*A survey on business processes management suites*“ [2] у коме су аутори проценили употребну вредност развојних окружења за пословне процесе на примеру управљања развојем софтверских решења као и њихов потенцијал за даља унапређења.

Чланак има за циљ да прикаже анкету коју свака информатичка компанија може користити да пореди постојећа развојна окружења схондо својим циљевима и потребама. Анкета узима у обзир сваку фазу животног циклуса а комбинује две добро познате технике: *Systematic Literature Review* [3] и *Quality Frameworks* [4] базиран на схеми карактеризације ревидиране од стране релевантних ИКТ компанија у Шпанији. Анеката се базирала на *open-source* решењима, која су у првом кораку прошла SLR тест, табела 1.

	QA1	QA2	QA3		QA1	QA2	QA3
Activiti	•	•	•	YAWL	•	•	•
BonitaOS	•	•	•	XFLOW			
Intalio/BPMS	•	•	•	TAVERNA			
jBPM	•	•	•	OpenWFE			
Kbee workflow				WfMOpen			
ProcessMaker	•	•	•	Freeflow			
uEngine BPM	•	•	•	Syrup			
Camunda	•	•	•	JawFlow			
Enhydra shark							

Табела 1 *Open-source* развојна окружења

За другу фазу развијен је квалитативни модел, који се заснива на схеми карактеризације, у сврху евалуације *BPM* развојних окружења а узимајући у обзир потребе компанија и задатке истраживања.



Слика 1. Хилов модел животног циклуса

Аутори су користили Хилов модел животног циклуса [5], те у складу са њим, слика 1, оцењивали окружења.

Након завршене друге фазе истраживања, добијени су резултати приказани у табели 2.

Табела 2. Резултати истраживања

	<i>Bonita</i>	<i>ProcessMaker</i>	<i>YAWL</i>
Резултат	7.08	6.28	7.43
<i>Camunda</i>	<i>Activiti</i>	<i>jBPM</i>	<i>uEngine</i>
7.79	6.92	7.99	6.21

Аутори предлажу своју методологију за избор адекватног развојног окружења из два разлога:

- детаљност анализе где свака компанија може фаворизовати кријеријуме који су њој најбитнији,
- из овакве анализе постоји могућност прављења планова унапређења.

У другом раду, истраживачи са немачких универзитета у Бамбергу и Карлштаду [6] су се бавили интересантним аспектом пословних процеса у стандарду *BPMN 2*. Наиме, иако је стандард активан још од 2011. године, покушали су утврдити колико су постојећа развојна окружења за моделовање пословних процеса (*Business Process Management Systems/Suites*) имплементирала стандард.

У складу са изнешеним, циљ је био препознати софтвере који су у могућности да директно извршавају семантику модела по стандарду и обрађују серијализован формат. Након анализе тренутног тржишта, препознато је 47 алата за моделовање пословних процеса, а директно извршавање модела и серијализован формат подржала су само три: *Camunda BPM*, *jBPM* и *Activiti*.

За утврђивање степена усклађености са стандардом, коришћене су тада (30. јун 2016.) најновије верзије изабраних софтвера: *Activiti 5.20.0*, *Camunda BPM 7.5.0* и *jBPM 6.4.0*. За тестирање је кориштено тест окружење *Betsy* [7] (*BPEL/BPMN Engine Test System*). У тестном окружењу изведене су три врсте тестова:

- Тестирање елемената језика за моделовање (*Language constructs testing*),
- Тестирање образаца за контролу тока (*Workflow control-flow patterns testing*) и
- Анализа модела (*Static analysis rules*).

Сумирано, у првом делу истраживања, од 47 алата доступних на тржишту, испоставило се да су само 3 у могућности да обрађују серијализовани формат и дозвољавају аутоматизовану евалуацију. Даље, изведена су побројана три теста. Приказано је да, иако је најбољи резултат подржаност 57% елемената језика (*jBPM 6.4.0*), основни образци за контролу тока су поуздани и добро имплементирани. Такође, постојећи капацитети су више него довољни за свакодневне захтеве у пракси.

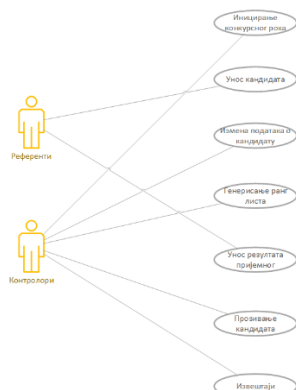
3. ПРОЦЕС УПИСА НА МЕДИЦИНСКИ ФАКУЛТЕТ

Медицински факултет Универзитета у Новом Саду сваке године расписује конкурс за упис кандидата на студијске програме Факултета који су акредитовани од стране Министарства просвете, науке и технолошког развоја и то у оквиру квоте за рад одобрене од стране истог. Упис се реализује кроз два конкурсна рока, и то јунски и септембарски.

Процес уписа на Факултет регулисан је Правилником о ближим условима за упис на интегрисане, основне и мастер студије на Факултету и начину спровођења пријемног испита.

Упис се реализује кроз секвенцијалне кораке и то у следећем редоследу: Иницирање новог конкурсног

рока; Пријава кандидата; Измена података о кандидату; Прелиминарне ранг листе са успехом из средње школе/факултета; Коначне ранг листе са успехом из средње школе/факултета; Уношење резултата пријемног испита; Увид у тестове; Коначна ранг листа; Прозивка; Упис; Извештаји.



Слика 2. Use case дијаграм процеса

На UML дијаграму на слици 2 приказана је статичка структура која представља интеракцију корисника система који учествују у процесу уписа на Факултет са самим системом.

3.1. Избор развојног окружења

На почетку израде мастер рада послављало се логично питање: Које развојно окружење изабрати? Једини увид у проблематику сам стекао током похађања курса Управљање пословним процесима те се на почетку израде *Activiti* наметао као логичан избор. Но, како бих стекао стваран увид у ситуацију на тржишту, читајући блогове и прегледе, као кандидати поред *Activiti*-а издвојили су се још и *BonitaSoft BPM*, *Camunda* и *jBPM*. Полазна основа за оцењивање били су могућности развојног окружења у складу са задатком рада, постојање бесплатне верзије и заснованост на отвореној (*open source*) платформи.

У складу са задатком, захтев је био да развојно окружење подржава једноставно графичко моделовање и програмирање дијаграма. Даље, као предност је урачунат развијен *IDE* као самостална апликација. На крају, врло пожељним се сматрала могућност креирања корисничких форми из самог развојног окружења.

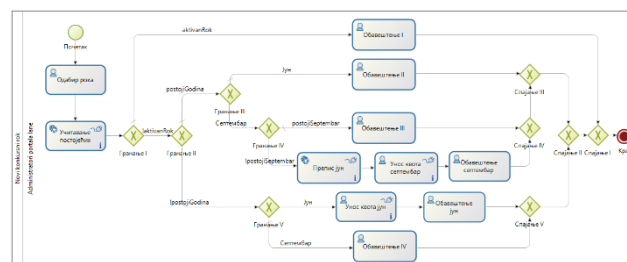
На тестирање сваког алата утрошено је по недељу дана, а одлука, чисто субјективне природе и у односу на постављени циљ, донешена је у корист развојног пакета *BonitaSoft BPM*, и то из следећих неколико разлога: комплетан развој и тестирање је могуће извршити у једном *IDE*-у, без потребе за коришћењем, нпр. *Eclipse*-а; могућност лаког креирања *web* форми; и можда најбитније, крива учења се показала најбоља.

3.2. Моделовање процеса

Обзиром на природу процеса и чињеницу да су сви кораци у временском следу, тј. један се наставља на други, процес је могуће поделити на више временски независних процеса. Овако моделовани процеси одигравају се један за другим и чине комплетан процес уписа.

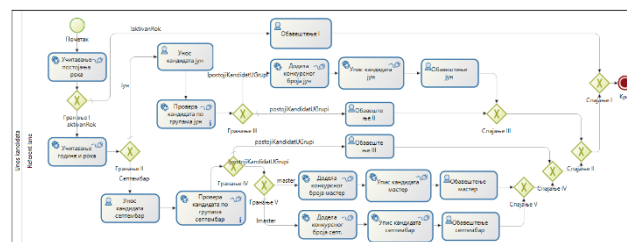
Први корак у процесу је иницирање новог конкурсног рока, у коме се задају критеријуми за конкурс (као

што је број места) који су преписани из дозволе за рад коју високошколска установа добија од Министарства. Модел процеса приказан је на слици 3.



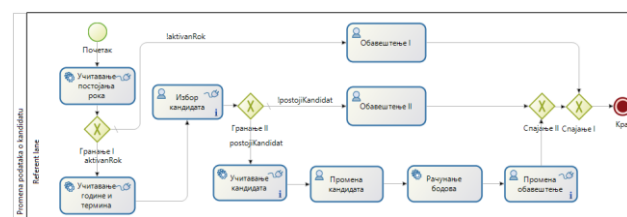
Слика 3. Иницирање конкурсних рока

Следећи корак је унос података о кандидатима који испуњавају услове предвиђене конкурсом. Модел је представљен на слици 4.



Слика 4. Унос новог кандидата

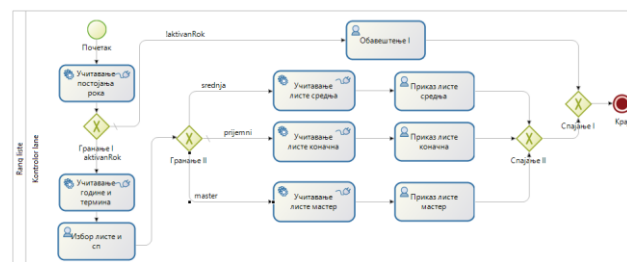
Уколико се јави потреба за изменом података о кандидатима, покреће се процес Измене података о кандидату (слика 5).



Слика 5. Промена података о кандидату

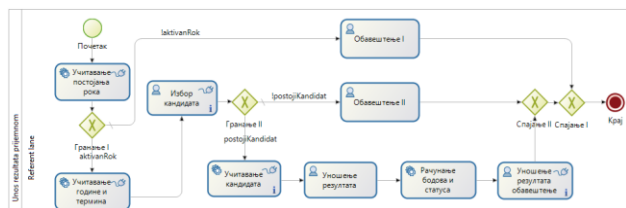
Ова ситуација се може десити након објављивања ранг листе, и то: након објављивања прелиминарних ранг листе кандидата уколико кандидат поднесе приговор и утврди се да је неки од података погрешно унесен (као што су лични подаци); након објављивања прелиминарних ранг листе са успехом на пријемном уколико се утврди да бодови на пријемном нису добро обрачунати (након приговора и увида у пријемни).

Након завршеног периода за конкурсисање, неопходно је генерисати ранг листе кандидата (слика 6). Такође, ранг листе је потребно генерисати и након уноса резултата пријемног испита.



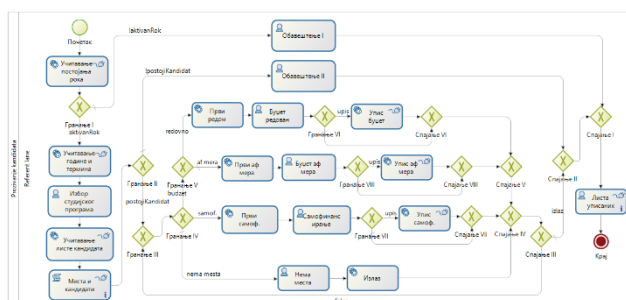
Слика 6. Генерисање ранг листа

Након завршеног пријемног испита и оцењивања тестова од стране Комисије за спровођење пријемног испита, на ред долази уношење резултата у систем. Модел је представљен на слици 7.



Слика 7. Унос резултата пријемног

Прозивање кандидата је првенствено осмишљено и имплементирано за упис на студијски програм Основне академске студије специјалне едукације и рехабилитације који садржи модуле. Модел је приказан на слици 8.



Слика 8. Прозивање кандидата

Када се заврши процес уписа у конкурсном року, постоје извештаји које је потребно како похранити у сопственој архиви, тако и послати у Покрајину и Министарство. На слици 9 приказан је модел.



Слика 9. Генерисање извештаја

Процес пријемног испита заокружен је са претходно представљених шест подпроцеса.

4. ЗАКЉУЧАК

Мастер рад имао је за циљ, кориштењем алата и техника за моделовање пословних процеса, развити апликацију кроз коју би се организовао и спроводио процес пријемног испита за академске, основне и мастер студије на Медицинском факултету у Новом Саду.

Опсежно и једнозначно захтеви су описани а апликација, која је развијена коришћењем техника управљања пословним процесима, тестирањем је потврдила функционалност. Овим је испуњен примарни задатак и јасно је показано да је моделовањем пословних процеса изводљиво бијективно пресликавање пословних метода везаних за процес пријемног испита у горепоменутој установи.

Штавише, испојили су се додатни бенефити које приступ заснован на моделовању пословних процеса нуди:

- Крајњи софтверски производ је високо конфигурабилан. У случају захтева за променама, лако и јасно се дефинише шта се треба преправити, те се не мора све програмирати, што убрзава и појефтиније процес.
- У првој фази, након завршеног дизајна долази анализа модела. У току анализе „испливавају“ потенцијалне грешке у моделу, тј. у организацији предузећа чије се пословање моделује. Овим се заправо може унапредити постојећи систем.

Крајњи бенефит упознавања са области управљања пословним процесима, са аспекта кандидата за израду мастер рада из поља примењених рачунарска наука и информатике, је пружање једне нове перспективе инжењерима, где се види како поред класичног програмирања информатички фокус полако захвата неке друге гране индустрије.

5. ЛИТЕРАТУРА

- [1] Object Management Group, доступно на страници: <http://www.omg.org/>, датум: 12.08.2016.
- [2] A survey on business processes management suites, A. Meidan, J. A. García-García, M. J. Escalona, I. Ramos, (2016), Computer Standards & Interfaces.
- [3] A systematic review of systematic review process research in software engineering, Information and Software Technology, (2013), B. Kitchenham, <http://dx.doi.org/10.1016/j.infsof.2013.07.010>, (ISSN0950-5849)
- [4] QuEF (Quality Evaluation Framework for model-driven web methodologies), F. J. Domínguez-Mayo, M. J. Escalona, M. Mejías, (2010) LNCS6385(2010)571–575
- [5] Gartner's position on business process management, J. B. Hill, J. Sinur, D. Flint, M. J. Melenovsky, in: C.T. Stamford (Ed.), Business Issues Gartner, 2006.
- [6] BPMN 2.0: The state of support and implementation, Matthias Geiger, Simon Harrer, Jörg Lenhard, Guido Wirtz, (2017) Future Generation Computer Systems.
- [7] On the Evolution of BPMN 2.0 Support and Implementation, M. Geiger, S. Harrer, J. Lenhard, G. Wirtz, (2016) 10th IEEE SOSE, Oxford, UK, 2016.

Кратка биографија:



Владимир Поповић рођен је у Новом Саду 1982. год. Мастер рад на Факултету техничких наука из области Примењених рачунарска наука и информатике одбранио је 2017. год.

REALIZACIJA MODBUS I MODBUS SUNSPEC SERVERA NA XILINX ZYNQ PLATFORMI**MODBUS AND MODBUS SUNSPEC SERVER IMPLEMENTATION ON XILINX ZYNQ PLATFORM**

Predrag Krža, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je objašnjen postupak razvijanja Modbus i Modbus SunSpec server aplikacija na Xilinx Zynq platformi, kao i razvijanje grafičkog interfejsa za konfiguraciju servera u vidu signal procesing komponenti. Rad takođe prikazuje rezultate testiranja servera korišćenjem testova definisanih od strane Modbus organizacije.

Ključne reči: Modbus, SunSpec, server, Xilinx, Zynq

Abstract – The method of developing Modbus and Modbus SunSpec server applications, as well as development of graphical interface for server configuration in the form of signal processing components is shown in this paper. Besides, the testing results of server obtained using the tests defined by the Modbus Organization is also given.

Ključne reči: Modbus, SunSpec, server, Xilinx, Zynq

1. UVOD

Komunikacija igra važnu ulogu u automatizaciji i kontroli procesa unutar industrijskih sistema. Kako bi se omogućila automatizacija u kontroli rada, uređaji moraju biti u međusobnoj sprezi i informacije o stanjima pojedinih delova sistema moraju biti dostupne centralnom uređaju za nadgledanje procesa, tj. SCADA (eng. *Supervisory Control And Data Acquisition*) sistemu. Kako bi se omogućila komunikacija između uređaja, mora postojati neki skup pravila za razmenu poruka, odnosno protokol. Danas postoji veliki broj industrijskih komunikacionih protokola, a Modbus, kao prvi međunarodno priznat procesni protokol, je postao standardan protokol velikog broja proizvođača.

Serijski interfejs je prvo korišćen u industrijskoj komunikaciji. Tako je i prvi Modbus protokol komunicirao preko RS-232, a kasnije preko RS-485 interfejsa. Razvojem komunikacionih tehnologija, pojavio se Ethernet i lokalne mreže kao nov, brz i jeftin način komunikacije velikog broja uređaja. Veliki broj industrijskih protokola prelazi na rešenja bazirana na Ethernet tehnologiji, među kojima je i Modbus. Danas postoji više verzija Modbus protokola koji koriste metode serijske i Ethernet komunikacije, a najrasprostranjeniji su Modbus RTU protokol i Modbus TCP protokol, koji je i glavna tema rada.

Kao krajnji cilj razvoja Modbus i Modbus Sunspec servera je interakcija sa simulacijom elektroenergetskih sistema putem Modbus protokola koji se simuliraju na HIL (eng. *Hardware In the Loop*) uređajima kompanije *Tajfun HIL*.

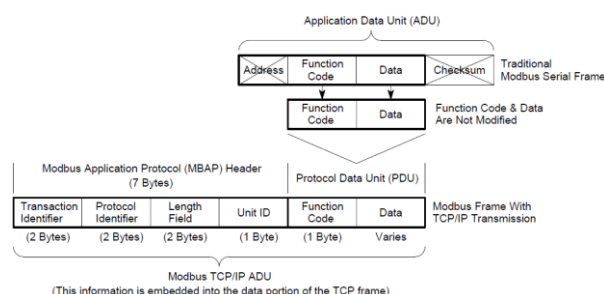
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Ivan Mezei, docent.

2. MODBUS TCP

Modbus protokol je baziran na *master-slave* komunikaciji u kojoj *master* uređaj inicira komunikaciju šaljući zahtev *slave* uređaju. Svaki *slave* uređaj se sastoji iz 1-bitnih i 16-bitnih registara i sve funkcije koje zahteva *master* uređaj se izvršavaju nad tim registrima. Namena svakog registra je unapred poznata i nju definiše proizvođač. Protokol jasno definiše koje funkcije može da podrži *slave* uređaj i osnovni skup funkcija koje svaki *slave* mora da podrži. Kako su *server* i *client* pojmovi zastupljeni u Ethernet terminologiji, oni se više koriste umesto *master* i *slave*. U ovakvoj nomenklaturi, *master* uređaj postaje *client*, a *slave* uređaj postaje *server*.

Modbus TCP protokol je aplikativni protokol koje se nalazi na sedmom sloju OSI modela. Modbus TCP kombinuje fizičku mrežu (Ethernet), mrežni standard (TCP/IP) i standardnu reprezentaciju podataka (Modbus kao aplikativni protokol) za prenos poruka između uređaja. Modbus TCP poruka je jednostavno Modbus komunikacija enkapsulirana u Ethernet TCP/IP zaglavlje (Slika 1).



Slika 1. Konstruisanje Modbus TCP paketa

Postoje četiri tipa registara koje definiše Modbus protokol: *Discrete Inputs*, *Discrete Outputs* (ili *Coils*), *Input Registers* i *Output Registers* (ili *Holding Registers*). Tabela 1 opisuje ove registre.

Ako se šalju 1-bitni podaci, osam statusnih bita se pakuje u jedan bajt tako da LSB bit predstavlja status registra najmanje adrese. Vrednost 16-bitnih registara se pakuju u dva bajta u *big-endian* formatu, tako da se prvo šalje bajt veće, a zatim bajt manje težine. Iako nije definisano Modbus protokolom, više 16-bitnih registara se može kombinovati za reprezentaciju 32-bitnih i 64-bitnih *int*, *unsigned* i *float* vrednosti. Ne postoji jasno definisan način kako se ove vrednosti mapiraju na protokol, ali najveći broj uređaja zadržava *big-endian* reprezentaciju kao i kod 16-bitnih registara. U radu je takođe prihvaćen ovaj metod mapiranja.

Svakom registru se pristupa preko njegove adrese, gde se vrednost adrese kreće u opsegu 0 – 65535.

Tabela 1. Tipovi Modbus registara

Referenca	Tip	Veličina	Pristup*
0xxxx	Coils	1 bit	R/W
1xxxx	Discrete Inputs	1 bit	R
3xxxx	Input Registers	16 bita	R
4xxxx	Holding Registers	16 bita	R/W

* R – read, W – write

Sve funkcije definisane Modbus protokolom se izvršavaju nad nekim od četiri tipa registara. Svaka funkcija ima definisan jedinstven kod. Modbus server mora podržati minimalan skup funkcija koji se sastoji iz funkcija čitanja i pisanja. Tabela 2 prikazuje ovaj osnovni skup funkcija. Detaljan opis svake funkcije se može pronaći u [1].

Tabela 2. Osnovni skup funkcija

		Kod funkcije	
Ime funkcije		Dec	Hex
Bit access	Read Discrete Inputs	02	0x02
	Read Coils	01	0x01
	Write Single Coil	05	0x05
	Write Multiple Coils	15	0x0F
16 bit access	Read Input Registers	04	0x04
	Read Holding Registers	03	0x03
	Write Single Register	06	0x06
	Write Multiple Registers	16	0x10
	Read/Write Multiple Registers	23	0x17

Poruke izuzetaka klijent uzvraća serveru ukoliko iz nekog razloga primljen zahtev nije ispravan. PDU polje poruke izuzetka se sastoji iz modifikovanog koda funkcije i koda izuzetka (eng. *Exception Code*). Kod funkcije se modifikuje tako što se MSB koda postavi na 1. Lista kodova izuzetaka se može pronaći u [1].

3. MODBUS SUNSPEC

SunSpec alijansa (eng. *SunSpec Alliance*) je savez više od stotinu učesnika u razvoju solarnih i distributivnih energetskih resursa sa ciljem razvoja standardizovanih informacionih modela, modela podataka i interfejsa koji bi omogućili interoperabilnost uređaja u sistemu. Informacioni model predstavlja unapred definisan skup registara nekog uređaja, njihovo značenje, tip, dužinu i pristup. Modeli ne zavise od protokola koji se koriste za komunikaciju, ali mapiranje modela je drugačije za svaki protokol. Modbus SunSpec predstavlja mapiranje SunSpec informacionih modela na Modbus protokol. Ovi informacioni modeli su standardno opisani u XML formatu u SMDX (eng. *SunSpec Model Definition XML*) notaciji. SunSpec definiše nekoliko tipova informacionih modela: *Common* model, *Standard* model, *Vendor* model i *End* model. Svaki SunSpec kompatibilan uređaj (u daljem tekstu SunSpec uređaj) mora da ima definisan *Common* model kao prvi model, *End* model kao krajnji model i mora posedovati barem jedan *Standard* ili jedan *Vendor* model. Slika 2 ilustruje strukturu SunSpec uređaja.

Common Model
Standard Model(s)
Vendor Models(s)
End Model

Slika 2. Struktura SunSpec uređaja

Common model pruža informacije o proizvođaču i uređaju. *Standard* modeli su modeli koji su testirani i odobreni od strane SunSpec zajednice, dok su *Vendor* modeli razvijeni od strane proizvođača. Registri SunSpec modela mogu biti sledećeg tipa: *int*, *uint*, *float*, *pad*, *acc*, *enum*, *bitfield*, *string*, *sunssf* ili *ip*. Značenje svakog tipa i načina mapiranja na Modbus protokol je detaljno opisano u [2].

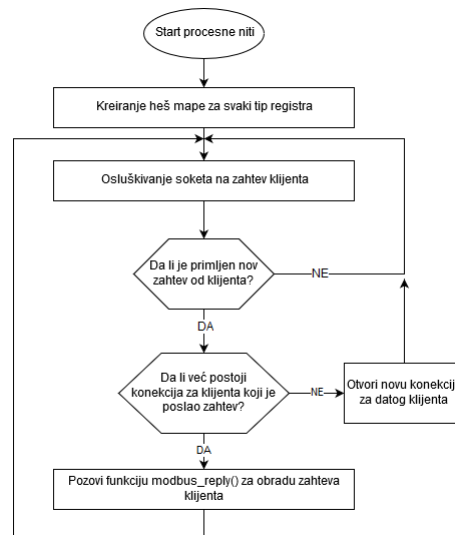
Bazni registar koji predstavlja početak registarskog prostora SunSpec uređaja ima jednu od tri unapred definisane vrednosti: 40001, 50001, ili 1. Prva dva registra imaju vrednost 'SunS' (0x53756e53) koja služi da identifikuje uređaj koji podržava SunSpec model.

4. REALIZACIJA I TESTIRANJE MODBUS SERVER APLIKACIJE

4.1 Realizacija Modbus server C aplikacije

Modbus server C aplikacije je realizovana sa ciljem izvršavanja na *Linux* operativnom sistemu. *Linux* ima ugrađenu podršku za rad sa Ethernetom, a podrška korišćenja procesnih niti (eng. *Threads*) pojednostavljuje realizaciju aplikacije koja podržava veći broj servera. Takođe, ovaj operativni sistem podržava koncept dodele više IP adresa jednom uređaju čime se omogućuje da svaki server ima sopstvenu IP adresu. Korišćena *libmodbus* biblioteka se oslanja na *Linux* Ethernet drajvere za prijem i slanje Modbus paketa.

Na početku aplikacije, konfigurišu se IP adrese za svaki Modbus server unutar aplikacije. Nakon konfiguracije adresa, pokreću se procesne niti, i to za svaki server po jedna. Dijagram izvršenja jedne niti je prikazan na slici 3.



Slika 3. Blok dijagram izvršenja procesne niti

Svaka nit kreira registarsku heš mapu (eng. *hashmap*) i čeka na konekciju klijenta. Korišćenje heš mape je neophodno kako bi se promenljivima definisanim u C kodu moglo pristupiti preko adrese registra koje zahteva klijent. Zasebna heš mapa se kreira za sve tipove registara. Pored heš mape koja povezuje adresu registra sa C varijablom, kreira se dodatna mapa koja povezuje adrese registara sa adresama HIL uređaja na kojima se nalaze potrebne vrednosti simulacije.

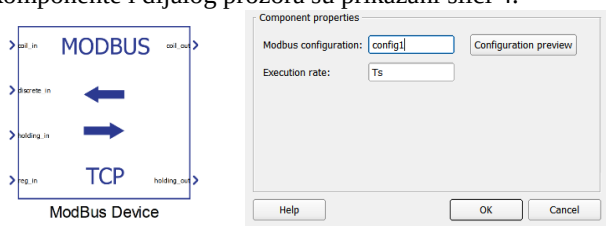
Uspostavljanje konekcije sa klijentom obrađuje *libmodbus* biblioteka. Na zahtev klijenta, poziva se

funkcija *modbus_reply()* koja obrađuje zahtev i uzvraća poruku klijentu. Zahtev se obrađuje na osnovu koda funkcije. Ukoliko kod nije podržan, uzvraća se poruka izuzetka. U suprotnom, vrednosti se ažuriraju i uzvraća se poruka klijentu. Ako je primljen zahtev za čitanje, varijable se prvo ažuriraju vrednostima simulacije, a ako je primljen zahtev za upis, varijablama se prvo promene vrednosti, a zatim se ažuriraju vrednosti simulacije.

Sve informacije o serverima i registrima se dinamički generišu u zavisnosti od konfiguracije servera pomoću šablona za automatsko generisanje koda (eng. *template*).

4.2 Kreiranje Modbus server komponente

Kako bi konfigurisanje bilo što lakše, potrebno je kreirati neki vid grafičkog interfejsa koji će omogućiti lako zadavanje parametara Modbus servera. Razvijen je grafički interfejs u vidu *signal processing* komponente koji se iz biblioteke može ubaciti u šemu, a kroz dijalog prozor komponente zadati odgovarajući parametri. Izgled komponente i dijalog prozora su prikazani slici 4.



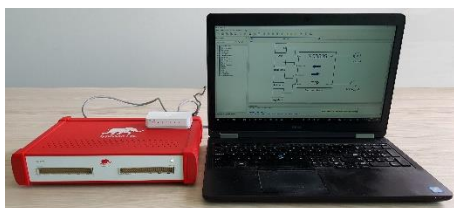
Slika 4. Izgled i dijalog prozora Modbus komponente

U polju *Modbus configuration* se specificira konfiguracija servera, a polje *Execution rate* definiše vreme osvežavanja vrednosti simulacije. Konfiguracija servera se definiše *python dictionary* tipom sa unapred definisanim poljima. Vrednostima ovih polja se definiše komunikacioni deo konfiguracije (*ip*, *netmask*, *port* i *slave id*) i registarska mapa servera gde se mogu definisati adresa, veličina i tip registra. Kompajliranjem modela se validira i parsira konfiguracija, a zatim se generiše konfiguracioni fajl koji se koristi u C aplikaciju.

4.3 Testiranje

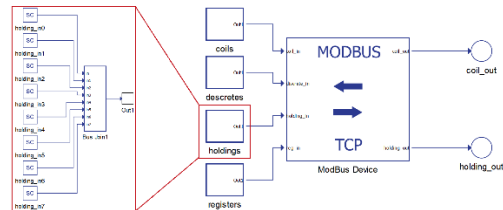
Prilikom testiranja Modbus server uređaja, postoje određeni kriterijumi koji moraju biti zadovoljeni da bi se on tretirao kao Modbus kompatibilan. Ovi kriterijumi su zadati od same Modbus organizacije i izneti su u [3]. Svaki modbus kompatibilan uređaj mora da odgovori uspešno na sve ispravne i neispravne zahteve servera, kao i da uspešno može da izvrši svaku podržanu funkciju na registru najmanje i najveće adrese, kao i na najvećem broju uzastopnih registara. Takođe, vrednosti registara koje se pročitaju i upišu Modbus klijentom moraju da se poklapaju sa vrednostima simulacije.

Postavku sistema za testiranje čine HIL uređaj i PC računar umreženi preko Ethernet kablova i mrežnog razdelnika (Slika 5).



Slika 5. Postavka sistema pri testiranju Modbus servera

Na HIL uređaju je pokrenut Modbus server dok je na računaru pokrenut *Typhoon HIL Control Center* program za upravljanje simulacije i Modbus klijent kojim se šalju zahtevi serveru. Server je konfigurisan tako da ima po 5 *Coil* i *Discrete Input* registara i po 8 *Holding* i *Input* registara od kojih su dva 16-bitna registra tipa *int* i *unsigned*, tri 32-bitna i tri 64-bitna registra tipa *int*, *unsigned* i *float*. Korišćeni model pri testiranju je prikazan na slici 6.



Slika 6. Model korišćen za testiranje Modbus servera

4.4 Rezultati testiranja

Testiranjem Modbus servera su dobijeni rezultati koji ukazuju na ispravnost rada aplikacije. Server uspešno odgovara na sve ispravne i neispravne zahteve klijenta, a odgovori imaju ispravnu formu kako je definisano u [1]. Vrednosti koje klijent čita i upisuje za 1-bitne i 16-bitne registre se poklapaju sa vrednostima simulacije. Neslaganje rezultata se uočava kod manipulacije 32-bitnih i 64-bitnih vrednosti. Upisivanjem velikih vrednosti u ove registre dovodi do gubitka preciznosti unutar simulacije kao što prikazuje tabela 3. Razlog ovakve greške se nalazi u samoj arhitekturi HIL uređaja i samog modela. Naime, sve memorijske lokacije HIL uređaja su veličine 32 bita. Da bi se mogle predstaviti 64 bitne *int* i *unsigned* vrednosti, one se u aplikaciji konvertuju u tip *float* i kao takve se upisuju u HIL memoriju. Ova konverzija dovodi do gubitka preciznosti za vrednosti koje su veće od 24 bita. Problem preciznosti se može prevazići ako se svi ulazi u Modbus komponentu definišu kao *int* ili *unsigned* tip. U ovom slučaju konverzija u *float* se neće izvršiti, ali kao rezultat se ne mogu koristiti vrednosti duže od 32 bita.

Tabela 3. Poređenje očitanih i zadatih vrednosti

Tip vrednosti	Zadata vrednost	Vrednost simulacije
uint16	54894	54894
int16	-31394	-31394
uint32	3564821368	3,56482e+09
int32	-2117546328	-2,11755e+09
float32	346,184	346,184
uint64	14239567456983121412	1,42396e+19
int64	-8543978536212438961	-8,54398e+18
float64	-12152,78854212	-12152,8

5. REALIZACIJA I TESTIRANJE MODBUS SUNSPEC SERVER APLIKACIJE

5.1 Unapređenje Modbus C aplikacije

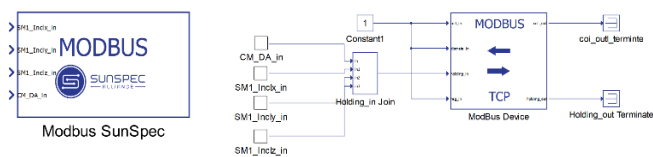
Kako se Modbus SunSpec server aplikacija oslanja na Modbus komponentu za uspostavljanje komunikacije sa klijentom, nije neophodno razvijanje zasebne C aplikacije za ovu komponentu. Međutim, novi tipovi registra koje definiše SunSpec zahteva određene izmene Modbus aplikacije.

Vrednosti određenih tipova registara se ne mogu menjati iz simulacije, bilo zato što podrška tog tipa ne postoji u simulaciji (*string* i *ip* tip) ili definicija SunSpec-a ne dozvoljava promenu vrednosti registra (*pad* tip). Za ove

tipove registara se ne definišu HIL adrese i vrednosti simulacije se ne čitaju u aplikaciji. Za ove tipove registara je takođe potrebno definisati inicijalne vrednosti, kako se one nikad ne ažuriraju. Inicijalizacija početnih vrednosti je dodata u konfiguracionom fajlu servera. Takođe, promena vrednosti ID servera u toku rada je podržana od strane SunSpec-a, pa je dodata podrška za promenu ove vrednosti od strane klijenta i simulacije.

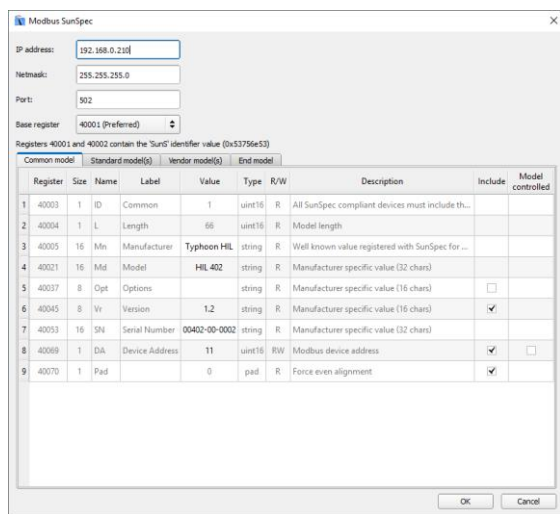
5.2 Kreiranje Modbus SunSpec server komponente

Ova komponenta koristi već realizovanu Modbus komponentu za uspostavljanje Modbus komunikacije sa klijentom. Iz tog razloga, Modbus SunSpec komponenta je realizovana kao kompozitna komponenta u čijem centralnom delu se nalazi Modbus sa dodatnim komponentama za povezivanje sa modelom. Iz tog razloga se ova komponenta može smatrati grafičkim interfejsom za kreiranje konfiguracije Modbus komponente. Izgled Modbus SunSpec komponente i iz kojih komponenti se sastoji prikazuje slika 7.



Slika 7. Izgled i sadržaj Modbus SunSpec komponente

Dijalog Modbus SunSpec komponente je dosta složeniji od Modbus komponente i koristi posebne kartice za prikazivanje različitih informacionih modela u vidu tabela (Slika 8).



Slika 8. Dijalog prozor Modbus SunSpec komponente

Common i *End* modeli su fiksni dok se *Standard* modeli mogu odabrati iz liste svih postojećih informacionih modela. Dodavanje *Vendor* modela je omogućeno kroz standardni Windows-ov prozor za odabir fajlova (eng. *Windows file browser*). Ako je registar definisan kao opcioni, korisnik može odabrati da li će registar biti implementiran ili ne. Takođe se može odabrati da li će se vrednost registra kontrolisati signalima iz modela ili se vrednost može menjati samo zahtevom klijenta.

5.3 Testiranje

Kako se uspostavljanje komunikacije oslanja na korišćenje Modbus komponente, pri testiranju Modbus SunSpec servera nije stavljen akcenat na čitanje i pisanje vrednosti

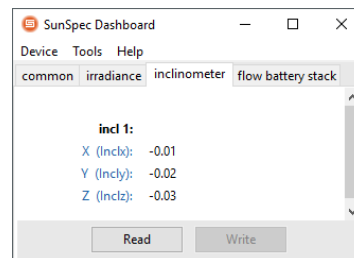
od strane klijenta. Akcenat testiranja je stavljen na proveru ispravnosti kreirane konfiguracije i inicijalizovanih vrednosti. Korišćeni program za testiranje, *SunSpec Dashboard*, je oficijalni program SunSpec alijanse koji se koristi za proveru da li je uređaj SunSpec kompatibilan. Nakon uspostave konekcije sa serverom, program automatski detektuje konfiguraciju i prikazuje vrednosti svih implementiranih registara.

Postavka sistema za testiranje Modbus SunSpec servera je identična kao i za testiranje Modbus servera (Slika 5). Na HIL uređaju je pokrenuta aplikacija, dok je na PC računaru pokrenut *SunSpec Dashboard* program.

Testirani su slučajevi kada postoji jedan i više *Standard* modela, jedan i više *Vendor* modela i kombinacija *Standard* i *Vendor* modela.

5.4 Rezultati testiranja

Testiranjem Modbus SunSpec servera su dobijeni rezultati koji ukazuju na ispravnost rada aplikacije. Sve konfiguracije se uspešno detektuju *SunSpec Dashboard* programom i sve inicijalne vrednosti definisane u konfiguraciji se poklapaju sa očitanim vrednostima. Primer očitavanja konfiguracije sa više *Standard* modela je prikazan na slici 9.



Slika 9. Očitana konfiguracija Modbus SunSpec servera

6. ZAKLJUČAK

U ovom radu je predstavljen proces razvoja i testiranja Modbus i Modbus SunSpec server aplikacije, kao i razvoj grafičkog interfejsa za konfiguraciju servera u cilju spajanja komponenata sa modelom i čitanja i upravljanja vrednosti simulacije pomoću spoljašnjih Modbus klijenata. Povećanje skupa podržanih funkcija Modbus servera, kao i grafički metod dodavanja i definisanja registara se može razmotriti kao unapređenje Modbus server aplikacije, a kao unapređenje Modbus SunSpec servera se može razmotriti grafičko definisanje *Vendor* modela i snimanje modela u standardnom SMDX formatu koji bi se verifikovali od strane SunSpec zajednice.

7. LITERATURA

- [1] Modbus Organization, Inc., *Modbus Application Protocol Specification V1.1b3*, April 2012.
- [2] SunSpec Alliance, *SunSpec Information Model Specification*, version 1.9
- [3] Modbus organization, Inc., *Conformance Test Specification for Modbus TCP Version 3.0*, 2009.

Kratka biografija:



Predrag Krža je rođen 24.09.1990. god. u Kikindi, Srbija. Osnovnu i srednju školu završio je u Kikindi. Osnovne akademske studije završio je na Fakultetu tehničkih nauka 2015. god. i nakon toga upisao master akademske studije na istom fakultetu Master rad iz oblasti Elektrotehnike i računarstva – Embedded sistemi i algoritmi, odbranio je 2017. god

TERMOVIZIJSKA DIJAGNOSTIKA ELEMENATA U RAZVODNIM ORMANIMA**THERMAL IMAGING DIAGNOSTICS OF ELEMENTS IN THE DISTRIBUTOR HOUSINGS**Jovan Knežević, Vladimir Katić, *Fakultet tehničkih nauka, Novi Sad***Oblast – Energetika, elektronika i telekomunikacije**

Kratak sadržaj – Termovizija ima široku primenu u raznim oblastima. U ovom radu su opisane fizičke osnove termovizije i mogućnosti primene u nekim oblastima. Posebno je analizirano održavanje elektroenergetskih sistema i upotreba termografije u upravljanju održavanjem. Detaljno su opisani uzroci nastajanja jedne vrste otkaza u elektroenergetskim uređajima i primena termovizije u njihovom otkrivanju. Na konkretnom primeru su pokazani svi potrebni postupci pri otkrivanju nesavršenosti u distributivnim razvodnim ormanima i predložene mere za njihovo otklanjanje.

Abstract – Thermography is widely used in various fields. This paper describes the physical basics of thermal imaging and application possibilities in some areas. Special analyzes the maintenance of power systems and the use of thermography in the maintenance management. There are detailed causes of one kind of failure in power devices and application of thermal imaging in their detection. In the particular case are shown all the necessary procedures for the detection of imperfections in the distribution electric boards and proposed measures for their elimination.

Cljučne reči: Termografija, Termovizija, Termovizijski snimak.

1. UVOD

Termovizija podrazumeva prikupljanje i obradu informacija na osnovu snimaka koji su dobijeni pomoću infracrvenog zračenja sa analiziranih objekata. Ima velike prednosti jer ne zahteva direktan kontakt sa mernim objektom i pri tome se dobijaju informacije o objektu pri punom radnom opterećenju. Potrebno je napomenuti da bezkontaktnost obezbeđuje to da merni uređaj ne utiče na merni objekat u smislu da ne remeti njegov rad. Takođe, velika prednost je i mogućnost merenja objekata koji su u pokretu.

Termovizijski snimak je dvodimenzionalan i omogućuje upoređivanje temperature više delova mernog objekta, čime se mogu jasno razlikovati mesta sa nepovoljnim karakteristikama, te preduzeti mere za otklanjanje uočenih nedostataka.

2. PRIMENA TERMOVIZIJE

Tokom svog razvoja termovizija je uglavnom svoju primenu nalazila u vojnoj industriji. U ranim periodima razvoja te primene se odnose na otkrivanje aviona na

noćnom nebu i otkrivanje traga podmornice, koji je u obliku trake toplije morske vode. Kasnije, kada su termovizijske kamere izrađivane sa većom rezolucijom, nalaze svoju primenu za noćno osmatranje, za noćnu vožnju tenkova i borbenih vozila. Krajem 90-ih godina dvadesetog veka proizvode se kamere koje su pristupačnije po ceni i tada dolazi do širenja oblasti primene termovizije na sledeće oblasti nauke i tehnike:

1.Mašinstvo. Na mestima gde se javlja trenje dolazi do pojave zagrevanja, i važno je uočiti da li ta zagrevanja prelaze dozvoljene granice da ne dođe do havarija na opremi. Povišena temperatura na delovima, gde se javlja trenje, uzrokuje smanjenje viskoznosti maziva i oštećenja kliznih ili kotrljajnih površina.

2.Elektrotehnika. Povećano zagrevanje se javlja na mestima loših električnih kontakata ili oštećenih provodnika. Takođe, javlja se i u elektronici pri preopterećenju aktivnih komponenti ili mikroprocesora.

3.Građevinarstvo. Za otkrivanje vlage na ravnim krovovima ili u zidovima, vlažni delovi imaju drugačije toplotne karakteristike. Za otkrivanje loših mesta zaptivanja na vratima, prozorima ili drugim vrstama poklopaca. Mogu se otkriti nepravilnosti u naleganju fasade na kruti deo zida, na osnovu promenjenih parametara toplotne provodljivosti usled pojave sloja vazduha. Na osnovu temperaturnih razlika mogu se prepoznati prevelika naprezanja u konstrukciji, tj. nosećim zidovima i nosećim stubovima. Termovizija pomaže i prilikom dodeljivanja energetskih pasoša objektima.

4.Medicina i veterina. Razlike u temperaturi u raznim delovima tkiva može ukazivati na neki upalni proces u tkivu ili na razoj ćelija kancera. Važno je napomenuti da termovizijsko snimanje nema štetnih efekata na živi organizam.

3. FIZIČKE OSNOVE TERMOVIZIJE

Toplota je oblik energije koji se ispoljava kroz unutrašnje oscilacije u atomu. Količina toplote utiče na temperaturu tela, koja se lako može meriti. Toplota prelazi sa tela sa većom temperaturom na tela sa manjom. Postoje tri osnovna načina prostiranja toplote kako je dato u [2]: provođenjem (kondukcija), prelaženjem (konvekcija) i zračenjem (radijacija).

Provođenje toplote je način prostiranja kod koga se unutrašnja energija jednog molekula provodi na drugi. Atomi tela se haotično kreću i međusobno sudaraju, atomi toplijeg dela tela koji se susudaraju sa susednim hladnijim i predaju im deo svoje kinetičke energije. Taj proces se

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Vladimir Katić, red.prof.

postepeno prostire kroz čitavo telo. Ovaj način prostiranja toplote karakterističan je za čvrsta tela.

Prelazjenje toplote je molarni vid prostiranja toplote tj. mora da postoji kretanje dela mase (mola) fluida (tečnosti ili gasa). Kretanjem materije u prostoru različito zagrejani delovi fluida prenose određenu količinu toplotne energije. Konvekcija može biti prouzrokovana prirodnim strujanjem fluida ili prinudnim (ventilatorom, kompresorom, pumpom, fenom i sl.).

Zračenje (radijacija) toplote je emitovanje unutrašnje energije u obliku elektromagnetnih talasa. Sva tela koja imaju temperaturu iznad apsolutne nule zrače toplotu ali i apsorbuju energiju koja do tela stiže u obliku elektromagnetnih talasa.

Na ovaj način se toplota može prenositi kroz prostor u kome nema materije tj kroz vakuum. Zračenje toplote nastaje kao posledica unutrašnjih oscilacija atoma, koje su ustvari kretanje naelektrisanih čestica protona i elektrona, čiji je rezultat emitovanje elektromagnetnih talasa.

3.1. Princip rada termovizijskog uređaja

Sva tela sa temperaturom višom od apsolutne nule emituju toplotu pomoću elektromagnetnih talasa. Ukupna emitovana energija i njena spektralna raspodela zavise od temperature tela i od koeficijenta emisivnosti. Tela najveći deo energije emituju u infracrvenom delu spektra (770 nm do 1 mm) i to se često naziva toplotno zračenje. Toplotno zračenje se fizički opisuje preko zračenja apsolutno crnog tela.

Kod tela koje se nalaze na sobnoj temperaturi (300 K ili 27°C), maksimum energije zračenja nalazi se na talasnoj dužini od 10 μm u dugotalasnoj infracrvenoj oblasti. Kao što je poznato, kroz atmosferu se može prostirati samo zračenje određenih talasnih dužina.

Postoje dva glavna prozora koja se nalaze na talasnim dužinama od 3 – 5 μm i od 8 – 14 μm, odnosno na frekvencijama od 100 THz – 60 THz i od 37,5 THz – 21 THz. Oba ova prozora kroz koje je moguće prostiranje infracrvenog zračenja kroz atmosferu koriste se za rad termovizijskih uređaja.

Termovizijski uređaji koji rade u infracrvenoj oblasti često se nazivaju skraćenicom FLIR (eng. forward looking real time infrared imaging system). Termovizijski uređaj služi za prevođenje nevidljivog infracrvenog zračenja u vidljivo zračenje i formiranje lika predmeta. Sopstveno zračenje posmatranog tela zajedno sa reflektovanim zračenjem drugih prirodnih izvora zračenja prostire se kroz atmosferu.

Neke informacije o predmetima u posmatranoj sceni mogu se izraziti kroz toplotni kontrast. Termovizijski uređaj omogućava vizuelizaciju toplotnog kontrasta zahvaljujući mogućnosti detektora da razlike u fluksu (snazi) primljenog infracrvenog zračenja pretvori u električni signal. Električni signal služi za generisanje kontrasta vidljive slike srazmerno toplotnom kontrastu posmatrane scene.

4. ODRŽAVANJE ELEKTROENERGETSKIH POSTROJENJA

Da bi se postigao efikasan i pouzdan rad elektroenergetskog sistema potrebno je izvršavati sledeće aktivnosti: preventivno održavanje sistema i testiranje, otklanjanje nedostataka, analiza kvarova i analiza trenda održavanja.

4.1. Održavanje EE postrojenja uz primenu IC uređaja

Za svakog korisnika IC termografije, snimak se obrađuje na računaru posebnim programskim paketom za analizu IC zapisa, a završno se izdaje Protokol - Izveštaj (ili nalaz - atest) snimljenog sa oznakom defekta i preporukom sanacije. Upravo je definisanje i prepoznavanje graničnih vrednosti otkrivenih nepravilnosti predmet izučavanja i poznavanja pojedine grane delatnosti i svakako zahteva stručne analize koje se stiču kroz obavezno školovanje termografista i naravno iskustvo u raznim radnim uslovima.

Kao bezkontaktna temperaturna merna metoda infracrvena termografija omogućuje korisniku otkrivanje raznih potencijalnih grešaka i to bez potrebe prekida procesa proizvodnje i troškova koji su prouzrokovani tim prekidom. Pojava pregrevanja u mnogim slučajevima ukazuje na stanje kvara. Termografskim snimanjem mogu se pratiti stanja elemenata za prenos električne energije, rashladnih postrojenja, transformatorskih stanica, kao i same proizvodnje električne energije. Isto tako može se pratiti stanje izolacije, mreže cevovoda u procesnoj industriji, vrelovoda, kvalitete obloga peći za topljenje, rotacionih peći u cementnoj industriji, stanje ležajeva. U okviru zaštite od požara, termografsko snimanje se koristi za otkrivanje latentnih požara, pronalaženja osoba u objektu zahvaćenom požarom, ispitivanja elemenata na otpornost od požara. IC termografija se može koristiti za nadzor objekata, prostora, saobraćaja i zagađenja okoline. U građevinarstvu se primenjuje kod ispitivanja kvaliteta izolacije objekta, utvrđivanja mesta s povećanom vlagom, itd. Primenjuje se i u medicini, naučno-istraživačkom radu iz područja provođenja toplote i mehanike fluida.

Pomoću primene IC termografske tehnike i IC uređaja moguće je u toku normalnog procesa rada, bez ostvarivanja kontakta s objektom koji se ispituje, otkrivanje povišenog zagrevanja električkih i mehaničkih komponenti, čime se sprečavaju kvarovi odnosno ukazuje se na lošu ili neodgovarajuću toplotnu izolaciju. Na taj način smanjuje se broj neplaniranih ispada pogona. Direktni efekti su ubrzanje postupka dijagnostike kvara i provera preduzetih mera, ušteda energije, zaštita kapitalne opreme. Povećanje raspoloživosti opreme, uz potvrdu njene pouzdanosti, ali i ukazujući na moguća kritična mesta, povećava se i ukupno vreme rada, odnosno proizvodnje.

Ispitivanje električne opreme najčešće ukazuje na probleme uzrokovane odnosima struje i napona. Prosto rečeno, "vruće mesto" u električnom strujnom kolu pojavljuje se kao posledica npr. nedovoljno pritegnutog, oksidiranog ili korodiranog spoja, ali i nepravilnog rada samog aparata. Zato i IC termografija nalazi primenu u područjima kao: proizvodnja el. opreme, preventivno održavanje postrojenja (u proizvodnji el. energije, u prenosu i distribuciji, u industriji i transportu).

Tabela 4.1: Klasifikacija stanja EE opreme

Delta T	Preporučena akcija	Klasa
>30°C	HITNA intervencija!	'A'
od 5°C do 30°C	Intervencija kod prvog zaustavljanja pogona	'B'
do 5°C	Praćenje stanja i planiranje skore intervencije	

Stanje EE opreme na kojoj je uočeno povećanje temperature (prema novoj klasifikaciji) može se po Džulovom zakonu uzeti za osnovu definisanja kriterijuma prema poznatoj referentnoj i korektnoj temperaturi, gledano pod istim radnim uslovima, i uzimanjem u obzir dobijene temperaturne razlike izražene kroz „delta t“. Ova klasifikacija stanja elektroenergetske opreme urađena je prema "ITC"-u (Infrared Training Center) i opisana u [3]. Da bi se odredilo nazivno povećanje temperature kod opterećenja koja su manja od 100% -nog, koristimo dakle za proračun Džulov zakon, što je takođe predmet pomenutog školovanja termografista.

Preporučuje se za kontrolu i inspekciju EE postrojenja uvođenje tzv. "ciklusa održavanja" pomoću IC praćenja stanja el. opreme, a koji se može prilagoditi gotovo svakoj industrijskoj sredini. Tako se npr. mogu dobiti podaci: o tipovima popravaka, o broju komponenti koje su zamenjene, o potrebnom broju rezervnih delova, o ostalim detaljima važnim za pogonsku spremnost postrojenja. Moguće je izvršiti ocenu efikasnosti urađenih popravki kao i njihovog periodičnog praćenja, a koje ima za cilj smanjenje kvarova, posebno onih najtežih.

5. REALIZACIJA IC SNIMANJA

Objekti predviđeni za merenje pomoću snimanja termovizijskom kamerom nalaze se u Univerzitetskom kampusu u Novom Sadu. Oni su sastavni deo elektroenergetske mreže za napajanje Univerzitetskog kampusa. Reč je o razvodnim ormanima za: solarnu elektranu na krovu mašinskog instituta, za kulu univerziteta, za amfiteatar A1 i A2 i za razvodni orman kantine.

5.1. Oprema za snimanje

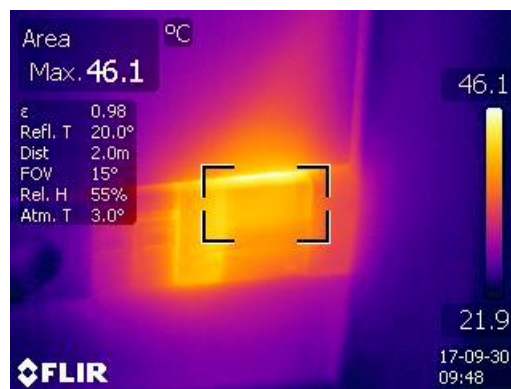
Termovizijsko snimanje obavljeno je sa termovizijskom kamerom „FLIR T335“. U tabeli su prikazane tehničke karakteristike kamere.

Tabela 5.1: Osnovne karakteristike IC kamere

Karakteristike snimanja	Vrednosti za IC kameru: T335
Termička osetljivost	0,05°C kod 30°C
Rezolucija	320x240
Zumiranje	1 do 4 neprekidno, digitalni zum
Temperaturni opseg mernog objekta	- 20°C do 120°C 0°C do 350°C 200°C do 650°C

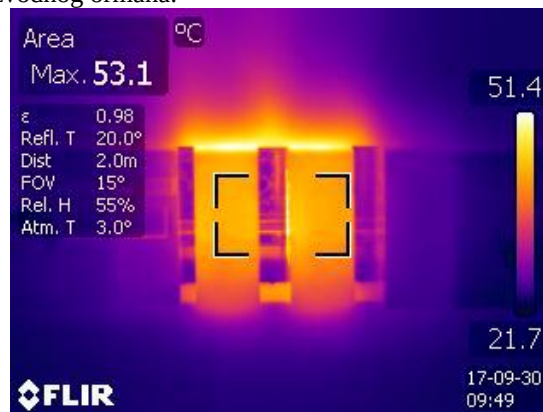
5.2. Analiza rezultata merenja

Termografskim snimanjem razvodnog ormana za napajanje anfiteatra A1 uočeni su nedostaci na kontaktoru sve tri faze u razvodnom ormanu rasvete i sistema ventilacije. Merna tačka je postavljena na spoj kontaktora i dovodnih kablova sve tri faze. Uočeno je toplo mesto (slika 1) na kontaktoru faza i izmerena temperatura toplog mesta iznosi 26°C. Toplo mesto je okarakterisano kao srednje pregrevanje i u skorijem periodu potrebno je zameniti kontaktor sa ispravnim.



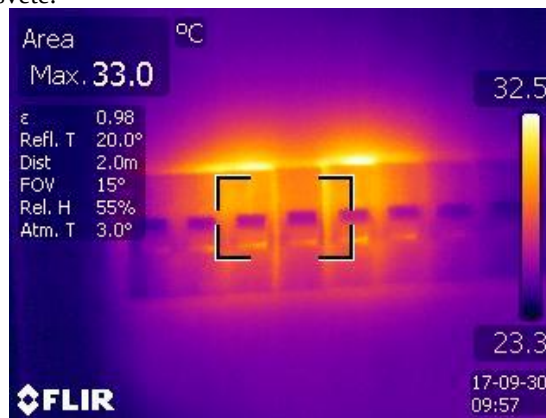
Slika 1. Snimak kontaktora

Termografskim snimanjem razvodnog ormana za napajanje amfiteatra A2 uočeni (slika 2) su nedostaci na kontaktorima sve tri faze K1 i K2. Merna tačka je postavljena na spoj kontaktora K1 i dovodnih kablova sve tri faze. Uočeno je toplo mesto na kontaktorima K1 i K2 i izmerena temperatura toplog mesta iznosi 33°C. Toplo mesto je okarakterisano kao veliko pregrevanje. Potrebno je hitno zameniti kontaktore K1 i K2 i uraditi reviziju razvodnog ormana.



Slika 2. Snimak kontaktora

Termografskim snimanjem razvodnog ormana za napajanje rasvete anfiteatra A2 uočeni (slika 3) su nedostaci na grupi osigurača u razvodnom ormanu rasvete.



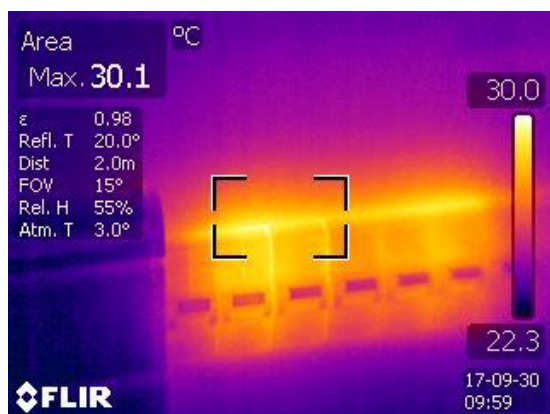
Slika 3. Snimak grupe osigurača

Merna tačka je postavljena na spoj osigurača broj 25. Uočeno je toplo mesto na osiguračima br.25 i br.27 i izmerena temperatura toplog mesta iznosi 13°C. Toplo

mesto je klasifikovano kao srednje pregrevanje i u skorijem periodu potrebno je očistiti spojeve na osiguračima i dotegnuti kontakte.

Termografskim snimanjem razvodnog ormana za napajanje rasvete amfiteatra A1 uočeni (slika 4) su nedostaci na osiguračima br.24 i br.28 u razvodnom ormanu rasvete. Merna tačka je postavljena na spoj osigurača broj 24.

Uočeno je toplo mesto na osiguračima br.24 i br.28 i izmerena temperatura toplog mesta iznosi 15°C. Toplo mesto je klasifikovano kao srednje pregrevanje i u skorijem periodu potrebno je očistiti spojeve na osiguračima i dotegnuti kontakte, pa zatim uraditi reviziju.



Slika 4. Snimak grupe osigurača

Termografskim snimanjem razvodnog ormana za napajanje amfiteatra A1 uočeni (slika 6) su nedostaci na kontaktorima sve tri faze K1 i K2. Merna tačka je postavljena na spoj kontaktora K1 i dovodnih kablova sve tri faze. Uočeno je toplo mesto na kontaktorima K1 i K2 i izmerena temperatura toplog mesta iznosi 47°C. Toplo mesto je klasifikovano kao veliko pregrevanje. Potrebno je hitno zameniti kontaktore K1 i K2 i uraditi reviziju grejnog mesta.



Slika 6. Snimak kontaktora

6. ZAKLJUČAK

Održavanje funkcionalnosti elektroenergetskih sistema je važan segment u privredama razvijenih država. Prvenstveno što stabilno napajanje električnom energijom za privredu znači izbegavanje gubitaka u prekidu proizvodnje zbog nestanka napajanja.

Primena termografije u održavanju elektroenergetskih sistema znači traženje uzroka neispravnosti na samom izvoru nastanka. Registrovanje mesta na kome se javlja pregrevanje jasan je pokazatelj nesavršenosti izrade određenog elementa u elektroenergetskom sistemu. Raznim postupcima mogu se pregrevanja klasifikovati i odrediti pravila po kojima se donose odluke u smislu koje mere je potrebno preduzeti u cilju otklanjanja nedostataka.

U ovom radu su opisani postupci kod termovizijskog snimanja, analize snimaka i predlaganje konkretnih mera za otklanjanje nedostataka. Postupci su opisani na konkretnom primeru snimanja ključnih elektroenergetskih objekata lociranih u univerzitetskom kampusu Novi Sad.

Opšte je poznato da kvarovi u elektroenergetskom sistemu mogu prouzrokovati velike materijalne štete i teško procenjive indirektno štete. Kod tradicionalnih načina održavanja ne mogu se na efikasan način oceniti loša mesta u elementima sistema. Termovizija je ovde našla praktična i efikasna rešenja za pronalaženje nedostataka. Posebno je značajno što se termovizijsko snimanje može obavljati kada je sistem u pogonu i što se merenja mogu obavljati i na većim udaljenostima.

7. LITERATURA

- [1] D. Stanivuković, S. Kecojević, I. Beker, "Projektovanje postupaka održavanja proizvodnih sistema", IIS, Novi Sad, 1992.
- [2] A. Dedić, „Osnovi mašinstva I“, Beograd, 2009.
- [3] I. Kuzle, H. Pandžić, „Održavanje elektroenergetskog sustava“, FER, Zagreb, 2012.
- [4] Lj. Čičkarić, „Primena termografije u dijagnostici toplotnih stanja energetske transformatora“, Elektrotehnički institut „Nikola Tesla“, Beograd, 2009.
- [5] K. Petrović, „Infracrvena termografija – pravi izbor za redovno održavanje“, 1. Konferencija „ODRŽAVANJE 2010“, Zenica, 2010.

Kratka biografija:



Jovan Knežević rođen je u Kninu (R.Hrvatska) 21.6.1987. god. Master rad na Fakultetu tehničkih nauka iz oblasti Energetike, elektronike i telekomunikacija – Obnovljivi izvori energije odbranio je 2017.god.



dr Vladimir A. Katić, red.prof. rođen je 1954. godine u Novom Sadu. Diplomirao je na Fakultetu tehničkih nauka u Novom Sadu 1978. god., a magistrirao i doktorirao na Univerzitetu u Beogradu 1981. i 1991. godine, respektivno. Od 2002. godine je redovni profesor Univerziteta u Novom Sadu. Trenutno je prodekan Fakulteta tehničkih nauka i šef Katedre za energetske elektroniku i pretvarače. Oblasti interesovanja su energetska elektronika, obnovljivi izvori električne energije, kvalitet električne energije i električna vozila.

ПРОЈЕКТОВАЊЕ МОБИЛНЕ АПЛИКАЦИЈЕ ЗА ЕВИДЕНЦИЈУ СТАЊА У АЗИЛИМА ПАСА У ПРОГРАМСКОМ ЈЕЗИКУ ЈАВА**DESIGNING A MOBILE APPLICATION FOR RECORDING THE STATE OF DOG'S SHELTERS IN JAVA PROGRAMING LANGUAGE**

Ана Лукић, Факултет техничких наука, Нови Сад

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – У овом раду је приказана имплементација мобилне апликације за евиденцију стања у азилима паса. Циљ овог рада јесте приказ примене програмских језика Андроид и Јава приликом пројектовања мобилне апликације. Поред тога, акценат је стављен и на примену како основних објектно-оријентисаних концепата тако и на примену дизајн шаблона који процес имплементације чине лакшим и једноставнијим.

Abstract – *Implementation of mobile app for registering the status in animal shelters for dogs. The aim is to show the application of Android and Java programming language in the application design. Object oriented concepts and design pattern application is shown here, as a mean for simplifying the design.*

Кључне речи: Java, Android, Hibernate

1. УВОД

Суштина овог рада је примена објектно-оријентисаног програмирања за развој мобилне апликације која би требала да презентује стање у азилима паса на нивоу града.

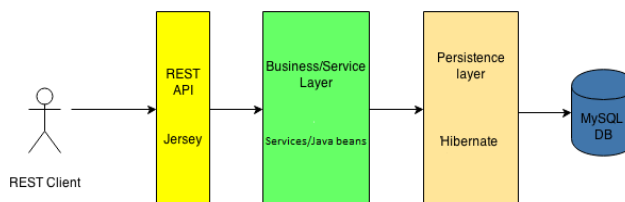
С обзром да смо сведоци убрзаног развоја мобилних и web технологија, као и то да смо упознати са манама и врлинама истих, употреба оваких технологија као средство подизања свести о стању живота животиња око нас представља добар пример како се исте могу искористити у хумане сврхе. У наставку је дат кратак опис сваког поглавља :

- Поглавље „Организација серверске стране апликације”, описује имплементацију серверске стране у програмском језику Јава, од мапирања података коришћењем Hibernate-а па све до слања података, при чему се посебан акценат ставља на имплементацију REST-ful сервиса ...
- Поглавље „Организација клијентске стране апликације”, описује имплементацију клијентске стране апликације, архитектуру решења, опис коришћеног шаблона, при чему се посебан акценат ставља на имплементацију Retrofitа ...
- Поглавље „Модел података”, садржи приказ концептуалног модела података имплементације мобилне апликације за евиденцију стања паса у азилима на нивоу града.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Александар Купусинац, ред.проф.

- Поглавље „Кориснички интерфејс апликације“, садржи детаљан приказ имплементираних решења за евиденцију стања паса у азилима, укључујући илустрације коришћења апликације.
- Поглавље „Коришћени алати и технологије приликом имплементације“, садржи кратак опис

2. ИМПЛЕМЕНТАЦИЈА**2.1. Организација серверске стране апликације**

Слика 1. Софтверска архитектура имплементације серверске стране апликације

Апликација која је тема овог рада се састоји од две апликације (клијентске и серверске). Клијентска апликација представља Андроид апликацију док је за имплементацију серверске апликације изабран *Spring framework*.

Овај развојни систем преставља систем за развој апликација базиран на *Java* програмском језику.

Приликом имплементације апликације која је тема овог рада дошло се до закључка да архитектуру серверске стране апликације је потребно имплементирати тако да садржи три слоја као што се може приметити на слици 1. Овај начин имплементације апликацију дели на три функционална слоја: слој за комуникацију са клијентима (*Rest API*), слој за имплементацију основних функционалности система (ткз. Бизнис логика), и слој за руковање подацима.

Сваки од ових слојева је функционално независан од других слојева што доприноси флексибилности система, лакше кодирање и прегледност система.

Овај вид архитектуре се базира на *MVC* шаблону који је чест и врло користан шаблон код веб апликација пошто испуњава све услове за лакше кодирање и одржавање система.

Апликација се састоји од три слоја:

- *Jersey REST*-фул web сервиси
- *Business/Services* слој
- *DAO* слој

2.1.1. Jersey REST-ful сервиси

Приликом имплементације серверске стране апликације описане у овом раду, за имплементацију *web* сервиса коришћени су *Jersey REST*-фул сервиси. *Jersey REST*-фул сервиси представљају *open-source* компоненту за развој сервиса на *Java* програмском језику која пружа подршку кроз *JAX-RS API*.

REST је архитектура имплементације сервиса која се базира на *web* стандарде и ослања се на *HTTP* протокол. У овој архитектури свака компонента система се може посматрати као ресурс којем се приступа преко интерфејса базираном на *HTTP* стандардним методама. Код ове архитектуре постоји *REST* сервер који пружа приступ ресурсима. Основне методе које су подржане код овог вида сервиса су:

- *GET* – дефинише читање и приступање ресурсу без нежељених промена. Код ове методе ресурс се не мења.
- *PUT* – ова метода се користи када је потребно креирати нови ресурс на серверској страни.
- *DELETE* – брише ресурс на серверској страни и може се понављати више пута
- *POST* – ова метода мења садржај и стање постојећег ресурса или креира нови ресурс уколико не постоји

У наставку биће приказан пример имплементације *REST*-фул сервиса за *login* функционалност:

```
@Path("/userservice")
public class UserRestService extends BaseRestService{
    public UserRestService() { super(); }
    @GET
    @Path("/login/{username}/{password}")
    @Produces(MediaType.APPLICATION_JSON)
    public String login(@PathParam("username") String
        username, @PathParam("password") String password) {
        User user = userService.
            findUserWithUsernameAndPassword(username, password);
        if(user != null){
            return gson.toJson(user);
        }
        return "NotExist";
    }
}
```

Као што се може видети у коду, коришћене су анотације типа *@Path* и *@GET* како би се дефинисао начин на који се обрађује *HTTP* захтев.

2.1.2. DAO слој

Последњи слој је слој који је намењен за руковање и управљање базом података. Имплементација овог слоја се базира на *DAO* објекте који комуницирају са базом података. *DAO* објекат (*database access object*) је објекат који пружа апстрактни интерфејс за одређене типове базе података или друге перзистентне системе. Приликом позива неке методе из овог слоја, *DAO* објекат обезбеђује приступ методи без пружања информације о начину имплементирања те методе као ни информације о томе која је база података у питању.

Предност коришћења ових објеката огледа се у релативно једноставном и ригорозном одвајању два битна дела апликације на тај начин да један и други део апликације немају никакве информације о њиховој имплементацији. Уколико дође до промене у бизнис логици,

имплементација *DAO* објеката може да остане нетакнута и исто важи за супротну ситуацију. Сви подаци о складишту су сакривени од остатка апликације и самим тим, могућа промена на перзистентном механизму може бити имплементирана само модификацијом једног *DAO* објекта док ће остатак апликације бити нетакнут.

Пример имплементације *DAO* слоја за ентитет *Корисник* приказан је у наставку:

```
@Repository („userDao“)
public class UserDaoImpl extends
    CustomHibernateDaoSupport implements UserDao {
    @Override
    public User findUserWithUsernameAndPassword(String
        username, String password) {
        Object[] params = {username, password};
        String[] paramsS = {„username“, „password“};
        List<User> list = getHibernateTemplate().
            findNamedQueryAndNamedParam (
                User.FIND_USER_WITH_USERNAME_AND_PASSWORD,
                paramsS, params);
        if(!list.isEmpty()){
            return list.get(0);
        }
        return null;
    }
}
```

2.1.3. Hibernate

Hibernate ORM представља алат за објектно-релационо мапирање објеката базиран на *Java* програмском језику. Овај алат пружа развојни систем за мапирање објектно оријентисаног модела на релациони модел базе података. *Hibernate ORM* је *open-source* библиотека која мапира *Java* класе са табелама базе података, као и *Java* типове података са типовима података дефинисаним у релационој бази.

Мапирање *Java* класа на табеле базе података се може имплементирати помоћу конфигурације система уз помоћ *XML* фајла или преко анотација. У апликацији која је тема овог рада коришћен је други приступ који је савременији и прегледнији за кодирање. Поред табела, *Hibernate* пружа могућност и мапирање веза између табела уз помоћ одговарајућих анотација (*one-to-many* или *many-to-one*).

Пример мапирања табеле *Корисник* који је имплементиран коришћењем *Hibernate* библиотеке приказан је у наставку:

```
@Entity
@NamedQuery (name =
    User.FIND_USER_WITH_USERNAME_AND_PASSWORD,
    query = "FROM User user WHERE user.username = :username AND
        user.password = :password")

@Table(name = "User", catalog = "mydb")
public class User implements Serializable {

    public final static String
        FIND_USER_WITH_USERNAME_AND_PASSWORD =
            "findUserWithUsernameAndPassword";

    private int idUser;
    private String username;
    private String password;
    public User() {}

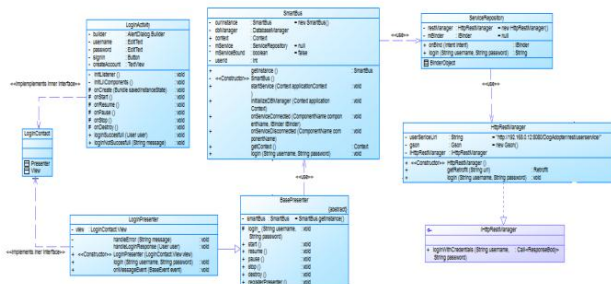
    @Id
    @GeneratedValue(strategy = IDENTITY)
```

```
@Column(name = "idUser", unique = true, nullable = false)
public int getIdUser() { return idUser; }

@Column(name = "username", unique = true, nullable = false, length
= 45)
public String getUsername() { return username; }

@Column(name = "password", unique = true, nullable = false, length
= 45)
public String getPassword() { return password; }
```

2.2. Ориганизација клијентске стране апликације



Слика 2. Концептуални модел клијентске стране апликације

У овом поглављу описано је мало више о самој имплементацији клијентске стране апликације. На слици 2 приказан је концептуални модел клијентске стране апликације. Улазну тачку представља Андроид компонента *Activity* која поседује графички интерфејс и директно комуницира са самим корисником апликације. Она спада у 4 основних Андроид компоненти: *Activity*, *Service*, *BroadcastReceiver*, *ContentProvider* при чему сваку од њих одликује засебни животни циклус.

Да би активити био видљив од стране система, потребно је да буде дефинисан у склопу *AndroidManifest.xml* фајла који представља документ у коме су дате есенцијалне информације о апликацији које су потребне оперативном систему пре покретања апликације.

У апликацији која је тема овог рада је коришћен *MVP* шаблон. Сама идеја *MVP* шаблона и имплементираних решења јесте таква да активити не комуницира директно са сервисима или осталим компонентама бизнис логике и апликације, већ комуникације иде преко презентера у коме се налази сва логика која се не тиче корисничког интерфејса активитија. Највећа предност *MVP* шаблона јесте постизање независности кода корисничког интерфејса од кода који је део бизнис логике.

Једна од компоненти која је коришћена у апликацији која је описана у овом раду јесте сервис. Начин на који је дефинисана у манифест фајлу је дат у следећем коду:

```
<service android:name=".backend.service.ServiceRepository" />
```

Сервис представља компоненту која нема свој кориснички интерфејс и користи се за извршавање корисничких задатака у позадинском процесу који је одвојен од главне апликационе нити. Постоје два типа сервиса: *Start* и *Bound*. Оба типа се дефинишу на исти начин али се покрећу на различите начине.

Начин покретања *Start* сервиса се разликује од покретања *Bound* сервиса. *Start* сервис се покрећу експлицитно у коду, тј. у коду мора постојати тачно команда која покреће сервис и команда која исти сервис гаси. Уколико се команда за деактивацију сервиса не изврши сервис ће се извршавати независно од апликације, тј. уколико се апликација угаси сервис ће и даље бити активан. *Bound* сервис се разликује у томе што се не извршава самостално и независно од апликације. *Bound* сервис су активни све док је активна компонента на коју је тај сервис референциран.

У апликацији која је тема овог рада је коришћен *Bound* тип сервиса али на начин који омогућава да се сервис извршава све док је апликација активна, тј. сервис није везан за активити која је тренутно приказан.

Сваки сервис мора наследити класу *Service* која захтева имплементацију одређених њених метода. Једна од метода које је потребно имплементирати јесте *onBind()* метода. Ова методу је обавезно имплементирати код *Bound* сервиса и имплементација треба да садржи креирање *Binder* објекта који се користи за комуникацију између сервиса и компоненте која је повезана са сервисом.

У апликацији се сервис покреће на следећи начин:

```
public void startService(Context applicationContext) {
    Intent intent = new Intent(applicationContext, ServiceRepository.class);
    context.bindService(intent, this, Context.BIND_AUTO_CREATE);
}
```

Када се сервис покреће на горе дати начин, оперативни систем тражи класу сервиса у манифест фајлу апликације. Када га пронађе, прва метода коју оперативни систем извршава јесте *onBind()*. Систем позива метод *onBind()* да генерише *IBinder* објекат само када се први корисник везује. Систем испоручује исти објекат свим додатним клијентима који се везују за исти сервис без поновног позивања *onBind()* методе. Када последњи клијент одустане од услуге, систем уништава сервис.

Један од разлога зашто је у апликацији коришћен сервис јесте комуникације са сервером преко интернета.

Као компонента преко које ће се обављати интернет комуникација одабрана је *Retrofit* компонента. *Retrofit* је *REST* клијент за Андроид базиран на *Java* језику и у тој компоненти се релативно лако обавља слање и пријем порука (са или без *JSON* објекта у *body* секцији).

Пре слања захтева на сервер, подаци се конвртују у *JSON* објекат и као такави се уграђује у тело захтева. За конвертовање података у *JSON* у апликацији је коришћена *Gson* библиотека. У наставку приказана је имплементација *login()* методе уз употребу *Retrofit*-а:

```
public class HttpRestManager {
    private static final String userSeviceUrl =
        "http://192.168.0.12:8080/DogAdopter/rest/userservice/";
    private final Gson gson = new Gson();
    public HttpRestManager() {}
    private IHttpRestManager iHttpRestManager;
    public Retrofit getRetrofit(String url) {}
}
```



```

HttpLoggingInterceptor interceptor = new HttpLoggingInterceptor();
interceptor.setLevel(HttpLoggingInterceptor.Level.BODY);
OkHttpClient client = new OkHttpClient.Builder()
    .addInterceptor(interceptor).build();
Retrofit retrofit = new Retrofit.Builder()
    .client(client)
    .addConverterFactory(GsonConverterFactory.create())
    .baseUrl(url)
    .build();
return retrofit;
}

public void login(String username, String password) {
    Retrofit retrofit = getRetrofit(userServiceUrl);
    IHttpRequestManager = retrofit.create(IHttpRequestManager.class);
    IHttpRequestManager.loginWithCredentials(username,
        password).enqueue(new Callback<ResponseBody>() {
        @Override
        public void onResponse(Call<ResponseBody> call,
            Response<ResponseBody> response) {
            if (response.isSuccessful()) {
                try {
                    String stringResponse = response.body().string();
                    User user = gson.fromJson(stringResponse, User.class);
                    EventBus.getDefault().post(new LoginEvent(user));
                } catch (IOException e) {
                    EventBus.getDefault().post(new
                        ErrorEvent(e.getMessage()));
                    e.printStackTrace();
                }
            }
        }
    });
    @Override
    public void onFailure(Call<ResponseBody> call, Throwable t) {
        EventBus.getDefault().post(new ErrorEvent(t.getMessage()));
    });
}
}
}

```

Оно што одликује *Retrofit* као такав јесте његова једноставна употреба пошто не захтева пуно кодирања, подаци се лако конвертују и није потребно водити рачуна о *multitasking*-у. С саме методе које користе *Retrofit* дефинисане у склопу интерфејса *IHttpRequestManager*. За сваку методу, коришћене су *HTTP* алатице на основу којих ретрофит генерише *HTTP* захтев.

```

public interface IHttpRequestManager {
    @GET("login/{username}/{password}")
    Call<ResponseBody> loginWithCredentials(@Path("username")
        String username, @Path("password") String password);
}

```

3. ЗАКЉУЧАК

У претходним деловима раде презентоване су урађене функционалности у постојећој апликацији и представљено је како развој мобилних и web технологија може имати позитиван утицај на друштво. Постоји велики простор за даљу надоградњу саме апликације.

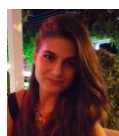
На пример, било би врло корисно уколико би изгубљени пас могао да се лоцира на мапи и то у истом тренутку на локацији на којој је примећен. Уколико корисник дозволи апликацији пристизање најновијих обавештења и приступ тренутној локацији корисника преко GPS-а, корисник би могао да добије обавештење да се у кругу од 1,5 км налази изгубљен пас. То би могло утицати на умањење временског периода потребног за проналажење изгубљених паса.

Следећа могућа функционалност која би требала да се развије јесте могућност да службе за зоохигијену чим ухвате и доведу пса могу да објаве слику пса и његове основне податке у склопу апликације (ово би могло и да се регулише обавезним законским актом где ће пас пре даљих поступања са њим морати да стоји објављен у одређеном временском периоду). Ово би било имплементирано као посебан део апликације.

4. ЛИТЕРАТУРА

- [1] Wikipedia, <https://sh.wikipedia.org>
- [2] Vogella, <http://www.vogella.com/tutorials/>
- [3] Imgur, <https://imgur.com/>
- [4] Spring, <https://spring.io/tools>
- [5] AndroidStudio, <https://developer.android.com/studio/index.html>
- [6] Facebook documentation, <https://www.programmableweb.com>
- [7] I.Dejanović, Софтверски обрасци и компоненте, материјали са предавања, Универзитет у Новом Саду, Факултет техничких наука, 2015
- [8] Jersey REST-ful services, <https://jersey.github.io/>
- [9] Hibernate, <https://docs.jboss.org/hibernate/>

Кратка биографија:



Ана Лукић је рођена 19.08.1992. у Приштини. Године 2011. уписала се на Факултет техничких наука, одсек Рачунарство и аутоматика, где је и дипломирала 2016. године.

DINAMIČKA ANALIZA RADA MIKROMREŽE**PERFORMANCE ANALYSIS OF MICROGRID DECENTRALIZED CONTROL**Njegoš Janković, *Fakultet tehničkih nauka, Novi Sad***Oblast – Energetika, elektronika i telekomunikacije**

Kratak sadržaj – Sa porastom broja distributivnih generatora (DG) povezanih na mrežu, javlja se potreba za usvajanjem novog koncepta pogodnog za njihovo povezivanje i pravilan rad. Mikromreža je koncept koji omogućava rješavanje ovih problema i omogućava poboljšanje stabilnosti u glavnoj mreži. Nekoliko distributivnih generatora je povezano na mikromrežu preko invertora. U ovom radu pažnja je posvećena načinu kontrole koji bi omogućio raspodjelu proizvedene snage između generatora. Svaki modul može povećavati frekvenciju i amplitudu izlaznog napona kako bi raspodijelio snagu sa drugim generatorima u skladu sa odnosom njihovih nazivnih snaga. Ovaj princip je preuzet iz principa rada sinhronog generatora.

Abstract – As number of distributed generators connected to the grid increases, there is need for implementation of concept suitable for their proper connection and operation. Microgrid is concept able to solve those problems and to increase stability of the main grid. Several distributed generators (DG) are connected to the microgrid via inverters. This paper focuses on control method for power sharing between generators in decentralized fashion. Every module is capable of adjusting output voltage frequency and magnitude in order to share power generation according to power rating of generators. This concept is taken from working principle of synchronous generator.

Ključne reči: mikromreža, microgrid, decentralizovana kontrola, droop karakteristike

1. UVOD

Povezivanje sve većeg broja obnovljivih izvora energije i uređaja za skladištenje energije na distributivnu mrežu zahtijeva promjenu koncepta centralizovanog sistema mreže. Ova promjena treba da prepozna proizvodnju energije na nivou distributivne mreže, bliže krajnjim potrošačima. Koncept mikromreže se nameće kao rješenje koje je u stanju da integriše distributivne generatore u postojeći sistem. Mikromreže se koriste kako bi se prevazišao problem povezivanja distributivnih generatora na decentralizovani način, pojednostavio upravljački algoritam zadužen za pravilan rad glavne mreže i samim tim otvorio put ka implementaciji koncepta pametne mreže u budućnosti.

Mikromreža je skup distributivnih generatora, uređaja za skladištenje energije i potrošača povezanih u mrežu koji radi u ostrvskom režimu rada ili povezan na glavnu mrežu.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Grabić, vanr.prof.

Ovdje, pažnja je usmjerena na rad mikromreže u ostrvskom režimu rada sa nekoliko distributivnih generatora i način raspodjele snage.

Kada se govori o upravljačkom algoritmu, mogu se definisati dva pristup [1]:

- Decentralizovani pristup koji iziskuje lokalni kontroler za svaki modul bez bilo kakvih informacija o aktivnostima drugih modula u mikromreži.
- Centralizovani pristup zahtijeva centralni kontroler i složen upravljački algoritam kako bi se ispunili svi uslovi u pogledu stabilnosti i pouzdanosti. Oslanja se na podatke prikupljene na svim elementima u mikromreži i obimnoj komunikaciji između centralnog kontrolera i svih kontrolisanih uređaja.

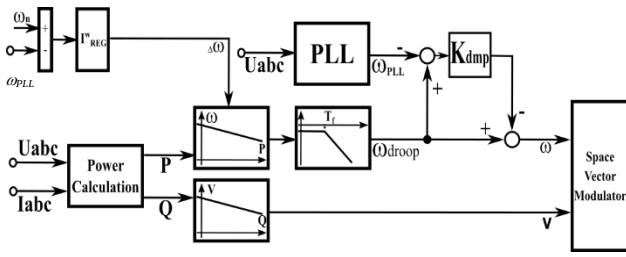
Oba pristupa imaju prednosti i mane te je neophodno naći kompromis. Kompromis može biti postignut uvođenjem nivoa kontrole: primarnog i sekundarnog. Prvi nivo kontrole je preuzet iz potpuno decentralizovanog pristupa i zadužen je za upravljanje amplitudom i frekvencijom izlaznog napona kako bi se proizvela snaga za potrošnju, ali istovremeno i ostvarila raspodjela snage između generatora. Drugi nivo kontrole je preuzet iz potpuno centralizovanog pristupa i zadužen je za dovođenje frekvencije na nazivnu vrijednost u cijelom sistemu i dovođenje efektivne vrijednosti napona na nazivnu vrijednost u jednom čvoru u mikromreži.

2. DECENTRALIZOVANO UPRAVLJANJE RADOM MIKROMREŽE

Primarna kontrola je odgovorna za izvršenje raspodjele snage između distributivnih generatora primjenjujući $P-\omega$ i $Q-V$ droop upravljačke karakteristike i podešavanje izlaznog napona generator bez direktne komunikacije između modula. Način na koji se ostvaruje raspodjela snage je zasnovan na droop metodi, koja je preuzeta iz načina rada sinhronog generatora.

Naime, sinhroni generator povezan na mrežu mijenja amplitudu ili učestanost izlaznog napona ukoliko dođe do nejednakosti proizvedene električne snage na izlazu i mehaničke snage na ulazu. U slučaju nejednakosti mehaničke i električne aktivne snage, brzina obrtanja rotora se mijenja. Ovo se prenosi na promjene u učestanosti. Na isti način, promjene u izlaznoj reaktivnoj snazi dovode do promjena u amplitudi izlaznog napona.

U slučaju distributivnog generatora povezanog na mikromrežu preko invertora, ovakvo ponašanje se može oponašati regulacijom amplitude i učestanosti izlaznog napona. Upravljanje ovim veličinama se ostvaruje podešavanjem modula i argumenta referentnog vektora napona za prostorni vektorski modulator (slika 1).



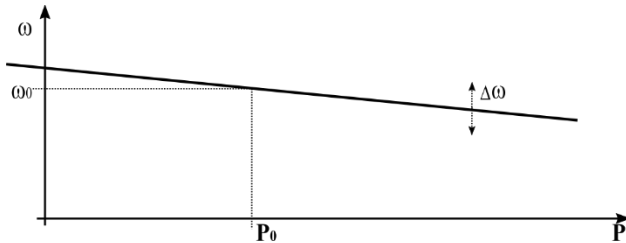
Slika 1. Upravljački algoritam

Napon i struja se mjere na mjestu zajedničkog povezivanja modula sa mikromrežom.

Aktivna snaga određuje referentnu vrijednost učestanosti izlaznog napona na osnovu $P-\omega$ karakteristike. Ovom karakteristikom su opisane nazivna snaga distributivnog generatora i brzina njegovog reagovanja na promjene u sistemu. Pored toga, podešavanjem nagiba karakteristike moguće je ostvariti raspodjelu proizvedene snage između više generatora pri svakoj promjeni u mikromreži tako da generatori veće nazivne snage reaguju sporije od generatora manje nazivne snage na promjene u mreži. Ovo je u saglasnosti sa ponašanjem sinhronog generatora, gdje je brzina reagovanja opisana inercijom rotora, koja je veća kod generatora veće nazivne snage [2]. Veća inercija rotora uslovljava sporiju reakciju na promjene. Generatori manje nazivne snage se koriste za brzu reakciju na promjene u frekvenciji. $P-\omega$ droop karakteristika može biti opisana:

$$\omega_{droop} = \omega_0 + m*(P_0 - P) + \Delta\omega \quad (1)$$

Gdje je P_0 nazivna aktivna snaga, ω_0 nazivna učestanost u sistemu, m parametar kojim se opisuje brzina reagovanja distributivnog generator na promjene u toku aktivne snage, P aktivna snaga u tački povezivanja generator na mikromrežu, a $\Delta\omega$ veličina kojom se sekundarnom regulacijom učestanost održava na nazivnoj vrijednosti. Jednačina (1) je prikazana na slici 2.



Slika 2. $P-\omega$ droop karakteristika

Kako bi se ublažile oscilacije u prelaznom režimu koje prate uključenje i isključenje distributivnog generatora, u ovom radu je predložena dodatna komponenta u upravljačkom algoritmu, *active damping* [3]. Time se smanjuju neželjene oscilacije tokom trajanja prelaznog procesa. *Active damping control* oponaša princip rada *linearog prigušivača koji oblikuje prelazni proces povećavajući prigušenje*.

Active damping komponenta može biti opisana relacijom:

$$\omega_{dmp} = (\omega_{droop} - \omega_{PLL}) * K_{dmp} \quad (2)$$

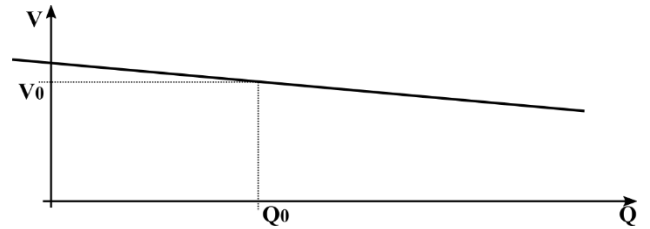
Za korištenje ove komponente u upravljačkom algoritmu potrebna je učestanost dobijena na osnovu $P-\omega$ karakteristike, ω_{droop} , i učestanost u sistemu, proračunata od strane PLL-a, ω_{PLL} .

Proizvedena reaktivna snaga reguliše amplitudu napona na osnovu odgovarajuće $Q-V$ karakteristike.

Poput regulacije učestanosti na osnovu $P-\omega$ karakteristike, ova karakteristika uzima u obzir nazivnu snagu distributivnog generatora i opisuje brzinu reagovanja na promjene u toku reaktivne snage. Princip rada je preuzet iz rada sinhronog generatora koji radi na sopstvenoj mreži, gdje promjena u toku reaktivne snage dovodi do odgovarajuće promjene amplitude napona. Ukoliko potrošnja reaktivne snage u mreži poraste, amplituda napona opada, i obratno. $Q-V$ karakteristika može biti opisana:

$$V = V_0 + n*(Q_0 - Q) + \Delta V \quad (3)$$

Gdje je Q_0 nazivna reaktivna snaga distributivnog generatora, a V_0 nazivni napon u mreži, n parametar kojim se opisuje brzina reagovanja distributivnog generator na promjene u toku reaktivne snage, Q merena reaktivna snaga u tački povezivanja generator na mikromrežu, a ΔV veličina kojom se sekundarnom regulacijom napon održava na nazivnoj vrijednosti. Relacija (3) je opisana na slici 3.

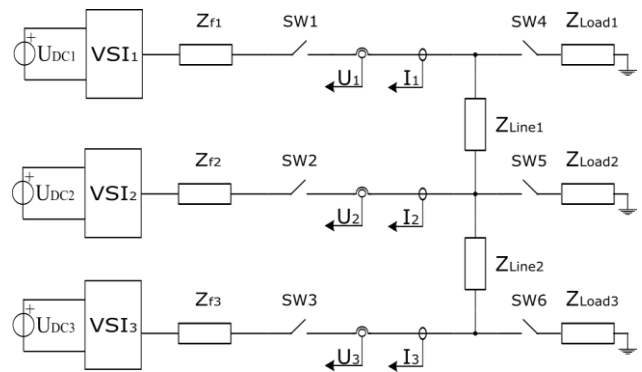


Slika 3. $Q-V$ droop karakteristika

Sekundarna kontrola ima za cilj održavanje učestanosti na nazivnoj vrijednosti, i održavanje amplitude napona u jednoj tački u mreži na nazivnoj vrijednosti. Ovo se postiže izdizanjem ili spuštanjem $P-\omega$ i $Q-V$ karakteristike pomoću veličina $\Delta\omega$ i ΔV opisanih u relacijama (1) i (3).

3. SIMULACIJA I REZULTATI

U cilju potvrde rada upravljačkog algoritma, korištena je mikromreža prikazana na slici 4.



Slika 4. Topologija mikromreže

Model se sastoji od tri distributivna generatora ($VSI+Z_r$), tri potrošača konstante impedance (Z_{LOAD}) i sjedinjenih parametara voda (Z_{Line}). Parametri svih elemenata su dati u prilogu. Svaki element se na mikromrežu povezuje pomoću kontaktora (SW). Mjerenjem napona i struja na mjestu zajedničkog povezivanja, svaki generator podešava amplitudu i frekvenciju kako bi ispunio zadatak

proizvodnje snage za namirenje potrošnje i raspodjele proizvodnje sa ostalim generatorima povezanim na mikromrežu.

Rezultati za svako stacionarno stanje su dati u tabeli 1.

Proračunata aktivna snaga svakog generatora (slika 5.) pokazuje da priključenjem novog generatora dolazi do smanjenja proizvodnje aktivne snage prethodno povezanih generatora, dok priključenje potrošača dovodi do povećanja proizvodnje svih generatora.

Prateći iste sekvence povezivanja elementa na mikromrežu uočava se promjena u toku reaktivne snage sa priključenjem novog generatora (slika 6.). Kao što je slučaj sa tokom aktivne snage, postoji raspodjela snage između generatora koji se priključuje na mrežu i prethodno povezanih generatora i to tako da generator koji se priključuje preuzima dio snage, umanjujući proizvedenu snagu drugih generatora, dok priključenje novog potrošača povećava proizvodnju svih generatora srazmjerno.

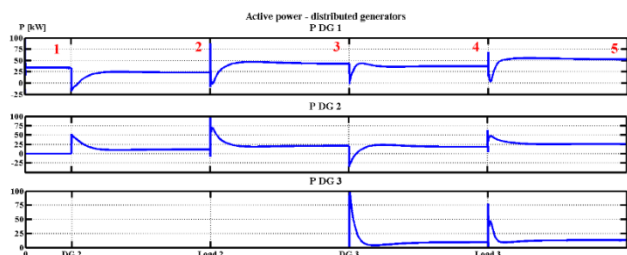
Promjene u frekvenciji napona (slika 7.) prate promjene u toku aktivne snage. Na osnovu $P-\omega$ droop karakteristike opisane u relaciji (1) povećanje u toku aktivne snage dovodi do smanjenja frekvencije i obratno.

Tok reaktivne snage uslovljava promjene u amplitudi napona svakog generatora (slika 8.) na osnovu $Q-V$ karakteristike opisane relacijom (2). Prateći datu karakteristiku, upravljački algoritam povećava amplitudu napona kako se smanjuje tok reaktivne snage i obratno.

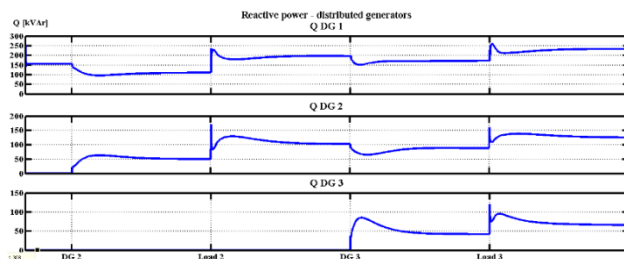
Tabela 1. Rezultati simulacije u Simulink okruženju

		Period				
		1	2	3	4	5
P [MW]	DG 1	0.339	0.239	0.431	0.375	0.527
	DG 2	/	0.116	0.214	0.186	0.26
	DG 3	/	/	/	0.09	0.13
Q [MVar]	DG 1	0.15	0.11	0.197	0.17	0.234
	DG 2	/	0.05	0.103	0.088	0.126
	DG 3	/	/	/	0.042	0.067
w [rad/s]		314.16	314.16	314.16	314.16	314.16
V [kV]	DG 1	2.76	2.78	2.72	2.72	2.69
	DG 2	/	2.77	2.68	2.69	2.61
	DG 3	/	/	/	2.68	2.57

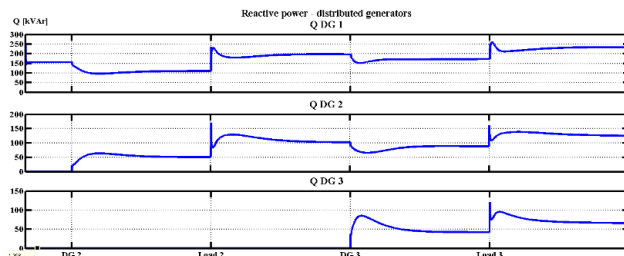
Vrijeme trajanja simulacije podijeljeno je na periode kako bi se uočile vrijednosti svih veličina bitnih za verifikaciju algoritma u svakom ustaljenom period. Simulacija počinje povezivanjem prvog generatora na mikromrežu, zatim se povezuje prvi potrošač - period 1, a potom se naizmjenično povezuju generatori i potrošači.



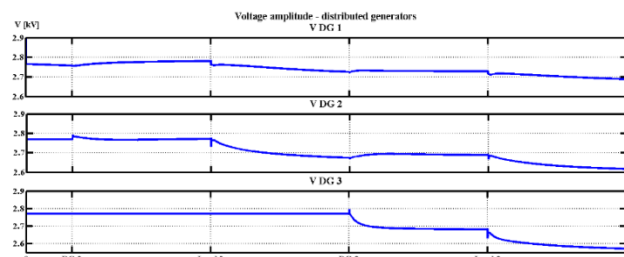
Slika 5. Tok aktivne snage – rezultati simulacije



Slika 6. Tok reaktivne snage – rezultati simulacije



Slika 7. Učestanost napona – rezultati simulacije



Slika 8. Amplituda napona generatora – rezultati simulacije

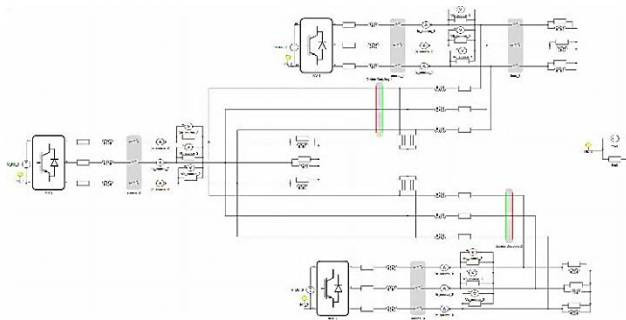
Na osnovu rezultata prikazanih u tabeli 1. i slikama od 5. do 8. uočava se da raspodjela proizvedene aktivne i reaktivne snage u dovoljnoj mjeri prati odnos nazivnih snaga generatora.

4. TESTIRANJE U TYPHOON HIL OKRUŽENJU

Kako Simulink okruženje ne daje mogućnost testiranja u realnom vremenu, a samim tim ostavlja mogućnost da se neke pojave ne uoče ili zanemare, razvijeni upravljački algoritam je testiran i u Typhoon HIL okruženju.

Emulacijom modela mikromreže razvijenog u HIL okruženju i upotrebom mikrokontrolera TI F28335 dobijeni su rezultati koji ukazuju da predloženi upravljački algoritam ostvaruje zadovoljavajuće rezultate i u realnom vremenu. Model mikromreže, prikazan na slici 9., koji je korišten za testiranje je ostvaren po uzoru na model korišten prilikom simulacije u Matlab-Simulink okruženju uz uvođenje novih elemenata neophodnih za podjelu modela na nekoliko HIL uređaja tokom emulacije.

Parametri korišteni u emulaciji su navedeni u prilogu. Za tu svrhu korišteni su elementi *Device Coupling*, koji razdvaja kolo na dva dijela i dodatni otpor vezan u paralelno kako bi se kompenzovao uticaj diskretizacije na procesu emulacije.



Slika 9. Model u Typhoon HIL okruženju

Tabela 2. Rezultati simulacije u Typhoon HIL okruženju

		Period			
		1	2	3	4
P [MW]	DG 1	0.52	0.38	0.31	0.375
	DG 2	/	0.18	0.16	0.186
	DG 3	/	/	0.11	0.09
Q [MVar]	DG 1	0.36	0.26	0.22	0.17
	DG 2	/	0.14	0.11	0.088
	DG 3	/	/	0.07	0.042
w [rad/s]		314.16	314.16	314.16	314.16
V [kV]	DG 1	2.76	2.85	2.9	2.72
	DG 2	/	2.81	2.87	2.69
	DG 3	/	/	2.84	2.68

Stacionarna stanja data u tabeli 2. nakon manipulacije kontaktorima su označeni od 1-4. Prvo stacionarno stanje je nakon povezivanja jednog generatora i tri potrošača. Stacionarno stanje broj 2 nastaje nakon povezivanja drugog generatora, stacionarno stanje broj 3 nakon povezivanja trećeg generatora i stacionarno stanje broj 4 opisuje otvaranja kontaktora jednog od potrošača impedanse jednake impedansama druga dva potrošača.

5. ZAKLJUČAK

Upotreba i razvoj mikromreža se smatra za jedno od rešenja koje najviše obećava u pogledu integracije većeg broja distributivnih izvora energije manje snage u mrežu. Upravljački algoritam izložen i testiran u ovom radu predlaže način rješavanja ovog problema na decentralizovan način, gdje svaki modul koji se povezuje na mrežu ima zaseban primarni nivo kontrole.

Predloženi način kontrole omogućava povezivanje i rad distributivnih generatora na mikromreži.

Kako bi se bolje razumio i poboljšao rad mikromreže potrebna je detaljnija dinamička analiza. Ovo će biti jedan od ciljeva budućeg rada.

6. LITERATURA

- [1] D. E. Olivares et al., "Trends in Microgrid Control", IEEE Trans. Smart Grid, vol. 5, no. 4, pp. 1905-1919, Jul. 2014.
- [2] M. C. Chandorkar, D. M. Divan, R. Adapa, "Control of parallel connected inverters in standalone AC supply systems", IEEE Trans. Ind. Appl., vol. 29, no. 1, pp. 136-143, Jan. 1993.
- [3] S. Grabic, N. Celanovic, V. Katic, "Series converter stabilized wind turbine with permanent magnet synchronous generator", 2004 IEEE 35th Annual Power Electronics Specialists Conference (IEEE Cat. No. 04CH37551), 2004, vol. 1, p. 464-468 Vol. 1.

Kratka biografija:

Njegoš Janković rođen je u Doboju 1993. god. 2012 god upisao osnovne akademske studije na Fakultetu Tehničkih Nauka, smer Energetika, elektronika i telekomunikacije. 2016 upisao master akademske studije na Fakultetu Tehničkih Nauka.

PRILOG

Tabela 3. Parametri elemenata korištenih u simulaciji

Label	U_{DC}	Parametri	
		$P-\omega$ characteristic $\omega = \omega_0 + m \cdot (P_0 - P)$	$Q-V$ characteristic $V = V_0 + n \cdot (Q_0 - Q)$
DG 1	5 kV	$\omega_0 = 314.16$ [rad/s]	$V_0 = 2.9$ kV
		$P_0 = 0.42$ MW	$Q_0 = 0.42$ MVar
		$m = 1.49 \cdot 10^{-5}$ rad/s*MW	$n = 1.4 \cdot 10^{-4}$ V/VAr
DG2	5 kV	$\omega_0 = 314.16$ [rad/s]	$V_0 = 2.9$ kV
		$P_0 = 0.21$ MW	$Q_0 = 0.21$ MVar
		$m = 2.98 \cdot 10^{-5}$ rad/s*MW	$n = 2.8 \cdot 10^{-4}$ V/VAr
DG3	5 kV	$\omega_0 = 314.16$ [rad/s]	$V_0 = 2.9$ kV
		$P_0 = 0.08$ MW	$Q_0 = 0.08$ MVar
		$m = 4.47 \cdot 10^{-5}$ rad/s*MW	$n = 4.2 \cdot 10^{-4}$ V/VAr
	$Z = R + j\omega L$	R [Ω]	ωL [Ω]
Filter VSI 1	Z_{f1}	0.05	0.94
Filter VSI 2	Z_{f2}	0.05	0.94
Filter VSI 3	Z_{f3}	0.05	0.94
Line 1	$Z_{Line 1}$	0.4	0.94
Line 2	$Z_{Line 2}$	0.4	0.94
Load 1	$Z_{Load 1}$	63	62.8
Load 2	$Z_{Load 2}$	63	62.8
Load 3	$Z_{Load 3}$	63	62.8

VERIFIKACIJA POTENCIJALA VETRA U VOJVODINI**WIND POTENTIAL VERIFICATION IN VOJVODINA**Vladan Stevanović, Vladimir Katić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu rađena je verifikacija parametara vetra dobijenih iz studije „Atlas Vetrova AP Vojvodine“ pomoću „Global Wind Atlas“ softvera. Verifikacija je obuhvatila osam različitih lokacija na teritoriji Vojvodine, na kojima se nalaze hidro-meteorološke stanice.

Abstract – In this paper, the verification of wind parameters obtained from the "Atlas of the Winds of AP Vojvodina" study was carried out using the "Global Wind Atlas" software. The verification included eight different locations on the territory of Vojvodina, which include hydro-meteorological stations.

Ključne reči: Vetar u Vojvodini, snaga vetra, gustina snage vetra.

1. UVOD

Energija vetra spada u obnovljive izvore energije i danas predstavlja jednu od najviše korišćenjih. Vetar je pojava kretanja vazduha, kao posledica razlike atmosferskih pritisaka između dva područja. Glavni uticaj na ovu pojavu ima Sunce, koje u različitoj meri zagreva oblasti zemlje između polova i ekvatora. Kada postoji razlika u pritisku, vazduh se keće sa područja višeg ka području nižeg pritiska, pa se kao posledica javljaju vetrovi različitih brzina. Glavni parametri energije vetra su njegova brzina, pravac i smer, te gustina vazduha, temperatura i pritisak [1].

U cilju korišćenja energije vetra za generisanje električne energije, potrebno je njegovu linearnu kinetičku energiju pretvoriti u rotacionu pomoću vetroturbine. Rotaciono kretanje se prenosi na osovinu električnog generatora na čijim priključcima se generiše električna struja. Snaga vetroturbine definisana je poznatim izrazom [1]:

$$P_{meh} = \frac{1}{2} \rho C_p A v^3 [W] \quad (1)$$

je ρ gustina vazduha (standardna gustina pri temperaturi 15°C iznosi $\rho = 1,225 \text{ kg/m}^3$), C_p koeficijent iskorišćenja vetroturbine (obično 0,4 – 0,45; max 0,59 Betzova konstanta), A površina obuhvata krilaca trokrake vetroturbine ($r^2\pi$), a v brzina vetra u m/s. Ključna promenljiva u ovom izrazu je brzina vetra. Da bi se dobila jednostavnija predstava snage za analiziranje primene raznih tipova turbina definiše se gustina snage kao odnos snage (P) i površine obuhvata trokrake turbine (A):

$$G = P/A [W/m^2] \quad (2)$$

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Vladimir Katić.

Ova dva parametra, brzina vetra i gustina snage, su osnovni za određivanje energetskog potencijala vetra na nekoj lokaciji ili regionu. Cilj ovog rada je da uporedi dva izvora u kojima je predstavljen vetro-potencijal u Vojvodini; „Atlas vetrova AP Vojvodine“ [2] i rezultate koje se dobijaju softverskim alatom „Global Wind Atlas“ [3]. Oba izvora su zasnovana su na korišćenju poznatog softvera WasP [4], a rezultati zavise od visine posmatranja (h), hrapavosti tla (z) i drugih parametara, koji će biti uzeti u obzir prilikom poređenja.

2. ENERGIJA VETRA**2.1 Pravac vetra**

Pravac vetra predstavlja stranu sveta sa koje vetar duva (Istok, Zapad, Sever, Jug i njihove kombinacije). Pravac se definiše na bazi ugla od 0° do 360° (u pravcu kazaljke na satu), gde je za sever ugao 0° i prikazuje se u ruži vetrova. Podela može biti na od 12 do 16 odsečaka kruga, odnosno po 30° ili po 22,5°. Ugao od 0°, odnosno pravac sever je na sredini prvog odsečka.

2.2 Brzina i snaga vetra

Brzina vetra predstavlja put koji čestice vazduha pređu u jedinici vremena (m/s).

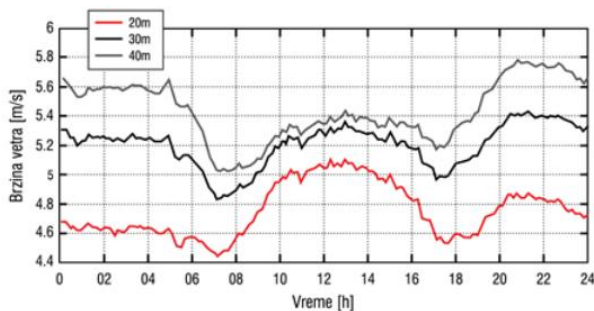
Jačina (snaga) vetra se definiše kao efekat njegovog dejstva na određene predmete i date su u Boforovoj skali (od 0 – 12). Ona je bazirana na subjektivnoj predstavi brzine, odnosno dejstva vetra.

Ako se energija vetra pretvara u rotacionu obrtnu energiju turbine, snaga vetra se definiše izrazom (1). Kako je snaga proporcionalna trećem stepenu brzine vetra, i male promene brzine vetra odražavaju na promenu snage. Zbog toga, da bi se u punoj meri sagledao potencijal energije vetra treba meriti njegovu brzinu u što dužem periodu, a minimalni period je godinu dana (kako bi se obuhvatile sezonske varijacije).

Brzina vetra je veoma promenljiva veličina, i to je jedan od glavnih nedostataka pri korišćenju energije vetra. Zbog toga se brzine obično mere svakih 10 minuta (analogni zapisivači), odnosno u intervalima 1-5 min. digitalnim loggerima. Promene brzine vetra zavise od godišnjeg doba, ali takođe su značajne i u toku dana. Pored toga, brzina vetra se značajno menja po visini od tla, tako da se merenje treba vršiti na različitim visinama. Uobičajeno je da se u stanicama Hidrometeorološkog zavoda Srbije (HMZS) merenja vrše na 12m, dok se za određivanje potencijala vetra merenja vrše i na većim visinama (i do 200 m sa specijalnim mernim stubovima). Brzina vetra na pojedinim visinama može se i računski odrediti, ako se poznaje vrednost na jednoj visini [1]:

$$v = v_1 \cdot \left(\frac{h}{h_1} \right)^m \quad (3)$$

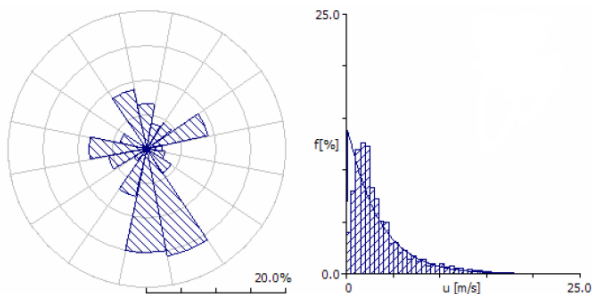
gde je v_1 poznata vrednost brzine na visini h_1 (na primer 12 m), h visina za koju se računa brzina vetra v , a m koeficijent koji zavisi od grubosti (hrapavosti) terena i ide od 0,01 za mirnu morsku površinu do 0,29 za seoske (izgrađene) zone i pošumljene terene. Brzina još zavisi i od rapavosti terena, temperature i pritiska vazduha i dr. Na slici 1 predstavljen je dnevni dijagram promene brzine vetra meren na tri visine na lokaciji u južnom Banatu [6]:



Slika 1. Dnevna promena brzine vetra za južni Banat [6]

2.2. Ruža vetrova

Ružom vetrova opisuju se srednje brzine i učestalost vetra po pravcima (u %), dok histogram raspodele vetra prebrojava koliko ima izmerenih podataka o vetru u nekom zadatom intervalu brzina (slika 2). Histogrami empirijske raspodele često se prave i po pravcima i za ceo uzorak, a izražava se i matematičkim funkcijama (načešće Weibull-ova raspodela).



Slika 2. Ruža vetrova i histogram empirijske raspodele izmerenog vetra [2]

Pomoću ovih podataka može se lako odrediti iz kojih smerova najčešće duvaju vetrovi i sa kojom najvećom prosečnom brzinom. Uglavnom se najčešći smerovi duvanja vetrova poklapaju sa smerovima vetrova sa najvećom prosečnom brzinom.

3. MERENJE BRZINE VETRA

Područja na kojima se izvršavaju meteorološka merenja i osmatranja nazivju se meteorološkim stanicama. Pozicija meteorološke stanice mora biti postavljena što dalje od većih objekata koji svojim prisustvom mogu uticati na meteorološke parametre, stoga je neophodno stanicu postaviti na nekoj reprezentativnoj lokaciji koja će što realnije prikazivati vreme i klimu neke oblasti [7]. Krug oko meteorološke stanice (slika 3) bi trebalo da tokom celog dana bude obasjan Suncem. U njemu se postavlja

meteorološka kućica unutar koje su smešteni određeni merni instrumenti. Dok se druge neophodne stvari za merenje i osmatranje nalaze obično pored kućice. Neki od instrumenata mogu biti smešteni i u unutrašnjosti zgrade meteorološke stanice kako bi zaposleni kadar mogao očitavati parametre na licu mesta.



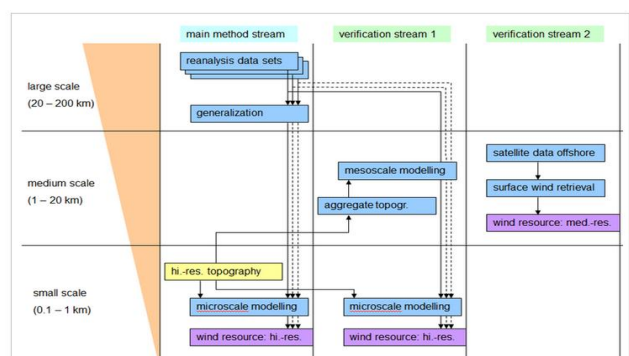
Slika 3. Meteorološka stanica u Zrenjaninu [9].

4. METODOLOGIJA ATLASA VETROVA

Za dobijanje studije “Atlas vetrova AP Vojvodine”[2] čije ćemo rezultate u daljem radu uporediti sa podacima vetra iz “Global Wind Atlas-a”[3], korišćeni su podaci RHMZ-a Srbije prikupljeni sa 8 glavnih meteoroloških stanica (GMS) Vojvodine smeštenih u Novom Sadu (Rimski Šančevi), Paliću, Somboru, Zrenjaninu, Sremskoj Mitrovici, Kikindi, Banatskom Karlovcu i Vršcu, koji su dalje obrađeni u WasP softveru.

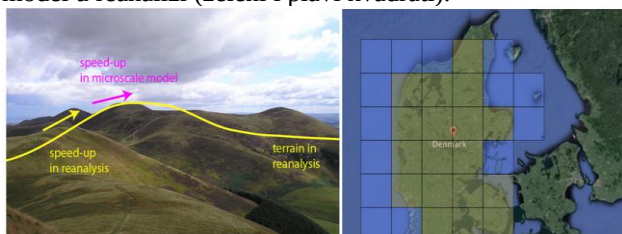
5. “GLOBAL WIND ATLAS”

„Global Wind Atlas“ [3] zasnovan je na procesu smanjenja (downscaling), slika 4. Započinje se sa podacima o klimi vetra u velikoj razmeri, a završava se sa podacima o vetru na nekom mikromestu. Podaci o klimi vetra u velikoj razmeri, obezbeđeni su uz pomoć reanaliza atmosfere iz meteoroloških stanica širom sveta. Ovi podaci se nalaze na mreži sa razmakom od oko 50 km u zavisnosti od skupa podataka. Na ovim podacima vrši se proces generalizacije. Rezultat je skup opštih klimatskih uslova koji imaju isti razmak na mreži kao i podaci o reanalizi koji su korišćeni za njihovo stvaranje.



Slika 4. Metodologija „Global Wind Atlas-a“ i metodologija verifikacije [3].

Metoda “Global Wind Atlasa” zasnovana je na generalizaciji klimatologije vetra dobijenih od globalnog modelovanja. Ovaj post-procesni metod generalizacije obimno se koristi u brojnim studijama o proceni vetra, posebno u okviru “WasP” metode. Početno mesto za primenu postupka generalizacije na podatke o re-analizi su “TAB netcdf” fajlovi sadržani u Databazama. Ove datoteke sadrže statističke podatke potrebne za dalje smanjenje. Na slici 5. desno, se ilustruju razlike u lokaciji obale u stvarnosti (mapa Google Earth) u odnosu na model u reanalizi (zeleni i plavi kvadrati).



Slika 5. Ilustracija razlike terena u stvarnosti i u reanalizi “Global Wind Atlasa” [3]

6. REZULTATI VERIFIKACIJE VETRA

U ovom radu izvršena je verifikacija “Atlasa vetrova AP Vojvodine” [2], tj. prikazan je energetski potencijal i brzine vetrova za osam mesta u Vojvodini koji će biti upoređivani sa vetrovima dobijenim iz “Global Wind Atlasa” [3] za ista mesta u Vojvodini. Predpostavlja se da razlika potencijala vetrova iz dva predhodno pomenuta izvora neće biti velika, kako bi oba izvora mogli smatrati pouzdanim. Rezultati iz oba izvora su međusobno upoređeni. Utvrđeno je njihovo relativno odstupanje (u %). tj. između “Atlasa Vetrova AP Vojvodine” i “Global Wind Atlasa”, s tim da su rezultati iz Global Wing Atlasa uzeti kao referentni. Za osam lokacija na kojima su vršena merenja i obrađivani podaci rezultati poređenja su predstavljeni u tabelama 1 do 8. Slike 6. i 7. predstavljaju srednje godišnje brzine vetra iz dva pomenuta izvora.

Napomena: Odstupanja u procentima za ruže vetrova (pravac ruže i učestanost, odnosno poslednje dve cifre u tabelama) računane su samo za dominantni pravac duvanja vetra, a ne za celu ružu.

Tabela 1. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Vršca

	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	3,1	6,2	6,0	6,4	25,0	6,0

Tabela 2. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Palića

	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	24,8	27,6	28,0	27,0	19,0	18,8

Tabela 3. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Sremske Mitrovice

	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	22,6	25,5	25,3	25,1	14,2	7,2

Tabela 4. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Banatskog Karlovca

	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	2,0	5,9	6,1	5,9	8,3	12,5

Tabela 5. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Zrenjanina

	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	21,8	24,9	24,9	24,4	2,5	17,6

Tabela 6. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Kikinde

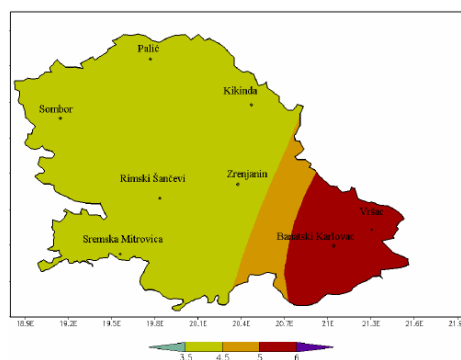
	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	26,4	28,9	28,7	28,6	4,2	25,0

Tabela 7. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Sombora

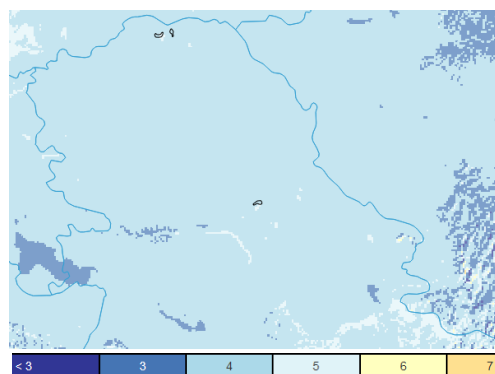
	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	26,9	29,9	29,9	29,5	5,6	8,3

Tabela 8. Odstupanje srednje brzine i ruže vetra u %, između dva pomenuta izvora za područje Rimskih Šančeva

	z = 0.00m	z = 0.03m	z = 0.10m	z = 0.40m	Pravac ruže	Učestanost
h = 50m	30,6	33,0	32,7	32,0	19,0	12,5



Slika 6. Srednja godišnja brzina vetra na 50m visine, dobije iz studije “Atlasa Vetrova AP Vojvodine” [2]



Slika 7. Srednja godišnja brzina vetra na 50m visine, dobijena prema „Global Wind Atlas“ softveru [3]

7. DISKUSIJA

Prvo što treba napomenuti jeste da se ruže vetrova ne menjaju značajno u zavisnosti od podloge, odnosno od hrapavosti terena i visine. Ruže vetrova pri različitim vrednostima hrapavosti podloge su gotovo identične. Što je donekle i logično jer se pravci iz kojih vetrovi duvaju ne bi trebali menjati zbog podloge terena. Što se tiče dva različita izvora podataka vidimo da se ruže vetrova minimalno razlikuju, odnosno u manjoj ili nešto većoj meri u zavisnosti od područja.

Ukoliko se pažljivo uporede podaci u tabelama u radu, odgovarajuće parametre sa odgovarajućim parametrima pri istoj visini i hrapavosti terena, može se primetiti da su brzine vetrova dobijene iz "Global Wind Atlasa" [3] nešto veće u odnosu na brzine vetrova pri istoj visini i hrapavosti dobijene iz "Atlasa Vetrova AP Vojvodine" [2]. To za posledicu ima da su godišnje srednje snage vetra dobijene iz "Global Wind Atlasa" veće, a to proizilazi iz činjenice da gustina snage vetra zavisi od kuba brzine, stoga i male promene u brzini vetra prouzrokuju velike promene u gustini snage vetra. Jedina komparacija koja se u nekoj meri razlikuje od drugih je dobijena za područje Vršca, gde se prema "Global Wind Atlasu" uprkos za nijansu jačim vetrovima, gustina srednje snage vetra je ipak nešto manja.

Najmanje razlike rezultata dobijena iz dva različita izvora su za područja gde su vetrovi najjači, odnosno za područja Vršca i Banatskog Karlovca (Tabele 1 i 4), dok su razlike u parametrima između dva pomenuta izvora najveće, za oblasti gde su vetrovi najslabiji (Tabela 8). Na tim mestima "Global Wind Atlas" prikazuje nešto veće srednje godišnje brzine vetrova, a samim tim i nešto veće srednje gustine snage.

8. ZAKLJUČAK

Ovim radom obuhvaćena je verifikacija vetrova na teritoriji Vojvodine, tj. urađena je komparacija karakteristika vetrova iz dva izvora: „Atlas Vetrova AP Vojvodine“ [2] i „Global Wind Atlasa“ [3]. Pažljivom analizom rada došlo se do zaključka da podaci vetrova dobijeni iz „Global Wind Atlasa“ generalno daju nešto veće vrednosti za srednje godišnje brzine vetrova u odnosu na prvi pomenuti izvor. Razlike između dva izvora u zavisnosti od lokacije se u određenoj meri razlikuju. Od minimalnih razlika za područje Vršca do nešto većih za Kikindu.

Može se zaključiti da za osam obrađenih lokacija na kojima je izvršena verifikacija, tj. svih lokacija na kojima su u Vojvodini postavljene hidrometeorološke stanice, "Atlas Vetrova AP Vojvodine" daje rezultate, koji više variraju od mesta do mesta, za razliku od "Global Wind Atlasa", koji ima manje oscilacija i razlika između obrađivanih mesta. Dolazi se do subjektivnog generalnog utiska da su parametri vetrova dobijeni iz "Global Wind Atlasa" u nekoj ili određenoj meri više generalizovani za obrađivane lokacije.

Svakako podaci iz dva različita izvora ne mogu biti potpuno isti. Međutim zbog velikog broja podataka koje su hidrometeorološke stanice prikupile tokom više

godina, i koje su sve obuhvaćene u "Atlasu Vetrova AP Vojvodine", kao i zbog opšteg utiska da "Global Wind Atlas" u nekoj meri daje slične podatke za lokacije koje se nalaze u međusobnoj blizine, celokupni utisak je da se prednost može dati "Atlasu Vetrova AP Vojvodine". Ipak, "Global Wind Atlas" predstavlja dobar, jednostavan i u velikom procentu tačan softver za verifikaciju vetrova, koji nudi veliki broj slikovito i tabelarno prikazanih parametara vetra.

9. LITERATURA

- [1] V.Katić i dr., „Obnovljivi izvori električne energije“, TEMPUS-CEFES, Novi Sad, 2007.
- [2] V. Katić i dr., „Atlas vetrova AP Vojvodine“, Studija za Pokraj. sekretar. za energetiku i min. sirovine, Novi Sad, 2008.
- [3] <http://globalwindatlas.com/map.html>
- [4] <http://www.wasp.dk/dataandtools#climate-analyst>
- [5] Vladan Stevanovic „Idejno rešenje samostalnog hibridnog vetro-fotonaponskog sistema za napajanje salaša u okolini Vršca“, Diplomski rad, Fakultet tehničkih nauka, Novi Sad, 2016.
- [6] Milan Srndović, mr Željko Đurišić, prof. dr Zlatan Stojković, doc. dr Jovan Mikulović, Boban Panajotović: „Hibridni vetar-solar-dizel sistemi za napajanje opreme mobilne telefonije na izolovanim lokacijama“, Telekomunikacije, RATEL, Br.9, 2012, pp.54-64
- [7] <https://meteoplaneta.rs/meteoroloski-instrumenti-i-osmatranja/meteoroloska-stanica-i-instrumenti/>
- [8] <http://globalwindatlas.com/methods.html>
- [9] <http://ilovezrenjanin.com/aktuelno/meteoroloska-stanica-zrenjanin/>

Kratka biografija:



Vladan Stevanović rođen je u Zrenjaninu 1991. god. Zrenjaninsku Gimnaziju, završio je 2010. god. Fakultet tehničkih nauka, studijski program Energetika, elektronika i telekomunikacije upisao je školske 2010/2011. godine. Na studijama se opredelio za smer Elektroenergetika-energetika elektronika i električne mašine, i diplomirao 2016. god. Master studije na istom fakultetu upisuje iste godine, koje je završio 2017. godine.



dr Vladimir A. Katić, red.prof. rođen je 1954. god. u Novom Sadu. Diplomirao je na Fakultetu tehničkih nauka u Novom Sadu 1978. god., a magistrirao i doktorirao na Univerzitetu u Beogradu 1981. i 1991. god., respektivno. Od 2002. godine je redovni profesor Univerziteta u Novom Sadu. Trenutno je prodekan Fakulteta tehničkih nauka i šef Katedre za energetsku elektroniku i pretvarače. Oblasti interesovanja su energetska elektronika, obnovljivi izvori električne energije, kvalitet električne energije i električna vozila.

MODELOVANJE TOPLOTNE PUMPE ZA RAD U REŽIMU GRIJANJA MODELING OF A HEAT PUMP FOR HEATING OPERATING MODE

Dajana Mitrović, *Fakultet tehničkih nauka, Novi Sad*

Oblast — AUTOMATIKA I UPRAVLJANJE SISTEMIMA

Kratak sadržaj – Ovaj rad predstavlja jedno od mogućih softverskih rješenja za modelovanje toplotne pumpe u režimu grijanja, na osnovu matematičke formule. Rješenje je moguće primijeniti upotrebom različitih tipova toplotnih pumpi, u zavisnosti od toplotnog izvora i toplotnog ponora.

Abstract – This paper presents one of the possible software solutions for modeling heat pump system for heating operating mode based on mathematical formula. This solution is applicable to various heat pump systems, based on heat source and heat

Ključne riječi: Toplotne pumpe, kompresor, isparivač, izmjenjivač, COP

1. UVOD

Povećanje energetske efikasnosti je najbrži i najjeftiniji način da se zadovolji sve veća potražnja za energijom, kao i da se ispune ekološki i ekonomski izazovi koji se javljaju sa njenom potrošnjom.

Sistem za grijanje i hlađenje jedan je od najvećih potrošača električne energije u stambeno-poslovnim objektima, i njegova potrošnja čini 34% od ukupne potrošnje električne energije [1].

Sistem grijanja i hlađenja ima zadatak da u prostoriji održava željenu temperaturu vazduha. Za ove namjene u osnovi se najčešće koriste toplotne pumpe. Današnji klima-uređaji koji se koriste u domaćinstvima su kompaktne toplotne pumpe sa dodatnim mogućnostima koje služe za klimatizaciju prostora.

2. ZADATAK I CILJ RADA

Zadatak rada jeste da se izvrši modelovanje sistema toplotnih pumpi za režim grijanja, upotrebom dva softverska paketa: Matlab i Labview, koji se koriste za simulaciju, kao i prikaz dobijenih rezultata.

Cilj rada jeste da se modeluje toplotna pumpa sa što većim koeficijentom efikasnosti COP, tačnije pumpa koja će, uz što manji utrošak električne energije, na izlazu dati željenu temperaturu.

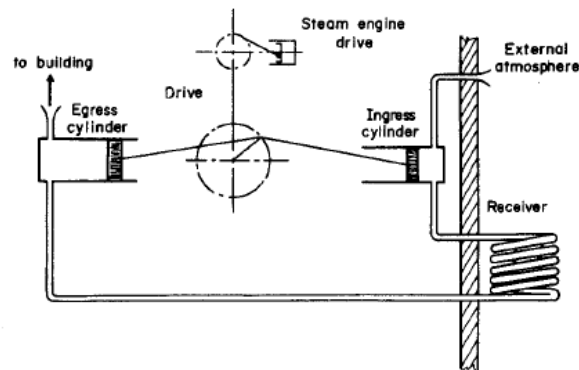
3. SISTEM TOPLOTNIH PUMPI

3.1. Uvod

Teoretske osnove rada toplotnih pumpi bile su poznate još u XIX vijeku. Međutim, iako je prva toplotna pumpa konstruisana već 1920. godine, tek početkom 70-ih godina XX vijeka započinje njihov intenzivan razvoj.

Ovo je svakako bilo prouzrokovano pojavom energetske krize, kada energetske efikasne sistemi postaju aktuelni i intenzivira se rad na poboljšanju njihovih energetskih karakteristika.

Prva toplotna pumpa, predstavljena na slici 1, predstavljala je otvoreni sistem, i koristila okolni vazduh kao toplotni izvor, i kao radni fluid. Ovi sistemi su se razvili u zatvorene sisteme sa kružnim ciklusima koji koriste različite izvore toplote [2].



Slika 1. Toplotna pumpa sa otvorenim sistemom

3.2. Kružni proces

Da bi se u nekom toplotnom aparatu iz toplotne energije dobio rad ili se uz pomoć rada obavio prenos toplote od jednog predmeta ili prostora drugom, neophodno je da radni medijum koji se nalazi u toplotnom aparatu prolazi kroz tzv. kružni proces. Kružnim procesom se smatra onaj kod koga se radni medijum nakon nekoliko uzastopnih promjena stanja vraća u početno.

U zavisnosti od načina prikazivanja promjena radne materije na dijagramima razlikujemo desnokretne i levokretne kružne procese. Bitno je naglasiti da se razlikuju levokretni kružni ciklus kome je osnovi zadatak ostvariti grijanje, i onaj kome je primarni zadatak ostvariti hlađenje.

Ako su promjene stanja radne materije ravnotežne i istovremeno i povratne tj. bez mehaničkih gubitaka, tada se postiže najveći mogući termički stepen iskorišćenja za posmatrani ciklus. Takav kružni ciklus je moguće samo teorijski zamisliti i naziva se idealni Karnoov kružni ciklus i on nam služi kao referentan za ocjenu efikasnosti svih ostalih kružnih ciklusa [2].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Velimir Čongradac, van.prof.

3.3. Izvori energije

Podzemne vode, vazduh i geotermalna energija mogu se koristiti kao izvori toplote za toplotne pumpe. Isto tako otpadna toplota i toplota koja nastaje u proizvodnim i prerađivačkim procesima može se koristiti kao izvor energije. U odnosu na izvor toplote razlikujemo tri tipa toplotnih pumpi:

- Zemlja - voda toplotne pumpe
- Voda - voda toplotne pumpe
- Vazduh - voda toplotne pumpe

3.4. Princip rada

Toplotna pumpa predstavlja toplotnu mašinu koja je zasnovana na Karnoovom kružnom ciklusu, i uz „stabilan“ toplotni izvor konstantne temperature može ostvariti visok toplotni učinak. Rashladni učinak ovog uređaja je sekundaran i nije neophodno da se koristi, ali je veoma atraktivno njegovo korišćenje za istovremeno hlađenje nekih obližnjih objekata. Osnovna karakteristika toplotne pumpe je ta da se za 1 kW utrošene električne energije dobije do 6 kW toplotne energije. To praktično znači da je grijanje ili hlađenje objekta uz pomoć toplotne pumpe jeftinije za 75% u odnosu na bilo koji drugi energent (gas, uglj, drva, struja). [2]

Zato što troše manje primarne energije u odnosu na druge konvencionalne sisteme, one predstavljaju važnu tehnologiju za smanjenje emisije gasova koji negativno utiču na životnu sredinu, kao što su ugljen dioksid CO₂, sumpor dioksid SO₂ i oksidi azota NO_x. Međutim, sveobuhvatni uticaj na zaštitu životne sredine zavisi od načina na koji je električna energija proizvedena.

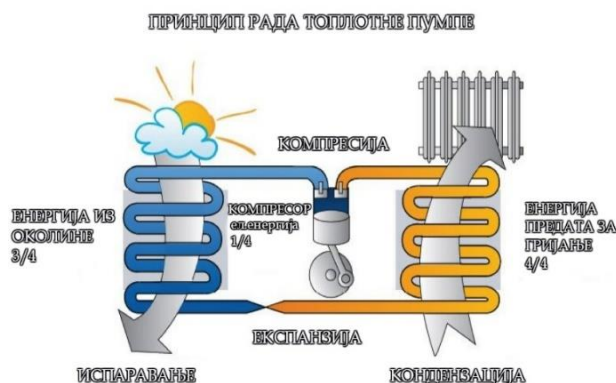
Preuzimanjem energije iz jedne sredine, i njenim transportom do druge sredine, se može dobiti i do 80% energije potrebne nekom objektu. Tek 20% energije treba dodati u obliku električne energije, koja je potrebna za rad same pumpe.

Toplotna pumpa je termodinamički sistem koji oduzetu energiju jednog medijuma predaje drugom medijumu. Energija niske temperature se transformiše u energiju više temperature, koja je pogodna za potrebe grijanja. Ovaj proces se odvija u zatvorenom krugu unutar kojeg radni fluid stalno mijenja stanje. U režimu grijanja, princip funkcionisanja je obrnut principu funkcionisanja frižidera. Dok frižider toplotu prenosi iz unutrašnjosti prema spoljašnjosti, toplotna pumpa energiju preuzetu iz okoline proslijeđuje ka objektu. [3]

Koristi se specijalni rashladni fluid, koji, u zavisnosti od temperature i pritiska pod kojim se nalazi, može biti u tečnom ili gasovitom stanju. Rashladno sredstvo apsorbira energiju preuzetu iz podzemnih voda, zemlje ili vazduha i nakon kompresije je koristi za dalje snabdevanje toplotnom energijom.

Na slici 2 predstavljen je rashladni krug, koji je sastavljen od :

1. Kompresora
2. Kondenzatora
3. Ekspanzionog ventila
4. Isparivača



Slika 2. Princip rada toplotne pumpe

Rashladni fluid se, u gasovitom stanju, komprimuje i cirkuliše kroz sistem uz pomoć kompresora.

Po izlasku iz kompresora, rashladni fluid u obliku vrelog gasa pod visokim pritiskom ulazi u kondenzator, gdje se hladi i kondenzuje u tečnost umjerene temperature pod visokim pritiskom.

Kondenzovani rashladni fluid zatim prolazi kroz uređaj za umanjenje pritiska (ekspanzioni ventil). Potom kondenzovani rashladni fluid ulazi u isparivač.

Toplotne pumpe se tokom ljetnjih mjeseci mogu koristiti i za hlađenje. U slučaju sistema zemlja/voda, toplota iz stambenog prostora se sistemom ventilokonvektora ili podnog grijanja, preuzima iz prostora, a zatim, upotrebom geosondi ili površinskih kolektora odvodi u zemlju. U zavisnosti od potreba performansi hlađenja, kao i postojećeg Sistema hlađenja, moguće je odabrati sistem aktivnog ili pasivnog hlađenja [4,5].

Aktivacijom reverzibilnog načina rada, toplotne pumpe se mogu koristiti za hlađenje objekata. Toplotna energija iz objekta se aktivno, pomoću kompresora, prenosi do zemlje. Kondenzator, u funkciji hlađenja, ima ulogu isparivača.

Koristi se da toplotu iz prostorija preda radnom fluidu, koji tu isparava, i prelazi u kompresor, gdje se komprimuje. Nakon kompresora, radni fluid se vodi u kondenzator, gdje se toplotna energija predaje zemlji, pomoću geosondi ili površinskih kolektora.

Pasivno hlađenje je tehnički vrlo jednostavan proces. Ako je u ljetnjim mjesecima sobna temperatura veća od temperature zemlje, toplotna pumpa može upravljati prirodnim, odnosno pasivnim hlađenjem. Za vrijeme pasivnog hlađenja, pokreće se pumpa pasivnog kruga, koja vrši cirkulaciju geosodne, a pumpa u krug unutrašnje instalacije grijanja obezbjeđuje protok vode koja preuzima toplotu iz prostora.

Tokom oba procesa, moguća je priprema potrošne tople vode. Pasivno hlađenje štedi energiju, koja je potrebna samo za rad cirkulacionih pumpi. Međutim, ovaj metod ima ograničenja po pitanju performansi i efikasnosti.

3.5. Koeficijent efikasnosti COP

Osnovni pokazatelj efikasnosti toplotnih pumpi, u režimu grijanja/hlađenja, naziva se koeficijent efikasnosti (COP, coefficient of performance). On je određen kao odnos toplotne energije koju je toplotna pumpa dovela nekom prostoru ili medijumu i pogonske energije kojom se ostvaruje proces u njoj.

Što je manja razlika između temperature toplotnog izvora i temperature toplotnog ponora, veća je efikasnost i ekonomičniji je rad toplotne pumpe, zato što kompresor treba utrošiti manju količinu energije kako bi podigao temperaturu rashladnog sredstva. Zbog toga je veoma važno optimizovati sistem grijanja/hlađenja.

4. MATERIJAL I METOD RADA

Za izradu ovog rada je bilo potrebno obezbijediti informacije o različitim toplotnim izvorima, kao i toplotnim ponorima, prosječne masene protoke fluida kroz sistem, matematičke modele koji će predstaviti korelaciju između svih komponenti toplotne pumpe, kao i jednačinu za izračunavanje koeficijenta efikasnosti toplotne pumpe. Računanje COP-a predstavljeno je sljedećim izrazom [6,7]:

$$COP = k * \frac{0.5 * T_{cin} + 0.5 * T_{cout} + 273.15}{(0.5 * T_{cin} + 0.5 * T_{cout}) - (0.5 * T_{vin} + 0.5 * T_{vout})}$$

- k – koeficijent efikasnosti kompresora
- T_{cin} – temperatura freona na ulazu u kondenzator [°C]
- T_{cout} – temperatura freona na izlazu iz kondenzatora [°C]
- T_{vin} – temperatura freona na ulazu u isparivač [°C]
- T_{vout} – temperatura freona na izlazu iz isparivača [°C]

Sljedećim izrazom se predstavlja ponašanje kondenzatora [6,7]:

$$C_c * \frac{dT_{cout}}{dt} = MF_{cin} * c_w * (T_{cin} - T_{cout}) + COP * E_{hp}$$

- C_c – toplotni kapacitet kondenzatora [J/K]
- MF_{cin} – maseni protok vode [kg/s]
- c_w – specifični toplotni kapacitet vode
- T_{cin} – temperatura freona na ulazu u kondenzator [°C]
- T_{cout} – temperatura freona na izlazu iz kondenzatora [°C]
- COP – koeficijent efikasnosti pumpe
- E_{hp} – električna energija toplotne pumpe [W]

Ponašanje isparivača je predstavljeno sljedećom jednačinom [6,7]:

$$C_v * \frac{dT_{vout}}{dt} = MF_{vin} * c_w * (T_{vin} - T_{vout}) - (COP - 1) * E_{hp}$$

- C_v – toplotni kapacitet isparivača [J/K]
- MF_{vin} maseni protok vode [kg/s]
- c_w – specifični toplotni kapacitet vode
- T_{vin} – Temperatura freona na ulazu u isparivač [°C]
- T_{vout} – Temperatura freona na izlazu iz isparivača [°C]

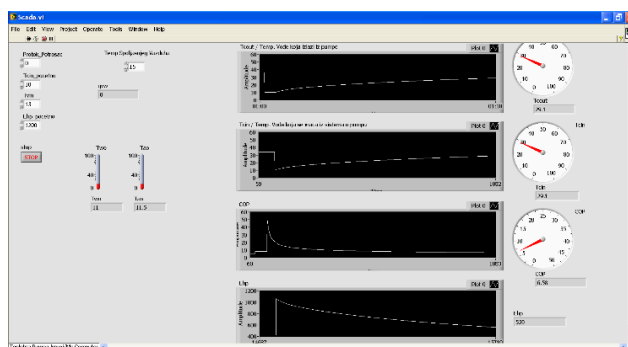
- COP – koeficijent efikasnosti toplotne pumpe
 - E_{hp} – električna energija toplotne pumpe [W]
- Matematički model rezervoara, koji simulira potrošač je predstavljeno sljedećom jednačinom [6,7]:

$$m * c_w * \frac{dT_{bout}}{dt} = MF_{bin} * c_w * T_{bin} - MF_{bout} * c_w * T_{bout}$$

gde su:

- m – masa rezervoara [kg]
- T_{bin} – temperatura vode koja ulazi u rezervoar [°C]
- T_{bout} – temperatura vode koja izlazi iz rezervoara [°C]
- MF_{bin} – maseni protok vode koja ulazi u rezervoar [kg/s]
- MF_{bout} – maseni protok vode koja izlazi iz rezervoara [kg/s]
- c_w – specifični toplotni kapacitet vode

Na slici 3 prikazan je SCADA sistem odrađen u programskom paketu LabWiev, i kao rezultat ovaj sistem prikazuje protok u izmjenjivaču, temperaturu vode na izlazu iz izmjenjivača, temperaturu koju odaje izmjenjivač, koeficijent efikasnosti COP, kao i utrošenu količinu električne energije.



Slika 3. Izgled SCADA sistema

Na slici 4, u nastavku, prikazan je izgled Simulink šeme, koji predstavlja reprezentaciju matematičkog modela toplotne pumpe u programskom jeziku Matlab.

5. REZULTATI SA DISKUSIJOM

Rezultati koji su prikazani su dobijeni u odnosu na formule navedene u prethodnom poglavlju, i predstavljaju pojednostavljen model toplotne pumpe, koju je moguće koristiti samo u režimu grijanja.

Kontrolna strategija je prilično jednostavna. Svi maseni protoci u sistemu su konstantni, i iznose 0.02kg/s, a toplotnom pumpom se upravlja pomoću vrijednosti temperature povratne vode iz sistema.

Simulacijom sistema unošenjem različitih vrijednosti ulaznih parametara je donešen zaključak da pumpa ima najbolje performanse ukoliko je temperaturna razlika između toplotnog izvora i toplotnog ponora manja. Što je manja razlika, kompresor treba raditi manje kako bi podigao temperaturu rashladnog sredstva, što dovodi to toga da COP raste, a rad pumpe je ekonomičniji.

OPTIMALNA REKONFIGURACIJA TOPOLOGIJE KAO METODA ZA UPRAVLJANJE ZAGUŠENJIMA U PRENOSNOJ MREŽI

OPTIMAL TOPOLOGY CHANGE AS METHOD FOR CONGESTION MANAGEMENT IN TRANSMISSION NETWORK

Biljana Durlević, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu opisani su problemi sa kojima se prenosna mreža susreće u deregulisanoj i otvorenom tržištu, pri čemu je zagušenje prenosnih vodova identifikovano kao glavni problem. Glavni cilj rada je da se pokaže kako se rekonfiguracija topologije prenosne mreže može koristiti kao korektivni mehanizam u slučaju pojave mogućih zagušenja u mreži. Odrađene su simulacije u sistemu za upravljanje prenosnom mrežom, kojima su potvrđene teoretske tvrdnje i pokazano kako se rekonfiguracija topologije prenosne mreže može koristiti kao efikasno sredstvo za rasterećenje od zagušenja prenosnih vodova.

Abstract – In this paper problems encountered by the transmission network in the deregulated and open market are described, with the congestion of the transmission lines being identified as the main problem. The main goal of this work is to show how the reconfiguration of topology of transmission network can be used as a corrective mechanism in case of possible congestion in the power network. Simulations were done in the transmission network management system, which confirmed theoretical assumptions and demonstrated how the reconfiguration of the topology of the transmission network can be used as an effective mean for congestion relief of transmission lines.

Cljučne reči: deregulacija, zagušenja prenosnih vodova, rekonfiguracija prenosne mreže, EMS

1. UVOD

Prenos električne energije igra ključnu ulogu u modernim elektroenergetskim sistemima. Savremeni trendovi u eksploataciji elektroenergetskih sistema orijentisani su na sve intenzivnije iskorišćenje postojećih proizvodnih i prenosnih kapaciteta, umesto na izgradnju novih. Otvoreni pristup prenosnoj mreži rezultirao je novim izazovima u upravljanju prenosnim sistemima. Zagušenje u električnoj mreži je ključna prepreka u živom konkurentskom tržištu električne energije. Da bi se obezbedili što veći ekonomski dobici i sigurno i pouzdano snabdevanje konzumenata električnom energijom mora da se odgovori na važno pitanje, a to je, koji je najbolji način da se organizuje prenos, a da ne dođe do remećenja pouzdanosti i trgovanja na tržištu. Zagušenje pojedinih vodova u elektroenergetskom sistemu se može tretirati na različite načine.

NAPOMENA:

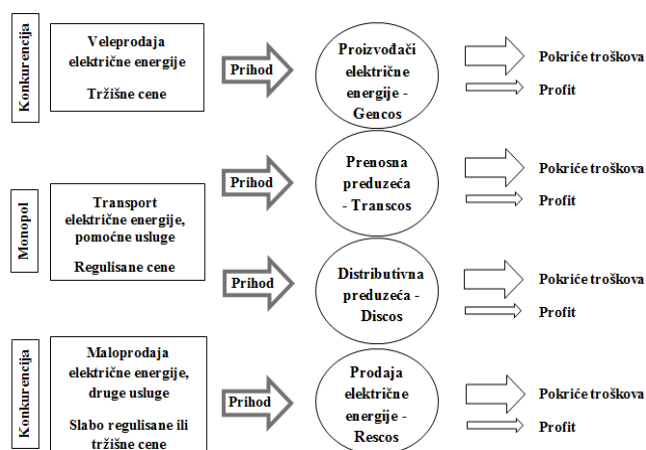
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Nenad Katić, red.prof.

Klasične metode koje se koriste bi mogle biti efikasnije, upravo zbog toga dolazi do potrebe za novim alatima. Ovde je pokazano upravo kako se rekonfiguracijom prenosne mreže tj. primenom sistema za upravljanje prenosnim mrežama u okviru koga se nalazi funkcija optimalne rekonfiguracije mreže, ovakva opterećena mreža može osloboditi od zagušenja.

2. DEREGULACIJA U ELEKTROPRIVREDI

Deregulacija u elektroprivredi zapravo predstavlja proces menjanja pravila i propisa koji kontrolišu elektroprivredu, a sve sa ciljem da dozvoljavanjem konkurencije na tržištu električne energije omogući kupcima izbor dobavljača električne energije koji mogu biti ili preprodavci (prodavci na malo) ili sami proizvođači (prodavci na veliko). Deregulacija bi trebalo da poboljša ekonomsku efikasnost proizvodnje i korišćenja električne energije, a usled konkurencije na tržištu, cene bi trebalo da opadnu i da se na taj način obezbedi benefit za kupce [1].

Osnovni principi deregulacije su razdvajanje ili razvezi vanje procesa proizvodnje i transporta električne energije od prodaje i usluga, kao i procesa proizvodnje od procesa prenosa i distribucije prenosne energije. U deregulisanim elektroprivredama dolazi do razdvajanja njihovih funkcija, disagregacije ili restrukturiranja. Uvođenje deregulacije je donelo nekoliko novih entiteta na tržište električne energije, dok je sa druge strane redefinisalo granice mnogih aktivnosti postojećih učesnika. Varijacije postoje na različitim tržištima prema tome kako je svaki entitet definisan i prema ulozi koju odigravaju u sistemu. Ipak, na širem nivou, sledeće celine se mogu identifikovati kao što je prikazano na Slici 1.



Slika 1. Restrukturiranje elektroprivrede [1]

su zagušenja prenosnih vodova, nestabilnost napona i nestanci struje. Sa postojećim mehanizmima za kontrolu mreže i neprekidnim razvojem, verovatnoća da će postojeća mreža preživeti kvarove ili ispade značajno opada, što naglašava potrebu za novim alatima za analizu i upravljanje električnom mrežom [2].

3. PROBLEM ZAGUŠENJA PRENOSNIH KAPACITETA

Zagušenje u prenosnoj mreži predstavlja stanje prenosne mreže u kome proizvođači ili potrošači električne energije žele da proizvedu, odnosno potroše električnu energiju na način koji bi doveo pogon prenosnog sistema na granicu jednog ili više ograničenja.

Ta ograničenja podrazumevaju prekoračenja termičkih granica opterećenja elemenata, nezadovoljenje naponskih prilika i nestabilnost rada sistema.

Uglavnom je interna mreža nekog sistema razvijenija od interkonektivnih veza između susednih sistema, pa se zbog toga nakon otvaranja međunarodnog tržišta električne energije zagušenja često javljaju na prekograničnim interkonektivnim vodovima.

Zagušenje je posledica ograničenja koja postoje u prenosnoj mreži, a koja se karakterišu konačnim kapacitetom mreže, koji sprečava istovremenu isporuku električne energije nekoliko prethodno predviđenih transakcija.

Ispadi vodova i veliki zahtevi za električnom energijom na određenoj putanji su razlozi zbog kojih nastaje zagušenje mreže.

Uvođenjem strukturne reorganizacije i deregulacije, u eksploataciju elektroenergetskih sistema, uočilo se da, zbog većeg broja transakcija, dolazi do sve češćih zagušenja prenosne mreže, a s tim i do naglašenije potrebe da se sistematskim dejstvom preventivnog i/ili korekcionog upravljanja, ta zagušenja izbegnu ili, ako se već pojave, da se efikasno otklone. U tu svrhu koriste se različiti metodi i sredstva, ekonomske, upravljačke i tržišne prirode, kao što su otkazivanje transakcija, preraspodela opterećenja generatora, rekonfiguracija prenosne mreže i redukcija potrošnje, kao krajnja mera.

Preduzimanje bilo koje od tih akcija ima značajan efekat na ekonomsko poslovanje učesnika na tržištu električne energije. Međutim, podjednaki ili čak i veći, ali negativni efekti imali bi se ako bi se dozvolio rad sistema sa tim, ili nekim drugim smetnjama, koje bi ugrozile sigurnosne granice rada sistema. Posledica tih činjenica je da su svi učesnici na tržištu električne energije zainteresovani da se problemi zagušenja i naponskih ograničenja prenosne mreže reše na vreme i na odgovarajući način.

Upravljanje zagušenjima je nešto složenije u konkurentnom tržištu i dovodi do nekoliko sporova. Zagušenja mogu biti ublažena kroz različite načine. Među tehničkim rešenjima postoje, redispečing sistema, rekonfiguraciju sistema, isključenje zagušenih vodova, korišćenje FACTS uređaja i regulacije na transformatorima.

Takođe, brojne studije su posvećene problemu upravljanja zagušenjima koristeći ekonomski redispečing generatora, sigurnosno ograničene optimalne tokove snaga, i isključenje opterećenja u kombinaciji sa redispečingom sistema [3].

4. SISTEMI ZA UPRAVLJANJE PRENOSNIM MREŽAMA

Sistemom za upravljanje prenosnim mrežama, EMS, obično nazivamo skup alata i računarske podrške koju operateri prenosnog sistema obično koriste za nadzor, upravljanje i optimizaciju rada sistema za proizvodnju i prenos električne energije. Generalno, EMS izvršava dve grupe zadataka: optimizaciju rada sistema u različitim radnim režimima (normalni sigurni, kritični, predhavarijski, havarijski i posthavarijski) i detekciju radnog režima sistema u pogledu granica stabilnosti. Većina industrijskih upravljačkih sistema (kakvi su i elektroenergetski sistemi) sakuplja podatke sa velikog broja mernih uređaja u sistemu na centralnoj lokaciji, ili više distribuiranih lokacija. Na osnovu tih podataka vrši se upravljanje sistemom. To prikupljanje podataka se vrši pomoću SCADA sistema. Prva namena SCADA sistema je bila praćenje stanja tehničkih procesa, međutim sada razvojem tehnologije njena uloga je proširena i funkcijama upravljanja. SCADA sistem sada obavlja dve važne funkcije a to su prikupljanje podataka (merenja i signalizacija) iz polja (sistema) i sprovođenje upravljačkih akcija. Sprovedene upravljačke akcije u sistemu moraju biti tačne i bezbedne. Pošto se one iniciraju iz regionalnih ili centralnog centra upravljanja, a sprovođenje na pojedinim uređajima na terenu, onda se mora vršiti njihovo nadgledanje. Nadgledanje (monitorisanje) podataka generisanih procesom u realnom vremenu je tipičan primer aktivnosti u realnom vremenu. Međutim, informacije generisane SCADA-om mogu se koristiti i na razne druge načine, a ne samo u realnom vremenu.

5. REKONFIGURACIJA TOPOLOGIJE PRENOSNE MREŽE

Uobičajena je praksa da operater sistema tretira elemente prenosne mreže (vodove i transformatore) kao statičke elemente u formulaciji problema optimalnih tokova snaga. Ovaj tradicionalni pogled ne opisuje prenosne elemente kao elemente koje operater ima mogućnost da kontroliše. Ipak, potvrđeno je, kako u teoriji tako i u praksi, da operateri sistema mogu da menjaju topologiju mreže da bi popravili naponske profile, povećali kapacitet prenosa, i čak i povećaju pouzdanost sistema [4]. Ovakva mogućnost bi bila više nego poželjna imajući u vidu rad današnjih električnih mreža. Zbog fizike koja upravlja tokovima električne energije i zbog složenosti u okviru ovog problema sa mrežnim tokovima snaga, izuzetno je malo verovatno da postoji jedna optimalna topologija mreže za sve periode i za sve moguće realizacije tržišta tokom dužeg vremenskog perioda. Optimizacija topologije prenosne mreže je obećavajuća opcija jer koristi postojeća sredstva tj. elemente prenosne mreže da postigne važne i pravovremenske ciljeve: povećanu fleksibilnost mreže i njenu efikasnost.

Optimalna rekonfiguracija topologije prenosne mreže je svakako inovativna metoda kada je u pitanju upravljanje prenosnom mrežom u smislu korektivnih akcija. Ona maksimalno koristi kapacitete prenosne mreže koristeći njene elemente kao dinamičke, tj sa mogućnošću njihovog uključenja/isključenja iz sistema. Prenosni vodovi se mogu uključiti ili isključiti iz pogona u cilju maksimiziranja ekonomske efikasnosti prilikom trgovine elek. energijom.

Rekonfiguracija prenosne mreže može se koristiti upravo kao sredstvo za uklanjanje prekoračenja koja su se javila usled zagušenja prenosnih vodova, tj. za relaksaciju vodova od zagušenja. Benefit od rekonfiguracije se može imati kod potrebe za smanjenjem preopterećenja i naponskih prekoračenja. Glavni problem koji ograničava primenu rekonfiguracije mreže je opasnost koju može da izazove vezano za sigurnost mreže.

U ovakvim okvirima, svaka metodologija koja može da pomogne operaterima prenosnih mreža u upravljanju mrežom (sa najmanjim uticajem na već isplanirane tržišne akcije) koja poseduje sigurnosna ograničenja čini se veoma korisnom. U nekim slučajevima, rekonfiguracija je u stanju da značajno smanji preopterećenja, i na taj način mogu da se izbegnu radnje vezane za preraspodelu opterećenja (redispečing) koje podrazumevaju menjanje unapred ugovorenih tržišnih transakcija [5].

Optimalna konfiguracija mreže je kombinatorni optimizacioni problem koji može da zahteva izuzetan napor prilikom proračuna za velike elektroenergetske sisteme. Međutim razvojem kompjuterizovanih softvera i hardvera novijih generacija javlja se realna mogućnost da se rekonfiguracija mreže i koristi u korektivne svrhe.

Da bi se prevazišla ova brojna ograničenja, a pre svega problem zagušenja prenosnih vodova mogu se upravo koristiti pametni sistemi za menadžment prenosnih mreža (Energetski menadžment sistemi – EMS) pomoću kojih se može pratiti stanje mreže i takođe u okviru kojih postoje različiti softverski alati za upravljanje mrežom koji mogu doprineti optimizaciji rada i iskorišćenja postojećih resursa mreže.

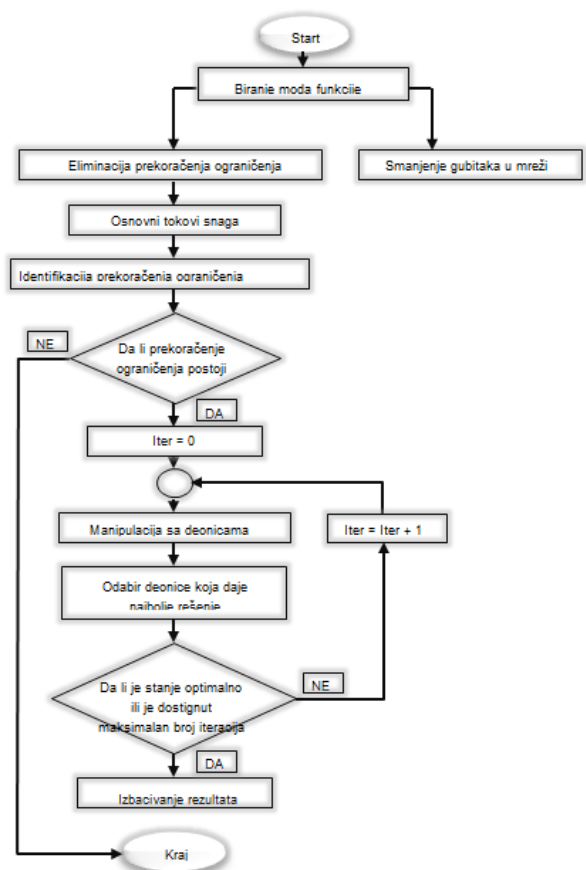
U ovom radu je upravo takav jedan alat preporučen za optimizaciju rada kada su u pitanju preopterećenja nastala kao posledica zagušenja prenosnih vodova prilikom trgovine električne energije.

5.1 Funkcija optimalne rekonfiguracije mreže

Funkcija optimalne rekonfiguracije mreže je funkcija koja se koristi da izračuna i predloži najbolju moguću konfiguraciju topologije prenosne mreže. Glavni zadatak ove funkcije je da eliminiše prekoračenja limita napona i preopterećenja vodova promenom topologije prenosne mreže.

Kao rezultat funkcije prenosni vodovi koji moraju da budu uključeni ili isključeni su izlistani. Rezultati operacija mogu da se koriste da: optimizuju topologiju prenosne mreže kako bi naponi bili u odgovarajućim granicama, smanje gubitke u prenosu prilagođavanjem topologije mreže, spreče kvarove na elementima mreže za trenutno stanje mreže signaliziranjem preopterećenja i potencijalnih kršenja ograničenja, pomognu u ispravljanju rada mreže u kvaru, gde preopterećenja i prekoračenja ograničenja mogu da se eliminišu, pomoć u studiji planiranja za postizanje boljih performansi buduće električne mreže, ukloni zagušenja vodova i napravi optimalni prolaz za snagu gde je prisutno tržište električne energije, itd.

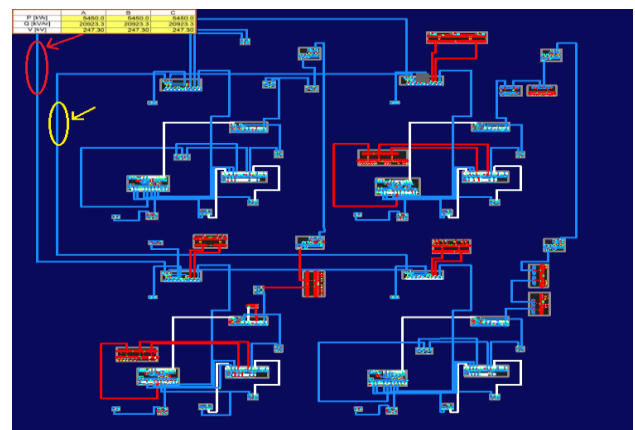
Ovaj rad je upravo fokusiran na poslednje navedenu mogućnost ove funkcije, a to je uklanjanje zagušenja na prenosnim vodovima. Algoritam rada ove funkcije dat je na Slici 2.



Slika 2. Algoritam funkcije za optimalnu rekonfiguraciju

6. SIMULACIJE OPTIMALNE REKONFIGURACIJE PRENOSNE MREŽE

U ovom primeru posmatra se situacija sa srednje opterećenom prenosnom mrežom. U simulacionom režimu je podešena takva situacija da je povećanjem proizvodnje određenih generatora za koje bi mogli uzeti pretpostavku da imaju najnižu cenu električne energije na tržištu, a isto tako smanjenjem proizvodnje na nekim drugim generatorima (onima sa višom cenom električne energije) došlo do povećanja opterećenja prenosnog voda koji se nalazi na putanji tokova snaga između prethodno pomenutih proizvođača i potrošača. U realnom sistemu upravo bi ovako izgledalo zagušenje voda nastalo usled povećanih transakcija zbog kupovine električne energije od proizvođača sa nižom cenom električne energije.



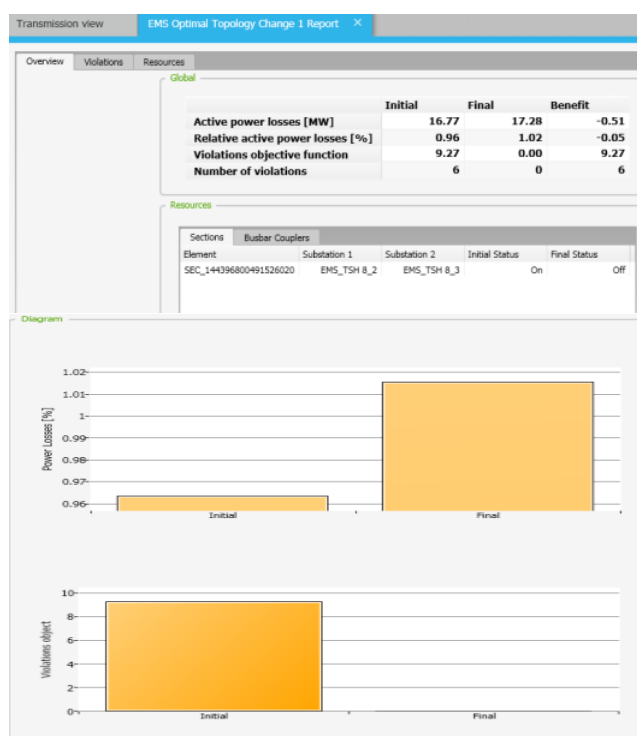
Slika 3. Razmatrana prenosna mreža

Na Slici 3 je prikazana razmatrana prenosna mreža sa prenosnim vodom (označenim crvenom bojom) na kome je simulirano zagušenje usled povećanih transakcija električne energije kao i opterećenje te deonice u momentu kada je nastalo preopterećenje.

Pošto je funkcija pokrenuta i izveštaj funkcije otvoren, mogu se uočiti sledeći rezultati. Algoritam prepoznaje upravo deonicu na kojoj smo simulirali zagušenje tj. deonicu sa preopterećenjem kao kritičnu, Slika 4.

Element	Substation 1	Substation 2	Loading [%]	Loading [MW]	Violations	Loading [%]	Loading [MW]	Violations
SEC_144396800491526023	EMS_TSH 8	EMS_TSH 8_4	117.57	115.02	Critical	-	-	OK

Slika 4. Kritična deonica



Slika 5. Rezultat rada algoritma

Ponudeno rešenje rada funkcije kao i opšti pregled stanja pre rekonfiguracije i posle može se videti u sekciji za pregled (Overview) Slika 5.

Ovde se vidi da je rešenje za rekonfiguraciju koje je funkcija ponudila za uklanjanje zagušenja na vodu upravo isključenje deonice iz pogona. Ovo zapravo predstavlja dokaz teoretske tvrdnje koja je ranije navedena da se čak i samo isključenjem pojedinih vodova iz sistema može popraviti sigurnost mreže i da se uklone preopterećenja na vodovima od važnosti.

Iako, isključenje prenosnog voda sa ciljem rasterećenja drugog prenosnog voda koji se želi rasteretiti od zagušenja i od preopterećenja, ne zvuči baš kao najprirodnije i najlogičnije rešenje, ovde se može videti da se upravo ovakvom akcijom postiže najoptimalnije stanje mreže za date transakcije električne energije i uklanjaju se mnoga prekoračenja ograničenja koja su postojala pre isključenja voda.

Sa Slike 5 se vidi da bi se ponuđenom rekonfiguracijom gubici aktivnih snaga u razmatranom delu prenosne mreže nešto malo povećali u odnosu na početno stanje, ali bi se stanje mreže svakako poboljšalo, što se može videti iz rezultata rada funkcije za eliminaciju prekoračenja i samim brojem prekoračenja koji postoje u inicijalnom i finalnom stanju. Vidi se da bi se isključenjem ove sekcije uklonilo svih 6 prekoračenja ograničenja koja postoje u prenosnoj mreži, tako da od početnih 6 koja su postojala na kraju rekonfiguracije ne bi ostao ni jedan, tj. cela mreža bi se nalazila u optimalnom stanju i svi elementi unutar svojih granica sigurnosti.

Među tih 6 prekoračenja ograničenja koja bi se eliminisala rekonfiguracijom, eliminisao bi se upravo naš kritični slučaj, tj. zagušena deonica koja bi se ovom akcijom rekonfiguracije rasteretila i rad prenosnog sistema bi se mogao nastaviti sa unapred dogovorenim transakcijama električne energije, dakle bez remećenja tržišta. Na Slici 3. žutom bojom je označena deonica koja je predložena za isključenje.

7. ZAKLJUČAK

Može se zaključiti kako se rekonfiguracijom prenosne mreže mogu postići itekakvi benefiti vezani za uklanjanje zagušenja u prenosnoj mreži i da se na taj način omogući nesmetana trgovina električnom energijom i povećati sigurnost rada mreže. Upotrebom dobrih softverskih alata za upravljanje prenosnim mrežama, koji mogu brzo i efikasno da odrede raspodelu tokova snaga u prenosnoj mreži pre i posle moguće rekonfiguracije, može se na brz i efikasan način reagovati prilikom zagušenja, što ovu metodu čini veoma dobrom za upravljanje i rešavanje problema sa zagušenjima.

8. LITERATURA

- [1] Nenad Katić: Elektroprivreda u uslovima slobodnog tržišta, FTN izdavaštvo, Novi Sad 2012
- [2] L.Philipson, H. Lee Willis: Understanding Electric Utilities and De-Regulation, Taylor & Francis Group, Boca Raton, 2006
- [3] Snežana Mihajlović: Metode za upravljanje zagušenjima u visokonaponskoj mreži, Magistarski rad, Fakultet tehničkih nauka, Novi Sad, 2009.
- [4] W. Shao: Transmission system reconfiguration for corrective control, Iowa State University, 2006, <http://lib.dr.iastate.edu/rtd/1304>
- [5] H.Ahmadi, M. Khanabadi, H. Ghasemi: Transmission System Reconfiguration for Congestion Management Ensuring Transient and Voltage Stability, University of British Columbia Vancouver, BC, 2013

Kratka biografija:



Biljana Durlević rođena je u Skoplju 1987. godine. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranila je 2017.godine.

VEB APLIKACIJA ZA ORGANIZOVANJE FUDBALSKOG TAKMIČENJA

WEB APPLICATION FOR ORGANIZING A FOOTBALL COMPETITION

Rajko Ilić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U okviru ovog rada je prikazana implementacija aplikacije za realizaciju organizovanja fudbalskog takmičenja. Za upravljanje bazom podataka korišćen je MySQL. Alat Power Designer je omogućio kreiranje modela na osnovu kojeg je automatski izgenerisan skript za formiranje šeme baze. Aplikacija je radjena u programskom jeziku Java.

Abstract – This thesis presents implementation of organizing football competition. MySQL is used as a database management system. Power Designer is used for definition of a physical data model in order to support automatic generation of database schema. Client application is implemented in Java.

Ključne reči: JPA, Spring, Kvota, Žreb, ORM

1. UVOD

Pod organizovanjem fudbalskog takmičenja podrazumeva se formiranje rasporeda utakmica u toku sezone za određenu ligu, osposobljavanje stručnih ljudi koji će predstavljati administrativni i kontrolni organ kao i kontrolisanje novčanih nagrada, pravila i medijskih prava vezanih za to takmičenje.

Previše je nejednakosti u fudbalu, a samo je mali broj elitnih klubova koji drže sportski i finansijski monopol ove igre. U većini današnjih liga širom sveta, izdvajaju se klubovi koji su dominantniji i odskaču od drugih i samim tim su uvek pobednici tih liga ili su u samom vrhu. Razlog za dominaciju leži upravo u činjenici što pojedini klubovi iskorišćavaju slabije. Sprovodi se 'bullying' na fudbalskom tržištu i manjim klubovima se otimaju najveći talenti. Kao jedan od zadataka neke organizacije fudbalskog takmičenja je da spreči takvu zloupotrebu moći i dozvoli manjim klubovima da se ravnopravno takmiče. Velika količina novca se obrće u fudbalu. To je razlog što se sve veći broj ljudi okreće ka njemu. Između ostalog, profit ostvaruju i kladionice. Za formiranje kvota one koriste svoje kompjuterske programe koje sadrže u sebi mnogo statističkih informacija. Kladionice na svaki meč imaju određenu svotu novca koju rizikuju. One ne smeju da dozvole sebi da izgube sav novac. U protivnom one mogu da bankrotiraju. Medjutim, kladionice imaju razradjene mehanizme da se od toga odbrane. Kladionica na osnovu svoje statistike ponudi kvotu, ali sa malom količinom novca za rizik. One prate uplate svojih klijenata, kao i kretanje kvota drugih kladionica i na osnovu toga spuštaju ili dižu kvotu kako bi smanjile rizik.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Aleksandar Kupusinac, red.prof.

Zadatak ovog rada je projektovanje i implementacija podsistema za organizaciju fudbalskog takmičenja gde su glavni zadaci generički formiranje žreba za fudbalsku ligu kao i formiranje kvota na osnovu statističkih podataka. Neki od tih statističkih podataka su klupski koeficijenti na osnovu FIFA-ine rang liste, ostvareni broj bodova, da li se igra na domaćem terenu, žuti, crveni kartoni, povredjeni igrači, golovi, asistencije i drugi. Korisniku ovog sistema je omogućeno vođenje evidencije o trenutnom stanju neke fudbalske lige, mogućnost pretrage, preglednost i to sve doprinosi jednostavnijem, bržem i lakšem pristupu.

2. OPIS REŠAVANOG PROBLEMA

2.1 Opis funkcionisanja

Da bi se uopšte započeo proces rada za organizovanje fudbalskog takmičenja, najpre je potrebno definisati raspored utakmica, tačnije definisati žreb. Žreb se formira generički na osnovu ulaznih podataka. Ulazne podatke predstavlja broj klubova kao i njihovi nazivi. Formiranje žreba je složen proces i u ovom radu se odvija se na sledeći način:

1. Odrediti ligu za koji se formira žreb
2. Definisati broj timova (N), stim što broj timova mora biti paran broj
3. Formirati N-1 lista gde je u okviru svake od nje moguće smestiti N/2 parova
4. Odabira se "slučajno odabrani tim" i njegov par sa svakim različitim timom ponaosob se smešta u svaku od lista
5. U svakoj sledećoj iteraciji bira se naredni slučajno odabrani tim od preostalih, stim što se pazi na sledeće stvari:
 - a. Ukoliko „slučajno odabrani tim” ili tim koji mu je pridružen već postoje u listi gde bi trebalo da se upišu, ne upisuju se tj. ignoriše se i prelazi se na sledeću listu
 - b. Slučajno odabrani tim i pridruženi tim ne smeju biti isti
 - c. Pamte se upisani parovi, koji će poslužiti kao proveru da li je taj par već negde upisan
6. Od svih nepopunjenih polja za svaku od lista pronalaze se koji klubovi fale u tim listama
7. Ukoliko u nekojod lista fali samo jedan par da se upiše, on se automatski upisuje tako što se proverava koja su to dva tima koja fale
8. Od klubova koji fale pravimo sortiranu listu sa vrednostima koliko koji klub još treba da se upiše
9. Iz sortirane liste se pronalazi klub koji treba na najmanji broj mesta da se upiše, i on se upisuje u nepopunjena polja sa odgovarajućim klubovima

10. Postupak se ponavlja dok ne dodjemo do kluba koji treba na najveći broj mesta da se upiše i on se sa odgovarajućim klubovima upisuje na preostala nepopunjena mesta.
11. Na ovaj način smo dobili N-1kola od ukupno (N-1) * 2. Potrebno je formirati još N-1kola sa promenjenim mestima timova u svakom paru iz prvih N-1 kola. Time je obezbedjeno da će klub igrati i kao domaćin i kao gost protiv njemu pridruženog tima. Dobili smo ukupno (N-1) * 2 kola i time je žreb završen.

Kroz aplikaciju sa administratorske strane postoji mogućnost unošenja klubova i igrača, mada s obzirom na veliki broj klubova a samim tim i ogroman broj igrača, sve ove podatke treba čuvati u skriptama kako bi se u slučaju njihovog ponovnog unosa smanjilo vreme unošenja i automatski izgenerisalo u bazi podataka.

Prilikom unosa klubova od obaveznih polja za unos navode se ime kluba, grad i država kojoj pripada, naziv stadiona na kome igra svoje takmičarske mečeve. Unosi se i logo kluba, postoje različiti sajtovi gde se može preuzeti logo približno fiksne veličine pa ih je poželjno preuzeti pre njihovog unosa u aplikaciju. Navodi se klupski koeficijent na osnovu FIFA-ine rang liste. Klubovi koji češće igraju evropska takmičenja imaju i veći koeficijent, jer dobri rezultati tamo doprinose mnogo većem koeficijentu. Klupski koeficijent se računa tako što se gledaju ostvareni rezultati kluba u zadnjih 5

godina, a u godinama pre toga se zanemaruju. Unosi se i rang kluba koji predstavlja koju ligu igrate u državi. Ovaj rang je jako bitan prilikom generičkog formiranja žreba jer se za ulazne podatke mogu automatski pokupiti klubovi na osnovu njihovog ranga.

Podaci koji se unose za klub kada počne takmičarska sezona su:

1. Broj odigranih mečeva
2. Broj pobeda
3. Broj nerešenih rezultata
4. Broj poraza
5. Broj postignutih golova
6. Broj primljenih golova
7. Broj bodova

Ovi podaci biće sadržani u formiranju tabele određene fudbalske lige. Ukoliko je broj bodova dva ili više tima isti, kriterijum koji odlučuje koji od ta dva ili više timova ima bolji plasman je sledeći:

1. Međusobni dueli dva ili više tima (sabira se broj ostvarenih bodova između njih)
2. Bolja gol razlika
3. Veći broj postignutih golova
4. Veći broj golova postignutih u gostima
5. Bolji klupski koeficijent

Prilikom unosa igrača od obaveznih polja se unose ime igrača, klub za koji trenutno nastupa, ime igrača, pozicija na kojoj igra kao i broj na dresu. Treba napomeniti da igrač može biti i bez trenutnog kluba ali nikako ne može u jednom trenutku biti učlanjen u dva kluba. U toku takmičarske sezone za igrača se unose i postignuti golovi, asistencije, broj žutih i crvenih kartona kao i to da li je povredjen. U slučaju crvenog ili drugog žutog kartona igrač propušta narednu utakmicu, a u slučaju povrede

navodi se koji broj utakmica propušta, tačnije na koliko će biti odsutan. Sve su ovo merila koja će imati bitan faktor prilikom računanja kvota.

Kladionice zaradjuju na osnovu marže koje imaju. Ona se kreće od 12%-2%. Na primer za određeni meč kladionica je ponudila sledeće kvote:

- 1 13
- X 5.5
- 2 1.2.

Ako primenimo formulu verovatnoće: $1/13 + 1/5.5 + 1/1.2 = 1.09$. Dakle ovaj broj je veći od 1 čime kladionica obezbeđuje sebi dobitak, $1 / 1.09 = 0.915\%$ što znači da je marža kladionice na ovom meču bila 8.5 %. Na svakih uplaćenih 100 e na ovom meču, kladionica na duže staze ima čist profit od 8.5 e.

U ovom radu kvote su formirane tako da obuhvataju i maržu, tj procenat od koje će kladionice imati profit.

Formiranje kvota je realizovan na sledeći način:

1. Odabere se jedno kolo koje je formirano prilikom žreba
2. Prolazi se kroz svaki par u okviru tog kola
3. U okviru jednog para pristupa se i jednom i drugom klubu i dobavljaju statistički podaci vezani za njih kao što su:
 - Klupski koeficijent na osnovu FIFA-ine rang liste
 - Broj ostvarenih bodova na tabeli
 - Da li neki igrač propušta utakmicu zbog dva žuta kartona
 - Da li neki igrač propušta utakmicu zbog crvenog kartona
 - Da li neki igrač propušta utakmicu zbog povrede
 - Koji tim je domaćin na ovoj utakmici
4. Klubu se za igrača koji će biti odsutan sa tog meča smanjuje ukupni koeficijent po formuli:
 - $2.5 * \text{koeficijent igrača}$
 - koeficijent igrača se menja iz svake utakmice u utakmicu u zavisnosti od toga kako je odigrao meč (da li je postigao gol, asistenciju)
5. Timu koji je domaćin na tom meču se ukupni koeficijent povećava za 60
6. Ukupni koeficijent domaćina = Klupski koeficijent + Broj ostvarenih bodova na tabeli – $2.5 * (\text{zbir koeficijenata igrača koji odsustvuju sa te utakmice}) + 60$
7. Ukupni koeficijent gosta = Klupski koeficijent + Broj ostvarenih bodova na tabeli – $2.5 * (\text{zbir koeficijenata igrača koji odsustvuju sa te utakmice})$
8. Verovatnoća da će tim domaćin pobediti na toj utakmici:
 - $uk1 / (uk1 + uk2) * 100$, gde je
 - $uk1$ - Ukupni koeficijent domaćin, $uk2$ - Ukupni koeficijent gost
9. Verovatnoća da će tim gost pobediti na toj utakmici:
 - $uk2 / (uk1 + uk2) * 100$, gde je
 - $uk1$ - Ukupni koeficijent domaćin, $uk2$ - Ukupni koeficijent gost
10. Kvota na domaćina:
 - $kvota1 = 125 / v1$, gde je
 - $v1$ - Verovatnoća da će tim domaćin pobediti na toj utakmici
11. Kvota na gosta:

- kvota2 = 125 / v2, gde je
- v2 - Verovatnoća da ce tim gost pobediti na taj utakmici

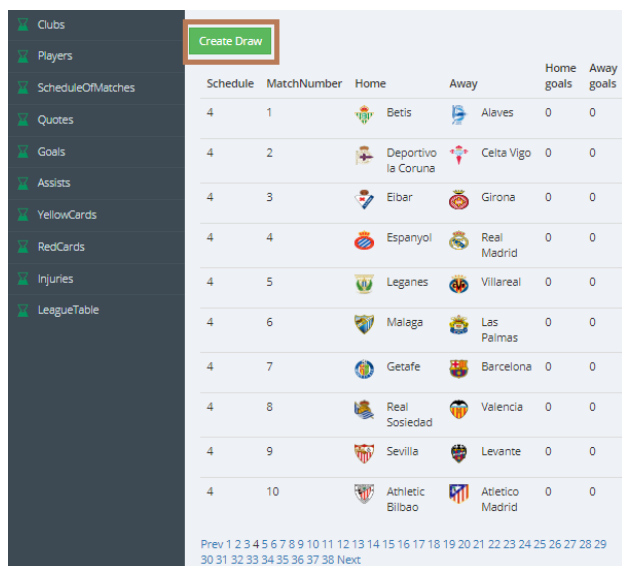
12. Kvota na nerešen ishod:

- kvotaX = (kvota1 + kvota2) / 2 + 0.5, gde je
- kvota1 - Kvota na domaćina, kvota2 - Kvota na gosta

13. Ovim smo obezbedili da marža kladionice bude oko 12 %

3. FORMIRANJE ŽREBA

Prilikom kreiranja klubova za svaki od njih se definiše rang, tj u kojoj ligi igra u određenoj državi, npr u Španiji postoje La Liga, La Segunda, Segunda Division B i druge. Prilikom kreiranja žreba neke lige određene države potrebno je navesti rang lige za koju želimo da izgenerišemo žreb kao što je prikazano na slikama 3.1 i 3.2. Posle toga je sve automatizovan proces. Prolazi se kroz sve klubove u bazi koji sadrže taj rang i u odnosu na njihov broj generiše se žreb. Treba napomenuti da broj timova za neki rang prilikom generisanja žreba mora biti paran broj. O ovome brine administrator.



Slika 3.1 Formiranje žreba se vrši klikom na dugme uokvireno braon bojom



Slika 3.2 Rang lige za koju želimo da izgenerišemo žreb

Treba napomenuti da kreiranje žreba za formiranje neke lige koja sadrži 20 timova traje manje od minute, pošto je ovo složen proces koji je opisan u prethodnim poglavljima. Medjutim, sobzirom da se žreb vrši jednom godišnje ovo svakako nalazi veliku primenu. Na slici 3.3 prikazano je formiranje žreba za 20 timova, a samim tim imamo 38 kola, gde kroz paginaciju na dnu označeno braon bojom možemo da pristupimo svakom.

Schedule	MatchNumber	Home	Away	Home goals	Away goals
2	1	Atletico Madrid	Alaves	0	0
2	2	Barcelona	Celta Vigo	0	0
2	3	Betis	Girona	0	0
2	4	Deportivo la Coruna	Real Madrid	0	0
2	5	Sevilla	Villareal	0	0
2	6	Getafe	Eibar	0	0
2	7	Espanyol	Leganes	0	0
2	8	Levante	Valencia	0	0
2	9	Las Palmas	Real Sociedad	0	0
2	10	Athletic Bilbao	Malaga	0	0

Slika 3.3 Formiranje žreba za 20 timova

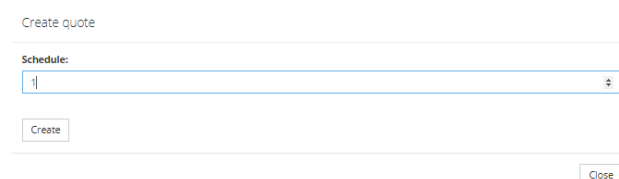
4. FORMIRANJE KVOTA

Formiranje kvota predstavlja takodje složen proces, i pretpostavlja se da svaka kladionica ima svoje mehanizme za njihovo definisanje. U ovom radu parametri koji su uzimani u obzir prilikom njihovog definisanja su:

1. Klupski koeficijent
2. Broj ostvarenih bodova na tabeli
3. Koji od timova igra na domaćem terenu
4. Odsustvo igrača zbog dva žuta kartona
5. Odsustvo igrača zbog crvenog kartona
6. Odsustvo igrača zbog povrede

Ukupni koeficijenti za svaki od timova se dobijaju tako što se sabere prva 3 parametra i od njih oduzmu zadnja tri čiji se koeficijenti množe sa 2.5. O ovome je bilo više reči u prethodnim poglavljima. Na taj način se dobijaju verovatnoće timova za pobedu i otuda se formiraju kvote, stim što zbir verovatnoća za pobedu domaćina, pobedu gosta i nerešen ishod mora da prelazi 100, jer se tako formira marža, a to je izvor zarade za kladionice na duže staze.

Administrator aplikacije formira kvote tako što bira za koje kolo želi da formira kvote kao što je prikazano na slici 4.1. Kvote za 1 i 2 kolo su redom prikazane na slikama 4.2 i 4.3. Neophodan uslov da bi se kreirale kvote za određeno kolo jeste da sve utakmice u prethodnom kolu moraju biti završene. O ovome takodje brine administrator.



Slika 4.1 Administrator aplikacije formira kvote tako što bira za koje kolo želi da formira kvote

Schedule	MatchNumber	Home	Away	Win	Draw	Lost
1	1	 Alaves	 Athletic Bilbao	2.09	3.1	3.11
1	2	 Celta Vigo	 Atletico Madrid	2.99	3.07	2.15
1	3	 Girona	 Barcelona	3.38	3.18	1.98
1	4	 Real Madrid	 Betis	1.41	6.57	10.73
1	5	 Villareal	 Valencia	1.8	3.44	4.08
1	6	 Deportivo la Coruna	 Eibar	1.54	4.6	6.67
1	7	 Getafe	 Espanyol	1.61	4.09	5.57
1	8	 Leganes	 Las Palmas	1.57	4.33	6.09
1	9	 Real Sociedad	 Sevilla	2.94	3.06	2.18
1	10	 Malaga	 Levante	1.57	4.36	6.15

1 2 Next

Slika 4.2 Formirane kvote za prvo kolo

Schedule	MatchNumber	Home	Away	Win	Draw	Lost
2	1	 Alaves	 Atletico Madrid	3.3	3.16	2.01
2	2	 Celta Vigo	 Barcelona	2.94	3.06	2.18
2	3	 Girona	 Betis	1.71	3.67	4.63
2	4	 Real Madrid	 Deportivo la Coruna	1.38	7.68	12.98
2	5	 Villareal	 Sevilla	2.44	3.0	2.56
2	6	 Eibar	 Getafe	1.58	4.26	5.95
2	7	 Leganes	 Espanyol	1.63	4.02	5.41
2	8	 Valencia	 Levante	1.49	5.11	7.74
2	9	 Real Sociedad	 Las Palmas	1.54	4.59	6.65
2	10	 Malaga	 Athletic Bilbao	2.05	3.12	3.2

Prev 1 2

Slika 4.3 Formirane kvote za drugo kolo

5. ZAKLJUČAK

Ovom aplikacijom je omogućeno korisnicima da na brz i efikasan način pretražuju rezultate i da imaju uvid o učinkovitosti njihovih omiljenih timova. Isto tako, mogu da se informišu o kvotama za pojedine mečeve, i na osnovu toga da sastavljaju tikete kod kuće. Treba napomenuti da kvote koje su formirane u ovoj aplikaciji ne odskaku mnogo od kvota u kladionicama, što je dobar pokazatelj da su parametri koji su uzimani u obzir prilikom sastavljanja kvota više nego dobri.

Danas se u svetu ogromna količina novca vrti oko fudbala. Ogroman kapital vlasnika klubova i pojedinih kompanija odlazi na kupovinu igrača ili čak deonice nekih klubova. Naravno, taj novac im se kroz izvesno vreme i povrati, kroz marketing, udeo od TV gledanosti, prodaje dresova i slično. Može se slobodno reći da fudbal danas nije samo sport ili igra iz zabave, već i biznis kojim se bavi veliki broj ljudi.

Iz tog razloga ova aplikacija može imati veliku primenu. Da bi ova aplikacija mogla da obuhvati sve lige sveta, potrebno je proširiti ovu aplikaciju, a isto tako za njeno funkcionisanje je potrebno angažovati veliki broj ljudi koji će je stalno ažurirati.

6. LITERATURA

- [1] MySQL:
http://www.pmf.ni.ac.rs/pmf/licne_prezentacije/230/przentacija/MySQL-1.pptx
- [2] Sybase PowerDesigner okruženje:
http://www.academia.edu/6471375/Uvod_u_Sybase_PowerDesigner_okru%C5%BEenje
- [3] Spring/hibernate:
<http://yuinfo.artkey.rs/zbornici/2009/html/pdf/072.pdf>
- [4] Learning Betting:
<https://www.bettingexpert.com/rs/akademija/osnove-kladjenja/kvote-za-kladionicu>

Kratka biografija:



Rajko Ilić, rođen je u Sremskoj Mitrovici 14.5.1993. godine. Završio je osnovnu školu „Dušan Jerković“ u Rumi. Nakon toga u istom mestu upisuje gimnaziju „Stevan Puzić“, opšti smer. Ubrzo posle toga upisuje Fakultet tehničkih nauka u Novom Sadu 2012 godine, a 2014 se usmerava na Primenjene računarske nauke. Ispunio je sve obaveze i položio sve ispite predviđene studijskim programom. Nakon završenih osnovnih studija stiče status diplomiranog inženjera elektrotehnike i računarstva 2016 godine, nakon čega upisuje master studije, smer elektronsko poslovanje.

STANICE ZA PUNJENJE ELEKTRIČNIH AUTOMOBILA NA TRASI SUBOTICA-BAR CHARGING STATIONS FOR ELECTRIC AUTOMOBILES ALONG THE ROUTE SUBOTICA-BAR

Radisav Ristić, Vladimir Katić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je opisan metod za pozicioniranje stanica za punjenje električnih vozila na trasi Subotica-Bar upotrebom određenih kriterijuma koji definišu potrebnu lokaciju, tip punjača, dozvoljeno rastojanje i veličinu stanice. Zatim je kreiran model električnog automobila u okviru softverskog paketa Matlab/Simulink sa ciljem preciznijeg određivanja lokacija za punjače, upotrebom ciklusa vožnje koji odgovara uslovima puta na predviđenoj trasi i bloka koji simulira nagib puta u određenom vremenskom periodu. Potom je izabrana oprema koja čini stanicu i odrađena je tehnno-ekonomska analiza i isplativost ovog projekta.

Abstract – The paper describes the method for positioning stations for charging electric vehicles on the Subotica-Bar route using certain criteria that define the required location, the type of charger, the allowed distance and the size of the station. Then an electric car model within the Matlab / Simulink software package was created with the aim of more precisely determining the locations for the chargers, using a driving cycle that corresponds to the road conditions on the predicted route and a block that simulates the inclination of the road in a specific time period. After that the equipment that makes the station was selected and the techno-economic analysis and cost-effectiveness of this project was made.

Ključne reči: Stanice za punjenje električnih vozila, Modelovanje i simulacija električnog vozila.

1. UVOD

U poslednjih par godina primećuje se značajan porast prodaje vozila na električni pogon, napajanih isključivo iz baterije. Podaci govore da je broj električnih vozila od par stotina iz 2005. porastao na jedan milion u 2015, a zatim još dodatnih 60% tokom naredne godine. Ovaj rast je posledica kako državnih subvencija i olakšica, tako i rasta infrastrukture za punjenje ove vrste vozila.

Prema tome, možda i bitniji faktor za brže prihvatanje ovakvog vida transporta od subvencija jeste pokrivenost glavnih koridora stanicama za punjenje i ako je moguće da one budu što veće snage kako bi omogućile minimalno zadržavanje vozača.

U ovom radu je obrađena trasa Subotica-Bar koju čine koridor X i koridor XI koji se trenutno gradi.

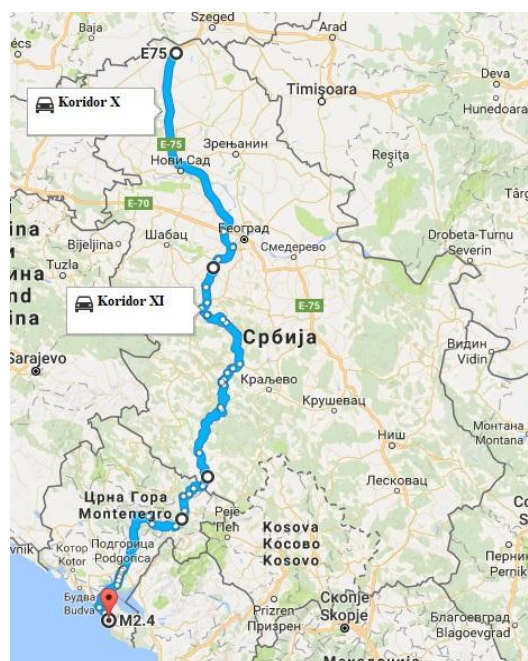
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio prof. dr Vladimir Katić.

Određenim kriterijumima se došlo do lokacija za stanice, a zatim je upotrebom simulacionog modela električnog vozila preciznije lociran pogodan položaj punjača na trasi. Parametri modela su odabrani tako da što bliže predstavljaju vozilo Nissan Leaf. U modelu je korišćen vektorski upravljani sinhroni motor sa permanentnim magnetima utisnutim u rotor, koji se putem invertora napaja iz baterije [1].

2. KARAKTERISTIKE TRASE

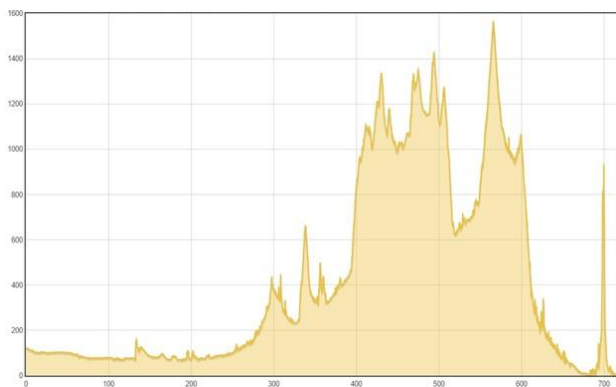
Početak trase na kojoj se razmatra postavljanje stanica za punjenje električnih vozila je petlja „Subotica-jug“ u blizini Subotice. Dužina razmatrane trase koja je prikazana na slici 1. je oko 587km i završava se u primorskom gradu-luci Bar na jugu Crne Gore. Trasa se prostire autoputem E75 koji na državnom nivou nosi oznaku A1 od Subotice do Beograda u dužini od 170km, zatim autoputem E763 (deonica je delimično izgrađena) koji na državnom nivou nosi oznaku A2 od Beograda do graničnog prelaza Boljare sa Crnom Gorom u dužini od 248km i na kraju autoputem Boljare-Bar (trenutno u izgradnji) u dužini od 165km.



Slika 1. Izgled trase Subotica-Bar

Projektovana brzina na autoputevima se kreće od 80km/h do 120km/h, brzina varira usled delova puta koji prolazi kroz tunel ili zbog uzdužnih nagiba. Do 2020. godine se očekuje promet na autoputu E763 od 18.000 do 37.000 vozila.

Na slici 2. prikazana je promena vertikalnog profila puta duž trase. Promena nadmorske visine je značajna i kreće se od 340m kod Gorobilja do 1.270m kod graničnog prelaza Boljare.



Slika 2. Promena nadmorske visine na trasi Subotica-Bar

3. IZBOR LOKACIJA ZA POSTAVLJANJE STANICA

Kako bi se odredile lokacije za postavljanje punjača neophodno je formirati određene kriterijume koje potencijalna lokacija mora da zadovolji. Prema tome kriterijumi koji su uzeti u obzir su oni koji definišu tip lokacije tj da li se uz autoput nalazi objekat poput benzinske stanice, odmorišta, naplatne rampe ili motela koji mogu da smanje troškove instalacije stanice zbog već dostupne elektroenergetske instalacije, kao i sigurnosti same stanice.

$$MDFC = R_m \cdot (1 - (M_w + k \cdot M_f)) \quad (1)$$

Zatim se kao drugi kriterijum nameće rastojanje između stanica, a ono zavisi od dometa električnog vozila. Uzimajući vreme realizacije razmatrane trase izabran je domet vozila Nissan Leaf za 2018. godinu sa baterijom većeg dometa kao referentni, koji iznosi 376km (EPA).

Prilikom proračuna maksimalnog rastojanja između stanica koristi se izraz (1). koji u obzir uzima domet vozila R_m , vremenski faktor M_w koji utiče na češće korišćenje uređaja za klimatizaciju što dalje utiče na veću redukciju dometa kao i stila vožnje i brzinu vožnje M_f koji takođe mogu da brže prazne bateriju.

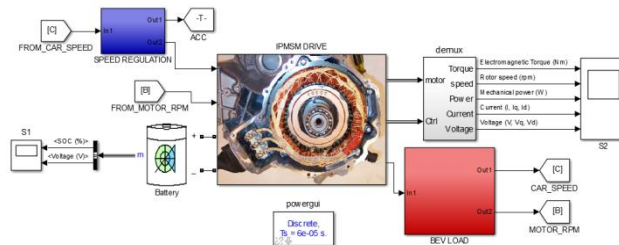
Kriterijumi vezani za tip punjača i veličinu stanice uzimaju u obzir najprodavanije modele u Evropi, na osnovu kojih je određen standard konektora kojim se vozilo priključuje na punjač, pa je zaključak da stanica mora da poseduje tri tipa konektora: Type 2, CHAdeMO i CCS Combo.

Zatim je zaključeno da zbog što kraćeg zadržavanja, punjači moraju da budu brzi odnosno veće snage, a to je moguće upotrebom brzih DC punjača.

Na mestima gde postoje moteli predlaže se upotreba sporijih punjača. Kriterijum za veličinu stanice na osnovu pretpostavke broja električnih vozila na trasi Subotica-Bar definiše upotrebu dva DC punjača u blizini većih mesta, a jedan sa opcijom naknadnog proširenja u blizini manjih naselja ili u blizini već postavljenih punjača [2].

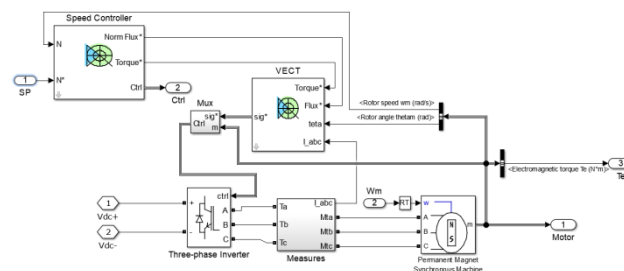
4. MODEL ELEKTRIČNOG AUTOMOBILA

Kako izraz (1) daje dosta grube rezultate po pitanju maksimalnog rastojanja između dve stanice, jer ne uzima u obzir nagib puta, regenerativno kočenje itd, kreira se model električnog automobila u okviru Matlab/Simulink softvera koji će obuhvatiti i ove faktore i na osnovu toga preciznije odrediti rastojanje između punjača.



Slika 3. Model pogona električnog automobila napajanog iz baterije

Na slici 3. je prikazan model pogona električnog automobila napajanog iz baterije. Kao pogonski električni motor korišćen je sinhroni sa stalnim magnetima utisnutim u rotor snage 100kW, napona 288Vdc, momenta 254Nm, nominalne brzine 2600rpm i sa 4 para polova.



Slika 4. Podsystem IPMSM DRIVE bloka

Vektorsko upravljanje, regulacija momenta i napajanje motora obuhvaćeno je blokom IPMSM DRIVE čiji je podsystem prikazan na slici 4. Pošto je cilj vektorskog upravljanja nezavisna kontrola momenta i fluksa u regionu brzina nižih od nominalne, kontrolišući d komponentu struje uz $i_d=0$, moment mašine T_e je funkcija struje i_q prema:

$$T_e = \frac{3}{2} P \Psi_f i_q \quad (2)$$

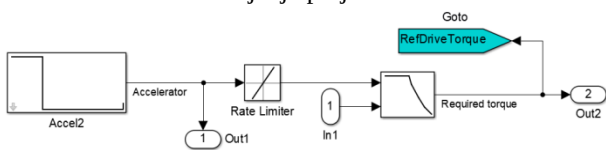
Takođe brzinu u ovom regionu kada su momenta T_e i struja i_q konstantni regulišemo preko napona statora u_s , a taj napon je limitiran onim na izlazu invertora.

$$\omega_e = \frac{\sqrt{u_s^2}}{\sqrt{(L_q i_q)^2 + \Psi_f^2}} \quad (3)$$

Pošto električna vozila imaju maksimalnu brzinu koja ide preko 100km/h, neophodno je omogućiti brzine rotora koje su i nekoliko puta veće od nominalne, to se postiže regulacijom d komponente struje, odakle prema izrazu (4) utiskivanjem negativne vrednosti ove struje drugi član imenioca se smanjuje, a brzina povećava. [3]

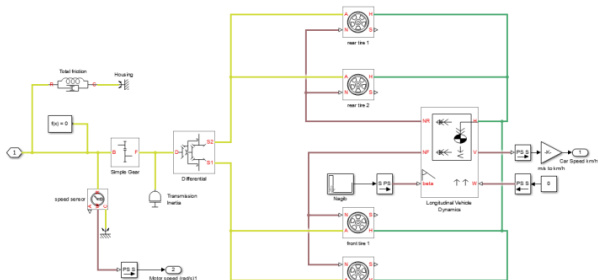
$$\omega_e = \frac{\sqrt{u_s^2}}{\sqrt{(L_q i_q)^2 + (L_d i_d + \Psi_f)^2}} \quad (4)$$

Podsistem bloka SPEED REGULATION prikazan na slici 5. kao ulaz ima referencu ubrzanja koja fizički predstavlja pedal „gasa“ koja na osnovu look up tabele i brzine iz povratne sprege zadaje dozvoljeni moment i na taj način ostvaruje željenu brzinu, a pri tom pazi na pad momenta u oblasti slabljenja polja.



Slika 5. Podsistem bloka SPEED REGULATION

Na izlazu se referentni moment dovodi na ulaz bloka speed controller slika 4. koji radi u režimu kontrole momenta i opet na osnovu look up tabele zadaje referencu fluksa koji se vodi na ulaz bloka vektorske kontrole.

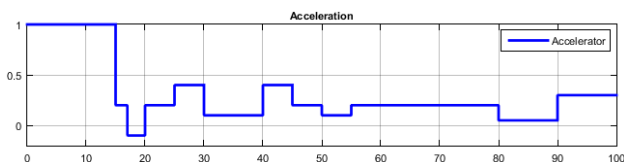


Slika 6. Podsistem BEV LOAD bloka

Blok BEV LOAD prikazan na slici 6. modeluje mehanički deo vozila u kome je moguće uneti parametre vezane za masu vozila, radijus pneumatika, koeficijente aerodinamike, trenja i prenosnog odnosa transmisije. Takođe upotrebom stair bloka moguće je zadavati promenu u nagibu puta u određenim vremenskim trenucima.

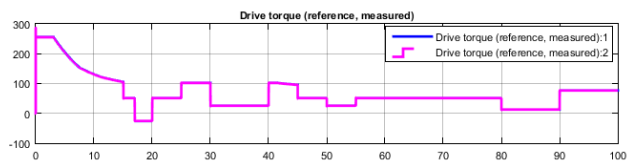
4.1. Simulacija vožnje na delu trase Subotica-Bar

Na sledećim slikama su prikazani rezultati simulacije vožnje na delu razmatrane trase od Subotice do Ljiga. Simulacija je vremenski ograničena na 100s i brzina je regulisana tako da prati modifikovani ciklus vožnje za ovu trasu, čija se brzina kreće oko 120km/h.



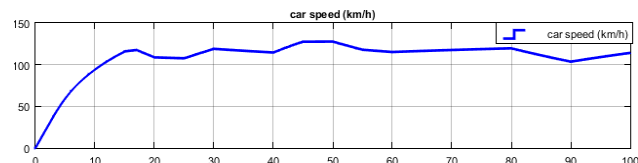
Slika 7. Referenca ubrzanja automobila

Na slici 7. je prikazana referenca ubrzanja automobila pri startu izražena u relativnim jedinicama. Pri samom startu ona iznosi 1 r.j. i ostaje na toj vrednosti dok vozilo ne dostigne brzinu od 120km/h, zatim se referenca menja, jer motor treba da posle zaleta savlada manji moment opterećenja ili se snižava zbog faktora na putu ili preticanja.



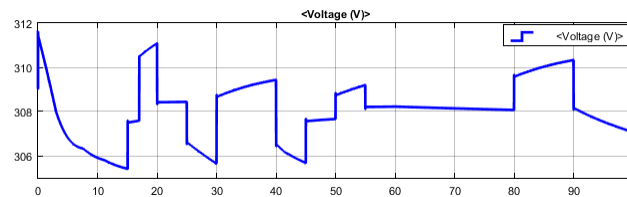
Slika 8. Odziv momenta sinhronog motora

Pri zadavanju željenog ubrzanja, vrši se indirektno zadavanje momenta koji dalje utiče na povećanje brzine, pa su odziv momenta na slici 8. i referenca ubrzanja na slici 7. slični, razlika postoji u prvih 15s, jer rotor brzo distiže svoju nominalnu brzinu pri maksimalnom momentu od 254Nm i zatim sledi povećanje brzine iznad nominalne i izlazak iz oblasti konstantnog momenta. Odakle on opada prema 1/n, kako raste brzina.



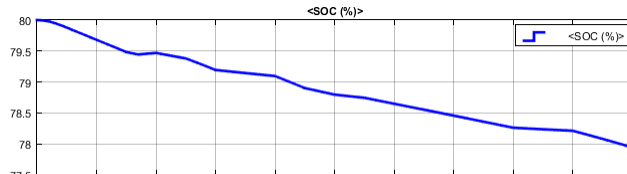
Slika 9. Promena brzine automobila

Na slici 9. se vidi promena brzine vozila tokom perioda od 100s, gde je obuhvaćeno ubrzavanje do 120km/h i zatim održavanje te brzine zbog kretanja po autoputu, uz povećanje pri prestizanju drugih vozila i kočenju zbog nailaska na radove ili krivine.



Slika 10. Odziv napona baterije

Na slici 10. je prikazana promena napona na bateriji, koja ima pad prilikom ubrzavanja vozila do 15. sekunde, kad motor povlači veliku struju i skok posle tog perioda kada vozilo usporava, jer dolazi do vraćanja energije u bateriju i zbog toga raste napon na njoj.

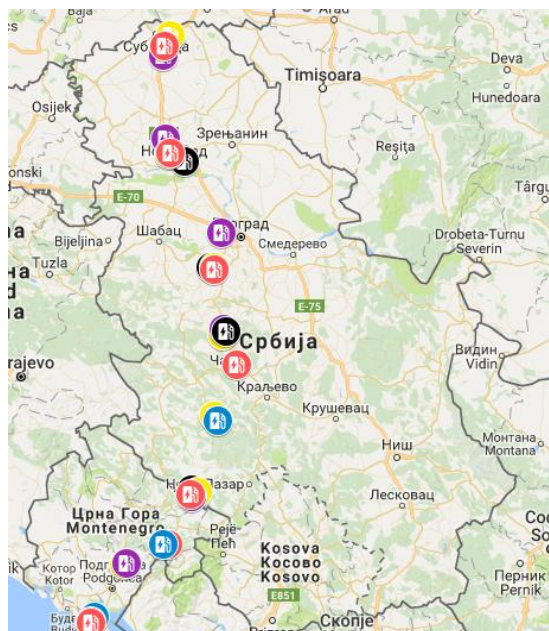


Slika 11. Promena napunjenosti baterije

Cilj realizacije modela električnog vozila je da se dobiju odzivi stanja napunjenosti baterije tokom kretanja određenom brzinom i pri određenom nagibu puta koji simulira stanje na trasi i na osnovu pražnjenja baterije u periodu trajanja simulacije da se izračuna vreme pri kojem će da padne na 15%, a poznavanjem prosečne brzine i pređeni put od početka trase do mesta gde je potrebno postaviti sledeću stanicu. Prema tome na slici 11. vidimo pražnjenje bateriji sa 80% na 77.95% u roku od 100s pa se proračunom utvrđuje da je potrebno sledeću stanicu postaviti na 96. kilometru puta.

4.2. Korigovanje lokacija za postavljanje stanica

Upotrebom simulacionog modela došlo se do sledećih rezultata predstavljenih na slici 12. Na kojoj su objedinjene lokacije za postavljanje stanica dobijene izrazom (1) i one dobijene upotrebom simulacionog modela električnog vozila.



Slika 12. Lokacije za postavljanje punjača dobijene proračunom i simulacionim modelom

Crnim, crvenim i plavim simbolom su označene lokacije dobijene simulacijom, dok su žutim i ljubičastim označene one dobijene izrazom (1). Prema slici 12. se vidi da rastojanje između lokacija dobijenih na osnovu rezultata simulacije varira uglavnom od nagiba puta tj od stanja opterećenosti električnog motora. Pa tako u predelu zapadne Srbije gde se projektovane brzine autoputa kreću od 100km/h do 120km/h i gde je uspon značajan, rastojanje stanica znatno skraćuje za smer ka graničnom prelazu, dok između graničnog prelaza i Bara, gde zbog deonica sa negativnim gradijentom dolazi do malog opterećenja i čestog regenerativnog kočenja rastojanje između punjača značajno povećava.

5. TEHNO-EKONOMSKA ANALIZA

Na osnovu predhodnih rezultata potrebno je postaviti 21 punjač za brzo punjenje i 4 za sporije. Izabrani su modeli ABB Terra 53 CJG i 23 CJG sa cenom od 40.760€ i 33.600€, respektivno. Ukupna investicija bi iznosila 967.709 €. Da bi se jedan punjač isplatio u roku od 4 godine uz kamatnu stopu od 6% potrebno je prema izrazu (5) da cena iznajmljivanja punjača bude 0.065€/kWh ili za slabiji 0.136€/kWh.

$$m = \frac{c \cdot i}{t_d \cdot p_i \cdot 365} \cdot \frac{(1+i)^{t_a}}{(1+i)^{t_a} - 1} \quad (5)$$

Gde je:

t_d - dnevni angažman

c - rata po godini za otplatu uložених sredstava

i - kamata na kredit

t_a - vreme otplate na kredit

p_a - prihod od iznajmljivanja punjača

p_i - snaga punjača

m - cena električne energije

5. ZAKLJUČAK

Jedan od problema vezanih za brže prihvatanje ove nove/stare tehnologije transporta se svodi na to da potencijalni kupci električnih automobila odustaju od njih, uglavnom jer ne postoji dovoljno razvijena mreža za punjenje, a sa druge strane država ne želi da ulaže u infrastrukturu, jer ne postoji dovoljno veliki broj električnih vozila na putu, kako bi investicija bila opravdana. Odavde se donosi zaključak da je ipak država ta koja mora da preduzme prve korake, jer ona takođe ostvaruje određene benefite, uklanjanjem konvencionalnih vozila sa puteva.

Primer gde je država napravila prvi korak ima dosta, pa tako postoje programi gde države daju podsticaje pri kupovini električnih vozila, dok neke i ne tako bogate poput Estonije već imaju realizovan projekat mreže brzih DC punjača na prometnim deonicama u čitavoj zemlji.

6. LITERATURA

- [1] <https://www.theguardian.com/environment/2017/jun/07/electric-cars-sales-2-million-worldwide-global-sales>
- [2] Antonio Colmenar-Santos, Carlos de Palacio, David Borge-Diez and Oscar Monzón-Alejandro, Planning Minimum Interurban Fast Charging Infrastructure for Electric Vehicles: Methodology and Application to Spain, Madrid, 2013
- [3] Xiangyu Du, Jialei Hu, Xuguang Li, Simulation of Flux-weakening Control of PMSM for Electrical Vehicle, International Conference on Electrical Machines and Systems (ICEMS), School of Electrical Information and Electrical Engineering, Shanghai Jiao Tong University, China, 2014

Kratka biografija:



Radisav Ristić rođen je u Novom Sadu 1990. god. Fakultet tehničkih nauka, studentski program Energetika, elektronika i telekomunikacije upisao je školske 2009/2010. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva - Energetska elektronika i električne mašine odbranio je 2017.god.



Vladimir Katić je rođen u Novom Sadu. Doktorirao je na Univerzitetu u Beogradu 1991. Godine . Od 2002. Godine je redovni profesor Univerziteta u Novom Sadu. Trenutno je i prodekan Fakulteta tehničkih nauka i šef katedre za energetsku elektroniku i pretvarače. Oblasti interesovanja su energetska elektronika, obnovljivi izvori električne energije, električna vozila i kvalitet električne energije.

RAZVOJ MOBILNE IGRE ZA REHABILITACIJU RUČNOG ZGLOBA**DEVELOPMENT OF A MOBILE GAME FOR THE REHABILITATION OF THE WRIST**

Aleksandra Mitrović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Rad se bavi istraživanjem upotrebe video igara u fizikalnoj rehabilitaciji sa koncentracijom na oporavak ručnog zgloba. Razvijena je jednostavna igra za Android platformu koja na zabavan način podstiče rad vežbi potrebnih za oporavak. Dostupna su tri režima igranja i podesivi parametri koji omogućavaju prilagođavanje specifičnim potrebama.

Abstract – The paper deals with the research of the use of video games in physical rehabilitation with a concentration on the recovery of the wrist. A simple game for the Android platform has been developed, which in a fun way stimulates the work of exercises needed for recovery. Three game modes are available and there are also adjustable parameters that allow customization to meet specific needs.

Ključne reči: Ručni zglob, rehabilitacija, Android, akcelerometar

1. UVOD

Industrija video igara donosi izuzetno veliki profit. Iako joj je glavna svrha zabava, svojim razvojem podstakla je i intenzivan razvoj tehnologije, koja je potom, veoma brzo našla primenu i u drugim oblastima kao što je fizikalna terapija. Oporavak od povreda uglavnom prati rad vežbi kako bi se ojačali mišići i tetive i vratila im se pokretljivost i snaga kakva je bila pre povrede. Vežbe često nisu bolne ni teške, ali zahtevaju redovnost i veliki broj ponavljanja. Ne samo deci, nego i odraslim osobama, ovo vrlo brzo postaje zamorno i dosadno i od vežbi se odustaje iako nije postignut potpun oporavak. Igra, sama po sebi, unosi zabavu u ovaj proces, a kada se uvedu atraktivne tehnologije, vežbe prestaju da se tretiraju kao vežbe i motivacija za njihovim radom je u znatnom porastu. Kontroleri igračkih konzola (*Nintendo Wii, Microsoft Kinect, Sony's Move*) svojim senzorima za praćenje pokreta mogu da pokriju značajan deo vežbi fizikalne terapije. Iako igračke konzole nude niz igara koje su veoma imerzivne i koje svojom tehnologijom postižu visok stepen kontrole pokreta, činjenica je da su ovi uređaji skuplji i samim tim nisu sastavni deo svakog domaćinstva. Isto tako, nedovoljna portabilnost ovih uređaja podrazumeva da pacijent izdvaja posebno vreme namenjeno vežbanju na određenom mestu, što takođe može predstavljati prepreku redovnom radu i uspešnom oporavku. Tako se javlja potreba za prenosivim rešenjem u vidu igre na mobilnom telefonu koja upotrebom akcelerometra angažuje ručni zglob.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragan Ivetić, red.prof.

U narednom poglavlju biće reči o ozbiljnim video igrama i njihovoj upotrebi u fizikalnoj rehabilitaciji, a u trećem poglavlju će se opisati scenario upotrebe video igre koja je tema rada. Potom sledi zaključak i spisak korišćene literature.

2. OZBILJNE VIDEO IGRE U REHABILITACIJI

Razvijena je posebna vrsta video igara, ozbiljne igre, koje koriste prednosti koje pružaju video igre u medicinske ili edukativne svrhe.

Glavni problem kod fizikalne terapije jeste motivacija pacijenta. Vežbanje ili učenje motoričkih veština zahteva puno ponavljanja istih pokreta. Kada se ovaj proces razvuče na više sesija, kojih takođe mora biti mnogo, jasno je da se ljudi demotivisu i izgube volju, pokreti postanu dosadni, zbog tolike monotonosti ljudi prestanu da se angažuju, odnosno ne izvode potrebne vežbe savesno i pažljivo. Ovo veoma utiče na krajnji rezultat terapije, jer je jedini način da ista bude u potpunosti uspešna, da se dosledno sprovede do kraja, bez preskakivanja, skraćivanja sesija ili nepravilnog izvršavanja. Upravo se u ozbiljnim igrama prepoznaje potencijal korišćenja igara za prevazilaženje problema motivacije. Integrisanje igranja u terapiju povećava vreme provedeno u obavljanju terapije tako što povećava verovatnoću početka nove sesije terapije, vreme potrošeno na terapiju, odlaganje momenta prestanka sa sesijom terapije i šansu ponovnog vraćanja terapiji [1].

Sledeći prepoznat problem, koji fizioterapija korišćenjem ozbiljnih igara može da nadomesti, jeste dostupnost. Ukoliko se omogući terapija i u kućnim uslovima, a ne samo u posebno opremljenim centrima, povećava se broj pacijenata koji će zaista imati mogućnost da joj se posvete.

Dostupnost se ne odnosi samo na pitanje vremenske i prostorne udaljenosti od bolničkog centra koji pruža fizioterapiju. Poseban problem predstavlja dostupnost i broj lekara i vreme kojim oni raspolažu i koje mogu da posvete jednom pacijentu. Ukoliko se terapija odvija tradicionalno, jedan terapeut može da se posveti samo jednom pacijentu u jednom trenutku, a veoma je česta pojava da nema dovoljno resursa za ovakav pristup. Kućna fizikalna terapija rešava i ovaj problem.

Predlagana su mnoga zanimljiva rešenja, koja sakrivaju pravi cilj iza zabavnog i za korisnika primamljivog sadržaja. Kada ne postoji igra za određeni problem, ukoliko se problem dovoljno detaljno istraži, uspešno se može doći do specifičnog rešenja koje će omogućiti prevazilaženje datog problema. Osnovna ideja je da se pokreti koji se koriste za vreme vežbanja iskoriste za savladavanje određenog zadatka u igri. U daljem delu

teksta biće ilustrovano samo par primera kreativne primene tehnologija virtuelne i augmentovane realnosti, kao i tehnologija kućnih igračkih konzola, u ozbiljnim igrama u terapeutske svrhe.

[2] se skoncentrisao na problem povećane pojave cerebralne paralize kod odraslih. Kako ovi bolesnici mogu imati i blago kognitivno oštećenje, postojeće igre su se pokazale kao nedovoljno dobre kada je u pitanju povratna informacija. Kao moguće rešenje ovog problema predložena je „ogledalo“ povratna informacija. Na ekranu su prikazani objekti do kojih korisnik mora da dođe i izvrši neku akciju, a da bi to učinio potrebno je da načini određene pokrete tela. Ekran zapravo funkcioniše kao ogledalo jer korisnik može i sebe da vidi na njemu i na taj način sazna svoj trenutni položaj.

[3] se bavi problemom oporavka od moždanog udara i razvija rešenje zasnovano na augmentovanoj stvarnosti. Jedan od razloga i motiva je činjenica da samo 37% kanadskih pacijenata koji su pretrpeli moždani udar ima pristup posebno opremljenim centrima predviđenim za oporavak. Ovaj projekat predlaže postavljanje vibratkih aktuatora u opipljive predmete. Na taj način se pacijentima omogućava da osele kontakt sa virtuelnim objektima. Vibracija kao povratna informacija obaveštava pacijenta o uspehu izvedenih akcija.

Gotovo svi današnji smart telefoni poseduju više različitih senzora (senzori pokreta, osvetljenja, pozicije, itd.). Jedan od njih je akcelerometar - senzor za merenje ubrzanja. Kako je ubrzanje vektorska veličina, njegovim merenjem može se izračunati ugao neke zamišljene ose telefona u odnosu na zamišljenu referentnu vertikalu. Detekcija okretanja, drmanja svodi se na merenje ubrzanja. Automatski prelazak iz *portrait* u *landscape* prikaz na ekranu omogućen je upotrebom akcelerometra [4, 5, 6]. *Unity engine* omogućava lako korišćenje akcelerometra telefona ili tableta. Akcelerometru se pristupa *Vector3* svojstvom *acceleration* koje je deo standardne *Input* klase.

Ukoliko se traži igra za oporavak ručnog zgloba u prodavnici igara mobilnih telefona, pronalaze se samo priručnici i uputstva za vežbanje. Međutim, veliki broj je igara koje omogućavaju upravljanje tiltovanjem, što angažuje ručni zglob, pa su prepoznate kao potencijalna pomoć terapiji. Zabavne i odlične za razbibrigu, a nisu ni skupe ili su čak besplatne. Uglavnom su teme lavirint ili trke. Trkačke verzije ovih igara ne izazivaju ekstremne položaje zgloba, jer je putanja najčešće samo prava i zahteva mala, kontrolisana pomeranja. Lavirint igre, s druge strane, mogu da izazovu krajnje položaje zgloba, a neke i dozvoljavaju promenu brzine, što može da oteža ili olakša igru u skladu sa trenutnim stanjem zgloba. Igra koja je predmet rada je po načinu upravljanja slična njima, samo, kako je nastala sa ciljem pomoći rehabilitaciji, ima podesive parametre, ali i režime koji mogu da izoluju određenu vrstu vežbi, što ove igre ne mogu.

Scena ove igre je veoma jednostavna i sastoji se od platforme, koja je u svemiru, okružena zvezdama. Na platformi se nalazi svemirski brod u obliku sfere koji predstavlja igrača. Cilj igre je skupiti što više zvezda, svaka zvezda vredi deset poena. Opasnosti su pad sa platforme ili sudar sa asteroidima koji se kreću po sceni. Dinamičnost zapravo i unose asteroidi koji prate igrača, kao i čestice koje pokupljene zvezde ostavljaju za sobom, čime

se smanjuje vidljivost u kratkom vremenskom intervalu i unosi se opšti kaos, koji sa sobom povlači adrenalin i uzbuđenje.

3. SCENARIO UPOTREBE

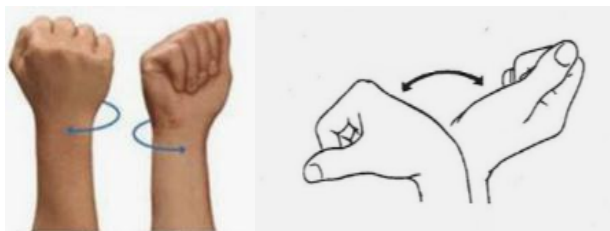
Na ulasku u igru vidljiv je početni ekran na kome se nalazi prikaz zapamćenog najvećeg rezultata i rezultata koji je osvojen u poslednjem igranju. Nakon toga ponuđena su četiri dugmeta od kojih prvo vodi do podešavanja opcija (koje mogu biti namenjene lekaru), a preostala tri služe za odabir nivoa, odnosno režima.

Opcije i parametri koje može da unese sam igrač, ali i fizioterapeut, uglavnom se odnose na podešavanje težine. Ponuđeni promenljivi parametri su koliko puta dnevno igra treba da se igra, što će rezultirati tolikim brojem notifikacija pomoću kojih igra podseća korisnika da odradi svoje vežbe. Sledeći parametar je vreme u minutima koje se smatra optimalnim za jednu sesiju igranja. Sledeća dva parametra odnose se na brzinu igrača i brzinu neprijatelja, odnosno asteroida. Brzina asteroida direktno utiče na težinu igranja, jer oni prate igrača i igrač sam kad ovlada igrom i dobije potrebu da je učini zanimljivijom, može da poveća ovu vrednost. Parametar za brzinu igrača se koristi u kontroli igrača i množi se sa ulazom akcelerometra. On može uticati na željene vežbe, jer ukoliko je ova vrednost veća, biće teže preciznije kontrolisati igrača i najverovatnije će biti potrebni izuzetno nagli pokreti za ostajanje u životu. Ovako će se ručni zglob dovoditi do krajnjih položaja.

Ponuđena su tri režima igre koja su dizajnirana tako da pokriju različite vežbe. Na raspolaganju je samo akcelerometar, tako da se koristi orijentacija telefona za različito opterećenje zgloba. U pitanju su slobodan režim, gore dole režim, i levo desno režim. Orijehtacija ekrana se predefiniše kroz kod tako da se ne može menjati kasnije u zavisnosti od fizičkog položaja uređaja. Ukoliko bi se ostavilo automatsko menjanje orijentacije ekrana u zavisnosti od položaja uređaja, ovu igru bi bilo nemoguće igrati jer bi se ekran tokom igranja stalno okretao zbog upravljanja.

3.1. Slobodni režim

Slobodan režim je najzanimljiviji, nema nametnut pravac kretanja, mogući su pokreti u svim pravcima i svim brzinama. Zvezde se pojavljuju na nasumičnim mestima, asteroidi dolaze iz svih pravaca. Platforma je veće širine pa je tako veći prostor po kome je dozvoljeno kretanje. Ovaj režim pokriva vežbe za pokretljivost ručnog zgloba u svim pravcima, najteži je za ostati dugo u životu, ali zbog toga izaziva različite brzine i time različitu naglost pokreta neophodnu za uspešno igranje. Ovako se izazivaju izuzetno nagli pokreti koji kao reakciju povlače nove nagle pokrete zbog velikih brzina, ali takođe i finiji, precizniji pokreti, kako bi se umirilo ubrzanje i postigla veća kontrola nad upravljanjem. Predefinisana orijentacija uređaja u ovom režimu je horizontalna. Na slici 1 prikazane su vežbe koje ovaj režim obuhvata, a na slici 2 izgled scene u ovom režimu.



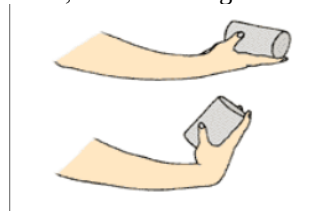
Slika 1: Vežbe obuhvaćene slobodnim režimom



Slika 2: Prikaz scene slobodnog režima

3.2. Gore dole režim

Gore dole režim ima za cilj vežbe ekstenzije i fleksije. Telefon se u ovom slučaju drži uspravno. Ukoliko se položi na celu šaku i čitava sesija igranja se odvija na ovaj način vežbe se izvode mnogo pravilnije. U ovom režimu cilj je postići što veće ubrzanje, jer ove vežbe zahtevaju dolazak do krajnjih položaja šake i ručnog zgloba. Platforma je uža i tako smanjuje prostor za kretanje levo desno, a zvezde i asteroidi se pojavljuju naizmenično na suprotnim stranama, i primoravaju igrača da uvek ide na suprotnu stranu platforme i menja položaj šake i zgloba do krajnje tačke. Na slici 3 prikazana je vežba obuhvaćena ovim režimom, a na slici 4 izgled scene.



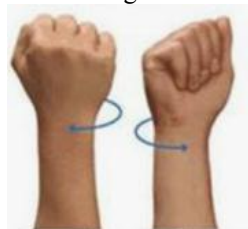
Slika 3: Vežba obuhvaćena gore dole režimom



Slika 4: Prikaz scene gore dole režima

3.3. Levo desno režim

Levo desno režim forsira ograničavanje pokreta po horizontalnoj osi. Mobilni telefon je orijentisan horizontalno. I ovde je slučaj da ukoliko je ubrzanje veće, postižu se ekstremni položaji i njihovo naizmenično smenjivanje. U suštini ideja je ista kao za gore dole režim, odnosno kombinacija zvezda i asteroida i njihovog naizmeničnog pojavljivanja na suprotnim stranama primoravaju korisnika da okreće zglob levo desno, a platforma je sužena kako bi smanjila mogućnost pokreta gore dole. Razlog zašto su ova dva režima razdvojena i orijentacija ekrana je predefinisana jeste izolovanje određenih pokreta i koncentrisanje na samo jednu vrstu vežbi. Na slici 5 prikazana je vežba, a na slici 6 scena ovog režima.



Slika 5: Vežba levo desno režima



Slika 6: Prikaz scene levo desno režima

U svim režimima, ukoliko se koristi teži uređaj poput tableta ili nekog većeg telefona, postiže se efekat rada vežbi sa tegovima.

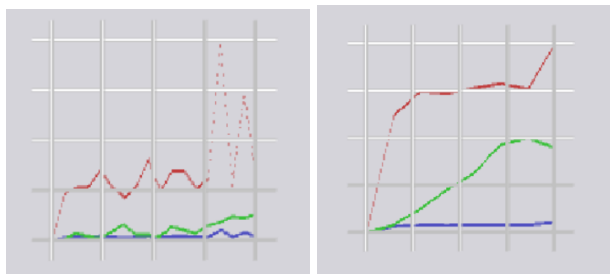
Navođenje pomoću zvezda i asteroida bi moglo napraviti preciznu putanju i unapred definisan set vežbi, ali ovo zahteva sprovođenje preciznog i detaljnog istraživanja, koji set bi bio najpogodniji za određenog pacijenta. Međutim, što je veća ograničenost pokreta, element zabave i igre se umanjuje i korisnik stiče utisak da radi samo redovne monotone vežbe.

3.4. Memorisanje i prezentacija podataka vežbanja

Osim za vežbanje, ova video igra može pomoći i fizioterapeutu u proceni trenutnog stanja pacijenta. Ako se prati ubrzanje po vremenu, može se videti kada je došlo do gubitka života, odnosno kritične situacije za pacijenta. Uzrok gubitka života može biti loša precizna kontrola, što se može pretpostaviti ukoliko je do gubitka došlo pri malom ubrzanju ili nemogućnost pomeranja zgloba u ekstremne položaje, što se može pretpostaviti ukoliko je do gubitka života došlo pri velikom ubrzanju. Naravno, i igrači koji imaju potpuno zdrav ručni zglob mogu izgubiti život ili imati problema sa upravljanjem, ali analiziranjem

igranja pravih pacijenata na dužem periodu mogu se ustanoviti problemi i napredak i shodno tome promeniti parametri.

Nakon gubitka života, na ekranu se privremeno prikazuje dijagram sa brzinama. Potom se on skrinšotuje i čuva u telefonu, a prikaz se nakon tri sekunde vraća na početnu scenu, u kojoj se bira novi režim ukoliko se igranje nastavlja. Izgled dijagrama prikazan je na slici 7.



Slika 7: Grafici brzina i sila

Na slici su data dva primera kako bi se lakše uočila razlika. Zelenom bojom je označena prava brzina svemirskog broda. Plavom bojom je prikazan ulaz akcelerometra, odnosno jačina vektora, a crvenom bojom proizvod brzine (koja je podesiv parametar) i ulaza akcelerometra, odnosno sila koja se dodaje sferi i tako utiče na njeno pomeranje. Ove vrednosti se mere svake sekunde i biće prikazane sve, do gubitka života. Promena u ulazu akcelerometra (plavo) je mnogo uočljivija ukoliko se gleda proizvod sa brzinom (crveno). Slika levo je imala velike oscilacije u brzini i ubrzanju akcelerometra, što znači da je prilikom igranja došlo do naglih pokreta koji su dovodili uređaj, a samim tim i zglob, u ekstremne položaje, na primer skroz naopako u vertikalnom položaju, što je i očekivano ponašanje za gore dole režim igranja. Desni grafik ima pravilniju kretanju, ulaz akcelerometra nema naglih promena, postignuta brzina se ravnomerno povećava i pred kraj smanjuje. Gubitak života u tom slučaju je ili prekasno usporavanje pri približavanju ivici ili zalutali neprimećeni asteroid.

4. ZAKLJUČAK

Istraživanje je pokazalo da se primena virtuelne realnosti i tehnologija industrije video igara u medicini ubrzano razvija, i da doktori u saradnji sa profesionalcima iz ove oblasti konstantno pronalaze nove mogućnosti lečenja koje sa prethodnom tehnologijom nisu bile ostvarive.

Kada se radi o lečenju pacijenata od povreda, konsultovana literatura je pokazala potencijal za ostvarivanjem značajno većeg uspeha prilikom lečenja povreda upravo zato što se pacijentima nudi zabavan sadržaj koji drži pažnju. Zabavna, a ozbiljna igra, dostupna na telefonu u svakom trenutku i na svakom mestu gotovo u potpunosti eliminiše potrebu za svakodnevnim posetama lecarskoj ordinaciji i znatno povećava motivaciju pacijenta za radom vežbi.

Aplikacija o kojoj je bila reč u ovom radu služi za oporavak ručnog zgloba. Za upravljanje se koristi akcelerometar već prisutan u mobilnom telefonu koji zahteva pomeranje ručnog zgloba. Budući razvoj ove aplikacije, koji ovaj rad prepoznaje kao pravac koji bi

dodatno unapredio lečenje i negu i bio koristan ne samo za ovu konkretnu aplikaciju nego za sve aplikacije u ovoj oblasti je mogućnost udaljenog praćenja napretka pacijenta. Integrisanje mogućnosti snimanja sesija u samu aplikaciju i mogućnost direktnog slanja snimaka na onlajn prostor kome lekar ima pristup, bi omogućilo lekaru da proverava svakodnevni ili nedeljni napredak pacijenta bez dolaska u ambulantu. Lekar može da prati intenzitet, brzinu i greške prilikom igranja i ustanovi koje su trenutne slabosti pacijenta. Ne samo da bi smanjio potrebu za stalnim posetama lekaru, ovakav pristup bi omogućio i preciznije ustanovljavanje razvoja povrede – koje nije zasnovano isključivo na opisu tegoba od strane pacijenta.

Kao još jedan pravac budućeg razvoja prepoznaje se uvođenje mogućnosti multiplejerskog igranja. Ovakav vid igranja bi pozitivno uticao na razvoj takmičarskog duha, i podstakli bi se pacijenti da igraju duže kako bi nadmašili protivnika. Na taj način bi se dodatno podigla motivacija za igranjem, što znači da bi periodi vežbanja bili učestaliji i duži i samim tim bi se olakšao i ubrzao oporavak.

5. LITERATURA

- [1] K. Lohse, N. Shirzad, A. Verster, N. Hodges, H. F. Machiel Van der Loos, *Using Design Principles to Enhance Engagement in Physical Therapy*, *Journal of Neurologic Physical Therapy*, 37(4), pp. 166-175, December 2013.
- [2] A. Jaume-i-Capó and A. Samčović, *Interactive Multimedia System Using Serious Game for Users with Motor Disabilities*, *Telfor Journal*, vol. 7, no. 2, pp. 97-102, 2015.
- [3] M. S. Hossain, S. Hardy, A. Alamri, A. Alelaiwi, V. Hardy, C. Wilhelm, *AR-based serious game framework for post-stroke rehabilitation*, *Multimedia Systems*, vol. 22, issue. 6, pp. 659-674, November 2016.
- [4] Dimension Engineering, *A beginner's guide to accelerometers*, <https://www.dimensionengineering.com/info/accelerometers>, Sajt namenjen temama iz oblasti robotike, radio kontrole i elektronike
- [5] Aliasing Games Devblog, *Accelerometer in Unity*, <http://devblog.aliasinggames.com/accelerometer-unity/>, Sajt namenjen blog postovima na teme u oblasti gejmla razvoja
- [6] Developer Android, *Sensors Overview*, https://developer.android.com/guide/topics/sensors/sensors_overview.html, Sajt namenjen developerima za android

Kratka biografija:



Aleksandra Mitrović rođena je u Novom Sadu 1992. god. 2011. godine upisuje Fakultetu tehničkih nauka, oblast Elektrotehnika i računarstvo – Računarstvo i automatika. Završava osnovne akademske studije i upisuje master 2015. godine. Diplomski-master rad odbranila je 2017. god.

APLIKACIJA ZA UPRAVLJANJE AUTONOMNIM VOZILOM ZASNOVANA NA ANDROID I ARDUINO PLATFORMAMA**APPLICATION FOR CONTROLLING AUTONOMOUS VEHICLE DESIGNED ON ANDROID AND ARDUINO PLATFORM**

Dušan Đumić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je prikazano rešenje daljinskog sistema za nadzor i upravljanje autonomnim vozilom zasnovano na Arduino i Android platformama.

Abstract – The paper presents a solution for remote monitoring and control of autonomous vehicle based on Arduino and Android platforms.

Ključne reči: Daljinski nadzor, Daljinsko upravljanje, Android platforma, Arduino platforma, autonomna vozila

1. UVOD

U ovom radu je prikazano rešenje i dat opis aplikacije koja služi za upravljanje autonomnim vozilom pomoću Arduino kontrolera i nadzorno-upravljačkog sistema implementiranog uz oslonac na Android mobilnu platformu. Komunikacija između kontrolera i nadzorno-upravljačke jedinice ostvarena je putem Bluetooth veze.

Prilikom razvoja programske podrške za Android platformu, prirodno okruženje za razvoj programeru predstavlja Java programski jezik, a sve u cilju približavanja granici mogućnosti same fizičke arhitekture [1]. Kako bi omogućili daljinsku kontrolu kretanja vozila i očitavanje izmerenih podataka kreirana je Android aplikacija. Dat je pregled nekih ključnih funkcija na koje treba obratiti pažnju, objašnjen je postupak pokretanja i korišćenja aplikacije. Objašnjen je algoritam rada same aplikacije i karakteristike. Prikazana je primena aplikacije.

Primena ove aplikacije je raznovrsna, ona se može primeniti u svim sferama života. Veliku prednost ima na teško dostupnim mestima, s obzirom da su sve komponente sistema montirane na samo vozilo, a da se upravljanje vrši daljinskim putem i to pomoću lako dostupnog hardvera. Samo vozilo je lako je upravljivo, stabilno, bezbedno i lako pokretno. Prednost Bluetooth-a je niska cena i sigurnost, a nedostatak je mali domet signala. Navešćemo nekoliko primera primene:

- Industrijska primena,
- Vojna primena,
- Sistemi za preživljavanje (spasilačke službe),
- Sistem za nadzor i obilazak tesko pristupnih terena,
- Upravljanje invalidskim kolicima,
- Razvoj elektro-hibridnih vozila,
- Kucna automatizacija malim izmenama na platformi.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Rapaić, red.prof.

2. KRATAK OPIS RAZVOJNIH PLATFORMI**2.1. Android platforma**

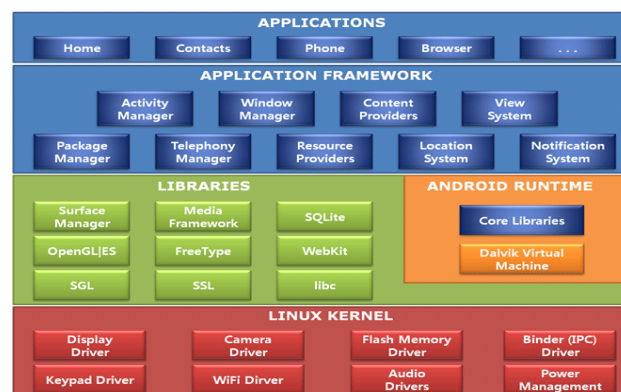
U ovom poglavlju dat je opis operativnog sistema Android platforme i osnovnih modula koji ulaze u sklop programske podrške ovog rešenja. Android predstavlja programsku podršku namenjenu prvenstveno za mobilne uređaje, koji u sebe uključuje operativni sistem, sprežni sloj za povezivanje Java i C programskog jezika i pojedine aplikacije koje su dostupne korisnicima. Sistem je zasnovan na Linux jezgu.[2]

Najvažniji gradivni blokovi svake Android aplikacije su aktivnost (eng. Activity) i servis (eng. Service). Aktivnost je blok u okviru aplikacije koji omogućuje interakciju sa krajnjim korisnikom, gde korisnik zadaje odgovarajuće akcije putem podržanih periferija (miš, tastatura, dodir, itd.). Svaka aktivnost sadrži prozor koji se koristi za iscertavanje korisničkog interfejsa. Prozor obično zauzima ceo ekran, ali postoji mogućnost da on bude i manji ili da bude postavljen u ravni iznad drugih prozora (eng. Z-ordering).

Aktivnost se definiše nasleđivanjem klase Activity, koja je definisana u okviru Android razvojnog okruženja i implementiranjem određenih povratnih metoda koje definišu životni ciklus svake aktivnosti.[3]

Arhitektura Android platforme (Slika 1.) predstavlja standardnu arhitekturu Linux operativnog sistema gde su segmenti razdvojeni po nivoima na kojima rade. Svaki nivo predstavlja grupu od nekoliko programskih komponenti.

Najniži sloj programske podrške je zasnovan na Linux jezgu verzije 2.6. Linux jezgro je iskorišćeno kao sloj koji omogućava apstrakciju fizičke arhitekture. Neka od proširenja su *alarm*, *ashmem*, *binder*, *logger*, *low memory killer*. [4]



Slika 1. Arhitektura Android platforme

2.2. Arduino platforma

Arduino je mikrokontrolerska platforma razvijena u Italiji 2005. godine na Institutu za interaktivni dizajn u Ivreji.

Napravljen je od open-source hardverskih komponenti koje su spojene na 8-bitni Atmel AVR mikrokontroler ili 32-bitni ARM procesor. Trenutno postoji više od petnaest različitih modela, a svi se mogu proširivati sa tzv. shield-ovima, koji imaju različite funkcije. Arduino dolazi u različitim veličinama. Trenutno aktuelne verzije ove popularne ploče su Arduino Uno R3, Arduino Due, Arduino Leonardo, Arduino MEGA 2560 R3, Arduino Nano, Pro Mini...

Na ulaze ovog kontrolera moguće je dovesti tastere, prekidače, gotove tastature, različite senzore (temperature, pritiska, protoka, IC senzore...), dok izlaze možete povezati na širok spektar izvršnih uređaja – LE dioda, sijalica, zujalica, motora, ekrana za prikaz podataka...

3. OPIS REŠENJA NADZORNO-UPRAVLJAČKE APLIKACIJE

Prilikom pokretanja aplikacije „Android Arduino Robot“ dobijamo početni ekran (Slika 2):

Aplikacija se sastoji od šest tastera (*Button*):

- *Akcelometar* – ulazimo u upravljanje vozilom preko akcelometra ugrađenog u samom telefonu,
- *Kontrola preko tastera* – ulazimo u upravljanje vozilom preko tastera,
- *Unos preko terminala* – unos željenih komandi preko tastature telefona,
- *Kompas* – meni za pozicioniranje vozila preko ugrađenog kompasa u telefonu,
- *Kontrola glasom* – zadavanje glasovnih komandi za pomeranje vozila,
- *Info* – prikaz informacija o samim funkcionalnostima aplikacije.

3.1. Akcelometar

Prozor *Akcelometar* se sastoji od više celina koje nam prikazuju informacije o položaju telefona, brzini levog i desnog točka. Unutar *SurfaceView*-a nalazi se linija koja se pomera u zavisnosti od položaja i nagiba telefona. Osvežavanje se vrši konstantno i linija se pomera definisanim načinom (Slika 3.). Slanje podataka na vozilo počinje tek kada pritisnemo dugme (*Button*) *Start*, koordinate X i Y koristimo u proračunu u kojoj se oblasti nalaze i samim tim šaljemo komandu na vozilo.

Dugme *Start* prelazi u *Stop* i kada želimo da prekinemo komunikaciju i slanje podataka pritisnemo *Stop* dugme. U suprotnom da nemamo ovaj taster vozilo bih krenulo pri samom ulasku u ovaj prozor i korisnik ne bih mogao da ga kontroliše i zaustavi.

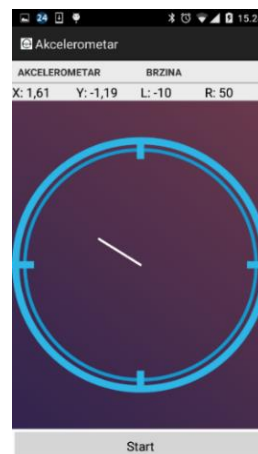
3.2. Kontrola preko tastera

Korisniku se pruža mogućnost upravljanja preko tastera, radi preciznosti upravljanja tasteri moraju adekvatno i brzo da reaguju. Vizuelno aplikacija sadrži 14 tastera i ispis rezultata u tri reda. Tasteri *Connection* i *Disconnect* služe za povezivanje i prekidanje veze sa *Bluetooth* modulom na vozilu. Tasteri za pomeranje i zaustavljanje

vozila su realizovani preko *ImageView*(prikaz slike preko tastera). Rade na principu kada se dodirnu i drže pritisnuti, vozilo se pokreće. Čim podignemo prst sa tastera automatski se vrši zaustavljanje. Taster *STOP* resetuje sve na početne vrednosti i zaustavlja kretanje. Tasteri za brzinu 25%, 50% i 100% služe za postavljanje brzine vozila na najsporiju četvrtinu brzine, pola brzine i na maksimalnu brzinu.

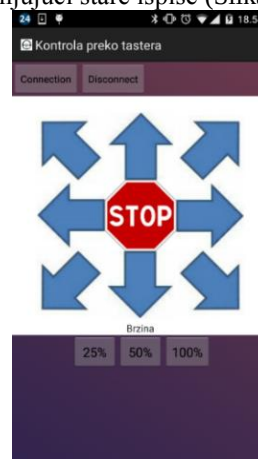


Slika 2. Izgled početnog ekrana



Slika 3. Izgled prozora Akcelometar

Ispis rezultata koji se nalazi u gornjem desnom uglu, dobijamo kao rezultat interakcije između aplikacije i vozila, a menja se ažuriranjem svakog ispisa od dole prema gore smenjujući stare ispise (Slika 4.).



Slika 4. Izgled prozora Kontrola preko tastera

3.3. Unos preko terminala

Prozor se sastoji od dva ispisa (*TextView*) i jednog unosa (*EditText*). Prvi ispis nam prikazuje vrednost koju smo uneli a drugi ispis je povratna informacija sa vozila da li je komanda uspesna ili je neispravan unos.

Pritiskom na *Unesite komandu* otvara nam se tastatura preko koje vršimo zadavanje komandi. Potvrdu komande vršimo slanjem sa tasterom na tastaturi, ujedno resetuje se polje za unos.

3.4. Kompas

Prozor *Kompas* se sastoji od više celina (Slika 5.) koje nam prikazuju informacije o uglu i položaju strana sveta kao i ispis *Pozicija* (željeni smer kretanja), ispis druge pozicije sa dugmeta *Orijentisi se* (trenutni smer vozila koji ce se promeniti), zatim u centralnom delu kompas koji se rotira, i ispis smera slovima i u stepenima. Na dnu se nalazi ispis podataka koje dobijamo sa vozila kao rezultat. Aplikacija treba da obezbedi da se vozilo postavlja na zadatu poziciju uvek najkraćim putem i samim tim štedi se vreme.



Slika 5. Izgled prozora Kompas

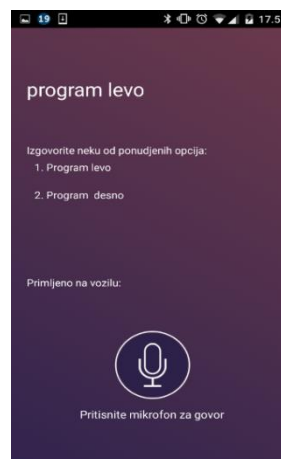
3.5. Kontrola glasom

Prozor se sastoji od više celina od čega je informativni deo na početku, zatim imamo povratnu informaciju sa vozila u vidu ispisa i na kraju taster mikrofona koji pokreće novi prozor za govor.

Korišćenje započinjemo pritiskom na taster mikrofona, nakon čega nam se otvara novi prozor. U novom prozoru aktiviran je mikrofona na telefonu i možemo zadati glasovnu komandu. Obrada podataka se vrši na ranije opisan način, a pri unosu moramo tačno i razgovetno da kažemo komandu. Prepoznavanje reči može da se obavlja i *Offline* kada nismo povezani na internet. Na (sl.6.), možemo videti rezultat posle izgovorene i prepoznate komande, ona nam se prikazuje na vrhu prozora **program levo**.

3.6. Info

Prozor *Info* nam pruža informacije o aplikaciji, broj verzije (ver. 1.0) kao i spisak samih funkcionalnosti o svakoj stavci koju pruža sama aplikacija. Dato je objašnjenje za svih pet stavki, načinu korišćenja, pokretanja i rada u samim aktivnostima određenih programa.

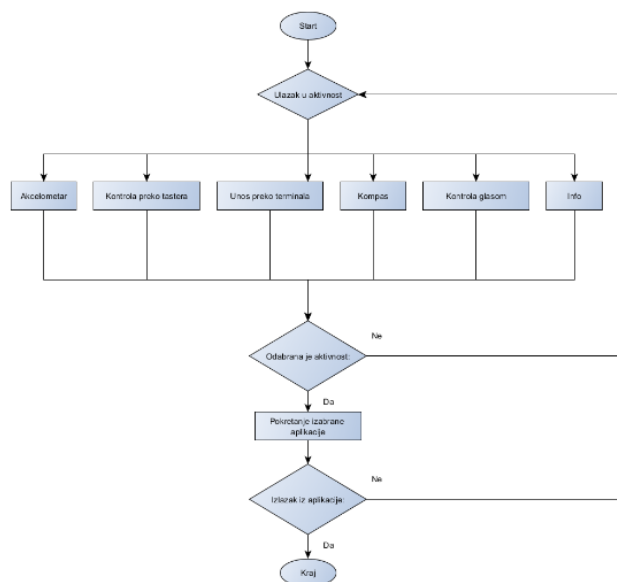


Slika 6. Izgled prozora sa prikazom rezultata

3.7. Algoritam rada aplikacije

Najpopularniji i najrasprostranjeniji je standardni dijagram toka kada se tok prati po pravcu kretanja strelica. U pravougaonike se upisuju kratki opisi operacija i poslova, a u rombove uslovi za grananje.[5]

Na slici 7. je prikazan algoritam rada aplikacije.



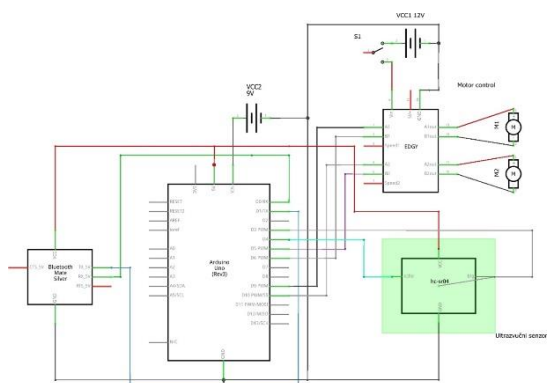
Slika 7. Izgled dijagrama toka početnog prozora

3. VERIFIKACIJI REŠENJA

Komponente koju smo koristili za izradu ovog rada:

- Samsung Galaxy S4 GT-I9505G verzija: 5.1
- Vozilo sa točkovima
- Arduino Uno
- Bluetooth modul HC-05
- Ultrazvučni senzor
- Modul za pogon DC motora
- Baterija: 9V
- Baterija: 1.5V 6 komada
- Kablovi

Šema povezivanje vozila je kreirana u programu Fritzing i prikazana je na slici 8.



Slika 8. Šema povezivanja elemenata

3.1. Telefon Samsung Galaxy S4

S4 I9505 je opremljen fantastičnim 5 inch-nim Super AMOLED displejom sa *Corning Gorilla Glass 3* zaštitom, kamerom od 13 MP i memorijom proširivom do maksimalnih 64GB. Pokretač ovog modela je moćni Quad-core procesor od 1.9 GHz, i funkcionira sa najsvežijim Android v5.1 (*Lollipop*) operativnim sistemom.

3.2. Vozilo sa točkovima

Promena pravca kretanja vozila se vrši simulacijom skretanja kao kod pravog automobila, preko zupčaste letve i spona. Zupčastu letvu dodiruje zupčanik na koji je fiksiran DC motor. Na vozilu imamo dva DC motora, prvi koristimo za skretanje vozila levo/desno, a drugi za pogon napred/nazad.

3.3. Arduino Uno

Napajanje: Arduino Uno ploča se može napajati preko USB veze ili sa spoljnim napajanjem. Izvor energije se automatski izabere. Spoljno napajanje može doći ili iz adaptera *AC-to-DC* (zidni punjač) ili baterije. Adapter može da se poveže na 2.1mm u ploči. Vod iz baterije može se ubaciti u GND i Vin pin zaglavlja POWER konektora. Ploča može raditi na spoljnom napajanju od 6 do 20 volti. Međutim, ako se isporučuje sa manje od 7V, pin 5V može da obezbedi manje od 5V i ploča može postati nestabilna. Ako koristite više od 12V, regulator napona može se pregrejeti i oštetiti ploču. Preporučeni opseg je 7 do 12 volti.

3.4. Bluetooth modul

Komunikacija između vozila i Android aplikacije je ostvarena Bluetooth bežičnom serijskom konekcijom, a zasniva se na stalnom „oslušivanju“ serijskog RX pina na mikrokontroleru.

Komunikacija u drugom smeru, od mikrokontrolera ka Bluetooth uređaju je ostvarena preko TX pina. Svi podaci, koji su primljeni na Android uređaju, prikazani su na samom interfejsu aplikacije a mogu biti prosleđeni na bilo koji savremeni uređaj (Telefon, Smart TV, Kompjuter,...)

3.5. Ultrazvučni senzor

Ultrazvučni senzor omogućava beskontaktno merenje razdaljine u opsegu od 2cm do 400cm uz veliku preciznost od oko 3mm zavisno od uslova u kojima se senzor nalazi. Napaja se naponom od 5V DC a potrošnja struje iznosi manje od 2mA. Ultrazvučni senzor se povezuje na 2 digitalna pina Arduinoa (trigger i echo) i na kontrolisani napon 5V.

3.6. Modul za pogon DC motora

Integralno kolo L298N smešteno je u 15-to pinsko MULTIWATT kućište i prvenstveno je namenjeno za pobudu koračnih motora. Sastoji se iz 4 izlazna kanala koji čine 2 puna tranzistorska mosta. Može da pogoni jedan step motor bipolarni ili dva DC motora. Za napajanje ima ugrađeni regulator napona 78M05.

4. ZAKLJUČAK

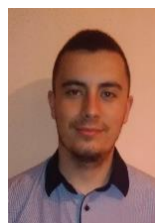
U ovom radu je prikazano rešenje i dat opis aplikacije koja služi za upravljanje vozilom. Realizovano je upravljanje na više principa i korisniku je pruženo da odabere njemu šta odgovara. Kompletan rad je rađen na Android platformi a testiranje je vršeno na Samsung Galaxy S4 verzija: 5.1 Lollipop.

Objašnjeni su osnovni teorijski principi Arduino i Android platforme. Srž realizovanog vozila predstavlja Arduino mikrokontroler, dok senzori koji se nalaze na vozilu služe za merenje parametara okoline. Kako bi omogućili kontrolu kretanja vozila i očitavanje izmerenih podataka kreirana je Android aplikacija. Za detekciju rastojanja od prepreke, promenu brzine kretanja i pomoć kod zaobilaženja iste iskorišćen je ultrazvučni senzor rastojanja od prepreke. Sve opisane funkcionalnosti su potvrđene eksperimentima u realnim uslovima, a još jedna prednost rešenja prikazanog u ovom radu je njegova modularnost, tj. mogućnost dodavanja novih funkcionalnosti i novih senzora.

5. LITERATURA

- [1] V. Kovačević, „Logičko projektovanje računarskih sistema I –projektovanje digitalnih sistema“, Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka, 2001
- [2] D. Bovet, M. Cesati, „Understanding the Linux Kernel“, 2005
- [3] Android Developers, official Google Android documentation, www.developers.google.com, 2014
- [4] R. Meier, „Professional Android 2 Application Development“, Wrox, 2010
- [5] F. B. Gilbreth, L. M. Gilbreth, „Process Charts“, American Society of Mechanical Engineers, 1921

Kratka biografija:



Dušan Đumić rođen je u Vrbasu 1989. god. Osnovne studije upisao je 2008. god. na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva. Master studije na Fakultetu tehničkih nauka iz oblasti Automatika i upravljanje sistemima upisao je 2014.god.

JEDNO REŠENJE PRIMENE VEŠTAČKIH NEURONSKIH MREŽA U SISTEMIMA INTERNET STVARI**A SOLUTION FOR THE USAGE OF ARTIFICIAL NEURAL NETWORKS IN THE SYSTEMS OF INTERNET OF THINGS**

Bojan Baltić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Ovaj rad razmatra jedan primer primene veštačkih neuronskih mreža u sistemima interneta stvari. Rad sadrži kratak pregled teorijskih osnova veštačkih neuronskih mreža u meri potrebnoj za razumevanje zadatka prepoznavanja rukom napisanih brojeva. Takođe, u ovom radu je opisana arhitektura sistema interneta stvari u okviru koje bi se ovakav algoritam mogao primeniti..

Abstract – This paper presents an example of application of artificial neural networks in the context of Internet of Things. It contains basic theory of artificial neural network, in the extent which is necessary for understanding the recognition of hand written numbers. This paper also describes an architecture of the Internet of Things system in which such an algorithm could be applied.

Ključne reči: Zbornik FTN, Studentski radovi, Uputstvo

1. UVOD

Internet stvari (engl. Internet of Things), je mreža uređaja povezanih preko interneta koji su u mogućnosti da razmenjuju podatke. Svaki uređaj, odnosno svaka stvar u „internet stvarima“ je jedinstveno identifikovana na mreži. Namena tih uređaja može biti izuzetno raznolika, odnosno može se reći da se IoT može iskoristiti kao rešenje za široku lepezu problema iz gotovo svih oblasti savremenog života i privrede. Uređaj može biti senzor koji detektuje medicinske signale, poput otkucaja srca, kao i senzor koji proverava pritisak u automobilskim gumama. Takođe, praćenje kretanja divljih životinja je moguće uz pomoć senzora koji bi se prethodno stavio u njih. U suštini svaki uređaj koji ima mogućnost komunikacije preko interneta može biti stvar u internet stvarima. Na osnovu navedenog, jasno je da je broj uređaja u sistemima internet stvari izuzetno velik.

Veliki broj uređaja generišu i velik broj podataka, a to je jedan od najvećih problema sistema internet stvari. U ovom radu će biti prikazan mehanizam smanjivanja broja podataka uz pomoć veštačkih neuronskih mreža. Ideja je da se rukom napisan jednocifren broj obradi uz pomoć veštačkih neuronskih mreža, detektuje koji je broj u pitanju, a da se zatim taj broj sačuva u bazu podataka.

Da bi se prikazala crno-bela slika jednocifrenog broja veličine 28x28 piksela, potrebno je 28*28, odnosno 784 bajtova.

Sa druge strane, jednocifren broj se može prikazati sa 4 bita, što znači da bi broj podataka potrebnih za čuvanje rukom napisanog broja bio manji 1568 puta. Dodatno, u ovom radu će biti predstavljene osnovne razlike između relacionih i nerelacionih baza podataka za upotrebu u sistemima internet stvari.

2. ARHITEKTURA U SISTEMIMA INTERNET STVARI

Jedna od najvažnijih komponenti, a možda i najvažnija komponenta svakog inženjerskog problema je arhitektura. Ulaganje vremena u dizajniranje arhitekture se uvek isplati ako se na problem posmatra dugoročno. U slučaju dobro dizajnirane arhitekture svaki pojedinačni deo sistema ne zavisi od bilo kog drugog dela sistema.

Ako se posmatra primer iz ovog rada, odnosno problem prepoznavanja rukom napisanih jednocifrenih brojeva uz pomoć veštačkih neuronskih mreža, jasno je da deo rešenja zadužen za prepoznavanje brojeva treba da bude nezavisan deo rešenja. Sistemi internet stvari nisu izuzetak od ovog pravila. Naprotiv, sistemi internet stvari su kompleksni sistemi u kojima je razdvajanje sistema na više manjih komponenti gotovo obavezan zadatak.

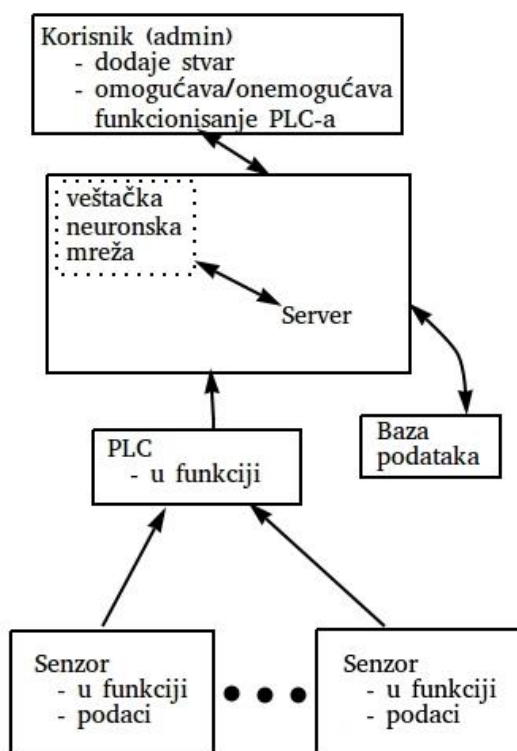
Prednost razdvajanja sistema na više manjih komponenti gde nijedna komponenta ne zavisi od neke druge ima dve velike prednosti. Prva prednost je ta što je mnogostruko jednostavnije posmatrati i unapređivati manji problem. Druga prednost je ta što je moguće promeniti celu komponentu, a da se ne naruši funkcionisanje celog sistema.

Na primer, prepoznavanje rukom napisanih brojeva se nakon nekog vremena može vršiti uz pomoć neke druge tehnologije umesto veštačkih neuronskih mreža. Postoji mogućnost da će se prepoznavanje brojeva u budućnosti vršiti uz pomoć tehnologije koja trenutno ni ne postoji. Zbog toga je važno omogućiti nezavisnost svake komponente unutar sistema internet stvari.

Na slici 1 je prikazan predložen izgled jednostavne arhitekture sistema internet stvari koja je iskorišćena u ovom radu, a u nastavku će biti objašnjene pojedinačne komponente predložene arhitekture.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Rapaić, red.prof.



Slika 1. Predložen izgled jednostavne arhitekture sistema internet stvari

Ako se posmatraju senzori u sistemima internet stvari, potrebno je dizajnirati sistem na takav način da dodavanje novih senzora ne utiče na rad celog sistema. Postoji mogućnost da dođe do kvara senzora, a da ne postoji način da se nađe isti takav senzor. Kada se to desi potrebno je na jednostavan način omogućiti rad novog senzora.

Senzori u ovom radu su zamišljeni kao kamere koje su u mogućnosti da šalju svoje podatke u sledeći sloj u arhitekturi. Podatak je slika veličine 28x28 piksela. Svaki piksel može da ima vrednosti od 0 do 255, gde 0 predstavlja totalno crn piksel, a 255 u potpunosti beo piksel.

Sledeći sloj u arhitekturi može biti sloj programabilno logičkih kontrolera (engl. Programmable logic controller). On je u mogućnosti da komunicira sa jednim ili više senzora koji mu šalju podatke. Može biti više programabilno logičkih kontrolera i u svakom momentu se bilo koji može zameniti sa drugačijim. Zbog jednostavnosti, u ovom radu je zamišljeno da senzori šalju podatke pomoću http protokola. Naravno, u realnom sistemu bilo bi neophodno koristiti neki bezbedniji način komunikacije, odnosno https protokol je mnogo bolje rešenje.

Programabilno logički kontroleri šalju podatke do servera. Tehnologija u kojoj je server napravljen je apsolutno nezavisna od ostatka aplikacije. Kao što je opisano na početku ovog poglavlja, sistemi sa dobrom arhitekturom mogu na jednostavan način da se modifikuju i unapređuju. Samim tim, server može da bude implementiran u jednom programskom jeziku, a da se nakon nekog vremena implementira u nekom drugom. Kao deo servera se može posmatrati i deo aplikacije

zadužen za veštačku neuronsku mrežu zbog toga što se može reći da se obrada podataka treba izvršiti na serveru. Sa druge strane, da bi se dodatno razdvojile veštačke neuronske mreže od servera, mogu se posmatrati kao posebna komponenta sistema. Na ovaj način je moguće implementirati „serverless“ arhitekturu koja je nov koncept u informacionim tehnologijama. Amazon veb servisi (engl. amazon web services) su predstavili koncept „serverless“ arhitekture 2014. godine uz pomoć tzv. „lambda“ funkcija.

Nakon obrade podataka u veštačkoj neuronskoj mreži server šalje podatak u bazu podataka. Tendencija je da se u sistemima internet stvari koriste nerelacione baze podataka, uprkos tome što je zastupljenost relacionih baza podataka u ovom momentu značajno veća. U ovom radu neće biti objašnjene razlike između relacionih i nerelacionih baza podataka.

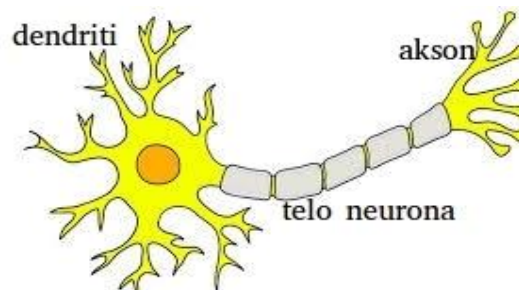
Konačno, korisniku sa administratorskim privilegijama je omogućeno da dodaje nove programabilno logičke kontrolere, ali i onemogućava komunikaciju postojećih. Takođe, korisnik je u mogućnosti da pogleda obrađene podatke iz baze podataka. Korisnik može da pristupi serveru uz pomoć internet pretraživača. Korisnik bez administratorskih privilegija nije prikazan u predloženoj arhitekturi sistema, a ako bi postojala potreba za njim bio bi implementiran na sličan način kao i korisnik sa administratorskim privilegijama. Razlika je samo u tome što korisnik bez administratorskih privilegija ne bi bio u mogućnosti da modifikuje funkcionalnost programabilno logičkih kontrolera.

3. VEŠTAČKE NEURONSKE MREŽE

3.1. Princip rada prirodnog neurona

Inspiracija za nastanak veštačkih neuronskih mreža je rad ljudskog nervnog sistema. Uprkos tome što nauka nije u potpunosti došla do odgovora kako radi nervni sistem i ljudski mozak, došlo se do zaključka da bi se kopiranje poznatog modela moglo iskoristiti za pravljenje veštačkih neuronskih mreža.

Prirodni neuron se sastoji iz dendrita, tela neurona i aksona na koji se odmah nastavljaju dendriti sledećeg neurona kao što je prikazano na slici 2.



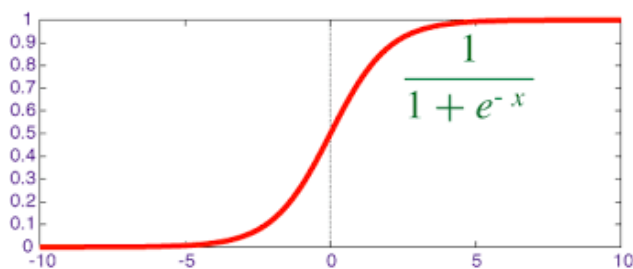
Slika 2. Izgled prirodnog neurona

Neuron na svom ulazu, odnosno na dendritima prima električni impuls, a zatim ga nakon obrade prosleđuje na svoj izlaz odnosno akson. Naravno, neuron može da ima više ulaza. Obrada signala se može posmatrati kao funkcija aktivacije, odnosno neuron neće proslediti električni impuls ako je impuls na ulazu jako mali. Realan

primer za to je kada čovek dotakne neku zanemarljivo toplu površinu. Iako je površina topla, do mozga neće stići informacija o tome da je nekakva opasnost u pitanju i čovek neće morati da refleksno skloni ruku sa te tople površine.

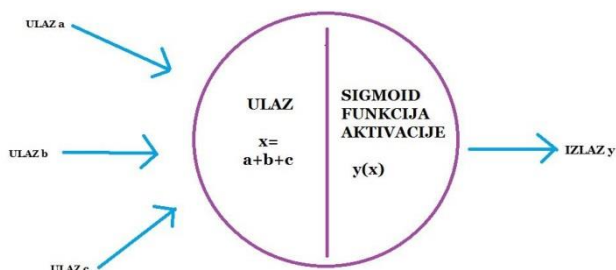
3.2. Kopiranje rada prirodnog neurona na veštačku neuronsku mrežu

Pošto je cilj kopirati model rada prirodnog neurona, jasno je da veštački neuron treba da ima mogućnost primanja više ulaza, a da se nakon toga ti ulazi trebaju obraditi uz pomoć funkcije aktivacije. Postoje razne funkcije aktivacije, a jedna od najčešće korišćenih u veštačkim neuronskim mrežama je „sigmoid“ funkcija aktivacije čiji je izgled prikazan na slici 3.



Slika 3. Izgled „sigmoid“ funkcije

Na osnovu navedenih komponenti potrebnih za predstavljanje veštačkog neurona, moguće je prikazati njegov izgled. Na slici 4 je predstavljen izgled veštačkog neurona.

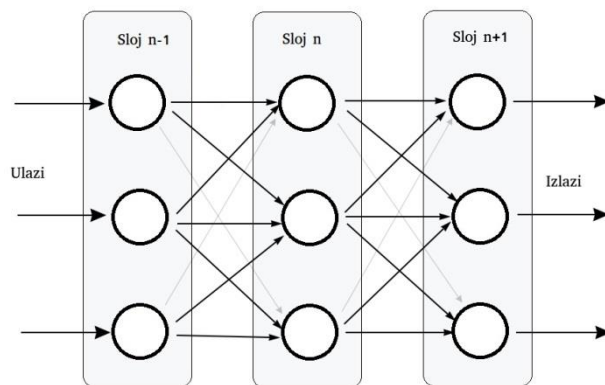


Slika 4. Izgled veštačkog neurona

Jasno je da su u prirodi neuroni međusobno povezani i da je potrebno tu povezanost preneti i na veštačke neuronske mreže. Način na koji bi se to moglo predstaviti je da postoje slojevi neurona gde je svaki neuron povezan sa svakim neuronom iz prethodnog i narednog sloja. Neuroni u veštačkim neuronskim mrežama se nazivaju nodovi.

Da bi neuronska mreža funkcionisala potrebno je definisati u kojoj meri svaki nod deluje na nod iz narednog sloja. U prirodi nemaju svi signali istu vrednost, odnosno težinu.

Na primer, ako čovek prstom dotakne vrelu vodu receptori na prstu će detektovati da je čovek dodirnuo nešto vlažno i vrelo, međutim do mozga će stići samo informacija da je prst dodirnuo nešto vrelo što preti da naškodi čoveku, što će dovesti do instinktivnog odmicanja prsta od opasnosti. Težine između nodova u veštačkoj neuronskoj mreži se obeležavaju slovom „w“. Na slici 5 je prikazana veštačka neuronska mreža.



Slika 5. Izgled kompletne veštačke neuronske mreže

Naredni sloj neurona se može opisati jednačinom

$$X = W \cdot I \quad (1)$$

gde X predstavlja matricu sledećeg sloja neurona, W matricu težina, a I matricu ulaza u sloj.

Na dobijenoj matrici izlaza je potrebno primeniti funkciju aktivacije, pa je formula za izlaz iz narednog sloja nodova:

$$O = \text{sigmoid}(X) \quad (2)$$

Na osnovu jednačina (1) i (2), moguće je na osnovu ulaza izračunati izlaze iz veštačke neuronske mreže. Da bi izlazi bili tačni, odnosno da bi veštačka neuronska mreža imala smisla, potrebno je da težine između nodova imaju dobre vrednosti. Određivanje težina nodova se vrši tako što se trenira veštačka neuronska mreža. Veštačka neuronska mreža se trenira tako što se dobijen izlaz poredi sa tačnom vrednosti izlaza, a onda se na osnovu greške modifikuju težine između nodova. Izvođenje formule za modifikovanje težina neće biti predstavljeno u ovom radu. U [1] je prikazano da je formula za modifikovanje težina:

$$\Delta W_{jk} = \alpha * E_k * \text{sigmoid}(O_k) * (1 - \text{sigmoid}(O_k)) O_j^T \quad (3)$$

gde je „j“ nod iz prethodnog sloja, dok je „k“ nod iz trenutnog sloja. E je matrica greške, O matrica izlaza prethodnog sloja, dok α predstavlja težinski faktor koji omogućava postepeno učenje, odnosno modifikovanje težina.

Da bi veštačka neuronska mreža radila potrebno je podesiti ulaze i izlaze na moguće vrednosti, odnosno potrebno je skalirati ulaze i izlaze. Ulazi u neuronsku mrežu trebaju da budu u granicama od 0.1 do 1 iz razloga što funkcija aktivacije postaje izuzetno ravna za vrednost 0 ili za vrednosti veće od 1. Iz istog razloga, odnosno zbog funkcije aktivacije poslednjeg sloja, izlazi veštačke neuronske mreže su u opsegu 0.01 – 0.99. Poslednja stvar koju je potrebno podesiti su težine nodova. Jedan od načina za inicijalizaciju težina predstavljenih u [1] je da matrica W bude popunjena sa normalnom distribucijom sa centrom u nuli i sa standardnom devijacijom koja je predstavljena u jednačini (4).

$$\frac{1}{\sqrt{\text{broj ulaza u nod}}} \quad (4)$$

3.3 Prepoznavanje rukom napisanih brojeva uz pomoć veštačke neuronske mreže

Rukom napisan jednocifren broj se može predstaviti slikom veličine 28x28 piksela, gde svaki piksel ima vrednosti od 0 do 255. Ta slika je ulaz u veštačku neuronsku mrežu i potrebno ga je skalirati na vrednosti između 0.01 i 1 kao što je opisano u prethodnom poglavlju. Mogući izlazi iz neuronske mreže su brojevi od 0 do 9, što znači da izlaz neuronske mreže treba da ima 10 nodova.

Prilikom treniranja veštačke neuronske mreže potrebno je postaviti sve nodove koji ne odgovaraju stvarnom izlazu na 0.01, dok nod koji odgovara stvarnom izlazu treba postaviti na vrednost 0.99. Na primer, ako se na slici koja je ulaz u veštačku neuronsku mrežu nalazi broj 0, to znači da prvi nod (nod koji predstavlja broj nula) treba da ima vrednost 0.99, a svi ostali vrednost 0.01.

Iz gore navedenog, broj ulaznih nodova je 28*28, odnosno 784. Broj izlaznih nodova je 10. Da bi se kreirala veštačka neuronska mreža potrebno je još postaviti broj nodova u skrivenom sloju, odnosno sloju između ulaza i izlaza, kao i odrediti težinski faktor. Ne postoji jasan način za određivanje broja skrivenih nodova, kao i težinskog faktora, odnosno ove dve vrednosti se eksperimentalno određuju. Kvalitet treniranja veštačke neuronske mreže se može poboljšati većim brojem pokušaja treniranja veštačke neuronske mreže. Svaki proces treniranja veštačke neuronske mreže se naziva epoha. U ovom radu neće biti detaljno predstavljene razlike u uspešnosti veštačke neuronske mreže na osnovu postavljenih parametara. Na osnovu [1] pokazano je da je sa 200 nodova u skrivenom sloju, težinskim faktorom od 0,1 i sa 5 epoha, uspešnost veštačke neuronske mreže prilikom prepoznavanja rukom napisanih jednocifrenih brojeva 97,51%.

4. BAZE PODATAKA U SISTEMIMA INTERNET STVARI

Baze podataka se mogu podeliti na relacione i nerelacione baze podataka. U ovom radu su predstavljene osnovne razlike između ove dve grupe baza podataka i prikazano je koja vrsta se bolje pokazala u sistemima internet stvari.

Zahtev sistema internet stvari su takvi da podaci moraju biti nestrukturirani zbog toga što se podaci generišu uz pomoć različitih senzora koji se mogu i menjati tokom vremena. Relacione baze podataka, koje su trenutno mnogo popularnije od nerelacionih imaju jasno struktuirane tabele i upite baza podataka, pa samim tim nisu u mogućnosti da na dobar način čuvaju nestruktuirane podatke. Zbog toga, u sistemima internet stvari, sve je veća popularnost nerelacionih baza podataka.

U [2] je rečeno : "Relacione baze podataka poput Oracle-a su odlične za podatke iz jedne kompanije, ali ne mogu da pruže skalabilnost ili dostupnost poput baza podataka napravljenih za cloud poput Cassandra baze podataka". Zanimljivo je primetiti, u skladu sa statističkim podacima o popularnosti baza podataka navedenim u [3], da je trenutna popularnost Oracle baze podataka značajno veća od Cassandra baze podataka. To se može objasniti time što su sistemi internet stvari relativno nova oblast i može se očekivati budući rast popularnosti nerelacionih baza podataka usled sve veće zastupljenosti sistema internet stvari.

5. ZAKLJUČAK

U ovom radu je prikazan značaj arhitekture u sistemima internet stvari i pokazan je primer jednostavne arhitekture koja omogućava prenos podataka od senzora do servera na kome se izvršava obrada podataka.

Takođe, prikazana je obrada podataka uz pomoć veštačkih neuronskih mreža. Prikazan je jednostavan primer obrade rukom napisanih jednocifrenih brojeva.

Ovaj rad može poslužiti kao osnova za dizajniranje arhitekture sistema internet stvari, ali i za razumevanje principa rada veštačkih neuronskih mreža.

6. LITERATURA

- [1] Tariq Rashid, "Make your own neural network", CreateSpace Independent Publishing Platform, 2016.
- [2] <https://readwrite.com/2014/11/28/internet-of-things-nosql-data/>
- [3] <https://db-engines.com/en/ranking>

Kratka biografija:



Bojan Baltić rođen je u Zagrebu 1991. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Automatika i upravljanje sistemima odbranio je 2017.god.

ZAŠTITA VISOKONAPONSKOG TRANSFORMATORA PROTECTION OF HIGH VOLTAGE TRANSFORMER

Anđelka Dražić, Đura Oros, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ENERGETSKA ELEKTRONIKA I ELEKTRIČNE MAŠINE

Kratak sadržaj – Vrste i podela energetskih transformatora, njihova zaštita i primer zaštite visokonaponskog transformatora u TS 400/110kV „Vranje 4“

Abstract – Types and division of power transformers, their protection and example of protection of high voltage transformer in substation 400 / 110kV "Vranje 4"

Cljučne reči: Energetika, Transformator, Zaštita

1. UVOD

Transformator je statički elektrotehnički aparat koji transformiše električnu energiju jednog naizmennog napona i naizmennu struju u električnu energiju drugog napona i druge struje, po pravilu iste učestanosti. Ovu funkciju mogu da obavljaju za jednu ili više faza te imamo jednofazne ili višefazne, najčešće trofazne transformatore. Pri svakoj transformaciji energije javljaju se gubici. Step en iskorišćenja energije kod energetskih transformatora je vrlo visok, što s jedne strane omogućava tehnički opravdan prenos električne energije, a sa druge strane ih uvrštava u klasu najbrojnijih električnih mašina u elektroenergetskom sistemu. Oblast njihove primene počinje od samog prenosa i raspodele električne energije, razne transformacije struje i napona, primene u tehnici veza, automatike, telemehanike, radiotehnike, merenjima i mnogim drugim oblastima.

Najvažnija vrsta transformatora bez kojih je nemoguć prenos električne energije, koji se najviše upotrebljavaju, a snage im se kreću od stotinu do više desetina hiljada kilovoltampera, a naponi do nekoliko stotina kilovolta, jesu energetski transformatori.

Kada je reč o zaštiti, osnovni zahtevi koji se postavljaju pred nju su selektivnost, brzina delovanja i osetljivost.

2. ZAŠTITA VISOKONAPONSKOG TRANSFORMATORA U TS 400/110 kV „VRANJE 4“

Transformatorsko polje 400kV u transformatorskoj stanici „Vranje 4“ sastoji se od sledećih elemenata:

- Trofazni uljni transformator Končar (2ARZd 300000-420) 400/110/10,5kV; Yna0+d5; 300/300/70MVA, slika 1;
- Strujni transformatori IOSK 420 Trench, 2x500/1/1 A, 3x800/1/1/1 A;
- Jednopolni prekidači 420kV, 3AP2 F1 Siemens
- Jednopolni pantografski rastavljači 420kV, sa noževima za uzemljenje, P RF2 420+AE BF2 Siemens

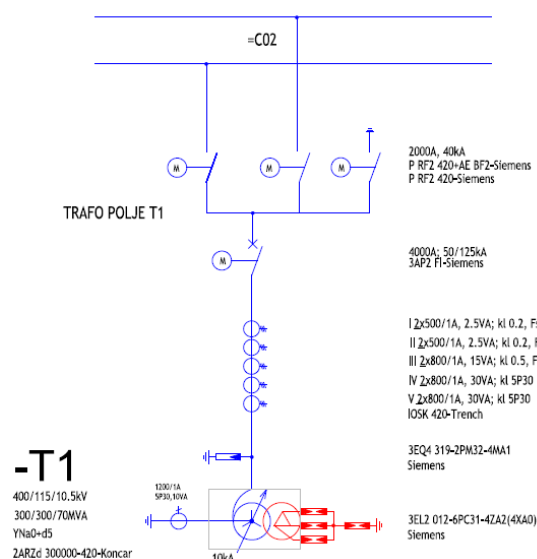
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Đura Oros, vanr.prof.



Slika 1. Trofazni uljni transformator 2ARZd 300000-420, Končar

Na slici 2 prikazana je jednopolna šema transformatorskog polja 400kV trafostanice „Vranje 4“.

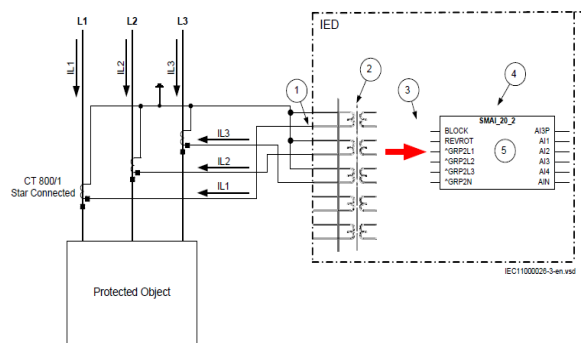


Slika 2. Jednopolna šema trafo polja

2.1. Diferencijalna zaštita

Zaštita visokonaponskog transformatora 400/110/10,5kV urađena je pomoću zaštitnog uređaja RET670 (main 1), proizvođača ABB.

Četvrto jezgro strujnog mernog transformatora se preko ispitne utičnice povezuje na zaštitni uređaj koji kontroliše struju po fazama. Šema vezivanja uređaja sa strujnim transformatorom prikazana je na slici 3.



Slika 3. Šema vezivanja uređaja sa strujnim transformatorom

Zaštita ima sledeće funkcije:

- diferencijalna zaštita autotransformatora
- ograničena zemljospojna zaštita
- višestepena trofazna prekostrujna zaštita
- višestepena zemljospojna zaštita
- zaštita od preopterećenja
- funkcija snimanja poremećaja u mreži
- funkcija hronološke registracije događaja
- funkcija samonadzora
- monitoring ulaznih mernih veličina na displeju
- komunikacija sa staničnom centralnom jedinicom

Lične zaštite transformatora, kao što su buholc, zaštita od nadpritiska, zaštita od nad-temperature ulja i namotaja, kontaktni termometar, nizak nivo ulja, su izvedene nezavisno, a signalizacija se uvodi u sistem zaštite i upravljanja.

Uređaj RET670 (slika 4), ima svoj LCD displej za prikaz merenja, signalizacije, ima alfanumeričku tastaturu, LED diode s mogućnošću indikacije alarma i prednji port za parametrizaciju.



Slika 4. RET 670, ABB

2.2. Kontrola isključnih krugova

Isključenjem i kontrolom prekidača direktno utičemo na zaštitu štićenog transformatora. Za kontrolu isključnih krugova prekidača koristi se relej FR02, proizvođača IED Beograd (slika 5).

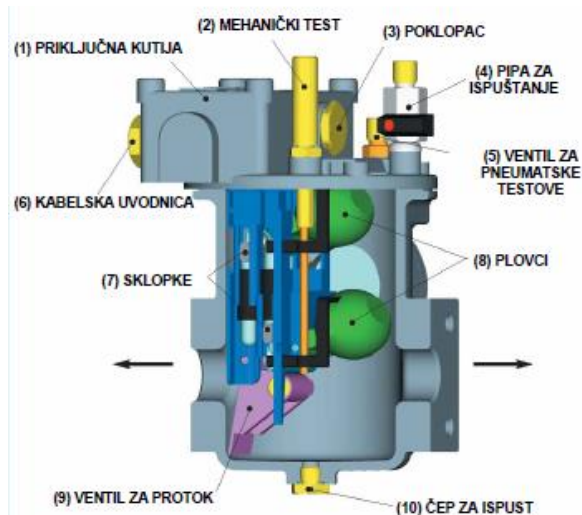


Slika 5. FR02, IED Beograd

2.3. Buholc relej

Plinski (Buholc) relej se montira između transformatora i njegovog konzervatora i tokom normalnog rada transformatora je napunjen uljem. Kada se u transformatoru stvori plin, on se diže prema konzervatoru i skuplja se u gornjoj komori releja. Nivo ulja padne i gornji plovak pokrene alarmnu sklopku. Plin ne sme slobodno prolaziti kroz telo releja i pobeći u cevi pre nego što se aktivira kontakt alarma.

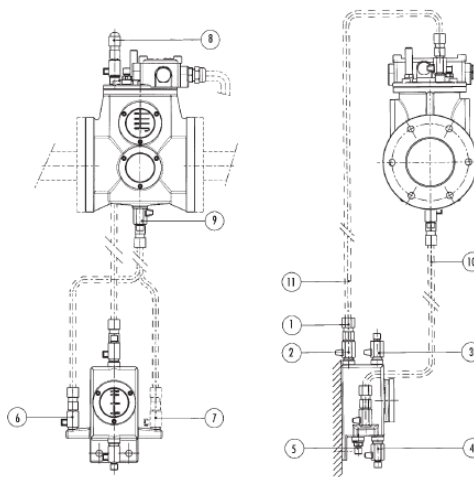
Korišćen je relej BR 80 MLO, proizvođača „COMEM“, prikazan na slici 6.



Slika 6. Buholc relej

2.3.1. Uređaj za uzimanje uzorka plina

Analizom nastalih plinova dobija se dobar pokazatelj o kojoj vrsti kvara se radi, ali pristup Buholc releju dok je transformator u pogonu može biti opasan. Uređaj za uzimanje uzorka plina je dizajniran da se izbegne taj problem postavljanjem na sigurno mesto s kojeg će poslužitelj uređaja neometano moći uzimati uzorak plina dok je transformator u radu. Na slici 7 prikazan je položaj uređaja za uzimanje uzorka plina u odnosu na Buholc relej; korišćen je uređaj proizvođača „COMEM“.



Slika 7. Uređaj za uzimanje uzorka plina

2.4. Termometar

Merni instrument za merenje temperature kod energetskih transformatora je termometar. Sakupljanje i prikazivanje

izmerenih vrednosti temperatura se obavlja preko patentiranog mernog sistema koji se sastoji od 4 elementa: 1. Termometarska sonda, 2. Kapilara, 3. Bourdonovo pero i kod izvedbe “/TT”, 4. Čelija za merenje pritiska. Sva 4 elementa sačinjavaju zatvoreni sistem sa tečnošću. Korišćen je termometar MT-ST160F, proizvođača „MESSKO“, slika 8.



Slika 8. MT-ST160F, MESSKO

2.4.1. Termoslika

Korišćen je uređaj EPT202, proizvođača „MESSKO“, slika 9.



Slika 9. EPT202, MESSKO

2.5. Odušnik

Sigurnosni uređaj štiti transformatore od neprihvatljivog i pretećeg porasta pritiska. Po dostizanju određene vrednosti pritiska, koji se unapred utvrđuje, sigurnosni će se uređaj otvoriti, smanjiti pritisak te se hermetički iznova zatvoriti nakon što postigne smanjenje pritiska. Korišćen odušnik: LMPRD 685-08-22-N00-00, proizvođača „MESSKO“, slika 10.

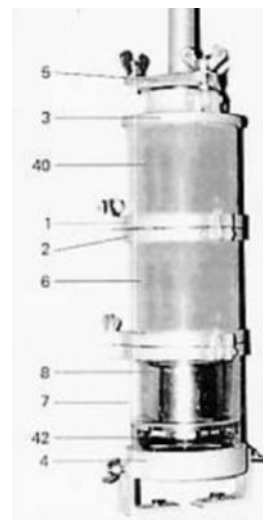


Slika 10. LMPRD 685, MESSKO

2.6. Sušionik vazduha

Sušionik vazduha eliminiše praktično svu vlagu iz vazduha koji struji kroz njega u konzervator za ulje, kad se transformator hladi. Na taj se način sprečava smanjivanje dielektričke otpornosti izolacije zbog vlažnog spoljnog vazduha i stvaranja kondenzata u konzervatoru ulja. Time ovaj uređaj povećava pogonski integritet transformatora.

Korišćen je tip DIN 42 562-L5, proizvođača „Siemens“, prikazan na slici 11.

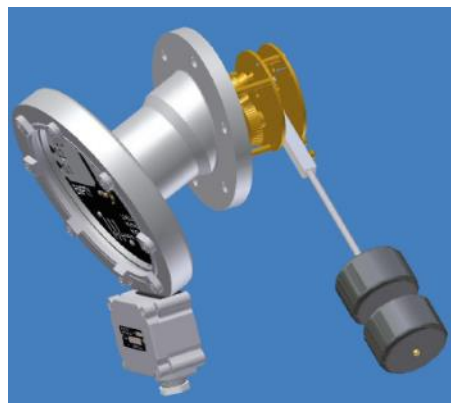


Slika 11. DIN 42 562-L5, Siemens

Delovi pomenutog sušionika vazduha su: 1-međuploča, 2-međupokrov, 3-pokrov, 4-donji prsten, 5-prirubnica, 6-stakleni cilindar, 7-staklena posuda, 8-zvono separatora, 40-kristali za sušenje vazduha, 42-hvatač ulja.

2.7. Indikator nivoa ulja

Indikator nivoa ulja je potreban da bi se dobila analogna indikacija unutar konzervatora pomoću brojčanika sa strelicom plus jedan ili više električnih signala (maksimalno četiri) kada ulje unutar konzervatora dostigne maksimalni ili minimalni nivo. Korišćen je IFG FK2, proizvođača „CEDASPE“, slika 12.



Slika 12. IFG FK2, CEDASPE

2.8. Releј regulacione sklopke

Zaštitni releј služi za dojavu greške unutar uljne posude teretne preklopke regulacione sklopke, odnosno uljne posude teretnog birača i ima zadatak da očuva sklopku i transformator od oštećenja.

On reaguje samo na strujanja ulja koja su usmerena od glave regulacione sklopke prema konzervatoru. Preklapanje regulacione sklopke pod teretom kod nazivnog tereta ili dopuštenog preopterećenja, ne dovodi do prorade zaštitnog releja.

3. ZAŠTITA VISOKONAPONSKIH TRANSFORMATORA OD POŽARA

Ako se elektroenergetsko postrojenje visokog napona postavlja na otvoren prostor, uljni transformatori moraju biti udaljeni od susjednih zgrada prema tabeli 1.

Tabela 1. Udaljenost transformatora od susjednih zgrada

Transformator nazivne snage (MVA)	Najmanja udaljenost (m)
do 10	3
preko 10 do 40	5
preko 40 do 200	10
preko 200	15

Ako je elektroenergetsko postrojenje visokog napona postavljeno u blizini stambene ili poslovne zgrade, bolnice, vrtića, škole, garaže, udaljenost iz tabele se povećavaju dvostruko. Ako se to ne može ispuniti, zaštita od širenja požara se mora predvidjeti pregradnim zidom. Za transformatorske stanice smeštene na stubu, energetske transformatori moraju biti udaljeni od susjednih objekata najmanje 3 metra, a energetske transformatori postavljeni na otvoren prostor bez međuprostora, moraju se odvojiti vatrootpornim zidovima.

4. ZAKLJUČAK

Prema tehničkim preporukama TP-07 i TP-4b, u elektroenergetskom sistemu Srbije za energetske transformatore se koriste uređaji za zaštitu navedeni u tabeli broj 2.

Tabela 2. Zaštita transformatora prema postojećim preporukama

Zaštita 400 i 220 kV transformatora
Diferencijalna zaštita
Zemljospojna (kućišna) zaštita
Zemljospojna diferencijalna zaštita
Distantna (dvosmerna) zaštita na niženaponskoj strani
Prekostrujna zaštita na višenaponskoj strani
Zaštita od preopterećenja
Termička slika
Kontaktne termometre
Zemljospojna zaštita tercijera
Zaštita od nesimetrije polova prekidača
Zaštita od trajnog nesimetričnog pogona
Uređaj za automatsku regulaciju napona (ako je regulacioni transformator)
Prebacivanje delova zaštite dalekovoda na prekidač spojnog polja pomoćnog sistema sabirnica (po potrebi)
Uređaj za kontrolu isključnih krugova
Osciloperturbograf

Najvažniji element svake transformatorske stanice predstavlja energetske transformator te je osnovni cilj da se on zaštiti najboljom i najefikasnijom opremom, zbog svoje uloge, a i cene. Održavanje transformatora je u istoj meri važno te je bitno uočiti i reagovati na iznenadne i neuobičajene promene.

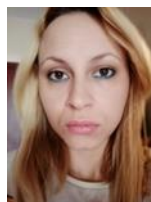
Iznenadnim promenama se smatraju odstupanja rezultata merenja od uobičajenih vrednosti i prorade zaštite energetske transformatora.

Nakon prorade zaštite transformatora treba utvrditi verodostojnost iste, s ciljem eliminisanja lažnih signala. Transformator se ne sme ponovo pustiti u pogon dok se ne otkloni sumnja da je kvar u transformatoru.

5. LITERATURA

- [1] Mr Gojko Dotlić, dipl.inž., Savez mašinskih i elektrotehničkih inženjera i tehničara Srbije, „Elektroenergetika kroz standard, zakone, pravilnike i tehničke preporuke“, Beograd, 2013.
- [2] Branko Mitraković, dipl.inž., Naučna knjiga, „Transformatori“, Beograd, 1968.
- [3] Franjo Božuta, dipl.el.inž., „Automatski zaštitni uređaji elektroenergetskih postrojenja“, Svjetlost OOUR zavod za udžbenike i nastavna sredstva sarajevo, 1989.

Kratka biografija:



Andelka Dražić, rođena je u Mokrinu 1986. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine odbranila je 2017.god.



Đura Oros rođen je u Ruskom Krsturu 1957. Doktorirao je na Fakultetu tehničkih nauka 2008. godine. Vanredni je profesor. Oblast interesovanja su električne mašine.

**BRZA SELEKCIJA KODNIH JEDINICA ZA INTRA MOD SCC EKSTENZIJE HEVC
KODERA****FAST INTRA MODE CODING UNIT SELECTION FOR HEVC SCREEN CONTENT
CODING**

Stevan Filipović, *Fakultet tehničkih nauka, Novi Sad*

**Oblast – ENERGETIKA, ELEKTRONIKA I
TELEKOMUNIKACIJE**

Kratak sadržaj – *High Efficiency Video Coding (HEVC)* predstavlja standard video kompresije koji obezbeđuje znatno veću efikasnost kodovanja od svojih prethodnika, po cenu povećane kompleksnosti. Kako bi se ubrzao proces odabira kodnih parametara, predlaže se algoritam brze selekcije kodnih jedinica na osnovu teksture frejma, namenjen intra-predikcionom modu i sekvencama koje se mogu tretirati kao kompjuterski generisan sadržaj.

Abstract – *High Efficiency Video Coding (HEVC)* is a video compression standard providing higher coding efficiency levels than its predecessors at the cost of increased complexity. In order to speed up the coding parameters selection process, a fast coding unit selection algorithm is proposed. Based on frame texture information derived from local binary patterns, it is intended for intra-prediction mode screen content coding.

Ključne reči: Zbornik FTN, Studentski radovi, Uputstvo

1. UVOD

High Efficiency Video Coding (HEVC) [1] standard je video kodovanja koji koristi hibridnu arhitekturu koderi i podelu slike na blokove. Nastao je u okviru organizacije JCT-VC (*Joint Collaborative Team on Video Coding*) i dokumentovan je kao ITU-T preporuka H.265 i ISO/IEC standard MPEG-H Part 2. Veća rasprostranjenost video sadržaja visoke rezolucije, rast opterećenja mreža koje isporučuju video sadržaj, kao i veći udeo mobilnih uređaja kao korisnika, doveli su do potrebe za razvojem novog standarda koji bi obezbedio efikasniji postupak kodovanja, sa naglaskom na snimke visoke rezolucije i značajniju upotrebu paralelnih arhitektura obrade podataka. Cena efikasnijeg kodovanja prevashodno se manifestuje kao povećanje kompleksnosti.

ITU-T i ISO/IEC standardi određuju strukturu i sintaksu izlaznog bitstrima, kao i postupak dekodovanja. Time se garantuje da će se dekodovanjem bitstrima generisanog u skladu sa standardom uvek dobiti isti izlazni video signal. Pošto je jedino ograničenje koderi obaveza mapiranja ulaznog signala u standardizovani bittrim, sam algoritam i implementacija kojima se on dobija mogu se prilagoditi potrebama aplikacije i željenim performansama.

Kompleksnost koderi u velikoj meri zavisi od broja funkcionalnosti koje su mu na raspolaganju. HEVC između ostalog podrazumeva veći broj kombinacija modova, kao posledicu fleksibilne hijerarhijske strukture podele na blokove i većeg broja predikcionih modova. Zahtevi pretraživanja ovakvog proširenog prostora dozvoljenih kodnih parametara negativno se odražavaju na kompleksnost.

Algoritam predložen u ovom radu za cilj ima smanjenje kompleksnosti koderi. Fokus je bio na ubrzanju donošenja odluka o podeli blokova za frejmove u kojima dominira kompjuterski generisan sadržaj - *screen content (SC)*. Ispitivane su metode detekcije teksture i obrade slike koje bi ukazale na korelaciju izdvojenih obeležja i očekivanih veličina blokova. Kako je za ovakav sadržaj specijalizovana SCC (*Screen Content Coding*) ekstenzija HEVC standarda [2], rešenje je realizovano kao modifikacija SCC ekstenzije ograničena na intra-predikcioni mod.

2. HIBRIDNI KODER

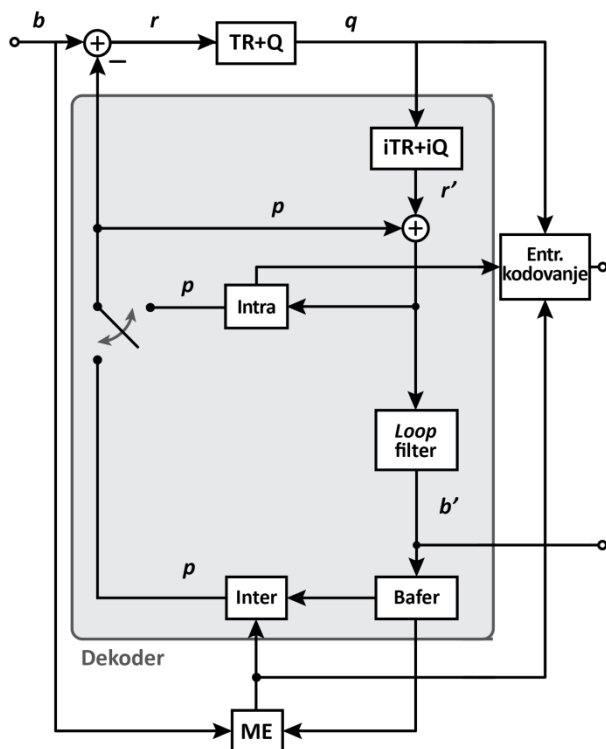
Video koderi sa hibridnom arhitekturom koriste spoj dva pristupa – predikcije, odnosno diferencijalne impulsne kodne modulacije, i transformacionog kodovanja – kako bi uklonili suvišne podatke iz ulaznog signala. Ulazni odmerci se prostorno i vremenski dekorelišu generisanjem razlike između ulaznog signala i signala predikcije dobijenog na osnovu prethodno kodovanih odmeraka. Transformacija signala razlike takođe dekorelišu podatke projektovanjem na ortogonalne vektore bazisa. Pored redundantnih podataka uklonjenih na ovaj način, kvantizacijom transformacionih koeficijenata eliminišu se i manje značajni podaci [3]. Kao transformacija najčešće se koristi diskretna kosinusna transformacija.

Na slici 1. prikazana je uopštena šema hibridnog koderi. Frejm se prilikom kodovanja deli na blokove b . Rezidualni signal r dobija se kao razlika ulaza b i bloka predikcije p . Transformacijom bloka r i kvantizacijom koeficijenata određuju se vrednosti kojima će odmerci bloka biti predstavljeni u izlaznom bitstrimu.

Poslednji korak potreban za generisanje bitstrima podrazumeva entropijsko kodovanje kvantizovanih koeficijenata q , zajedno sa odabranim kodnim parametrima i višim sintaksnim elementima. U HEVC standardu ovaj postupak podrazumeva binarizaciju vrednosti elemenata koje se u tom obliku prosleđuju kontekstno-adaptivnom binarnom aritmetičkom koderu (CABAC) [4].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dejan Vukobratović, red.prof.



Slika 1. Šema hibridnog kodera [3]

Dekoder će rekonstruisati b sabiranjem bloka predikcije i rezidualnog signala. Potrebna predikcija dobija se na osnovu prethodno dekodovanih odmeraka određenih kodnim parametrima preuzetim iz bitstrima. Sa r' je označen rekonstruisani rezidualni blok dobijen dekvantizacijom (iQ) i inverznom transformacijom (iT).

Referentni signal na osnovu kojeg se generiše predikcija određen je modom. Odmerci aktuelnog frejma koriste se u intra modu, dok se za inter-predikciju upotrebljavaju odmerci drugih, prethodno kodovanih frejmova.

2.1. Intra-predikcija

Intra-predikcija predstavlja proces dobijanja rezidualnog signala poređenjem bloka sa rekonstruisanim odmercima iz okolnih blokova unutar iste slike. HEVC podržava 35 modova inter-predikcije: DC, ravanski, i 33 ugaona moda koji se dalje dele na 17 vertikalnih i 16 horizontalnih.

Na luminentne blokove se može primeniti celokupni opseg modova. Za hrominentne je rezervisano pet opcija: DC, ravanski, V0, H0 i tzv. izvedeni mod koji je jednak modu odabranom za luminentnu komponentu [1].

3. PODELA SLIKE NA BLOKOVE

Raširena upotreba HD i Ultra-HD snimaka stvorila je potrebu za unapređenjem mehanizma podele slike. Sa porastom rezolucije, javljaju se veće oblasti kojima se mogu dodeliti isti parametri predikcije, odnosno značajnija je prostorna korelacija između odmeraka.

Ovo svojstvo većih formata se može iskoristiti povećanjem jedinica obrade slike, čime se obezbeđuje bolje poklapanje sa strukturama u snimku i smanjenje redundantnog signaliziranja. Istovremeno je značajno da se zadrže manje jedinice kako bi se omogućilo prilagođavanje kodera lokalizovanim svojstvima slika. Fleksibilnost obezbeđena na ovaj način povećava kompleksnost algoritma.

HEVC definiše četiri vrste jedinica: jedinice kodnog drveta (CTU – *coding tree unit*), kodne jedinice (CU – *coding unit*), predikcione jedinice (PU – *prediction unit*) i transformacione jedinice (TU – *transform unit*). Svaka jedinica sadrži potrebne sintaksne elemente i po jedan istoimeni blok odmeraka za svaku komponentu slike (CTB – *coding tree block*, CB – *coding block*, PB – *prediction block* i TB – *transform block*).

CTU je osnovna jedinica obrade veličine $N \times N$, gde N uzima vrednosti iz skupa $\{16, 32, 64\}$. Odabirom veličine CTU-a, kompleksnost procesa može se prilagoditi potrebama ili ograničenjima kodera. Veće vrednosti N povećavaju kompleksnost, ali ujedno pružaju efikasnije modelovanje HD sadržaja. CTB luminentne komponente obuhvata $N \times N$ odmeraka, dok dimenzije CTB-ova hrominentnih komponenti zavise od formata slike. Npr. kod formata 4:2:0, hrominentni CTB-ovi će obuhvatati $(N/2) \times (N/2)$ odmeraka.

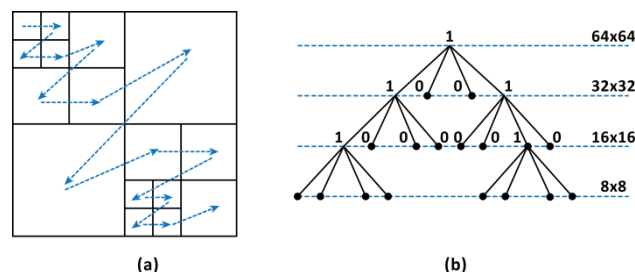
3.1. Kodna jedinica

CTU se deli na kodne jedinice na nivou kojih se bira vrsta predikcije. Promenljiva veličina jedinica obezbeđuje fleksibilnost prilikom odlučivanja o modu. Poput CTU-a, CU se sastoji od jednog luminentnog i dva hrominentna kodna bloka. Dozvoljene dimenzije CU-a se kreću od 8×8 do 64×64 luminentnih odmeraka.

Struktura podele CTU-a odgovara kvadratnom drvetu (QT – *QuadTree*) čiji početni čvor predstavlja nepodeljeni CTU koji sadrži samo jedan CU iste veličine. CTU veličine $N \times N$ može se podeliti na četiri jednaka CU-a veličine $(N/2) \times (N/2)$, čime bi se dobila četiri nova čvora. Svaki od njih se dalje može deliti na isti način, sve dok se ne dostigne minimalna veličina unapred određena za CU. Čvorovi u kojima je odlučeno da se prestaje sa podelom postaju listovi drveća. Svaki list predstavlja jedan CU.

Na slici 2.a. prikazana je struktura podele CTU-a veličine 64×64 . 16 kodnih jedinica veličina od 8×8 do 32×32 formirano je redosledom označenim isprekidanim strelicama. Ovaj tzv. z-redosled garantuje da će pre aktuelnog bloka biti kodovani blokovi koji se nalaze levo i iznad njega. Slika 2.b. sadrži kodno drvo [1].

Za odmerke unutar svakog CU-a određuje se jedinstveni mod kodovanja. Odluka se donosi na osnovu odmeraka luminentnog CB-a, i važi takođe za hrominentne CB-ove.



Slika 2. Grafički prikaz kodnog drveta podele CTU-a

4. OPTIMIZACIJA CENE KODOVANJA

Kvalitet kodovanja datog signala određuje se na osnovu dva kriterijuma – izobličenja odmeraka i protoka izlaznih podataka. Nijedan sam po sebi ne daje dovoljno informacija o performansama kodera, i ne mogu se posmatrati nezavisno jedan od drugog. Pored konačne procene kvaliteta, takođe služe kao optimizaciona metrika

prilikom odabira kodnih parametara. Optimizacija nije propisana standardom, ali se mnogi predviđeni mehanizmi implicitno oslanjaju na nju i znatno gube na efikasnosti ukoliko je izostavljena [5]. Kontrola kodera upravlja pretragom svih dozvoljenih skupova kodnih parametara p rukovodeći se minimizacijom *Rate-Distortion* (RD) cene slike definisane kao

$$J(p) = D(p) + \lambda R(p), \quad (1)$$

gde je D izobličenje, R protok, i λ Langranžov multiplikator koji određuje udeo izobličenja i protoka u vrednosti J . Izraz za cenu predstavlja problem minimizacije bez ograničenja [5].

4.1. Pretraga kodnog drveta

Da bi se odredila optimalna podela CTB-a, potrebno je među svim mogućim podelama na kodne blokove pronaći onu koja minimizuje *Rate-Distortion* cenu. Za svaki od potencijalnih kodnih blokova traže se optimalni predikcioni parametri i predikcioni blokovi. Da bi se oni dobili, unutar kodnog bloka se takođe pretražuju sve moguće podele na transformacione blokove – za svaki predikcioni mod.

G-BFOS [5] pretraga generiše celo kvadratno drvo dubine d , i ujedno računa cenu J za svaki čvor i list. Počev od svakog čvora na dubini $d-1$, poredi se njegova cena sa zbirom cena listova koji mu pripadaju. Ako je cena čvora manja, listovi se odbacuju. U suprotnom, čvoru se dodeljuje nova manja cena i nadalje se tretira kao list. Postupak se ponavlja dok se ne provere sve dubine.

5. SCC

Screen Content obuhvata video sadržaj u kojem dominiraju kompjuterski generisani elementi umesto uobičajenih sekvenci usvojenih kamerom, tj. prirodnog sadržaja. SCC ekstenzija povećava efikasnost kodovanja odgovarajućih snimaka prilagođavanjem standarda svojstvima specifičnim za SC.

Dok su kod prirodnog sadržaja prisutni senzorski šum, izraženi pokreti i relativno visok nivo korelacije između različitih boja, za SC su karakteristične česte uniformne površine i ponavljanja obrazaca unutar frejma, kao i prisustvo identičnih oblasti u različitim frejmovima. Takođe, kod boja se može očekivati visoka saturacija i sveden skup različitih nijansi u jednom frejmu ili u nizu frejmova.

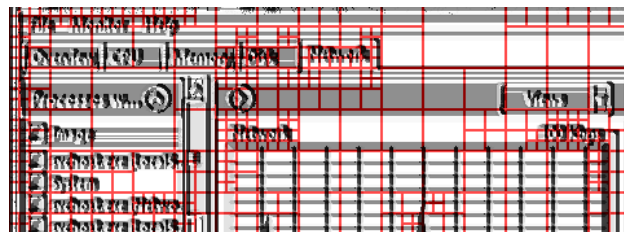
6. PREDLOG ALGORITMA ZA BRZU SELEKCIJU KODNIH JEDINICA

Hijerarhijska struktura blokova i proračuni u okviru RD optimizacije predstavljaju ključne izvore kompleksnosti HEVC kodera. U intra modu u kom se bira jedan od 35 intra-predikcionih modova, *brute force* pristup zahteva formiranje 35 ugnježenih rezidualnih QT-ova u svakom čvoru QT-a bloka, kao i operacije generisanja reziduala, transformacije, kvantizacije, entropijskog kodovanja, inverzne transformacije i dekvantizacije na nivou svakog čvora rezidualnog QT-a.

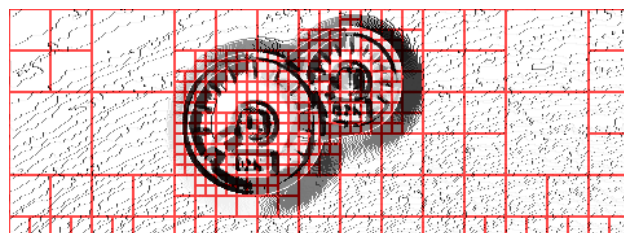
Pristupi ubrzanju podele na blokove mogu se svrstati u tri grupe: pojednostavljenje modela RD optimizacije, smanjenje broja modova predikcije za potpunu RD optimizaciju, i smanjenje opsega veličina kodnih, predikcionih ili transformacionih jedinica za potpunu RD optimizaciju.

Algoritam predložen u ovom radu spada u treću grupu, i podrazumeva donošenje brzih odluka o podeli kodnih jedinica na osnovu tekstone slike dobijene na primenom lokalnih binarnih obrazaca (LBP – *Local Binary Patterns*) [7]. Osnovna pretpostavka je da će zbog karakteristika SC-a biti moguće da se na efikasan način iskoristi korelacija između tekstone i veličine odabranog bloka.

Na slikama 3. i 4. prikazana je superpozicija LBP tekstone delova frejma i strukture podele na kodne jedinice dobijene referentnom implementacijom HEVC kodera HM-16.15+SCM-8.5. Može se primetiti da se u uniformnim oblastima sa slabo izraženom teksturom, kao i u oblastima u kojima se tekstura sastoji od pravilnih obrazaca, biraju veći blokovi, dok se delovi slike sa puno detalja koduju u manjim blokovima.



Slika 3. LBP tekstura i podela na blokove dela frejma sa pravilnom strukturom



Slika 4. LBP tekstura i podela na blokove dela frejma sa pozadinskim gradijentom

Kako bi se iskoristile uočene veze između veličine blokova i tekstone, uvedena su dva uslova bez kojih nije dozvoljena RDO:

1. Ako blok ne sadrži nikakvu teksturu, RDO se dozvoljava samo za čvor QT-a koji odgovara datom bloku.
2. Ako su tekstone bilo koja dva susedna podbloka u izvesnoj meri slične, RDO se dozvoljava za čvor koji odgovara datom bloku i za njegovu decu.

Deo frejma izdvojen u slici 4. prikazuje teksturu koja odgovara pozadinskom gradijentu. Prethodno definisani koraci u ovakvim elementima detektuju izraženu nepravilnu teksturu, po čemu znatno odstupaju od ponašanja referentnog kodera. Kako bi se nadoknadio ovaj nedostatak, uveden je još jedan uslov:

3. Ako blok ne sadrži ivice, RDO se dozvoljava za čvor koji odgovara datom bloku i za njegovu decu.

6.1. Implementacija algoritma

Kako bi se proverili prethodno navedeni uslovi, u sledećim koracima generišu se po jedna slika tekstone T i slika ivica I za svaki frejm:

1. Frejm se propušta kroz dvodimenzionalni Gausov filter veličine 5×5 , čime se dobija usrednjena slika F_g .
2. T se dobija u dva koraka:

- a) Tekstura F_g , signala detektuje se primenom LBP koda.
- b) Rezultat prethodnog međukoraka binarizuje se Otsuovom metodom.

3. I se dobija primenom Kanijevog detektora na F_g .

Provera uslova 1. i 3. na osnovu T i I je trivijalna. Sličnost tekstura podblokova detektuje se jednostavnim *template matching*-om parova susednih blokova. Rezultat R predstavlja vektor kros-korelacionih koeficijenata.

R se na kraju binarizuje pragom $t(k)$ definisanim kao $k \cdot \max(R)$. Ako preostaju nenulti elementi sa indeksom većim od polovine broja elemenata vektora, smatra se da su podbloкови slični.

7. REZULTATI

Predloženi algoritam implementiran je kao modifikacija referentnog softvera HM16.15-SCM8.5 i testiran na MPEG sekvencama klase F (*screen content*). Uslovi i upotrebljena konfiguracija koda definisani su u CTC-SCC dokumentaciji [7]. Eksperimenti su izvedeni na procesoru Intel Core i7 Q720@1.60GHzx4 sa 3.8GB RAM koristeći platformu Linux Mint 18.2.

Učinak algoritma ocenjen je metrikama $T_{\%}$ i BD -rate, dobijenim poređenjem performansi referentnog i modifikovanog koda. $T_{\%}$ definiše se kao

$$T_{\%} = \frac{T}{T_{ref}} 100\%, \quad (2)$$

gde su T i T_{ref} vremenski intervali potrebni za kodovanje sekvence referentnim odnosno modifikovanim koderom. Bjontegaard Delta-rate (BD) daje prosečnu procentualnu razliku protoka između dve RD krive [8]. Krive predstavljaju efikasnost koda u zavisnosti od kvantizacionog parametra, i određuju se interpolacijom na osnovu četiri RD vrednosti dobijene za kvantizacione parametre iz skupa {22, 27, 32, 37}.

Sekvence su prema rezoluciji podeljene u klase F1 (1920px×1080px), F2 (1280px×720px) i F3 (1024px×784px). RD cena se dobija kao prosek skupa od 10 frejmova, za svaku YUV sekvencu. Rezultati u tabeli 1. dobijeni su koderom kod kojeg koeficijent k iznosi 0.8.

Tabela 1. Prosečne vrednosti BD -rate i $T_{\%}$, $k=0.8$

	$BD(Y)$	$BD(U)$	$BD(V)$	$T_{\%}$
Klasa F1	0.80	0.95	1.09	88
Klasa F2	1.75	1.95	2.38	82
Klasa F3	0.82	1.11	1.29	88

Sa porastom k , kvalitet kodovanja opada, dok se ubrzanje koda povećava. U tabeli 2. prikazane su performanse modifikovanog koda za različite vrednosti k .

Tabela 2. Učinak koda u zavisnosti od k

	$BD(Y)$	$BD(U)$	$BD(V)$	$T_{\%}$
$k=0.80$				
F1	0.43	0.55	0.63	92
F2	1.25	1.30	1.49	92
$k=0.85$				
F1	0.80	0.95	1.09	88
F2	1.75	1.95	2.38	82
$k=0.9$				
F1	1.37	1.77	1.91	84
F2	2.43	2.86	3.67	80

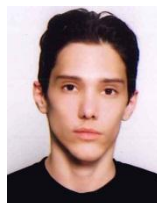
8. ZAKLJUČAK

U radu je predložen algoritam koji koristi informacije o teksturi frejmova SC sekvenci kako bi ubrzao rad HEVC koda u intra modu. Pokazano je da između teksture i podele slike na blokove postoji korelacija takva da je moguće zanemariti delove RD optimizacije uz prihvatljive gubitke kvaliteta kodovanja. Dalji razvoj bi mogao da podrazumeva optimalniji način odabira parametra k , opsežnim testiranjem različitih vrednosti ili adaptivnim pristupom korekcije u toku kodovanja. S druge strane, analizom LBP histograma bi možda bilo moguće formirati skup podataka koji bi služio za treniranje klasifikatora teksture kodnih jedinica. U opštem slučaju, poželjno bi bilo proširivanje algoritma i na inter mod.

9. LITERATURA

- [1] G. Sullivan, J. Ohm, W. Han, T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard", *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12), 2012.
- [2] J. Xu, R. Joshi, R. Cohen, "Overview of the emerging hevc screen content coding extension", *IEEE Transactions on Circuits and Systems for Video Technology*, 26(1), 2015.
- [3] M. Wien, "High Efficiency Video Coding, Coding Tools and Specification", *Springer*, 2015.
- [4] D. Marpe, H. Schwarz, T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard", *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7), 2003.
- [5] G. Sullivan V. Sze, M. Budagavi, "High Efficiency Video Coding, Algorithms and Architectures", *Springer*, 2014.
- [6] T. Ahonen, A. Hadid, M. Pietikäinen, "Face description with local binary patterns: Application to face recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), 2006.
- [7] H. Yu, R. Cohen, K. Rapaka, J. Xu, "Common test conditions for screen content coding", *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 21st Meeting: Warsaw, PL*, 2015.
- [8] G. Bjontegaard, "Calculation of average psnr differences between rd-curves", *ITU - Telecommunications Standardization Sector*, 2001.

Kratka biografija:



Stevan Filipović rođen je u Kikindi 1989. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Komunikacione tehnologije i obrada signala odbranio je 2017.god.

**KREIRANJE VEB APLIKACIJE ZA BUDŽETIRANJE TROŠKOVA U .NET
TEHNOLOGIJI KORIŠĆENJEM C# JEZIKA****CREATING A WEB APPLICATION FOR COSTS BUDGETING IN .NET TECHNOLOGY
USING C# LANGUAGE**Marko Mandić, Živko Bojović, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu je opisan koncept primene novih rešenja koja su zasnovana na .NET tehnologiji. Objašnjen je trenutni proces budžetiranja troškova u preduzećima. Detaljno je objašnjen predlog idejnog rešenja koje predstavlja razvoj aplikacije za budžetiranje troškova. Iskorišćeno je dobro poznato Microsoft-ovo razvojno rešenje Microsoft Visual Studio.

Abstract – This paper describes the concept of implementing new solutions based on .NET technology. The current process of budgeting costs in enterprises is explained. The idea of a conceptual solution that represents the development of applications for budgeting costs is explained in detail. A well-known Microsoft development solution, Microsoft Visual Studio, was used.

Ključne reči: C#, SQL baze podataka, .NET, veb aplikacija

1. UVOD

Informacione tehnologije (IT), kao pojam, imaju dva značenja. U najširem smislu, pojam informacionih tehnologija se često koristi umesto pojma računarstvo, tj. njim se označava sve što je vezano za računare. Međutim, u akademskom smislu, pojam informacionih tehnologija se vezuje za studijske programe koji pripremaju studente da uspešno zadovolje organizacije koje imaju potrebu da primene računare. Informacione tehnologije je disciplina koja se vrlo brzo razvija kao rezultat zahteva iz prakse, odnosno potreba preduzeća i drugih organizacija. Danas poslovanje u svim oblastima potpuno zavisi od informacionih tehnologija, jer sve organizacije koriste sisteme zasnovane na IT. Ovakvi sistemi moraju da rade u skladu sa definisanim potrebama, da budu pouzdani i bezbedni, da mogu da se usavršavaju, održavaju i, ako je potrebno, zamene drugim. Podrška inženjera za IT je neophodna, jer oni razumeju računarske sisteme i njihov softver [1]. Jedna od najvažnijih primena informacionih tehnologija svakako jeste razvoj i održavanje veb aplikacija. Veb aplikacije su programska rešenja kojima se pristupa putem Internet pretraživača koristeći Internet.

Motivacija za ovaj rad je u sve većoj primeni veb aplikacija u svakodnevnom životu, kao i u činjenici da nam veb aplikacije omogućavaju da mnoge naše zahteve dobijemo samo na jedan klik. U tom smislu, razvoj veb aplikacije za budžetiranje troškova jeste korak u smeru automatizacije

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Živko Bojović, docent.

velikog broja excel fajl-ova koji se trenutno koriste u procesu budžetiranja troškova u velikim kompanijama.

U ovom radu je predstavljena veb aplikacija koja omogućava unos troškova budžeta za narednu godinu. Cilj realizacije ove aplikacije je da se ručno unošenje podataka i dodavanje formula u svaki od mnoštvo excel fajlova zameni sa veb aplikacijom. Zato je kroz ovu veb aplikaciju implementirano rešenje koje omogućava sjedinjavanje svih excel fajlova u jedan kao i praćenje stanja unosa troškova koristeći tzv. „semafor“ koji je razvijen u okviru same aplikacije. Sam razvoj aplikacije je zasnovan na korišćenju napredne tehnologije .NET programiranja koja uključuje razne alate i programske jezike kao što su C#, JavaScript, jQuery, CSS. Za bazu podataka i smeštanje podataka troškova iskorišćeno je dobro poznato Microsoft-ovo rešenje MS SQL.

Veb aplikacija za budžetiranje troškova se sastoji iz četiri dela:

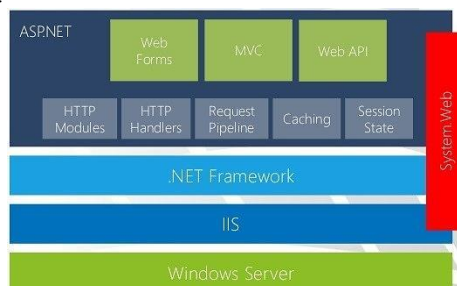
- Izrada verzije budžeta - administrator aplikacije kreira verziju budžeta koja će omogućiti korisnicima da unose troškove,
- Unos troškova – softverski modul koji pruža mogućnost unosa troškova sa svim neophodnim formulama i pretragama po različitim ključevima,
- Praćenje stanja troškova po korisniku – implementiran je semafor koji omogućava administratoru pregled stanja svih korisnika,
- Izveštaj o svim troškovima predstavljen u vidu jednog excel fajla.

2. ASP.NET

ASP.NET je okruženje za razvoj softvera, razvijano od strane Microsoft-a za Windows platforme, iOS, Android OS i uključuje veliku klasu biblioteka (*Framework Class Library*). Programi se izvršavaju kroz softversko okruženje CLR (*Common Language Runtime*) virtualnu mašinu koja sadrži: *memory management, exception handling, garbage collector*... Omogućeno je korišćenje preko 20 programskih jezika od kojih su najpopularniji C#, C++ i Visual Basic [2]. Jezici se, svaki preko svog kompajlera, kompajliraju u *Common Intermediate Language (CIL)* međujezik. Zatim, u zavisnosti od toga na kojoj se platformi izvršava, CLR kompajlira CIL u mašinski kod. Glavni razvojni alat je *Microsoft Visual Studio*.

ASP.NET je tehnologija za razvoj dinamičkih veb sajtova, interaktivnih veb aplikacija i veb servisa sa korišćenjem baza podataka za PC i mobilne uređaje. ASP.NET stranice se izvršavaju na serverskoj strani i generišu HTML, WML ili XML koji se šalje desktop ili mobilnim pretraživačima.

ASP.NET koristi 'event driven' model programiranja koji poboljšava performanse i omogućava separaciju korisničkog interfejsa od logike aplikacije. ASP.NET radi na vrhu HTTP protokola koristeći HTTP komande i pravila kako bi omogućio obostranu komunikaciju između klijenta i servera.



Slika 1. ASP.NET [2]

ASP.NET (slika 1.) omogućava 3 metode razvoja:

- *Web forms* (tradicionalni ASP.NET) – koja je korišćena u razvoju ove aplikacije
- *WEB pages*
- *MVC (Model View Controller)*

Web pages predstavlja najjednostavniji model, HTML i kod se nalaze u istom fajlu.

MVC omogućava moćan, *pattern*-orijentisan razvoj u tri sloja. *Model*: na ovom sloju je biznis logika, najčešće za komunikaciju sa bazom, *View*: omogućava prikaz i najčešće podatke iz modela, *Controller*: deo aplikacije koji čita sa *View*-a kontroliše korisničke input-e i prosleđuje input podatke *Model*-u.

Web Forms: Omogućavaju separaciju HTML-a i ostalog *UserInterface* koda od logike aplikacije, pristup podacima, moćni *data binding*, podršku za skripting sa klijentske strane kao i ostale mogućnosti kao što su rutiranje, bezbednost... Koristi programski model „event-driven“.

3. MOTIVACIJA ZA RAZVOJ APLIKACIJE

U velikim kompanijama koji sadrže nekoliko entiteta koji obuhvataju preko 1000 zaposlenih, mora se voditi računa o troškovima. Mesta troškova (*Cost Centri* u daljem tekstu, skr. CC) su podeljeni od pogona do pogona, od oblasti rada do oblasti rada. Vrlo je teško voditi računa o svim troškovima ako se podaci o troškovima ne čuvaju u nekoj bazi podataka.

Trenutni proces budžetiranja troškova u kompanijama koje koriste SAP rešenja prikazan je na slici 2.



Slika 2. Trenutni proces budžetiranja troškova

Podaci o aktuelnim ostvarenjima troškova se preuzimaju iz SAP-a. Kada se završi proces budžetiranja (kroz unos troškova u Excel), podaci o troškovima unose se u SAP za potrebe budžeta koji će se odvijati naredne godine.

SAP je najveća evropska softverska kompanija (treća po veličini u svetu iza *Microsoft*-a i *IBM*-a) koja je najpoznatija po svom ERP sistemu (*Enterprise Resource Planning*) iz 90ih godina XX veka i koji vuče korene od softvera nastalih 70-ih godina [3]. Doslovce prevedeno, ERP bi

značilo poslovno planiranje resursa međutim, ERP je ipak više od ovoga. To je zapravo poslovno rešenje ili sistem koji preduzeću omogućava da upravlja poslovnim procesima i vrši koordinaciju pojedinačnih poslovnih jedinica. SAP rešenja podržavaju baze podataka, aplikacije, operativne sisteme i hardver skoro svih proizvođača.

Dakle, kada se podaci preuzmu iz SAP-a dobija se preko 100 excel fajlova za svako mesto troška. Svako mesto troška (CC) sadrži na desetine troškovnih elemenata (*Cost Elementi*, skr. CE). Primer nekih od *Cost Centar*-a su proizvodni pogoni, prodajne jedinice, održavanje mašina, dizajn parketa itd. Na ovaj način se dolazi do imponentnog broja od preko 100 mesta troškova. Što se tiče CE, jedan od najvažnijih primera je potrošnja struje, potrošnja vode, potrošnja gasa itd. Javlja se potreba da se za svaki *Cost Centar* zaduži po jedna osoba koja će se baviti analizom troškova. Te podatke jednom godišnje svaka osoba zadužena za određeni CC unosi u jedan excel fajl. Problem sa excel fajlom je što za svako računanje mora da se odabere određena formula. A kada svaki od 100+ zaduženih CC završi sa unosom javlja se potreba da neko to sakupi kako bi menadžment mogao da vidi stanje o troškovima. Ta osoba na neki način mora da bude administrator. Najveći problem nastaje kada sve excele treba objediniti u jedan. Ali pre toga, administrator mora da zna trenutak kada objediniti podatke. Dva su tipa budžetiranja, jedno je prognoza troškova kada se radi presek stanja i u nekom određenom mesecu neophodno je budžetirati preostale mesece do kraja godine, a drugi tip je budžetiranje za svaki mesec naredne godine (budžet za narednu godinu).

Ovakvi problemi su glavni motiv koji vodi ka razvoju ideje da se napravi jedna veb aplikacija koja bi za svakog nosioca ili vlasnika CC omogućila unos troškova na veb stranici pri čemu bi aplikacija u potpunosti zamenila staru tehnologiju poput *Microsoft*-ovog paketa *Microsoft Office Excel* (slika 3.). Prilikom unosa podataka svaki nosilac CC nema potrebu da razmišlja o formulama koje su mu neophodne da bi izvršio proces budžetiranja. Veliki benefit aplikacija bi dala što bi sve podatke smeštala u samo jednu tabelu. U tom slučaju administrator na samo jedan klik dobija vrednosti troškova za sva mesta troškova. Administrator kroz aplikaciju obaveštava nosioce CC da je verzija za budžetiranje kreirana tako što svakom od nosioca šalje *e-mail* sa linkom aplikacije.



Slika 3. Idejni proces budžetiranja troškova

Ovaj rad kao svoj cilj prikazuje jedan moderan pristup i proces razvoja jedne vrlo funkcionalne veb aplikacije. Proces koji opisuje funkcionalnost aplikacije, u ovom slučaju je unos i održavanje troškova budžeta u bazu podataka.

Glavni motiv ovog rada je pomoć korisnicima da prilikom unosa troškova u svakom trenutku na jedan klik mogu da dobiju objedinjene excel fajlove kao i status svakog excel fajla, dakle sve podatke na jednom mestu. Razvoj jedne takve aplikacije koja koristi i spaja više različitih veb tehnologija, koja ima mogućnost višestruke implementacije i nadogradnje te nudi potpuno siguran, bezbedan sistem.

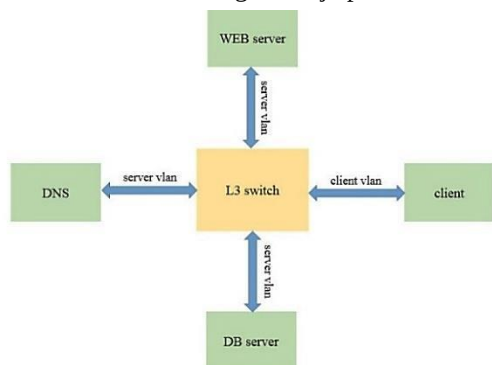
4. IMPLEMENTACIJA REŠENJA

4.1. Komunikacioni blok

Što se tiče komunikacionog bloka infrastruktura na kojoj je instalirana aplikacija je virtualizovana, a koristi se VMware-ovo rešenje za virtualizaciju. To znači da su svi serveri virtualni odnosno instaliran je gost operativni sistem na virtualni hardver, samim tim mu je dodeljena i virtualna mrežna kartica kao i statička privatna IP adresa [4]. IP adrese zavise od VLAN-a u kojem se nalaze. VLAN je virtualni LAN koji se kreira na VMware-u i predstavlja se server kao običan LAN. To znači da ukoliko se serveri nalaze u istom VLAN-u mogu da komuniciraju međusobno zato što imaju IP adrese iz istog pool-a i iste subnet adrese. Konkretno komunikacija od jednog servera ka drugom može da se odvija na više načina:

- Direktno korišćenjem IP adrese (ukoliko je preko IP adrese zahtev će da se rutira do sledećeg servera sa tom adresom u istom VLAN-u)
- Indirektno korišćenjem host name-a (svaki server ima svoj host name (ime koje se koristi za konekciju na server) za čije se razrešavanje u IP adresu koristi DNS server (na njemu se nalaze svi zapisi sa host name-ovima i njihovim IP adresama) koji klijentu odgovara na njegov zahtev i dostavlja odgovarajuću adresu.
- Preko alias-a (za određeni server možemo da kreiramo alias na DNS serveru i na taj način je moguće pristupiti serveru)

Blok šema komunikacionog bloka je prikazana na slici 4.



Slika 4. Komunikacioni blok veb aplikacije

Kao što se vidi na slici 4, jezgro komunikacije je L3 multilayer switch koji radi i kao ruter, odnosno koji rutira saobraćaj između client VLAN-a i server VLAN -ova. Svi uređaji su zakačeni na switch. Postoji i L2 switch koji rutira saobraćaj na osnovu MAC adresne tabele i odgovarajućih zapisa koji se odnose na MAC adrese. Dakle, u opisanom sistemu postoji više VLAN-ova i za razmenu saobraćaja između njih potrebno je uraditi tzv. intervlan rutiranje, a zaštita se koristi L3 switch. Svaki fizički interfejs na koji je priključen neki uređaj dodat je u neki VLAN. Switch ne zna ime da razreši u IP adresu i za to se koristi DNS server.

Klijent šalje zahtev WEB serveru preko DNS servera kroz svoj VLAN (client vlan) koji je povezan sa serverskim VLAN -om (komuniciraju preko gateway-a). Web server dobija request i šalje upit bazi (DB server) preko DNS servera i server vlan-a, DB server odgovara istim putem i vraća na WEB server koji zatim klijentu šalje podatke koje je zahtevao.

4.2. Temeljne funkcionalnosti veb aplikacije

Funkcionalnost razvijene veb aplikacije je prikazana na slici 5.



Slika 5. Glavne funkcije aplikacije

Može se podeliti u 4 glavne celine:

1. Aplikacija preko svoje početne stranice omogućava administratru da kreira verziju budžeta
2. Korisnici aplikacije unose troškove za svaki mesec od ukupno 12 meseci ili ako je verzija za prognozu onda korisnici unose samo poslednjih par meseci vrednosti za troškove
3. Administrator može da prati rad korisnika preko realizovanog semafora
4. Svi korisnici dobijaju kroz aplikaciju objedinjen excel fajl

Navedeni celine čine osnovnu funkcionalnost veb aplikacije. Vrlo je važno istaći da postoje dve grupe korisnika aplikacije. To su administrator i svi ostali korisnici. Administrator je posebna grupa koja ima najveća prava pristupa nad aplikacijom, a korisnici samo deo aplikacije gde se očekuje njihova operativnost i izvršnost.

Za potrebe pretrage kroz veb stranice, dostupno je Microsoftovo rešenje CTRL+F. Administrator kroz aplikaciju dodeljuje razna prava korisnicima. Korisnici prilikom unosa troškova mogu da koriste i automatski unos prema ključevima koji se nalaze u bazi podataka a koji su korisnicima na raspolaganju čim pristupe određenoj veb stranici. Odabirom jednog od nekoliko ključeva moguće je automatski raspodeliti troškove za svih 12 meseci bez ručnog unosa svakog od meseci.

4.3. Pregled najvažnijih veb stranica

Najvažnije veb stranice koje su razvijene u okviru ovog projekta su:

- Troškovi
- ForecastAndBduget
- NestedGridView
- Monitoring

Veb stranica Troškovi

Ova stranica je razvijena za administratora aplikacije. Administrator odabira vrstu unosa troškova i kreira verziju. U okviru fajla *Troskovi.aspx* nalaze se nekoliko komponenti od kojih su najvažniji *button*, *label*, *dropdown* lista. U okviru ove stranice razvijeno je nekoliko metoda pisanih u programskom jeziku C# kao i nekoliko procedura pisanih u SQL programskom jeziku.

Veb stranica ForecastAndBudget

Na ovoj stranici se nalaze *hyperlink*-ovi kreiranih verzija za unos troškova budžeta. Stranica je razvijena da bi kori-

snici imali prava pristupa nad njom, a naravno i administrator. Korisnici biraju verziju za koju žele da unose troškove i na taj način započinje prvi korak unosa troškova.

Veb stranica *NestedGridView*

Ključna i najvažnija stranica u celoj aplikaciji a ujedno i najkompleksnija. Na ovoj stranici prava pristupa imaju i korisnici i administrator. Veb stranica koja omogućava korisnicima da unose troškove, da vrše razne pretrage po mestima troškova. Korisnici kroz ovu stranicu mogu da vrše objedinjavanja nekoliko troškovnih elemenata u jedan. Omogućeno je automatsko izračunavanje preko *JavaScript* funkcija. Moguće je unositi troškove za svako mesto troška pojedinačno za svaki mesec ali i automatski po ključu za sve mesece odjednom. Ključ sadrži predefinisane podatke za raspodelu troškova svih meseci. Primera radi, jedan ključ može da označava linearnu raspodelu po svim mesecima pri čemu korisnik unosi samo jednu vrednost koja se linearno raspodeljuje na sve mesece. Administrator na ovoj stranici može da zaključa/otključa određenu verziju.

Veb stranica *Monitoring*

Na ovoj veb stranici je realizovan semafor. Korisnici nemaju prava da pristupe ovoj stranici, administrator je taj koji vrši monitoring unosa troškova. U pitanju je klasičan semafor sa tri boje: crvena, žuta i zelena boja. Crvena boja označava da korisnik nije ni krenuo da unosi troškove, žuta boja administratoru govori da je u toku unos troškova, dok zelena boja kazuje da je korisnik uneo troškove. Za korisnike nosioce mesta troška kod kojih semafor daje crveno i žuto dugme u svakom trenutku administrator može da pošalje mejl kroz aplikaciju u kojem će ih podsetiti o krajnjem roku unosa troškova. Administrator može da vidi sva mesta troška, može da vidi po određenoj kategoriji kao i po pogonu.

5. POSTUPAK TESTIRANJA I REZULTATI

Nakon razvoja veb aplikacije izvršeno je nekoliko testiranja sa ciljem provere ispunjenosti zadatka. Testiranje je obuhvatilo nekoliko celina:

- Keiranje verzije za unos troškova
- Slanje *e-mail*-a obaveštenja kroz aplikaciju
- Prava pristupa korisnicima kada dobiju link aplikacije
- Unos troškova
- Prikazivanje *pop-up* prozora kada korisnik želi automatski unos podataka za sve mesece odjednom
- Automatski unos podataka
- Prikaz *pop-up* dugmeta kada korisnik želi da validira unete podatke
- Prikaz semafora kada administrator pristupi stranici *Monitoring*
- Prikaz kategorija i pogona u padajućim listama na stranici *Monitoring*
- Slanje mejla podsetnika kroz aplikaciju

Svi testovi su dali očekivane rezultate u skladu sa traženim zahtevima.

6. ZAKLJUČAK

U ovom radu predstavljen je jedan moderan pristup rešavanju problema prilikom procesa budžetiranja troškova. Kreirana je veb aplikacija koristeći napredne tehnologije kao što je *Microsoft*-ovo rešenje u vidu *.NET* tehnologije. Objašnjeni su i predstavljeni svi metodi i korišćeni alati prilikom razvoja aplikacije. Aplikacija je razvijena u komercijalne svrhe za veoma veliki broj korisnika koji su je sa oduševljenjem prihvatili.

Razvoj ove aplikacije je omogućio korisnicima vrlo brz, moderan i bezbedan pristup rešavanju problema. Brzina aplikacije se ogleda u tome što se svaki proces odvija samo na jedan klik. Aplikacija je automatizovala sve neophodne korake koji su do tada korišćeni i omogućila sve podatke na jednom mestu. Pružila je odličan monitoring u vidu semafora i drastično smanjila šanse za nastanak grešaka. Nakon završetka razvoja aplikacije pristupilo se testiranju.

Što se tiče daljeg razvoja aplikacije, naredni koraci bi svakako bili da se u postojeću aplikaciju razvije i integriše budžetiranje zarada na sličan način kako je to predstavljeno u ovom radu gde je glavna funkcionalnost budžetiranje troškova. Tendencija je da će ovakvih aplikacija na klik biti sve više u budućnosti koja je pred nama.

7. LITERATURA

- [1] Informacione tehnologije, online:
<http://www.metropolitan.ac.rs/files/2017/09/oas-informacione-tehnologije-2017.pdf>
- [2] *.NET*, online:
<https://raf.edu.rs/citaliste/programiranje/4079-sta-je-xa-net-xa>
- [3] *ERP*, online:
<https://beleske.com/sta-je-erp-znacaj-erp-resenja-u-poslovanju-preduzeca/>
- [4] VMware, „*Infrastructure Architecture Overview*“, 2006

Kratka biografija:



Marko Mandić rođen je u Novom Sadu 1990. god. Fakultet tehničkih nauka u Novom Sadu je upisao 2009. godine. Ispunio je sve obaveze položio sve ispite predviđene studijskim programom. Bsc diplomu stekao je 2014. godine na usmerenju Mikroračunarska elektronika.



Živko Bojović rođen je u Peći 1967. god. Završio je Elektrotehnički fakultet u Prištini, i magistrsku tezu iz oblasti obrade govornog signala i doktorsku disertaciju na temu testiranje kvaliteta servisa u živoj IP MPS mreži, odbranio je na Fakultetu tehničkih nauka u Novom Sadu gde od 2015. god. radi u zvanju docenta.

PRIMJENA NETWORK PROTECTORA U UPETLJANIM DISTRIBUTIVNIM MREŽAMA I MREŽAMA SA DISTRIBUTIVNIM GENERATORIMA**APPLICATION OF NETWORK PROTECTORS IN MESHED DISTRIBUTION NETWORKS AND NETWORKS WITH DISTRIBUTION GENERATION**

Ilija Pantić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu je predstavljena primena network protectora u upetljanim distributivnim mrežama i mrežama sa distributivnim generatorima, kao jednog od najvažnijih mrežnih zaštitnih uređaja za sprečavanje obrnutog toka snage. Verifikacija njihove primene i simulacija rada realizovana je na test mreži od 15 čvorova.

Abstract – This paper presents application of network protectors in meshed distribution networks and networks with distribution generation, as one of the most important protective network device for preventing from reverse power flow. Verification of their application and simulation is realized on the test case of networks consisting of 15 nodes.

Cljučne reči: network protectori, distributivni generatori, upetljane distributivne mreže

1. UVOD

Poslednjih decenija napredak tehnologije doveo je do povećane potrebe za proizvodnjom električne energije, pa se elektroenergetski sistemi usložnjavaju. U isto vrijeme, količina fosilnih goriva se smanjuje, a fokus razvoja se prebacuje na instaliranje distribuiranih izvora energije – distributivnih generatora. Potrošačka područja sve više zavise od kontinuiteta distribucije električne energije, pa se zahtjeva konstantno napajanje i povećavanje sigurnosti i pouzdanosti sistema. U tu svrhu dolazi do upetljavanja distributivnih mreža. Integracija network protectora kao važnog zaštitnog uređaja za sprečavanje obrnutog toka snage jedan je od najvećih izazova u analizi zaštite mreža.

2. KONCEPTI DISTRIBUTIVNIH MREŽA U SVIJETU

Distributivne mreže distribuiraju električnu energiju od distributivnih transformatorskih stanica do krajnjih potrošača. Veliki broj potrošača, glomaznost mreže i ekonomska neisplativost upetljavanja doprinijeli su opredjeljenju za koncept radijalnosti distributivnih mreža. Takođe, u početku je proizvodnja bila izmještena, pa su distributivne mreže bile pasivne.

2.1. Upetljavanje distributivnih mreža

Ipak, potrebe za konstantnim napajanjem i povećanjem pouzdanosti dovele su do kasnijeg upetljavanja distributivnih mreža. Neke od prednosti u odnosu na konvencionalne radijalne mreže su [1]:

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Duško Bekut, red. prof.

- Poboljšavanje naponskih prilika u mreži,
- Potrošači su napajani iz više smjerova, što smanjuje mogućnost ispada,
- Dinamičke promjene u mreži (npr. pokretanje asihronih motora) ne uzrokuju veće poremećaje,
- Smanjuju se vršna opterećenja transformatora u upetljanim mrežama.

Instaliranjem distributivnih generatora narušava se i koncept pasivnosti, pa se mreže dodatno komplikuju. Ovakve mreže zahtjevaju dodatne analize i modifikovanje postojećih proračuna električnih veličina. Generalno, postoje dva tipa konfiguracija distributivnih mreža u svijetu: evropske i sjevernoameričke.

2.2. Slaboupetljane distributivne mreže

Evropske slaboupetljane mreže se najčešće izvode na sredjenaponskim nivoima, u obliku prstena. Na niskonaponskom delu distributivne mreže ređe dolazi do upetljavanja, mada se ta praksa sve češće primjenjuje. Američke slaboupetljane mreže se odlikuju različitošću konfiguracija, a slaba upetljanost je jedna od glavnih njihovih karakteristika. Postoje dvije osnovne konfiguracije upetljenih mreža: grid (eng. *Grid Networks*) i spot (eng. *Spot networks*) distributivne mreže [1]. Obje mreže su otporne na kvarove unutar mreže i na jednom od fidera, jer je mreža tako dimenzionisana da ostali fideri mogu da preuzmu opterećenje fidera u ispadu, dok se kvarovi unutar mreže otklanjaju prekidačima ili osiguračima. U slučaju kvara na primaru transformatora koji napajaju ove mreže može da dođe do pojave obrnutog toka snage, jer se mjesto kvara napaja preko NN mreže, sa drugih fidera.

3. DISTRIBUTIVNI GENERATORI I PRIMJENA U DISTRIBUTIVNIM MREŽAMA

Osnovno opredjeljenje za instaliranje distributivnih generatora odnosi se na zaštitu životne sredine i smanjivanje upotrebe fosilnih goriva.

Dodatno, uvođenjem distributivnih generatora se oslobađa prenosni kapacitet, čime se povećava faktor efikasnosti mreža.

Takođe, smanjuju se gubici u prenosu energije a samim tim i troškovi eksploatacije. Naponski profil u distributivnim mrežama je povoljniji a pouzdanost i kvalitet snabdijevanja krajnjih potrošača je na znatno višem nivou, pri čemu se mogu zadovoljiti i pikovi potrošnje.

3.1. Tipovi distributivnih generatora

Osnovna podjela distributivnih generatora odnosi se na vrstu obnovljivih izvora koji služe za pogon generatora, pa se dijele na:

- vjetrogeneratore,
- solarne generatore,
- mikro hidrogeneratore,
- generatore na geotermalni pogon,
- generatore na pogon biomase (biodizel, biogas...),
- kogeneracione generatore.

Vrsta električne mašine, način instalacije, cijena instalacije i snaga se dosta razlikuju u zavisnosti od tipa.

3.2. Prednosti i mane distributivnih generatora

U analizi opravdanosti instalacije distributivnih generatora moraju se jasno predstaviti potencijalne prednosti i mane. Prednosti distributivnih generatora su:

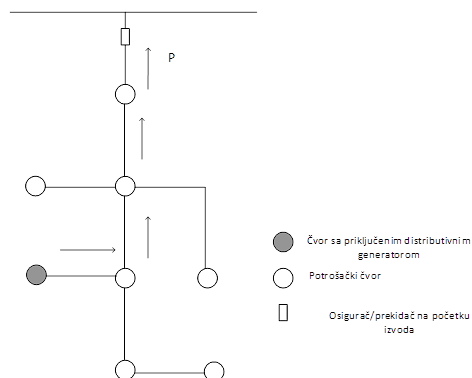
- Pokrivaju pikove potrošnje,
- Smanjuju ili odlažu investicije u prenosne elemente,
- Lako se instaliraju u najrazličitijim uslovima,
- Pozitivno utiču na naponske prilike,
- Jednostavno se proširuje proizvodnja,
- Mogu da troše višak reaktivne energije.

Mane su uglavnom vezane za promenu konfiguracije mreža i uticaj na relejnu zaštitu:

- Promene vrednosti struja kratkog spoja,
- Smanjen doseg distantnih releja,
- Greške u koordinaciji zaštite,
- Problemi sa APU,
- Pojava obrnutog toka snage.

3.3. Pojava obrnutog toka snage u distributivnim mrežama

Pojaava obrnutog toka snage nastaje kada generatori proizvode veću snagu od one koja je potrebna tom potrošačkom području. U literaturi se ravnopravno koriste izrazi „suprotan tok snage“ i „snaga suprotnog smijera“ [2]. Ono što predstavlja najveći izazov je usklađivanje i praćenje potrošnje i proizvodnje i držanje u specificiranim granicama. Na slici 1 prikazana je distributivna mreža sa potrošačima i generatorom i strelicom je obeležen tok aktivne snage usled povećane proizvodnje generatora.



Slika 1. Distributivna mreža sa obilježenim obrnutim tokom aktivne snage

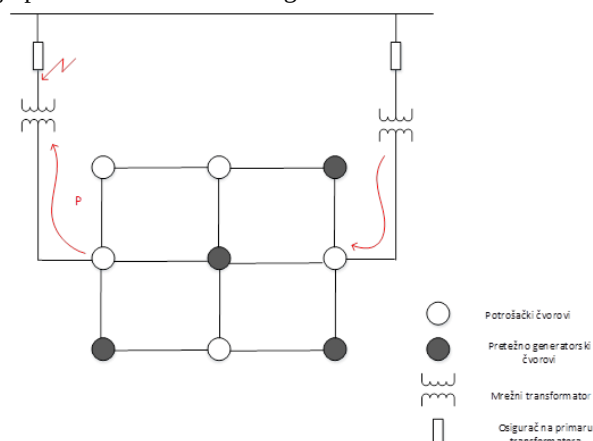
U analizi uticaja obrnutog toka snage definiše se najgori mogući scenario (The Worst-Case Scenario), gdje su sve potrošnje na minimumu a proizvodnje generatora na maksimumu [3]. Posljedice obrnutog toka snage su:

- Poremećaj koordinacije i rada relejne zaštite u prenosnoj mreži
- Preopterećenje elemenata distributivne i prenosne mreže
- Promena naponskih prilika
- Prenosnje harmonika u prenosnu mrežu
- Formiranje potrošačkih ostrva

Distributivni generatori unose problem sa proračunima električnih veličina i otežavaju praćenje dijagrama potrošnje.

4. NETWORK PROTECTORI I NJIHOVA PRIMJENA U DISTRIBUTIVNIM MREŽAMA

Najefikasniji mrežni uređaj za detekciju i sprečavanje obrnutog toka snage je network protector. Na slici 2 prikazana je uprošćena šema slaboupetljane distributivne mreže sjevernoameričkog tipa, tzv. grid mreže. Na primaru transformatora je iniciran kratak spoj i strelicom je prikazan tok aktivne snage.



Slika 2. Uprošćena šema niskonaponske grid mreže sa kratkim spojem na primaru transformatora

Ukoliko se desi kratak spoj na primaru transformatora ili primarnim sabirnicama i primarni prekidač odreaguje, mesto kvara nastavlja da se napaja sa drugog fidera, preko NN mreže. Ovakvo stanje je opasno po opremu unutar mreže, a može i da dovede do pojave opasnih napona pa se kvar mora eliminisati.

Zbog toga se na sekundar transformatora postavlja network protector koji će u ovom slučaju detektovati obrnut tok aktivne snage i odvojiti transformator od NN mreže. Iz prethodnih razmatranja se zaključuje da network protector služi za sprečavanje obrnutog toka snage usled:

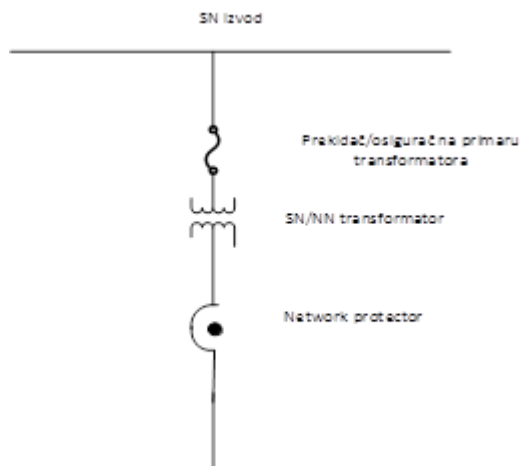
- Povećanja proizvodnje distributivnih generatora.
- Kvara na primaru transformatora slaboupetljane mreže.

4.1. Električna prezentacija network protectora

Network protectori su specijalni mrežni uređaji koji se sastoje od:

- Specijalne prekidačke opreme (prekidač ili rastavljač),
- Mehanizma za manipulaciju,
- Releja za detektovanje obrnutog toka snage,
- Kontrolne opreme.

Na slici 3 prikazan je primjer postavljanja *network protectora* na sekundaru transformatora.



Slika 3. Primjer postavljanja *network protectora*

Imaju potpunu kontrolu nad strujom i snagom transformatora, kao i funkciju relejne zaštite koja se sastoji od zaštite sekundara i distributivne mreže na istom. Nominalno, oni nemaju prekostrujnu zaštitu za direktnu komponentu, ali se ista može integrisati, ako ima potrebe.

4.2. Struktura network protectora

Fizički, kućišta u koja su smješteni *network protectori* mogu biti potopljena i nepotopljena. Obje vrste kućišta mogu se montirati direktno na sekundare transformatora ili na zidove, u zavisnosti od vrste transformatora. Kao osnovna prekidačka oprema u sklopu protectora koriste se vazdušni prekidači ili rastavljači. Vazdušni prekidači su sigurniji ali i skuplji, dok rastavljači mogu da se koriste u kombinaciji sa osiguračima.

4.3. Relejna zaštita network protectora

Releji koji su integrisani u *network protectore* da detektuju obrnut tok snage su „releji povratne snage“ ili „releji obrnutog toka snage“ [2]. Dat je izraz koji opisuje parametre releja:

$$U \cdot I \cdot k \cdot \cos(\varphi + \alpha + \beta) \geq Mk \quad (1)$$

gde je:

- U, I – napon i struja koji se dovode na relej, respektivno
- φ – fazni pomeraj - ugao između napona i struje,
- α – unutrašnji ugao releja,
- k – konstanta releja,
- β – fazni pomeraj.

Za slučaj releja povratne snage $\beta = -\alpha + 180^\circ$, pa se dobija da je $\cos(\varphi + 180^\circ) = -\cos \varphi$. Kritičan je i jednopolni kratak spoj sa zemljom na primaru, koji se često ne može detektovati primarnim prekostrujnim prekidačima, pa se mora integrisati i relej nulte komponente struje. Takođe,

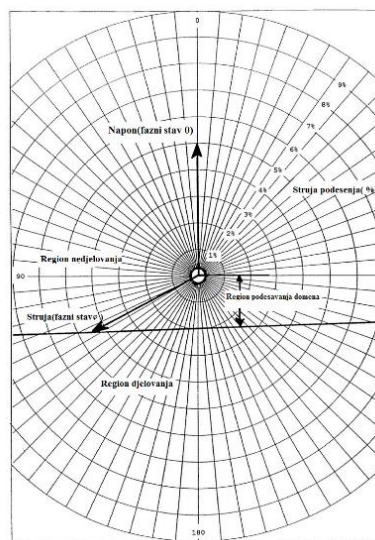
kada je transformator deenergizovan, procedure za vraćanje u pogon se moraju definisati na osnovu očitavanja naponskog i faznog releja. Svi ovi releji su ranije bili izvedeni kao elektromehanički dok danas mikroprocesorski releji objedinjuju sve funkcije. Sve prekidačke akcije moraju biti strogo nadgledane i kontrolisane, a u tu svrhu se vrši automatizacija mreža i integrisanje SCADA uređaja. Podešavanja releja se vrše na osnovu jedne od četiri trip funkcije:

- Senzitivna trip funkcija,
- Vremenski odložena trip funkcija,
- Nesenzitivna trip funkcija,
- Watt-War trip funkcija.

Parametri koji figurišu u podešavanju releja su:

- Fazni stav i intenzitet napona,
- Fazni stav i intenzitet struje,
- Region djelovanja releja,
- Region nedjelovanja releja,
- Domet djelovanja releja koji se podešava.

Na slici 4 dat je dijagram podešenja djelovanja *network protectora*.



Slika 4. Dijagram podešenja delovanja *network protectora*

5. ANALIZE PRORAČUNA TOKOVA SNAGA U SLABOUPETLJANIM DISTRIBUTIVNIM MREŽAMA I MREŽAMA SA DISTRIBUTIVNIM GENERATORIMA

Da bi se u potpunosti shvatila uloga i način delovanja *network protectora*, mora da se postavi jedinstven model proračuna i da se izvrši modifikovanje postojećih algoritama za najvažnije proračune u distributivnim mrežama.

Proračun simetričnih tokova snaga se sastoji od proračuna promenljivih stanja distributivne mreže, na bazi poznatog izvora napajanja mreže (korena) i poznatih potrošnji u svim čvorovima. Standardne metode za proračun tokova snaga u prenosnim mrežama imaju znatno manju efikasnost kada se primenjuju u distributivnim mrežama. Numerički postupci koji su bazirani na matričnom pristupu postaju neefikasni u rešavanju problema

proračuna tokova snaga u distributivnim mrežama.

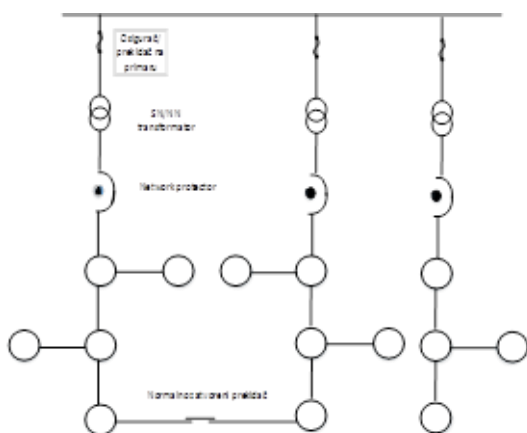
Da bi se rešili problemi primene algoritama za prenosne mreže u distribuciji, razvijeni su specijalizovani algoritmi za proračun tokova snaga u elektrodistributivnim mrežama [4]:

- Algoritmi sumiranja struja,
- Algoritmi sumiranja snaga,
- Algoritmi sumiranja admitansi,
- Kombinovani algoritmi.

U praksi su se najbolje pokazali algoritmi sumiranja struja i to Shirmohammadi-jev algoritam, koji se sa sastoji od postupaka sumiranja struja, korekcija napona i provere kriterijuma konvergencije [4]. Ipak, ovaj algoritam se koristi samo za radijalne mreže, pa se u slaboupetljivim mrežama mora pristupiti njegovoj modifikaciji i tako se dobija kompenzovani algoritam [5].

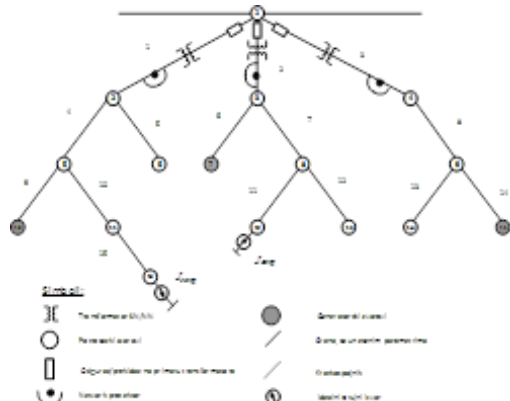
6. SIMULACIJA RADA NETWORK PROTECTORA NA OSNOVU PRORAČUNA TOKOVA SNAGA

Da bi se u potpunosti analiziralo djelovanje network protectora, moraju se izvršiti proračuni tokova snaga na referentnoj test mreži, koja je prikazana na slici 5.



Slika 5. Referentna test mreža

Test mreža se sastoji od 14 potrošačko-generatorskih PQ čvorova. Mreža je trofazna niskonaponska, blagoupetljana sa jednom petljom. Na sekundaru transformatora su postavljeni network protectori i na osnovu injektiranja snaga distributivnih generatora treba odrediti da li će oni reagovati ili ne. Na slici 6 prikazana je raspeljana kompenzovana mreža, u vidu stabla.



Slika 6. Kompenzovana mreža prikazana u vidu stabla

Ispis detalja simulacije rada u slučaju kada je proizvodnja svih generatora veća od potrošnje dat je na slici 7.

```
Network protector ce odreagovati u grani:      1
Smanjite snagu generatora
=====
Network protector ce odreagovati u grani:      2
Smanjite snagu generatora
=====
Network protector ce odreagovati u grani:      3
Smanjite snagu generatora
=====
```

Slika 7. Simulacija rada network protectora za slučaj povećane proizvodnje generatora

Za slučaj smanjene proizvodnje generatora, network protector neće odreagovati i dobija se ispis kao na slici 8.

```
Network protector neće odreagovati u grani:     1
=====
Network protector neće odreagovati u grani:     2
=====
Network protector neće odreagovati u grani:     3
=====
```

Slika 8. Simulacija rada network protectora za slučaj smanjene proizvodnje generatora

7. ZAKLJUČAK

Snage generatora i snage potrošača u upetljivim distributivnim mrežama moraju da se konstantno kontrolišu. Odabir instalacije distributivnih generatora kao i njihova instalisana snaga zavise najviše od analize mana i prenosti koje oni donose.

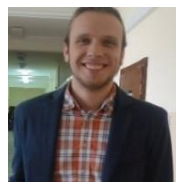
Pri tome se treba voditi računa da integrisanje network protectora odgovara unapred urađenim proračunima i mjerenjima u mreži.

U ovakvim mrežama na izboru kad će network protector reagovati najviše zavisi od usvojenih smernica samog distributivnog preduzeća, pa se prema tome i bira odgovarajuće podešenje releja.

8. LITERATURA

- [1] T.A.Short, *Electric Power Distribution Handbook*, Second Edition, 2014.
- [2] D.Bekut, *Relejna zaštita*, Fakultet Tehničkih nauka, Novi Sad, 2013.
- [3] A.Keane, M.O'Malley, *Optimal utilization of distribution networks for energy harvesting*, IEEE Trans. Power Syst., vol. 22, no. 1, pp. 467-475, Feb. 2007.
- [4] D.Popović, D.Bekut, V.Treskanica, *Specijalizovani DMS algoritmi*, Novi Sad, DMS Grupa, 2004.
- [5] D.Shirmohammadi, H.W.Hong, A.Semlyen, G.X.Luo, *A Compensation-Based Power Flow Method for Weekly Meshed Distribution and Transmission Network*, IEEE Trans. PS, Vol.3, No.2, May 1988, pp. 753-762

Kratka biografija:



Ilija Pantić rođen je u Milićima 11.11.1992. god. Srednju školu, završio je 2011. god. u Vlasenici. Diplomski rad na Fakultetu tehničkih nauka, iz oblasti Elektrotehnike i računarstva odbranio je 2015. god.

GENERATOR KODA ZA SISTEME ZA DINAMIČKO UPRAVLJANJE SADRŽAJEM CODE GENERATION FOR CONTENT MANAGEMENT SYSTEMS

Dragoljub Ilić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je predstavljena implementacija generatora koda koja predstavlja osnovu za dalji razvoj aplikacije u okviru sistema za dinamičko upravljanje sadržajem u okviru Kentico platforme.

Abstract – This thesis presents implementation of a code generator which is the basis for further development of the application within content management system by using Kentico platform.

Ključne reči: Content Management Systems (CMS), Code Generation

1. UVOD

U današnje vreme informacionim tržištem polako preovladavaju sistemi za dinamičko upravljanje sadržajem. Ukoliko internet prezentacija predstavlja neku oblast koja se često menja i razvija, i kada je neophodno dopunjavati njen sadržaj, odabir ovog sistema kao platforme je očekivan. Sve je veća potražnja za novim sistemima koji korisnicima olakšavaju korišćenje, i samim tim daju utisak potpune kontrole. Dizajniran je tako da se u njemu mogu snalaziti prosečni korisnici.

Iako su ovakvi sistemi jednostavni za upotrebu i pružaju korisnicima osećaj lagodnosti, veoma su zahtevni sa stanovišta razvoja. Krajnji korisnici ne mogu menjati komponente koje čine strukturu i organizaciju njihovog proizvoda, već prvenstveno upravljati tekstualnim i vizuelnim sadržajem. Upravo iz tog razloga veliku pažnju treba posvetiti organizaciji i strukturi sadržaja pre početka implementacije, kako krajnji korisnici ne bi imali problema u budućem razvoju. Jednom kvalitetno odrađen sistem zahteva jednostavno i lako održavanje.

Kao što krajnji korisnici teže ka što jednostavnijoj upotrebi sistema, tako i inženjeri, zaduženi za njegov razvoj, teže ka što bržoj i kvalitetnijoj izradi istog. U velikoj meri na kvalitet i brzinu utiču generatori koda. Generatori koda smanjuju redundantnost i vreme izrade, ali povećavaju nivo apstrakcije. Prilikom implementacije generatora, neophodno je ostaviti mogućnost naknadne dorade, u zavisnosti od konkretnog sistema koji predstavlja predmet generisanja. Generičkim programiranjem se postiže optimizacija programskog rešenja.

Glavni cilj ovog rada jeste projektovanje i implementacija generatora koda koji je namenjen prvenstveno ljudima koji rade na implementaciji samog sistema.

Generator, kao jedinka, ne predstavlja čitav sistem, već samo jedan njegov deo čija je namena dobavljanje sadržaja iz sistema implementiranog korišćenjem MVC modela. Način dobavljanja je identičan za svaki sistem, s tim što se menjaju gradivne komponente, kao i određeni delovi sistema koji nisu od velike važnosti za samu implementaciju, ali jesu za kvalitet kao i za lakše održavanje samog sistema u budućim iteracijama.

2. TEORIJSKE OSNOVE

CMS (Content Management System) je sistem za dinamičko upravljanje sadržajem [1]. CMS predstavlja sistem uz pomoć kojeg se može, bez poznavanja programiranja, upravljati sadržajem. Osnovne osobine svakog CMS – a su:

- Korisnički interfejs za prikaz sadržaja
- Veb interfejs za administraciju
- Odvojenost sadržaja i interfejsa
- Sadržaj se čuva u bazi podataka

Što se tiče funkcionalnosti, one variraju u zavisnosti od CMS – a, ali neke osnovne funkcionalnosti koje su zajedničke za većinu sistema su:

- Intuitivne funkcije indeksiranja, pretraživanje i dobavljanje indeksiranog sadržaja na osnovu ključnih reči, kao i dodatnih atributa koje specificira korisnik
- Funkcije upravljanja sadržajem naknadno, nakon početne publikacije, kao i kontrola i istorija upravljanja sadržajem
- Funkcije za objavljivanje sadržaja, koje omogućavaju korišćenje određenih skupova šablona i komponenti, prethodno definisanih i odobrenih od strane organizacije, kao i korišćenje određenih alata za izmenu sadržaja
- SEO (Search Engine Options) – adresa do sajta je laka i prepoznatljiva za korisnika [2]
- Hijerarhija i organizacija sadržaja
- CRM (Customer Relationship Management), poznatiji kao 1:1 marketing, koji predstavlja strategiju zasnovanu na personalizaciji sadržaja [3]

CRM predstavlja sposobnost internet prezentacije da prilagođava sadržaj i reklamiranje korisniku na osnovu podataka, koje čine specifične karakteristike samog korisnika, prikupljenih od strane same prezentacije.

NAPOMENA:

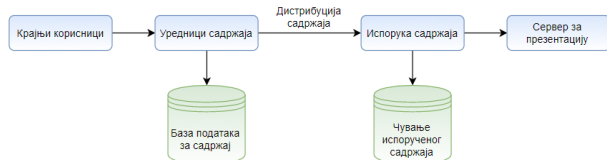
Ovaj rad proistekao je iz master rada čiji mentor je bila Dr Gordana Milosavljević

CMS je širok pojam i mnoštvo drugačijih tipova sistema se svrstavaju pod isti pojam. Dele se na više kategorija, a svaka od njih je stvorena kako bi se rešio neki specifičan problem. Godinama se razvijalo rešenje oko problema koji je naknadno prerastao u čitavu oblast. Ovo su neke od opšte prihvaćenih oblasti:

- **WCM (Web Content Management)** – sistem za upravljanje javnim sadržajem
- **ECM (Enterprise Content Management)** – sistem za upravljanje skeniranjem, skladištenjem, pretraživanjem i dobavljanjem fizičkih dokumenata. ECM olakšava integraciju za upravljanje dokumentima, upravljanje digitalnim sredstvima i arhiviranje evidencije pružajući krajnjim korisnicima pristup digitalnim sredstvima
- **DAM (Digital Asset Management)** – sistem za upravljanje širokim spektrom digitalnih sredstava. Medijske kuće često koriste ove sisteme da katalogizuju, anotiraju, čuvaju, preuzimaju i distribuiraju audio, video, animacije i druge medijske sadržaje
- **DXP (Digital Experience Platform)** – sistem za pravljenje sadržaja za svaki digitalni kanal dostupan u današnje vreme, kao i za poboljšanje poslovanja kroz digitalizaciju i integraciju

ECM i WCM sistemi se sastoje iz dve komponente (vidi sliku 1):

- **CMA (Content Management Application)**
- **CDA (Content Delivery Application)**



Slika 1. Prikaz veze između CMA i CDA

CMA je grafički korisnički interfejs koji omogućava korisniku da kontroliše kreiranje, modifikaciju i uklanjanje sadržaja sa veb stranice bez potrebe da poseduje osnovno znanje o razvoju softvera. Ovi sistemi obično uključuju šablone koji automatizuju mnoge komponente koje se ponavljaju prilikom dizajniranja. Sve više su popularniji takozvani WYSIWYG (*What You See Is What you Get*) koncepti. Oni omogućavaju korisnicima da direktnim pristupom manipulišu izgledom sadržaja i šablona same veb stranice. Na taj način korisnik u realnom vremenu može da testira prikaz njegovih izmena.

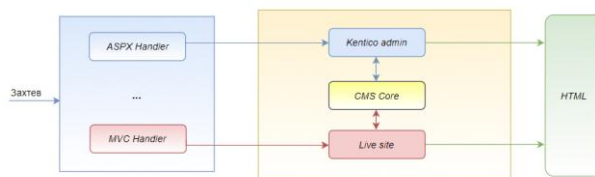
CDA predstavlja usluge koje podržavaju upravljanje i isporuku sadržaja koji je napravljen u okviru CMA sistema. Ovaj deo zadužen je uglavnom za prebacivanje sadržaja na glavni server, kao i za procesiranje sadržaja koji je označen da treba da se prebaci sa predprodukcionog na produkcion server.

Svaki napredni CMS sistem je modularan, tj. sastoji se iz manjih celina (modula). Svaki modul poseduje određenu funkciju i korisnik može iskoristiti samo one module koji su mu potrebni.

Ovaj princip rada poboljšava prilagodljivost CMS sistema za razne tipove marketinga i implementacija.

Kentico administrativni portal [4] se sastoji iz dve sekcije (vidi sliku 2):

- **CMS Site Manager** – za konfiguraciju i razvoj aplikacije
- **CMS Desk** – za upravljanje sadržajem i podacima



Slika 2. Arhitektura Kentico CMS sistema [5]

3. OPIS KORIŠĆENIH TEHNOLOGIJA

Tehnologije i tehnike korišćene za projektno rešenje mogu se podeliti na tri celine, od kojih prva obuhvata *MVC Framework* i implementaciju generatora koda u okviru programskog jezika C#, dok su druge dve vezane za meta-model i implementaciju obrađivača šablona.

3.1 STRUKTURA IMPLEMENTACIJE

U današnje vreme, mnoge aplikacije podacima od velikog značaja za poslovnu logiku pristupaju direktno iz baze podataka. Direktnim pristupom podacima može doći do neželjenih pojava kao što su:

- Dupliciranje koda
- Povećan rizik za programske greške
- Poteškoće u centralizaciji politika vezanih za podatke kao što je keširanje
- Nemogućnost jednostavnog testiranja poslovne logike bez uticaja spoljnih zavisnosti

Da bi se navedeni problemi izbegli koristi se Repository šablon. On se koristi kako bi se razdvojila logika za preuzimanje podataka i mapiranje izvornih podataka na podatke poslovne logike. Ovaj sloj izvornih podataka uglavnom je predstavljen kao baza podataka ili kao veb servisi.

Svaka nova funkcionalnost koja se implementira u okviru Kentico sistema se implementira u obliku modula. Moduli su veoma pogodni za rad jer podržavaju mnogo već predefinisanih funkcionalnosti od strane samog sistema. Veoma je jednostavno napraviti novi grafički korisnički interfejs i iščitavati podatke smeštene u modulu. Za svaki modul se može napraviti prečica i izmestiti na radno okruženje. Ono što je takođe važno jeste da se prava pristupa mogu dodeljivati i na nivou modula, tako da urednici sadržaja nemaju pristup tehničkim stvarima. Imajući prethodno u vidu, predložen je i implementiran novi modul pod nazivom Code Generator.

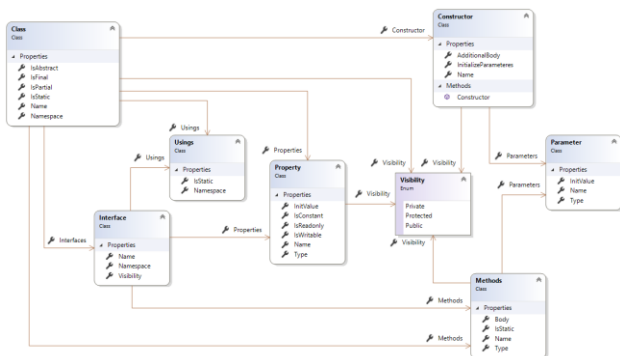
Sama implementacija se sastoji iz dva interfejsa:

- Dobavljanje tipova stranica
- Generator koda

Ovaj modul je namenjen prevashodno generisanju koda u odnosu na tipove stranica u okviru Kentico sistema koji je zasnovan na organizaciji koristeći MVC model. Generisanje koda se vrši na osnovu tipa stranica koje su korišćene prilikom izgradnje samog sistema. Sam generator je implementiran tako da bude generički, ali podaci koji se dostave generatoru moraju se strukturirati tako da odgovaraju samoj implementaciji. Implementacija ovog modula je zasnovana na tome da može da se koristi nezavisno od Kentico sistema. Dovoljno je samo uvesti i registrovati ovaj modul u bilo koji sistem i pokrenuti generator.

3.2 META-MODEL

Meta-model na apstraktan način opisuje moguću strukturu modela definišući pritom koncepte domena i njihove veze kao i ograničenja i pravila modelovanja [11]. U ovoj implementaciji meta-model predstavlja model klasa generatora koda. Ovaj model je generički (vidi sliku 3) i pokriva veliki broj situacija do kojih može doći u programskom jeziku C#.



Slika 3. Prikaz meta-modela generatora koda

Podaci dobijeni iz Kentico sistema se mapiraju na ovaj model. Osnovni modeli su Klasa (Class) i Interfejs (Interface). Svi ostali čine gradivne elemente ova dva modela. Na osnovu meta-modela implementirani su i šabloni, koji koriste isti princip.

3.3 ŠABLONI

Šablon (Template) predstavlja tekstualni fajl koji se sastoji iz statičkog dela (statičkog teksta) i dinamičkog dela (u koji spadaju naredbe, promenljive, izrazi, pozivi funkcija, filteri). Šablon se povezuje sa modelom podataka u cilju kreiranja rezultujućih dokumenata. Dinamički deo šablona se evaluiira na osnovu podataka iz modela a zatim se spaja sa statičkim čime se kreira rezultujući dokument. Šabloni se koriste za generisanje proizvoljnog tekstualnog sadržaja poput HTML, JSON, XML, CSS, JavaScript, C#, Java, Python itd. Šabloni omogućuju generisanje koda, kao i drugih datoteka u svim programskim jezicima. Oni se pišu tako da transformišu kod koji se generiše prema podacima iz modela.

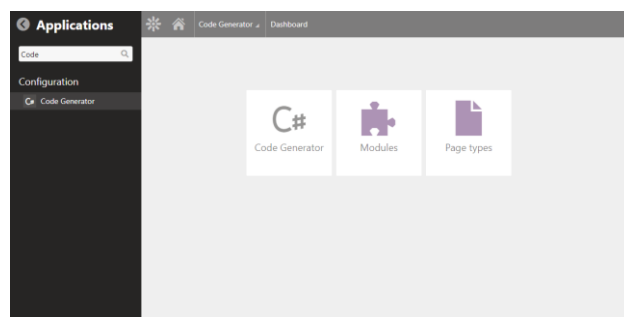
Šablon sadrži kombinaciju teksta koji je potrebno izgenerisati i programski kod koji generiše promenljive delove teksta. Programski kod omogućava da se određeni delovi teksta ponavljaju ili uslovno izostavljaju iz generisanog teksta.

Tekst može biti programski kod koji će biti deo same aplikacije. Ukoliko se generiše programski kod važno je ostaviti mogućnost ručnog proširenja tako da se kod prilikom ponovnog generisanja ne pregazi. Programski kod koji se generiše može biti u bilo kom programskom jeziku.

U implementaciji ovog rešenja korišćeni su T4 šabloni. T4 šablon je veoma moćan alat koji se koristi pri generisanju koda u C# programskom jeziku. Sve karakteristike koje važe generalno za šablone, važe i za ovu vrstu. Da bi rasteretili kompleksnost šablona, svaki entitet koji dolazi iz meta-modela ima svoj šablon koji ga transformiše u tekst. Upravo iz tog razloga u ovoj implementaciji napravljen je još jedan međusloj. Taj sloj je zadužen da premapira podatke koji dolaze iz meta-modela u podatke koji su pogodniji i intuitivniji generatorima koda zasnovanim na šablonima.

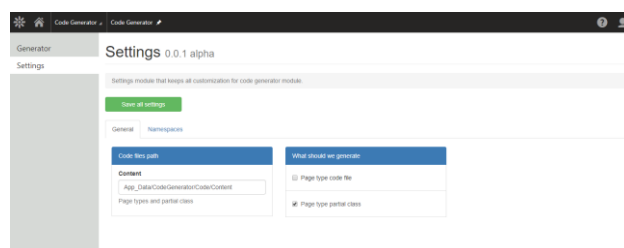
4. PRIKAZ IMPLEMENTIRANOG REŠENJA

Da bi se pristupilo ovom modulu prethodno ga je potrebno registrovati u Kentico sistemu. Ovo se radi jednostavnom instalacijom modula kroz modul aplikaciju. Nakon toga CodeGenerator modulu se može pristupiti iz liste aplikacija sa leve strane (vidi sliku 4). Takođe, prečica do aplikacije se može postaviti i na radni prostor. Na taj način se modulu može pristupiti direktno sa početne strane administrativnog portala.



Slika 4. Početna strana u radnom okruženju

Sama konfiguracija modula jednako je važna kao i sam generator. U okviru ove sekcije vrše se detaljna podešavanja vezana za generator (vidi sliku 5).



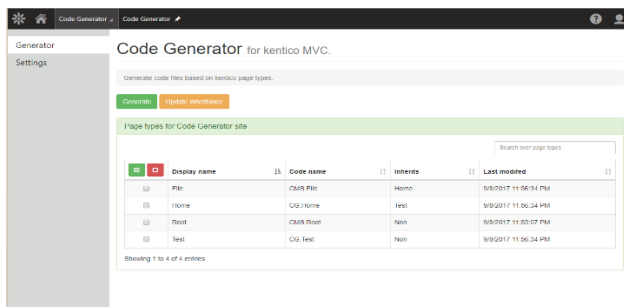
Slika 5. Početna strana sekcije za konfiguraciju generatora

Prva strana u sekciji za podešavanja modula poseduje naziv modula, kao i njegovu trenutnu verziju. Opšta podešavanja poseduju mogućnost postavljanja putanje gde će se generisati svi fajlovi iz generatora, kao i mogućnost odabira tipova željenih fajlova koji će se generisati. U trenutnoj verziji, podržana su dva tipa:

- Tipovi stranica
- Deljene klase datih tipova

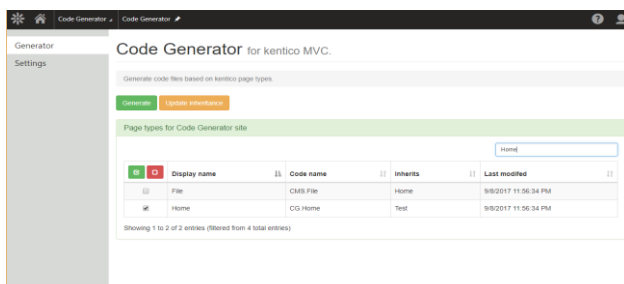
Druga sekcija u okviru podešavanja jeste sekcija za podešavanje opsega vidljivosti. Za svaki od tipova klasa predloženih u prvoj sekciji, moguće je dodeliti specifičan opseg vidljivosti. Na taj način se omogućuje korišćenje modula u više različitih projekata jer su sve promenljive konfigurabilne. Klikom na dugme za skladištenje informacija, sva podešavanja, iz obe sekcije se čuvaju direktno u modulu.

Početna strana generatora koda predstavlja grafički korisnički interfejs pomoću koga se vrši generisanje koda (vidi sliku 6). Dizajn stranice je prilično jednostavan i sastoji se od dva dugmeta za generisanje koda, kao i tabele za prikaz svih tipova stranica iz Kentico sistema.



Slika 6. Početna strana modula za generator koda

Tabela za prikaz tipova stranica poseduje pretragu u gornjem desnom delu (vidi sliku 7), kao i dva dugmeta za odabir svih tipova i za uklanjanje selekcije. Pretraga se odvija po svim vrednostima polja iz tabele.

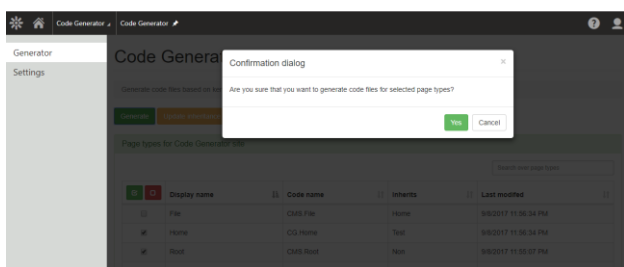


Slika 7. Primer pretrage tabele za prikaz tipova stranica

Generator koda poseduje dve akcije za generisanje koda:

- Generisanje koda
- Ažuriraj nasleđene tipove

Klikom na dugme za generisanje koda, pojavljuje se modalni dijalog koji zahteva potvrdu korisnika da li želi da nastavi sa ovom akcijom (vidi sliku 8). Kod se generiše za sve tipove stranica obeležene u tabeli za konfiguraciju odabranu iz sekcije za konfiguraciju.



Slika 8. Modalni dijalog za potvrdu generisanja koda

Klikom na dugme za ažuriranje nasleđenih tipova, odvija se ista akcija kao i za generisanje koda, sa bitnom razlikom koji tipovi stanica se generišu. Na primer, ako odaberemo ovu akciju za tip *CG.Test*, neće se samo izgenerisati ovaj tip već i svi ostali tipovi koji nasleđuju specifično ovaj tip. Ova akcija ažurira sve nasleđene tipove. To znači da će se u ovom slučaju ažurirati i *CG.Home* kao i tipovi koji nasleđuju Home, što je u ovom slučaju *CMS.File*.

5. ZAKLJUČAK

Zadatak ovog rada je bio da prikaže implementaciju generatora koda u okviru sistema za dinamičko upravljanje sadržajem. Prikazano je konkretno rešenje u okviru Kentico platforme koje direktno utiče na optimizaciju programskog rešenja, smanjuje vreme izrade i rešavanje budućih problema, ali isto tako utiče na kompleksnost. U okviru platforme generiše se samo deo sistema za dobavljanje sadržaja. Način dobavljanja je identičan za svaki sistem, s tim što se menjaju gradivne komponente, kao i pojedinačni delovi sistema koji nisu od velike važnosti za samu implementaciju, ali jesu za kvalitet kao i za lakše održavanje sistema u budućim iteracijama.

6. LITERATURA

- [1] A. Mauthe, P. Thomas, Professional Content Management Systems: Handling Digital Media Assets, John Wiley & Sons 2004
- [2] What is SEO?
<http://whatis.techtarget.com/definition/search-engine-optimization-SEO>
- [3] One to one marketing – Strategija zasnovana na personalizaciji sadržaja
<http://searchcrm.techtarget.com/definition/one-to-one-marketing>
- [4] Kentico 10 – Dokumentacija
<https://docs.kentico.com/>
- [5] Using Kentico MVC
<https://www.slideshare.net/thomrobbins/using-mvc-with-kentico-8>
- [6] M. Voelter, T. Stahl, Model-Driven Software Development: Technology, Engineering, Management, John Wiley & Sons, 2006, Foreword

Kratka biografija:

Dragoljub Ilić je rođen 20.06.1992. godine u Novom Sadu. Osnovnu školu „Žarko Zrenjanin“ u Bačkom Magliču završio je 2007. godine, a srednju elektrotehničku školu „Mihajlo Pupin“, smer Elektrotehničar računara, u Novom Sadu 2011. Iste godine upisao je osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu, odsek Elektrotehnika i računarstvo, smer Računarstvo i automatika. Osnovne akademske studije završava 2015. godine sa prosečnom ocenom 8,03. Diplomski rad sa temom „Generisanje poslovnih aplikacija pomoću generičkog modela“ odbranio je 26.10.2015. godine sa ocenom 10. Master akademske studije Fakulteta tehničkih nauka, odsek Elektrotehnika i računarstvo, smer Softversko inženjerstvo i informacione tehnologije, usmerenje Softversko inženjerstvo upisuje školske 2015/2016. Sve ispite predviđene planom i programom položio je sa prosečnom ocenom 10,00.

UPRAVLJANJE I NADZOR RADA GENERATORA U DISTRIBUTIVNIM MREŽAMA CONTROL AND MONITORING OF GENERATOR OPERATION IN DISTRIBUTION NETWORKS

Dorđe Bogdanović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je opisan je skup funkcija koje se koriste za regulaciju učestanosti i aktivnih snaga u distributivnim mrežama. Funkcije koje su opisane su: automatska regulacija aktivnih snaga, ekonomski dispečing, proračun optimalnog angažovanja agregata i rasterećenje. Akcenat je stavljen na prikazivanje načina testiranja navedenih funkcija u DMS okruženju i procenu ostvarenih ušteda usled njihovog korišćenja.

Abstract – This paper describes set of functions for regulation of frequency and active power in distribution networks. The following functions are described: automatic generation control, economic dispatching, unit commitment and load shedding. The accent was on the testing of these functions in DMS environment and estimation of achieved benefit during its use.

Ključne reči: Regulacija učestanosti i aktivnih snaga, distributivna mreža, generatori, testiranje.

1. UVOD

Elektroenergetski sistem je dinamičan sistem u kome se u svakom trenutku dešavaju promene prije svega sa strane potrošnje. Osim toga moguće su i velike promene u proizvodnji električne energije usled velikih kvarova i havarija na elektranama. Sve te oscilacije dovode do disbalansa u sistemu, do promena frekvencije i tokova aktivnih snaga a samim tim i do smanjenja stabilnosti samog elektroenergetskog sistema. Zato je opravdano postojanje sistema za regulaciju učestanosti i aktivnih snaga, kako u prenosnoj tako i u distributivnoj mreži. Poseban problem predstavljaju generatori u distributivnim mrežama koji pored svih prednosti koje donose mogu dovesti i do niza kritičnih situacija, počevši od pojave ostrvskog rada pa do niza drugih.

U drugoj glavi opisan je način na koji distribuirani generatori (DG) utiču na elektroenergetski sistem i koje su prednosti i nedostaci njihovog korišćenja sa ekonomskog i tehničkog aspekta.

U trećoj glavi opisane su primarna, sekundarna i tercijarna regulacija koje se primenjuju u elektroenergetskim sistemima.

U četvrtoj glavi objašnjeno je kako funkcioniše automatska regulacija proizvedene aktivne snage u sklopu softverskog paketa za distributivni menadžment sistem (DMS).

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Duško Bekut, red. prof.

Peta glava opisuje ekonomski dispečing, funkciju koja vrši proračune nad generatorskim jedinicama i na osnovu toga diktira način na koji će se upravljati generatorima da bi se električna energija proizvodila uz minimalne troškove.

Šesta glava daje opis funkcije za proračun optimalnog angažovanja agregata, koja ima za cilj da odredi najjeftiniju kombinaciju generatora, kako tradicionalnih tako i obnovljivih.

U sedmoj glavi dat je opis funkcije rasterećenja koja ima za cilj održavanje stabilnosti sistema prilikom prekida proizvodnje generatora, odnosno kada ukupna potrošnja iznenada postane mnogo veća od proizvodnje.

Osma glava opisuje način na koji se vrši testiranje funkcija za regulaciju učestanosti i aktivnih snaga u DMS okruženju uz adekvatne primere koji doprinose boljem razumevanju same problematike testiranja.

2. DISTRIBUIRANI GENERATORI

Ugradnja distribuiranih generatora dovodi do poboljšanja kvaliteta električne energije, do stabilnijeg naponskog profila, međutim istovremeno utiče na fluktuacije napona u tački priključenja, naročito ako je u pitanju asinhroni generator.

2.1. Prednosti i nedostaci upotrebe distribuiranih generatora u mreži

Prednosti sa ekonomskog stanovišta:

1. DG mogu da obezbede pokrivanje povećanja potrošnje određenog broja potrošača, čime se smanjuju ili odlažu investicije za proširenje postojećih ili izgradnju novih prenosnih i distributivnih objekata tokom razmatranog planskog perioda.
2. Pojedini tipovi distribuiranih izvora mogu se postaviti na bilo kom mestu i za relativno kratko vreme (gorivne ćelije, mikroturbine). Njihova instalirana snaga može se menjati sa brojem instaliranih modula.
3. Svi tipovi izvora mogu se proširivati u malim koracima, te na taj način zadovoljiti potrošnju [1].

Osim poboljšanja koje donosi postavka DG-a u sistem, takođe dolazi i do određenih tehničkih poteškoća. Tradicionalno, mreže su projektovane tako da je tok snage od mreže višeg ka mreži nižeg naponskog nivoa. Usled postavljanja DG-a u distributivne mreže, tokovi snaga su promenjeni od jednosmernih (od izvora) ka dvo-smernim (smer od izvora ili od generatora). Kritične promene su sledeće:

- Postojanje DG utiče na rad uređaja relejne zaštite, te se moraju uvoditi novi postupci koordinacije između pojedinih uređaja.
- Moguće je da nakon ugradnje DG-a dođe do izmena u planiranju distributivnih sistema.
- Visoko učešće DG-a u potrošnji sistema značajno utiče na stabilnost.
- Neobnovljivi distribuirani izvori su upravljivi, dok većina obnovljivih nije. Upravlјivost znači da je moguće odgovarajućim merama kontrolisati snagu na krajevima generatora [1].

3. VRSTE REGULACIJA UČESTANOSTI I AKTIVNIH SNAGA

U elektroenergetskim sistemima postoje primarna, sekundarna i tercijarna regulacija.

3.1. Primarna regulacija

Proizvodni agregati sa svoje strane mogu biti više ili manje osetljivi na promene učestanosti. Opseg promena u granicama ± 1 Hz, a ponekad i ± 3 Hz propisuje se kao dozvoljeni opseg normalnih promena u kojima agregati moraju odavati punu snagu, ali u ograničenom vremenskom period.

U nekim posebnim pogonskim uslovima dolazi do velikih i naglih promena učestanosti. Ovo uzrokuje potrebu za brzom promenom dovoda radnog fluida u primarni motor što u nekim slučajevima predstavlja veliki problem pa se takve abnormalne situacije otklanjaju delovanjem zaštita koje deluju na potrošače isključujući ih sa mreže (automatsko frekventno rasterećenje). U slučaju naglog ubrzanja rotora generatora, što se dešava u slučaju kratkih spojeva u blizini elektrana, deluje se na zaštitne ventile koji zatvaraju dovod radnog fluida u primarni motor.

3.2. Sekundarna regulacija

Sekundarna regulacija, u suštini, predstavlja naknadno i superponirano dejstvo na primarnu regulaciju, koje potiče od centralizovanog, ili lokalnih regulatora u nekom elektroenergetskom sistemu.

Postoje tri različita moda funkcionisanja sekundarne regulacije, koji zadovoljavaju sva moguća radna stanja u jednom elektroenergetskom sistemu. To su osnovni, korekcion i havarijski mod [2].

U osnovnom regulacionom modu, sekundarna regulacija jednog povezanog elektroenergetskog sistema ima dva osnovna aspekta: održavanje učestanosti i aktivnih snaga razmjene po međusistemskim spojnim vodovima na programiranim, željenim vrednostima i kontrolu raspodele aktivnih opterećenja između generatorskih jedinica u pogonu.

Funkcije sekundarne regulacije, u ostala dva regulaciona moda se nešto razlikuju. Dok je ekonomija od primarnog interesa u osnovnom regulacionom modu, ona se u slučaju teških poremećaja ili neočekivano velikih opterećenja zamenjuje prioriteto koji se prebacuje na sigurnost pogona. Sekundarna regulacija tada radi u havarijskom modu.

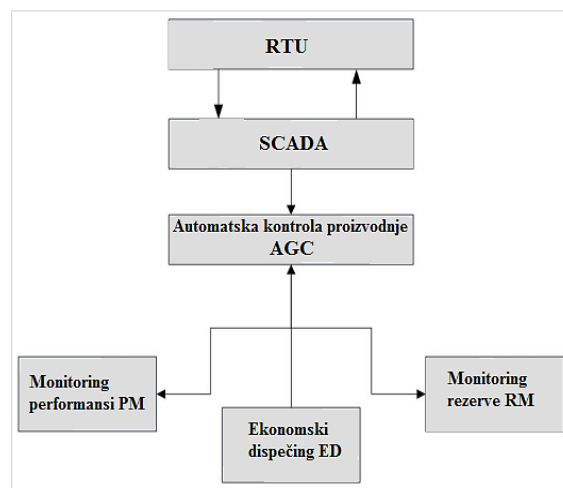
Treći regulacioni mod sekundarne regulacije je korekcion mod, koji se koristi u periodima korekcije grešaka sinhronog vremena i ugovorenih energija razmjene.

3.3. Tercijarna regulacija

Tercijarna regulacija je proces koji se superponira na primarnu i sekundarnu regulaciju učestanosti i snaga, a cilj joj je da se raspodela opterećenja između proizvodnih agregata u pogonu obavi na najekonomičniji način, obično na bazi kriterijuma minimalnih pogonskih troškova.

4. AGC U DMS OKRUŽENJU

AGC (automatska regulacija proizvedene aktivne snage) održava učestanost sistema i vrši transakciju sa susednim kontrolnim područjima koristeći povratnu informaciju sa PI (D) regulatora koji su sastavni deo proizvodnih jedinica (generatora). To je potrebno raditi sa minimalnim operativnim troškovima poštujući sva ograničenja u radu sistema i uređaja, npr. minimalno i maksimalno opterećenje, maksimalna brzina odziva, itd. Slika 1 prikazuje šematski dijagram AGC kontrole jedne elektrodistributivne kompanije [2].



Slika 1. Šematski prikaz AGC regulacije jedne elektrodistributivne kompanije

Sekundarna regulacija se uglavnom zasniva na kombinovanom signalu izračunatom iz međusobne razmjene sistema i odstupanja učestanosti, nazvanom greška kontrolnog područja (engl. Area Control Error – ACE).

5.1. Režimi rada AGC funkcije

AGC može biti u četiri različita režima rada. To su:

- Monitoring – u ovom režimu funkcija AGC je ručno pauzirana i obavlja sve potrebne proračune ali ne vrši kontrolu pojedinačnih jedinica.
- Aktivan – funkcija obavlja sve potrebne proračune i vrši kontrolu pojedinačnih jedinica. U aktivan režim se može preći samo iz monitoring režima.
- Van mreže – ovo je režim mirovanja i AGC funkcija samo čita trenutni status. Ovaj režim je od suštinskog značaja za vršenje čitanja i upisivanja u bazu podataka.
- Suspendovan – AGC se automatski suspenduje usled abnormalnih stanja:
 - ACE je iznad dozvoljenih granica,
 - učestanost je iznad dozvoljenog opsega,
 - važna merenja imaju loš kvalitet,
 - baza podataka je oštećena.

5. EKONOMSKI DISPEČING

Ekonomski dispečing je generatorska funkcija koja proračunima nad generatorskim jedinicama omogućava proizvodnju električne energije uz minimalne troškove, uz pouzdano napajanje potrošača, uzimajući u obzir sva operativna ograničenja samih generatorskih jedinica i prenosnih elemenata.

Sigurnost sistema ograničava opcije ekonomskog dispečinga, jer operativni uslovi utiču na mogućnosti kombinacije jedinica koje mogu da zadovolje potrošnju i istovremeno održe sistem sigurnim. Pravila pouzdanosti zahtevaju da operacije nad sistemom uvažavaju naponske prilike, termička i ograničenja samih elemenata sistema (kao što su generatori i prenosni vodovi). Operatori uvek moraju voditi sistem tako da obezbede pomenutu sigurnost tipa N-1 i da povratak napajanja i sigurnog stanja bude u okviru 30 minuta.

5.1. Uloga ekonomskog dispečinga u okviru funkcije AGC

Nakon ispada, sistem kontrolnog područja u kom je došlo do ispada generatora šalje signal ostalim jedinicama u cilju vraćanja učestanosti na željenu vrijednost (50 Hz). Ukoliko kontrolno područje ne može otkloniti posljedice ispada, mora zahtijevati hitno djelovanje (pomoć) susjednih sistema sa kojima je u interkonekciji. Vremenski okvir ove faze je od nekoliko sekundi do 10 minuta nakon ispada.

Finalna etapa reakcije sistema može biti nazvana fazom ekonomskog dispečinga. Ona predstavlja dio AGC faze. Za vrijeme ekonomskog dispečinga AGC kontrolni signali se šalju kako bi se osigurala ekonomična zamjena generisanja snage. Tokom finalne etape proizvodnja će biti preraspodijeljena na jedinice koje omogućavaju minimalne proizvodne troškove, tako da se nadomjesti ispala snaga i obezbijedi ponovno napajanje potrošača [3].

6. PRORAČUN OPTIMALNOG ANGAŽOVANJA AGREGATA

Proračun optimalnog angažovanja agregata ima za cilj da odredi najjeftiniju kombinaciju obnovljivih i tradicionalnih proizvodnih jedinica kako bi se zadovoljile potrebe projektovane potrošnje i obavezne rezerve, uz uvažavanje ograničenja generatora i elektroenergetskog sistema. Raspored se pravi za vremenski interval od osam dana i određuje koji generatori će raditi tokom kojih sati. Korist od ove funkcije je smanjenje ukupnih troškova proizvodnje i smanjenje broja pokretanja i isključivanja generatorskih jedinica tokom određenog vremenskog perioda. U problem angažovanja generatorskih jedinica mogu biti uključene i rezerve koje su izvan mreže, a mogu se relativno brzo dovesti u stanje maksimalne proizvodnje i sinhronizovati na mrežu. To uključuje dizel agregate, gasne turbine, kao i većinu hidrogenatora i reverzibilnih hidrogenatora [4].

7. RASTEREĆENJE

Rasterećenje (engl. *Load Shedding*) je složena funkcija namijenjena za ručno ili automatsko odspajanje potrošača sa mreže usljed prekida proizvodnje ili odspajanje dijelova mreže u vanrednim situacijama, kako bi se održala stabilnost sistema i smanjio ukupni ispad potrošača.

Sistem za rasterećenje mora obezbediti napajanje svim bitnim i najkritičnijim potrošačima, kako u mreži, tako i u samoj elektrani. Ovo se postiže isključivanjem ne-esencijalnih opterećenja u mreži u slučaju nedostatka proizvedene snage potrebne za napajanje potrošača i kao i sopstvene potrošnje u elektranama. Ovo je neophodno kada se desi neki od sljedećih ozbiljnih poremećaja:

- Ispad generatora.
- Ispad partnera (sve veze prema susednom sistemu).
- Učestanost sistema je ispod definisanog ograničenja.

U tim slučajevima, PLC prima signale koji definišu problem i generišu odgovarajuću komandu za prekid napajanja.

Komande za prekid napajanja potrošača se definišu kroz tabelu rasterećenja, koju sistem izračunava svakih pet sekundi, i nakon svake akcije. Nakon svakog izračunavanja, tabela se šalje PLC-u ili RTU-u. Na taj način je osigurano da vreme isključenja nije više od 200 milisekundi (vreme isključenja se meri između ispada generatora i komande koju izvršava aktuator).

8. TESTIRANJE FUNKCIJE ZA REGULACIJU UČESTANOSTI I AKTIVNIH SNAGA U DMS OKRUŽENJU

Da bi softver pravilno funkcionisao, potrebno je izvršiti i adekvatno testiranje da bi se utvrdilo da pojedine funkcije u okviru sistema za regulaciju učestanosti i aktivnih snaga daju očekivane rezultate na način na koji su definisane.

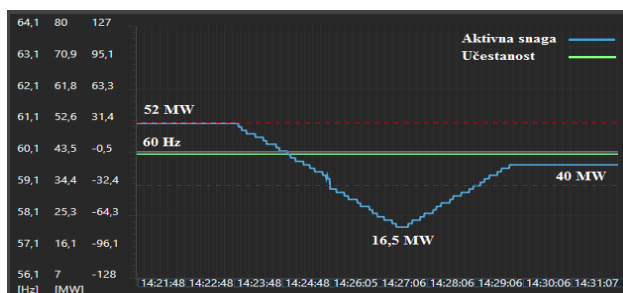
Tabela 1. *Kataloški podaci generatora nad kojima se vrši regulacija*

	Vrsta	Minimalna snaga	Maksimalna snaga	Regulacioni opseg
Generator A	Konvencion alni	14 [MW]	167 [MW]	75-105 [MW]
Generator B	Rotirajući fotonaponsk i paneli	7 [MW]	80 [MW]	9-52 [MW]

8.1. Upravljanje generatorom u linearnom režimu

Jedan od načina na koji može da se vrši upravljanje generatorom je u linearnom režimu. To znači da generator sa regulatora dobija signal koji se može predstaviti pomoću linearne funkcije. Takav način upravljanja omogućuje generatoru da postepeno dođe do željene vrednosti i novog stacionarnog stanja.

Na slici 2. se može videti kako se kreće aktivna snaga generatora tokom regulacije u linearnom režimu (generator B iz tabele 1). Vidi se da kriva aktivne snage nije glatka nego se regulacija odvija u koracima, ali se u principu kriva može aproksimirati linearnom funkcijom.

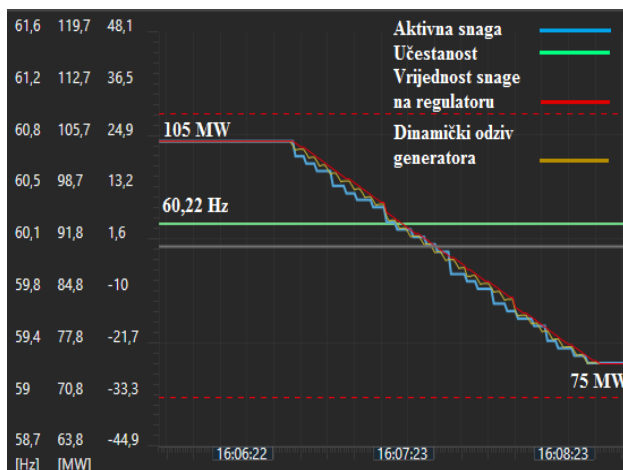


Slika 2. Kriva aktivne snage generatora B tokom regulacije u linearnom režimu

8.2. Upravljanje generatorima promenom greške kontrolnog područja

Da bi ovaj način regulacije imao smisla, potrebno je posmatrati generatore kao delove jedne kontrolne oblasti, koja vrši razmjenu snage sa drugom kontrolnom oblašću odnosno drugom distributivnom kompanijom. Između njih postoji ugovorena snaga razmjene koja može biti u oba smera i ona nam diktira kako će se ponašati generatori, odnosno na koji način će automatska regulacija proizvedene aktivne snage izračunati i poslati odgovarajuće signale regulatorima.

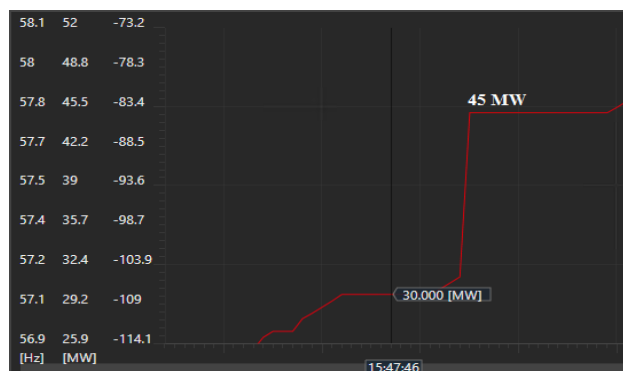
Neka se pretpostavi da je ugovorena snaga razmjene između jedne i susedne oblasti pozitivna i da je AGC u režimu održavanja konstantne snage razmjene i da je došlo do ispada nekoliko generatora, kao i da se vrednost snage koju se isporučuje susednoj oblasti smanjila. Onda funkcija proračunava signal greške kontrolnog područja (ACE) i na osnovu njega šalje odgovarajućim generatorima, odnosno njihovim regulatorima signal za povećanje proizvodnje da bi se ACE greška svela na minimum, ili u potpunosti neutralisala. ACE greška može imati pozitivnu ili negativnu vrednost.



Slika 3. Kriva aktivne snage generatora A tokom automatske regulacije proizvedene aktivne snage

8.3. Zabranjene zone rada generatora

Većina konvencionalnih generatora ima određene opsege u kojima nije dozvoljen njihov trajan rad. Ti opsezi se zajedno sa ostalim kataloškim podacima o generatorima upisuju u bazu koju koristi sistem za regulaciju učestanosti i aktivnih snaga, tako da se prilikom regulacije mogu uvažiti na adekvatan način. Na slici 4 može se videti kako izgleda grafik aktivne snage generatora koji je u linearnom režimu upravljanja sa zabranjenim opsegom upravljanja od 30-45 [MW].



Slika 4. Kriva aktivne snage generatora u linearnom režimu sa zabranjenim opsegom upravljanja

9. ZAKLJUČAK

Sekundarna regulacija učestanosti i aktivnih snaga je jedan od prvih primera automatskog upravljanja velikim sistemima. Ovako realizovan sistem se pokazao izrazito pouzdanim, pa se njegova osnovna struktura ne menja, samo se vrše usavršavanja i estetske izmene. U ovom radu dat je jedan kompletan osvrt na problematiku regulacije učestanosti i aktivnih snaga u elektro-energetskom sistemu (AGC), ekonomski dispečing (ED), optimalno angažovanje agregata (UC), kao i na funkciju rasterećenja (LS) koja se koristi kao prva faza odbrane EES prilikom ispadanja velikih sistema proizvodnje sa mreže.

Predstavljen je način testiranja funkcije za regulaciju učestanosti i aktivnih snaga i dobijeni rezultati u DMS okruženju. Testiranje je vršeno na malom broju generator-skih jedinica, ali sasvim dovoljno da se shvati na koji način sistem funkcioniše.

10. LITERATURA

- [1] M.Bratić: Distribuirani generatori, master rad, Fakultet tehničkih nauka, Novi Sad, 2012.
- [2] V.Perić: Realizacija funkcije sekundarne regulacije učestanosti i aktivnih snaga u elektroenergetskom sistemu, master rad, Fakultet tehničkih nauka, Novi Sad, 2008.
- [3] M.Galić: Primena funkcije Ekonomski dispečing za optimizaciju pogona elektroenergetskog sistema, diplomski rad, Fakultet tehničkih nauka, Novi Sad, 2008.
- [4] A.J.Wood, B.F.Wollenberg: Power Generation, Operation and Control, Wiley, 1996.

Kratka biografija:



Đorđe Bogdanović rođen je u Sarajevu 1991. god. Diplomirao je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine 2015. god. i nakon toga upisao master studije na istom fakultetu.

INTEGRISANI IZVOR REFERENTNOG NAPONA VREDNOSTI 0.6 V U 55nm CMOS TEHNOLOGIJI**A 0.6 V BANDGAP VOLTAGE REFERENCE IN 55nm CMOS TECHNOLOGY**Ivan Bogosavljević, *Fakultet tehničkih nauka, Novi Sad***Oblast –ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Projektovan je CMOS izvor referentnog napona vrednosti od 0.6 V za primene u temperaturskom opsegu od -40 do 85°C. Teženo je da se postigne što manja vrednost temperaturnog koeficijenta (eng. Temperature Coefficient - TC) [pmm/°C]. Kao sastavni deo ovog kola projektovan je i operacioni pojačavač sa preklopljenom kaskodom, uključujući i neophodno polarizacijo i start up kolo. Izvršena je simulaciona provera primenom korner i Monte Carlo analize i prikazani su rezultati.

Abstract – This paper presents an example of 0.6 V CMOS bandgap voltage reference design, which will operate in the temperature range from -40 to 85 ° C. It is desired to minimize the value of temperature coefficient (TC) [pmm/°C]. As an essential building block, folded cascode operational amplifier has been designed, including a necessary polarization and a start-up circuit. A simulation check was performed using Corner and Monte Carlo analysis and results were presented.

Cljučne reči: Izvori referentnog napona, operacioni pojačavač, integrisana kola, CMOS tehnologija.

1. UVOD

U savremenim integrisanim kolima (eng. *Integrated Circuit* - IC) koja predstavljaju kola mešovityh signal (eng. *mixed-signal*) na visokim učestanostima (eng. *Radio Frequency* - RF), nezaobilazna je upotreba izvora referentnih napona. Uobičajno je da takvi sistemi koriste više međusobno nezavisnih naponskih referenci različitih karakteristika shodno potrebama blokova u kojima se primenjuju. Tako na primer svaki linearni stabilizator- regulator sa malim padom napona (eng. *Low Drop Output* - LDO) koristi svoju naponsku referencu za generisanje napajanja nekog vitalnog bloka, kako bi se izbeglo unošenje smetnji na druge signale. Upotreba visoko kvalitetnih naponskih izvora je od ključne važnosti za tačnost i postizanje visoke rezolucije pri konverziji podataka u AD (eng. *Analog to Digital*) i DA (eng. *Digital to Analog*) konvertorima.

U CMOS integrisanim kolima, izvori referentnog napona ili naponske reference se realizuju na osnovu napona energetskog procepa u silicijumu (eng. *Bandgap Voltage Reference*). Sa tendencijom snižavanja napona napajanja IC sistema rastu zahtevi za poboljšanje performansi naponskih referenci koje se postižu preko unapređenja postojećih i pronalaženjem novih topologija kola.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bila dr Jelena Radić, docent.

2. PRINCIP RADA BANDGAP IZVORA REFERENTNOG NAPONA

Princip rada *bandgap* naponske reference se zasniva na sabiranju dva napona suprotnih temperaturnih koeficijenata (TC) sa ciljem generisanja konstantnog napona u željenom temperaturskom opsegu. Poznato je da direktno polarisana dioda na svojim krajevima daje napon koji se smanjuje sa povećanjem temperature. Taj napon ima negativni TC, koji približno iznosi -1.6 mV/°C, i u literaturi se naziva CTAT (eng. *Complementar to Absolut Temperature*) napon. Napon sa pozitivnim TC je termički napon pn spoja koji je proporcionalan sa apsolutnom temperaturom, pa se zato i naziva PTAT (eng. *Proportional To Absolute Temperature*) napon. Vrednost PTAT napona na sobnoj temperaturi je približno 26 mV, dok njegov TC približno iznosi 0.86 mV/°C. PTAT napon se opisuje jednačinom:

$$V_T = \frac{kT}{q}, \quad (2.1)$$

gde je k Boltzmanova konstanta, T apsolutna temperatura i q naelektrisanje elektrona. Napon V_{BE} je po prirodi CTAT i može se opisati jednačinom

$$V_{BE} = V_T \ln \frac{I_C}{I_S}, \quad (2.2)$$

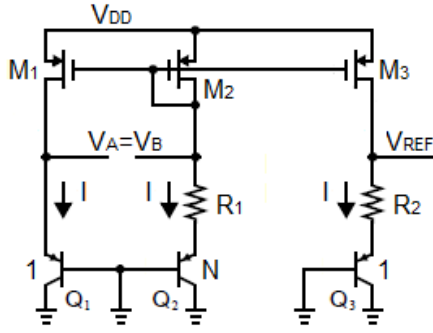
gde je I_S inverzna struja zasićenja bipolarnog tranzistora, a I_C je kolektorska struja.

PTAT napon se može generisati oduzimanjem dva CTAT napona V_{BE1} i V_{BE2} , napona sa dva različita pn spoja čiji odnos površina ima vrednost N , jednačina (2.3). Na slici 1 je prikazano kolo za generisanje PTAT struje $I = \Delta V_{EB}/R_1$ i referentnog napona.

$$\Delta V_{EB} = V_{EB1} - V_{EB1} = V_T \ln N \quad (2.3)$$

Da bi kolo generisalo PTAT struju glavni uslov koji treba ispuniti je da naponi u tačkama A i B budu jednaki. U tu svrhu se koristi strujno ogledalo. Da bi se postigla veća tačnost referentne struje i referentnog napona koriste se i druge topologije sa kaskodnim strujnim ogledalima i operacionim pojačavačima.

Da bi se obezbedile dobre karakteristike *bandgap* naponske reference i pri tome izbegle skupe procedure podešavanja komponenti, pribegava se tehnici uparivanja komponenti kola. Poželjno je da se tranzistori strujnog ogledala, otpornici i bipolarni tranzistori uparuju primenom *Common Centroid* tehnike projektovanja lejauta kao jedne od najboljih za postizanje istih (približnih) karakteristika komponenti kola [1].



Slika 1. Izvor referentnog napona i PTAT struje

Pri primeni operacionog pojačavača kao pojačavača greške, potrebno je da pojačavač ima veliko pojačanje A i što manji napon ofseta.

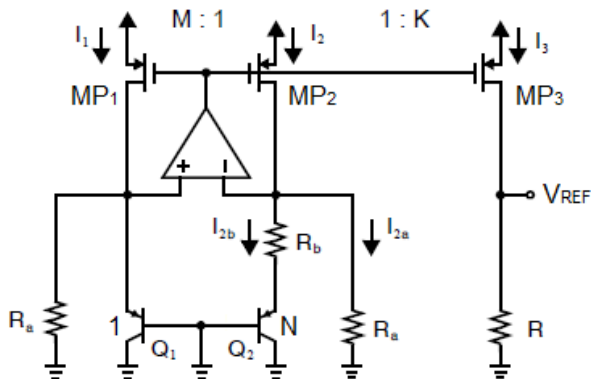
2.1 Izvor referentnog napona vrednosti manje od 1 V

Prvobitne topologije *bandgap* naponskih referenci ne mogu se koristiti u novijim CMOS tehnologijama sa skraćenim dužinama kanala tranzistora i naponima napajanja čije vrednosti su niže od 1.5 V. Fundamentalni problem je što je vrednost referentnog napona, koji odgovara energetskom procepu u silicijumu na $T=0$ K, od približno 1.205 V bliska ili pak može biti i viša od vrednosti napona napajanja, čineći tako rad *bandgap* kola nemogućim [2]. Uzrok tome je što se ne može ostvariti dovoljan pad napona na elementima kola, koji je neophodan za njihovo pravilno funkcionisanje. Vremenom se došlo do novih rešenja gde se struja koja predstavlja zbir PTAT i CTAT struje propusti kroz otpornik i na taj način dobije referentni napon vrednosti manje od 1 V [3].

Na slici 2 je prikazana topologija koja koristi pomenutu ideju. Povratna sprega formirana od operacionog pojačavača i tranzistora strujnog ogledala (MP1 i MP2) osigurava da naponi na drejnovima istih tranzistora budu dobro upareni i jednaki CTAT naponu V_{EB1} , na pn spoju bipolarnog vertikalnog tranzistora Q_1 . Napon na krajevima otpornika R_b je jednak razlici napona na tranzistorima Q_1 i Q_2 . Označava se sa ΔV_{EB} , jednačina (2.5). Promena tog napona je direktno proporcionalna sa promenom temperature, tj. u pitanju je PTAT napon.

$$\Delta V_{EB} = \frac{kT}{q} \ln(MN) \quad (2.5)$$

CTAT napon V_{EB1} i PTAT napon ΔV_{EB} se konvertuju u struje I_{2a} i I_{2b} preko otpornika R_a i R_b . Te dve struje



Slika 2. Šematič bandgap reference niže vrednosti napona

$I_{2a} = V_{EB1}/R_a$ i $I_{2b} = \Delta V_{EB}/R_b$ se sabiraju u drejnu tranzistora MP2 i preslikavaju strujnim ogledalom u granu sa tranzistorom MP3. Ta struja prolazeći kroz otpornik R stvara pad napona koji je jednak referentnom naponu:

$$V_{REF} = V_{EB1} \left(\frac{KR}{R_a} \right) + \frac{kT}{q} \ln(MN) \left(\frac{KR}{R_b} \right). \quad (2.6)$$

Potrebno je pažljivo podesiti vrednosti parametara M , N , R_a i R_b tako da se promene PTAT i CTAT napona usled promene temperature međusobno potiru kako bi se ostvario približno nulti TC referentnog napona V_{REF} . Promenom vrednosti R i K može se podešavati vrednost referentnog napona. Glavni izvori greške ovog izvora referentnog napona su ofset napon operacionog pojačavača i neuparenost komponenti (strujnih ogledala, otpornika i bipolarnih tranzistora).

2.2 Operacioni pojačavač sa preklopljenom kaskodom

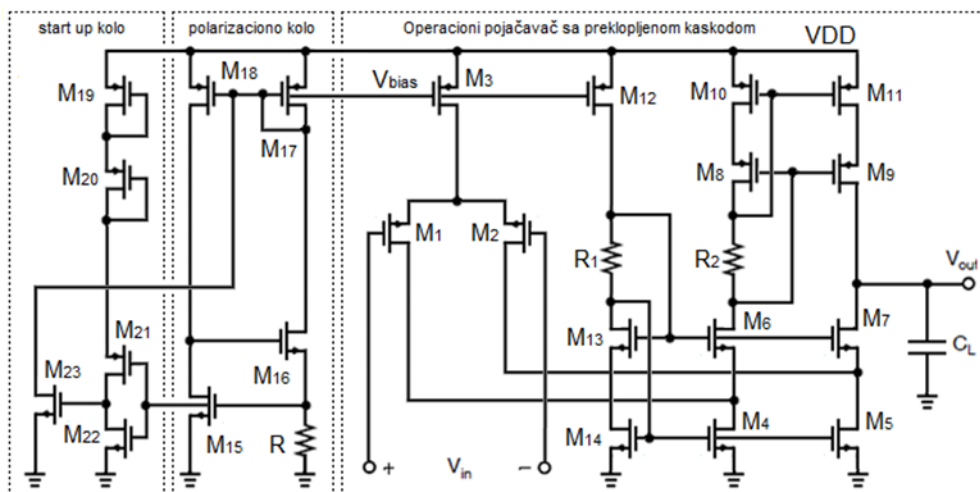
U većini slučajeva integrisani CMOS operacioni pojačavači na svojim izlazima imaju kapacitivna opterećenja što dozvoljava da se veliko naponsko pojačanje ostvari na osnovu velike otpornosti na izlazu pojačavača. U slučaju otporničkog opterećenja, ukupna otpornost kola je jednaka paralelnoj vezi izlazne otpornosti pojačavača i otpornosti opterećenja, što znatno smanjuje vrednost naponskog pojačanja. Problem se uglavnom rešava upotrebom odvojnog naponskog stepena ili bafera, tj. dodatnog pojačavačkog stepena, čime se često ugrožava stabilnost pojačavača u radu zbog smanjene fazne margine (eng. *phase margin* - PM).

Primer pojačavača sa velikom izlaznom otpornošću je preklopljeni kaskodni operacioni pojačavač (eng. *Folded Cascade Operational Amplifier* – FC OP AMP) prikazan na slici 3. Ovaj operacioni pojačavač se smatra jednostepenim pojačavačem, iako u suštini predstavlja kombinaciju pojačavača sa zajedničkim sorsom i pojačavača sa zajedničkim gejtom. Proistekao je iz teleskopsko-kaskodnog pojačavača (eng. *telescopic-cascade operational amplifier*), čiji nedostaci u vidu nemogućnosti kratkog spajanja izlaza i ulaza i smanjenog izlaznog naponskog opsega su otklonjeni ovim tipom pojačavača.

Proračun vrednosti parametara komponenti kola za FC operacioni pojačavač sa PMOS ulaznim diferencijalnim parom izvršen je na način objašnjen u [4], [5] i [6], a zahtevane vrednosti parametara su date u tabeli 1.

Tabela1. Zadate specifikacije operacionog pojačavača

Parametar	Vrednost	Jedinica
A_{v0}	80	dB
GBW	1	MHz
Fazna margina (PM)	80	°
Slew Rate (SR)	1	V/μs
C_L	10	pF
ICMR	Veći od 2.5 V	V
P_{diss}	1	mW
V_{DD}	3.3	V
V_{SS}	0	V



Slika 3. Šematik operacionog pojačavača sa preklopljenom kaskodom

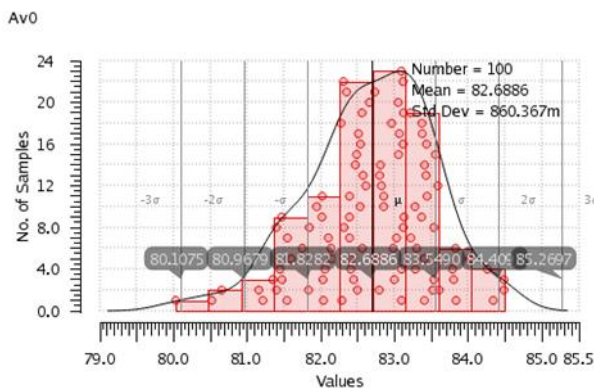
3. REZULTATI SIMULACIJA

Predložena topologija *bandgap* izvora referentnog napona je projektovana u 55nm CMOS tehnologiji proizvođača *GlobalFoundries*. Za proveru karakteristika FC operacionog pojačavača izvršena je korner analiza za 81 graničnu vrednost parametra korišćenog procesa. Dobijeni rezultati su prikazani u tabeli 2.

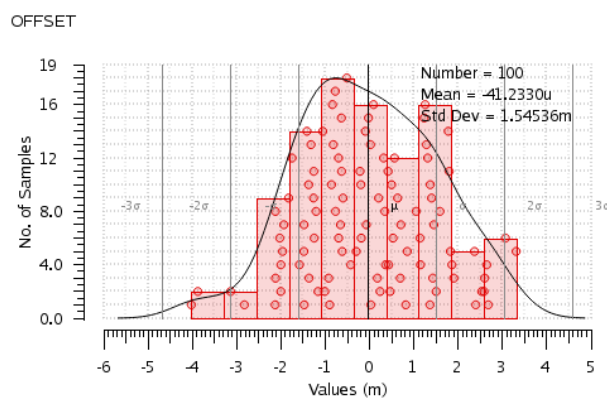
Tabela 2. Rezultati korner analize za operacioni pojačavač

Parametar	Vrednost			Jedinica
	min	nominalna	max	
A_{v0}	76.51	83.99	87.53	dB
PM	84.62	85.09	85.48	°
GBW	1.09	1.5	1.98	MHz
SR	0.56	0.76	1.04	μS
$ICMR$	359m-2.24	355m-2.65	353m-2.94	V
V_{noise}	2.616	2.825	3.064	μV/√Hz
$V_{noise, f=1kHz}$	77.07	83.68	91.22	nV/√Hz
V_{offset}	17.41	54.96	216.5	μV
P_{diss}		162.77		μW

Izvršena je i Monte Carlo analiza za 100 tačaka. Kao ključni parametri pojačavača za primenu kao pojačavača greške prikazani su grafici za pojačanje otvorene petlje A_{v0} (slika 4) i grafik ofseta napona V_{os} (slika 5).

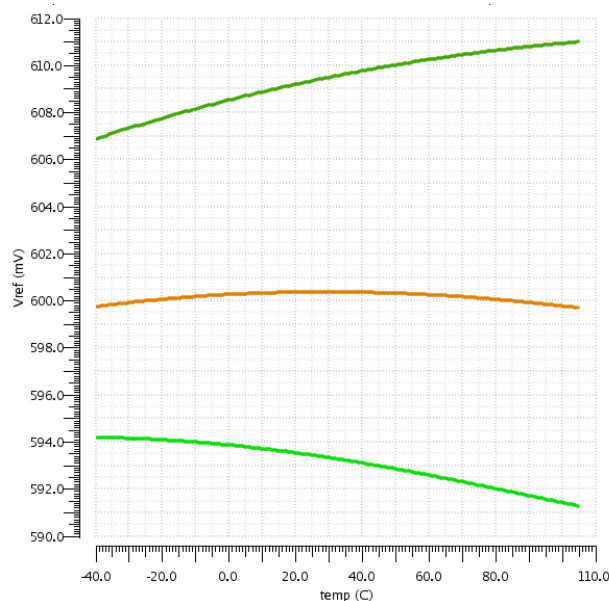


Slika 4. Pojačanje otvorene petlje operacionog pojačavača



Slika 5. Ofset napon operacionog pojačavača

Na slici 6 su prikazane karakteristike referentnog napona za tri karakteristična slučaja korner analize. Korner analizom je utvrđeno da se TC nalazi u opsegu od -35 do +48 ppm/°C.

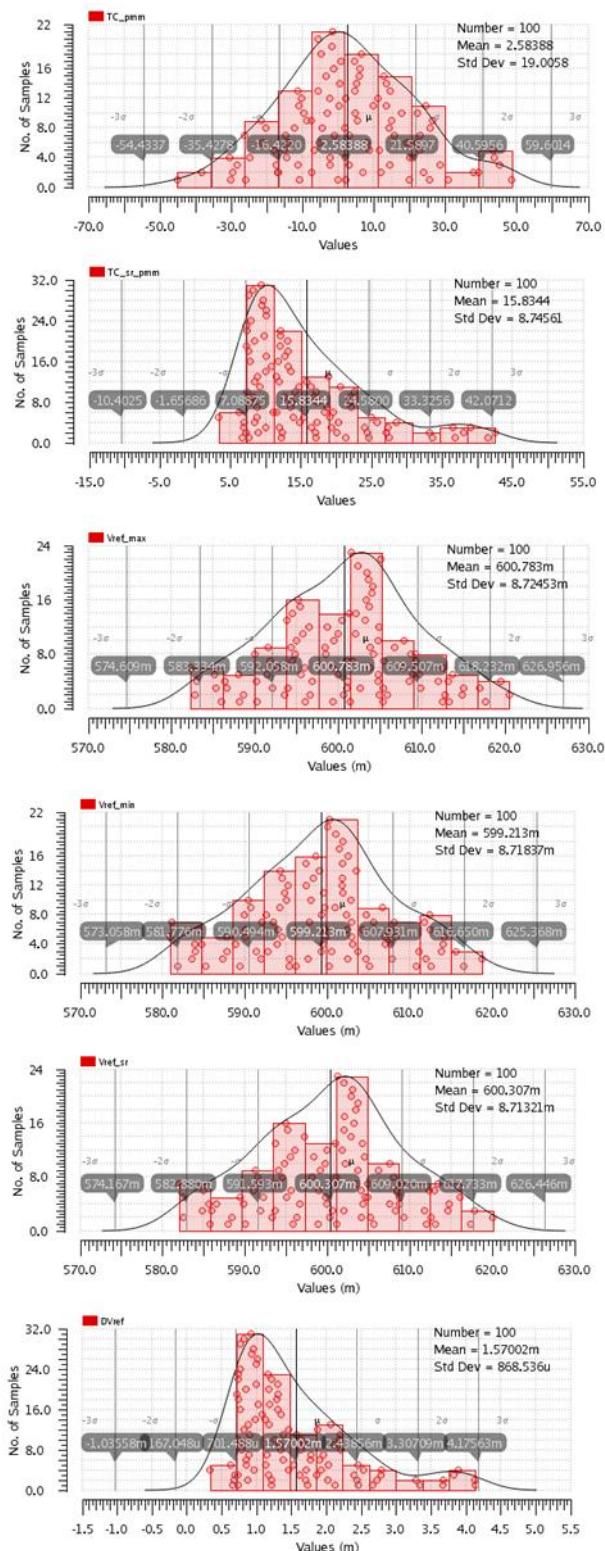


Slika 6. Referentni napon V_{REF} za tri karakteristična slučaja korner analize

Monte Carlo analizom testirane su karakteristike projektovanog kola pri naponu napajanja od 3.3 V. Na slici 7 su

prikazani grafici za sledeće parametre kola: TC, odstupanje TC-a od nominalne temperature, maksimalna vrednost referentnog napona, minimalna vrednost referentnog napona, srednja vrednost referentnog napona i razlika maksimalnog i minimalnog napona.

Može se videti da je srednja vrednost temperaturnog koeficijenta $TC=2.58 \text{ ppm}/^{\circ}\text{C}$, dok je standardna trostruka devijacija $3\sigma=57 \text{ ppm}/^{\circ}\text{C}$.



Slika 7. Rezultati Monte Carlo analize

4. ZAKLJUČAK

Projektovanim *bandgap* izvorom referentnog napona je prema corner analizi postignut TC u opsegu od -35 do $48 \text{ ppm}/^{\circ}\text{C}$. Monte Carlo analizom je dobijeno da je srednja vrednost temperaturnog koeficijenta $TC=2.58 \text{ ppm}/^{\circ}\text{C}$, dok je standardna trostruka devijacija $3\sigma=57 \text{ ppm}/^{\circ}\text{C}$.

Kao mogući načini za postizanje boljih rezultata može se koristiti operacioni pojačavač koji ima manji ofset napon. Takođe umesto FC operacionog pojačavača može se koristiti i neki drugi tip pojačavača, kao na primer dvostepeni operacioni pojačavač. Upotrebom kaskodnih umesto običnih strujnih ogledala u kolu *bandgap* izvora referentnog napona može se postići bolja uparenost struja i time uticati da varijacija referentnog napona bude manja.

5. LITERATURA

- [1] P.K.T. Mok, and K.N. Leung, "Design Considerations of Recent Advanced Low-Voltage, Low-Temperature-Coefficient CMOS Bandgap Voltage Reference", *IEEE Custom Integrated Circuits Conference*, 2004.
- [2] T. C. Carusone, "Analog Integrated Circuit Design", Second Edition, John Wiley&Sons, Inc., 2012.
- [3] H.Banba, H. Shiga, A. Umezawa, T. Tanzawa, S. Atsumi, and K. Sauki, "A CMOS Bandgap Reference Circuit with Sub-1-V Operation", *IEEE Journal of Solid-State Circuits*, Vol. 34, No. 5, May 1999.
- [4] P. Gray, P.J. Hurst, S.H.Lewis and R.G. Meyer, "Analysis and Design of Analog Integrated Circuits", 4th Edition, Wiley, 2001.
- [5] P. M. Petković. "Projektovanje CMOS integrisanih kola sa mešovitim signalima", Elektronski fakultet u Nišu, Niš, 2009.
- [6] Prezentacije iz predmeta Napredno računarsko projektovanje mikroračunarskih kola. Fakultet tehničkih nauka u Novom Sadu, Novi Sad, 2011.

Kratka biografija:



Ivan Bogosavljević je odbranio diplomski rad 2014. godine. Zaposlen je više od šest godina u kompaniji *Elsys Easter Europe* na poziciji projektant lejauta analognih integrisanih kola.

УПРАВЉАЊЕ КОНТРОЛОМ ПРИСТУПА НА ЗАТВОРЕНИМ ПАРКИНЗИМА

ACCESS CONTROL IN CLOSED PARKING LOTS

Саво Токић, Факултет техничких наука, Нови Сад

Област – МЕХАТРОНИКА

Kratak sadržaj – У овом раду описано је решење за контролу приступа кориштењем система са GSM модулом који је повезан на ПЛК преко серијске комуникације. За приступ паркингу довољно је посједовање мобилног телефона којим се позива GSM модул. Читав систем је аутоматизован али поред аутоматског рада потребно је имати могућност ручног рада система, као и праћење рада преко SCADA система.

Abstract – This paper describes an access control solution using the GSM module module that is connected to the PLK via serial communication. To access the parking lot, it is enough to own a mobile phone that calls the GSM module. The entire system is automated, but in addition to the automatic operation of the system, it is necessary to have the possibility of manual operation of the system, as well as monitoring system operation via SCADA system.

1. УВОД

У модерним индустријским и неиндустријским процесима управљање се најчешће врши програмабилно логичким контролерима (ПЛК). Програмабилно логички контролери (ПЛК) су индустријски рачунари чији је хардвер и софтвер посебно прилагођен раду у индустријским условима, а који се могу лако програмирати и уграђивати у постојеће индустријске и неиндустријске системе. ПЛК је, у суштини, микропроцесорски уређај који користи програмабилну меморију за памћење наредби којима се захтјева извођење специфичних функција, као што су логичке функције, секвенцирање, пребројавање, мјерење времена, израчунавање, а у циљу управљања различитим типовима машина и процеса преко дигиталних и аналогних улазно-излазних модула [1]. У овом раду је описано решење за контролу приступа кориштењем система са GSM модулом који је повезан са ПЛК преко серијске комуникације.

За приступ паркингу довољно је посједовање мобилног телефона којим се позива GSM модул. Читав систем је аутоматизован али поред аутоматског рада система потребно је имати могућност ручног рада система, као и праћење рада система преко SCADA система.

Рад система у ручном режиму се користи само у ријетким ситуацијама када је потребно ручно подићи рампу, услед неких непредвиђених ситуација или детекције отказа неког дијела система приликом редовног одржавања.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Стеван Станковски, ред. проф.

2. УПРАВЉАЊЕ КОНТРОЛОМ ПРИСТУПА

Задатак овог рада је да се да се осмисли, пројектује, и реализује систем за контролу приступа на затвореном паркингу користећи програмабилно логички контролер и GSM модул.

Систем треба да садржи три врсте корисника тј. три нивоа овлаштења приступа паркингу, а то су надзор, вип и обични корисници. Улазак и излазак је могућ само ономе ко је на списку корисника. На ПЛК је потребно направити програм који стално прати позиве на GSM терминалу. Ако је позив упућен од регистрованог корисника рампа ће се отворити и корисник може да уђе/изађе са паркинга. Корисник који је у списку за надзор може увијек да омогући улазак/излазак. Вип корисник значи да га увијек чека празно мјесто на паркингу. Обичан корисник може да користи паркинг само ако има слободних мјеста. У случају већег броја вип мјеста или да је надзор омогућио улазак, смањује се број мјеста за обичне кориснике. Приликом сваког уласка и изласка корисник мора да назове број који је на GSM терминалу. Рампа се не подиже у случају да су активни сензори испред и иза рампе истовремено.

3. GSM МОДУЛ

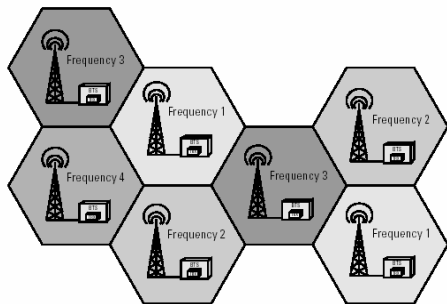
GSM је најраспрострањенији дигитални комуникациони систем који се и данас користи спада у мобилне комуникационе мреже друге генерације (2G). GSM је осмислила група формирана 1982. године са задатком да развије стандард за пан европску целуларну мобилну мрежу, а прва комерцијална GSM мрежа подигнута је 1992. године.

GSM Bands				
System	Reverse (MHz)	Forward (MHz)	Channel Width (kHz)	Max. No. of Channels
GSM-900	890-915	935-960	200	125
E-GSM	880-915	925-960	200	175
GSM-1800	1710-1785	1805-1880	200	375
GSM-1900	1850-1910	1930-1990	200	300

Слика 1. GSM опсези [2]

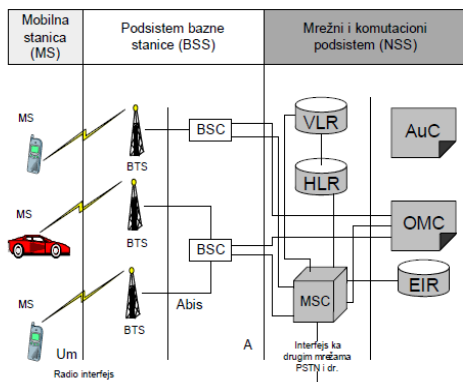
На слици 1 је приказана табела на којој се виде GSM опсези рада. Ти опсези рада представљају главне стандардне вриједности фреквенција рада GSM уређаја. GSM 900 и GSM 1800 представљају стандарде који се користе углавном у цијелом свијету. То је фреквенција на којој раде телефони у Европи као и у већем дијелу Азије и Аустралије. GSM 850 и GSM 1900 се користе у Америци. GSM користи целуларну структуру, на слици 2. Телефонска мрежа има задатак да повеже мобилне и

фиксне кориснике као и мобилне кориснике међусобно. Такође има задатак да обезбиједи везу између корисника који имају различите провајдере. Базична идеја целуларне мреже јесте подјела расположивог фреквенцијског опсега доступних фреквенција, и додјела дијела фреквенцијског спектра сваком базном примопредајнику, као и смањење домета базних станица како би било могуће поновно кориштење дефицитарних фреквенцијау несусједним ћелијама. Један од главних циљева мреже јесте да смањи интерференцију која се појављује између различитих базних станица.



Слика 2. Изглед једне целуларне мреже [3]

На слици 3 може се видјети архитектура једне GSM мреже. GSM мрежу чине: Мобилна станица (MS), Модул идентификације корисника (SIM картица), Базна примопредајна станица (BTS), Контролер базне станице (BSC), Комутациони центар мобилне телефоније (MSC), Регистар домаћих корисника (HLR), Регистар гостујућих корисника (VLR) и Регистар идентификација корисничке опреме (EIR).



Слика 3. Архитектура GSM мреже

Преко серијског комуникационог порта можемо вршити комуникацију са рачунаром, ПЛЦ-ом или неким другим уређајем. Размјена информација се врши слањем тзв AT команди. AT команде су стандардизоване и њихова синтакса почиње са словима AT. Разликујемо четири врсте AT команди: Test команде, команде које се шаљу опреми и очекује се листа параметара и вредносних опсега у кореспонденцији са одговарајућом Write командом или са неким процесом. Изглед синтаксе је следећи „AT+CXXX=?“. Read команде, ове команде враћају тренутно подешене вриједности параметара или неке параметре. Изглед синтаксе је следећи „AT+CXXX?“ Write команде, ове команде подешавају вриједности параметара. Изглед синтаксе је следећи: AT+CXXX=<...>. Execution команде, ове команде су команде извршења које читају непромјениве параметре одређене интерним процесором. Изглед синтаксе је следећи „AT+CXXX“.

4. УОПШТЕНО О ПЛК

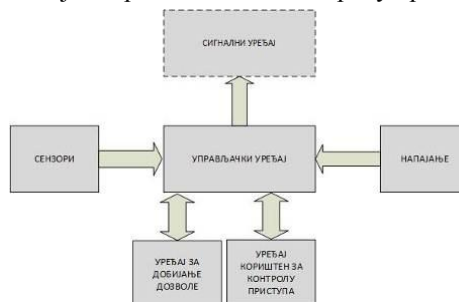
ПЛК нуди више могућности и далеко је поузданији него релејно управљање. Највећа предност ПЛК у односу на релејну технику је у томе што је управљање далеко једноставније. Управљање је централизовано на ПЛК и самим тим је количина опреме, која се користи, која је раније била велика сада драстично мања. Релеји се користе као актуатори тј. за укључење уређаја велике снаге и то само уколико је излазни степен ПЛК слаб да то одради. На улазни степен ПЛК се повезују сензори који дају улазне сигнале на основу којих се врши управљање цијелим системом. ПЛК има могућност прихватања различитих врста улазних и излазних сигнала како дигиталних тако и аналогних. Највећа предност ПЛК управљања у односу на релејну технику је у томе што је дијагностика кварова изузетно олакшана као и сама измјена управљачког система. Уколико је из неког разлога управљање потребно измјенити, додати неки сензор, склоп или сасвим измјенити логику рада често је потребно само измјенити само програмски код у ПЛК уз само мање измјене у управљачком ормару чиме је постигнута велика флексибилност самог уређаја. Флексибилност додатно побољшава и сам начин програмирања ПЛК чије је програмирање дефинисано у оквиру стандарда IEC61131-3[8].

5. АНАЛИЗА ПРОБЛЕМА

Да би се правилно испројектовао систем за контролу приступа потребно је знати неколико ствари о самом паркингу, као што су:

- Број мјеста на паркингу,
- Број корисника по нивоу приступа,
- Број улаза и излаза са паркинга и
- Захтјевани ниво безбједности паркинга.

Ове ставке су нам потребне како би правилно одабрали начин управљања паркингом. У овом случају број места на паркингу је шеснаест а број корисника је 19. Паркинг је грађен тако да има само један улаз на који је могуће ући аутом. На слици 4 су приказани типични елементи које садржи систем за контролу приступа.



Слика 4. Блок шема система за контролу приступа

Управљачки уређаји који се користе у оваквим системима у зависности од захтјева система су обично микроконтролери или ПЛК уређаји. У овом случају кориштен је ПЛК из разлога што је потребно остварити високу поузданост самог система. За контролу приступа паркингу постоји велики број решења која су тренутно у употреби. Најчешће су то рампе, потапајући стубићи, разне врсте капија и слично. На слици 5 је приказан улаз на паркинг прије уградње система.



Слика 5. Изглед улаза прије поставке система

Такође на слици 4 види се да су на приступном путу за улаз постављене бехатон коцке. За постављање потапајућих стубића потребно је пре свега припремити мјесто на коме ће стубић бити постављен. Такође за рад потапајућих стубића потребан је и извор ваздуха под притиском. Што сам систем чини јако компликованим и у већини случајева обезбјеђивање ваздуха под притиском представља изазов који је тешко решив, услед скученог простора у ормарима и немогућности постављања довољно квалитетног извора напајања. Самим тим ово решење је због горе наведених ставки је драстично скупље и захтјева далеко више рада на самој припреми. Аутоматске паркинг рампе као једно од главних средстава за контролу приступа се широко користе код јавних и приватних паркинг површина, код гаража, градских зона и пословно - стамбених објеката. Велика флексибилност, брзина, поузданост, квалитет, једноставност руковања, као и прихватљива цена су главне предности које рампе имају у контроли приступа. Због наведених особина као и саме конфигурације паркинга у овом случају аутоматска паркинг рампа је идеално решење за контролу приступа. У оваквом систему потребно је да имамо сензоре за детекцију присуства аута на улазу и излазу као и испод саме рампе, сензори су такође потребни за бројање аута која су унутра. Број сензора и њихова конфигурација зависи од тога на који начин се одобрава приступ. У овом случају се захтјева строго контролисан улазак и излазак са паркинга па самим тим је потребна конфигурација од три сензора. Један на улазу један испод рампе као и један после рампе. Ови сензори су потребни како би се детектовао пролазак аута. Одлука о месту и начину постављања сензора игра кључну улогу у исправном функционисању и безбедности система.

6. СИСТЕМ ЗА КОНТРОЛУ ПРИСТУПА

Начин рада система за контролу приступа на паркингу у просторима Департамана за грађевину Факултета Техничких Наука Универзитета у Новом Саду је следећи. Када се возило креће ка рампи тј. кад улазу у паркинг неопходно је да возач уочи сигнализацију са семафора:

- Црвено – Паркинг је пун и није могуће ући, осим за возаче са вип дозволом.
- Зелено – Паркинг има слободних мјеста и могуће је ући.

Ако је на семафору упаљено црвено свјетло возач без вип приступа нема никаквог разлога да се приближава паркингу, јер у њему нема слободних мјеста, тако да може потражити паркинг место на другој локацији.

Ако улазни семафор показује зелено, аутомобил треба зауставити испред рампе.

Како је систем конфигуриран, возило ће се зауставити на саму индуктивну петљу. Када је то учињено, активирана је улазна петља и систем је спреман да прими одговарајућу дозволу за улаз од стране корисника. Дозвола мора бити добијена помоћу позива на број телефона док GSM модул не одбије позив. Када се то деси рампа ће бити подигнута и ауто може да уђе на паркинг. Када ауто прође рампа се спушта истог тренутка када ауто напусти петљу. Уколико се ауто нађе на средишњој петљи у интервалу од двије секунде од напуштања последње петље рампа се поново диже како би се избјегло ударање рампе у ауто и оштећење истог или повређивање корисника. Процедура изласка је иста с тим што ауто мора да се позиционира на излазну петљу. Рампа се неће подићи уколико се и на улазу и на излазу налази ауто. У том случају неко мора да се склони и да се поново позове рампа. Када ауто уђе на паркинг врши се инкрементирање бројача, супротно се ради приликом изласка са паркинга када се бројач декрементира.

6.1. Аутоматска рампа

Рампа представља физичку препреку која забрањује пролаз неовлашћеним возилима. Систем за контролу приступа на паркинг простор као централни актуаторски систем садржи аутоматску рампу произвођача CAME. Аутоматске рампе CAME су намењене за интензивну употребу. Сви модели су идеално избалансирани, брзи (отварање и затварање могуће чак и за 1,5 секунди) и имају трогодишњу гаранцију. Модели се разликују у зависности од дизајна и од дужине ручице.

6.2. Индуктивне петље

Индуктивне петље су намјењене за бесконтакнуетекцију возила. Свој рад заснивају на елетромагнетском принципу. У датом систему користе се три петље. Једна код улазне рампе, једна непосредно испод ручице рампе и једна петља за контролу изласа. Петља се формира намотавањем (двоструко, троструко или четвороструко), флексибилног, електричног кабла, који се у једном делу и упреда и формира завој слика 6. Овако направљен кабал је пожељно поставити у изолациону цев или бужир да би се обезбједила механичка и термичка заштита приликом постављања петљи. Петља се понаша као сензор, повезана је са управљачким ормаром и омогућава безконтактну детекцију металних објеката. Петља је повезана са детектором са $2 \times 1.5 \text{ mm}^2$ уплетеним, изолованим каблом.



Слика 6. Дебљина пресека

Кабал мора имати најмање петнаест завоја по метру у дијелу у коме се кабал уплиће. Максимална дужина кабла једне петље је сто метара. Употреба оклопљеног STP (Shielded Twisted Pair) кабла је обавезна у случају да се неколико каблова постављају у исти отвор (две или више петљи) или у нестабилним окружењима (високо-

напонски каблови, итд). Веза петљи са управљачким ормаром, тачније пријемником, би требала бити што краћа.

Табела 1. Број кругова који се мота на петљу у зависности од површине

Област	Број кругова
$<3\text{m}^2$	4
$3\text{-}5\text{m}^2$	3
$6\text{-}10\text{m}^2$	2

Облик индуктивних петљи у највећем броју случајева је правоугаоног или ређе квадратног облика. У специјалним случајевима може бити ромбоидног, шестоугаоног или чак елипсастог или кружног облика.

6.3. UNITRONICS V350-35-R34

Овај ПЛК припада тзв. класи микро ПЛК. Посједује двадесетдва дигитална улаза и два аналогна улаза као и дванаест релејних излаза. Осим тога посједује и екран величине 3,5 инча. Овај контролер се испоручује са једним серијским портом за комуникацију и програмирање. Такође има и један мини USB који се користи за програмирање.

6.4. Управљачки систем рампе

Приликом пројектовања управљачког ормара потребно је прије свега изградити електро шему. На основу електро шеме бира се број редних стезалки.



Слика 7. Завршен и постављен ормар

Када се утврде сви елементи који су потребни за ормар врши се њихово премјеравање а затим и утврђивање њиховог распореда у ормару. Након тога на основу димензија и жељеног распореда елемената утврђује се величина ормара који је потребан. Након тога се приступа изради ормара. На слици 7 је слика завршеног и постављеног ормара.

7. ЗАКЉУЧАК

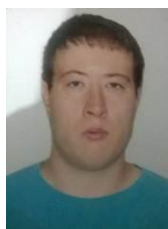
Употреба аутоматизованих рампи у значајној мери повећала је ефикасност система у погледу флексибилности и брзине рада. Кориштење GSM технологије за дозволу приступа показало се прилично практично. Одликује је једноставност и поузданост рада. Могућности злоупотреба су практично сведене на нулу. Практично једино што је потребно је мобилни телефон корисника што увелико појефтиније кориштење овог система. У коду је елиминисана могућност вишеструког пуштања корисника на паркинг. Једини са том дозволом су корисници надзора који су уједно овлашћени

за одржавање система. Једно од могућих унапређења је је додавање система видео надзора као и скада система који би употпунио безбједносни аспект цијелог система тј његових корисника. Приказано решење у потпуности задовољава постављене захтеве, и то пре свега контролисани улазак и излазак, са кратким задржавањем приликом уласка, односно изласка. Задржавање приликом кориштења система зависи од брзине реаговања система, брзине позивања GSM модула и брзине подизања и спуштања ручице рампе. Мора се напоменути да дати паркинг простор добро саобраћајно уређен и постављена је правилна саобраћајна сигнализација. Уређен је и ограђен паркинг простор што у великој мери олакшава проблеме контроле и аутоматизације система.

8. ЛИТЕРАТУРА

- [1] Контрола приступа паркинзима универзитета у Новом Саду, Александар Шипоски, 2011.
- [2] Overview Of The GSM System and Protocol Architecture, Moe Rahema
- [3] Архитектура, протоколи и сервиси GSM мрежа, Марко Лалић
- [4] drakic.files.wordpress.com/2012/02/predavanje-7.ppt, (Приступило 25.09.2017.)
- [5] GSM Networks, protocols and implementation, Gunnar Heine
- [6] http://www.portech.com.tw/data/Siemens%20mc55-i_AT.pdf, (Приступило 25.09.2017.)
- [7] Програмабилни логички контролери и комуникациони протоколи у електроенергетици, Дарко Марчетић, Марко Гецић, Борис Марчетић, Нови Сад 2016
- [8] Превођење програмског кода из лествичастог дијаграма у структурирани текст у оквиру стандарда IEC 61131-3, Саво Ђокић, Стеван Станковски, 2015
- [9] <http://www.came.it>, (Приступило 25.09.2017)

Кратка биографија:



Саво Ђокић рођен је у Бијељини 1991. год. Основне академске студије на Факултету техничких наука у Новом Саду уписује 2010 год, на смијеру Мехатроника. Мастер академске студије уписује на истом факултету 2015. године на смијеру Мехатроника, роботика и аутоматизација. Мастер рад одбранио 2017. године.

**PREDLOG REŠENJA POLUAUTONOMNOG UPRAVLJANJA NA ADAS PLATFORMI
PROPOSED SOLUTION OF SEMIAUTONOMOUS DRIVING ON ADAS PLATFORM**Marko Markov, *Fakultet tehničkih nauka, Novi Sad***Oblast – MEHATRONIKA**

Kratak sadržaj – Ovaj rad se bavi problematikom skladištenja podataka primljenih sa digitalnog kalorimetra na klaud servis pod imenom Firebase. Kao rezultat rada, razvijen je softver u programskom okruženju Arduino, koji je zadužen da podatke temperature i protoka primi preko serijskog porta, integriše ih sa podacima sa uređaja o GPS poziciji i vremenskim trenutkom i pošalje ih na klaud servis. Za komunikaciju sa servisom zadužen je GSM modul koji se povezuje sa internetom preko SIM kartice. Time je omogućena konekcija u većem delu države.

Abstract – This paper is related to the issue of storing data collected from digital calorimeter on to cloud service called Firebase. As a result, a software was created in programming environment called Arduino. Software is in charge of getting temperature and flow data over serial port, integrating them with data from GPS module and timestamp and sending them to cloud service. Service communication is established through GSM module, that connects through internet over SIM card. By using SIM card the connections is possible in most parts of the country.

1. UVOD

Kalorimetar^[1] je uređaj za merenje količine toplote koju je apsorbovalo neko telo ili koja se oslobađa pri sagorevanju određene količine goriva. Koristi se za obračunavanje potrošnje toplotne energije u zagrevanim objektima priključenim na sistem daljinskog grejanja. Može se koristiti u kombinaciji sa: mehaničkim vodomernom (bilo da je u pitanju reed rele, blizinski prekidač ili fotočelija), ultrazvučnim ili sa elektromagnetnim vodomernom. Pored protoka fluida, potrebno je meriti ulaznu i izlaznu temperaturu. U zavisnosti od broja ulaznih veličina, kalorimetar omogućava očitavanje dolazne i odlazne temperature na toplovodu, trenutni i ukupni protok tople vode kao i trenutne snage i utrošene toplotne energije. Uređaj se zasniva na modernim mikrokontrolerima što mu daje brzinu rada neophodnu za postizanje visoke tačnosti merenja, jednostavnost u dizajnu i proizvodnji.

Funkcija uređaja je da broji impulse dobijene od vodomera i da meri temperature u dolaznoj i odlaznoj grani toplovođa. Na osnovu dobijenih rezultata, izračunavaju se svi parametri neophodni za praćenje utroška toplotne energije.

Grejanje po utrošku se više isplati od daljinskog jer na ovaj način mesečni račun može biti i za trećinu niži. Ideja za uređaj opisan u ovom radu javila se posmatranjem postojećih rešenja.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Stankovski, red. prof.

Javila se potreba napraviti daljinsko nadgledanje potrošnje energije u jedinstvenoj bazi podataka, kojoj bi samo određeni ljudi imali pristup. Uglavnom bi to bila nekolicina ljudi zaduženi za unapređenje postojećih metoda regulacije potrošnje goriva. Podaci bi bili korišćeni u cilju predviđanja naglih skokova potrošnje gde bi se adekvatno reagovalo na vreme sa optimalnim utroškom gorive jedinice. Na taj način se postojećem rešenju dodaje Data logger (uređaj opisan u ovom radu) koji ima mogućnost lokalnog skladištenja podataka kao i periodičnog slanja istih na jedinstveni server. Svaki uređaj ima jedinstveni identifikacioni broj koji služi u prepoznavanju uređaja. Pored identifikacionog broja na server se šalje ulazni protok tople vode u radiator, izlazni protok, ulazna temperatura, izlazna temperatura, temperatura u sredini radiatora, ambijentalna temperatura kao i geografski podaci.

2. TEORIJSKE OSNOVE

Za razumevanje rešenja problema potrebno je za početak upoznati se sa principom rada pojedinačnih komponenti od kojih je uređaj sastavljen. Jedne od najvažnijih su svakako mikrokontoler (Atmega328p), GPS modul (GY-GPSV3-NEO), GSM modul za slanje i primanje poruka sa interneta (SIM900), RTC tj. real time clock (DS1307), SD kartica za skladištenje podataka (Kingston 2GB) i modul za bežičnu komunikaciju (nRF24L01). Pored glavnih komponenti uređaj sadrži i mnoge druge komponente zadužene za kontrolu i upravljanje baterijskim napajanjem, komunikacijom sa drugim modulima i dr.

2.1. Mikrokontroleri

Mikrokontoler predstavlja mini računar na jednom integrisanom kolu. U modernoj terminologiji sličan je takozvanom SoC (*System on Chip*), s tim da se u SoC-u on nalazi kao samo jedna od komponenata. Mikrokontoler sadrži centralnu procesorsku jedinicu (CPU), RAM memoriju (operativnu memoriju), fleš memoriju (memorija za skladištenje programskog koda i promenljivih) kao i programabilne ulazno-izlazne pinove. Oni su dizajnirani za embeded aplikacije kao i za komunikaciju sa mikrop procesorima korišćenim u personalnim računarima (PC).

2.2 Globalni pozicioni sistem-GPS

GPS^[2] je trenutno jedini potpuno funkcionalan globalni satelitski navigacioni sistem. GPS se sastoji od 24 satelita raspoređenih u orbiti Zemlje koji šalju radio signal na površinu Zemlje. GPS prijemnici na osnovu ovih radio signala mogu da odrede svoju tačnu poziciju: nadmorsku visinu, geografsku širinu i geografsku dužinu na bilo kom mestu na planeti danju i noću, pri svim vremenskim uslovima.

GPS prijemnik je uređaj koji proračunava svoju poziciju na osnovu merenja udaljenosti od tri ili više GPS satelita. Pošto su poznate pozicije tri satelita i udaljenost prijemnika od svakog od njih, postupkom triangulacije se može odrediti pozicija prijemnika. Triangulacija se bazira na činjenici da se tri sfere seku u najviše dve tačke (od kojih jedna obično nema smisla).

2.3 Globalni sistem za mobilne komunikacije – GSM

GSM^[3] je standard koji je razvio Evropski institut za telekomunikacijske standarde (ETSI) da bi opisao protokole za digitalne mreže mobilne telefonije druge generacije koje koriste mobilni uređaji. Unutar tog standarda spadaju 2G, 3G i 4G mreže.

2G mreža omogućava glasovne pozive, slanje/primanje SMS/MMS poruka, kao i osnovne internet usluge. 3G mreža se pojavila kao potreba za unapređenjem brzine prenosa podataka. Daljim razvijanjem nastala je 4G mreža koja pruža superiornije brzine prenosa podataka (3-4 puta veće od 3G). Merač protoka će ovaj standard koristiti za razmenu podataka preko interneta. SIM900 GSM modul Preko SIM kartice je zadužen za primanje i slanje poruka.

2.4 RF komunikacija (bežična komunikacija)

Za potrebe prenosa informacije između mrežnih komponenta, bežične komunikacije ne koriste se fizički vodovi nego radio talasi. Bežične mreže mogu obezbediti mrežni pristup telefonima, računarima, aplikacijama, bazama podataka i internetu, i to na nivou zgrada, između zgrada, između naselja i između udaljenih lokacija. Korisnici imaju mogućnost da kopiraju, pribavljaju, upravljaju ili manipulišu podacima virtuelno sa bilo koje lokacije.

Postoje četiri glavna sredstva koja se kod bežičnih komunikacija koriste za prenos informacije a ona se baziraju na:

- Radio-prenosu,
- Mikrotalasnom-prenosu,
- Satelitskom prenosu i
- Optičko-infracrvenom prenosu.

Najstarije bežične aplikacije su one koje koriste radio talase za potrebe emisije u etar (broadcast), sa jedne strane i individualnog prijema, sa druge strane. Zadnjih nekoliko godina radio-frekventne (RF) tehnologije su evoluirale od analognih sistema do digitalnih, tj, od onih koje su ne tako davno prenosile samo govor i sliku, do onih koje danas prenose i podatke.

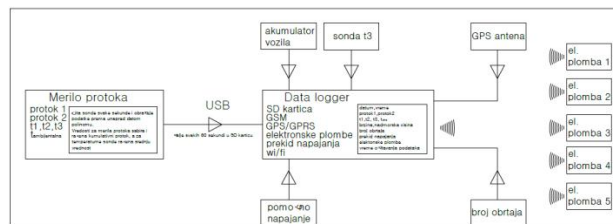
Radio-talasni modul je obično mali elektronski uređaj koji se koristi za prenos i / ili prijem radio signala između dva uređaja. Ova komunikacija je bežična i može biti ostvarena optičkom komunikacijom ili putem RF komunikacije. Ima široku primenu jer ne zahteva fizički vidljivu vezu. RF sistem se sastoji iz predajnika i prijemnika. Oni su različitih vrsta i opsega. Neki mogu obavljati funkciju i na udaljenosti od 1000 m. Komercijalno dostupni RF moduli, uključujući one u industrijskim, naučnim i medicinskim oblastima, koriste frekvenciju u opsezima od 433.92 MHz, 915 MHz i 2400 MHz.

3. KONCEPT REŠENJA

Slika 1. ilustruje blok šemu principa rada. Sa leve strane se nalazi digitalni kolorimetar. On je zaseban i nezavistan modul koji sadrži svoj programski kod koji je zadužen za merenje ulaznog i izlaznog protoka, ulazne temperature, temperature u sredini radijatora i izlazne temperature. Podaci se zajedno sa ID (identifikacionim brojem), šalju u *Data logger* (uređaj opisan u ovom radu).

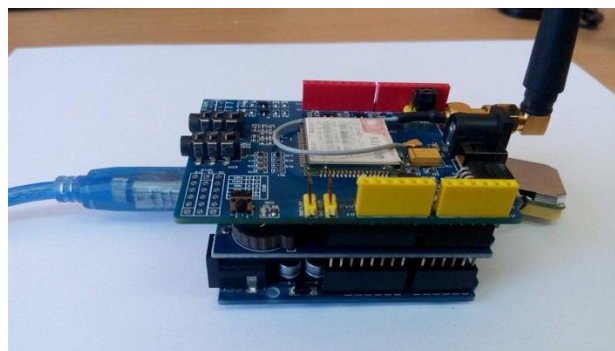
Data logger, pored ovih informacija dobija i signale bežično sa tzv. elektronskih plombi, kojih ima 5, preko nRF-24L01 modula. Opciono je ostavljena mogućnost merenja broja obrtaja potencijalnog motora, koji bi se kontrolisao i dodatna sonda (t3) koja bi unela veću preciznost. Uređaj može biti baterijski napajan i/ili regularno priključen na mrežu. Pored toga sa GPS modula dobija se informacija o geografskim koordinatama i svi podaci se čuvaju na SD kartici i šalju preko GSM-a na klaud.

Elektronske plombe predstavljaju jednostavne granične prekidače povezane sa mikrokontrolerom koji sadrži nRF24L01 modul za bežičnu komunikaciju. Ovo su baterijski napajani moduli, koji su u uspavanom režimu sve dok se granični prekidač ne uključi. Tada prelaze u režim normalnog rada i šalju tu informaciju glavnom uređaju nakon čega se vraćaju se u uspavani režim. Na taj način obična litijumska baterija od 3V (CR2030) može da ih drži aktivnim više od godinu dana. Njihova uloga jeste da jave da li je kućište uređaja bilo otvoreno i ako jeste gde i kad (bitna informacija vezana za garanciju).



Slika 1. Blok šema uređaja

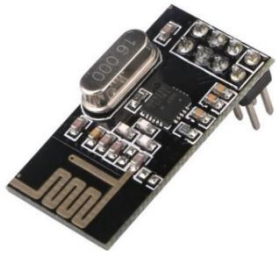
Na Slici 2. nalazi se izgled uređaja u krajnjem stadijumu. Prijemna strana plombe povezuje se preko pinova sposobnih za hardverski SPI. To su Pinovi 13, 12, 11, 10.



Slika 2. Uređaj u krajnjem stadijumu

3.1 RF Komunikacija između uređaja

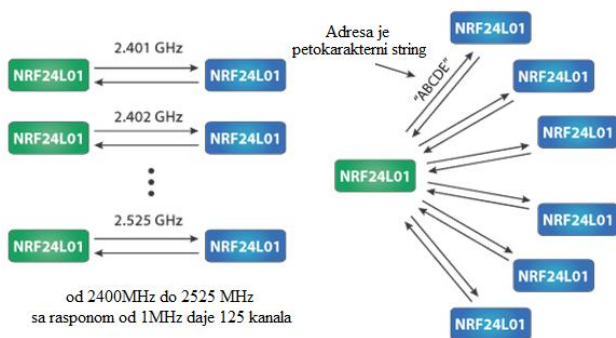
Za celokupnu bežičnu komunikaciju zadužen je primo-predajnik nRF24L01, slika 15. Sa frekvencijom prenosa od 2.4GHz može imati brzinu prenosa u rasponu od 250kb/s do 2Mb/s. U koliko radi u otvorenom prostoru bez prepreka, može dostići do met i do 100m.



Slika 3. Izgled nRF24L01 modula

Modul može komunicirati na 125 različitih kanala, odnosno, može raditi 125 nezavisnih modula. Svaki kanal može imati i do 6 adresa, što govori o tome sa koliko uređaja može u isto vreme komunicirati.

Na Slici 4, vidi se komunikacija na više kanala sa leve strane kao i komunikacija sa više uređaja na desnoj strani. Svaki kanal se razlikuje od prethodnog za frekvenciju od 1MHz tako da modul radi u rasponu od 2.4GHz do 2.525GHz.



Slika 4. Komunikacija na mogućim kanalima i komunikacija sa drugim modulima

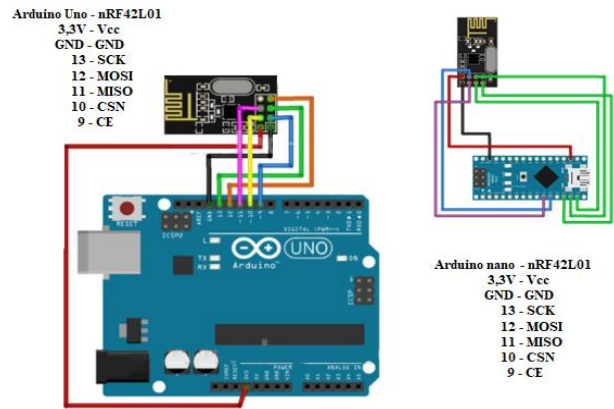
Struja pri slanju od 12mA govori o maloj potrošnji modula, dok on može raditi na rasponu napona od 1.9V do 3.6V. Pozitivna strana je činjenica da su pinovi za komunikaciju sa modulom tolerantni na 5V i mogu se direktno povezati sa mikrokontrolerom.

Sam modul komunicira preko SPI protokola i koristi 3 pina u tome (MISO, MOSI, SCK). Pinovi CSN i CE mogu biti povezani za bilo koji digitalni pin i oni služe za prebacivanje modula iz moda čekanja u aktivni mod (i obrnuto) i za prebacivanje iz komandnog u predajni mod (i obrnuto). Raspored pinova prikazan je sa Slici 5.^[5]



Slika 5. Raspored pinova nRF24L01 modula

Na Slici 6. prikazana je blok šema povezivanja Arduino UNO pločice sa modulom i povezivanje plombe, Arduino nano sa modulom.



Slika 6. Blok šema povezivanja RF modula i mikrokontrolera

4. PROGRAMSKO REŠENJE

Programski kod pisan je u Arduino razvojnom okruženju, u istoimenom programskom jeziku. Arduino programski jezik baziran je na C++, tako da su i biblioteke pisane u C++. Moguće je pisati i kod u C programskom jeziku, za podešavanje registara, kao i u assembleru. Za komunikaciju sa postojećim uređajima korišćene su gotove biblioteke.

GPS modul *GY-GPSV3-NEO* komunicira preko UART komunikacionog protokola. Za komunikaciju sa mikrokontrolerom zadužena je TinyGPS++ biblioteka kojoj je potrebna pomoćna biblioteka SoftwareSerial. Njena uloga je da softverski otvori port koji će se koristiti za komunikaciju, kako su svi hardverski portovi zauzeti. Nakon toga treba napraviti objekat i inicijalizovati komunikaciju serijskog porta odgovarajućom brzinom (Boud rate) koja je u ovom slučaju 9600. Zatim je potrebno dohvatiti podatke pozivom metode `gps.location.lat`, `gps.location.log` i `gps.altitude.meters` U koliko su podaci validni, smeštaju se u promenjive predviđene za njihovo privremeno skladištenje u suprotnom promenjive se postavljaju na 0. Ovim se završava manipulacija sa GPS podacima, teži deo posla obavlja sam modul.

Dobijanje tačnog vremena i datuma vrši se uz pomoć RTC modula *DS1307* koji komunicira preko I²C komunikacionog protokola. Za komunikaciju sa mikrokontrolerom zadužena je RTCLib biblioteka. Ona ne može funkcionisati samostalno i potrebno je uključiti i pomoćnu biblioteku Wire. Nakon inicijalizacije, podaci se povlače preko metoda `now.year`, `now.month`, `now.day`, `now.hour`, `now.minute`, `now.second`.

GSM modul *SIM900* komunicira preko UART komunikacionog protokola i zadužen je za komunikaciju sa mikrokontrolerom. Program koristi SIM900 biblioteku koja šalje AT komande a koristi metode sintaksno pojednostavljene korisniku. Prilikom otvaranja konekcije ka serveru potrebno je konstruktoru proslediti određene parametre. Parametri koji se prosleđuju su pin kod same SIM kartice koju koristimo, u slučaju da je zaključana (mada je preporučljivo ubaciti je u telefon i otključati pre korišćenja), APN za korišćenu mrežu (Tabela 1.), APN korisničko ime i lozinka, pin koji će služiti za softversku kontrolu stanja modula, hardverska serijska komunikacija.

Upotrebom RESTfull servisa komunicirati sa bazom uz pomoć GET i POST metoda.

PROJEKTOVANJE HIDRAULIČNE POLUPLATFORME ZA KIPOVANJE DRUMSKIH TRANSPORTNIH SREDSTAVA SA RASTRESITIM TERETOM**DESIGN OF HYDRAULIC SEMI PLATFORM FOR THE UNLOADING OF ROAD TRANSPORT VEHICLES WITH A GRANULATED LOAD**

Nikola Pucar, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MEHATRONIKA

Kratak sadržaj – U ovom radu je isprojektovana hidraulična kip poluplatforma za kipovanje drumskih transportnih sredstava sa rastresitim teretom. Prikazana je hidraulična šema upravljanja. Izvršen je neophodan proračun. Kip platforma ima mogućnost upravljanja putem mobilne android aplikacije, a pored toga urađena je i standardna električna šema upravljanja u relejnoj tehnici.

Abstract – In this paper, a hydraulic semi platform for the unloading of road transport vehicles with a granulated load is designed. A hydraulic control scheme is displayed. The necessary calculate has been executed. The semi platform has the ability to operate with a mobile android application, and in addition, a standard electrical circuit diagram for relay technology has been developed.

1. UVOD

Ulaganje u automatizaciju tehnoloških procesa u današnje vreme, a i u budućnosti, jeste poduhvat koji zasigurno donosi napredak u industrijskoj proizvodnji i samim time, ostavlja Vas u konkurenciji na beskompromisnom tržištu, tamo gde zahtevi i očekivanja korisnika iz dana u dan rastu, traži se povećanje kvaliteta i usluge, dok gubitak koraka za „većima“, čini da sve što je do tada učinjeno bio samo dobar pokušaj. Zahvaljujući globalizaciji i tehnološkom napretku, svet i način života se menjaju velikom brzinom, tako da se ljudi u budućnosti moraju prilagoditi tehnologiji, za razliku od prošlosti, kada se tehnologija prilagođavala njihovim potrebama. U skladu sa tim, jasno je da danas ne postoji fabrika ili manje preduzeće koje ne sadrži makar jedan delić automatizacije proizvodnje. Automatizacija označava tok prenosa rada čoveka na mašine, obično kroz tehnički napredak. U industriji se nadograđuje na već postojeću mehanizaciju. Dok mehanizacija rada omogućava ljudima u pogonu lakše uslove rada, odnosno automatizacija smanjuje ljudsku potrebu za prisutnost u obavljanju određenih delatnosti. Automatizacija igra važnu ulogu u svetskoj ekonomiji. Ciljevi automatizacije su:

- Smanjenje ili eliminisanje neposrednog učešća čoveka u procesu
- Skraćenje ciklusa proizvodnje
- Skraćenje roka izrade
- Povećanje proizvodnosti
- Povećanje kvaliteta
- Povećanje sigurnosti

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Mitar Jcanović, van.prof.

Kako je prethodno bilo govora o automatizaciji, nadalje može se naslutiti da je cilj ovog rada predstavljanje automatizovanog postupka izvođenja nekog procesa, konkretno istovara zrnastog tereta iz prevoznog sredstva, u okviru prijema sirovina u mlinovima, silosima, fabrikama stočne hrane i slično. Rad sadrži hidrauličnu šemu upravljanja kao i proračun za izbor komponenti potrebnih za funkcionalnost rada.

2. TEORIJSKE OSNOVE

Kip platforme su namenjene za istovar rasutog tereta kao što su: žitarice, korenski usevi, jabuke i ostali zrnasti materijali. Istovar se ostvaruje na taj način tako što hidroinstalacija podiže platformu koja se zajedno sa vozilom naginje za određeni ugao, te tovar klizi i pada iz vozila u usipni koš.

Kip platforme koje su najzastupljenije u industriji mogu se podeliti na tri vrste:

- Jednostazna kip platforma (slika 1)
- Jednostazna nagibna kip platforma
- Potpuna kip platforma



Slika 1. Jednostazna kip platforma (poluplatforma)

Ove kip platforme su konstruisane tako da se podiže samo jedna strana vozila sa teretom, i za vreme podizanja platforma ostaje paralelna sa horizontalnom ravni. Mana ove platforme jeste u nepotpunom kontaktu između platforme i guma prevoznog sredstva, te stoga spadaju u najnebezbednije platforme jer postoji šansa proklizavanja, ali samim tim su najjeftinije za ugradnju. Jednostazna nagibna kip platforma (slika 2) je poboljšana verzija prethodne kip platforme, koja se razlikuje u tome što se kod ove platforme na vrhove klipnjače cilindra postavljaju spojnice koje ostvaruju zglobovu vezu, koja omogućava zakretanje platforme u pravcu naginjanja vozila prilikom podizanja tereta. Ove platforme su pouzdanije od prethodnih.



Slika 2. Prikaz jednostazne nagibne kip platforme

Treći tip platforme koji se koristi jeste onaj kod kojeg je širina platforme veća od širine vozila, koje se pri tome u potpunosti nalazi na platformi i prilikom istovarivanja tereta dolazi do zakretanja i platforme i vozila na stranu istovara. Na samom kraju platforme se postavlja čelična konstrukcija koja ne dozvoljava da vozilo sklizne sa platforme prilikom istovarivanja. Ovaj tip platforme (potpuna kip platforma) je najbezbednije i najbolje rešenje, ali je zbog toga i cena srazmerna kvalitetu. Prikazana je na slici 3.



Slika 3. Prikaz potpune kip platforme

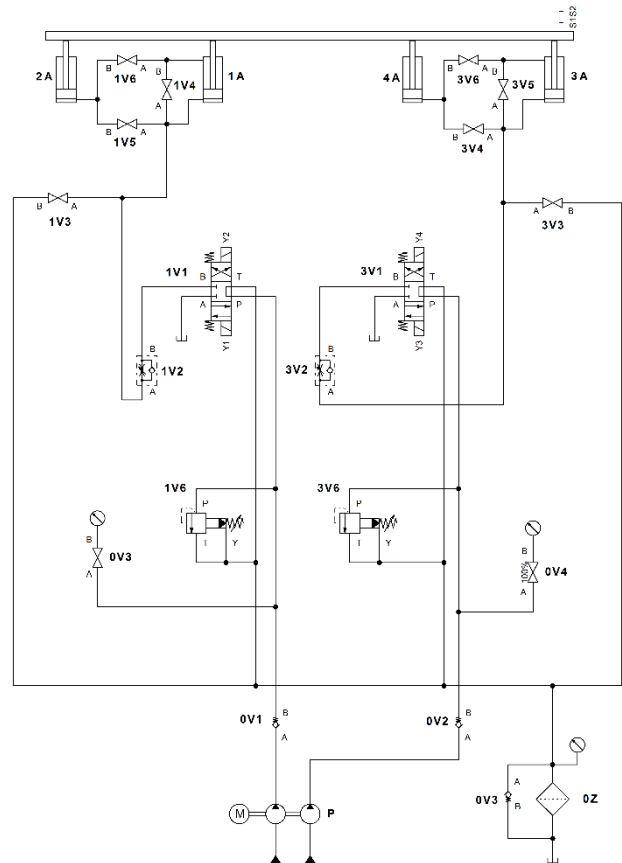
3. HIDRAULIČNA ŠEMA UPRAVLJANJA

Sistem pokreće elektromotor snage 11 kW, dok su za snabdevanje radnog medijuma iskorišćene dve zupčaste pumpe kapaciteta 27 l/min. Upravljačke elemente sačinjavaju 2 elektromagnetna ventila sa 4 priključka i 3 razvodna položaja. Izvršne elemente predstavljaju 2 hidraulična cilindra dvosmernog dejstva i 2 cilindra jednosmernog dejstva tj. 2 plužer cilindra.

Pritiskom na taster „PODIZI_PLATFORMU“ započinje se ciklus istovarivanja tereta tj. platforma se podiže. Hidraulično ulje iz pumpe kreće preko priključka A i priključka B elektromagnetnih ventila da puni zadnju komoru cilindra 1A i 3A. Podizanjem klipova ova dva cilindra pune se komore druga dva cilindra čija je zapremina jednaka klipnjačinoj zapremini cilindra 1A i 3A i oni se počinju izvlačiti. Podešavanjem prigušno–

nepovratnih ventila prilagođava se brzina spuštanja platforme. S2 granični prekidač predstavlja maksimalni položaj podizanja platforme. Dva ventila za ograničenje pritiska – indirektnog dejstva imaju zadatak da održavaju vrednost pritiska na podešenoj vrednosti i sprečavaju da pritisak u sistemu pređe dozvoljenu–podešenu vrednost, u ovom slučaju 100 bar-a. Hidraulične slavine 1V4 i 3V4 su iskorišćene kao pomoć pri rasterećenju sistema prilikom podizanja i spuštanja platforme.

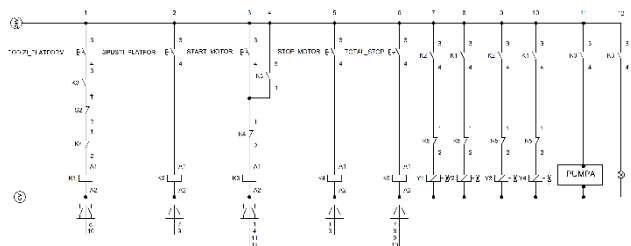
Pritiskom na taster „SPUSTI_PLATFORMU“ dolazi do spajanja priključaka B i T i sada se ulje iz cilindra vraća u rezervoar, odnosno platforma se spušta u početni položaj. Hidraulična šema upravljanja je prikazana na slici 4.



Slika 4. Hidraulična šema upravljanja

4. ELEKTRIČNA ŠEMA UPRAVLJANJA

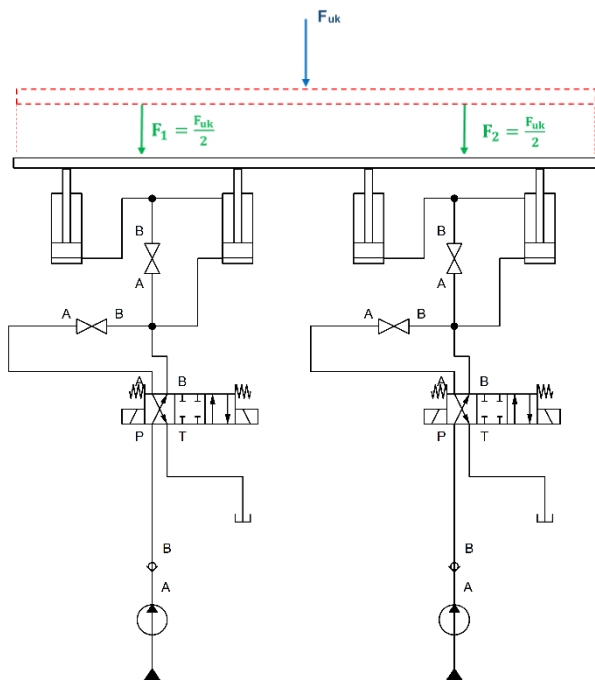
Napajanje mreže iznosi 24 V. Šema (slika 5) prikazuje 4 tastera kojima se aktiviraju releji, čiji kontakti upravljaju elektromagnetima razvodnih ventila (Y1, Y2, Y3 i Y4).



Slika 5. Električna šema upravljanja

5. HIDRAULIČNI PRORAČUN

U ovom poglavlju je prikazan proračun: hidrauličnih cilindara, kapacitet pumpi, snaga elektromotora. Krenućemo od najjednostavnijeg modela, sa koga se vidi da se ukupna sila deli sa dva zbog deljenja opterećenja duž platforme.



Slika 6. Raspodela ukupne sile na platformu

$G = 30 \text{ t}$ – nosivost platforme

$F = 30000 \cdot 9,81 = 294300 \text{ N}$ – sila kojom kamion opterećuje platformu

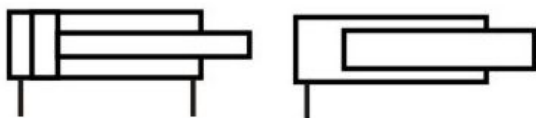
$M = 3,5 \text{ t}$ – masa konstrukcije

$F_k = 3500 \cdot 9,81 = 34335 \text{ N}$ – opterećenje konstrukcije

$F_{uk} = F + F_k \approx 329000 \text{ N}$

$F_1, F_2 = F_{uk} \div 2 = 164500 \text{ N}$

Usvojeni prečnici cilindara su:



1A i 3A: $D_1=160 \text{ mm}$

$d_1=110 \text{ mm}$

$L=1115 \text{ mm}$

2A i 4A: $D_2=116,2 \text{ mm}$

$L=1115 \text{ mm}$

$$D_1=160 \text{ mm} \rightarrow A_1 = \frac{D_1^2 \cdot \pi}{4} = \frac{160^2 \cdot \pi}{4} = 0,02 \text{ m}^2$$

$$d_1=110 \text{ mm} \rightarrow a_1 = \frac{d_1^2 \cdot \pi}{4} = \frac{110^2 \cdot \pi}{4} = 0,0095 \text{ m}^2$$

Površina klipnog prstena:

$$A_{p1} = A_1 - a_1 = 0,02 - 0,0095 = 0,0106 \text{ m}^2$$

$$D_2=116,2 \rightarrow A_2 = \frac{D_2^2 \cdot \pi}{4} = \frac{116,2^2 \cdot \pi}{4} = 0,0106 \text{ m}^2$$

Željeni pritisak kroz sistem se računa:

$$p = \frac{F}{A} = \frac{164500}{0,02} = 82 \text{ bar}$$

Kako su klipne i klipnjačine komore susednih cilindara povezane, a njihove klipnjače povezuje platforma, sledi da su hodovi klipova jednaki i to znači jednake su i zapremine kao i brzine kretanja izvršnih elemenata, što se može videti iz sledećih jednačina:

$$V_2 = V_{p1} \quad \text{i} \quad v_1 = v_2$$

Zapremina cilindara A1 i A3:

$$V_1 = A_1 \cdot L = 0,02 \cdot 1,115 = 0,02241 \text{ m}^3$$

Zapremina cilindara A2 i A4:

$$V_2 = A_2 \cdot L = 0,0106 \cdot 1,115 = 0,0118 \text{ m}^3$$

Zapremina klipnjačine komore cilindara A1 i A2:

$$V_{p1} = A_{p1} \cdot L = 0,0106 \cdot 1,115 = 0,0118 \text{ m}^3$$

Kapacitet pumpe se proračunava:

$$Q = A_1 \cdot v = 0,0201 \cdot 10^2 \cdot 0,02 \cdot 10 \cdot 60 = 24,12 \frac{\text{dm}^3}{\text{min}}$$

Usvaja se potreban kapacitet pumpe: $Q = 27 \frac{\text{l}}{\text{min}}$

Potrebna snaga elektromotora jednaka je:

$$P = 2 \cdot \frac{p \cdot Q}{600 \cdot \eta_{uk}} = 2 \cdot \frac{82 \cdot 27}{600 \cdot 0,85} = 8,68 \text{ kW}$$

Usvaja se snaga elektromotora: $P = 11 \text{ kW}$

Brzina izvlačenja klipnjača cilindra se računa:

$$v = \frac{Q}{A} = \frac{27 \cdot 10^{-3}}{60 \cdot 0,02} = 0,02 \frac{\text{m}}{\text{s}}$$

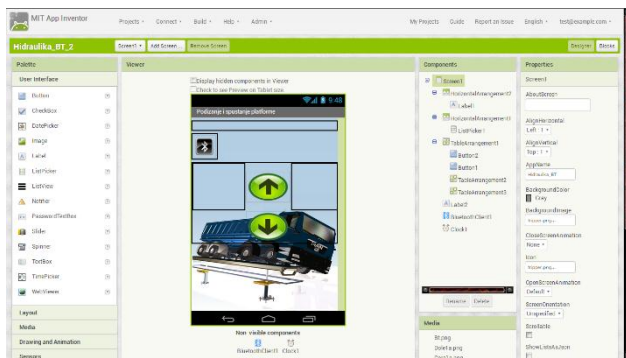
5. UPRAVLJANJE KIP PLATFORMOM POMOĆU TELEFONSKE APLIKACIJE

Kao što je bilo govora ranije kip platforma je realizovana da se upravlja putem android aplikacije „Hidraulika–_BT_2“. za pametne telefone (slika 7). Aplikacija je napravljena u MIT App Inventor programu.



Slika 7. Prikaz android aplikacije za upravljanje kip platformom

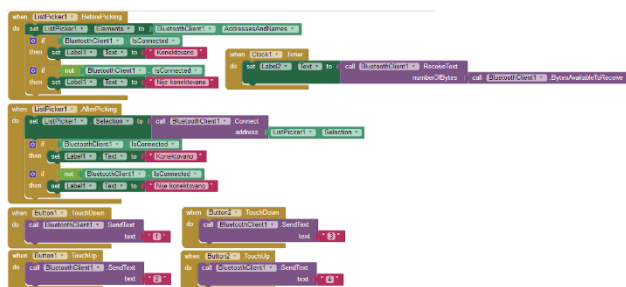
Aplikacija je krajnje jednostavnog dizajna i prilikom otvaranja aplikacije na ekranu se prikazuju dve strelice koje označavaju smer kretanja kip platforme. Povezivanje aplikacije sa upravljačkim delom se ostvaruje preko bežične komunikacije kratkog dometa – bluetooth-a. Pritiskom strelice na gore platforma se podiže, dok strelica na dole označava spuštanje platforme. Na slici 8 je prikazan proces izrade aplikacije u MIT App Inventor programu.



Slika 8. Proces izrade aplikacije

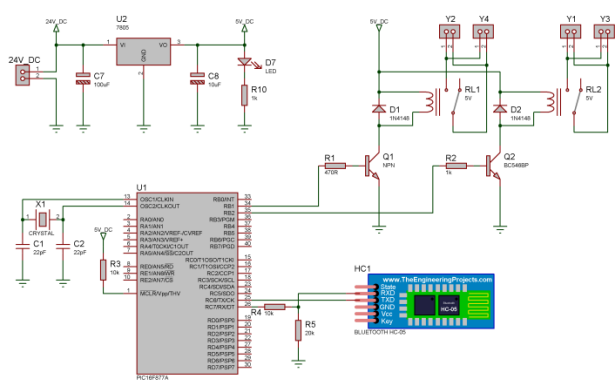
Program za kreiranje aplikacije je prikazan na slici 9. Prvi deo programa se odnosi na konekciju sa bluetooth modulom, tako što se u pretrazi za bluetooth klijentima odabere naziv ugrađenog modula koji komunicira sa mikrokontrolerom. Ukoliko je to učinjeno, saopštava se povezivanje porukom „konektovano“, u suprotnom na ekranu mobilnog telefona piše „nije konektovano“. Mikrokontroler i bluetooth modul komuniciraju putem serijske komunikacije.

Nakon izvršene konekcije sledi upravljanje radom kip platforme pomoću strelica gore – dole. Pritiskom strelice okrenute ka gore, bluetooth modul prosleđuje 1 na ulaz mikrokontrolera PIC16F877A i ona je aktivna sve do otpuštanja strelice, kada se prosleđuje 2. Pritiskom strelice ka dole na mikrokontroler se prosleđuje 3, dok otpuštanjem strelice prosleđuje se 4.



Slika 9. Prikaz programa za aplikaciju

Slika 10 prikazuje električnu šemu upravljanja kip platformom pomoću telefonske aplikacije. Na šemi je prikazan naponski regulator LM7805 koji spušta napon sa 24V na 5V, kojim se napaja mikrokontroler PIC16F877A kao i releji. Kada se na ulaz mikrokontrolera prosledi 1 sa bluetooth modula aktivan je pin RB2. Tada provodi tranzistor Q2 i dolazi do aktiviranja releja RL2 koji dovodi pod napajanjem elektromagnete Y1 i Y3 razvodnih ventila 1V1 i 3V1 zaduženih za izvlačenje klipnjača cilindara. Kada se na ulaz mikrokontrolera prosledi 2, tranzistor Q2 više nije pod napajanjem i ostatak kola postaje neaktivno što čini da je sistem u neutralnom položaju, tj. u stanju mirovanja. Ukoliko se na ulaz mikrokontrolera prosledi 3 tada je aktivan pin RB1 i provodi tranzistor Q1. Sada relej RL1 čini aktivnima elektromagnete Y2 i Y4 što znači da se klipnjače cilindara vraćaju u početni položaj. U momentu kada na ulaz mikrokontrolera stigne 4, pin RB1 postaje neaktivan i tranzistor Q1 više ne provodi i sistem se postavlja u neutralan položaj.



Slika 10. Električna šema upravljanja tel. aplikacijom

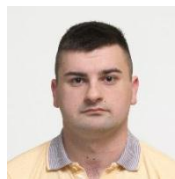
6. ZAKLJUČAK

U ovom radu imali smo priliku da vidimo razne tipove hidrauličnih kip platformi, čiji je zadatak da izvrše istovaranje tereta zrnaste kulture. Takođe, uočili smo da su principi rada opisanih kip platformi isti, ali razlike i te kako postoje, a ona koja je najpresudnija jeste – cena. Upravo ova veličina diktira stepen automatizovanosti nekog procesa i određuje tok razvoja tehnologije. Kako god bilo, uvek se biralo najefikasnije i najjednostavnije rešenje, ono je samim tim i najjeftinije. Može se primetiti da je energetska veoma efikasno rešenje iako je moglo drugačije da se isprojektuje kao npr: da protok bude veći, ali onda je potreban jači elektromotor i drugi kapaciteti pumpi ili da su se koristili teleskopski cilindri koji su skuplji i veće zapremine. Upotreba manjeg broja cilindara bi povećala njihov prečnik, a to bi značilo da je i cena cilindara znatno veća. Takođe, prikazan je i potrebni proračun. Pored svega ovoga, prikazano je i rešenje upravljanja putem mobilne aplikacije. To znači da su načinjeni prvi koraci ka modernizaciji ovakvih procesa koji su se godinama zasnivali na primeni relejne tehnike. Jasno je ovde da je potreba za poznavanjem elektronike neizostavna. Sve ovo upućuje na potrebu za neprekidnim radom i usavršavanjem jer takva je tehnika; ili koračaš krupnim korakom sa njom ili stojiš u mestu.

7. LITERATURA

- [1] http://www.vtssa.edu.rs/Teholoski%20procesi%20i%20zivotna%20sredina/TPZS_1.pdf [Dostupno 14.10.2017.]
- [2] <http://mssgracanica.com/skripte/AUTOMATIZACIJA%20PROIZVODNJE.pdf> [Dostupno 14.10.2017.]
- [3] <http://www.kompenzacionifond.gov.rs/file/strucni%20tekstovi/savremeni-pristup-upravljanju-skladistem-za-znaste-kulture.pdf> [Dostupno 14.10.2017.]

Kratka biografija:



Nikola Pucar rođen je u Jajcu, BiH 1992. god. Osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu upisuje 2011. godine, na smeru Mehatronika. Master akademske studije upisuje na istom fakultetu 2015. godine na smeru Mehatronika, robotika i automatizacija. Master rad odbranio je 2017. godine.

RAZVOJ DESKTOP APLIKACIJA KOJE MOGU DA SE IZVRŠAVAJU NA RAZLIČITIM OPERATIVNIM SISTEMIMA**DESKTOP APPLICATION DEVELOPMENT WITH POSSIBILITY OF EXECUTION ON DIFFERENT OPERATIVE SYSTEMS**

Nemanja Đekić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MEHATRONIKA

Kratak sadržaj – U ovom radu opisan je razvoj desktop aplikacije pomoću web tehnologija koja može da se izvršava na različitim operativnim sistemima. (Windows, Linux, MacOS).

Abstract – This papers describe desktop application development via web technologies with possibility of execution on different operative systems. (Windows, Linux, MacOS)

Ključne reči: Electron, Angular4, TypeScript, Firebase

1. UVOD

U eri interneta podaci se svakodnevno prikupljaju iz raznih izvora (npr. društvene mreže) i skladište se na cloud-e, odnosno određene servere. Prikupljeni podaci se koriste za razna ispitivanja, a sve u svrhu unapređenja kvaliteta određenih usluga i životnih standarda. Tokom ovog rada realizovana je desktop aplikacija koja ima mogućnost izvršavanja na različitim operativnim sistemima, a koja se bavi analizom i obradom prikupljenih podataka. Nauka o podacima [1] (engl. *Data Science*) je postala izuzetno popularna u današnje vreme. *Data Science* predstavlja interdisciplinarno polje o naučnim metodama, procesima i sistemima za izvlačenje znanja ili uvida iz podataka u različitim oblicima.

Svrha realizovane aplikacije je prikupljanje podataka sa cloud-a i njihova obrada, odnosno sortiranje i parsiranje podataka na osnovu zadatih parametara. Podatke je moguće sačuvati u .csv tabelu i na taj način prikazivati ih tabelarno, analizirati iste i iscrtavati potrebne grafike.

Navedena aplikacija ima za cilj praćenje određenih parametara koji mogu imati uticaj na kvalitet određenog proizvoda. Prikupljeni podaci daju korisniku uvid u ponašanje sistema (termički ili rashladni uređaj, prostorija u kojoj se odvija odgovarajući hemijski proces, ...)

2. TEORIJSKE OSNOVE

U ovom delu ukratko su opisane tehnologije i alati koji su korišćeni prilikom izrade ovog rada.

2.1. Electron

Electron [2] je biblioteka slobodno otvorenog koda koju je razvila *GitHub* kompanija za izradu desktop aplikacija pomoću web tehnologija za različite platforme koristeći *HTML*, *CSS* i *JavaScript* programske jezike.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Gordana Ostojić, vanr.prof.

Kombinacijom *Chromium*-a i *node.js* –a u jedinstvenu celinu *Electron* omogućava razvoj desktop aplikacija za Windows, Linux i MacOS platforme.

Neke od poznatih aplikacija koje su napravljene pomoću *Electron* biblioteke su: *GitHub Desktop*, *WordPress*, *Hyper Terminal*, *Atom*, *Visual Studio Code*...

Chromium [3] pretraživač je projekat slobodno otvorenog koda, koji za cilj ima razvoj sigurnijih, brzih i mnogo stabilnijih načina u svrhu poboljšanja korisničkog iskustva sa web aplikacijama.

Node.js [4] je izgrađena *JavaScript* biblioteka u *Chrome V8 JavaScript* sistemu. *Node.js* koristi neblokirajući ulazno/izlazni model, koji čini ovaj sistem efektivnijim i lakšim za korišćenje, a koji reaguje na neki događaj (npr. klik miša). Ekosistemski paket *npm* je najveći ekosistem biblioteka slobodno otvorenog koda.

2.2. HTML

HTML [5] (engl. *Hyper Text Markup Language*) je standardni meta jezik za razvoj Web stranica. U početku je bio ograničen što se tiče označavanja sadržaja i pružao je uglavnom elementarne stvari za označavanje i formatiranje teksta (paragrafi, naslovi, citati, itd.). Kako je internet rastao, rasla je i potreba za bogatijim sadržajem te je u tom smeru razvijan i *HTML* standard. Tada su standardu dodati elementi za opis tabela, slika, slojeva, napredno formatiranje teksta, itd.

Karakteristične su mu sledeće osobine:

- opisuje strukturu web stranica koristeći meta jezik
- jednostavno odvajanje elemenata kao što su naslovi, paragrafi, citati i slično
- elementi su predstavljeni pomoću etiketa (engl. *Tags*)
- ugrađeni elementi koji detaljnije opisuju sam dokument kao što su kratak opis dokumenta, ključne reči, podaci o autoru i slično
- internet pretraživači ne prikazuju *HTML* etikete, ali ih koriste za prikaz sadržaja

2.3. CSS

CSS [6] (engl. *Cascading Style Sheets*) je jezik formatiranja pomoću kog se definiše izgled elemenata web stranice. Prvobitno *HTML* je služio da definiše kompletan izgled, strukturu i sadržaj web stranice, ali je od verzije *HTML 4.0* uveden *CSS* koji definiše konkretan izgled, dok je *HTML* ostao u funkciji definisanja strukture i sadržaja.

CSS sintaksa se sastoji od opisa izgleda elemenata u dokumentu. Opis može da definiše izgled više elemenata i više opisa može da definiše jedan element. Na taj način se

opisi slažu jedan preko drugog da bi se definisao konačan izgled određenog elementa.

Svaki opis se sastoji od tri elementa:

- definicija ciljnih elemenata
- svojstva
- vrednosti

Nakon što se definišu ciljni elementi, tj. elementi na koje će se trenutni opis odnositi, nizom parova svojstvo-vrednost, definiše se izgled svakog ciljnog elementa. Sintaksa koja se pri tome definiše je sledećeg oblika:

```
ciljni-elementi {  
  → svojstvo: vrednost;  
  → svojstvo: vrednost;  
  → ...  
}
```

Slika 1. Primer definisanja CSS elementa

2.4 JavaScript

JavaScript [7] je jedan od najpopularnijih programskih jezika i koristi se za dodavanje interaktivnosti na web stranice, obradu podataka, kao i stvaranje različitih aplikacija (mobilne aplikacije, desktop aplikacije, igre i još mnogo toga). Službeno ime *JavaScript* –a je *ECMAScript* čije je standard donela organizacija *ECMA International*. Standard *ECMA-262* donešen je 1997., a *ISO* ga je prihvatio 1998. *JavaScript* kod izvršava web pretraživač koji pregleda stranicu na kojoj se taj kod nalazi. Ovakav oblik skriptovanja naziva se skriptovanje na klijentskoj strani (engl. *Client side scripting*). Važno: *JavaScript* i *Java* su dva različita jezika, čija je jedina poveznica slična sintaksa koju dele sa *C* i *C++* programskim jezicima.

JavaScript ima mogućnost da:

- ubaci dinamički sadržaj na web stranicu
- reaguje na događaje (npr. prelazak miša preko određenog objekta, klik miša, završetak učitavanja stranice, ...)
- čita i menja sadržaj HTML dokumenta
- izvrši validaciju podataka, pod uslovom da korisnik šalje podatke serveru
- otkrije koji internet pretraživač korisnik koristi (na ovaj način moguće je prikazati stranicu prilagođenu korisnikovom pretraživaču)
- kreira „kolačiće“ (engl. *Cookies*) (mehanizam zapisivanja i čitanja podataka na korisnikovom računaru)

2.4.1. TypeScript

TypeScript [8] je besplatan programski jezik slobodno otvorenog koda koji razvija i održava *Microsoft* kompanija. Strog je nadskup *JavaScript*-a, i dodaje jeziku opcionu statičku tipizaciju i objektnu orijentisanost. *TypeScript* se može koristiti za razvoj *JavaScript* aplikacija za izvršavanje na klijentu ili serveru (*Node.js*). Dizajniran je za razvoj velikih aplikacija i prevodi (engl. *Compile*) se u *JavaScript*-u. Kako je *TypeScript* nadskup *JavaScript*-a svi postojeći *JavaScript* programi su takođe validni i *TypeScript* programi. Podržava *header* datoteke koje mogu da sadrže tipske informacije za postojeće *JavaScript* biblioteke, omogućavajući time drugim programima da koriste objete definisane u *header* datotekama kao da su snažno tipični *TypeScript* objekti.

TypeScript je nastao zbog percipiranih nedostataka *JavaScript*-a za razvoj velikih aplikacija od strane *Microsoft*-a i njihovih klijenata. Izazovi sa kompleksnim *JavaScript* kodom doveli su do potražnje za prilagođenim alatima da bi se olakšao razvoj komponenti u jeziku. Dizajneri *TypeScript*-a tražili su rešenje koje neće izgubiti kompatibilnost sa standardom i njegovom višepatformskom podrškom.

Dodatne mogućnosti *TypeScript*-a su: tipski potpisi i provera tipova prilikom programskog prevodenja koda, dedukcija tipova podataka, klase, interfejsi, brojački tip, generisane promenljive, modularno programiranje, tuplovi (engl. *Tuples*).

Sintaksno, *TypeScript* je vrlo sličan *Jscript* .NET *Microsoft*-ovoj implementaciji *ECMA-262* jezičkog standarda, koji je dodao podršku za statičku standardizaciju i klasičnu objektnu orijentisanost kroz klase, nasleđivanje, interfejsa i imenske prostore.

2.5 Angular

Angular [9] je okvir (engl. *Framework*) koji služi za razvoj klijentskih aplikacija u *HTML*-u i/ili *JavaScript*-u ili jezicima poput *TypeScript*-a koji se prevode pomoću *JavaScript* programskog prevodioca.

Sam okvir se sastoji od nekoliko biblioteka, neke od njih su osnovne, a neke su izborne. Spajanje *HTML* šablona (engl. *Templates*) sa *Angular* oznakom, pisanjem klasa za komponente, koje upravljaju šablonima, dodavanjem aplikacione logike u usluge, i pakovanjem komponenti i usluga u pakete naastaju *Angular* aplikacije.

Angular aplikacije se pokreću pomoću korenskog modula, zatim *Angular* preuzima kontrolu i prikazuje sadržaj aplikacije u internet pretraživaču i odgovara na korisničke akcije u zavisnosti od uputstva koja su definisana unutar aplikacije.

2.6 Firebase

Firebase [10] je *Google*-ova platforma za razvoj aplikacija sledeće generacije. Ova tehnologija omogućava razvoj web aplikacija bez potrebe programiranja na serverskoj strani, tako da je razvoj aplikacija olakšan, jednostavan i znatno ubrzan.

Firebase je zadužen za skladištenje podataka, verifikaciju korisnika na server i implementaciju pristupnih pravila. Podržava *web*, *iOS*, *OS X* i *Android* klijentske aplikacije. Pomoću *Firebase*-a aplikacije mogu da koriste podatke i da ih menjaju (dodaju/brišu), a da ne moraju da vode računa o njihovom skladištenju i sinhronizovanošću sa drugim korisničkim aplikacijama u realnom vremenu. Zahvaljujući *Firebase* –u nije potrebno pisati kod za serversku aplikaciju ili razvijati kompleksni serverski okvir da bi aplikacija radila sa *Firebase*-om.

2.6.1. AngularFire2

Angularfire2 [11] je biblioteka koja je implementirana unutar *Angular* okvira, a koja služi za interakciju sa *Firebase*-om. Unutar ove biblioteke implementirani su paketi sa klasama koje sadrže metode za komunikaciju sa *Firebase*-om, autentifikaciju korisnika koji pristupaju *Firebase* serveru, kao i metode za manipulaciju podacima (čitavanje, pisanje, brisanje, menjanje...)

3. KONCEPT PROGRAMSKOG REŠENJA

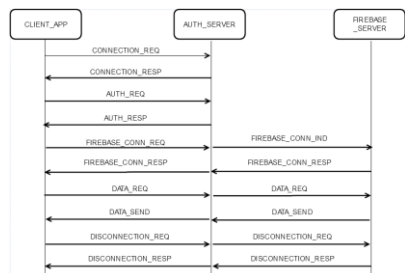
U ovom poglavlju predstavljen je koncept rešenja, odnosno ponašanje aplikacije i njena komunikacija sa *Firebase* servisom. Koncept rešenja predstavljen je pomoću *UML* dijagrama. (Slika 2) Zamišljeno je da se prilikom pokretanja aplikacije, na početnom ekranu, pojavi stranica za autentifikaciju. Korisnik može da izabere jednu od tri ponuđene opcije pomoću koje može da utvrdi da ima pravo pristupa bazi podataka. Ponuđene opcije su:

1. prijavljivanje pomoću *Facebook* naloga
2. prijavljivanje pomoću *Google+* naloga
3. prijavljivanje pomoću elektronske pošte i određene lozinke

Ukoliko korisnik nije registrovan ili nema ni jedan od navedenih naloga, korisnik može da se registruje pomoću elektronske pošte i da napravi novi nalog, pod uslovom da je unutar *Firebase*-a omogućena registracija novih korisnika. Aplikacija je modularna i slobodno otvorenog koda je, tako da vrlo lako može da se prilagodi korisničkim potrebama. Ukoliko korisnik želi da se registruje, na ekranu se otvara nova stranica u kojoj korisnik unosi potrebne podatke i pravi novi nalog. Kada je registracija uspešno izvršena, korisnik se vraća na početnu stranu i prijavljuje se na svoj nalog. Kada je korisnik prijavljen, prebačen je na stranicu na kojoj se podaci iz baze podataka prikazuju tabelarno.

3.1 UML dijagram

Pomoću *UML* dijagrama predstavljena je komunikacija između klijentske aplikacije i *Firebase*-a. Prvo se korisnička aplikacija šalje zahtev za povezivanje sa slojem na *Firebase* serveru koji je zadužen za autentifikaciju i čeka odgovor. Ukoliko je povezivanje uspešno, šalju se podaci za autentifikaciju. U zavisnosti od odgovora klijent može da se poveže sa *Firebase*-om, tj. ukoliko je autentifikacija uspešna, šalje se potvrđan odgovor klijentu i on može da pošalje zahtev da se poveže sa serverom. Kada se klijent povezuje sa serverom, čeka se takođe odgovor i ukoliko je povezivanje uspešno, klijent i server mogu da međusobno razmenjuju podatke.



Slika 2. Prikaz komunikacije između aplikacije i baze podataka pomoću *UML* dijagrama

4. REALIZACIJA REŠENJA

U ovom poglavlju detaljno je opisan proces razvoja aplikacije. Prilikom izrade ovog rada korišćen je *VisualStudioCode* uređivač datoteka sa integrisanim *cmd* terminalom. Aplikacija je razvijana na računaru sa *Windows 10 x64* operativnim sistemom.

Početak razvoja aplikacije podrazumevao je podešavanje razvojnog okruženja, instaliranje potrebnih biblioteka i

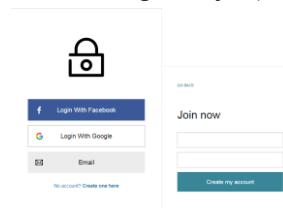
alata. Instaliran je *JavaScript* programski jezik sa pratećim alatima, zatim *node.js* biblioteka pomoću *Windows installer* -a. Ostale biblioteke su instalirane pomoću *cmd* -a i *npm* ekosistema, unoseći potrebne komande za instalaciju, kao npr. *npm install -g @angular/cli* (instaliranje *AngularCli* biblioteke). Datoteke u kojima je razvijana aplikacija kreirane su pomoću komande *ng new ime_aplikacije*. Kada su instalirane sve potrebne biblioteke, pomoću uređivača se otvaraju datoteke i počinje da se piše potrebni programski kod.

Većina metoda je implementirana iz *Angular* i *angularfire2* biblioteka i prikazane su ispod. (Tabela 1).

Tabela 1. Prikaz implementiranih metoda

Naziv funkcije	Opis funkcije
<code>firebase.auth.FacebookAuthProvider();</code>	Ova metoda omogućava prijavljivanje pomoću <i>Facebook</i> naloga i pristup bazi podataka
<code>firebase.auth.GoogleAuthProvider();</code>	Ova metoda omogućava prijavljivanje pomoću <i>Google</i> naloga i pristup bazi podataka
<code>firebase.auth().signInWithEmailAndPassword(email, password);</code>	Navedena metoda omogućava prijavljivanje registrovanih korisnika pomoću elektronske pošte i lozinke. Parametri koji se prosleđuju ovoj metodi su upravo lozinka i elektronska pošta.
<code>firebase.auth().createUserWithEmailAndPassword(email, password);</code>	Omogućava registraciju novih korisnika pomoću elektronske pošte i lozinke.
<code>firebase.auth().signOut();</code>	Odjavljivanje korisnika.
<code>firebase.storage().getDownloadURL();</code>	Čuvanje podataka sa <i>Firebase</i> -a
<code>firebase.storage().put();</code>	Postavlja podatke na <i>cloud</i> .
<code>firebase.storage().delete();</code>	Brisanje podataka na <i>cloud</i> -u.

Nakon odrađene logike, stavlja se fokus na grafičku korisničku spregu i kako bi trebalo aplikacija da izgleda. Pomoću *CSS*-a odrađen je dizajn, koji je prilično jednostavan i na slici ispod moguće je videti izgled početne strane i strane za registraciju. (Slika 3)



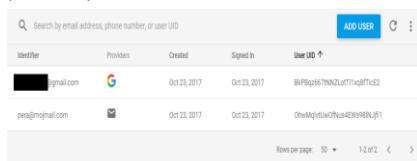
Slika 3. Izgled početne strane i strane za registraciju

Na samom kraju vrši se ispitivanje i verifikacija funkcionalnosti same aplikacije. Ispitivanje se vrši u četiri faze:

Prva faza - ispitivanje i verifikacija sistema za prijavljivanje na *Firebase* servis: Ovaj korak podrazumeva prijavljivanje pomoću sva tri navedena servisa (*Facebook*, *Google+*, elektronska pošta). Proverava se prijavljivanje pomoću jednog servisa, pa drugog i trećeg, da bi se utvrdilo da li je zaista moguće prijaviti se na *Firebase* servis. Ukoliko je prijavljivanje uspešno prelazi se na korak u kome se ispituje odjavljivanje sa servisa.

Druga faza - ispitivanje i verifikacija registracije novi korisnika: Ukoliko korisnik ne poseduje ni jedan od

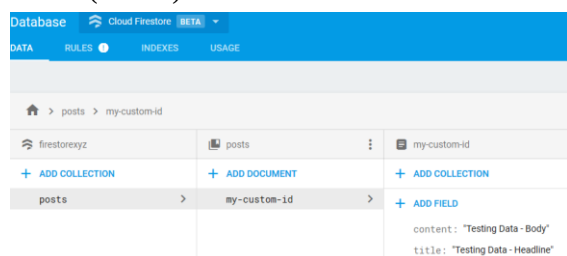
ponuđenih naloga za prijavljivanje, moguće je napraviti novi ukoliko je to odobreno od strane *Firebase* baze podataka. Ova provera je vrlo jednostavna. Korisnik unosi elektronsku poštu (korisničko ime) i lozinku. Kada klikne na opciju *Create my account* pojavljuje se tekst na klijentskoj strani sa odgovorom da li je registracija bila uspešna ili ne. Ukoliko je korisnik uspešno registrovan, potrebno je proveriti na samom *Firebase* servisu da li postoji korisnik pod registrovanim korisničkim nalogom. (Slika 4)



Search by email address, phone number, or user UID	Avatar	Provider	Created	Signed in	User UID
[redacted]	Google	Oct 23, 2017	Oct 23, 2017	BPtqjK6T7N2Lz7TtqgP7dE2	
penaj@gmail.com	Google	Oct 23, 2017	Oct 23, 2017	ChkAqjDwCfKua4EhKvNkUj1	

Slika 4. Verifikacija registrovanih korisnika na *Firebase* servisu

Treća faza - Ispitivanje i verifikacija prikaza i manipulacije podacima: U ovom koraku ispituje se manipulacija podacima, odnosno čitanje, pisanje i brisanje istih. Na slici ispod moguće je videti server stranu prikaza podataka. (Slika 5)



Database	Cloud Firestore	BETA
DATA	RULES	INDEXES
posts > my-custom-id		
firestorexyz	posts	my-custom-id
+ ADD COLLECTION	+ ADD DOCUMENT	+ ADD COLLECTION
posts	my-custom-id	+ ADD FIELD
		content: "Testing Data - Body"
		title: "Testing Data - Headline"

Slika 5. Prikaz podataka na server strani

Ukoliko se prikazivanje podataka, pisanje i brisanje odvija u realnom vremenu, odnosno promene se dešavaju istovremeno i na klijentskoj i na serverskoj strani, verifikacija je uspešna.

Četvrta faza - Ispitivanje .csv tabele : Ukoliko su svi prethodni koraci uspešno izvršeni, potrebno je samo proveriti da li su podaci sačuvani dobro u .csv tabelu. S obzirom da *Firebase* servis ne može direktno da čuva podatke u .csv tabelu, on ih čuva u *json* datoteci. Na osnovu identifikacionih oznaka elemenata i njihovog sadržaja, iščitavaju se podaci iz *json* datoteke, parsiraju se i sortiraju na osnovu navedenih oznaka i kao takvi čuvaju se u .csv tabeli. Ovaj postupak nije vidljiv krajnjem korisniku, ali mu je dostupna .csv tabela koja se kreira svaki put kada se korisnik odjavi i čuva se na lokalnom računaru u fascikli CSV, na mestu na kome je instalirana ova aplikacija.

5. ZAKLJUČAK

Pre nekoliko godina bilo je nezamislivo da se pomoću *web* tehnologija (*HTML*, *CSS*,...) razvijaju desktop aplikacije, koje ujedno mogu da rade i kao internet pretraživač aplikacije. Prednost ovakvih aplikacija je brz odziv, jednostavnost, ali i modularnost, jer je vrlo lako instalirati proširenja unutar istih. Prednost aplikacije razvijene tokom izrade ovog rada je takođe modularnost. Moguće je prilagoditi istu različitim bazama podataka na *Firebase* servisu, kao i menjati algoritam za parsiranje i sortiranje podataka, odnosno parametre po kojima se podaci sortiraju.

S obzirom da je modularnost jako bitna, jedan od mogućih pravaca daljeg razvoja je jednostavnije povezivanje sa određenom bazom podataka. Bilo bi poželjno kada bi *Firebase* servis povezivanje na bazu sveo na svega 2 – 3 jednostavna parametra, koje korisnik može da unese na početnom ekranu i da se na osnovu njih povezuje na željenu *Firebase* bazu podataka. Pored toga, jedan od predloga je i pravljenje nove stranice u kojoj će korisnik moći da unese parametre na osnovu kojih će se podaci parsirati i sortirati. U ovom slučaju ima ponuđeno sortiranje po kolonama (od najveće ka najmanjoj vrednosti i obrnuto). Finalni predlog je korišćenje *Python* programskog jezika umesto *JavaScript*-a za manipulaciju podacima. *Python* ima integrisane razne biblioteke za obradu podataka poput *NumPy*, *Pandas*, *Matplotlib*,... Bilo bi poželjno kada bi se integrisao alat napravljen u *Python* programskom jeziku u ovu aplikaciju, koji bi služio samo za analizu podataka, tako da korisnik ne bi morao da čuva podatke posebno u neku .csv tabelu, već bi direktno mogao da ih obrađuje unutar aplikacije i da ih čuva na primer u *PDF* formatu.

6. LITERATURA

- [1] Preuzeto sa: <https://datascience.berkeley.edu/about/what-is-data-science>, 25.10.2017.
- [2] Preuzeto sa: <https://electron.atom.io/docs/tutorial/about>, 25.10.2017.
- [3] Preuzeto sa: <https://www.chromium.org>, 25.10.2017.
- [4] Preuzeto sa: <https://nodejs.org/en/about>, 25.10.2017.
- [5] Preuzeto sa: https://www.w3schools.com/html/html_intro.asp, 25.10.2017.
- [6] Preuzeto sa: https://www.w3schools.com/css/css_intro.asp, 25.10.2017.
- [7] Preuzeto sa: https://www.w3schools.com/js/js_intro.asp, 25.10.2017.
- [8] Preuzeto sa: <https://www.typescriptlang.org>, 26.10.2017.
- [9] Preuzeto sa: <https://angular.io/guide/quickstart>, 26.10.2017.
- [10] Preuzeto sa: <https://firebase.google.com/docs>, 26.10.2017.
- [11] Preuzeto sa: <https://github.com/angular/angularfire2/tree/master/docs>, 26.10.2017.

Kratka biografija:

Nemanja Đekić rođen je u Doboju 1993. godine. Master rad na Fakultetu tehničkih nauka iz oblasti Mehatronika, robotika i automatizacija odbranio je 2017 godine.

APLIKACIJA ZA UNOS I KNJIŽENJE BANKARSKIH SPECIFIKACIJA

APPLICATION FOR ENTERING AND POSTING BANKING SPECIFICATIONS

Jelena Raca, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MEHATRONIKA

Kratak sadržaj –U ovom radu je opisan jedan od karakterističnih automatizovanih procesa današnjih bankarskih sistema. Obradeni su delovi aplikacije, koji uključuju unos i knjiženje bankarskih specifikacija, kreiranje i štampu izveštaja, prebacivanje u Excel unešenih specifikacija, kao i učitavanje podataka iz Excela i njihovo ubacivanje u bazu podataka. Razvojna okruženja korišćena za razvoj aplikacije za unos i knjiženje bankarskih specifikacija, jesu Visual Studio 2012 i SQL Management Studio 2014.

Abstract – This paper describes one of the characteristic automated processes of today's banking systems. Application modules are processed, which include the entry and posting of bank specifications, creation and printing of reports, switching into Excel entered specifications, as well as loading data from Excel and uploading them to the database. The development environments used to develop the application for entering and posting banking specifications are Visual Studio 2012 and SQL Management Studio 2014.

1. UVOD

Informacione tehnologije imaju fundamentalan uticaj na svaki aspekt ljudskog života i savremenog poslovanja, da je praktično nezamislivo funkcionisanje bez njih[10]. U poslovnom sistemu, informacioni sistem obezbeđuje prikupljanje podataka iz različitih izvora poslovanja, obradu, čuvanje i ažuriranje podataka, pravljenje izveštaja, klasifikovanje podataka, kao i distribuisanje podataka organima koji donose odluke. Sektor na koji je razvoj informacionih tehnologija najradikalnije uticao jeste bankarski sistem.

Bankarska industrija je industrija koja se u velikoj meri oslanjala na kompjuterizovane zapise i mogućnosti pristupa ključnim informacijama brzo i jednostavno.

Automatizacija procesa rada je veoma značajna u današnjim bankarskim sistemima, ne samo za zaposlene u banci, već i za klijente, jer povećava produktivnost i smanjuje utrošeno vreme. U ovom radu je opisan jedan od karakterističnih automatizovanih procesa današnjih bankarskih sistema, koji uključuju unos i knjiženje bankarskih specifikacija, kreiranje i štampu izveštaja, prebacivanje u Excel unešenih specifikacija, kao i učitavanje podataka iz Excela i njihovo ubacivanje u bazu podataka.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Stankovski, red.prof.

2. KORIŠĆENI ALATI I PROGRAMSKI JEZICI

Razvojna okruženja korišćena za razvoj aplikacije za unos i knjiženje bankarskih specifikacija, jesu Visual Studio 2012 i SQL Management Studio 2014.

2.1 Visual Studio 2012, C#

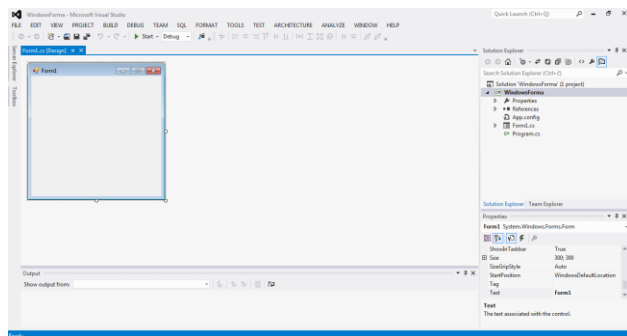
Microsoft Visual Studio predstavlja skup alata za kreiranje softvera, od faze planiranja kroz dizajn interfejsa (UI), kodiranje, testiranje, debugovanje, analizu kvaliteta koda i performansi, plasiranje kupcima i prikupljanje telemetrije prilikom korišćenja[13]. Ovi alati su dizajnirani da rade zajedno što besprekornije i svi su na raspolaganju zahvaljujući Visual Studio IDE (Integrated Development Environment) integrisanom razvojnom okruženju. Visual Studio može da se koristi za kreiranje različitih vrsta aplikacija, od jednostavnijih aplikacija za kupovinu i igrice za mobilne telefone, do velikih, kompleksnih sistema, koji upravljaju preduzećima i centrima za podatke. Mogu da se kreiraju aplikacije i igrice koje se pokreću ne samo na Windows operativnom sistemu, već i na Android OS i iOS-u, web stranice i web usluge bazirane na ASP.NET, JQuery, AngularJS i drugim popularnim platformama, grafički zahtevne aplikacije za različite Windows uređaje, uključujući Xbox koristeći DirectX. Visual Studio podrazumevano pruža podršku za C#, C i C++, zatim JavaScript, F# i Visual Basic. Visual Studio radi i dobro se integriše sa aplikacijama nezavisnih proizvođača, kao što je npr. Unity.

C# je moderan, objektno-orijentisan programski jezik opšte namene, koji je razvijen od strane Microsoft-a, a koji je odobrila ECMA (European Computer Manufacturers Association) i međunarodna organizacija za standarde (International Standards Organization-ISO). C# je programski jezik koji sadrži osobine kao što su jednostavnost, doslednost, kompatibilnost i fleksibilnost. C# je razvio Anders Hajlsberg i njegov tim tokom razvoja .NET platforme. C# je dizajniran za CLI (Common Language Infrastructure), koji se sastoji od izvršnog koda, i okruženja u vremenu izvršavanja (runtime environment-RTE), koje omogućava upotrebu različitih programskih jezika visokog nivoa na različitim kompjuterskim platformama sa različitom arhitekturom[12].

2.2 Windows Forms aplikacija

Windows forme predstavljaju grafički programski interfejs aplikativnog interfejsa (API), koji je uključen kao deo Microsoft-ove .NET platforme. Jednostavnije rečeno, WinForms je biblioteka za kreiranje GUI aplikacija.

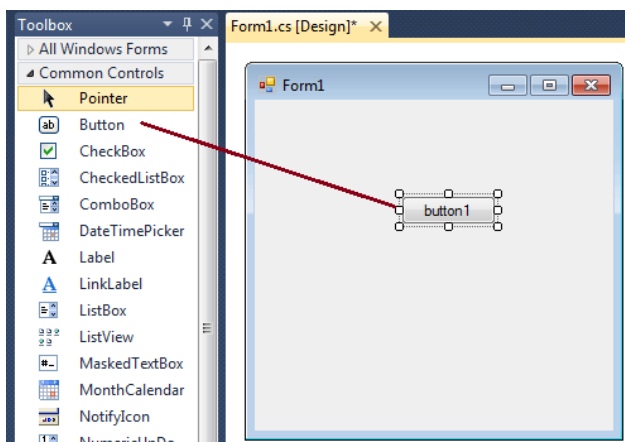
Na slici 2.1 prikazan je izgled kreirane Windows forme u Visual Studiju pod nazivom, u ovom slučaju, WindowsForma. C# programeri su u velikoj meri iskoristili forme za izgradnju korisničkih interfejsa. Svaki put kada se kreira nova Windows aplikacija, Visual Studio prikazuje podrazumevanu praznu formu (Slika 2.1).



Slika 2.1 Prikaz kreirane Windows forme

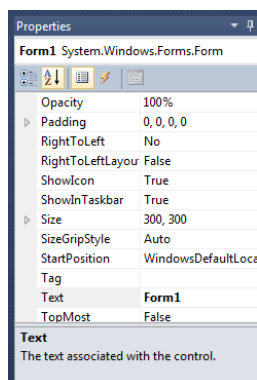
Na kreiranu praznu formu mogu da se prevuku kontrole i prilagodi njihova veličina i položaj (Slika 2.2).

Na vrhu forme nalazi se podrazumevani naziv forme Form1, koje se može promeniti u zavisnosti od potrebe i namene forme. Pored naziva forme, nalazi se i kontrolni deo, koji sadrži opcije za minimiziranje, maksimiziranje i zatvaranje forme.

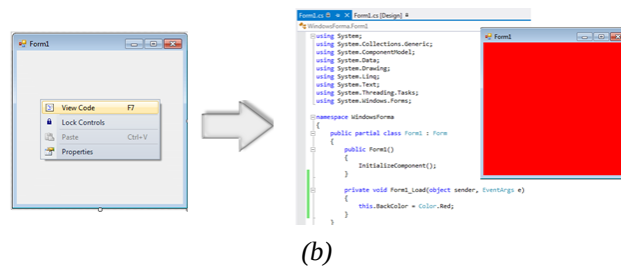


Slika 2.2 Prikaz kontrola unutar alatki (Toolbox)

Podešavanje svojstava forme može da se izvrši preko prozora Properties (2.3(a)), a takođe i kroz kod (2.3(b)).



(a)



(b)

Slika 2.3 Podešavanje svojstava forme

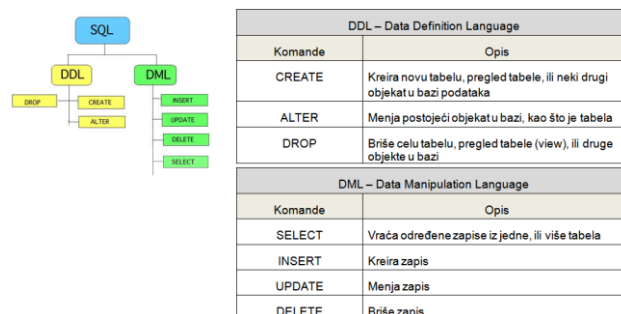
2.3 SQL, SQL Management Studio

SQL (engl. *Structured Query Language*) je relacioni upitni jezik za rad sa bazama podataka. To podrazumeva kreiranje baze podataka, čuvanje, manipulaciju i preuzimanje podataka. SQL je ANSI (*American National Standards Institute*) i ISO standardni jezik. Svi sistemi upravljanja relacionim bazama podataka (RDBMS-*Relational Database Management Systems*), kao što su MySQL, MS Access, Oracle, Sybase, Informix, Postgres i SQL Server, koriste SQL kao standardni jezik baze podataka.

Ocem relacionih baza podataka smatra se dr Edgar F. "Ted" Codd iz IBM-a, koji je 1970. godine opisao relacioni model baze podataka. SQL je inicijalno razvijen u IBM-ovoj istraživačkoj laboratoriji, a njegovi tvorci su Donald D. Chamberlin i Raymond F. Boyce početkom 1970-ih.

Standardne SQL komande za interakciju sa relacionim bazama podataka su CREATE, SELECT, INSERT, UPDATE, DELETE i DROP.

Jezici baze podataka su podeljeni u dve glavne kategorije: jezik za definisanje podataka (*DDL-Data Definition Language*) i jezik za manipulisanje podacima (*DML-Data Manipulation Language*). Oba jezika (2.3) koriste iskaze i komande SQL-a, koje se koriste za preuzimanje i manipulisanje podacima u relacionoj bazi podataka [7].



Slika 2.3 Prikaz i opis DLL i DML komandi

SQL Server predstavlja Microsoft-ov sistem za upravljanje relacionim bazama podataka (RDBMS)[11]. To je sveobuhvatna baza podataka primarno kreirana da se takmiči sa konkurentima, kao što su Oracle DB i MySQL. SQL Server, kao i svi veliki RDBMS sistemi, podržava ANSI SQL, standardni SQL jezik. Međutim, SQL Server sadrži i T-SQL, sopstvenu implementaciju SQL-a. Primarni jezik za upite, Transact SQL (T-SQL), znači da osim osnovnih i klasičnih (SELECT tipa) SQL upita dozvoljava i složenije stvari poput menjanja programskog toka (IF naredba) i slično.

SQL Server Management Studio (SSMS), ranije poznat kao Enterprise Manager, je glavni alat za pristup (interfejs) SQL Servera i podržava 32-bitna i 64-bitna okruženja. SQL Server Management Studio je aplikacija koja se koristi za podešavanje, upravljanje i nadgledanje svih komponenti Microsoft SQL Server-a.

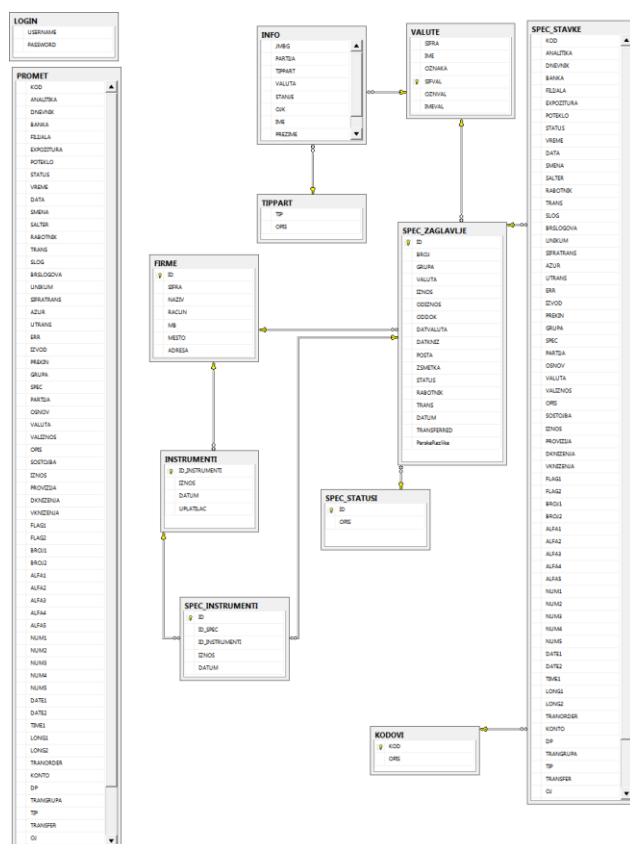
3. APLIKACIJA ZA UNOS I KNJIŽENJE BANKARSKIH SPECIFIKACIJA

3.1 Uvod, šema banarske specifikacije

Bankarski sistem je veoma složen sistem, koji sadrži mnogo različitih aplikacija, koje se svakodnevno koriste i umnogome olakšavaju rad. Prednosti koje proizilaze iz kompjuterizacije, odnosno napretka IT tehnologije su usmerene i prema kupcu i prema banci i prema zaposlenom. Neke od aplikacija u svakodnevnoj upotrebi jesu Devizna aplikacija, Trezor, Platni promet, CMS (Card Management System), E-Bank, Šalterska aplikacija, itd.

Šalterska aplikacija se najviše koristi od svih navedenih aplikacija i upravo neki od njenih delova i podsistema će biti obrađeni u ovom radu. Ova aplikacija se koristi u svakodnevnom radu i kroz nju mogu da se vrše uplate, isplate, otvaranje i zatvaranje računa, izveštavanje, itd.

Šema baze podataka predstavlja strukturu logičkog prikaza cele baze podataka. Na slici 3.1 se vidi šema (dijagram) baze podataka bankarske specifikacije.

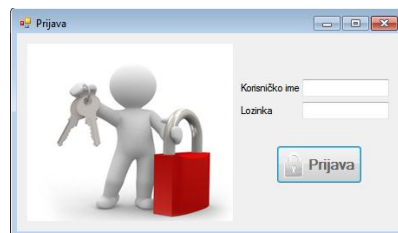


Slika 3.1 Šema bankarske specifikacije

3.2 Rad sa specifikacijama

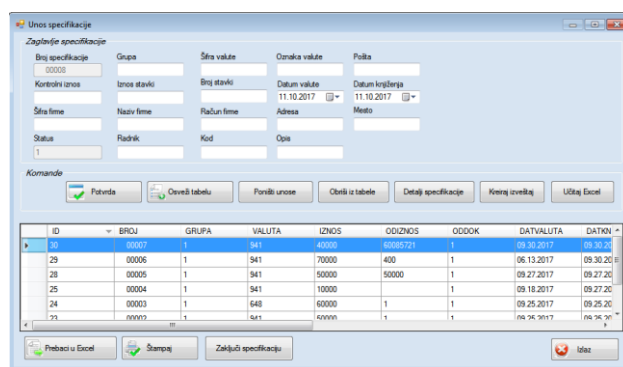
Aplikacija za rad sa specifikacijama sastoji se između ostalog od forme za prijavu, prikaza same specifikacije i detalja svake specifikacije.

Specifikacije su vrsta grupnih naloga, koji služe za rasknjižavanje plata zaposlenih, masovne nabavke i slično. Zamisao rešenja koje će biti prikazano, bila je da se olakša, a samim tim i ubrza rad zaposlenih u ovom bankarskom sektoru, kao i da se zameni nekadašnji način rada i poboljša funkcionalnost. Forma za prijavu, koja se sastoji od korisničkog imena i lozinke, prikazana je na slici 3.2.



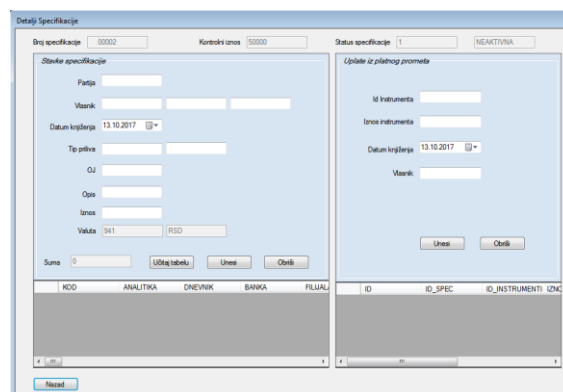
Slika 3.2 Forma za prijavu

Nakon uspešne prijave, otvara se nova forma, a to je forma za unos bankarskih specifikacija, koja se sastoji od zaglavlja specifikacije, komandi (operacija) za izvršavanje, kao i prikaza tabele unešenih specifikacija (3.3).



Slika 3.3 Prikaz forme za unos specifikacija

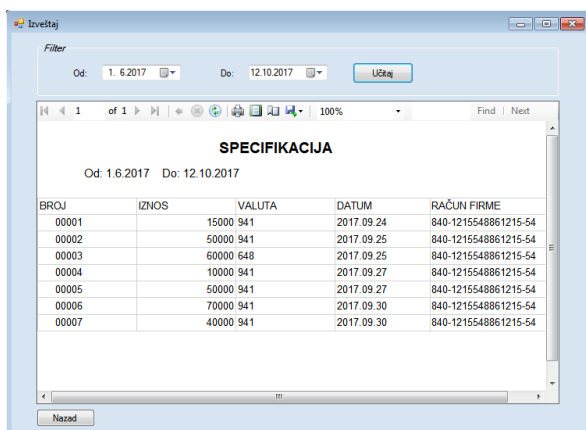
Moguće je pristupiti detaljima svake unešene specifikacije klikom na dugme Detalji specifikacije. Detalji se odnose na onu specifikaciju, koja je u trenutku klika na dugme, bila obeležena u tabeli. Takođe, moguće je i dvostrukim klikom na određeni red u tabeli, otvoriti detalje specifikacije. Izgled forme prikazan je na slici 3.4.



Slika 3.4 Prikaz detalja specifikacije

Na formi Detalji specifikacije se nalaze isplate koje radnik unosi (tzv. stavke specifikacije) i spoljašnja uplata, koja je prethodno proknjižena kroz platni promet. Uplate iz platnog prometa kao i stavke specifikacije mogu se upisivati i brisati sve do trenutka zaključenja specifikacije. Nakon toga, moguć je samo uvid. Nakon unosa zaglavlja specifikacije i detalja i provere da li se svi iznosi i sume slažu, specifikacija može da bude zaključena. To se vrši sa osnovne forme za unos specifikacija (slika 3.3), odabirom određene specifikacije iz liste specifikacija i klikom na dugme Zaključi specifikaciju. Suma iznosa stavki specifikacije, mora da bude jednaka kontrolnom iznosu specifikacije, kako bi specifikacija mogla da bude uspešno zaključena.

Unete specifikacije, koje se nalaze u tabeli unutar forme za unos, moguće je prebaciti u Excel datoteku. Prilikom čuvanja, moguće je odabrati željeni tip datoteke, da li datoteku sa ekstenzijom .xlsx (Excel Files 2007), ili .xls (Excel Files 2003), ili čuvanje bilo koje vrste datoteke. Kreiranje izveštaja je omogućeno klikom na dugme Kreiraj izveštaj. Izveštaj je kreiran kao RDLC Report (Report Definition Language Client-side), podrazumevani alat u sklopu Visual Studia. Na formi se vidi deo za filtriranje specifikacija u zavisnosti od datuma knjiženja (3.5).



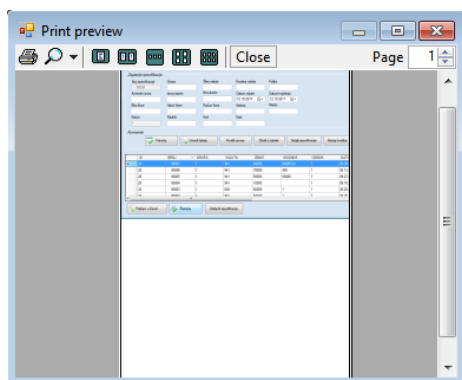
Filter: Od: 1. 6. 2017 Do: 12.10.2017 [Učitaj]

SPECIFIKACIJA
Od: 1.6.2017 Do: 12.10.2017

BROJ	IZNOS	VALUTA	DATUM	RAČUN FIRME
00001	15000	941	2017.09.24	840-1215548861215-54
00002	50000	941	2017.09.25	840-1215548861215-54
00003	60000	648	2017.09.25	840-1215548861215-54
00004	10000	941	2017.09.27	840-1215548861215-54
00005	50000	941	2017.09.27	840-1215548861215-54
00006	70000	941	2017.09.30	840-1215548861215-54
00007	40000	941	2017.09.30	840-1215548861215-54

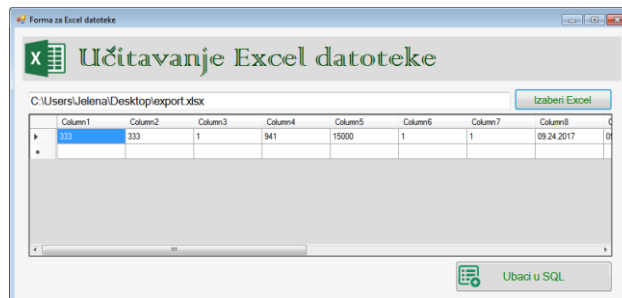
Slika 3.5 Izveštaj specifikacija

Moguće je kreirati izveštaj prebaciti u Word dokument, Excel i PDF. Izveštaj može i da se odštampa, zatim može da se podesi veličina papira, orijentacija, kao i margine. Omogućeno je i štampanje forme za unos specifikacije, tačnije svega onoga što se vidi u trenutku kada se klikne na dugme Štampaj.



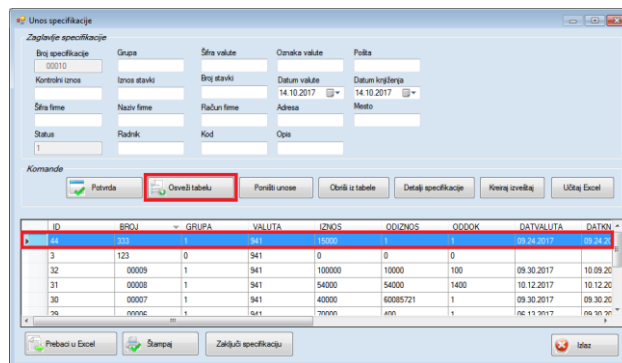
Slika 3.6 Prikaz dokumenta za štampanje

Na formi za unos specifikacija, nalazi se još jedna komanda, a to je komanda za otvaranje nove forme, u kojoj je omogućeno učitavanje Excel datoteka sa računara (3.7), kao i ubacivanje izabrane Excel datoteke u bazu podataka.



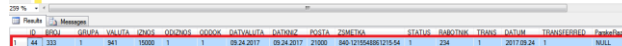
Slika 3.7 Prikaz učitane Excel-a unutar forme

Provera da li su vrednosti iz datoteke uspešno učitane u bazu, tj. u tabelu dbo.[SPEC_ZAGLAVLJE], moguća je klikom na dugme Osveži tabelu u formi za unos specifikacija (3.8(a)), a takođe i u bazi podataka, preko SELECT komande, pregledom svih vrednosti u tabeli (3.8 (b)).



(a)

```
SELECT * FROM [virtuelnabanka].[dbo].[SPEC_ZAGLAVLJE]
```



(b)

Slika 3.8 Provera učitanih vrednosti iz Excel-a

(a) provera u formi za unos

(b) provera u bazi podataka

4. SQL SERVER PROFILER

SQL Server Profiler je alat, uz pomoć koga će mogu da se prate promene koje se dešavaju nad bazom podataka. SQL Server Profiler sakuplja sve informacije koje se dešavaju unutar instance SQL servera i time omogućava nadgledanje svih aktivnosti, od redovne korisničke aktivnosti/transakcija, do blokade/zaključavanja i sistemskih greški[14]. Može da se izvrši proaktivno održavanje instance SQL servera, pomoću koga bi se identifikovali bilo kakvi loše obavljeni upiti, kako bi se kasnije mogli analizirati i popraviti.

5. ZAKLJUČAK

Ključnu stavku prilikom stvaranja poslovne politike banaka predstavlja dobro organizovana koncepcija i struktura informacionog sistema[15]. Savremene informacione tehnologije uticale su na povećanje brzine i tačnosti transakcija, kao i na smanjenje vremena potrebnog za procesuiranje transakcija, povećanje produktivnosti zaposlenih, a takođe i manji utrošak vremena klijenata[6].

Upravo iz navedenih razloga, važnost aplikacije za unos i knjiženje bankarskih specifikacija je od velikog značaja za uspešno poslovanje banke. Zahvaljujući toj aplikaciji, moguće je obavljati neke od najvažnijih funkcija bankarskog sistema, koje se koriste u svakodnevnom radu, kao što su uplate, isplate, izvršenja bilo kakvih priliva na račun, zatim izveštavanje, kao i učitavanje i eksportovanje potrebnih dokumenata. Takođe, aplikacija za unos i obradu bankarskih specifikacija omogućava zaposlenima da u bilo kom trenutku imaju uvid u prethodno unete specifikacije, kao i u njihove detalje.

S obzirom da se proizvodi i usluge koje banka nudi svojim klijentima, svakim danom sve više proširuju i menjaju, te promene moraju da budu ispraćene u njihovom postojećem informacionom sistemu, što je u ovoj aplikaciji omogućeno, zahvaljujući korišćenim alatima.

Postoje i određeni nedostaci kada se govori o IT tehnologijama u poslovnom svetu, a to su npr. zloupotreba računara za lične interese u službenim radnim satima, zatim dugoročno korišćenje računara, koje može da ima značajne posledice na zdravlje, a tu je i neizostavno pitanje sigurnosti informacija, pogotovo u bankarskim i finansijskim sistemima.

Industrija informacionih tehnologija je jedno intenzivno konkurentno okruženje, koje se odlikuje brzom promenom tehnologija, razvojem industrijskih standarda, čestim uvođenjem novih proizvoda i smanjenjem cena i troškova. Potrebna su velika ulaganja u informacione tehnologije i blagovremeno reagovanje i prilagođavanje na promene zahteva tržišta, kako bi se uspešno održala konkurencija u ovom okruženju.

6. LITERATURA

- [1] Alan Beaulieu, "Learning SQL", O'Reilly Media, Inc., Sebastopol, 2009.
- [2] Eric Brown – Windows Forms Programming with C#, Manning, 2006.
- [3] Ross Mistry and Stacia Misner, „Introducing Microsoft SQL Server 2014“, Microsoft Corporation, Washington, 2014.
- [4] Mladen Čudanov, „Organizacija i strateška primena IKT-a“, Zadužbina Andrejević, Beograd, 2011.

- [5] <https://www.tutorialspoint.com/sql/sql-overview.htm> [pristupljeno 02.09.2017.]
- [6] <http://www.ubs-asb.com/Portals/0/Casopis/2014/5/UBS-Bankarstvo-5-2014-DusicaSanader.pdf> [pristupljeno 07.10.2017.]
- [7] <http://www.differencebetween.info/difference-between-ddl-and-dml-in-tabular-form> [pristupljeno 02.09.2017.]
- [8] [https://en.wikipedia.org/wiki/C_Sharp_\(programming_language\)](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)) [pristupljeno 09.09.2017.]
- [9] https://www.tutorialspoint.com/ms_sql_server/ms_sql_server_tutorial.pdf [pristupljeno 02.09.2017.]
- [10] <https://www.scribd.com/doc/37928469/SEMINARSKI-Uticaj-razvoja-informacionih-tehnologija-na-poboljsanje-realizacije-poslovnih-aktivnosti> [pristupljeno 07.10.2017.]
- [11] https://en.wikipedia.org/wiki/Microsoft_SQL_Server [pristupljeno 08.09.2017.]
- [12] <https://www.tutorialspoint.com/csharp/index.htm> [pristupljeno 23.09.2017.]
- [13] <https://msdn.microsoft.com/en-us/library/dn762121.aspx> [pristupljeno 16.09.2017.]
- [14] <https://www.mssqltips.com/sqlservertutorial/3500/introduction-to-sql-server-profiler/> [pristupljeno 03.09.2017.]
- [15] <https://raf.edu.rs/citaliste/racunarske-mreze/3929-racunari-u-oblasti-bankarstva> [pristupljeno 10.10.2017.]

Kratka biografija:



Jelena Raca rođena je u Sarajevu 1991. god. Osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu upisuje 2010. godine, na smeru Mehatronika, usmerenje Mehatronika, robotika i automatizacija. Master akademske studije upisuje na istom fakultetu 2016. godine na smeru Mehatronika, robotika i automatizacija.

APLIKACIJA ZA NAPLATU PARKIRANJA NA OTVORENIM PARKING PROSTORIMA**APPLICATION FOR CHARGING PARKING FEE AT OUTDOOR PARKING SPACES**

Atila Smieško, Fakultet tehničkih nauka, Novi Sad

Oblast – MEHATRONIKA

Kratak sadržaj – U ovom radu je opisan proces evidentiranja i naplate parkiranja vozila. Obradeni su uređaji koji uključuju prepoznavanje vozila koja se parkiraju, prepoznavanje kovanice i novčanica kojima se plaća parkiranja, evidentiranje trajanja parkiranja i davanje priznanice o parkiranju. Uređaji koji su korišćeni za gore navedene funkcije su: NV10, RM5HD, TG1260H.

Ključne reči: Aplikacija, parking, programiranje, protokol.

Abstract – This paper describes the process of recording and charging a vehicle for parking fee. Used devices include that recognize coins and banknotes that will be paid for parking, recording of the duration of parking and issuance of parking receipts. The devices used for the above mentioned functions are: NV10, RM5HD, TG1260H.

Key words: Application, parking, programming, protocol.

1. UVOD

Postoji više načina evidentiranja i naplate na javnim parkiralištima. Svima je zajedničko da se mora tačno odrediti vozilo koje je parkirano, trajanje parkiranja, cena parkiranja i dokaz o naplati usluge parkiranja. U ovom radu je opisan način rada i karakteristike aplikacije za naplatu parkiranja na otvorenim parking prostorima.

Samouslužni automatizovani sistem za naplatu parkiranja treba da omogući pouzdan i jednostavan sistem naplate parkiranja.

Mobilnost automata za naplatu parkiranja treba da omogući solarni panel preko koga sistem treba da se napaja električnom energijom a autonomiju sistema treba da obezbedi baterija. Sistem za naplatu parkiranja treba da bude pouzdan i da bude na raspolaganju za samostalno korišćenje 24 sata dnevno, 7 dana u nedelji u svim vremenskim uslovima.

Automat za naplatu parkiranja treba da omogući korisniku unos registracije vozila i plaćanje parkinga putem gotovine (novčanice i kovanice).

Automat ne treba da radi vraćanje kusura, već se, ukoliko korisnik uplati veći iznos nego što je potrebno, preostali iznos vezuje za registraciju vozila.

Korisnik prilikom sledećeg korišćenja parking prostora može iskoristiti taj iznos ili ga podići na ovlašćenom mestu. Nakon ubacivanja novca u automat, potrebno je da se odabere broj sati za parkiranje.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Stankovski, red.prof.

Korisnik treba da ima mogućnost da odabere broj sati za parkiranje, pri čemu se maksimalan broj sati određuje na osnovu ubačenog novca i cene jednog sata parkiranja.

Automat pruža mogućnost da se uradi izdavanja štampane potvrde o plaćenom parkiranju, ili da se potvrdi o plaćenom parkiranju pošalje na e-mail adresu koju korisnik unese na automatu. Unetu e-mail adresu sistem treba da zabeleži i veže za registraciju vozila.

2. KORIŠĆEN ALAT

Razvoj aplikacije je urađen pomoću softverskog okruženja Visual Studio 2008. Aplikacija je napisana pomoću C# programskog jezika.[3]

2.1. Visual Studio 2008 C#

Microsoft Visual Studio predstavlja alat za kreiranje softvera, od faze planiranja kroz dizajn interfejsa (UI), kodiranje, testiranje, debugovanje, analizu kvaliteta koda i performansi, plasiranje kupcima i prikupljanje telemetrije prilikom korišćenja. Ovi alati su dizajnirani da rade zajedno što besprekornije i svi su na raspolaganju zahvaljujući Visual Studio IDE (Integrated Development Environment) integrisanom razvojnom okruženju. Visual Studio može da se koristi za kreiranje različitih vrsta aplikacija, od jednostavnijih aplikacija za kupovinu i igrice za mobilne telefone, do velikih, kompleksnih sistema, koji upravljaju preduzećima i centrima za podatke.

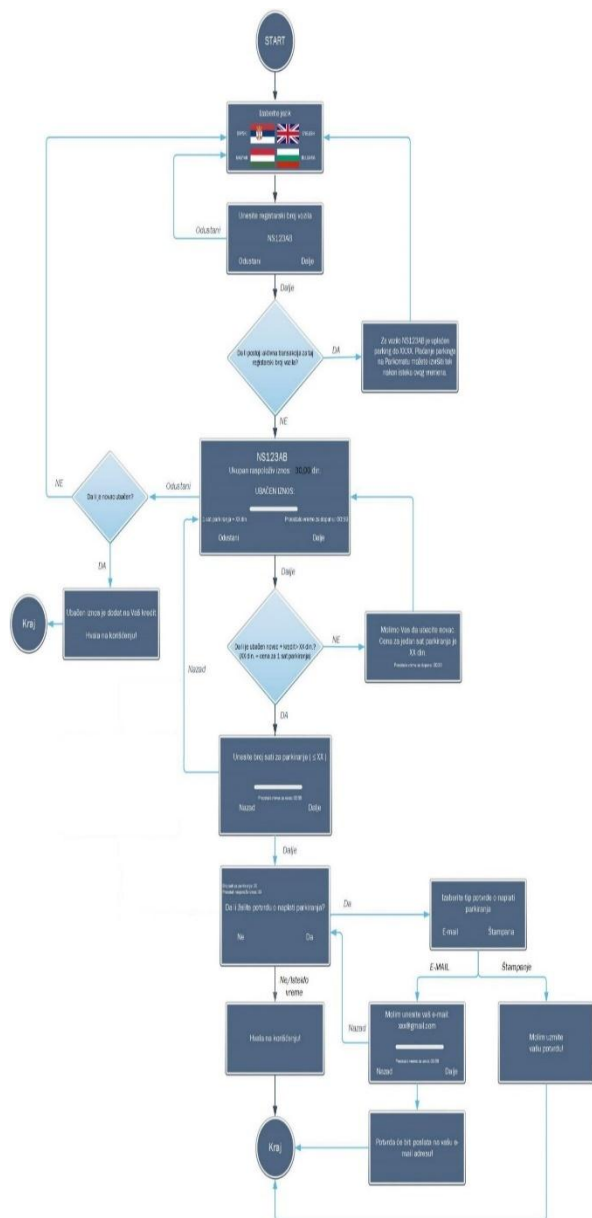
C# je moderan, objektno-orijentisan programski jezik opšte namene, koji je razvijen od strane Microsoft-a, a koji je odobrila ECMA (**European Computer Manufacturers Association**) i međunarodna organizacija za standarde (**International Standards Organization-ISO**). C# je programski jezik koji sadrži osobine kao što su jednostavnost, doslednost, kompatibilnost i fleksibilnost. C# je razvio Anders Hajlsberg i njegov tim tokom razvoja .NET platforme. C# je dizajniran za CLI (**Common Language Infrastructure**), koji se sastoji od izvršnog koda, i okruženja u vremenu izvršavanja (**runtime environment-RTE**), koje omogućava upotrebu različitih programskih jezika visokog nivoa na različitim kompjuterskim platformama sa različitom arhitekturom.

2.2. Implementacija komunikacije sa uređajima

Algoritam je ključni pojam u računarskoj obradi podataka jer je računarski program izvestan algoritam koji računaru objašnjava koje korake (naredbe) i kojim redosledom treba da obavlja. Tako se algoritmom može smatrati bilo koji niz instrukcija. Pošto je algoritam niz preciznih koraka (naredbi), redosled izračunavanja je

uvek kritičan za funkcionisanje algoritma. Pretpostavlja se da su naredbe navedene jasno, da počinju od vrha i da se izvršavaju do dna. Ova ideja se formalno opisuje kontrolom toka.

Na slici 2.1 je prikazan algoritam koji opisuje rad aplikacije. Aplikacija je razvijena pomoću ovog algoritma.



Slika 2.1 Algoritam rada aplikacije za naplatu parkiranja

2.3. Implementacija algoritma za naplatu parkiranja

Zahtevi koji su dati se implementiraju distribuirano u aplikaciji, pomoću klasa. Softver koji je napisan na ovaj način je pregledniji i jednostavnije je tražiti greške u kodu.

Klase koje u pozadini obrađuju komunikaciju sa uređajima su sledeće:

- Klasa koja izvršava komunikaciju sa uređajem koji prepoznaje novčanice.
- Klasa koja izvršava komunikaciju sa uređajem koji prepoznaje kovanice.
- Klasa koja izvršava komunikaciju sa štampačem.

- Klasa koja sadrži izgled potvrde.
- Klasa koja izvršava čitanje i upis u bazu podataka.

Identifikatori klasa su sledeći: NV10[4], RH5HD, Štampač, Ticket2017PS i LokalnaBazaPodataka. Zajednički deo u klasama NV10, RM5HD i Štampaču je da u svakoj pozivamo klasu serijski port, koji omogućava komunikaciju sa uređajem.

2.4. Uređaj za prepoznavanje kovanica

RM5HD je uređaj koji vrši prepoznavanje kovanica. Na uređaj se prvo mora upisati firmver koji omogućuje prepoznavanje valute-kovanice Republike Srbije. Prepoznavanje kovanice se vrši pomoću težine kovanice. Upisani firmvare prepoznaje kovanice od 5, 10 i 20 dinara, i uređaj koristi CCTalk protokol.

Važan deo naredbi je reset broja ubačenih kovanica i vrednost ubačenih kovanica. Uređaj nije opremljen motorima koji bi zadržali kovanicu kako bi sa našim programom mogli vršiti prihvatanje kovanice. Ovu situaciju koriguje naredba “229”, tako što prethodno ubačenu kovanicu sačuva u bafer memoriju. Na slici 2.2 prikazana je obrada odgovora od uređaja kada se pošalje naredba “229”. [1]

Odgovor od uređaja se sastoji od 21 bajta, u ovom slučaju prvih pet bajtova je komanda za zahtev koju šaljemo u funkciju “requestStatus”. Posledica je da je potrebno proveravati odgovor, da li je broj bajtova 21. Samo u slučaju da jeste, program će parsirati odgovor. Potrebno je izdvojiti prvih pet bajtova odgovora. Metoda “copy” će napraviti kopiju počev od 5. indeksa u nizu. Proveravanje ispravnosti odgovora vrši funkcija “calculateCHK” i samo ako je ispravan odgovor, program će nastaviti parsiranje odgovora. Potrebno je izvršiti proveru prosleđenih događaja, ako se desio novi događaj, treba izračunati koliko se događaja desilo. Posle toga sa “for” petljom pokupimo sve bajtove koji su novi događaji.

```

private void parseData(int[] data){
    if (data.Length == 21) {
        int[] _temp = new int[16];
        Array.Copy(data, 5, _temp, 0, 16);
        int[] _data = new int[10];
        if (calculateCHK(_temp)){
            if (lastEvent != _temp[4]){
                int _brojac = _temp[4] - lastEvent;
                if (_brojac > 5){
                    _brojac = 5;
                }
                logger.Info("LASTEVENT: " + _temp[4]);
                lastEvent = _temp[4];
                for (int i = 0; i < _brojac; i++){
                    _data[2 * i] = _temp[(i + 1) * 5];
                    _data[i + 1] = _temp[2 * i + 5];
                }
                for (int i = 0; i < _brojac; i++){
                    if (_data[i + 1] != 0){
                        int pare = 0;
                        switch (_data[i + 1]){
                            case 1: pare = 5; break;
                            case 2: pare = 10; break;
                            case 3: pare = 20; break;
                            default: pare = 0; break;
                        }
                        if (eUbacenApoen != null)
                            eUbacenApoen.Invoke(pare);
                    }
                }
            }
        }
    }
}
  
```

Slika 2.2 Obrada podataka o vrednosti kovanice

2.5. Uređaj za prepoznavanje novčanice

Ovaj uređaj prepoznavanje novčanica vrši pomoću optičkog senzora. Senzor prepoznaje na novčanici vođeni žig, kontrolnu metalnu nit, ultraljubičaste markere, potpis guvernera, serijski broj i dimenziju same novčanice. Kvalitet novčanica koje su optičaju je loš jer novčanice nisu plastificirane. Plastificirane novčanice se teže cepaju i savijaju. Zbog toga senzori često ne mogu izvršiti prepoznavanje novčanice jer ne mogu prepoznati parametre koji definišu određenu novčanicu te ih uređaj vraća.[2] Na slici 2.3 prikazana je obrada podataka kada uređaj prepoznao validnu novčanicu.

```
void NV10_DataReceived(object
sender, System.IO.Ports.SerialDataReceivedEventArgs e)
{
    try
    {
        int prvi, drugi, novac;
        int ubacenIznos = 0;
        while ((prvi = comNV10.ReadByte()) != 120 && bNV10radi)
            Thread.Sleep(5);
        drugi = comNV10.ReadByte();
        Thread.Sleep(100);
        if (drugi == 121)
        {
            zauzet = true;
            novac = comNV10.ReadByte();
            primi_novac();
            switch (novac)
            {
                case 1: ubacenIznos = 10; break;
                case 2: ubacenIznos = 20; break;
                case 3: ubacenIznos = 50; break;
                case 4: ubacenIznos = 100; break;
                case 5: ubacenIznos = 200; break;
                case 6: ubacenIznos = 500; break;
                case 7: ubacenIznos = 1000; break;
            }
            if (eUbacenApoen != null)
            {
                eUbacenApoen.Invoke(ubacenIznos);
            }
        }
    }
    catch (Exception ex)
    {
        loggerNV10.Info("NV10 ... Greska: " + ex);
    }
}
```

Slika 2.3 Obrada podataka o vrednosti novčanice

2.6. Pristup lokalnoj bazi podataka

Kada je potrebno pristupiti lokalnoj bazi podataka, moramo napraviti komandu tj. upit. Upit eng. "query" u svakoj funkciji je definisan u lokalnoj promenljivoj. Dodatna biblioteka "sqlite" ima metodu koja će napraviti komandu od lokalne promenljive. Konekcija sa lokalnom bazom je izvršena pomoću klase „SQLiteConnection“. Na slici 2.4 je prikazana funkcija koja vrši upis u bazu podataka.

```
public void insertBazaPodataka(String reg_tablica, int kredit,
DateTime Vazi_Do, String Email)
{
    lock (zakljuci_2)
    {
        if (postoji == false || strCount == "0")
        {
            strKom = "INSERT INTO Baza_Pod" + "
(TABLEICA, KREDIT,EMAIL , VAZIDO) VALUES " +
"(@Reg, @Kredit,@Email ,@Vazi)";
            using (SQLiteConnection conn = new
SQLiteConnection(strKoneccija))
            {
                conn.Open();
                using (var comm = conn.CreateCommand())
                {
                    comm.CommandText = strKom;
                    comm.Parameters.Add("@Reg",
System.Data.DbType.String).Value = reg_tablica;
                    comm.Parameters.Add("@Kredit",
System.Data.DbType.Int32).Value = kredit;
                    comm.Parameters.Add("@Email",
System.Data.DbType.String).Value = Email;
                    comm.Parameters.Add("@Vazi",
System.Data.DbType.DateTime).Value = Vazi_Do;

                    comm.ExecuteNonQuery();
                    strCount = "1";
                }
                conn.Close();
            }
        }
    }
}
```

Slika 2.4 Funcija koja vrši upis u bazu podataka

Podaci koji se upisuju u bazu su registraska tablica, preostali iznos, e-mail adresa, vreme trajanja parkiranja.

2.7. Rezultati

Testiranje softvera za naplatu parkiranja na otvorenim parkiralištima je urađeno na automatu čiji je izgled prikazan na slici 2.5 Razvijeni korisnički interfejs omogućuje da se jasno vide sve strane i svaki uneti podatak koji se prikazuje na stranama. Komunikacija sa uređajem je u testnom periodu pokazala da nije bilo greške u njegovom radu, ali su se javljale greške zbog lošeg kvaliteta novčanica ili kovanica, odnosno zbog ubacivanja novčanica/kovanica drugih valuta. Uočeno je da postoji u nekim slučajevima sporiji odziv posle unosa registrarske tablice, budući da aplikacija pristupa lokalnoj bazi podataka registrarskih tablica.



Slika 2.5 Izgled automata za naplatu parkiranja na otvorenim parkiralištima

U toku testiranja rada aplikacije svi korisnici su bili zadovoljni načinom kojim se uspostavlja komunikacija sa automatom i bili su zadovoljni predloženim konceptom naplate parkiranja na otvorenim parkiralištima.

3. ZAKLJUČAK

Automat za naplatu parkiranja se dobro pokazao prilikom testiranja na spoljne uslove korišćenja. To znači da smo koristili komponente koje su bile otporne na vremenske prilike, s jedne strane, a sa druge strane su optimalno odgovarale postavljenim zahtevima.

Ugrađeni elementi su korektno prepoznavali novčanice i kovanice. Eventualne teškoće sa prepoznavanjem novčanica su se javile u situaciji kada su novčanice bile pohabane odnosno ukoliko ubačene kovanice nisu bile naše platežno sredstvo.

Zbog kapaciteta korišćenog računara, program je sporije reagovao na informacije. Ukoliko bi se u praksi pokazala potreba da se vreme reagovanja ubrza, tada to možemo postići korišćenjem drugog računara. Imajući u vidu obim ulaganja u automat za naplatu parkiranja smatram da za sada nije potrebno korišćenje drugog računara jer postignuta brzina odgovara svakodnevnim potrebama korisnika a ugradnja drugog računara bi uvećala troškove. Iskazana je odlična pouzdanost softvera, za vreme testiranja nije bilo prekida njegovog rada.

Uočeni su problemi sa povezivanjem na Internet zbog operativnog sistema računara. Ovaj problem se može razrešiti tako što bi se ugradila dodatna biblioteka za komunikaciju sa serverom.

Poboljšanje sistema je moguće u domenu brzine reagovanja, ukoliko se za to ukaže potreba, na taj način što možemo optimizovati softver ali to iziskuje dodatan rad.

Imajući u vidu uložena sredstva (rad i korišćenu opremu) u odnosu na postignuti rezultat, možemo zaključiti da je postignut rezultat dobar jer automat za naplatu parkiranja korektno obavlja sve postavljene zahteve tj. podoban je za svakodnevnu upotrebu.

4. LITERATURA

- [1] https://www.casino-software.de/download/manual_rm5.pdf
- [2] <http://www.digislot.ro/download/brosuri-acceptoare-de-bancnote/DGL-PRO-NV10.pdf>
- [3] C# 6.0 and the .NET 4.6 Framework by Andrew Troelsen and Philip Japikse
- [4] <http://www.innovative-technology.com/products/products-main/127-nv10-usb>

Kratka biografija:



Atila Smieško rođen je u Senti 1993. god. Diplomirao je na Fakultetu tehničkih nauka u Novom Sadu 2016.g. sa diplomskim radom Prepoznavanje alfanumeričkih karaktera u okviru mašinskog učenja.

U realizaciji Zbornika radova Fakulteta tehničkih nauka u toku 2017. godine učestvovali su sledeći recenzenti:

Aco Antić	Dušan Gvozdenac	Milan Trifković	Slavko Đurić
Aleksandar	Dušan Kovačević	Milan Trivunić	Slobodan Dudić
Erdeljan	Dušan Uzelac	Milan Vidaković	Slobodan Krnjetin
Aleksandar Ristić	Duško Bekut	Milena Krklješ	Slobodan Morača
Bato Kamberović	Đorđe Ćosić	Milica Kostreš	Sonja Ristić
Biljana Njegovan	Đorđe Lađinović	Milica Miličić	Srđan Kolaković
Bogdan	Đorđe Obradović	Milinko Vasić	Srđan Popov
Kuzmanović	Đorđe Vukelić	Miloš Slankamenac	Srđan Vukmirović
Bojan Batinić	Đura Oros	Milovan Lazarević	Staniša Dautović
Bojan Lalić	Đurđica Stojanović	Miodrag	Stevan Milisavljević
Bojan Tepavčević	Filip Kulić	Hadžistević	Stevan Stankovski
Bojana Beronja	Goran Sladić	Miodrag Zuković	Strahil Gušavac
Branislav Atlagić	Goran Švenda	Mirjana	Svetlana Nikoličić
Branislav Nerandžić	Gordana	Damnjanović	Tanja Kočetov
Branislav Veselinov	Milosavljević	Mirjana Malešev	Tatjana Lončar
Branislava Kostić	Gordana Ostojić	Mirjana Radeka	Turukalo
Branislava	Igor Budak	Mirko Borisov	Todor Bačkalić
Novaković	Igor Dejanović	Miro Govedarica	Toša Ninkov
Branka Nakomčić	Igor Karlović	Miroslav	Uroš Nedeljković
Branko	Ivan Beker	Hajduković	Valentina Basarić
Milosavljević	Ivana Katić	Miroslav Plančak	Velimir Čongradec
Branko Škorić	Ivana Kovačić	Miroslav Popović	Velimir Todić
Cvijan Krsmanović	Jasmina Dražić	Mitar Jocanović	Veran Vasić
Damir Đaković	Jelena Atanacković	Mladen Kovačević	Veselin Avdalović
Danijela Lalić	Jeličić	Mladen Radišić	Veselin Perović
Darko Čapko	Jelena Borocki	Momčilo Kujačić	Vladimir Katić
Darko Marčetić	Jelena Kiurski	Nađa Kurtović	Vladimir
Darko Reba	Jelena Radonić	Nebojša Pjevalica	Radenković
Dejan Ubavin	Jovan Petrović	Neda Pekarić Nađ	Vladimir Strezoski
Dragan Ivanović	Jovan Vladić	Nemanja	Vladimir Škiljajica
Dragan Ivetić	Jovanka Pantović	Stanisavljević	Vlado Delić
Dragan Jovanović	Karl Mičkei	Nenad Katić	Vlastimir
Dragan Kukolj	Katarina Gerić	Nikola Brkljač	Radonjanin
Dragan Mrkšić	Ksenija Hiel	Nikola Đurić	Vuk Bogdanović
Dragan Pejić	Laslo Nađ	Nikola	Zdravko Tešić
Dragan Šešlija	Leposava Grubić	Jorgovanović	Zoran Anišić
Dragana Bajić	Nešić	Nikola Radaković	Zoran Brujić
Dragana	Livija Cvetičanin	Ninoslav Zuber	Zoran Jeličić
Konstantinović	Ljiljana Vukajlov	Ognjen Lužanin	Zoran Mijatović
Dragana Šarac	Ljiljana Cvetković	Pavel Kovač	Zoran Milojević
Dragana Štrbac	Ljubica Duđak	Peđa Atanasković	Zoran Mitrović
Dragiša Vilotić	Maja Turk Sekulić	Petar Malešev	Zoran Papić
Dragoljub	Maša Bukurov	Predrag Šiđanin	Željen Trpovski
Novaković	Matija Stipić	Radivoje Dinulović	Željko Jakšić
Dragoljub Šević	Milan Rackov	Radovan Štulić	
Dubravka Bojanić	Milan Rapajić	Rastislav Šostakov	
Dušan Dobromirov	Milan Simeunović	Slavica Mitrović	

