



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



ЗБОРНИК РАДОВА ФАКУЛТЕТА ТЕХНИЧКИХ НАУКА

Едиција: Техничке науке - зборници

Година: XXXIV

Број: 9/2019

Нови Сад

Едиција: „Техничке науке – Зборници“

Година: XXXIV

Свеска: 9

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 девета 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 11.01.2019. до 14.06.2019. год., а који се промовишу у октобру 2019. 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 две свеске.

У овој свесци, са редним бројем 9., објављени су радови из области:

- машинства,
- електротехнике и рачунарства,
- саобраћаја,
- мехатронике и
- инжењерства информационих система.

У свесци са редним бројем 10. објављени су радови из области:

- грађевинарства,
- графичког инжењерства и дизајна,
- архитектуре,
- инжењерског менаџмента.
- инжењерства заштите на раду и заштите животне средине,
- математике у техници,
- геодезије и геоматике,
- управљања ризиком од катастрофалних догађаја и пожара и
- информационог инжењеринга.

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. Lazar Ristić, Đorđe Vukelić, TRIBODIJAGNOSTIKA STATIČKOG KOEFICIJENTA TRENJA PO PRINCIPU KOSE RAVNI	1491-1494
2. Nikola Milivojević, Damir Đaković, PRORAČUN KOMORNE SUŠARE ZA PEČURKE	1495-1498
3. Veljko Mladenović, PROJEKTOVANJE PREDNJEG DELA KAROSERIJE LAKOG HIBRIDNOG VOZILA	1499-1502
4. Marko Narić, ISTRAŽIVANJE UTICAJA LOKALNOG OTPORA POSTAVLJENOG ISPRED MERNE BLENDE NA STRUJNE PARAMETRE	1503-1506
5. Вук Спремо, ИСТРАЖИВАЊЕ УТИЦАЈА ЛОКАЛНОГ ОТПОРА ПОСТАВЉЕНОГ ИЗА МЕРНЕ БЛЕНДЕ НА СТРУЈНЕ ПАРАМЕТРЕ	1507-1510
6. Novica Nikolić, PROJEKTOVANJE TEHNOLOGIJE TERMIČKE OBRADJE LEŽAJNOG PRSTENA PRIMENOM RAČUNARA	1511-1514

Radovi iz oblasti: Elektrotehnika i računarstvo

1. Јелена Јаковљевић, НАМЕНСКИ ЈЕЗИК ЗА МОДЕЛОВАЊЕ РАЧУНАРСКИХ МРЕЖА И ГЕНЕРИСАЊЕ КОНФИГУРАЦИЈА SWITCH УРЕЂАЈА	1515-1518
2. Simonida Kadić, ARHITEKTURA I IMPLEMENTACIJA SISTEMA ZA ANALIZU PODATAKA PAMETNOG GRADA	1519-1522
3. Teodora Gavrilov, UTICAJ SOLARNIH PANELA NA STRUJU KRATKOG SPOJA U NISKONAPONSKOJ DISTRIBUTIVNOJ MREŽI	1523-1526
4. Miloš Spasić, MODELOVANJE MODERNIH DISTRIBUTIVNIH ENERGETSKIH RESURSA ZASNOVANIH NA UREĐAJIMA ENERGETSKE ELEKTRONIKE ZA POTREBE PRORAČUNA KRATKIH SPOJEVA	1527-1530
5. Luka Pavlica, SMERNICE KVALITETA ANDROID APLIKACIJA	1531-1534

	STRANA
6. Branislav Vejnović, OPTIMALNO ANGAŽOVANJE KONDENZATORA U DISTRIBUTIVNOJ MREŽI ZA PERIOD OD 24H	1535-1538
7. Miroslav Korica, UPRAVLJAČKE STRATEGIJE U POSLOVNO-STAMBENIM OBJEKTIMA	1539-1542
8. Helena Petričević, Vladan Krsman, UTICAJ DISTRIBUTIVNIH GENERATORA NA KOORDINACIJU ZAŠTITE IZMEĐU PREKIDAČA I OSIGURAČA U RADIJALNIM DISTRIBUTIVNIM MREŽAMA	1543-1546
9. Nikola Nešković, UVAŽAVANJE PETERSENOVE PRIGUŠNICE U RESTAURACIJI NAPAJANJA VELIKOG PODRUČJA	1547-1550
10. Igor Bjelica, UTICAJ PRIKLJUČENJA FOTONAPONSKE ELEKTRANE NA DISTRIBUTIVNU MREŽU	1551-1554
11. Mirko Đurić, PRORAČUN REŽIMA ELEKTROENERGETSKIH SISTEMA SA KVAROM U FAZNOM DOMENU....	1555-1558
12. Milica Milutinović, PREDIKCIJA VREDNOSTI KRIPTOVALUTA ANALIZOM ISTORIJSKIH CENA, BLOKČEJN INFORMACIJA I SENTIMENTA TVITOVA	1559-1562
13. Vladan Rakić, ENEL OPTIMALNA REKONFIGURACIJA MREŽE	1563-1566
14. Željko Bal, RAZVOJ JEZIKA ZA OPIS KOMANDNIH KORISNIČKIH INTERFEJSA	1567-1570
15. Jan Šulja, VREMENSKO I LOKACIJSKO PROŠIRENJE KONTROLE PRISTUPA BAZIRANE NA ULOGAMA U SPRING SECURITY OKRUŽENJU	1571-1574
16. Igor Pušin, Vladimir Katić, MODELOVANJE I ISPITIVANJE BIOGASNE ELEKTRANE – PRIMER “PIK MORAVICA”	1575-1578
17. Бојан Батало, НАМЕНСКИ ЈЕЗИК И ОКРУЖЕЊЕ ЗА МОДЕЛОВАЊЕ И ГЕНЕРИСАЊЕ СИСТЕМА СПОСОБНОСТИ У ИГРАМА СА УЛОГАМА	1579-1582
18. Mina Lakić, PRORAČUN PREKIDA FAZE U DISTRIBUTIVNOJ MREŽI.....	1583-1586
19. Siniša Kolarević, IMPLEMENTACIJA SERVERLESS APLIKACIJE ZA IOS UREĐAJE POMOĆU FIREBASE SERVISA....	1587-1589
20. Tamara Žepinić, VIŠEKRITERIJUMSKO ODLUČIVANJE U PLANIRANJU RAZVOJA DISTRIBUTIVNIH MREŽA	1590-1593
21. Александар-Вук Павловић, Мирослав Поповић, РЕАЛИЗАЦИЈА БЕЖИЧНОГ АКУСТИЧНОГ СЕНЗОРА БАЗИРАНА НА БЛУТУТ ТЕХНОЛОГИЈИ	1594-1597
22. Vasilija Šešlija, UTICAJ DISTRIBUTIVNIH GENERATORA NA AUTOMATSKU REGULACIJU NAPONA	1598-1601
23. Zlatan Prečanica, Milan Vidaković, RAZVOJ SKLADIŠTENJA I PRETRAGE DIGITALNE BIBLIOTEKE DIPLOMSKIH RADOVA UPOTREBOM MongoDB, Elasticsearch I Blockchain TEHNOLOGIJA	1602-1605
24. Јасмина Еминовски, Милан Видаковић, ИЗРАДА ВЕБ АПЛИКАЦИЈЕ ЗА УПРАВЉАЊЕ БИБЛИОТЕЧКИМ ИНФОРМАЦИОНИМ СИСТЕМОМ НА ПЛАТФОРМИ ASP.NET	1606-1609
25. Zoltan Berec, OPTIMIZACIJA AUTOMATSKOG PODFREKVENTNOG RASTEREĆENJA POTROŠAČA PRIMENOM GENETSKOG ALGORITMA	1610-1613

	STRANA
26. Slađan Aćimović, PRILAGOĐENA KLASIČNA REGULACIJA NAPONA U AKTIVNIM DISTRIBUTIVNIM MREŽAMA	1614-1617
27. Jovan Marković, AUTOMATSKO UPRAVLJANJE VIŠE SPREGNUTIH KLIMA KOMORA I BALANSIRANJE HVAC SISTEMA ČISTIH PROSTORA U NATPRITISKU	1618-1621
28. Bilja Sjeran, MIKROPROCESORSKA ZAŠTITA ELEKTROMOTORA	1622-1625
29. Balaž Petrović, REALIZACIJA EDUKACIONOG MIKRORAČUNARA NA ZYNQ KONFIGURABILNOM SISTEMU NA ČIPU	1626-1629
30. Биљана Анђелић, ЈЕЗИК И ОКРУЖЕЊЕ ЗА МОДЕЛОВАЊЕ И ГЕНЕРИСАЊЕ ВЕБ СТРАНИЦА ИНФОРМАЦИОНИХ СИСТЕМА ОПШТЕ НАМЈЕНЕ	1630-1633
31. Balaž Kereši, IZBOR OPTIMALNIH POZICIJA I TIPA DISTRIBUTIVNIH GENERATORA I KONDEZATORSKIH BATERIJA U DISTRIBUTIVNOJ MREŽI PRIMENOM MOPSO	1634-1637
32. Aleksandra Tešić, Vladan Krsman, UPRAVLJANJE NEPLANIRANIM ISPADIMA U ELEKTRODISTRIBUTIVNIM MREŽAMA	1638-1641
33. Maja Zetko, NAMENSKI JEZIK I OKRUŽENJE ZA MODELOVANJE I GENERISANJE INFORMACIONIH SISTEMA OPŠTE NAMENE	1642-1645
34. Nataša Subić, IMPLEMENTACIJA MOBILNE APLIKACIJE TRAVELING POMOĆU REACT NATIVE RADNOG OKVIRA	1646-1649
35. Marko Kljajić, POSLOVNI INTEGRACIONI ŠABLONI I NJIHOVA UPOTREBA U APACHE CAMEL OKRUŽENJU	1650-1653
36. Aleksandra Kerezović, Željko Trpovski, Dejan Nemec, PROJEKTOVANJE OPTIČKE MREŽE U NASELJU SREDNJE VELIČINE	1654-1657
37. Stojan Mitrić, TEHNOLOGIJE I SOFTVER U 3D ŠTAMPI SA PRIMEROM PRIMENE U PČELARSTVU	1658-1661
38. Дарко Китановић, ПРИМЕНА СОФТ СТАРТЕРА SIEMENS 3RW4426-1BC44 У ИНДУСТРИЈСКОЈ ПУМПНОЈ СТАНИЦИ DARKHAN – MONGOLIA	1662-1665
39. Nevena Dovičin, KOMBINATORNI ALGORITMI- PROBLEM TRGOVAČKOG PUTNIKA I RASPOREĐIVANJE PROCESA PROCESORA	1666-1669
40. Emir Murtezić, Vladimir Popović, Darko Marčetić, ROBUSTNI KONTROLER MAKSIMALNE EFIKASNOSTI U POGONU AM ZA ELEKTRIČNA VOZILA	1670-1673
41. Ивана Савин, Милан Видаковић, РАЗВОЈ JavaScript БИБЛИОТЕКЕ ЗА РАД СА КОМПОНЕНТАМА КОРИСНИЧКОГ ИНТЕРФЕЈСА	1674-1677

Radovi iz oblasti: Saobraćaj

1. Mihajlo Blagojević, ANALIZA POTREBA ZA IZGRADNJOM SKLADIŠTA ZA VOĆE I POVRĆE U REPUBLICI SRBIJI	1678-1681
--	-----------

	STRANA
2. Слободан Нинков, Павле Питка, УТИЦАЈ ЕЛЕКТРОНСКОГ СИСТЕМА НАПЛАТЕ НА ДИНАМИЧКЕ ЕЛЕМЕНТЕ ЛИНИЈЕ	1682-1685

Radovi iz oblasti: Mehatronika

1. Miloš Kovačević, IMPLEMENTACIJA STANDARDA ENERGETSKE EFIKASNOSTI ISO 50001 U PROIZVODNOM SISTEMU	1686-1689
---	-----------

Radovi iz oblasti: Inženjerstvo informacionih sistema

1. Sandra Veljkov, IMPLEMENTACIJA BIKONA U POSLOVANJE KOMPANIJE	1690-1693
2. Đorđo Kalaković, SISTEM ZA AUTOMATIZACIJU PROCESA IZRADE LIČNIH DOKUMENATA	1694-1697

TRIBODIJAGNOSTIKA STATIČKOG KOEFICIJENTA TRENJA PO PRINCIPU KOSE RAVNI**TRIBODIAGNOSTICS OF THE STATIC FRICTION COEFFICIENT BY PRINCIPLE OF INCLINED PLANE**Lazar Ristić, Đorđe Vukelić, *Fakultet tehničkih nauka, Novi Sad***Oblast – PROIZVODNO MAŠINSTVO**

Kratak sadržaj – U radu su prezentovani rezultati merenja statičkog koeficijenta trenja po principu kose ravni. Eksperimentalna istraživanja su sprovedena za kontaktne parove napravljene od različitih materijala. Dobijeni rezultati ukazuju da kontaktne parove čiji su elementi napravljeni od čelika, sivog liva i mesinga karakterišu niže vrednosti statičkog koeficijenta trenja. Gumu kao materijal kontaktnog para karakterišu visoke vrednosti statičkog koeficijenta trenja. Na kraju su dati odgovarajući zaključci i pravci budućih istraživanja.

Ključne reči: Tribometar, statički koeficijent trenja

Abstract – The results of the measurement of the static friction coefficient on the principle of inclined plane are presented. Experimental investigations were carried out for contact pairs made of different materials. The obtained results indicate that contact pairs whose elements are made of steel, cast iron and brass are characterized by lower values of the static friction coefficient. The rubber as a contact-pair material is characterized by high values of the static friction coefficient. At the end, appropriate conclusions and future directions research are given.

Keywords: Tribometer, static friction coefficient

1. UVOD

Tribologija je jedna od tehničkih disciplina čiji razvoj ima izuzetan uticaj na pouzdanost i vek mašinskih sistema. Tribološka istraživanja i korišćenje već ostvarenih znanja, kako u procesu konstruisanja tako i u proizvodnji i eksploataciji, predstavljaju neophodnost savremene industrije kada se posmatraju njeni današnji zahtevi za ekonomično korišćenje materijala, racionalno trošenje energije, povećanje pouzdanosti mašinskih sistema i smanjenje troškova održavanja. Tribologija je naučno-tehnička disciplina koja obuhvata istraživanja procesa trenja, habanja i podmazivanja, kao i postupaka za smanjenje i optimizaciju ovih procesa kod različitih mašinskih sistema [1].

Trenje se definiše kao sila koja deluje između površina kontakta dva tela suprotstavljajući se njihovom relativnom kretanju.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Đorđe Vukelić, vanr. prof.

Trenje je složena pojava koja zavisi od velikog broja faktora kontaktnih površina, kao što su: materijal, geometrija, brzina klizanja (kotrljanja), normalno opterećenje i dr. [2].

Sila trenja je proporcionalna normalnoj sili koja deluje na površine koje su u kontaktu, što se može prikazati sledećim izrazom [3]:

$$F_t = \mu \cdot F_n \quad (1)$$

gde je: F_t sila trenja, μ koeficijent trenja a F_n normalna sila. Veličina sile trenja ne zavisi od veličine kontaktne površine i ravnomerno je raspoređena po njoj. Sila trenja deluje tangencijalno u odnosu na kontaktnu površinu i ima smer dejstva suprotan smeru tendencije kretanja.

Prema stanju kretanja tela, trenje se deli na [4]:

- trenje mirovanja (statičko trenje) - otpor koji se javlja na dodirnim površinama dva tela koja miruju,
- trenje kretanja (kinematsko trenje) - otpor koji se javlja na dodirnim površinama dva tela koja se kreću.

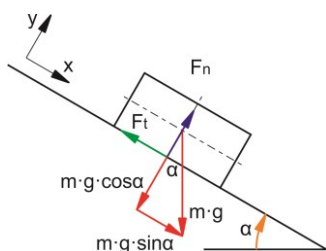
U stanju mirovanja sila trenja je na mestima kontakta površina veća ili jednaka sa rezultantom tangencijalnih sila koja izaziva kretanje. Trenje postoji na površini kontakta čvrstih tela i kada su tela u relativnom mirovanju, jer je potrebno uložiti znatnu silu da bi se izazvalo početno relativno kretanje. Nakon neke granične vrednosti sile telo se pokreće. Sila trenja potom raste do neke maksimalne vrednosti statičke sile trenja. Nakon toga javlja se relativno kretanje između tela a sila trenja postaje kinematska sila trenja, a njena vrednost je nešto manja od statičke sile trenja. Pri tome, dok tela miruju koeficijent trenja ima jednu vrednost a tokom kretanja drugu vrednost. Vrednosti koeficijenta trenja tokom kretanja su manje nego tokom mirovanja u identičnim uslovima ostvarivanja kontakta [5,6].

U realnim industrijskim sistemima vrlo je teško, a u većini slučajeva i nemoguće, kvantitativno odrediti veličinu koeficijenta trenja. Merenje sile trenja i normalnog opterećenja, odnosno određivanje koeficijenta trenja vrši se na posebnim mernim uređajima – tribometrima.

2. METODOLOGIJA

Merenje statičkog koeficijenta trenja je moguće izvesti na različite načine, a jedan od njih je princip kose ravni (slika 1) koji se zasniva na gravitacionoj sili [7,8]. Na slici je predstavljeno telo mase (m) koje se nalazi u stanju statičke ravnoteže. Telo se nalazi na kosoj ravni koja je nagnuta u odnosu na horizontalu za određeni ugao (α). Na

telo deluje gravitaciona sila ($Q=m \cdot g$) u vertikalnom pravcu, sila trenja (F_t) u pravcu kose ravni i normalna reakcija podloge (F_n) upravno na pravac kose ravni.



Slika 1. Raspored sila

Statički koeficijent trenja, po definiciji, jednak je:

$$\mu = F_t / F_n \quad (2)$$

Na osnovu statičkih uslova ravnoteže za x i y osu (slika 1) sledi da je:

$$F_t = m \cdot g \cdot \sin \alpha \quad (3)$$

$$F_n = m \cdot g \cdot \cos \alpha \quad (4)$$

Zamenom izraza (3) i (4) u (2) dobija se izraz za izračunavanje statičkog koeficijenta trenja kontaktnog para:

$$\mu = m \cdot g \cdot \sin \alpha / m \cdot g \cdot \cos \alpha \quad (5)$$

$$\mu = \sin \alpha / \cos \alpha \quad (6)$$

$$\mu = \tan \alpha \quad (7)$$

Na osnovu izraza (7) moguće je izračunati statički koeficijent trenja ako se predhodno izmeri ugao nagiba kose ravni (α) pri kojoj dolazi do pokretanja jednog elementa kontaktnog para u odnosu na drugi.

Tribometar upotrebljen u eksperimentalnim istraživanjima koji funkcioniše na principu kose ravni prikazan je na slici 2. Sklop tribometra sastavljen je iz funkcionalnih celina koji omogućavaju ostvarivanje potrebnog zakretanja kose ravni jednog elementa kontaktnog para u odnosu na drugi.



Slika 2. Tribometar

Pre izvođenja postupka merenja potrebno je izvršiti nivelaciju uređaja i dovođenje kose ravni u početni (horizontalni) položaj zakretanjem diska.

Na taj način se obezbeđuje zauzimanje horizontalnog položaja prvog elementa kontaktnog para. Nakon izvršene pripreme postavlja se prvi element kontaktnog para tako

da nalegne na čivije i jednu stranu udubljenja na disku. Potom se na njega postavlja drugi element. Time je formiran kontaktni par (slika 3).



Slika 3. Kontaktni par

Nakon formiranja kontaktnog par vrši se zakretanje diska a istovremeno sa njim i zakretanje kose ravni (slika 4). Poželjno je da se zakretanje vrši što manjom brzinom. kako bi se povećala tačnost merenja. Posle svakog zakretanja, od približno $0,5^\circ$, pravi se kratka pauza, i posmatra se da li drugi element kontaktnog para miruje ili je započelo njegovo kretanje. U trenutku kada se uoči početak kretanja, drugog elementa kontaktnog para u odnosu na prvi element, vrši se očitavanje ugla zakretanja kose ravni - α .



Slika 4. Zakretanje kose ravni

3. REZULTATI

Elementi kontaktnih parova na kojima su izvršena eksperimentalna istraživanja napravljeni su od sledećih materijala: čelik (S235JR), sivi liv (SL25), mesing (MS58) i silikonska guma.

Merenje ugla nagiba kose ravni izvršeno je za 10 kombinacija materijala kontaktnih parova, i to:

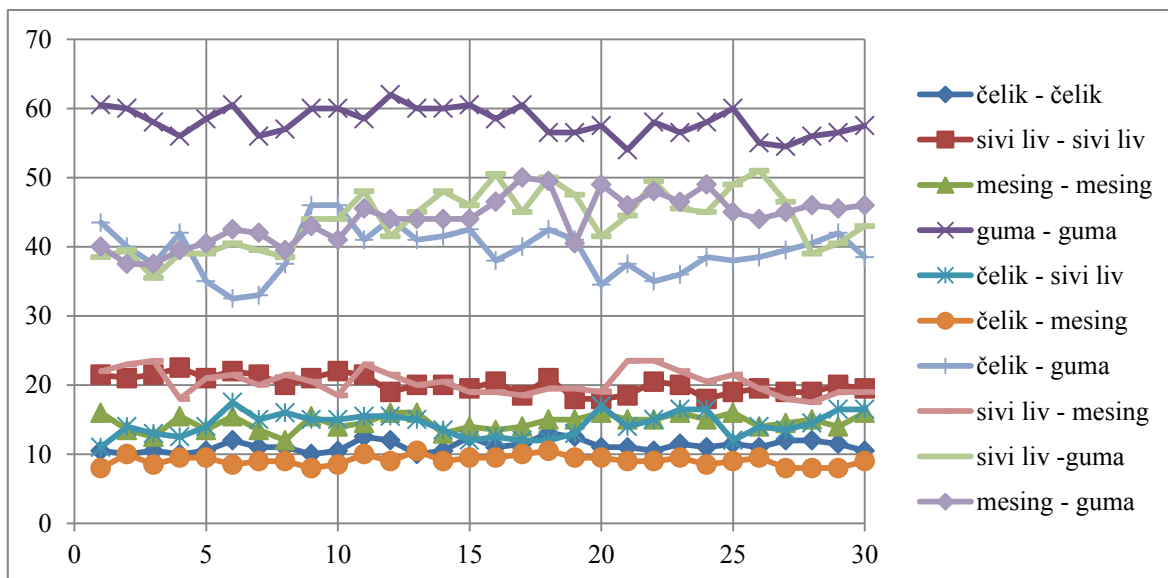
- kontaktni par "čelik - čelik",
- kontaktni par "sivi liv - sivi liv",
- kontaktni par "mesing - mesing",
- kontaktni par "guma - guma",
- kontaktni par "čelik - sivi liv",
- kontaktni par "čelik - mesing",
- kontaktni par "čelik - guma",
- kontaktni par "sivi liv - mesing",
- kontaktni par "sivi liv - guma",
- kontaktni par "mesing - guma".

Za svaki kontaktni par izvršeno je po 30 merenja ugla nagiba kose ravni. Izmerenje vrednosti ugla nagiba kose ravni su prikazane na slici 5.

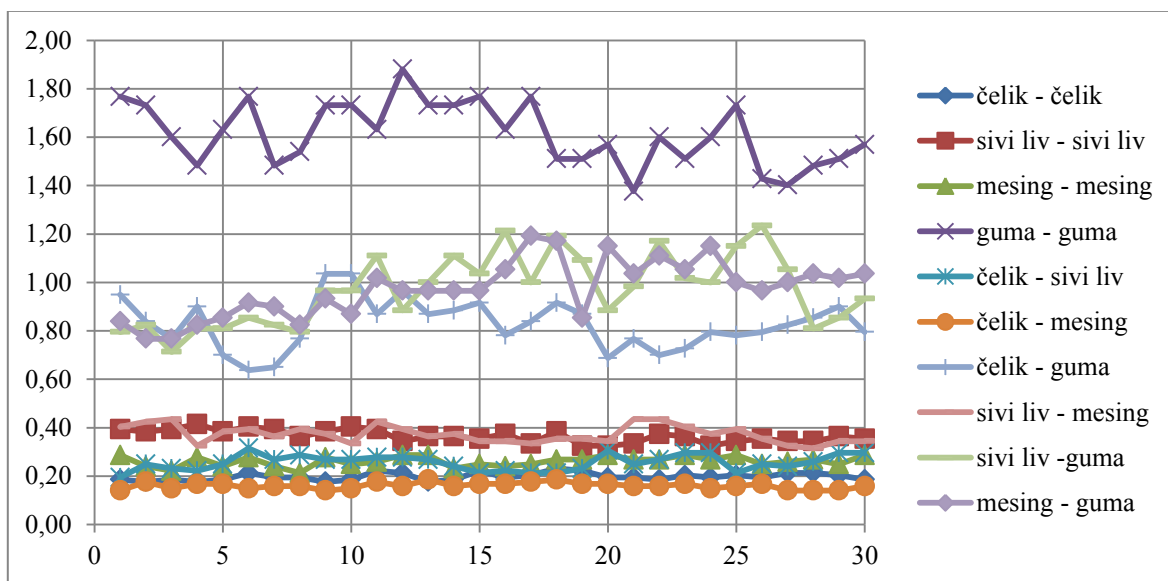
Na osnovu izmerenih vrednosti ugla nagiba kose ravni izračunate su vrednosti statičkog koeficijenta trenja za pojedine kontaktne parove (slika 6).

U tabeli 1 prikazane su srednje aritmetičke vrednosti statičkog koeficijenta trenja, njegove minimalne i maksimalne vrednosti, kao i standardne devijacije za pojedine kontaktne parove.

Na slici 7 su prikazane srednje aritmetičke vrednosti statičkog koeficijenta trenja za sve kontaktne parove.



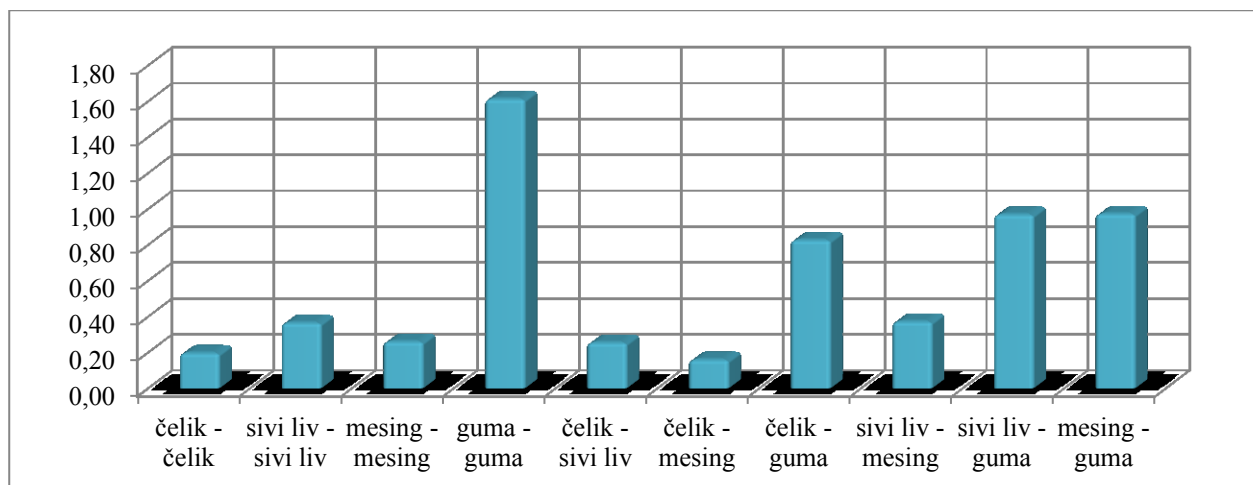
Slika 5. Izmerene vrednosti ugla nagiba kose ravni



Slika 6. Izračunate vrednosti statičkog koeficijenta trenja klizanja

Tabela 1. Statistički parametri

Parametar	Kontakti par									
	čelik čelik	sivi liv sivi liv	mesing mesing	guma guma	čelik sivi liv	čelik mesing	čelik guma	sivi liv mesing	sivi liv guma	mesing guma
$\bar{\mu}$	0,20	0,37	0,26	1,61	0,26	0,16	0,83	0,37	0,97	0,97
$\mu_{\min}$	0,18	0,32	0,21	1,38	0,19	0,14	0,64	0,32	0,71	0,77
$\mu_{\max}$	0,23	0,41	0,29	1,88	0,32	0,19	1,04	0,43	1,23	1,19
σ_{μ}	0,02	0,03	0,02	0,13	0,03	0,01	0,10	0,04	0,15	0,12



Slika 7. Srednje aritmetičke vrednosti statičkog koeficijenta trenja

Na bazi prikazanih rezultata može se konstatovati da su najveće vrednosti statičkog koeficijenta trenja klizanja dobijene za kontaktni par guma-guma, a najmanje vrednosti statičkog koeficijenta trenja za kontaktni par čelik-mesing.

Guma pokazuje izuzetno dobre tribološke karakteristike u smislu ostvarivanja izuzetno visokih vrednosti koeficijenta trenja. Srednje aritmetičke vrednosti statičkog koeficijenta trenja kontaktnih parova gde je jedan od elemenata od gume se kreću u rasponu od 0,83 do 0,97. To znači da je sila trenja veoma velika, tj. bliska normalnoj sili. S obzirom na to, preporučuje se upotreba elemenata od gume u slučajevima kada je potrebno ostvariti visoke koeficijente trenja. To se može postići „oslojavanjem“ elemenata u kontaktu sa gumom ili umetanjem podmetača i podloški napravljenim od gume između dva elementa. U slučaju da je potrebno obezbediti niske vrednosti koeficijenta trenja gumu kao materijal kontaktnog elementa ne treba upotrebljavati.

Niže vrednosti statičkog koeficijenta trenja se dobijaju za sve kontaktne parove u kojima ni jedan od elemenata nije napravljen od gume. U ovom slučaju srednje aritmetičke vrednosti statičkog koeficijenta trenja se kreću u rasponu od 0,16 do 0,37. Posebno niske vrednosti kinematskog koeficijenta trenja karakterišu kontaktno par čelik-mesing i kontaktni par čelik-čelik.

4. ZAKLJUČCI

U većini slučajeva, u različitim pokretnim elementima mašina, uređaja i postrojenja, trenje je nepoželjna i štetna pojava. Na savlađivanje sile trenja kao otpora kretanja bilo koje vrste, troši se znatna količina mehaničke energije. Kao posledica trenja na površinama u kontaktu dolazi do trošenja i zagrevanja materijala površina. Rezultati eksperimentalnih istraživanja ukazu, da u ovim slučajevima, kada je funkcija cilja optimizacije tribomehaničkog sistema minimum trenja, kao elemente kontaktnog para treba upotrebljavati elemente od čelika, sivog liva i mesinga.

Međutim, trenje u nekim slučajevima može biti korisna fizička pojava. Stoga, ako je funkcija cilja optimalizacije tribomehaničkog sistema maksimum trenja kao jedan od kontaktnih elemenata treba upotrebljavati gumu. Gumu karakteriše odlična rastegljivost i fleksibilnost. Velike

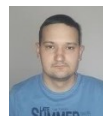
vrednosti statičkog koeficijenta trenja su posledica toga što guma može da podnese veće elastične deformacije od većine drugih materijala i da se vrati u početno stanje bez permanentne deformacije.

Buduća istraživanja biće usmerena na merenja statičkog koeficijenta trenja kontaktnih elemenata napravljenih od drugih materijala sa različitim karakteristikama.

5. LITERATURA

- [1] A. Rac, *Osnovi tribologije*, Mašinski fakultet, Beograd, 1991.
- [2] B. Ivković, A. Rac, *Tribologija*, Jugoslovensko društvo za tribologiju, Kragujevac, 1995.
- [3] J. P. Blau, „The significance and use of the friction coefficient“, *Tribology International*, Vol. 34, pp. 585-91, 2001.
- [4] A. Devaraju, A review on important factors affecting dry sliding friction, *Journal of Surface Science and Technology*, Vol. 32, No. 3-4, pp. 73-78, 2016.
- [5] I. M. Hutchings, Leonardo da Vinci's studies of friction, *Wear*, Vol. 360-361, pp. 51-66, 2016.
- [6] K. Kato, Wear in relation to friction - A review, *Wear*, Vol. 241, No. 2, pp. 151-157, 2000.
- [7] ASTM G219-18: *Standard Guide for Determination of Static Coefficient of Friction of Test Couples Using an Inclined Plane Testing Device*, ASTM International, 2018.
- [8] ASTM G115-10: *Standard Guide for Measuring and Reporting Friction Coefficients*, ASTM International, 2010.

Kratka biografija:



Lazar Ristić rođen je u Novom Sadu 1993. god. Master rad na Fakultetu tehničkih nauka iz oblasti Mašinskog inženjerstva odbranio je 2019.god.
kontakt: laleristic.rs@gmail.com@gmail.com



Đorđe Vukelić rođen je u Novom Sadu 1974. . godine. Doktorirao je na Fakultetu tehničkih nauka 2010. god., a od 2015 je zvanju vanrednog profesora.

PRORAČUN KOMORNE SUŠARE ZA PEČURKE DESIGN OF CHAMBER DRYER FOR MUSHROOMS

Nikola Milivojević, Damir Đaković, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – U okviru rada prikazan je primer proračuna i konstrukcije jedne konvektivne komorne sušare sa podovima za sušenje pečurki. Cilj rada je razmatranje više mogućih rešenja za zagrevanje agensa sušenja, sa fokusom na upotrebu obnovljivih izvora energije umesto konvencionalnih goriva, pre svega toplotnih pumpi. Na osnovu definisanog sistema, odabrana je odgovarajuća konstruktivna izvedba sušare.

Ključne reči: komorna sušara, pečurke, toplotna pumpa

Abstract – This paper represents an example of calculation and design of convective hot air chamber dryer with trays for drying mushrooms. Aim of this paper is analysis of few possible solutions for heating of drying air, with focus on use of renewable energy sources instead conventional fuels, especially heat pumps. Based on defined system for heating air, suitable design of represented dryer is selected.

Keywords: Chamber dryer, mushrooms, heat pump

1. UVOD

Sušenje predstavlja jedan od osnovnih načina preradu i čuvanje prehrambenih proizvoda sa visokim sadržajem vlage, sa težnjom da se u tom procesu ostvari što manje uticaja na njihove nutritivne karakteristike. Sam proces sušenja zahteva značajnu potrošnju toplotne energije, pa kao takav ostavlja prostora za mnoga istraživanja na polju poboljšanja energetske efikasnosti procesa. Zadržavanje kvaliteta osušenog proizvoda visokom nivou, uz što manju potrošnju energije, predstavlja jedan od najvećih izazova za stručnjake iz ove oblasti

Kao jedan od najčešće uzgajanih prehrambenih proizvoda i u svetu i kod nas smatraju se pečurke. Spadaju u grupu jestivih gljiva, sa preko 2000 vrsta koje danas postoje, od kojih su 22 intenzivno uzgajane u komercijalne svrhe, širom sveta [1]. U našoj zemlji je zastupljeno uzgajanje i korišćenje gljiva, i to uglavnom sledećih vrsta:

- Šampinjoni (*Agraricus bispourus*)
- Bukovače (*Pleurotus plumonarius*)
- Šitaki pečurke (*Lentinula edodes*)

Pečurke spadaju u red proizvoda sa visokim sadržajem vlage, što ih čini izuzetno osetljivim na uslove u kojima se čuvaju i transportuju. Vlažnosti svežih pečurki može

znatno da varira i zavisi od mnogobrojnih faktora, od uslova u kojima su odgajani, preko klimatskih uslova podneblja na kojima se gaje ili rastu u prirodi, do perioda godine u kome su sveže pečurke ubrane. Prema [1], vlažnost svežih pečurki varira u opsegu od 87% - 95% po vlažnoj osnovi, što ih čini proizvodima lako podložnim kvarenju.

2. PRORAČUN KOMORNE SUŠARE

Proračunom odabrane komorne sušare obuhvaćen je materijalni bilans sušare i termički proračun sušare. Cilj ovog proračuna je određivanje radnih parametara procesa sušenja, usvajanje potrebne opreme i određivanje odgovarajuće konstruktivne izvedbe sušare.

2.1 Materijalni bilans sušare

Materijalnim bilansom sušare određuju se količine osušenog materijala i izdvojene vlage, na osnovu zadatog kapaciteta sušare, na bazi vlažnog materijala. Sušara se projektuje za sušenje 100 kg pečurki po jednom punjenju. Usvojena početna vlažnost pečurki je $w_1 = 90\%$ (po vlažnoj osnovi) i proces sušenja traje do dostizanja krajnje vlažnosti osušenog materijala od $w_2 = 10\%$ (po vlažnoj osnovi). Materijalni bilans sušare po ukupnoj masi i po količini vlage je definisan jednačinama:

$$G_1 \cdot w_1 = W + G_2 \cdot w_2 \quad (1)$$

$$G_1 \cdot w_1 = W + (G_1 - W) \cdot w_2 \quad (2)$$

U prikazanom slučaju, od 100 kg svežih pečurki, izdvojena količina vlage je $W = 88,9$ kg, dok je dobijena količina osušenih pečurki iz jedne šarže $G_2 = 11,1$ kg.

2.2 Termički proračun sušare

U okviru termičkog proračuna sušare određena je potrebna količina toplote koju je potrebno dovesti u procesu sušenja. Planirani period rada sušare u toku godine je april - septembar, zbog povoljnih vremenskih uslova Predviđeno je sušenje pečurki u tankom sloju, poprečnim nastrujavanjem zagrejanog vazduha preko vlažnog materijala postavljenog na lese unutar komore za sušenje. Prema preporukama iz [2], odabrana temperatura vazduha za sušenje pečurki je $t_1 = 60^\circ\text{C}$ na ulazu u sušaru, i $t_2 = 40^\circ\text{C}$ na izlazu iz sušare, sa brzinom strujanja od 1 m/s. Sistem je projektovan prema najnepovoljnijim uslovima u odabranom periodu, najnižoj jutarnjoj temperaturi vazduha za mesec april od $8,3^\circ\text{C}$ i relativnoj vlažnosti od 82 % [3]. Potrebna količina zagrejanog vazduha za sušenje zadate količine pečurki, na osnovu usvojenih parametara sušenja se, prema [4], izračunava preko jednačine

$$L = \frac{W}{x_2 - x_1} \quad (3)$$

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Damir Đaković, vanr. prof.

i iznosi $L = 12.054 \text{ kg}_{\text{sv}}$, gde su $x_1 = 0,0057 \text{ kg}_w/\text{kg}_{\text{sv}}$ i $x_2 = 0,013 \text{ kg}_w/\text{kg}_{\text{sv}}$ apsolutne vlažnosti vazduha na ulazu i izlazu iz sušare [4,5]. Potrebna količina toplote u procesu sušenja, prema [4], je

$$Q = Q_i + Q_m + Q_o + Q_2 + Q_d + Q_t \pm Q_x \quad (4)$$

gde su:

- $Q_i = 225.446 \text{ kJ}$ - količina toplote potrebna za isparavanje vlage iz materijala
- $Q_m = 266,4 \text{ kJ}$ - količina toplote potrebna za zagrevanje materijala
- $Q_o = 24828 \text{ kJ}$ - količina toplote koja se predaje okolini
- $Q_2 = 402.974 \text{ kJ}$ - količina toplote dovedena agensu sušenja
- $Q_d = 0 \text{ kJ}$ - količina toplote potrebna za izdvajanje hemijski vezane vlage
- $Q_t = 0 \text{ kJ}$ - količina toplote za zagrevanje naknadno unetih masa i transportnih uređaja.

3. KINETIKA SUŠENJA

U okviru kinetike procesa sušenja analizirana je zavisnost procesa sušenja od vremena. Određivanje vremena trajanja procesa sušenja predstavlja veoma kompleksan problem, jer se vreme sušenja razlikuje od materijala do materijala. Najpotpunija slika odvijanja procesa sušenja se dobija analiziranjem krive sušenja, koja daje zavisnost srednje vlažnosti materijala od vremena trajanja procesa sušenja i krive brzine sušenja, koja daje zavisnost između brzine sušenja i vlažnosti materijala. Procena vremena trajanja sušenja za konkretan primer sušenja pečurki, odabran u okviru ovog rada, izvršeno je na osnovu eksperimentalno dobijenih podataka iz dostupne literature. Prema [2], kriva sušenja pečurki, konkretno šampinjona, za sušenje u tankom sloju pri temperaturi vazduha 60°C i brzini strujanja 1 m/s , može se opisati logaritamskim matematičkim modelom

$$MR = a \cdot \exp(-k \cdot \tau) + c \quad (5)$$

gde je MR sadržaj vlage (moisture ratio), τ vreme sušenja u časovima, i $a = 1,052187$, $k = 0,23897$, $c = -0,06599$, eksperimentalne konstante. Vrednost MR se određuje jednačinom

$$MR = \frac{M - M_e}{M_0 - M_e} \quad (6)$$

gde je M vlažnost materijala po suvoj osnovi u bilo kom trenutku sušenja, M_0 početna vlažnost materijala po suvoj osnovi i M_e ravnotežna vlažnost. Za početnu vlažnost pečurki 90% i krajnju 10% , sa temperaturom vazduha 60°C i brzinom strujanja vazduha 1 m/s , vreme potrebno za sušenje odabrane količine pečurki je 11 h . Specifična potrošnja vazduha po kilogramu isparene vlage se izračunava iz jednačine

$$l = \frac{1}{x_2 - x_1} \quad (7)$$

i iznosi $137,1 \text{ kg}_{\text{sv}}/\text{kg}_w$. Na osnovu maksimalne potrebne količine vazduha u prvom satu sušenja od $0,9 \text{ kg/s}$, dimenzionisan je odgovarajući razmenjivač toplote za zagrevanje vazduha i odgovarajući ventilator.

4. ANALIZA RADA SISTEMA ZA ZAGREVANJE VAZDUHA

Zagrevanje vazduha se vrši cevnim razmenjivačem toplote sa orebrenim bakarnim cevima, Proračun potrebne toplotne snage razmenjivača toplote izvršen je za temperaturu vazduha u mesecu aprilu u 7 h pre podne ($8,3^\circ\text{C}$ [3]) i temperaturu zagrevanja vazduha 60°C . Potrebna toplotna snaga razmenjivača toplote je određena jednačinom

$$\dot{Q} = \dot{L} \cdot c_{pv} \cdot (t_2 - t_1) \quad (8)$$

gde je:

- $\dot{Q} = 46,7 \text{ kW}$ - toplotna snaga razmenjivača toplote
- $\dot{L} = 0,9 \text{ kg/s}$ - potrebna količina vazduha za sušenje
- $c_{pv} = 1,005 \text{ kJ}/(\text{kg} \cdot \text{K})$ - specifični toplotni kapacitet vazduha
- $t_2 = 60^\circ\text{C}$ - temperatura zagrejanog vazduha
- $t_1 = 8,3^\circ\text{C}$ - temperatura okolnog vazduha

Toplotna snaga razmenjivača toplote iznosi $46,7 \text{ kW}$. Zagrevanje vazduha se vrši vodom temperature na ulazu u razmenjivač od 80°C i na izlazu 65°C . Za zagrevanje vode će biti usvojen odgovarajući sistem, čija će analiza rada biti data u nastavku. Na osnovu potrebnog protoka vazduha, usvojen je zidni aksijalni ventilator proizvođača S&P, serije COMPACT, tip HCBT/2-315/G.

4.1 Definisane periode rada sušare

Odabrana komorna sušara je predviđena za dnevni rad u periodu $7 \text{ h} - 18 \text{ h}$, zbog mogućnosti korišćenja sunčeve energije. Kako je potrebna toplotna snaga od $46,7 \text{ kWh}$ neophodna samo u prvom satu sušenja, kako proces odmiče, sa rastom temperature okoline smanjuje se i potrebna količina toplote za zagrevanje vazduha. Ukupna potrebna količina toplote za zagrevanje vazduha u toku sušenja jedne šarže pečurki data je u tabeli 1.

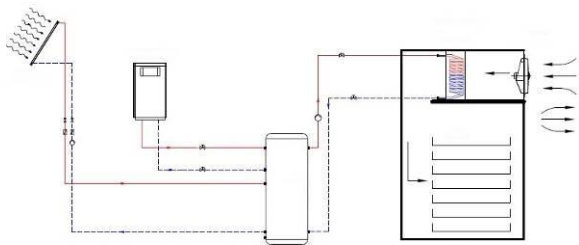
Tabela 1. Časovno toplotno opterećenje sistema za zagrevanje vazduha

Vreme [h]	Prosečna temperatura vazduha [$^\circ\text{C}$]	Potrebna toplotna energija [kWh]
7	8,3	46,76
8	12,5	32,05
9	14,1	24,35
10	15,5	18,53
11	16,3	14,29
12	17,0	11,14
13	17,4	8,69
14	17,6	6,85
15	17,2	5,45
16	16,5	4,25
17	15,5	3,02
Ukupno		175,4

Za usvojeni period rada sušare, sistem za zagrevanje vazduha mora biti dimenzionisan tako da obezbedi $175,4 \text{ kWh}$ toplotne energije za zagrevanje vazduha u toku jedne šarže sušenja. U nastavku će biti dati rezultati analize rada više različitih sistema za zagrevanje vazduha

4.2 Sistem za zagrevanje vazduha sa gasnim kondenzacionim kotlom solarnim kolektorima

Sistem se sastoji od gasnog kondenzacionog kotla, snage 51,5 kW, 20 solarnih kolektora i akumulacionog rezervoara od 2000 l. Simulacija rada sistema je izvršena u softveru „TSol”, u kom je prethodno definisan navedeni sistem, za klimatske uslove na području Novog Sada na mesečnom nivou za april. Na slici 1. je prikazana konfiguracija odabranog sistema i raspored opreme.



Slika 1. Sistem zagrevanja vazduha sa gasnim kondenzacionim kotlom i solarnim kolektorima

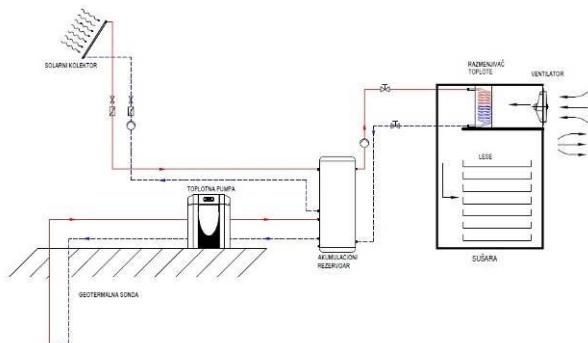
Simulacijom rada navedenog sistema, maksimalna dnevna potrošnja prirodnog gasa u toku meseca je 28 m³, odnosno 267 kWh. Za vrednost donje toplotne moći prirodnog gasa od 34.432 kJ/m³ i cenu prirodnog gasa 38,736 RSD/m³ [7], troškovi goriva po šarži osušenog materijala iznose 1084,6 RSD/šarži. Veličina prema kojoj će se upoređivati efikasnost rada prikazanih sistema je SMER (specific moisture evaporation rate), koji predstavlja odnos količine isparene vlage W u procesu sušenja i potrošnje energije u sistemu E.

$$SMER = \frac{W}{E} \left[\frac{kg_w}{kWh} \right] \quad (9)$$

Za prikazani sistem SMER iznosi 0,33 kg_w/kWh.

4.3 Sistem za zagrevanje vazduha sa toplotnom pumpom zemlja - voda i solarnim kolektorima

Sistem se sastoji od istih elemenata kao i prethodni, sa razlikom što je umesto gasnog kotla osnovni izvor toplote toplotna pumpa zemlja - voda snage 46 kW sa COP = 4,5. Kod ove vrste toplotne pumpe vrednost COP je konstantna, zbog konstantne temperature zemlje na većim dubinama, koja predstavlja izvor toplote za rad toplotne pumpe. Simulacija rada sistema je takođe izvršena u softveru „TSol”. Na slici 2. je dat šematski prikaz navedenog sistema.



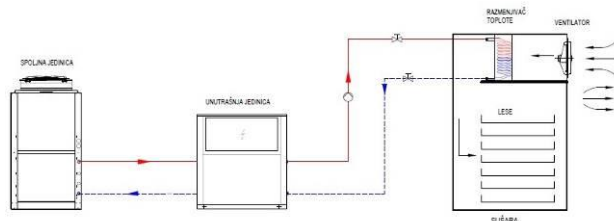
Slika 2. Sistem za zagrevanje vazduha sa toplotnom pumpom zemlja - voda i solarnim kolektorima

Prema dobijenim rezultatima simulacije rada u toku meseca aprila, ukupna potrošnja električne energije za rad

toplotne pumpe iznosi 1062 kWh, što daje prosečnu potrošnju električne energije po jednoj šarži sušenja od 35,4 kWh. Za cenu električne energije u zelenoj i plavoj zoni potrošnje u višoj tarifi od 5,962 RSD/kWh i 8,943 RSD/kWh prema [6], troškovi električne energije za sušenje jedne šarže pečurki iznose 396,2 RSD/šarži. Vrednost SMER za prikazani sistem je 2,51 kg_w/kWh.

4.4 Sistem za zagrevanje vazduha sa toplotnom pumpom vazduh - voda

U navedenom sistemu, za zagrevanje vazduha samostalno je primenjena toplotna pumpa vazduh - voda, iz razloga što ovaj tip toplotnih pumpi znatno zavisi od temperature okolnog vazduha. Zbog toga, za zagrevanje vode na temperature od preko 65 °C, neophodna je upotreba dvostepenih toplotnih pumpi većih snaga, koje su znatno skuplje od ostalih, pa bi cena ukupne investicije sa solarnim kolektorima bila jako visoka. Analizirana toplotna pumpa snage 60 kW ima promenljiv COP u odnosu na temperaturu okolnog vazduha, pa se za varijacije temperature u periodu sušenja koji je usvojen, kreće u opsegu 2,29 - 2,43. Na slici 3. je prikazana šema razmatranog sistema.



Slika 3. Sistem za zagrevanje vazduha sa toplotnom pumpom vazduh - voda

Za prikazani sistem, dnevna potrošnja električne energije za rad toplotne pumpe u toku meseca aprila, a u svrhu obezbeđivanja potrebne količine toplote za zagrevanje vazduha je određena iz odnosa potrebne količine toplote Q iz tabele 1. i vrednosti COP posmatrane toplotne pumpe

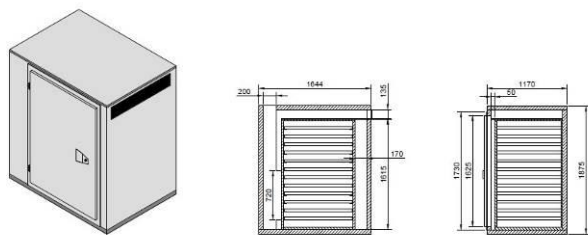
$$E_{el} = \frac{Q}{COP} \quad (10)$$

iznosi 73,42 kWh. Ukupna mesečna potrošnja električne energije za rad toplotne pumpe iznosi 2202 kWh, čime se ulazi u crvenu zonu potrošnje što iziskuje veće troškove rada. Za ranije navedene cene električne energije u zelenoj i plavoj zoni, i uz cenu električne energije u crvenoj zoni od 17,877 RSD/kWh [6], troškovi električne energije za sušenje jedne šarže pečurki iznose 1070,6 RSD/šarži. Vrednost SMER za posmatrani sistem iznosi 1,21 kg_w/kWh.

5. KONSTRUKCIJA KOMORNE SUŠARE

Konstrukcija komore za sušenje izvedena je od termoizolacionih panela sa spoljnim zidovima od nerđajućeg čelika (prohrom) čija je upotreba obavezna u prehrambenoj industriji, i ispunom od mineralne vune. Unutar komore za sušenje je na odgovarajući ram postavljeno 12 lesa za vlažni materijal dimenzija 1x1m, takođe izrađenih od nerđajućeg čelika, sa dnom od prohromske mreže. Strujanje vazduha u komori za sušenje ostvareno je u dva prolaza.

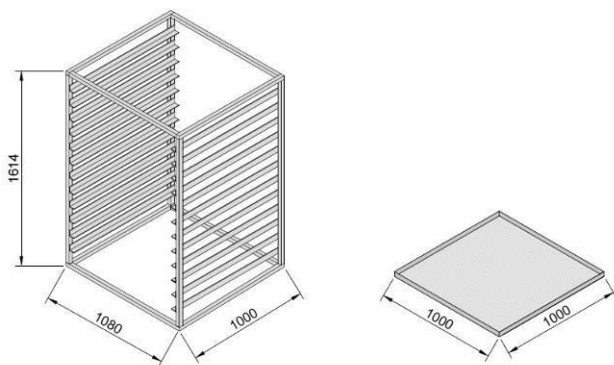
Kanal za dovod vazduha u kome su postavljeni razmenjivač toplote i ventilator se nalazi na konstrukciji komore za sušenje. Gabaritne dimenzije komore za sušenje su 1644 x 1170 x 1875 mm (dužina x širina x visina). Na slici 4. je prikazana komora za sušenje.



Slika 4. Komora za sušenje

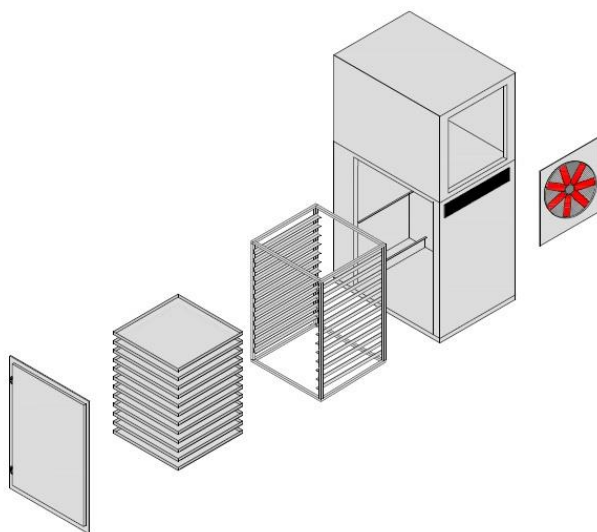
Lese sa vlažnim materijalom se postavljaju na ram prikazan na slici 5. preko odgovarajućih L profila.

Pošto se ram sa lesama nalazi u komori za sušenje, u dodiru je sa vlažnim vazduhom, pa je neophodno da bude od materijala koji nije podložan koroziji.



Slika 5. Dimenzije rama i lese

Na slici 6. prikazan je raspored osnovnih elemenata prikazane komorne sušare.



Slika 6. Osnovni elementi komorne sušare

6. ZAKLJUČAK

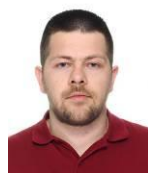
Iz prikazane analize sistema za zagrevanje vazduha, očigledno je da, sa energetske strane, primena toplotnih pumpi u odnosu na kotlove na konvencionalna goriva ima veliku prednost. Upotrebom odgovarajuće vrste toplotne pumpe, troškovi rada sušare u prikazanim sistemima se mogu umanjiti za oko 60 %. Rešenje sa toplotnom pumpom zemlja - voda i solarnim kolektorima se pokazalo kao najbolje i sa strane troškova rada, i sa strane vrednosti SMER, kao pokazatelja efikasnosti rada sušare. Toplotna pumpa vazduh - voda u samostalnom radu dala je lošije rezultate nego kombinacija toplotne pumpe zemlja - voda i solarni kolektori, ali je bila efikasnija od sistema sa gasnim kotlom i solarnim kolektorima. Takođe, upotreba električne energije umesto prirodnog gasa kao energenta, omogućava nezavisnost od mogućih povećanja cena prirodnog gasa.

Najveća prednost toplotnih pumpi se ogleda u tome što za svoj rad koriste niskotemperaturske obnovljive izvore energije, kojih ima u neograničenim količinama i mogu se primenjivati na bilo kojoj lokaciji. Iz svega navedenog, može se zaključiti da je primena toplotnih pumpi u procesima sušenja apsolutno opravdana, čak i pored relativno visokih investicionih ulaganja.

7. LITERATURA

- [1] Arora, S., Shivhare U. S., Ahmed, J., Raghavan, G. S. V., (2003), "Drying kinetics of *agraricus bisporus* and *pleurotus florida* mushrooms", Transactions of the ASAE, Vol 46(3), pp. 721-724
- [2] Stegou-Sagia, A., Fragkou, D. V, (2015), „Influence of drying conditions and mathematical models on the drying curves and the moisture diffusivity of mushrooms”, Journal of Thermal Engineering, Vol 1, No 4, pp. 235-244.
- [3] "Meteorološki godišnjak - klimatološki podaci", (2017), Beograd, Republički hidrometeorološki Zavod Republike Srbije.
- [4] Topić, R. (2014), "Sušenje i sušare", Beograd, SMEITS.
- [5] Topić, R. (1989), "Osnove projektovanja, proračuna i konstruisanja sušara", Beograd: IRO Naučna knjiga.
- [6] www.eps-snabdevanje.rs, (pristupljeno u martu 2019.)
- [7] www.srbijagas.com, (pristupljeno u martu 2019.)

Kratka biografija



Nikola Milivojević je rođen u Požarevcu 1993. god. Upisao osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu 2012. god, oblast Mašinstvo - Energetika i procesna tehnika. Trenutno zaposlen na Fakultetu tehničkih nauka kao saradnik u nastavi na Katedri za procesnu tehniku.



Damir Đaković je rođen 1975. godine u Sremskoj Mitrovici. Diplomirao, magistrirao i doktorirao na Fakultetu tehničkih nauka u Novom Sadu. Zainteresovan za mogućnosti praktične primene spregnutih problema prenosa toplote i mase u sušenju i drugim oblastima energetike i procesne tehnike.

PROJEKTOVANJE PREDNJEG DELA KAROSERIJE LAKOG HIBRIDNOG VOZILA**DESIGN OF LIGHT HYBRID VEHICLE'S FRONT END**Veljko Mladenović, *Fakultet tehničkih nauka, Novi Sad***Oblast – MAŠINSTVO**

Kratak sadržaj – U radu je opisan proces projektovanja prednjeg dela karoserije lakog hibridnog vozila (HEV Hermes), uz pomoć programa „CATIA P3 V5-6R2016“. Izrađena su dva različita modela, pridržavajući se ograničenja koja su zadata na samom početku, i upoređena su sa više različitih aspekata. Objasnjen je i proces izrade delova od plastike (branik), kao i sam programski paket koji je poslužio kao alat za projektovanje modela.

Ključne reči: Motorno vozilo, CATIA, dizajn, branik, plastični delovi

Abstract – This paper describes the process of designing the front part of the lightweight hybrid vehicle body (HEV Hermes), using CATIA P3 V5-6R2016. Two different models were created, adhering to the constraints imposed at the beginning, and compared with several different aspects. The process of making plastic parts (bumper) is explained, as well as the software package itself, which served as a tool for designing the model.

Ključne reči: Vehicle, CATIA, design, bumper, plastic parts

1. UVOD

U oblasti motornih vozila, kojom se bavi ovaj rad, vozilo je sredstvo koje je po konstrukciji, uređajima, sklopovima i opremi namenjeno i osposobljeno za kretanje po putu [1]. Cilj ovog rada je da na zadatoj, postojećoj, konstrukciji HEV HERMES, koja je razvijana i pravljen na Departmanu za mehanizaciju i konstrukciono mašinstvo FTN-a u Novom Sadu, projektovati i konstruisati prednji deo karoserije (branik).



Slika 1. HEV HERMES [2]

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji su mentori bili dr Dragan Ružić i dr Boris Stojić.

2. OGRANIČENJA PRILIKOM PROJEKTOVANJA PREDNJEG DELA KAROSERIJE AUTOMOBILA

Biće urađena dva modela pomoću programskog paketa „CATIA“ (modili: *part, shape, product design*), koji će na kraju biti međusobno upoređeni.

Prilikom projektovanja karoserije automobila, projekat od samog starta prolazi kroz veći broj faza razvoja. Pojedine faze mogu trajati i više meseci, a do same proizvodnje preko raznih ispitivanja i testiranja može proći i par godina.

Prednji deo karoserije automobila čine svi spoljašnji delovi kao što su: branik, mreže za ulaz vazduha za hlađenje, farovi, i drugi delovi koji zavise od proizvođača i vrste modela.

Neke od osnovnih funkcija branika i neka druga ograničenja na koja se mora obratiti pažnja prilikom projektovanja su:

- Poboljšavanje aerodinamike vozila,
- Sprečavanje nastanka štete pri sudarima manje brzine,
- Absorbovanje dela sile prilikom sudara sa pešacima,
- Smanjenje težine vozila korišćenjem odgovarajućih materijala,
- Povećanje atraktivnosti vozila,
- Tačke veze (branik-glavna konstrukcija, ostali delovi sa branikom),
- Prostor za elemente oslanjanja i druge komponente (raspored u prostoru),
- Pristup radi održavanja,
- Tehnologija izrade branika,
- Reciklaža...

2.1. Tehnologije izrade branika

Postoji nekoliko vrsta branika, a danas su najzastupljeniji:

- Plastični branici – najviše zastupljeni na modernim automobilima, a odlikuje ih jeftina izrada (serijska i masovna proizvodnja), i velika absorpcija energije pri udaru,
- Karbonski branici – veoma su laki i čvrsti, ali zbog svoje visoke cene koriste ih najviše sportska i vozila više klase,
- Hibridni branici – kombinacija više materijala. Uglavnom za maloserijsku i pojedinačnu proizvodnju.

Procesi izrade branika:

- Brizganjem plastike,
- 3D štampači
- Kompozitni materijali (fiberglass, carbon fiber)

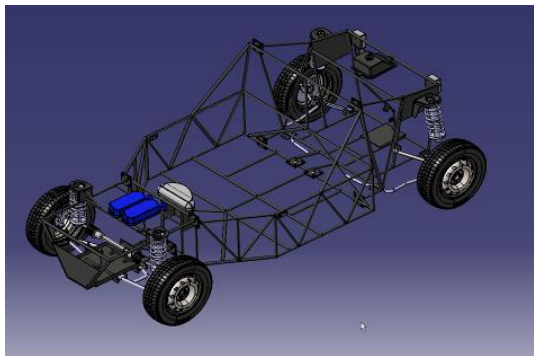
3. PROJEKTOVANJE MODELA POMOĆU SOFTVERSKOG PAKETA „CATIA”

CATIA (*Computer Aided Three-Dimensional Interactive Application*) je računarski program firme *Dassault Systemes* za projektovanje i tehničko konstruisanje. Ovaj program može poslužiti i za dizajniranje, simulacije, analizu i proizvodnju u različitim industrijama. Iako je prvobitno bio namenjen za konstruisanje aviona, vrlo brzo je našao primenu u oblastima kao što je automobilska industrija (*BMW, Audi, Daimler AG, VW,*), brodogradnja, industrijske mašine.

- **PART DESIGN** - predstavlja modul koji koristimo za konstruisanje preciznih pojedinačnih 3D modela koje posle sklapamo u Product-u. Part Design nam omogućava konstruisanje od jednostavnijih do veoma komplikovanih delova.
- **GENERATIVE SHAPE DESIGN** - je modul koji omogućava modelovanje delova jednostavne i veoma komplikovane nepravilne geometrije. *Shape Design* nam omogućava parametrijsku izradu delova i brze izmene postojećih delova. *Shape Design* se koristi u kombinaciji sa *Part Design*-om. U *shape design* se prave površine na osnovu kojih dobijamo gotove 3D modele dodavanjem materijala u *Part Design*-u.
- **PRODUCT/ASSEMBLY DESIGN**- predstavlja modul u koji ubacujemo gotove 3D elemente koje smo konstruisali predhodno ili ih generisali iz baze podataka *Catie*. *Produkt* nam omogućava sklapanje kompletnih proizvoda i mogućnost analize finalnog proizvoda.

3.1. Koncept 1.

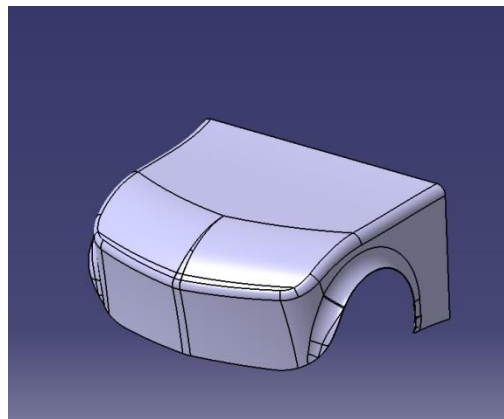
Na samom startu uzimamo početni model od koga krećemo (Slika 2.)...



Slika 2. Prikaz osnovne (početne) konstrukcije

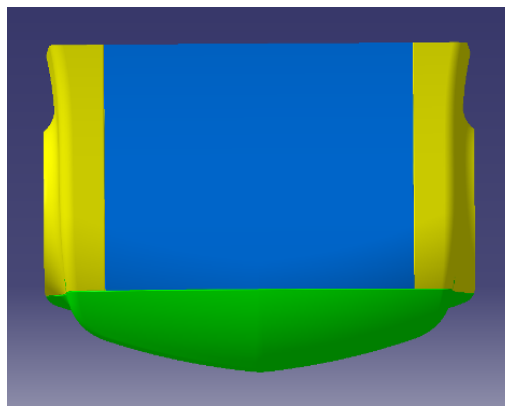
Zadatak je da na prednjem delu izmodeliramo branik. Rad na prednjem delu ćemo podeliti u nekoliko faza:

- **Faza 1.** (rad sa površinama u *Shape Design*-u-početna faza) – U ovoj početnoj fazi postavljamo tačke i Plane-ove (referentne), u odnosu na apsolutni koordinatni sistem. Dalje pomoću opcije *Sketch* crtamo odgovarajuće profile od kojih kasnije pomoću opcija *Sweep*, *Extrude*, *Split*, *Trim*, dobijamo željeni izgled.
- **Faza 2.** (rad u *Part Design*-u sa debljinom (Body)), - U fazi dva prelazimo u drugo okruženje i pomoću opcija *Mirror*, *Chamfer* i *Fillet* ulepšavamo izgled prednjeg dela. Nakon toga pomoću opcije *Shell* dobijamo debljinu dela 3 [mm] (Slika 3.).



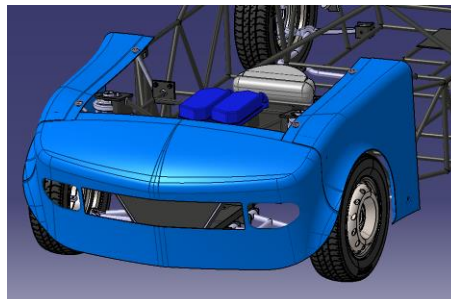
Slika 3. Part Design – Body (debljina 3 [mm])

- **Faza 3.** (Podela prednjeg dela na više segmenata) – U fazi tri ćemo podeliti prednji deo na više segmenata zbog lakše izrade i montaže, i posle toga uraditi otvore na prednjem delu za svetla i mrežice za vazduh (Slika 4.).



Slika 4. Podela prednjeg dela na više delova

- **Faza 4.** (Rad na kačenju - veza delova sa konstrukcijom i između sebe) – U ovoj fazi treba osmisliti i dati predlog kačenja-veze delova između sebe i sa konstrukcijom. Pošto imamo ograničenje i zahtev da to bude što jednostavnije, odstupamo od nekih standardnih rešenja pomoću raznih vrsta klipova koji su dosta komplikovani za izradu. Prvo je potrebno napraviti odgovarajuće držače na konstrukciji, a nakon toga ćemo pomoću jednostavnih kopči i vijaka spojiti delove (Slika 5.).



Slika 5. Montirani delovi sa predloženim kačenjem

- **Faza 5.** (dodatni element – farovi, pokazivači pravca, mreža i logo) – Na kraji dodajemo svetla i pokazivače pravca koji su objedinjeni u jednom kućištu, kao i mrežu na srednjem delu sa logom (Slika 6.).



Slika 6. Prikaz završenog prednjeg dela

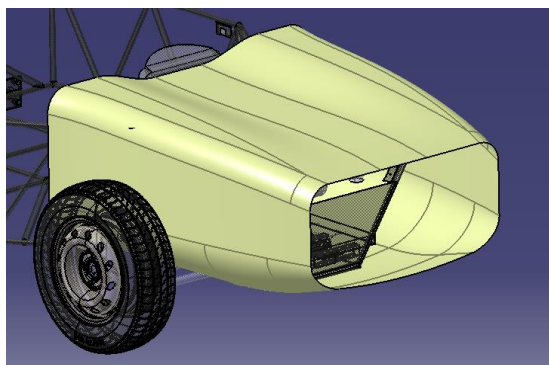
- **Faza 6.** (Provera i izračunavanje parametara) – U poslednjoj fazi, u istom programu, izvršićemo proračun težine dela (prvo zadamo odgovarajući materijal-plastika:gustina $0,857 \text{ [g/cm}^3\text{]}$) - $m=7,1 \text{ [kg]}$. Pomoću programa *Flow design*, firme *Autodesk*, uradićemo simulaciju, (za brzinu od 60 [km/h] ($16,6 \text{ [m/s]}$)), aerodinamike i merenje koeficijenta otpora vazduha - $C_w=0,52$.

Napomena: Ove vrednosti dobijene simulacijom treba uzeti sa rezervom pošto ne predstavljaju realno stanje, konstrukcija nije kompletna, pa se ne mogu koristiti za proračune.

3.2. Koncept 2.

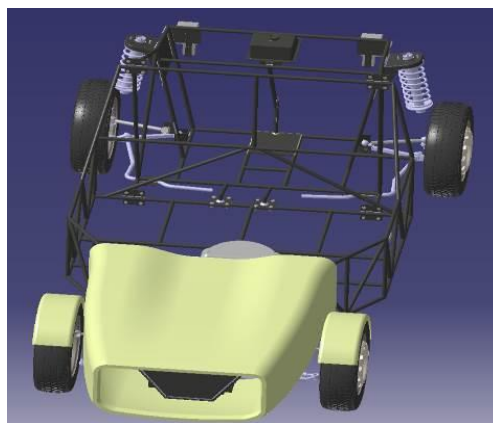
Krećemo od početnog modela... (Slika 2.).

- **Faza 1., Faza 2.** (rad sa površinama u *Shape Design*-u-početne faze) – Isto kao kod prvog modela, u *Shape Design*-u, postavljamo referentnu tačku i u odnosu na nju u odgovarajućim ravnima koordinatnog sistema postavljamo *Plane*-ove. Dalje pomoću opcije *Sketch* crtamo odgovarajuće profile od kojih kasnije pomoću opcija *Sweep*, *Extrude*, *Split*, *Trim*, *Shape Fillet*, dobijamo željeni izgled. (Slika 7.).



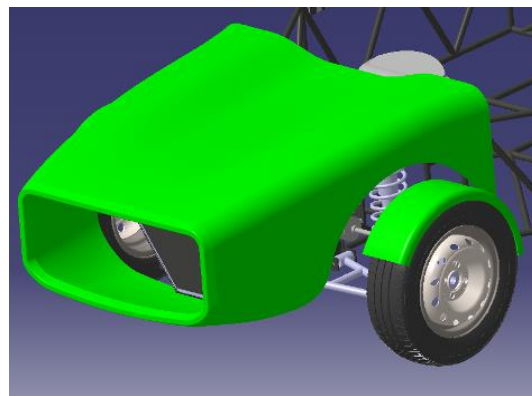
Slika 7. Izgled zatvorenog profila prednjeg dela

- **Faza 3., Faza 4.** (Rad sa površinama u *Shape Design*-u – prednji deo i blatobrani) – U ovoj fazi crtamo profil prednjeg dela pomoću opcije *Sketch*, pozicionirajući se u odgovarajućoj ravni. Zatim pomoću opcije *Sweep* i *Shape fillet* dobijamo oblik prednjeg dela. Iznad točkova dodajemo blatobrane koji su propisani pravilnikom (Slika 8.).



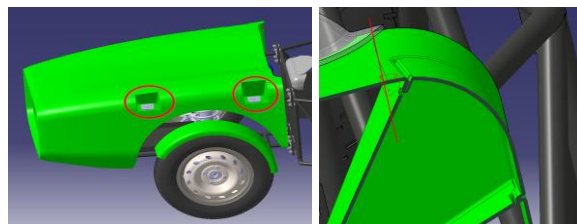
Slika 8. Izgled zatvorenog profila prednjeg dela sa blatobranima

- **Faza 5.** (rad u *Part Design*-u sa debljinom (*Body*)) - Prelazimo u drugo okruženje i pomoću opcije *Close surface* dobijamo *Body* i dodeljujemo materijal. Sređujemo prednji deo oko točkova da bi se mogli zakretati, pošto su blatobrani predviđeni da se zakreću zajedno sa točkovima (Slika 9).

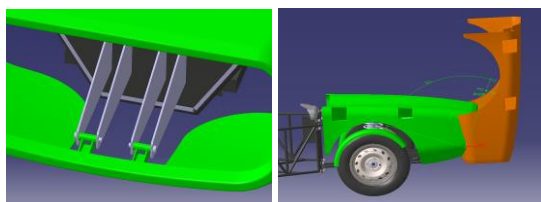


Slika 9. Izgled prednjeg dela sa prostorom za zakretanje točkova

- **Faza 6.** (rad u *Part Design*-u - veza delova) - Tačke veze-kačenja će biti što jednostavnije urađene, biće u skladu sa zahtevom da se što lakše pristupi prednjem delu radi održavanja. Biće urađene tako da omoguće da se ceo prednji deo podiže napred i oslobodi prostor za pristup prednjem delu konstrukcije. Na glavnu konstrukciju ćemo dodati profile na kojima su vijci koji čine jedan deo. Prilikom spuštanja prednjeg kraja-haube, ona će se u krajnjem položaju, preko ojačanja napravljenih za vezu, spustiti na vijke. Dodatnim leptir navrtkama (2 sa jedne, 2 sa druge strane-simetrično postavljene) i elastičnim podloškama biće učvršćen gornji deo i osiguran od pomeranja. Kada se fiksira gornji deo dodajemo poklopce koji će sakriti mesta kačenja (Slika 10.).

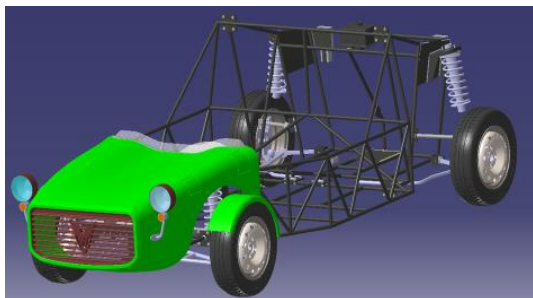


Slika 10. Izgled kačenja sa gornje strane i rešenje poklopca za skrivanje vijaka



Slika 11. Rotaciona veza na prednjem kraju i prikaz podignute haube

- **Faza 7.** (rad u *Part Design*-u - rad na dodatnim delovima) - Na kraju nam ostaje da dodamo preostale delove kao što su: svetla, maska sa znakom i zaštita za vozača. Prednja signalizacija je urađena po standardima (Slika 12.).



Slika 11. Izgled prednjeg dela

Kao i na prvom modelu i ovde ćemo dodeliti materijal i izmeriti težinu. Ako bude izrađen od plastike (gustina $0,857 \text{ [g/cm}^3\text{]}$), težina je $m=7,4 \text{ [kg]}$.

Kao i kod prvog modela urađena je simulacija otpora vazduha na isti način - $C_w=0,42$.

Nakon izrade delova i njihovog testiranja i provere u pogledu krutosti, biće potrebno za svaki deo uraditi ojačanja u vidu dodatnih, poprečnih i/ili uzdužnih rebara. To će omogućiti da delovi budu bolji u pogledu krutosti i postojaniji usled spoljnih uticaja.

4. POREĐENJE MODELA

Ova dva modela (Koncept 1., Koncept 2.), upoređićemo sa više različitih aspekata (aerodinamika, tačke veze, tehnologija izrade, težina, pristup radi održavanja...).

Sa strane tačaka veze, prvo je trebalo ispuniti uslov da bude što jednostavnija veza između dela koji pravimo i konstrukcije. Pošto je na samom početku rečeno da treba ići u pravcu pojedinačne proizvodnje, odmah smo zaobišli skup i komplikovan sistem veze. U ovoj fazi treba napraviti prototipe branika-prednjih delova koje smo osmislili i ispitati tačke veze i njihovu krutost i izdržljivost.

Tehnologija izrade može bilo koja od navedenih. Gledajući sa ekonomske strane, za pojedinačnu proizvodnju pravljenje kalupa za brizganje plastike bi bilo skupo i neisplativo. Treba gledati u pravcu kompozitnih materijala i od njih izraditi delove.

Još jedan bitan zahtev i ograničavajući element pre samog konstruisanja bio je omogućiti funkcionalnost svih delova a samim tim obezbediti što lakši pristup radi održavanja. Kod oba koncepta skida se nekoliko navrtki koje omogućavaju skidanje ili podizanje određenog dela i dobija se pristup prostoru za održavanje. Kod prvog koncepta skida se samo poklopac prednjeg dela i nešto je manji pristup delu za održavanje. Kod drugog koncepta, ceo prednji deo

je predviđen da se uradi iz jednog dela, pa tako podizanjem i zakretanjem oko odgovarajuće ose, podize se ceo prednji deo i omogućava u potpunosti pristup.

Treba napomenuti da Koncept 2. nije u potpunosti urađen u skladu sa propisima. Prednji deo mora imati napred branik kao poseban deo, što nije slučaj sa našim drugim modelom jer je predviđen da se uradi iz jednog dela po uzoru na trkačka vozila.

5. ZAKLJUČAK

U ovom radu je prikazan kompletan postupak izrade prednjeg dela (branika), za HEV HERMES. Kao što vidimo od samog početka, od ideje, pa preko skica i izrade modela pomoću programskog paketa „CATIA“, zadato je mnogo ograničenja sa različitih aspekata koje je bilo potrebno ispoštovati.

Prednost prvog modela (Koncept 1.), je mogućnost lakše izrade. Izrađuje se više jednostavnijih delova, jednostavniji kalupi, samim tim i jeftinija proizvodnja. Mana ovog modela, pored malo lošijeg koeficijenta otpora vazduha (koji nisu dovoljno relevantni), je što se lakim skidanjem poklopca, radi održavanja, dobija pristup samo odozgo. Drugi model (Koncept 2.), je na neki način suprotnost prvom. On se izrađuje iz jednog dela (sa dodatkom blatobrana koji su posebni delovi), pa je tako i izrada komplikovanija i skuplja. Prednost je što se podizanjem prednjeg dela dobija pristup celom prednjem kraju.

6. LITERATURA

- [1] Ružić D.: *Motorna vozila – skripta*, Fakultet tehničkih nauka, Novi Sad, 2017.
- [2] www.mehanizacija.ftn.uns.ac.rs, pristupljeno 31.10.2018.
- [3] Stojić, B.: *Teorija kretanja drumskih vozila*, Fakultet tehničkih nauka, Novi Sad, 2012.
- [4] Brzaković R., Marjanović, Z.: *Reciklaža kao segment životnog ciklusa vozila*, I nacionalna konferencija o reciklaži motornih vozila, Ečka, 2009.
- [5] Happian-Smith J.: *An Introduction to Modern Vehicle Design*, Oxford, 2002.
- [6] IRJET- Vishwanatha R. H., Ajith A., Anand N., Punith Raju K. B., Vasanth K.: *Design and Strength Validation of Front Car Bumper Using Composite Material*, Volume: 05 Issue: 05, India, 2018.
- [7] *Advanced in Mechanical Engineering*-Hu Y., Liu C., Zhang J., Ding G., Wu Q.: *Research on carbon fiber-reinforced plastic bumper beam subjected to low-velocity frontal impact*, Vol. 7(6) 1-15. China, 2015.

Kratka biografija:



Veljko Mladenović, rođen je u Ivanjici 1986. godine. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Mašinstva odbranio 2015. godine.

**ISTRAŽIVANJE UTICAJA LOKALNOG OTPORA POSTAVLJENOG ISPRED MERNE
BLENDE NA STRUJNE PARAMETRE****RESEARCHING OF INFLUENCE OF LOCAL FLOW RESISTANCE PLACED IN THE
FRONT OF ORIFICE METER ON FLOW PARAMETERS**Marko Narić, *Fakultet tehničkih nauka, Novi Sad***Oblast - MAŠINSTVO**

Kratak sadržaj - Predmet rada bila je merna blenda, uređaj za merenje protoka fluida. Cilj rada bio je da se istraži uticaj lokalnog otpora postavljenog ispred merne blende na sledeće strujne parametre: razliku pritiska, pad pritiska, gubitak strujne snage i koeficijent protoka. Razmatrana je merna blenda sa jednim (SHO) i merna blenda sa više otvora (MHO). Analiza je urađena primenom računarske dinamike fluida. Zaključeno je da lokalni otpor postavljen ispred merne blende utiče na navedene strujne parametre.

Ključne reči: Merna blenda, vazduh, lokalni otpor, merenje protoka.

Abstract: - The subject of this thesis is orifice meter, device for fluid flow rate measurement. The aim was to analyse and determine influence of local flow resistance placed in front of orifice meter on the flow parameters: pressure drop, pressure loss, fluid energy loss and discharge coefficient. It's considered orifice meter with single (SHO) and orifice meter with multiple fluid hole openings (MHO). Analysis was done by using computational fluid dynamics (CFD). It was concluded that local flow resistance placed in the front of the orifice meter have influence on listed fluid parameters.

Key words: orifice meter, air, local flow resistance, fluid flow rate measurement.

1. UVOD

Predmet rada bila je merna blenda, uređaj koji predstavlja jedan od najzastupljenijih merila protoka zbog niza svojih prednosti. Merna blenda se postavlja u pravolinijski deo cevovoda i služi za stvaranje razlike strujnih pritisaka kojim se uz poznavanje stanja fluida i geometrijskih osobina cevovoda izračunava protok.

Cilj rada bio je da se istraži uticaj lokalnog otpora postavljenog ispred merne blende na sledeće strujne parametre: razliku pritiska, pad pritiska, gubitak strujne snage i koeficijent protoka. Razmatrane su dve merne blende:

- merna blenda sa jednim (SHO) i
- merna blenda sa više otvora (MHO).

NAPOMENA:

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

Obe razmatrane merne blende nalazile su se u cevovodu prečnika $D=70,3$ mm gde je odnos površina poprečnog preseka svetlog dela merne blende i cevi iznosio $\beta=0,55$. Dobijene vrednosti strujnih parametara su se upoređivale za različite položaje lokalnog otpora postavljenog ispred merne blende sa vrednostima dobijenim u slučaju kada se u cevovodu ne nalazi lokalni otpor. Vrednosti položaja lokalnog otpora postavljenog ispred merne blende iznosile su $5D$, $10D$, $15D$, $22D$ i $30D$. Kao lokalni otpor koristilo se koleno sa uglom od 90° i poluprečnikom $r=D=70,3$ mm. Istraživanje je urađeno pri parametrima strujanja suvog vazduha na temperaturi od 22°C , pritisku od 4 bar i brzinama strujanja na ulazu u cevovod od 5 m/s.

2. PREGLED STANJA U OBLASTI

Veliki broj autora bavi se računarskom dinamikom fluida gde je kao predmet istraživanja korišćena merna blenda. U poslednje vreme je čest predmet istraživanja velikih razvojnih i naučnih instituta.

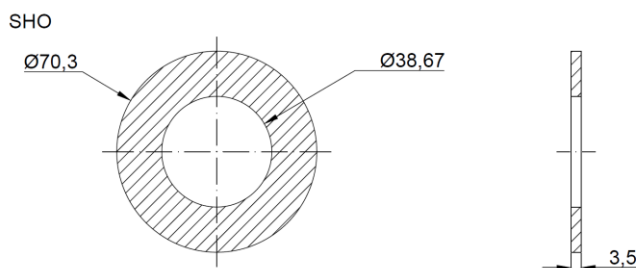
Istraživanjem povrata pritiska primenom računarske dinamike fluida ustanovljeno je da rezultati koji su dobijeni u korišćenim numeričkim simulacijama u radu [1] odstupaju oko 7 % od rezultata dobijenih eksperimentalnom metodom. Takođe, autori su ustanovili da merna blenda sa više otvora ima znatno viši koeficijent protoka u odnosu na mernu blendu sa jednim otvorom i manji pad pritiska.

Istraživanjem merne blende sa jednim otvorom primenom računarske dinamike fluida [2] detaljno je prikazan uticaj merne blende na turbulenciju strujnica prilikom strujanja fluida. Posebno je dat akcenat na analizu kinetičke energije.

Pored navedenih, postoje i mnogi drugi radovi čija je analiza merenje protoka mernom blendom primenom računarske dinamike fluida bila predmet interesovanja autora.

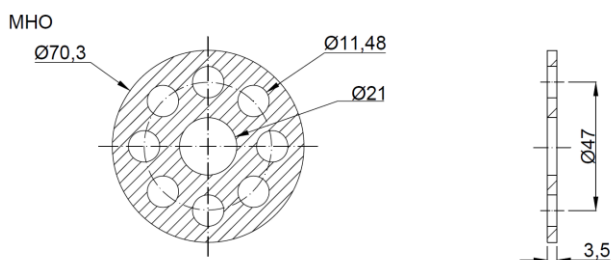
3. PLAN NUMERIČKIH SIMULACIJA**3.1. Geometrija mernih blendi i cevovoda**

Posmatrao se protok vazduha kroz dve vrste mernih blendi različitih konstrukcija i dimenzija, ali istim odnosom površina poprečnog preseka svetlih otvora merne blende i cevi β . Prva blenda SHO je sa jednim, standardnim otvorom čiji je prečnik 38,67 mm, koncentrično postavljen u odnosu na osu cevi, slika 1.



Slika 1. Prikaz merne blende sa jednim otvorom

Druga blenda MHO je sa jednim, centralnim, koncentrično postavljenim otvorom prečnika 21 mm i sa osam perifernih otvora prečnika 11,48 mm pravilno raspoređenih na prečniku od 47 mm, slika 2.



Slika 2. Prikaz merne blende sa više otvora

Radi poređenja rezultata obe razmatrane merne blende imale su istu vrednost parametra β :

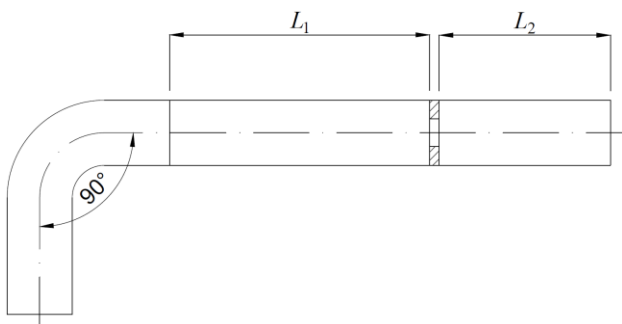
$$\beta = \sqrt{\frac{A_2}{A_1}} = 0,55,$$

gde su:

- β - odnos površina poprečnog preseka svetlih otvora merne blende i cevi [-];
- A_1 - površina poprečnog preseka svetlog dela cevi [m²] i
- A_2 - ukupna površina poprečnog preseka svetlih otvora merne blende [m²].

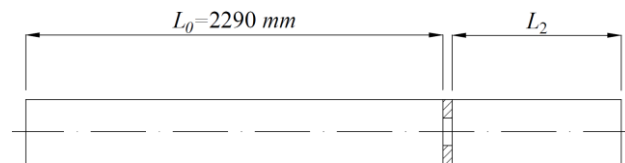
Kao jedino uprošćenje u geometriji merne blende smatralo se da nema zakošenih ivica.

Kako bi se istražio uticaj lokalnog otpora na strujne parametre bilo je potrebno definisati položaj kolena ispred merne blende. Koleno od 90° i prečnika $r=D=70,3$ mm bilo je postavljeno na rastojanju ispred merne blende L_1 u vrednosti $5D$, $10D$, $15D$, $22D$ i $30D$, slika 3.



Slika 3. Prikaz položaja lokalnog otpora L_1

Dobijene vrednosti strujnih parametara u slučaju kada je lokalni otpor postavljen ispred merne blende su se upoređivali sa vrednostima dobijenim u slučaju kada se u cevovodu ne nalazi lokalni otpor referentne deonice L_0 , slika 4.



Slika 4. Prikaz referentne deonice L_0

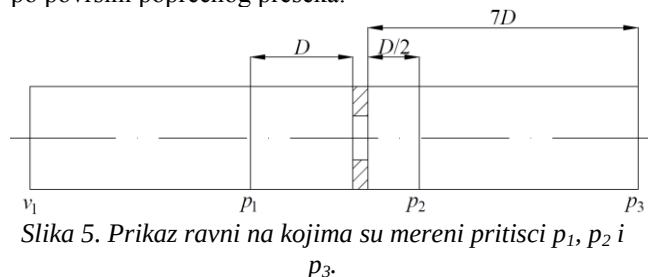
Vazduh masenog protoka $0,103$ kg/s, temperature 22 °C i pritiska 4 bar, strujao je kroz navedeni cevovod gde je u numeričkim simulacijama brzina strujanja na ulazu u cevovod iznosila 5 m/s.

3.2. Strujni parametri

Numeričkim simulacijama mogu da se dobiju vrednosti različitih osnovnih i izvedenih fizičkih veličina (brzine, pritiska, gustine, inteziteta vrtloženja, sile itd). Pažnja je bila usredsređena na sledeće strujne parametre:

- razliku pritisaka Δp_{1-2} koja nastaje na blendi;
- pad pritiska Δp_{1-3} ;
- gubitak strujne snage ΔP i
- koeficijent protoka μ .

Na slici 5, prikazane su ravni na kojima su mereni pritisci vazduha p_1 , p_2 i p_3 . Radi se o pritiscima koji su osrednjeni po površini poprečnog preseka.



Slika 5. Prikaz ravni na kojima su mereni pritisci p_1 , p_2 i

p_3 . Razlika pritiska nastaje usled suženja preseka na samoj blendi i određuje se kao:

$$\Delta p_{1-2} = p_1 - p_2,$$

gde su:

- Δp_{1-2} - razlika pritisaka na mernoj blendi [Pa];
- p_1 - pritisak ispred merne blende [Pa] i
- p_2 - pritisak iza merne blende [Pa].

Pad pritiska ima manju vrednost od razlike pritiska jer se deo kinetičke energije vratio u pritisnu energiju nakon blende, a određen je kao:

$$\Delta p_{1-3} = p_1 - p_3,$$

gde su:

- Δp_{1-3} - pad pritiska [Pa];
- p_1 - pritisak ispred merne blende [Pa] i
- p_3 - pritisak na kraju cevne deonice [Pa].

Gubitak strujne snage na mernoj blendi određen je kao:

$$\Delta P = Q \cdot \Delta p_{1-3},$$

gde su:

ΔP - gubitak strujne snage [W];
 Q - zapreminski protok vazduha [m³/s] i
 Δp_{1-3} - pad pritiska [Pa].

Uticaj položaja lokalnog otpora postavljenog ispred merne blende L_1 na gubitak strujne snage prikazan je preko relativnog gubitka strujne snage koji je određen kao:

$$\varepsilon_{\Delta P} = \frac{|(\Delta P)_{L_1} - (\Delta P)_{L_0}|}{(\Delta P)_{L_0}} \cdot 100,$$

gde su:

$\varepsilon_{\Delta P}$ - relativni gubitak strujne snage [%];
 $(\Delta P)_{L_1}$ - gubitak strujne snage pri položaju lokalnog otpora L_1 [W] i
 $(\Delta P)_{L_0}$ - gubitak strujne snage pri referentnoj deonici L_0 [W].

Koeficijent protoka predstavlja bezdimenzionu vrednost koji pokazuje koliki je stvaran protok u odnosu na teorijski i određen je kao:

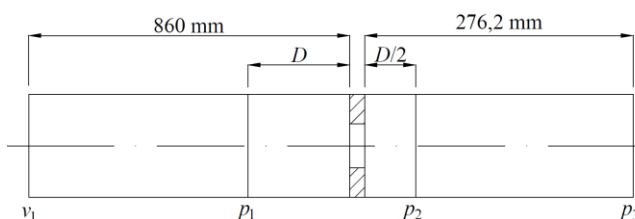
$$\mu = \frac{Q}{\sqrt{\rho \left[1 - \left(\frac{A_2}{A_1} \right)^2 \right]} \cdot A_2},$$

gde su:

μ - koeficijent protoka [-];
 Q - zapreminski protok vazduha [m³/s];
 Δp_{1-2} - razlika pritisaka na mernoj blendi [Pa];
 ρ - gustina vazduha [kg/m³];
 A_1 - površina poprečnog preseka svetlog dela cevi [m²] i
 A_2 - površina svetlih otvora merne blende [m²].

3.3. Verifikacija rezultata numeričkih simulacija sa eksperimentalnim rezultatima

Eksperimentalno merenje urađeno je u laboratoriji za ispitivanje gasnih uređaja koja se nalazi u okviru administrativnog i proizvodnog prostora u krugu kompanije **GasTeh d.o.o.** iz Indije. Ispitna instalacija koja se nalazi u laboratoriji gde je urađen eksperiment i na osnovu koga je napravljen numerički model prikazana je na slici 6.



Slika 6. Prikaz geometrije numeričkih simulacija prilikom verifikacije rezultata

Relativna razlika pritiska između numeričkih i eksperimentalnih rezultata određena je kao:

$$\varepsilon_{sim} = \frac{|(\Delta p_{1-2})_{sim} - (\Delta p_{1-2})_{exp}|}{(\Delta p_{1-2})_{exp}} \cdot 100,$$

gde su:

ε_{sim} - relativna razlika pritiska numeričkih simulacija [%];
 $(\Delta p_{1-2})_{sim}$ - razlika pritiska dobijena numeričkim simulacijama [Pa] i
 $(\Delta p_{1-2})_{exp}$ - razlika pritiska dobijena eksperimentalnom metodom [Pa].

Verifikacija rezultata je urađena pri pritisku 4 bar i temperaturi vazduha oko 27 °C. Na ulazu cevne deonice strujao je vazduh masenog protoka $\dot{m}_{exp}$ i brzini strujanja v_1 koji su prikazani u tabeli 1.

Tabela 1. Maseni protok $\dot{m}_{exp}$ i brzina strujanja vazduha v_1 tokom verifikacije rezultata

$(\dot{m}_{exp})_{SHO}$ [kg/s]	$(v_1)_{SHO}$ [m/s]	$(\dot{m}_{exp})_{MHO}$ [kg/s]	$(v_1)_{MHO}$ [m/s]
0,046	1,94	0,053	2,27
0,093	3,93	0,085	3,64
0,132	5,59	0,121	5,15
0,173	7,32	0,161	6,87
0,212	9,01	0,195	8,33
0,241	10,27	0,216	9,23

4. ANALIZA I DISKUSIJA

4.1. Verifikacija rezultata numeričkih simulacija

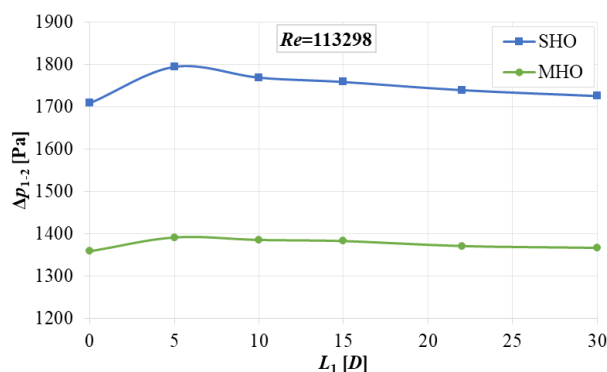
U tabeli 2, su prikazane vrednosti relativne razlike pritiska za mernu blendu sa jednim $(\varepsilon_{sim})_{SHO}$ i mernu blendu sa više otvora $(\varepsilon_{sim})_{MHO}$. Može da se zaključi da merna blenda sa jednim otvorom ima srednju vrednost relativne razlike pritiska od 2,42 % dok merna blenda sa više otvora ima srednju vrednost od 11,11 % u odnosu na rezultate dobijene eksperimentom.

Tabela 2. Relativna razlika pritiska numeričkih simulacija

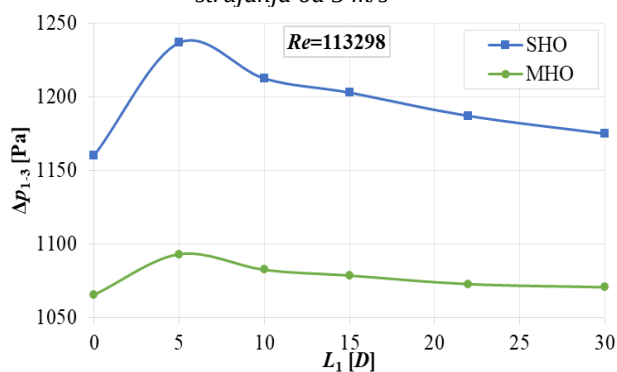
$(\varepsilon_{sim})_{SHO}$ [%]	$(\varepsilon_{sim})_{MHO}$ [%]
10,81	1,36
9,48	3,05
12,11	1,53
9,57	0,85
9,60	5,22
15,10	2,53

4.2. Strujni parametri merne blende

Na slikama 7 i 8, su grafički prikazani rezultati razlike i pada pritisaka $(\Delta p_{1-2}, \Delta p_{1-3})$ za mernu blendu sa jednim i mernu blendu sa više otvora u funkciji položaja lokalnog otpora L_1 ispred merne blende u cevovodu.



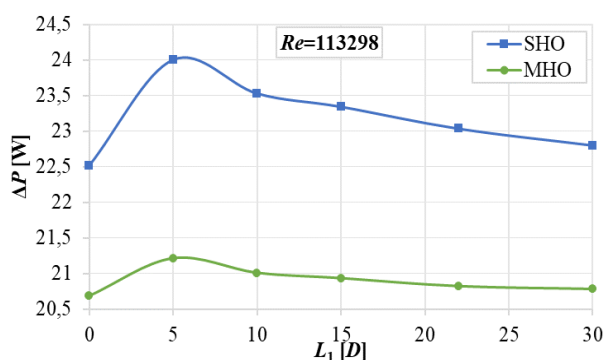
Slika 7. Razlika pritiska na pritisku 4 bar i pri brzini strujanja od 5 m/s



Slika 8. Pad pritiska na pritisku 4 bar i pri brzini strujanja od 5 m/s

Analizom dobijenih rezultata, uočeno je da potreban položaj lokalnog otpora L_1 ispred merne blende sa jednim otvorom kako bi se uticaj lokalnog otpora na razliku i pad pritiska mogao zanemariti iznosi $22D$, dok je kod merne blende sa više otvora $15D$.

Na slici 9, su grafički prikazani rezultati gubitka strujne snage za mernu blendu sa jednim i mernu blendu sa više otvora u funkciji položaja lokalnog otpora L_1 ispred merne blende u cevovodu.



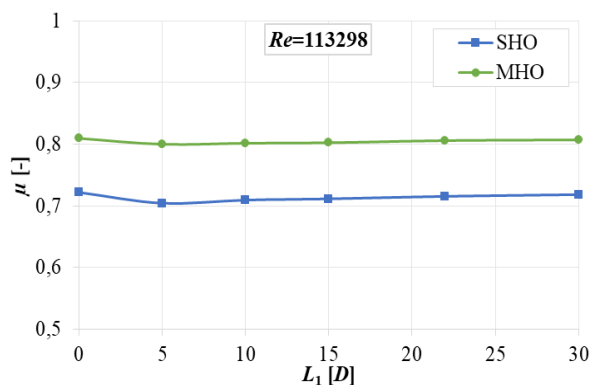
Slika 9. Gubitak strujne snage na pritisku 4 bar i brzini strujanja vazduha od 5 m/s

Analizom dobijenih rezultata može se uočiti da položaj lokalnog otpora L_1 ima značajno veći uticaj na pad pritiska kod merne blende sa jednim otvorom u odnosu na mernu blendu sa više otvora. To je i očekivano, pošto su gubici u direktnoj vezi sa padom pritiska. U tabeli 4, su prikazani rezultati relativnog pada pritiska za razmatrane merne blende. Primećuje se da lokalni otpor više utiče na pad pritiska kod merne blende sa jednim otvorom.

Tabela 4. Relativni pad pritiska

Položaj kolena L_1	$\varepsilon_{\Delta P, SHO}$ [%]	$\varepsilon_{\Delta P, MHO}$ [%]
5D	6,58	2,58
10D	4,49	1,58
15D	3,66	1,21
22D	2,3	0,67
30D	1,25	0,47

Na slici 10, su grafički prikazani rezultati koeficijenta protoka μ za mernu blendu sa jednim i mernu blendu sa više otvora u funkciji položaja lokalnog otpora L_1 ispred merne blende u cevovodu.



Slika 10. Koeficijent protoka na pritisku 4 bar i brzini strujanja od 5 m/s

Može se uočiti da položaj lokalnog otpora nema značajan uticaj na vrednosti koeficijenta protoka kod razmatranih mernih blendi.

5. ZAKLJUČAK

Na osnovu analiziranih rezultata može da se zaključi da lokalni otpor kolena utiče na sledeće strujne parametre. Ukoliko se žele smanjiti gubici strujne energije, preporuka je da se koleno, kao otpor, ne postavlja na cevovod ispred merne blende sa jednim otvorom na manjoj dužini ravne deonice od $22D$ dok se kod merne blende sa više otvora ne preporučuje manje od $10D$.

6. REFERENCE

- [1] Singh V.K. i Tharakan John T., (2015), Numerical simulations for multy-hole orifice flow meter, Flow Measurement and Instrumentation, Vol. 45, Issue 1, 375-383.
- [2] Manisch S. i dr., (2012), Analysis of flow through an orifice meter: CFD simulation, Chemical Engineering Science, Vol. 71, Issue 1, 300-309.

Kratka biografija:



Marko Narić rođen je u Novom Sadu 1990., gde je završio osnovnu i srednju školu. Master rad na fakultetu tehničkih nauka iz oblasti mašinstva - Gasna i naftna tehnika odbranio je 2019. god.

ИСТРАЖИВАЊЕ УТИЦАЈА ЛОКАЛНОГ ОТПОРА ПОСТАВЉЕНОГ ИЗА МЕРНЕ БЛЕНДЕ НА СТРУЈНЕ ПАРАМЕТРЕ

RESEARCH ON INFLUENCE OF LOCAL RESISTANCE PLACED AFTER THE ORIFICE ON THE FLUID PARAMETERS

Вук Спремо, Факултет техничких наука, Нови Сад

Област – МАШИНСТВО

Кратак садржај – Предмет рада је мерна бленда, уређај за мерење протока флуида. Анализиран је утицај колена ($\varphi=90^\circ$ и $R/D=1$) као локалног отпора на вредности разлике притисака, пада притиска, коефицијента протока, губитка струјне снаге и поврата притиска код мерне бленде са једним и код мерне бленде са више отвора приликом струјања сувог ваздуха. Анализа је урађена применом експерименталне методе и рачунарске динамике флуида. Нумеричке симулације су урађене у комерцијалном пакету STARCCM+. Циљ рада је био да се установи да ли различити положаји колена иза мерне бленде на цевоводу имају утицај на наведене струјне параметре. Закључак је да колено нема скоро никаквог утицаја када су у питању разлика притисака, коефицијент протока и поврат притиска. Са друге стране, уочен је утицај, који није занемарујући, на вредности пада притиска и губитка струјне снаге.

Кључне речи: Мерна бленда, мерење протока

Abstract – The subject of this paper was an orifice for measurement of fluid flow rate. The influence of a local loss - elbow on pressure drop, pressure loss, fluid power loss, discharge coefficient and pressure recovery for single-hole and multi-hole orifices was analyzed. The stream of a dry air through both types of orifice was considered. Analysis was done using computational fluid dynamics (CFD). The numerical simulations were done in commercial software package STAR CCM+. The aim of this work was to determine if different positions of elbow behind the orifice have the influence on the mentioned fluid parameters. It was concluded that elbow doesn't have the influence on the pressure drop, discharge coefficient and pressure recovery. However, the elbow does have the influence on the pressure loss and fluid power loss for the both types of orifice.

Key words: Orifice, Fluid flow rate

1. УВОД

Предмет рада била је мерна бленда за мерење протока гаса. Разматран је утицај колена ($\varphi=90^\circ$ и $R/D=1$) као локалног отпора постављеног на цевовод унутрашњег пречника $D=70,3$ mm иза мерне бленде на струјне параметре. Утицај присуства колена са ниструјне стране сагледан је за мерну бленду са једним и са више отвора, обе параметра $\beta=0,55$.

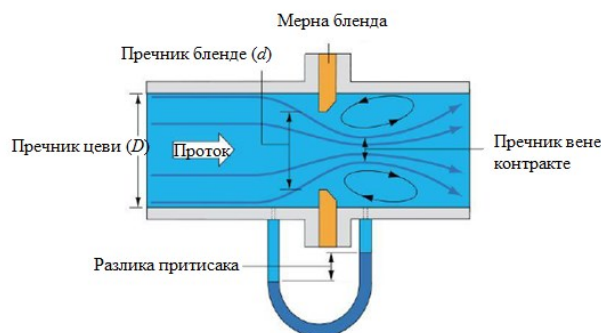
НАПОМЕНА:

Овај рад је проистекао из мастер рада чији ментор је био проф. др Синиша Бикић.

Анализа је урађена за струјање сувог ваздуха кроз цевовод температуре ваздуха испред бленде од 22°C . Разматрани су струјни услови у цевоводу испред мерне бленде брзине од 5 и 10 m/s и притисци од 4, 8 и 12 bar.

2. ТЕОРИЈСКЕ ОСНОВЕ

Неколико типова мерила протока (протокомера) користи принцип пригушивања флуидне струје. Међу њима је и мерна бленда. Показатељ протока код овог мерног уређаја је разлика притисака, па се понекад овај уређај назива и диференцијални протокомер. Величине релевантне при изучавању струјања флуида у цеви произвољног попречног пресека су притисак p , густина ρ , температура T и брзина струјања флуида v .



Слика 2.1 Мерна бленда [1]

Математички модел којим се може, према разлици притиска, израчуати запремински проток гласи:

$$Q = \psi A_0 v_2 = \frac{\psi A_0}{\sqrt{\alpha_2 - \alpha_1 m^2 \psi^2}} \sqrt{\frac{2}{\rho} (p_1 - p_2)}$$

где су:

Q – запремински проток флуида [m^3/s];

p_1 – притисак испред бленде [Pa];

p_2 – притисак иза бленде [Pa];

ρ – густина флуида [kg/m^3];

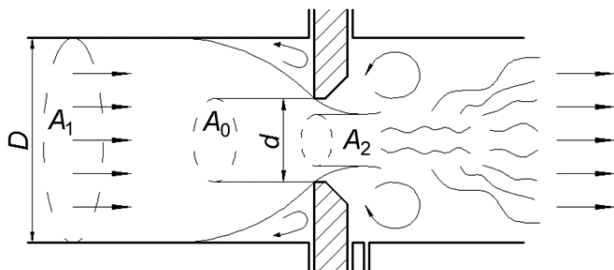
v_2 – средња брзина струјања флуида иза бленде [m/s];

A_0 – површина отвора бленде [m^2];

ψ – коефицијент контракције [-].

Како флуидна струја дотиче бленди, притисак расте незнатно, па затим нагло опада како пролази кроз исту. Наставља да опада све док флуидна струја не формира тзв. вену контракту, а затим постепено расте све до приближно 5-8 пречника цеви низводно. Највиша вредност притиска иза бленде је мања од

вредности притиска испред бленде због локалних губитака и вртлога који настају убрзавањем флуидне струје протицањем кроз бленду. Треба разликовати појмове **разлика притисака** и **пад притиска**. Наиме, разлика притисака се појављује непосредно испред и иза бленде, док пад притиска представља разлику између неузнемирене флуидне струје испред и иза мерне бленде.

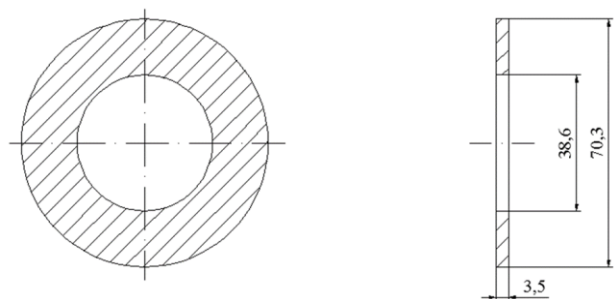


Слика 2.2 Струјање флуида кроз бленду [2]

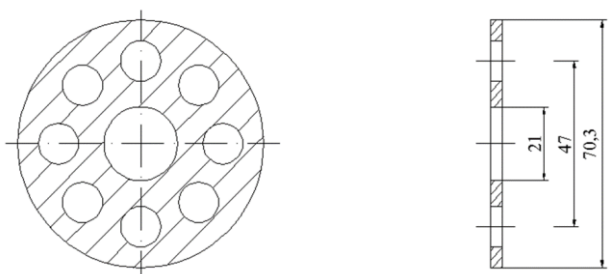
3. МАТЕРИЈАЛ И МЕТОД

Нумеричка симулација је реализована у програмском пакету STARCCM+ који за дискретизацију парцијалних диференцијалних једначина користи метод коначних запремина [3].

Посматран је проток ваздуха кроз мерну бленду са једним отвором (ШО), слика 3.1, и мерну бленду са више отвора (МНО), слика 3.2. Треба нагласити да се ради о сувом ваздуху, тако да утицај влаге на резултате струјних параметара неће бити разматран.



Слика 3.1 Мерна бленда са једним отвором



Слика 3.2 Мерна бленда са више отвора

Мерне бленде су постављене у цеовод унутрашњег пречника $D=70,3$ mm, док је ширина мерних бленди $3,5$ mm. Иако је циљ овог рада био да се анализира утицај колена на струјне параметре мерењем протока мерном блендом са више отвора, урађене су и симулације са мерном блендом са једним отвором, како би се ова два типа бленди могли међусобно упоредити.

Као једино упрошћење у геометрији мерне бленде сматраће се да нема закошених ивица. У табели Т.1 приказани су услови при којима су урађене симулације.

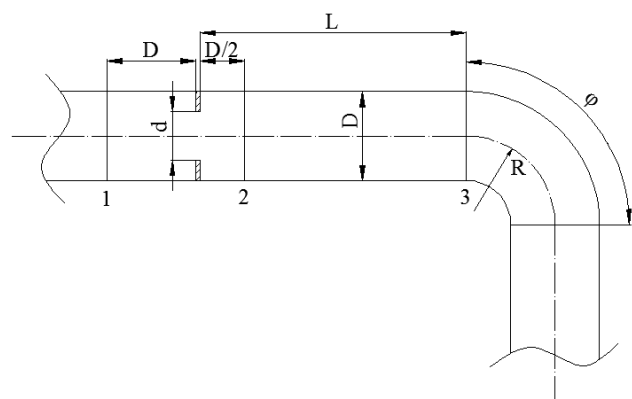
Т.3.1 Параметри са којима ће се подешавати нумеричке симулације

Радни флуид:	Суви ваздух
Температура флуида t :	22 °C
Притисак p :	4 bar 8 bar 12 bar
Густина ρ :	5,92 kg/m ³ 10,65 kg/m ³ 15,417 kg/m ³
Брзина струјања v :	5 m/s 10 m/s
Модел турбуленције:	$k-\epsilon$
Растојање колена од бленде L :	1D, 2D, 3D, 4D, 5D, 6D, 7D

Напомиње се да је урађен и експеримент који ће послужити као верификација добијених података симулација. Експериментална метода је урађена са равним деоницама испред (860 mm) и иза (276,2 mm) мерне бленде без присуства колена. Радни флуид је био ваздух у опсегу притиска од 2 до 12 bar, температуре од 20 до 32° C и запреминским протоцима ваздуха од 14 до 150 m³/h.

Пажња је била усредсређена на пет величина: разлику притисака Δp_{1-2} која настаје на бленди, пад притиска Δp_{1-3} који настаје због постојања мерне бленде, коефицијент протока μ , губитак струјне снаге ΔP и поврат притиска p_x/p_1 .

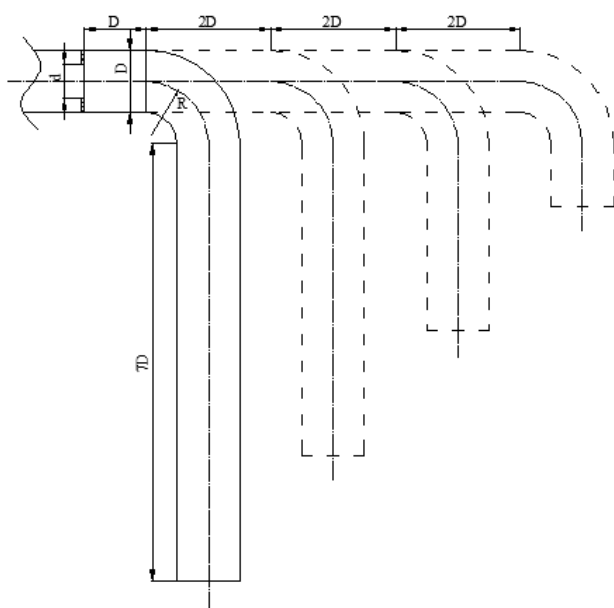
На слици 3.3 приказане су равни на којима ће се мерити вредности притисака испред бленде p_1 на растојању од $1D$, иза бленде p_2 на растојању $D/2$ и на улазу у колено p_3 (и тако за сваку деоницу). Растојање L представља дужину деонице цевова од мерне бленде до колена и оно варира од $1D$ до $7D$. Реч је о притисцима који су осредњени по површини попречног пресека (статички притисци при мерењу притиска флуида у неком пресеку).



Слика 3.3 Приказ равни на којима су мерени притисци p_1 , p_2 и p_3

Анализирана је енергетска ефикасност бленде за случај када се иза мерне бленде у цевоводу налази колено на различитим растојањима.

Овде је битно напоменути да се приликом померања колена за свако D од бленде, неће мењати запремина флуидног простора, јер за свако повећање растојања колена од бленде L , дужина равне деонице иза колена ће се смањити за исто растојање, слика 3.4.



Слика 3.4 Приказ померања колена иза мерне бленде

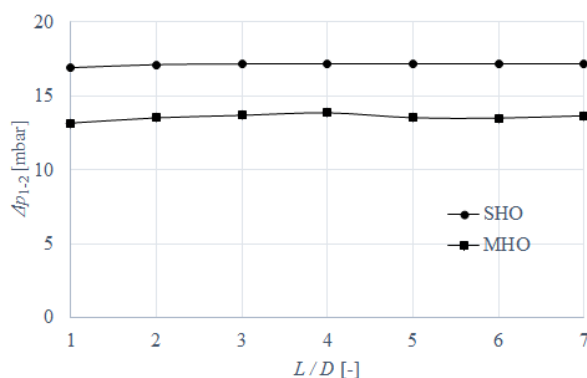
4. РЕЗУЛТАТИ

На основу добијених резултата установљен је утицај положаја колена иза мерне бленде на наведене струјне параметре.

Анализом добијених резултата експеримента и нумеричких симулација примећује се да су вредности симулација релативно блиске са вредностима експеримента, као и да је вредност релативне грешке у просеку мања код мерне бленде са једним отвором и износи 5%. Релативна грешка код мерних бленди са више отвора у просеку износи 13,5%.

4.1 Разлика притисака

На слици 4.1 су графички приказани резултати разлике притисака Δp_{1-2} за мерне бленде са једним и са више отвора.



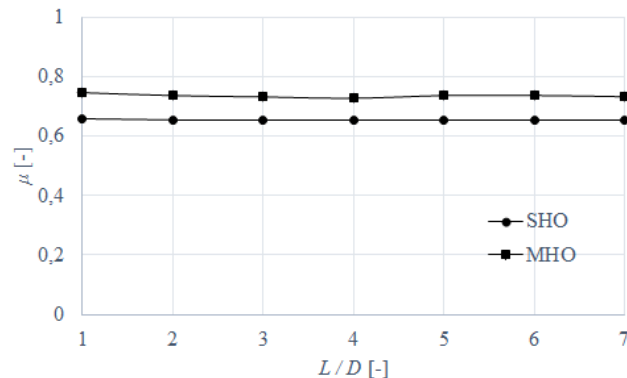
Слика 4.1 Разлика притисака при условима испред бленде: притисак ваздуха од 4 bar и брзина струјања ваздуха од 5 m/s

Анализом добијених резултата уочено је да растојање колена од мерне бленде нема значајан утицај када је у питању разлика притисака, и за случај мерне бленде са једним и за случај мерне бленде са више отвора, што је и било очекивано. Мерне бленде са једним отвором стварају већу разлику притисака у односу на бленде са више отвора, што је такође било очекивано.

Бленда са једним отвором је осетљивија па се, при истим условима, на њој мери тачније разлика притисака а самим тим, и запремински проток ваздуха.

4.2 Коефицијент протока

На слици 4.2 графички су приказане вредности коефицијента протока μ за мерне бленде са једним и са више отвора.

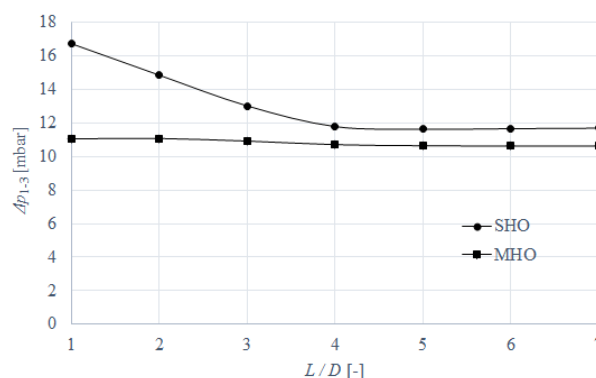


Слика 4.2 Коефицијент протока при условима испред бленде: притисак ваздуха од 4 bar и брзина струјања ваздуха од 5 m/s

Анализом добијених резултата може се закључити да разматрано колена као локални отпор нема готово никакав значајан утицај на вредност коефицијента протока, ни за мерну бленду са једним ни са више отвора. То је и очекивано на основу предходне анализе, јер је коефицијент протока μ у директној вези са разликом притисака Δp_{1-2} . Такође, може се закључити и да је вредност коефицијента протока већа код мерних бленди са више отвора у односу на бленду са једним отвором.

4.3 Пад притиска

На слици 4.3 графички су приказане вредности пада притиска Δp_{1-3} за мерну бленду са једним и са више отвора.



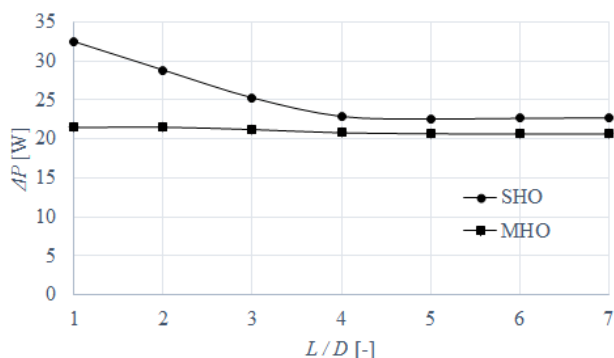
Слика 4.3 Пад притиска при условима испред бленде: притисак ваздуха од 4 bar и брзина струјања ваздуха од 5 m/s

Анализом добијених резултата може се закључити да присуство колена иза мерне бленде има утицај на пад притиска ваздуха. Код мерне бленде са једним отвором пад притиска ваздуха је услучају присуства разматраног колена на растојању колена од мерне бленде мањем од 4D за 20-30% већи од случаја кад цевовод има равну деоницу иза мерне бленде без

присуства колена. Присуство разматраног колена има мањи утицај на пад притиска ваздуха код мерне бленде са више отвора, где је на растојању колена од мерне бленде мањем од $4D$ пад притиска за 5-10% већи у односу на случај равне деонице без присуства колена. Такође, треба напоменути и да је пад притиска ваздуха код мерне бленде са више отвора мањи у односу на мерну бленду са једним отвором, што је и очекивано. За случај већег растојања од $4D$, колена нема значајнијег утицаја на вредности струјних параметара.

4.4 Губитак струјне снаге

На слици 4.4 графички су приказане вредности губитка струјне снаге ΔP за мерну бленду са једним и са више отвора.



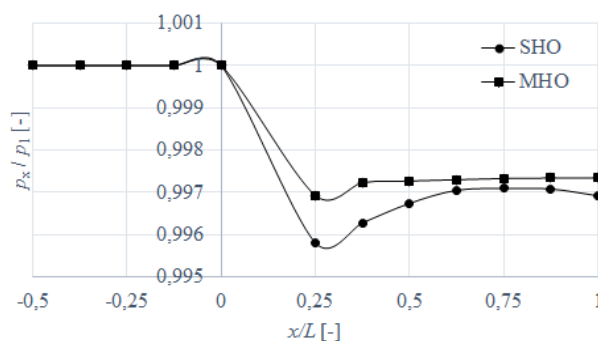
Слика 4.4 Губитак струјне снаге при условима испред бленде: притисак ваздуха од 4 bar и брзина струјања ваздуха од 5 m/s

Анализом добијених резултата може се закључити да присуство разматраног колена такође има утицај, који није занемарљив, када су у питању губици струјне снаге. То је и савим очекивано, пошто су губици у директној вези са падом притиска. Губици струјне снаге су мањи код мерне бленде са више отвора од губитака код мерне бленде са једним отвором. За случај присуства разматраног колена на растојању од бленде са једним отвором мањем од $4D$ за 20-30% су већи губици струјне снаге него за случај кад цевовод има равну деоницу иза мерне бленде без присуства колена. Када је у питању бленда са више отвора, губици струјне снаге су за 5-10% већи него за случај равне деонице иза бленде тј. без присуства колена. Јасно је да што је колена ближе мерној бленди, то је његов утицај већи на вредности губитка струјне снаге.

4.5 Поврат притиска

На слици 4.5 графички су приказане вредности поврата притиска p_x/p_1 .

Анализом добијених резултата може се закључити да положај разматраног колена у односу оба типа мерних бленди нема готово никакав утицај на поврат притиска. Наиме, најнижа вредност поврата притиска се налази на око 0,25 вредности бездимензионог растојања x/L , и за случај мерне бленде са једним и за случај мерне бленде са више отвора.



Слика 4.5 Поврат притиска при условима испред бленде: притисак од 4 bar брзина струјања ваздуха од 5 m/s

Оно што се такође може закључити је да је поврат притиска већи код бленди са више отвора у односу на бленде са једним отвором.

5. ЗАКЉУЧАК

Анализирањем је установљено да колена има већи утицај на губитак када се налази ближе мерној бленди, што доводи до закључка да уколико се жели избећи велики губитак струјне снаге, препорука је да се колена, као отпор, не поставља на цевовод на мањој дужини равне деонице од $4D$ од мерне бленде. Када су у питању разлика притисака, коефицијент протока и поврат притиска, установљено је да колена нема готово никакав утицај на ове струјне параметре. Упоредицањем мерних бленди са једним и са више отвора закључено је да оба типа имају своје предности у односу на други тип. Уколико захтеви у пракси налажу да мерење протока буде тачније, тада се треба користити бленда са једним отвором. Овај тип бленде ствара већу разлику притиска и то је чини осетљивијом што доводи до веће тачности при мерењу. Ако, са друге стране, захтеви налажу да губици струјне снаге буду што мањи, тада се треба ослонити на мерне бленде са више отвора.

6. РЕФЕРЕНЦЕ

- [1] Instrumentation Tools, 2016.
- [2] П. Цвијановић, (1998). Мерење флуидних величина, Стилос.
- [3] STARCCM+ Tutorial, 2019.

Кратка биографија:



Вук Спремо рођен је у Новом Саду, 1992. године, где је завршио основну и средњу школу. Дипломирао је 2016. године на Факултету техничких наука, студијски програм Машинство, на Департману за Енергетику и процесну технику, смер Гасна и нафтна техника.

**PROJEKTOVANJE TEHNOLOGIJE TERMIČKE OBRADJE LEŽAJNOG PRSTENA
PRIMENOM RAČUNARA****DESIGN OF THE HEAT TREATMENT TECHNOLOGY OF A BEARING RING USING A
COMPUTER**

Novica Nikolić, *Fakultet tehničkih nauka, Novi Sad*

Oblast- MAŠINSTVO

Kratak sadržaj – U radu je objašnjena tehnologija termičke obrade prstena kugličnih ležaja. Data je analiza procesa kaljenja i otpuštanja, kao i problemi koji se mogu javiti pri radu. Definisan je pogon termičke obrade FKL-a i spisak potrebnih peći za termičku obradu. Prikazana je upotreba programskog paketa MESAVIS za zadavanje i praćenje parametara procesa termičke obrade putem računara.

Ključne reči: Tehnologija termičke obrade, ležajni prsten, MESAVIS.

Abstract – In this paper heat treatment technology of bearing rings is explained. An analysis of the hardening and tempering process is given, as well as problems that can arise during work. The FKL plant and list of necessary furnaces for heat treatment are defined. The use of the software package MESAVIS for setting and monitoring the parameters of heat treatment process through the computer is shown.

Keywords: Heat treatment technology, bearing rings, MESAVIS.

1. UVOD

Ležajevi su mašinski elementi čija se osnovna funkcija sastoji u obezbeđivanju uslova za relativno kretanje obrtnih delova i za prenošenje opterećenja između njih. Zbog prirode opterećenja, ležajevi su izloženi habanju, jer se kotrljajna tela kreću po stazama između unutrašnjeg i spoljašnjeg prstena. Zbog toga je neophodno uraditi termičku obradu, koja doprinosi povećanju mehaničkih karakteristika i otpornosti na habanje. Najvažniji zahtevi koje treba da ispuni čelik za ležajeve jesu visoka čvrstoća, otpornost na habanje kotrljanjem i trajna stabilnost dimenzija i oblika.

2. TEHNOLOGIJA TERMIČKE OBRADJE I FAZNE TRANSFORMACIJE

Termička obrada se može definisati kao operacija, ili skup operacija koje uključuju zagrevanje, držanje na određenoj temperaturi i potom hlađenje, u cilji dobijanja materijala koji će imati željene osobine, ili koji će biti doveden u željeno stanje za dalju obradu.

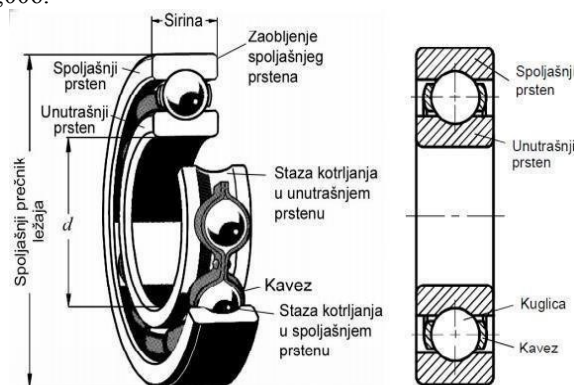
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Branko Škorić, red.prof.

Najznačajnije fazne promene su one koje se odigravaju kao rezultat smanjenja temperature, na kojoj se metal nalazi u ravnotežnom stanju kao jedna faza, na drugu temperaturu na kojoj se stabilno stanje sastoji od neke druge faze ili od dve različite faze. Nastanak prve male pločice cementita je uslovljen difuzijom ugljenika iz neposredne okoline pločice, jer se koncentracija ugljenika u pločici mora uvećati sa 0,8% C na 6,67% C, koliko je potrebno za obrazovanje cementita. Beinitna transformacija. Ako je brzina hlađenja pri kontinualnom hlađenju podešena tako da se transformacija dešava između 200 °C i 450 °C ili ako se izotermalno transformiše u ovom opsegu temperatura, nastaje beinitna struktura. Martenzitna transformacija. Ako se hlađenje austenita obavi dovoljno velikom brzinom, ili na dovoljno niskoj temperaturi procesom transformacije nastaje martenzit. Sigurno je samo da je proces transformacije bez difuzije.

3. KOTRLJAJUĆI LEŽAJEVI

Ležajevi su mašinski elementi koji služe za smanjenje trenja u ležištima osovina. Oni su oslonci vratila i obrtnih osovina, primaju radijalne i aksijalne sile koje deluju i zadržavaju zadati položaj ose obrtanja vratila odnosno osovina. Ležajevi prenose samo sile a ne i obrtne momente. Relativni koeficijent trenja kotrljanja je mali i veoma je blizak koeficijentu tečnog trenja, koji iznosi 0,0015 – 0,006.



Slika 1. Delovi kotrljajnog ležaja [1]

4. TEHNOLOGIJA TERMIČKE OBRADJE PRSTENOVA LEŽAJA

Zbog habanja između prstenova i kuglica ležaja, mora se odraditi odgovarajuća termička obrada. Prstenovima ležajnog čelika koji se koriste u FKL se termičkom obradom dobija tvrdoća od 57-62 HRC. Čelici za kotrljajuće ležajeve su po svojoj nameni konstrukcioni, ali

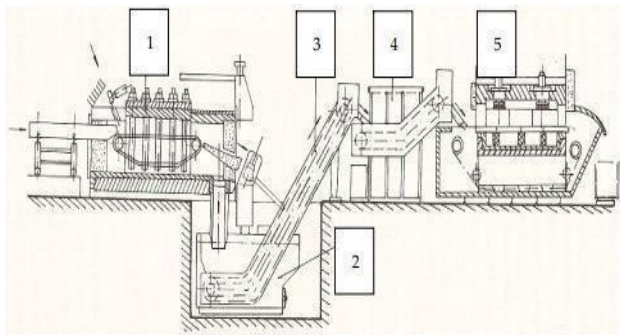
po sastavu, termičkoj obradi i najvažnijim osobinama zapravo alatni čelici. Najvažniji zahtevi koje treba da ispuni čelik za ležajeve jesu visoka čvrstoća, otpornost na habanje kotrljanjem i trajna stabilnost dimenzija i oblika. Jedan od osnovnih parametara kaljenja je temperatura kaljenja, čije su granice određene greškama u strukturi, koje će se javiti posle kaljenja.

Suviše visoka temperatura ima za posledicu gruboigličasti martenzit i nepovoljno utiče na krupnoću zrna, dok suviše niska temperatura omogućava prisustvo beinita i grubih karbida u strukturi, te smanjuje tvrdoću. Brzina zagrevanja čelika za kotrljajuće ležajeve na temperaturu kaljenja može biti nešto veća. Kaljenje se vrši u komornim ili protočnim pećima. Najveća količina dispergovanih karbida rastapa se za kratko vreme, nakon što se postigne temperatura kaljenja.

Za kotrljajuće ležajeve, koji moraju imati relativno visoku tvrdoću, temperatura otpuštanja iznosi od 150 °C do 180 °C. Kod otpuštanja natoj temperaturi dolazi do sniženja tvrdoće za 1 do 2 HRC. Ležajni čelik koji se koristi za proizvodnju ležajeva FKL-a – Č.4146 (100Cr6) komercijalne oznake OCR 4 spada u nadeutektoidne čelike sa sadržajem ugljenika preko 0,8%. Duboko hlađenje je veoma značajan proces termičke obrade u industriji ležajeva, jer se ovom metodom količina zaostalog austenita svodi nanajmanju moguću meru. Ovim postupkom se postiže odgovarajuće povećanje tvrdoće i bolja stabilizacija strukture, smanjuju se unutrašnji strukturni naponi i time umanjuje sklonost ka spontanoj promeni specifične zapremine utoku vremena – tzv. starenje.

5. PEĆI ZA TERMIČKU OBRADU

Najvažnija namena peći za termičku obradu je osiguravanje ravnomernog zagrevanja komada. Njihova konstrukcija mora odgovarati svim zahtevima standarda kvaliteta. Postrojenje komornih peći je izvedeno kao niz komornih peći ispred kojih su, ispod kote nula, smeštene kade za kaljenje u ulju. Ispred komornih peći postoji transporter koji omogućava prenos obrađenog materijala. Protočna peć za termičku obradu je sastavljena iz osnovnih tehnoloških jedinica koje su međusobno povezane transportnim i programskim sistemom koji omogućava unapred programiran automatski rad. Postoje dve osnovne jedinice protočne peći u FKL: peć za kaljenje i peć za otpuštanje.



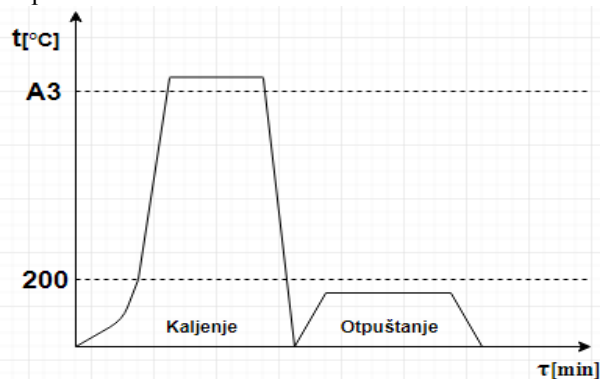
Slika 2. Šema protočne peći: 1 - peć za kaljenje, 2 - kada za kaljenje u ulju, 3 - lift, 4 - mašina za pranje, 5 - peć za otpuštanje

Prstenovi ležajnog čelika imaju bolje mehaničke karakteristike ukoliko se njihova termička obrada vrši u protočnim pećima a ne u komornim. Razlog tome je

postepeno zagrevanje u zonama – tri zone kaljenja i četiri zone otpuštanja.

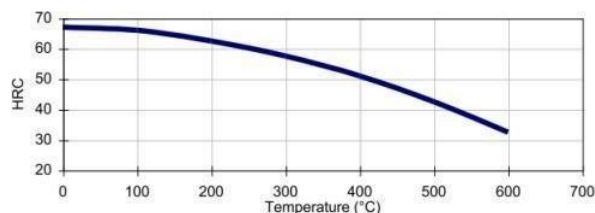
6. TERMIČKA OBRADA SPOLJAŠNJEG PRSTENA UH 210N-3/1

Radi dobijanja potrebne tvrdoće, radi se termička obrada i spoljašnjih i unutrašnjih prstenova. Termička obrada se sastoji od kaljenja i otpuštanja. Struktura čelika se, posle kaljenja, sastoji iz martenzita i zaostalog austenita. Kaljenje podrazumeva proces termičke obrade čelika u kome se komad zagreva iznad temperature A3, a zatim dovoljno brzo hladi kako bi se dobila neravnotežna struktura martenzita velike tvrdoće. Kaljenje se vrši na temperaturi od 840 °C.



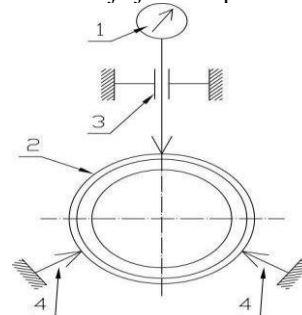
Slika 3. Dijagram termičke obrade ležajnog prstena

Pošto su ove faze nestabilne, postoji prirodna težnja da pređu u stabilniji oblik sa nižim energetske nivoom. Kod delova ležajeva posle kaljenja, vrši se samo nisko otpuštanje, sa povećanjem žilavosti i uklanjanjem zaostalog austenita.



Slika 4. Zavisnost tvrdoće od temperature otpuštanja

Tehnološki, ovalnost je rezultat nesimetrične raspodele unutrašnjih napona pre kaljenja i nejednakog zagrevanja i hlađenja. Ovalnost je određena vrsta kružnog odstupanja. Ovalnost se proverava nakon i pre procesa termičke obrade. [2] Za merenje se koristi specijalni mikrometar čija je šema prikazana na slici 5.



Slika 5. Šema uređaja za merenje ovalnosti

Merena je ovalnost tri prstena, i pravljene su zapisi nakon svake rotacije za 10°. Rezultati su očitavani sa mernog sata. Ukoliko je odstupanje veće od 0.2 mm, prsten nije

dobar i ide na doradnu operaciju španovanja, a ukoliko i tada bude preveliko odstupanje, prsten se smatra škartom. Nije prošao kriterijum kvaliteta i neće se slati na dalju obradu brušenja staza.

Veliko odstupanje ovalnosti može biti posledica nejednakog zagrevanja ili hlađenja u toku procesa termičke obrade. Ovalnost je merena na tri unutrašnja prstena uzeta nasumično iz različitih serija. Prsten A pokazuje zadovoljavajuću ovalnost do 0.2 mm. Prsten B ima ovalnost preko 0.2 mm, tako da nije pogodan za dalju mašinsku obradu. Najniža ovalnost je kod prstena C. Za prsten B je neophodno identifikovati uzrok tako velikog odstupanja [3].

ROTACIJA PRSTENA	0	10	20	30	40	50	60	70	80	90	100
Prsten A	0.02	0.04	0.09	0.13	0.16	0.17	0.17	0.18	0.19	0.2	0.19
Prsten B	0.21	0.2	0.19	0.18	0.17	0.16	0.13	0.09	0.04	0.03	0.03
Prsten C	0.09	0.05	0.02	0	0.02	0.04	0.08	0.09	0.1	0.11	0.12

Slika 6. Odstupanje ovalnosti prstena -1 [2]

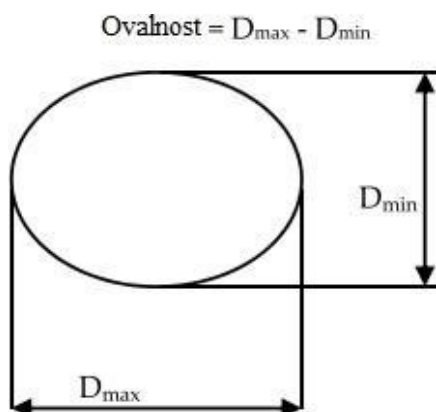
110	120	130	140	150	160	170	180	190	200	210	220	230
0.18	0.17	0.17	0.16	0.13	0.09	0.04	0.02	0.03	0.04	0.09	0.13	0.16
0.04	0.09	0.13	0.16	0.17	0.18	0.19	0.21	0.2	0.19	0.18	0.17	0.17
0.13	0.14	0.15	0.15	0.14	0.13	0.12	0.11	0.1	0.09	0.05	0.02	0.04

Slika 7. Odstupanje ovalnosti prstena -2 [2]

240	250	260	270	280	290	300	310	320	330	340	350	360
0.17	0.17	0.18	0.19	0.2	0.19	0.18	0.17	0.17	0.16	0.13	0.09	0.05
0.16	0.13	0.09	0.05	0.02	0.04	0.09	0.13	0.16	0.18	0.19	0.2	0.21
0.08	0.1	0.1	0.11	0.12	0.13	0.14	0.15	0.15	0.14	0.13	0.12	0.1

Slika 8. Odstupanje ovalnosti prstena -3 [2]

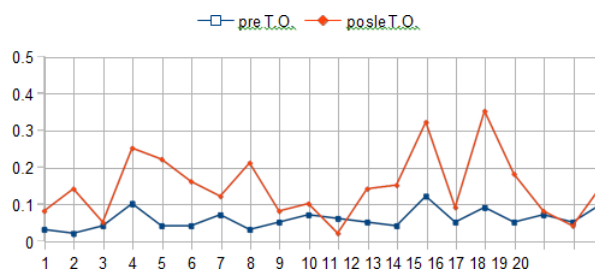
Deformacija materijala i ovalnost prstenova su izazvane zaostalim naponima mašinske obrade i u procesu termičke obrade. Deformacije mogu biti izazvane pogrešnim štelovanjem ili nepoštovanjem tehnoloških instrukcija. Ovalnost se definiše kao razlika dijametara rotiranih za 90° [3].



Slika 9. Definicija ovalnosti [2]

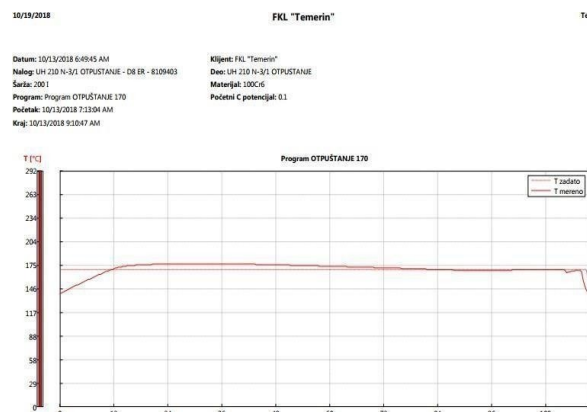
Ovalnost se najčešće povećava nakon procesa termičke obrade. Prikaz merenja jednog prstena pre i nakon termičke obrade prikazan je na slici 10.

Veliki problem kod delova kotrljajućih ležajeva i preciznih alata je deformisanje za vreme kaljenja, što je posledica zaostalih napona nastalih transformacijom, neravnomernim hlađenjem ili eventualnom pogrešnom manipulacijom komada za vreme hlađenja. Posledice ovih napona se javljaju u dva vida: kao promena oblika i kao promena dimenzija.



Slika 10. Ovalnost pre i posle procesa t.o.

7. PRIKAZ PROGRAMSKOG PEKETA MESAVIS ZA UNUTRAŠNJI PRSTEN UH 210N-3/1



Slika 11. Dijagram kaljenja ležajnog prstena u Mesavisu

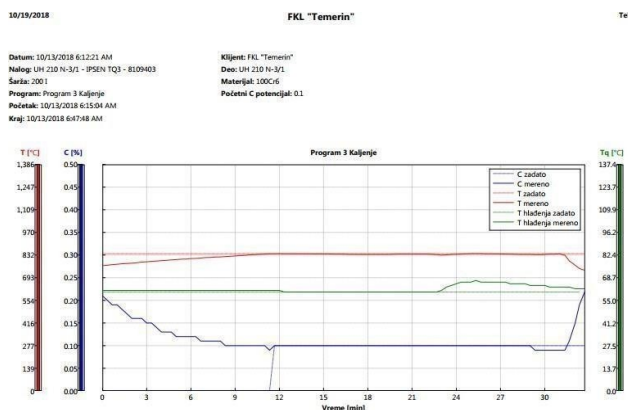
Za spoljašnji prsten UH 210 N-3/1, za koji je ranije obrađena problematika termičke obrade, urađeno je softversko zadavanje parametara procesa. Podaci koje dobijamo ovakvim radom se očitavaju na dijagramu, koji predstavlja izlazni rezultat nakon završetka procesa termičke obrade. Iznad dijagrama se nalaze sledeći podaci: Datum i vreme početka i kraja procesa, broj radnog naloga, broj komada u šarži, program koji se koristio, naziv radnog komada, materijal komada i početni C potencijal. Ovo su opšti podaci, bitni za razlikovanje i kasniju analizu završenih procesa.

Temperatura atmosfere: Zadana temperatura atmosfere je 840 °C, što je standard kod ležajnih čelika Č.4146. Nakon ubacivanja šarže u peć temperatura je bila nešto niža, i na dijagramu se vidi kako ona postepeno raste dok ne dođe do zadate vrednosti. Na kraju dijagrama uočava pad temperature, to je trenutak otvaranja peći nakon završenog procesa kaljenja.

Temperatura ulja: Zadana temperatura ulja je 60 °C, ta temperatura je bila i izmerena u toku celog procesa, sve dok platforma sa zagrejanim komadima nije spuštена u kadu sa uljem, kada je očitao skok temperature ulja. Nakon toga dolazi do paljenja sistema za hlađenje kade sa uljem, pa se vidi i postepeni pad temperature.

C potencijal: Zadati C potencijal za ovaj proces je 0,1%. Na početku procesa očitao je C potencijal 0,2% jer su se prednja vrata peći otvarala zbog ubacivanja šarže. Nakon toga sledi pad vrednosti zasićenosti atmosfere ugljenikom do zadatih 0,1%. Na kraju sledi ponovni rast C potencijala, kada se otvaraju zadnja vrata i šarža vadi iz peći.

Vreme programa 3 (10 min) će krenuti tek nakon što se ostvare svi uslovi za proces, a to je da temperatura atmosfere bude 840 °C (uz zadatu toleranciju 5 °C) a vrednost C potencijala bude 0,1% (uz toleranciju 0.03%).



Slika 12. Dijagram otpuštanja ležajnog prstena u MESAVISU

Kod procesa otpuštanja, dijagram je znatno jednostavniji. Opšti podaci su, kao i kod procesa kaljenja, prikazani iznad dijagrama. Na dijagramu otpuštanja očitava se vrednost temperature u toku vremena. Zadana temperatura otpuštanja za spoljašnji prsten UH 210 N-3/1 je 175 °C, ali se u programu 170 °C. Razlog tome su mikroprekidači u peći za otpuštanje D8 ER, koji se ne isključuju na vreme grejače tako da ima malog "prebačaja" temperature. To ne predstavlja ozbiljan problem jer se temperatura zadržava u granicama tolerancija.

Prednosti softverskog zadavanja i praćenja parametara termičke obrade su prvenstveno u dobijanju dodatne stabilnosti procesa. Ukoliko neki od parametara procesa nije u graničnim vrednostima, odmah se analizira razlog tome. Ako vrednosti koje odstupaju nisu znatno uticajne na kvalitet obrađenih komada, rad u pećima se nastavlja i istovremeno se radi na rešavanju problema.

Kada su vrednosti znatno van granica tolerancije parametara u pećima, rad se zaustavlja i hitno vrši popravka peći. Kvar peći je najčešći uzrok loših parametara procesa. Ukoliko se desi na merenju tvrdoće nakon procesa termičke obrade, da neki komadi nemaju dovoljnu tvrdoću, ulazi se u bazu podataka završenih radnih naloga Mesavisa, kako bi se našao uzrok problema.

8. ZAKLJUČAK

Zbog prirode opterećenja, ležajevi su izloženi habanju, jer se kotrljajna tela kreću po stazama između unutrašnjeg i spoljašnjeg prstena. Termičkom obradom spoljašnjeg prstena UH 210-N-3/1 postižu se potrebne mehaničke karakteristike. Kaljenje je vršeno na 840 °C sa zadržavanjem na toj temperaturi 10 minuta, uz C potencijal 0,1. Otpuštanjem na 175 °C dobijaju se završne vrednosti tvrdoće i relaksacija strukture. Radi dodatne sigurnosti procesa i zahteva tržišta, uvedena je i kontrola postupaka termičke obrade putem računara.

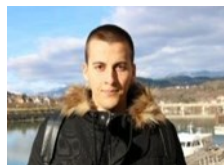
Programski paket Mesavis omogućava zadavanje i praćenje parametara procesa termičke obrade. Izlazni zapisi su dijagrami sa kojih se očitava promena temperature atmosfere i ulja, zatim C potencijal u toku vremena. Ovi podaci se analiziraju i pružaju dodatnu sigurnost u stabilnost procesa.

Na ovaj način se mogu na vreme otkriti greške unutar peći za termičku obradu i pronaći uzroci njihovog nastanka.

9. LITERATURA

- [1] Ilija Pantelić, "Tehnologija termičke obrade čelika", Subotica 1976
- [2] Vladimir M. Blanuša, "Analiza cilindrično valjkastih ležaja za specijalne namene", Fakultet tehničkih nauka, Novi Sad, 2017
- [3] J. Burke, "Kinetika faznih transformacija u metalima", Ujedinjeno Kraljevstvo, 1981

Kratka biografija:



Novica Nikolić je rođen u Bijeljini 1994.god. Diplomski-master rad na Fakultetu tehničkih nauka u Novom Sadu iz oblasti tehnologije termičke obrade odbranio je 2018.god.

Dr Branko Škorić, redovni profesor na Fakultetu tehničkih nauka u Novom Sadu, rođen 1955. god. Doktorirao iz oblasti inženjerstva površina i nanotehnologija.

НАМЕНСКИ ЈЕЗИК ЗА МОДЕЛОВАЊЕ РАЧУНАРСКИХ МРЕЖА И ГЕНЕРИСАЊЕ КОНФИГУРАЦИЈА SWITCH УРЕЂАЈА**A DOMAIN-SPECIFIC LANGUAGE FOR COMPUTER NETWORK MODELING AND GENERATION OF SWITCH CONFIGURATIONS**

Јелена Јаковљевић, Факултет техничких наука, Нови Сад

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – У овом раду представљен је наменски језик за моделовање рачунарских мрежа и генератор за генерисање конфигурационих скрипти повезаних уређаја, где је посебна пажња посвећена switch уређајима. Описан је алат који се заснива на основном конфигурисању switch уређаја компаније Cisco. За развој коришћено је окружење EMF за креирање мета-модела и окружење Sirius за развој графичке синтаксе наменског језика.

Кључне речи: Наменски језици, развој софтвера вођен моделима, рачунарске мреже, switch уређај, детекција циклуса у графу

Abstract – In this work we present a domain-specific language for computer network modeling and a generator for generating configuration scripts for connected devices. The focus of this work is on switch devices. The developed tool is focused on the basic configuration of Cisco switch devices. The EMF environment was used for the creation of the meta-model while the Sirius environment was used for developing the graphical syntax of the domain-specific language.

Keywords: Domain-Specific Language, Model-Driven Software Development, Computer Networks, Network Switch, Cycle Detection in Graph

1. УВОД

Рачунарска мрежа је појам који се односи на рачунаре и друге уређаје који су међусобно повезани, кабловима или на други начин, у сврху међусобне комуникације и размене података. У зависности од географске области коју обухватају, рачунарске мреже се деле на локалне и глобалне. Рачунарску мрежу у локалној области, LAN (Local Area Network), чини група мрежних уређаја у истој области међусобно повезаних одговарајућим кабловима или бежичном везом. Тема овог рада јесу LAN мреже. Најчешће су то мреже које се користе у истој згради или комплексу зграда.

Како се рачунарска мрежна система развијају из дана у дан, постоји оправдана потреба за развојем алата за пројектовање и анализу мрежних система. Најбољи начин за представљање мрежних система јесте шематски приказ топологије рачунарских мрежа јер је

најразумљивији особама које раде са таквим системима. Зато се као логичан избор за моделовање рачунарских мрежа намеће графички наменски језик. Графички језик представља информације користећи просторни размештај графичких елемената. Визуелна нотација уводи графичке симболе за концепте моделовања тако што повезује концепте моделовања описане у мета-моделу тј. апстрактној синтакси, на њихову визуелну, конкретну представу.

Приликом пројектовања и подешавања рачунарских мрежа потребно је за сваки уређај подесити његове мрежне карактеристике. Што је рачунарска мрежа већа и садржи више повезаних уређаја, овај посао је сложенији. Ручно подешавање сваког појединачног уређаја је напорно и подложно грешкама. Стога би било пожељно у што већој мери аутоматизовати и олакшати описани процес. Аутоматизација процеса подешавања рачунарских мрежа представља и основну мотивацију за овај рад. Циљ је убрзати и поједноставити поступак подешавања рачунарских мрежа. Да би овај циљ био остварен потребно је развити графички наменски језик за моделовање конкретних рачунарских мрежа и генератор конфигурационих скрипти. Помоћу развијеног графичког наменског језика треба омогућити моделовање мрежних уређаја и веза између њих. Такође, потребно је развити и генератор који ће, на основу података из модела специфицираног помоћу наменског језика, генерисати конфигурационе скрипте, односно извршне текстуалне датотеке. Покретањем генерисаних скрипти на конкретним мрежним уређајима аутоматски би се подесиле њихове мрежне карактеристике.

2. АРХИТЕКТУРА СИСТЕМА

У овом поглављу биће описани делови развијеног софтверског алата, као и генератор конфигурационих скрипти развијен на основу њега. На слици 1. приказана је архитектура развијеног софтверског алата.

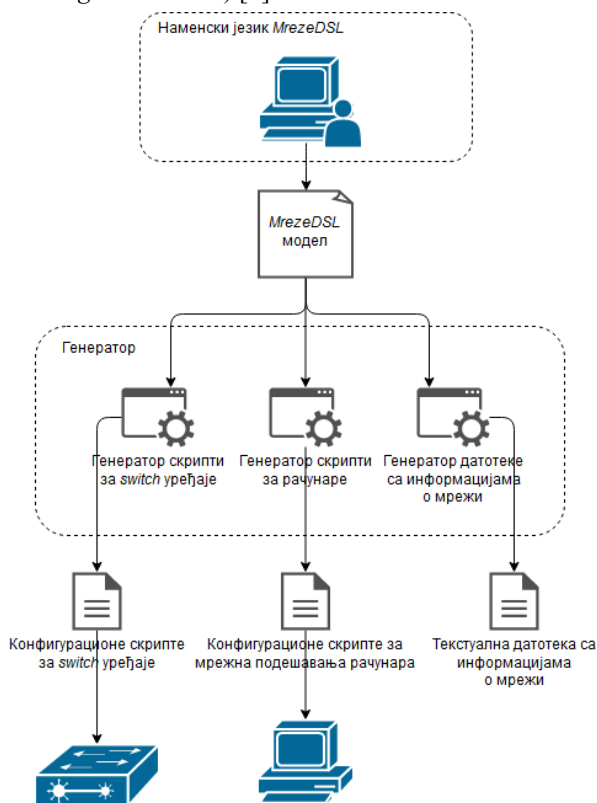
У оквиру овог рада развијен је графички наменски језик *MrezeDSL*. Помоћу графичког наменског језика *MrezeDSL* креира се *MrezeDSL* модел рачунарске мреже са жељеним изгледом и понашањем уређаја.

Графички наменски језик *MrezeDSL* развијен је у оквиру алата *Eclipse*. Имплементирани су алати за синтаксну и семантичку проверу модела, као ограничења дефинисана над мета-моделом. Мета-

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био др Иван Луковић, ред. проф.

модел који представља апстрактну синтаксу језика *MrezeDSL* имплементиран је у окружењу *EMF (Eclipse Modeling Framework)* [1].



Слика 1. Архитектура система

Конкретна синтакса имплементирана је у два облика, као текстуална и као графичка синтакса. Текстуални део конкретне синтаксе развијен је у алату *Xtext* [2], док је за графички део коришћено окружење *Sirius* [3].

У оквиру рада развијен је генератор за генерисање конфигурационих скрипти за одређене мрежне уређаје. Након креирања модела помоћу језика *MrezeDSL* корисник покреће генерисање конфигурационих скрипти. Генерисање скрипти представља *M2T (Model to Text)* трансформацију и своди се на трансформацију креираног модела рачунарске мреже у текстуалне датотеке.

Како је за један наменски језик могуће развити више различитих генератора, у раду су развијени генератор конфигурационих скрипти за *switch* уређаје, генератор скрипти за мрежна подешавања рачунара и генератор текстуалне датотеке са информацијама о рачунаској мрежи. Сви генератори кода написани су у програмском језику *Xtend* [4].

Покретањем скрипти на уређајима за које су предвиђене, уређајима се задаје жељена мрежна конфигурација без потребе за додатним подешавањима.

3. ЈЕЗИК ЗА ОПИС РАЧУНАРСКИХ МРЕЖА

Наменски језик поседује апстрактну и конкретну синтаксу. Апстрактна синтакса представља структуру и граматичка правила језика, док се конкретна синтакса бави нотацијом и репрезентацијом језика [5]. Апстрактна синтакса моделује концепте из домена на основу концепата мета-мета-модела. Избор нотације, конкретне синтаксе, прати актуелну презентацију

доменских концепата и идеално сваком концепту језика одговара једна репрезентација, као што је симбол. На основу развијене апстрактне синтаксе могуће је развити више конкретних синтакси различитих нотација. Приликом развоја конкретне синтаксе језика *MrezeDSL* коришћене су графичка и текстуална синтакса.

Концепти апстрактне синтаксе су:

- **Дијаграм** - представља коренски елемент мета-модела и одговара самом дијаграму који се креира;
- **Тип кабла** - дефинише тип кабла;
- **Атрибут** - дефинише додатне атрибуте за тип кабла;
- **Мрежа** - одговара рачунаској мрежи;
- **IP адреса** - одговара интернет адреси додељеној уређају или рачунаској мрежи;
- **Уређај на мрежи** - одговара уређајима који чине рачунарску мрежу;
- **Порт** - одговара порту уређаја на рачунаској мрежи;
- **Кабл** - одговара каблу који повезује уређаје на рачунаској мрежи;
- **OSI ниво два** - представља уређаје који припадају другом *OSI* нивоу;
- **OSI ниво три** - представља уређаје који припадају трећем *OSI* нивоу;
- **Hub** - представља уређај *hub*;
- **Switch** - представља подешавања која су заједничка за уређај *switch* другог и трећег *OSI* нивоа;
- **Switch2** - представља подешавања која су специфична за уређај *switch* другог *OSI* нивоа;
- **Switch3** - представља подешавања која су специфична за уређај *switch* трећег *OSI* нивоа;
- **WorkingStation** - представља уређај који је радна станица, односно персонални рачунар или нека врста сервера;
- **Рутер** - одговара уређају рутер;
- **Рутинг табела** - одговара рутинг табели одређених уређаја трећег *OSI* нивоа;
- **Рута** - одговара једном реду рутинг табеле;
- **VLAN** - представља *VLAN* подешавање *switch* уређаја.

Део конкретне синтаксе језика *MrezeDSL* који се односи на дефинисање типова каблова имплементиран је као текстуална конкретна синтакса. Тип кабла представља опис кабла и не приказује се посебно на дијаграму. Из тог разлога дефинисање типова каблова имплементирано је као текстуална синтакса. На слици 2. приказан је део конкретне текстуалне синтаксе којом су дефинисани типови каблова за пример *MrezeDSL* модела.

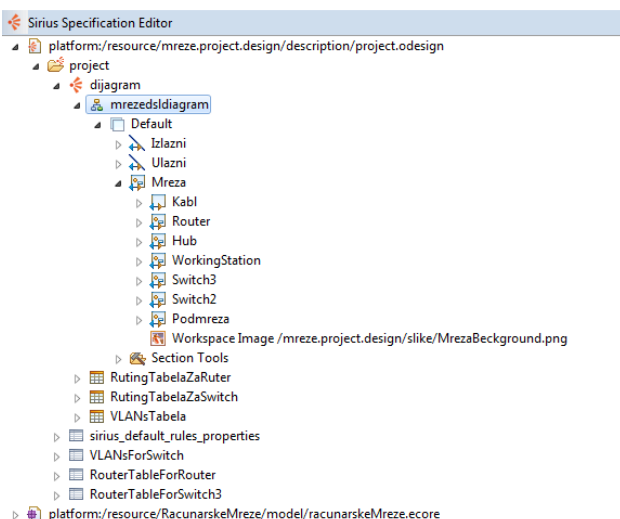
Алати за графичко моделовање углавном користе радну површину која омогућава позиционирање моделованих елемената у две димензије. Такав дијаграм репрезентује графички поглед на модел. Спецификација конкретне синтаксе наменског језика *MrezeDSL* приказан је на слици 3.

```

TipKablo {
  "crossover" {
    maxDuzina 100
    povezuje Iste
    PovezujePortove RJ_45
  },
  "straightthrough" {
    maxDuzina 100
    povezuje Razlicite
    PovezujePortove RJ_45
    atribut {
      {
        nazivAtributa "shielded"
        tipAtributa "boolean"
      }
    }
  },
  "koaksijalac" {
    povezuje Iste
    PovezujePortove koaksijalni
  },
  "opticki" {
    povezuje Iste
    PovezujePortove opticki
  }
}

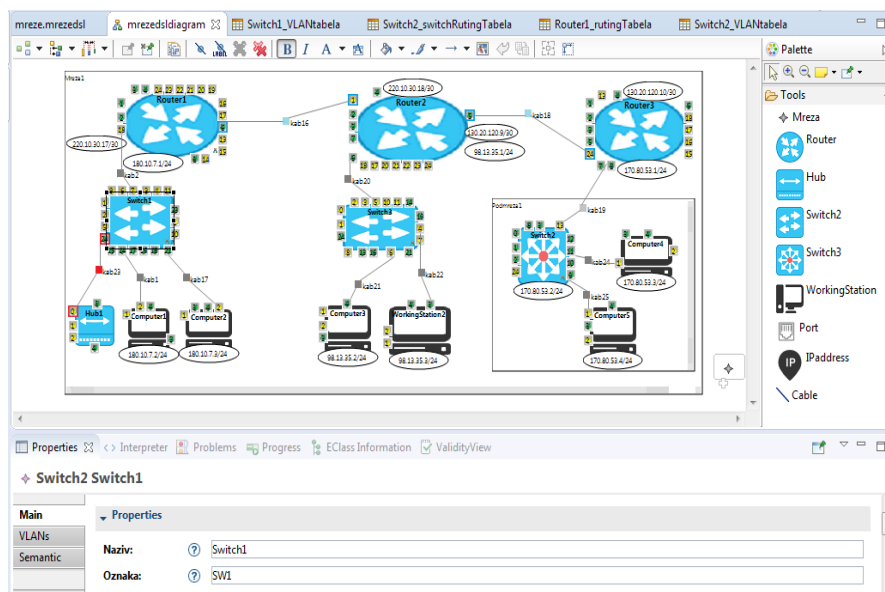
```

Слика 2. Конкретна текстуална синтакса



Слика 3. Спецификација графичке синтаксе наменског језика MrezeDSL

На слици 4. приказан је изглед радне површине креираног алата. Палета са алатима за креирање



Слика 4. Изглед радне површине креираног алата

елемената налази се са десне стране радне површине. У доњем делу радне површине налази се део за измену вредности атрибута селектованог елемента. Такође, креирани модел рачунарске мреже могуће је семантички валидирати по креирним доменским ограничењима.

Нека од креираних ограничења су ограничења јединствености назива мреже, јединствености ознаке каблова, уређаја на мрежи и IP адреса, као и ограничење да се само претходно слободни портови могу повезати каблом. Све интернет адресе у оквиру модела треба да буду задате у децималном запису. За уређаје на мрежи креирана су ограничења јединствених редних бројева портова.

Портови могу бити обележени само позитивним бројем, и само на портovima switch уређаја може бити дефинисано VLAN подешавање. Такође, постоји ограничење јединствености ознаке VLAN подешавања за сваки switch уређај. Кабл на основу свог дефинисаног типа кабла мора да повезује или уређаје истог или уређаје различитог OSI нивоа. Затим, кабл мора да повеже један улазни и један излазни порт и само врсту порта за коју је дефинисан по свом типу кабла.

У оквиру рада имплементирана је и детекција циклуса како би се могли уочити и отклонити потенцијални проблеми у рачунаској мрежи која има циклусе. За детекцију циклуса коришћена је претрага први у дубину, а модел рачунарске мреже је посматран као неусмерени граф.

Уређаји на рачунаској мрежи представљају чворове, а каблови везе између њих. Детекција циклуса имплементирана је у екстерној класи у програмском језику Java, а позива се из контекстног менија развијеног алата.

Резултат провере исписује се на дијалогу, уколико постоји пронађен циклус исписују се називи елемената сваког пронађеног циклуса.

4. ГЕНЕРАТОР КОНФИГУРАЦИОНИХ СКРИПТИ

За сваки *switch* уређај на мрежи генерише се скрипта као датотеке са *.cfg* и *.txt* екстензијама како би се подешавања могла покренути на више начина. Садржај скрипте представљају *Cisco IOS* команде којима се постиже жељено понашање уређаја задато у креираном моделу. Скрипте је довољно само покренути на правом *Cisco switch* уређају и подешавања ће бити учитана и сачувана. Тиме се уређај подешава онако како је означено у графичком алату, без потребе за познавањем самих команди и процеса конфигурације што у многоструко олакшава цео процес. На слици 5. приказан је део генерисане конфигурационе скрипте за *switch* уређај.

Поред конфигурационих скрипти за *switch* уређаје генеришу се и скрипте са основним интернет подешавањем рачунара за сваки такав уређај на мрежи. У скрипти се подешава мрежни интерфејс уређаја, односно његова *IP* адреса и маска. На крају генерише се и сама спецификација рачунарске мреже као текстуална датотека. Она садржи податке о дефинисаној рачунарској мрежи као што су назив и адреса мреже, број уређаја, број каблова и листу уређаја на мрежи. У листи уређаја за сваки од њих пишу њихове карактеристике.

```
10 configure terminal
11
12 interface vlan 1
13 ip address 180.10.7.1 255.255.255.0
14 exit
15
16
17 vlan 5
18 name five
19
20 interface vlan 5
21 ip address 180.10.7.1 255.255.255.0
22 exit
23
24 spanning-tree vlan 5 priority 0
25
26 vlan 6
27 name six
28
29 interface vlan 6
30 ip address 180.10.7.1 255.255.255.0
31 exit
32
33 spanning-tree vlan 6 priority 0
34
35 no vlan 6
36 end
37
38 configure terminal
39 interface FastEthernet 0/8
40 switchport access vlan 5
41 end
42 configure terminal
43 interface FastEthernet 0/15
44 switchport access vlan 5
45 end
```

Слика 5. Део генерисане конфигурационе скрипте

5. ЗАКЉУЧАК

Приликом подешавања рачунарских мрежа за сваки уређај потребно је подесити његове карактеристике на мрежи. Повезивањем већег броја уређаја и проширивањем мреже конфигурирање рачунарске мреже постаје сложеније, а ручним конфигурирањем расте могућност појаве грешке.

На основу искуства стеченог током израде алата описаног у овом раду, закључује се да се за поједностављивање и аутоматизацију процеса

подешавања уређаја рачунарске мреже може користити графички наменски језик. На основу модела специфицираног помоћу наменског језика *MrezeDSL* генеришу се конфигурационе скрипте за подешавање мрежних карактеристика повезаних уређаја. Предност коришћења наменског језика представља и могућност укључивања доменских експерата у развој како би развијени алат био што прилагођенији стварној потреби корисника. Наменски језик уједно сакрива техничке детаље имплементације и омогућава корисницима, који не морају познавати конкретне наредбе уређаја, лако конфигурирање рачунарске мреже.

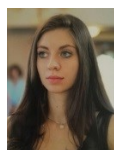
У овом раду описан је алат заснован на основном конфигурирању *switch* уређаја компаније *Cisco*. Да би описани алат доживео ширу практичну примену потребно је проширити мета-модел. Увођењем нових концепата и атрибута у мета-модел, била би омогућена потпуна конфигурација *switch* уређаја, али и уређаја трећег *OSI* нивоа, превасходно рутера. Такође, проширивањем мета-модела може се подржати концепт бежичне рачунарске мреже. На основу ових проширивања, поред имплементираних детекције циклуса у рачунарској мрежи, биле би омогућене и комплексније анализе мреже за којима постоји реална потреба. Проширивањем генератора скрипти било би могуће подржати и уређаје других произвођача мрежне опреме.

Пожељно би било имплементирати и додатне функционалности попут аутоматског задавања *IP* адреса повезаним уређајима на основу адресе мреже, као и проверу опсега адреса у локалном сегменту рачунарске мреже. Неопходно је подржати нову стандардну *IP* адресу верзије шест, наследника верзије четри, која решава проблем исцрпљености адреса претходне верзије. Такође, могла би бити подржана и функционалност аутоматског попуњавања рутинг табеле на основу моделованих веза између уређаја. Оваквим функционалностима поступак моделовања рачунарске мреже био би додатно убрзан.

6. ЛИТЕРАТУРА

- [1] Eclipse Modeling Framework, Доступно на: <https://eclipse.org/modeling/emf/>, август 2018.
- [2] Xtext, Доступно на: <https://www.eclipse.org/Xtext/>, август 2018.
- [3] Sirius, Доступно на: <https://www.eclipse.org/sirius/>, август 2018.
- [4] Xtend, Доступно на: <https://www.eclipse.org/xtend/>, август 2018.
- [5] M. Brambilla, J. Cabot, M. Wimmer: Model-Driven Software Engineering in Practice, Morgan & Claypool Publishers, 2012.

Кратка биографија:



Јелена Јаковљевић рођена је у Новом Саду 1992. године. Факултет техничких наука, одсек Рачунарство и аутоматика, уписала је 2011. године. Основне академске студије завршила је 2015. године.

**ARHITEKTURA I IMPLEMENTACIJA SISTEMA ZA ANALIZU PODATAKA
PAMETNOG GRADA****AN ARCHITECTURE AND IMPLEMENTATION OF A SYSTEM FOR SMART CITY
DATA ANALYSES**

Simonida Kadić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu prezentovan je sistem skladišta podataka čijom je upotrebom moguće vršiti analize kretanja ljudi na području grada Melburna u Australiji. U prvom delu rada opisani su izvori podataka. Podaci su prikupljeni iz baza podataka vladinih agencija grada Melburna. Dobavljanje podataka zahtevalo je razvoj podsistema za ekstrakciju podataka. Neophodno je bilo izvršiti eksplorativnu analizu prikupljenih podataka, kako bi se razvila OLTP šema baze podataka i programi za punjenje njenih tabela. Za projektovanje dimenzija i činjenica OLAP šeme baze podataka bilo je potrebno uočiti poslovne procese koji su od značaja za analizu kretanja ljudi. Takođe, razvijen je ETL proces čijom primenom se realizuje punjenje OLAP baze podataka. Izveštajna funkcija omogućava parametrizaciju upita.

Ključne reči: Sistemi skladišta podataka, OLTP, OLAP, ETL proces, Pametan grad.

Abstract – In this paper we present a data warehouse system that provides analyses of the movement of people in the city of Melbourne, Australia. In the first part of the paper we describe sources of data, provided by government agencies of the city of Melbourne. The extraction of data required the development of scraping programs. The development of the OLTP database schema and programs for its loading were based on exploratory data analysis. Next, a set of business processes was established to drive the development of dimensions and facts of the data warehouse. An ETL process was developed to load data into the data warehouse. A user interface was developed to allow customization of reports. Finally, the paper provides reports based on the collected data.

Ključne reči: Data warehouse system, OLTP, OLAP, ETL process, Smart city.

1. UVOD

U vreme rasta populacije već razvijenih gradova, tehnološke inovacije počinju da se primenjuju u korist boljeg svakodnevnog života građana. Podaci sa različitih senzora omogućavaju sprovođenje odluka koje proizilaze iz izveštaja iz prikupljenih podataka.

Uvođenje informacione tehnologije u proces praćenja različitih parametara bitnih za funkcionisanje gradova i njihovih promena, doprinelo je pojavi koncepta pametnog

grada (eng. Smart City) [1]. Pametan grad karakteriše primena mnogobrojnih senzora za prikupljanje podataka. Istraživanje je zasnovano na podacima preuzetim za grad Melburn u Australiji, kao primer gusto naseljenog grada koji je na visokom stepenu kulturnog i ekonomskog razvoja. Australijske državne agencije već duži vremenski period objavljuju skupove podataka kao odgovor na deklaraciju o otvorenoj vladi [2]. Omogućen je i pristup podacima u realnom vremenu sa mnogih javnih senzora. Decentralizacija podataka prikupljenih od strane različitih agencija onemogućuje jedinstven pristup podacima. Agencije objavljuju podatke u različitim formatima vremena i datoteka ili na odvojenim portalima. Podaci objavljeni na ovaj način moraju se dovesti u korelaciju i ujednačen format pre sprovođenja analiza. Količina prikupljenih podataka dodatno otežava njihovu analizu. Transakcije koje bi pružile izveštaje iz polaznih podataka ne mogu se izvršiti za vreme koje bi bilo adekvatno za praćenje podataka u skoro realnom vremenu. Zbog toga, podatke pojedinačnih izvora najpre je potrebno „očistiti“ od grešaka, a zatim ih dovesti u oblik koji bi omogućio kraće vreme generisanja izveštaja.

Različiti formati podataka, velike količine podataka, potrebe za „čišćenjem“ podataka i njihovo dovođenje u korelaciju, impliciraju potrebu za informacionim sistemom koji bi omogućio parametrizovano generisanje izveštaja. Ovaj rad opisuje postupak izgradnje informacionog sistema za analizu podataka pametnog grada. Podaci Melburna iskorišćeni su kako bi iz sistema proizišli realni izveštaji. Informacioni sistem pokriva proces praćenja saobraćaja automobila, pešaka, biciklista na nivou predgrađa. Cilj ovog rada je da se na osnovu prikupljenih podataka o kretanju ljudi dobiju informacije o gustini pešačkog i biciklističkog saobraćaja, kao i informacije o upotrebi parking mesta, u zavisnosti od obuhvaćenog vremenskog intervala, meteoroloških uslova i karakteristika posmatrane lokacije. Rezultati analiza prikupljenih podataka treba da odgovore na sledeća pitanja:

- Kako je kretanje pešaka, biciklista i upotreba parking mesta raspoređeno u toku različitih vremenskih intervala?
- Kako vremenski uslovi utiču na broj zabeleženih pešaka, biciklista i parkiranja?
- Kako je kretanje pešaka, biciklista i upotreba parking mesta raspoređeno na nivou zone grada?

Rezultati analiza mogu da ukažu na mesta koja zahtevaju više ili manje parking mesta, mesta čestih gužvi u saobraćaju ili mesta gde je potrebno podstaći saobraćaj pešaka i biciklista. Termini izvođenja radova mogu da se planiraju

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Ivan Luković, red. prof.

za periode kada će najmanje ometati saobraćaj. U toku izvođenja radova može se planirati preusmeravanje saobraćaja. Sistem omogućava procenu stanja u saobraćaju u toku radova i nakon završetka radova.

2. IZVORI PODATAKA

Podaci su preuzeti putem Internet stranica agencija vlade Australije. Sledeći skupovi podataka su preuzeti:

- podaci o predgrađima Melburna,
- podaci o poslovnim preduzećima,
- podaci o stambenim prostorima,
- podaci o parking sensorima,
- podaci o sensorima biciklističkih staza,
- podaci o meteorološkim stanicama,
- podaci o sensorima pešaka i
- podaci o sensorima kvaliteta vazduha.

Eksplorativnom analizom utvrđene su nepravilnosti u podacima. Do 33% senzora parking mesta nema zabeleženu lokaciju. Postoje očitavanja parkiranja koja nisu jedinstvena ili imaju negativnu dužinu trajanja parkiranja. Postoje parkiranja koja se sukcesivno nastavljaju tako da kraj jednog parkiranja je u trenutku početka sledećeg parkiranja. Utvrđeno je da se takve sekvence protežu kroz više sati, odnosno više dana pa čak i meseci. Sukcesivna parkiranja su u nekim slučajevima vrlo kratka i traju nekoliko sekundi, što doprinosi broju zabeleženih parkiranja u toku jednog sata. Na primer, na jednom parking mestu 19. januara 2012. u 11 sati zabeležena su 202 parkiranja. Očigledno je da takav broj parkiranja u toku jednog sata je malo verovatan, kao i da ovakve nepravilnosti bitno doprinose broju očitavanja sa senzora. Od 6.250 senzora čiji položaj je poznat, 3.125 senzora (50%) ima zabeleženo preko 2.037 parkiranja u toku 2016. godine. Neki senzori imaju očitavanja samo u toku jednog meseca. Takvi senzori su zabeležili ukupno preko 200 parkiranja, odnosno više parkiranja u intervalu od par meseci, nego senzori bez sekvencijalnih parkiranja u toku cele godine. Utvrđeno je da upravo takvi senzori imaju sukcesivna parkiranja.

U početnim podacima za 43 senzora pešaka postoji 390.423 zapisa koja pokazuju koliko pešaka je zabeleženo u toku predhodnog sata u 2016. godini. Idealno, svaki senzor bi imao 8.784 zapisa za svaki sat u toku 2016. godine, odnosno 24 zapisa po danu. Utvrđeno je da postoje nepravilnosti zbog kojih broj zapisa ni na jednom senzoru ne odgovara ovoj pretpostavci. Ove nepravilnosti nastaju usled dupliranja zapisa, nepostojeci zapisa i lošeg rešenja za letnje računanje vremena.

Duplikate čini do 8% očitavanja pešačkih senzora u 2016. godini. Nakon uklanjanja duplikata, senzori imaju najviše 8.783 očitavanja u toku 2016. godine. 3. Aprila 2016. godine prestalo je letnje računanje vremena i časovnik je pomešten sa 3 (tri) na 2 (dva) sata. U toku ovog datuma nema ni višak ni manjak zapisa. Ovde je problematično to što dolazi do gubljenja podataka jednog sata, s obzirom na to da vreme od 2 (dva) sata nastupa dva puta u toku tog dana: prvi put po letnjem računanju vremena, a zatim i po zimskom računanju vremena. Takođe, ne zna se da li zapisi očitani u 2 (dva) sata ujutru odgovaraju letnjem ili zimskom računanju vremena.

Senzori pešaka koja imaju očitavanja do poslednjeg dana u godini, a prvo očitavanje pre oktobra, nemaju očitavanje

2. oktobra 2016. godine u 2 sata. Ovo je sat koji nedostaje sensorima koji imaju 8.783 umesto 8.784 očitavanja nakon uklanjanja duplikata. Upravo 2. oktobra 2016. godine u 2 sata počelo je letnje računanje vremena i časovnik je pomešten za jedan sat u napred. Za taj datum i vreme nema zapisa ni za jedan senzor. Međutim, 1 (jedan) sat po zimskom računanju vremena i 3 (tri) sata po letnjem računanju vremena preslikavaju se na susedne sate po griničkom vremenu, zbog čega takvi zapisi ne treba ni da postoje.

Navedene nepravilnosti utiču na rezultate izveštaja, zbog čega se moraju očistiti putem *ETL* procesa. Detaljniji opis izvora podataka prikazan je u master radu autora.

3. ŠEMA OLTP BAZE PODATAKA

OLTP baza podataka (eng. *On-Line Transaction Processing*, transakciono procesiranje podataka) predstavlja izvor podataka za OLAP (eng. *On-Line Analytical Processing*, analitičko procesiranje podataka) bazu podataka. Sastoji se od sledećih tabela: tabela predgrađa - sadrži nazive i granice predgrađa, tabela stambenih prostora - sadrži podatke sa popisa stambenih prostora, tabela poslovnih preduzeća - sadrži podatke sa popisa poslovnih preduzeća, tabela pešačkih senzora - sadrži podatke o pešačkim sensorima, uključujući lokaciju i identifikator senzora, tabela očitavanja pešačkih senzora - sadrži očitavanja pešačkih senzora uključujući identifikator senzora, tabela lokacija biciklističkih senzora - sadrži podatke i identifikatore lokacija na biciklističkim stazama gde su postavljena najčešće dva senzora koja su usmerena u suprotnim smerovima, tabela biciklističkih senzora - sadrži podatke i identifikatore biciklističkih senzora sa identifikatorom lokacije na kojoj se nalaze, tabela očitavanja biciklističkih senzora - sadrži podatke o očitavanjima biciklističkih senzora, uključujući identifikator lokacije senzora i identifikator senzora, tabela lokacija parking senzora - sadrži identifikatore i lokacije na kojima se nalazi više parking senzora, tabela očitavanja parking senzora - sadrži podatke očitavanja parking senzora, uključujući identifikator lokacije senzora i identifikator parking senzora, tabela meteoroloških stanica - sadrži podatke o mestima očitavanja meteoroloških uslova, uključujući identifikatore meteoroloških stanica, tabela očitavanja meteoroloških stanica - sadrži podatke o očitavanjima meteoroloških uslova, uključujući opis očitano parametra i identifikator meteorološke stanice, tabela mesta očitavanja kvaliteta vazduha - sadrži identifikatore i lokacije na kojima se nalazi više senzora kvaliteta vazduha, tabela senzora kvaliteta vazduha - sadrži podatke i identifikatore senzora kvaliteta vazduha kao i identifikator lokacije merenja na kojoj se nalazi senzor, tabela vremenskih osnova očitavanja kvaliteta vazduha sadrži podatke o intervalima očitavanja parametara kvaliteta vazduha, uključujući identifikator senzora i identifikator lokacije očitavanja, tabela očitavanja kvaliteta vazduha - sadrži podatke o očitavanjima parametara kvaliteta vazduha, uključujući identifikator vremenske osnove očitavanja, identifikator senzora i identifikator lokacije očitavanja.

Detaljna specifikacija šeme OLTP baze podataka prikazana je u master radu autora.

4. ŠEMA OLAP BAZE PODATAKA

Šema *OLTP* baze podataka predstavlja izvor podataka za izveštajnu funkciju. Njen sadržaj obezbeđuje *ETL* proces. Sastoji se od tabela dimenzija i tabela činjenica. Dimenzije čine vremenska dimenzija, dimenzija lokacije, dimenzija meteoroloških uslova i dimenzija kvaliteta vazduha. Činjenične tabele opisuju pešački saobraćaj, biciklistički saobraćaj, automobilski saobraćaj i upotrebu prevoznih sredstava. Tabela 1 prikazuje matricu Poslovni procesi / Dimenzije (tzv. *Bus Matrix*), u okviru koje su izdvojeni ključni procesi poslovanja. Prikazani procesi poslovanja su predmet projektovanja skladišta podataka. Tabela prikazuje i dimenzije po kojima se posmatraju poslovni procesi.

Detaljna specifikacija šeme *OLAP* baze podataka data je u master radu autora.

Tabela 1. BUS Matrix

Dimenzija					
Poslovni proces	Vreme	Meteorološki uslovi	Predgrade	Senzor	Kvalitet vazduha
Praćenje upotrebe prevoznih	X	X	X		
Praćenje biciklističkog saobraćaja	X	X	X	X	
Praćenje kretanja pešaka	X	X	X	X	
Praćenje kvaliteta vazduha	X	X	X	X	
Praćenje upotrebe parking mesta	X	X	X	X	X

5. ETL PROCES

Na slici 2. prikazan je tok podataka kroz sistem, od sistema agencija do korisničkih servisa sistema. Programi za prikupljanje podataka unose podatke u *OLTP* bazu podataka. *ETL* proces obrađuje podatke iz *OLTP* baze podataka i beleži ih u *OLAP* bazu podataka. Korisnički servisi prikazuju rezultate analiza nad podacima u *OLAP* bazi podataka na osnovu zadatih kriterijuma.

Putem programa za prikupljanje podataka ekstrahuju se podaci sa servisa agencija. Kako bi se smanjila količina podataka koja se preuzima preko Interneta, podaci se snimaju u sistemu datoteka. Programi zatim prenose podatke u *OLTP* bazu podataka. Prenos može da bude ostvaren putem *Python* ili *SQL* skripti.

Upute nad *OLTP* bazom podataka karakteriše upotreba geoprostornih metoda. U takve upite spada poređenje položaja dve tačke na karti, koje mogu predstavljati građevinske objekte, senzore za prikupljanje podataka o saobraćaju ili poslovna preduzeća. Međutim, pokazalo se da su upiti vremenski zahtevni kada podrazumevaju veliki

broj provera preseka dva mnogougla, ili provera prisustva tačke unutar mnogougla.

Razvoj *ETL* procesa zahtevao je vremena za istraživanje mogućnosti korišćenja adekvatnog programskog alata. Microsoft integracioni servisi i servisi za analizu ne podržavaju geoprostorni tip podataka. Dalje unapređenje *ETL* procesa podrazumeva pronalaženje programskog paketa koji podržava geoprostorni tip podataka.

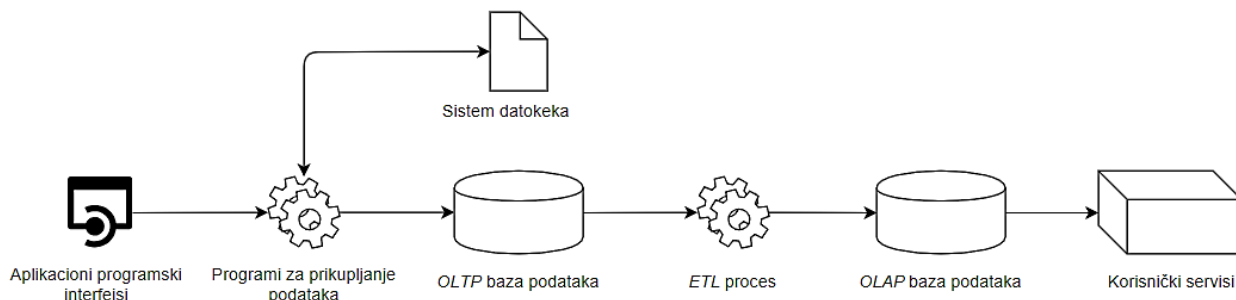
Željeni nivo kvaliteta je postignut usaglašavanjem formata različitih senzora. Projektovanjem ograničenja u *OLTP* bazi podataka sprečen je unos nekonzistentnih podataka u sistem (na primer upisivanje očitavanja nepostojećeg senzora). Radi ostvarenja polaznog cilja rada, analitička funkcija sistema zahteva da se upiti nad *OLAP* bazom podataka izvršavaju brže nego nad *OLTP* bazom podataka. *ETL* proces i *OLAP* baza podataka smanjuju broj spajanja tabela i poređenja kroz agregaciju podataka i kategorizaciju. U zahtevne procedure spadaju prevođenje atributa u geoprostorni tip i poređenje lokacija stanica i granica predgrađa. *ETL* proces omogućava da se ovakve procedure izvrše jednom, i to pre potrebe za izvršenjem upita.

Detaljna specifikacija *ETL* procesa prikazana je u master radu autora ovog rada.

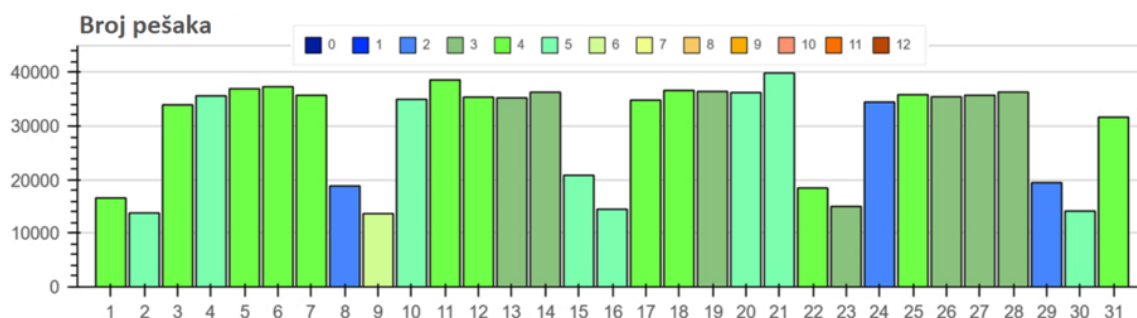
6. IZVEŠTAJI

Dinamički izveštaji imaju prednost u odnosu na statičke izveštaje, s obzirom da omogućavaju prilagođavanje upita izveštaja potrebama korisnika. Na ovaj način, korisnici bez informatičkog znanja i iskustva ne moraju da se upoznaju sa tehnologijom generatora izveštaja, kako bi prilagodili upite svojim potrebama. Sa druge strane, programeri nemaju potrebu za ručnim implementiranjem pojedinih tipova izveštaja.

Slika 3 prikazuje broj zabeleženih pešaka po danu u toku oktobra 2016. godine na senzoru u centru Melburna. Boje stubova grafikona odgovaraju maksimalnoj kategoriji vetra. Legenda povezuje boje sa kategorijama vetra. Sa slike može se utvrditi da maksimalne kategorije vetra su se kretale u intervalu od drugog do šestog stepena. 26. januar beleži manje pešaka, pošto je u pitanju državni praznik „Dan Australije“. Većina zaposlenog stanovništva tada koristi slobodan dan, državne ustanove i prodavnice su zatvorene ili imaju skraćeno radno vreme, a pojedini gradski servisi javnog prevoza ne rade. Pored ovog dana u godini, i prvi dan godine (1. januar) beleži manje pešaka. Na osnovu izveštaja o kretanju pešaka, mogu da se planiraju lokacije promocija, anketiranja ili poslovnih preduzeća. Preduzeća mogu da procene vrednost lokacija poslovanja, potreban broj radnika ili potrebu za obezbeđenjem.



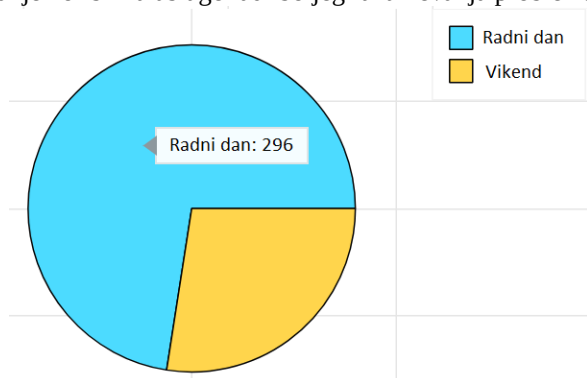
Slika 2. Tok podataka kroz sistem



Slika 3. Izveštaj o broju pešaka po danu u toku oktobra 2016. godine

Marketing može da se prilagodi izveštajima kako bi se privuklo što više kupaca.

Slika 4. prikazuje udeo parkiranja u prekršaju koji su zabeleženi u toku 2016. godine. Grafikon prikazuje da u toku 2016. godine, više od četvrtine prekršaja pri parkiranju zabeleženo je u toku vikenda. Može se zaključiti da je potrebno podstaći plaćanje parkinga vikendom, na primer kroz češće kontrole. Izgradnjom infrastrukture za bicikliste u blizini mesta gde se dešavaju prekršaji može da se podstakne upotreba bicikla u mesto automobila. Ovo je primer izveštaja koji može da podstakne anketiranje korisnika usluge radi boljeg razumevanja problema.



Slika 4. Izveštaj o broju parkiranja u prekršaju u toku 2016. godine, podeljeno na radne dane i vikend

S obzirom da implementirani izveštaji o kretanju pešaka, biciklista i upotrebi parking mesta proizilaze iz realnih podataka vladinih agencija Melburna, moguća je njihova upotreba pri planiranju radova na saobraćajnoj infrastrukturi i praćenje njihovog uticaja na saobraćaj. Izveštaji omogućavaju procenu isplativosti radova na saobraćajnoj infrastrukturi. Takođe, sistem omogućava praćenje uticaja manifestacija na saobraćaj i organizaciju bezbednosti u gradu.

7. ZAKLJUČAK

U ovom radu predstavljen je sistem skladišta podataka sa izveštajnom funkcijom koji omogućava uvid u kretanje građana grada Melburna. Podaci uključuju očitavanja senzora pešaka, biciklista, parkinga, meteoroloških uslova i kvaliteta vazduha, zatim podatke o granicama predgrađa, kao i podatke o poslovnim preduzećima i stambenim prostorima. Izvori podataka su portali vladinih agencija i API servisi. Podaci koji potiču sa servera zahtevaju programe kako bi se snimili i preneli u OLTP bazu podataka. Količina zabeleženih podataka nije prikladna za analizu nad datotekama ili nad OLTP sistemom.

Eksplorativna analiza prikupljenih podataka otkrila je nepravilnosti u prikupljenim podacima. Manifestuju se u

obliku duplikata ili neispravnih zapisa, što doprinosi još većoj količini zabeleženih podataka sa senzora. Kod pojedinih senzora postoje „rupe“ u podacima usled prekida rada senzora. Uočena je i nekonzistentnost u formatiranju vremena i lokacije kod različitih senzora. Poseban problem čine senzori koji beleže očitavanja u lokalnom vremenu, iz razloga što propuštaju sate pri prelasku iz letnjeg u zimsko računanje vremena.

Formirana je OLTP šema baze podataka koja je namenjena čuvanju prikupljenih podataka. Uočeni su poslovni procesi, kao i dimeznije po kojima se prate. Formirana je šema OLAP baze podataka, koja kasnije ispunjava ulogu izvora podataka za izveštaje.

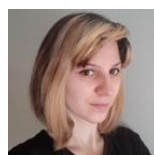
Prenos podataka iz OLTP baze podataka u OLAP bazu podataka zahteva implementaciju ETL procesa koji uzima u obzir navedene nedostatke u podacima. Izveštajna funkcija implementirana je na Web platformi, što omogućava dinamičko generisanje izveštaja i dostupnost izveštaja velikom broju uređaja. Predlaže se beleženje vremena koje je nezavisno od vremenske zone (griničkom vremenu), ili u Unix Epoch vremenu.

Dalji razvoj projekta podrazumeva implementaciju sistema skladišta podataka putem Azure SQL Data Warehouse servisa. Punjenje skladišta podataka obezbedio bi Azure Databricks servis, na osnovu podataka iz Azure Data Lake Store servisa. Prenos sistema na Azure service omogućava bolje performanse i veću dostupnost sistema, kao i jednostavniji razvoj i održavanje. Sistem nema pristup bazama podataka agencija, već ima ograničen pristup Internet servisima agencija. Istorijski podaci senzora pešaka agregirani su na jedan sat, a sat pri prelasku iz letnjeg u zimsko računanje vremena je preskočen. Zbog ovoga, podaci nekih senzora u OLTP bazi podataka već su agregirani. U cilju postizanja boljih rezultata, potrebno je razviti proces prenošenja podataka iz baza podataka agencija.

8. LITERATURA

- [1] Cocchia A. (2014) „Smart and Digital City: A Systematic Literature Review“, Dameri R. Rosenthal-Sabroux C. „Smart City“, pp 13-43, 2014
- [2] Melbourne Data, URL <https://data.melbourne.vic.gov.au>

Kratka biografija:



Simonida Kadić rođena je u Novom Sadu 1994. god. Fakultet tehničkih nauka upisala je 2013. god. Bečelor rad iz oblasti računarske grafike odbranila je 2017. god

UTICAJ SOLARNIH PANELA NA STRUJU KRATKOG SPOJA U NISKONAPONSKOJ DISTRIBUTIVNOJ MREŽI**IMPACT OF SOLAR GENERATORS ON THE SHORT-CIRCUIT CURRENT IN LOW VOLTAGE NETWORK**Teodora Gavrilov, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U radu se razmatra uticaj kućnih solarnih panela na vrednost struje kratkog spoja u niskonaponskoj distributivnoj mreži. Cilj je da se proceni pri kojoj vrednosti snage panela postoji značajan uticaj na rad zaštita u niskonaponskoj mreži. Razmatranja su ilustrovan na primeru jedne tzv. sremskom tipu naselja.

Ključne reči: relejna zaštita, struja kratkog spoja, solarni paneli

Abstract – This paper deals with the impact of home solar panels on the value of the short-circuit current in the low-voltage distribution network. The aim is to evaluate where the value of the panel has a significant impact on the operation of protection in the low voltage network. The considerations are illustrated in the case of one so-called. Srem type of settlement.

Keywords: Relay Protection, Short-Circuit Current, Solar Generators

1. UVOD

U radu se razmatra uticaj kućnih solarnih panela na vrednost struje kratkog spoja u niskonaponskoj distributivnoj mreži. Potrošači koji imaju ove solarne panele se ponašaju kao prosumeri, odnosno kao potrošači ali i proizvođači električne energije. Cilj je da se proceni pri kojoj vrednosti snage panela postoji značajan uticaj na rad zaštita u niskonaponskoj mreži. Razmatranja su ilustrovana na primeru sremskog tipa naselja. U glavi dva definisana je niskonaponska distributivna mreža. U glavi tri dati su osnovni pojmovi o solarnim panelima. Glava četiri sadrži teorijske osnove o kratkim spojevima u elektroenergetskim sistemima (EES) sa posebnim akcentom na distributivne mreže. U glavi pet definiše se zaštita od kratkih spojeva, kao i način zaštite transformatora od njih. U glavi šest prikazan je proračun struja kratkog spoja sa ciljem pokazivanja uticaja prosumera. Zaključci rada su predstavljeni u sedmoj glavi. Na kraju je data literature u glavi osam.

2. NISKONAPONSKA DISTRIBUTIVNA MREŽA

Delovi elektroenergetskih sistema kojima se električna energija preuzima sa prenosnih mreža i raspodeljuje potrošačima jesu distributivne mreže. Električna energija se, u srpskom i evropskim EES, iz trofazne prenosne mreže vrlo visokog napona, Označena sa VVN, 400 i 220 kV preuzima trofazni

transformatorima 400/115 i 220/115 kV/kV. Trofazni vodovi 110 kV jesu visokonaponski vodovi, koji se napajaju sa tih transformatora. Oni čine sub-prenosnu mrežu. Električna energija se sa trofazne sub-prenosne mreže preuzima trofaznim napojnim transformatorima 110/21(10,5) kV/kV. Sub-prenosna i distributivna mreža s tri naponska nivoa u Srbiji je 110 kV/20 kV/0,4 kV (npr. u Vojvodini) ili 110kV/10 kV/0,4 kV (npr. u Beogradu). Trofazni vodovi srednjeg napona 20(10) kV napajaju se sa trofaznih napojnih transformatora 110/21(10,5) kV/kV. Sa ovih vodova električna energija se preuzima trofaznim distributivnim transformatorima 20(10)/0,4 kV/kV. Na osnovu toga, s nominalnih napona niskonaponskih strana distributivnih transformatora od 0,4 kV prelazi se na 0,42 kV [1]. Individualni potrošači (stanovi, itd.) napajaju se trofazno električnom energijom sa ovih niskonaponskih vodova.

3. DISTRIBUIRANI GENERATORI

Elektroenergetski sistemi pre uvođenja slobodnog tržišta električne energije, sastojali su se iz proizvodnog sistema, prenosnog sistema, distributivnog sistema i potrošnje i njihovo uređenje i upravljanje je bilo vertikalno. U takvom EES-u, tok energije u distributivnom sistemu tekao je samo u jednom smeru, tj. od mesta preuzimanja energije (transformatorske stanice između prenosnog i distributivnog sistema) preko uglavnom radialnih vodova i transformatorskih stanica do potrošača.

Nakon uvođenja slobodnog tržišta električne energije, osnovne komponente EES-a su ostale iste, ali je sada uređenje i upravljanje EES-om iz „vertikalnog” prešlo u „horizontalno”. Pored uvođenja slobodnog tržišta električne energije došlo je do razvoja distribuiranih generatora [2]. Razvojem distribuiranih generatora, koji se priključuju direktno na distributivnu mrežu, dolazi do promene smera tokova snaga u distributivnom sistemu. Pored promene tokova snaga u samoj distributivnoj mreži, dolazi do toga da distributivni sistem predaje aktivnu snagu u prenosni sistem. Distributivna mreža više nije samo pasivan deo, nego postaje „aktivan” deo EES-a.

Pod distribuiranim generatorima se podrazumevaju generatori manjih snaga, koji se priključuju direktno na distributivnu mrežu i koji za pogon koriste obnovljive izvore energije ili viškove energije iz nekog drugog tehnološkog (proizvodnog) procesa. Samo ime distribuirani generator nastalo iz činjenice da su izvori energije koje oni koriste rasprostranjeni po celom EES-u. Zbog navedenog, dolazi do pojave velikog broja malih elektrana koje nisu centralizovane nego su široko rasprostranjene po celom EES-u, zavisno od izvora energije koji koriste.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je dr Duško Bekut, redovni profesor.

Distribuirani generatori se razlikuju po vrsti energenta koji koriste. Vrste distribuiranih generatora koji su trenutno najčešće u upotrebi su: mala hidroelektrana, mini hidroelektrana i mikro hidroelektrana, elektrana na vetar, solarna elektrana, termoelektrana, tj. toplana koja deo električne energije povremeno ili stalno isporučuje u mrežu, termoelektrana na biomasu, otpatke i gorivne ćelije.

3.1. Solarni Paneli

Solarna energija (energija Sunca) je najobilniji vid energije koji je dostupan. Sunce je neiscrpan izvor svetlosti i u budućnosti je neophodno razvijati načine korišćenja ovog oblika energije. Velika mana solarne energije je ta da ona nije konstantna tokom celog dana kao i da ima promenljivu prirodu tokom godine. Načini korišćenja solarne energije mogu da se podele u dva pravca, pretvaranje solarne energije u električnu energiju i pretvaranje solarne energije u termičku energiju [2].

Fotonaponska (FN) ćelija je osnova fotonaponskog sistema. Ona predstavlja p-n spoj koji pomoću fotonaponskog efekta sunčevo zračenje pretvara u električnu energiju. Sačinjena je od poluprovodničkog materijala (najčešće silicijuma). Usled apsorpcije sunčevog zračenja, u p-n spoju se javljaju parovi elektron-šupljina. Prilikom sunčevog zračenja unutar ili u blizini p-n spoja, unutrašnje električno polje razdvaja elektrone i šupljine. Pri tome se elektroni kreću prema n strani, a šupljine prema p strani p-n spoja, i kao posledica toga javlja se potencijalna razlika, odnosno napon. Kada se FN ćelija poveže u strujno kolo, kroz potrošač će poteći struja.

Više solarnih ćelija sjedinjenih u jednu celinu jeste solarni panel, a više panela sjedinjenih u jednu celinu predstavlja modul. U ovom radu, tokom analize uticaja solarnih panela na eventualno prepođenje relejne zaštite, uzima se u obzir samo struja koju ovakvi paneli prosleđuju dalje u mrežu.

4. KRATAK SPOJ

Kratak spoj je slučajni ili nameran spoj provodnika, preko malog otpora ili impedanse, između dve ili više tačaka, koje su na različitim potencijalima, [1]. Karakteristike uređaja koji vrše prekidanje struja kratkog spoja zavise od veličine struje koju prekidaju i od vremenskog trenutka u kom se eliminiše kratak spoj. Poznavanje odvijanja prelazne pojave i veličine struje kratkog spoja koja se prekida od posebnog je značaja za podešavanje zaštitnih releja i izbor parametara prekidača i druge rasklopne opreme. Rezultati proračuna krakih spojeva se koriste kako pri projektovanju tako i pri eksploataciji i planiranju elektroenergetskog sistema. U distributivnim mrežama, kao i u elektroenergetskim sistemima, analiziraju se sledeće vrste kratkih spojeva: jednopolni kratki spoj, dvopolni kratki spoj bez zemlje, dvopolni kratki spoj sa zemljom i troploni kratki spoj sa i bez zemlje.

U radu se razmatra trolni kratki spoj. Trolni kratki spoj predstavlja simetričan slučaj kratkog spoja. Trolni kratki spoj se zbog svojih simetričnih karakteristika najlakše interpretira i proračunava. Na mestu kratkog spoja, fazni naponi su brojno jednaki na nulu i struja kratkog spoja opterećuje fazne vodove simetrično. Neutralni vod ili zemlja ne učestvuju u provođenju struje pa je struja kratkog spoja ista, nezavisno od toga da li je zvezdište mreže uzemljeno ili izolovano. Zbog navedenog, proračun struje kratkog spoja se

računa samo za jednu fazu. Verovatnoća pojavljivanja trolnog kratkog spoja je vrlo mala, te na mestu kratkog spoja u najvećem broju slučajeva ova vrsta kratkog spoja daje najveću struju kratkog spoja.

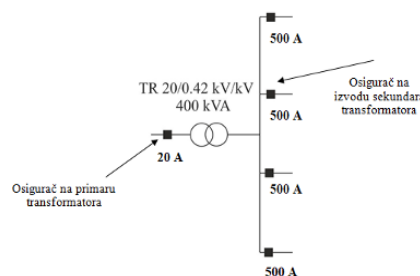
5. ZAŠTITA OD STRUJE KRATKOG SPOJA U NISKONAPONSKIM MREŽAMA

Zadatak zaštite od kratkog spoja jeste da eliminiše kratke spojeve i ograniči posledice, što znači da struje nastale kratkim spojem budu prekinute u što kraćem vremenu. Uređaji koji obezbeđuju samo zaštitu od struja kratkog spoja su uređaji koji mogu biti instalisani tamo gde se zaštita od preopterećenja postiže na drugi način, ili gde nije obavezna zaštita od preopterećenja. Ovi uređaji moraju da prekidaju struju kratkog spoja sve do vrednosti očekivane struje kratkog spoja na mestu ugradnje. Takvi uređaji su: prekidači sa prekostrujnim okidačem i osigurači [3].

Osigurač je uređaj koji služi za zaštitu od kratkog spoja i preopterećenja ili samo za zaštitu od kratkog spoja. Prekidanje strujnog kola se vrši topljenjem posebno oblikovanog topljivog elementa. Ugrađuje se u razvodnim ormanima niskog napona.

5.1. Zaštita Transformatora

U slučaju kratkih spojeva iza transformatora, kkratak spoj predstavlja značajno naprezanje za transformator preko koga se napaja mesto kratkog spoja. Ovi kratki spojevi se isključuju zaštitom smeštenom na niskonaponskoj strani (niskonaponski osigurači ili niskonaponski prekidači), ili zaštitom na visokonaponskoj strani ispred transformatora u slučaju da ne reaguje zaštita na niskonaponskoj strani. Za transformator u ovom radu, proračunate su nominalne struje primara i sekundara i spram njihovih nominalnih struja izabrani osigurači (slika 5.1).



Slika 5.1 – Zaštita transformatora

6. UTICAJ PROSUMERA NA KRATAK SPOJ I ZAŠTITU U SREMSKOM TIPU NASELJA

U ovom delu rada su definisani prosumeri i prikazan je sremski tip naselja. Takođe su i prikazani primeri uticaja prosumera na struju kratkog spoja.

6.1. Prosumeri

Prosumer je reč koja je nastala iz dve engleske reči: producer (proizvođač) i consumer (potrošač). U elektroenergetskom smislu, to je potrošač energije koji je u isto vreme i proizvođa (npr. zgrade, kuće ili domaćinstva, koji na svojim krovovima imaju solarne panele), tj. oni su istovremeno proizvođači i potrošači energije, što dovodi do dvosmernog toka energije. Prosumer je i jedinica koja se može posmatrati kao početak i kraj lanca snabdevanja energijom.

6.2. Sremski tip naselja

Sremski tip naselja obuhvata delove vangradskog naselja sa većom gustinom stanovanja, sa približno pravilnim rasporedom ulica i raskrsnica, spratnosti P i P+1.

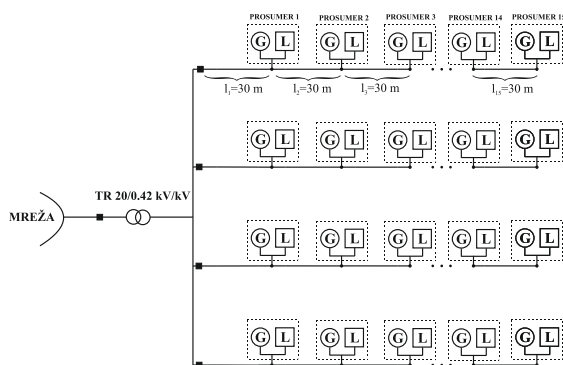
U ovom naselju, niskonaponska mreža je, po pravilu, nadzemna i povezuje susedne transformatorske stanice (TS), dok su transformatorske stanice slobodnostojeće ili stubne [4].

Za sremski tip stambenog naselja koriste se montažno betonska TS – MBTS (8 NN izvoda), stubne TS (4 NN izvoda) ili MTSP (4 NN izvoda).

Nadzemna niskonaponska mreža je na betonskim stubovima, povezuje susedne transformatorske stanice ili se gradi kao antenska. Granice razdvajanja mreža se izvode na stubu, na kome se obavezno povezuju neutralni provodnici. Po pravilu se koristi SKS, ali je dozvoljeno korišćenje i golih Al/Če užadi.

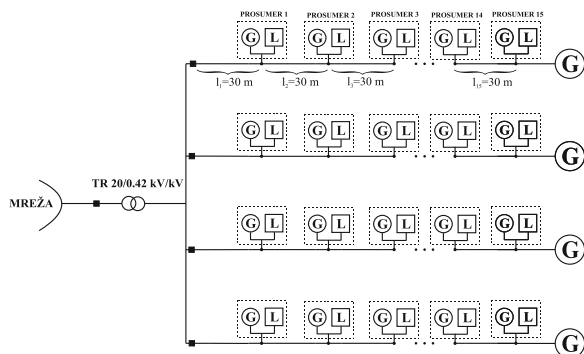
6.3. Primer uticaja prosumera

Primer uticaja prosumera na struju kratkog spoja prikazana je na delu niskonaponske mreže. Razmatrana je jednostavna distributivna mreža sa kratkim spojem na početku i kraju izvoda. Distributivna mreža je naponskog nivoa 0.42 kV, prikazana na slici 6.1.



Slika 6.1 – Razmatrani deo niskonaponske mreže

Razmatran je i slučaj kada su na krajevima izvoda priključeni veliki solarni paneli, koji proizvode 50 kW. Ovaj slučaj je prikazan na slici 6.2.



Slika 6.2 – Razmatrani deo niskonaponske mreže sa velikim solarnim panelima na krajevima izvoda

U radu su razmatrani sledeći slučajevi:

- prosumeri proizvode 1kW, i troše 5kW,
- prosumeri proizvode 2kW, i troše 5kW i
- prosumeri proizvode 5kW, i troše 5kW.

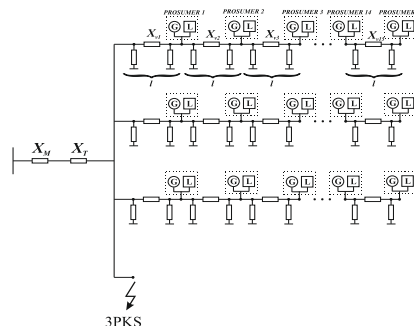
Posmatran je trolpolni kratak spoj na početku i na kraju izvoda za svaki od ova 3 slučaja.

6.3.1. Kratak spoj na početku izvoda

Prvo je posmatran slučaj kada na krajevima izvoda nisu vezani veliki solarni paneli (slika 6.3).

Struja trolpolnog kratkog spoja pri insertovanim generatorima je zbir stvarne struje trolpolnog kratkog spoja pre insertovanja i struje sume svih struja koje odaju solarni generatori pri kratkom spoju.

Za struju trolpolnog kratkog spoja pre insertovanja dobijeno je 15.961 kA.

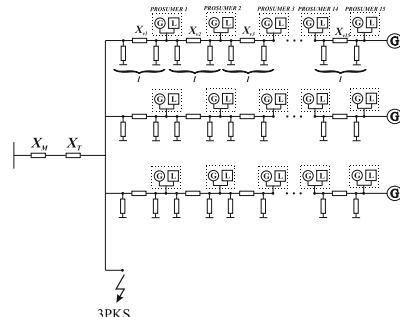


Slika 6.3 – Mreža bez velikih solarnih panela na kraju izvoda kada je kratak spoj na početku izvoda

Izračunate su sume svih struja koje odaju generatori pri kratkom spoju. Na svakom od 3 izvoda ima po 15 prosumera, te je ukupna struja prosumera pri kratkom spoju na početku izvoda jednaka 195.6 A (ako je proizvodnja prosumera 1 kW), 391.3 A (ako je proizvodnja prosumera 2 kW) i 978.2 A (ako je proizvodnja prosumera 5 kW). Kada se te vrednosti dodaju na izračunatu struju kratkog spoja, dobija se:

- ako je proizvodnja prosumera 1 kW:
 $I_{3PKS} + I_{3PKS\ 1kW} = 16.1\text{ kA},$
- ako je proizvodnja prosumera 2 kW:
 $I_{3PKS} + I_{3PKS\ 2kW} = 16.3\text{ kA},$
- ako je proizvodnja prosumera 5 kW:
 $I_{3PKS} + I_{3PKS\ 5kW} = 16.9\text{ kA}.$

Drugi slučaj je kada su za krajeve izvoda priključeni veliki paneli, koju proizvode snagu 50 kW (slika 6.4).



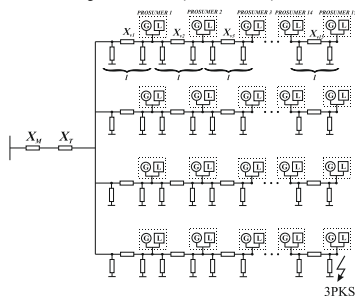
Slika 6.4 – Mreža sa velikim solarnim panelima na kraju izvoda kada je kratak spoj na početku izvoda

Struja koju veliki solarni panel proizvodi iznosi 68.7758 A. Na svakom od 3 izvoda ima po jedan veliki solarni panel, ta je ukupna struja velikih solarnih panela jednaka 206.327 A. Izračunata vrednost se sabira sa već izračunatom strujom kratkog spoja i dobija se:

- ako je proizvodnja prosumera 1 kW:
 $I_{3PKS} + I_{3PKS\ 1kW} + I_{3PKS\ 50kW} = 16.3\text{ kA},$
- ako je proizvodnja prosumera 2 kW:
 $I_{3PKS} + I_{3PKS\ 2kW} + I_{3PKS\ 50kW} = 16.5\text{ kA},$
- ako je proizvodnja prosumera 5 kW:
 $I_{3PKS} + I_{3PKS\ 5kW} + I_{3PKS\ 50kW} = 17.1\text{ kA}.$

6.3.2. Kratak spoj na kraju izvoda

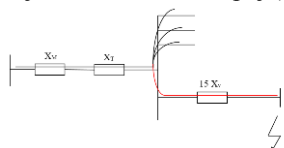
Posmatran je slučaj kada u mreži nema velikih solarnih panela vezanih za kraj izvoda (slika 6.5).



Slika 6.5 – Mreža bez velikih solarnih panela na kraju izvoda kada je kratak spoj na početku izvoda

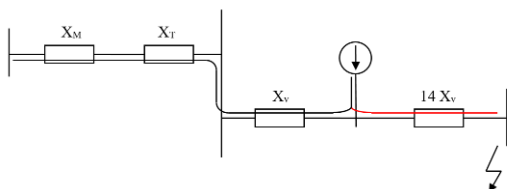
Struja trolnog kratkog spoja iznosi 2185.3 A.

Vrednosti struja koje proizvode prosumeri su ostale iste kao u slučaju kada je kvar na početku izvoda. Deo ukupne struje svih prosumera prolazi kroz granu u kojoj su ekvivalentna impedansa mreže kao i impedansa transformatora i deo kroz izvod na čijem kraju se računa kratak spoj (slika 6.6).



Slika 6.6 – Strujni razdelnik za ukupnu struju koju proizvode prosumeri iz izvoda na čijem kraju nije KS

Tražena vrednost struje kroz izvod na čijem kraju se računa kratak spoj dobija se relacijom strujnog razdelnika. Struju prosumera priključenih na poslednji izvod se takođe računa preko strujnog razdelnika iz istog razloga (slika 6.7).



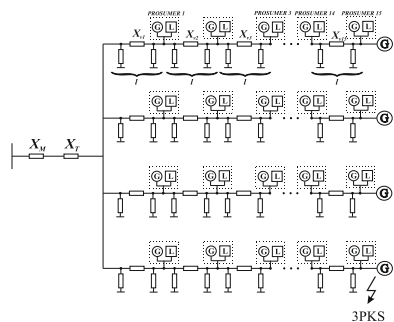
Slika 6.7 – Strujni razdelnik za struju prosumera koji su priključeni na izvod na čijem je kraju kratak spoj

Kada se dobijene vrednosti dodaju na izračunatu struju kratkog spoja, dobija se ukupna struja kratkog spoja za slučaj kada na krajevima izvoda nema velikih solarnih panela:

- ako je proizvodnja prosumera 1 kW:
 $I_{3PKS} + I_{3PKS\ 1kW} + I_{3PKS\ 1kW\ izvod\ sa\ KS} = 2257.1\ A,$
- ako je proizvodnja prosumera 2 kW:
 $I_{3PKS} + I_{3PKS\ 2kW} + I_{3PKS\ 2kW\ izvod\ sa\ KS} = 2328.9\ A,$
- ako je proizvodnja prosumera 5 kW:
 $I_{3PKS} + I_{3PKS\ 5kW} + I_{3PKS\ 5kW\ izvod\ sa\ KS} = 2544.4\ A.$

Na kraju, posmatran je slučaj kada su za krajeve izvoda priključeni veliki paneli, koji proizvode snagu 50 kW (slika 6.8).

Struja koju veliki solarni panel proizvodi je ostala ista kao kada je računata struju trolnog kratkog spoja i kada je kratak spoj bio na početku izvoda (68,7758 A). Kako na svakom od 3 izvoda ima po jedan veliki solarni panel, to je ukupna struja velikih solarnih panela nepromenjena (206,3A).



Slika 6.8 – Mreža sa velikim solarnim panelima na kraju izvoda kada je kratak spoj na kraju izvoda

Veliki solarni panel priključen na izvod na kom se računa kratak spoj ne utiče na struju kratkog spoja.

Izračunata vrednost se sabira sa već izračunatom strujom kratkog spoja kada solarni paneli nisu vezani za krajeve izvoda:

- ako je proizvodnja prosumera 1 kW:
 $I_{3PKS} + I_{3PKS\ 1kW} + I_{3PKS\ 1kW\ izvod\ sa\ KS} + I_{3PKS\ 50kW} = 2463.3\ A,$
- ako je proizvodnja prosumera 2 kW:
 $I_{3PKS} + I_{3PKS\ 2kW} + I_{3PKS\ 2kW\ izvod\ sa\ KS} + I_{3PKS\ 50kW} = 2535.1\ A,$
- ako je proizvodnja prosumera 5 kW:
 $I_{3PKS} + I_{3PKS\ 5kW} + I_{3PKS\ 5kW\ izvod\ sa\ KS} + I_{3PKS\ 50kW} = 2750.6\ A.$

Kao što se može iz svih ovih primera zaključiti jeste da je uticaj prosumera na struje kratkih spojeva relativno mali i da ne postoje suštinska ograničenja za primenu solarnih panela u niskonaponskoj distributivnoj mreži.

7. ZAKLJUČAK

U ovom radu obrađena je tema uticaja prosumera instalisanih na individualna domaćinstva na struju kratkog spoja u sremskom tipu naselja. Proračunate su vrednosti struja trolnog kratkog spoja na početku i na kraju izvoda, sa različitim vrednostima proizvodnje prosumera, čime je pokazano da povećanjem proizvodnje prosumera raste i struja kratkog spoja, dok vrednost potrošnje prosumera ne utiče na vrednost struje kratkog spoja. Činjenica je da što je mesto kratkog spoja bliže početku izvoda, to je struja veća. Kao što se može zaključiti iz primera u ovom radu je da je uticaj prosumera na struje kratkih spojeva relativno mali i da ne postoje ni suštinska ograničenja za primenu solarnih panela u niskonaponskoj distributivnoj mreži, niti za promene u podešenju zaštita.

8. LITERATURA

- [1] V. Strezoski, *Analiza elektroenergetskih Sistema 1 i 2*, Fakultet tehničkih nauka Novi Sad, 2017.
- [2] V. Katić, I. Kapetanović, N. Sarajlić, *Obnovljivi izvori električne energije*, Fakultet tehničkih nauka Novi Sad, 2007.
- [3] EPS, *Tehničke preporuke broj 14*, EPS 2001.
- [4] Strahil Gušavac, *Osnovni principi projektovanja u mrežama srednjeg i niskog napona*, Fakultet Tehničkih Nauka Novi Sad, 2004.

Kratka biografija:



Teodora Gavrilov rođena je u Novom Sadu 1994. godine. Master rad je odbranila 2019. godine na Fakultetu tehničkih nauka iz Elektrotehnike i računarstva – Elektroenergetski sistemi.

**MODELOVANJE MODERNIH DISTRIBUTIVNIH ENERGETSKIH RESURSA
ZASNOVANIH NA UREĐAJIMA ENERGETSKE ELEKTRONIKE ZA POTREBE
PRORAČUNA KRATKIH SPOJEVA****MODELING OF MODERN DISTRIBUTION ENERGY RESOURCES BASED ON
ENERGY ELECTRONIC DEVICES FOR THE NEEDS OF SHORT-CIRCUIT
CALCULATION**

Miloš Spasić, *Fakultet tehničkih nauka, Novi Sad*

Oblast- ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je opisan model distributivne mreže (DM) sa velikim brojem modernih energetske resursa (DER). Predstavljene su vrste postojećih DER kao i njihovi modeli u proračunima režima s kratkim spojem. Akcenat rada je na modelima DER koji su na DM povezani uređajima energetske elektronike. Predstavljen je i unapređen algoritam sumiranja struja i korekcije napona (USSKN) kako bi se ovi modeli uključili u proračune režima s kvarom.

Ključne reči: Modeli modernih distribuiranih izvora, proračun kratkih spojeva, generalizovano delta kolo.

Abstract – This paper describes the model of a power distribution system with a large number of modern energy resources. Types of existing energy resources, as well as their models in short-circuit calculations are presented. The emphasis of this paper is on energy resource models connected to the network by energy electronic devices. Improved backward/forward sweep algorithm is introduced in order to incorporate these models into calculations of the fault mode of distribution networks.

Keywords: Models of distributed energy resources, short circuits calculation, generalized delta circuit.

1. UVOD

U poslednjih nekoliko godina dolazi do velikog razvoja i sve većeg korišćenja distribuiranih energetske resursa (DER) zasnovanih na uređajima energetske elektronike. Modeli ovih DER razlikuju se od modela sinhronih i asinhronih generatora. Kako je moderni DER odvojen od DM pomoću invertora, njegov model zavisi isključivo od upravljanja invertorom [2]. Pokazano je i da se modeli modernih DER ne mogu koristiti u tradicionalnim procedurama za izračunavanje režima s kratkim spojem zato što sadrže veliki broj mogućih kontrolnih strategija. Stoga je opisan koncept generalizovanog delta kola, koji omogućava integraciju DER zasnovanih na savremenim tehnologijama, u postojeće proračune kratkih spojeva u DM [1]. Stanje sistema s kvarom se dobija superpozicijom režima pre kvara, koje je poznato, i režima generalizovanog delta kola.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Luka Strezoski.

Takođe, većina razvijenih zemalja ima svoj pravilnik o pogonu distributivnih mreža, gde su propisani i FRT (Fault Ride Through) zahtevi. FRT zahtevima se određuje kako bi moderni DER trebalo da se ponašaju u slučaju pada napona na mestima njihovih priključenja na mrežu. U zavisnosti od specifičnih zahteva država, inventori DER se mogu podešavati na različite načine.

Cilj ovog rada je prikaz modela savremenih DER za potrebe proračuna kratkih spojeva u DM na koje je moguće primeniti različite FRT zahteve. Opisani model je integrisan u unapređenu proceduru sumiranja struja i korekcije napona za proračun režima aktivnih DM s kratkim spojem.

2. MODELI ELEMENATA DISTRIBUTIVNE MREŽE

U ovom delu, predstavljeni su pofazni modeli, odnosno pogonski reprezenti trofaznih uravnoteženih elemenata DM koji se koriste u proračunima kratkih spojeva.

2.1. Potrošači

Potrošači se u proračunima režima DM s kratkim spojem najčešće modeluju idealnim strujnim generatorima. Te struje se izračunavaju na osnovu poznatih snaga i napona iz režima pre kratkog spoja.

2.2. Sekcije vodova

Sekcije vodova se za potrebe proračuna kratkih spojeva zamenjuju svojim pogonskim Π kolima određenog redosleda, između kojih, usled uravnoteženosti, u domenu simetričnih veličina ne postoje sprege između pogonskih kola direktnog, inverznog i nultog redosleda [4].

2.3. Transformatori

Transformatori su uređaji koji povezuju DM različitih naponskih nivoa. Prelaskom u domen relativnih vrednosti i relativnih uglova idealni transformator je eliminisan (sveden na jedinicu) [4], a njegov model se sastoji od redne impedanse $\hat{z}_l$ i otočne admitanse magnećenja $\hat{y}_m$.

2.4. Koren distributivne mreže

Koren DM predstavlja čvor u kojem se DM priključuje na prenosni sistem. Ona predstavlja glavni izvor električne energije u DM i modeluje se indentično kao DER tipa I, koji će biti objašnjen u nastavku.

3. MODELI DER U PRORAČUNIMA REŽIMA DISTRIBUTIVNIH MREŽA S KRATKIM SPOJEVIMA

U zavisnosti od vrste i načina povezivanja na sistem, DER se u proračunima kratkih spojeva mogu modelovati na sledeća četiri načina [1]:

- 1) DER tipa I - Sinhroni distributivni generatori koji su direktno priključeni na DM;
- 2) DER tipa II - Asinhroni distributivni generatori koji su direktno priključeni na DM;
- 3) DER tipa III - Dvostruko napajana asinhrona mašina (Doubly Fed Induction Machine - DFIM);
- 4) DER tipa IV - IBDER (Inverter Based Distributed Energy Resource) – DER koji je priključen na mrežu preko invertora.

3.1. DER tipa I – Sinhroni generator direktno priključen na mrežu

U ovom radu modeli DER tipa I, u proračunima režima s kratkim spojem se prikazuju njihovim Tevenenovim ekvivalentima linearnog kola [4]. U kolima direktnog redosleda, u sve tri vremenske sekvence, postoji direktna komponenta elektromotornih sila $\hat{E}^d$, dok se impedansa menja od suptranzitne $\hat{Z}_s^{d''}$, preko tranzitne $\hat{Z}_s^{d'}$, do impedanse u ustaljenom periodu - $\hat{Z}_s^d$. Impedanse u inverznom $\hat{Z}_s^i$ i nultom $\hat{Z}_s^0$ se ne menjaju tokom kratkog spoja.

3.2. DER tipa II – Asinhroni generator direktno priključen na mrežu

Asinhroni generatori se, slično sinhronim, tretiraju Tevenenovim ekvivalentima. Jedine razlike su što kod modela asinhrona mašine ne postoji tranzitni period i elektromotorne sile $\hat{E}^d$ su anulirane.

3.3. DER tipa III – DFIM

Kod ovih generatora, stator je povezan direktno na trofaznu DM, dok je rotor povezan preko uređaja energetske elektronike. Ovaj pretvarač je jako osetljiv na struje iznad nominalnih, tako da mora na neki način, biti zaštićen od kratkih spojeva u blizini generatora.

Kroubar i čoper su dve osnovne i najrasprostranjenije zaštite [3]. Kroubar predstavlja trofazan otpornik velike vrednosti koji se redno povezuje sa namotajem rotora mašine.

U slučaju ozbiljnog kvara, odnosno kratkog spoja blizu DFIM, kroubar reaguje i kratko spaja rotor mašine [1]. Time se pretvarač potpuno odvaja od ostatka mreže, a model DFIM u proračunima kratkih spojeva postaje jako sličan modelu asinhrona mašine (DER tipa II), tj. DFIM gubi kontrolu nad izlaznim strujama. Ako se DFIM štiti čoperom ili kvar nije toliko ozbiljan inverter će nastaviti da kontroliše injektirane struje i oni će imati iste modele kao DER IV, koji će biti opisan u sledećoj glavi.

3.4. DER tipa III – IBDER

Ovaj tip DER je uređajem energetske elektronike kompletno odvojen od DM. Ta činjenica, omogućava potpunu kontrolu izlaznih struja, kako u normalnim režimima, tako i u slučaju kratkih spojeva u DM.

4. DER TIPA III-IV

Ovom tipu DER će, u ovom radu, biti posvećeno najviše pažnje. Predstavnicima ovog tipa DER su IBDER i DFIM koji uspeju da zadrže kontrolu struje tokom kratkog spoja. Za ove vrste DER, koristiće se zajednički naziv – „DER tipa III-IV” [1]. Njihovi modeli, kao i injektirane struje tokom kvara, uopšte ne zavise od karakteristika samih generatora, već isključivo od primenjenih kontrolnih strategija koje su ugrađene u invertore [1]. Tokom kvara, da bi se uređaj energetske elektronike ovog tipa DER zaštitio, kontrolisana injektirana struja ne sme preći predefinisane granice. Najčešće se struja tokom kratkog spoja ograničava na 1 – 1.5 njene nominalne vrednosti.

DER tipa III-IV injektira struje isključivo direktnog redosleda, čak i tokom neuravnoteženih kratkih spojeva [3]. Odatle sledi da njihov model u proračunima kratkih spojeva sadrži idealan strujni generator jedino u kolu direktnog redosleda.

4.1. FRT standardi

Većina razvijenih zemalja ima svoja pravila o pogonu DM, među kojima se nalaze i FRT zahtevi. Njima se propisuje kako bi moderni DER trebalo da se ponašaju u slučaju pada napona usled kratkog spoja na mestima njihovih priključenja na DM.

FRT zahtevi za obnovljive izvore električne energije generalno opisuju sledeća četiri podzahteva:

- 1) Vrednost napona na kojem DER tipa III-IV mora ostati priključen na DM tokom kratkog spoja;
- 2) Nivo naponske podrške (injektirane reaktivne snage) koja se zahteva od DER tipa III-IV tokom kvara;
- 3) Injektiranje aktivne snage tokom kvara;
- 4) Injektiranje aktivne snage nakon čišćenja kvara.

Kada se DM nalazi u stanju sa kvarom, DER tipa III-IV treba da ostanu operativni i priključeni, čak i u slučaju smanjenih napona. Najčešće se, u okviru pravila o pogonu DM definišu LVRT (Low Voltage Ride Through) zahtevi kojima se propisuju vrednosti napona pri kojima DER tipa III-IV ostaje na mreži.

Naponi čvorova su jako pogođeni kratkim spojem u DM, tako da DER tipa III-IV treba da pruže podršku injektiranjem reaktivne snage. FRT standardima se propisuje vrednost reaktivne struje koja će biti injektirana u DM od strane DER tipa III-IV, sa ciljem što bržeg povratka napona u normalne granice [5]. Većina tih standarda zahteva da se sa jednim procentom pada napona od njegove nominalne vrednosti DER tipa III-IV poveća injektiranje reaktivne struje za 2%. Tokom kvarova prioritet injektiranja iz DER tipa III-IV je reaktivna snaga, dok se aktivna snaga injektira onoliko koliko je to moguće, a da se ne premaši strujno ograničenje.

4.2. Modeli DER tipa III-IV za potrebe proračuna režima s kratkim spojem

Kao što je ranije rečeno, DER tipa III-IV su podešeni tako da injektiraju simetrične struje direktnog redosleda čak i u slučaju neuravnoteženih kvarova u sistemu [1]. Režimi inverznog i nultog pogonskog kola su anulirani. Odatle sledi, da modeli ovih DER u simulacijama kratkih spojeva sadrže jedino idealne strujne generatore u kolima

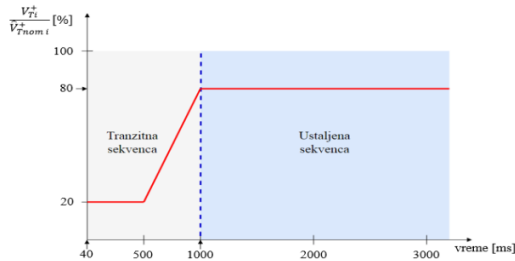
direktnog redosleda. Takođe, ovi DER reaguju isključivo na pad direktne komponente napona na njihovim priključcima na mrežu, tako da se njihovi modeli u proračunima režima s kratkim spojem sastoje samo od komponente direktnog redosleda [1], što je prikazano na slici 1. Naponi direktnog redosleda u kojima su priključeni DER tipa III-IV (V_{Ti}^+ , $i = 1, \dots, N_{DERIII-IV}$) se računaju u jednoj iteraciji USSKN i pomoću njih će biti proraračunate struje injektiranja ovih DER u svim vremenskim sekvencama kratkog spoja.



Slika 1. Model DER tipa III-IV u kolima direktnog redosleda za proračun kratkih spojeva u DM

Nakon ovog proračuna, modeli DER tipa III-IV se dele na suptranzitne, tranzitne i ustaljene. Na slici 1 je prikazan model ovog tipa DER u svim vremenskim sekvencama. Jedino što se menja kroz sekvence je vrednost injektirane struje kratkog spoja ($\hat{I}_{DERIII-IV i}^{ks}$).

Najčešće, suptranzitna sekvenca predstavlja vreme dok inverter DER tipa III-IV ne detektuje kratak spoj. To je najčešće do dve periode (40ms). Druge dve sekvence, tranzitna i ustaljena, definiše posmatrani FRT standard. U ovom radu biće obrađen španski FRT zahtev. Taj zahtev, kao i podela na tranzitnu i ustaljenu sekvencu je prikazana na slici 2. Ovako definisane vremenske sekvence modela DER tipa III-IV se prilično dobro slažu s sekvencama DER tipa I i II [4].



Slika 2. Tranzitna i ustaljena vremenska sekvenca za DER tipa III-IV tokom kratkog spoja

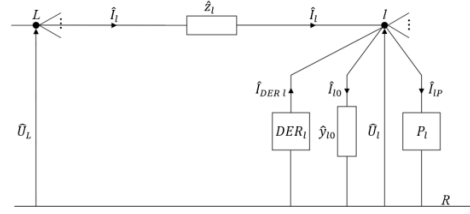
4.3. Γ segment distributivne mreže

Proračun režima DM s kratkim spojem je najlakše vršiti u domenu simetričnih veličina. Time se model DM predstavlja sa tri pogonska kola direktnog, inverznog i nultog redosleda. Kako se ovde posmatraju samo uravnotežene trofazne mreže ova kola su spregnuta samo na mestu neuravnoteženog kratkog spoja (čvor k). Ta kola su prikazana na slici 3.

5. GENERALIZOVANO DELTA KOLO

Koncept tradicionalnog delta kola, generalno nije pogodan za DM koji sadrže DER tipa III-IV sa velikim rasponom mogućih kontrolnih strategija [1].

Koristeći drugu kontrolnu strategiju, pretpostavka da je delta kolo pasivno svuda osim na mestu s kratkim spojem više ne važi [1].



Slika 3. Γ segment DM u sva tri redosleda

U ovom slučaju, delovi struja koji premašuju injektirane struje DER tipa III-IV pre kvara se injektiraju u delta kolo. Stoga, delta kolo postaje aktivno i u svim čvorovima u kojima su ovi DER priključeni na DM [1]. Kako DER tipa III-IV injektira samo direktnu komponentu struje, samo tradicionalno delta kolo direktnog redosleda se menja. Ovo kolo se naziva generalizovano delta kolo, a struje koje prekoračuju struje pre kvara se nazivaju ekcesnim i računaju se na sledeći način [1]:

$$\hat{I}_{DERIII-IV i}^{\Delta+} = \hat{I}_{DERIII-IV i}^{ks} - \hat{I}_{DERIII-IV i}^{preks}, \quad i = 1, \dots, N_{DERIII-IV} \quad (1)$$

gde $N_{DERIII-IV}$ predstavlja čvorove u kojima su priključeni DER tipa III-IV sa ekcesnim strujama ($\hat{I}_{DERIII-IV i}^{\Delta+}$). Generalizovana delta kola inverznog i nultog redosleda su i dalje pasivna svuda osim na mestu kvara, jer su ekcesne struje u tim kolima jednake nuli.

5.1. Ključne veličine za proračun režima generalizovanog delta kola

U ovom radu se obrađuju četiri vrste metalnih kratkih spojeva. Struje kvara u domenu simetričnih veličina na mestu kratkog spoja (čvor k) se tradicionalno računaju jednačinama prikazanim u sledećoj tabeli [4]:

Tabela 1. Formule za proračun struja kvara na mestu kratkog spoja

	1FKS(faza a)	2FKSZ(faze b i c)	2FKS(faze b i c)	3FKS
$\hat{J}_k^+$	$\frac{\hat{U}_{ka}}{\hat{Z}_k^+ + \hat{Z}_k^- + \hat{Z}_k^0}$	$\frac{(\hat{Z}_k^- + \hat{Z}_k^0) \hat{U}_{ka}}{\hat{Z}_k^+ \hat{Z}_k^- + \hat{Z}_k^+ \hat{Z}_k^0 + \hat{Z}_k^- \hat{Z}_k^0}$	$\frac{\hat{U}_{ka}}{\hat{Z}_k^+ + \hat{Z}_k^-}$	$\frac{\hat{U}_{ka}}{\hat{Z}_k^+}$
$\hat{J}_k^-$	$\frac{\hat{U}_{ka}}{\hat{Z}_k^+ + \hat{Z}_k^- + \hat{Z}_k^0}$	$\frac{-\hat{Z}_k^0 \hat{U}_{ka}}{\hat{Z}_k^+ \hat{Z}_k^- + \hat{Z}_k^+ \hat{Z}_k^0 + \hat{Z}_k^- \hat{Z}_k^0}$	$\frac{-\hat{U}_{ka}}{\hat{Z}_k^+ + \hat{Z}_k^-}$	0
$\hat{J}_k^0$	$\frac{\hat{U}_{ka}}{\hat{Z}_k^+ + \hat{Z}_k^- + \hat{Z}_k^0}$	$\frac{-\hat{Z}_k^- \hat{U}_{ka}}{\hat{Z}_k^+ \hat{Z}_k^- + \hat{Z}_k^+ \hat{Z}_k^0 + \hat{Z}_k^- \hat{Z}_k^0}$	0	0

U formulama iznad, $\hat{Z}_k^+$, $\hat{Z}_k^-$, $\hat{Z}_k^0$ predstavljaju Tevenenove impedanse u domenu simetričnih veličina viđenih sa mesta kratkog spoja. $\hat{U}_{ka}$ je poznati napon čvora k pre kvara. Kao što je pokazano u [2], uticaj ekcesnih struja DER tipa III-IV na struje kratkog spoja na mestu kvara je zanemarljiv.

5.2. Proračun režima generalizovanog delta kola u domenu simetričnih veličina

Režim generalizovanog delta kola u direktnom redosledu se računa sledećim algoritmom (USSKN) [1]:

Sumiranje struja:

$$(\hat{f}_l^{\Delta+})^{h+1} = \hat{f}_l^{\Delta+} + \hat{y}_{l0}^+ (\hat{U}_l^{\Delta+})^h + \sum_{j \in \alpha_l} (\hat{f}_j^{\Delta+})^{h+1} \quad (2)$$

$$+ \hat{f}_{DERIII-IV}^{\Delta+}, l = n, \dots, 3, 2$$

Korekcija napona:

$$(\hat{U}_l^{\Delta+})^{h+1} = (\hat{U}_l^{\Delta+})^{h+1} - \hat{z}_l^+ (\hat{f}_l^{\Delta+})^{h+1}, \quad (3)$$

$$l = 2, 3, \dots, n$$

Injektirane struje na mestu kvara su poznate iz tabele 1:

$$\hat{f}_l^{\Delta+} = \begin{cases} 0, & l \neq k \\ \hat{f}_{k+}^{\Delta+} \neq 0, & l = k \end{cases} \quad (4)$$

Ekcesne struje DER tipa III-IV su poznate iz sledeće formule:

$$\hat{f}_{DER3-4}^{\Delta+} = \begin{cases} 0, & l \notin \alpha_{DERIII-IV} \\ -\hat{f}_{DERIII-IV}^{\Delta+}, & l \in \alpha_{DERIII-IV} \end{cases} \quad (5)$$

Iterativna procedura se zaustavlja kada se ispuni kriterijum konvergencije:

$$|(\hat{U}_l^{\Delta+})^{h+1} - (\hat{U}_l^{\Delta+})^h| < \varepsilon_U, l = 2, 3, \dots, n \quad (6)$$

Režimi delta kola u inverznom i nultom redosledu se izračunavaju indentično kao tradicionalna delta kola, kao što je prikazano u [4].

6. KOMPLETAN TOK PRORAČUNA REŽIMA S KRATKIM SPOJEM DISTRIBUTIVNE MREŽE U KOJOJ SE NALAZE DER TIPa III-IV

Na osnovu svih izlaganja u prethodnim glavama, sada je na jednom mestu, na slici 4, prikazan kompletan tok proračuna uravnotežene DM bez petlje u režimu s kratkim spojem.

7. ZAKLJUČAK

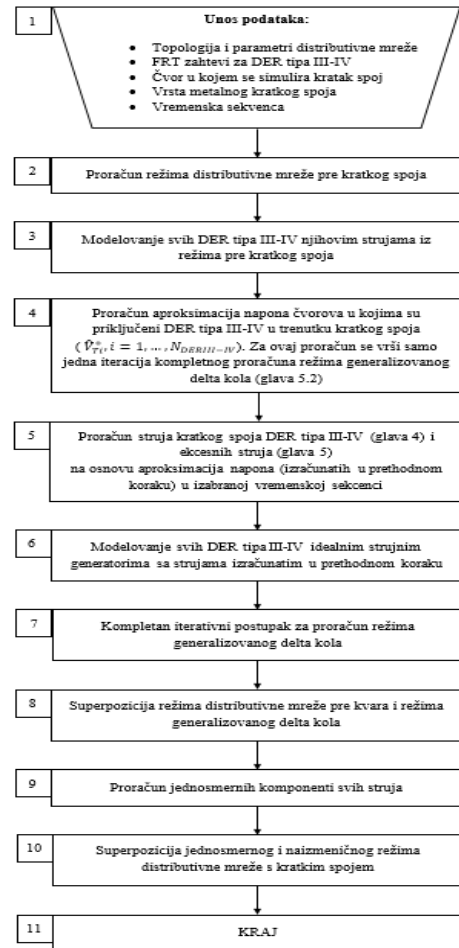
U skladu sa [1], u ovom radu je urađeno sledeće:

1. Predstavljen je pojam FRT standarda;
2. Detaljno su opisani modeli DER tipa III-IV za potrebe proračuna DM s kratkim spojem. Da bi se predloženi model lakše uklopio u postojeće proračune on je predstavljen u suptranzitnoj, tranzitnoj i ustaljenoj vremenskoj sekvenci;
3. Objašnjen je koncept generalizovanog delta kola koji omogućuje integraciju modela DER tipa III-IV u proračune režima DM s kratkim spojem;
4. Da bi se režim generalizovanog delta kola odredio, predstavljen je unapređeni algoritam sumiranja struja i korekcije napona.

Realizacijom ovih ciljeva, ispunjena su dva osnovna uslova proračuna kratkih spojeva u DM, a to su visoka preciznost i efikasnost.

Preciznost je rezultat što je moguće tačnijeg modelovanja elemenata DM, dok su u isto vreme ti modeli i dovoljno jednostavni za proračune kratkih spojeva u DM velikih dimenzija, koje sadrže relativno veliki broj DER tipa III-IV.

Korišćenjem predstavljenog unapređenog algoritma sumiranja struja i korekcije napona moguće je, sa podjednako lakoćom, proračunavati režime DM kako sa malim brojem čvorova, tako i DM ogromnih dimenzija.



Slika 4. Algoritam proračuna stanja mreže s kratkim spojem

8. LITERATURA

- [1] Luka Strezoski, „Proračun kompleksnih kratkih spojeva neuravnoteženih distributivnih mreža sa distribuiranim energetske resursima”, doktorska disertacija, Fakultet tehničkih nauka, Novi Sad, 2017.
- [2] Luka Strezoski and Marija Prica, „Real-Time Short-Circuit Analysis of Active Distribution Systems”, IEEE Power and Energy Conference in Illinois (PECI), Champagne, IL, USA, 2016.
- [3] Joint Working Group, „Fault Current Contribution from Wind Plants”, Report to the T&D Committee of the IEEE Power and Energy Society, Pro Relay, 2015.
- [4] Vladimir Strezoski, „Analiza elektroenergetskih sistema”, Fakultet tehničkih nauka, Novi Sad, 2014.
- [5] Luka Strezoski, Marija Prica, Vladimir Katić, Boris Dumnić, „Short-Circuit Modeling of Inverter Based Distributed Generators Considering the FRT Requirements”, North American Power Symposium (NAPS), IEEE, Denver, Septembar 2016.

Kratka biografija:



Miloš Spasić rođen je 1994. godine u Krajujevcu. Fakultet tehničkih nauka, studijski program Energetika, elektronika i telekomunikacije upisao je 2013. Na studijama se opredelio za smer Elektroenergetika - elektroenergetski sistemi i diplomirao u oktobru 2017. godine. Trenutno je na master studijama na smeru Elektroenergetski sistemi na Fakultetu tehničkih nauka u Novom Sadu.

SMERNICE KVALITETA ANDROID APLIKACIJA**ANDROID QUALITY GUIDELINES**Luka Pavlica, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom master radu analizirane su smernice kvaliteta mobilnih aplikacija implementiranih na Android platformi. Izabrane smernice kvaliteta primenjene su na dizajn i implementaciju Travel aplikacije, Android aplikacije za pregled hotela, restorana, spomenika i prodavnica. Opis dizajna i implementacije ove aplikacije može da posluži kao uputstvo za dizajniranje i implementaciju kvalitetnih Android aplikacija.

Ključne reči: Mobilne aplikacije, Android, smernice kvaliteta

Abstract – This master thesis analyses Android app quality guidelines. The selected quality guidelines are applied to the design and development of the Travel app, an Android app for reviewing hotels, restaurants, monuments and shops. The description of the design and implementation can be used as an instruction manual for developing Android apps adhering to quality guidelines.

Keywords: Mobile apps, Android, quality guidelines

1. UVOD

Sa rastom broja korisnika mobilnih aplikacija rasli su i zahtevi koje aplikacija mora da ispuni. Nakon što su kreirane razne smernice kvaliteta, Android [1] aplikacije su postale kvalitetnije. Danas da bi korisnik bio zadovoljan opštim izgledom i funkcionisanjem aplikacije, programeri moraju da naprave aplikaciju koja će poštovati smernice za vizuelni i navigacioni dizajn, kompatibilnost, sigurnost, performanse, itd.

2. SMERNICE KVALITETA

Ovaj rad se bavi načinom poboljšanja kvaliteta aplikacije, kako bi direktno uticao na njen uspeh na tržištu. Pridržavanjem određenih smernica [2] se može unaprediti kvalitet aplikacije. Te smernice mogu se podeliti u nekoliko kategorija.

2.1. Vizuelni dizajn i interakcija sa korisnikom

Kriterijumi koji pripadaju ovoj kategoriji podeljeni su u grupe: standardni dizajn, navigacija i notifikacije. Ovi kriterijumi osiguravaju da aplikacija pruža standardni Android vizuelni dizajn za dosledno i intuitivno korisničko iskustvo.

2.2. Funkcionalnost

Ovim smernicama kvaliteta pripadaju: dozvole, lokacija instalacije, audio, korisnički interfejs i grafika i stanje korisnika/aplikacije. Ovi kriterijumi obezbeđuju da aplikacija bude funkcionalna.

2.3. Kompatibilnost, performanse i stabilnost

Ovim smernicama kvaliteta pripadaju: stabilnost, performanse, SDK, multimediji i vizuelni kvalitet. Ovo su kriterijumi koji obezbeđuju da aplikacija pruža kompatibilnost, performanse, stabilnost, kao i odziv koji očekuju korisnici.

2.4. Google Play

Ukoliko aplikacija treba da završi na tržištu, ove smernice kvaliteta predstavljaju bitan uslov koji se mora ispoštovati. Ovi kriterijumi osiguravaju da je aplikacija spremna za objavljivanje na Google Play [3]. Sastoji se od kriterijuma: polise, detalji stranica aplikacije i korisnička podrška.

2.5. Build for billions

Aplikacije koje koriste memoriju, procesorsku snagu i mrežnu propusnost na bolji način biće efikasnije i proizvešće bolje iskustvo za sve korisnike. Ova kategorija smernice kvaliteta se sastoji od grupa: konektivnost, mogućnosti uređaja, troškovi podataka, potrošnja baterije i korisnički interfejs i sadržaj.

2.6. Lokalizacija

Android korisnici mogu biti iz bilo kog dela sveta i jezici na kojim se oni najčešće sporazumevaju i na koji su navikli koristiti se veoma razlikuju. Zbog tog razloga aplikaciju bi trebalo lokalizovati na bar ključne regionalne jezike. Aplikacija treba da bude dizajnirana tako da se lako lokalizuje prilagođavanjem varijacija između jezika.

2.7. Bezbednost

Smernice kvaliteta koje pripadaju ovoj kategoriji obezbeđuju da aplikacije bezbedno rukuju ličnim podacima. Sastoji se od kriterijuma: podaci, komponente aplikacije, umrežavanje, biblioteke, WebViews, izvršenje i kriptografija.

3. ZAHTEVI APLIKACIJE**3.1. Funkcionalni zahtevi**

Aplikacija treba da omogućiti neprijavljenom korisniku registraciju, prijavu na sistem, promenu jezika i promenu orijentacije ekrana.

Uslov za pregled svih gradova i njihovih hotela, restorana, spomenika i prodavnica je da se korisnik

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stevan Gostojić, vanr. prof.

prethodno uspešno prijavio. Dodatno treba da postoji mogućnost filtracije svakog od njih. Aplikacija prijavljenom korisniku treba da omogući pregled informacija konkretnog hotela, restorana, spomenika, ili prodavnice. Pored pregleda treba da nudi i prikaze na mapi svakog od njih. Takođe aplikacija treba da omogući prijavljenim korisnicima rezervaciju hotela i restorana. Dodatno korisnik treba da ima pregled rezervacija hotela i restorana, kao i pregled svog profila. Ukoliko poželi treba da može i da ažurira svoje podatke.

3.2. Nefunkcionalni zahtevi

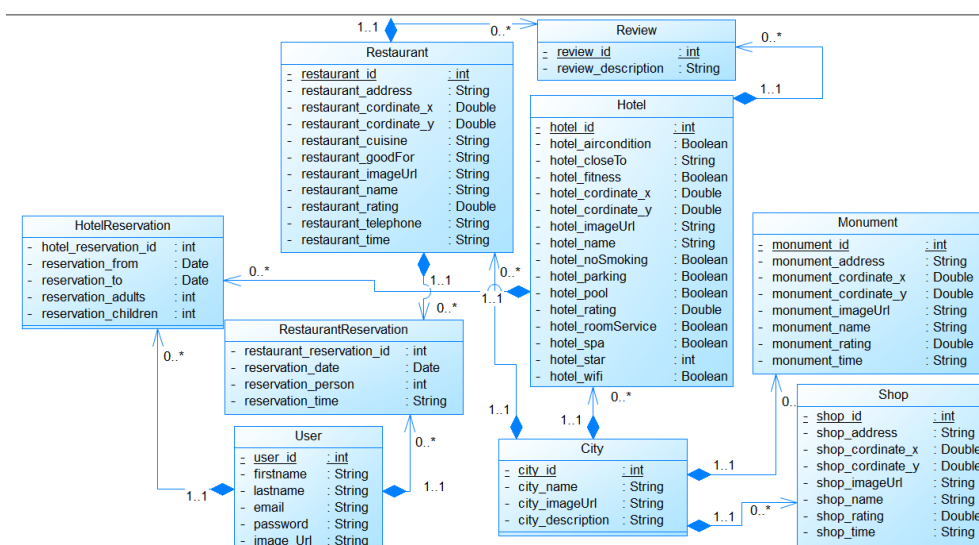
U aplikaciji će biti implementirane smernice kvaliteta tako da će se prikaz ekrana, kao i promena prikaza ekrana, vršiti brzo. Isto tako sva stanja će biti sačuvana prilikom prelaska na drugu aplikaciju, ili prelaska stanja

telefona na sleep mod. Korisnik će biti obezbeđen od gubitka podataka. Do iznenadnog prestanka rada aplikacije ne bi trebalo da dođe, osim kao posledica grešaka u radu operativnog sistema. Slike u aplikaciji će biti prebačene na .webp format i koristiće se odgovarajuće biblioteke za učitavanje slika. Приликом уноса, или измене података врши ће се провера и валидација свих вредности.

4. DIZAJN APLIKACIJE

Za opis modela podataka upotrebljen je dijagram klasa na kom su prikazane klase i njihove međusobne veze.

Na slici 1 prikazane su klase i njihov međusobni odnos.



Slika 1. Dijagram klasa Travel aplikacije

Klasa User predstavlja korisnika aplikacije. Korisnik može da ima više, ili nijednu rezervaciju. Klasa HotelReservation predstavlja rezervacije hotela koje je neki korisnik izvršio.

Slično kao i prethodna klasa RestaurantReservation predstavlja rezervacije restorana koje je neki korisnik izvršio. Klasa Hotel predstavlja hotel u jednom gradu. U jednom gradu može biti nijedan, ili više hotela. Takođe hotel može biti rezervisan od strane korisnika više puta, ili nijednom.

Slično kao prethodna klasa Restaurant predstavlja restoran jednog grada. Restoran se može rezervisati nijednom, ili više puta, a svaki restoran pripada jednom gradu. U jednom gradu može da se nalazi više spomenika, a ne mora da postoji nijedan.

Klasa Monument sadrži jedan i samo jedan grad. Identično kao za prethodnu klasu, klasa Shop predstavlja prodavnicu u nekom gradu. Prodavnica se nalazi u samo jednom gradu, dok grad može da sadrži više prodavnica.

Klasa City predstavlja klasu grad u kojem se nalaze hoteli, restorani, spomenici i prodavnice. Klasa Review predstavlja komentare korisnika na hotele i restorane. Jedan hotel, ili restoran mogu imati više, ili nijedan komentar. Jedan komentar pripada tačno jednom hotelu, ili restoranu.

Za prikaz rasporeda komponenti aplikacije upotrebljen je dijagram rasporeda. Android aplikacija je napisana u

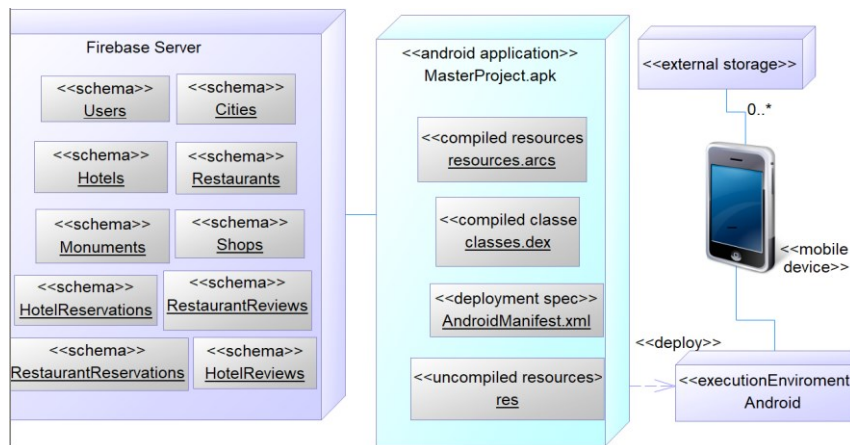
programskom jeziku Java [4]. SDK alati kompajliraju i pakuju izvršni kod zajedno sa svim potrebnim podacima i resursnim datotekama u jednu datoteku sa .apk sufixom. Ova .apk datoteka predstavlja jednu Android aplikaciju koja će biti raspoređena na mobilne uređaje. Aplikacija se sastoji od jedne, ili više komponenti. Instaliranje aplikacije na eksterni uređaj je opciono, a mesto instalacije se definiše u AndroidManifest.xml fajlu. Dijagram rasporeda Travel aplikacije prikazan je na slici 2.

5. IMPLEMENTACIJA APLIKACIJE

Kao primer aplikacije koja ispunjava smernice kvaliteta implementirana je Travel aplikacija.

5.1. Delovi implementacije

Projekat je podeljen na više manjih delova (celina): konfiguracija, model i view i kontroler. Konfiguracija je deo projekta koji se sastoji od konfiguracionih fajlova, kao i .gradle skripti u kojima se nalaze sve zavisnosti projekta. Model je deo projekta u kome se nalaze klase koje opisuju model podataka. Upotrebom Firebase-a [5] i njegovim unapred definisanim metodama, vrednosti atributa ovih klasa se dovlače do baze podataka. View i kontroler predstavlja podrazumevani pristup pravljenja Android aplikacije gde je Activity klasa kontroler, ali isto tako ima ulogu i u upravljanju view delom.



Slika2. Dijagram rasporeda naše aplikacije

5.2. Implementacija bitnih elemenata

Aplikacija je napravljena u skladu sa Material Design [6] smernicama kvaliteta i koristi uobičajne Android ikonice za uobičajene funkcije. Ova aplikacija podržava standardni back button navigation.

Korisnik ima mogućnost vraćanja pritiskom na back dugme uređaja, ili preko up mehanizma. Veličine svake komponente u projektu su predstavljene u dp jedinicama, osim teksta koja je predstavljena u sp jedinicama. Na ovaj način je rešen problem sa prikazima sadržaja aplikacije na ekranima uređaja sa različitim veličinama.

U ovom projektu ne zahtevaju se dozvole za pristup osetljivim podacima (kao što su kontakti), ili usluge koje se naplaćuju.

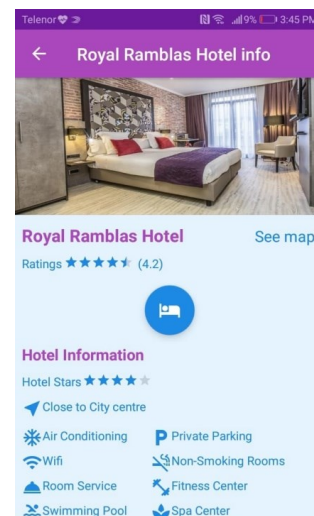
Lokacija instalacije aplikacije je na SD kartici ukoliko je to moguće. Prikaz sadržaja je omogućen i pri horizontalnoj i pri vertikalnoj orijentaciji ekrana uvođenjem komponente ScrollView. U toku korišćenja aplikacije ni u jednom trenutku neće doći do situacije da je korisnik primoran da izađe iz nje.

Aplikacija će se ponašati normalno, bez iznenadnog gašenja, ili „smrzavanja“.

Korišćenjem Picasso biblioteke za učitavanje slika, kao i .webp formata za slike, smanjeno je vreme učitavanja slika. Dodirom na neku komponentu, korisnik dobija odgovarajući feedback. Implementirana je višejezičnost.

6. DEMONSTRACIJA APLIKACIJE

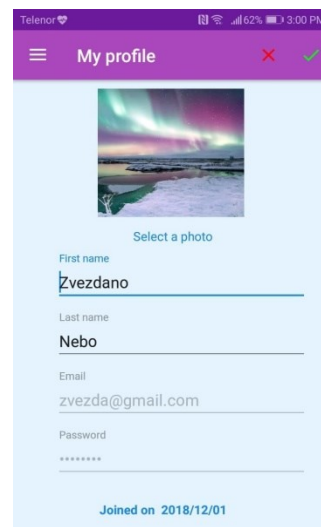
Pri pokretanju aplikacije prikazuje se početni ekran (LoginActivity). Takođe, postoji mogućnost registracije korisničkog naloga ako nije registrovan. Ukoliko se korisnik uspešno prijavio na sistem, otvara mu se strana sa prikazom svih gradova u kojima može kasnije da gleda hotele, restorane, spomenike i prodavnice i pravi rezervacije. Mehanizam pretrage postoji u svakom od ovih prikaza. U bilo kojem trenutku korisnik može da promeni jezik na svom uređaju i aplikacija će se automatski prilagoditi tom jeziku. Prikaz hotela, kao i restorana, sastoji se od njihove slike, naziva i ocene. Prikaz spomenika i prodavnica sastoji se od slike i naziva. Nakon odabira određenog prikazuju mu se informacije. Dodatno se pruža mogućnost prikaza kordinata na mapi. Na slici 3 se nalazi prikaz informacija jednog hotela.



Slika 3. Prikaz informacija jednog hotela

Pri rezervaciji, korisnik je dužan da unese sva polja koja se traže od njega. U suprotnom korisnik se obaveštava o grešci.

Korisnik ove aplikacije, ukoliko želi, može menjati svoje podatke. Podaci koji mogu da se ažuriraju su ime, prezime i profilna slika, dok se lozinka i e-mail ne mogu promeniti.



Slika 4. Prikaz profila korisnika

7. ZAKLJUČAK

U radu su opisane smernice kvaliteta kojih bi svaka Android aplikacija trebalo da se pridržava i po tim smernicama je implementirana jedna Android aplikacija kao studija slučaja.

Iako implementirana aplikacija ispunjava veliki broj smernica kvaliteta, tokom razvoja i korišćenja aplikacije primećeni su neki nedostaci. Korisnik je u mogućnosti da vidi samo jednu sliku hotela, ili restorana kojeg želi da rezerviše. U profesionalnim aplikacijama ovo nije slučaj. Takođe ne postoji opcija otkazivanja rezervacija. Pored navedenih nedostataka jedan od narednih koraka u razvoju jeste i omogućavanje korišćenja aplikacije bez upotrebe interneta.

LITERATURA

- [1] Android, <https://www.android.com>
- [2] Core app quality, <https://developer.android.com/docs/quality-guidelines/core-app-quality>
- [3] Google Play, <https://play.google.com/about/developer-content-policy/>
- [4] Java, <https://java.com/en>
- [5] Firebase, <https://firebase.google.com/>
- [6] MaterialDesign, <https://material.io/design/introduction/>

Kratka biografija



Luka Pavlica rođen je 26.07.1993 u Novom Sadu. Završio je Osnovnu Školu „Jovan Popović“ i gimnaziju Laza Kostić u Novom Sadu. Osnovne akademske studije na Fakultetu tehničkih nauka je upisao 2012. godine. Stekao je zvanje diplomiranog inženjera elektrotehnike i računarstva 2017. god.. Master akademske studije na Fakultetu tehničkih nauka upisao je 2017. godine.

OPTIMALNO ANGAŽOVANJE KONDENZATORA U DISTRIBUTIVNOJ MREŽI ZA PERIOD OD 24H**OPTIMAL ENGAGEMENT OF CAPACITORS IN DISTRIBUTION NETWORK FOR 24H PERIOD**

Branislav Vejnović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je razmatran problem optimalnog angažovanja kondenzatora u distributivnoj mreži za period od 24h. Matematički model za njegovo rešavanje je razvijen i implementiran u programskom jeziku C++. Potom je izvršena verifikacija i praćenje toka algoritma na primeru test mreže.

Ključne reči: Distributivna mreža, Reaktivna snaga, Faktor snage, Upravljanje kondenzatorima

Abstract – In this paper optimal engagement of capacitor in distribution network for period of 24h is analyzed. Mathematical model of optimization problem is described and then implemented in programming language C++. Verification of mathematical model and algorithm tracking is done on simple example of test distribution network.

Keywords: Distribution network, Reactive power, Power factor control, Shunt capacitor optimization

1 UVOD

Prilikom koordinacije dva uređaja, plan angažovanja kondenzatora je sastavljen "off-line", na osnovu prognoziranih karakteristika potrošnje. Kondenzatori rade u skladu sa tako definisanim rasporedom. Pritom, otočni kondenzatori uglavnom imaju samo dva diskretna stanja, uključeno i isključeno, koja nisu dovoljna za iole finu regulaciju napona i tokova reaktivnih snaga. Nemoćnosti kondenzatorskih baterija i odstupanja prognoziраног i ostvarenog režima, pokrivaju se primenom regulacioni transformatori sa mogućnošću regulacije pod naponom (ULTC Tr).

Primenom ULTC Tr sa više pozicija regulacione sklopke, omogućeno je da se u realnom vremenu obezbedi dovoljno fino regulacija napona. Da bi se što više pomoglo ULTC Tr potrebno je razviti algoritam čijom primenom bi se dobio što kvalitetniji plan angažovanja kondenzatora za ceo dan [1]. Jedan takav algoritam treba da na osnovu prognozirane potrošnje i uklopnog stanja mreže, da napravi teorijski optimalan plan po kojem kondenzatori treba da rade. Dodatnu kompleksnost algoritma uvode ograničenja dozvoljenih manipulacija nad kondenzatorima u toku dana.

Nakon Uvoda u drugom delu rada je predstavljen problem optimalnog angažovanja otočnih kondenzatora za naredna 24h. U trećem delu je dat prikaz metoda za rešavanje postavljenog problema – izbor optimalnog angažovanja kondenzatora u zadatom periodu. Nakon toga u četrtom delu date su teorijske osnove genetskog algoritma. Pre svega njegove komponente i princip rada.

Peti deo se nadovezuje na četvrti. On sadrži detalje implementacije genetskog algoritma za rešenje datog problema, kao i opis parametara optimizacije i ograničenja koja genetski algoritam uvažava.

U šestom delu su prikazani rezultati verifikacije korišćenog algoritma. Nakon zaključka referentno je navedena literatura korišćena za realizaciju ovog rada.

2 POSTAVKA PROBLEMA

Dimenzija problema optimizacije, izbor optimalnog angažovanja kondenzatora u zadatom periodu, zavisi od broja i karakteristika regulacionih resursa, vrste i broj kriterijumske funkcije i naravno ograničenja koje te kriterijumske funkcije moraju da ispoštuju. Kriterijumska funkcija je funkcija kojom je izražen cilj VVO procedure. Ona može da predstavlja potrebu distributivnog preduzeća da smanji troškove, ostvari dobit, da poveća sigurnost pogona distributivne mreže (DM), itd.

Svaka kriterijumska funkcija podleže ograničenjima, što znači da njen minimum ne može uvek, po svaku cenu, da se ostvari.

Ograničenja su granice unutar kojih posmatrane veličine moraju da budu. Pri izboru konačnog, optimalnog rešenja mogu da se razmatraju samo rešenja kod kojih su sva ograničenja zadovoljena. Svaki dodatni kriterijum i svako dodatno ograničenje znatno otežava mogućnost postizanja optimalnog rešenja [2].

Cilj optimalnog angažovanja kondenzatora u naredna 24 časa jeste da se minimiziraju gubici aktivne snage u mreži i da se pritom smanji reaktivna snaga koja se preuzima iz prenosne mreže. Sve to uvažavajući naponska ograničenja i ograničenja broja manipulacija nad ULTC Tr i kondenzatorima [2].

Pritom, treba naglasiti tri velika problema koji narušavaju validnost planiranog angažovanja kondenzatora [3]:

1. Greške kratkoročne prognoze snage distribuiranih generatora i snage potrošača.
2. Neplanirana promena uklopnog stanja mreže.
3. Zavisnost snage potrošača od napona.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor bio dr Goran Švenda, red. prof.

3 PREGLED LITERATURE

U protekloj deceniji je razvijano nekoliko metaheurističkih metoda koje su u stanju da reše optimizacione probleme koji su smatrani izuzetnom teškim ili čak nemogućim. Pod pojmom metaheurističkih metoda, prvenstveno se misli na evolucione algoritme (algoritme mravljih kolonija [4] metode roja [5], kaljenje čelika [6]), metode veštačke inteligencije [7] i druge. Izveštaji o primeni tih metoda su uveliko objavljeni. Za rešavanje tih ekstremno zahtevnih problema, ove metaheurističke metode su kombinovane. Dodatno, elementi znanja i drugi tradicionalniji pristupi, kao što je statistička analiza, su inkorporirani u metaheurističke metode.

Rešavanje optimizacionih problema sa ovim metodama nudi dve velike prednosti: i) vreme razvoja je mnogo kraće nego u slučaju korišćenja tradicionalnijih metoda optimizacije, ii) mogu se razmatrati problemi ogromnih razmera, relativno neosetljivi na šumove ili nedostajuće podatke. Osnovni nedostatak metaheurističkih metoda je da, i pored kvalitetne teorijske osnove po kojoj su sposobni da se prilagođavaju promenama stanja u DM, njihova primena još uvek nije dovoljno praktično potvrđena u realnim sistemima. Ovo je i glavna barijera za njihovu primenu u distributivnim sistemima na kojima se često vrši rekonfiguracija, ili imaju distributivne generatore.

Problem vremenskog intervala u primeni funkcije Volt/Var kontrol razmatra se u [8]. U tom radu problem pokušava da se reši primenom genetskog algoritma, tako što se kriva dnevnog opterećenja podeli na nekoliko nivoa opterećenja i za svaki od tih nivoa nađu otpimalne pozicije ULTC Tr i angažovanje kondenzatora. U [6] je predstavljen i na realnoj distributivnoj mreži testiran model na bazi fuzzy logike. Model uzima u obzir ograničenja broja operacija nad resursima, i da bi rešio problem formiranja operativnog rasporeda rada kondenzatora koristi metod kaljenja čelika.

Autori rada [4] koriste algoritam kolonije mrava da reše problem određivanja optimalnog angažovanja kondenzatora na sekundaru transformatora i kondenzatora duž fidera, kao i određivanje optimalnih pozicija ULTC Tr za isti period. Algoritam je testiran na IEEE test mrežama. Metod baziran na neuronskim mrežama prikazan u [7] vrši kontrolu otočnih kondenzatora, ali pritom ne uvažavanja ULTC Tr.

4 GENETSKI ALGORITAM

U oblasti veštačke inteligencije genetski algoritam (GA) je pretraživačka heuristika koja oponaša proces prirodne selekcije. Ova heuristika (takođe ponekad nazivana metaheuristika) se rutinski koristi da generiše korisna rešenja za optimizaciju i probleme pretrage [9].

4.1 Komponente

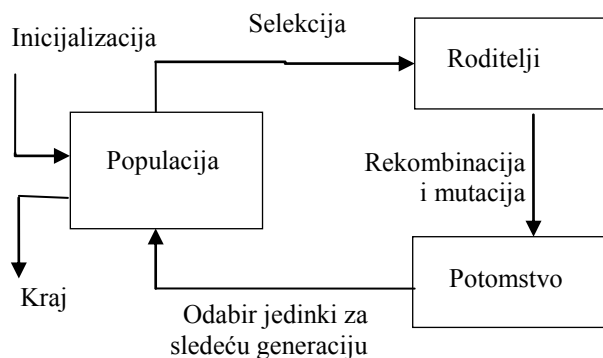
Osnovne komponente, neophodne za rad GA su [9]:

1. Reprezentacija jedinki,
2. Funkcija sposobnosti,
3. Populacija,
4. Mehanizam odabira roditelja,
5. Operatori mutacije i rekombinacije,
6. Način inicijalizacije populacije,
7. Uslov zaustavljanja.

4.2 Princip rada

Rad GA se može opisati u seldećih osam koraka [9]:

1. Definisane svih potrebnih parametara problema i GA,
2. Formiranje početne populacije,
3. Dekodiranje hromozoma – ovaj korak se javlja samo kod binarnih GA,
4. Određivanje cena hromozomima (funkcijom sposobnosti),
5. Odabir selekcije hromozoma koji će opstati za parenje,
6. Parenje – obično se iz boljeg dela populacije odaberu roditelji koji će na neki način ukrstiti svoj genetski materijal i dati jednog ili više potomaka,
7. Mutacija, pri kojima se menja genetski sadržaj hromozoma,
8. Ispitivanje konvergencije, da bi se utvrdilo da li ima osnova da se tok algoritma prekine. Ukoliko nije ispunjen uslov konvergencije, vratiti se na korak 3 odnosno 4.



Slika 1. Šema GA algoritma [9]

4.3 Ograničenja

Ponovljena procena funkcije sposobnosti (fitnes funkcije) za složene probleme je zaštitnički i ograničavajući segment veštačkih evolucionihih algoritama. Pronalaženje optimalnog rešenja za složene visoko-dimenzione, multimodalne probleme često zahteva veoma skupe procene fitnes funkcija. Genetski algoritmi ne rade dobro srazmerno složenosti, tj kada je broj elemenata koji su izloženi mutaciji veliki često postoji eksponencijalni porast u veličini prostora za pretragu [9].

Zbog toga je teško koristiti tehniku na problemima kao što su projektovanje motora, kuće, ili aviona. Na primer, problemi optimizacije, nemaju tačan cilj, koje rešenje mora da se dostigne, već se pokušava naći najbolje moguće rešenje. Za specifične probleme optimizacije i instance problema, drugi algoritmi optimizacije mogu biti efikasniji od genetskih algoritama u pogledu brzine konvergencije. Alternativni i komplementarni algoritmi uključuju evolucione strategije, evoluciono programiranje, Gausovu adaptaciju, pretraživanje usponom i inteligenciju jata [9].

5 IMPLEMENTACIJA GA

Nakon postavke problema pravljenja operativnog rasporeda kondenzatora za naredna 24h i teoretskog pojašnjenja genetskog algoritma, na red je došlo da se prezentuje jedan od mogućih načina implementacije genetskog algoritma sa ciljem da reši problem razmatran u ovom zadatku.

5.1 Optimizacioni kriterijumi

Odabrani optimizacioni ciljevi i ograničenja nisu jedinstvena za ovaj problem i zavise od potreba distributivnih preduzeća i njihovih zahteva. Vrednovanje optimizacionih kriterijuma služi da genetski algoritam proceni vrednost svakog mogućeg rešenja, što je osnova rada genetskog algoritma.

5.1.1 Minimizacija gubitaka aktivne snage

Kriterijumska funkcija minimizacije gubitaka aktivne snage je najčešće korišćena kriterijumska funkcija. Dobija se minimizacijom sume gubitaka po vodovima u posmatranom periodu:

$$P_{loss, sum} = \sum_{t=1}^{24} P_{loss, t}, \quad (1)$$

gde $P_{loss, t}$ predstavlja snagu trofaznih gubitaka aktivne snage u satu t .

5.1.2 Minimizacija injektirane reaktivne snage

Zahtev za minimizacijom injektirane reaktivne snage se uvodi u svrhu podrške prenosnoj mreži, da se kroz nju spreči nepotreban transport reaktivne snage. Kriterijumska funkcija za minimizaciju injektirane reaktivne snage je data relacijom [2].

$$Z_{QHV} = \sum_{i=1}^{24} C_{QHV} \cdot \Delta Q_{inj, i}, \quad (2)$$

$$\Delta Q_{inj} = \begin{cases} 0, & Q_{inj} < Q^{max} \\ Q_{inj} - Q^{max}, & Q^{max} < Q_{inj} \end{cases} \quad (3)$$

gde je:

C_{QHV} – cena reaktivne snage injektirane u DM;

Q^{max} – maksimalna vrednost injektirane reaktivne snage koju prenosna mreža ne naplaćuje distributivnoj;

$Q_{inj, i}$ – reaktivna snaga injektirana u DM sa primarne strane napojnog transformatora u i -tom času.

5.2 Ograničenja

Kriterijumske funkcije se moraju ograničiti standardnim ograničenjima koje mogu biti tipa jednakosti i nejednakosti. Razlika između kriterijumskih funkcija i ograničenja je prioritet u procesu optimizacije. Ograničenja su višeg prioriteta u optimizaciji. U slučaju da postoji narušenje ograničenja, prvi korak je uklanjanje tog narušenja, pa zatim u dozvoljenim granicama vršiti optimizaciju sa željenim kriterijumskim funkcijama, ako je moguće. Dakle, ograničenja su nešto što se mora ispoštovati, dok su kriterijumske funkcije cilj kome se teži što je više moguće [2].

5.2.1 Ograničenje po faktoru snage

Ograničenje po faktoru snage je jedno od standardnih ograničenja, koje direktno utiče na promenu tokova reaktivnih snaga. Uobičajeno se definiše na sledeći način:

$$\cos\varphi_{min} \leq \cos\varphi \leq \cos\varphi_{max} \quad (4)$$

gde je:

$\cos\varphi_{min}$ – minimalni dozvoljeni faktor snage;

$\cos\varphi_{max}$ – maksimalni dozvoljeni faktor snage;

$\cos\varphi$ – trenutni faktor snage [6].

5.2.2 Broj operacija na dnevnom nivou za kondenzatorske baterije

Ograničenje po broju akcija na dnevnom nivou za kondenzatorske baterije (npr. definisano specifikacijom proizvođača) je prikazano relacijom 5 [1]:

$$\sum_{i=1}^{24} (C_{n, i} \oplus C_{n, i-1}) \leq K_n; \quad n = 1, 2, \dots, n_c \quad (5)$$

gde je:

$C_{n, i}$ – status kondenzatora n u trenutku i ;

$C_{n, i-1}$ – status kondenzatora n u trenutku $i-1$;

K_n – maksimalni broj operacija nad kondenzatorom n ;

n_c – broj kondenzatora.

5.2.3 Ograničenje po naponima čvorova

Ograničenje po naponima čvorova predstavlja uslov koji se odnosi na vrednosti napona svakog čvora. Vrednosti napona svakog čvora razmatrane mreže se moraju kretati između minimalne i maksimalne vrednosti koje su definisane ograničenjem. Relacija kojom je uvaženo ovo ograničenje dato je u nastavku:

$$V_{min} \leq V_i \leq V_{max} \quad (6)$$

gde je:

V_{min} – minimalni dozvoljeni napon;

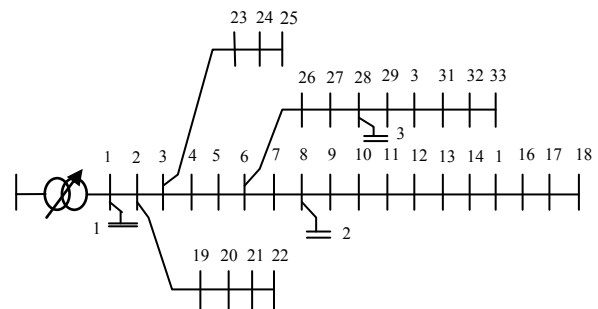
V_{max} – maksimalni dozvoljeni napon;

V_i – napon u i -tom čvoru;

i – indeks čvora [2].

6 VERIFIKACIJA ALGORITMA NA TEST MREŽI

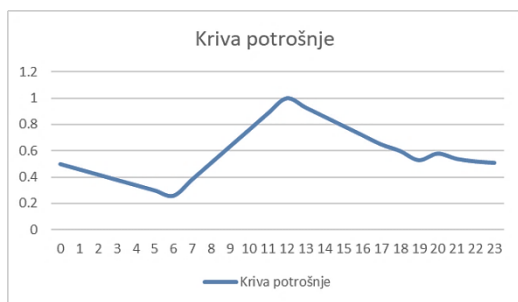
Na primeru test distributivne mreže od 33 čvora [1] korišćenjem GA, formiran je operacioni raspored rada kondenzatora za 24h. GA koristi rezultate tokova snaga, koji se izvršavaju sa predviđenim vrednostima potrošnje.



Slika 2. Izgled standardne IEEE test mreže [1]

6.1 Rešenje problema primenom GA

Algoritam je tokom izvršavanja krenuo od rasporeda koji daje rezultate prikazane u tabeli 1. U tabeli su prikazane vrednosti reaktivne energije injektirane u distributivnu mrežu iz prenosne (Q_{inj}), reaktivne energije injektirane iz distributivne u prenosnu (Q_{exp}), energiju gubitaka aktivne snage (P_{loss}) i prosečnog faktora snage. Prikazane vrednosti se odnose na period od 24h. Snaga potrošača je modelovana prema dnevnoj krivi potrošnje prikazanoj na slici 3.

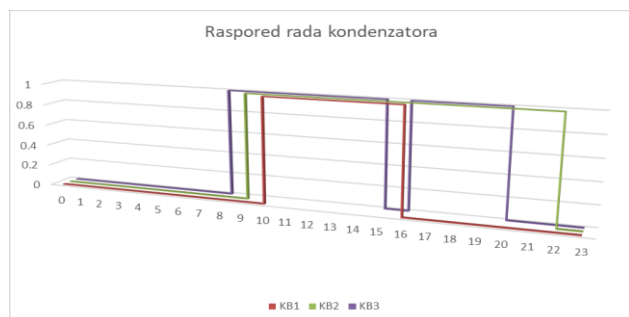


Slika 3. Dnevni dijagram potrošnje

Tabela 1. Rezultati dobijeni korišćenjem inicijalnog plana angažovanja

Q_{inj} [MVarh]	Q_{exp} [MVarh]	P_{loss} [MWh]	$\cos\phi$
4.085	22.768	2.3244	0.902

Nakon 450. iteracija GA, dobijen je operativni plan prikazan na slici 4, a rezultati korišćenja generisanog plana na test mreži su prikazani u tabeli 2

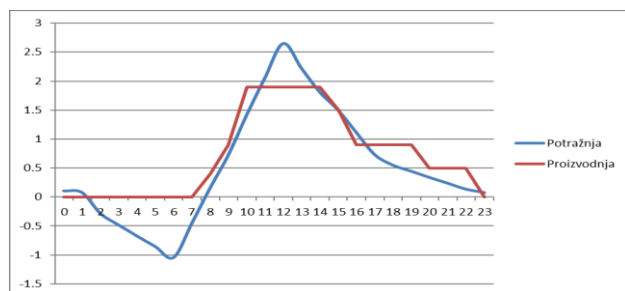


Slika 4. Grafički prikaz rasporeda rada kondenzatora

Tabela 2. Rezultati dobijeni korišćenjem generisanog plana angažovanja

Q_{inj} [MVarh]	Q_{exp} [MVarh]	P_{loss} [MWh]	$\cos\phi$
0.221	12.72	2.16	0.9699

Na slici 5 prikazan je odnos reaktivne energije koja se injektira iz prenosne u DM kad nisu uključeni kondenzatori (plava boja) i proizvodnje reaktivne snage od strane kondenzatora u skladu sa rasporedom angažovanja koji je za dati period dobijen primenom GA (crvenom bojom).



Slika 5. Potrošnja reaktivne snage DM i proizvodnja reaktivne snage od strane kondenzatora za period 24h

Iz tabele 2 se vidi da je ispoštovano ograničenje po dnevnim operacijama nad kondenzatorima i da je prosečni faktor snage u granicama ograničenja. Vidi se poboljšanje rešenja u vidu manjih gubitaka aktivne snage u poređenju sa prethodno razmatranim rešenjem. Poboljšanje prosečnog faktora snage, kao i manje proizvodnje reaktivne snage u periodima niske potrošnje, što se ogleda u manjim vrednostima Q_{exp} .

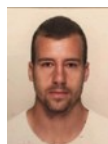
7 ZAKLJUČAK

U radu je razmatran problem optimalnog angažovanja kondenzatora za period od 24h, koristeći prognozirane vrednosti potrošnje. Nakon uvodnih razmatranja, čitaocu su ukratko predstavljene metode koje su u literaturi korišćene za rešavanje razmatranog problema. U radu je posebno razmatran genetski algoritam. Genetski algoritam je primenjen na IEEE mreži i za vreme izvršavanja od 7 sati generisao rešenje razmatranog problema. Uočeno je da se tokom vremena generišu sve kvalitetnija rešenja, u smislu boljih rezultata optimizacionih kriterijuma i pokazalo se da je jedan od problema genetskog algoritma upravo uslov zaustavljanja. Tako da je neophodno pronaći optimalan odnos kvaliteta rešenja i dozvoljenog vremena izvršavanja algoritma.

8 LITERATURA

- [1] A.Ulinuha, M.A.S.Masoum, S.M.Islam: Optimal Scheduling of LTC and Shunt Capacitors in Large Distorted Distribution Systems Using Evolutionary-Based Algorithms, *IEEE Trans. on PWRD*, vol. 23, Issue: 1, Jan. 2008.
- [2] S.Bajić: *Optimizacija napona i reaktivnih snaga u distributivnim mrežama*, master rad, Fakultet tehničkih nauka u Novom Sadu, Novi Sad, 2014.
- [3] J.Park, S.Nam, J.-K.Park: Control of a ULTC Considering the Dispatch Schedule of Capacitors in a Distribution System, *IEEE Trans. on PS*, vol. 22, Issue: 2, May 2007.
- [4] R.H.Liang, Y.S.Wang: Fuzzy-based reactive power and voltage control in a distribution system; *IEEE Trans. on PWRD*, vol. 18, no. 2, pp. 610-618, Apr. 2003.
- [5] G.Carpinelli, C.Noce: Voltage Regulators and Capacitor Placement in Three-phase Distribution Systems with Non-linear and Unbalanced Loads; *International Journal of Emerging Electric Power Systems*, No.4, 1-17, (2006).
- [6] T.Ghose, S.K.Goswami: Effects of unbalances and harmonics on optimal capacitor placement in distribution system; *Electric Power Systems Research*, 68, 2, 167-173, (2003)
- [7] T.Niimura, R.Yokoyama, B.J.Cory: A fuzzy coordination approach for multi-objective voltage and reactive power scheduling of an electric power system; in *Proc. 2nd IEEE Int. Conf. Fuzzy Systems*, pp. 267-272, Mar. 1993.
- [8] L.Zhang, W.Tang, J.Liang, P.Cong, Y.Cai: *Coordinated Day-Ahead Reactive Power Dispatch in Distribution Network Based on Real Power Forecast Errors*. IEEE Transactions on Power Systems. 31. 1-9. 10.1109/TPWRS.2015.2466435
- [9] Mitchell, Melanie (1996). *An Introduction to Genetic Algorithms*. Cambridge: MA: MIT Press. crp. 2, 2015.

Kratka biografija:



Branislav Vejnović rođen je u Novom Sadu 1994. godine. Master rad je odbranio 2019. godine na Fakultetu tehničkih nauka iz Elektrotehnike i računarstva – Elektroenergetski sistemi.

UPRAVLJAČKE STRATEGIJE U POSLOVNO-STAMBENIM OBJEKTIMA**CONTROL STRATEGIES FOR BUILDINGS**Miroslav Korica, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Rad se bazira na analiziranju i upravljanju HVAC sistemima. Upravlja se ventilacijom i klimatizacijom u zgradama i rešavaju se određeni problemi koji mogu da se pojave tokom izvođenja radova na zgradi. Sagledavaju se koraci u pripremi projekata zgrade, izvođenju radova, puštanju sistema u rad, prikupljanju i analizi podataka kao i održavanje zgrade za što duži životni vek..

Cljučne reči: *Fancoil, BMS, temperature, projektovanje, puštanje u rad*

Abstract – *The work is based on the analysis and management of HVAC systems. It is managed by ventilation and air conditioning in buildings and certain problems are solved which can occur during the execution of works on the building. It is based on the preparation of building projects, the execution of works, the commissioning of the system, the collection and analysis of data as well as the maintenance of the building for a longer life.*

Keywords: *Fancoil, BMS, temperature, designing, commissioning*

1. UVOD

Sistemi klimatizacije, grejanja i hlađenja-KGH, tj. HVAC (eng. Heating, Ventilation and Air Conditioning) igraju centralnu ulogu u regulisanju unutrašnjih uslova komfora i zdrave sredine za ljude u zgradi.

HVAC sistemi su važni i veoma popularni u projektovanju srednjih i velikih poslovno-stambenih zgrada jer obezbeđuju povoljne uslove za rad. Pod povoljnim uslovima za rad podrazumeva se održavanje željene temperature vazduha, željene vlažnosti vazduha itd. Ovaj deo se bavi strategijama kontrole za kompletne zgrade (građevine). Strategija za kompletan HVAC sistem mora uzeti u obzir interakcije među podsistemima. Sistem može funkcionisati u nekoliko različitih modova i uslovi svakog podsistema u svakom posebnom modu moraju biti specificirani. Praktične i teorijske tehnike projekata su opisane, što može koristiti za predviđanje kontrolisanih performansi u (pod) različitim uslovima.

BMS je bitan izvor informacija, koje se mogu preneti u druge aplikacije. Performanse zgrade i potrošnja energije se analiziraju softverom za nadzor i ciljanje energije, kao pomoć u efikasnom upravljanju energijom. Informacije iz BMS-a se koriste kao deo upravljanja postrojenjima, uključujući administraciju planiranog održavanja.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Velimir Čongradac, vanr. prof.

Važnost dobrog planiranja je naglašena kako bi sistem bio dobro specificiran, instaliran i pušten u rad. Rad i održavanje sistema kontrole nakon predaje je takođe obrađen. Finansijska korist investicije u BMS se može analizirati korišćenjem metoda troška životnog ciklusa opisanim u poslednjem odeljku.

2. OPERATIVNI MODOVI

Opis strategija kontrole podsistema indicira mnoge interakcije koje je neophodno uvrstiti u strategije kontrole. Prilikom projekata HVAC i kontrolnog sistema za novu zgradu, postoje nesumnjivo kombinacije podsistema koje ranije nisu bile dokumentovane i ključno je da se projekat teorijski preispita (izazove) po principu različitih „šta ako“ scenarija. Raznovrsnost mogućih, očekivanih i neočekivanih situacija je potrebno planirati i predviđene reakcije sistema kontrole ispratiti, kako bi se osiguralo da nema neočekivanih efekata.

Operativni modovi za HVAC sistem:

- Niska spoljna temperatura vazduha, blokiranje (za zaštitu postrojenja)
- Niska temperatura povratne vode, blokiranje (za zaštitu sistema)
- Isključenje postrojenja
- Optimalni start grejanja
- Optimalni start hlađenja
- Noćno hlađenje (Freecooling)
- Požar

Dokumentacija kompletne strategije kontrole sastoji se iz sledećeg:

- Šematski dijagram postrojenja
- Opis postrojenja
- Strategija kontrole
- Klauzule specifikacije
- BMS kontrolna lista
- Rezime operacije postrojenja
- Kontrolni dijagram

Važno je odabrati HVAC sistem i sistem kontrole koji su odgovarajući za zahteve zgrade (građevine) i operacije koje podržava. Nema prednosti u posedovanju sofisticiranog BMS-a niti u akumuliranju informacija koje nisu neophodne i neće biti upotrebljene. Neke proste kontrole, možda spremne postojeće kontrole isporučene od proizvođača postrojenja, mogu imati adekvatne performanse, njih treba koristiti osim ako ne postoji dobar razlog za instalaciju kontrolnog sistema po narudžbini. Snaga i fleksibilnost modernog BMS-a omogućava praktično implementaciju bilo koje strategije kontrole i kasniju

modifikaciju iste. Promene u zahtevima korisnika, razvoj u HVAC tehnologiji i razvijajuće politike o očuvanju energije osiguravaju da su inovacije i promene u projekatu gradnje kontinuiran proces.

Zahtevi očuvanja energije i želja da se umanjí ili eliminiše potreba za klimatizacijom su doveli do projekata zgrada koje su namenjene da funkcionišu na granici svojih mogućnosti u periodu predviđenih vremenskih uslova; više nije prihvatljivo da se osiguravamo od problema kroz predimenzioniranje fabrike.

Ovo je dovelo da raznovrsnosti zgrada, opsega od kompletno klimatizovanih, zatvorenih zgrada do onih sa kompletnom prirodnom ventilacijom. U mnogim slučajevima, ipak, javila se potreba za uvođenjem pomoćne opreme za prevazilaženje specifičnog vremena ili operativnih uslova [1][2].

3. FAN COIL JEDINICE

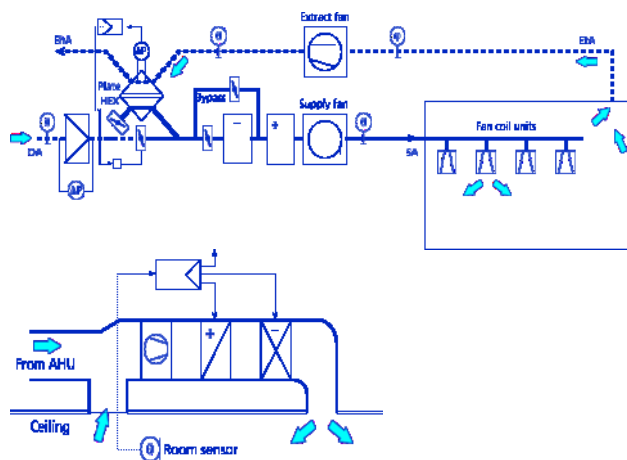
Fan coil jedinice su raspoložive u velikom rasponu konfiguracija, uključujući i one ispod podova, konzole projektirane za na zid ili ispod prozora, do uobičajenih koje se ugrađuju na plafon. Jedinice su povezane na dovod tople i hladne vode. Distribucija vode u zgradi je jednostavnija i zauzima manje prostora od distribucije tretiranog vazduha. Ovo omogućava fleksibilniju instalaciju fan coil jedinica od VAV sistema. FCU se nalaze u svim tipovima zgrada i sistem kontrole se razlikuje od lokalne kontrole termostata ventilatora do potpuno integriranih kompletnih sistema za zgrade.

Sam FCU recirkulira vazduh iz prostorije, grejanje ili hlađenje je neophodno za održavanje definisane tačke vazduha. Svež vazduh je obezbeđen kroz centralnu jedinicu koja dovodi 100% spoljni vazduh na kontrolisanoj temperaturi i uobičajeno kontrolisanoj vlažnosti. Uključivanje temperature primarno obezbeđenog vazduha u strategiju kontrole omogućuje uštedu energije kroz kontrolu operacije na izmenjivaču toplote koji se koristi za besplatno hlađenje. Fan coil jedinice nemaju potrebu za odvlaživanjem osim kondenzacije vlage na izmenjivaču. Fan coil jedinice zato moraju uvek imati kondenzacionu tacnu i mogućnost otklanjanja kondenzata. Poželjan metod je da se reši odvlaživanje primarnog spoljnog vazduha na kontrolisanu unutrašnju vlažnost na željenom nivou, ovo je obično dovoljno da spreči kondenzaciju na FCU u hlađenju. Uobičajena praksa je da se kanališe primarni vazduh u svaki FCU, gde se meša sa recirkuliranim sobnim vazduhom u ventilacionoj jedinici.

U integrisanom sistemu, svaka inteligentna fan coil jedinica sadrži svoj kontroler koji komunicira sa BMS-om. Definisane temperature prostorije i raspored rada mogu biti podešeni daljinski za svaku jedinicu ili grupu jedinica koje rade zajedno. Jedna od mana fan coil jedinica je potencijalni nivo održavanja neophodan i remećenje prilikom identifikacije i servisiranja jedinica u kvaru koje su u prostoru u upotrebi. BMS se koristi da pruži upozorenje za kvarove ili potrebu za rutinskim održavanjem, poput potrebe za promenom filtera.

Slika 1 pokazuje jedinicu za vazduh za sistem fan coil, koji isporučuje 100% spoljnog vazduha za sve jedinice u prostoriji. Izmenjivač toplote je postavljen za predaju toplote između izbacnog i svežeg vazduha koji se uzima.

Vazduh se uzima konstantnim protokom za vreme perioda zauzetosti; ventilator za izvlačenje je vezan da radi u isto vreme. Vlažnost dostavljenog vazduha se kontroliše da bi se održala na nivou koji je komforan i treba sprečiti, ili bar minimizovati, kondenzaciju fan coil-a.



Slika 1. Integriran sistem fan coil jedinica. Temperatura ubacnog vazduha je podešena prema spoljoj temperaturi.

4. INFORMACIONE TEHNOLOGIJE

Velika opravdanost za instalaciju BMS-a je u prevenciji gubitka energije. Ispravan i efikasan HVAC će doprineti efikasnoj upotrebi energije ali dodatno je neophodno da energetski menadžer prima tačne i pravovremene informacije o utrošku energije u zgradama za koje je odgovoran. Pristup generalno usvojen u UK se naziva *nadzor i ciljanje energije* (M&T). Termin nadzor i ciljanje u suštini ne impliciraju upotrebu BMS-a ili bilo koje opreme za prikupljanje podataka i analizu. M&T zahteva da se podaci o potrošnji energije redovno prikupljaju, sumiraju i porede sa planiranim(ciljanim) ciframa potrošnje. Kompjuterizovano prikupljanje i analiza podataka čine M&T snažnim oruđem za kontrolu i smanjenje potrošnje energije [3].

4.1. Nadzor i ciljanje

M&T ima dve vodeće funkcije:

- kontrola trenutne upotrebe energije, nadzorom potrošnje i poređenjem sa istorijskim podacima i relevantnim podacima za slične zgrade
- poboljšanje u efikasnosti potrošnje energije podešavanjem budućih ciljeva.

U velikim zgradama ili kompleksima, efektivno upravljanje energijom zahteva da se uspostave oblasti računanja, nazivaju se centri računanja energije (EAC). Računanje implicira i odgovornost za potrošenu energiju i nadležnost za kontrolisanje potrošnje. Potrošnja energije za EAC treba da bude

- merljiva
- upravljiva
- u odnosu sa merenim aktivnostima.

EAC može biti deo, proizvodni proces energetski intenzivan, iznajmljeni deo zgrade ili pojedinačna zgrada na parceli. Standardizovane performanse energije se inicijalno utvrđuju za svaki EAC, koje postavljaju

potrošnju energije u odnosu na odgovarajuće varijable, kao stepen-dani(degree-days) ili rezultat proizvodnje.

Nadzorni deo M&T procesa uključuje četiri faze:

Prikupljanje podataka. Podaci o potrošnji energije se prikupljaju. Kompletan sistem uključuje merače za struju, gas i vodu na svakom EAC-u, osposobljene za slanje podataka u centralnu bazu. Raspoloživost drugih relevantnih podataka mora se uzeti u obzir, npr. podaci o stepen-danima ili vrednosti proizvodnje.

Analiza podataka. Podaci primljeni od računara se kontrolišu na greške i skladište u formi adekvatnoj za dalju analizu. Analiza se sprovodi u skladu sa potrebama sistema, uobičajeno sa nedeljnim i mesečnim sumama. Svaki rezultat koji je indicija problema ili kvara trebalo bi da odmah pokrene upozorenje.

Izveštavanje. Sistemski menadžment izveštaji koji prikazuju potrošnju energije za svaki EAC uporedo sa ciljevima.

Akcije. Esencijalno je da postoji struktura rukovođenja koja će efikasno upotrebiti izveštaje koje generiše M&T sistem.

Da bi M&T sistem bio efikasan, sve četiri faze moraju biti implementirane. Prikupljanje podataka, koliko god ispravno, koje se ne prati niti izaziva neku akciju ne služi ničemu. Efikasan M&T zahteva punu podršku menadžmenta. Otprilike, M&T može se očekivati da umanju račun za energiju za oko 5%, veće uštede su moguće tamo gde su prepoznati kvarovi opreme ili ozbiljni propusti u kontroli kao rezultat podataka merenja. Ova cifra dozvoljava da se odabere odgovarajuć nivo M&T-a u odnosu na ukupni trošak energije. Sistem upravljanja energijom ima mnoge karakteristike kao BMS. U zgradi sistem instaliran za nadzor energije i ciljanje može deliti opremu sa BMS-om i može se upravljati sa istog terminala i od strane istog osoblja. Ipak, oni nisu isto. BMS pruža kontrolu u realnom vremenu servisa u zgradi, dok se M&T sistem bavi prikupljanjem podataka i analizom istorijata.

5. UPRAVLJANJE ZGRADOM

Forma nabavke definisana od strane klijenta može imati važan uticaj na način kako će se kontrole projektovati i instalirati. Bez obzira na način nabavke, možemo definisati glavne učesnike u procesu izgradnje kao:

- *Klijent:* kupac za koga se zgrada nabavlja i ko snosi trošak.
- *Izvodjači:* kompanije koje grade zgradu. Mogu postojati firme koje proizvode i instaliraju specijalne podsisteme i rade kao podizvodjači za glavnog izvođača.
- *Konsultanti:* firme koje pružaju projekat i servis kontrole troškova i nezavisne su od komercijalnih interesa izvođača

5.1 Projektovanje i gradnja

U ovoj varijanti, klijent ugovara samo jednog izvođača koji ima odgovornost i za projekat i izgradnju. Klijent može ovlastiti savetnika koji radi kao zaposleni agent, daje savete u pripremi dokumentacije, procenama tendera i pružanju nezavisnih saveta u toku projekta. Projektovanje i gradnja ima prednosti:

- Klijent ima jednu tačku odgovornosti od jedne organizacije.
- Kako izvodjač ima odgovornost za projekat i izvođenje, uštede bi trebale biti moguće.

Dobro napisana klijentova dokumentacija je osnova za postizanje zadovoljavajućih rešenja kontrole u završenoj zgradi. Nedovoljno detaljna inicijalna dokumentacija može uzrokovati da izvodjač dostavi minimalnu specifikaciju. Poznata je varijanta razvoj i izvođenje, gde klijent koristi konsultanta za projekat da pruži opšti projekat, pre pribavljanja tendera od izvođača koji razvijaju i dovršavaju projekat i potom grade zgradu.

Jedna firma je ovlašćena za projekat, rukovođenje i predaju projekta. Ovo je slično kao projektovanje i gradnja, samo što ovaj izvođač ne izvodi gradnju, već koristi ugovorenog izvođača. Uobičajene varijante projektovanja i rukovođenja su:

- *Izvodjač.* Organizacija za projekat i rukovođenje koja projektuje i vodi radove, generalno za naknadu, i predaje projekat zapošljavajući druge da projektuju i/ili grade.
- *Konsultant.* Organizacija koja je klijentov agent projektuje, vodi radove i obezbeđuje izvođače koji dalje ulaze u direktan odnos sa klijentom.

6. PROJEKAT I SPECIFIKACIJA SISTEMA KONTROLE

Prvi korak je da klijent postavi zahteve za zgradu kao projektantski koncept. Što su zahtevi jasnije postavljeni u projektantskoj dokumentaciji, bolje će biti kontrolisan ceo projekat. Šematski projekat procesa treba da uzme u obzir operative zahteve krajnjih korisnika i opšti nivo kontrole i IT sisteme. Klijent može imenovati savetnika u ovoj fazi da pruži nezavisne savete sada i kasnije u toku projekta. Upotreba savetnika može biti ključna za uspeh projekta gde klijent ima malo iskustva u materiji. Kada je projekat dogovoren, projekat se prosleđuje konsultantima, čiji je posao da pripreme finalni projekat koji će biti osnova za ugovor o izgradnji. Za kompleksnu zgradu, kontrolni projekat će verovatno raditi konsultant specijalista, ili nezavisna firma ili grupa specijalista u okviru M&E konsultanata. Nivo detalja zavisi od projekta do projekta, može biti kompletna specifikacija, ili prosta specifikacija performansi, koja ostavlja da specijalista kontrole odradi detaljan projekatantski rad i odabere dobavljače.

Podela odgovornosti među izvođačima mora biti jasna u specifikacijama. Primer relevantan za kontrolu su kablovi. Kablovi za struju i signalni kablovi mogu biti od različitih izvođača i signali za BMS mogu delom biti preneti preko deljene IT mreže.

7. PUŠTANJE U RAD

Ispravno puštanje u rad je glavno za zadovoljavajuću operaciju HVAC sistema i ključno je da se za ovaj zadatak opredeli dovoljno vremena i resursa. Kako je puštanje u rad poslednja velika operacija u procesu izgradnje i sistem kontrole je poslednji koji se pušta u rad, postoji velika opasnost da će se odvijati pod velikim vremenskim pritiskom ili čak nakon što je zgrada useljena. Iskustvo je pokazalo da ovo može stvoriti mnoge probleme, sa neprihvatljivo velikim postotkom zgrade koja ne

funkcionira ispravno, sa posledicama velike potrošnje energije i nezadovoljstvom korisnika. Ovaj deo naglašava važnost da puštanje u rad mora imati svoje adekvatno mesto u procesu nabavke od početka planiranja projekta.

Puštanje u rad kontrolnog sistema je zavisno od zadovoljavajućeg rada električnih i mehaničkih usluga. Sistem takođe može imati interakciju sa specijalnim uslugama kao što je obezbeđenje, kontrola pristupa i IT sistem. Što je veći stepen integracije, više planiranja je neophodno i pažnje da se definišu polja odgovornosti [4]. Zapisi svih faza pre puštanja u rad i puštanja u rad svih drugih aspekata zgrade moraju biti izdati pre puštanja u rad automatskog sistema kontrole. Tabela 1 sumira faze procesa nabavke i aktivnosti relevantne za puštanje u rad kontrolnog sistema.

Tabela 1. Aktivnosti puštanja u rad za vreme procesa nabavke

Faza	Aktivnost
Tehnička dokumentacija	Ciljevi puštanja u rad Imenovati menadžera za puštanje u rad
Projekat	Smanjiti potrebu za puštanjem u rad Projekat za puštanje u rad
Instalacija	Pre-puštanja u rad Puštanje u rad
Predaja	Svedočenje O&M dokumentacija Obuka korisnika
Rad	Nadzor i izveštavanje Potvrda sistema Fino podešavanje Ponovno puštanje

Originalna tehnička dokumentacija će biti osnova za organizaciju puštanja u rad. Ovo je faza u kojoj se organizacija celog procesa nabavke odlučuje. Za velike projekte, imenovanje menadžera za puštanje u rad treba razmotriti, sa odgovornošću da se osigura da su zahtevi za puštanje u rad uzeti u obzir u svim fazama. Menadžer će biti odgovoran za imenovanje tima i da osigura da su adekvatni resursi za puštanje u rad uneti u specifikaciju. Pre početka puštanja u rad, sva oprema mora biti instalirana i pripremljena za puštanje u rad. Sve kontrole koje se povezuju na BMS treba da su puštene u rad, ili da se puštaju paralelno sa BMS-om. HVAC postrojenje takođe treba da je pušteno u rad, sve stavke pod kontrolom BMS moraju biti puštene u rad i sistem očišćen i ventiliran. Sve elekto-motorne sigurnosne blokade i sigurnosni uslovi moraju biti implementirani i operativni. Puštanje u rad BMS-a sastoji se od dve glavne aktivnosti:

- provera da kontrolni sistem radi
- podešavanje svih parametara i prekidača na odgovarajuće vrednosti, uključujući kontrolne petlje za podešavanje.

8. ZAKLJUČAK

Nekoliko tehnika je raspoloživo da se predvide performanse zgrade, ili analogno modelovanje, maketa u punoj razmeri, ili razne tehnike matematičkog modelovanja. Modelovanje može predvideti potencijalne probleme kontrole i omogućiti da se isprave u fazi projektovanja.

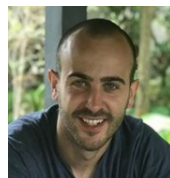
BMS se može koristiti kao podrška menadžmentu i ciljanju energije kroz merenje komponenti potrošnje energije i slanjem podataka u M&T aplikacioni paket.

Kada su HVAC i kontrolni sistem instalirani, efikasno puštanje u rad je ključno. U modernim kompleksnim zgradama, kontrolni sistem je u interakciji sa drugim sistemima i pažnja je neophodna u planiranju procesa puštanja u rad i podeli područja odgovornosti.

9. LITERATURA

- [1] Honeywell *Engineering manual of automatic control* SI ed Minneapolis: Honeywell Inc.) (1995)
- [2] Siemens *MSR Planungshandbuch* (CD-ROM) (Karlsruhe: Siemens AG) (1997)
- [3] ETSU *Computer-aided monitoring and targeting for industry* Good Practice Guide 31 (Harwell: ETSU) (1991)
- [4] Wild L J *Commissioning HVAC systems. Division of responsibilities* AG 3/89 (Bracknell: Building Services Research and Information Association) (1989)

Kratka biografija:



Miroslav Korica, rođen je 24.12.1991. godine u Velebitu. Osnovnu školu „Jovan Jovanović Zmaj“ završio je u Kanjiži. Srednju tehničku školu je pohađao u Subotici, smer elektrotehničar računara koju završava 2010. godine. Iste godine upisuje Fakultet Tehničkih Nauka u Novom Sadu, smer Računarstvo i automatika. Kasnije se usmerava na Automatiku i upravljanje sistemima gde je diplomirao u junu 2015. godine i zatim upisuje master akademske studije na Fakultetu Tehničkih Nauka.

UTICAJ DISTRIBUTIVNIH GENERATORA NA KOORDINACIJU ZAŠTITE IZMEĐU PREKIDAČA I OSIGURAČA U RADIJALNIM DISTRIBUTIVNIM MREŽAMA**THE IMPACT OF DISTRIBUTION GENERATORS ON PROTECTION COORDINATION IN FUSE SAVING SCHEME OF RADIAL DISTRIBUTION GRIDS**Helena Petričević, Vladan Krsman, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U radu je prikazano kako priključenje jednog ili više distributivnih generatora utiče na tradicionalni proračun relejne zaštite u radijalnim distributivnim mrežama. Cilj ovog rada je da se utvrdi koliko je narušen kvalitet proračuna relejne zaštite koji se koristi u svakodnevnom upravljačkim akcijama unutar distributivne mreže. Poseban akcenat je stavljen na koordinaciju zaštite između prekidača i osigurača. Verifikacija problema je izvršena na mreži od 50 čvorova.

Ključne riječi: Relejna zaštita, Distributivna mreža, Distributivni generator

Abstract – This paper describes how the presence of one or more distribution generators affects the traditional calculation of relay protection applied in radial distribution networks. The goal of this paper is to determine the existence of any issue in relay protection calculation in the day-to-day management of the distribution network. The focus was put to protection coordination in Fuse Saving Scheme. Verification of results was carried out on network with 50 nodes.

Keywords: Relay protection, Distribution network, Distribution generator, Fuse Saving Scheme

1. UVOD

Uvođenjem distributivnih generatora oslobađa se prenosni kapacitet, čime se povećava faktor efikasnosti samih mreža. Smanjuju se gubici u prenosu energije, a samim tim i troškovi eksploatacije. Naponski profil u distributivnim mrežama je povoljniji, a pouzdanost i kvalitet snabdjevanja krajnjih potrošača je na znatno većem nivou [1].

Pored pozitivnih osobina koje donosi priključenje distributivnih generatora postoje i loše osobine koje mogu da utiču na stabilnost čitavog distributivnog sistema. Uvođenjem distributivnih generatora mijenja se koncept radikalnosti, mreža postaje aktivna i mnogo dinamičnija.

Priključenje distributivnih generatora direktno na distributivnu mrežu uzrokuje da se mjesto kvara napaja iz više izvora, čime se mijenjaju vrijednosti varijabli poremećenog stanja na osnovu kojih je podešena relejna zaštita mreže [1].

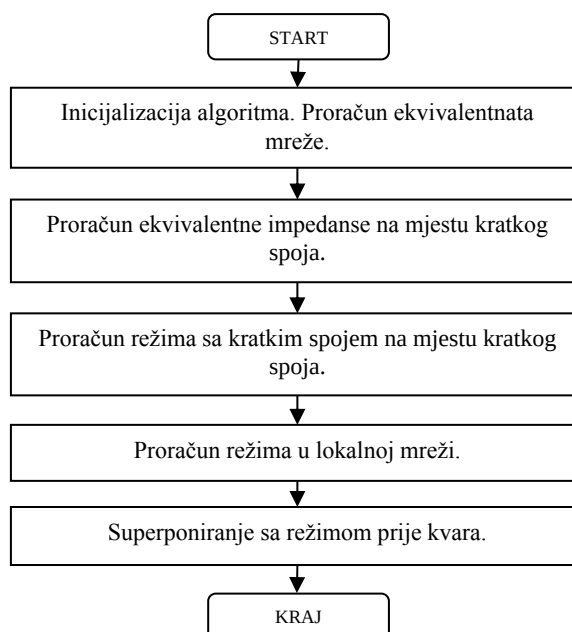
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je dr Vladan Krsman, docent.

Selektivnost između prekidača za napajanje i nizvodnih uređaja očekuje se u tipičnom distributivnom sistemu. U nekim situacijama (u slučaju prolaznih kvarova), prekidač za napajanje je u slučaju kvara podešen tako da reaguje prije osigurača koji se nalazi nizvodno od prekidača za napajanje. Ova praksa poznata je kao "Fuse Saving" šema [2]. Posebna pažnja u ovom radu će biti posvećena uticaju distributivnih generatora na rad (koordinaciju) relejne zaštite u "Fuse Saving" šemi.

2. PRORAČUN KRATKIH SPOJEVA

Osnova za dobro podešenje relejne zaštite je poznavanje vrijednosti struja kvara kroz zaštitne uređaje. U tu svrhu u radu je korišćen Admitantno impedantni algoritam, čiji blok dijagram je prikazan na slici 2.1.



Slika 2.1 – Admitantno impedantni algoritam[3]

3. RELEJNA ZAŠTITA

Osnovna uloga relejne zaštite je da obezbijedi siguran rad elektroenergetskih sistema i da zaštiti korisnike i opremu. Pošto su kvarovi u distributivnim mrežama česta pojava, cilj relejne zaštite je da uticaj tih kvarova na potrošače svede na minimum. Cilj relejne zaštite je da detektuje i na odgovarajući način reaguje na sve moguće tipove kvarova koji mogu da se dese u šticejnoj mreži, ali na takav način da bez napajanja ostaje minimalan broj potrošača. Nije

uvijek moguće obezbijediti ispunjenje svih uslova koji se postavljaju pred relejnu zaštitu i često se moraju praviti kompromisi, kako bi se našlo rješenje najbliže idealnom.

Sa povećanjem potrošnje i opterećenja sistema, sve je značajnija uloga prekostrujne zaštite. Neophodne karakteristike prekostrujne zaštite su osjetljivost, selektivnost, brzina, pouzdanost i ekonomičnost [4].

Osjetljivost je sposobnost releja da odreaguje na najmanju očekivanu struju kvara. **Selektivnost** je sposobnost releja da razlikuje stanja u kojima je potrebno odmah reagovati, reagovati sa nekom vremenskom zadržkom ili uopšte ne reagovati. Svaki relej ima svoju zonu šticećenja, ali pored toga, vrlo često služi i kao rezervna zaštita opremi u drugim zonama. **Brzina** je sposobnost releja da reaguje u predviđenom vremenskom intervalu. Struje kratkog spoja direktno utiču na naprezanja opreme, oštećuju je i smanjuju vijek trajanja. Stoga je u interesu kvar otkloniti što je prije moguće. **Pouzdanost** relejne zaštite bi trebalo da obezbijedi da sistem radi na predviđen način, kako je i konfigurisan. Pouzdanost predstavlja sigurnost da će relejna zaštita pravilno raditi, bez obzira na spoljne faktore. **Ekonomičnost** predstavlja obezbeđenje maksimuma zaštite uz minimalne troškove. Troškovi su najvažniji faktor koji dominantno utiče na sve prethodno navedene zahtjeve. Bitno je naći kompromisno rješenje između cijene i nivoa zaštite[4].

4. UTICAJ DISTRIBUTIVNIH GENERATORA NA RELEJNU ZAŠTITU

Distributivne mreže su konstruisane i planirane da budu pasivne, da se napajanje vrši u smjeru iz centralizovanog izvora (napojna transformatorska stanica VN/SN) ka individualnim potrošačima. Sekcije se sastoje od monofaznih, dvofaznih i trofaznih vodova, na čijim krajevima se nalaze potrošači. Kada se projektovala distributivna mreža, nisu postojali distributivni generatori i nije se vidjela njihova potreba, tako da su distributivne mreže projektovane kao radijalne mreže sa napajanjem samo sa jedne strane. S obzirom da su centralizovani izvori energije i dalje dominantni, distributivni izvori se moraju prilagoditi već postojećim konfiguracijama.

Pouzdanost, sigurnost i kvalitet napajanja potrošača je zahtjev koji se stavlja pred projektante koji treba da priključe distributivne generatore u već postojeće distributivne mreže. Skup problema vezan za relejnu zaštitu, koji se mogu pojaviti priključenjem distributivnih izvora su [5]:

- pogrešno okidanje releja,
- smanjen doseg distantnih releja,
- greške u koordinaciji između zaštitnih uređaja,
- neželjen ostrvski pogon,
- automatsko ponovno uključenje,
- ugrožavanje osjetljivosti zaštite.

Greške u koordinaciji između zaštitnih uređaja

Dobrom koordinacijom među zaštitnim elementima se omogućuje selektivnost u radu. Selektivnost obezbjeđuje da se pri trajnim kvarovima, taj kvar izoluje na vrijeme, tako da ostavlja bez napajanja najmanji broj potrošača.

Radijalne distributivne mreže obično imaju neusmjerene prekostrujne releje, prekidače sa automatskim ponovnim uključanjem (APU, eng. *Recloser*) i osigurače u svojim zaštitnim šemama. Kako ovi uređaji ne uzimaju u obzir

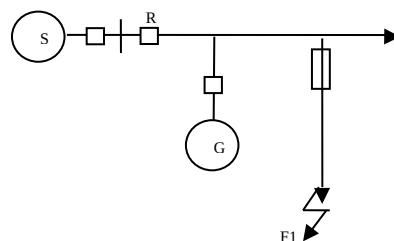
smjer toka snage, to može dovesti do nepotrebnog ili nekorektnog djelovanja zaštite u slučaju kada distributivni generatori doprinose struji kratkog spoja.

Priključenjem distributivnih generatora može doći do ugrožavanja koordinacije. Koordinacija među zaštitnim elementima je urađena sa pretpostavkom da je mreža radijalna, sa jedinstvenim izvorom napajanja.

Postoji nekoliko šema koje se koriste za zaštitu distributivnih mreža. Najčešće su šeme u kojima figuriše prekostrujna zaštita i šeme koje predstavljaju kombinaciju prekostrujnih releja sa automatskim ponovnim uključanjem i osigurača.

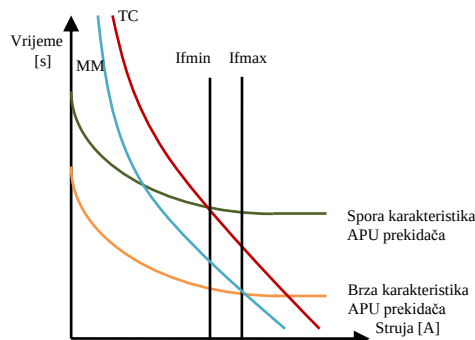
Na slici 4.1 je prikazana distributivna mreža sa zaštitnom šemom koja osiguračima ne daje prioritet u otklanjanju kvara ("*Fuse Saving*").

U slučaju da se kvar desio na mestu F1 (slika 4.1), APU prekidač bi po svojoj brznoj karakteristici trebao da otvori prekidač. Prekidač ostaje u tom stanju određeno vrijeme, dok APU ne inicira njegovo ponovno zatvaranje. U slučaju da je kvar prolaznog tipa, to vrijeme omogućava da se kvar sam od sebe otkloni. Ukoliko je kvar i dalje prisutan, ponovo se izvršava ista akcija i dalje pod pretpostavkom da je kvar prolazan. Ako i posle dvije operacije APU prekidača kvar nije otklonjen, smatra se da je kvar trajnog tipa [6].



Slika 4.1 – "*Fuse Saving*" šema

Nakon ove dvije operacije, prekostrujna karakteristika APU prekidača se prebacuje sa brze na sporu i na taj način omogućava da osigurač djeluje prije APU prekidača i izoluje dio mreže u kojoj je nastao kvar. Da bi se ovakav scenario u stvarnosti odigrao, APU prekidač i osigurač moraju da budu dobro koordinisani.



Slika 4.2 – Koordinacija između APU prekidača i osigurača

Na slici 4.2 su prikazane brza i spora karakteristika APU prekidača i dvije karakteristike osigurača. Dvije karakteristike osigurača su na dijagramu označene sa MM (*Minimum Melting*) i TC (*Total Clearing*). One se razlikuju po tome što MM kriva pokazuje vrijeme početka

topljenja osigurača usljed struje kvara, a TC kriva pokazuje vrijeme koje je potrebno da se osigurač potpuno istopi i otkloni kvar.

Ukoliko je struja kvara u okvirima opsega (I_{fmin} , I_{fmax}), tada će ovakav dio mreže biti dobro koordinisan. Vidi se da će za struje kvara, koje se nalaze u pomenutom opsegu, brza karakteristika APU prekidača prva biti aktivna. Zatim, ukoliko APU ne odreaguje, osigurač služi kao rezervna zaštita. Ukoliko i on u predviđenom vremenu ne prekine napajanje mjesta kvara, spora karakteristika APU prekidača služi kao rezerva.

Ako se razmatra slučaj da je distributivni generator priključen, dolazi do promjena koje mogu da utiču na koordinaciju sistema:

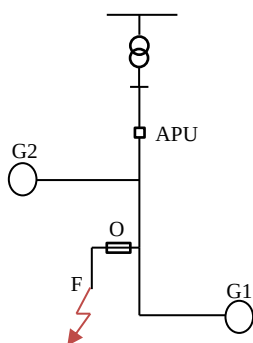
- Vrijednosti struja I_{fmin} i I_{fmax} su promijenjene usljed doprinosa distributivni generatora,
- Struja koju mjeri APU prekidač nije ista kao struja koja prolazi kroz osigurač.

Moguće je da struja kvara poraste i da izađe iz predviđenog opsega. Ta situacija dovodi do ugrožavanja zaštite, jer je koordinacija izgubljena. Struja viđena od strane APU prekidača i osigurača može da se razlikuje. To zavisi od konfiguracije mreže, od mjesta na kom je priključen distributivni generator, kao i od snage tog generatora.

5. VERIFIKACIJA PROBLEMA

U ovom radu izvršena je verifikacija problema na test mreži sa 50 čvorova, jednim napojnim transformatorom i 49 sekcija. Vršene su provjere koordinacije između APU prekidača koji se nalazi na glavnom izvodu (slika 5.1 – APU) i osigurača (slika 5.1. – O) koji štiti izvod na kom se simulira kvar. Provjere su vršene bez generatora, zatim pri priključenju jednog ili dva generatora. Snage generatora su mijenjane u različitim opsezima, što će biti prikazano u primjerima koji slijede.

Vrijednosti struja prikazanih u tabelama rezultat su proračuna režima delta kola. Proračun je vršen upotrebom Admitantno impedantnog algoritma.



Slika 5.1–Test mreža

Prvi primjer – simuliran je jednopolan kratak spoj na mjestu F - slika 5.1. Priključen je jedan generator G1 čija se snaga mijenja u opsegu od 12 do 67 MVA, sa korakom od 5 MVA. U tabeli 5.1 su prikazani rezultati proračuna.

U tabelama 5.1 i 5.2 su takođe prikazani i rezultati proračuna kada na mrežu nije priključen nijedan generator, odnosno kada su snage generatora 0 MVA. S obzirom da je koordinacija među zaštitnim elementima

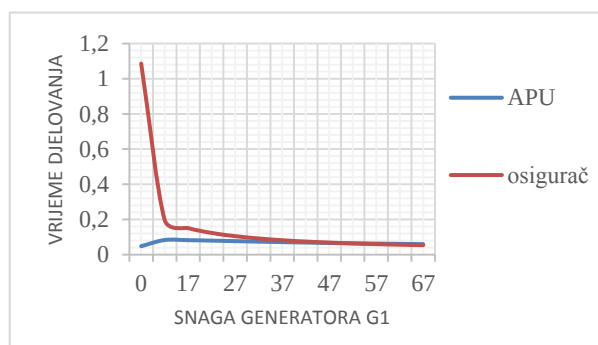
urađena sa pretpostavkom da je mreža radijalna, sa jedinstvenim izvorom napajanja u ovom slučaju neće doći do loše koordinacije.

Tabela 5.1 – Rezultati proračuna koordinacije zaštite u slučaju jednopolnog kratkog spoja na mjestu F kada je na mrežu priključen jedan generator

Snaga G1 [MVA]	Struja kroz APU [A]	Struja kroz osigurač [A]	Vreme delovanja APU prekidača [s]	Vreme delovanja osigurača [s]
0	2610.22	2610.20	0.048017	1.08575
12	1912.29	5785.54	0.0831907	0.1975690
17	1924.23	6606.03	0.0822621	0.1509080
22	1957.94	7308.34	0.0797310	0.1232550
27	1999.43	7921.91	0.0767852	0.1050830
32	2043.01	8464.94	0.0738773	0.0918601
37	2086.17	8950.08	0.0711707	0.0819564
42	2127.76	9386.71	0.0687125	0.0742507
47	2167.29	9782.07	0.0665022	0.0684967
52	2204.59	10141.90	0.0645214	0.0635839
57	2239.66	10471.00	0.0627459	0.0597904
62	2272.58	10773.10	0.0611515	0.0563790
67	2303.45	11051.60	0.0597156	0.0537546

Iz table 5.1 se može zaključiti da će za snage generatora veće od 47 MVA doći do loše koordinacije u "Fuse Saving" šemi, odnosno da će osigurač odreagovati prije APU prekidača. Struje kroz APU prekidač prikazane u tabeli 5.1 predstavljaju struje koje vidi relej nulte komponente (mjeri trostruku nultu komponentu struje, odnosno zbir kompleksnih struja faza L1, L2 i L3).

Na slici 5.2 su prikazane krive djelovanja APU prekidača i osigurača pri različitim snagama generatora. Tačka gdje se krive sijeku pokazuje snagu generatora pri kojoj će doći do loše koordinacije zaštite.



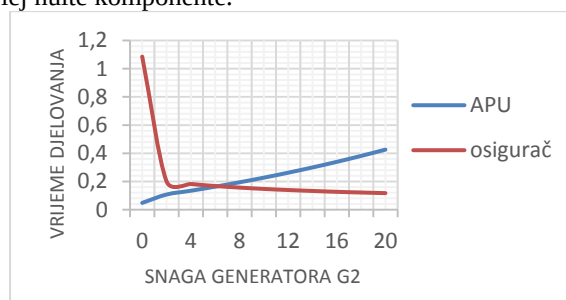
Slika 5.2 - Krive djelovanja APU prekidača i osigurača u slučaju jednopolnog kratkog spoja kada je na mrežu priključen jedan generator

Drugi primjer – simuliran je takođe jednopolan kratak spoj na mjestu F. Priključena su dva generatora G1 i G2. G1 je snage 10 MVA i tokom proračuna se snaga ne mijenja. Snaga generatora G2 je u opsegu od 2 do 20 MVA i mijenja se u koracima od po 2 MVA. U tabeli 5.2 su prikazani rezultati proračuna.

Tabela 5.2 – Rezultati proračuna koordinacije zaštite u slučaju jednopolnog kratkog spoja na mjestu F kada su na mrežu priključena dva generatora

Snaga G1 [MVA]	Snaga G2 [MVA]	Struja kroz APU [A]	Struja kroz osigurač [A]	Vreme delovanja APU prekidača [s]	Vreme delovanja osigurača [s]
0	0	2610.2	2610.2	0.048017	1.08575
2	10	1663.4	5737.9	0.107227	0.201034
4	10	1474.3	6017.2	0.134086	0.181996
6	10	1329.0	6264.3	0.163000	0.167694
8	10	1213.5	6486.9	0.193953	0.156479
10	10	1119.1	6689.9	0.226973	0.146986
12	10	1040.4	6876.7	0.262116	0.139182
14	10	973.44	7049.7	0.299466	0.132580
16	10	915.76	7211	0.339126	0.126430
18	10	865.43	7361.8	0.381224	0.121584
20	10	821.06	7503.5	0.425905	0.117158

Iz tabele 5.2 se može zaključiti da će za snage generatora G2 veće od 8 MVA doći do loše koordinacije u "Fuse Saving" šemi, odnosno da će osigurač odreagovati prije APU prekidača. Struje kroz APU prikazane u tabeli 5.2, kao i u prethodnom primjeru predstavljaju struje koje vidi relej nulte komponente.



Slika 5.3 - Krive djelovanja osigurača i APU prekidača u slučaju jednopolnog kratkog spoja kada su na mrežu priključena dva generatora

Na slici 5.3 su prikazane krive djelovanja APU prekidača i osigurača pri različitim snagama generatora. Tačka gdje se krive sijeku pokazuje snagu generatora pri kojoj će doći do loše koordinacije zaštite.

6. ZAKLJUČAK

U radu je razmatrana koordinacija reagovanja zaštitnih uređaja u "Fuse Saving" šemi, kada u mrežu nema generatora ili kada su na mrežu priključeni jedan ili dva generatora. Ovakav način zaštite primjenjuje se da bi se zaštitilo topljenje osigurača usljed prolaznih kvarova.

U slučaju priključenja jednog generatora u dubini mreže do loše koordinacije dolazi tek za prilično visoke i nerealne vrijednosti snage distributivnog generatora s obzirom da je snaga napojnog transformatora skoro dva puta niža. Na primjer, ako se uzmu ekstremni uslovi rada sa potrošnjom 20% većom od snage transformatora, priključenje generatora ove snage bi pravilo inverzan tok snage na napojnom transformatoru. Na osnovu ove analize može se reći da je zaštita dobro koordinisana.

U slučaju priključenja jednog generatora u dubini mreže i drugog koji se nalazi na putanji između dva zaštitna uređaja, do loše koordinacije dolazi kada ukupna generacija iznosi oko 70% snage napojnog transformatora. Može se reći da u ovom slučaju zaštita nije dobro koordinisana.

Koordinacija zaštite iz razmatranih primjera zavisi od mjesta i broja priključenih generatora. Zaključak je da u zavisnosti od konkretne potrebe priključenja generatora zaštitu treba prilagoditi, odnosno prepodesiti podešenje APU prekidača ili eventualno primijeniti neku drugu vrstu zaštite.

7. LITERATURA

- [1] James A. Silva, Hammed B. Funmilayo, Karen L. Butler – Purry, *Impacts of Distribution Generator on the IEEE 34 Node radial Test Feeder with Overcurrent Protection*, Power Symposium, 2007. NAPS '07. 39th North American, pp. 49-57
- [2] B. Hussain, S. M. Sharkh, S. Hussain, and M. A. Abusara, *An Adaptive Relaying Scheme for Fuse Saving in Distribution Networks With Distributed Generation*, IEEE Transactions on Power Delivery Volume:28, Issue: 2, 2013
- [3] D.Popović, Z.Gorečan, J.Dujić, V.Vasić, V.Perić: *Modelovanje u elektroenergetici*, DMS Grupa, Prosveta, Novi Sad, 2011.
- [4] Duško Bekut, *Relejna zaštita*, Fakultet Tehničkih Nauka Novi Sad, 2009.
- [5] Ioanna Xyngi: *An Intelligent Algorithm for Smart Grid Protection Applications*, Delft University of Technology, Delft, Netherlands.
- [6] Hamed B. Funmilayo, Karen L. Butler-Purry, *An Approach to Mitigate the Impact of Distributed Generation on the Overcurrent Protection Scheme for Radial Feeders*, Power Systems Conference and Exposition, Mar. 2009. PSCE '09. IEEE/PES, pp. 1 – 11

Kratka biografija:

Helena Petričević rođena je na Sokocu 1993. godine. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranila je 2016. godine.

Vladan Krsman rođen je u Sarajevu 1985. godine. Doktorsku disertaciju odbranio je 2017. godine na Fakultetu tehničkih nauka iz oblasti Elektroenergetski sistemi.

UVAŽAVANJE PETERSENOVE PRIGUŠNICE U RESTAURACIJI NAPAJANJA VELIKOG PODRUČJA**CONSIDERATION OF PETERSEN COIL IN LARGE AREA RESTORATION**Nikola Nešković, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu predstavljen je problem restauracije napajanja velikog područja. Prikazana su opasna pogonska stanja mreže u slučaju kratkih spojeva. Dat je algoritam za skoro optimalnu restauraciju napajanja velikog područja koji uvažava ograničenje Petersenove prigušnice. Sprovedena je verifikacija ovog algoritma sa uvaženim ograničenjem Petersenove prigušnice na jednostavnoj test mreži sa različitim dužinama srednjenaponskih (SN) kablovskih izvoda.

Ključne reči: Restauracija napajanja, Kratki spojevi, Petersenova prigušnica

Abstract – This paper presents Large Area Restoration of Distribution Systems. A dangerous network condition is presented in case of short circuits. An algorithm for near optimal Large Area Restoration with respect of Petersen coil constraint is given. Verification of this algorithm was carried out with acknowledged Petersen coil constraint on the test network with different lengths of MV cable lines.

Keywords: Supply Restoration, Short Circuits, Petersen coil

1. UVOD

Nagli razvoj računarskih tehnologija je u velikoj meri uticao na razvoj elektroenergetskih sistema. Uz pomoć složenih softverskih alata, koji predstavljaju ključni deo „Smart Grid“ rešenja, distributivne mreže (DM) su znatno stabilnije, sigurnije i pouzdanije. Restauracija napajanja velikog područja predstavlja jednu od osnovnih funkcionalnosti takvih složenih softverskih alata, kako za planiranje tako i za optimalno upravljanje DM [1].

U DM većina kvarova su posledica kratkih spojeva [2]. Zbog toga su tehnologije procesiranja kvarova u DM od velikog značaja za poboljšanje pouzdanosti. Jedna od tehnologija procesiranja kvarova je i restauracija napajanja velikog područja [3].

Restauracija napajanja velikog područja je funkcija koja se koristi za određivanje optimalnog plana kreiranja prekidačkih manipulacija koje dovode do restauracije napajanja delu mreže koji je ostao bez napajanja, bilo da je to usled neplaniranih ispada, bilo da se radi o planiranom održavanju mreže [2].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor bio dr Goran Švenda, red. prof.

2. JEDNOPOLNI KRATAK SPOJ SA ZEMLJOM

U ovoj glavi dati su neki osnovni statistički podaci vezani za jednopolni kratak spoj u mreži. Nabrojane su prilike koje se javljaju u mreži tokom tih kratkih spojeva.

2.1. Prilike u mreži za vreme zemljospoja

Kvarovi u DM su pojave koje se događaju i teško se mogu svesti na relativno zanemarljiv broj. Oni su uzrokovani električnim i mehaničkim uzrocima. Na mehaničke uzroke kvarova je vrlo teško i skupo uticati, dok se na električne uzroke mora uticati zbog potencijalno velikih šteta i opasnosti za ljude i materijalna dobra.

Statistički podaci pokazuju da se u mešovitim DM najčešće pojavljuju jednopolni dozemni kratki spojevi [3]. Znatno uticaj na iznose napona i struja u poremećenom stanju ima tretman neutralne tačke, odnosno uzemljenje neutralne tačke elektroenergetske mreže. U zavisnosti od tretmana zvezdišta, dozemni kvar predstavlja zemljospoj ili jednopolni kratak spoj. Jednopolni kratak spoj sa zemljom ima značenje kratkog spoja samo onda ako je zvezdište DM direktno uzemljeno. U tom slučaju iznos veličine struje kratkog spoja reda je veličine iznosa struje trolnog kratkog spoja. Jednopolni kratak spoj u mreži sa izolovanim zvezdištem naziva se zemljospoj. Struje zemljospoja su kapacitivne i teku kroz dozemne kapacitete.

Nezavisno od načina tretmana neutralne tačke SN DM, jednopolni dozemni spoj u mreži karakterišu sledeće pojave [3]:

- [1] Pad napona u fazi koja je zahvaćena zemljospojem i pojava nesimetrije napona u sistemu.
- [2] Pojava napona u zvezdištu, odnosno u neutralnoj tački mreže, čiji iznos zavisi od vrednosti prelaznog otpora na mestu kvara.
- [3] Novi tok kapacitivnih struja kvara u mreži koje teku od zdravih vodova prema mestu kvara, nezavisno od načina uzemljenja neutralne tačke.
- [4] Kapacitivne struje na zdravim vodovima imaju suprotan smer u odnosu na kapacitivnu struju voda koji je u zemljospoju.
- [5] Pojava nulte komponente struje kao posledica nesimetrije struja, odnosno kvara u mreži, bez obzira da li se radi samo o kapacitivnoj struji u izolovanim mrežama ili i aktivnoj komponenti struje kroz otpornik u slučaju mreže uzemljene preko niskoomskog otpornika.

3. NAČINI UZEMLJENJA ZVEZDIŠTA TRANSFORMATORA DISTRIBUTIVNIH MREŽA

Na kvalitet snabdevanja električnom energijom značajno utiče formirana struktura mreže, odnosno dužina i tip mreže (kablovska, nadzemna ili mešovita mreža), gustina i priroda opterećenja kao i način uzemljenja zvezdišta [3].

Način uzemljenja neutralne tačke utiče na pogonske uslove mreže, stalnost napajanja, sigurnost ljudi, vrstu i cenu ugrađene opreme, zaštite i automatike. Ne postoji jedinstveni koncept uzemljenja zvezdišta SN mreža, jer postoje mnogi faktori koji utiču na izbor koji se ne mogu istovremeno zadovoljiti. Prilikom izbora vrste uzemljenja zvezdišta, osim tehničkog, važan je i ekonomski pristup, s obzirom na to da se pojedina rešenja mogu značajno razlikovati.

Razlozi za uzemljenje zvezdišta [3]:

1. Prelazak na viši naponski nivo (niže dopuštene vrednosti kapacitivne struje kvara).
2. Porast struje jednopolnog kratkog spoja (opasni naponi dodira i koraka).
3. Kabliranje (veći iznos kapacitivne vrednosti struje kvara te se luk ne može sam ugasiti).
4. Širenje mreže (posledice kao i kod kabliranja).
5. Zahtev za većim kvalitetom snabdevanja električnom energijom.

Koncepti uzemljenja neutralne tačke biraju se prema konkretnim specifičnostima određene DM, kako bi se zadovoljili svi neophodni tehnički kriterijumi za normalan i bezbedan rad.

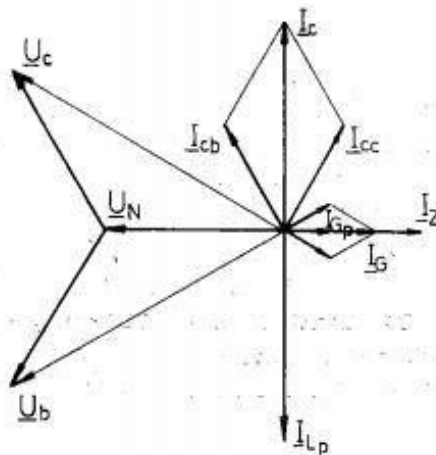
Sledeći koncepti koriste se za uzemljenje zvezdišta SN mreža u Evropi [3]:

1. Izolovano zvezdište.
2. Zvezdište uzemljeno preko male impedanse.
3. Rezonantno uzemljenje (uzemljenje preko Petersenove prigušnice).
4. Delimična kompenzacija (zvezdište uzemljeno pomoću niskoomskog otpornika i prigušnice).
5. Direktno uzemljeno zvezdište.
6. Transformatori za uzemljenje.

Mreža sa rezonantnim uzemljenjem zvezdišta energetskih transformatora spojena je sa zemljom preko induktivne reaktanse, u praksi poznate kao Petersenova prigušnica. Osnovni razlog za ugradnju automatskih kompenzacionih prigušnica za uzemljenje je povećanje kvaliteta snabdevanja električnom energijom sa stanovišta pouzdanosti napajanja. Pomenuta prigušnica smanjuje kapacitivnu komponentu struje zemljospoja jer kroz mesto kvara proizvodi induktivnu komponentu struje koja se superponira na kapacitivnu komponentu struje kvara i poništava je delimično ili u potpunosti, u zavisnosti od podešenja prigušnice.

Princip rada kompenzacije prigušnice je jednostavan. U slučaju zemljospoja jedne faze, ukupna kapacitivnost mreže prema zemlji i induktivnost prigušnice stvaraju jedan rezonantni krug čija je impedansa teoretski beskonačna. U ovakvoj situaciji reaktivna komponenta struje na mestu kvara, nezavisno od prelaznog otpora između faze i zemlje, se poništava, odnosno induktivna struja kompenzuje kapacitivnu struju zemljospoja.

Da bi se dobila jasnija slika o prilikama pri jednopolnom kratkom spoju u mrežama u kojima je ugrađena prigušnica za kompenzaciju kapacitivne komponente struje zemljospoja, na slici 1 prikazan je vektorski dijagram napona i struja. Ukupna kapacitivna struja mreže dobija se kao vektorska suma svih kapacitivnih struja zdravih faza. Ova struja prednjači naponu u zvezdištu za 90° . Za razliku od nje, induktivna komponenta struje kroz prigušnicu I_L kasni za pomenutim naponom za 90° , dok mu aktivna komponenta struje kroz prigušnicu I_Z stoji u opoziciji.



Slika 1. Vektorski dijagram napona i struja u rezonantno uzemljenoj mreži za vreme zemljospoja [3]

4. RESTAURACIJA NAPAJANJA VELIKOG PODRUČJA

Restauracija napajanja velikog područja je postupak koji prema određenom kriterijumu omogućava određivanje optimalnog plana prekidačkih akcija za obnavljanje napajanja delu mreže koji je ostao bez napajanja nakon izolacije kvara. Velika dimenzionalnost SN DM čini problem restauracije napajanja znatno složenijim. Zbog toga se zadnjih decenija razvilo dosta funkcija koje imaju cilj da za što kraći vremenski period, uz minimalne troškove, što većem broju potrošača koji su ostali bez napajanja obnove napajanje.

Rezultat funkcija za restauraciju napajanja velikog područja je predlog svih varijanti alternativnog napajanja koje su rangirane u skladu sa korisnički specificiranim kriterijumom, koji zavise od osobina razmatrane DM kao i tipa problema koji se rešava restauracijom napajanja [4].

4.1 Problem optimizacije

Restauracija napajanja velikog područja pripada klasi mešovito-celobrojnih nelinearnih optimizacionih problema sa ograničenjima. Pored toga, u praktičnoj primeni ovaj problem je izuzetno velike dimenzionalnosti, što ga dodatno usložnjava. Naime, konfiguracija DM određena je statusom prekidača (otvoren/zatvoren) koji se modeluje binarnim (0/1) promenljivim. Sa druge strane, režim DM (tokovi snaga, gubici aktivne snage, itd.) modeluju se nelinearnim algebarskim jednačinama u kojima figurišu kontinualne promenljive (moduli i fazni stavovi fazora napona u čvorovima DM). Optimizacione metode koje se koriste za rešavanje navedenog problema mogu se podeliti na [2]:

1. Heurističke metode.
2. Meta-heurističke metode.
3. Matematički zasnovane metode.

Cilj heurističkih metoda jeste da znajući fiziku određenog problema u što kraćem roku dođe do kvalitetnih praktičnih, ali suboptimalnih rešenja. Ove metode su razvijene na bazi maksimalnog poznavanja karakteristika DM. Meta-heurističke metode su zasnovane na stohastičkom pretraživanju, analogiji sa prirodnim procesima kao i na modelima veštačke inteligencije. Matematički zasnovane metode, koje u potpunosti odgovaraju prirodi navedenog problema, pokazale su se nepraktičnim za primenu u realnim DM. Naime, usled velike dimenzionalnosti realnih DM, odgovarajući optimizacioni modeli za rešavanje problema restauracije napajanja velikog područja su veoma velikih dimenzija i za njihovo rešavanje su potrebni značajni računarski resursi.

4.2 Višekriterijumski algoritam za restauraciju

U [4] predložena je metoda za (skoro) optimalnu restauraciju napajanja velikog područja nakon velikih poremećaja. Metoda je osmišljena tako da obezbedi minimalan trošak usled prekida napajanja uzimajući u obzir strujna, naponska, sistemska i druga ograničenja. Efikasna restauracija opterećenja postiže se korišćenjem SN sabirnice u VN/SN transformatorskim stanicama koje olakšavaju prenos opterećenja između transformatora i SN izvoda, uz istovremeno smanjenje broja uklopnih operacija. Optimalne varijante za snabdevanje potrošačkih čvorova koji su ostali bez napajanja generišu se na osnovu najkraćih puteva između delova mreže koji su ostali bez napajanja i dela mreže koji i dalje ima napajanje, što značajno skraćuje vreme potrebno za restauraciju. Globalni tok algoritma:

1. Određivanje dela mreže od interesa.
2. Selekcija rezervnih varijanti napajanja.
3. Rangiranje varijanti.

4.2.1 Određivanje dela mreže od interesa

Potrebno je odrediti elemente koji su zahvaćeni kvarom ili na kojima se vrši plansko održavanje, i te elemente upisati u listu elemenata kojima se ne sme vratiti napajanje. Nakon ovog koraka potrebno je u drugu listu elemenata upisati sve elemente koji ostaju bez napajanja, a nisu zahvaćeni kvarom, ili se na njima ne vrši plansko održavanje.

4.2.2 Selekcija rezervnih varijanti napajanja

Restauracija napajanja velikog područja koristi se za velike strukturalne poremećaje, kada je zahvaćen veliki broj potrošača (npr. kvar na VNvodu i/ili TS VN/SN). Kako je TS VN/SN prelaz između prenosnog i distributivnog dela mreže, područje operativnog delovanja ove funkcije je i prenos i distribucija. Topološka varijanta je varijanta koja posmatra samo električnu povezanost elemenata u mreži (kako prenosnoj, tako distributivnoj). Od svih elemenata iz listi koje su dobijene u prethodnom koraku vrši se topološka pretraga u svim pravcima do normalno otvorenih (NO) rasklopnih uređaja. Ako je jedan kraj NO uređaja pod napajanjem, a drugi ne, njegovim zatvaranjem dobija se rezervna varijanta napajanja. Ovako dobijena rezervna varijanta restauracije napajanja naziva se jednostavna varijanta, zato što se osim topoloških ne gledaju nikakva druga ograničenja. Simulacijom primene

svake jednostavne varijante dobija se režim i proveravaju se ograničenja (strujna, naponska, ograničenja relejne zaštite, ograničenje Petersenove prigušnice, itd.). Pored ovih jednostavnih postoje i složene varijante za restauraciju u kojima se odseca jedan deo potrošnje kako bi se zadovoljila ograničenja. Cepanje opterećenja podrazumeva restauraciju sa dva različita izvora.

4.2.3 Rangiranje varijanti

Pri rangiranju varijanti rezervnog napajanja razmatraju se sve jednostavne varijante i unapred korisnički definisan broj složenih varijanti. Rangiranje varijanti vrši se na osnovu jednog od korisnički specificiranih kriterijuma ili nekoj od njihovih kombinacija. Ti kriterijumi su [2]:

1. Iznos snaga nerestauriranog opterećenja.
2. Troškovi manipulacija.
3. Kritična rezerva snage napojnog transformatora.
4. Kritična strujna rezerva izvoda.
5. Kritičan pad napona.

Gore navedeni kriterijumi se pojavljuju kao sabirnici u integralnoj funkciji uz dodatak težinskih faktora kojima se definiše koji od kriterijuma je važniji za korisnika.

5. MODEL PETERSENOVE PRIGUŠNICE U RESTAURACIJI NAPAJANJA VELIKOG PODRUČJA

Vrednost reaktanse Petersenove prigušnice koja je potrebna da bi se kompenzovala ukupna kapacitivnost napojne mreže može se dobiti iz struje na mestu kvara. Struja nultog redosleda na mestu kvara je:

$$I_0 = I_{C0} + I_{L0} = j \left(\omega C_0 - \frac{1}{3\omega L} \right) V_0. \quad (1)$$

Kako je ideja korišćenja Petersenove prigušnice da se kompenzuje kapacitivna komponenta struje kvara, idealna kompenzacija bila bi opisana relacijom $I_{C0} + I_{L0} = 0$. Sa druge strane u slučaju da nemamo idealnu kompenzaciju, ukupna kompenzacija se može se opisati sa koeficijentom kompenzacije α tako da bi relacija 1 glasila:

$$\omega C_0 - \frac{1}{3\omega L} = (1 - \alpha) \omega C_0. \quad (2)$$

Odnosno minimalna i maksimalna vrednost koeficijenta kompenzacije može se iskazati u terminima reaktanse Petersenove prigušnice i ukupne otočne reaktanse mreže.

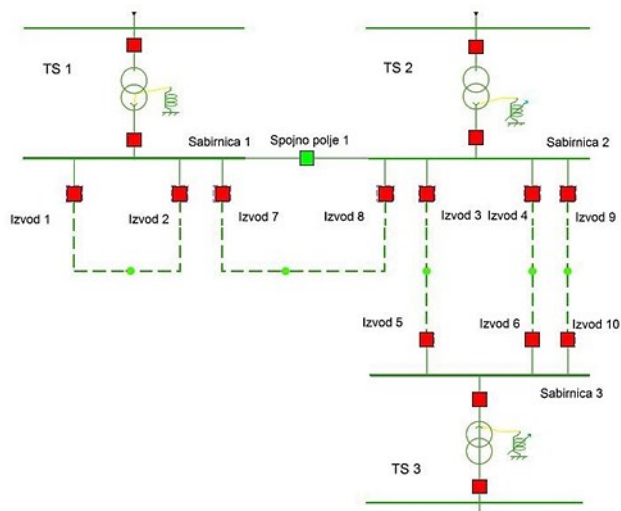
$$\alpha_{min} X_L \leq \frac{X_{C0}}{3} \leq \alpha_{max} X_L. \quad (3)$$

Menjanje topološke strukture mreže prilikom restauracije mora da zadovolji jednačinu 3 kako se ne bi desilo da prilikom primenjene varijante za restauraciju ukupna reaktansa mreže X_{C0} prekorači gornje ograničenje.

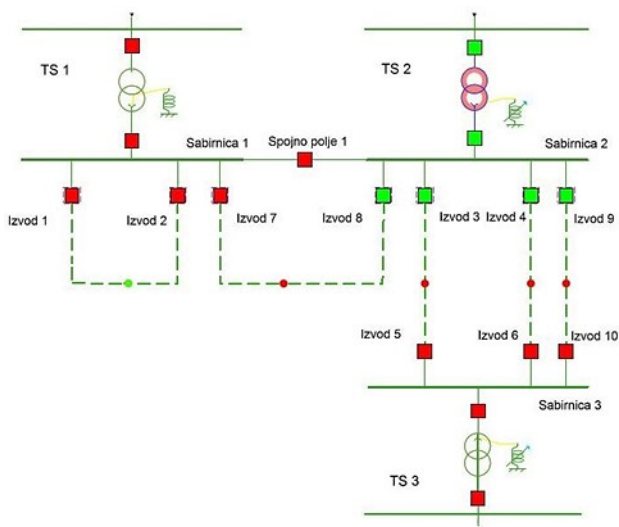
6. NUMERIČKA VERIFIKACIJA MODELA

U okviru ove glave prikazan je primer restauracije napajanja velikog područja gde se kvar desio u TS 2, sve dužine SN izvoda jednake su i iznose 20 km. Svi transformatori na sekundarnoj strani imaju priključenu Petersenovu prigušnicu. Na slici 2 prikazana je osnovna topološka struktura mreže pre kvara. Crvenom bojom prikazani su zatvoreni, dok su zelenom bojom prikazani otvoreni rasklopní uređaji.

Nakon što relejna zaštita detektuje kvar na TS 2 i otvori prekidač na VN strani transformatora, potrebno je otvoriti prekidač na SN strani kako bi se kvar izolovao. Tada izvod 8, 3, 4 i 9 ostaju bez napajanja i potrebno im je vratiti napajanje sa susjednih transformatorskih stanica. Nakon simulacije zatvaranja rastavljača snaga koji se nalaze između izvoda proverava se ukupno X_{C0} i ako tako dobijeno X_{C0} zadovoljava jednačinu 3 ta varijanta se prihvata, to se radi sve dok svi izvodi ne budu restaurisani. Konačna topološka struktura mreže nakon primenjenih manipulacija prikazana je na slici 3.



Slika 2. Inicijalna topološka struktura mreže



Slika 3. Konačna topološka struktura mreže

Na slici 3 se jasno vidi da su svi SN izvodi restaurisani preko izvoda iz susjednih stanica, kao i da je SN sabirnica restaurisana zatvaranjem prekidača u spojnom polju. Rezultati proračuna kao i vrednost X_{C0} prikazani su i tabelarno u tabeli 1.

Prva kolona tabele predstavlja redni broj izvoda koji se restauriše. Ti izvodi su sortirani prema prioritetu izvoda. Prioritet izvoda definisan je korisničkim kriterijumima (broj potrošača, broj daljinski kontrolisanih prekidačkih uređaja, itd.). U drugoj koloni označen je izvod koji je ostao bez napajanja i kome se treba vratiti napajanje. Treća kolona predstavlja izvod preko koga se restauriše izvod iz prethodne kolone. U četvrtoj koloni nalazi se

ukupna nulta vrednost reaktanse napojne mreže koja se izračunava nakon što se varijanta primeni. Četvrta kolona je indikator da li je varijanta zadovoljila kriterijum opisan relacijom 3 ili ne, odnosno da li će biti primenjena.

Tabela 1. Primer restauracije za dužinu izvoda 20 km

Redni br.	Za restauraciju	Restauriše se preko	$X_{C0}[\Omega]$	Varijanta prihvaćena
1	Izvod 9	Izvod 10	230.52	Da
2	Izvod 8	Izvod 7	208.01	Da
3	Izvod 4	Izvod 6	175.30	Da
4	Izvod 3	Izvod 5	149.38	Da
5	Sabirnica 2	Sabirnica 1	208.01	Da

U tabeli 2 prikazan je rezultat proračuna za dužinu SN izvoda 25 km. Tu je potrebno primetiti da izvodi 8 i 3 kao i sabirnica 2 ostaju bez napajanja jer ukoliko bi se zatvorili NO rasklopni uređaji doveli bi do narušavanja ograničenja Petersenove prigušnice.

Tabela 2. Primer restauracije za dužinu izvoda 25 km

Redni br.	Za restauraciju	Restauriše se preko	$X_{C0}[\Omega]$	Varijanta prihvaćena
1	Izvod 9	Izvod 10	138.31	Da
2	Izvod 4	Izvod 6	124.80	Da
3	Izvod 8	Izvod 7	105.18	Ne
4	Izvod 3	Izvod 5	89.63	Ne
5	Sabirnica 2	Sabirnica 1	103.49	Ne

7. ZAKLJUČAK

U okviru ovog rada prikazana je restauracija napajanja velikog područja nakon velikih sistemskih poremećaja. Kako ova funkcija menja topološku strukturu mreže, potrebno je uvažiti ograničenje Petersenove prigušnice prilikom kreiranja rezervnih varijanti za restauraciju, kako se primenom tih varijanti ne bi izgubilo pozitivno dejstvo koje bi prigušnica imala kada bi se desili novi kratki spojevi u mreži.

8. LITERATURA

- [1] N.Katić: *Menadžment sistemi u elektroenergetici - DMS*; skripta sa predavanja, Fakultet tehničkih nauka, Novi Sad, novembar 2013.
- [2] D.Popović, D.Bekut, V.Treskanica: *Specijalizovani DMS algoritmi*; Prosveta, Novi Sad, 2011.
- [3] J.Nahman: *Uzemljenje neutralne tačke distributivnih transformatora*; Naučna knjiga, Beograd, 1980.
- [4] J.Nahman, G.Štrbac: *A new algorithm for service restoration in large-scale urban distribution systems*; EPSR, Vol. 29, 1994, str. 181-192.

Kratka biografija:



Nikola Nešković rođen je u Kragujevcu, Srbija, 1994. god. Osnovne studije završio je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi 2017. god. i iste godine upisao master studije.

UTICAJ PRIKLJUČENJA FOTONAPONSKE ELEKTRANE NA DISTRIBUTIVNU MREŽU**THE IMPACT OF PHOTOVOLTAIC POWER PLANT CONNECTION ON DISTRIBUTED NETWORK**Igor Bjelica, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U radu se razmatraju dva osnovna problema koja nastaju priključenjem fotonaponske elektrane na distributivnu mrežu (DM): problem varijabilnosti snage i problem "Duck curve". Prvo je prikazan teorijski uvod o oba problema, a kasnije se predlažu strategije za njihovo suzbijanje. Uticaj priključenja fotonaponske elektrane na DM razmatra se na primeru jednostavne test mreže. U tu svrhu, razvijen je matematički model u programskom jeziku C#. Dobijeni rezultati predstavljeni su kroz dijagrame na osnovu kojih se vrši analiza.

Ključne reči: Distributivna mreža, Fotonaponska elektrana

Abstract – In this paper are considered two fundamental problems which occur in connection of photovoltaic power plant on distributed network. It is analyzed the problem of power variability and the problem of "Duck curve". First, it is shown the theoretical introduction about both problems and then proposes strategies for their suppression. In the second part of the work it was analyzed the impact of power plant connection on distributed network. This research is being conducted on test network. The mathematical model for analyzing was developed in the C # programming language. The obtained results are presented through diagrams on the basis of which the analysis is performed.

Keywords: Distribution network, Photovoltaic power plant

1 UVOD

Brz razvoj solarne industrije, povećanje efikasnosti solarnih ćelija, pad cene fotonaponskih (FN) panela i invertora, te razni načini podsticaja rezultovali su brzim i kontinuiranim povećanjem instalirane snage fotonaponskih elektrana (FNE) globalno [1].

Kako bi se smanjili gubici u prenosu električne energije, a time i troškovi, električna energija se počinje proizvoditi na mestu ili blizu mesta potrošnje. To se zove distributivna proizvodnja električne energije [2].

Jedan od glavnih problema obnovljivih izvora energije jeste nestabilnost i zavisnost od vremenskih prilika. Brze promene vremenskih uslova u toku dana mogu prouzro-

kovati varijabilnost proizvedene snage. Još jedan problem nastao velikom ekspanzijom solarnih elektrana jeste pokrivanje dnevnog pika u poslepodnevnim časovima za vreme zalaska sunca kada proizvodnja FN elektrana polako iščezava a potrošnja energije doseže svoj maksimum. Tada se pojavljuje velika razlika između potrošnje i proizvodnje koju je potrebno nadomestiti iz konvencionalnih izvora energije [3].

U drugom poglavlju pažnja se posvećuje osnovnim problemima prouzrokovanim povezivanjem FN elektrane na distributivnu mrežu (DM). Obrađuju se dva problema: "Duck curve" [4] i varijabilnost snage [5].

Treće poglavlje se bavi strategijama u cilju suzbijanja prethodno navedenih problema koji nastaju priključivanjem FNE na DM.

U četvrtom poglavlju prikazani su rezultati proračuna dobijeni na primeru jednostavne test mreže. Na osnovu dobijenih rezultata vrši se analiza.

U petom poglavlju dat je zaključak o prednostima i manama priključenja FNE na DM na osnovu teorijskih razmatranja i prikazanih rezultata. Nakon zaključka referentno je navedena literatura korišćena za realizaciju ovog rada.

2 POSTAVKA PROBLEMA

U ovom delu je prikazan problem integracije FNE na DM. Pritom, razmatraju se dva osnovna problema FNE: varijabilnost proizvodnje i "Duck Curve".

2.1 Problem varijabilnosti proizvodnje FNE

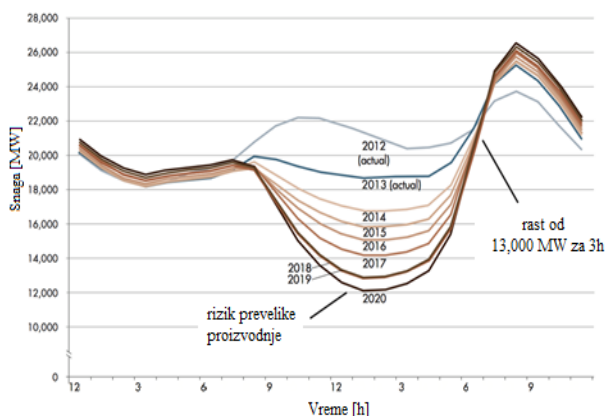
Varijacije u proizvodnji FNE zavise od promena intenziteta sunčevog zračenja [5], koje se menja pre svega usled kretanja oblaka (temperatura, brzina vetra i vlažnost, ne uzrokuju značajne varijacije u proizvodnji). Tako izazvane varijacije injektirane snage u mrežu imaju negativan uticaj na kvalitet i pouzdanost sistema. Skladišta električne energije su jedno od pouzdanih rešenja za smanjenje varijacija snage ispod dozvoljene vrednosti [5].

2.2 Duck curve

U ovom delu, razmatra se dnevni dijagram opterećenja specifičnog oblika, poznat kao "Duck Curve" [4]. Oblik dijagrama ukazuje na potencijal za prekomernu proizvodnju, koja se javlja usled sve veće penetracije FNE.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor bio dr Goran Švenda, red. prof.



Slika 1. Duck Curve [4]

Na slici 1, za period 2012-2020, prikazani su dijagrami ukupnog opterećenja sistema – ukupna potrošnja umanjena za proizvodnju (umanjeno za energiju koja se proizvodi uz pomoću vetra i solarnih elektranama). Može se uočiti značajno smanjenje vrednosti dijagrama ukupnog opterećenja sistema za svaku narednu godinu. Konačno, za 2020. godinu izvršena je procena rizika od prekomerne proizvodnje. Glavni problem nastaje u poslepodnevnim časovima za vreme zalaska sunca kada je proizvodnja FNE polako iščezava a potrošnja energije doseže svoj maksimum. Tu razliku koja nastaje između zahtevane i proizvedene snage potrebno je nadomestiti iz konvencionalnih izvora energije. Oni zahtevaju više vremena za stupanje u pogon pre nego što budu iskorišćeni, stoga moraju da proizvode minimum izlazne snage u periodima kada to nije potrebno. To predstavlja glavni problem za operatore mreže sa puno FNE [4].

3 STRATEGIJE ZA REŠAVANJE PROBLEMA

U ovom delu prikazane su strategije i algoritmi čijom primenom se mogu izravnati dijagrami opterećenja i varijacija snage. U [6] su date strategije za promenu izgleda Duck Curve i izravnjanja dijagrama opterećenja.

3.1 Strategije za rešavanje Duck curve

U ovom delu razmatra se oblik Duck Curve i strategije za smanjenje i preraspodelu opterećenja [6]. Razmatrane su sledeće strategije za pokrivanje velikih potreba za energijom:

1. Strategija – Energetska efikasnost u časovima naglog porasta opterećenja
2. Strategija – Korišćenje i raspodela obnovljivih izvora u vreme maksimalnog opterećenja
3. Strategija – Upravljanje distribucijom vode i ispuštanja otpadnih voda
4. Strategija – Kontrolisanje električnih grejača vode
5. Strategija – Upotreba skladišta leda umesto klima uređaja
6. Strategija – Promena opterećenja uvođenjem tarifnog sistema
7. Strategija – Raspodela skladišta električne energije na određene lokacije
8. Strategija – Implementirati programe za regulisanje potrošnje energije
9. Strategija – Korišćenje međuregionalne razmene

energije

10. Strategija – Povlačenje iz upotrebe nefleksibilnih i neisplativih elektrana

Navedene strategije ne podrazumevaju samo integraciju obnovljivih izvora, već i povećanje pouzdanosti sistema, smanjenje cene proizvodnje i prenosa energije, kao i cene goriva [6].

3.2 Strategije za rešavanje varijabilnosti snage

Varijabilnost sunčevog zračenja kojima su uzloženi FN paneli prouzrokovana je kretanjem oblaka. Na osnovu promenljivosti sunčevog zračenja javljaju se brze promene u proizvodnji FNE. Povećavanje udela FNE u mreži može prouzrokovati negativne uticaje na kvalitet i pouzdanost. Iz tog razloga skladišta energije su neophodna da smanje oscilacije snage FNE ispod dozvoljene vrednosti. Instalacija skladišta energije ima veliki uticaj na energetske-ekonomsku isplativost FN sistema za koje se smatra da će imati ključnu ulogu u održivosti FN sistema. U ovom delu predlažu se tri kontrolne strategije za rešavanje problema varijabilnosti snage [7]:

1. Ramp rate
2. Moving average
3. Step rate

Jednogodišnja simulacija navedenih strategija pokazala je mane i prednosti svake od kontrolnih strategija. Moving average strategija zahteva najmanji kapacitet skladišta energije, međutim povećanjem gubitaka od 2-3 puta gube se ti benefiti. Ramp rate strategija ima manje gubitke ali zahteva najveći kapacitet skladišta energije. Detaljna analiza isplativosti za svaki slučaj pojedinačno predstavlja najbolje rešenje. Analiza treba obuhvatiti uticaj broja ciklusa punjenja/praznjenja i stanja napunjenosti baterije na izabrano skladište. Strategija Step rate predstavlja najbolje rešenje za većinu razmatranih slučajeva, jer omogućuje najefikasnije korišćenje sistema za skladištenje energije, sa manjim brojem ciklusa punjenja/praznjenja i smanjenim gubicima [7].

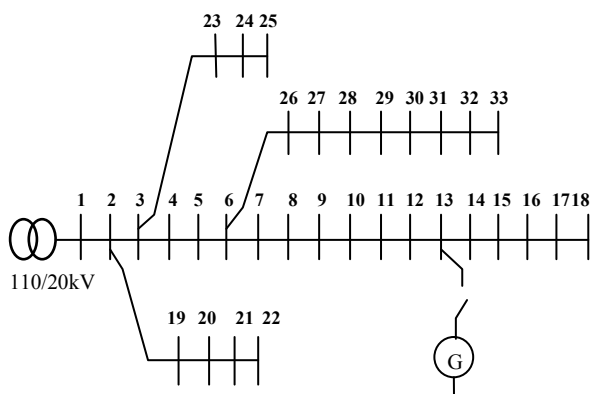
4 PRIMER PRORAČUNA

Na primeru test mreže prikazane su prednosti i mane priključenja FNE na DM. Prvo se razmatra uticaj priključenja FNE na DM, a zatim i uticaj energetskih skladišta. Pritom deo energije proizvedene iz FNE skladišti se u energetskim skladištima i upotrebljava u večernjim časovima kada je potrošnja najveća. Rezultati su dobijeni na osnovu matematičkog modela razvijenog u C# programskom jeziku.

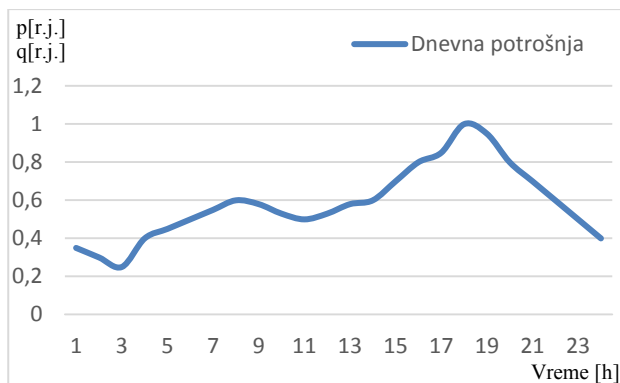
4.1.1 Jednostavna test mreža

Analize su izvršene na IEEE test mreži [8], slika 2, za koju su poznati parametri i vrednosti potrošnje potrošača, slika 3 i proizvodnje FNE slika 4. Proračunom tokova snaga dobijene su vrednosti gubitaka aktivne snage i snage preuzete od prenosne mreže. Posebno su razmatrana dva trenutka:

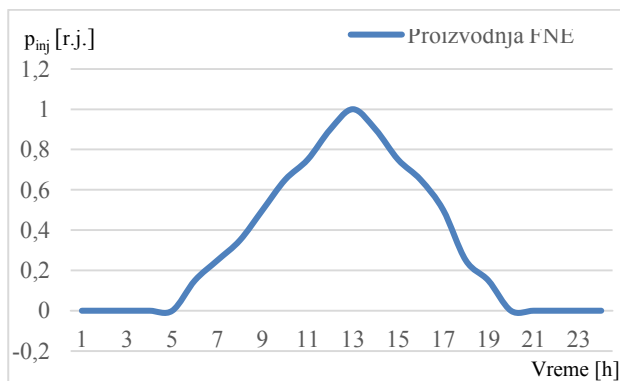
- 1) 18h – maksimalno opterećenje mreže i
- 2) 13h – maksimalna proizvodnja FNE.



Slika 2. Test mreža [8]



Slika 3. Koeficijenti potrošnje DM



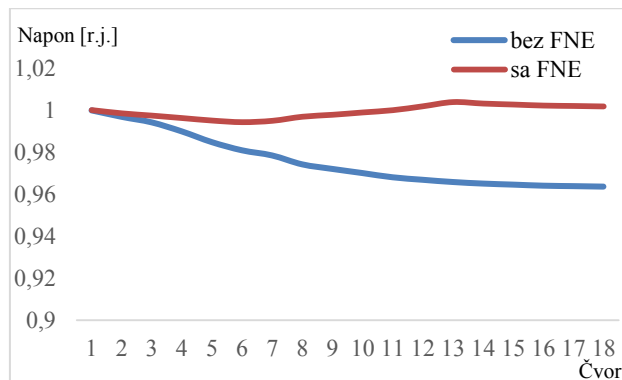
Slika 4. Koeficijenti proizvodnje FNE

4.2 Prikaz rezultata u 13h

Maksimalna proizvodnja FNE se desila u 13h. Priključenjem FNE gubici test mreže su smanjeni sa 0.131 MW na 0.058 MW. Snaga potrebna iz prenosne mreže smanjuje se sa 3.714 MW na 2.372 MW, tabela 1. Značajno smanjenje gubitaka u DM povezano je sa proizvodnjom električne energije u neposrednoj blizini njene potrošnje. Priključenjem FNE značajno su popravljene i naponske prilike u celoj mreži, slika 5.

Tabela 1. Rezultati proračuna u 13h

	bez FNE	sa FNE
Gubici u mreži [MW]	0.131	0.058
Snaga potrebna iz prenosne mreže [MW]	3.714	2.372



Slika 5. Profil naponska sa i bez FNE, u 13h

4.3 Prikaz rezultata u 18h

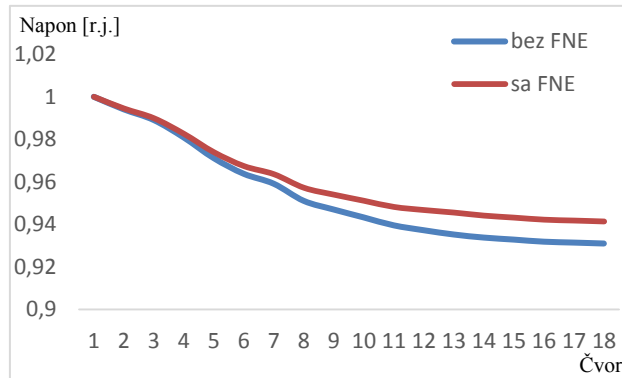
Maksimalna potrošnja DM se desila u 18h. U ovom režimu, priključenjem FNE gubici test mreže su smanjeni sa 0.410 MW na 0.334 MW.

Snaga potrebna iz prenosne mreže smanjuje se sa 6.508 MW na 6.140 MW, tabela 2.

Iako je proizvodnja FNE u 18h znatno manja od svoje maksimalne, postiže se značajno smanjenje gubitaka u mreži. Priključenjem FNE popravljene su i naponske prilike u celoj mreži, slika 6.

Tabela 2. Rezultati proračuna u 18h

	bez FNE	sa FNE
Gubici u mreži [MW]	0,410	0,334
Snaga potrebna iz prenosne mreže [MW]	6,508	6,140



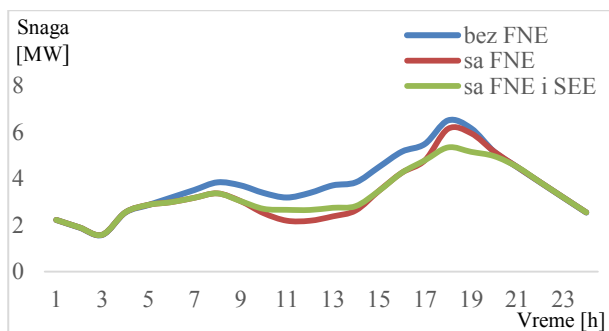
Slika 6. Profil napona DM sa i bez FNE u 18h

4.4 Prikaz dnevnih dijagrama

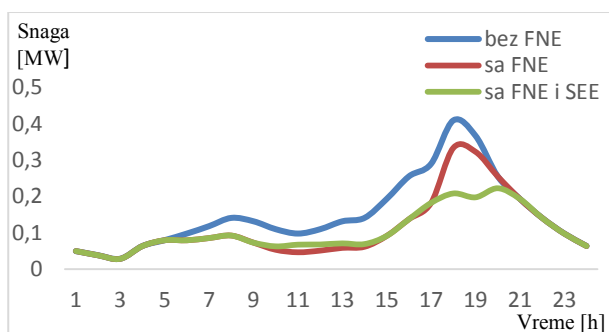
Osim primera DM sa i bez FNE, razmatran je i primer sa skladištem električne energije (SEE) koje je postavljeno neposredno uz FNE.

Na slikama 7 i 8 prikazani su dnevni dijagrami potrošnje i gubitaka aktivne snage, respektivno (na dijagramima su prikazane vrednosti režima test mreže bez FNE, sa FNE i sa FNE i SEE).

U test mreži sa FNE i SEE deo proizvodnje FNE se injektira u mrežu, a deo se skladišti u SEE i iz njega u poslepodnevnim časovima injektira u mrežu.



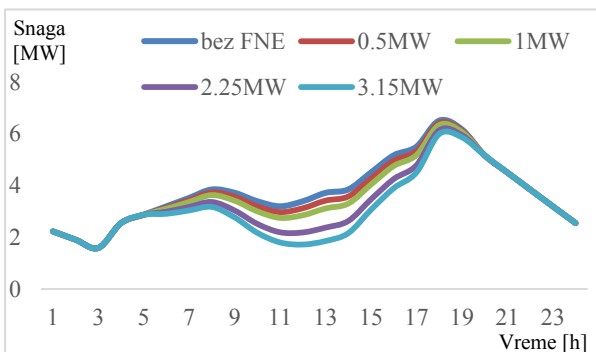
Slika 7. Dnevni dijagram potrošnje



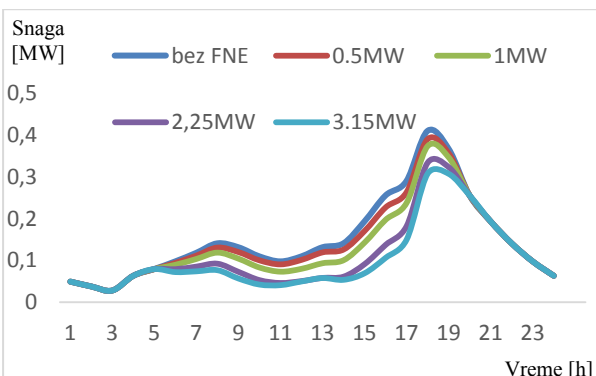
Slika 8. Dnevni dijagram gubitaka aktivne snage

4.5 Uticaj veličine FNE

Povećavanjem kapaciteta FNE u test mreži raste problem "Duck curve". Na slikama 9 i 10 prikazani su dnevni dijagrami potrošnje i gubitaka aktivne snage u zavisnosti od veličine FNE, respektivno (na dijagramima su prikazane vrednosti režima test mreže bez FNE i sa FNE vršne snage 0.5MW, 1MW, 2.25MW i 3.15MW).



Slika 9. Dijagram potrošnje u zavisnosti od snage FNE



Slika 10. Dijagram gubitaka u zavisnosti od snage FNE

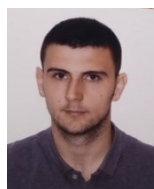
5 ZAKLJUČAK

Distributivni generatori zasnovani na suncu i vetru nisu upravljivi u smislu kontrolisane promene aktivne snage. Priključenjem FNE u DM mogu da se ostvare mnoge prednosti, kao npr. bolje naponske prilike i smanjenje gubitaka distributivne mreže. Ali istovremeno njihovim priklučenjem u DM mogu da se dese i neželjene pojave, kao što su npr. Duck curve i varijacije izlazne snage. Navedeni problemi se uspešno rešavaju primenom električnih skladišta energije. Visoka cena i realtivno mali kapaciteti sprečavaju njihovu širu primenu u današnjim DM. U budućnosti se očekuje njihov razvoj koji će uticati na povećanje kapaciteta i smanjenje cena a samim tim i na širu primenu.

6 LITERATURA

1. H.Wirth: *Recent Facts about Photovoltaics in Germany*, Fraunhofer Institute for Solar Energy Systems ISE, Freiburg, January 2017.
2. D.Pajan: *Utjecaj solarne elektrane na mrežu u točki priključenja*, Završni rad, Sveučilište Sjever, Varaždin, 2016.
3. K.Brellas: *Modeling of Hybrid Operation of a Concentrated Solar Power production unit*, Technical University of Crete, Chania, Greece, 2017.
4. <https://www.aiso.com/Documents/FlexibleResourcesHelpRenewablesFastFacts.pdf>.
5. S.Vodopija: *Prognostiranje proizvodnje iz OIE i upravljanje distribuiranim sustavima za skladištenje električne energije u elektroenergetskoj mreži u tržišnim uvjetima*, doktorski rad, Sveučilište u Splitu, Split, 2017.
6. J.Lazar: *Teaching the Duck to Fly*, Second Edition. Montpelier, VT: The Regulatory Assistance Project, 2016.
7. J.Marcos, I.Parra, M.Garcia, L.Marroyo: *Control strategies to smooth short-term power fluctuations in large photovoltaic plants using battery storage systems*, Article, Department of Electrical and Electronic Engineering, University of Navarre, Pamplona, Spain, 2014.
8. A.Ulinuha, M.A.S.Masoum, S.M.Islam: *Optimal scheduling of LTC and shunt capacitors in large distorted distribution systems using evolutionary-based algorithms*, IEEE Trans. on PWRD, vol. 23, Issue: 1, Jan. 2008.

Kratka biografija:



Igor Bjelica rođen je u Novom Sadu 1994. godine. Osnovne studije završio na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i Računarstvo – Elektroenergetski sistemi 2017. god. i 2017. god. upisao master studije na istom fakultetu.

PRORAČUN REŽIMA ELEKTROENERGETSKIH SISTEMA SA KVAROM U FAZNOM DOMENU

POWER SYSTEMS SHORT CIRCUIT CALCULATION IN PHASE DOMAIN

Mirko Đurić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu prikazan je postupak za formiranje matematičkih modela elemenata elektroenergetskih sistema u faznom domenu, njihovu integraciju u jedinstveni model i proračun režima sa kratkim spojem elektroenergetskih sistema. Proračuni su izvršeni u programskom jeziku Fortran.

Ključne reči: Modelovanje elemenata elektroenergetskih sistema, režim sa kratkim spojem, fazni domen

Abstract – This paper presents the procedure for creating mathematical models of the elements of electric power systems in the phase domain, their integration into a unique model and the calculation of the regime with a short circuit of the electric power systems. All calculations are realized in Fortran programming language.

Keywords: Modeling Elements of Power Systems, Short Circuit Regime, Phase Domain

1. UVOD

Proračun režima elektroenergetskog sistema (EES) sa kvarom je, uz proračun tokova snaga, najznačajniji problem koji se razmatra u domenu (kvazi)stacionarne elektroenergetike [1-4].

Postupci za proračun režima EES sa kvarom se mogu razlikovati u zavisnosti od krajnjeg cilja proračuna koji se provodi. Uobičajeno, cilj proračuna je podešenje relejne zaštite, mehaničko i termičko dimenzionisanja opreme, sigurnost sa aspekta opreme, itd. U tom slučaju proračun režima EES sa kvarom se sprovoditi samo za fazni deo sistema. Ukoliko je cilj proračuna dimenzionisanje sistema uzemljenja, proračun napona koraka i dodira, odnosno sigurnost sa aspekta ljudi, itd. tada se osim faznog dela mora obuhvatiti i sistem uzemljenja.

Generalno, metode za proračun režima EES sa kvarom mogu da se podele na metode bazirane na kompleksnim analitičkim izrazima i numeričke metode. Metode bazirane na analitičkim izrazima karakterišu relativno jednostavni matematički modeli. One su razvijene na osnovu većeg broja aproksimacija i samim tim imaju umanjenu tačnost rezultata i značajna ograničenje praktične primene.

Numeričke metode imaju znatno manje aproksimacija i pogodne su za obradu problema velikih dimenzija. Samim ove metode imaju znatno veće mogućnosti i njihovom primenom se dobijaju znatno tačniji rezultati. Njihova praktična primena zahteva primenu računara.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor dr Goran Švenda, red.prof.

Proračuni režima EES sa kvarom mogu se vršiti u domenu simetričnih komponenti i u faznom domenu. Proračuni u domenu simetričnih komponenti su karakteristični za uravnotežene EES, kada je od interesa proračun samo režima u faznom delu sistema. Proračuni u faznom domenu karakteristični za neuravnotežene EES, odnosno kada je od interesa režim faznog i sistema uzemljenja.

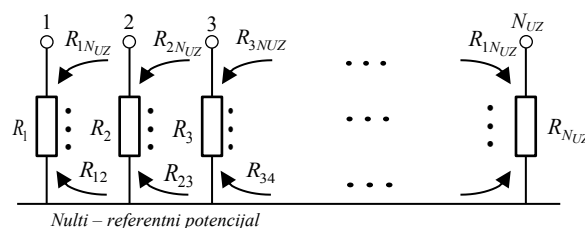
2. PARCIJALNI MATEMATIČKI MODELI ELEMENATA EES

Veličina i složenost realnih EES kao i veliki broj efekata koji se u njima ispoljavaju za posledicu imaju velike dimenzije matematičkih modela kojima se opisuju. Ovi matematički modeli se, radi jednostavnijeg generisanja i obrade, zasnivaju na metodi potencijala čvorova, odnosno na odgovarajućim matricama admitansi nezavisnih čvorova $\underline{Y}_{BUS}$. Tri osnovne osobine matrica admitansi su: trivijalno se generišu, simetrične su i retke [1,5]. Treba naglasiti da se ove tri osobine ispoljavaju u električnim kolima u kojima nema kontrolisanih izvora (uzemljeni delovi EES spregnuti efektom blizine, elektromagnetski spregnuti provodnici i idealni transformatori).

U cilju modelovanja EES sa kvarom, moguće je definisati pojedinačne matematičke modele elemenata (pod sistema) EES. Razmatrani EES moguće je dekomponovati na *pod sistem uzemljivača*, *pod sistem provodnika*, *pod sistem transformatora* i *pod sistem elektroenergetskog okruženja* između kojih postoje isključivo galvanske veze. Uticaji kontrolisanih izvora, u okviru pod sistema, ekvivalentirani su uticajem admitantnih grana. Nakon formiranja matematičkih modela pojedinačnih pod sistema, moguće je te parcijalne modele integrisati u jedinstven matematički model EES – jedinstvenu matricu admitansi celokupnih EES.

2.1. Pod sistem uzemljivača

Pod sistem uzemljivača, sa N_{UZ} uzemljivača [6], prikazan je na slici 1. Uzemljivači su predstavljeni sa svojim međusobnim uticajima - efekt blizine. Sa 1,2,..., N_{UZ} su naznačene tačke u kojima su uzemljivači galvanski spojeni sa elementima EES-a.



Slika 1. Pod sistem uzemljivača koji se sastoji od N uzemljivača

Efekat blizine predstavljen je sopstvenim i međusobnim otpornostima uzemljivača kao mera uticaja struje kvara kroz uzemljivač na vrednosti sopstvenih potencijala i potencijala ostalih uzemljivača, ali i na raspodelu struje kvara unutar podsistema uzemljivača.

Zemlja predstavlja čist aktivni otpor, a celo uzemljenje sa zemljovodnim provodnikom i uzemljivačem ima vrlo malu induktivnost za učestanost od 50 Hz, tako da je svaki uzemljivač predstavljen svojom aktivnom otpornošću $R_1, R_2, \dots, R_{N_{UZ}}$. Dakle, sopstvene i međusobne impedanse uzemljivača sadrže samo realne delove. **Matematički model** za podsistem uzemljivača sa slike 1 može se napisati u *impedantnoj formi*:

$$[\underline{V}_{UZ}]_{N \times 1} = [\underline{Z}_{UZ}]_{N \times N} \cdot [\underline{I}_{UZ}]_{N \times 1}, \quad (1)$$

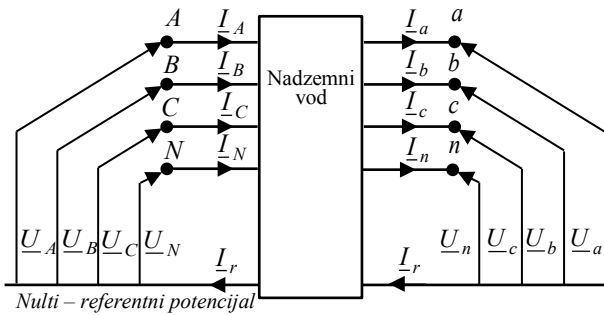
i *admitantnoj formi*:

$$[\underline{I}_{UZ}]_{N \times 1} = [\underline{Y}_{UZ}]_{N \times N} \cdot [\underline{V}_{UZ}]_{N \times 1} \quad (2)$$

$$([\underline{Y}_{UZ}]_{N \times N} = [\underline{Z}_{UZ}]_{N \times N}^{-1})$$

2.2. Podsistem provodnika

Na slici 2 prikazana je blok šema četvorožičnog nadzemnog voda (uvažavajući uticaj povratnog puta kroz zemlju) [6].



Slika 2. Blok šema nadzemnog voda

Kao inicijalnu tačku posmatranja prilikom izvođenja matematičkog modela nadzemnog voda, može se uzeti jednačina za **padove napona** duž provodnika, iskazanu u impedantnoj formi:

$$[\underline{\Delta U}_{Pr}]_{4 \times 1} = [\underline{Z}_{Pr}]_{4 \times 4} \cdot [\underline{i}_{Pr}]_{4 \times 1}. \quad (3)$$

Relacija (3) predstavlja odnos struja duž provodnika i pada napona na provodnicima (razlike potencijala sa krajeva provodnika). U cilju formiranja matematičkog modela, u skladu sa metodom potencijala čvorova, vektor kolona padova napona duž provodnika, mora se iskazati kao razlika potencijala sa početka i kraja provodnika, tj.:

$$[\underline{\Delta U}_{Pr}]_{4 \times 1} = [\underline{V}_{pocetak}]_{4 \times 1} - [\underline{V}_{kraj}]_{4 \times 1}. \quad (4)$$

Posmatrano u admitantnoj formi, jednačina za padove napona duž provodnika dobija sledeći oblik:

$$[\underline{i}_{Pr}] = [\underline{Y}_{Pr}] \cdot [\underline{\Delta U}_{Pr}], \quad [\underline{Y}_{Pr}] = [\underline{Z}_{Pr}]^{-1}, \quad (5)$$

pri čemu je sa $[\underline{Y}_{Pr}]_{4 \times 4}$ naznačena matrica sopstvenih i međusobnih admitansi provodnika nadzemnog voda.

Ako se za referentni smer struja provodnika usvoji smer od početka prema kraju provodnika, onda su struje injektirane u čvorove na početku provodnika jednake strujama provodnika sa pozitivnim predznakom, a struje

injektirane u čvorove na kraju provodnika negativnim vrednostima struja provodnika:

$$[\underline{I}_{pocetak}] = [\underline{i}_{Pr}], \quad (6)$$

$$[\underline{I}_{kraj}] = -[\underline{i}_{Pr}]. \quad (7)$$

Na osnovu relaciju (5), (6) i (7) može se formirati matematički model podsistema provodnika u skladu sa metodom potencijala čvorova:

$$\begin{bmatrix} \underline{I}_{pocetak} \\ \underline{I}_{kraj} \end{bmatrix} = \begin{bmatrix} \underline{Y}_{pr} & -\underline{Y}_{pr} \\ -\underline{Y}_{pr} & \underline{Y}_{pr} \end{bmatrix} \cdot \begin{bmatrix} \underline{V}_{pocetak} \\ \underline{V}_{kraj} \end{bmatrix} = [\underline{Y}_{2pr}] \cdot \begin{bmatrix} \underline{V}_{pocetak} \\ \underline{V}_{kraj} \end{bmatrix} \quad (8)$$

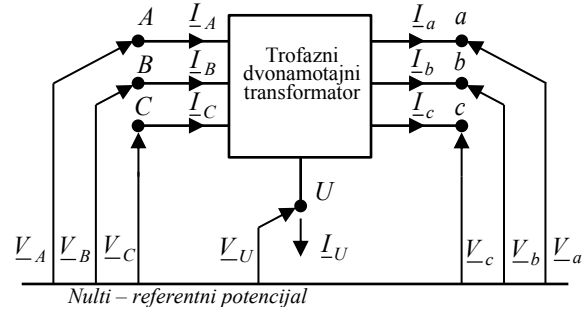
Sopstvene i međusobne vrednosti parametara provodnika mogu se izračunati na osnovu Karsonovih i Polačkovih formule [2]. Jednostavni Karsonovi izrazi za proračun podužne vrednosti sopstvenih i međusobnih impedansi provodnika, korišćeni u ovom radu su [3,6]:

$$z_{ii} = r_z + r_{pi} + j0.1445 \cdot \log_{10} \frac{D_e}{a_{ei}}, \quad (9)$$

$$z_{ij} = r_z + j0.1445 \cdot \log_{10} \frac{D_e}{d_{ij}}. \quad (10)$$

2.3. Podsistem transformatora

Blok šema trofaznog dvonamotajnog transformatora [6] prikazana je na slici 3.



Slika 3. Blok šema trofaznog dvonamotajnog transformatora

Razmatrani transformator ima ukupno sedam tačaka za galvansko povezivanje sa ostatkom sistema. Sa A, B i C, su naznačeni krajevi visokonaponske (VN) strane, a sa a, b i c krajevi niskonaponske (NN) strane transformatora. Priključak za uzemljenje transformatora, povezivanje sa podsistemom uzemljenja, naznačen je sa U.

Matematički model dvonamotajnog transformatora u **domenu simetričnih komponenti**, napisan u formi koja odgovara **metodu potencijala čvorova**, glasi:

$$[\underline{I}_{T2}^{dio}]_{7 \times 1} = [\underline{Y}_{T2}^{dio}]_{7 \times 7} \cdot [\underline{V}_{T2}^{dio}]_{7 \times 1}. \quad (11)$$

U cilju dobijanja konačnog **matematičkog modela transformatora u faznom domenu**, potrebno je izvršiti transformaciju matematičkog modela napisanog u domenu simetričnih komponenti. U tu svrhu koriste se matrica transformacija (domena simetričnih komponentni u fazni domen) $[\underline{A}]$ i njena inverzna matrica $[\underline{A}]^{-1}$ [1,5,7,9], odnosno proširena matrica transformacije $[\underline{A}_2]$ i njena inverzna matrica $[\underline{A}_2]^{-1}$.

Matematički model transformatora u **faznom domenu**:

$$[\underline{I}_{T2}]_{7 \times 1} = [\underline{Y}_{T2}]_{7 \times 7} \cdot [\underline{V}_{T2}]_{7 \times 1} \quad (12)$$

2.3. Podsystem elektroenergetskog okruženja

Podsystem elektroenergetskog okruženja predstavlja ceo EES iz koga je potisnut sistem od interesa. Elektroenergetsko okruženje se može predstaviti kao „crna kutija” koja sadrži sve elemente EES osim elemenata sistema od interesa, a sa sistemom od interesa je povezan isključivo galvanski, preko graničnih čvorova.

Model elektroenergetskog okruženja se može definisati uz pomoć snage jednogpolnog i trofaznog kratkog spoja, kao i napona u graničnim čvorovima u kojima se izvode potrebni proračuni.

Matematički model podsystema elektroenergetskog okruženja (sa N_E čvorova) u domenu simetričnih komponenti opisan je sledećom relacijom:

$$[\underline{I}_{EEO}^{dio}]_{N_E \times 1} = [\underline{Y}_{EEO}^{dio}]_{N_E \times N_E} \cdot [\underline{V}_{EEO}^{dio}]_{N_E \times 1} \quad (13)$$

Matematički model podsystema elektroenergetskog okruženja u faznom domenu ima izgled:

$$[\underline{I}_{EEO}]_{3N_E \times 1} = [\underline{Y}_{EEO}]_{3N_E \times 3N_E} \cdot [\underline{V}_{EEO}]_{3N_E \times 1} \quad (14)$$

3. INTEGRACIJA PARCIJALNIH MATEMATIČKIH MODELA

Nakon određivanja pojedinačnih matematičkih modela sistema, potrebno je izvršiti njihovu integraciju u cilju formiranja konačnog matematičkog modela EES.

U cilju formiranja matematičkog modela EES potrebno je definisati jedinstvenu numeraciju svih čvorova. Ukupan broj čvorova definiše dimenziju celokupnog sistema, odnosno broj jednačina preko kojih se može odrediti režim EES-a.

Jedinstvena numeracija svih čvorova omogućava da se za svaki podsystem, formira matrica admitansi čije su dimenzija jednake ukupnom broju čvorova celokupnog EES. Integracijom četiri matrice admitansi, koje su nezavisno formirane za četiri podsystema, jednostavno se formira matrica admitansi EES. Integracija matrica admitansi podsystema se svodi na sabiranjem elemenata tih matrica.

Konačno, matematički model EES sa N čvorova, glasi:

$$[\underline{I}_{EES}]_{N \times 1} = [\underline{Y}_{EES}]_{N \times N} \cdot [\underline{V}_{EES}]_{N \times 1} \quad (15)$$

4. PRORAČUN REŽIMA EES SA KVAROM

Postupak za proračuna režima EES sa kratkim spojem se, generalno, sastoji iz četiri dekompozicije [3,5,7]:

- dekompozicija režima s kratkim spojem na poznati režim pre kratkog spoja i režim jednog izvedenog trofaznog delta kola (Δ -kola) – delta režim (Δ -režim) – I dekompozicija;
- dekompozicija Δ -režima na naizmeničnu i jednosmernu komponentu – II dekompozicija;
- dekompozicija naizmenične komponente Δ -kola na tri vremenske sekvence suptranzitni, tranzitni i ustaljeni period (režim) – III dekompozicija;

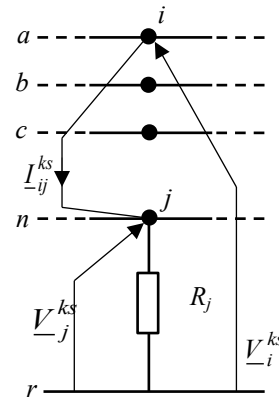
- transformacija naizmeničnih režima Δ -kola u sve tri vremenske sekvence iz domena faznih u domen simetričnih komponenti – IV dekompozicija.

U ovom radu proračun režima EES sa kratkim spojem se bazira na prvoj dekompoziciji, odnosno na proračun režima Δ -kola – Δ -režima. Motiv za tretman režima EES sa kratkim spojem u skladu sa ostale tri dekompozicije ogleda se u postojanju naizmeničnih mašina u EES, koje diktiraju promenu amplitude ma koje veličine naizmenične komponente režima Δ -kola u vremenu.

4.1. Proračun režima Δ -kola

U slučaju dekompozicije kola sa kvarom režim pre kvara je poznat, dok se režim Δ -kola računa, tako što se idealni naponski generator umetne samo na mestu kvara, dok se svi ostali izvori pasiviziraju.

Na slici 4 prikazan je deo EES na kome je nastupio jednofazni zemljospoj [6].



Slika 4. – Deo EES na kome je nastupio jednofazni zemljospoj

U ovom radu razmatra se metalni kratak spoj (kratak spoj bez luka, koji se predstavlja kao kvar preko impedanse). U skladu sa tim, grana sa strujom kratkog spoja je bezimpedantna grana.

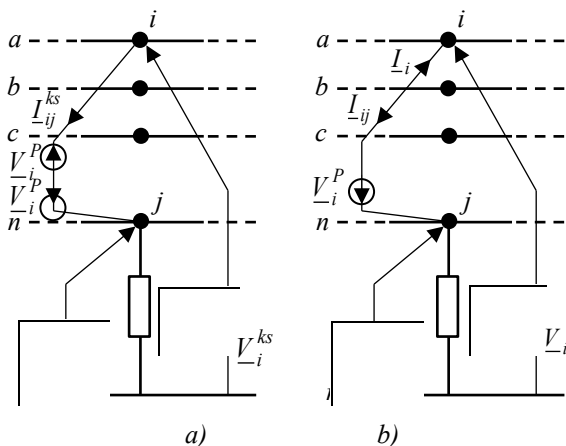
U granu sa kvarom se insertuju dva jednaka idealna naponska izvora vezana u opoziciju (zajedno kao jedan nulti idealni naponski izvor), slika 5a.

Napon insertovanog idealnog naponskog izvora $\underline{V}_i^P$ sa slike 5a jednak je naponu na mestu kvara pre kvara. Ako se na ovom mestu, a u skladu sa prvom dekompozicijom, kolo sa slike 5a dekomponuje na dva kola, i to tako da se u prvom kolu zadrži idealni naponski izvor usmeren od zemlje, odnosno tačke j , ka sistemu, a u drugom kolu idealni naponski izvor, koji je usmeren od sistema ka zemlji, odnosno tački j .

To drugo kolo, nad kojim će se vršiti proračuni od interesa naziva se Δ -kolo i prikazano je na slici 5b.

Naponi tačaka i i j u ekvivalentnom, Δ -kolu, sa slike 5b, naznačeni su sa $\underline{V}_i$ i $\underline{V}_j$, respektivno.

Struje injektirane u čvorove i i j , naznačene su sa $\underline{I}_i$ i $\underline{I}_j$, respektivno, dok je struja kratkog spoja na mestu kratkog spoja naznačena sa $\underline{I}_{ij}$.



Slika 5. Sistem sa kratkim spojom u tački i a) sa insertovanim parom idealnih naponskih izvora i b) Δ-kolo na kojem je simuliran jednofazni zemljospoj

Na osnovu slike, sledi:

$$V_i = V_j - V_i^P, \quad (16)$$

$$I_i = -I_j \quad (= -I_{ij}), \quad (17)$$

Konačno, matematički model Δ-kola, u matricnoj formi, raspregnuti izraz (15):

$$\begin{bmatrix} V_1 \\ \vdots \\ V_i \\ \vdots \\ V_j \\ \vdots \\ V_N \end{bmatrix} = \begin{bmatrix} Z_{11} & \cdots & Z_{1i} & \cdots & Z_{1j} & \cdots & Z_{1N} \\ \vdots & & \vdots & & \vdots & & \vdots \\ Z_{1i} & \cdots & Z_{ii} & \cdots & Z_{ij} & \cdots & Z_{iN} \\ \vdots & & \vdots & & \vdots & & \vdots \\ Z_{j1} & \cdots & Z_{ji} & \cdots & Z_{jj} & \cdots & Z_{jN} \\ \vdots & & \vdots & & \vdots & & \vdots \\ Z_{N1} & \cdots & Z_{Ni} & \cdots & Z_{Nj} & \cdots & Z_{NN} \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ I_i \\ \vdots \\ I_j \\ \vdots \\ 0 \end{bmatrix} \quad (18)$$

Izraz (18) zapisan u skraćenoj formi:

$$[\underline{V}]_{N \times 1} = [\underline{Z}_{EES}]_{N \times N} \cdot [\underline{I}]_{N \times 1}. \quad (19)$$

Treba uočiti da su jedine nenulte vrednosti struja, struja injektiranih u čvorove, struje kvara injektirane u čvorove zahvaćene kvarom. Na osnovu izraza (16), (17) i (18), mogu se odrediti vrednosti struja na mestu kvara (za kratak spoj između tačaka i i j):

$$I_i = -I_j = \frac{-V_i^P}{Z_{ii} - Z_{ij} - Z_{ji} + Z_{jj}} = \frac{-V_i^P}{Z_{EKV}} (= -I_{ij}). \quad (20)$$

Za poznate vrednosti struja kvara, koje su injektirane u čvorove i i j, na osnovu izraza (18), mogu se odrediti vrednosti potencijala Δ-kola za sve čvorove EES. Za poznate vrednosti potencijala svih čvorova, jednoznačno je definisan režim celokupnog Δ-kola. Superpozicijom režima Δ-kola i kola pre kvara dobija se režima kola sa kvarom.

5. ZAKLJUČAK

U ovom radu prikazan je postupak za formiranje matematičkog modela EES u faznom domenu, i na osnovu njega proračun režima EES sa kvarom. Postupak modelovanja je pojednostavljen podelom sistema na četiri

podсистema između kojih postoje isključivo galvanske veze. Za svaki podsystem formiran je poseban nezavisan model, a modeli su razvijeni u faznom domenu, u admittantnoj formi. Integracijom parcijalnih modela formiran je model EES sa kvarom.

Primenom prikazanog postupka izbegnute su uobičajene aproksimacije pri modelovanju i proračunima struje kvara, raspodele struje kvara i raspodele potencijala režima EES sa kvarom. Neke od njih su:

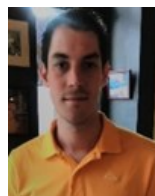
- uticaj galvanske veze krajeva provodnika električne zaštite i uzemljivača transformatora;
- sistem uzemljenja transformatorske stanice i stubova i zvezdišta transformatora nisu na nultom potencijalu.

Time je prikazan model pogodan za proračun (dimenzionisanje) uzemljivača postrojenja, zaštitnih užadi, električne zaštite, napona koraka i dodira itd. Istovremeno prikazani model može da bude merodavan za vrednovanje tačnosti i verodostojnosti rezultata dobijenih korišćenjem uprošćenih matematičkih modela i za verifikaciju (ne)opravdanosti uvođenja aproksimacija na tačnost proračuna režima sa kvarom.

6. LITERATURA

- [1] V.C.Strezoski: *Analiza elektroenergetskih sistema*, Fakultet tehničkih nauka, Novi Sad, 2007.
- [2] G.S.Švenda: *Osnovi elektroenergetike: matematički modeli i proračuni*, Fakultet tehničkih nauka, Novi Sad, 2007.
- [3] A.R.Bergen, V.Vittal: *Power Systems Analysis*, Prentice-Hall Inc, New Jersey, USA, 2000.
- [4] J.J.Grainger, W.D.Stevenson: *Power Systems Analysis*, McGraw-Hill, Inc., New Jersey, USA, 1994.
- [5] Vladimir C. Strezoski, Dragan S. Popović: *Proračuni stacionarnih režima elektroenergetskih sistema*, Fakultet tehničkih nauka, Novi Sad, 2008.
- [6] G.S.Švenda: *Specijalizovani softveri u elektroenergetici*, materijal sa predavanja, Fakultet tehničkih nauka, Novi Sad, 2015.
- [7] N.D.Tleis: *Power Systems Modelling and Fault Analysis: Theory and Practice*, Newnes Power Engineering, Elsevier Ltd, Oxford, UK, 2008.
- [8] K.Kasaš-Lažetić: *Modelovanje impedanse zemlje kao povratnog provodnika*, doktorska disertacija, Fakultet tehničkih nauka, Novi Sad, 2015.
- [9] W.D.Stevenson: *Elements of Power System Analysis*, Fourth Edition, McGraw-Hill, New York, USA, 1982.

Kratka biografija:



Mirko Đurić rođen je u Valjevu 1989. god. Srednju tehničku školu (Elektrotehničar računara) završio je 2008. god. u Valjevu. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranio je 2015. god. Iste godine upisao se na master studije.

PREDIKCIJA VREDNOSTI KRIPTOVALUTA ANALIZOM ISTORIJSKIH CENA, BLOKČEJN INFORMACIJA I SENTIMENTA TVITOVA**PREDICTING THE VALUES OF CRYPTOCURRENCIES USING HISTORICAL VALUES, BLOCKCHAIN INFORMATION AND TWITTER SENTIMENT**

Milica Milutinović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**1. UVOD**

Kratak sadržaj – Ovaj rad bavi se problemom predikcije cene Bitkoina na osnovu istorijskih cena i blokčejn informacija. Cenu kriptovaluta velikim delom diktiraju špekulacije pa je ispitano u kojoj meri sentiment analiza tvitova doprinosi padu ili rastu cena. Pored podataka o ceni i sentimentu, korišćeni su blokčejn podaci i istorijski podaci za još tri popularne kriptovalute: Lajtkoin, Etereum i Ripl. Za sentiment analizu ispitane su tri tehnike: konvolucione neuronske mreže, ansambli leksikona i metoda zasnovana na jezičkim pravilima. Rezultati sentiment analize upotrebljeni su u daljem procesu predviđanja kriptovaluta. Izabrani model za predikciju je rekurentna neuronska mreža sa GRU ćelijama. U fazi evaluacije posmatrala se tačnost predviđanja kretanja cene, odnosno da li će ona da raste ili opada. Takođe, upotrebljena je i relativna tačnost u kojoj se posmatra odnos stvarne cene i prediktovane. Najbolji rezultat dostigao je 57,3% tačnosti u predikciji kretanja cene, odnosno 99,39% relativne tačnosti.

Ključne reči: sentiment analiza; predikcija; Bitkoin;

Abstract – This paper studies the problem of Bitcoin price prediction based on historical prices and blockchain information. As the price of a cryptocurrency is dictated largely by speculation we also examine the influence of twitter sentiment on the performance of our predictive models. We have also experimented with historical data for three other popular cryptocurrencies: Litecoin, Ethereum and Ripple. We experimented with three techniques for sentiment mining: convolutional neural networks, ensemble of lexicons and linguistic rules. The results of sentiment analysis were integrated with other data to train a recurrent neural network with GRU cell as our price prediction model. We evaluated our model both as a binary classifier (predicting whether the price will go up or down) and a regression predictor (predicting the actual price). The accuracy of the classification model was 57.3%, while the relative accuracy of the regression model was 99.39%.

Keywords: sentiment analysis; prediction; Bitcoin;

Kriptovalute predstavljaju oblik digitalne imovine koje su zasnovane na kriptografskom protokolu koje omogućavaju korisnicima da ih čuvaju i razmenjuju putem mreže [1]. Satoshi Nakamoto je u januaru 2009. implementirao Bitkoin (en. *Bitcoin*), što predstavlja prvu kriptovalutu. Od tada njihova popularnost stalno raste, najviše zbog osobina kao što su decentralizovanost, transparentnost i sigurnost. U ovom radu će biti opisani pristupi i rešenja za predviđanje cene Bitkoina. U poređenju sa „klasičnim“ akcijama na berzi, za kriptovalute se mogu vezati informacije o samoj blokčejn tehnologiji koje su javno dostupne što uvodi dodatnu semantiku [2].

U narednoj sekciji je izložen pregled postojeće literature vezane za problem predviđanja cena kriptovaluta i sentiment analize podataka. Treća sekcija sadrži informacije o prikupljanju podataka. Četvrta sekcija opisuje metodologije i alate korišćene u radu. U petoj sekciji objašnjena je eksperimentalna evaluacija rezultata, dok poslednja sekcija zaključuje rad predlaže pravce mogućeg daljeg istraživanja.

2. PRETHODNA REŠENJA

Dosadašnja literatura pokriva veliki broj problema koji teže da se objasne u radu. Proučavanja sentiment analize teksta su dostigla visok nivo, a pristupi koji su rađeni variraju od metoda koje ne koriste nikakav vid učenja odnosno koja se baziraju na pravilima i/ili leksikonima, do metoda koje koriste mašinsko učenje za te svrhe. Metod zasnovan na pravilima [3], VADER alat, teži da odredi sentiment na osnovu jezičkih konstrukcija, odnosno pravila koje vladaju u njima. Zbog izuzetnih rezultata koji su dobijeni (0,96 F1 mera tačnosti za sentiment tvitova), implementacija ovog metoda je korišćena kao jedan od načina za određivanje sentimenta. Pristup koji određuje sentiment teksta baziran na ansamblima leksikonima [4] traži reči u formiranim leksikonima i u zavisnosti od broja ponavljanja u njima, dodeljuju svakoj reči jedinstven sentiment. Ovaj metod unapredio je proste metode bazirane na leksikonima koji su imali mnogih nedostataka [5], a dostigao je približnu tačnost kao i metode mašinskog učenja u određivanju sentimenta teksta. Iz tih razloga, implementiran je, a detalji će biti objašnjeni kasnije u radu. Analiziranje sentimenta u tekstu često se rešava primenom algoritama dubokog učenja. Rad zasnovan na konvolucionim neuronskim mrežama [6] koristi konvolucionu arhitekturu neuronske mreže i tzv. Bag-of-Words model [7] kao

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Aleksandar Kovačević, van. prof.

ulazne podatke u mrežu. Iako arhitektura u pomenutom radu nije velike dubine, podešavanjem parametara mreže i korišćenjem regularizacije u procesu učenja, ovaj model pokazao je bolje rezultate od drugih koji su upoređivani u istom radu.

U radu koji se bavi predikcijom volatilnosti akcija na kineskom tržištu [8], korišćenjem rekurentne neuronske mreže sa sentimentalnim podacima, dobijena je tačnost od 65%. Međutim, u tom radu predikcija je vršena samo za jedan vremenski korak unapred.

3. SAKUPLJANJE PODATAKA

Skup podataka koji je analiziran dobijen je prikupljanjem tvitova influensera u domenu kriptovaluta, istorijskih cena i informacija o blokčejnu u periodu od Januara 2016. godine do kraja Juna 2018. godine.

Tvitovi su prikupljeni zahvaljujući projektu [9] koji formira url na osnovu koga skrepuje (eng. *scrapping*) stranicu preko koje sakuplja tvitove. Zahvaljujući ovom pristupu preuzet je veliki broj tvitova u kratkom vremenskom intervalu. Nakon procesa sentiment analize, dobijeni rezultati prosleđeni su u fazu predikcije. Iskorišćeno je 7 numeričkih atributa u daljem toku:

- vader_sent i vader_pol – intezitet i polaritet sentimenta dobijen korišćenjem VADER alata
- lex_sent - sentiment dobijen korišćenjem modela ansambla leksikona
- cnn_sent i cnn_pol - intezitet i polaritet sentimenta dobijen korišćenjem CNN arhitekture
- favorites - predstavlja broj "lajkovanja" tvita
- retweets – predstavlja broj deljenja tvita

Istorijski podaci o cenama su preuzeti sa Poloniex API-ja za Bitkoin, Lajtkoin, Etereum i Ripl. Podaci se nalaze u vremenskom koraku od 5 minuta za period od Januara 2016. godine do Aprila 2018. godine. Eksplorativnom analizom atributa je utvrđeno da podaci o ceni imaju veliku međusobnu korelaciju (približno 1) te je za dalju obradu korišćeno: close (cena na kraju intervala), date (vreme u *unix timestamp*-ovima), volume (trgovani iznos u poslednja 24h u dolarima) i quoteVolume (trgovani iznos u poslednja 24h u datoj kriptovaluti). Takođe je zaključeno da postoji korelacija između različitih valuta, tako da su za konačnu predikciju cene Bitkoina validirani i podaci od drugih valuta.

Preostalih 10 atributa su podaci o Bitcoin blokčejnu koji su preuzeti sa blockchain.info API-ja i unose dodatnu semantiku o transakcijama ove kriptovalute.

S obzirom da prikupljeni tvitovi nemaju pravilan vremenski interval, a podaci o blokčejnu se nalaze na uniformnom intervalu od 1 dan, nije ih bilo moguće spojiti sa podacima o cenama koje imaju unforman interval od 5 minuta. Stoga su oni interpolirani na isti vremenski raspon. U radu je upotrebljena interpolacija na 1 sat i na 10 minuta.

Za proces formiranja rečnika u metodologiji koja koristi ansamble leksikona upotrebljeno je 5 skupova podataka koji sadrže anotirane tvitove: *Stanford Twitter Sentiment Test Set* [10], *Sentiment Strength Twitter Dataset* [11], skup od 9637 tvitova skinutih sa *Kaggle*-a, skup od 50800 tvitova skinutih sa *Kaggle*-a i *Senticnet* [12].

4. METODOLOGIJA

U cilju istraživanja sentimenta tvitova, isprobana su tri popularna rešenja. Rezultati metoda su upotrebljeni da se odredi u kojoj meri doprinose poboljšanju predikcije.

U sva tri pristupa, tvitovi su pretprocesirani primenom nekoliko koraka: izbacivanjem emotikona, akronima i skraćenica, pretvaranjem velikih u mala slova, uklanjanjem *URL*-ova, praznih mesta, stop reči, znakova interpunkcije i specijalnih karaktera. Potom je primenjena morfološka normalizacija reči. Skupovi podataka korišćeni u procesu formiranja leksikona pretprocesirani su na sličan način, s tim da su im i ciljne labele svedene na isti interval, odnosno 1 za pozitivan, -1 za negativan, a 0 za neutralan tvit.

Prvi isprobani pristup koristi jezička pravila za računanje sentimenta [3]. U pomenutom pristupu posmatra se grupa reči koja predstavlja neku celinu, pa se njoj određuje sentiment, a ne samo pojedinačna reč. Najbolji rezultati se dobijaju ukoliko se radi analiza tzv. mikro-blogova, odnosno kratkih statusa. Implementacija ovog pristupa je javno dostupna i preuzeta je.

Drugi način bazira se na formiranju leksikona, koji se potom koriste da se pronađu reči koje imaju pozitivan ili negativan sentiment. Ovom metodologijom najpre su generisani leksikoni koristeći pomenute skupove. Leksikoni su kreirani tako što su se pronalazile reči koje su najzastupljenije u pozitivnim, odnosno negativnim recenzijama. Na ovaj način, napravljen je skup od 10 rečnika za navedenih 5 skupova, odnosno rečnici sa po 5 i 25 najpozitivnijih i najnegativnijih reči za svaki skup posebno. Formiran je i jedan rečnik koji je agregira sve skupove podataka zajedno i među njima traži skup od 25 najpozitivnijih i najnegativnijih reči. Preostala 4 rečnika formirana su na osnovu reči koje jednoznačno doprinose promeni inteziteta sentimenta i preuzeti su iz rada [13].

Sledeći korak predstavlja klasifikaciju sentimenta na pozitivan ili negativan na osnovu ansambla rečnika. Ovaj model uzima pojednačne reči iz rečenice i predviđa njihov sentiment tako što traži da li se ta reč pojavljuje više kao pozitivna ili u negativna. Ukoliko preovladava pojavljivanje reči u leksikonima sa negativnim rečima, vrednost sentimenta reči će biti -1, odnosno 1 u suprotnom. Ukoliko se reč ne pojavljuje ni u jednom leksikonu, vrednost njenog sentimenta će biti 0. Nakon što se svakoj reči dodeli sentiment, računa se ukupna vrednost sentimenta na nivou teksta, odnosno tvita u našem slučaju. Ukupan sentiment za leksikon l i tvit t računa se po formuli:

$$s_l = \begin{cases} 1, \wedge \text{ if } \sum (l, t) > 0 \\ -1, \wedge \text{ if } \sum (l, t) < 0 \\ 0, \wedge \text{ otherwise} \end{cases}$$

Konačan rezultat sentiment analize za jedan tvit dobija se usrednjavanjem rezultata svih leksikona. Ukoliko je izračunata vrednost veća od 0, uzima se da je tvit pozitivan, ukoliko je manja od 0, smatra se da je tvit negativan, a u suprotnom je tvit neutralan.

Treći pristup korišćen u ovom radu zasniva se na primeni konvolucionih neuronskih mreža, koje koriste konvolucion operator da proizvode nove osobine. Ulaz u mrežu dobijen je formiranjem tzv. Bag-of-Words modela i predstavlja matricu fiksne dužine $N \times K$, gde N predstavlja

broj tvitova u datom intervalu, a K fiksnu dužinu rečenica na kojima je izvršeno upotpunjavanje (*padding*) ili odsecanje (*truncate*) u odnosu na fiksnu dužinu. Izlaz iz mreže daje dve vrednosti, tj. odnos rasta ili pada cene i oznaku da li cena raste ili opada.

Kako se o tvitovima čuvaju informacije o danu kada su kreirani, izlazni podaci u procesu treniranja se dobijaju tako što se računa odnos cena tog dana i sutrašnjeg dana i oznaku da li cena raste ili opada. Na primer, ukoliko je tvit postavljen dana 10.12.2017. posmatra se cena tog dana i cena 11.12.2017, računa se količnik cena ta dva dana, a postavlja se i indikator da li cena raste ili opada. Zbog nemogućnosti da se odredi čiji je tvit važniji, svi tvitovi objavljeni istog dana imaju isti odnos cena i isti indikator koji govori o kretanju cene kriptovalute.

Za proces računanja predikcije korišćene su rekurentne neuronske mreže jer imaju sposobnost da adresiraju vremenske podatke koristeći unutrašnju memoriju.

S obzirom da su svi atributi numerički, odrađena je normalizacija u intervalu $[0,1]$ na svim atributima. Kako bi se kreirala veća količina podataka upotrebljen je preklapajući obučavajući i trening skup. Ova redundantnost omogućena je korišćenjem „klizećeg” vremenskog okvira za svaki skup ulaznih i izlaznih vrednosti.

Odabir arhitekture RNN mreže je dobijen eksperimentalno nakon što je testirano više različitih arhitektura. GRU koristi manje parametara od LSTM (eng. *Long short-term memory*), a performanse se neznatno razlikuju. Korišćene su GRU ćelije koje sadrže 96 neurona (ćelija) u jednom sloju. Na izlazu je Fully-Connected sloj sa 24 izlaza koji predstavljaju vremenske korake predikcije. Kako bi se sprečio overfitting, dodata je Dropout regularizacija. Testirani su modeli sa jednim i dva unutrašnja skrivena sloja. Za funkciju greške odabrana je MSE (*mean squared error*), gde je takođe korišćena Adamova optimizacija.

5. EKSPERIMENTI I REZULTATI

Za osnovnu meru tačnosti je uzet broj uspešno predviđenih kretanja valuta – pad ili rast što može da se svede na binarnu klasifikaciju. Formula po kojoj se ona izračunava:

$$Acc = \frac{successfullpredictions}{totalpredictions}$$

Relativna tačnost, dobija se računanjem odnosa stvarne cene i prediktovane. Osim pomenutih metrika, korišćene su i srednja kvadratna i srednja apsolutna greška, radi upoređivanja tačnosti rezultata.

U postupku evaluacije kreirano je više skupova kako bi se validirali podaci dobijeni u postupku sentiment analize. Urađeni su i testovi radi ispitivanja parametara koji dovode do poboljšanja rezultata. Tim putem ispitano od koliko je značaja vremenski raspon u kojem su interpolirani podaci, broj ulaznih vremenskih koraka u proces obučavanja, kao i informacije o drugim vrednostima kriptovaluta.

Interesantno je primetiti da informacije o sentimentu influensera dovode do boljih rezultata u određenoj meri. Uočeno je da povećanje broja ulaznih jedinica, kao i informacije o vrednostima drugih kriptovaluta ne doprinose boljim rezultatima. Ipak, najveće poboljšanje postignuto je u slučaju

smanjenja intervala interpolacije sa 1. sata na 10 minuta. U Tabeli 1. prikazani su rezultati za najbolji validirani test, odnosno za slučaj interpolacije na 10 minuta, gde se prediktuju vrednosti za naredna 2 sata, na osnovu vrednosti u prethodna 24 sata. Oznake skupova sastoje se od atributa koji ih sačinjavaju: **b** (blokčejn podaci), **h** (istorijske podaci), **s** (sentiment podaci), **v** (sentiment podaci dobijeni vader metodom), **c** (sentiment podaci dobijeni cnn metodom) i **l** (sentiment podaci dobijeni ansamblima leksikona).

Tabela 1. Rezultati

	acc (%)	rel_acc (%)	mse	mae
bps	53.31	98,83	9180.78	79.47
bp	52.74	97.8	27721.35	151.15
bpv	47.36	96.1	79079.49	272.38
bpl	54.32	99.2	5056.30	54.41
bpc	52.76	97.22	40463.36	190.12
s	71.22	38.57	56364.57	100.56
ps	57.3	99.39	3731.67	40.63



Slika 1. Grafik predikcije za skup ps.

Na slici 1. dat je prikaz najboljeg skupa atributa ps, za date parametre. Može se uočiti da su prediktovani rezultati prilično korelirani sa stvarnim.

6. ZAKLJUČAK

U ovom radu predložen je novi pristup predviđanja cena kombinacijom različitih skupova podataka u cilju preciznijih rezultata. Podaci o sentimentalnoj analizi uticajnih ljudi na Tviteru doprineli su poboljšanju predikcije, međutim, potrebno je dodatno istraživanje kako bi se ispitao stepen uspešnosti. Ovim rezultatima je potvrđena težina i kompleksnost problema predviđanja cene na berzi. Moguća obrazloženja lošijih rezultata je relativno mali period skupa podataka (2 godine i 6 meseci) i činjenica da je u proteklih godinu dana zabeležen rast nekih kriptovaluta čak i do 10,000% te je stvoren tržišni balon (*market bubble*) koji je polako počeo da puca u januaru 2018. godine.

Budući pravci istraživanja bi mogli da razmotre korišćenje nekih eksperimentalnih modela neuronskih mreža u rešavanju problematike. Takođe, bilo bi pogodno istražiti koliko pojedini tvitovi utiču na promenu cena, pa u skladu sa tim da im se dodele odgovarajuće težine. Pored ovoga, bilo bi zanimljivo istražiti i kako finansijske vesti i neki drugi indikatori utiču i na druge cene na berzi i pored kriptovaluta.

7. LITERATURA

- [1] Wikipedia contributors. (2018, April 11). Cryptocurrency. In *Wikipedia, The Free Encyclopedia*. Retrieved 18:46, April 13, 2018, from <https://en.wikipedia.org/w/index.php?title=Cryptocurrency&oldid=835965914>
- [2] Blockchain. (2018, April 13). Retrieved April 13, 2018, from <https://en.wikipedia.org/wiki/Blockchain>
- [3] Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014
- [4] Augustyniak, L., Kajdanowicz, T., Szymanski, P., Tuligłowicz, W., Kazienko, P., Alhajj, R., & Szymanski, B.K. (2014). Simpler is better? Lexicon-based ensemble sentiment classification beats supervised methods. 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), 924-929.
- [5] S.-H. Na, Y. Lee, S.-H. Nam, and J.-H. Lee, "Improving opinion retrieval based on query-specific sentiment lexicon," in *Advances in Information Retrieval*, ser. Lecture Notes in Computer Science, vol. 5478. Springer Berlin / Heidelberg, 2009, pp. 734–738.
- [6] K., & Y. (2014, September 03). Convolutional Neural Networks for Sentence Classification. Retrieved April 13, 2018, from <https://arxiv.org/abs/1408.5882>
- [7] Wikipedia contributors. (2018, March 26). Bag-of-words model. In *Wikipedia, The Free Encyclopedia*. Retrieved 20:07, April 13, 2018, from https://en.wikipedia.org/w/index.php?title=Bag-of-words_model&oldid=832576977
- [8] Liu, Y., Qin, Z., Li, P., & Wan, T. (2017). Stock Volatility Prediction Using Recurrent Neural Networks with Sentiment Analysis. *Advances in Artificial Intelligence: From Theory to Practice Lecture Notes in Computer Science*, 192-201. Doi:10.1007/978-3-319-60042-0_22
- [9] J. (2018, January 29). Jefferson-Henrique/GetOldTweets-python. Retrieved April 13, 2018, from <https://github.com/Jefferson-Henrique/GetOldTweets-python>
- [10] Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford (2009)
- [11] Thelwall, M., Buckley, K., Paltoglou, G.: Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology* 63(1), 163–173 (2012)
- [12] SenticNet. (n.d.). Retrieved from <https://sentic.net/>
- [13] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 142–150.

Kratka biografija:



Milica Milutinović rođena je 6.1.1995. godine u Rumi. Osnovne studije je upisala 2013. godine na Fakultet tehničkih nauka, odsek Računarstvo i automatika. Osnovne studije je završila 2017. godine, nakon čega upisuje master akademske studije na Fakultetu tehničkih nauka, smer Inteligentni sistemi.

ENEL OPTIMALNA REKONFIGURACIJA MREŽE**ENEL OPTIMAL NETWORK RECONFIGURATION**Vladan Rakić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U radu je prikazan problem optimalne rekonfiguracije mreža sa osvrtom na minimalne gubitke aktivne električne energije kao optimizacionog kriterijuma. Detaljno su opisani algoritmi za proračune tokova snaga, proračune gubitaka električne energije i algoritam izmene grana za optimalnu rekonfiguraciju mreže koji objedinjeni sačinjavaju inovativni pristup rešenja postavljenog problema pod nazivom *Enel optimalna rekonfiguracija mreža*. Na kraju je izvršena verifikacija izloženog algoritma na test mreži.

Ključne reči: *Optimalna rekonfiguracija mreže, gubici električne energije, distributivni sistemi*

Abstract – This paper displays a problem of Optimal Network Reconfiguration with emphasis on minimal active Energy Losses optimization criterion. Algorithms for Load Flow, Energy Losses and Network Reconfiguration, Branch Exchange combined comprise innovative solution for proposed problem named *Enel Optimal Network Reconfiguration*. Finally, proposed algorithm is verified on test scheme.

Key words: *Optimal Network Reconfiguration, energy loss, distributive systems*

1. UVOD

Većina distributivnih mreža (DM) se projektuje u vidu radijalnih mreža, a sve u cilju postizanja efikasnije koordinacije zaštite pojedinih elemenata mreže [1]. U prisustvu kvara, prekidačkom opremom se može manipulirati u cilju izolovanja dela mreže sa kvarom i prebacivanja izolovane potrošnje na zdrav deo mreže. U normalnim uslovima, ovi prekidači se mogu koristiti za rekonfiguraciju, u cilju smanjenja gubitaka aktivne snage, povećanja pouzdanosti ili balansiranja opterećenja u mreži [2]. Isporuka električne energije od izvora do krajnjih potrošača je uvek praćena gubicima. Ako se u obzir uzme da su distributivna preduzeća suočena sa povećanim pritiskom za ostvarivanje veće efikasnosti, da su uvedeni podsticaji i penali koji elektrodistribucija plaća potrošačima zbog nenajavljenih prekida napajanja ili lošeg kvaliteta isporučene električne energije, i da gubici direktno utiču na finansijska pitanja i efikasnost distributivnih postrojenja, jasno je da redukcija gubitaka predstavlja cilj svakog distributivnog preduzeća [3]. Jedan od efikasnih načina za postizanje ovog cilja jeste optimalna rekonfiguracija DM (ORDM).

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Goran Švenda, red.prof.

2. PRORAČUN TOKOVA SNAGA

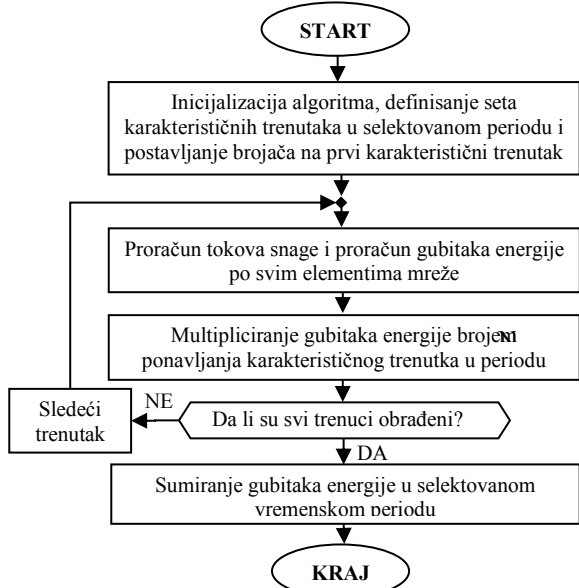
Proračun tokova snaga predstavlja proračun promenljivih stanja (režima) mreže, gde su poznati napon izvora napajanja mreže i potrošnje u svim čvorovima mreže. Na osnovu ovih vrednosti se proračunavaju vrednosti modula i faznih stavova napona u preostalim čvorovima mreže. Ovim je definisano stanje mreže. Dalje se vrši proračun ostalih režimskih promenljivih. Metode za proračun tokova snaga koje se koriste u DM razlikuju se od onih koje se koriste u prenosnim mrežama. Većina poznatih iterativnih postupaka koji se koriste u prenosnim mrežama bazirani su na matričnom postupku, dok se u DM koriste specijalizovani algoritmi koji su orijentisani ka granama [1]. Ovi postupci polaze od glavne pretpostavke, a to je da je DM radijalna. Kod metoda orijentisanih ka granama, sa porastom dimenzija, potrebe za memorijom i vreme proračuna rastu linearno, dok kod matričnih metoda potrebe za memorijom rastu kvadratno, a vreme proračuna sa trećim stepenom. Pored ovoga, pokazalo se da algoritmi orijentisani ka granama veoma brzo konvergiraju, postizajući vrlo veliku tačnost proračuna. Metoda izabrana za proračune u ovom radu je orijentisana ka granama, a to je metoda sumiranja struja, poznatija kao Shirmohammadi-ev algoritam [1].

3. ANALIZA GUBITAKA ELEKTRIČNE ENERGIJE

Vrednost gubitaka je jedan od direktnih pokazatelja efikasnosti realizacije osnovne funkcije DM, prenosa energije od prenosne mreže do krajnjih potrošača. Analiza gubitaka u DM je funkcija koja se koristi za proračun gubitaka aktivne i reaktivne snage/energije u mreži kao celini, kao i gubitka na elementima te mreže. Kada se analiza odvija za neki proizvoljan vremenski trenutak, tada je u pitanju analiza gubitaka snage. Kod analize gubitaka energije potrebno je raspolagati sa vrednostima gubitaka snage u funkciji vremena. Integral gubitaka snage u intervalu od interesa predstavlja ukupnu vrednost gubitaka energije.

Algoritam za proračun gubitaka električne energije se u svojoj osnovi oslanja na rezultate tokova snaga [1]. Prvo se pristupa inicijalizaciji algoritma što podrazumeva selektovanje i postavljanje vremenskog perioda za koji će biti vršen proračun gubitaka električne energije, vremenski korak prolaska kroz selektovani vremenski period i deo mreže nad kojim će se vršiti proračuni gubitaka električne energije. Posle inicijalizacije algoritma na osnovu selektovanog vremenskog perioda, koraka iteracije kao i same mreže, kreira se set karakterističnih trenutaka. Skup karakterističnih trenutaka se dobija izvlačenjem sezona iz selektovanog vremenskog perioda, karakteri-

stičnih dana iz sezona i karakterističnih trenutaka iz karakterističnih dana. Karakterističan dan je dan koji se ciklično ponavlja jednom ili više puta svake sedmice unutar jedne sezone. Sezone su periodi u okviru kojih se promena režima DM ponavlja ciklično na nivou jedne sedmice. Kada se ima set karakterističnih trenutaka, pristupa se iterativnom postupku. Postavlja se na prvi karakteristični trenutak, za isti se vrši proračun tokova snage i računaju se gubici snage odnosno energije za svaki element selektovanog dela DM. Dobijeni gubici električne energije se multiplicitiraju brojem ponavljanja karakterističnog trenutka unutar selektovanog vremenskog perioda. Iterativni postupak se vrši sve dok se ne obrade svi karakteristični trenuci iz prethodno definisanog seta. Kada se obrade svi karakteristični trenuci sumiraju se gubici električne energije za ceo selektovani vremenski period po elementima mreže, a sumiranjem istih se dobijaju ukupni gubici aktivne i reaktivne električne energije. Ovim se završava algoritam proračuna gubitaka električne energije prikazan na slici 2.



Slika 1. Blok dijagram algoritma proračuna gubitaka električne energije

4. OPTIMALNA REKONFIGURACIJA DM

Problem određivanja optimalne konfiguracije mreže je kompleksan, kombinatoran, nelinearan i diskretan optimizacioni problem. Za rešenje ovog problema koriste se: optimizacione metode [1,4], kombinatorno pretraživanje [1] i heurističke metode [1,6]. Izvorni algoritmi vezani su za metode bazirane na heuristici i optimalnim tokovima snaga. Suština ovih algoritama je da se maksimalnim poznavanjem fizike problema na najkraći način dođe do kvalitetnih radijalnih konfiguracija.

Heuristički algoritam korišćen u radu je algoritam izmene grana. "Izmena grana" zapravo znači "izmena mesta" normalno otvorenog (NO) rasklopnog uređaja sa njime spregnutim normalno zatvorenim (NZ) rasklopnim uređajem [4]. Glavna prednost ovog algoritma jeste izvršavanje tokova snaga samo u slučaju kada je postignuto poboljšanje vrednosti kriterijumske funkcije. Osnovna mana algoritma jeste zavisnost konačnog rešenja od početnog režima i topološkog stanja mreže. Kako je u svakom trenutku proračuna radijalnost uvek održana,

mogu se primenjivati metode za proračun tokova snaga orijentisane ka granama. Blok dijagram algoritma prikazan je na slici 1.

Kao kriterijum, odabrana je minimizacija gubitaka aktivne snage. Kriterijumska funkcija (KF) za procenu smanjenja gubitaka aktivne snage nakon izmene mesta NO/NZ prekidača, može da se predstavi kao [3]:

$$IG^{(h)} = \sum_{i=1}^{n_{ev}} \sum_{j \in a_{d_i}^h} r_{ij} (J_{ij}^h)^2, \quad (1)$$

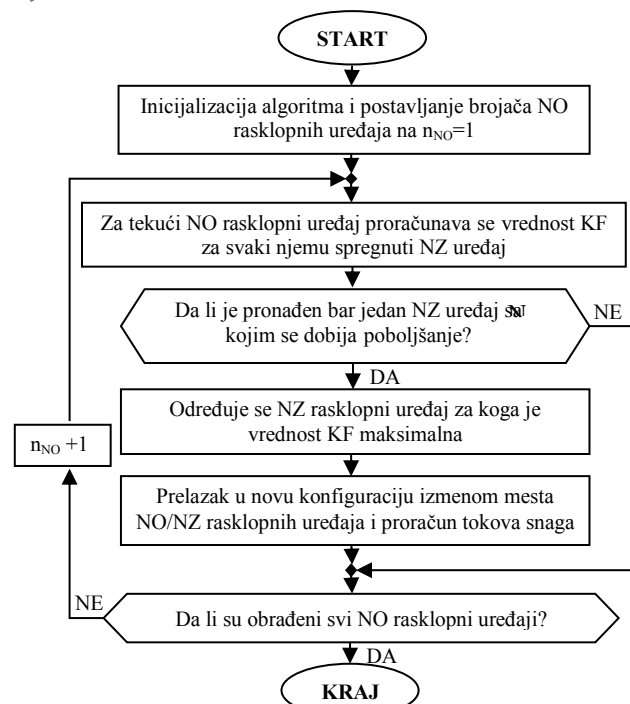
gde je:

n_{ev} – ukupan broj izvoda,

r_{ij} – rezistansa grane j koja pripada izvodu i ,

$a_{d_i}^h$ – skup indeksa grana izvoda i u konfiguraciji h ,

J_{ij}^h – moduo struje grane j izvoda i u konfiguraciji h .



Slika 2. Blok dijagram algoritma izmene grana [1]

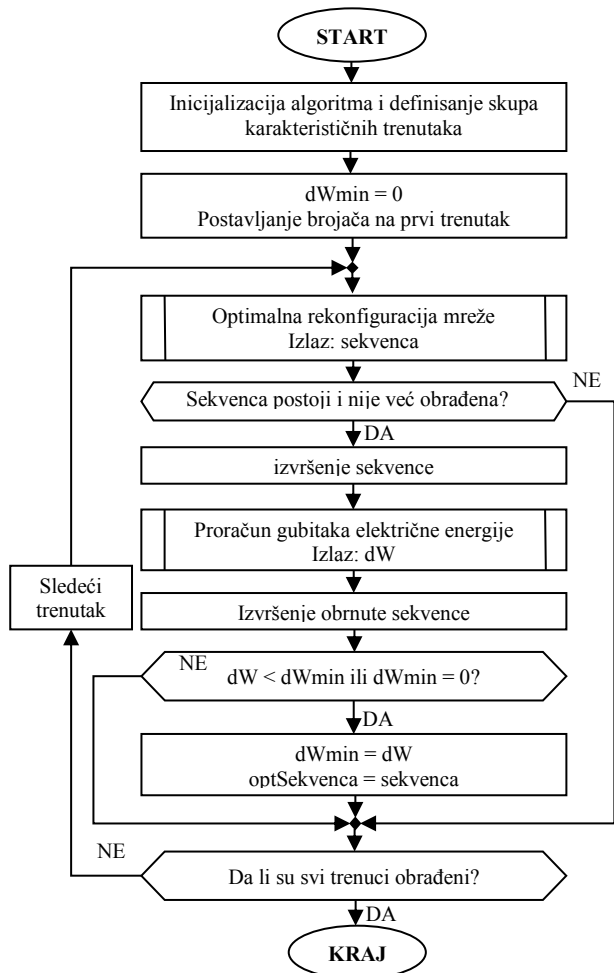
Ako je sa supskriptom h označena konfiguracija nakon izmene mesta NO/NZ rasklopnih uređaja, a sa $h-1$ konfiguracija pre te izmene, vrednost kriterijuma je [1]:

$$IG^{(h)} = IG^{(h-1)} - \Delta IG \quad (2)$$

$$\Delta IG^{(h-1)} = 2 \cdot \text{Re} \left[(J_j^{h-1})^* (\Delta V_k^{h-1} - \Delta V_m^{h-1}) \right] - |J_j^{h-1}|^2 \cdot R_{petlje} \quad (3)$$

gde su sa $IG^{(h)}$, $IG^{(h-1)}$ i ΔIG naznačene vrednosti gubitaka aktivne snage za konfiguracije h i $h-1$ i promena te vrednosti, respektivno; sa J_j^{h-1} struja grane j u radijalnoj konfiguraciji $h-1$; sa ΔV_k^{h-1} i ΔV_m^{h-1} vrednosti padova napona čvorova levo i desno od razmatranog NO rasklopnog uređaja i sa R_{petlje} rezistansa konture između dva napojna čvora koja sadrže granu sa NO rasklopnim uređajem. Pri izvođenju algoritma izmene grana dovoljno je proračunati samo vrednost izraza (1). Ako je ta vrednost veća od nule, tada su izmenom mesta razmatranog spregnutog para NO/NZ rasklopnih uređaja gubici aktivne snage smanjeni.

Pošto je problem rekonfiguracije vezan za određeni vremenski trenutak odnosno jedan režim, dok se gubici aktivne električne energije vezuju za određeni vremenski period, odnosno veći broj režima, postavlja se pitanje da li je moguće iskoristiti algoritam izmene grana za pronalaženje topološke konfiguracije mreže pri kojoj bi se imali minimalni gubici aktivne električne energije za selektovani vremenski period. Kao rešenje problema izneta je Enel optimalna rekonfiguracija DM, slika 3.



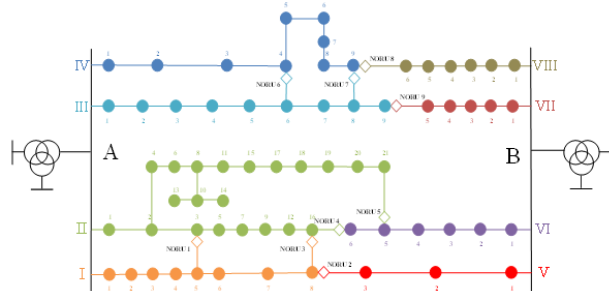
Slika 3 – Algoritam Enel optimalne rekonfiguracije mreže

Osnovna ideja metodologije je objedinjenje datih algoritama za proračun gubitaka električne energije DM i algoritma rekonfiguracije DM zasnovanog na razmeni grana. Početni korak algoritma je inicijalizaciji, što podrazumeva postavljanje željenog vremenskog perioda proračuna, odabir vremenskog korak prolaska kroz selektovani vremenski period i selektovanje dela mreže nad kojim će se vršiti proračuni. Posle inicijalizacije algoritma, na osnovu selektovanog vremenskog perioda, koraka i selektovane mreže, kreira se set karakterističnih trenutaka. Inicijalno, vrednost minimalnih gubitaka aktivne električne energije (dW_{min}) se postavlja na nulu. Zatim se pristupa iterativnom postupku sukcesivnog izvršenja algoritma izmene grana i po potrebi izvršenja algoritma proračuna gubitaka električne energije. Postavlja se na prvi karakteristični trenutak i nad istim se vrši algoritam rekonfiguracije DM. Koriste se minimalni gubici aktivne snage kao jedini optimizacioni kriterijum, željeni tipovi rasklopne opreme kao resursi i opterećenje sekcija i transformatora kao ograničenja. Kao rezultat se dobija

sekvenca manipulacija NO i NZ rasklopnih uređaja, čijim izvršenjem se prelazi u novo topološko stanje mreže. U slučaju da dobijena sekvenca nije prazna, niti prethodno obrađena, ista se izvrši i za novo dobijeno topološko stanje se proračunaju gubici električne energije nad selektovanim vremenskim periodom i sa istim korakom. Dobijena vrednost gubitaka aktivne električne energije (dW) se poredi sa trenutnom vrednošću dW_{min} i u slučaju da im je vrednost manja ili da je trenutna vrednost dW_{min} jednaka nuli, isti se postavljaju za minimalne, a trenutna sekvenca za optimalnu sekvencu ($optSekvenca$) manipulacija rasklopnim uređajima. Zatim se izvršava obrnuta sekvenca, sekvenca u kojoj su svi rasklopnih uređaji ostali isti, a redosled manipulacija i tip akcija (otvaranje, zatvaranje) koji se vrše nad rasklopnim uređajima obrnuti u odnosu na sekvencu dobijenu od strane optimalne rekonfiguracije DM. Ovim se selektovana mreža vraća u prvobitno topološko stanje, posle čega se prelazi na naredni karakteristični trenutak, odnosno iteraciju. U slučaju da je pak sekvenca prazna ili već obrađivana, prelazi se direktno na narednu iteraciju. Postupak se izvršava sve dok se ne obrade svi karakteristični trenuci. Kao rezultat proračuna se ima vrednost minimalnih gubitaka aktivne električne energije kao i optimalna sekvenca kojom se prelazi iz inicijalnog topološkog stanja u stanje za koje se imaju minimalni gubici. Dodatno pored optimalne sekvence i njoj korespondentnih gubitaka aktivne električne energije moguć je uvid u predložene sekvence tokom iterativnog postupka kao i vrednosti gubitaka aktivne električne energije svakog proračuna gubitaka električne energije.

5. VERIFIKACIJA ALGORITAMA

Verifikacija algoritama Enel optimalne rekonfiguracije mreže izvršena je na test mreži. Mreža se napaja preko dva transformatora nominalnog prenosnog odnosa 110/20 kV/kV i sastoji se od 8, 20 kV izvoda i 9 NO rasklopnih uređaja. Na mreži postoje dva tipa potrošača (po jedan za obe napojna transformatora) sa različitim normalizovanim dnevnim P, Q krivama potrošnje. Oba tipa karakterišu 4 sezone kojima odgovaraju 4 godišnja doba i po 2 karakteristična dana u sezoni, radni dan i dan vikenda. U nastavku su razmatrana 2 primera izvršavanja algoritma.

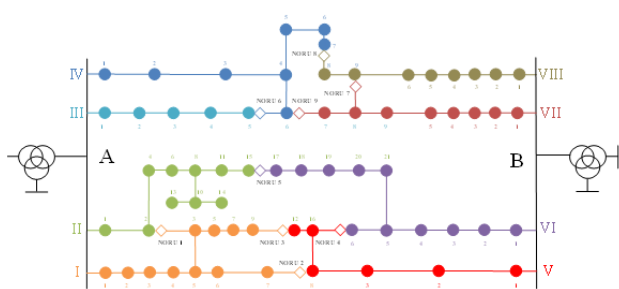


Slika 4 – Test mreža

5.1. Primer 1– Jednom godišnje

Proračun Enel optimalne rekonfiguracije mreže za period od godinu dana sa korakom iteracije od 1h. Prvi korak je izvršenje proračuna inicijalnih gubitaka električne energije na test mreži. Ukupni gubici aktivne električne energije za period od godinu dana iznose 24302420 kWh odnosno 4.08% ukupne godišnje injektirane aktivne

električne energije. Sledeći korak je izvršenje samog algoritma Enel optimalne rekonfiguracije mreže. Kako period od godinu dana obuhvata sve 4 sezone sa po 2 karakteristična dan, onda se sa korakom od 1h imaju ukupno 192 iteracije. Za 192 puštanja proračuna ORDM dobijene su ukupno 54 jedinstvene sekvence pa je izvršeno 54 proračuna gubitaka električne energije. Sekvenca nakon čije primene se ostvaruju najmanji gubici aktivne električne energije od 22566240 kWh, odnosno 3.80%, je dobijena za trenutak 03:00 h, radni dan, jesen. Sekvenca se sastoji od 8 parova NO/NZ rasklopnih uređaja. Ostvareni benefit primenom optimalne sekvence iznosi **1736180 kWh** što je **0.28%** godišnje injektirane aktivne električne energije.

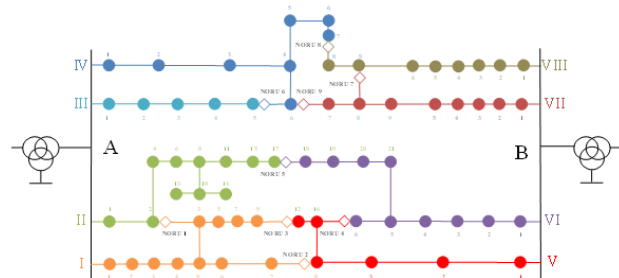


Slika 5 – Mreža nakon primene sekvence

5.2. Primer 2– Dva puta godišnje

Proračun Enel optimalne rekonfiguracije mreže 2 puta godišnje i to za period proleće-letno i period jesen-zima. Inicijalno topološko stanje i potrošnja su identični kao u prvom primeru. Period proleće-letno počinje 21. marta, a završava se zaključno sa 22. septembrom, dok period jesen-zima počinje 23 septembra, a završava se zaključno sa 20. martom naredne godine. Prvi se proračunavaju inicijalni gubici električne energije za period proleće-letno. Dobijeni ukupni gubici aktivne električne energije su 13690040 kWh odnosno 4.28%. Zatim se izvršava Enel optimalne rekonfiguracije mreža za period proleće-letno. Izabrani vremenski period obuhvata dve sezone pa se imaju ukupno 96 puštanja proračuna ORDM. Dobijeno je 40 jedinstvenih sekvenci i izvršeno je 40 proračuna gubitaka električne energije.

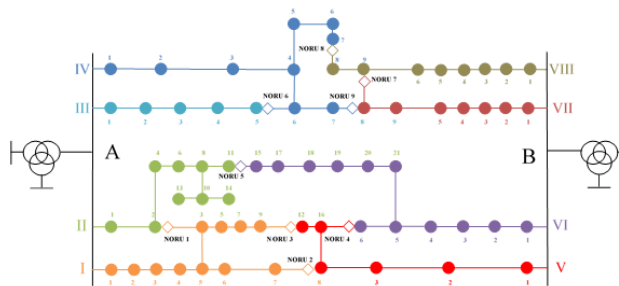
Optimalna sekvenca je dobijena za trenutak 03:00 h, radni dan, proleće i njenom primenom dobijeni su gubici od 12742850 kWh. Sekvenca se sastoji od 8 parova NO/NZ rasklopnih uređaja.



Slika 6 – Mreža nakon primene sekvence (proleće-letno)

Ponovo se pušta inicijalni proračun gubitaka električne energije za period jesen-zima i novo topološko stanje. Dobijena vrednosti gubitaka aktivne električne energije je 9827190 kWh. Potom se izvršava algoritma Enel opti-

malne rekonfiguracije mreže za period jesen-zima. Sada se za 96 puštanja ORDM ima jedinstvenih sekvenci, odnosno 12 proračuna gubitaka električne energije. Optimalna sekvenca se javlja za karakteristični trenutak 09:00 h, dan vikenda, jesen i njenom primenom dobijaju se gubici od 9813981 kWh. Sekvenca se sastoji od 2 para NO/NZ rasklopnih uređaja. Za oba perioda, imaju se ukupni gubici aktivne električne energije koji iznose 22556831 kWh (12742850+9813981). U odnosu na gubitke koji su dobijeni inicijalnim puštanjem proračuna gubitaka za period od godinu dana, 24302420 kWh, ostvaren je benefit od **1745589 kWh** što predstavlja **0.29%** godišnje injektirane aktivne električne energije.



Slika 7 – Mreža nakon primene sekvence (jesen-zima)

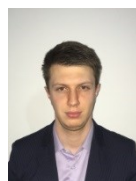
6. ZAKLJUČAK

Enel optimalna rekonfiguracija DM, kao rešenje problema optimalne konfiguracije mreža spram vrednosti gubitaka aktivne električne energije je ostvarila svoj cilj. Metoda je uvek pronalazila sekvencu kojom se ostvaruje smanjenje gubitaka aktivne električne energije za selektovani vremenski period.

7. LITERATURA

- [1] D.Popović, D.Bekut, V.Dabić: *Specijalizovani DMS Algoritmi*; Prosveta, Novi Sad, 2011.
- [2] D.Shirmohammadi, H.W.Hong: Reconfiguration of Electric Distribution Networks for Resistive Line Losses Reduction, *IEEE Transactions on Power Delivery*, Vol. 4, No. 2, April 1989, pp 1492-1498
- [3] Y.Al-Mahrogi, I.A.Metwally, A.Al-Hinai, A.Al-Badi: Reduction of Power Losses in Distribution Systems; *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, Vol:6, No:3, 2012.
- [4] M.F.Sulaima, M.F.Mohamad, M.H.Jali, W.M.Bukhari, M.F.Baharom: A Comparative Study of Optimization Methods for 33kV Distribution Network Feeder Reconfiguration; *International Journal of Applied Engineering Research*, Vol. 6, No. 4, 2014, pp. 1169-1182
- [5] V.C.Strezoski: *Osnovi Elektroenergetike*; Fakultet tehničkih nauka u Novom Sadu, Novi Sad, 2014.
- [6] A.Charlansut, N.Rugthaicharoencheep, S.Auchariyament: Heuristic Optimization Techniques for Network Reconfiguration in Distribution System; *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, Vol. 6, No. 4, 2012, pp. 365-368

Kratka biografija:



Vladan Rakić rođen je u Užicu 03.09.1992. god. Srednju školu, završio je 2011. god. u Šapcu. Diplomski rad na Fakultetu tehničkih nauka i oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranio je 2017. god. Ist godine upisao se na master akademske studije.

RAZVOJ JEZIKA ZA OPIS KOMANDNIH KORISNIČKIH INTERFEJSA DEVELOPMENT OF A LANGUAGE FOR SPECIFICATION OF COMMAND USER INTERFACES

Željko Bal, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Ovaj rad opisuje implementaciju jezika specifičnog za domen opisivanja korisničkih interfejsa, pomoću koga je moguće izgenerisati kod za interfejs komandne linije, kao i za grafički korisnički interfejs određene aplikacije.

Ključne reči: Jezik specifičan za domen, JSD, textX, CLI, GUI

Abstract – This thesis describes the implementation of a domain specific language for describing user interfaces, that can be used to generate the code for a command line interface, as well as a graphical user interface for a certain application.

Keywords: Domain specific language, DSL, textX, CLI, GUI

1. UVOD

U ovom radu je opisan jezik specifičan za domen opisivanja komandnih korisničkih interfejsa nazvan „Command Interface Description Language” (u daljem tekstu skraćeno CID). Glavni pojmovi koji čine meta-model jezika su komande i njihovi parametri, čime se opisuje hijerarhijska struktura komandi određenog korisničkog interfejsa. Na osnovu ovog opisa CID alat može da generiše korisnički interfejs komandne linije, kao i grafički korisnički interfejs, koji se oslanja na izgenerisani ili već postojeći interfejs komandne linije. Na ovaj način se omogućava lako kreiranje grafičkog interfejsa i interfejsa komandne linije za određenu aplikaciju na osnovu jedinstvene specifikacije, što ubrzava razvoj i olakšava održavanje oba interfejsa. Takođe je moguće na lak način opisati već postojeći interfejs komandne linije neke aplikacije i izgenerisati grafički interfejs koji se oslanja na postojeći korisnički interfejs.

Gramatika CID jezika je opisana u TextX [1] meta-jeziku za opisivanje jezika specifičnih za domen. Na osnovu ovog opisa je generisan parser za sam CID jezik. Generatori za korisničke interfejse u okviru CID alata su implementirani u programskom jeziku Python, dok je za samo generisanje koda interfejsa korišćen šablon-generator Jinja 2 [2]. Korisnički interfejs komandne linije koji se generiše pomoću CID alata je implementiran u programskom jeziku Python, dok je grafički interfejs implementiran kao Electron [3] desktop aplikacija. Sam CID alat poseduje grafički interfejs i interfejs komandne linije koji su izgenerisani na osnovu opisa u CID jeziku.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Igor Dejanović, vanr. prof.

2. TEORIJSKE OSNOVE

2.1 Jezici Specifični za Domen

Jezici specifični za domen (eng. *Domain Specific Language* - DSL) su računarski jezici koji su specijalizovani za određenu oblast primene (domen), za razliku od jezika opšte namene koji su široko primenjivi u različitim oblastima [4]. Jezici specifični za domen barataju pojmovima iz određenog usko definisanog domena i na taj način omogućavaju da se rešenja za određenu klasu problema iskažu jasnije i konciznije nego pomoću jezika opšte namene. Takođe, na isti način, omogućavaju stručnjacima iz date oblasti da lakše razumeju rešenje ili čak učestvuju u izradi samog rešenja.

2.2 Korisnički Interfejs

Korisnički interfejs je tačka interakcije između računara i ljudi (korisnika). Korisnički interfejsi podrazumevaju bilo koju vrstu interakcije (kao što su slika, zvuk, pozicija, pokret i druge...) prilikom koje se prenose informacije između korisnika i računarskog sistema [5]. Cilj korisničkih interfejsa je da omoguće lak, efikasan i prijatan način za upotrebu sistema koji proizvodi željeni rezultat [6]. U ovom radu će posebno biti razmatrana dva, trenutno najčešće korišćena, tipa korisničkih interfejsa na personalnim računarima: interfejs komandne linije (eng. *Command Line Interface* - CLI) i grafički korisnički interfejs (eng. *Graphical User Interface* - GUI).

2.3 Interfejs Komandne Linije

Interfejs komandne linije je način interakcije sa računarskim programom pri kom korisnik zadaje komande programu u vidu sukcesivnih redova teksta (komandnih linija) [7]. Interfejsi komandne linije omogućuju brže i konciznije zadavanje komandi od grafičkih korisničkih interfejsa i takođe omogućuju lakšu automatizaciju zadavanja komandi, zbog čega ih često preferiraju stručnjaci u oblasti računarstva.

2.4 Grafički Korisnički Interfejs

Grafički korisnički interfejs omogućuje korisniku interakciju sa računarskim programom putem “direktne manipulacije” grafičkim elementima [8]. Prosečni korisnici računara često vide grafički korisnički interfejs kao lakši za učenje i upotrebu od interfejsa komandne linije zbog vizuelnih metafora koje omogućavaju korisniku da poveže elemente interfejsa sa svojim mentalnim modelima. Iz istog razloga korisnici koji su navikli na upotrebu grafičkih korisničkih interfejsa često oklevaju da koriste aplikacije koje poseduju isključivo interfejs komandne linije.

3. KORIŠĆENE TEHNOLOGIJE

3.1 TextX

TextX [1] je meta jezik (jezik za opisivanje jezika) za specifikaciju jezika specifičnih za domen, napisan u programskom jeziku Python. Pomoću ovog meta jezika je moguće na lak način opisati gramatiku određenog tekstualnog jezika. Opis gramatike se sastoji od niza produkcionih pravila koja ujedno opisuju koncepte (apstraktnu sintaksu) i konkretnu sintaksu jezika. Na osnovu gramatike opisane na ovaj način TextX alat dinamički gradi meta-model i parser za opisani jezik.

Rezultat parsiranja dobijenim parserom je objektni model koji odgovara meta-modelu jezika i predstavlja apstraktno sintakšno stablo (graf) parsiranog teksta. Ovaj model je sačinjen od Python objekata, sa dinamički kreiranim klasama koje predstavljaju koncepte jezika opisane gramatikom. TextX projekat je inspirisan Xtext [9] alatom za izgradnju jezika specifičnih za domen, implementiranom u programskom jeziku Java i baziranom na Eclipse platformi [10].

TextX se u ovom projektu koristi za opisivanje gramatike samog CID jezika i parsiranje specifikacije korisničkih interfejsa, kao i za inicijalno pre-procesiranje dobijenog modela. Na ovaj način se smanjuje kompleksnost celog sistema korišćenjem gotovog rešenja za generisanje parsera, a ujedno se dobija i formalna definicija gramatike CID jezika.

3.2 Jinja

Jinja [2] je šablon-generator (eng. *template engine*) implementiran u programskom jeziku Python. Ovaj alat omogućava definisanje tekstualnih šablona i generisanje tekstualnog sadržaja na osnovu definisanih šablona i određenih ulaznih podataka. Definicija šablona sadrži statičku strukturu teksta koji se generiše, zajedno sa rezervisanim poljima na određenim mestima u tekstu, koje predstavljaju dinamički deo šablona i koje šablon-generator popunjava sa odgovarajućim prosleđenim ulaznim podacima prilikom generisanja.

U ovom projektu, Jinja se koristi za generisanje koda korisničkih interfejsa. Definisani su šabloni za Python objektni model koji koristi apstraktni parser argumenata komandne linije, kao i za HTML kôd i JavaScript objektni model koji se koristi za grafički korisnički interfejs.

3.3 Electron

Electron [3] je platformski nezavisan (eng. *cross platform*) programski okvir koji omogućava izgradnju desktop aplikacija korišćenjem *web* tehnologija (HTML, CSS i JavaScript). Electron se oslanja na Node.js okvir za *back-end*, pomoću koga se izvršava poslovna logika i komunikacija sa operativnim sistemom, implementirana u JavaScript jeziku.

Za *front-end*, Electron se oslanja na Chromium projekat [11] uz pomoć koga prikazuje grafički interfejs implementiran pomoću *web* tehnologija u desktop okruženju. Na ovaj način se postiže velika fleksibilnost pri kreiranju grafičkog interfejsa i mogućnost upotrebe širokog spektra *web* tehnologija, dok sa druge strane prikazani grafički interfejs ima isti izgled na svim podržanim platformama (Windows, Linux i OSX).

Electron je u ovom projektu korišćen za implementaciju generisanog grafičkog korisničkog interfejsa.

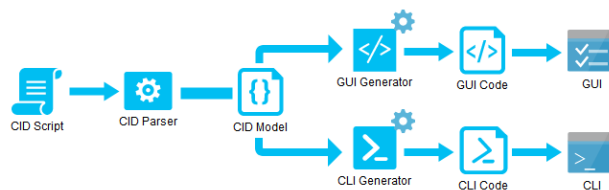
4. SPECIFIKACIJA I IMPLEMENTACIJA

4.1 Ciljevi

Glavni cilj sistema opisanog u ovom radu je da omogućiti lak i brz razvoj korisničkih interfejsa za aplikacije koje bi mogle imati koristi od dobrih osobina grafičkih korisničkih interfejsa i interfejsa komandne linije istovremeno. S obzirom na to da razvoj oba ova tipa korisničkih interfejsa za jednu aplikaciju uglavnom zahteva previše resursa, najčešće aplikacije poseduju jedan od ova dva tipa interfejsa, zavisno od toga koji od njih nudi benefite koji su bitniji za samu aplikaciju. Kako bi se prevazišao ovaj problem, u okviru ovog rada je razvijen sistem pomoću koga se mogu generisati ujedno interfejs komandne linije i grafički korisnički interfejs za određenu aplikaciju na osnovu jedinstvene specifikacije. Predviđena su dva slučaja korišćenja. U prvom je potrebno generisati grafički korisnički interfejs i interfejs komandne linije za novoimplementiranu aplikaciju (ili pak aplikaciju koja tek treba da bude implementirana). Ova dva izgenerisana korisnička interfejsa su kompatibilni jedan sa drugim u smislu da se grafički interfejs oslanja na interfejs komandne linije prilikom izvršavanja komande. U drugom slučaju korišćenja, već postoji implementiran interfejs komandne linije, za koji je potrebno generisati grafički interfejs. U ovom slučaju se opisuje već postojeći interfejs komandne linije i generiše se samo grafički interfejs koji se oslanja na postojeći interfejs aplikacije pri izvršavanju.

4.2 Arhitektura

CID jezik je implementiran kao eksterni jezik specifičan za domen. Osnovna komponenta sistema je parser CID jezika koji na osnovu definisane gramatike parsira specifikaciju koja opisuje korisnički interfejs. Parser vrši osnovnu validaciju specifikacije i kao rezultat vraća model interfejsa kao objektnu strukturu. Ovaj model se potom prosleđuje generatoru interfejsa komandne linije i/ili generatoru grafičkog korisničkog interfejsa. Generatori vrše dodatne validacije i transformacije modela specifične za tip korisničkog interfejsa i potom na osnovu modela generišu kôd za odgovarajuće korisničke interfejse. Na slici 1 je šematski prikazana arhitektura sistema:



Slika 1. Šematski prikaz arhitekture CID sistema

4.3 CID Parser

CID Parser kao ulaz ima jednu ili više datoteka koje sadrže opis korisničkog interfejsa u CID jeziku, dok kao rezultat vraća objektni model datog interfejsa, koji dalje može biti korišćen kao ulaz generatorima koda samih korisničkih interfejsa. *CID Parser* se, za samo parsiranje CID jezika i formiranje stabla apstraktne sintakse, oslanja na dinamički generisan parser na osnovu definisane gramatike jezika, pomoću TextX alata opisanog u odeljku 3.1. Nakon toga se stablo apstraktne sintakse validira i transformiše u model koji na ulazu očekuju generatori koda. Takođe, uloga *CID Parser* komponente je i da razreši reference na parametre i komande definisane

unutar iste ili odvojene datoteke. Objekti koji predstavljaju reference na parametre ili komande u stablu apstraktnih sintakse bivaju zamenjeni instancama koje opisuju referencirane elemente. Samim tim, *CID Parser* parsira i kombinuje sve referencirane datoteke i kreira objektni model sa strukturom grafa.

4.4 CLI Generator

CLI Generator modul predstavlja generator koda interfejsa komandne linije. Ulaz u *CLI Generator* je model korsničkog interfejsa dobijen iz *CID Parser-a*, dok je izlaz kôd interfejsa komandne linije i primer koda koji implementira logiku aplikacije, a koji se poziva od strane izgenerisanog koda. Pre samog generisanja, *CLI Generator* dodatno procesira i validira model primenjujući transformacije i validacije koje su specifične za interfejs komandne linije. Nakon toga generator kopira statički kôd, koji uključuje generički parser argumenata komandne linije, koji parsira argumente na osnovu modela interfejsa komandne linije. Nakon kopiranja statičkog koda, generator generiše objektni model korisničkog interfejsa za generički parser argumenata. Generisanje koda se vrši pomoću Jinja šablon-generatora opisanog u odeljku 3.2. Prilikom ponovnog generisanja, generator ne menja deo koda koji implementira logiku same aplikacije, tako da korisnik može slobodno menjati ručno pisani kôd.

4.5 Generički CLI Parser

Generički CLI parser predstavlja parser argumenata komandne linije, koji na osnovu modela interfejsa komandne linije prepoznaje kojim komandama i parametrima pripadaju parsirani argumenti i kao rezultat daje objektni model datih argumenata. U okviru modela generičkog parsera su opisane komande korisničkog interfejsa i njihovi parametri.

Pre samog parsiranja argumenata, proverava se prisutnost argumenata koji iniciraju prikaz ugrađene korisničke pomoći definisanih za svaku komandu u okviru CID modela. Ukoliko se pronađe jedan od ovih argumenata, prikazuje se definisana korisnička pomoć za odgovarajuću komandu (prikazana na slici 2) i ostali argumenti se ignorišu.

Parsiranje argumenata komandne linije funkcioniše tako što se prvo argumenti klasifikuju po komandama kojima pripadaju i dalje na prepoznate ne-pozicione i preostale pozicione parametre. Zatim se za klasifikovane argumente, na osnovu modela, prepoznaju šabloni korišćenja za svaku od komandi i pri tom se prepoznaju i pozicioni parametri. Uz to se takođe proverava: da li su zadovoljeni šabloni korišćenja, da li postoje ne prepoznati argumenti ili argumenti koji su prepoznati, ali ne pripadaju prepoznatom šablonu korišćenja, kao i da prisustvo potkomande odgovara prepoznatom šablonu korišćenja. Potom se za vrednosti parametara koje nisu prisutne dodaju podrazumevane vrednosti definisane u modelu samih parametara i nakon toga se vrednosti svih parametara konvertuju u odgovarajuće Python tipove, u skladu sa definisanim tipovima podataka. Na posletku se proveravaju definisana ograničenja parametara i komandi i kao rezultat se vraća objektni model prepoznatih vrednosti parametara.

```
cid_generator --help -a
Generate a graphical and/or command line interface from a CID script.

Usage: cid_generator <CID_FILE> <ROOT_COMMAND> <DEST_PATH> <sub_command>

Parameters:
  <ROOT_COMMAND> The name of the entry point command of the interface.
  <CID_FILE>       The CID script file that contains a description of the command
                  interface to generate. (-cid)
  <DEST_PATH>      The destination directory where to generate the interface.
  -h, --help      Shows a shorter help message. For this help message type:
                  cid_generator -h -a. All detailed help parameters: -a, --all.

Sub Commands:
  generate_cli     Generate a command line parser based on the provided description.
  generate_gui     Generate a graphical user interface based on the provided
                  description.
  generate_both    Generate both GUI and CLI.

For help about a specific sub command type: cid_generator <sub_command> -h.

CID (Command Interface Description) is a language for describing command interfaces (such as this one), that can be used
to generate graphical or command line interfaces. For more information see https://github.com/sajko-bal/CID.
```

Slika 2: Prikaz korisničke pomoći za *cid_generator* komandu

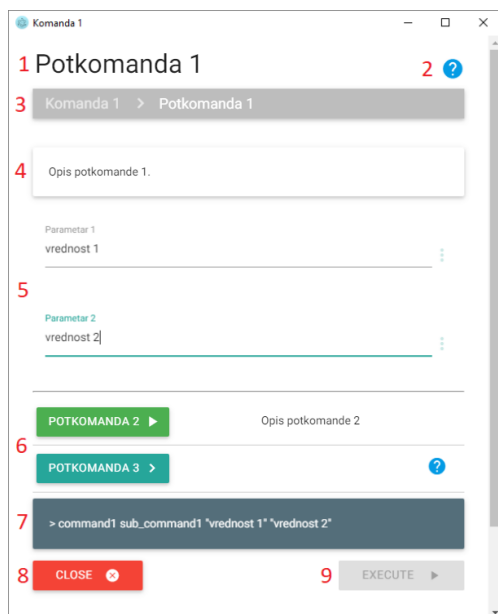
4.6 GUI Generator

GUI Generator modul predstavlja generator koda grafičkog korisničkog interfejsa. Proces generisanja koda je sličan gore opisanom procesu generisanja koda za interfejs komandne linije. Ulaz u *GUI Generator* je model korsničkog interfejsa dobijen iz *CID Parser-a*, dok je izlaz kôd grafičkog korisničkog interfejsa, koji se oslanja na interfejs komandne linije opisanog komandnog interfejsa. Pre samog generisanja, *GUI Generator* dodatno procesira i validira model primenjujući transformacije i validacije koje su specifične za grafički korisnički interfejs. Nakon toga generator kopira statički kôd, koji uključuje delove aplikacije grafičkog korisničkog interfejsa sa generičkim grafičkim komponentama i kodom za validaciju argumenata i generisanje sadržaja komandne linije. Nakon kopiranja statičkog koda, generator generiše *front-end* kôd kompozicijom i konfiguracijom predefinisanih grafičkih komponenti, kao i objektni model korisničkog interfejsa, koji se koristi za validaciju argumenata i generisanje sadržaja komandne linije na osnovu unetih vrednosti. Generisanje koda se, kao i kod *CLI Generator-a*, vrši pomoću šablon-generatora.

4.7 GUI programski okvir

Programski okvir grafičkog korisničkog interfejsa je implementiran kao Electron desktop aplikacija. Aplikacija sadrži jedan prozor (prikazan na slici 3) koji prikazuje jednu ili više ulaznih formi. Svaka forma predstavlja jednu od komandi opisanog korisničkog interfejsa i sadrži ulazne elemente koji odgovaraju parametrima date komande, kao i dugmad koja vode ka formama odgovarajućih potkomandi. Aplikacija omogućuje prikaz korisničke pomoći i opisa komandi i parametara, kao i validaciju korisničkog ulaza. Pored validacije pojedinačnih parametara i komandi, validiraju se i definisani šabloni korišćenja parametara. Na posletku, aplikacija omogućava generisanje sadržaja komandne linije za opisani komandni interfejs na osnovu korisničkog unosa i slanje generisanog sadržaja na standardni izlaz. Ovo dalje omogućava izvršavanje date komande sa odgovarajućim argumentima u interpreteru komandne linije.

Validacija ulaza i generisanje sadržaja komandne linije se vrše na generičan način, pomoću izgenerisanog JavaScript objektnog modela, koji je definisan u okviru grafičkog programskog okvira. Ovaj model opisuje strukturu korisničkog interfejsa gde su glavni koncepti, kao i kod CID modela, komande i parametri.



Slika 3: Primer prozora grafičkog interfejsa: 1. naslov komande, 2. dugme za korisničku pomoć, 3. navigaciona traka nad-komandi, 4. opis komande, 5. parametri, 6. potkomande, 7. izgenerisani sadržaj komandne linije, 8. dugme za zatvaranje, 9. dugme za izvršavanje komande

5. ZAKLJUČAK

Razvijeni sistem omogućava generisanje grafičkih korisničkih interfejsa i interfejsa komandne linije na osnovu jedinstvene specifikacije u jeziku specifičnom za domen. Jezik je dovoljno koncizan da omogućava brzo i jednostavno opisivanje komandi i parametara korisničkog interfejsa. Domen jezika je ograničen na generisanje komandnih interfejsa, što omogućava da pojmovi jezika budu dovoljno apstraktni da istovremeno opisuju elemente oba tipa korisničkih interfejsa. Na ovaj način se izbegava dupliranje specifikacije i time skraćuje vreme implementacije i olakšava održavanje oba tipa korisničkih interfejsa. Sa druge strane jezik nudi fleksibilnost u opisivanju izgleda i strukture tekstualnog sadržaja komandne linije čime se omogućava opis velikog broja postojećih aplikacija sa interfejsom komandne linije i generisanje grafičkog interfejsa za iste. Takođe, jezik nudi fleksibilnost i u opisivanju strukture i rasporeda grafičkih elemenata na ulaznoj formi grafičkog korisničkog interfejsa. Pored pomenute fleksibilnosti, generatori koda imaju mogućnost određivanja smislenih podrazumevanih vrednosti za delove specifikacije koji nisu navedeni i na taj način omogućavaju korisniku da navede minimalan opis korisničkog interfejsa na osnovu koga se može izgenerisati funkcionalan kôd korisničkih interfejsa.

Takođe, pored izbegavanja dupliranja specifikacije elemenata korisničkog interfejsa, razvijeni sistem olakšava razvoj pojedinačnih korisničkih interfejsa tako što omogućava njihov opis na višem nivou apstrakcije, na osnovu čega se generiše kôd tih interfejsa, koji inkorporira dobre prakse, dizajn šablone i standarde u oblasti dizajna korisničkih interfejsa.

Dva glavna slučaja korišćenja implementiranog sistema su generisanje grafičkog korisničkog interfejsa i interfejsa komandne linije za određenu aplikaciju i generisanje grafičkog korisničkog interfejsa za aplikaciju sa već postojećim interfejsom komandne linije. Sistem se može upotrebiti za razvoj korisničkih interfejsa za jednostavne alate koje je potrebno isporučiti klijentima koji preferiraju grafičke korisničke interfejse, a za koje takođe postoji potreba da budu automatizovani, tj. da se izvršavaju u okviru skripte za automatizaciju ili grupnu obradu.

Takođe sistem se može koristiti za razvoj grafičkih korisničkih interfejsa koji se oslanjaju na već postojeće interfejse komandne linije postojećih aplikacija, kako bi se upotreba tih aplikacija olakšala korisnicima koji iz različitih razloga izbegavaju upotrebu komandne linije. Kako je domen implementiranog sistema ograničen na generisanje jednostavnih komandnih korisničkih interfejsa, on se ne može koristiti za generisanje korisničkih interfejsa za aplikacije koje zahtevaju složenu interakciju sa korisnikom.

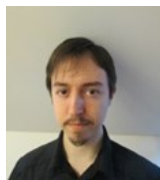
Neki od bitnih aspekata sistema kojima bi se trebala posvetiti pažnja u daljem razvoju su: testiranje, internacionalizacija generisanih korisničkih interfejsa, implementacija dodatnih ulaznih grafičkih elemenata i podrška za druge operativne sisteme (pored Microsoft Windows OS-a).

CID projekat je otvorenog koda, pod MIT licencom i dostupan na adresi: <https://github.com/zeljko-bal/CID>

6. LITERATURA

- [1] www.igordejanovic.net/textX/
 - [2] jinja.pocoo.org/
 - [3] electronjs.org/
 - [4] en.wikipedia.org/wiki/Domain-specific_language
 - [5] [en.wikipedia.org/wiki/Interface_\(computing\)](http://en.wikipedia.org/wiki/Interface_(computing))
 - [6] en.wikipedia.org/wiki/User_interface
 - [7] en.wikipedia.org/wiki/Command-line_interface
 - [8] en.wikipedia.org/wiki/Graphical_user_interface
 - [9] www.eclipse.org/Xtext/
 - [10] www.eclipse.org/
 - [11] www.chromium.org/Home
- (svim stranicama je pristupljeno u novembru 2018.)

Kratka biografija:



Željko Bal rođen je u Novom Sadu 1990. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Softversko inženjerstvo, odbranio je 2019.god. kontakt: zeljko.bal@gmail.com

VREMENSKO I LOKACIJSKO PROŠIRENJE KONTROLE PRISTUPA BAZIRANE NA ULOGAMA U SPRING SECURITY OKRUŽENJU**TEMPORAL AND LOCATION-BASED EXTENSION OF ROLE-BASED ACCESS CONTROL WITHIN SPRING SECURITY FRAMEWORK**

Jan Šulja, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je prikazana implementacija vremenskog i lokacijskog proširenja kontrole pristupa bazirane na ulogama u Spring Security okruženju. Implementacija je podeljena na dve celine, gde je prva celina generički modul za kontrolu pristupa, a druga, veb aplikacija koja taj modul koristi za kontrolu pristupa.

Ključne reči: Java, Spring, kontrola pristupa, autorizacija

Abstract – This paper presents the implementation of temporal and location-based extension of role-based access control within Spring Security framework. Implementation is divided into two units, where the first unit is a generic access control module, and the second unit is a web application that uses this module to control the access.

Keywords: Java, Spring, access control, authorization

1. UVOD

Vreme i lokacija su važni atributi koji mogu odrediti politiku kontrole pristupa u organizaciji. Brzi porast korišćenja uređaja koji imaju ugrađen GPS, senzora i druge mobilne opreme, doveo je do pojave značajnih istraživanja u oblasti kontrole pristupa na osnovu lokacije i vremena pristupa. Politika kontrole pristupa na osnovu lokacije proširuje bezbednost aplikacije tako što dozvoljava ili zabranjuje pristup resursu sa određenih lokacija. Sa druge strane, vremensko proširenje pruža mogućnost zabrane ili dozvole pristupa resursu u željenim vremenskim intervalima [1].

Autori rada [2] daju predlog modela koji proširuje RBAC prostornim i vremenskim elementima. Njihov predlog predstavlja model za izdavanje zahteva kontrole pristupa izuzetno mobilnih i dinamičnih okruženja kako bi obezbedili kontrolu pristupa zasnovanu na lokaciji i vremenu. U radu je predstavljen obiman prostorni i vremenski RBAC model. Pojam vremena je jedinstven nad svim entitetima RBAC modela (korisnici, uloge, dozvole) dok kontekst lokacije može biti različit za svaki od ovih entiteta.

Prema tome, korisnici koji se nalaze na nekoj lokaciji L1 mogu dobiti ulogu koja je omogućena u lokaciji L2 za dobijanje dozvola za operacije nad objektima koji se

nalaze na lokaciji L3, pri čemu lokacije L1, L2 i L3 ne moraju biti iste lokacije [2].

Još jedan predlog modela koji opisuje vremensko i prostorno proširenje kontrole pristupa na osnovu uloga predstavljen je u radu [1]. U radu je dat predlog prostorno vremenskog modela kontrole pristupa zasnovanog na ulogama (STRBAC – *Spatio Temporal Role-Based Access Control*), kao proširenje prvobitnog RBAC modela. STRBAC model podržava dinamičko aktiviranje uloga i dozvola u zavisnosti od navedenih ograničenja. Takođe, daje novi pravac za kategorizaciju logičke lokacije i vremena, a pogodan je za organizacije koje zahtevaju kontrolu pristupa na osnovu vremena i lokacije [1].

2. KONTROLA PRISTUPA

Pod pojmom kontrola pristupa, podrazumeva se dodela prava pristupa, kontrole ili upravljanja objektima od značaja zainteresovanom subjektu. Subjekt na osnovu precizno određenih pravila može dobiti pravo na vršenje određenih operacija nad traženim objektom ukoliko su zadovoljeni tačno definisani uslovi. Subjekt i objekat predstavljaju entitete u modelima kontrole pristupa, gde je subjekt entitet koji vrši operacije nad objektom, a objekat resurs sistema [3].

Način na koji se pravila kontrole pristupa definišu, opisuje se modelom kontrole pristupa, a neki od najzastupljenijih modela su:

- RBAC - role based access control [4]
- ABAC - attribute based access control [5]
- DAC - discretionary access control [6]
- MAC - mandatory access control [7]

2.1. Kontrola pristupa na osnovu uloga

Kod ove kontrole pristupa administratori sistema kreiraju uloge u skladu sa poslovnim pozicijama unutar kompanije ili organizacije, odobravaju dozvole za te uloge, i dodeljuju subjektima uloge na osnovu njihovih radnih obaveza i kvalifikacija. Uloga predstavlja sposobnost subjekta da obavlja specifičan zadatak. Kombinacija korisnika sa dozvolama dodeljenim putem uloga, se obično često menja vremenom. Sa druge strane, veze između dozvole i uloge su stabilnijeg karaktera, ređe se menjaju nego ljudi koji rade posao predstavljen tom ulogom. Zbog toga se prilikom administracije bezbednosti koristi uloga, a ne dozvola. Ukoliko su promene potrebne, korisnicima se mogu jednostavno dodeljivati ili brisati uloge. Slično tome kako se sistem u kompaniji razvija, ulogama mogu biti dodeljivane nove ili uklonjene neke

NAPOMENA:

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

stare dozvole [8]. Neke varijacije ove kontrole pristupa uključuju sposobnost uspostavljanja odnosa između uloga, kao i između korisnika i uloga.

Na primer, dve uloge mogu biti uzajamno isključive, istom korisniku nije dozvoljeno da ima dodeljene obe uloge. Uloge mogu takođe sticati i veze nasleđivanja, pri čemu jedna uloga nasleđuje dozvole dodeljene drugoj ulozi.

Veze između uloge i dozvole mogu biti a obično i jesu predefinisane, što olakšava dodelu uloga korisnicima.

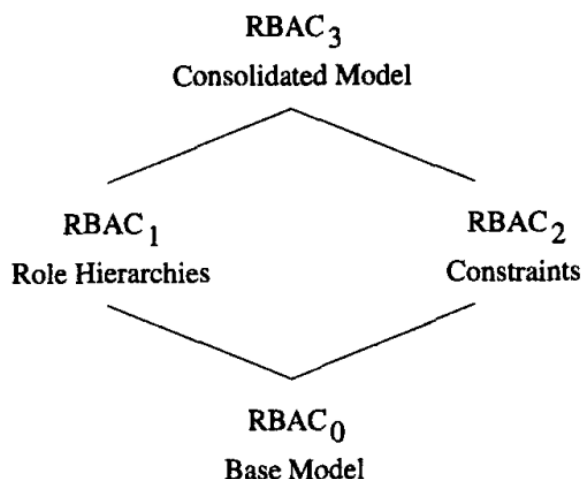
Politika kontrole pristupa na osnovu uloga obuhvaćena je vezama kao što su uloga-dozvola, korisnik-uloga i uloga-uloga.

Ove veze kolektivno određuju da li je korisniku dozvoljen pristup određenom delu sistema i mogu direktno biti konfigurisane od strane administratora sistema ili indirektno putem odgovarajućih uloga koje delegira administrator sistema. Politika koja se sprovodi u datom sistemu, rezultat je specifične konfiguracije ovih veza koje navodi administrator sistema.

Pošto politika kontrole pristupa može, a obično se i menja tokom životnog ciklusa sistema, kontrola pristupa na osnovu uloga pruža mogućnost modifikacije kontrole pristupa kako bi organizacione potrebe bile zadovoljene u svakom trenutku životnog ciklusa sistema [8]. RBAC modeli kontrole pristupa definisani su familijom od četiri modela (slika 1). $RBAC_0$ je osnovni model i predstavlja minimalni deo koji mora da bude zadovoljen od strane sistema, da bi se za taj sistem moglo reći da koristi RBAC model za kontrolu pristupa.

Komponente $RBAC_0$ modela su: korisnici, uloge, dozvole i sesije. $RBAC_1$ i $RBAC_2$ uključuju osnovni RBAC model, ali mu dodaju mu neke nezavisne funkcionalnosti. $RBAC_1$ dodaje koncept hijerarhije uloga (uloge mogu naslediti dozvole iz drugih uloga). $RBAC_2$ dodaje ograničenja (na primer uzajamno isključive uloge).

Ujedinjeni model $RBAC_3$ uključuje $RBAC_1$, $RBAC_2$ i tranzitivno $RBAC_0$ [8].

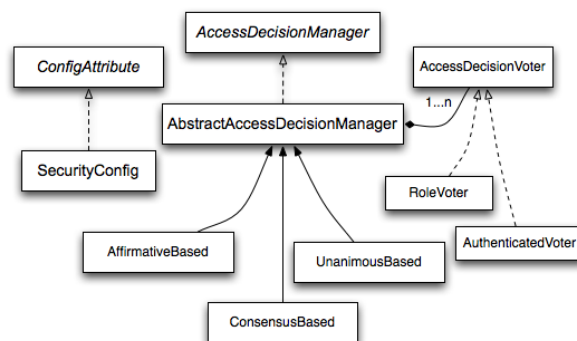


Slika 1. Veze RBAC modela

2.2. Kontrola pristupa u spring okruženju

U spring okruženju interfejs *AccessDecisionManager* je zadužen za kontrolu pristupa (slika 2). Utvrđivanje prava pristupa zasnovano je na glasanju, a postoje tri konkretne implementacije *AccessDecisionManager*-a:

- *AffirmativeBased* – pristup je dozvoljen ukoliko je bar jedan glasač dao glasao pozitivno
- *UnanimousBased* – pristup je dozvoljen ako su svi glasači bili uzdržani ili glasali pozitivno
- *ConsensusBased* - pristup je dozvoljen ako je većina glasača glasala pozitivno



Slika 2. Arhitektura autorizacije u Spring okruženju

Najčešće se koristi glasač tipa *RoleVoter*, kod kontrole pristupa na osnovu uloga. Ovaj glasač će glasati ukoliko postoji bar jedan konfiguracioni atribut koji počinje prefiksom „ROLE_“. Ako korisnik ima dodeljenu ulogu definisanu u konfiguracionom atributu, glasač će glasati pozitivno, u suprotnom glasač daje negativan glas. U slučaju da u konfiguracionim atributima ne postoji atribut koji počinje prefiksom „ROLE_“ glasač neće glasati, odnosno biće uzdržan [9].

3. MODEL SISTEMA

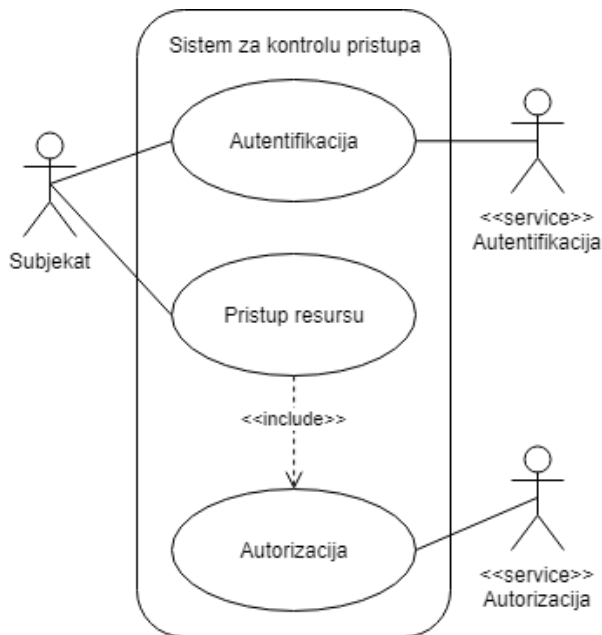
Projektovanje modula za kontrolu pristupa zasnovano je na ideji da se za svaki resurs u sistemu kome je kontrola pristupa potrebna definiše lista „brava“ koje je potrebno otključati tačno određenim „ključevima“. Pristup resursu se dozvoljava samo ako su sve brave na resursu uspešno otključane, u suprotnom zahtev za pristup traženom resursu se odbija. Brava je u modulu za kontrolu pristupa modelovana interfejsom *AccessLock*, dok ključ može biti atribut subjekta, atribut sistema, tip *AccessTicket* kao što je slučaj u veb aplikaciji ovog rada ili neke druge vrednosti.

Prilikom procesa odluke o dodeli prava pristupa resursu, svaki od *AccessLock* tipova definisanih za taj resurs prolazi kroz lanac odgovornosti, gde će ga odgovarajuća komponenta koja je zadužena za konkretan *AccessLock* obraditi. Ukoliko uslov potreban za dobijanje prava pristupa nije zadovoljen baca se izuzetak, u suprotnom izvršavanje procesa obrade zahteva se nastavlja dalje dok neka komponenta ne baci izuzetak ili proces obrade ne dođe do kraja lanca obrade. Ukoliko ni jedna komponenta ne baci izuzetak u ovom procesu obrade pristup resursu će biti uspešno odobren.

3.1 Učesnici u sistemu

U sistemu postoje tri učesnika, subjekat, servis za autentifikaciju i servis za autorizaciju. Na slici 3 prikazan je dijagram slučajeva korišćenja za pristup resursima. Ukoliko se pristupa javnom resursu koji je dostupan svima, servisi za autentifikaciju i autorizaciju se neće koristiti. Ako je resurs kome se pristupa zaštićen, onda subjekat pre pristupa resursu obavlja autentifikaciju a nakon uspešne autentifikacije sa token-om koji mu je

dodeljen može da zatraži pristup resursu. Pre dobijanja dozvole pristupa, servis za autorizaciju proverava da li subjekat ima prava da dobije pristup traženom resursu, ako je odgovor da, subjekat dobija pravo pristupa u suprotnom pristup je zabranjen.



Slika 3. Dijagram slučajeve korišćenja pristupa resursu

3.2. Model podataka

Osnovna ideja modela podataka je podržavanje više različitih vrsta privilegija za kontrolu pristupa, uz jednostavno dodavanje novih tipova privilegija kontrole pristupa. Jedan od ciljeva ovog modela je i da se omogući proizvoljno modelovanje kompleksnih hijerarhija privilegija kontrole pristupa, tako da može na najbolji mogući način da zadovolji potrebe što većeg broja različitih aplikacija, čime se dobija i mogućnost proširenja kontrole pristupa na osnovu uloga vremenskim i prostornim komponentama što je i zadatak ovog rada. Pored pomenutih komponenti, vrlo je jednostavno proširiti sistem drugim vrstama privilegija za kontrolu pristupa.

4. IMPLEMENTACIJA SISTEMA

Kada govorimo o tehnologijama koje su korišćene u ovom radu, moramo napomenuti da aplikaciju možemo podeliti na dve celine, prva celina predstavlja generičko rešenje koje omogućava kreiranje različitih tipova kontrole pristupa, dok druga celina predstavlja samo korišćenje alata koje nudi prva celina. Nakon što smo to ustanovili možemo zasebno opisati tehnologije koje su korišćene u ove dve celine.

Prva celina koja nudi generičko rešenje kontrole pristupa je implementirana koristeći Java [10] programski jezik, stoga može biti korišćena u bilo kojoj drugoj aplikaciji koja se bazira na Java programskom jeziku, bilo da je to veb bazirana aplikacija, Desktop aplikacija, mobilna aplikacija ili aplikacija nekog drugog tipa.

Druga celina predstavlja rešenje koje demonstrira primenu različitih tipova kontrole pristupa uz mogućnost proširenja baznih implementacija kontrole pristupa i implementirano je u Java programskom jeziku koristeći Spring okruženje [11].

Komponente zadužene za odlučivanje o dodeli prava pristupa date su kroz projektni obrazac Lanac odgovornosti, sa osnovnom klasom *AuthorizationHandler* (listing 1) koju nasleđuju sve konkretne komponente u lancu odgovornosti. Klasa *AuthorizationHandler* sadrži jedan atribut *successor* istog tipa koji predstavlja referencu na sledeću komponentu u lancu, dve apstraktne metode i jednu metodu sa implementacijom.

Prva metoda *doAuthorization(AccessLock)* zadužena je za obradu *AccessLock-a*, odnosno odlučivanju o dodeli ili zabrani pristupa resursu koji je zaštićen određenim *AccessLock* tipom.

Druga metoda *getResponsibility()* ima za ulogu da vrati konkretni tip *AccessLock-a* za koji je ta konkretna komponenta zadužena.

Treća metoda *handleRequest(AccessLock)* odlučuje na osnovu odgovornosti *AccessLock-a* da li je trenutna komponenta u lancu odgovorna za taj *AccessLock* pozivom prve metode *getResponsibility()*, ako jeste, poziva dalje drugu metodu *doAuthorization(AccessLock)* i dalje poziva *handleRequest(AccessLock)* nad sledećom komponentom u lancu, ovim se obezbeđuje obrada jednog konkretnog *AccessLock* tipa od strane više komponenti u lancu, a ako nije odgovorna u tom slučaju se samo poziva *handleRequest(AccessLock)* nad sledećom komponentom u lancu.

Metoda *doAuthorization(AccessLock)* odlučuje da li će pristup biti dozvoljen ili zabranjen, ukoliko dođe do zabrane ta metoda baca izuzetak.

```

public abstract class AuthorizationHandler {

    protected AuthorizationHandler successor;

    protected abstract void doAuthorization(AccessLock lock) throws Exception;
    protected abstract Class<? extends AccessLock> getResponsibility();

    public void setSuccessor(AuthorizationHandler successor) {
        this.successor = successor;
    }

    public void handleRequest(AccessLock lock) throws Exception {

        if(lock.getClass().equals(this.getResponsibility())) {
            this.doAuthorization(lock);
        }

        if(nonNull(successor)) {
            successor.handleRequest(lock);
        }
    }
}

```

Listing 1 Klasa *AuthorizationHandler*

Kako bi se omogućilo korišćenje modula za kontrolu pristupa registruje se novi filter koji će automatski biti dodat posle Spring security filter chain-a. Za donošenje odluke o kontroli pristupa koristi se *AuthorizationService* pozivom metode *authorize*. Ukoliko ova metoda baci izuzetak, kao odgovor se vraća poruka o grešci sa HTTP status kodom 401.

Servis klasa *AuthorizationService* je zadužena za konfigurisanje modula za kontrolu pristupa. Konfigurisanje se sastoji iz konstrukcije lanca odgovornosti i delegiranja zahteva za obradu u lanac odgovornosti.

Klasa *SimpleAuthorizationService* predstavlja jednu od mogućih implementacija ovog servisa (listing 2).

Korišćenjem Spring constructor dependency injection mehanizma u trenutku kada Spring okruženje instancira ovaj servis, u konstruktoru kreiramo lanac odgovornosti i inicijalizujemo atribut *securityConfig* koji sadrži konfiguraciju za kontrolu pristupa resursima.


```

@Service
public class SimpleAuthorizationService implements AuthorizationService {

    private SecurityConfig securityConfig;
    private AuthorizationHandler authorizationChain;

    public SimpleAuthorizationService(SecurityConfig securityConfig,
                                     List<AuthorizationHandler> links) {

        this.securityConfig = securityConfig;
        this.authorizationChain = links.stream()
            .reduce(null, (a,b) -> { b.setSuccessor(a); return b; });
    }

    @Override
    public void authorize(HttpServletRequest request) throws Exception {

        securityConfig.getLocks(request)
            .stream()
            .forEach(this::authorize);
    }

    @SneakyThrows
    private void authorize(AccessLock lock) {

        authorizationChain.handleRequest(lock);
    }
}

```

Listing 2. Implementacija servisa za autorizaciju

Nakon inicijalizacije lanca odgovornosti atribut *authorizationChain* predstavlja referencu prve komponente u lancu. Kao rezultat ovog načina inicijalizacije lanca odgovornosti, dobili smo vrlo jednostavno registrovanje novih komponenti u lanac odgovornosti. Za dodavanje nove autorizacione komponente nije potrebno menjati postojeću logiku za kreiranje lanca odgovornosti, potrebno je samo naslediti klasu *AuthorizationHandler* sa anotacijom *@Component* a prilikom sledećeg pokretanja aplikacije ova komponenta će automatski biti dodata u lanac odgovornosti, čime smo dobili vrlo jednostavno dodavanje novih autorizacionih komponenti u postojeći sistem.

Uloga javne metode *authorize* je da za resurs za koji se proverava kontrola pristupa preuzme iz *SecurityConfig* objekta sve *AccessLock* tipove koji definišu pravila o kontroli pristupa i da svaki od njih pošalje u lanac odgovornosti na kontrolu. Ova javna metoda se poziva iz filtera *AuthorizationFilter* prilikom svakog primljenog HTTP zahteva.

Za konfiguraciju kontrole pristupa koristi se klasa *SecurityConfig* a način konfigurisanja kontrole pristupa je sličan kao u Spring okruženju. U prvom delu koda (*listing 3*) data je konfiguracija *SecurityConfig* klase, koja predstavlja deo rešenja ove aplikacije, dok drugi deo prikazuje konfiguraciju Spring Security modula, dakle dozvoljena je kombinacija korišćenja obe implementacije kontrole pristupa.

```

securityConfig
    .antMatcher("/access_control_test/test3")
    .access(rbac("ADMIN"))
    .antMatcher("/access_control_test/test4")
    .access(rbac("USER"))
    .antMatcher("/access_control_test/test5")
    .access(ibac("178.220.0.0-178.223.255.255"))
    .antMatcher("/access_control_test/test6")
    .access(tbac())
    .antMatcher("/access_control_test/test7")
    .access(ibac("178.220.0.0-178.223.255.255"), tbac());

http.authorizeRequests()
    .antMatchers("/access_control_test/test2")
    .authenticated();

```

Listing 3. Konfiguracija kontrole pristupa

5. ZAKLJUČAK

U radu je prikazana implementacija vremenskog i lokacijskog proširenja kontrole pristupa bazirane na ulogama. Pored glavnog zadatka čiju je implementaciju pratio ovaj rad prikazano je generičko rešenje modula za kontrolu pristupa koji je omogućio jednostavnu implementaciju ovog tipa kontrole pristupa i otvorilo

vrata za implementaciju bilo koje druge vrste kontrole pristupa. Pored ovih implementacija dat je prikaz konceptualnog modela i dijagrama klasa ovih rešenja, kao i demonstracija upotrebe ovih aplikacija kroz interaktivni korisnički interfejs. Prikazan je i primer jednostavnog proširenja modula za kontrolu pristupa, slično tom primeru ovaj modul se može proširiti bilo kojom novom vrstom kontrole pristupa.

6. LITERATURA

- [1] Kumar, Mahendra / Newman, Richard E. *STRBAC - AN APPROACH TOWARDS SPATIO-TEMPORAL ROLE-BASED*. Gainesville, FL 32608 : CISE Department Gainesville, FL 32608, 2006.
- [2] *LoT-RBAC: A Location and Time-Based RBAC Model*. Chandran, Suroop Mohan / Joshi, J. B. D. Department of Information Sciences and Telecommunications, University of Pittsburgh : s.n., 2005.
- [3] Access control. *Wikipedia*. [Na mreži] https://en.wikipedia.org/wiki/Access_control.
- [4] *Role based access control*. [Na mreži] https://en.wikipedia.org/wiki/Role-based_access_control.
- [5] *Attribute based access control*. [Na mreži] https://en.wikipedia.org/wiki/Attribute-based_access_control.
- [6] *Discretionary access control*. [Na mreži] https://en.wikipedia.org/wiki/Discretionary_access_control.
- [7] *Mandatory access control*. [Na mreži] https://en.wikipedia.org/wiki/Mandatory_access_control.
- [8] Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. Role-based access control models. [Na mreži] 1996. <https://ieeexplore.ieee.org/abstract/document/485845>.
- [9] Authorization. *Spring*. [Na mreži] <https://docs.spring.io/spring-security/site/docs/current/reference/html/authorization.html>.
- [10] *Java*. [Na mreži] <https://docs.oracle.com/javase/8/docs/api/>.
- [11] *Spring*. [Na mreži] <https://spring.io/>.

Kratka biografija:



Jan Šulja je rođen 17.12.1993 godine u Zrenjaninu. Osnovnu školu "Maršal Tito" u Padini je završio 2008. godine. Elektrotehničku i građevinsku školu „Nikola Tesla“ u Zrenjaninu završio je 2012. godine. Osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu završio je 2016. godine na studijskom programu Računarstvo i automatika, smer Primjenjene računarske nauke i informatika. Godine 2016. upisuje master akademske studije na Fakultetu tehničkih nauka u Novom Sadu, na studijskom programu Računarstvo i automatika, smer Primjenjene računarske nauke i informatika, modul Elektronsko poslovanje.

MODELOVANJE I ISPITIVANJE BIOGASNE ELEKTRANE – PRIMER “PIK MORAVICA”**BIOGAS POWER PLANT MODELLING AND TESTING – CASE STUDY “PIK MORAVICA”**Igor Pušin, Vladimir Katić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ENERGETIKA, ELEKTRONIKA I TELEKOMUNIKACIJE**

Kratak sadržaj – U radu je razmatrana primena biomase kao obnovljivog izvora energije, Obradena je primena biogasa sa opisom biogasnog postrojenja „PIK Moravica“. Poređenje rezultata merenja izlazne struje napona i snage sa simulacijama i kriterijumima za priključenje malih elektrana su dati.

Ključne reči: Obnovljivi izvori, Biogas, Kvalitet električne energije.

Abstract – The paper discusses the usage of biomass as a renewable energy source. Use of biogas is treated and the description of the biomass plant "PIK Moravica" is presented. Comparison of the output current, voltage and power measurement results with simulations and criteria for connection of small power plants are given.

Keywords: Renewable energy, Biogas, Power quality

1. UVOD

Prirodni oblici energije se dele na neobnovljive (fosilna goriva, nuklearna goriva) i obnovljive (energija Sunca, vodotokova, vetra, mora i okeana, biomasa) izvore energije. Danas se teži primeni obnovljivih izvora i njihovoj transformaciji u električnu energiju iz razloga njihove sveprisutnosti, neiscrpnosti, usklađenosti sa prirodom (ne utiču na klimatske promene) i dr., a istovremeno zbog visokog kvaliteta i efikasnosti (preko 95%) korišćenja električne energije. Međutim, njih karakteriše promenljivost (na primer: brzina vetra ili intenzitet solarnog zračenja mogu se promeniti jako brzo), manja gustina snage, nemogućnost skladištenja (vetar, solar) ili visoka cena skladištenja (voda, biomasa), kompleksnost i cena transporta, viša cena izlazne energije, te nivo investicije za izgradnju energetskih postrojenja za transformisanje energije. Prema dosadašnjim procenama, najveći potencijal od obnovljivih izvora energije u Srbiji ima energija iz biomase i on se procenjuje na oko 60% ukupnog [1]. U biomasu spadaju biorazgradivi delovi nusproizvoda drvne industrije, otpaci i ostaci biološkog porekla iz poljoprivrede (biljnog i životinjskog porekla), kao i industrijskog i komunalnog otpada. U Vojvodini je dostupna poljoprivredna biomasa, koja se najčešće koristi za dobijanje biogasa [2].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Vladimir Katić, red. prof.

Pod biogasom se podrazumeva gas sa velikom količinom metana, koji nastaje fermentacijom organskih supstanci, kao što su stajnjak, kukuruzna silaža, slama i sl., otpaci od hrane, ili bilo koji drugi biorazgradivi materijali pri anaerobnim uslovima. U odnosu na fosilna goriva, prednost biomase (biogasa) je velika efikasnost transformacije u električnu i toplotnu energiju (preko 80%), znatno manja emisija štetnih gasova, a mogućnost daljeg korišćenja otpadnih materijala [3].

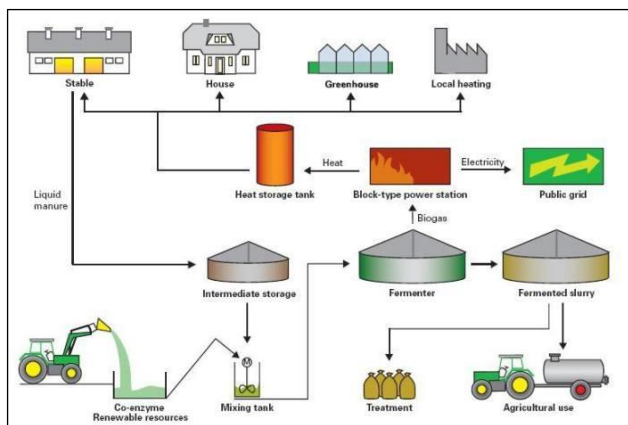
U radu će biti opisan način rada biogasnog agregata za dobijanje električne energije i izveden njegov Matlab/Simulink računarski model. Biće predstavljeni i diskutovani rezultati merenja parametara električne energije na izlazu biogasne elektrane „PIK Moravica“ i upoređeni sa standardima kvaliteta električne energije, koje elektrana mora da zadovolji kod priključenja na javnu distributivnu mrežu.

2. DOBIJANJE BIOGASA

Kod biogasnih postrojenja rezultat procesa anaerobne digestije je nastanak biogasa i digestata. Biogas je mešavina gasova čiju zapreminu čine oko dve trećine metan i jednu trećinu ugljen-dioksid. Postoje različite tehnologije za dobijanje biogasa, to jest različite tehnologije fermentacije, različite vrste fermentora, mešanja, skladištenja biogasa i na kraju iskorišćenje dobijenog biogasa. Biogas ima relativno veliku toplotnu moć, što omogućava različite primene kao izvor energije.

Na slici 1 prikazan je kompletan proces dobijanja biogasa od anaerobne digestije tečnog stajnjaka sa farme goveda, fermentacije i dobijanja biogasa (skuplja se u gasnom rezervoaru), njegovog sagorevanja u kogeneracijskoj jedinici (motor-generator postrojenju za generaciju električne i toplotne energije), njenu predaju električnoj mreži, odnosno distribuciju za grejanje raznih objekata, do načina daljeg korišćenja ostataka fermentacije (iskorišćen supstrat može se dalje upotrebiti kao đubrivo i vraćati na oranice).

Najefikasniji način korišćenje biogasa predstavlja proizvodnja električne i toplotne energije u kogeneracijskom postrojenju. Ono se sastoji od motora sa unutrašnjem sagorevanjem i električnog generatora. Na generator (obično sinhrona mašina) prenosi se mehanička energija dobijena na vratilu motora. Savremeni motori sa unutrašnjim sagorevanjem imaju stepen iskorišćenja oko 40%, ali je i tada na raspolaganju velika količina toplotne energije. Iskorišćavanje toplotne energije se ostvaruje preko rashladne tečnosti. Topla voda se koristi za zagrevanje fermentora i drugih objekata.



Slika 1. Šema procesa anaerobne digestije tečnog stajnjaka sa farme [4]

3. BIOGASNA ELEKTRANA "PIK MORAVICA"

3.1. Opšte o firmi

Poljoprivredno-industrijski kombinat „PIK Moravica“ je osnovalo firmu „Gakovac“ d.o.o., kao posebnu firmu za malu elektranu. „PIK Moravica“, a takođe i „Gakovac“ d.o.o. nalaze se u Staroj Moravici u opštini Bačka Topola. „PIK Moravica“ bavi se ratarstvom na površini od 3.500 ha i stočarstvom tovom junadi kapaciteta oko 2.000 grla. Da bi iskoristili ostaci primarne proizvodnje i stočarstva (slama, kukuruzovina, stajnjak, i dr.), kao najpogodnije rešenje urađeno biogasno postrojenje („Gakovac“ d.o.o.). Ova mala elektrana na biogas ima zadatak da proizvodi biogas iz biomase i konačno električnu i toplotnu energiju. Električnu energiju prodaje Elektrodistribuciji u kategoriji povlašćenih proizvođača, a višak toplotne energije iz rada gasnog motora se koristi za grejanje fermentora i supstrata sa planom dalje distribucije prema obližnjim objektima. Upravljanjem biomasom na ovaj način došlo se do velikih ušteda, a pored toga i dobijeno đubrivo je kvalitetnije za dalju ratarsku proizvodnju.

3.2. Tehnički parametri

Proizvodnja biogasa sastoji se od tri glavna procesa: pripremanje supstrata za doziranje, razgradnja i tretman ostatka. Glavni supstrat predstavlja stajnjak sa farme junadi, kokošiji stajnjak (koji se kupuje sa druge farme), kao i kukuruzna i pšenična silaža iz sopstvene proizvodnje. U biogasnoj elektrani se nalaze dva glavna fermentora, i jedan postfermentor, a vreme zadržavanja supstrata u ovim fermentorima je oko 30-60 dana.

Materijalni bilans ulaznih sirovina, proizvodnja biogasa, proizvodnja električne i toplotne energije data je u tabelama 1 i 2. Elektrana ima dve kogeneracijske jedinice, koje se sastoje od biogasnih motora sa unutrašnjim sagorevanjem i od generatora izlazne snage blizu 1 MW. U tabeli 3 predstavljeni su ključni materijalni bilansi proizvodnje biogasa, kao dobijene električne i toplotne energije na godišnjem nivou [5].

4. MODEL SINHRONOG GENERATORA

Na slici 2 se može videti uprošćen model jedne kogeneracijske jedinice (CHP), to jest blok dijagram motora i generator [6]. Na gornjem delu slike mogu se videti blokovi vezani za motor sa unutrašnjem sagorevanjem (može biti dizel, prirodni gas, biogas, itd).

Na donjem delu slike se vidi jedan veliki blok koji sam po sebi obuhvata sve delove sinhronog generatora, a na izlazu generatora je električna mreža ili potrošač.

Matematički model sinhronog generatora za biogasnu elektranu „PIK Moravica“ napravljen je u Matlab/Simulink okruženju, gde su osnovni podaci uneti na osnovu natpisne pločice sinhronog generatora u elektrani „Gakovac“. Na slikama 3 i 4 mogu videti odzivi za struju i napon dobijeni računarskim simulacijama.

Tabela 1. Dimenzije za pojedinačne objekte biogasne elektrane [5]

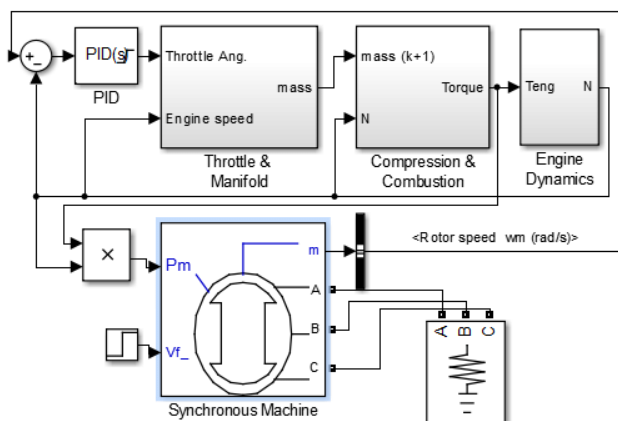
r. br.	OBJEKAT	Prečnik [m]	Zapremina [m ³]	Max. kol. [m ³]
1	Fermentor F1	24	3843	3250
2	Fermentor F2	24	3843	3250
3	Postfermentor F3	32	6832	5300
4	Mešna Jama MJ1	8	150	145
5	Dozator silaže		50	50
6	Mešna Jama MJ1	8	150	145
7	Dozator stajnjaka		20	20
8	Separatna jama JS	8	150	145

Tabela 2. Godišnji plan proizvodnje gasa i potrošnje sirovina [5]

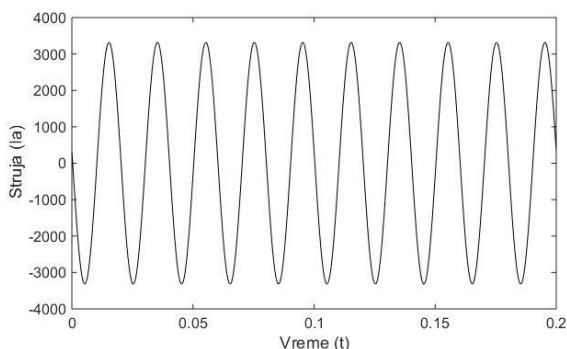
SIROVINA	Godišnja potrošnja [t]	% metana	Gasa po planu [m ³]
Kukuruzna silaža	20000	52,1	4.935.291
Pšenična silaža	1620	57,2	348.585
Otpad iz silosa i mešaone	1440	53,9	817.085
Ražena silaža	1620	56,5	317.027
Govedi stajnjak	18000	53,8	2.068.855
Živinski stajnjak	3600	58,4	389.900
UKUPNO planirano gasa		55	8.876.742
Ukupno potrebno za 8500h x 2 x 522		55	8.874.000

Tabela 3. Materijalni bilans izlaznih produkata [5]

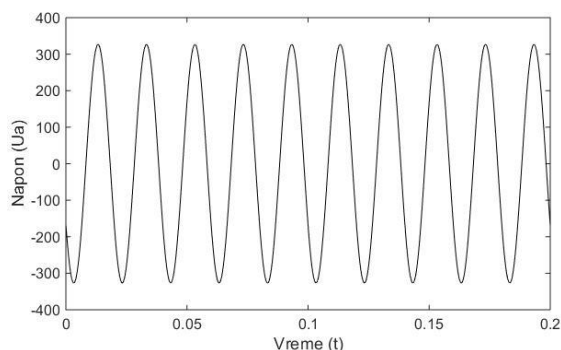
Proizvodnja biogasa:	8.874.000 Nm ³ /god.
Metan iz biogasa (55 %)	4.880.700 Nm ³ /god.
Proizvodnja električne energ.	17.748 MWh/god.
Proizvodnja toplotne energije	18.165 MWh/god.
Sopstvene potrebe el. energija	1.420 MWh/god.
Sopstvene potrebe topl. energ.	1.728 MWh/god.
Raspoloživa el. energija	16.328 MWh/god.
Raspoloživa topl. energija	16.437 MWh/god.



Slika 2. Simulacioni model kogeneracijskog postrojenja (biogasni motor i sinhroni generator) [6]



Slika 3. Izlazna (fazna) struja prema simulacijama.



Slika 4. Izlazni (fazni) napon prema simulacijama

5. REZULTATI MERENJA

Merenje parametara kvaliteta električne energije na izlazu biogasne elektrane urađeno je profesionalnim mernim instrumentom Chauvin Arnoux CA-8332 u skladu sa zahtevima standarda SRPS EN 50160. Karakteristični rezultati merenja prikazani su na slikama 5-8.

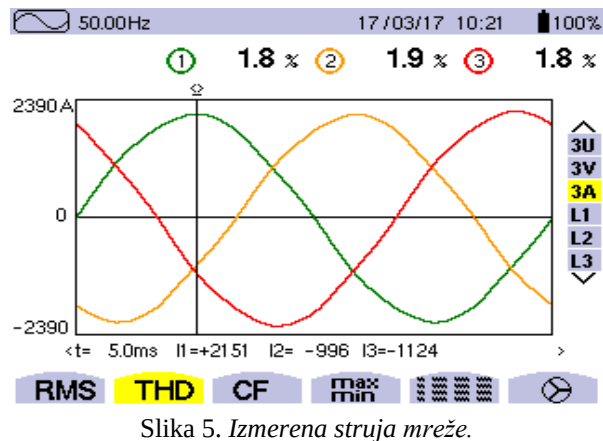
5.1. Trenutne vrednosti napona i struje

Na slikama 5 i 6 prikazane su tri linijske struje i napona, po priključenju mernog instrumenta. Naponi i struje su sinusoidalni sa malim izobličenjem, što se poklapa i sa rezultatima računarskih simulacija.

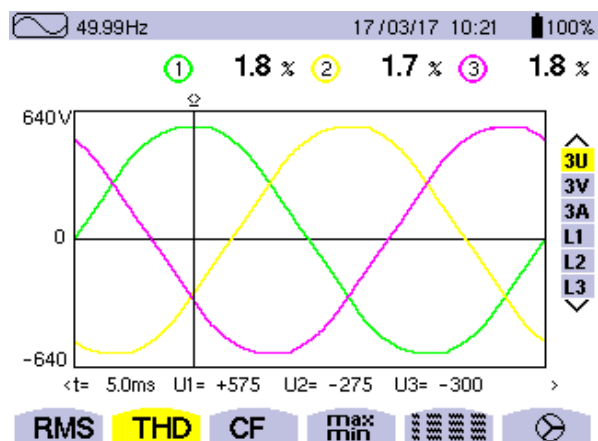
5.2. Statistički rezultati merenja harmonika

Na slikama 7 i 8 prikazan je harmonijski sastav izlaznog napona i struje. Vide se maksimalne i srednje vrednosti, kao i vrednosti koje nisu premašene tokom 95% vremena za ukupno harmonijsko izobličenje napona i struje i za

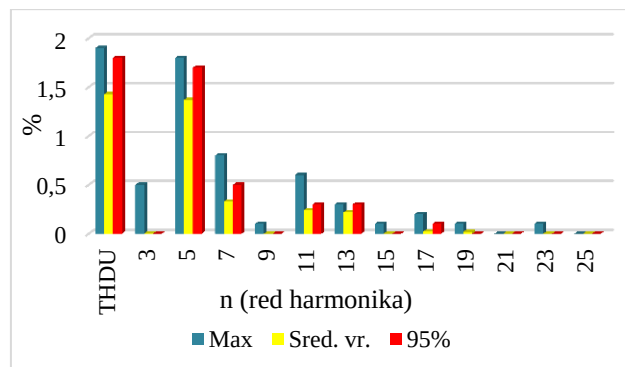
pojedinačne neparne harmonike napona i struje. Na prvom dijagramu, koji pokazuje upoređivanje harmonika napona vidi se da nema velike razlike između vrednosti. Drugi dijagram pokazuje upoređivanje harmonika struje i vidi se, da su maksimalne vrednosti mnogo veće od srednje ili od vrednosti koji nisu premašene tokom 95% slučajeva. To je posledica uključenja i isključenja generatora, jer pri uključenju i isključenju viši harmonici se povećavaju na nedozvoljeni nivo, ali to su samo kratkotrajni skokovi, koji nisu uzet u obzir pri upoređivanju sa standardima.



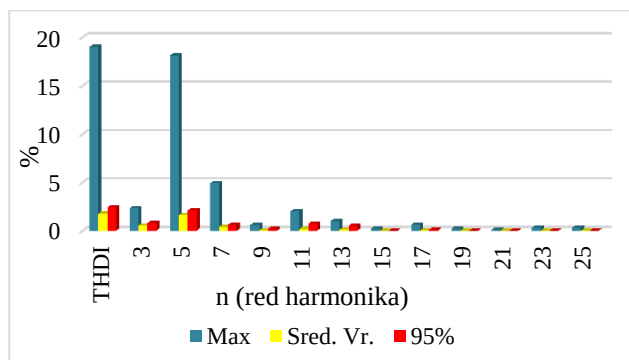
Slika 5. Izmerena struja mreže.



Slika 6. Izmereni napon mreže



Slika 7. Najveća vrednost, srednja vrednost i vrednost koja nije premašena tokom 95% vremena za THDU (ukupni napon distorzije) i za neparne harmonike napona



Slika 8. Najveća vrednost, srednja vrednost i vrednost koja nije premašena tokom 95% vremena za THDI (ukupnu struju distorzije) i za neparne harmonike struje

5.3 Upoređivanje rezultata sa preporukama

Dobijeni rezultati merenja upoređeni su sa dozvoljenim vrednostima viših harmonika struje i napona u skladu sa standardom SRPS EN 50160 (napon) i „Pravilima o radu distributivnih sistema“ (struja). Kao što se vidi u tabelama 4 i 5 ni jedna vrednost višeg harmonika napona ili struje ne prelazi dozvoljenu vrednost. Jedino peti harmonik napona i struje imaju nešto izraženiju vrednost, ali i oni su u intervalu dozvoljenog. Na osnovu toga, može se zaključiti da biogasna elektrana zadovoljava uslove priključenja na distributivnu mrežu EPS-a.

Tabela 4. Upoređivanje merene vrednosti viših harmonika napona sa dozvoljenim vrednostima

Redni broj višeg harmonika [v]	Merena vrednost [V]	Dozvoljena vrednost [V]
5	7,0686	9,355
7	1,6624	18,71
11	1,245	18,71
13	1,246	15,90

Tabela 5. Upoređivanje merene vrednosti viših harmonika struja sa dozvoljenim vrednostima

Redni broj višeg harmonika [v]	Merena vrednost [A] (svedeno na 10 [kV])	Dozvoljena vrednost [A]
2	0,1221	1,085
3	0,4727	/
4	0	0,355
5	1,0084	1,085
6	0	0,430
7	0,3062	1,534
8	0	0,150
9	0,1221	/
10	0	0,262
11	0,1772	0,973
12	0	0,206
13	0,1223	0,711

6. ZAKLJUČAK

Iskorišćenje energije iz biogasa je još u fazi razvoja u Vojvodini, iako su uslovi za dobijanje ove vrste obnovljivih izvora energije jako dobri. Jedan dobar primer za biogasnu elektranu u Vojvodini je u Staroj Moravici, „PIK Moravica“, odnosno firma „Gakovac“. Ova biogasna elektrana ima dve kogeneracijske jedinice, koje se sastoje od biogasnih motora sa unutrašnjim sagorevanjem i od sinhronih generatora 1 MW izlazne snage, tako da je ukupna nazivna snaga 2 MW.

Rezultati merenja parametara kvaliteta električne energije (struje, napona i snage) na izlazu generatora su pokazali odlične vrednosti, i poklapaju se sa rezultatima simulacije modela generatora. Na bazi upoređivanja sa zahtevima odgovarajućih domaćih standarda za kvalitet električne energije (SRPS EN 50160) i odgovarajućih preporuka EPS-a („Pravila o radu distributivnih sistema“) može se zaključiti da oni odgovaraju svim uslovima za priključenje male elektrane na električnu mrežu.

4. LITERATURA

- [1] Ministarstvo energetike i rudarstva, „Nacionalni akcioni plan za korišćenje obnovljivih izvora energije Republike Srbije“, Beograd, 2013.
- [2] ***, „Biomasa – najrasprostranjeniji i nedovoljno iskorišćen izvor energije“, Energetski portal, 2019, <https://www.energetskiportal.rs/obnovljivi-izvori-energije/biomasa/>
- [3] M. Stanojević, S. Simić, A. Jovović, D. Radić, M. Obradović, D. Todorović, „Biogas: Dobijanje i primena“, Mašinski fakultet, Beograd, 2014
- [4] S. Zafar, „Trends in utilization of biogas“, 2019 <https://www.bioenergyconsult.com/utilization-of-biogas>
- [5] Program i plan održavanja biogasnog postrojenja (1MW x 2) „Gakovac“-Stara Moravica
- [6] S. Benhamed i dr., „Dynamic modeling of diesel generator based on electrical and mechanical aspects“, 2016 IEEE Electrical Power and Energy Conference (EPEC), Ottawa (Canada), Oct.2016, DOI: 10.1109/EPEC.2016.7771756

Kratka biografija:



Igor Pušin rođen je u Subotici 1993. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine odbranio je 2019.god.

kontakt: igorpusin@gmail.com



Vladimir Katić rođen je u Novom Sadu 1954. Doktorirao je na Univerzitetu u Beogradu 1991. god., a od 2002 je redovni profesor Fakultetu tehničkih nauka u Novom Sadu. Oblast interesovanja su energetska elektronika, obnovljivi izvori električne energije, električna vozila i kvalitet električne energije.

**НАМЕНСКИ ЈЕЗИК И ОКРУЖЕЊЕ ЗА МОДЕЛОВАЊЕ И ГЕНЕРИСАЊЕ
СИСТЕМА СПОСОБНОСТИ У ИГРАМА СА УЛОГАМА****A DOMAIN SPECIFIC LANGUAGE AND A FRAMEWORK FOR MODELING AND
GENERATION OF ABILITY SYSTEMS IN ROLE-PLAYING GAMES**

Бојан Батало, Факултет техничких наука, Нови Сад

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – У овом раду описан је развијени софтверски алат који омогућава специфицирање и генерисање система способности у видео играма са играњем улога. Систем се састоји из наменског језика, једне Model-To-Text трансформације и оквира за развој видео игара. Употребом описаног система могуће је генерисати текстуалне игре role-playing жанра и покренути их у окружењу Python.

Кључне речи: Наменски језици, развој софтвера заснован на моделима, видео игре, игре са улогама

Abstract – In this paper we present a system for specification and code generation of ability systems for role-playing video games. The system comprises of a domain specific language, one Model-To-Text transformation and a framework for developing video games. Using the proposed system, it is possible to generate runnable textual role-playing games for Python environment.

Keywords: Domain-Specific Languages, Model-Driven Software Development, video games, role-playing games

1. УВОД

Видео игре с улогама (Computer role-playing games, скраћено CRPG или RPG) припадају жанру видео игара које садрже богату причу и садржај, као и сложена правила игре, која су прецизно и формално дефинисана. Оне потичу од класичних друштвених игара с улогама, од којих су наследиле скоро све главне карактеристике, као што су комплексна правила, богати светови и наратив у првом плану. Најпознатија класична друштвена игра с улогама је *Dungeons & Dragons* (D&D), која је настала 1974. године [1]. D&D се сматра почетком развоја модерних игара с улогама и већина игара насталих након ње су бар делимично преузеле неке од елемената D&D.

Развојем рачунара за личну употребу и технологија рачунарске графике, развијају се и верзије ових игара на рачунарима, прво у текстуалној форми на терминалима, а затим и у 3D форми [2].

Игре RPG жанра стављају посебан нагласак на ликове или групу ликова којом играч управља, њиховим

способностима, снази и развоју кроз причу. Како лик напредује, развијају се његове способности и отварају нове могућности за интеракцију са светом, или нови начини за борбу са непријатељима. Способности које лик може да стекне и начини на које их је могуће стећи дефинисани су у оквиру система способности. Систем способности је генерички појам који обухвата скуп способности и правила стицања способности и структурира их на одређени начин, који се разликује од игре до игре. Структуром система способности дефинише се путања развоја лика и диктира се динамика игре, јер ликови који су јачи и имају боље способности лакше решавају задатке и заобилазе препреке.

1.1. Мотивација

Дизајнирање система способности представља изазов, јер систем првенствено мора бити довољно занимљив и привлачан играчима. Поред тога, препоручљиво је да систем буде уравнотежен, односно да се играчима пружи могућност избора различитих способности и да не постоји један очигледан најбољи избор. Играчима је битно да доносе одлуке у вези развоја лика, јер се тако ближе поистовећују са њима [3].

Системи способности морају бити довољно сложени како би наградили играче за напредовање од почетних нивоа игре до крајњих. Како је нарација главни нагласак у играма са играњем улога, битно је играча оставити са утиском напредовања и јачања лика којим рукује. Неретко се од развојног тима игре додатно очекује да је у стању да мења и прилагођава систем способности кроз време. Наведени захтеви према систему способности изискују флексибилан приступ дизајну самог система, који је подложен променама у било којој фази развоја игре, укључујући и период када је игра издата и увелико се игра.

Главни проблем је недостатак наменских алата за дизајн видео игара. Алати које дизајнери користе за дизајн игре, описе дијаграма, визуелизацију одређених делова система способности и нарацију помоћу природног језика су већином алати опште намене. Они нису у стању да обезбеде формално специфицирање видео игара или њених делова, већ искључиво опис путем природног језика. Недостатак одговарајућих алата доводи до тога да се свака спецификација игре и њеног система способности мора ручно имплементирати на нивоу програмског кода, чиме се ствара проблем раздвојености спецификације и имплементације. Тиме се успорава процес

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био др Иван Луковић, ред. проф.

развоја игре, што у изузетно конкурентном тржиštu [4] може да има негативне последице по будућност игре.

1.2. Циљ и задаци

Главни циљ овог рада јесте решавање проблема раздвојености дизајна система способности и његове имплементације у оквиру видео игре. Потребно је омогућити дизајнерима да специфицирају систем способности у оквиру игре помоћу наменских алата, који користе концепте са којима су дизајнери упознати из искуства, а који су довољно формално дефинисани како би се могли искористити за имплементације.

Концепти система способности су довољно генерички, те је могуће направити наменски језик који би служио за дизајн система способности.

За потребе остварења наведеног циља, потребно је креирати систем који садржи наменски језик за специфицирање система способности у видео играма са улогама и трансформације које ће концепте наменског језика пресликати у концепте неког од имплементационих окружења. Поред тога, потребно је развити низ алата који ће дизајнерима омогућити употребу наменског језика и олакшати развој видео игара овим приступом.

Очекује се успешан развој наменског језика за спецификацију система способности, алата за његову употребу и оквира за тестирање система дизајнираних употребом ових алата. Наменски језик мора да обухвати све битне концепте домена проблема, тако да омогући његову несметану употребу. Успешно остваривање постављених циљева доприноси развоју наменских алата за развој видео игара, а пре свега система способности. Овим се олакшава експертима дизајн игара са системима способности и отвара могућност за даљи развој наменских алата.

1.3. Структура рада

Овај рад садржи увод, закључак и три поглавља. Поглавље 2 описује архитектуру имплементираних алата. Поглавље 3 описује апстрактну и конкретну синтаксу наменског језика, док поглавље 4 описује *Model-To-Text* трансформацију, заједно са примером генерисане видео игре.

2. АРХИТЕКТУРА СИСТЕМА

У овом поглављу описана је архитектура система који чине имплементирани алати и наменски језик за дизајнирање система способности у играма са играњем улога. Укратко су описани сви модули, њихова улога у систему, као и технологија употребљена приликом њихове имплементације. На слици 1. илустрована је архитектура решења и организација модула система.

Архитектура обухвата три модула, једну врсту корисника и један тип крајњег производа. Дизајнер видео игре има улогу корисника система и користи наменски језик да опише, односно моделује систем способности. Наменски језик, *RPGLang*, подржава и моделовање основног скелета видео игре ради генерисања кода који омогућује интерактивно тестирање система способности.



Слика 1. Приказ архитектуре система и процеса генерисања видео игре

Моделована игра се снима у .rpg датотеку, која се користи као улаз за модул генератора кода. Модул генератора кода генерише програмски код у програмском језику Python, посебно припремљеном тако да буде употребљив у наредном модулу, погону за видео игре. Модул погона за видео игре садржи оквир који може да integriше генерисани код и произведе функционалну конзолну видео игру. Овако генерисана видео игра може да послужи као оквир за тестирање карактеристика система способности.

3. НАМЕНСКИ ЈЕЗИК ЗА МОДЕЛОВАЊЕ И ГЕНЕРИСАЊЕ СИСТЕМА СПОСОБНОСТИ

Језици за моделовање су алати који дизајнерима система пружају могућност моделовања система у различите сврхе, као што су дескриптивно моделовање система, моделовање будућих екстензија система и моделовање решења за постојеће проблеме. Наменски језици су они језици за моделовање који су намењени за узак домен примене. Основни чиниоци језика за моделовање су апстрактна синтакса, конкретна синтакса и семантика језика. Апстрактна синтакса дефинише структуру језика, примитиве које је могуће користити и начине на које се оне могу комбиновати. Она је независна од конкретних репрезентација и корисници језика веома често немају додирних тачака са њом. Конкретна синтакса дефинише нотације наменског језика и описује његов визуелни изглед, а може бити текстуална и графичка.

Апстрактна синтакса наменског језика *RPGLang* развијена је у Eclipse развојном окружењу, употребом *Eclipse Modeling Framework* (скраћено *EMF*) оквира за метамоделовање. *EMF* је имплементација *Meta-Object Facility* (скраћено *MOF*) стандарда од стране *Eclipse Foundation* и представља мета-метамодел, помоћу којег се дефинишу метамодел. Метамодел наменског језика је уједно и апстрактна синтакса наменског језика и садржи се од концепата уочених у домену употребе језика, те су самим тиме блиски доменским експертима.

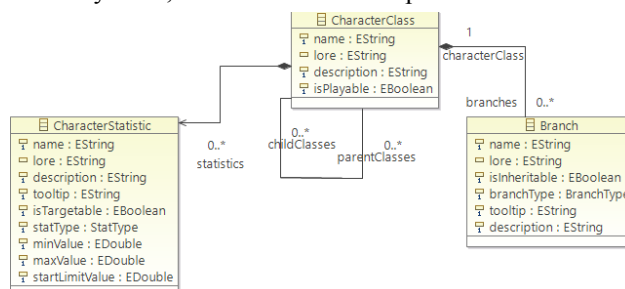
Метамодел наменског језика *RPGLang* састоји се од следећих концепата:

- i. Игра - основни и коренски концепт у наменском језику и представља саму игру која се моделује,
- ii. Јединица домета - дефинише основну јединицу простора која се користи за израчунавање домета способности,
- iii. Класа лика – представља групу сличних ликова, који деле карактеристике и способности,
- iv. Статистика лика - моделују атрибуте класе лика, као што су здравље, снага, тежина, магијска моћ и слично,

- v. Грана – скуп чворова уређен у усмерени ациклични граф,
- vi. Чвор – сваки чвор садржи способности или таленте, зависно од типа гране којој припада,
- vii. Активација чвора – механизам откључавања чвора са способностима или талентима и унапређења лика, у замену за одређене ресурсе,
- viii. Цена активације - услов активације чвора који се заснива на употреби ресурса у замену за активирање наредног нивоа,
- ix. Услов пропагације - услов активације чвора који се заснива на задовољености активационог нивоа неког другог чвора,
- x. Способност - централни део система способности, помоћу којих играч може да се бори са непријатељима, реши загонетке и друге задатке и напредује кроз игру,
- xi. Ефекат - саставни део способности и дефинише шта се дешава када се способност активира и погоди мету,
- xii. Промена - означава начин на који се стање у игри мења активацијом ефекта, односно који елементи игре се мењају и како,
- xiii. Промена статистике - моделује случај када ефекат директно утиче на статистике одређеног лика,
- xiv. Промена периода чекања - односи на способности које лик (или његови противници) поседују, повећавајући или смањујући период чекања одговарајуће способности,
- xv. Откуцај - моделује концепт промене статистике лика кроз време,
- xvi. Модификатор – моделује механизам промене статистике лика у оквиру ефекта неке способности који зависе од статистика лика који користи ту способност, као и статистика лика ка ком је способност усмерена,
- xvii. Генератор насумичних бројева - дефинишу као вероватноћа да се одређени догађај деси, а користе се да уведу несигурност у исходе борби, загонетке и других концепата у оквиру игара,
- xviii. Таленат - дефинише унапређења за статистике лика и ликове способности, додаје одређене ефекте постојећим способностима, смањују цену употребе или откључава посебне способности,
- xix. Таленат статистике - тип талента који има утицај на статистике лика,
- xx. Таленат периода чекања - тип талента који има утицај на периоде чекања способности лика који поседује тај таленат и
- xxi. Таленат откључавања - тип талента који има утицај на откључавање способности, ефеката, промена и модификатора.

Приликом дизајна метамодела наменског језика тежило се увођењу нивоа апстракције и генеричности како би се омогућила што већа флексибилност приликом дизајна игара употребом имплементираног језика. Са друге стране, решење је постављено тако да не буде превише генеричко, јер би се тиме значајно

изгубило на интуитивности и употребљивости језика. На слици 2 приказан је део метамодела који се односи на класу лика, њене статистике и гране способности.



Слика 2. Приказ дела метамодела

Конкретна синтакса је дефинисана на основу концепата садржаних у оквиру апстрактне синтаксе, односно метамодела. Реализована је као графичка конкретна синтакса. Имплементирана је употребом *Sirius* оквира за креирање графичке конкретне синтаксе.

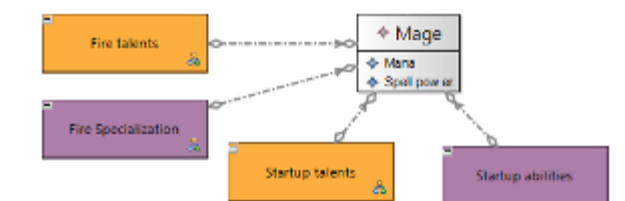
Овај оквир користи приступ пресликавања за развој графичке конкретне синтаксе, при чему се за сваки концепт апстрактне синтаксе експлицитно дефинише графичка репрезентација. Упоредо је аутоматски креиран едитор, који се користи за моделовање употребом графичке конкретне синтаксе.

Овај приступ употребљен је и за развијање графичке конкретне синтаксе наменског језика *RPGLang*. Она је развијена тако да обезбеди прегледност и функционалност, док критеријум естетике није био у првом плану.

Сваком концепту апстрактне синтаксе додељена је графичка репрезентација у облику квадрата, правоугаоника, ромба или кружнице, различите боје. Везе између концепата приказане су као линије са стрелицама, или се наводе у оквиру графичког окружења за измену атрибута концепта.

Имплементирано је графичко окружење за моделовање, које пружа могућности креирања, измене и брисања свих концепата наменског језика.

Игра, као коренски концепт, приказује се у основном дијаграму, а концепти које садржи приказани су на том дијаграму. Ради прегледности, концепти као што су грана, способност и таленат отварају се у сопствене дијаграме, који садрже детаљније приказе. На слици 3 приказан је пример графичке конкретне синтаксе.



Слика 3. Приказ употребе конкретне графичке синтаксе

4. MODEL-TO-TEXT ТРАНСФОРМАЦИЈА И ПРИМЕР ГЕНЕРИСАНЕ ИГРЕ

Model-To-Text трансформација која представља генератор кода, развијена је као *Java* апликација без графичког корисничког интерфејса. Ова апликација преузима модел креиран наменским језиком као улаз, а као излаз генерише *Python* програмски код. Само генерисање кода имплементирано је употребом *Xtend* програмског језика.

Генератор кода у потпуности је зависан од мета-модела наменског језика, јер користи концепте дефинисане у оквиру метамодела и трансформише их у текст. *Xtend* омогућава пресликавање концепата метамодела у конкретан текст употребом шаблона. Генерисани *Python* код могуће је покренути, чиме се добија текстуална игра у рачунарској конзоли, која може да послужи за тестирање моделованог система способности. Апликација генератора кода имплементирана је тако да омогућава једноставну имплементацију додатних генератора кода, за различите циљне системе, као што су погони за видео игре попут *Unreal Engine* и *Unity*. На слици 4 приказан је пример скупа способности генерисане видео игре.

```
rpg$: show_abilities
Fireball 1/3 | Tier 1
Cost: 4% Mana | Deals: 20 Fire Damage
Cast time: 2 turns

Pyroblast 1/2 | Tier 2
Cost: 20% Mana | Deals: 100 Fire Damage
Cast time: 6 turns

Fire Blast: 0/1 | Tier 3
Cost: 8% Mana | Deals: 25 Fire Damage
Chance: 10% of 1 turn Cooldown Reduction
Cast time: Instant | Cooldown: 5 turns

Flamestrike: 0/2 | Tier 4
Cost: 20% Mana | Deals: 30 AOE Fire Damage
Cast time: 5 turns

rpg$:
```

Слика 4. Пример скупа способности у генерисаној видео игри

5. ЗАКЉУЧАК

Због пораста броја видео игара са играњем улога и раста тржишта видео игара јавља се потреба за њиховим брзим и ефикасним развојем. Предложени приступ развоју игара ослања се на употребу методологије развоја софтвера заснованог на моделима и резултује имплементацијом наменског језика.

Наменски језик се састоји из апстрактне и конкретне синтаксе. Апстрактна синтакса представља метамодел наменског језика и апстрахује концепте који су пронађени као заједнички међу више игара са играњем улога и кључни за описивање система способности за такве игре. Дефинисана су ограничења у оквиру метамодела како би се ојачала структура, пружили основи и механизми за валидацију креираних модела и онемогућила неисправна употреба наменског језика. Конкретна синтакса је имплементирана као графичка конкретна синтакса, чиме су се дефинисаним концептима апстрактне синтаксе придружиле одговарајуће графичке репрезентације и имплементирали алати за моделовање употребом графичке конкретне синтаксе.

Даљи правци развоја су вишеструки. Што се тиче апстрактне синтаксе, могуће је проширити метамодел концептима који ће подржавати додатне врсте талената, модификатора и ефеката. Пожељно је посветити посебну пажњу концептима који ће дефинисати интеракцију између способности, јер су такви концепти учестали у модерним играма са играњем улога, а нису подржани у наменском језику *RPGLang*.

Потребно је унапредити графичку конкретну синтаксу како би се добило на изражајности и прегледности, а посебну пажњу посветити естетском делу синтаксе. Пожељан правац даљег развоја био би развој генератора кода за један од популарних погона за игре као што су *Unity* и *Unreal Engine*. Спајањем наменског језика *RPGLang* и популарног погона за игре би пружио аутору платформу да добије битне повратне информације о валидности употребе овог наменског језика у индустријске сврхе од стране развојних студија видео игара.

Решење описано у раду, иако није погодно за комерцијалну употребу, може да послужи као добар пример како би се решили генерички проблеми у развоју видео игара и како би се скратило време развоја. Такође, решење помаже у смањивању јаза између дизајнера, односно доменских експерата који су циљна група наменског језика *RPGLang*, и инжењера који имплементирају игре. У широј слици, даљи развој решења може довести до пробоја развоја софтвера заснованог на моделима у индустрију видео игара. Тиме би се допринело формализацији и аутоматизацији развоја игара на основу образаца који постоје у домену већ десетинама година.

6. ЛИТЕРАТУРА

- [1] <http://dnd.wizards.com/> (приступљено у фебруару 2019.)
- [2] M. Barton: *Dungeons and Desktops*, New York: A K Peters/CRC Press, 2008
- [3] Sean Q. Hendricks, W. Keith Wriker: *Gaming as Culture - Essays on Reality, Identity and Experience in Fantasy Games*, McFarland, 2006
- [4] <https://www.statista.com/statistics/552623/number-games-released-steam/> (приступљено у фебруару 2019.)

Кратка биографија:



Бојан Батало рођен је у Руми 1994. год. Факултет техничких наука уписао је 2013. године. Бечелор рад из области Електротехнике и рачунарства – Рачунарске науке и информатика одбранио је 2017. године. Мастер рад из исте области одбранио је 2019. године.

контакт: bbatalo@hotmail.com

PRORAČUN PREKIDA FAZE U DISTRIBUTIVNOJ MREŽI**ANALYSIS OF PHASE INTERRUPTION IN DISTRIBUTION NETWORKS**

Mina Lakić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu se razmatra prekid faze u radijalnoj distributivnoj mreži. Cilj je da se izračunaju naponi i struje u distributivnoj mreži posle prekida faze. Razmatranja su ilustrovan na primeru jedne distributivne mreže 20kV sa 3 izvoda. Razvijeno je programsko rešenje za prekid faze u distributivnoj mreži koje omogućava interakciju sa korisnikom.

Ključne reči: prekid faze, fiktivno kolo, distributivna mreža

Abstract – This paper deals with phase interruption in radial distribution network. The goal is to calculate voltages and currents in distribution network after the phase has been interrupted. Considerations are illustrated on example with one 20kV distribution network and 3 feeders. Therefore, the software solution with graphical user interface has been developed, for one phase interruption in distribution network, which allows interaction with user.

Keywords: Phase interruption, Δ -circuit, Distribution network

1. UVOD

U radu se razmatra prekid faze u distributivnoj 20kV mreži. Da bi se na neki način predupredili takvi poremećaji u radu distributivne mreže, potrebno je analizirati takve poremećaje [1,2,3,4]. Ovakvi poremećaji su usloveli razvoj specijalizovanog algoritma za proračune režima sa prekidom faza u distributivnoj mreži.

Da bi se takav model i realizovao, u drugom delu teze se detaljno izvode sve potrebne jednačine za opis režima sa prekidom faza na samom mestu prekida.

U trećem delu rada je započeto sa rešavanjem prethodno definisanog problema proračuna režima sa prekidom faza u distributivnoj mreži.

U četvrtom delu se generalizuju razmatranja iz drugog dela i uz uvažavanje modela distributivne mreže iz trećeg dela izvodi se odgovarajući model za proračun režima sa prekidom faza u radijalnoj mreži i daje se odgovarajući algoritam za proračun.

U petom delu je dat primer proračuna režima sa prekidom faza u radijalnoj mreži, dok je u šestom delu dat zaključak.

2. OPŠTA RAZMATRANJA U VEZI SA PRORAČUNOM REŽIMA SA PREKIDOM FAZA

U ovom delu će biti analizirani „čisti“ prekidi jedne faze u distributivnoj mreži. Za početak se uvode dve pretpostavke kojima se obezbeđuju određena pojednostavljenja koja su pogodna za izvođenja koja slede, a koja kasnije neće predstavljati smetnju da se izvrši eventualna generalizacija.

Te pretpostavke su:

1. Da su prekidi faza koji se događaju potpuni.

Odnosno, da se admitansa nulte brojne vrednosti nalazi na mestu prekida između tačaka prekida.

2. Da se prekidi faza događaju na krajevima grana.

Cilj ovih razmatranja je da se detaljno opiše režim na mestu prekida i da se za prekid jedne faze u radijalnoj mreži izvedu svi izrazi od interesa, a da se u narednim delovima rada na osnovu ovih izvođenja daju izrazi za proračune za prekide u proizvoljno izabranoj jednoj fazi, pa je za analize koje slede izabrana faza L1.

2.1. Osnove proračuna režima sa prekidom faza

Proračun ustaljenog režima sa prekidom faza globalno se sastoji od dva koraka:

1. Primenom principa superpozicije/dekompozicije režima linearnih kola, režimi distributivne mreže sa kvarom se razlažu – dekomponuju na režim iste distributivne mreže pre kvara i režim tzv. fiktivnog kola. Trofazni simetrični režim distributivne mreže pre kvara se smatra poznatim. Režim fiktivnog kola je, u opštem slučaju, trofazan nesimetričan, sa nesimetrijom istog karaktera kao nesimetrija distributivne mreže sa kvarom. Osnovna pogodnost fiktivnog kola je u njegovoj pasivnosti (nepostojanja naponskih i strujnih izvora) u svim tačkama osim mesta kvara.
2. Proračun nesimetričnog trofaznog režima fiktivnog kola primenom metode simetričnih komponenti. Ovde se trofazno fiktivno kolo sa nesimetričnim režimom, primenom pomenute metode, razlaže na tri trofazna kola sa direktnim, inverznim i nultim režimima. Svako od njih, s obzirom na simetričnost njegovog režima, prikazuje se pofazno. To je omogućeno uvođenjem pretpostavke o uravnoteženosti svih trofaznih elemenata distributivne mreže.

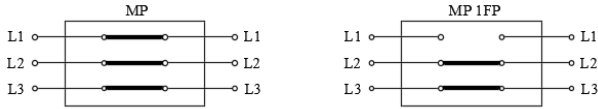
Fiktivno trofazno kolo čine uravnoteženi trofazni elementi i potrošači, a neuravnoteženost unosi samo mesto prekida. Razmatra se slučaj prekida faza isključivo na krajevima trofaznih elemenata, pa prekid ne remeti uravnoteženost elemenata koji konstituišu distributivnu mrežu.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je dr Duško Bekut, redovni profesor.

2.2. Prikaz i definicija prekida faze

U razmatranjima koja slede analiziraju se „čisti“ jednofazni prekidi faza (bez dodatnih impedansi između tačaka prekida). Topološki prikaz mesta prekida faza pre nastanka kvara, kao i pomenuti kvar – dat je na slici 2.2.



Slika 2.2. – Topološki prikaz prekida faze:
a – mesto prekida pre nastanka prekida (MP)
b – jednofazni prekid faze L1 (MP 1FP)

Ako se umesto topologije prekida MP postavi topologija bilo kog prekida MPX (prekid jedne 1FP, dve 2FP ili tri faze 3FP), dobije se distributivna mreža sa prekidom faze izabranog tipa u grani između čvora k i ustanovljenog čvora m. Ovakav slučaj prikazan je na slici 2.3. Na njoj su sa $u_{L1k}^p, u_{L2k}^p, u_{L3k}^p$, odnosno $u_{L1m}^p, u_{L2m}^p, u_{L3m}^p$, označeni fazori faznih napona čvora k i m, respektivno (indeks „p“ – posle kvara). Sa $i_{L1k}^p, i_{L2k}^p, i_{L3k}^p$, odnosno sa $i_{L1m}^p, i_{L2m}^p, i_{L3m}^p$, fazori injektiranih struja (iz mesta prekida) u čvorove k i m, respektivno. Sa $i_{L1}^p, i_{L2}^p, i_{L3}^p$, označeni su fazori struja kroz mesto prekida posle prekida (različiti od nulte brojne vrednosti samo u fazama u kojima nije došlo do prekida):

$$\begin{aligned} i_{L1}^p &= -i_{L1k}^p = i_{L1m}^p \\ i_{L2}^p &= -i_{L2k}^p = i_{L2m}^p \\ i_{L3}^p &= -i_{L3k}^p = i_{L3m}^p \end{aligned} \quad (2.1)$$

Matematički opis topologije prekida – definicija prekida data je izrazom 2.2:

$$\begin{aligned} H_1[u_{L1k}^p, u_{L2k}^p, u_{L3k}^p, u_{L1m}^p, u_{L2m}^p, u_{L3m}^p, i_{L1}^p, i_{L2}^p, i_{L3}^p] &= 0, \\ H_2[u_{L1k}^p, u_{L2k}^p, u_{L3k}^p, u_{L1m}^p, u_{L2m}^p, u_{L3m}^p, i_{L1}^p, i_{L2}^p, i_{L3}^p] &= 0, \\ H_3[u_{L1k}^p, u_{L2k}^p, u_{L3k}^p, u_{L1m}^p, u_{L2m}^p, u_{L3m}^p, i_{L1}^p, i_{L2}^p, i_{L3}^p] &= 0. \end{aligned} \quad (2.2)$$

Specifikacija linearnih zavisnosti, H_1 , H_2 i H_3 saglasno topologiji prekida (slika 2.2.), za prekid faze, data je u tabeli 2.1:

Tabela 2.1. Matematički opis prekida faze između čvorova k i m

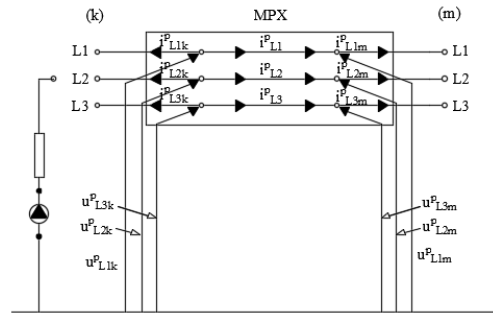
Tip prekida faza funkcije	Prekid jedne faze (faza L1)
H_1	$i_{L1}^p = 0$
H_2	$u_{L2k}^p - u_{L2m}^p = 0$
H_3	$u_{L3k}^p - u_{L3m}^p = 0$

Izrazi specificirani u tabeli 2.1 predstavljaju definiciju prekida jedne faze.

2.3. Dekompozicija režima sa prekidom faza na režim pre kvara i režim fiktivnog kola

U opštim izlaganjima iznetim u ovoj glavi, nagoveštene su dve dekompozicije režima sa prekidom faze, na kojim se zasniva proračun tih režima. U ovom delu je reč o prvoj od dve dekompozicije, koja se odnosi na dekompoziciju nesimetričnog trofaznog režima sa

prekidom faze na poznati simetričan režim pre nastanka prekida i nesimetrični režim u fiktivnom trofaznom kolu. Ako se pođe od globalnog prikaza distributivne mreže, sa prekidom faze, datog na slici 2.3, njen se režim neće promeniti ako se između tačaka gde se dešava prekid faze (čvorovi k i m), paralelno sa topologijom prekida, ustanove nulti (po brojnoj vrednosti) idealni strujni izvori prikazani na slici 2.4.



Slika 2.3 – Globalni prikaz mreže sa prekidom faze X između čvorova m i k ($X = 1FP, 2FP, 3FP$)



Slika 2.4 – Ekvivalentni sistem sistema sa prekidom faze

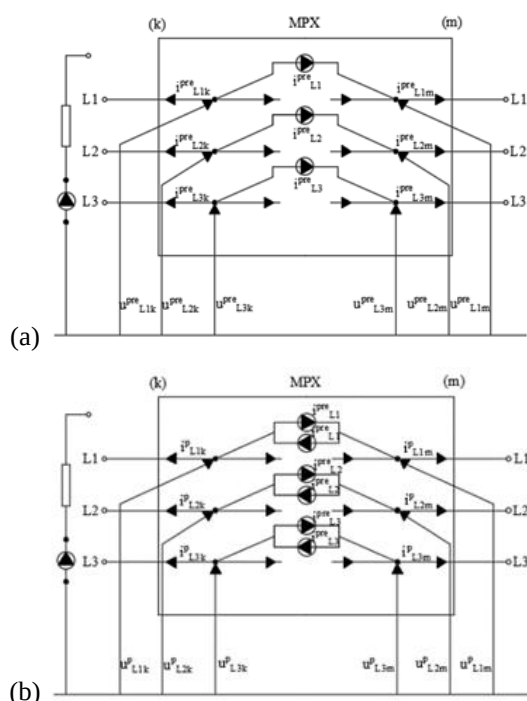
Kolo sa slike 2.4 zajedno sa svojim režimom, može se prikazati pomoću 2 kola. Prvo je dobijeno iz razmatrane distributivne mreže sa prekidom faze (slika 2.4) zapažajući identičnu topologiju i sve idealne naponske izvore, uz tri idealna strujna izvora u sve tri faze između tačaka prekida k i m, usmerena od čvora k ka čvoru m. Preostala tri strujna izvora su prekinuta.

Ako je određena faza u prekidu, ta je struja jednaka nuli, a struja između čvorova k i m, jednaka sa onom pre prekida, obezbeđuje odgovarajući idealni strujni izvor. Za fazu koja nije u prekidu, ta je struja jednaka odgovarajućoj struji pre prekida u toj fazi, a odgovarajući strujni izvor ne igra nikakvu ulogu.

Drugo kolo – fiktivno, dobija se takođe zapažanjem topologije osnovnog kola (slika 2.4), kratkim spajanjem svih idealnih naponskih izvora i prekidanjem sva tri strujna izvora koji postoje u prethodnom kolu, a zapažanjem samo tri preostala idealna strujna izvora između tačaka prekida, usmerena od čvora m ka čvoru k. Struje kroz mesto prekida u ovom kolu su obeležene sa i_{L1}, i_{L2}, i_{L3} .

Vrednosti struja u grani sa prekidom faze dati su izrazima 2.3:

$$\begin{aligned} i_{L1k}^p &= i_{L1k}^{\text{pre}} + i_{L1k} = -i_{L1}^{\text{pre}} + i_{L1k}, \\ i_{L1m}^p &= i_{L1m}^{\text{pre}} + i_{L1m} = i_{L1}^{\text{pre}} + i_{L1m}, \\ i_{L2k}^p &= i_{L2k}^{\text{pre}} + i_{L2k} = -i_{L2}^{\text{pre}} + i_{L2k}, \\ i_{L2m}^p &= i_{L2m}^{\text{pre}} + i_{L2m} = i_{L2}^{\text{pre}} + i_{L2m}, \\ i_{L3k}^p &= i_{L3k}^{\text{pre}} + i_{L3k} = -i_{L3}^{\text{pre}} + i_{L3k}, \\ i_{L3m}^p &= i_{L3m}^{\text{pre}} + i_{L3m} = i_{L3}^{\text{pre}} + i_{L3m}. \end{aligned} \quad (2.3)$$



Slika 2.5. Dekompozicija režima distributivne mreže sa prekidom faze na: a – sistem – režim pre nastanka prekida faze; b – fiktivno kolo

2.4. Dekompozicija fiktivnog kola sa nesimetričnim režimom na tri kola sa simetričnim režimom

Pri proračunu režima sa prekidom faze iz mesta kvara u trofazni fiktivni sistem se „gleda“ iz dva čvora k i m između kojih se dešava prekid faze. Trofazno fiktivno kolo se može prikazati pomoću tri monofazna kola koja su svuda raspregnuta osim na mestu prekida faze, kao na slici 2.6.

$$\begin{aligned} \mathbf{u}_j^{d,i,0} &= \mathbf{A} \cdot \mathbf{u}_j, \quad j = 1, 2, \dots, n, \\ \mathbf{i}_j^{d,i,0} &= \mathbf{A} \cdot \mathbf{i}_j, \quad j = 1, 2, \dots, n, \end{aligned} \quad (2.5)$$

Slika 2.6 – Rasprezanje trofaznog fiktivnog kola sa nesimetričnim režimom na tri trofazna, po fazno prikazana kola: a – direktnog, b – inverznog i c – nultog režima

Za transformaciju faznih veličina u simetrične komponente koristi se sledeća matrica:

$$\mathbf{A} = \frac{1}{3} \begin{bmatrix} 1 & a & a^2 \\ 1 & a^2 & a \\ 1 & 1 & 1 \end{bmatrix}, \quad a = e^{j\frac{2\pi}{3}}, \quad (2.7)$$

Za simetrične komponente fazora injektiranih struja u čvorove fiktivnog kola važi:

$$\mathbf{i}_j^{d,i,0} = \mathbf{A} \cdot \mathbf{i}_j \begin{cases} = [0], & \text{za } j \neq k, m, \\ \neq [0], & \text{za } j = k, m, \end{cases} \quad (2.9)$$

Tabela 2.2 – Specifikacija – matematički opis prekida faze u trofaznom fiktivnom kolu sa prekidom između čvorova k i m

Tip prekida faze funkcije	Prekid jedne faze (faza L1)
H ₁	$i_{L1} = 0$
H ₂	$u_{L2k} - u_{L2m} = 0$
H ₃	$u_{L3k} - u_{L3m} = 0$

Fazori napona i injektiranih struja u čvorovima između kojih nastaje prekid faze transformišu se u simetrične komponente na sledeći način:

$$\begin{aligned} \mathbf{u}_k^{d,i,0} &= \mathbf{A} \cdot \mathbf{u}_k, \\ \mathbf{u}_m^{d,i,0} &= \mathbf{A} \cdot \mathbf{u}_m, \\ \mathbf{i}_k^{d,i,0} &= \mathbf{A} \cdot \mathbf{i}_k, \\ \mathbf{i}_m^{d,i,0} &= \mathbf{A} \cdot \mathbf{i}_m, \end{aligned} \quad (2.11)$$

3. MATEMATIČKI MODEL MREŽE

Neophodno je za svako potencijalno mesto prekida izračunati odgovarajuće ekvivalentne admitanse koje se “vide” iz tog čvora ka ostatku kola. Pri konstruisanju matematičkog modela potrebno je, dakle, proračunati potrebne ekvivalente, a tako izračunati ekvivalenti efikasno se koriste za proračun režima (struja i napona) u proizvoljno odabranom lokalnom kolu.

4. PRORAČUN REŽIMA SA PREKIDOM FAZA

U izlaganju koje sledi daju se odgovarajući izrazi i postupci za proračun režima sa prekidom faza, na osnovu matematičkog modela mreže.

4.1. Proračun režima sa prekidom faza u radijalnom kolu

Proračun režima sa prekidom faze se sastoji iz dva dela:

1. Proračun režima sa prekidom na mestu prekida,
2. Proračun režima u lokalnom kolu.

4.1.1. Proračun režima na mestu prekida

Kada se raspolaze sa ekvivalentnim admitansama koje se vide iz čvora ka korenu (y_u) i iz čvora ka poslednjem lejeru (y_d) može se izračunati režim na mestu prekida.

$$y_{pe}^x(m) = \frac{y_u^x(m) \cdot y_d^x(m)}{y_u^x(m) + y_d^x(m)}, \quad x \in \{d, i, 0\} \quad (4.1)$$

Simetrične komponente napona u čvoru m sa jedne i druge strane prekida se izračunavaju:

$$\begin{aligned} u_m^d &= i^d / y_u^d(m), \quad u_m^{d'} = -i^d / y_d^d(m), \\ u_m^i &= i^i / y_u^i(m), \quad u_m^{i'} = -i^i / y_d^i(m), \\ u_m^0 &= i^0 / y_u^0(m), \quad u_m^{0'} = -i^0 / y_d^0(m), \end{aligned} \quad (4.2)$$

Za zadati tip prekida u m-tom čvoru, vrednosti simetričnih komponenti struja na mestu prekida faze su funkcija struje na mestu prekida faze pre prekida i^{pre} , izračunate pri proračunu tokova snaga (izraz 4.3).

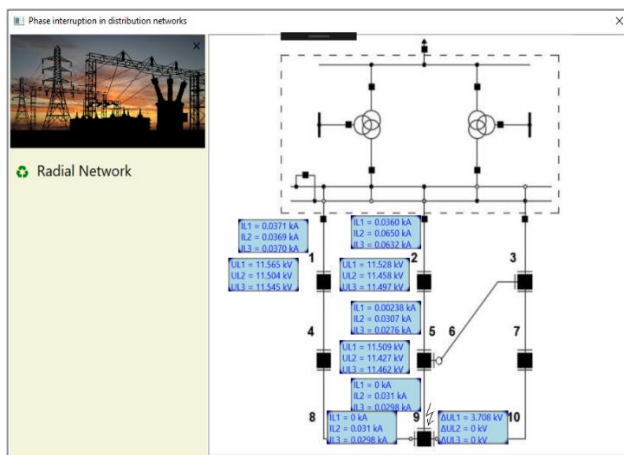
$$\begin{aligned} i^d &= \frac{z^i z^0}{z^d z^i + z^d z^0 + z^i z^0} \cdot i^{pre} \\ i^i &= \frac{z^d z^0}{z^d z^i + z^d z^0 + z^i z^0} \cdot i^{pre} \\ i^0 &= \frac{z^d z^i}{z^d z^i + z^d z^0 + z^i z^0} \cdot i^{pre} \end{aligned} \quad (4.3)$$

5. TEST PRIMER

U ovom delu teze prvo je dat opis test distributivne mreže, a zatim i primer proračuna prekida jedne faze (faze L1).

5.1. Opis test distributivne mreže

Razmatrana distributivna mreža prikazana je na slici 5.1. Ona se sastoji od jedne transformatorske stanice VN/SN₁/SN₂ (110/20/10 kV/kV/kV) i sedam transformatorskih stanica SN/NN (20/0.4 kV/kV). Sve transformatorske stanice su povezane kablovskim deonicama. Transformatori VN/SN₁/SN₂ su u paralelnom pogonu. Sekundari ovih transformatora su vezani za različite sabirnice, ali je spojno polje između tih sabirnica zatvoreno. Na slici je prikazan korisnički interfejs, program koji omogućava interakciju korisnika sa softverom. Moguće je zadati granu u kojoj se desio prekid faze L1, servis obradi proračun i na korisničkom interfejsu se vide rezultati. U softveru, u C# jeziku, je urađen prekid faze u deonici 9 i dati su rezultati u lokalnom kolu.



Slika 5.1 – Program kojim se računa prekid faze L1

Lokalno kolo čine čvorovi: 5, 2, 1 kao i grane: 9, 5, 2, 1. Kod mesta prekida koji se desio pri kraju grane 9, dati su rezultati struja i napona u sve tri faze. Vidi se da je struja u L1 fazi jednaka nuli, dok su naponi u zdravim fazama takođe nula kako je objašnjeno u teoriji iz glave 2. Simulacija poremećaja u softveru omogućava da se odrede vrednosti karakterističnih veličina kojima se opisuje rad distributivne mreže za vreme trajanja poremećaja.

6. ZAKLJUČAK

U radu je obrađen problem proračuna režima distributivne mreže sa prekidom faze. To je pre svega model za proračun režima u distributivnoj mreži koji se zasniva na proračunu ekvivalenata za svaki od čvorova mreže.

Radi se o veoma efikasnom modelu kojim se omogućuje generalno vrlo efikasan proračun, a zatim i vrlo efikasan proračun u proizvoljno izabranom delu mreže.

Matematički model je numerički potpuno stabilan, a vreme proračuna se linearno menja sa promenom dimenzije mreže.

Kada se proračun izvodi u radijalnim mrežama (a distributivne mreže su u najvećem broju slučajeva upravo radijalne) do izražaja dolazi još jedna inherentna pogodnost ovog modela, a to je da se kod proračuna režima sa prekidom faze ne mora uvoditi dodatni čvor na mestu prekida.

7. LITERATURA

- [1] N.Čupin: Spoj simetričnih sistema i strujne nesimetrije na transformatorima u slučaju prekida faza, *JUKO CIGRE*, Dubrovnik, 1970., ref. 41.29, str. 289-302.
- [2] E.Sterpin: Problem zaštite od uzdužnog kvara u izoliranim distributivnim zračnim mrežama, *Energija*, br. 9-10., 1973., str. 324-326.
- [3] V.Strezoski, G.Švenda, D.Bekut: Extension of the Canonical Model to Grounded Parts of Power System Under Fault Condition, *EPSR*, 2003.
- [4] D.Bekut, S.Kuzmanović: Proračun režima sa prekidom faza u radijalnoj distributivnoj mreži, *Elektroprivreda*, br. 4, 2002., str. 3-12.
- [5] D.Rajacic, R.Taleski: Two novel methods for radial and weakly-meshed network analysis, *Electric Power System Research*, 48, (1998), pp. 79-87

Kratka biografija:



Mina Lakić rođena je u Novom Sadu 1994. godine. Master rad je odbranila 2019. godine na Fakultetu tehničkih nauka iz Elektrotehnike i računarstva – Elektroenergetski sistemi

IMPLEMENTACIJA SERVERLESS APLIKACIJE ZA IOS UREĐAJE POMOĆU FIREBASE SERVISIA**IMPLEMENTATION OF SERVERLESS APPLICATION FOR IOS DEVICES WITH FIREBASE SERVICE**

Siniša Kolarević, *Fakultet tehničkih nauka, Novi Sad*

Oblast: RAČUNARSTVO I AUTOMATIKA

Kratak sadržaj – U radu je opisano rešenje implementacije serverless aplikacije za iOS uređaje pomoću Firebase servisa. Prezentovane su i analizirane mogućnosti Firebase servisa kao Baas (backend-as-a-service) usluge. Specificirana je i implementirana mobilna aplikacija za komentarisanje proizvoda identifikovanih barkodom za iOS uređaje pomoću usluga koje nudi Firebase Baas servis.

Ključne reči: Mobilne aplikacije, Serverless, iOS

Abstract – This paper describes the solution of implementation of serverless application for iOS devices using Firebase service. Firebase services are presented and analyzed as Baas (backend-as-a-service) services. A mobile application for commenting on products identified by barcodes has been specially designed and implemented for iOS devices using the services offered by Firebase Baas service.

Keywords: Mobile applications, Serverless, iOS

1. UVOD

U ovom radu opisan je dizajn i implementacija serverless [1] iOS [2] aplikacije za komentarisanje barkodova uz korišćenje Firebase Cloud [3] platforme. Takođe, opisani su i moduli koji komuniciraju u okviru sistema, njihove međusobne veze i odgovornosti. Na najvišem nivou apstrakcije mogu se definisati 2 modula u sistemu:

- iOS aplikacija (koju koristi krajnji korisnik)
- Firebase Cloud platforma – Cloud platforma koja služi za prijem, obradu i skladištenje podataka

1.1. Komunikacija između modula

Tok komunikacije između modula, na najvišem nivou apstrakcije, može se opisati na sledeći način:

- Krajnji korisnik inicira određenu akciju koristeći iOS aplikaciju
- iOS aplikacija (po potrebi) kontaktira Firebase Cloud platformu radi izvršenja date akcije

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Branko Milosavljević, red. prof.

- Firebase Cloud platforma prima zahtev od iOS aplikacije, procesira ga i vraća odgovor iOS aplikaciji
- iOS aplikacija prima odgovor od Firebase Cloud platforme, procesira ga i prikazuje odgovor na određenu akciju krajnjeg korisnika

2. SPECIFIKACIJA SISTEMA

U ovom poglavlju biće prikazana specifikacija sistema.

2.1. Funkcionalnosti iOS aplikacije

iOS aplikacija treba da obezbedi krajnjim korisnicima sledeće funkcionalnosti:

- Mogućnost kreiranja korisničkog naloga
- Mogućnost logovanja sa kreiranim korisničkim nalogom
- Mogućnost skeniranja proizvoljnog barkoda koristeći kameru uređaja
- Prikaz komentara za skenirani barkod
- Mogućnost komentarisanja skeniranog barkoda (sa ili bez slike)
- Mogućnost prikaza skeniranih barkodova i komentara za barkodove za korisnika
- Mogućnost podešavanja korisničkog imena
- Mogućnost podešavanja korisničke slike

2.2. Funkcionalnosti Firebase Authentication modula

Firebase Authentication modul treba da obezbedi sledeće funkcionalnosti:

- Mogućnost kreiranja korisničkih naloga
- Mogućnost logovanja korisnika
- Mogućnost čuvanja podataka o korisnicima
- Mogućnost izmene korisničkog imena
- Mogućnost izmene korisničke slike

2.3. Funkcionalnosti Firebase Realtime Database modula

Firebase Realtime Database baza podataka treba da obezbedi sledeće funkcionalnosti:

- Mogućnost čuvanja podataka o skeniranim barkodovima
- Mogućnost čuvanja podataka o kreiranim komentarima za skenirane barkodove

2.4. Funkcionalnosti Firebase Cloud Storage modula

Firebase Cloud Storage modul treba da obezbedi sledeće funkcionalnosti:

- Mogućnost čuvanja podataka o slikama za barkodove
- Mogućnost čuvanja podataka o slikama za korisnike

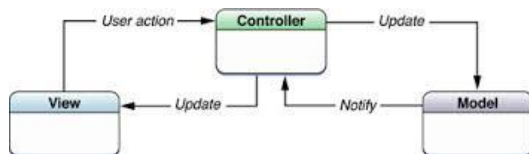
3. IMPLEMENTACIJA SISTEMA

U ovom poglavlju biće prikazana implementacija sistema.

3.1. Implementacija iOS aplikacije

iOS aplikacija je dizajnirana, implementirana i testirana u *Xcode* [4] razvojnom okruženju, koristeći *Swift* [5] programski jezik i *Cocoa Touch framework* [6]. *Xcode* je razvojno okruženje za razvoj aplikacija za iOS operativne sisteme. *Swift* je programski jezik u kome se pišu aplikacije za iOS operativne sisteme. *Cocoa Touch framework* je skup funkcionalnosti (skup klasa) koje dolaze sa *Xcode* razvojnim okruženjem, a koje se koriste za razvoj aplikacija za iOS operativne sisteme.

Pre prikaza implementacije ekrana iOS aplikacije biće objašnjen *MVC (Model, View, Controller)* [7] dizajn šablon koji se koristi u razvoju softvera, a posebno se koristi u razvoju aplikacija za iOS operativne sisteme. *MVC (Model, View, Controller)* dizajn šablon podrazumeva razdvajanje modela podataka od njihovog prikaza i rukovanja datim podacima. Na slici 3.1 prikazan je *MVC* dizajn šablon.



Slika 3.1. *MVC* dizajn šablon [7]

iOS aplikacija ima sledeće ekrane koji su na raspolaganju krajnjem korisniku:

- *Welcome* ekran
- *Login* ekran
- *Register* ekran
- *Scan barcode* ekran
- *No comments for barcode* ekran
- *Comments for barcode* ekran
- *Create comment for barcode* ekran
- *Feed* ekran
- *Menu* ekran
- *Housekeeping* ekran

3.2. Implementacija Firebase Authentication modula

Firebase Authentication modul obezbeđuje *backend* servise za autentifikaciju korisnika. Podržava autentifikaciju korišćenjem lozinki, brojeva telefona, kao i *Google*, *Facebook* i *Twitter* autentifikaciju.

Takođe se integriše i sa drugim *Firebase* servisima. U ovom radu korišćena je metoda autentifikacije putem *email*-a i lozinke.

3.3. Implementacija Firebase Realtime Database modula

Firebase Realtime Database modul obezbeđuje *NoSQL cloud-hosted* bazu podataka. Podaci se čuvaju u *JSON* formatu i sinhronizuju se u realnom vremenu. *Firebase Realtime Database* baza podataka se sastoji od 3 tipa entiteta: *topic*, *comment* i *feed*.

Topic predstavlja skenirani barcode i sastoji se od sledećih atributa:

- *code* - predstavlja barkod idnetifikator koji se očitava prilikom skeniranja barkoda

Comment predstavlja komentar za određeni barkod i sastoji se od sledećih atributa:

- *avatarUrl* – putanja do korisničke slike (ukoliko postoji) u okviru *Firebase Storage* modula
- *code* – barkod identifikator za koga je kreiran dati komentar
- *commentDate* – datum komentarisanja u *timestamp* formatu
- *commentPhotoUrl* – putanja do slike za komentar (ukoliko postoji) u okviru *Firebase Storage* modula
- *commentText* – tekst komentara
- *displayName* – korisničko ime
- *userId* – identifikator korisnika

Feed lista predstavlja listu skeniranih barkodova i komentara za barkodove za svakog korisnika i sastoji se od sledećih atributa:

- *feedType* – predstavlja tip feed elementa: Može imati 2 vrednosti: *COMMENT_CREATED* i *TOPIC_SCANNED*
- *timestamp* – datum kreiranja
- *topicCode* – barkod identifikator
- *commentText* – tekst komentara (ukoliko postoji). Ovo polje je popunjeno u slučaju kada je *feedType* tipa *COMMENT_CREATED*

3.4. Implementacija Firebase Cloud Storage modula

Firebase Cloud Storage obezbeđuje skladištenje i serviranje sadržaja kao što su slike ili video zapisi. U okviru implementacije ovog rada, *Firebase Cloud Storage* se koristi za skladištenje i serviranje slika za komentare korisnike.

4. ZAKLJUČAK

U radu je predstavljeno rešenje implementacije iOS aplikacije za komentarisanje barkodova korišćenjem *Firebase Cloud* platforme.

U rešenju koje je opisano u okviru ovog rada, iOS aplikacija se integriše sa 3 *Firebase* modula:

- *Firebase Authentication*
- *Firebase Realtime Database*
- *Firebase Cloud Storage*

Firebase Cloud platforma pored navedenih modula nudi još i mnogo drugih korisnih modula, kao što su:

- *Firebase Cloud Functions*
- *Firebase ML Kit*
- *Firebase Hosting*
- *Firebase Crashlytics*
- *Firebase Performance Monitoring*
- *Firebase Google Analytics*
- *Firebase Cloud Messaging*
- *Firebase Remote Config*

Jedan od glavnih ciljeva *Firebase Cloud* platforme je brži razvoj mobilnih (i *Web*) aplikacija, bez potrebe vođenja računa o infrastrukturi. *Firebase Cloud* platforma je ustvari *BaaS (Backend-as-a-Service)* platforma.

Generalno govoreći, *Web* i mobilne aplikacije zahtijevaju sličan skup serverskih funkcija uključujući čuvanje podataka, obradu podataka, notifikacije, integraciju sa društvenim mrežama itd. Svaki od navedenih serverskih servisa se mora posebno integrisati za svaku od aplikacija, što je proces koji je istovremeno dugotrajan a može biti i relativno komplikovan, u zavisnosti od servisa koji se integriše. *BaaS (Backend-as-a-Service)* platforme, odnosno, *Firebase Cloud* platforma, kao jedna od instanci *BaaS (Backend-as-a-Service)* platformi, pokušava da reši dati problem tako što na uniforman način spaja serverske funkcije sa mobilnim i *Web* aplikacijama.

BaaS (Backend-as-a-Service) platforme predstavljaju relativno skorašnji razvoj u okviru *Cloud Computing* oblasti računarstva, sa većinom platformi pokrenutih od 2011 godine i kasnije. Iako još uvek mlada industrija, trendovi do sada pokazuju da se usluge *BaaS (Backend-as-a-Service)* platformi sve više i više koriste, razvijaju i dobijaju na značaju.

5. LITERATURA

- [1] Serverless - https://en.wikipedia.org/wiki/Serverless_computing (Serverless computing allows you to build and run applications and services without thinking about servers)
- [2] iOS - <https://en.wikipedia.org/wiki/iOS> (iOS is a mobile operating system created and developed by Apple Inc. exclusively for its hardware)
- [3] *Firebase Cloud* - <https://firebase.google.com/> (Firebase helps mobile app teams succeed. Firebase is a Backend-as-a-Service - BaaS. Firebase is your server, your API and your datastore)
- [4] *Xcode* - <http://en.wikipedia.org/wiki/Xcode> (Xcode is an integrated development environment (IDE) for macOS containing a suite of software development tools developed by Apple for developing software for macOS, iOS, watchOS, and tvOS)
- [5] *Swift* - <https://developer.apple.com/swift/> (Swift is a powerful and intuitive programming language for macOS, iOS, watchOS and tvOS)
- [6] *Cocoa Touch Framework* - <https://developer.apple.com/library/archive/documentation/General/Conceptual/DevPedia-CocoaCore/Cocoa.html> (Cocoa and Cocoa Touch are the application development environments for OS X and iOS, respectively)
- [7] *MVC (Model-View-Controller) Design Pattern* - <http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller> (Model–view–controller is an architectural pattern commonly used for developing user interfaces that divides an application into three interconnected parts)

Kratka biografija:



Siniša Kolarević rođen je u Vukovaru 1986. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Internet Softverske Arhitekture odbranio je 2019.god. kontakt: sinisa.kolarevic@gmail.com

VIŠEKRITERIJUMSKO ODLUČIVANJE U PLANIRANJU RAZVOJA DISTRIBUTIVNIH MREŽA**MULTIPLE ATTRIBUTE DECISION MAKING IN DEVELOPMENT PLANNING OF DISTRIBUTION NETWORKS**

Tamara Žepinić, Fakultet tehničkih nauka, Novi Sad

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu su implementirane metode za višekriterijumsko odlučivanje VIKOR i TOPSIS na primeru ekonomske analize u planiranju razvoja distributivne mreže. Nakon implementacije, napravljeno je poređenje metoda na osnovu analize dobijenih rezultata i njihovih matematičkih osnova.

Ključne reči: Planiranje razvoja distributivnih mreža, višekriterijumska optimizacija i odlučivanje

Abstract – In this paper, the methods for multiple attribute decision making VIKOR and TOPSIS are implemented in the case of economic analysis in the development planning of distribution network. After implementation, a comparison was made between the methods based on the analysis of the obtained results and their mathematical bases.

Keywords: Development planning of electricity distribution networks, multiple attribute decision making

1. UVOD

Ciljevi planiranja razvoja distributivne mreže uključuju minimizaciju gubitaka energije i potrebnih investicija (troškova ulaganja), povećanje pouzdanosti i lične sigurnosti i dobar kvalitet električne energije, kao i očuvanje životne sredine [1]. U ovom radu je razmatrano planiranje razvoja distributivne mreže sa ciljem povećanja pouzdanosti uz minimizaciju potrebnih investicija. Pouzdanost odnosno neprekidnost napajanja sa stanovišta korisnika najčešće predstavlja najznačajniji tehnički aspekt kvaliteta električne energije.

Pouzdanost se najčešće meri učestanošću i dužinom trajanja prekida napajanja korisnika, odnosno indeksima (pokazateljima) pouzdanosti SAIFI i SAIDI. Uobičajene strategije korišćene za unapređenje pouzdanosti su smanjenje broja kvarova u mreži, smanjenje dužine trajanja prekida u napajanju i smanjenje broja korisnika pogođenih kvarovima. Indeksi pouzdanosti SAIDI i SAIFI uvek su bili među najvažnijim pokazateljima koji su se koristili kod planiranja izgradnje i unapređenja distributivne mreže. Postoji više različitih strategija za unapređenje ovih pokazatelja pouzdanosti [2]. Jedna od strategija za njihovo unapređenje je i smanjenje broja kvarova kroz smanjenje intenziteta otkaza elemenata distributivnih

mreža. Ova strategija je i razmatrana u ovom radu. Ona je zasnovana na zameni postojećih sekcija mreže sa novim sekcijama koje imaju manji intenzitet otkaza (λ). U cilju određivanja najboljeg rešenja, odnosno skupa postojećih sekcija koje treba zameniti novim tako da se minimiziraju SAIDI, SAIFI i investicioni troškovi, potrebno je definisati skup većeg broja različitih alternativa zamene postojećih sekcija. Zatim, na osnovu navedenog kriterijuma treba izabrati najbolju varijantu iz skupa generisanih alternativa. Sve alternative zamene postojećih sekcija će biti poređene po nekoliko kriterijuma i to vrednost investicije [n.j.], vrednost indeksa pouzdanosti SAIDI [$\frac{h}{\text{god}}$] i vrednost indeksa pouzdanosti SAIFI [$\frac{\text{kvar}}{\text{god}}$].

Pošto su svi kriterijumi izraženi u različitim jedinicama, potrebno je koristiti optimizaciju po više kriterijuma, odnosno višekriterijumsku optimizaciju. Da bi se rešio zadatak izbora najboljeg rešenja iz skupa alternativa, razvijene su metode za višekriterijumsku optimizaciju. U nastavku će detaljnije biti objašnjene metode VIKOR i TOPSIS koje se koriste za određivanje višekriterijumski optimalnog rešenja i koje će biti implementirane na primeru višekriterijumskog odlučivanja u radu.

2. VIŠEKRITERIJUMSKO ODLUČIVANJE I KOMPROMISNO REŠENJE

Zadatak višekriterijumskog odlučivanja može se formulisati na sledeći način [3]:

$$vko_{a \in A}(f_1(a), f_2(a), \dots, f_n(a)), \quad (1)$$

gde je: A - skup dopustivih (potencijalnih) alternativa, $a = (x_1, x_2, \dots)$ - alternativa dobijena za određene vrednosti promenljivih (x) sistema, f_i - i -ta kriterijumska funkcija, vko - operator za višekriterijumsko optimalno rešenje.

Metode za višekriterijumsku optimizaciju koje se koriste za rešavanje zadataka izbora najboljeg rešenja iz datog (konačnog) skupa alternativa $\{a_1, a_2, \dots, a_j\}$ često imaju i zadatak da odrede redosled (rang-listu) sa stanovišta svih datih kriterijuma. Ako postoji rešenje koje istovremeno maksimizira/minimizira sve kriterijumske funkcije, onda je ono optimalno rešenje postavljenog zadatka. Međutim, takvo rešenje retko postoji u realnim (praktičnim) problemima višekriterijumske optimizacije, pa se mora usvojiti jedno rešenje kao najbolje.

Metode za višekriterijumsku optimizaciju VIKOR i TOPSIS su nastale kao verzije metode kompromisnog programiranja kojom se određuje skup kompromisnih rešenja. Kompromisno rešenje je najbliže idealnom prema

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Željko Popović.

datoj meri rastojanja. Obe metode zahtevaju da svaka alternativa bude vrednovana po svim kriterijumima, pa se vrednost i -te kriterijumske funkcije za alternativu a_j može označiti sa f_{ij} . Neophodno je da se traži maksimum ako i -ta kriterijumska funkcija predstavlja korist ili dobit, ili minimum za štete ili troškove.

Takođe, obe metode koriste normalizovane vrednosti težina kriterijumskih funkcija w_i , $i = 1, \dots, n$ koje su neophodne jer se ne može pretpostaviti da sve kriterijumske funkcije imaju jednaku važnost, pri čemu mora važiti da je $\sum w_i = 1$, $w_i \geq 0, \forall i$. Težina uz i -tu kriterijumsku funkciju može imati veću ili manju vrednost u odnosu na zadatu početnu vrednost w_i , ali promenom jedne težine (w_i) moraju se promeniti i ostale težine tako da i dalje važi da je $\sum w_i = 1$, $w_i \geq 0, \forall i$. Konačnu odluku o tome koja alternativa će biti izabrana donosi donosilac odluke (nadležno i odgovorno lice za donošenje konačne odluke, odnosno za usvajanje konačnog, optimalnog rešenja) [3].

2.1. Metoda VIKOR

Metoda VIKOR za Višekriterijumsko Kompromisno Rangiranje ili (Višekriterijumsko Kompromisno Rešenje) [3] zahteva da su poznate vrednosti svih kriterijumskih funkcija za sve alternative, u vidu matrice $|f_{ij}|_{n \times J}$. Metoda VIKOR određuje kompromisno rešenje koje može biti jedna alternativa ili skup alternativa, za date (ulazne) vrednosti težina kriterijuma w_i , $i = 1, \dots, n$. Za određivanje kompromisne rang-liste treba najpre odrediti meru S_j kojom se određuje "najbolja" alternativa koja zadovoljava "većinu" (sumu) kriterijuma, kao i meru R_j kojom se određuje alternativa koja nije "izrazito loša" prema nekom kriterijumu. Pomenute mere se određuju za sve analizirane alternative a_j , $j = 1, \dots, J$ pomoću sledećih relacija:

$$S_j = \sum_{i=1}^n w_i (f_i^* - f_{ij}) / (f_i^* - f_i^-), \quad (2)$$

$$R_j = \max_i [w_i (f_i^* - f_{ij}) / (f_i^* - f_i^-)], \quad (3)$$

gde je: n - broj kriterijuma, w_i - težina i -te kriterijumske funkcije, f_{ij} - vrednost i -te kriterijumske funkcije za j -tu alternativu, f_i^* - najbolja alternativa svakog i -tog kriterijuma, f_i^- najlošija alternativa svakog i -tog kriterijuma. Rangiranje se vrši sortiranjem alternativa prema vrednostima mere Q_j koja predstavlja linearnu funkciju težine strategije "zadovoljavanja većine kriterijuma" i određuje se pomoću sledeće relacije:

$$Q_j = v \frac{(S_j - S^*)}{(S^- - S^*)} + (1 - v) \frac{(R_j - R^*)}{(R^- - R^*)}, \quad (4)$$

gde je: v - vrednost težine koja zavisi od broja kriterijuma (n) i u primeru koji sledi usvojeno je da je $v = 0.5$ za $n \leq 4$, $S^* = \min_j S_j$, $S^- = \max_j S_j$, $R^* = \min_j R_j$ i $R^- = \max_j R_j$. Najbolja alternativa je ona za koju je vrednost mere najmanja i ona zauzima prvo mesto na rang-listi. Metoda VIKOR predlaže kao višekriterijumski najbolju alternativu (za date težine w_i) onu koja je na prvoj poziciji na kompromisnoj rang-listi samo ukoliko ima "dovoljnu prednost" nad alternativom sa sledeće pozicije. Za vrednovanje "prednosti" koristi se razlika između mera Q_j . Alternativa a' ima dovoljnu prednost nad sledećom a'' sa rang-liste ako je:

$$Q(a'') - Q(a') \geq DQ, \quad (5)$$

gde je DQ "prag prednosti" koji se određuje kao:

$$DQ = \min(0.25; 1/(J - 1)), \quad (6)$$

gde je J ukupan broj analiziranih alternativa.

Sa 0.25 je ograničen prag za slučajeve sa malim brojem alternativa. Uslov "dovoljne prednosti" omogućava da se donosiocu odluke prikažu sve alternative koje su "bliske" u višekriterijumskom smislu. Ne bi bilo opravdano da se donosiocu odluke prikaže samo alternativa sa prve pozicije na rang-listi prema meri Q_j i da se zanemare alternative koje imaju njoj "bliske" vrednosti mere Q_j . Ukoliko prva alternativa sa kompromisne rang-liste ne ispunjava taj uslov tada skup kompromisnih rešenja sadrži alternative sa kompromisne rang-liste a' i a'' , ..., $a^{(K)}$ za koje je $Q(a^{(K)}) - Q(a') < DQ$, kao što će biti pokazano u razmatranom primeru u radu [3].

2.2. Metoda TOPSIS

Metoda TOPSIS (Technique for Order Preference by Similiarity to Ideal Solution) [4] je još jedna verzija proširenja kompromisnog programiranja. Ova metoda je zasnovana na konceptu da izabrana alternativa treba da ima najkraću Euklidovu udaljenost od "idealnog" rešenja i najdalju Euklidovu udaljenost od "anti-idealnog" rešenja, pa se na taj način uvode dve referentne tačke: "idealno rešenje" i "anti-idealno rešenje". Metoda TOPSIS zahteva da su poznate vrednosti svih kriterijumskih funkcija f_i iz F vektorske kriterijumske funkcije, $F = (f_1, f_2, \dots, f_n)$ u vidu matrice $|f_{ij}|_{n \times J}$ za sve alternative iz skupa dopustivih (potencijalnih) alternativa $A = \{a_1, a_2, \dots, a_J\}$. Metoda TOPSIS na kraju svog algoritma uvek pruža jedinstveno kompromisno rešenje tako što se formiranjem rang-liste svih alternativa na osnovu vrednosti mere C_j^* bira najbolja alternativa sa najvećom vrednošću C_j^* . Mera C_j^* proračunava relativnu blizinu "idealnom rešenju" svake alternative a_j , $j = 1, \dots, J$ i računa se kao:

$$C_j^* = S_j^- / (S_j^* + S_j^-), \quad (7)$$

gde je: S_j^* - Euklidova udaljenost od "idealnog rešenja", S_j^- - Euklidova udaljenost od "anti-idealnog rešenja". S_j^* se računa kao:

$$S_j^* = \sqrt{\sum_{i=1}^n \left[w_i \frac{f_{ij}}{\sqrt{\sum_{j=1}^J f_{ij}^2}} - v_i^* \right]^2}, \quad (8)$$

gde je: n - broj kriterijuma, w_i - težina i -te kriterijumske funkcije, f_{ij} - vrednost i -te kriterijumske funkcije za j -tu

alternativu, $v_i^* = \max_j \left[w_i \frac{f_{ij}}{\sqrt{\sum_{j=1}^J f_{ij}^2}} \right]$ ako kriterijumska

funkcija predstavlja korist ili $v_i^* = \min_j \left[w_i \frac{f_{ij}}{\sqrt{\sum_{j=1}^J f_{ij}^2}} \right]$ ako

kriterijumska funkcija predstavlja štetu. S_j^- se računa kao:

$$S_j^- = \sqrt{\sum_{i=1}^n \left[w_i \frac{f_{ij}}{\sqrt{\sum_{j=1}^J f_{ij}^2}} - v_i^- \right]^2}, \quad (9)$$

gde je : $v_i^* = \min_j \left[w_i \frac{f_{ij}}{\sqrt{\sum_{j=1}^J f_{ij}^2}} \right]$ ako kriterijumska funkcija

predstavlja korist ili $v_i^* = \max_j \left[w_i \frac{f_{ij}}{\sqrt{\sum_{j=1}^J f_{ij}^2}} \right]$ ako kriterijumska funkcija predstavlja štetu.

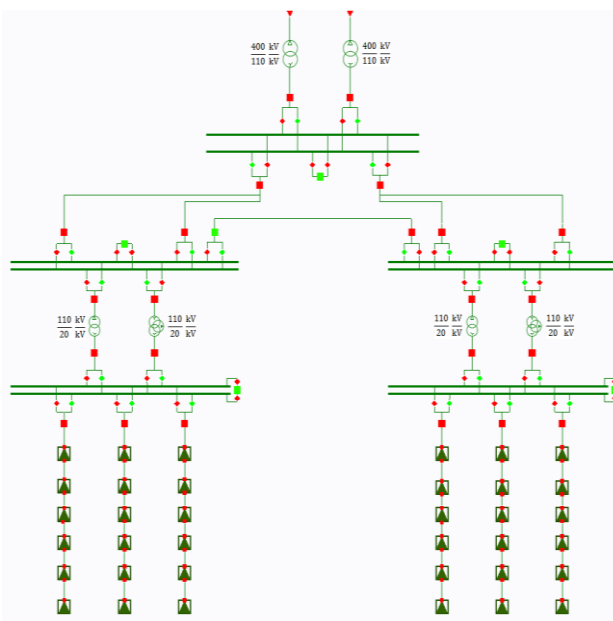
3. PRIMENA METODA VIKOR I TOPSIS

U ovom delu rada je prikazana primena metoda VIKOR i TOPSIS na primeru ekonomske analize u planiranju razvoja distributivnih mreža. U razmatranom primeru je primenjena uobičajena strategija za unapređenje pouzdanosti napajanja i to smanjenje broja kvarova u mreži zamenom postojećih sekcija mreže sa novim sekcijama koje imaju manji intenzitet otkaza (λ). Problem višekriterijumske optimizacije se u ovome radu rešava korišćenjem heurističkog pristupa. Ovaj pristup definiše skup heurističkih pravila za određivanje skupa sekcija mreže koje bi bile zamenjene novim sekcijama koje imaju manji intenzitet otkaza (λ). Heuristička pravila koja će biti primenjena u nastavku su:

1. Zamena svih sekcija mreže;
2. Zamena svih sekcija najkritičnijeg/ih izvoda. Pod najkritičnijim izvodima se smatraju svi izvodi koji sadrže najveći broj sekcija čiji je intenzitet otkaza $\lambda \geq \lambda_{prosečno}$, gde je $\lambda_{prosečno}$ prosečni intenzitet otkaza svih sekcija mreže;
3. Zamena najkritičnijih sekcija jednog ili više izvoda. Pod najkritičnijim sekcijama se smatraju sve sekcije čiji je intenzitet otkaza $\lambda \geq \lambda_{prosečno}$, gde je $\lambda_{prosečno}$ prosečni intenzitet otkaza svih sekcija mreže;
4. Zamena svih najkritičnijih sekcija mreže. Pod najkritičnijim sekcijama se smatraju sve sekcije čiji je intenzitet otkaza $\lambda = \lambda_{najgore}$, gde je $\lambda_{najgore}$ najveći zabeleženi intenzitet otkaza sekcije λ u mreži;
5. Zamena sekcije svakog izvoda koja je najbliža izvoru napajanja, pod uslovom da je intenzitet otkaza te sekcije $\lambda \geq \frac{\lambda_{najgore}}{2}$, gde je $\lambda_{najgore}$ najveći zabeleženi intenzitet otkaza sekcije λ u mreži;
6. Zamena sekcije svakog izvoda koja je najdalja od izvora napajanja, pod uslovom da je intenzitet otkaza te sekcije $\lambda \geq \frac{\lambda_{najgore}}{2}$, gde je $\lambda_{najgore}$ najveći zabeleženi intenzitet otkaza sekcije λ u mreži;
7. Zamena svih sekcija čija je dužina $l \geq l_{prosečno}$ i čiji je intenzitet otkaza $\lambda \geq \lambda_{prosečno}$, gde je $l_{prosečno}$ prosečna dužina svih sekcija mreže, a $\lambda_{prosečno}$ prosečni intenzitet otkaza svih sekcija mreže.

Po uzoru na ova heuristička pravila na primeru test mreže sa slike 1 definisano je 10 različitih alternativa zamene postojećih sekcija, te je potrebno izabrati najbolju tako što će se sve alternative porediti po nekoliko kriterijuma. Zamena starih sekcija novim sekcijama je simulirana promenom intenziteta otkaza sa stare vrednosti na $0.1 \frac{\text{kvar}}{\text{god}}$. U mreži postoji šest izvoda sa različitim dužinama sekcija, tako da su ukupne dužine izvoda u opsegu 2.5 – 7 km, dok su intenziteti otkaza sekcija u opsegu 0.01 – 0.06

$\frac{\text{kvar}}{\text{god}}$. Broj potrošača na svakom izvodu je između 100 i 700.



Slika 1. Test mreža

3.1. Razmatrani kriterijumi

Sve alternative se vrednuju na osnovu tri kriterijuma, odnosno kriterijumske funkcije:

1. Vrednost investicije [n.j.].
2. Vrednost indeksa pouzdanosti SAIDI $\left[\frac{h}{\text{god}} \right]$.
3. Vrednost indeksa pouzdanosti SAIFI $\left[\frac{\text{kvar}}{\text{god}} \right]$.

Na osnovu prethodno definisanih kriterijumskih funkcija, vidi se da njihove vrednosti nisu izražene u istim jedinicama mere, zbog čega je neophodno koristiti metode višekriterijumskog odlučivanja, VIKOR i TOPSIS. Za proračune je potrebno imati vrednosti težina kriterijuma i u ovom primeru svim kriterijumima su dodeljene iste težine u vrednosti od 0.33. Finalna VIKOR rang-lista prema meri Q_j za prve dve rangirane alternative prikazana je u tabeli 1.

Tabela 1. Rang-lista VIKOR prema meri Q_j

Pozicija na rang-listi	Alternativa	Q_j
1.	5.	0.060
2.	4.	0.145

Na osnovu podataka iz tabele 1 može se izračunati “prednost” alternative koja zauzima prvo mesto na kompromisnoj rang-listi nad alternativom koja zauzima drugo mesto. Ta “prednost” iznosi 0.085, pa se može doneti zaključak da prva alternativa ne ispunjava uslov “dovoljne prednosti” nad alternativom sa sledeće pozicije rang-liste jer “prag prednosti” koji iznosi 0.11 nije postignut. Zbog toga se metodom VIKOR dobija skup kompromisnih rešenja u koji ulaze i prva i druga alternativa sa kompromisne rang-liste koje se zajedno predlažu donosiocu odluke. Prva alternativa sa kompromisne rang-liste predstavlja zamenu svih sekcija

čiji su intenziteti otkaza 0.6 ili $0.5 \frac{\text{kvar}}{\text{god}}$. Ova alternativa je definisana po uzoru na četvrto predloženo heurističko pravilo. Druga alternativa sa kompromisne rang-liste predstavlja zamenu po tri najkritičnije sekcije svakog izvoda. Ova alternativa je definisana po uzoru na treće predloženo heurističko pravilo.

Finalna TOPSIS rang-lista prema meri C_j^* za prve dve rangirane alternative prikazana je u tabeli 2.

Tabela 2. Rang-lista TOPSIS prema meri C_j^*

Pozicija na rang-listi	Alternativa	C_j^*
1.	5.	0.651
2.	10.	0.590

Metodom TOPSIS je alternativa koja zauzima prvo mesto na kompromisnoj rang-listi određena kao jedinstveno kompromisno rešenje i to je alternativa zamene svih sekcija čiji su intenziteti otkaza 0.6 ili $0.5 \frac{\text{kvar}}{\text{god}}$.

Kao što je napomenuto u 2. glavi, težina uz i -tu kriterijumsku funkciju tj. kriterijum može imati veću ili manju vrednost u odnosu na zadatu početnu vrednost w_i , ali promenom jedne težine (w_i) moraju se promeniti i ostale težine tako da važi da je $\sum w_i = 1$, $w_i \geq 0, \forall i$. U tabeli 3 su prikazane rang-liste koje su dobijene primenom metoda VIKOR i TOPSIS za nove vrednosti težina kriterijuma, tako da je prvom kriterijumu dodeljena težina od 0.2, a drugom i trećem kriterijumu od 0.4. U rang-listama su prikazane samo pozicije alternativa koje su bile najbolje rangirane u inicijalnom slučaju kada su sve težine kriterijuma bile jednake. Pozicije alternativa na rang-listama koje su se promenile u odnosu na taj slučaj obeležene su podebljanim brojevima.

Tabela 3. Rang-liste VIKOR i TOPSIS za nove vrednosti težina kriterijuma

Alternativa	Pozicija na rang-listi VIKOR	Pozicija na rang-listi TOPSIS
4.	3	3
5.	5	2
10.	9	7

3.2. Diskusija dobijenog kompromisnog rešenja

Analizom dobijenih rezultata, može se uočiti da su obe metode odredile kao kompromisno rešenje petu alternativu, odnosno zamenu svih sekcija čiji su intenziteti otkaza 0.6 ili $0.5 \frac{\text{kvar}}{\text{god}}$. Međutim, metodom VIKOR je određen skup kompromisnih rešenja u koji pored pomenute alternative ulazi i alternativa koja zauzima drugo mesto na kompromisnoj rang-listi. Donosilac odluke treba da izabere, na osnovu njegovih preferencija, jednu od te dve alternative kao konačno rešenje. Za razliku od metode VIKOR, metoda TOPSIS je na kraju svog algoritma pružila jedinstveno kompromisno rešenje. Na osnovu podataka iz tabele 3 vidi se da se promenom težina kriterijumskih funkcija dobijaju bitno drugačije

rang-liste alternativa u odnosu na prvobitne, što ukazuje na to koliko su metode VIKOR i TOPSIS osetljive na promenu vrednosti težina kriterijumskih funkcija.

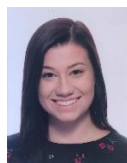
4. ZAKLJUČAK

Na osnovu izloženih razmatranja, može se zaključiti da primena metode VIKOR predstavlja bolji izbor za višekriterijumsko odlučivanje, jer ova metoda omogućava da se donosiocu odluke prikaže više alternativa koje pripadaju skupu kompromisnih rešenja kada prva alternativa sa rang-liste nije "dovoljno" bolja od ostalih, kao što je to bio slučaj u primeru višekriterijumskog odlučivanja prikazanom u ovom radu. Sa druge strane, metodom TOPSIS se uvek dobija jedinstveno kompromisno rešenje. Alternativa koja je definisana po uzoru na četvrto predloženo heurističko pravilo je određena kao kompromisno rešenje, pa se heurističko pravilo zamene svih najkritičnijih sekcija mreže predlaže kao najbolje pravilo za unapređenje pouzdanosti. Pošto su sve alternative definisane po uzoru na nekoliko predloženih heurističkih pravila, predmet daljeg istraživanja može biti unapređenje tih heurističkih pravila. Na kraju primera je pokazano da se promenom težina kriterijumskih funkcija dobijaju bitno drugačije rang-liste alternativa primenom metoda VIKOR i TOPSIS, što ukazuje na to koliko su te metode osetljive na promenu vrednosti težina kriterijumskih funkcija. Zbog navedenog je važno uraditi analizu težina i stabilnosti kompromisnog rešenja sa promenom težina. Ispitivanje ostalih metoda za višekriterijumsku optimizaciju je svakako od veoma velike važnosti i predstavlja predmet daljeg istraživanja.

5. LITERATURA

- [1] Viktoria Neimane: *On development planning of electricity distribution networks*; Doctoral Dissertation, Royal Institute of Technology, Stockholm, 2001.
- [2] Ž. Popović, S. Knežević, B. Brbaklić: *Optimalna strategija za unapređenje pouzdanosti u distributivnim mrežama*; Zbornik radova sa desetog savetovanja o elektrodistributivnim mrežama CIREC Srbija, Vrnjačka Banja, R-3.01, 26.-30. Septembar 2016.
- [3] Serafim Opricović: *Višekriterijumska optimizacija sistema u građevinarstvu*; Građevinski fakultet Univerziteta u Beogradu, Beograd, 1998.
- [4] Ching-Lai Hwang, Kwangsun Yoon: *Multiple Attribute Decision Making- Methods and Applications*; Springer-Verlag, Berlin, 1981.
- [5] Gwo-Hshiung Tzeng, Jih-Jeng Huang: *Multiple Attribute Decision Making- Methods and Applications*; CRC Press, London, 2011.

Kratka biografija:



Tamara Žepinić rođena je u Bijeljini 1994. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranila je 2017.god. i 2017. god. upisala je master studije na istom fakultetu.

РЕАЛИЗАЦИЈА БЕЖИЧНОГ АКУСТИЧНОГ СЕНЗОРА БАЗИРАНА НА БЛУТУТ ТЕХНОЛОГИЈИ**IMPLEMENTATION OF A WIRELESS ACOUSTIC SENSOR BASED ON BLUETOOTH TECHNOLOGY**

Александар-Вук Павловић, Мирослав Поповић, *Факултет техничких наука, Нови Сад*

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – Развој технологије довео је до повећања перформанси и пада цена електронских уређаја, што је проузроковало појаву Интернета Ствари – велики број умрежених уграђених уређаја чија је примена побољшање квалитета живота људи. Ови уређаји се уграђују и у домове, доводећи до појаве паметних кућа. Великом брзином, главна спрега комуникације са уређајима у паметној кући постаје паметни звучник. Уколико је корисник удаљен од уређаја, квалитет комуникације опада због деформација звука. Претпоставља се да се квалитет комуникације може побољшати уградњом уређаја са микрофоном унутар просторије. У овом раду је представљена реализација оваквог уређаја заснована на Блутут технологији. Испитана је потрошња електричне енергије и квалитет везе на различитим удаљеностима и добијени су задовољавајући резултати.

Кључне речи: Звук, Блутут, Бежична комуникација, Микрофон, Сензор

Abstract – Continued development of technology led to an increase in performance and decrease in price of electronic devices, which further led to Internet of Things – a term used to describe a multitude of connected embedded devices designed to improve quality of life. These devices can also be found in homes, giving rise to smart homes. Increasingly, the main communication interface in a smart home is a smart speaker. If the user is further away from the device, the quality of communication suffers due to sound deformation. An assumption is made that the quality of communication can be improved by placing devices that contain a microphone inside a room. This thesis presents an implementation of such a device based on Bluetooth technology. Power consumption and connection quality at different distances were measured and tested and acceptable results were obtained.

Keywords: Sound, Bluetooth, Wireless Communication, Microphone, Sensor

1. УВОД

Развој технологије и електронике довео је до повећања перформанси електронских компонената, смањења њихове физичке величине и пада њихових цена. Ово је проузроковало настанак нове брзоразвијајуће парадигме – Интернета ствари (енгл. IoT). Интернет ствари представља умрежавање уграђених и наменских уређаја у циљу постизања комплекснијих захтева које појединачни уређаји нису у стању да подрже. Често се проналази примена оваквих „ствари“ у виду уређаја потрошачке електронике који имају за циљ побољшање квалитета живота људи. Овакви уређаји се све чешће могу пронаћи у домовима, тј. у тзв. паметним кућама.

Великом брзином, главна спрега комуникације са уређајима у паметној кући постаје такозвани паметни звучник – уређај који је у стању да прима гласовне наредбе од корисника, проследи их у одговарајућем формату другим уређајима и кориснику да повратну информацију, у виду синтетизованог људског гласа [1]. Оваква спрега поседује једну битну ману, а то је да спрега између корисника и уређаја не постоји уколико корисник и уређај нису у могућности да се међусобно чују. У овом раду претпостављамо да се поменути проблем може решити постављањем већег броја уређаја са микрофоном унутар просторије који би били спојени са паметним звучником. Представимо мотивацију за избором Блутут технологије и ESP развојне плоче, идеју и реализацију алгорита за управљање поменутим уређајима и резултате испитивања карактеристика решења.

2. МОТИВАЦИЈА И РЕАЛИЗАЦИЈА

Већ је поменуто да је битна мана гласовне комуникације са паметним звучником постојање ситуација у којој корисник и уређај нису у могућности да се међусобно чују. Удаљеност (када се микрофон паметног звучника налази у удаљеном пољу у односу на извор звука, тј. корисника), бука, акустичне карактеристике просторије и други фактори који се одликују разноврсним звучним деформацијама имају велики утицај на квалитет спреге. Претпоставили смо да се овај проблем може ублажити или решити постављањем већег броја јефтиних нископотрошних уређаја опремљених микрофоном унутар једне и/или више просторија. Коришћењем алгорита за избор, микрофон на

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био проф. др Мирослав Поповић.

најбољој позицији да чује корисника (микрофон који се налази у блиском пољу корисника) би био постављен као активан и звучни сигнал би од њега био пренет до паметног звучника. Пратећи ову претпоставку и ослањајући се на нека претходна истраживања [2] [3], осмислили смо поменути алгоритам и одлучили се да га имплементирамо на ESP развојној плочи, ослањајући се на Блутут технологију за пренос команди и звучног сигнала. Избор ESP развојне плоче је био мотивисан њеном великом доступношћу, постојањем интегрисаног Блутут контролера и малом ценом, док је избор Блутут технологије био мотивисан позданошћу преноса и ниском потрошњом електричне енергије.

Алгоритам смо реализовали кроз клијент-послужилац архитектуру са већим бројем послужилаца и једним клијентом. Сваки извор звука представља један послужилац, док пријемник представља клијента.

Сваки послужилац се понаша на идентичан начин, тако што се прво учини видљивим другим Блутут уређајима, а затим чека захтеве од стране клијента. Ако добије захтев од клијента, повезује се са њим коришћењем Блутут технологије и шаље вредности звучног сигнала на клијента.

Клијент се понаша тако што прво претражи све видљиве Блутут уређаје и формира листу свих доступних послужилаца. Уколико у задатом временском интервалу није пронашао ниједног послуживоца, наставља претрагу. Након завршетка тражења, клијент се повезује са послуживоцем коришћењем Блутут технологије и са њега преузима вредности звучног сигнала уколико послужилац назначи да корисник тренутно издаје говорну наредбу. Када клијент закључи да је корисник престао да издаје наредбу, престаје да преузима вредности звучног сигнала и поново почиње да проверава код свих послужилаца да ли је корисник кренуо да издаје говорну наредбу. Овај процес се понавља непрестано.

Фокус овог рада је био на испитивању квалитета Блутут везе између послуживоца и клијента и испитивању потрошње електричне енергије. С тим на уму, уместо реалних гласовних наредби од стране корисника, користили смо апстрактнији вид активирања преноса између послуживоца и клијента – псеудонасумичне бројеве који су били интерпретирани као ознака која је клијента упућивала да ли да преузима звучни садржај са послуживоца или не. Ово је олакшало испитивање послуживоца и клијента.

Алгоритам смо имплементирали за Xtensa LX6 архитектуру, која долази као саставни део ESP интегрисаног микроконтролера. Производи се од стране већег броја компанија, а у овом раду је коришћена развојна плоча ESP32 Devkit V1, приказана на слици 1. Ова развојна плоча се одликује двојезгарним процесором Xtensa LX6, радног такта до 240MHz, радном SRAM меморијом величине 520KiB и уграђеним Блутут контролером који подржава Блутут верзију 4.2. Ова развојна плоча се одликује и ниском ценом, која је у тренутку писања овог рада била мања од 7 америчких долара.



Слика 1: Изглед ESP DevKit v1 развојне плоче

Са стране програмске подршке, коришћено је Espressif IDF програмско окружење, које се може слободно преузети са Интернета и састоји се од преводиоца, стандардне библиотеке, библиотека за уграђене компоненте (од највећег значаја је наравно библиотека за управљање Блутут везом и пакетима на физичком нивоу), система за аутоматско преводићење са зависностима, документације и др. Преводиоца је заснован на познатом FLOSS GCC преводиоцу, што је умногоме олакшало писање програмске подршке, због претходног искуства које смо имали са овим преводиоцем.

Одређени проблем при реализацији настао је због архитектуре библиотеке за Блутут подршку. Функције из ове библиотеке обављају своје задатке асинхронно и користе повратне функције као начин сигнализације да је задатак (успешно или неуспешно) завршен. Због овакве архитектуре, одређене функције унутар кода клијента нисмо могли да имплементирамо линеарно, па смо их имплементирали коришћењем аутомата са коначним бројем стања. Сам аутомат је описан функцијом, док су стања представљена вредностима енумерације, а прелази се иницирају на основу горепомених повратних функција. Целокупан код решења се може пронаћи на Интернету, на следећим адресама:

- Послужилац на http://github.com/authext/bt_server
- Клијент на http://github.com/authext/bt_client

Целокупан код је лиценциран под BSD-3 лиценцом.

3. РЕЗУЛТАТИ

Две одлике које су биле у фокусу овог рада су потрошња електричне енергије и квалитет везе, који се огледа у односу послатих и примљених Блутут пакета. Вредности ових параметара су мерење засебно.

За мерење потрошње електричне енергије, коришћен је мерни уређај Gembird EG-EMU-03 USB power meter,

који је приказан на слици 2. Овај уређај приказује неколико параметера везаних за електричне одлике испитиваног уређаја, од којих су најважније утрошена електрична енергија (на уређају изражена у Wh) и протекло време од почетка мерења (на уређају изражено у часовима, минутима и секундама), на основу којих се може израчунати просечна снага мереног уређаја. На мерном уређају се, осим поменутих одлика, могу уочити и тренутни интензитет струје (изражен у A), тренутни напон (изражен у V), количина пренесеног наелектрисања (изражена у mAh) и тренутна снага (изражена у W). Мерни уређај је декларисан са резолуцијом очитавања напона од $10mV$, резолуцијом очитавања струје од $1mA$ и резолуцијом очитавања времена од $1s$. Мерење смо вршили у неколико независних конфигурација:

- Послужилац који није повезан са клијентом,
- Послужилац спојен са клијентом коришћењем Блутут технологије
- Клијент спојен са послужиоцем коришћењем Блутут технологије



Слика 2: Gembird EG-EMU-03 USB power meter мерни уређај

Да бисмо извршили само мерење, *ESP* уређај смо повезали на мерни уређај (тестер). Након тога, тестер смо прикључили на *USB* порт рачунара. Тестер је искључиво електрични уређај и што се тиче *USB* протокола у потпуности је транспарентан. Рачунар овако повезан склоп препознаје као *ESP* уређај, на исти начин као и када тестер није повезан. Преко рачунара смо на *ESP* уређају покренули жељени програм (послужилац или клијент). Уколико је потребно, на другим *ESP* уређајима смо покренули програм (клијент или послужилац) неопходан да се оствари комуникација описана конфигурацијом. Сваку конфигурацију смо пустили да ради одређену количину времена, након чега смо измерили утрошену електричну енергију и израчунали просечну снагу. Резултати се могу видети у табелама 1, 2 и 3.

ТАБЕЛА 1: Приказ потрошње електричне енергије послужиоца који није повезан са клијентом

Трајање (min)	Утрошена Енергија (Wh)	Просечна снага (W)
80	0.420	0.315
81	0.434	0.317
89	0.472	0.318

ТАБЕЛА 2: Приказ потрошње електричне енергије послужиоца који је повезан са клијентом

Трајање (min)	Утрошена Енергија (Wh)	Просечна снага (W)
62	0.407	0.393
99	0.648	0.393
1028	6.760	0.394

ТАБЕЛА 3: Приказ потрошње електричне енергије клијента који је повезан са послужиоцем

Трајање (min)	Утрошена Енергија (Wh)	Просечна снага (W)
62	0.345	0.333
100	0.549	0.329
1030	5.686	0.331

На основу измерених резултата и тренутне цене електричне енергије [4], израчунате су цене годишње потрошње електричне енергије у најгорем случају (једнотарифни обрачун, црвена зона, уређај ради непрекидно целу годину) и представљене у табели 4.

ТАБЕЛА 4: Годишња цена за непрестано коришћење уређаја у црвеној зони, при једнотарифном обрачуну

Програм	Снага (W)	Цена ($RSD/година$)
Послужилац	0.394	50.812
Клијент	0.333	42.945

Друга битна одлика која је мерена је квалитет везе. Вредности звучног сигнала се шаљу путем Блутут пакета. Лако се закључује да ако одређени Блутут пакет који је послат са послужиоца не буде примљен на клијенту да то негативно утиче на квалитет везе.

Што више пакета буде изгубљено на путу до клијента, то ће квалитет везе више трпети, што се огледа у разним деформацијама звучног сигнала који се реконструише на клијенту. Број изгубљених пакета се рачуна као разлика послатих пакета на послужиоцу и примљених пакета на клијенту.

Извршено је мерење послатих и примљених пакета. Блутут веза између послужиоца и клијента је била прекидана и поново успостављана, на сваких 20s. Коришћене су следеће конфигурације:

- Послужилац и клијент су постављени на мање од $10cm$.
- Послужилац и клијент су постављени на даљину од $270cm$, са различитих страна зида.
- Послужилац и клијент су постављени на даљину од $690cm$, са различитих страна зида.

Резултати мерења приказани су у табелама 5, 6 и 7.

ТАБЕЛА 5: Резултати мерења где су послужилац и клијент постављени на даљину од 10cm

Послатих пакета	Примљених пакета	Изгубљених пакета	Проценат изгубљених
119986	119981	5	0.004
120098	120094	4	0.003
120220	120212	8	0.007

Табела 6: Резултати мерења где су послужилац и клијент постављени на даљину од 270cm

Послатих пакета	Примљених пакета	Изгубљених пакета	Проценат изгубљених
119861	119842	19	0.016
119553	119531	22	0.018
119570	119562	8	0.007

ТАБЕЛА 7: Резултати мерења где су послужилац и клијент постављени на даљину од 690cm

Послатих пакета	Примљених пакета	Изгубљених пакета	Проценат изгубљених
105320	105027	293	0.278
108227	108013	214	0.197
114812	114421	391	0.341

4. ЗАКЉУЧАК

Током израде овог рада, реализовали смо алгоритам за управљање већим бројем уређаја опремљених микрофоном, са циљем решавања проблема који настају при интерпретацији звучног сигнала који је снимљен из удаљеног поља. Овај алгоритам је реализован на јефтиној ESP развојној плочи. Алгоритам подразумева да постоји један клијентски уређај и већи број уређаја послужилаца. Алгоритам уређује начин на који се клијентски уређај повезује на одређени послужилац и са њега преузима вредности звучног сигнала у реалном времену. Након имплементације алгоритма, извршено је мерење потрошње електричне енергије и рачунање цене коришћења оваквог уређаја током дужег временског периода. Такође, извршено је и мерење квалитета везе између клијентског и уређаја послуживоца.

На основу резултата добијених мерењем потрошње електричне енергије и рачунањем годишње цене, долази се до закључка да уколико би корисник користио 10 оваквих уређаја (1 клијент, 9 послужилаца) непрестано током целе године, морао би у најгорем случају да плати нешто више од 500RSD годишње или нешто више од 40RSD месечно.

На основу ових података, закључује се да је овако реализован уређај задовољавајуће јефтин за коришћење.

На основу резултата добијених мерењем квалитета везе примећује се да је проценат изгубљених пакета мањи од 0.5%, што не представља проблем за алгоритам прикривања губитка пакета SBC кодека [5] [6]. Закључује се да овако реализован уређај поседује задовољавајући квалитет преноса звучних података.

5. ЛИТЕРАТУРА

- [1] Н. Рајић, Н. Четић, М. Поповић, С. Танацковић, „Систем за испитивање гласовних команди у паметној кући“ на 62. конференција за електронику, телекомуникације, рачунарство, аутоматику и нуклеарну технику ЕТРАН, 2018.
- [2] Д. Додер, Н. Четић, Ј. Ковачевић, М. Поповић, „Implementation of low cost wireless acoustic sensor based on ESP8266 module“ (енгл) на 2nd International Conference on Eletrical, Eletronic and Computing Engineering, 2016.
- [3] Д. Додер, Н. Четић, М. Поповић, Ј. Ковачевић, „Realisation of server application for acoustic sensors based on RTSP, RTP protocols“ (енгл) на 23. Телекомуникациони форум ТЕЛФОР, 2015.
- [4] Електропривреда Србије, „Методологија за одређивање цене електричне енергије за јавно снабдевање“, Службени гласник Републике Србије 84/14, 2014, ISSN 0353-8389
- [5] С. Hoene, М. Hyder, „Optimally Using the Bluetooth Subband Codec“, на The 35th Annual IEEE Conference on Local Computer Networks, 2010.
- [6] С. Hoene, М. Hyder, „Considering Bluetooth's Subband Codec (SBC) for Wideband Speech and Audio on the Internet“, Технички извештај WSI-2009-3, 2009, ISSN 0946-3852

Кратка биографија:



Александар-Вук Павловић рођен је у Новом Саду 1994. године. Дипломски рад на Факултету техничких наука из области Електротехнике и рачунарства – Рачунарство и аутоматика одбранио је 2017. године.
контакт: wolf.srb.ns@hotmail.com



Др Мирослав Поповић је редовни професор на Факултету техничких наука (катедра за рачунарску технику и међурачунарске комуникације). Докторирао је на Факултету техничких наука 1990. године. Ужа научна област му је рачунарска техника и рачунарске комуникације.

UTICAJ DISTRIBUTIVNIH GENERATORA NA AUTOMATSKU REGULACIJU NAPONA**IMPACT OF DISTRIBUTED GENERATION ON AUTOMATIC VOLTAGE REGULATION**

Vasilija Šešlija, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je razmatran uticaj distributivnih generatora na regulaciju napona. Izložena su i upoređena rešenja dobijena korišćenjem klasične i modifikovane regulacije napona. Za potrebe verifikacije razvijen je program u programskom jeziku FORTRAN.

Ključne reči: Regulacija napona, distributivni generatori

Abstract – In this paper impact of distributed generation on voltage regulation with automatic voltage regulator is presented. Solutions obtained by using classical and modified voltage regulation are shown and compared. Solution verification is developed in programming language FORTRAN

Keywords: Voltage regulation, distributed generation

1 UVOD

Prateći svetske trendove, poslednjih godina i u Srbiji intenzivno raste broj distributivnih generatora (DG) priključenih na distributivnu mrežu (DM). Njihovim priključenjem tradicionalno pasivne DM postale su aktivne DM, što ima direktan uticaj na promenu tokova snaga, vrednosti struja kratkih spojeva, kvalitet električne energije, itd.

U takvim okolnostima ugrožena je pouzdanost i efikasnost široko primenjenih klasičnih postupaka za regulaciju napona (RN) [1], rad i koordinaciju relejne zaštite, restauraciju napajanja nakon kvara, planiranje distributivnih kompanija, itd. U skladu sa tim, distributivne kompanije moraju da se prilagode i/ili u potpunosti promene klasične principe u okviru svog delovanja [2]. Od interesa za razmatranja u ovom radu jeste uticaj DG na rad klasične automatske regulacije napona (KRN) u DM.

2 OSNOVNI POJMOVI

U ovom delu razmotreni su pojmovi koji su od interesa za razumevanje koja slede:

- Potrošnja,
- Slični i karakteristični potrošači,
- Karakteristični dani i sezone,
- Regulacioni transformatori,
- Automatski regulatori napona.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Goran Švenda, red.prof.

2.1 Potrošnja

Pod potrošnjom se smatra ukupna potrošnja svih potrošača koji se napajaju sa NN sabirnica transformatora. Njena vrednost definisana je bar sa dve veličine, npr: modulom struje i faktorom snage, odnosno aktivnom i reaktivnom snagom. Za razmatrani period, potrebne su njihove [3]:

- minimalna i maksimalna vrednost,
- dnevni hronološki dijagrami (DHD),
- normalizovani dnevni hronološki dijagrami (NDHD).

2.2 Slični i karakteristični potrošači

Svi potrošači čija potrošnja može da se opiše jedinstvenim NDHD predstavljaju grupu sličnih potrošača. Reprezent te grupe je karakteristični potrošač sa istim tim NDHD i jediničnim kvantitativnim pokazateljem potrošnje.

2.3 Karakteristični dani i sezone

Period unutar kojeg svaki od karakterističnih potrošača, za sve karakteristične dane, može da se opiše jedinstvenim skupom NDHD, a svaki od potrošača jednim kvantitativnim pokazateljem potrošnje, predstavlja sezonu – karakteristični period.

2.4 Regulacioni transformatori

Najčešći regulacioni transformatori u praksi su transformatori za regulaciju odnosa veličina napona uz fiksni odnos faznog pomeraja. Oni se mogu podeliti na:

- regulacione transformatore sa regulacijom pod opterećenjem (RTPO),
- regulacione transformatore sa regulacijom u beznaponskom stanju (RTBS).

RTPO su uobičajeno naponskog nivoa $V_N/SN_1/SN_2$ sa stepenom regulacije $\pm 12 \times 1.25 \%$, odnosno $\pm 10 \times 1.6 \%$. Promena odnosa transformacije je diskretna.

2.5 Automatski regulatori napona

Uloga automatskog regulatora napona (ARN) je da se na osnovu izmerene vrednosti modula struje na sekundaru RTPO definiše željena (referentna) vrednost modula napona na kontrolisanim SN sabirnicama. Željena vrednost napona se poredi sa izmerenom vrednošću. Ako je njihova razlika veća od dozvoljene razlike, preko kontrolnih releja se inicira promena položaja regulacionog otepa na RTPO [3].

3 POSTAVKA PROBLEMA

Osnovni pokazatelj kvaliteta regulacije napona jeste šteta koju elementarni potrošači električne energije (električni aparati) trpe usled odstupanja napona od nominalnih vrednosti [1].

U postavci problema regulacije napona izvršena je prostorna i vremenska dekompozicija problema, pri čemu je kriterijum štete postavljen kao osnovni motiv za regulaciju napona. U tradicionalnim distributivnim mrežama (DM) problem je u potpunosti rešen primenom klasične regulacije napona (KRN), međutim osnovni princip na kome ona počiva, a to je jednosmeran tok snage, prestaje da važi u DM sa DG.

3.1 Prostorno- vremenska dekompozicija problema

Regulacija napona je kompleksan problem koji se može razložiti u prostoru i vremenu [1]. U zavisnosti od veličine mreže nad kojom se vrši upravljanje može se izvršiti prostorna dekompozicija problema. U zavisnosti od brzine upravljanja regulacionim resursima, što pre svega zavisi od tipa i mogućnosti regulacionog resursa, može se izvršiti vremenska dekompozicija problema kontrole napona.

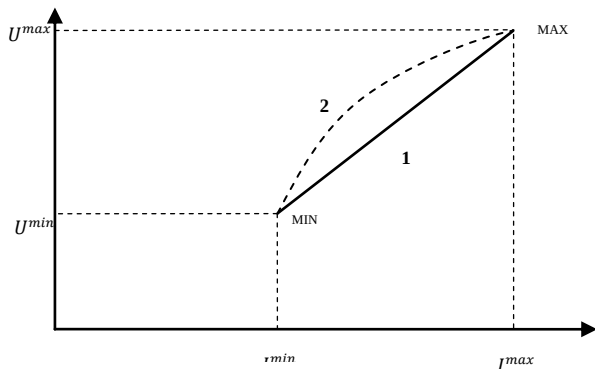
3.2 Kvantifikacija naponskog profila mreže – Šteta

Distribuciju električne energije neminovno prate padovima napona na njenim elementima (vod, transformator). Pad napona je promenljiva veličina, njena vrednost zavisi od parametara i režima DM. Promena pada napona ima za posledicu promenu napona na potrošačima i štete koju potrošači trpe jer se napajaju električnom energijom s naponima različitim od nominalnih. Iako se primenom RTPO i unapred definisanim naponskim podrškama na RTBS (Tr SN/NN i SN₁/SN₂), vrednosti napona mogu regulisati, štete se ne može potpuno izbeći [1].

3.3 Klasična regulacija napona

Klasična regulacija napona (KRN) se zasniva na [3,5]:

- predstavi DM samo sa RTPO, sa ARN i ekvivalentnim vodom, s rednim parametrima R i X , na čijem je kraju priključen jedinstven (ekvivalentni) potrošač,
- proceni (proračunu) minimalnog i maksimalnog opterećenja na sekundaru napojnog RTPO (I_{Tr}^{min} , I_{Tr}^{max}) i
- proračunu optimalnih vrednosti napona na SN sabirnici RTPO za min. i max. režim DM (V_{min} , V_{max}).



Slika 1. Karakteristika zakon regulacije:
(1) linearna i (2) nelinearna [1]

Na osnovu rezultata proračuna, za ta dva režima, definisana je karakteristika zakona regulacije ARN, tačke MIN i MAX, odnosno funkcionalna zavisnost napona sekundara

RTPO, V_{sek} od njegovog opterećenja, struje I_{Tr}^{sek} , slika 1.

3.4 Problem KRN u DM sa distributivnim generatorima

Jedan od osnovnih preduslova za korektan rad KRN jeste da je smer tokova aktivne snage od korena DM ka njenim potrošačima, odnosno da je DM pasivna. U aktivnim DM merenje modula struje na sekundaru RTPO nije jednako opterećenju DM. U skladu sa tim primenom KRN donose se odluke na osnovu pogrešnih informacija (lažne slike o opterećenju DM), čime je onemogućena kvalitetna procena optimalne vrednosti napona na sekundaru RTPO, odnosno onemogućen je njen kvalitetan rad [2].

U pasivnim mrežama napon opada duž izvoda. Pri maksimalnoj potrošnji, maksimalan je i pad napona. Da bi poslednji potrošač na izvodu imao napon u dozvoljenim granicama, potrebno je podići napon na RTPO. Kada je potrošnja mala, napon na RTPO potrebno je sniziti, kako naponi ne bi bili iznad dozvoljene granice. Ove aktivnosti koordinisane su zakonom regulacije.

Problem nastaje kada se deo potrošnje DM pokriva iz DM, proizvodnjom DG. Tada, RTPO nema korektnu sliku o vrednosti potrošnje DM za koju treba da izvrši kompenzaciju pada napona.

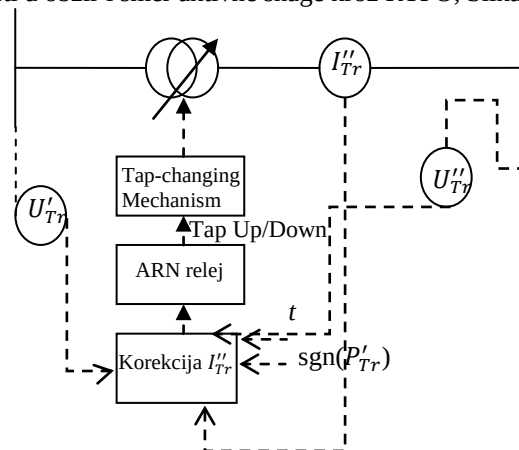
4 MATEMATIČKI MODEL

Ulazni podaci u ARN su vrednosti modula struje i napona sa sekundara RTPO. Kada DG ne radi, potrošači se napajaju iz prenosne mreže, preko RTPO, tako da je struja kroz transformator jednaka struji potrošnje. Kad se uključi DG, te struje više nisu jednake i ulazni podaci u ARN nisu validni. Kada DG proizvodi više nego što potrošnja zahteva, deo snage se preko RTPO šalje u prenosnu mrežu, u suprotnom smeru od podrazumevanog. Ampermetar na RTPO, meri samo moduo fazora struje (ali ne i njen smer). Konačno, struja koja se šalje na ARN ne odgovara režimu DM, odnosno vrednosti njene ukupne potrošnje.

4.1 Blok dijagram modifikovane KRN

Modifikovana KRN (ModKRN) zasniva se na korekciji vrednosti modula struje I_{Tr}'' sa sekundara RTTr. Ulaz u ARN više neće biti samo modul struje sa sekundara, već vektorski zbir struja sa sekundara RTPO i struje DG.

Pored modula napona sa primara i sekundara, modula struje sa sekundara RTPO i pozicije sklopke, potrebno je uzeti u obzir i smer aktivne snage kroz RTPO, Slika 2.



Slika 2. Blok dijagram sistema modifikovane KRN [2]

4.2 Procena režima grane sa generatorom

Procena režima grane sa priključenim DG se vrši na osnovu unapred definisane vrednosti faktora snage DG $\cos(\varphi_G) \approx 1$, izmerenih modula struje DG, kao i napona sa sekundara RTPO [2]:

$$\hat{I}_G = -I_G \cos(\hat{\varphi}'_G + \alpha) + jI_G \sin(\hat{\varphi}'_G + \alpha) \quad (1)$$

gde su:

I_G – modul struje sa početka izvoda sa DG,

$\hat{\varphi}'_G$ – fazni stav struje generatora,

α – ugao između fazora struja sa početka i kraja izvoda sa DG.

4.3 Procena režima regulacionog transformatora

Potrebno je, na osnovu izmerenih vrednosti modula struje na sekundaru I''_{Tr} i modula napona na sabirnicama primara U'_{Tr} i sekundara U''_{Tr} , proceniti režim RTPO. Dve moguće varijante fazora struja i napona su takve da je vrednost modula napona primara, svedena na sekundar, veća od vrednosti modula napona koja je izmerena na sekundaru i obrnuto.

Moguće su sledeće vrednosti ugla fazora struje na sekundaru RTPO [2]:

$$\psi''_{Tr} = -90^\circ \pm \arccos \left[\frac{dU''_{Tr}}{X''_k \cdot I''_{Tr}} \right] \quad (2)$$

gde su:

dU''_{Tr} – pad napona na RTPO,

X''_k – impendansa RTPO svedena na sekundar,

I''_{Tr} – struja sekundara RTPO.

U skladu sa izrazom (2), i za pozitivnu i negativnu vrednost pada napona, postoje po dva matematički korektna rešenja.

4.4 Procena potrošnje potrošača

Za korektan rad KRN sa ugrađenim ARN, sa zakonom regulacije koji je podešen bez uticaja DG, ARN treba da se prosledi sledeća vrednost modula struje [2]:

$$I_{ARN} = \{ [I''_{Tr} \cos \psi''_{Tr} + I_G \cos(\varphi'_G + \alpha)]^2 + [I''_{Tr} \sin \psi''_{Tr} - I_G \sin(\varphi'_G + \alpha)]^2 \}^{\frac{1}{2}}, \quad (3)$$

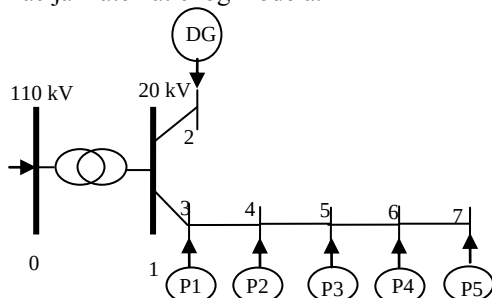
gde je sa I_{ARN} naznačena vrednost modula struje koja ulazi u ARN.

5 VERIFIKACIJA MATEMATIČKOG MODELA

Verifikacija prikazanog matematičkog modela izvršena je na primeru jednostavne DM. Analizirane su mogućnosti rada KRN i ModKRN u zavisnosti od režima DM.

5.1 Test mreža

Na slici 3 je prikazana test mreža nad kojom je izvršena verifikacija matematičkog modela.



Slika 3. Jednostavna test mreža

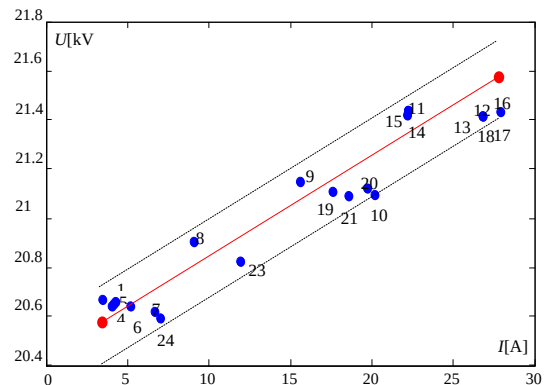
5.2 Potrošnja

Potrošači su modelovani kao potrošači konstantne struje (samo pri nominalnom naponu imaju međusobno jednake vrednosti koje se kreću od 8.4 do 57 A).

Proizvodnja DG je modelovana kao negativna potrošnja (DG injektira snagu u DM). Za potrebe verifikacije modela, proizvodnja se menja od 0 do 10 MW.

5.3 Automatska regulacija napona

Na slici 4 je prikazana karakteristika zakona regulacije za razmatran primer, bez DG. Crvene tačke su optimalne vrednosti napona za minimalnu (3h) i maksimalnu (16h) potrošnju – na osnovu njih je formirana karakteristika. Plave tačke predstavljaju optimalne vrednosti napona za svaki sat.



Slika 4 – Karakteristika zakona regulacije (DM bez DG)

5.4 Regulacija napona – DM bez DG

Struja koja ulazi u ARN jednaka je struji potrošnje, stoga regulacija radi korektno. Vrednosti dobijene u ovom primeru su referentne vrednosti, one treba da se primenom regulacije napona dobiju bez obzira na proizvodnju DG.

U nastavku se razmatraju tri premera: DG proizvodi 1, 5 i 10 MW. U tabelama 1, 2 i 3 su prikazane vrednosti dobijene proračunom tokova snaga, u skladu sa zakonom regulacije, slika 4, koji dobija vrednost struje sa sekundara RTPO (KRN) i modifikovanu struju sa sekundara RTPO (ModKRN).

5.5 Regulacija napona – DG proizvodi 1 MW

Primena KRN kritičnija je za režime sa malim opterećenjem, 3 h, kada je potrošnja mala i bliska proizvodnji DG (kada se većim delom napaja iz DG). Posledica je mala struju kroz RTPO, primenom KRN nije dobro definisana optimalna vrednost napona, tako da raste šteta koju trpe potrošači. Za režime sa velikom potrošnjom, 16 h, potrošači se pretežno napajaju preko RTPO, a manje iz DG, pa je struja I_{ARN} bliska struji potrošnje.

Korišćenjem ModKRN naponski profil na SN sabirnici isti je kao kada je DG bio isključen.

Tabela 1. DG proizvodi 1 MW

	3h		16h	
	KRN	ModKRN	KRN	ModKRN
I''_{Tr} [A]	7.62		252.26	
I_{DG} [A]	27.87		28.18	
I_P [A]	37.10		282.28	
V_{opt} [V]	20669.97		21434.98	
I_{ARN} [A]	7.62	35.47	252.26	276.31
V_{SN} [V]	20347.93	20669.87	21439.10	21439.10
$\dot{S}$ [n.j.]	30.47	1.012	354.64	354.64

5.6 Regulacija napona – DG proizvodi 5 MW

Vidi se da je u 3h, pri minimalnom opterećenju proizvodnja DG daleko veća od potrošnje, pa se određeni deo snage vraća u prenosnu mrežu. To za KRN znači da je I_{ARN} znatno veća od stvarne potrošnje, stoga u skladu sa karakteristikom po kojoj deluje podiže napon na sekundaru.

Kod ModKRN, I_{ARN} dobija se korekcijom I''_{Tr} , odnosno njenim vektorskim sabiranjem sa I_{DG} . Tako se dobija struja koja je dovoljno bliska struji potrošnje, pa regulator reaguje odgovarajućim naponom.

U 16 h, DG samo delimično napaja potrošnju, odnosno tok snage je jednosmeran. I_{ARN} vrlo je bliska struji potrošnje, pa se na sekundaru RTr dobija napon kao u slučaju kada je DG isključen.

Tabela 2. DG proizvodi 5 MW

	3h		16h	
	KRN	ModKRN	KRN	ModKRN
$I''_{Tr}[A]$	97.00		154.58	
$I_{DG}[A]$	131.73		132.19	
$I_P[A]$	37.10		282.28	
$V_{opt}[V]$	20669.97		21434.98	
$I_{ARN}[A]$	97.00	35.42	154.58	272.74
$V_{SN}[V]$	20980.00	20646.39	21077.76	21434.1
$\dot{S} [n.j.]$	43.55	3.15	894.48	357.14

5.7 Regulacija napona – DG proizvodi 10 MW

Može se uočiti da je, pri velikoj proizvodnji DG, KRN u režimima minimalne potrošnje 3h, realizovana kao da je režim sa maksimalnom potrošnjom. I obrnuto, u 16h, režim sa maksimalnom potrošnjom je tretiran kao režim sa minimalnom potrošnjom.

Vrednosti napona značajno su korigovane korišćenjem ModKRN, čime je skoro 5 smanjena ukupna šteta koju trpe potrošači.

Tabela 3. DG proizvodi 10 MW

	3h		16h	
	KRN	ModKRN	KRN	ModKRN
$I''_{Tr}[A]$	220.14		85.29	
$I_{DG}[A]$	254.49		256.92	
$I_P[A]$	37.10		282.28	
$V_{opt}[V]$	20669.97		21434.98	
$I_{ARN}[A]$	220.14	61.48	85.29	295
$V_{SN}[V]$	21257.62	20571.39	20680.39	21760.37
$\dot{S} [n.j.]$	132.6	2.5	2237.87	451.037

6 ZAKLJUČAK

Karakteristika regulacije za KRN zasniva se na vrednosti modula struje izmerene na sekundaru RTPO. U pasivnoj DM, ona je jednaka struji potrošnje, dok je u aktivnoj jednaka struji potrošnje umanjenoj za struju DG. U skladu sa tim, u aktivnim DM narušena je osnovna ideja rada KRN. Posledica je da će se njenom primenom na potrošačima realizovati loši naponi, odnosno nepotrebna šteta.

Rezultati prikazani u ovom radu pokazuju da se primenom ModKRN prethodno navedeni problemi uspešno prevazilaze.

7 LITERATURA

- [1] V.Strezoski, D.Janjić: *Sistemi regulacije napona radijalnih distributivnih mreža*; Fakultet tehničkih nauka, Novi Sad, 1997.
- [2] Z.Simendić, G.Švenda, V.Mijatović, P.Bajčetić: Modeli regulacije napona u klasičnim distributivnim mrežama sa generatorima; *XI Savetovanje o elektrodistributivnim mrežama Srbije, sa regionalnim učešćem – CIRED*, Kopaonik, 24-28. septembar 2018, Smart Grid forum, R-F.07, str. 1-10
- [3] D.Đanić: *Primena URN u ED Sombor*; master rad, Fakultet tehničkih nauka, Novi Sad, Srbija, 2011.
- [4] G.Švenda, V.Strezoski, R.Bibić: Regulacija napona u eksploataciji distributivnih mreža; *Elektroprivreda*, br.3, 2008, str. 28-42.

Kratka biografija:



Vasilija Šešlija rođena je u Somboru, 1994. god. Osnovne studije završila je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi 2017. god. i iste godine upisala master studije.

RAZVOJ SKLADIŠTENJA I PRETRAGE DIGITALNE BIBLIOTEKE DIPLOMSKIH RADOVA UPOTREBOM MongoDB, Elasticsearch I Blockchain TEHNOLOGIJA**DEVELOPMENT OF DATA STORAGE AND RETRIEVAL OF DIGITAL LIBRARY APPLICATION USING MongoDB, Elasticsearch AND Blockchain TECHNOLOGIES**

Zlatan Prečanica, Milan Vidaković, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj: U radu je implementirana aplikacija digitalne biblioteke studentskih diplomskih radova. Svaki diplomski rad predstavlja digitalni dokument, te se projekat bavi metodama skladištenja i pretrage diplomskih dokumenata. U te svrhe su korišćene MongoDB, Elasticsearch i Blockchain tehnologije.

Keywords: Java, SpringBoot, MongoDB, ElasticSearch, Blockchain

Abstract: This thesis deals with the development of a digital library application. The digital library is used to store and retrieve student's thesis. Each thesis has been presented as a digital document. For storage and search MongoDB, Elasticsearch and Blockchain technologies are used.

Ključne reči: Java, SpringBoot, MongoDB, ElasticSearch, Blockchain

1. UVOD

Biblioteka je sređena zbirka knjiga po autorskom, predmetnom ili nekom drugom katalogu, koja pripada privatnim licima, državnim institucijama ili ustanovama u kojima se čuvaju zbirke knjiga.

Prve biblioteke nastale su pre više hiljada godina, a prve knjige u tim bibliotekama su bile glinene pločice ispisane klinastim pismom. Nove tehnologije su uticale na promenu paradigme bibliotekara pri čemu su bibliotekari uočili dve važne činjenice: da je današnje digitalno okruženje, kao što su internet i razne društvene mreže, postalo konkurencija bibliotekama i da je bibliotekama u 21. veku osigurana budućnost samo ukoliko uspeju da se prilagode novim tehnologijama.

Digitalna biblioteka je biblioteka u kojoj se čuvaju knjige u digitalnom formatu, a svaka knjiga koja se nalazi u digitalnoj biblioteci dostupna je putem računara i interneta.

Tema ovog rada jeste implementacija jedne takve digitalne biblioteke koja sadrži bazu studentskih završnih (diplomskih) radova.

NAPOMENA:

Ovaj rad je proistekao iz master rada čiji mentor je bio dr Milan Vidaković.

2. KORIŠĆENE TEHNOLOGIJE I TERMINOLOGIJE

JAVA [1] - objektno-orijentisani programski jezik koji je razvila kompanija Sun Microsystems početkom 1990-ih godina. Dizajnirana je tako da što manje zavisi od specifičnih karakteristika konkretnog računarskog sistema i njegove arhitekture. Ono što Javu izdvaja od ostalih programskih jezika jeste to što se jednom kompajliran kod može pokrenuti na svim platformama koje podržavaju Javu. Java se izvršava korišćenjem Java virtualne mašine (JVM eng. *Java Virtual Machine*), apstraktne mašine koja postoji samo u memoriji i koja izvorni kod izvršava na računaru.

SpringBoot [2] - je Java radni okvir koji se koristi za kreiranje *mikro servisa*. Mikroservisi su tehnički razvoja softvera koja predstavlja varijantu servisno-orijentisanog arhitekturnog (SOA) stila koji strukturira aplikaciju kao kolekciju labavo povezanih servisa. U mikroservis arhitekturi, servisi su sitno granulirani, a protokoli su jednostavni.

MongoDB [3] - je vodeća *NoSQL* baza podataka. U skladu sa rastućim interesovanjem za alternativne sisteme za upravljanje bazama podataka, koji se razlikuju od tradicionalnih relacionih baza podataka, pojavio se koncept takozvanih *NoSQL* baza podataka, koje ne koriste *SQL* za povezivanje, nerelacione su, distribuirane, otvorenog koda i horizontalno skalabilne. *MongoDB* je napisan u C++ programskom jeziku i otvorenog je koda, izdata pod kombinacijom *GNU Affero General Public Licence* i *Apache Licence*. *MongoDB* čuva podatke kao *JSON* dokumente sa dinamičkim šemama. *JSON* (*JavaScript Object Notation*) je otvoreni standard zasnovan na tekstu, osmišljen za razmenu podataka koji su pogodni za čitanje ljudima.

ElasticSearch [4] - je server koji služi za pretragu i analizu podataka. Napisan je u Java programskom jeziku što omogućava pokretanje na svim platformama. Baziran je na *Lucene* indeksima i omogućava korisnicima da pretraže veliku količinu podataka vrlo brzo. *Lucene* indeksi imaju dodatne strukture podataka koje sadrže jednu ili više kolona tabele i kod kojih su podaci sortirani. Sortirani podaci se mogu mnogo brže pretražiti pomoću struktura kao što je B stablo. Indeksi koji se koriste u relacionim bazama podataka uglavnom koriste liste i stabla za pretragu, dok se *Lucene* indeks razlikuje po tome što koristi nizove.

Blockchain [8] - je baza podataka koja se ne nalazi na jednom mestu, već je čine manje baze (blokovi) koji su međusobno digitalno povezani, a koji sadrže informacije o digitalnim transakcijama bilo koje vrste (od vlasničkih listova, preko podataka iz knjige rođenih, do ugovora kojim se regulišu autorska prava). Za razliku od klasične *online* baze podataka, *blockchain* tehnologija omogućava komunikaciju sa nekoliko računara (servera) između kojih se transakcija obavlja. Svako obavljenom transakciji pripisuje se vremenska odrednica, dok se sama transakcija po hronološkom redu dodaje već postojećem lancu blokova. Taj lanac predstavlja niz blokova koji sadrže podatke o svakoj transakciji koja se u okviru nje izvršila. Kako je svaki blok logički povezan sa drugim, *blockchain* je prevazišao ono što ljudski rod nije uspeo od svog nastanka – postao je imun na prevaru.

Frontend tehnologije

Za implementaciju *frontend* dela aplikacije digitalne biblioteke korišćene su *JavaScript* [5], *jQuery* [6] i *Ajax* [7] tehnologije. Zahvaljujući *jQuery* i *Ajax* tehnologijama postignuta je “*single page*” arhitektura aplikacije. To znači da prilikom prelaska sa jedne funkcionalnosti na drugu, ili prilikom pritiska na bilo koje dugme na stranici, korisnik ostaje na istoj stranici na kojoj je i bio, ne postoji druga stranica koja se učitava, već se sve dešava na jednoj. Zahvaljujući *jQuery* biblioteci omogućeno je skrivanje i prikazivanje elemenata koji u datom kontekstu treba da budu prikazani ili sakriveni dok je glavna prednost *Ajax* biblioteke to što omogućava asinhronu pozive ka *backend-u* odnosno ka *REST API*-ju aplikacije.

3. SPECIFIKACIJA APLIKACIJE

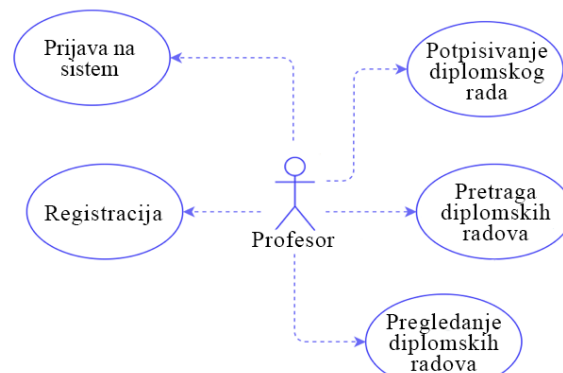
Zadatak obuhvata izradu digitalne biblioteke diplomskih radova. Ideja projekta je da se prikaže funkcionisanje jednog takvog sistema. Postoje dva tipa korisnika, studenti i profesori. Oba tipa korisnika imaju mogućnost registrovanja i prijave na sistem kao i pretraživanja i pregledanja diplomskih radova koji se nalaze u sistemu. Studenti pored ovih mogućnosti imaju i mogućnost dodavanja novih diplomskih radova, dok profesori imaju mogućnost digitalnog potpisivanja radova (Slika 1: Dijagram slučajeva korišćenja profesora). Sistem je zasnovan na tehnologiji mikroservisa i sastoji se od dve aplikacije i tri velike celine.

Prva celina obuhvata *backend* koji je zasnovan na *Java* programskom jeziku i *SpringBoot* radnom okviru za kreiranje web servisa koji ima mogućnost rada sa bazom podataka, indeksiranju dokumenata. Dokumenti (diplomski radovi) se čuvaju u *MongoDB* bazi podataka. Pored čuvanja, dokumenti se i indeksiraju zbog bržeg i efikasnijeg pretraživanja.

U ove svrhe se koristi *Elasticsearch* servis. Druga celina obuhvata *frontend* deo koji ima za zadatak da obezbedi korisnicima lako i efikasno pristupanje sadržaju. Ovaj deo aplikacije je zasnovan na *JavaScript*, *jQuery* i *Ajax* tehnologijama.

Treća i poslednja celina je zasebna aplikacija (mikro servis), koja predstavlja *Blockchain* servis. Ovaj servis je takođe zasnovan na *Java* programskom jeziku i *SpringBoot* radnom okviru. Glavna uloga ovog servisa

jest da vodi računa o svakoj transakciji (dodavanju novog dokumenta, potpisivanju dokumenta...) i sprečava bilo koju promenu podataka o obavljenim transakcijama, uključujući njihovo brisanje, dodavanje i bilo koji drugi vid falsifikovanja. Ovaj servis takođe ima podršku za rad sa *MongoDB* bazom podataka. Oba mikro servisa imaju podršku za *REST API endpoint-e* preko kojih ove dve aplikacije i komuniciraju.



Slika 1: Dijagram slučajeva korišćenja profesora

4. IMPLEMENTACIJA

Projekat se sastoji od dva mikroservisa koji su u osnovi dve *SpringBoot* aplikacije. To su *library* i *blockchain* servisi. Prvi servis predstavlja samu digitalnu biblioteku i sadrži operacije za rad sa korisnicima i digitalnim dokumentima, njihovim čuvanjem, pretraživanjem kao i potpisivanjem. Drugi servis predstavlja, kao što mu i sam naziv kaže, *blockchain* deo koji se koristi za perzistenciju informacija i koji garantuje nepromenljivost digitalnih dokumenata. Oba servisa su zasnovana na *Java* programskom jeziku i imaju podršku za *REST API* servis.

SpringBoot projekat

Prilikom kreiranja novog *SpringBoot* projekta koristi se *online* koji se nalazi direktno na oficijalnom sajtu *SpringBoot* radnog okruženja (<https://start.spring.io/>) i pruža veoma brzo i efikasno kreiranje novog projekta zasnovanog na ovom radnom okviru.

Konfiguracija projekta

Početna konfiguracija projekta vrši se u *application.properties* fajlu. U njemu se podešavaju parametri aplikacije kao što su port na kome će se aplikacija pokretati, kredencijali za bazu podataka, naziv *elasticsearch* klastera kao i host i port *elasticsearch* servisa na kome se nalazi.

Indeksiranje dokumenata

Indeksiranje dokumenata je odrađeno kao dio procesa čuvanja novih dokumenata sistemu. Pošto *SpringBoot* radni okvir pruža odličnu podršku za rad sa *elasticsearch* servisom, procedura je slična kao i za rad sa običnom bazom podataka. Prvo je potrebno kreirati model podataka koji će se čuvati, a zatim i repozitorijum koji će vršiti indeksiranje prosleđenog modela. Pošto se *elasticsearch* repozitorijumu prosleđuje model dokumenta koji će se indeksirati, prilikom prvog pokretanja aplikacije u *elasticsearch* servisu će se kreirati indeks koji će imati polja

ekvivalentna poljima modela koji će se u njih upisivati. Primer indeksa koji će kreirati *elasticsearch* servis je prikazana na slici 2.

```
1 {
2   "article" : {
3     "aliases" : { },
4     "mappings" : {
5       "default" : {
6         "properties" : {
7           "abstract" : {
8             "type" : "text",
9             "fields" : {
10              "keyword" : {
11                "type" : "keyword",
12                "ignore_above" : 256
13              }
14            }
15          },
16          "areaCode" : {=
38          "author" : {=
105          "committeeMembers" : {=
199          "filename" : {=
208          "keywords" : {=
217          "status" : {=
226          "title" : {=
235        }
236      },
237    },
238    "settings" : {
239      "index" : {
240        "creation_date" : "1552997028653",
241        "number_of_shards" : "5",
242        "number_of_replicas" : "1",
243        "uuid" : "rDUChIEvRmGax23auFiORw",
244        "version" : {
245          "created" : "5060499"
246        },
247        "provided_name" : "article"
248      }
249    }
250  }
251 }
```

Slika 2: ElasticArticle indeks

Za indeks "article" se definišu polja po kojima će se vršiti indeksiranje, a kasnije pretraga. Za svako polje se definiše njegov tip. Pored polja indeks sadrži i polje "settings" u kome se nalaze neki osnovni podaci o indeksu kao što je datum kreiranja, naziv, uuid (*Universally Unique Identifier*) itd.

Rad sa bazom

Za rad sa *MongoDB* bazom se koristi *Hibernate* okvir koji je sastavni deo *SpringBoot* okruženja. Nakon što su kreirani modeli podataka koji će biti čuvani u bazi podataka, potrebno je kreirati repozitorijume i servise koji će da vrše osnovne *CRUD* (*Create Read Update Delete*) operacije nad podacima. Pošto su osnovne *CRUD* operacije već definisane u *Hibernate* okviru, njih nije potrebno ponovo definisati, već se definišu samo one operacije koje su specifične za dati model.

Na primer, ukoliko je potrebno kreirati servis koji vraća samo jednog korisnika sa specifičnim email-om ili nekim drugim atributom iz baze podataka, prvo je neophodno kreirati repozitorijum koji će u bazi pronalaziti korisnika sa specifičnim atributom.

Nakon toga je potrebno kreirati servis koji preko prethodno kreiranog repozitorijuma komunicira sa bazom. Servis predstavlja glavnu vezu između kontrolera web servisa i samog *MongoDB API*-ja.

Komunikacija sa Blockchain aplikacijom

Pri dodavanju novog dokumenta u biblioteku, pored čuvanja u *MongoDB* bazu i indeksiranja u *Elasticsearch* servis, vrši se i čuvanje u *Blockchain* bloku. Za to je zadužen zaseban blockchain mikroservis. Ovaj servis vodi

računa o svim blokovima i čuva ih u bazi podataka. Aplikacija digitalne biblioteke sa ovom aplikacijom komunicira pomoću *RestTemplate*-a koji je ugrađen u *SpringBoot* okvir i omogućuje komunikaciju sa *REST API endpoint*-ima drugih aplikacija.

Implementacija pretrage

Metoda prima svih šest parametara po kojima se vrši indeksiranje dokumenata kao i tip pretrage koji predstavlja logičku operaciju. Ti parametri stižu sa *frontend*-a i šalje ih korisnik koji vrši pretraživanje. Logička operacija može da bude logičko I (*AND*) ili logičko ILI (*OR*).

Ukoliko je između polja pretrage operacija logičko I, zahteva se da dokumenti budu filtrirani po svim unesenim parametrima i da samo dokumenti koji zadovoljavaju sve parametre budu vraćeni korisniku kao rezultat pretrage. Ukoliko je operacija logičko ILI, u tom slučaju rezultat su svi dokumenti koji zadovoljavaju uslov bar jednog polja.

Digitalno potpisivanje dokumenata

Ukoliko profesor odluči da potpiše određene radove, potrebno je da prethodno selektuje one koje želi da potpiše. Nakon toga potreban je unos PIN koda. PIN kod je izgenerisan prilikom registracije na sistem upotrebom posebnog algoritma od privatnog ključa korisnika. Ukoliko je PIN kod dobar, izabrani radovi se potpisuju. Za svrhe digitalnog potpisivanja se koriste klase iz *java.security* paketa.

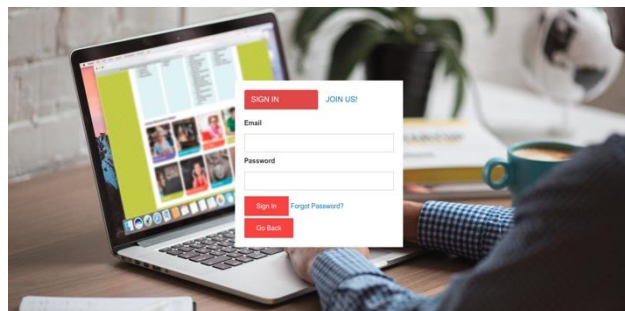
Prvi korak je učitavanje privatnog i javnog ključa korisnika. Nakon toga se učitava PDF dokument. Treći korak je potpisivanje PDF fajla koristeći metode *JAVA API*-ja i privatni ključ korisnika.

Četvrti i poslednji korak je čuvanje tako potpisanog fajla u bazi podataka. Provera da li je fajl uspešno potpisan se vrši pomoću javnog ključa.

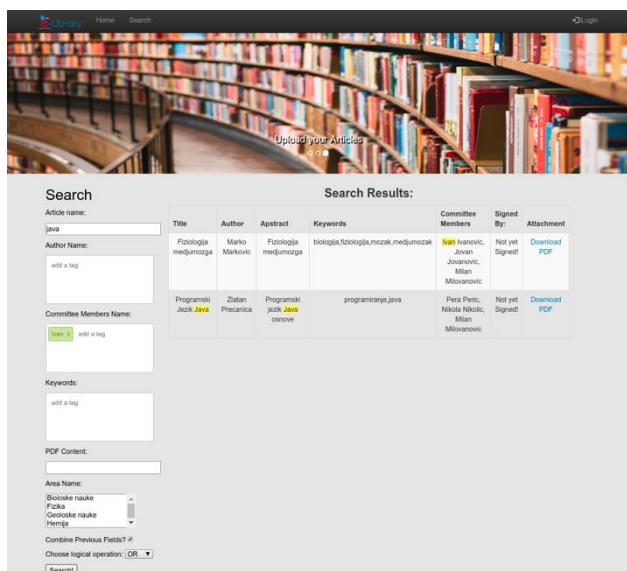
5. DEMONSTRACIJA

Stranica za registraciju i prijavu na sistem (Slika 3: *Početna stranica*) je prva stranica koju korisnik vidi. Nakon uspešnog prijavljivanja na sistem korisniku je prikazan spisak svih dokumenata u digitalnoj biblioteci. Ukoliko je prijavljen korisnik tipa student, njemu se nudi opcija dodavanja novog dokumenta.

Ukoliko je na sistem prijavljen korisnik tipa profesor i ukoliko ima nepotpisanih radova njemu se šalje obaveštenje o pristiglim radovima koji čekaju da budu potpisani. I student i profesor imaju mogućnost pretrage diplomskih radova. Rezultat kombinovane pretrage je prikazan na slici 4.



Slika 3: Početna stranica



Slika 4: Rezultat pretrage

6. ZAKLJUČAK

Ovaj rad nastoji da ukaže na bogatstvo elektronskih servisa koje nude savremene biblioteke, kao i na potrebu da se pri kreiranju digitalne biblioteke rukovodimo merilima efikasnosti. Zadatak ovog rada je bio kreiranje digitalne biblioteke u kojoj se čuvaju studentski završni (diplomski) radovi. Implementacija se u najvećoj meri bavi pitanjima čuvanja i pretrage digitalnih dokumenata, te su u te svrhe korišćeni *MongoDB* i *Elasticsearch* mehanizmi.

Pored implementacije digitalne biblioteke zadatak je bio i istraživanje blockchain tehnologija kao i pronalaženje načina upotrebe blockchain servisa u unapređenju jednog bibliotečakog informacionog sistema. Što se tiče daljeg istraživanja na ovu temu, prvenstveno je bitno pratiti razvoj informacionih sistema digitalnih biblioteka i mehanizama za njihovu implementaciju.

7. LITERATURA

- [1] Java, <https://www.java.com/en/download/>
- [2] SpringBoot, <https://spring.io/projects/spring-boot>
- [3] MongoDB, <https://www.mongodb.com/>
- [4] Elasticsearch, <https://www.elastic.co/>
- [5] JavaScript, <https://www.javascript.com/>
- [6] jQuery, <https://jquery.com/>
- [7] Ajax, [https://en.wikipedia.org/wiki/Ajax_\(programming\)](https://en.wikipedia.org/wiki/Ajax_(programming))
- [8] Blockchain, <https://www.blockchain.com/>
- [9] Wikipedia, <https://www.wikipedia.org/>

Kratka biografija:

Zlatan Prečanica rođen je 7.5.1994, Glina, Hrvatska. Završio osnovnu školu "Boško Palkovljević Pinki" u Sremskoj Mitrovici, srednju elektrotehničku školu "Nikola Tesla", takođe u Sremskoj Mitrovici. Nakon toga upisao se na Fakultet Tehničkih Nauka u Novom Sadu, odsek računarstvo i automatika.

Milan Vidaković rođen je u Novom Sadu 1971. godine. Na Fakultetu tehničkih nauka u Novom Sadu završio je doktorske studije 2003. godine. Na istom fakultetu je 2014. godine izabran za redovnog profesora iz oblasti *Primenjene računarske nauke i informatika*.

ИЗРАДА ВЕБ АПЛИКАЦИЈЕ ЗА УПРАВЉАЊЕ БИБЛИОТЕЧКИМ ИНФОРМАЦИОНИМ СИСТЕМОМ НА ПЛАТФОРМИ ASP.NET

WEB APPLICATION FOR MANAGING LIBRARY INFORMATION SYSTEM ON THE ASP.NET PLATFORM

Јасмина Еминовски, Милан Видаковић, *Факултет техничких наука, Нови Сад*

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај: У раду је анализиран ASP.NET оквир и начин на који је имплементирана апликација за управљање библиотечким информационим системом. Фокус рада је на наменски направљеном object-relation маперу и шаблону репозиторијума помоћу којег се мапер користи. Такође, приказан је и објашњен начин креирања табеларног приказа података помоћу наменски направљеног JavaScript додатка за приказ, филтрирање и сортирање података у табели.

Abstract: The paper analyzes ASP.NET framework and one implementation of an application for managing library information system. The focus is on the custom-made object-relation mapper and usage of the repository pattern. The paper also explains a way of creating data tables in UI using a custom-made JavaScript plugin to display, filter and sort data.

Кључне речи: .NET, ASP.NET, Repository pattern, custom ORM, JavaScript, plugin.

1. УВОД

У склопу Мајкрософтове .NET платформе [1] која се бави развојем различитих врста апликација (десктоп, мобилне, итд.), важно место заузимају web апликације. Скуп алата и библиотека намењених за развој web апликација се назива ASP.NET [2]. Ова технологија тренутно представља једну од водећих технологија за развој web апликација у свету. Задатак овог рада је приказ могућности које ASP.NET технологија пружа. Апликација за управљање радом библиотеке – LMS (Library management system), не представља реални библиотечки информациони систем, већ један његов модел и треба је посматрати као анализу случаја (енгл. case study) за конкретни задатак рада.

2. СПЕЦИФИКАЦИЈА АПЛИКАЦИЈЕ

Апликација за управљање радом библиотеке представља информациони систем намењен за лакше управљање пословним процесима у библиотеци али и сервис који могу користити чланови библиотеке. Постоје три типа регистрованих корисника: администратор, библиотекар и претплатник.

НАПОМЕНА:

Овај рад је проистекао из мастер рада чији ментор је био проф. др Милан Видаковић.

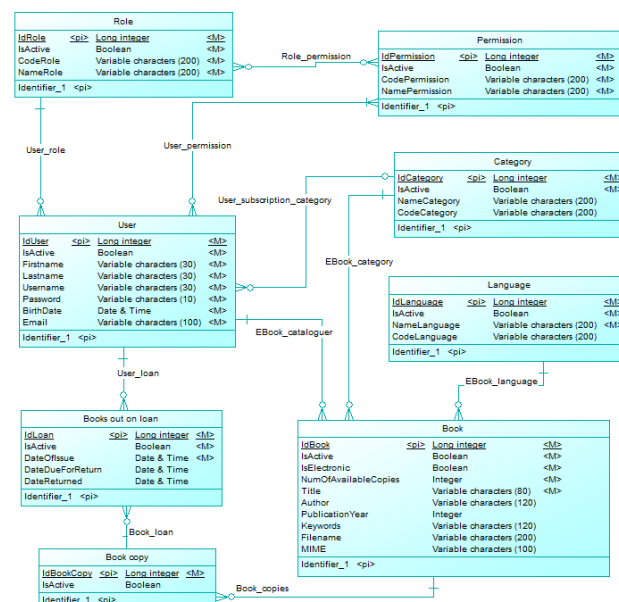
Осим њих сервису могу приступи и посетиоци – нерегистровани корисници који имају могућност да виде доступне категорије књига и књиге (традиционалне и електронске).

Сви регистровани корисници поред пријаве и одјаве са система могу да врше преглед и измену личних података, преглед основних ентитета у систему (књиге и категорије), претражују електронске књиге и размењују поруке – чет.

Претплатници додатно могу да виде историју својих изнајмљених књига и да преузму дигиталне књиге које припадају категорији на коју су претплаћени.

Администратор и библиотекар обављају CRUD (engl. create, read, update, delete) операције над ентитетима. Администратор обавља регистрацију нових корисника било ког типа, док библиотекар може додати само новог претплатника.

Поред тога обавља и задуживање и раздуживање књига у систему. Модел података приказан је на слици број 1 у наставку.



Слика 1. Дијаграм модела података

Сваки ентитет има идентификатор и индикатор активности који служи за логичко брисање појаве ентитета. Поред улога у систему користе се и дозволе за обављање одређених активности. Дозволе се могу директно доделити кориснику од стране администратора.

3. ТЕХНОЛОГИЈЕ И АЛАТИ

ASP.NET MVC 5

Постоје три различите технологије које се могу користити у оквиру ASP.NET оквира: Web Forms [3], MVC [4], и ASP.NET Web Pages [5]. Сходно претходном знању и искуству у раду коришћена је MVC технологија у верзији 5.2.3 која се ослања на истоимени архитектонски шаблон. *Model-View-Controller* [6] је један од најважнијих архитектонских шаблона у рачунарству и заснива се на подели архитектуре према одговорности на ове три компоненте. Модел представљају класе које описују податке. Поглед – *view* представља изглед апликације односно *.cshtml* странице. Контролер је задужен за комуникацију између корисника и тока апликације односно начина функционисања. MVC је у ASP.NET оквиру покривен одређеним конвенцијама које олакшавају процес развоја апликације.

RAZOR СИНТАКСА

Razor [7] синтакса је једноставан начин за уградњу програмске логике у web страницу. Ова синтакса је основ *view* компоненте у ASP.NET MVC. На *.cshtml* страницама могу постојати две врсте садржаја: клијентски и серверски. Серверски код је задужен за логику и динамичко креирање клијентског садржаја кога чине HTML елементи, CSS стилови или нека JavaScript логика. Оваквом употребом серверског кода омогућено је комплетно креирање web апликација без посебног познавања неке клијентске технологије, а отуда и потиче назив ASP (*active server pages*). Сама синтакса је врло једноставна и нема много правила.

ИМПЛЕМЕНТАЦИОНО ОКРУЖЕЊЕ

Коришћен је *Microsoft Visual Studio Community 2015* [8].

БАЗА ПОДАТАКА

За чување података коришћена је *Microsoft SQL Server 2014* релациона база података и *Microsoft SQL Server Management Studio 2014*. Иако је база релационог карактера, није корићено ограничење референцијалног интегритета, већ је брига о вези између података обављена на апликативном нивоу.

LUCENE .NET

За потребе рада са електронским књигама коришћена је библиотека *Lucene.NET* [9] у верзији 4.8.0 која представља *open-source* програмску библиотеку за индексирање, претрагу и анализу текста. Електронске књиге представљају *.pdf* фајлове приликом чијег отпремања се врши индексирање мета-података и садржаја ради напредне претраге.

SIGNALR

Да би обављање функција било могуће у реалном времену коришћена је библиотека *SignalR* [10] која је део ASP.NET оквира. Примену је у конкретној имплементацији нашла за потребе размене порука између корисника – чет. Овим приступом је омогућена двосмерна комуникација између клијента и

сервера. Сервер може да иницира неку функционалност на клијентској страни ако је остварена потребна конекција и ако је клијент доступан. *SignalR* се заснива на принципу мрежних сокета (*web socket*) [11].

4. ИМПЛЕМЕНТАЦИЈА

Пројекат чине два дела: први ASP.NET MVC апликација и други DAL (енг. *data access layer*) *class library* део намењен за рад са базом. MVC апликација има референцу на DAL потпројекат. Део клијентске логике је обављен уз помоћ *razor* синтаксе, али се велики део базира на чистом JavaScript језику и *jQuery* [12] библиотеци.

СТРУКТУРА И КОНФИГУРАЦИЈА

MVC апликација представља скуп компоненти које функционишу по принципима MVC архитектонског шаблона. Приликом креирања овакве апликације аутоматски се генеришу главни фолдери намењени за груписање класа модела, контролера и *view* фајлова. Том приликом настаје и фолдер који садржи статичке ресурсе за скрипт, стил фајлове и слике. Конфигурациона логика неопходна за покретање апликације је у *Global.asax* фајлу. У њему се обавља регистрација DI (енг. *dependency injection*) *Autofac* [13] контејнера неопходног за разрешавање зависности између класа у апликацији. *Global.asax* такође служи за регистрацију свих зона (енг. *area*), рута, филтера и пакета скрипт и стил фајлова који се користе на клијентској страни. Ради чистије архитектуре коришћен је концепт зоне – *area*, која представља MVC шаблон у малом. За сваки ентитет постоји посебна зона у којој се налазе контролери и *view* странице. Одвојен је део који обухвата глобалне сервисе (за ауторизацију и активацију одређених инфраструктурних елемената) од дела који обухвата пословну логику и дела који обухвата класе инфраструктуре (помоћни сервиси за валидацију, филтрирање, сортирање, изградњу објеката итд.). Такође, груписани су у фолдере делови намењени за индексирање и претрагу електронских књига и део за размену порука.

LMS OBJECT-RELATION MAPPING И РАД СА БАЗОМ

У овом поглављу је описан део архитектуре задужен за рад са базом података и обављање основних CRUD (енг. *create, read, update, delete*) операција. Имплементирано решење се заснива на *Repository* шаблону [14] без коришћења *Entity Framework*-а [15] уз *custom object-relational* мапер компоненту. Сврха шаблона репозиторијума је сакривање логике чувања, добављања и мапирања података из базе у доменске моделе и обрнуто. Класа која представља омотач око базе података је *SqlServerDatabase*. Она користи елементе ADO.NET [16] библиотеке и представља извор података за репозиторијум. Репозиторијума има онолико колико и ентитета односно табела у бази. Разликују се три нивоа апстракције. Најнижи ниво апстракције представља *BaseRepository* са заједничким операцијама за све табеле – читање и брисање. Ове SQL команде са

[illegible]

MappingModel је генеричког типа и веже се за тип ентитета о чијим конверзијама и командама за чување ће бити задужен. Сваки доменски модел има пропертије, који се преко атрибута *DBCColumn* означавају као одговарајуће колоне у бази. Типови података не морају бити исти у бази као и у доменском моделу у апликацији. За конверзију типова се приликом мапирања података користе методе које се добављају помоћу својих назива и складиште у колекције објекта *MappingModel* класе. Како би логика била генерична коришћени су делегати и стабла израза за њихову припрему. *LMSMapper* класа има улогу да повеже репозиторијум са мапинг моделом и тиме омогући позив метода са листинга у наставку преко мапинг модела у оквиру репозиторијума.

```

else
{
    command = updateCommandStatement(
        (T)Convert.ChangeType(Model, typeof(T)));
    query = ExecuteUpdateQuery();
}
command.CommandText = query;
return command;
}

```

LMSGRID

List of active books

#	Author	Title	Year	Available copies	Category	Language
2	Anne Frank	The story of a young girl	2003	3	Hirography	English
3	Ara Rambovic	Kad je svet bio takav nekada	2017	2	Dierography	Serbian
4	Walter Isaacson	EIve Jobs	2011	2	Bioraphy	English
5	Jeaner Dasein	Mage para Kalliozina	2018	2	Classics	Romanian
6	Federic M. Desiderius	Zvezin Lumea	2013	4	Classics	Serbian
7	Robert C. Martin	Clean Code	2015	1	Computers	English
8	Hannah-ny	Helle wene	2017	1	Companys	English
9	Romanao Klenka	Litica i Uspodnja	2003	3	Kids	Serbian
10	Elena Pesante	Muse gentiana e pitagorica	2017	1	Computers	Serbian

W H Page: 1 | of 1 Page: 20 - View 1 - 3 of 0

Слика 3. *LMSGrid* табела са приказом активних књига

```
public JsonResult GetAllActive(FilterSorterModel filterSorterModel)
{
    var viewModels = BookService.GetAll(true);
    var filterSorter = new
        DataCollectionFilterSorter<BookViewModel>();
    IEnumerable<BookViewModel> enumBooks = viewModels
        .AsEnumerable();
    enumBooks = filterSorter.FilterAndSort
        (enumBooks, filterSorterModel);
    return Json(enumBooks,
        JsonRequestBehavior.AllowGet);
}
```

Листинг 2. Метода за добављање активних књига

Кључни елемент је *DataCollectionFilterSorter* класа која обавља филтрирање и сортирање помоћу прослеђених података са клијентске стране уз ослонац на *LINQ* [17] и делегате. За сортирање је довољна само *OrderBy* операција из *LINQ*. За филтрирање је потребно направити делегат који ће на основу прослеђеног назива пропертија, тражене вредности, формата и типа траженог података и *view* модела, вратити *bool* вредност као индикатор припадности тог истог модела филтрираној колекцији.

5. ЗАКЉУЧАК

Задатак овог рада била је демонстрација рада .NET платформи и њеном оквиру намењеном web апликацијама – ASP.NET. Проблем библиотечког система треба посматрати као case-study за побољшање вештина у раду са базом, структурама података, JavaScript објектима... Такође имплементација апликације, која је основ овог рада, је служила за боље упознавање са архитектонским и дизајн шаблонима као што су *MVC*, *Repository*, *Builder*, *Factory* и други.

У раду нису приказани сви делови решења већ они који би читаоцу били можда најзанимљивији. Интересантно би такође било описати начин размене порука између корисника апликације помоћу SignalR технологије, као и индексирање и претрагу електронских књига помоћу Lucene.NET библиотеке. Ови делови имплементације се у великој мери свode на коришћење готовог кода из библиотеке па је у раду остављен простор за опис компоненти које су јединствене и специфичне за ово конкретно решење.

Показало се да ASP.NET оквир оставља пуно простора за сопствене идеје и решења, али истовремено пружа стабилно окружење за брз и ефикасан развој web апликација. Конфигурација је врло једноставна и лако је укључити додатне библиотеке.

Овај систем би се могао унапредити у више различитих праваца. Филтрирање помоћу *LMSGrid* компоненте би могло бити проширено додатним параметром по каквом услову се одређени појам филтрира: услов једнакости, услов садржавања, услов веће/мање за бројеве. Тренутно је могуће филтрирати само по једнакости за било који тип. Из перспективе корисничког интерфејса унос појмова за филтрирање би могао да се унапреди тако да се једноставна инпут поља замене компонентама попут *datepicker*-а или *selectbox*-а. У раду са базом би у случају усложњавања шеме и повећања броја рекорда у бази, требало би размотрити о увођењу референцијалног интегритета ради избегавања потенцијалних проблема на апликативном нивоу. *Repository* механизам би у том случају могао и да се прошири неком врстом кеш колекција које би допринеле побољшању перформанси смањењем броја упита ка бази.

6. ЛИТЕРАТУРА

- [1] Microsoft .NET, <https://dotnet.microsoft.com/>
- [2] ASP.NET, <https://docs.microsoft.com/en-gb/aspnet/overview>
- [3] ASP.NET Web Forms, <https://docs.microsoft.com/en-gb/aspnet/web-forms/>
- [4] ASP.NET MVC, <https://docs.microsoft.com/en-gb/aspnet/mvc/mvc4>
- [5] ASP.NET Web Pages, <https://docs.microsoft.com/en-gb/aspnet/web-pages/>
- [6] MVC pattern, <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
- [7] Razor view, <https://weblogs.asp.net/scottgu/introducing-razor>
- [8] Visual Studio IDE, <https://visualstudio.microsoft.com/vs/>
- [9] Lucene.NET, <https://lucene.apache.org/>
- [10] Signal R, <https://docs.microsoft.com/en-us/aspnet/signalr/overview/getting-started/introduction-to-signalr>
- [11] Web sockets, <https://medium.com/@dominik.t/what-are-web-sockets-what-about-rest-apis-b9c15fd72aac>
- [12] jQuery, <https://jquery.com/>
- [13] Autofac, <https://autofacn.readthedocs.io/en/latest/index.html>
- [14] Repository pattern, <https://deviq.com/repository-pattern/>
- [15] Entity Framework, <http://www.entityframeworktutorial.net/what-is-entityframework.aspx>
- [16] ADO.NET, <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/>
- [17] LINQ, <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/>

Кратка биографија:

Јасмина Еминовски рођена је 2.8.1994 у Војнићу, општина Војнић, Р.Хрватска. 2013. године је уписала Факултет техничких наука, одсек *Рачунарство и аутоматика*. Након завршених основних академских студија, уписала је мастер академске студије на истом факултету, смер *Примењене рачунарске науке и информатика, Електронско пословање*.

Милан Видаковић је рођен у Новом Саду 1971. године. На Факултету техничких наука у Новом Саду завршио је докторске студије 2003. године. На истом факултету је 2014. године изабран за редовног професора из области *Примењене рачунарске науке и информатика*.

OPTIMIZACIJA AUTOMATSKOG PODFREKVENTNOG RASTEREĆENJA POTROŠAČA PRIMENOM GENETSKOG ALGORITMA

OPTIMIZATION OF UNDERFREQUENCY LOAD SHEDDING USING GENETIC ALGORITHM

Zoltan Berec, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu predstaviće se rešenje optimizacije Automatskog podfrekventnog rasterećenja potrošača (AFRP) primenom genetskog algoritma. Opisuje se realizacija AFRP-a ILS metodom (Intelligent Load Shedding) koja se koriste za očuvanje stabilnosti kao krajnje opcije i kao rešenje je ponuđen Genetski algoritam kao metod optimizacije za pronalaženje adekvatne liste potrošača za isključenje.

Ključne reči: Stabilnost elektroenergetskog sistema, Genetski algoritam, automatsko podfrekventno rasterećenje potrošača

Abstract – This paper represents solution for optimizing Underfrequency Load Shedding (UFLS) in power electric systems using Genetic Algorithm. Through this paper ILS (Intelligent Load Shedding) method will be described as solutions for preservation of stability in power systems as last option and for solution will propose Genetic algorithm as a method of optimization for UFLS in finding the best suited list of consumers to curtail.

Key words: Power system stability, Underfrequency Load Shedding, Genetic Algorithm

1. UVOD

Razvojem industrije i njene infrastrukture naglasila se potreba za povećanjem potrošnje električne energije. Nesrazmera u velikom rastu opterećenja s jedne strane, i značajno manjem rastu proizvodnih i prenosnih kapaciteta s druge strane, dovelo je do potrebe za pojačanim iskorištenjem postojećih kapaciteta. Prenos velike količine energije kroz mrežu dovodi do rada prenosnih linija blizu njihovih limita. Zbog ovih razloga sistemi napajanja postaju podložni smetnjama i ispadima sistema. Neke smetnje koje se iskuse elektroenergetskim sistemima su greške zbog opreme, gubitak generatora, iznenadna promena potrošnje... Ove smetnje se menjaju po intenzitetu. U nekim trenucima ove smetnje mogu da utiču na sistem da bude nestabilan. Veliki deficit snage u sistemu sa proizvodnjom nedovoljnom da pokrije potrošnju mogu da dovedu do potpunog frekventnog i naponskog sloma odnosno raspada sistema. Kada proizvodni kapaciteti rade na svojim gornjim granicama i kada ne postoji mogućnost za pokrivanje deficita, potrebno je primeniti dodatne akcije, a u cilju sprečavanja opadanja frekvencije i frekventnog sloma.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Platon Sovilj.

U toku vanrednog stanja elektroenergetskog sistema potrebno je vršiti rasterećenje mreže isključivanjem pojedinih manje bitnih potrošača kako bi omogućili snabdevanje potrošača kao poslednji korak očuvanja sistemskog integriteta.

Rasterećenje mreže automatskim podfrekventnim rasterećenjem potrošača - AFRP jedna je od efektivnijih metoda koja očuvava stabilnost elektroenergetskog sistema u kritičnim vremenima i uspostavlja ravnotežu aktivnih snaga u sistemu.

2. Automatsko podfrekventno rasterećenje potrošača

Veoma je važno da elektroenergetski sistem (ESS) osigura dovoljnu generaciju snage kako bi mogle da zadovolje sve potrošače kako u normalnoj fazi rada sistema tako i u kritičnim fazama. U normalnim uslovima EES se održava u balansu generišući snagu dovoljnu da nadomesti sve gubitke i napoji sve potrošače.

$$\text{Proizvodnja} = \text{Potražnja} + \text{Gubici}$$

Pod balansiranim stanjem sistem će funkcionisati pod sinhronom frekvencijom od 50 Hz. Frekvencija u EES-u se menja kroz sledeću tabelu:

Tabela 1. Promena frekvencije u zavisnosti od opterećenja mreže

Stanje EES	Promena frekvencije u sistemu
Proizvodnja > Potražnja + Gubici	Povećanje
Proizvodnja = Potražnja + Gubici	Nema promene
Proizvodnja < Potražnja + Gubici	Smanjenje

2.1. Posledice od prekomerne devijacije učestanosti

Devijacije frekvencije u sistemu moraju biti otklonjene inače rad sistema nije moguć. Proizvodne jedinice i transformatori imaju ugrađene zaštitne frekvencione releje koji ih štite od povećane ili smanjene frekvencije. Generatorske turbine nisu u stanju da podnesu velike smetnje. U slučaju prekoračenja, zaštita se uključuje, isključujući proizvodnju i povećavajući deficit snage.

Posledice povećanja učestanosti na generatorima

Ako frekvencija poraste, generatori osećaju promenu kao gubitak potrošača. Posledično oni će početi da rotiraju brže i primarna regulacija će se uključiti, što će smanjiti mehaničku snagu turbine, posle toga će se frekvencija stabilizovati u stacionarno stanje i generatori mogu normalno da funkcionišu. Ovo predstavlja problem zbog

povećanja napona usled gubitka potrošača. U ovom slučaju generator će biti isključen sa mreže.

Posledice redukovanja učestanosti generatora

U slučaju da su generatori preopterećeni, brzina hlađenja sistema je ispod normalnih vrednosti. Napon opada, regulator povećava jednosmernu pobudu rotora, ovo predstavlja rizik opterećenja rotora i statora. Dozvoljeno vremensko prekoračenje može se izračunati kroz sledeću jednačinu:

$$t = \frac{K}{(x^2 - 1)}$$

gde K predstavlja konstantu koja ima 2 različite vrednosti: 44 je za vremensko opterećenje statora a 33 za rotor; x predstavlja vrednost p.u struje statora ili napona.

Iako većina uređaja može izdržati malo odstupanje frekvencije, jačina i dužina devijacije je strogo ograničena posebno za skupe jedinice kao što su sinhroni generatori. Generatorske turbine su osetljive na rad prilikom odstupanja frekvencije od nominalnih vrednosti, abnormalne frekventne operacije zbog prekomernih vibracija mogu dovesti do istrošenosti i ubrzanog starenja. Prema istraživanjima maksimalno vreme rada parnih turbina prilikom devijacije frekvencije je predstavljen kroz tabelu 2.

Tabela 2. Radna sposobnost parne turbine pri devijaciji frekvencije

Devijacija frekvencije pod punim opterećenjem	Maksimalno vreme rada
$\Delta f = 1\% f_n$	Neprekidno
$\Delta f = 2\% f_n$	100 minuta
$\Delta f = 3\% f_n$	10 minuta
$\Delta f = 4\% f_n$	1 minut
$\Delta f = 5\% f_n$	0.1 minut
$\Delta f = 6\% f_n$	1 sekund

Pomoćni uređaji elektrane kao što su indukcijske mašine u sistemu hlađenja se oslanjaju na stabilnu frekvenciju. Niža frekvencija dovodi do manje brzine rotacije i manje efikasne ventilacije.

2.2. Osnovne funkcije automatskog podfrekventnog rasterećenja potrošača

Svrha ovih korektivnih akcija jeste da se narušeni sistem prinudno vrati u novo stanje ravnoteže, balansiranjem proizvodnje i potrošnje, dakle, održavanjem frekvencije u dozvoljenom opsegu. Ukupnu snagu rasterećenja, je potrebno za različite vrednosti manjka proizvodnje, optimalno rasporediti po stepenima, za svaki stepen karakteriše učestanost na kojoj se okida, količina opterećenja i vremenska razlika između svakog koraka. Konačni cilj je da se izvrši optimalno rasterećenje potrošača za minimalnu potrebnu snagu rasterećenja, a u isto vreme uspostavi brza, mirna i sigurna tranzicija sistema od vanrednog do normalnog stanja.

AFRP mora da ispuni sledeće relevantne funkcije:

- Frekventni odziv nakon poremećaja mora biti kvalitetan, a što podrazumeva brz oporavak frekvencije na vrednost blisku nominalnoj, uz što manje rasterećenje i što manji pad frekvencije ispod nominalne (minimalna vrednost ne sme da pređe 47 Hz).
- Porast frekvencije iznad nominalne treba da je što manji (maksimalna dozvoljena frekvencija ne sme da pređe 51 Hz).

- Potrebno je ostvariti koordinaciju sa podfrekventnom zaštitom generatora.
 - Potrebno je uvažiti kategorije potrošača tj. njihove prioritete.
 - Potrebno je izbeći nepotrebno isključenje potrošača.
- Kompjutersko rešenje je potrebno kako bi se izračunala minimalna potrebna snaga rasterećenja, broj stepeni AFRP, operaciono vreme i snaga rasterećenja za svaki od stepena. Postoji mnogo tipova i rešenja za ove probleme kroz literaturu, ali generalno, može se podeliti u statičke i dinamičke (ili adaptivne) šeme za rasterećenje mreže.

2.3. Mane konvencionalnih APRP šema

Iako konvencionalne APRP šeme imaju široku implementaciju u elektroenergetskim sistemima imaju nekoliko glavnih nedostataka:

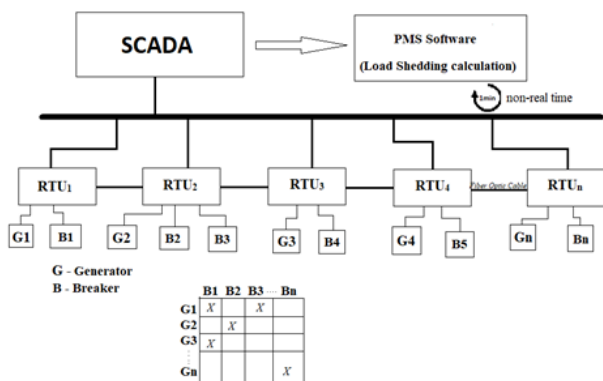
- Ne uzimaju u obzir dinamičko ponašanje opterećenja elektroenergetskog sistema.
- Ne uzimaju u obzir lokaciju poremećaja prilikom rasterećenja mreže.
- Fiksno određen procenat rasterećenja po stepenu ne može se dinamički prilagoditi debalansu snage, što može da dovede do nepreciznog rasterećenja.

Razvojem kompjuterizovanih sistema koji vrše upravljanje elektroenergetskim sistemima, zbog mana konvencionalnih metoda rasterećenja mreža, stvara se mogućnost zamene novim bržim i boljim metodama (ILS- Intelligent Load Shedding).

3. REALIZACIJA AUTOMATSKOG PODFREKVENTNOG RASTEREĆENJA POTROŠAČA ILS METODOM

Razvojem i modernizacijom kompjuterizovanih sistema za upravljanje elektroenergetskih sistema (*Power Management Systems -PMS*) imamo mogućnost za brzo i optimalno upravljanje potrošačima u aktuelnim operacionim uslovima rada sa znanjem o svim prethodnim smetnjama. PMS predstavlja sistem koji u realnom vremenu vrši akviziciju podataka i merenja sa terena i kontinualno ažuriraju kompjuterski model sistema. Jedna od funkcija koju obavlja PMS predstavlja i funkcija rasterećenja mreže (*Load Shedding (LS) function*). Funkciju koristimo za primenu metode pametnog rasterećenja mreže (*ILS*), gde funkcija za sve unapred definisane scenarije vandrednih situacija vrši skladištenje liste potrošača za rasterećenje na RTU ili PLC uređajima. Kada RTU ili PLC uređaj dobije signal koji opisuje određeno vandredno stanje šalje se komanda za isključenje potrošača iz mreže. Na ovaj način omogućujemo veoma brzo delovanje sistema za različite vandredne situacije.

PMS ima sve informacije o potrošačima kao i topologiji mreže. Kroz funkciju rasterećenja mreže vršimo proračun i odabir potrošača za rasterećenje prema prioritetu, statusu i količini potrošnje. Svi proračuni se vrše u ne-realnom vremenu, uzima se trenutak kad se pravi presek stanja i odabir potrošača se vrši za dato trenutno stanje mreže. Svaka lista potrošača se čuva u tabelama na RTU/PLC uređajima po tipu definisanih scenarija. Ova tabela se ažurira nakon svakih 60s, i nakon svakog rasterećenja mreže. Na ovaj način u slučaju neočekivanih ispada sistema RTU/PLC reaguje odma, bez potrebe za komunikacijom ka PMS-u, i tako čuva elektroenergetski sistem od totalnog raspada u roku od 100ms.



Slika 1. Realizacija automatskog rasterećenja mreže ILS metodom

Rasterećenje mreže vršimo kroz sledeće scenarije:

1. Kvara (ispada) generatora
2. Opadanja sistemske frekvencije ispod određenog limita

Kada imamo kvar (ispad) generatora u mreži ne čekamo da sistem reaguje na promene već odmah delujemo rasterećenjem sistema. Sistemska arhitektura je tako osmišljena da može da se reaguje skoro u istom trenutku od događaja kvara.

Rasterećenje mreže se vrši i kada frekvencija opadne ispod određenog limita. Operator mreže bira broj stepeni rasterećenja i zadaje procentualno količinu rasterećenja za svaki od stepena rasterećenja. PMS konstantno vrši merenja i proveru frekvencije u sistemu i kada frekvencija opadne ispod određenog limita vrši se rasterećenje po prvom stepenu. Nakon toga vrše se ponovni proračuni PMS-a i ako su potrebni sledeći stepeni rasterećenja će biti izvršeni. Za sve moguće predviđene scenarije potrošači se uvek biraju po prioritu i količini potrebne snage za rasterećenje. Potrošači sa većim prioritetom se uvek biraju poslednji za isključenje.

3.1. Potreba za optimizacijom ILS metode

Stanje električne mreže se stalno menja zbog promene konfiguracije i opterećenja mreže. Zbog stalnih promena, kroz proračun funkcije moraju stalno da se ažuriraju tabele na PLC/RTU uređajima, zbog čega se funkcija poziva veoma često. Kada imamo mali sistem od po nekoliko potrošača sa jednim ili dva generatora lako je izvršiti proračun i odabir potrošača za sve od mogućih scenarija ispada. Za PMS koji upravlja velikim sistemima gde imamo veliki broj potrošača i veći broj generatora ova funkcionalnost mora da se izvršava jednako brzo. Zbog čestih poziva funkcije bitno je obratiti pažnju na vremensko trajanje proračuna. Takođe ukupna snaga rasterećenja odabrane kombinacije potrošača mora da bude jednaka ili veća od potrebne snage za rasterećenje. Funkcija koja vrši odabir potrošača za rasterećenje mora biti veoma brza zbog stalnih poziva i mora da daje optimalno rešenje za dati problem uz određene kriterijume koje mora da ispunji. Kako bi rešili problem brzine traženja određene kombinacije potrošača za rasterećenje primenjen je genetski algoritam koji služi kao tehnika optimizacije i traženja adekvatnog rešenja.

4. GENETSKI ALGORITAM

Genetski algoritam (GA) predstavlja tehniku optimizacije i pretraživanja baziranu na principima genetike i prirodne selekcije. U genetskom algoritmu imamo populaciju mogućih rešenja za dati problem. Ova populacija rešenja prolazi kroz rekombinaciju i mutaciju (kao u genetici), kreirajući nova rešenja i proces se ponavlja kroz određen broj generacija. Za svako rešenje je dodeljena fitnes vrednost (sposobnost rešenja – opisuje koliko je rešenje dobro) koja se određuje kroz funkciju za dati problem. Rešenja koja imaju najveću fitnes vrednost imaju najveće šanse da učestvuju u kreiranju novih rešenja (odnosno najmanju fitnes vrednost od zavisnosti od postavke problema).

Implementacija genetskog algoritma se vrši kroz sledeće korake [4]:

1. Kreiranje inicijalne populacije
2. Evaluacija sposobnosti svakog pojedinca unutar populacije (*Fitness evaluation*)
3. Primena genetskih operatora

Genetski operatori menjaju genetski sastav potomaka. Predstavljaju ih sledeće procedure i funkcije:

- Selekcija najспособnijih pojedinaca za parenje (*Selection*)
- Rekombinacija izabranih pojedinaca za kreiranje njihovih naslednika-potomaka (*Crossover*)
- Mutacija (*Mutation*) – Mutacija gena
- Dodavanje novih potomaka u populaciju
- Selekcija najboljih u populaciji

U C# programskom jeziku napisana je GA funkcija kao rešenje problema optimizacije za potragu adekvatne liste potrošača. Kako PMS ima sve informacije o sistemu za ulazne parametre genetskog algoritma prosleđuje listu svih trenutno aktivnih potrošača u mreži i željenu snagu rasterećenja za određeni scenario. Potrošači su opisani statusom, opterećenjem (potrošnja izražena u MW) i prioritetom. Ovi ulazni parametri prolaze kroz proces genetskog algoritma i kao izlaz daju najbolje proračunato rešenje, tj listu potrošača za rasterećenje.

Pseudo kod koji opisuje genetski algoritam predstavljen je na sledećoj slici:

```
GA()
    initialize population
    find fitness of population

    while (termination criteria is reached) do
        parent selection
        crossover with probability pc
        mutation with probability pm
        decode and fitness calculation
        survivor selection
        find best
    return best
```

Slika 1 Pseudo kod GA

5. TESTIRANJE NAPISANE GA FUNKCIJE

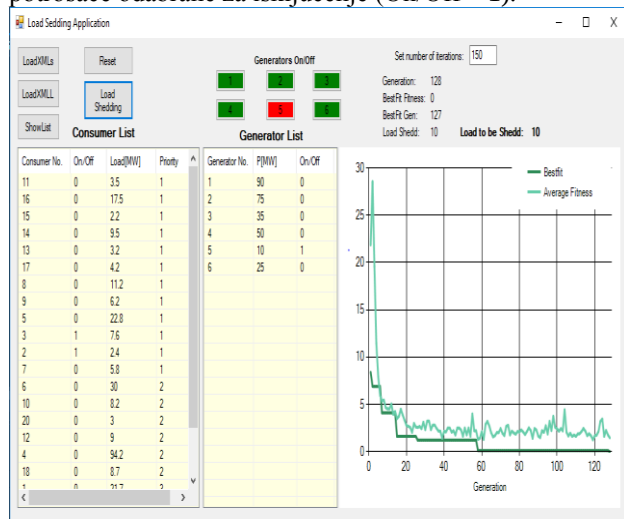
Za potrebe testiranja napisana je aplikacija u kojoj se nalazi i funkcija odabira potrošača za rasterećenje. Aplikacija je pisana u C# programskom jeziku i sačinjena

od UI prozora pomoću kojeg možemo da vidimo rezultate rada osmišljene GA funkcije.

Napravljena su dva statička modela od po 6 generatora i 18 potrošača, odnosno 18 generatora i 48 potrošača. UI aplikacija nudi odabir jednog od dva ponuđena modela. Svaki od modela nudi odabir scenarija, to jest mogućnost uključivanja i isključivanja željenog broja generatora za odabrani model.

Kako za očekivane scenarije očekujemo ispad jednog generatora sledeći primeri testiranje biće bazirani na jednom ispadu generatora.

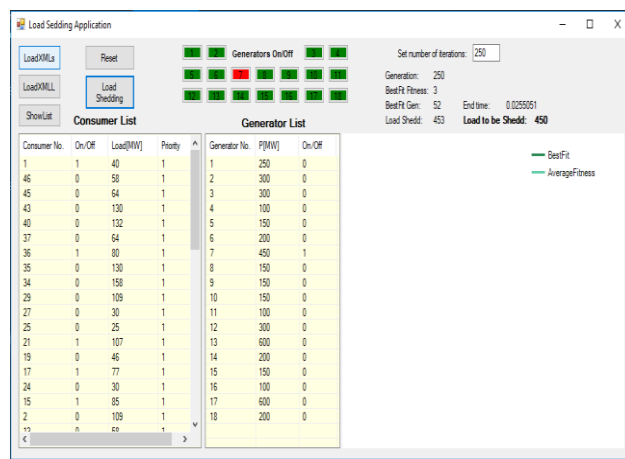
Test 1 - Biramo model od 6 generatora i 18 potrošača. Za određeni scenario uzimamo ispad generatora 5 [10 MW], slika 3. Na ovom primeru želimo da pratimo prosečnu fitnes vrednost cele populacije kroz svaku iteraciju, odnosno želimo da vidimo da se kroz svaku iteraciju dobija populacija boljih rešenja. Za maksimalni broj iteracija algoritma postavljena je vrednost od 150. Na UI prozoru može se videti grafik gde prati ovu promenu kroz svaku iteraciju, a u *Consumer List* možemo videti potrošače odabrane za isključenje (On/Off – 1).



Slika 2 Test 1, isključenje generatora 5

Odabrani potrošači za isključenje su potrošači od 7.6 MW i 2.4 MW, odgovarajuća kombinacija čija suma snage iznosi 10 MW. Svi potrošači su najmanjeg prioriteta (1). Najbolje rešenje je pronađeno u 127 generaciji. Na grafiku se može videti napredak cele populacije ka željenom rešenju. Može se uočiti da na početku GA funkcija veoma brzo napreduje i pronalazi mnogo bolja rešenja u svega nekoliko iteracija, ali to ima tendenciju za zasićenjem u kasnijim fazama.

Test 2 - Biramo model od 18 generatora i 48 potrošača. Testira se brzina algoritma da nađe željenu kombinaciju potrošača. Za scenario se uzima ispad generatora 7 od 450 MW. Za potrebe testiranja brzine isključuje se iscrtavanje grafika, na UI prozoru se dodaje labela **End time** koja opisuje potrebno vreme za završetak pretrage algoritma. Na slici 4 možemo videti sledeće : kroz 250 iteracija algoritma u roku od 25 ms GA funkcija pronalazi kombinaciju potrošača čija je fitnes vrednost 3. Ovo ne predstavlja najbolju moguću kombinaciju potrošača za dati scenario ali je dovoljno dobro rešenje, rasterećenje odabranom kombinacijom je 453 MW. Ovo rešenje je pronađeno u 52-oj iteraciji algoritma. Svi potrošači predviđeni za isključenje su najmanjeg prioriteta 1.



Slika 3 Test 2, testiranje brzine GA funkcije

6. ZAKLJUČAK

Kako nestabilnost elektroenergetskih sistema ima loš uticaj na kvalitet usluge, elektroenergetski sistemi su dizajnirani tako da stabilnost bude uvek očuvana. Kao poslednja odbrana od kolapsa sistema vrši se automatsko rasterećenje mreže, čime se održava balans snaga u mreži. Razvojem kompjuterizovanih sistema koji vrše upravljanje elektroenergetskim sistemima, zbog mana konvencionalnih metoda rasterećenja mreža, stvara se mogućnost zamene novim bržim i boljim metodama (ILS- Intelligent Load Shedding).

Upravljanjem velikim elektroenergetskim mrežama postoji potreba za optimizacijom funkcije odabira potrošača za ILS metode. Kako bi rešili problem optimizacije kao tehniku pretrage i optimizacije koristi se genetski algoritam. Genetski algoritam je razvijen po principima genetike i daje veoma dobre rezultate za različitu vrstu problema. Primenom genetskog algoritma dobili smo funkciju koju smo testirali za dati problem optimizacije i kao rezultat toga na brz način dobili adekvatna željena rešenja kroz različite primere.

7. LITERATURA

- [1] D. S. Popović, "Modelovanje u elektroenergetici", Institut za energetiku i elektroniku, FTN, Novi Sad, 1999
- [2] Meng Zhang, "Modeling and Verification of Advanced UnderFrequency Load Shedding Schemes", University of Technology, Delft, 2016
- [3] Farrokh Shokooh, Shervin Shokooh, Jacques Tastet, Hugo Castro, Tanuj Khandelwal, Gary Donner, "An Intelligent Load Shedding System Application in a Large Industrial Facility", 2005
- [4] Randy L. Haupt, Sue Ellen Haupt, "Practical Genetic Algorithms", February 2004

Kratka biografija



Zoltan Berec je rođen u Novom Sadu 1992. godine. Srednju tehničku školu "Mihajlo Pupin" u Kuli završio 2011. Diplomirao je na Fakultetu tehničkih nauka u Novom Sadu, na katedri za električna merenja 2016. godine.

PRILAGOĐENA KLASIČNA REGULACIJA NAPONA U AKTIVNIM DISTRIBUTIVNIM MREŽAMA

ADAPTED CLASSIC VOLTAGE REGULATION IN ACTIVE DISTRIBUTION NETWORKS

Sladjan Acimović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu predstavljen je jednostavan matematički model kojim se uzima u obzir uticaj distributivnih generatora na regulaciju napona u distributivnim mrežama. Primenom modela je omogućeno da se klasična regulacija napona prilagodi aktivnim distributivnim mrežama.

Ključne reči: Distributivni generator, Regulacija napona, Automatski regulator napona

Abstract – In this paper, simple mathematical model correcting the impact of distributed generation on voltage regulation in distribution networks is presented. Using this model, classical voltage regulation is adapted to active distribution networks.

Keywords: Distributed generation, Voltage regulation, Automatic voltage regulator

1. UVOD

Porast potreba za energijom u svetu mora biti praćen izgradnjom novih elektrana. Pored velikih elektrana koje se priključuju na prenosnu mrežu, u novije vreme sve više se koriste obnovljivi izvori energije i sve više se grade distributivni generatori (DG) koji se priključuju direktno na distributivnu mrežu (DM). Jedna od ključnih prednosti elektrana na obnovljive izvore električne energije je to da se one uglavnom nalaze na distributivnim područjima, pa su gubici mnogo manji [1]. Izvori električne energije su neiscrpn, emisija štetnih gasova je mala, pa su i ovo razlozi koji su doveli do povećanog broja instalacija elektrana na ove izvore [1].

Pored navedenih prednosti, postoje i određeni nedostaci u ovakvom vidu proizvodnje električne energije. Izgradnja DG dovodi do promena strukture DM. Mreže koje su ranije bile radikalne i napajale se iz jednog izvora, sada postaju mreže sa petljama, te se energija potrošačima sada isporučuje i iz samih DG.

Dakle, usled promene strukture mreže javljaju se i određeni negativni efekti priključenja DG. Negativni efekti mogu se prevazići ugradnjom novih vidova zaštite, novih elemenata mreže, promenom preseka vodova i slično. Međutim sve to ima svoju cenu i dobro bi bilo kada bi se uz male promene postojeći kapaciteti mogli iskoristiti što je više moguće.

NAPOMENA:

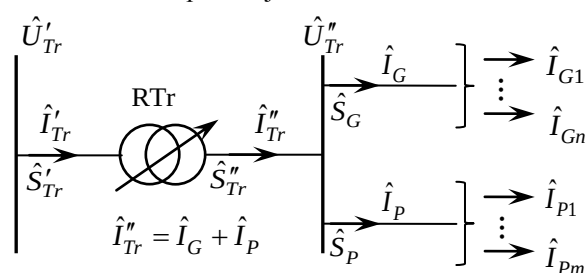
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Goran Švenda, red.prof.

U ovom radu razmatra se uticaj DG na regulaciju napona u DM. U radu je predstavljeno jednostavno i jeftino rešenje na osnovu koga se koriguje lažna slika koja se javlja u vidu struje kroz transformator, koja ne odgovara struji potrošnje. Za potrebe simulacija razvijen je program u programskom jeziku C++. Koristi se proračun tokova snaga zasnovan na algoritmu sumiranja struja [2].

2. MATEMATIČKI MODEL

Uticaj DG na regulaciju napona ogleda se u stvaranju lažne slike potrošnje. U okviru tzv. klasične regulacije napona regulacioni transformator (RTr) na osnovu struje na njegovom sekundaru, koristeći automatski regulator napona (ARN) određuje optimalnu vrednost napona, te eventualno menja prenosni odnos kako bi postigao napon blizak optimalnom. Struja sekundara RTr je u slučaju klasične radikalne DM u stvari ekvivalentna struja potrošnje. Priključenjem DG menjaju se tokovi snaga u mreži, te struja kroz sekundar RTr nije stvarna struja potrošnje. Posledica toga je da na osnovu merenja samo modula struje sekundara nije moguće proceniti prirodu i vrednost opterećenja, a samim tim ni optimalnu vrednost napona, odnosno odgovarajuću poziciju regulacione sklopke. U ovom radu predstavljeno je rešenje ovog problema, koje se zasniva na tome da se ARN-u ne prosleđuje struja sekundara, nego procenjena struja potrošnje. Ta procena se realizuje na osnovu izmerenih vrednosti modula napona na sekundaru RTr, modula struje na sekundaru RTr i modula struje na početku izvoda sa DG (pri čemu je vrednost faktora snage DG poznata).

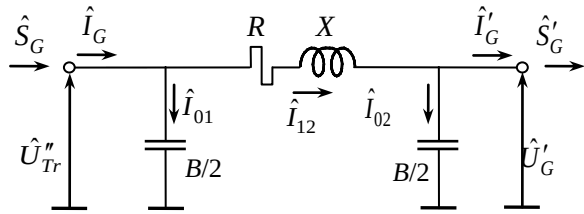
U svrhu izvođenja matematičkog modela, posmatra se jednostavna DM na slici 1. DM se sastoji iz RTr, izvoda sa DG i izvoda sa potrošnjom.



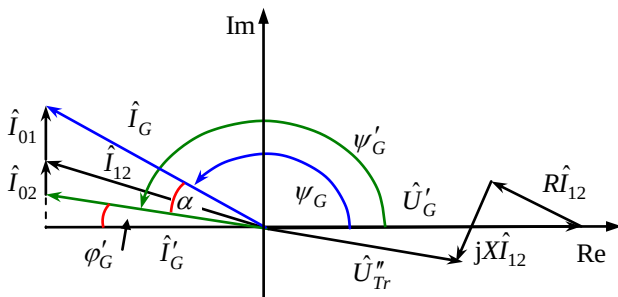
Slika 1 – Jednostavna DM

Na početku je potrebno proceniti režim izvoda sa DG. Ekvivalentna šema izvoda prikazana je na slici 2. Na osnovu unapred definisane vrednosti faktora snage DG

$\cos \phi'_G \approx 1$ i izmerenih vrednosti modula struje I_G i napona U''_{Tr} na početku izvoda, može se proceniti kompleksna struja $\hat{I}_G$. Fazori struja i napona razmatranog izvoda prikazani su na slici 3. Sa α je označen ugao između struja na kraju i na početku izvoda sa DG.



Slika 2 – Izvod sa DG



Slika 3 – Vektorski dijagram struja i napona izvoda sa DG

Na osnovu slike 3 može se izvesti izraz za proračun vrednosti ugla α . U tu svrhu, potrebno je struju na kraju izvoda sa DG izraziti preko aktivne i reaktivne snage DG, pa potom izraziti struju na početku ovog izvoda preko struje na kraju izvoda i otočnih struja. Pritom, potrebno je uvesti aproksimaciju da je napon na kraju izvoda sa DG jednak naponu na njegovom početku:

$$\hat{U}''_{Tr} \approx \hat{U}'_G = U''_{Tr} \angle 0^\circ. \quad (1)$$

Izjednačavanjem realnih delova struja $\hat{I}_G$ i $\hat{I}'_G$ i otočnih struja sa početka i kraja voda (posledica izraza (1), slika 3), primenom kosinusne teoreme na trougao sa stranicama I_G , I'_G i $I_{01} + I_{02}$ može da se proceni vrednost ugla:

$$\alpha = \arcsin \frac{B \cdot U''_{Tr}}{I_G} \cos \phi'_G, \quad (2)$$

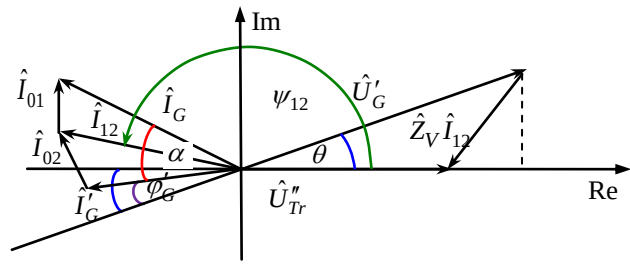
gde je B ukupna otočna susceptansa izvoda sa DG.

Konačno, na osnovu procenjene vrednosti ugla α i izmerene vrednosti modula, struja na početku izvoda sa DG ima sledeći kompleksni oblik:

$$\hat{I}_G = -I_G \cos(\phi'_G + \alpha) + j \cdot I_G \sin(\phi'_G + \alpha). \quad (3)$$

2.1 Korekcija ugla napona generatora

Aproksimacija koja se pravi izrazom (1) u zavisnosti od režima, može da prouzrokuje značajnu grešku pri proceni fazora struje potrošnje. U nastavku je pokazan postupak za korekciju greške koja je navedenom aproksimacijom načinjena. U tu svrhu, na slici 3 prikazan je novi vektorski dijagram struja i napona izvoda sa DG. Na ovoj slici usvojeno je da je fazor napon na početku izvoda sa DG referentni fazor, a da fazor napona na kraju izvoda prednjači za ugao θ .



Slika 4 – Vektorski dijagram struja i napona izvoda sa DG sa uvažanim uglom θ

Na osnovu dijagrama sa slike 4 sledi:

$$\theta = \arcsin \left(\frac{|\text{Im}\{\hat{Z}_V \cdot \hat{I}_{12}\}|}{U'_G} \right), \quad (4)$$

gde je :

$\hat{Z}_V$ – kompleksna impedansa izvoda sa DG,

$\hat{I}_{12}$ – struja duž izvoda sa DG.

Pretpostavlja se da fazor struje $\hat{I}_{12}$ približno deli ugao α na pola, zatim da je moduo napona na kraju izvoda sa DG U'_G približno jednak modulu napona na sekundaru RTr U''_{Tr} , te da je $I_{12} \approx I_G$. Imaginarni deo kompleksnog pada napona na izvodu sa DG je negativan (slika 4), pa se umesto apsolutne vrednosti u izrazu (4) uzima negativna vrednost. Kompleksna impedansa izvoda definisana je na osnovu poznavanja rezistanse i reaktanse izvoda. Izraz (4) na kraju se može zapisati kao:

$$\theta = \arctg \left[\frac{-A \sin B}{1 + A \cos B} \right], \quad (5)$$

gde je :

$$A = \frac{Z_V \cdot I_G}{U''_{Tr}}, \quad (6a)$$

$$B = \phi_V + 180^\circ - \frac{\alpha}{2} - \phi'_G, \quad (6b)$$

pri čemu je ϕ_V ugao kompleksne impedanse izvoda $\hat{Z}_V$.

Konačno, struja na početku izvoda sa DG ima sledeću kompleksnu vrednost:

$$\hat{I}_G = -I_G \cos(\phi'_G + \alpha - \theta) + j \cdot I_G \sin(\phi'_G + \alpha - \theta). \quad (7)$$

2.2 Procena režima transformatora

Za procenu ukupne struje potrošnje, potrebno je osim procene struje na početku izvoda sa DG proceniti i kompleksnu vrednost struje na sekundaru RTr (poznata je samo vrednost modula te struje). Pad napona na RTr, odnosno razlika modula napona primara svedenog na sekundar i napona sekundara je približno jednaka realnom delu kompleksnog pada napona (promene napona [4]):

$$\frac{U'_{Tr}}{m_{12}(t)} - U''_{Tr} = dU''_{Tr} \approx \text{Re}\{\Delta \hat{U}''_{Tr}\}. \quad (8)$$

Iz prethodnog izraza sledi:

$$dU''_{Tr} = X''_k \cdot \hat{I}''_{Tr} \cdot \cos(90^\circ + \psi''_{Tr}), \quad (9)$$

gde je:

- X_k'' reaktansa kratkog spoja RTr (svedena na sekundar,
 I_{Tr}'' moduo struje kroz sekundar RTr,
 ψ_{Tr}'' ugao struje kroz sekundar RTr.

Izraz za procenu vrednosti ugao struje na sekundaru RTr:

$$\psi_{Tr}'' = -90^\circ \pm \arccos \left[\frac{dU_{Tr}''}{X_k'' \cdot I_{Tr}''} \right] \quad (10)$$

Predznak drugog dela izraza definisan je sa smerom toka aktivne snage i vrednošću pada napona na RTr (odnosno toka reaktivne snage) [3]. Pregled mogućih slučajeva dat je u tabeli 1.

Tabela 1 – Četiri moguća rešenja za ugao ψ_{Tr}''

dU_{Tr}''	P	Znak u izrazu za ψ_{Tr}''
>0	>0	+
>0	<0	–
<0	>0	+
<0	<0	–

2.3 Procena struje potrošnje

Konačno, procenjena vrednost modula struje ukupne potrošnje, modul struje koju treba proslediti u ARN jeste:

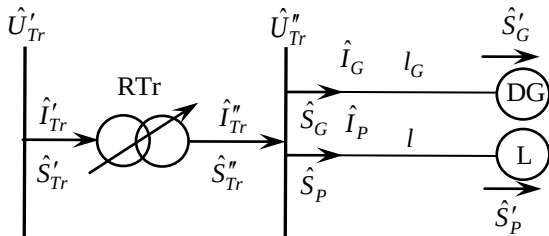
$$I_{ARN} = \{ [I_{Tr}'' \cos \psi_{Tr}'' + I_G \cos(\phi_G' + \alpha)]^2 + [I_{Tr}'' \sin \psi_{Tr}'' - I_G \sin(\phi_G' + \alpha)]^2 \}^{1/2}. \quad (11)$$

Ukoliko se izvrši i dodatna korekcija, u vidu procene ugla θ , na osnovu izraza (5), prethodni izraz glasi:

$$I_{ARN} = \{ [I_{Tr}'' \cos \psi_{Tr}'' + I_G \cos(\phi_G' + \alpha - \theta)]^2 + [I_{Tr}'' \sin \psi_{Tr}'' - I_G \sin(\phi_G' + \alpha - \theta)]^2 \}^{1/2}. \quad (12)$$

3. VERIFIKACIJA MATEMATIČKOG MODELA

Verifikacija mogućnosti prikazanog modela za procenu modula struje potrošnje u DM sa DG, izvršena je na primeru jednostavne test mreža prikazana na slici 5. U tu svrhu razvijen je program u programskom jeziku C++.



Slika 5. Test mreža

Tablični podaci RTr dati su u tabeli 2, a podaci o DM u tabeli 3.

Tabela 2 – Tablični podaci RTr

Prenosni odnos	(110±10×1.25%/20) kV/kV
Nominalna snaga	63 MVA
Napon kr. spoja	5 %
Aktuelna pozicija	0

Tabela 3 – Podaci o DM

l_G [km]	l [km]	Z_G [Ω /km]	Z_P [Ω /km]
10	10	0.420+j0.390	0.410+j0.360

U tabeli 4 prikazane su vrednosti od interesa, za tri karakteristična režima razmatrane DM: 1) DG ne radi, 2) DG injektira 5 MW i 3) DG injektira 10 MW. Sa boldom su naglašene vrednosti koje se koriste kao merenja (vrednosti potrebne za proračune).

Tabela 4 – Rezultati proračuna tokova snaga

	Režim 1	Režim 2	Režim 3
S_P'	5.0–j2.5	5.0–j2.5	5.0–j2.5
S_G'	0.0–j0.0	5.0–j0.0	10.0–j0.0
$\cos \phi_G'$	1.0	1.0	1.0
$\hat{I}_G$ [A]	11.5 ∠89.7°	137.8 ∠177.6°	264.1 ∠–177.4°
$\hat{I}_P$ [A]	170.7∠–24.8°	170.7∠–24.6°	170.8∠–24.4°
$\hat{I}_{Tr}''$ [A]	166.2 ∠–21.2°	67.6 ∠–75.0°	136.0 ∠–142.7°
$\hat{U}_{Tr}'$ [kV]	63.5 ∠0.0°	63.5 ∠0.0°	63.5 ∠0.0°
$\hat{U}_{Tr}''$ [kV]	11.5 ∠–0.3°	11.5 ∠–0.0°	11.5 ∠0.2°

U tabeli 5 prikazane su vrednosti karakterističnih veličina, koje su potrebne za procenu vrednosti struje potrošnje, dobijene na osnovu proračuna tokova snaga. Procena tih vrednosti, samo na osnovu izmerenih vrednosti (vrednosti u tabeli 4 naznačene sa boldom), primenom postupka koji je prikazan u glavi 2, prikazane su u tabeli 6. Procena je izvršena sa i bez korekcije ugla θ .

Tabela 5. Vrednosti karakterističnih veličina i struje potrošnje (tokovi snaga)

	Režim 1	Režim 2	Režim 3
α [°]	89.70	4.92	2.61
$\text{Re}\{\Delta \hat{U}_{Tr}''\}$ [V]	19.09	20.73	26.16
ψ_{Tr}'' [°]	–21.20	–74.97	–142.7
θ [°]	–0.12	2.54	5.00
I_P [A]	170.69	170.73	170.84

Tabela 6. Vrednosti karakterističnih veličina i struje potrošnje (matematički model)

	Režim 1	Režim 2	Režim 3
α [°]	87.47	4.80	2.50
dU_{Tr}'' [V]	18.98	20.73	26.10
ψ_{Tr}'' [°]	–21.08	–74.95	–142.81
θ [°]	–0.01	2.42	4.54
I_{ARN} bez θ [A]	171.19	172.89	181.53
I_{ARN} sa θ [A]	171.19	170.71	171.74

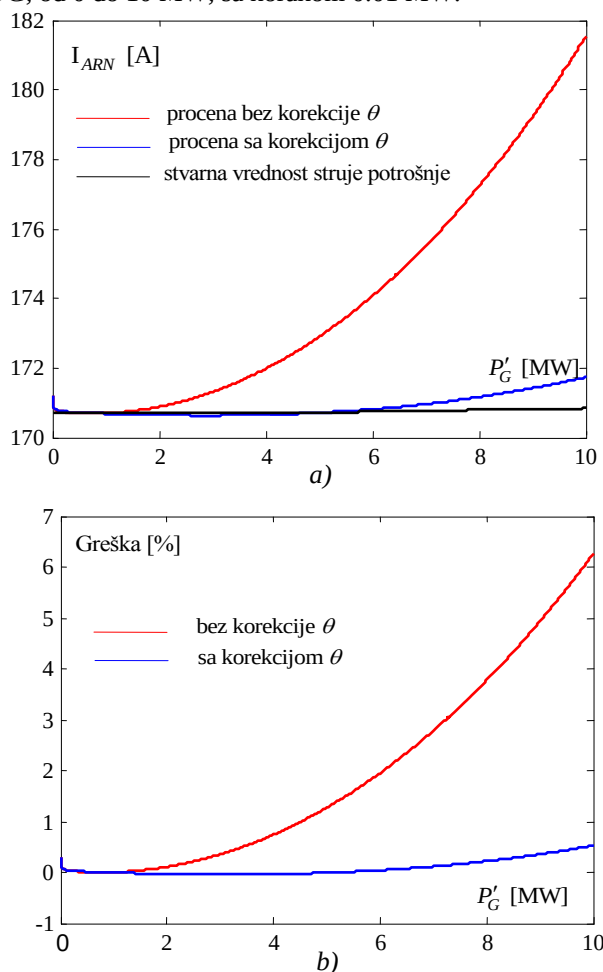
Poredeći tačne vrednosti, koje su prikazane u tabeli 5 sa procenjenim, izračunatim vrednostima, koje su prikazane u tabeli 6, može se uočiti sledeće:

- izračunate vrednosti svih veličina veoma su bliske tačnim vrednostima;
- ako se procena modula struje potrošnje uradi sa

korekcijom ugla θ na kraju izvoda sa DG, primenom izraza (12), dobija se izuzetno dobra vrednost (veoma bliska izračunatoj vrednosti struje potrošnje, za sva tri režima);

- ako se procena uradi bez korekcije ugla θ , primenom izraza (11), greška u proceni struje potrošnje raste sa porastom snage DG.

Na slici 6a prikazana su odstupanja procenjene vrednosti modula struje potrošnje, u odnosu na tačnu vrednost (proračun tokova snaga, crna linija). Procene su urađene bez korekcije ugla θ , crvena linija, i sa korekcijom ugla, plava linija. Na slici 6b je prikazana greška u procentima, u odnosu na stvarnu vrednost struje potrošnje. Analiza je urađena za jednostavnu DM, sa konstantnom potrošnjom 5.0–j2.5 MVA i različitom proizvodnjom aktivne snage DG, od 0 do 10 MW, sa korakom 0.01 MW.



Slika 6. a) Procena vrednosti modula struje potrošnje, b) Greške u odnosu na stvarnu vrednost

Može se uočiti da se za male vrednosti aktivne snage DG dovoljno tačne vrednosti dobijaju primenom oba modela, sa i bez korekcije ugla θ . Sa porastom vrednosti aktivne snage DG raste greška u proceni modela bez korekcije ugla θ , ali ne i greška modela sa korekcijom ugla θ . Na slici 6b može se uočiti da je prim malom uticaju DG, greška koja se pravi primenom modela bez korekcije ugla θ , praktično zanemarivo mala.

4. ZAKLJUČAK

U skladu sa savremenim trendovima u elektroenergetici, u mnogim državama sveta sve više se koriste obnovljivi izvori električne energije i izgrađuje se sve veći broj distributivnih generatora. U skladu sa tim, struktura distributivnih mreža se menja, pasivne radijalne mreže transformišu se u aktivne mreže sa petljama. Kao posledica toga, distributivna preduzeća će morati da se prilagode ili u potpunosti promene klasične principe svoga delovanja. Ugradnjom distributivnih generatora struja sekundara transformatora više ne odgovara struji potrošnje, te u skladu sa tim ARN donosi odluke na osnovu pogrešnih informacija - lažne slike o opterećenju. U okviru ovog rada predloženo je jednostavno rešenje ovog problema. Rešenje se sastoji iz modifikacije postojećeg sistema klasične regulacije napona, gde se struja na sekundaru koriguje koristeći postojeća SCADA merenja. Na ovaj način izbegnuta je skupa nadogradnja i ulaganje u distributivne mreže. U radu su razmatrana dva načina za procenu struje potrošnje: bez korekcije ugla na kraju izvoda sa generatorom (ugao θ) i sa navedenom korekcijom. Pokazano je da se za različite vrednosti snage generatora postiže prilično dobra procena struje potrošnje u oba slučaja, s tim da je za veće snage generatora ugao θ značajniji, pa se bez njegove korekcije dobija nešto veća greška u odnosu na stvarnu vrednost struje potrošnje. Odgovarajućom procenom struje potrošnje, automatski regulator napona će moći da odredi optimalnu vrednost napona na sekundaru transformatora, odnosno odgovarajuću poziciju regulacione sklopke.

5. LITERATURA

- [1] F.A.Viawan: *Voltage Control and Voltage Stability of Power Distribution Systems in the Presence of Distributed Generation*, PhD Thesis, Chalmers University of technology Göteborg, Sweden, 2008.
- [2] G.Švenda: *Specijalizovani softveri u elektroenergetici*, predavanja na FTN, Novi Sad, 2018.
- [3] G.Švenda, Z.Simendić: *Advanced voltage control in distribution network with DG*, International Conference Energy and Ecology Industry, October, Belgrade, 2018.
- [4] V.Strezoski: *Osnovi elektroenergetike*, Fakultet tehničkih nauka, Novi Sad, 2014.

Kratka biografija:



Slađan Aćimović rođen je u Trebinju 1994. godine. Diplomski rad na Fakultetu tehničkih nauka u Novom Sadu odbranio je 2017. godine u oblasti Elektrotehnika i računarstvo, smer Elektroenergetski sistemi. Iste godine upisuje master studije na istom smeru.

**AUTOMATSKO UPRAVLJANJE VIŠE SPREGNUTIH KLIMA KOMORA I
BALANSIRANJE HVAC SISTEMA ČISTIH PROSTORA U NATPRITISKU****AUTOMATIC CONTROL OF MULTIPLE AIR HANDLING UNITS SUPPLYING CLEAN
AIR IN OVER PRESSURE AREAS**

Jovan Marković, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO - Merenje i Regulacija

Kratak sadržaj – U ovom radu je prikazan elektromotorni pogon i BMS sistem čistih prostora objekata koji su u natpritisaku za proizvodnju farmaceutskih proizvoda.

Ključne reči: elektromotorni pogon, BMS, čisti prostori, natpritisak

Abstract – This paper presents design project of the BMS system of clean areas in over-pressure for the production of pharmaceutical products.

Keywords: BMS (building management system), clean areas, over-pressure

1. UVOD

Fabrički kompleks „Fresenius Medical Care Srbija“ se nalazi u Tehnološkom parku u Vršcu i sastoji se iz više objekata: portirnice, glavnog objekta za proizvodnju medicinskih sredstava, energo bloka, skladišta opasnih materija, nadstrešnice za palete, nadstrešnice za kontejnere za sekundarne sirovine, kao i od podzemnih instalacija koje povezuju energanu i napojne vodove sa objektima.

Projekat EMP i automatike urađen je na osnovu projektnog zadatka, arhitektonsko-građevinskih podloga, tehnološkog projekta i projekta termotehničkih instalacija, koji sadrže podatke o opremi, nameni prostora, kao i zahtevane parametre koje treba ostvariti projektovanim sistemom. Projektom EMP i automatike se definiše kontrolni sistem (CS) koji upravlja procesom.

Projektnim zadatkom i tehnološkim projektom se ne zahteva praćenje stanja svih tehničkih sistema (hidrotehničkih instalacija, PP vode, sprinklera, statusa rada gasne kotlarnice i drugih pratećih sistema fabrike za proizvodnju medicinskih sredstava) preko kontrolnog sistema. Prate se samo statusi rada i kvara kotlarnice i sprinkler sistema.

Kontrolni sistem meri, reguliše i upravlja procesom. Sa stanovišta dobre proizvođačke prakse (GMP), kontrolni sistem treba da obezbedi sledeće:

Kontrolni sistem se mora realizovati kao zatvoren kontrolni sistem, tj. kao sistem sa kontrolisanim pristupom gde će se dodatno, u vidu elektronskog zapisa, beležiti i arhivirati svaka prijava, odjava i bezuspešan pokušaj prijave korisnika u sistem.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio prof. dr Zoran Mitrović.

- Komandovanje, vizuelizaciju procesa i implementaciju svih potrebnih regulacija
- Praćenje i arhiviranje svih GMP kritičnih procesnih veličina
- Detekciju alarmnih stanja i arhiviranje alarmnih poruka
- Obradu podataka.

Kontrolni sistem se sastoji iz 3 nivoa:

NIVO OPREME U POLJU

Senzori i aktuatori, koji omogućavaju vezu nivoa automatizacije-upravljačkog sistema sa fizičkim parametrima procesa

NIVO AUTOMATIZACIJE (upravljački sistem)

PLC kontroleri koji omogućavaju:

- prihvatanje fizičkih merenja,
- regulaciju i vođenje procesa po unapred postavljenim parametrima
- vezu sa nadzornim nivoom
- praćenje parametara rada u toku samog delovanja sistema
- vođenje čitavog procesa bez komunikacijske veze sa nadzornim nivoom

NIVO NADZORA I UPRAVLJANJA (CSNU)

Prikaz, kontrola, arhiviranje, alarmiranje i protokoliranje za nadzirani objekat/objekte i samih veza između pojedinačnih nivoa

Uređaji u polju mogu pripadati grupi procesne instrumentacije (Process Instrumentation) ili pripadati grupi izvršnih elemenata-Aktuatora.

Uređaji iz grupe procesne instrumentacije (PI) pretvaraju fizičku veličinu u strujni signal. Strujni signali se kablovima zaštićenim od elektromagnetnih smetnji dovode na analogne i digitalne ulaze programabilno-logičkih kontrolera (PLC). PLC-ovi se nalaze u ormanima automatike i sa pratećom opremom prikupljaju signale iz polja i predaju naloge izvršnim elementima. Izvršni elementi služe da u skladu sa PLC programom izvršavaju aktivnosti propisane ovim projektom.

2. MODOVI RADA KLIMA SISTEMA

Klima sistemi su međusobno nezavisni. To znači da je moguće nezavisno svaki sistem uključiti, isključiti ili upravljati njime u ručnom modu.

Ručni mod klima sistema

Ovaj mod podrazumeva ručno upravljanje svakim izvršnim elementom ponaosob. Operater će sa HMI

određivati položaj elementa i njihovo stanje. Ručni mod će služiti prilikom ispitivanja opreme ili nekih specifičnih radnji. Ne preporučuje se upotreba ručnog moda u normalnom radu.

Automatski mod klima sistema

Operator jednostavnim komandama START/STOP pokreće, odnosno zaustavlja sistem. Prilikom komande starta, program prolazi kroz nekoliko pripremnih faza, da bi na kraju ušao u fazu RAD. Ovaj mod se preporučuje za svakodnevni rad.

Hibridni mod klima sistema

Kada je sistem u automatskom radu, moguće je izabrati element prebaciti u ručni mod i njime upravljati po želji. Ovaj mod je pogodan za rad ako dođe do kvara na instrumentaciji, pa je potrebno „premostiti“ automatsku regulaciju do neophodne poravke. Ne preporučuje se upotreba hibridnog moda u normalnom radu.

Eko mod klima sistema

Kada je sistem u automatskom radu, moguće ga je prebaciti u eko mod kada će sistem raditi sa smanjenim protokom, a uz održavanje kaskada pritisaka.

3. REGULACIJA PROTOKA VAZDUHA PO SOBAMA

Za regulacije protoka vazduha na ubacivanju u sobe se koriste CAV (constant air volume) elementi. Setpointi na CAV-ovima se postavljaju interno na samom uređaju, nezavisno od CS i oni nezavisno od CS održavaju konstantan protok, dok se CS-u šalje samo podatak o trenutnom protoku vazduha.

Za vrednosti internih setpointa treba konsultovati Tehnološki i Mašinski projekat.

Usled zahteva korisnika za održavanjem projektovanih pritisaka u ekonomskom režimu rada, na CAV-ovima će biti podešena 2 setpointa protoka. Vmax će biti setpoint protoka za normalan režim rada, a Vmin će biti setpoint redukovano protoka, za ekonomski režim rada. Vrednost Vmax će biti u skladu sa projektovanim vrednostima, dok će se vrednost Vmin naknadno odrediti u dogovoru sa investitorom, a konsultujući Tehnološki i Mašinski projekat.

CS će određivati koji od ta dva setpointa će CAV-ovi održavati, tj. CS će određivati da li će CAV-ovi održavati Vmin ili Vmax.

CAV uređaji će održavati zadovoljavajući protok ako potisni ventilator zadovoljavajuće održava protok.

Korišćeni su CAV pokretači proizvođača Belimo, tip: LMV-D3-MP, slika 1.



Slika 1. CAV regulator LMV-D3-MP

4. REGULACIJA PRITISKA PO SOBAMA

Ova regulacija postoji u čistim sobama gde je potrebno održavati konstantan pritisak. Ova regulacija se ostvaruje elementom za regulaciju pritiska (VAV) i senzorom diferencijalnog pritiska u sobi. Setpoint se zadaje na CSNU i odgovara vrednostima koje su specificirane zahtevima iz Tehnološkog i Mašinskog projekta. Kao podrška radu VAV-ova se koristi ventilator izvlačenja.

U slučaju pojave kritičnog prevelikog ili premalog pritiska koji bi mogao da ugrozi plafon sobe, sistem odlazi u STOP.

NAPOMENA: U cilju bezbednog održavanja kaskade pritisaka između propusnika i prostorija, kontrolni sistem upravlja radom interlokova sa dvoje ili troje vrata.

Korišćen je VAV pokretač proizvođača Belimo, tip: NM 24A-SR, slika 2.



Slika 2. VAV regulator NM-24A-SR

Korišćen je senzor diferencijalnog pritiska proizvođača Siemens tip QBM 4100-1D, slika 3.



Slika 3. Siemens QBM 4100-1D

5. OPIS RADA SISTEMA K8: AV SISTEMI

Sistem K8 je predviđen za potrebe AV sistema. Namenjen je za vazdušno grejanje, hlađenje i ventilaciju. Sistem je sa 100% svežim vazduhom.

Ovaj sistem pokriva prostorije koje su klasificirane kao ISO 8. Jedan od kriterijuma je bio i minimalan broj izmena vazduha.

Klima komora K8_V1/K8_O1 se sastoji iz: usisne sekcije, prigušivača, **glikolskog rekuperatora**, ventilatora, mešne sekcije, filtera G4, toplovodnog grejača, hladnjaka, ventilatora i filtera F9. Sistem ima mogućnost da radi i sa recirkulacijom vazduha, te je zato predviđena mešna komora. U nominalnom režimu rada sistem radi sa 100 % svežim vazduhom. Količina odsisnog vazduha je dimenzionisana na maksimalni kapacitet, ali se, ukoliko rade lokalni odnosi, količina odsisnog vazduha smanjuje.

Sa ovim sistemom spregnuti su sistemi K8_O2, K8_O3, K8_O4, K8_O5, K8_O6.

Svi sistemi osim sistema K8_O5 su namenjeni za odsis sa AV linije. AV linija je tehnološki uređaj koji na 3 mesta ima priključke za otpadni vazduh. U kanalu iznad plafona čiste sobe predviđen je rekuperator toplote.

Sistem K8_O5 je predviđen za lokalne odsise sa stolova gde se vrši rad sa organskim rastvaračima. U kanalu iznad plafona čiste sobe predviđen je rekuperator toplote.

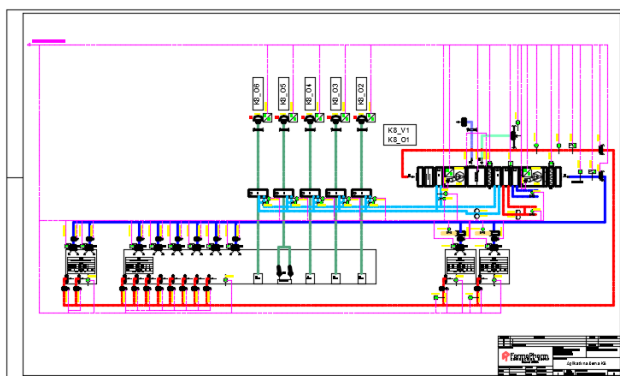
Glikolski rekuperatori u odsisnim sistemima su povezani u jedan sistem i dalje su povezani sa rekuperatorom u potisnoj sekciji. Predviđena je jedna zajednička centralna pumpa, dok je rekuperator u potisnoj sekciji sa trokrakim ventilom.

Za kancelariju je predviđen kanalski grejač koji dogрева vazduh i na taj način vrši lokalnu regulaciju temperature. Održavanje pritiska se vrši pomoću regulatora protoka na izvlačenju. Na ubacivanju se nalaze regulatori protoka koji obezbeđuju konstantnu količinu vazduha na ubacivanju.

Ubacivanje vazduha se vrši kroz difuzore sa ugrađenim HEPA filterima.

6. APLIKATIVNA ŠEMA JEDNOG OD SISTEMA VENTILACIJE ČISTOG PROSTORA U NATPRTISKU K-8

Aplikativna šema detaljno prikazuje spisak signala koji se preuzima sa periferne opreme u polju, kao i tip signala. (analogni ili digitalni). Signali se obrađuju u PLC-u (Siemens S7-400) i na osnovu predefinisanih programa vrši se delovanje na izvršne elemente, u ovom slučaju CAV i VAV regulatore čijim upravljanjem se i obezbeđuju projektovani parametri u prostoru. U nastavku je prikazana aplikativna šema klima komore K-08.



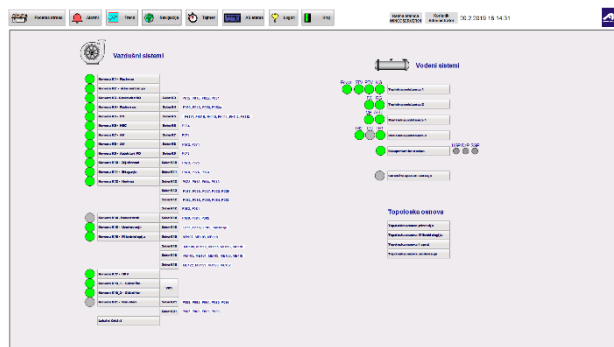
Šema 1. Aplikativna šema klima komore K-08

7. SKADA PRIKAZ SISTEMA K8

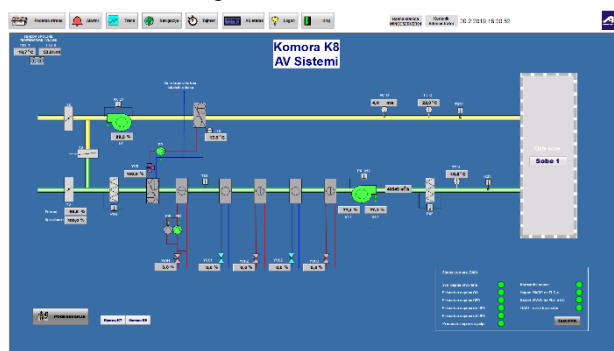
Zbog vrlo pozitivnog iskustva, jako velikih mogućnosti, visoke pouzdanosti i fleksibilnosti, projektom je predviđeno da se kao bazni programski paket za realizaciju CSNU koristi Vijeo Citect sa svim svojim komponentama.

Redundantni SCADA serveri su predviđeni zbog sigurnosti, s obzirom da su svih 12 PLC-ova povezani sa jednim serverom. U normalnom radu u funkciji je primarni (Primary) server dok je sekundarni (Stand By) u „stand by“ modu. U slučaju nekog problema sa računarom primarnog servera ili sa njegovom komunikacijom sa PLC-ovima, sekundarni server automatski prelazi u funkciju

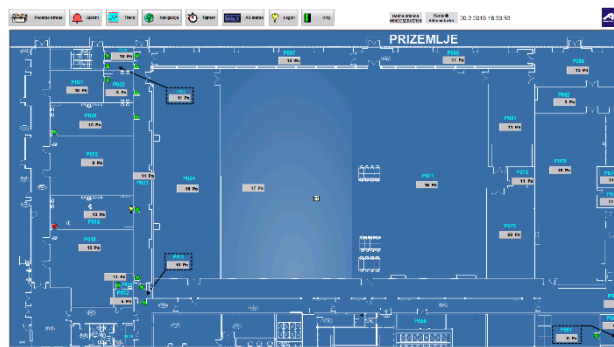
dok se primarni ponovo ne osposobi. Samo serveri obavljaju komunikaciju sa PLC-ovima i na njima se realizuju sve funkcionalnosti SCADA sistema. Svaki server se sastoji od IO servera, Alarm Servera, Trend servera i Report servera.



Slika 4. skada, početni ekran sa klima komorama



Slika 5. skada, početni ekran sa klima komorom K-8



Slika 6. skada, osnova sa prikazom ostvarenih pritisaka

8. ZAKLJUČAK

U radu su prikazani osnovni delovi elektromotornog pogona i BMS-a (building management system) primenjeni u fabričkom kompleksu „Fresenius Medical Care Srbija“ u Vršcu, koji je po svojoj prirodi, s obzirom na to da je u pitanju proizvodnja lekova i koncentrata, na najvišem mogućem nivou. Prikazan je princip merenja realnih parametara u polju, putem najpreciznije periferne senzoričke i ostvarivanje zadatih parametara delovanjem na izvršne elemente u polju.

Pošto su u pitanju osetljivi i precizni signali, poseban akcenat je stavljen na instalaterski deo, gde je prikazana tačna definicija signalnih kablova položenih za međuvezu ormana elektromotornog pogona periferne opreme, kako bi se eliminisala mogućnost smetnji i slabljenja signala kroz samu položenu instalaciju.

Kako je zaustavljanje rada ventilacije i zatvaranje PP klapni (protivpožarnih klapni) na granicama požarnih sektora preduslov za primanje objekta od strane PP inspekcije, eksplicitno su navedeni položaji, instalacije i tipovi ugrađenih PP klapni.

Korisnički deo, vizualizacija BMS sistema, tj. skada soft-ver je prikazan kroz nekoliko slika realno izvedenog i u potpunosti funkcionalnog sistema koji je u eksploataciji više od tri godine.

Kratka biografija:

Jovan Marković rođen je u Somboru 1985. god. Master studije mehatronike završio je 2013. god. Tokom školske 2018/2019. upisao se na master studije na studijskom programu Merenje i regulacija na Fakultetu tehničkih nauka u Novom Sadu.

kontakt: jovanmax@gmail.com

MIKROPROCESORSKA ZAŠTITA ELEKTROMOTORA MICROPROCESSOR PROTECTION OF ELECTRICAL MOTORS

Bilja Sjeran, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu se razmatraju savremeni mikroprocesorski zaštitni uređaji čija je funkcija da zaštite električne motore kao najzastupljenije električne mašine u elektroenergetskom sistemu. Cilj je da se prikažu prednosti i različite pogodnosti primjene ovakve vrste zaštite koja omogućava pouzdanost, te detaljnu analizu i predviđanje opasnih stanja u cilju njihovog sprečavanja. Razmatranja su izvršena na primjeni zaštitnog uređaja tipa „Sepam M61“ za zaštitu elektromotora.

Ključne reči: mikroprocesorska zaštita, zaštitni releji, elektromotori

Abstract – This paper deals with modern microprocessor protection devices whose function is the protection of electric motors as the most commonly used electric machine in the electric power system. The aim is to show the benefits and various benefits of applying this kind of protection that enables reliability, detailed analysis and predicting dangerous states in order to prevent it. Considerations have been made on the use of the „Sepam M61“ protective device for the protection of electric motors.

Keywords: microprocessor-based protection, protection relays, electric motors

1. UVOD

U ovom radu razmatra se realizacija zaštite motora mikroprocesorskim zaštitnim uređajima [1,2]. Mikroprocesorska zaštita podrazumjeva zaštitu koja je potpuno numerički orijentisana. Dakle, svi ulazi u dio u kojem se obrađuju mjerni signali su numeričke vrijednosti koje se dobijaju odgovarajućim proračunom. Do potrebnih veličina za proračune se dolazi mjerenjem naponskih i strujnih signala i njihovom konverzijom u binarne vrijednosti više puta u nekom periodu vremena. Ove veličine su podaci za proračune nakon kojih se dobija rezultat – vrijednost koja se logički poredi sa izabranim podešenjem. Kao rezultat poređenja dobijaju se signali na izlazima koji mogu biti djelovanje prekidačima, upravljanje, startovanje nekih drugih procedura ili samo obavještenje operatera o događajima. Od velikog broja mikroprocesorskih uređaja kojima se može realizovati zaštita motora, u drugom poglavlju ovog rada predstavljen je uređaj „Sepam M61“, njegove mogućnosti u pogledu zaštite, mjerenja, upravljanja i praćenja relevantnih veličina. U glavi tri prikazane su zaštitne funkcije koje su od interesa za zaštitu od kvarova i opasnih stanja električnih motora.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je dr Duško Bekut, redovni profesor.

Razmotrene su i informacije koje su od interesa za nadređeni SCADA sistem, kao i benefiti ovakvog načina automatizacije relejne zaštite i sistema za zaštitu i upravljanje. U posljednjim dijelovima izneseni su zaključak i prednosti ovakvog vida zaštite elektromotora, kao i literature korišćena za izradu ovog rada.

2. MIKROPROCESORSKA ZAŠTITA

Mikroprocesorski releji jesu savremena vrsta zaštitnih releja koja se sve češće primjenjuje danas. Krajem šezdesetih godina su se počeli razvijati mikroprocesorski zaštitni sistemi. Kao što je i logično pretpostaviti jeste da je tada mikroelektronika bila veoma skupa pa je bilo veoma nepraktično raditi sa ovakvom zaštitom koja pri tom nije imala veliku pouzdanost. Kako je vrijeme prolazilo, uređaji elektronike bivali sve jeftiniji, a način izrade mikroprocesora obezbjeđivao sve veću pouzdanost, tako se sve više išlo ka tome da se njihova primjena u relejnoj zaštiti poveća. Danas se sve češće viđaju ovakvi uređaji, kojima se zamjenjuju stari (mehanički i elektromagnetni) releji. Stari releji se nastavljaju primjenjivati dokle god mogu da ispune tražene zahtjeve, odnosno, dokle god je to isplativo. S obzirom na borbu za tržištem, proizvođači se konstantno nadmeću tako što iznova dodaju nove mogućnosti ovih releja, proizvode softverske alate za lakšu upotrebu i sl. Na slici 2.1 prikazan je izgled tri mikroprocesorska releja različitih proizvođača [3].



Slika 2.1. Izgled današnjih mikroprocesorskih zaštita različitih proizvođača

Način na koji se bira zaštitni uređaj za zaštite određenih dijelova sistema određuju funkcije koje taj uređaj ima. Tako su razvijeni specijalizovani zaštitni sistem za zaštitu električnih motora, transformatora, vodova... Jedan od najzastupljenijih uređaja je „Sepam M61“ na čijem primeru će biti izvršena analiza zaštite elektromotora. Ogromne mogućnosti uređaja „SEPAM“ tipa M61, odgovarajućeg za zaštitu motora, predstavljene su u sledećim tabelama. U tabeli 2.1. data je lista zaštitnih funkcija kao i dodatnih funkcija za upravljanje i praćenje dok su u tabeli 2.2. prikazane mogućnosti mjerenja [3].

Tabela 2.1. Zaštitne funkcije uređaja „Sepam M61“ [4]

Zaštita	ANSI kod
Fazna prekostrujna	50/51
Zemljospojna/osjetljiva zemljospojna	50N/51N, 50G/51G
Zaštita od otkaza prekidača	50BF
Nesimetrično opterećenje	46
Termičko preopterećenje za mašine	49RMS
Usmjerena zemljospojna	67N/67NC
Usmjerena prekom.aktivne snage	32P
Usmjerena prekom.reaktivne snage	32Q
Fazna podstrujna	37
Predugačko vrijeme startovanja, ukočen rotor	48/51LR/14
Startovanje na sat	66
Gubitak pobude	40
Podnaponska	27
Podnaponska direktne komponente	27D
Prenaponska	59
Zemljospojna od rezid. Napona	59N
Prenaponska inverzne komponente	49
Nadfrekventna	81H
Podfrekventna	81L
Termostat/Buholc	26/63
Praćenje temperature	38/49T
Provjera sinhronizma	25
Upravljanje i praćenje	
Rasterećenje/automatski restart	
Lečovanje/potvrđivanje	86
Proglašenje	30
Prebacivanje grupa podešenja	

Tabela 2.2. Moguća mjerenja za uređaj „SEPAM M61“ [4]

Mjerenje	M61
Fazna struja I1, I2, I3 eff	+
Rezidualna struja I ₀ , zbirna I ₀	+
Struja potrošnje I1, I2, I3	+
Vršna struja potrošnje IM1, IM2, IM3	+
Napon U21, U32, U13, V1, V2, V3	+
Rezidualni napon V0	+
Direktna komponenta napona Vd	+
Inverzna komponenta napona Vi	+
Frekvencija	+
Aktivna snaga P, P1, P2, P3	+
Reaktivna snaga Q1, Q2, Q3	+
Prividna snaga S, S1, S2, S3	+
Vršna potrošnja PM, QM	+
Faktor snage	+
Temperatura (16 RTDs)	Opciono
Brzina obrtaja	Opciono

Objašnjenje funkcija mikroprocesorske zaštite nije moguće, a da se prije toga ne spomene obilježavanje po ANSI standardu. U elektroenergetskim sistemima svijet je naširoko prihvatio notaciju američkog ANSI obilježavanja (ANSI/IEEE Standard C37.2 *Standard for Electrical Power System Device Function Numbers, Acronyms, and Contact Designations*) koja služi za olakšanu identifikaciju zaštitnih uređaja kao što su prekidači, zaštitni releji i sl.

Pored zaštitnih funkcija, ovakav uređaj ima funkcije korisne za dijagnostiku sistema kao što su: brojač isključenja kvara faze i zemljospoja, poremećaji usljed viših harmonika (THD), fazni pomjeraj, prikaz korištenog toplotnog kapaciteta, brojač radnih sati/vremena, mjerenje

struja i vremena startovanja, odbrojavanje vremena sprečavanja startovanja, nadzor strujnih i naponskih transformatora, nadzor kola za isključenje, broj operacija, vrijeme rada, vrijeme navijanja opruge, broj operacija izvlačenja.

3. ZAŠTITA ELEKTROMOTORA

U prethodnom poglavlju predstavljen je uređaj koji je odgovarajući za zaštitu najzastupljenijih električnih mašina u elektroenergetskom sistemu; asinhronih električnih motora, kojima su posvećena razmatranja u ovom radu.

Tokom svog razvoja dimenzije i težine motora bitno su se smanjivale. Smanjene dimenzije smanjile su i toplotnu i električnu preopteretivost. Ipak, s druge strane, rasli su zahtjevi korisnika motora iskoristivosti motora (zalijetanje velikih masa, reverzibilni pogon...). Velike snage kratkih spojeva zahtijevaju što brže isključenje u slučaju pojave kvara da bi se posljedice svele na minimum a omogućio što brži i efikasniji oporavak. Zbog toga, uloga zaštite elektromotora je od velikog značaja.

Na slici 3.1. se može vidjeti asinhrona kavezna mašina specijalne izvedbe, koja se koristi u pumpama visokog pritiska. U pitanju je šestopolni asinhroni motor ukupne snage 1203 kW, na naponskom nivou od 13,8kV i 60Hz, IP 56 zaštite [2].



Slika 3.1. Visokonaponski asinhroni kavezni motor proizvođača WEG

Kao i kod ostalih električnih mašina, tako i kod motora razlikujemo stanja sa kvarom i opasna pogonska stanja. Te dvije stvari je potrebno razdvojiti iz prostog razloga što se pri uspostavljanju istih vrše različiti postupci za povratak pogona u rad.

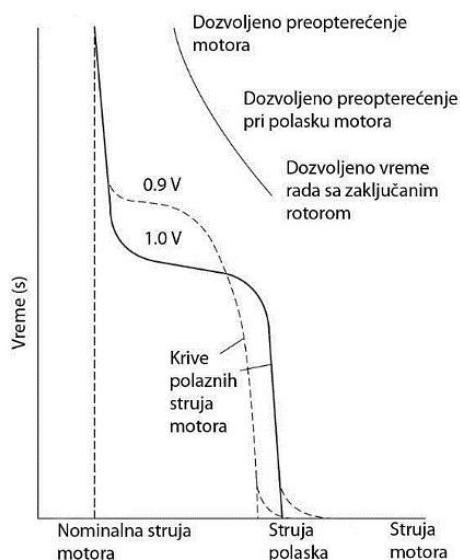
Potencijalni kvarovi asinhronih motora su [1]:

- 1) Kratak spoj između statorskih namotaja.
 - 2) Kratak spoj sa zemljom statorskog namota.
- Dok su opasna pogonska stanja [1]:
- 1) Preopterećenje motora.
 - 2) Zaključan rotor.
 - 3) Prenizak ili previsok napon napanja.
 - 4) Neodgovarajući redoslijed faza.
 - 5) Neodgovarajući ambijentalni uslovi (visoka koncentracija prašine, visoka ili niska temperatura).
 - 6) Gubitak faze napajanja.
 - 7) Mehanički kvarovi, kvarovi u ležajevima i sl.

Kvarovi na motorima često prouzrokuju mehaničku, kao i štetu na električnim dijelovima koji nisu pod kvarom. U vezi sa tim zaštita motora od preopterećenja se slobodno može svrstati u najvažniju zaštitu, te je potrebno obratiti posebnu pažnju da njena realizacija bude adekvatna. Pod tim pojmom se podrazumjeva njena odgovarajuća brzina reagovanja i vrijednost na kojoj će zaštita da prouzrokuje isključenje prekidača preko kog se napaja motor. Osobina koju posjeduju motori vezana za struju polaska koja traje

(nekada) i do jednog minuta, dodatno usložnjava podešavanje mikroprocesorske zaštite [5-7].

Na slici 3.2. je prikazana karakteristika struje polaska motora, zajedno sa karakteristikom zaštitnog releja i granice u kojoj je dozvoljen rad motora. Sa te slike se jasno može uočiti da kriva po kojoj radi relej mora da dozvoli direktan start (veliku polaznu struju motora), ali da isto tako ne smije da dozvoli da motor uđe u opseg zabranjenog rada (ne smije da dozvoli da određena struja preopterećenja ili kratkog spoja bude prisutna neko određeno vrijeme, koje zavisi od materijala izrade i gabarita dijelova od kojih je motor sačinjen).



Slika 3.2. Karakteristika struje polaska motora, zajedno sa karakteristikom zaštitnog releja i granice u kojoj je dozvoljen rad motora

3.1. Zaštita od preopterećenja

Prekostrujna zaštita, ili zaštita od preopterećenja ima funkciju zaštite od pregrijavanja izolacije namotaja električne mašine, odnosno od oštećenja njenih dijelova pri povišenoj temperaturi. Motori su konstruisani tako da njihove karakteristike dozvoljavaju preopterećenja; kako neki stručnjaci tvrde preopterećenja od 20% su dozvoljena i do dva sata, dok su preopterećenja od 50% dozvoljena nekoliko minuta.

Pomoću releja se realizuje zaštitna funkcija od preopterećenja koja koristi direktno snimanje struje štićenog uređaja pomoću senzora za mjerenje temperature (najčešće su to Pt100 sonde) i na osnovu poređenja tih vrijednosti sa podešenim dolazi do reagovanja releja. Velika pogodnost mikroprocesorske zaštite u odnosu na stare releje jeste u tome da sa jednom sondom za mjerenje temperature (i odgovarajućim podešenjima) vrši i signalizacija i isključenje. Kod starih releja za te dvije funkcije bilo potrebno kupiti dva odvojena uređaja, te se jedan uređaj povezivao u kolo za signalizaciju, a drugi u kolo za isključenje.

Takođe, današnje mikroprocesorske zaštite posjeduju neke funkcije kao što su broj startovanja motora (ANSI 66 - *Starts per hour*), gdje mi u zaštitu upisujemo vrijednosti vremena između zaustavljanja i mogućeg sledećeg startovanja mašine, broj uzastopnih startovanja iz hladnog stanja, broj uzastopnih startovanja iz toplog stanja, period hlađenja mašine itd.

Za motore koji se pokreću sa nominalnim naponom, čije vrijeme polaska je između 5s i 10s, C37.96-2000 standard predviđa da se napravi minimalna margina između vremena polaska i vremena prekidanja definisanog krivom za zaštitu, dok se za motore čije se vrijeme polaska kreće u opsegu od 40s do 50s predviđa margina od 5s.

3.2 Zaštita od kratkih spojeva između faznih namotaja

Zaštita od kratkih spojeva treba da bude sposobna da isključi kvarove kao što su kratak spoj između statorskih namotaja, statorskog namotaja sa zemljom, kratak spoj na kabl i sl. Kada se podešava ova zaštita potrebno je obratiti posebnu pažnju da je u pitanju motor i da zaštita ne bi smjela da odreaguje na polaznu struju koja se očekuje. Struja na koju je potrebno podesiti kratkospojnu zaštitu bi trebala da bude u opsegu između 1,65I_{pol} i 1,9I_{pol}.

3.3. Zaštita od kratkih spojeva fazih namotaja sa zemljom (zemljospoja)

Dva najčešće korišćena načina za detekciju struje zemljospoja kod mikroprocesorskih zaštita) jesu da se koriste tri mjerna transformatora, koji su postavljeni tako da pokazuju fazne struje kroz napojni kabal motora, ili korišćenjem jednog obuhvatnog transformatora. Način kada mjerimo tri struje (mjerene sa 3 mjerna transformatora) se dovode na relej, gde se na osnovu te tri očitane struje proračunava struju zemljospoja. Ta struja se zatim poredi sa strujom koja je podešena i ukoliko je ona veća od podešene, šalje se ili signal za isključenje prekidača ili signal za alarm. Ti signali se šalju putem digitalnih izlaza na releju. Drugi način jeste da se koristi obuhvatni transformator. Taj transformator na svom izlazu direktno daje redukovanu struju zemljospoja, u skladu sa svojim prenosnim odnosom. Tipična vrijednost na koju se podešava zemljospojna zaštita jeste na 20% od nominalne vrijednosti struje motora.

3.4. Zaštita od inverznih struja

Malo detaljnijom analizom dolazi se do zaključka da 5% inverzne komponente napona proizvodi oko 30% inverzne komponente struje, što predstavlja veoma visok procenat. Problem kod inverzne komponente struje nije samo u tome što je povećano zagrijavanje mašine, koje je takođe veoma bitan faktor. Ono što se može smatrati bitnijim faktorom jeste upravo to da se stvara negativna vrijednost momenta konverzije, a time i kočioni moment (kontramoment) mašine. U prilog ovome ide i sama činjenica da se motor neće pokrenuti ni u praznom hodu ukoliko mu je isključena jedna faza, pri čemu su direktna i inverzna struja jednake, odnosno rezultatni moment konverzije je jednak nuli. Zaključuje se da je ova zaštita veoma značajna kod asinhronih motora velikih snaga a mikroprocesorski uređaji omogućavaju efikasno podešavanje ove zaštite.

3.5. Podnaponska zaštita

Propad napona rezultuje povećanom strujom motora. Posljedice niskog napona se mogu ogledati u tome da se motor ne može pokrenuti (s obzirom na to da se smanjenjem napona znatno smanjuje moment motora). Ova zaštita se standardno podešava na 80% nominalnog napona, jer se za tu vrednost napona smatra da je moguće izvršiti pokretanje motora bez većih posledica. Ova karakteristika se obično podešava sa vremenskom zadržkom od nekoliko sekundi.

3.6. Realizacija funkcija zaštite

Za proračune parametara u stacionarnim stanjima kod asinhronih motora je moguće koristiti standardnu ekvivalentnu šemu motora koja služi za proračune karakterističnih veličina motora, na osnovu kojih mogu biti određeni parametri zaštite motora. Podešavanje tih proračunatih parametara za svaku od navedenih zaštita vrši se u odgovarajućem softveru u zavisno od korišćenog uređaja, prikazano kao primjer na slici 3.3. Softver „SFT2841“ služi za parametrisanje zaštitnih funkcija za uređaj tipa „Sepam“. Karakteristike ovakvog programa su lako lak rad i intuitivnost [5].



Slika 3.3. Parametrisanje zaštitnog releja

Još jedna od mogućnosti mikroprocesorske zaštite jeste prenos relevantnih informacija do SCADA (Supervisory Control And Data Acquisition) sistema koji služi za automatizaciju procesa; za prikupljanje podataka sa senzora i dislociranih mjernih stanica, za prenos i prikazivanje mjernih i prikupljenih podataka u centralnoj stanici u svrhu nadzora i/ili upravljanja, u realnom vremenu [6].

4. ZAKLJUČAK

Koncept zaštite u novijim elektroenergetskim sistemima danas nije moguće zamisliti bez prisustva mikroprocesorske zaštite koja je prevazišla sve ograničavajuće faktore starih tipova relejne zaštite. Posebno bitan aspekt jeste zaštita asinhronih motora, skupocijenih dijelova elektroenergetskih sistema čiji kvarovi za posljedicu mogu imati velike materijalne štete za popravak ili zamjenu oštećenog dijela. Štete prouzrokovane prestankom procesa proizvodnje usljed kvara mogu biti još i veće i zato se velika pažnja posvećuje analizi sistema i zaštititi visokonaponskih asinhronih motora. Primjenom mikroprocesora dobija se kvalitetnija zaštita i znatno širih mogućnosti u odnosu na klasičnu jer je omogućena primjena inteligentnijih funkcija i postupaka u okviru jedne zaštite. Omogućen je nadzor štićenog objekta i analiza podataka čak i u slučajevima kada nema kvara. U slučaju kvara dostupan je veliki broj informacija o samom kvaru kao i veliki broj informacija od interesa za daljnji nadzor i upravljanje.

5. LITERATURA

- [1] Duško Bekut, Relejna Zaštita – FTN Izdavaštvo, Novi Sad, 2009;
- [2] Franjo Božuta, Automatski zaštitni uređaji elektroenergetskih postrojenja, Zavod za udžbenike i nastavna sredstva Sarajevo, 1986.
- [3] Simon Richards Network protection & automation guide, Third edition, ISBN: 978-0-9568678-0-3, May 2011, Published by *Alstom Grid*,
- [4] Electrical network protection SEPAM 20,40,60,80, Schneider Electric
- [5] <https://docplayer.net/52232862-Proizvodni-programi-production-programme-pp17sren.html>
- [6] E. Veljković Grbić, J. Dragutinović, A. Car, Izrada aplikacije za konfigurisanje SCADA sistema u skladu sa savremenim konceptima modelovanja i konfigurisanja; Institut Mihailo Pupin, Beograd;
- [7] Reimert Donald, Protective relaying for power generation systems“ ISBN-13: 978-0-8247-0700-2 (alk. paper), 2006;

Kratka biografija:



Bilja Sjeran rođena je u Trebinju 1993. godine. Osnovne studije je završila na Fakultetu tehničkih nauka 2017. godine na departmanu Elektrotehnika i računarstvo – Elektroenergetski sistemi. Master rad je odbranila 2019. godine, na istom fakultetu na smjeru Energetska elektronika i električne mašine

REALIZACIJA EDUKACIONOG MIKORORAČUNARA NA ZYNQ KONFIGURABILNOM SISTEMU NA ČIPU

EDUCATIONAL MICROCOMPUTER IMPLEMENTATION ON ZYNQ RECONFIGURABLE SYSTEM ON CHIP

Balaž Petrović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu je opisan proces realizacije ZEdulent školskog mikrorračunara na ZYNQ sistemu na čipu. Prikazan je proces projektovanja mikrorračunara i grafičkog korisničkog interfejsa. Pored toga opisan je način verifikacije i testiranja.

Ključne reči: ZYNQ, ZYBO, FPGA, Edulent

Abstract – In this paper a process of ZEdulent educational microcomputer implementation on Zybo board is described. Microcomputer and graphical user interface design process is depicted. Besides, verification and testing methodology is given.

Keywords: : ZYNQ, ZYBO, FPGA, Edulent

1. UVOD

Jednostavnost opisivanja logičkih kola preko jezika za opis hardvera, kao što je VHDL, i dostupnost FPGA čipova su bili razlozi za projektovanje i realizaciju edukacionog 8-bitnog mikroprocesora (Edulent) [1] na kojem studenti mogu da izvršavaju sopstvene programe u assembleru. Pored procesora, takođe u okviru projekta dizajniran je grafički interfejs na PC računaru u Visual C++ jeziku koje omogućava studentima da prate rad mikroprocesora [1].

Takav sistem omogućava studentima da upišu program u memoriju mikroprocesora, zatim da izvršavaju program na mikroprocesoru baziranom na FPGA i da grafički prikazuju unutrašnje stanje procesora preko grafičkog interfejsa.

Koristeći Edulent, studenti su u mogućnosti da upoznaju osnovnu arhitekturu mikroprocesora i asembler jezik programiranja. Zbog toga Edulent simulator je korišćen godinama na laboratorijskim vežbama. Na osnovu pozitivnih povratnih informacija, ovaj simulator je olakšao korake studentima, kao što su: razumevanje funkcionalnosti mikroprocesora, lakše razumevanje dizajna mikroprocesora, programiranje u assembler jeziku, izvršenje programa i izvršavanje programa korak po korak.

Komunikacija između FPGA čipa i PC grafičkog interfejsa se odvijalo preko LPT paralelnog porta. Pošto na novijim računarima LPT port je zamenjen kompaktnijim USB portovima, postoji potreba za realizacijom Edulent mikrorračunara na novijim FPGA čipovima, koji se koristi za savremeniji vid komunikacije.

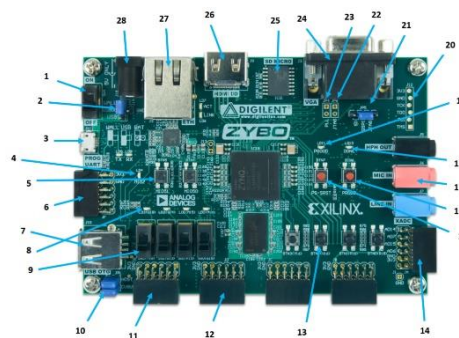
NAPOMENA:

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

Savremenija verzija Edulenta je realizovana na ZYNQ konfigurabilnom sistemu na čipu, pod nazivom ZEdulent. ZEdulent je edukacioni mikroprocesor projektovan da predstavlja evolutivni razvoj Edulent mikroprocesora [1]. Zahtevi koje moraju da zadovolje ZEdulent su isti kao i za Edulent: jednostavna arhitektura, grafički korisnički interfejs, sposoban za pokretanje jednostavnih programa korak-po-korak, assembler jezik za programiranje i jednostavno povezivanje sa periferijama.

1.1 Zybo razvojna ploča

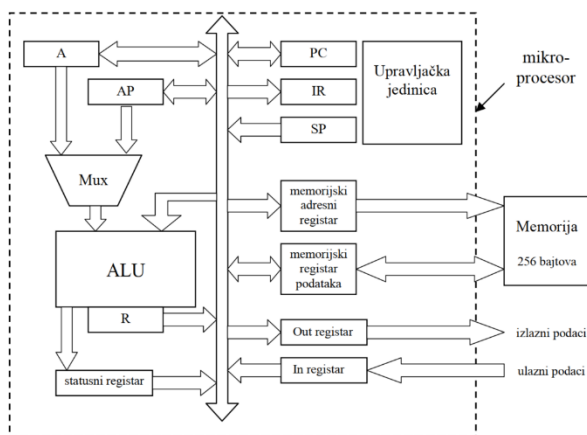
ZYBO (Zynq BOard) je fleksibilna, embedid razvojna platforma softvera i digitalnog dizajna, početnog nivoa. Razvojna platforma izgrađena oko najmanjeg člana porodice Xilinx Zynq-7000, Z-7010. Z-7010 je zasnovan na Xilinx All Programmable System-on-Chip (AP SoC) arhitekturi, koja integriše dual-core ARM Cortex-A9 procesor i programabilnu logiku Xilinx 7-series Field Programmable Gate Array (FPGA). Kada se SoC spoji sa bogatim skupom multimedijalnih i perifernih uređaja za povezivanje koji su dostupni na ZYBO-u, Zynq Z-7010 može da podržava dizajn čitavog sistema. Ugrađena memorija, video i audio ulaz/izlaz, USB, Ethernet i SD slot će omogućiti dizajn sistema bez dodatnog hardvera. Pored toga, na raspolaganju je šest Pmod portova za specijalne potrebe dizajna [2].



Slika 1. Zybo board

1.2 Model Edulent mikrorračunara

Uzimajući u obzir razmatranja iz uvodnog dela gde je naglašen značaj arhitekture za početno izučavanje iz oblasti mikroprocesora, odabrana je 8-bitna arhitektura. Postoji više različitih podela mikroprocesora prema arhitekturi. Odabrana je arhitektura koja predstavlja modifikovanu akumulatorsku arhitekturu von Neumann tipa tako da se linearan memorijski prostor koristi i za program i za podatke. U ovom poglavlju su prikazane osnovne karakteristike, načini adresiranja i skup instrukcija dati u [1].



Slika 2. Model mikroračunara

Mikroprocesor ima 11 registara i svi su 8-bitni. Dva su registra direktno programski dostupna korisniku. Dva registra služe kao pomoćni registri za realizaciju ulazno-izlazne komunikacije sa perifernim uređajima.

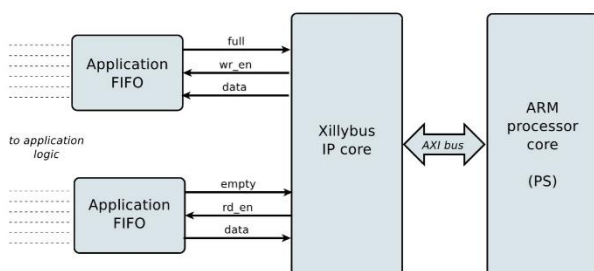
Edulent ima ukupno 40 instrukcija (uključujući različite načine adresiranja).

1.3 Xillybus

Xillybus je jednostavan, prenosiv, i efikasno rešenje za prenos podataka između FPGA i hosta na kojoj se izvršava operativni sistem (Linux ili Microsoft Windows). Dostupan je za personalne računare i za embedid sisteme koji koriste PCI Express bus za prenos podataka, također podržava ARM-bazirane procesore, koji koriste AMBA magistralu za prenos podataka (AXI3 / AXI4) [3].

FPGA dizajner kao i host programer komunicira sa Xillybus-om preko dobro poznatih interfejsa: FPGA programabilna logika se povezuje na IP core preko standardnih FIFO-a; korisnička aplikacija na host računaru obavlja obične I/O operacije na device fajlovima.

Tok podataka se kreće prirodno između FIFO-a i datoteke koju je otvorila host aplikacija. Ne postoji specifičan API, koji treba instalirati i koristiti.



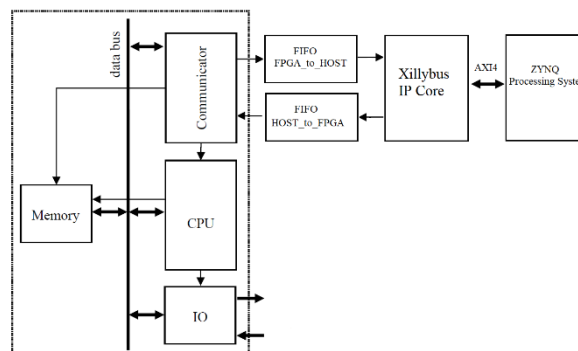
Slika 3. Xillybus ip core

Kao što je gore prikazano na Sl. 3, programabilna logika aplikacije na FPGA-u treba da bude u interakciji sa standardnim FIFO-ima.

Aplikacija na računaru čita i piše podatke u obične datoteke uređaja koje se ponašaju kao cevi. Xillybus IP jezgro i kernel drajveri prenose podatke efikasno i intuitivno između FIFO-a u FPGA i njihovih odgovarajućih device fajlova na hostu.

2. REALIZACIJA MIKRORAČUNARA NA ZYBO RAZVOJNOM SISTEMU

Blok šemu dizajna možemo videti na Sl. 4. Edulent IP jezgro je povezan preko FIFO bufera na Xillybus IP. Xillybus IP jezgro komunicira sa ARM procesorom preko AXI4 magistrale.



Slika 4. Blok šema sistema

Na blok šemi možemo videti modifikacije u odnosu na prethodne verzije Edulenta [1]. U ranijim verzijama komunikaciona jedinica, komunicirala je sa PC računalom preko LPT porta. U ovoj realizaciji komunikaciona jedinica komunicira sa host računalom preko FIFO bafera, zbog toga je potrebno projektovanje nove komunikacione jedinice.

Ostali delovi mikroprocesorskog sistema kao što su CPU, memorija i ulazna-izlazna jedinica ostaju suštinski nepromenjena. Potrebne su modifikacije u opisu dvosmerne magistrale. Vivado alat zahteva eksplicitno drajvovanje za dvosmernu magistralu.

2.1 Komunikaciona jedinica

Tabela 1. Rezultati implementacije

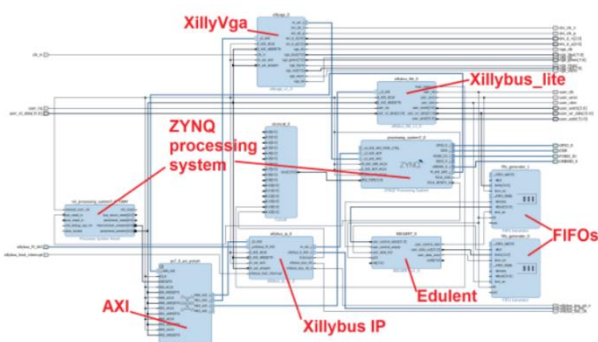
Slice blocks	5254 out of 17600 (30%)
Flip-flops	4986 out of 35200 (14%)
4 input LUTs	474 out of 4704 (10%)
I/O blocks	30 out of 146 (20%)
Tri state buffers	116 out of 2352 (4%)
RAM blocks	5 out of 60 (8%)
Clock frequency	20 MHz

Zbog edukacionog karaktera celog sistema i želje da se u svakom trenutku može imati uvid u kompletno stanje sistema, potrebno je osim mikroračunara realizovati i odgovarajući blok za komunikaciju sa personalnim računalom. Ovo je realizovano u komunikator bloku. Komunikator predstavlja vezu između mikroračunara i PC računara. Komunikacioni blok treba da omogući izvršavanje aktivnosti kao što su pokretanje programa, izvršavanje instrukcija korak po korak itd. U suštini komunikacioni blokima funkcionalnost jednostavne upravljačke jedinice, jer treba da obezbedi odgovarajuće kontrolne signale pri čitanju stanja registara i memorije ili pri upisu u njih. Pored toga ovaj blok mora da omogući i odgovarajuće zaustavljanje takta procesora ili na neki drugi način zaustavljanje rada procesora kada je to potrebno. Način komunikacije je takav da komunikacioni blok prima instrukcije od PC hosta po određenom

protokolu. Na osnovu instrukcije komunikator šalje vrednost registara, šalje vrednosti iz memorije, upiše novu vrednost u neki registar(ako je to dozvoljeno), upiše vrednost u memoriju na određenu adresu, i da pokrene procesor na jedan od tri moguća načina.

2.2 Realizacija na FPGA čipu

Na Sl. 5. možemo videti blok šemu sistema. FPGA deo sistema se sastoji od više modula. Postoje moduli koji su potrebni u svakom dizajnu. Ovoj grupi pripada: ZYNQ processing sistem, Processing System Reset, AXI Interconnect. Moduli koji su potrebni za prikazivanje na monitoru i za komunikaciju između PL i PS sistema su: xillyvga IP jezgro, xillybus IP jezgro, xillybus_lite IP jezgro, dva FIFO IP jezgro. Modul koji implementira funkcionalnost Edulent mikroračunara je Edulent IP jezgro.



Slika 5. Šema sistema IP jezgara [4]

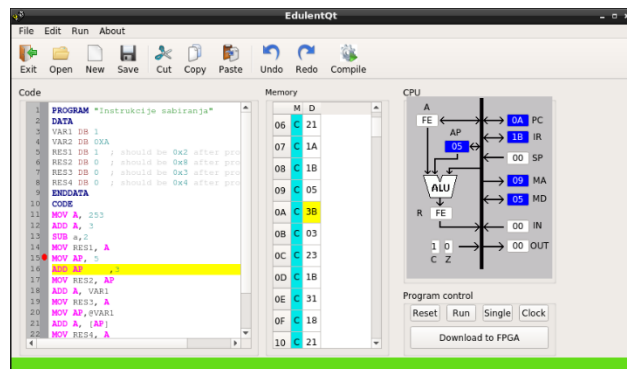
Sistem je implementiran pomoću Xilinx Vivado alata za dizajn koristeći VHDL jezik za opis hardvera za IP jezgra (Sl. 5.). Rezultati implementacije za XSC7Z010 SoC na ZYBO ploči su date u tabeli 1. Upoređivanjem rezultata implementacije sa rezultatima implementacije ranijih verzija Edulent mikroračunara [1] vidimo da broj Slice-blokova i flip-flop-ova su znatno veći. To je zbog činjenice da ZEdulent ima dodatnu logiku u odnosu na Edulent (npr. FIFO, Xillybus IP jezgro i XillyVga IP jezgro).

3. PROGRAMSKA PODRŠKA NA QT

Zbog edukacionog karaktera ZEdulenta, čiji je osnovni cilj da omogući studentima početna učenja iz oblasti mikrop procesora i mikroračunara, potrebna je vizualizacija stanja mikroračunara. U ovoj sekciji opisan je grafički korisnički interfejs.

Za projektovanje GUI korišćeno je razvojno okruženje Qt Creator. Ovaj program podržava metode potrebne za generisanje i korišćenje GUI elemenata. Qt koristi C++ jezik, s tim se program sastoji od više objekta.

Na Sl. 6. možemo videti Edulent aplikaciju u Run mod-u sa postavljenim Brake point-om. Brake point je postavljen na liniju 15 u CodeEditoru, mikrop procesor izvršio instrukciju na liniji 15 (MOV AP, 5), zatim je zaustavljen Brake point-om. Žuta linija u CodeEditor-u je postavljena na sledeću instrukciju, na liniju 16. U memorijskom prostoru žutom linijom označeno je mesto gde programski brojač pokazuje.



Slika 5 Edulent u Run modu

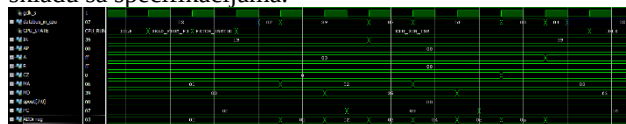
Brake point se postavlja duplim klikom na liniju teksta u CodeEditor-u, isto tako sa duplim klikom se briše. Brake point treba da bude postavljen pre kompajliranja.

4. VERIFIKACIJA

Verifikacija kao zadatak ima da potvrdi da logika dizajna radi usklađeno sa dizajniranom specifikacijom. Dakle, verifikacija pokušava da da odgovor na pitanje: "Da li ovaj predloženi dizajn radi ono što ste mislili?" Za ovaj složeni zadatak potrebno je dosta vremena i napora u većini projekata tokom projektovanja elektronskih sistema.

U okviru ovog rada je projektovana nova komunikaciona jedinica, samim tim je potrebna temeljna funkcionalna verifikacija ove jedinice. U ovom radu je izvršena funkcionalna verifikacija pomoću simulatora Xilinx Vivado alata i posmatranjem izlaznih signala.

Svaka instrukcija Edulent procesora je simulirana jedan po jedan sve dok se svi funkcionalno ne verifikuju. Svaka instrukcija komunikacione jedinice je simulirana, zatim posmatranjem izlaznih signala zaključeno je da radi u skladu sa specifikacijama.



Slika 6 Simulacija instrukcije ADD A, 0x0FD

Na Sl. 7. prikazani su signali mikrop procesora iz simulacije. Simulirana je u single modu. Komunikator modul je dražvovan porukom za pokretanje procesora za izvršavanje jedne instrukcije. Kada komunikator detektuje poruku u ulaznom FIFO baferu, prelazi u stanje *Read_from_fifo*. U ovom koraku učitava poruku u sopstvene registre. U narednom koraku je dekodovanje instrukcije *Fetch_instruction*. Nakon dekodovanja komunikator i/ili mikrop procesor u zavisnosti od instrukcije izvršavaju akcije vezane za instrukciju. U ovom slučaju to je izvršenje instrukcije ADD A, 0x0FD. Možemo videti da procesor radi ispravno.

Poslednja metoda u postupku verifikacije je provera rada na FPGA kolu. Ovaj korak spada delimično u verifikaciju, a delimično u upotrebu. Ova metoda predstavlja pogodanu metodu za proveru ispravnosti rada celog sistema. Međutim, u slučaju pojave greške ona ne ukazuje tačno na izvor greške već samo približno. Zato je često potrebno uraditi simulaciju da bi se precizno utvrdila

greška. Kao i kod prethodne metode, ovo ukazuje na činjenicu da sve metode verifikacije daju najbolje rezultate kada se koriste u kooperaciji.

U ovoj fazi je testirana Qt programska podrška i rad celog sistema. Da bi mogao da se testira rad celog sistema prvo je potrebno verifikovati ispravnost programskog prevodioca. Ovo je urađeno tako što su redom testirane sve instrukcije sa svim tipovima adresiranja i praćene su njihove vrednosti na mašinskom jeziku koje je prevodilac generisao.

Nakon verifikacije kompajlera moglo se pristupiti funkcionalnoj verifikaciji pomoću programa za vizualizaciju rada mikroracunara. Za te potrebe napisano je više programa u simboličkom mašinskom jeziku tako da se pokriju sve instrukcije i svi tipovi adresiranja. Ovi programi su izvršavani u tri režima rada. Prvi režim podrazumeva celokupno izvršavanje programa i poređenje rezultata sa očekivanim. Drugi režim rada podrazumeva izvršavanje programa po principu instrukcija po instrukcija. Ovaj režim omogućava proveru izvršavanja pojedinačnih instrukcija. U trećem režimu rada program se izvršava po principu takt po takt i on omogućava praćenje izvršavanja mikroinstrukcija. Na ovaj način detaljno je testiran i funkcionalno verifikovan ZEdulent sistem.

5. ZAKLJUČAK

U ovom radu je opisan proces realizacije novije verzije Edulent mikroracunara, pod nazivom ZEdulent. Na ZYNQ ZYBO razvojnoj ploči je instalirana osnovna verzija Linux Ubuntu operativnog sistema, pod nazivom Xillinux. Zajedno sa operativnom sistemom instalirani su i drajveri potrebni za korišćenje Xillybus IP jezgra, koje služe za realizaciju komunikacije između FPGA dela ploče i Linux operativnog sistema.

U okviru rada projektovana je nova komunikaciona jedinica za komunikaciju sa Xillybus IP jezgrom. Dizajn mikropcesora je prilagođen novoj jedinici. Funkcionalna verifikacija je urađena u simulatoru Vivado programskog paketa. Zatim je izvršena implementacija ZEdulent sistema zajedno sa potrebnim IP jezgrima za video izlaz. Programiranje FPGA dela ploče je rađeno zajedno sa podizanjem Xillinux operativnog sistema.

Razvoj programske podrške se odvijao na Windows PC računaru, korišćenjem Qt Creator programa. Razvijen je programski prevodilac, i uspostavljena je komunikacija sa drajverima Xillybus IP jezgra. Završna faza razvoja je urađena u Xillinux operativnom sistemu, na ZYBO platformi. Svrha programske podrške je aktivacija i praćenje različitih režima rada programa na mikroracunaru, kao i vizuelno prikazivanje dinamike izvršavanja programa.

Testiranje i funkcionalna verifikacija celog sistema je urađena pomoću nekoliko programa pisanim na simboličkom mašinskom jeziku koje pokrivaju sve instrukcije mikropcesora. Programi su izvršeni u sva tri načina rada mikropcesora.

Budući pravci razvoja su proširenje procesora sa nekim dodatnim funkcionalnostima (npr. rad sa prekidima, UART itd.). Osim toga moguće je projektovanje 16-bitne i/ili 32-bitne verzije Edulent mikropcesora.

6. LITERATURA

- [1] I. Mezei: "Formalna specifikacija i realizacija laboratorijskog mikroracunara na programabilnom integrisanom kolu" Magistarska teza, FTN, Novi Sad 2005
- [2] ZYBO Reference manual:Digilent
https://www.xilinx.com/support/documentation/university/XUP%20Boards/XUPZYBO/documentation/ZYBO_RM_B_V6.pdf accessed on: 29.03.2019
- [3] Xillybus IP datasheet: <http://xillybus.com/doc> accessed on: 22.03.2019
- [4] Balaz Petrovic, Ivan Mezei: „Educational microprocessor implementation on ZYNQ reconfigurable system on chip“, poslato za prikaz na međunarodnoj konferenciji

Kratka biografija:



Balaž Petrović rođen je u Senti 1993. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Embedid sistemi i algoritmi odbranio je 2019.god. kontakt: pbalazs@stcable.net

**ЈЕЗИК И ОКРУЖЕЊЕ ЗА МОДЕЛОВАЊЕ И ГЕНЕРИСАЊЕ ВЕБ СТРАНИЦА
ИНФОРМАЦИОНИХ СИСТЕМА ОПШТЕ НАМЈЕНЕ****LANGUAGE AND FRAMEWORK FOR MODELING AND GENERATING WEB PAGES
OF GENERAL PURPOSE INFORMATION SYSTEMS**Биљана Анђелић, *Факултет техничких наука, Нови Сад***Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО**

Кратак садржај – У овом раду представљени су језик за моделовање веб страница информационих система опште намјене и генератори прототипова веб страница. Посебна пажња усмјерена је на моделовање валидационих правила корисничких форми. За развој meta-modela коришћено је окружење EMF, а окружење Xtend за развој текстуалне синтаксе намјенског језика.

Abstract – In this work a domain-specific language for modeling web pages of general-purpose information systems is presented together with a generator of web page prototypes. The focus of this work is on modeling the user form validation. The EMF environment was used for the creation of the meta-model. The Xtend environment was used for developing the textual syntax of the domain-specific language.

Кључне речи: Информациони системи, намјенски језици, развој заснован на моделима, формална спецификација веб страница

1. УВОД

Глобализација која се увукла у све поре људског живота и дјеловања намеће многе захтјеве свјетској привреди и онима који желе бити дио ње. Потреба да производи и услуге у најкраћем року буду доступни подједнако у свим дјеловима свијета намеће висок ниво умрежености коју је немогуће замислити без употребе информационих система.

Информациони систем (енг. *Information System, IS*) као модел реалног система има неколико задатака: обухват, складиштење, пренос, презентовање и обраду података као и аутоматизацију управљачких функција. Обзиром да обухват података представља улазни корак за реализацију свих осталих задатака неопходно је обезбиједити једноставан обухват података, који омогућава кориснику унос искључиво валидних и добро форматираних података. Најчешће, обухват података у оквиру информационог система обавља се путем корисничких форми. У овом раду биће описан језик и одговарајући алати који омогућавају генерисање веб страница информационих система опште намене са посебном пажњом усмјереном ка корисничким формама које уједно пружају први вид валидације података.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији је ментор био др Владимир Димитријески, доцент.

Употребом традиционалних методологија развоја софтвера клијенти су били у могућности да виде функционалан систем тек на крају процеса развоја. То је могло да буде прекасно ако имплементирано рјешење није задовољавало захтјеве и очекивања. Примјеном agilних методологија развоја софтвера са брзим итерацијама, развојем прототипова као и честом комуникацијом са клијентима овај проблем би се могао ако не у потпуности уклонити онда бар значајно умањити.

Увођење моделом вођеног развоја софтвера (енг. *Model-Driven Software Engineering, MDSE*) и намјенских језика представља погодан оквир за успјешну реализацију комуникације између клијената и стручњака за информационе технологије (енг. *Information Technology, IT*). Коришћењем намјенских језика омогућава се клијентима да помоћу познатих концепата из домена учествују у креирању модела информационог система. У овом раду представљено је рјешење које би требало да превазиђе касну детекцију неусклађености захтјева и очекивања са имплементраним рјешењем. Из тог разлога настао је намјенски језик *fvalDSL* који омогућава развој веб страница информационих система опште намјене путем концепата блиских пројектантима *IS* и концепата графичког корисничког интерфејса.

2. ПРЕГЛЕД ПОСТОЈЕЋЕГ СТАЊА У ОБЛАСТИ

Постојање информационих система датира још од времена прије појаве рачунара. Појава рачунара олакшала је развој и одржавање информационих система, али је наметнула и нове захтјеве у смислу брзине обраде, капацитета, повјерљивости и приказа података.

2.1 Традиционални приступи

Велики број традиционалних методологија развоја *IS* засноване су на моделима. У оквиру ових приступа модели се посматрају као важан дио процеса и предлага се употреба различитих врста модела да би се представиле различите фазе развоја *IS*-а [1]. Једна од битнијих фаза у развоју *IS* је концептуално моделовање. Очекивани резултат ове фазе је креирање модела независних од платформе (енг. *Platform Independent Model, PIM*) на којој се систем извршава. Модели креирани употребом традиционалне методологије развоја софтвера углавном служе као документација. За разлику од традиционалних приступа, рјешење описано у овом раду омогућава да се креирани модел користи као централни артефакт за генерисање функционалног прототипа апликације.

2.2 MDE приступ

Увођење развоја софтвера вођеног моделима (енг. *Model Driven Engineering, MDE*) омогућава превазилажење недостатака који су се испојили у традиционалном приступу развоја софтвера. Креирани модели постају извршиви артефакти што је искоришћено и у овом раду.

2.3 Примјена намјенских језика

Већина алата за спецификацију корисничког интерфејса представљају графичке алате који су намјењени дизајнерима. Они омогућавају корисницима да на брз и једноставан начин специфицирају графички кориснички интерфејс. Постоје и намјенски језици за спецификацију *IS* који омогућавају опис и серверске и клијентске стране као што је намјенски језик за моделовање информационих система опште намјене *WISL* (енг. *Web Information System Language*) [2].

За разлику од намјенског језика *WISL*, намјенски језик *fvalDSL* посједује знатно шири скуп правила која се могу примјенити на атрибуте ентитета. Осим правила која важе над атрибутима, *fvalDSL* посједује и међуатрибутска ограничења. Када се *fvalDSL* упореди са дизајнерским алатима за генерисање графичког корисничког интерфејса може се примјетити да има знатно скромније могућности при тачном позиционирању компоненти као и подешавању њихових димензија. Неке од предности намјенског језика *fvalDSL* у односу на дизајнерске алате јесте и то што је конкретна синтакса реализована као текстуална што га чини једноставног за коришћење *IT* стручњацима. Постојање предефинисаних компоненти графичког корисничког интерфејса знатно скраћује вријеме развоја јер није нужно за сваки атрибут навести компоненту којом се представља.

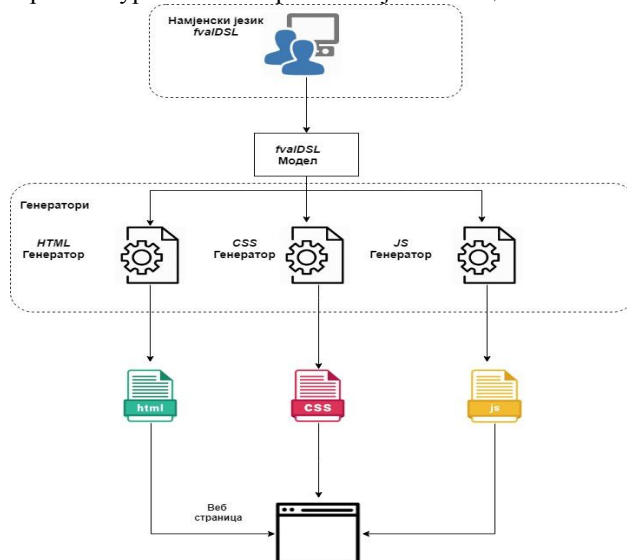
2.4. Интеракција човјек-рачунар

Седмдесетих година двадесетог вијека, појављује се парадигма са називом: „прозори, иконице, мени, показивачи“ (енг. *Window, Icon, Menu, Pointer, WIMP*) која пружа сасвим нови начин интеракције између човјека и рачунара. Овај вид интеракције корисницима је донио многе погодности као што су мулти-таскинг, једноставније коришћење за већину корисника, нема више памћења команди и контекстна помоћ. Поред погодности које је *WIMP* парадигма донијела постоје и недостаци од којих се посебно истиче смањење брзине интеракције напредних корисника.

Да би се превазишао наведени недостатак приликом попуњавања корисничких форми потребно је водити рачуна како су поља за унос груписана и распоређена у зависности од врсте података који се преко њих уносе, времена и мјеста када се приказују поруке о грешци и другим параметрима који могу имати утицаја. Приликом креирања генератора посебна пажња усмјерена је на избор предефинисаних компоненти. Предефинисане компоненте за атрибуте се креирају на основу типа атрибута и осталих ограничења тако да буду у складу са тренутно најбољом праксом из области човјек-рачунар. Међутим, пружена је и могућност да *IT* стручњак изабере произвољну компоненту којом ће представити атрибут уколико му предефинисана компонента из неког разлога не одговара.

3. АРХИТЕКТУРА СИСТЕМА

Архитектура система приказана је на слици 3.1.



Слика 3.1 - Архитектура система

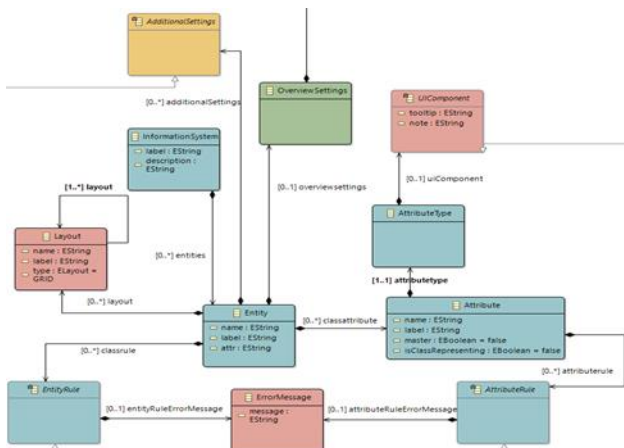
Систем се састоји од два модула: модул са намјенским језиком *fvalDSL* и модул са генераторима кода. Модули су на слици представљени правоугаоницима испрекиданих линија.

Први модул обухвата намјенски језик *fvalDSL* који посједује концепте блиске домену за креирање информационих система и нотацију која омогућава да се концепти инстанцирају. Излаз из првог модула је *fvalDSL* модел који уједно представља и улаз у други модул.

Други модул чине генератори кода. Описани систем посједује три генератора кода. Сва три генератора имају исти улаз, *fvalDSL* модел, а сваки од њих креира по један фајл као излаз. Генератор *HTML*-а (енг. *Hypertext Markup Language*) као излаз даје *HTML* фајл са статичким садржајем. Генератор *JavaScript*-а генерише *JavaScript* фајл са описом динамике, а генератор *CSS*-а (енг. *Cascading Style Sheets*) као резултат даје *CSS* фајл са описом стила.

4. НАМЈЕНСКИ ЈЕЗИК ЗА МОДЕЛОВАЊЕ ВЕБ СТРАНИЦА ИНФОРМАЦИОНИХ СИСТЕМА ОПШТЕ НАМЈЕНЕ

За развој апстрактне синтаксе коришћен је мета-мета-модел *Ecore* који омогућава представљање концепата из домена и веза које владају међу њима. На слици 4.1 представљен је најбитнији дио мета-модела намјенског језика *fvalDSL*. Централни концепт је *Entity*. Сваки ентитет посједује назив, а може да посједује лабелу и опис. Поред наведених обиљежја ентитет посједује атрибуте, опис изгледа корисничке форме, изглед табеле за приказ додатих записа и скуп међуатрибутских ограничења. У мастер раду из кога је овај рад произашао може се видјети комплетан мета-модел са конкретизацијама концепата *EntityRule*, *AttributeRule*, *uiComponent* и *AdditionalSetting* као и концептима које референцира концепт *OverviewSettings*.



Слика 4.1 Централни дио мета-модела намјенског језика *fvalDSL*

Поред ограничења која су јасно видљива на основу мета-модела као што су ограничења домена за вриједности атрибута и садржавања за описани језик постоје ограничења која није могуће специфицирати на нивоу мета-модела, а неопходна су да би се задовољила семантика. Обзиром да Layout може да садржи друге Layout-е, на једном нивоу сви Layout-и морају да буду истог типа што се не може видјети из мета-модела креирано је OCL (енг. Object Constraint Language)[3] правило приказано на листину 4.1.

Invariant

```
allLayoutsOnOneLevelMustBeSameType ('All
layouts on one level must be same type.'):
layouts -> forAll(fL1: Layout, fL2: Layout |
  if fL1.ocIsTypeOf(fL2.ocIsType())
  then true
  else false endif);
```

Листинг 4.1 Сви layout-и на једном нивоу морају бити истог типа

Конкретна синтакса за намјенски језик *fvalDSL* имплементирана је као текстуална. На листингу 4.2. приказан је дио информационог система који омогућава регистрацију корисника. Модел почиње називом система што је у приказаном случају *IS*, потом слиједи навођење лателе и описа који су опциони. Први ентитет уједно и једини у описаном примјеру је *user*. Ентитет *user* посједује три атрибута. Атрибут *username* је атрибут који представља корисничко име. Корисничко име је знаковни тип податка, обавезно је унијети његову вриједност и први карактер мора да буде велико слово, након тога слиједе мала слова. Минимална дужина корисничког имена је два карактера, а максимална 40 карактера. У случају да ограничења нису задовољена исписиваће се подразумеване поруке које постоје за свако ограничење. Атрибут ће бити представљен компонентом *TextInput*. Атрибути *password* и *confirmPassword* омогућавају унос и потврду лозинке. За оба атрибута важе иста ограничења. Лозинка мора да задовољи прописани формат што у овом примјеру значи да мора да има најмање једно мало слово, једно велико слово и једну цифру. Минималан број карактера које лозинка мора да има је 7, а максималан 40. Порука која се исписује уколико формат није задовољен специфицирана је по-

моћу атрибута *Error Message* у оквиру ограничења *StringPattern*. Атрибути *password* и *confirmPassword* су представљени компонентом *Password*. Омогућен је приказ карактера у изворном облику подешавањем параметра *Visible* на вриједност *true*. Да компонента графичког корисничког интерфејса није експлицитно наведена подразумјевало би се да се користи компонента *TextInput*. Након навођења атрибута наведена су и међу-атрибутска ограничења. Прво је наведено ограничење *Equals* које специфицира да вриједности атрибута *password* и *confirmPassword* морају бити једнаке. Друго специфицирано правило *NotEquals* дефинише да *password* и *username* не смију имати исту вриједност. Обзиром да распоред компоненти на форми није наведен подразумјева се да се компоненте приказују једна испод друге.

```
IS
{
  label "Information system"
  description "Short description"
  user{
    label "User"
    username{
      Data type String
      Required
      StringPattern '[A-Za-z0-9]+'
      MaxLength 40
      MinLength 2
    }
    password{
      Data type String
      Password (Visible true)
      Required
      StringPattern '(?=.*\\d)(?=.*[a-z])(?=.*[A-Z])'
      {
        Error message "Password must contain at least one upper case letter,
        lower case latter and digit."
      }
      MaxLength 40
      MinLength 7
    }
    confirmPassword{
      Data type String
      Password (Visible true)
      Required
      StringPattern '(?=.*\\d)(?=.*[a-z])(?=.*[A-Z])'
      {
        Error message "Password must contain at least one upper case letter,
        lower case latter and digit. "
      }
      MaxLength 40
      MinLength 7
    }
    Equals{
      attributes(password, confirmPassword)
      Error message ("Passwords must match.")
    }
    NotEquals {attributes(username,password)}
  }
}
```

Листинг 4.2 - Модел дијела информационог система за регистрацију

5. ГЕНЕРАТОР КОДА

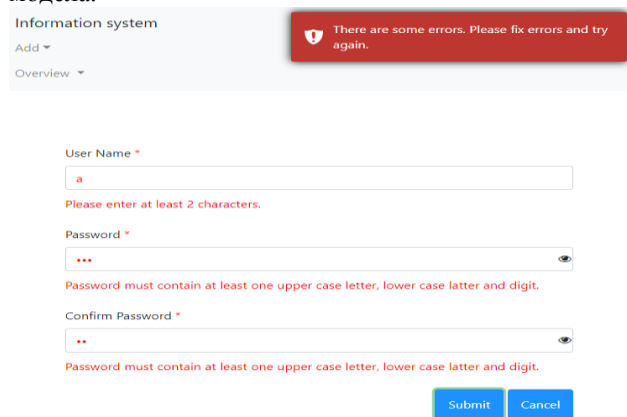
Као што је наведено у поглављу 3, постоје три генератора која имају исти улаз, али дају три излаза који могу да постоје независно. Уз одређене модификације могу се искористити и у другим рјешењима ако се испоштује начин на који се именују класе и *ID*-ијеви *HTML* елемената

У случају *HTML* генератора, приликом парсирања садржаја уčitаног модела, генератор проналази елементе који представљају компоненте графичког корисничког интерфејса и трансформише их у елементе *HTML*-а. Међутим, постоји могућност да за сваки концепт који треба бити приказан није специфицирана компонента којом ће бити представљена. Тада се приступа додатној провјери атрибута елемента како би се представио компонентом која ће корисницима омогућити најинтуитивнију и најбржу

интеракцију. Прво се врши провјера типа податка, а затим се приступа провјери и осталих ограничења која су специфицирана над елементом.

У случају имплементираниог CSS генератора за намјенски језик *fvalDSL* стил је дефинисан у екстерном фајлу, што чини јаснији и читљивији и CSS и HTML код. Прво се дефинише подразумевани стил за елементе као што су форма, панели и табеле за преглед унешених торки, а затим се дефинише специфичан стил за елементе уколико је он специфициран на нивоу модела. Уколико није специфициран или уколико за неки специфичан елемент није могуће специфицирати стил помоћу *fvalDSL*, могуће га је ручно додати или измијенити, навођењем селектора елемента за који се дефинише стил.

У имплементираниом JavaScript генератору прво се генеришу колекције које садрже мета податке о ентитетима који постоје у информационом систему, затим се врши генерисање помоћних функција које су исте за све ентитет. На крају се врши генерисање валидатора за сваки ентитет посебно на основу описаног модела.



Слика 5.1 - Део информационог система за регистрацију корисника

На слици 5.1 је приказан изгенерисани дио информационог система специфициран на листингу 4.1. Комплекснији примјер апликације је описан у мастер раду из кога је овај рад проистекао.

6. ЗАКЉУЧАК

У овом раду описан је развој веб страница за информационе системе опште намјене путем намјенског језика *fvalDSL*. За разлику од осталих алата који омогућавају развој крокија за графички кориснички интерфејс, који су углавном креирани за дизајнере, посједује текстуалну синтаксу која га чини ближим програмерима који су се навикли да програмирају у текстуалним програмским језицима. Приликом развоја модела путем намјенског језика *fvalDSL* није нужно за сваки атрибут навести компоненту графичког корисничког интерфејса. За сваки атрибут постоји предефинисана компонента којом ће бити представљен у складу са типом податка и скупом ограничења која важе над наведеним атрибутом, као и добрим праксама развоја дефинисаним у истраживачкој области интеракције човјек-рачунар.

Поред концепата који представљају компоненте графичког корисничког интерфејса посједује и концепте који су блиски *ER* моделу као што су ентитети, атрибути и везе међу ентитетима. Намјенски језик *fvalDSL* такође посједује и концепте који се односе на валидацију података како на нивоу атрибута, што је врло блиско ограничењу домена, тако и концепте за валидацију међу-атрибутских ограничења која су врло блиска ограничењима појаве типа ентитета из *ER* модела података. Наведени концепти омогућавају развој прототипа који може да врши симулацију и серверске стране што корисницима даје јаснији увид у понашање информационог система која си се развија.

Да би се помоћу *fvalDSL* могао описати цјелокупан информациони систем прво би се требало усмјерити на проширење мета-модела и увођењем концепата који би пружили подршку за ауторизацију, јер у тренутној имплементацији корисник има могућност да додаје, мијења и брише све ентитете који егзистирају у информационом систему. Потребно је развити и генераторе за серверски дио апликације.

Пожељно би било развити и генераторе за десктоп верзију апликације пошто није ријеткост да постоји потреба да један информациони систем буде развијен у више технологија. Ово би смањило могућност појаве разлике између веб и десктоп верзије информационог система.

7. ЛИТЕРАТУРА

- [1] Marco Branbilla, Jordi Cabot, Manuel Wimmer, Model-Driven Software Engineering in Practice Morgan & Claypool Publishers 2012.
- [2] Војислав Ђукић, Језик и окружење за моделовање и генерисање информационих система опште намене, Факултет техничких наука, 2015.
- [3] Object Constraint Language, <https://www.omg.org/spec/OCL/About-OCL/>, Приступљено 15.2.2019.

Кратка биографија:



Биљана Анђелић рођена је 1994. године у Бања Луци, Босна и Херцеговина. Факултет техничких наука уписала је 2013 године. Бечелор рад из области Електротехника и рачунарство – Рачунарске науке и информатика одбранила је 2017. год. Мастер рад одбранила је 2019. године.

IZBOR OPTIMALNIH POZICIJA I TIPA DISTRIBUTIVNIH GENERATORA I KONDENZATORSKIH BATERIJA U DISTRIBUTIVNOJ MREŽI PRIMENOM MOPSO

OPTIMAL PLACEMENT AND SELECTION OF DISTRIBUTION GENERATORS AND CAPACITOR BANKS IN DISTRIBUTION NETWORK WITH MOPSO

Balaž Kereši, *Fakultet tehničkih nauka, Novi Sad*

Oblast – Elektroenergetika – elektroenergetski sistemi

Kratak sadržaj – U ovom radu prikazan je postupak za izbor optimalne pozicije i optimalnih karakteristika distributivnih generatora i kondenzatorskih baterija u radialnoj distributivnoj mreži. Postupak se zasniva na Multi-objective Particle Swarm Optimization (MOPSO) algoritmu. Verifikacija njegovih mogućnosti realizovana je na primeru mreže IEEE-33.

Ključne reči: Distributivni generatori, Kondenzatorske baterije, Particle Swarm Optimization

Abstract – This master thesis describes the problem of optimal siting and sizing of Distribution Generators and Capacitor Banks on radial distribution network. This paper also discusses the usage of Multi-Objective Particle Swarm Optimization (MOPSO) algorithm to solve the proposed problem. Additionally, one simulation shows the problem solution on IEEE-33 standard system.

Keywords: Distribution Generators, Capacitor Banks, Particle Swarm Optimization

1. UVOD

Održavanje i razvoj elektroenergetskih sistema (EES) postaje sve kompleksniji zadatak čija realizacija ima sve veće troškove.

Pojavom distributivnih generatora (DG) klasične pasivne distributivne mreže (DM) postaju aktivne, tako da osim potrošnje mogu i da proizvode električnu energiju [1]. Postavljanjem kondenzatorskih baterija (KB) ostvarene su dodatne mogućnosti u regulaciji reaktivnih snaga i napona u DM [2].

Izbor optimalnih pozicija i optimalnih tipova DG (OPSDG) odnosno KB (OPSKB) predstavlja složen problem optimizacije (PO) koji se rešava primenom mešovito-celobrojnog nelinearnog programiranja [3]. Pored analitičkih i numeričkih metoda, pri rešavanju ovih problema veoma su efikasni evolucionari algoritmi i tehnike veštačkih inteligencija [1]. Zbog raznovrsnih i često kontradiktornih kriterijuma predstavljen PO može da se razmatra kao problem više kriterijumskih funkcija. U toj oblasti Multi-Objective Particle Swarm Optimization (MOPSO) predstavlja jedan od najčešće korišćenih algoritama [4] i kao takav koristi se i u ovom radu.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor dr Goran Švenda, red.prof.

Cilj ovog rada je da predstavi probleme OPSDG i OPSKB i da na osnovu MOPSO algoritma formira postupak za njihovo rešavanje. Verifikacija njegovih mogućnosti realizovana je na primeru mreže IEEE-33 [5].

Nakon Uvoda, u drugom delu rada izvršena je postavka problema OPSDG i OPSKB. U trećem delu prikazani su modeli elemenata DM i osnovni detalji MOPSO algoritma. U četvrtom delu su izvršeni eksperimenti sa različitim postavkama PO i dati rezultati pokazatelja efikasnosti ponuđenog rešenja. Nakon Zaključka koji je dat u petom delu u šestom delu je navedena korišćena literatura.

2. POSTAVKA PROBLEMA

OPSDG i OPSKB opisani su kao višekriterijumski problemi optimizacije (VPO). Nad skupom postignutih rešenja je kreiran Pareto front [5] koji sadrži sva nedominantna rešenja postignuta algoritmom.

2.1. Višekriterijumski problem optimizacije

Zadatak VPO jeste izbor vrednosti ulaznih promenljivih razmatranog matematičkog modela, sa ciljem poboljšanja vrednosti više kriterijumskih funkcija, uz uvažavanje aktivnih ograničenja jednakosti i/ili nejednakosti [6].

Matematički opis VPO glasi [6]:

$$\begin{aligned} \text{Minimizirati: } & f_l(\mathbf{u}) & l=1,2,\dots,M_{obj} \\ \text{uz ograničenja: } & g_i(\mathbf{u}) \leq 0 & i=1,2,\dots,m \\ & h_j(\mathbf{u}) = 0 & j=1,2,\dots,n \end{aligned} \quad (1)$$

gde je:

- $\mathbf{u}$ – vektor promenljivih, $\mathbf{u} = [u_1, u_2, \dots, u_{N_{prom}}]$, $\mathbf{u} \in U$, U je prostor odluke, a N_{prom} broj promenljivih;
- f_l – l -ta kriterijumska funkcija, $l=1,2, \dots, M_{obj}$, M_{obj} je broj elemenat kriterijumskog prostora;
- g_i – i -ta ograničenja nejednakosti, $i=1,2, \dots, m$, m je broj ograničenja nejednakosti;
- h_j – j -ta ograničenja jednakosti, $j=1,2, \dots, n$, n je broj ograničenja jednakosti.

2.2. Ulazni parametri modela optimizacije

Razmatrani PO ima sledeće ulazne promenljive:

- indeks čvora svih postavljenih uređaja;
- injektirane aktivne i reaktivne snage upravljivih DG;
- nominalne snage ne-dispečabilnih DG i KB;
- pozicija regulacione sklopke napojnog regulacionog transformatora (RTR).

3. MATEMATIČKI MODELI

Radikalna DM je modelovana pomoću čvor-grana modela. U skladu sa radikalnom strukturom DM za proračune tokova snaga koristi se Shirmohammadi-jev algoritam [7].

3.1. Modeli DG-a i KB-a

U ovom delu prikazani su model simuliranih DG i KB.

3.1.1. Modelovanje dispečabilnih generatora

U cilju kontrole, granice injektiranja aktivne i reaktivne snage DG su u opsegu tehničkih granica elektrane.

3.1.2. Modelovanje kondenzatorskih baterija

KB su modelovane kao dodatni otočni element u čvoru u kojem su postavljene [5]:

$$B = \frac{Q_n}{V_n^2}, \quad (2)$$

gde je:

B – supseptansa KB;

Q_n – nominalna reaktivna snaga KB, za $V = V_n$;

V_n – nominalni napon KB;

3.1.3. Modelovanje vetrogeneratora

Proizvodnja vetrogeneratora uglavnom zavisi od brzine vetra na lokaciji gde je vetroturbina postavljena [5]:

$$P_{DG,wind,p} = \begin{cases} 0, & v_p < v_{ci} \\ \alpha \cdot v_p^3 + \beta \cdot P_n, & v_{ci} \leq v_p < v_n \\ P_n, & v_n \leq v_p < v_{co} \\ 0, & v_{co} < v_p \end{cases}, \quad (3)$$

$$\alpha = \frac{P_n}{v_n^3 - v_{ci}^3}, \quad (4)$$

$$\beta = \frac{v_{ci}^3}{v_n^3 - v_{ci}^3}, \quad (5)$$

gde je:

$P_{DG,wind,p}$ – proizvodnja vetroturbine pri brzini vetra v_p ;

v_p – brzina vetra u stanju p ;

P_n – nominalna snaga vetrogeneratora;

v_{ci} – brzina uključenja vetroturbine (cut-in speed);

v_n – nominalna brzina vetroturbine;

v_{co} – brzina isključenja vetroturbine;

Za prognozu brzine vetra korišćena je Weibull-ova raspodela [5].

3.1.4. Modelovanje solarnih generatora

Proizvodnja solarnog panela u stanju p [5]:

$$P_{DG,solar,p} = N \cdot FF \cdot I_p \cdot V_p, \quad (6)$$

$$FF = \frac{V_{mpp} \cdot I_{mpp}}{V_{oc} \cdot I_{SC}}, \quad (7)$$

$$I_p = s_{avg,p} (I_{SC} + K_i (T_{C,p} - 25)), \quad (8)$$

$$V_p = V_{OC} - K_V \cdot T_{C,p}, \quad (9)$$

$$T_{C,p} = T_A + s_{avg,p} \left(\frac{N_{OT} - 20}{0.8} \right) \quad (10)$$

gde je:

$P_{DG,solar,p}$ – proizvodnja solarnog panela u stanju p ;

$T_{C,p}$ – nominalna temperatura ćelije u stanju p [°C];

T_A – nominalna temperatura ambijenta [°C];

$s_{avg,p}$ – srednja solarna radijacija u stanju p ;

N_{OT} – nominalna operaciona temperatura solarnih ćelija;

I_p – struja solarnih panela u stanju p ;

V_p – napon solarnih panela u stanju p ;

I_{SC} – struja kratkog spoja solarnih panela;

V_{OC} – napon praznog hoda solarnih panela;

K_i – temperaturni koeficijent struje [A/°C];

K_p – temperaturni koeficijent napona [V/°C];

N – broj solarnih panela;

FF – fill faktor solarnih panela.

U modelu vrednosti solarne iradijacije su prognoziranje pomoću Beta raspodele [5].

3.2. Kriterijumske funkcije

Ovom paragrafu su kvantifikovane kriterijumske funkcije na osnovu pojedinih pokazatelja mreže.

3.2.1. Aktivni gubici mreže

Gubici DM:

$$P_{Loss} = P_0 - \sum_{i=1}^{N_b} P_i \quad (11)$$

gde je:

P_{Loss} – aktivni gubici mreže;

P_0 – protok aktivne snage kroz koren mreže;

P_i – injektirana aktivna snaga potrošnje u čvoru i ;

N_b – broj čvorova.

3.2.2. Kriterijum indeksa naponske stabilnosti

Indeks naponske stabilnosti (VSI) predstavlja pokazatelj stabilnosti sistema na osnovu napona mreže i opterećenja. VSI je izračunat na osnovu sledeće formule [5]:

$$VSI_j = |V_i|^4 - 4.0 \cdot (P_j \cdot X_j - Q_j \cdot R_j)^2 - 4.0 \cdot (P_j \cdot R_j + Q_j \cdot X_j)^2 |V_i|^2 \quad (12)$$

gde je:

VSI_j – indeks naponske stabilnosti j -te grane;

X_j – reaktansa j -te grane;

R_j – aktivna otpornost j -te grane;

P_j – injektirana aktivna potrošnja u j -tom čvoru;

Q_j – injektirana reaktivna potrošnja u j -tom čvoru;

V_i – napon početnog čvora j -te grane.

Na nivoa mreže ova kriterijumska funkcija se izračunava na osnovu sledećeg izraza:

$$f_2 = \frac{1}{\sum_{i=2}^{N_b} VSI_i^t} \quad (13)$$

3.2.3. Kriterijum naponskog odstupanja

Kriterijumska funkcija odstupanja napona definisana je izrazom [5]:

$$f_3 = \sum_{i=1}^{N_b} |1 - \text{real}(V_i^t)|. \quad (14)$$

3.3. Ograničenja

PO ima sledeća ograničenja:

1. bilansa snage;
2. pozicija za postavljanje DG i/ili KB;
3. vrednosti modula napona svih čvora mreže;
4. injektirane aktivne i reaktivne snage postavljenih jedinica.

3.4. Mult-objective Particle Swarm Optimization

Osnovni entiteti MOPSO algoritma su čestice [5]. Jednu česticu karakterišu njeni atributi. U slučaju postavljenog problema atributi su pozicije DG i KB, odnosno njihove injektirane aktivne i reaktivne snage.

U prvoj iteraciji svaka čestica dobija slučajne attribute i izračunati su postignute vrednosti kriterijumskih funkcija. Njihovi globalno najbolji, Pareto dominantni rezultati su sačuvani u skladištu nedominantnih rešenja (REP). Algoritam dodatno pamti lično najbolje rezultate svih čestica.

U narednim iteracijama svaka čestica može da menja svoje attribute u smeru koji je određen na osnovu poznatih lično i globalno najboljih rešenja.

Zbog velike brzine konvergencije proračuna mutacioni operator takođe može da bude korišćen u kasnijim iteracijama. Prilikom mutacije atributi čestica ponovo dobiju slučajne vrednosti i zanemareni su lično i globalno najbolji rezultati.

3.5. Skladište nedominantnih rešenja (REP)

Globalno najbolji postignuti rezultati su sačuvani u REP. Prilikom procenjivanja novog rešenja posebna procedura proverice dominantnost novog rešenja nad ovim skupom. Ako ne postoji nijedno rešenje koje bi dominiralo nad novim, ono je dodato u REP-u. Ukoliko novo rešenje dominira nad drugima, oni nad kojim dominira će biti odbačeni.

REP ima ograničen broj elemenata. Ako prilikom dodavanja novog rešenja u skupu ova ograničenja budu preoračena, potrebno je odbaciti neki element od REP. Za izbor odbačenog elementa tehnika adaptivnih rešetaka je korišćena za podržavanje ravnomerne raspodele Pareto fronta.

Prilikom korišćenja adaptivnih rešetaka posebna procedura kategoriše postignute globalno najbolje rezultate na osnovu regiona kriterijumskog prostora u kojoj pripadaju. Prilikom odbacivanja elementa od REP-a rezultati najgušćih regiona imaju najveću šansu da budu odbačeni. Adaptivne rešetke takođe mogu da budu korišćene prilikom izbora elementa za pomeraj pozicije čestice.

3.6. Donošenje odluka pomoću kružne metrike

Za rangiranje rešenja korišćena je sledeća kružna metrika:

$$R_k = \sum_{i=1}^{M_{obj}} \left(\frac{f_i^{\min} - f_i^k}{f_i^0} \right)^2 \quad (16)$$

gde je:

- $f_1^{\min}$ – minimalna vrednost i -te kriterijumske funkcije;
- f_1^k – vrednost i -te kriterijumske funkcije u tački k ;
- f_i^0 – bazični vrednost kriterijumske i -te funkcije
- R_k – poluprečnik kruga tačke k ;
- k – indeks tačka Pareto fronta.

Izabrana čestica je ona koja ima najmanji poluprečnik kruga.

4. SIMULACIJE NA RAČUNARU

Verifikacija prikazanog MOPSO algoritma izvršena je na primeru 12.66 kV IEEE-33 mreže. Mreža ima 33 čvora, njeno maksimalno aktivno i reaktivno opterećenje je 3715 kW i 2300 kVAr, respektivno.

4.1. OPSDG za maksimalno opterećenje mreže

Traži se optimalno rešenje: gde postaviti i kakav upravljiv DG. Na raspolaganju su sledeći tipovi DG:

- Tip 1 – DG injektira samo aktivnu snagu u mreži;
- Tip 2 – DG injektira samo reaktivnu snagu u mreži;
- Tip 3 – DG injektira i aktivnu i reaktivnu snagu u mreži.

Injektiranja aktivne i reaktivne snage mogu da budu u opsegu 0÷2 MW i 0÷2 MVar, respektivno. Dozvoljene vrednosti napona svih čvora se nalaze u opsegu 0.95÷1.05 [r.j.]. DG mogu da se postave u svaki čvor DM, osim u koren DM. Za proračune tokova snaga potrošnja je modelovana konstantnim modulom struje i konstantnim faktorom snage. DM se napaja preko RTR prenosnog odnosa 1±12·1.25% [r.j.]. Optimalna rešenja su prikazana u tabelama 1 i 2.

Tabela 1. Optimalna rešenja: maksimalno opterećenje

Tip DG	Broj DG	Pozicija teretnog menjača	Pozicija DG	Injektirana aktivna i reaktivna snaga [MW][MVar]
Tip 1	1	3	10	1.873
	3	3	15, 24, 30	1.122, 1.227, 1.398
Tip 2	1	4	8	1.884
	3	4	7, 15, 30	0.878, 0.764, 1.819
Tip 3	1	3	8	P: 1.969 Q: 1.332
	3	3	11, 25, 30	P: 1.330, 0.508, 1.163 Q: 0.183, 0.834, 1.091

Tabela 2. Vrednosti kriterijumskih funkcija

Tip DG	Broj DG	Aktivni gubici mreže [kW]	VSI
Bez DG	–	182.49	25.96
Tip 1	1	124.54	29.59
	3	88.39	31.16
Tip 2	1	171.36	27.94
	3	177.27	29.26
Tip 3	1	80.98	30.62
	3	25.69	32.20

Napomena: ako pozicija regulacione sklopke nije obuhvaćena procesom optimizacije, ne može da se nađe optimalno rešenje za postavku jednog DG tipa 1 i tipa 2, a da se pritom ne naruše postavljena ograničenja.

Na osnovu prikazanih rezultata može se zaključiti da se postavljanjem više DG mogu značajno popraviti vrednosti kriterijumskih funkcija.

Tabela 3 su uporedno prikazane vrednosti rezultata

primene MOPSO i NSGA-II [8] algoritma. Rezultati su dobijeni prilikom postavljanja generatora tipa 3.

Tabela 3. Vrednosti kriterijumskih funkcija dva algoritma

Broj DG	1	2
Aktivni gubici [kW](MOPSO)	80.98	32.58
Aktivni gubici [kW] (NSGA-II)	71.75	33.93
VSI (MOPSO)	30.62	31.80
VSI (NSGA-II)	30.16	32.48

Na osnovu prikazanih rezultata može se uočiti da ni jedan od dva algoritma ne dominira. Na osnovu dodatne analize može se zaključiti da Pareto front NSGA-II algoritma ima bolju distribuciju u odnosu MOPSO, dok MOPSO algoritam karakteriše manje vreme proračuna u zavisnosti od broja kriterijumskih funkcija.

4.2. OPSDG i OPSKB za karakteristične dane i sezone

Traži se optimalno rešenje za postavljanje jedne vetroelektrane sa 1-4 generatora (nominalne snage 250 kW po generatoru), jedne solarne elektrane sa 1-4 generatora (nominalne snage 250 kW po generatoru) i jedne kondenzatorske baterije sa 1-8 kondenzatora (nominalne snage 125 kVAr po kondenzatoru). Napomena: primer odgovara postavljanju jednog upravljivog DG, tipa 1, čija injektirana snaga može da varira u opsegu 0-2 MW. U tabeli 4su prikazana optimalna rešenja:

Tabela 4. Optimalna rešenja: karakteristični dani i sezone

Element	Čvor	P [kW]	Q [kVAr]
Kondenzator	27	-	875
Solarna elektrana	33	1000	0
Vetroelektrana	18	1000	0
Upravljivi generator	6	1843	0

Vrednosti kriterijumskih funkcija su prikazane u tabeli 5:

Tabela 5. Vrednosti kriterijumskih funkcija

		Leto	Jesen	Zima	Proleće
Gubici energije za 24h [MWh]	Pre	2.481	0.294	0.637	0.385
	Posle	0.710	0.269	0.294	0.254
Prosečni VSI	Pre	0.775	0.916	0.879	0.904
	Posle	0.932	0.993	0.974	0.989
Prosečni naponski profil	Pre	0.937	0.978	0.968	0.975
	Posle	0.981	0.988	0.990	0.990

Rezultati jednoznačno ukazuju na smanjenje gubitaka u mreži, poboljšanje kriterijuma VSI-a i smanjenje odstupanja napona.

5. ZAKLJUČAK

U radu su razmatrane mogućnosti rešavanja OPSDG i OPSKB u radijalnim DM primenom MOPSO algoritma, za različite kriterijumske funkcije.

U prvom primeru, prikazano je postavljanje jednog ili višeg distributivnih generatora:

- MOPSO daje dobra rešenje problema OPSDG sa različitim tipovima DG;
- MOPSO algoritam vraća rezultate slične NSGA-II algoritma ako, međutim, povećanjem broja ulaznih promenljivih distribucija NSGA-II rezultata su bolji,

ali vremenski troškovi MOPSO-a su manji.

- Značajno bolji rezultati se dobijaju ako se optimizacijom obuhvate i RTR (često se bez njih ne može doći do rešenja koje je unutar propisanih granica).

U drugom primeru, razmatran je problem postavljanja vetro i solarnih generatora, odnosno KB-a na osnovu prognoziranih vrednosti proizvodnje. Na osnovu rezultata može se zaključiti:

- Rezultati MOPSO algoritma prikazuju dugoročne benefite postavljanja obnovljivih generatora u stanovištu kriterijumskih funkcija gubitaka aktivne snage, VSI i naponskih profila;
- MOPSO algoritam efektivno rešava PO prilikom povećanja broja kriterijumskih funkcija.

6. LITERATURA

- [1] S.A.A.Kazmi, H.K.Shahzad, D.R.Shin: Multi-Objective Planning Techniques in Distribution Networks: A Composite Review; *Energies*, Vol.10, Issue 2, Feb. 2017, pp. 208-251
- [2] V.Strezorski: *Analiza elektroenergetskih sistema*, Fakultet Tehničkih Nauka u Novom Sadu, Novi Sad, Srbija, 2006.
- [3] Y.M.Atwa, E.F.El-Saadany, M.M.A.Salama, R.Seethapathy: Optimal Renewable Resources Mix for Distribution System Energy Loss Minimization, *IEEE Transactions on Power Systems*, Vol 25, No. 1, Feb 2010, pp. 360-371.
- [4] R.Sirjani, A.Mohamed, H.Sharef:Heuristic Optimization Techniques to Determine Optimal Capacitor Placement and Sizing in Radial Distribution Networks:A Comprehensive Review, *Przeglad Elektrotechniczny*, Vol.88, Issue 7a, Jan. 2012, pp. 1-7.
- [5] M.Kumar, P.Nallagownden, I.Elamvazuthi: Optimal Placement and Sizing of Renewable Distributed Generations and Capacitor Banks into Radial Distribution Systems, *Energies*, Vol. 25, Issue 1, Feb. 2010, pp. 811-835.
- [6] C.A.Coello, G.T.Pulido, M.S.Lechuga: Handling Multiple Objectives With Particle Swarm Optimization, *IEEE Trans.on Evolutionary Computation*, Vol. 8. Issue 3, June 2004, pp. 256-279.
- [7] D.Shirmohammadi, H.W.Hong, A.Semlyen, G.X.Luo: A Compensation-Based Power Flow Method for Weekly Meshed Distribution and Transmission Networks, *IEEE Trans.on PS*, Vol. 3, Issue 2, May 1988, pp. 753-762.
- [8] K.Deb, A.Pratap, S.Agarwal, T.Meyarivan: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II, *IEEE Trans.on Evolutionary Computation*, Vol. 6, No. 2, April 2002, pp. 182-197.

Kratka biografija:



Balaž Kereši rođen je u Senti 1993. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetika – Elektroenergetski sistemi odbranio je 2019.god.

kontakt: kobalu@gmail.com

UPRAVLJANJE NEPLANIRANIM ISPADIMA U ELEKTRODISTRIBUTIVNIM MREŽAMA**UNPLANNED OUTAGE MANAGEMENT IN POWER DISTRIBUTION NETWORKS**

Aleksandra Tešić, Vladan Krsman, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu su obrađeni tipični procesi upravljanja neplaniranim ispadima koji su posljedica trajnih kvarova u radijalnim distributivnim mrežama u normalnim uslovima. Prikazane su razlike između upravljanja neplaniranim ispadima u neautomatizovanim i automatizovanim distributivnim mrežama. Dat je osvrt na prednosti koje se dobijaju automatizacijom distributivne mreže.

Ključne riječi: Neplanirani ispad, Distributivna mreža, Proces upravljanja

Abstract – This paper describes typical business processes for managing unplanned outages due to permanent faults in radial distribution networks under normal conditions. Differences between managing unplanned outages in non-automated and automated distribution networks are presented. A review of the benefits obtained from automatization of the distribution network is given.

Keywords: Unplanned outage, Distribution network, Business process

1. UVOD

Elektroenergetski sistem je složen tehnički sistem koji se bavi proizvodnjom, prenosom, distribucijom i potrošnjom električne energije, odnosno u okviru koga se vrše elektroenergetske transformacije iz električne energije ili u električnu energiju, njene unutrašnje transformacije, prenos i distribucija [1]. Kao dio elektroenergetskog sistema, elektrodistributivni sistem ima zadatak da raspodijeli i distribuira električnu energiju, od napojnih transformatorskih stanica do svakog potrošača.

Elektrodistributivni sistem, odnosno distributivnu električnu mrežu čine elektroenergetski objekti putem kojih se vrši distribucija električne energije do krajnjih potrošača. Distributivna mreža mora da bude izgrađena tako da i pri najvećim opterećenjima može da obavlja svoju funkciju, te da bude otporna na kvarove i poremećaje, odnosno da ih dovoljno brzo eliminiše i smanji trajanje prekida napajanja što je moguće više. To se postiže optimalnim upravljanjem naponskim prilikama, opterećenjima elemenata i gubicima u distributivnom sistemu, a takođe i uspostavljanjem preduzeća koja će biti zadužena za distribuciju električne energije. Ta preduzeća su distributivna preduzeća.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je dr Vladan Krsman, docent.

2. ORGANIZACIJA DISTRIBUTIVNOG PREDUZEĆA

Distributivna preduzeća se najčešće sastoje od slijedećih dijelova: kontrolni centar, departman za planiranje i investicije, departman za organizaciju posada, departman za odnose sa potrošačima i departman za informaciono-komunikacione tehnologije.

Kontrolni centar je glavni dio svakog distributivnog preduzeća. Kontrolna soba je operativni dio kontrolnog centra u kome se vrši nadzor i upravljanje svim aktivnostima u velikim distributivnim mrežama [2]. Kontrolna soba je uglavnom opremljena kontrolnim pločama i aplikacijama za praćenje, kontrolu i upravljanje distributivnom mrežom.

U svakodnevnom aktivnostima distributivnog preduzeća učestvuju nekoliko tipičnih korisnika, koji se mogu klasifikovati na: operatore – odgovorne za upravljanje mrežom, analizu primljenih promjena u polju, praćenje i kontrolisanje telemetrisanih tačaka u sistemu, donošenje odluka u oblastima za koje su nadležni da bi sistem bio siguran i pouzdan i da bi se obezbijedila sigurnost ekipa na terenu; dispečere – odgovorne za raspoređivanje i slanje posada na teren; osoblje na terenu – razne ekipe odgovorne za popravke ili održavanje mreže u polju.

3. ELEMENTARNE KARAKTERISTIKE DISTRIBUTIVNIH MREŽA

Distributivne mreže su konstruisane i planirane tako da se napajanje vrši u smjeru iz napojne transformatorske stanice VN/SN ka individualnim potrošačima, preko monofaznih, dvofaznih i trofaznih vodova. Vodovi povezuju napojnu transformatorsku stanicu VN/SN (110/35 kV/kV, 110/20 kV/kV, 110/10 kV/kV), sa transformatorskim stanicama SN/NN (35/0.4 kV/kV, 20/0.4 kV/kV, 10/0.4 kV/kV). Napajanje krajnjih potrošača na NN dijelu mreže se takođe vrši putem nadzemnih ili kablovskih vodova [2].

Svaki SN izvod na svom početku, odnosno u napojnoj TS VN/SN ima izvodni prekidač i relejnu zaštitu. Glavni zadatak izvodnih prekidača je da provode radne struje u svakodnevnom radu i da u kombinaciji sa zaštitnim relejom prekinu struju kvara ako se kratak spoj desi negdje na izvodu. U slučaju kvara u distributivnoj mreži, izvodni prekidači imaju zadatak da lokalizuju kvar, tako da što je moguće manje korisnika ostane bez napajanja. Većina kvarova na nadzemnim vodovima imaju prolazni karakter. Zbog toga se ugrađuju prekidači sa automatskim ponovnim uključanjem (APU, eng. recloser). Oni obez-

bjeđuju da nakon prekidanja struje kvara prekidač čeka neko predefinisano vrijeme i poslije toga se ponovo zatvara. Ako je kvar prolazan, prekidač ostaje u zatvorenom položaju (izvod je napojen i nikakvi dalji koraci nisu potrebni). Međutim, ako je kvar i dalje prisutan prekidač će ponovo prekinuti struju kvara, ostati otvoren neko vrijeme i ponovo se zatvoriti pod dejstvom APU.

Ako je i posle ovoga kvar još uvek prisutan prekidač će se otvoriti i ostati u tom položaju. Izvodni prekidači su najčešće telemetrisani, odnosno pod SCADA-om (eng. *Supervisory Control and Data Acquisition* – sistem za mjerenje, praćenje i kontrolu industrijskih sistema).

Linijski prekidači koji se nalaze duž izvoda su veoma često pod SCADA-om, dok osigurači koji štite TS SN/NN nisu pod SCADA-om, kao ni osigurači u NN dijelu distributivne mreže. SCADA prikuplja informacije sa elemenata u distributivnoj mreži, čime se omogućava praćenje stanja distributivne mreže u realnom vremenu i donošenje odluka u skladu sa trenutnim stanjem distributivne mreže.

4. ISPADI U ELEKTRODISTRIBUTIVNIM MREŽAMA

Neplanirani ispadi u elektrodistributivnim mrežama su najčešće posljedica kvarova u mreži koji uzrokuju otvaranje rasklopnih uređaja. Kvarovi mogu biti posljedica starosti ili neispravnosti opreme, električnih ili mehaničkih naprezanja, ali mogu biti uzrokovani i uticajima aktivnosti ljudi ili životinja, spoljašnjim faktorima kao što su vegetacija ili vremenski uslovi. S obzirom na to da su distributivne mreže uglavnom radijalne strukture, svaki ispad u mreži uzrokuje prekid isporuke električne energije potrošačima i to rezultuje neplaniranim ispadom. Distributivna preduzeća su u obavezi da u što kraćem vremenskom roku poprave kvar u mreži i da vrate napajanje potrošačima.

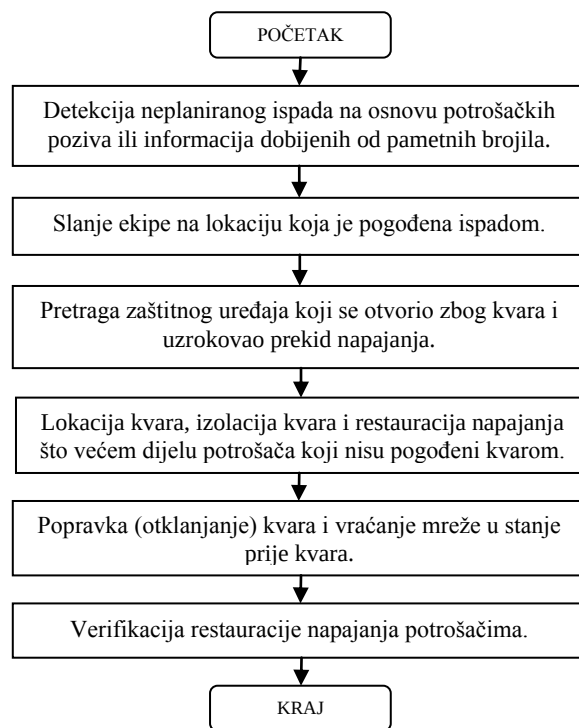
U kablovskim mrežama uzrok kvarova je najčešće starenje vodova, pojava i širenje parcijalnih pražnjenja u izolaciji kao i kidanje kablovskih vodova uslijed radova građevinskih mašina. Kvarovi u kablovskim mrežama su najčešće trajne prirode i zbog toga se ne mogu otkloniti korišćenjem prekidača sa APU [2].

U nadzemnim mrežama najčešći uzroci kvarova su naleti ptica, narasla vegetacija, atmosferska pražnjenja, zagađenost atmosfere, kidanje užadi usljed dodatnog opterećenja, vandalizam, starenje materijala od koga je napravljen vod i slično. Velika većina ovih kvarova je prolaznog karaktera.

U distributivnim preduzećima se razlikuje proces upravljanja neplaniranim ispadima, koji su posljedica trajnih kvarova, na neautomatizovanom dijelu distributivne mreže i proces upravljanja neplaniranim ispadima, koji su posljedica trajnih kvarova, na automatizovanom dijelu distributivne mreže.

5. NEPLANIRANI ISPAD U DUBINI MREŽE

Neplanirani ispad u dubini mreže je posljedica trajnog kvara koji nije uspješno eliminisan prekidačem sa APU.



Slika 5.1 – Algoritam upravljanja neplaniranim ispadom u dubini mreže

Na slici 5.1 prikazan je algoritam po kome se vrši upravljanje neplaniranim ispadom u dubini mreže, od detekcije samog ispada, preko otklanjanja kvara koji je prouzrokovao ispad, pa do vraćanja mreže u stanje prije kvara i verifikacije restauracije napajanja potrošačima.

Prvi korak u procesu upravljanja neplaniranim ispadom u dubini mreže je sama detekcija ispada. Pošto se radi o ispadu na netelemetrisanom uređaju, prve informacije o samom postojanju ispada se dobijaju preko korisničkih poziva. Ukoliko distributivno preduzeće koristi naprednu mjernu infrastrukturu (eng. *Advanced Metering Infrastructure* – AMI), informacije o ispadu se mogu dobiti i preko pametnih brojila. Nakon detekcije ispada, procjenom lokacija sa kojih su prijavljeni pozivi ili na kojima se nalaze pametna brojila, sistem može da odredi zaštitni uređaj koji se otvorio i uzrokovao prekid napajanja [3]. U sistemu se automatski kreira incident (softverski entitet) na zaštitnom uređaju, koji sadrži sve informacije o ispadu koje su potrebne operatoru u svrhu lakšeg dispečinga.

Operator u kontrolnoj sobi ima uvid u incidente koji trenutno postoje u sistemu. Nakon detekcije novog incidenta u sistemu, pristupa se njegovoj analizi. Operator vrši dodjelu najbliže ekipe koja ima vještine potrebne za detektovani tip ispada i obavještava ekipu da je potrebno da pristupe lokaciji na kojoj se desio ispad [4].

Nakon što ekipa stigne na lokaciju, vrši se provjera statusa zaštitnog uređaja. Ako je uređaj otvoren, incident se potvrđuje. Ako je uređaj zatvoren, ekipa traži novi zaštitni uređaj koji se otvorio i uzrokovao ispad, odnosno prekid napajanja. Suštinski, to je prvi korak koji ekipa mora da izvrši - da ustanovi koji zaštitni uređaj je otvoren. Drugi korak je pronalaženje same lokacije kvara, odnosno neispravne opreme. Nakon lokacije kvara, ekipa na osnovu instrukcija dobijenih od operatora vrši izolaciju

kvara otvaranjem rastavljača snage kako bi se odvojio dio mreže na kome je kvar. Zatim se vrši vraćanje napajanja dijelu mreže koji nije pogođen kvarom, kako bi se u što kraćem vremenskom roku vratilo napajanje što većem broju potrošača, preko alternativnih izvora - zatvaranjem normalno otvorenih rastavljača snage ka drugim izvodima. Nakon restauracije napajanja potrošačima koji se nalaze u dijelu izvoda koji nije pogođen kvarom, ekipa popravljiva kvar i oblast u mreži se vraća u normalno stanje, odnosno stanje koje je bilo prije kvara. Ukoliko ekipa nema potrebnu opremu da bi se otklonio kvar, operator mora angažovati neki spoljni departman, npr. departman za planiranje i investicije kako bi se izvršila zamjena opreme u polju.

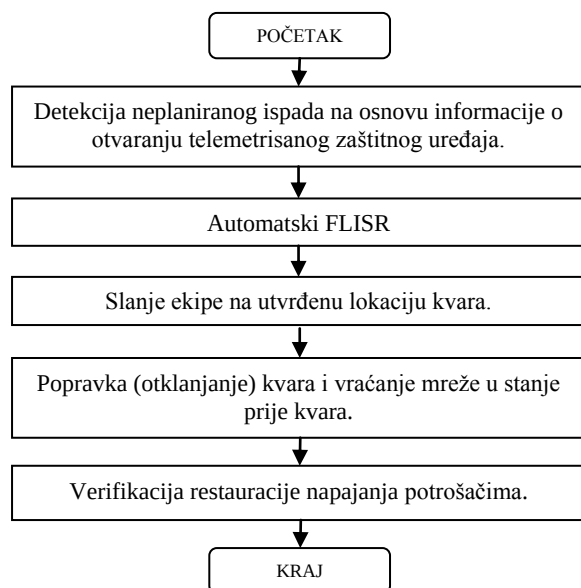
Kada se obnovi napajanje, potrebno je proći postupak verifikacije restauracije napajanja potrošača. Ta provjera se može sprovesti na dva načina:

- Preko telefonskog poziva – agenti u korisničkom centra pozivaju pojedinačne potrošače i pitaju da li im je vraćeno napajanje. Potrošači ili potvrđuju da im je napajanje vraćeno ili prijavljuju da su još uvijek bez napajanja.
- Preko pametnih brojlara – u distributivnim preduzećima sa AMI sistemom moguće je preko statusa pametnih brojila pojedinačnih potrošača provjeriti da li im je vraćeno napajanje.

Kada je sav rad u polju završen i verifikovana restauracija napajanja, operator zatvara incident, koji se nakon toga smješta u bazu podataka odakle se može koristiti za potrebe raznih dnevnih, mjesečnih i godišnjih izvještaja.

6. NEPLANIRANI ISPAD NA AUTOMATIZOVANOM DIJELU MREŽE

Neplanirani ispad na automatizovanom dijelu mreže je takođe posljedica trajnog kvara koji nije uspješno eliminisan prekidačem sa APU.



Slika 6.1 – Algoritam upravljanja neplaniranim ispadom u automatizovanom dijelu mreže

Na slici 6.1 prikazan je algoritam po kome se vrši upravljanje neplaniranim ispadom u automatizovanom

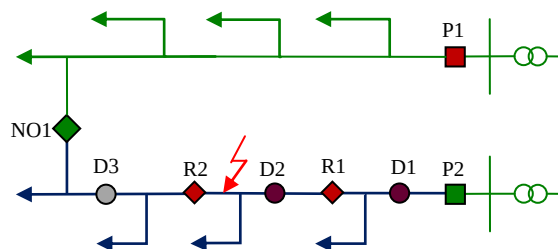
dijelu mreže, od detekcije ispada, preko izvršavanja automatskog FLISR-a (eng. *Fault Location, Isolation and Service Restoration* - proces lokacije kvara, izolacije dijela mreže na kome je kvar i restauracije napajanja dijelu mreže koji nije pogođen kvarom) i otklanjanja kvara, do vraćanja mreže u stanje prije kvara i verifikacije restauracije napajanja potrošačima.

Ukoliko su uređaji u napojnoj transformatorskoj stanici telemetrisani, kada se izvodni prekidač otvori uslijed kvara, ta informacija je odmah dostupna operatoru u kontrolnom centru. Nakon detekcije ispada, odnosno informacije da se desio neplanirani ispad na telemetrisanom izvodnom prekidaču, sistem automatski kreira incident [4].

S obzirom na to da otvaranjem izvodnog prekidača veliki broj potrošača ostaje bez napajanja, distributivnom preduzeću je u cilju da se vrijeme trajanja prekida napajanja smanji što je moguće više. Nakon detekcije ispada, vrši se lokacija kvara. Za lokaciju kvara se mogu koristiti mnogobrojni savremeni uređaji, a neki od njih su [2]:

- Detektori kvara – uređaji koji daju informaciju o tome da li je struja kvara protekla vodom na kome se nalazi detektor kvara. U slučaju naglog povećanja struje uzrokovanog kvarom, detektor se aktivira.
- Snimači kvara – uređaji čija je uloga da izmjere vrijednosti napona i struje kvara.

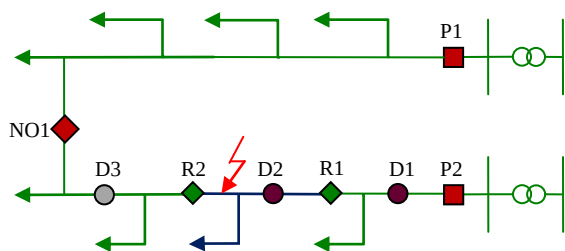
Ukoliko su detektori i snimači kvara telemetrisani, operator ima sve potrebne informacije da izvrši lokaciju kvara pokretanjem odgovarajuće softverske funkcije. Ulazni parametar za funkciju je zaštitni uređaj koji se otvorio uslijed kvara na štichenom dijelu mreže. Na osnovu statusa detektora kvara i vrijednosti napona i struja izmjerenih pomoću snimača kvara, kao i podataka o topologiji mreže, moguće je odrediti lokaciju kvara. Dodatno, operator ima mogućnost i ručnog unošenja dodatnih podataka koji mogu biti od koristi. Glavni cilj funkcije za lokaciju kvara je određivanje minimalnog skupa dijelova izvoda na kojima se pretpostavlja da je došlo do kvara, kao i procenta za svaki dio izvoda koji pokazuje kolika je vjerovatnoća da se kvar desio baš na tom dijelu. Što je veći broj detektora kvarova u mreži, rezultati funkcije će biti precizniji. Rezultati funkcije se mogu koristiti kao pomoć ekipama na terenu da brže lociraju kvar ili u automatskoj lokaciji kvara, praćenom automatskom izolacijom dijela mreže na kome se desio kvar i restauracijom napajanja preostalom dijelu mreže koji nije pogođen kvarom.



Slika 6.2 – Distributivna mreža sa kvarom na dijelu izvoda

Prilikom kvara na sekciji izvoda kao na slici 6.2, pod uticajem struje kvara aktivirani su detektori kvara D1 i D2. Pošto je kvar trajne prirode, prekidač P2 se otvorio i ostao u tom položaju, čime je uzrokovan prekid napajanja na dijelu izvoda ispod njega. Na osnovu aktiviranih detektora kvara, rezultat lokacije kvara je upravo sekcija izvoda na kojoj se desio kvar. U narednom koraku potrebno je izvršiti izolaciju tog dijela izvoda. To se omogućava otvaranjem rastavljača snage. Ukoliko su oni telemetrisani, otvaranje se može izvršiti direktno iz kontrolnog centra, a u suprotnom se mora poslati ekipa na teren koja će izvršiti potrebne akcije. Pošto je dio mreže razmatran u ovom primjeru telemetrisan, otvaranje rastavljača snage R1 i R2 se vrši automatski, bez potrebe za slanjem ekipa na teren.

Nakon izolacije dijela izvoda na kome se desio kvar, može se vratiti napajanje potrošačima koji se nalaze u zdravom dijelu izvoda, zatvaranjem normalno otvorenih rastavljača snage ka drugim izvodima. Skup uređaja čijim će se zatvaranjem vratiti napajanje svim dijelovima izvoda bez kvara se može dobiti pomoću funkcije za restauraciju napajanja. Ulazni podaci za funkciju su elementi, odnosno dijelovi izvoda kojima je potrebno vratiti napajanje i dijelovi izvoda koji moraju ostati bez napajanja, odnosno dijelovi izvoda na kojima je kvar. Funkcija prilikom traženja uređaja čijim će se zatvaranjem vratiti napajanje dijelovima izvoda može uzeti u obzir i potencijalna preopterećenja izvoda koji se koriste kao alternativni izvor napajanja, kao i preopterećenja zdravih dijelova izvoda kojima se vraća napajanje. Predloženi skup akcija se može izvršiti automatski, u slučaju da su predloženi uređaji telemetrisani, ili ručno – slanjem ekipa na teren. S obzirom na to da je dio distributivne mreže koji se razmatra u ovom primjeru telemetrisan, automatski se vrši zatvaranje normalno otvorenog rastavljača NO1 i prekidača P2.



Slika 6.3 - Distributivna mreža nakon izvršenog automatskog FLISR-a

Na slici 6.3 je prikazan izgled distributivne mreže nakon izvršenog automatskog FLISR-a. Potrošači koji su bez napajanja su samo oni potrošači koji se nalaze između rastavljača snage R1 i R2. Nakon izvršenog automatskog FLISR-a, operator šalje ekipu na utvrđenu lokaciju na kojoj se desio kvar. Ekipa treba da pristupi lokaciji na kojoj je utvrđeno da je kvar, da otkrije uzrok kvara i da pristupi njegovom otklanjanju. Ako ekipa ne može da izvrši popravku kvara, operator će inicirati zahtjev departmanu za planiranje i investicije, kako bi se izvršila zamjena opreme koja je u kvaru. Kada je kvar otklonjen, mreža se vraća u normalno stanje, odnosno stanje koje je bilo prije kvara.

Nakon što je obnovljeno napajanje, vrši se postupak verifikacije restauracije napajanja potrošača. Kada je utvrđeno da je svim potrošačima vraćeno napajanje, operator zatvara incident koji se zatim smješta u bazu podataka, odakle je dostupan za potrebe raznih izvještaja.

7. ZAKLJUČAK

Distributivna preduzeća u svakodnevnim aktivnostima obavljaju odgovarajuće procese upravljanja neplaniranim aktivnostima. Svi članovi osoblja distributivnog preduzeća imaju odgovarajuće uloge u tim procesima.

U slučaju da je distributivna mreža neautomatizovana i ukoliko se ne koriste napredna tehnička rješenja, u procesu upravljanja neplaniranim ispadom lokacija kvara traje jako dugo – nekoliko sati, pa čak i do 24h. Prije same lokacije kvara, ekipe na terenu moraju da pronađu zaštitni uređaj koji se otvorio i uzrokovao prekid napajanja, što takođe može da potraje ukoliko je dio mreže pogođen ispadom veliki. Za to vrijeme potrošači ostaju bez napajanja. Da bi se trajanje prekida napajanja potrošača smanjilo, koriste se napredna tehnička rješenja. Jedno od njih je i automatizacija distributivne mreže, pod kojom se podrazumijeva primjena SCADA sistema i softvera za upravljanje distributivnom mrežom – DMS (eng. *Distribution Management System*) u distributivnim preduzećima, sa daljinskom kontrolom svih napojnih transformatorskih stanica VN/SN i RTU (eng. *remote terminal units*) kod telemetrisanih SN prekidača i detektora kvara u dijelu distributivne mreže (10-20% od ukupne opreme) [5].

Upotrebom naprednih tehničkih rješenja se olakšava i ubrzava upravljanje pomenutim procesima, što se direktno odražava na profitabilnost distributivnog preduzeća.

8. LITERATURA

- [1] Vladimir Strezoski: *Osnovi elektroenergetike*, Fakultet tehničkih nauka Novi Sad, 2014.
- [2] Miroslav Nimrihter: *Elektrodistributivni sistemi*, Fakultet tehničkih nauka, Novi Sad, 2009.
- [3] Stuart Borlase: *Smart Grids: Infrastructure, Technology and Solutions*, Taylor & Francis Group, LCC, Boca Raton, Florida, 2013.
- [4] Vladan Krsman, Pisani materijal, Novi Sad, 2016.
- [5] Nenad Katic: *Menadžment sistemi u elektroenergetici DMS*, skripta, Fakultet tehničkih nauka, Novi Sad, 2013.

Kratka biografija:

Aleksandra Tešić rođena je u Tuzli 1992. godine. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranila je 2016. godine.

Vladan Krsman rođen je u Sarajevu 1985. godine. Doktorsku disertaciju odbranio je 2017. godine na Fakultetu tehničkih nauka iz oblasti Elektroenergetski sistemi.

**NAMENSKI JEZIK I OKRUŽENJE ZA MODELOVANJE I GENERISANJE
INFORMACIONIH SISTEMA OPŠTE NAMENE****A DOMAIN SPECIFIC LANGUAGE AND FRAMEWORK FOR MODELING AND
GENERATION OF GENERAL PURPOSE INFORMATION SYSTEMS**

Maja Zetko, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu prezentovan je generator izvornog Java koda. Dat je pregled modela i šablona, kao i metoda i tehnika koje su korišćene za dobijanje Java koda iz modela, kao i kreiranje namenskog jezika. Takođe, navedeno je poređenje između generatora programskog koda koji su trenutno dostupni na tržištu i generatora opisanog u ovom radu.

Ključne reči: Generator, definisanje šablona, model, namenski jezik

Abstract – In this paper, a generator of the Java source code is presented. We give a review of models and templates, as well as the methods and techniques used to obtain the Java code from a model, as well as creation of a DSL language. Also, we make a comparison of the code generators currently available in the market and the generator proposed in this paper.

Keywords: Generator, DSL, Template Detection, Model, Domain Specific Language

1. UVOD

U poslednjim decenijama primećuje se povećanje potrebe za specifičnim rešenjima informacionih sistema, kao i potreba da se postojeća rešenja menjaju u skladu sa promenama u poslovanju. Klijenti imaju potrebu za novim funkcionalnostima ili modifikacijama postojećih, koje mogu da zahtevaju značajne programske izmene. U neretkim situacijama izmene su prouzrokovane zakonskim izmenama kao i razvojem savremenih tehnologija iz oblasti poslovanja za čiju primenu je neophodno izvršiti izmene u načinu poslovanja. Razvojem tehnologija stvorena je mogućnost za automatizaciju velikog dela poslovnih procesa koje zahteva korišćenje više različitih informacionih sistema, koje je potrebno integrisati.

Sve veća složenost informacionih sistema u kombinaciji sa vremenskim ograničenjem za njihov razvoj motivisala je istraživanje pristupa razvoju programskih rešenja koji su u mogućnosti da ubrzaju isporuku proizvoda i smanjuju troškove. Konkurentnost na tržištu iziskuje da se izmene u načinu poslovanja implementiraju u što je moguće kraćem roku.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Ivan Luković, red. prof.

Model Driven Engineering (MDE) može doprineti bržem razvoju kvalitetnijih softverskih rešenja. U kontekstu MDE, model je opis sistema ili dela sistema zapisan korišćenjem dobro definisanog jezika koji podržava automatsku interpretaciju od strane računara. Model sistema na visokom nivou apstrakcije tretira se kao deo implementacije, za razliku od pristupa baziranih na modelima kod kojih se model koristio samo za potrebe analize, boljeg razumevanja i dokumentovanja sistema. Sistem predstavlja skup elemenata u interakciji i njihovih relacija. Apstrakcija je proces zanemarivanja nebitnih informacija prilikom kreiranja modela. Transformacija predstavlja automatsko generisanje ciljnog modela na osnovu početnog modela, u skladu sa definicijom transformacije. Definicija transformacije predstavlja skup pravila od kojih svako opisuje kako se jedna ili više konstrukcija iz početnog jezika može preslikati u jednu ili više konstrukcija ciljnog jezika.

Model je instanca metamodela, koji predstavlja jezik za specifikaciju modela. Metamodel se može definisati namenskim jezikom, koji je računarski jezik specijalizovan za određenu oblast primene [8]. Metamodel se specificira putem koncepata koji su definisani u meta-metamodelu. Meta-metamodel, metamodel, model sistema i instanca modela (instance klasa, korisnički podaci) formiraju četvoroslojnu arhitekturu metanivoa.

U ovom radu je analiziran pristup razvoja softvera zasnovanog na modelima prilikom izrade informacionih sistema. Proučeni su osnovni koncepti informacionih sistema opšte namene i formalno opisani namenskim jezikom. Opisan je razvoj i implementacija namenskog jezika, kao i razvoj programskog okvira za podršku modelovanju putem datog jezika. Kreiran je generator informacionog sistema zasnovan na pomenutom jeziku.

Pored navedenog, u samom master radu analizirana je strategija proširenja osnovnih funkcionalnosti generisanog informacionog sistema radi pružanja podrške u implementaciji realnih korisničkih zahteva. Proučena je opravdanost korišćenja kreiranog sistema i prateće metodologije razvoja softverskog rešenja u realnim uslovima.

2. PREGLED AKTUELNOG STANJA U OBLASTI

Na tržištu postoje generatori programskog koda čija kompleksnost u načinu korišćenja kao i dobijenog programskog koda, varira. Jedan od zastupljenijih je *Acceleo* [1], dodatak u *Eclipse* razvojnom okruženju, koji se koristi za generisanje tekstualnih programskih jezika iz

modela definisanog metamodelom Moguće je izabrati jedan od ponuđenih metamodela, dok je kreiranje specifičnog dosta jednostavno.

Kada se *Acceleo* dodatak instalira u *Eclipse*, dobije se nova *Acceleo* perspektiva i mogućnost kreiranja *Acceleo* projekta. Omogućeno je debugovanje za vreme generisanja, tako da se generisanje može pauzirati u određenom momentu kako bi se proverio dobijeni rezultat.

Jedan od jednostavnijih generatora je *Picocog* [2]. To je javno dostupna biblioteka za programski jezik *Java*. Šablon se kreira koristeći klasu *PicoWriter* i pozivajući njenih nekoliko jednostavnih metoda za ispisivanje teksta, kao što su *writeln_r()*, *writeln_l()* i *writeln_lr()*. Moguće je instancirati više puta klasu *PicoWriter* korišćenjem metode *createDeferredWriter()* i na taj način se može pisati kôd na različitim mestima.

Pored navedenih generatora programskog koda, pomoću kojih može da se generiše kôd pisan u više programskih jezika, postoje i oni koji su specijalizovani za određeni programski jezik. Jedan od njih je *Xtend*, koji generiše *Java* kôd. Može se koristiti bilo koja od postojećih biblioteka za programski jezik *Java*. Dobijeni kôd je čitljiv i ekvivalentan je ručno pisanom *Java* programskom kodu.

Spring Boot [6] je projekat koji omogućava jednostavniji način kreiranja i konfigurisanja aplikacija koje je moguće izvršavati. Uz razvoj *Spring Boot-a* razvijen je *Spring Roo* [3], koji naredbama kroz komandnu liniju generiše potreban kôd, uključujući dodavanje potrebnih biblioteka programskog jezika *Java* kao i kreiranje potrebnih metoda. Kada se *Spring Roo* instalira, preko *CMD* komandne linije se kucaju naredbe koje generišu programski kôd. Nije potrebno kreirati model, nego se generiše jedna po jedna komponenta. Moguće je dobiti kôd menjati u nekom od tekstualnih editora, i ta promena će biti vidljiva u komandnoj liniji. Ovim generatorom se brzo dolazi do funkcionalne aplikacije.

3. OPIS NAMENSKOG JEZIKA

Namenski jezik je računarski jezik specijalizovan za određenu oblast primene. U ovom radu je opisan namenski jezik koji sadrži sledeće koncepte:

- *Menu* - služi za generisanje strukture menija, koja može da ide u neograničenu dubinu. To se postiže tako što jedna instanca koncepta može da bude roditelj i/ili dete u odnosu na druge instance. Pri kreiranju menija prvo se uzimaju koncepti koji nemaju roditelja, i zatim njegova deca i to redosledom definisanim atributom *anOrder*. Ako je za stavku menija definisana validna oznaka objekta u atributu *anObject*, ta stavka postaje link za pristup formama i podacima vezanih za taj objekat.
- *Object* - predstavlja opis tabele čiji se podaci koriste na formi, kao i način prikazivanja podataka. Atributima koncepta se, između ostalog, definiše tabela u bazi, *where* i *order by* klauzule. Atributom *anPagination* se definiše da li se na listi podataka nalazi paginacija ili ne. Kada se na formi ne primenjuje paginacija, inicijalno se prikazuje prazna forma, i tek nakon izbora parametara za filtriranje i pretrage po tim parametrima, se prikazuju podaci.

- *ObjectItem* – povezan je sa konceptom *Object* i definiše kolone iz tabele koje će se prikazivati na formi aplikacije. Osnovni atribut ovog koncepta je *acType*, kojim se definiše tip podatka i način njegovog prikazivanja. *OCL* (Object Constraint Language) se koristi za definisanje obaveznih atributa u zavisnosti od tipa izabranog u *acType* atributu.
- *ObjectItemStyle* - pruža osnovne opcije koje utiču na izgled *jsp* stranica. Može se definisati *css* (Cascading Style Sheets) klase, *JavaScript* funkcije, kao i *HTML* (Hyper Text Markup Language) tagovi. Konkretno *css* klase i *JavaScript* funkcije nisu predmet ovog rada i potrebno ih je dodatno kreirati u importovati u projekat.
- *MenuLocalisation*, *ObjectLocalisation* i *ObjectItemLocalisation* – pružaju jednostavan oblik lokalizacije, pomoću koga je moguće generisati kôd aplikacije na više jezika. Navedeni koncepti su povezani sa adekvatnim osnovnim konceptom i omogućavaju da se tekstualni atributu, kao što je naziv, prevedu na više jezika.
- *PrivilegeMenu*, *PrivilegeObject* i *PrivilegeObjectItem* - služe da se obezbedi da svaka grupa korisnika ima odgovarajući pristup podacima. U navedenim konceptima je za svaku grupu korisnika definisao da li sme da vidi, menja ili dodaje podatke. Podaci o grupama korisnika i korisnicima nisu predmet ovog rada i potrebno ih je ručno implementirati u aplikaciji.

Definisanje konkretne sintakse podrazumeva definisanje rezervisanih reči koje će korisniku biti ponuđene i za koje će on trebati da veže konkretne vrednosti koje će se koristiti pri generisanju programskog koda. Rezervisane reči trebaju na jasan i nedvosmislen način da upute korisnika koje vrednosti treba da unese, kako bi dobio kôd odgovarao specifikaciji funkcionalnosti aplikacije koja je cilj generisanja.

Kada se završi popunjavanje tabela, korišćenjem *SQL* jezika baze koja se koristi generiše se tekst u formatu koji zahteva namenski jezik. Izbor baze podataka koja će se koristiti zavisi od potrebe klijenta. Za potrebe ovog rada koristi se *Microsoft SQL Database* baza i *T/SQL* jezik.

Korišćenjem *Eclipse* razvojnog okruženja može se direktno kreirati tekstualni fajl u formatu koji zahteva namenski jezik. Na ovaj način razvojno okruženje nudi predefinisane reči i vodi korisnika kroz definisanje koncepta i njihovih atributa.

4. GENERISANJE PROGRAMSKOG KODA

Generatori programskog koda su programi koji ulazne podatke prevode u programski kôd, za razliku od kompajlera, koji za rezultat daju izvršivi mašinski kôd. Važne odlike programskog koda dobijenog generisanjem su dobra čitljivost i brže testiranje, pošto se kôd dobija iz šablona koji su već formatirani i testirani.

Korišćenje *MVC2* (Model-view-controller) [4] obrazca i kreiranje višeslojnog programskog koda se pokazao veoma pregledno i korisno. *MVC2* se obično koristi za razvoj softvera koji deli aplikaciju u tri povezana dela. Ovo se radi radi razdvajanja internih prikaza informacija sa načinom na koji su informacije predstavljene i prihvaćene od strane korisnika. Model dizajnira ove

glavne komponente koje omogućavaju efikasnu ponovnu upotrebu kodova i paralelni razvoj.

- Model – Model sadrži podatke u obliku pogodnom za konkretnu primenu[11]. Metodi za generisanje modela su prosleđeni parametri: naziv paketa, objekat za koji se generiše model, lista *ObjectItem*-a vezanih za objekat koji će biti atributi modela i lista primarnih knjučeva. Modeli se generišu za sve tipove podataka koji se koriste u aplikaciji, što znači da će biti generisani za sve instance koncepta *Object*, kao i za sve instance koncepta *Choose* koji sadrže podatke o tabelama i kolonama koje se koriste u padajućim listama.
- Sloj za pristup podacima (*Data access object*, *DAO*) služi za direktan rad sa bazom podataka korišćenjem *Hibernate* biblioteke *Java* programskog jezika. Korišćenjem *Hibernate* anotacija nad atributima i pozivanjem *Hibernate* metoda za rad sa podacima, obezbeđuje se ispravnost podataka i koozistentno stanje baze podataka.
- Sloj za rad sa podacima (*Business object BO*) omogućava definisanje operacija koje je potrebno izvršiti nad podacima. *BO* sloj je smešten između kontrolera i *DAO* sloja, tako da kontroler komunicira samo sa *BO* slojem koji preuzima podatke iz *DAO* sloja.
- Kontroler – Kontroler ažurira model i prikaz. Unutar jedne klase može da bude više kontrolera koji su zasnovani na različitim funkcionalnostima aplikacije. Za pregled podataka se generišu dva kontrolera, jedan za *GET* poziv i jedan za *POST* poziv, dok se za otvaranje formi za izmenu podataka i za ažuriranje podatka generiše po jedan kontroler koji pozivaju odgovarajuće metode *BO* sloja.
- Komponente i komandni objekat – Pod komponentama se podrazumevaju pripereri (*Java* klase koje nasleđuju klasu *ViewPreparerSupport*) i validatori. Pripereri služe za pripremu podataka za prikaz. U njima se preuzimaju filteri sa forme, na osnovu kojih se kreira objekat za filtriranje podataka. Validatori služe za validaciju podataka preuzetih sa forme. Komandni objekti služe za jednostavnije prosleđivanje podataka na *jsp* stranicu.
- *JSP* stranice – Putem *jsp* stranice vrši se prikaz objekata u okviru aplikacije. Prva stranica koja se generiše predstavlja okvir unutar kojeg će se prikazivati lista podataka sa paginacijom. Na ovu formu se importuje stranica pod nazivom *taglib.jsp* u kojoj su odrađeni svi potrebni importi. Ova stranica je statička i ne generiše se nego se ručno kreira. Sledeće se kreira stranica na kojoj će se prikazivati lista sa podacima. Dalje se generiše stranica sa formom za pristup podacima u cilju dodavanja ili izmene.

Svaka *Java* klasa sadrži naziv paketa u kojem se nalazi. Pošto se klase različitog sloja programskog koda nalaze u različitim paketima koji ne moraju biti slični po nazivu, a cilj generatora opisanog u ovom radu je da omogućiti generisanje dela programskog koda koji će biti integrisan u postojeću aplikaciju, naziv paketa nije sadržan u modelu nego se prosleđuje kao ulazni parameter pri generisanju svakog sloja pojedinačno. Naziv paketa takođe

predstavlja lokaciju na disku na kojoj će se sačuvati dobijeni fajl.

5. PRIMER GENERISANE APLIKACIJE

Izabran je primer aplikacije koja služi za vođenje evidencije o zadacima radnika jedne konsultantske firme. Konsultanti komuniciraju sa komitentima i zajedničkim radom rešavaju svakodnevne probleme. Kako bi produktivno iskoristili vreme potrebno je da evidentiraju zadatak tako što će opisati uočeni problem i rezervisati vreme koje je predviđeno za njegovo rešavanje.

Osnovna funkcionalnost aplikacije je vođenje evidencije o zadacima. Zadatak, u najopštijem smislu znači bilo koja aktivnost radnika u toku radnog vremena. To može biti telefonski razgovor sa klijentom, konsultacija sa kolegom, sastanak sa nadređenima, odlazak na teren i sve ostalo što zaposleni radi u toku radnog vremena. Svaki zadatak ima tip koji se bira iz padajuće liste, i može da bude jedna od prethodno opisanih stvari. Zadatak može biti otvoren, zatvoren ili na čekanju. Za svaki zadatak je potrebno popuniti opis problema koji se rešava i na kraju opisati rešenje koje je sprovedeno.

Da bi bila napravljena forma za popunjavanje predhodno opisanih podataka, prvo je potrebno kreirati tabelu u bazi podataka. Za svaki *combobox* na formi se kreira posebna tabela u bazi podataka. Za *combobox* je u strukturi namenskog jezika kreirano više koncepata, koji kada se ispravno popune generišu potrebne *SQL* izraze. Ti koncepti su:

- *Choose* – za definisanje tabele i kolone primarnog ključa
- *Value* – za definisanje kolona koje treba prikazati pri izboru, a koje mogu i ne moraju biti deo primarnog ključa
- *Where* – za definisanje *where* klauzule, koji ne mora biti popunjen
- *Order* – za definisanje *order by* klauzule, koji ne mora biti popunjen

Kada se koristi baza podataka za kreiranje modela namenskog jezika, ti izrazi se pišu ručno.

- *SQL* izraz koji daje listu podataka - *select anUserId as 'acKey', ltrim(rtrim(acname)) + ' ' + LTRIM(rtrim(acSurname)) as 'acValue-1' from dbo.tHE_SetSubjcontact where acActive = 'T' and acUserId <> '' order by acName*
- *SQL* izraz koji daje jedan podatak na osnovu primarnog ključa - *(select ltrim(rtrim(acname)) + ' ' + ltrim(rtrim(acSurname)) from tHE_SetSubjcontact where anPrsn = anUserId) as anPrsn*

Osim navedenih, potrebno je kreirati instance koncepata *Object* za tabelu u kojoj se nalaze podaci o zadacima, *ObjectItem* sa poljima koji će se nalaziti na formi, *ObjectItemStyle* sa odacima o izgledu *jsp* stranice, kao i *ObjectPrivilege*, *ObjectLocalisation*, *ObjectItemPrivilege*, *ObjectItemLocalisation*, koji služe za definisanje prava pristupa podacima i lokalizaciju.

Da bi se dobio meni u aplikaciji potrebno je kreirati instance koncepata *Menu*, *MenuPrivilege* i *MenuLocalisation*.

Model za predhodno opisanu aplikaciju su kreirali menadžer prodaje i web dizajner, uz manju pomoć

programera, što je potvrdilo da model može biti kreiran od strane krajnjeg korisnika aplikacije.

Analizom dobijenog programskog koda uočena su dva problema.

Prvi problem su predstavljale *jsp* stranice koje nisu bile ispravne. *Jsp* stranica sa listom podataka nije bila ispravno povezana sa odgovarajućim kontrolerom, zbog čega nije bilo moguće filtriranje podataka, kao ni prelaz na stranicu za izmenu podataka. Na stranici za izmenu podataka nisu bile dodate *Java Script* funkcije definisane u modelu, zbog čega nije radila validacija i formatiranje podataka.

Drugi problem je predstavljala realizacija preslikavanja kompozitnog ključa u modelu podataka. Preslikavanje kompozitnog ključa se radi preko unutrašnje klase koja ima sufiks *PK* i koja sadrži atribut koji ulaze u kompozitni ključ. Pri generisanju unutrašnje klase, atributi nisu bili onih tipova podataka kako su definisani u modelu.

Navedeni problemi prate trenutno razvijenu verziju generatora i biće predmet rešavanja u budućem radu. Ostali delovi generisanog programskog koda su bili ispravni i nije bila potrebna dodatna korekcija. Generator programskog koda nije u potpunosti ispunio očekivanja, međutim brzina i količina dobijenog kvalitetnog koda opravdava dalja ulaganja u njegov razvoj.

6. ZAKLJUČAK

Usled povećane složenosti i brzine izmena u poslovnim procesima, kao i povećane konkurencije na tržištu, bilo je potrebno omogućiti brži razvoj informacionih sistema. Brži razvoj nije smeo da umanjí kvalitet dobijenog rešenja, kao ni definisanje zahteva od strane krajnjeg korisnika.

U ovom radu je opisan razvoj i korišćenje generatora koda za programski jezik *Java*. Namenski jezik kreiran za potrebe ovog rada sadrži koncepte *Object*, *ObjectItem* i *ObjectItemStyle*, na osnovu kojih se podaci čitaju iz baze podataka i prikazuju na formi u obliku definisanom atributom *acType* koncepta *ObjectItem*. Kreirani su koncepti *Choose*, *Value*, *Where* i *Order* koji služe za generisanje dodatnih upita za preuzimanje podataka za popunjavanje combobox elemenata. Konceptima *MenuLocalisation*, *ObjectLocalisation* i *ObjectItemLocalisation* definisan je jezik dobijene aplikacije, dok su konceptima *PrivilegeMenu*, *PrivilegeObject* i *PrivilegeObjectItem* definisana prava pritupa. Generatorom se dobija višeslojan kôd koji je lako čitljiv i razumljiv.

Prednost ovog generatora u odnosu na druge dostupne na tržištu je u tome da kreiranje modela može da uradi krajnji korisnik. Na taj način se olakšava komunikacija između programera i krajnjeg korisnika. Takođe je prednost što je model moguće definisati na dva načina, popunjavanjem tabele u bazi podataka i generisanjem tekstualnog fajla korišćenjem dostupnih alata izabrane baze, ili korišćenjem razvojnog okruženja *Eclipse*. Korišćenjem *Eclipse* razvojnog okruženja može se direktno kreirati tekstualni fajl u formatu koji zahteva namenski jezik. Na ovaj način razvojno okruženje nudi predefinisane reči i vodi korisnika kroz definisanje koncepta i njihovih atributa.

Programski kôd dobijen generisanjem nije potpun i potrebno ga je ručno korigovati, što nije preporučljivo. Sledeći korak u razvoju generata bi bio ispravka uočenih grešaka opisanih u prethodnim poglavljima. Zatim, je potrebno obezbediti način integracije ručno pisanog programskog koda sa generisanim, tako da se generisani kôd ne menja.

Korišćenjem namenskog jezika i generatora programskog koda opisanih u ovom radu, može se dobiti programski kôd aplikacije koji se može koristiti kao polazna tačka u razvoju složenih informacionih sistema. Iako u ovom momentu postoje nepokriveni aspekti u dobijenom programskom kodu, brzina i količina dobijenog kvalitetnog koda opravdava dalja ulaganja u njegov razvoj.

Dalji razvoj generatora programskog koda, opisanog u ovom radu, odnosio bi se na kreiranje novih tipova *ObjectItem* komponente. Kao što je navedeno, na osnovu vredosti atributa *acType* se definiše izgled i funkcionalnost polja na formi.

7. LITERATURA

- [1] "Acceleo/Getting Started" https://wiki.eclipse.org/Acceleo/Getting_Started
- [2] "Code Generation Is Easy - With Picocog" <https://dev.to/ainslec/metaprogramming-with-picocog-440d>
- [3] "Spring Roo" <https://docs.spring.io/spring-roo/reference/html/intro.html>
- [4] "Model-view-controller" <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
- [5] "MVC 2 Architectural Diagram" <http://springjavatutorial.blogspot.com/2014/05/mvc1-vs-mvc2.html>
- [6] "UI design with Tiles and Struts", Prakash Malani 2002., <https://www.javaworld.com/article/2073902/ui-design-with-tiles-and-struts.html>
- [7] "Uvod u MVC dizajn patern", Aleksa Vidović 2017., <https://startit.rs/dizajn-paterni-mvc/>
- [8] "Spring Boot" <https://spring.io/projects/spring-boot>
- [9] Model-driven Software Engineering (MDSE) in Practice. The book on MDD, MDE, MDA, MD* by Marco Brambilla, Jordi Cabot, and Manuel Wimmer. 2nd edition. Morgan & Claypool, 2017. ISBN
- [10] "Domain-specific language" https://en.wikipedia.org/wiki/Domain-specific_language

Kratka biografija:



Maja Zetko rođena je u Novom Sadu 1987. Fakultet tehničkih nauka upisala je 2006. god. Bečelorski rad iz oblasti Elektro-tehnike i računarstva odbranila je 2011. Master studije upisala je 2016. godine.

IMPLEMENTACIJA MOBILNE APLIKACIJE TRAVELING POMOĆU REACT NATIVE RADNOG OKVIRA**IMPLEMENTATION OF MOBILE APPLICATION TRAVELING USING REACT NATIVE FRAMEWORK**

Nataša Subić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – RAČUNARSTVO I AUTOMATIKA

Kratak sadržaj – U ovom radu prezentovan je alat za izgradnju mobilnih aplikacija React Native. Kroz primer implementacije rešenja prikazani su osnovni koncepti ovog radnog okvira kao i biblioteke pomoću kojih je rešenje implementirano.

Ključne reči: JavaScript, React Native, REST

Abstract – This paper presents React Native tool for mobile application development. The paper contains basic concepts of this tool and descriptions of libraries used for implementation of the application.

Keywords: JavaScript, React Native, REST

1. UVOD

Sa napretkom tehnologije došlo je do napretka mobilnih uređaja. Mobilni uređaji razvijaju se izuzetnom brzinom, povećava se pristupačnost istih i raste broj funkcionalnosti koje se pomoću njih mogu obaviti. Sve više i više veb aplikacija razvija se prvo za mobilne uređaje a kasnije im se prikaz prilagođava i za veće ekrane. Kako mobilni uređaji napreduju tako se razvijaju i mobilne aplikacije. Razvijaju se novi radni okviri za kreiranje mobilnih aplikacija i unapređuju postojeći. Jedan od radnih okvira za razvijanje mobilnih aplikacija jeste React Native. Ovaj radni okvir zasniva se na JavaScript programskom jeziku i o njemu će biti više reči u nastavku ovog rada. Tema ovog rada je mobilna aplikacija implementirana pomoću React Native radnog okvira.

2. REACT NATIVE

React Native je radni okvir pomoću kog se kreiraju mobilne aplikacije za Android i iOS mobilnu platformu [1]. Za razvijanje aplikacija koristi JavaScript programski jezik uz XML tagove i sve komponente prevodi u kod nativan odgovarajućoj platformi. Prednost ovog radnog okvira je što nema potrebe za razvojem aplikacija za pojedinačnu platformu. Mane su što se trenutno mogu razvijati aplikacije samo za Android i iOS mobilnu platformu. Bitno je istaći da postoji velika sličnost između ReactJS biblioteke i React Native radnog okvira.

ReactJS je JavaScript biblioteka pomoću koje se kreira korisnički prikaz za veb aplikacije [2]. Mnogi principi koji se koriste u ReactJS biblioteci koriste se i u React Native radnom okviru.

2.1. React Native komponenta

Komponenta predstavlja osnovnu jedincu za izgradnju prikaza aplikacije [3]. Ona omogućuje da čitav korisnički prikaz podelimo u više međusobno nezavisnih jedinica. Svaka od tih jedinica posmatra se kao izolovana celina koja se može više puta upotrebljavati u aplikaciji. Jedna izolovana celina čini blok čijim se spajanjem kreira potpun prikaz korisničkog interfejsa. React komponenta može biti JavaScript klasa ili funkcija. Funkcionalna komponenta koristi se za pisanje komponenti koje nemaju svoje stanje već samo prikazuju neke elemente korisničkog interfejsa. Klasne komponente služe za kreiranje kompleksnijih komponenti koje čuvaju i menjaju svoje stanje i koriste metode životnog ciklusa komponente.

2.2. JSX

JSX ili JavaScript eXtension predstavlja proširenje JavaScript-a i služi za opisivanje izgleda korisničkog interfejsa. JSX kombinuje XML sintaksu sa tagovima za označavanje i JavaScript kod [4]. JSX ne razdvaja logiku od prikaza nego ih sve spaja u jedinice odnosno komponente. Ona omogućuje ubacivanje prethodno definisanih promenljivih i poziv funkcija u HTML kod.

2.3. State i props

Razlikujemo dva tipa podataka koji kontrolišu komponentu. To su stanje komponente i svojstva komponente, odnosno props. Svojstva komponente su podaci koji se ne menjaju u toku života komponente. Za promenljive podatke koristi se stanje komponente.

Većina komponenti može se prilagoditi prilikom kreiranja pomoću različitih parametara. Ovi parametri nazivaju se props (skraćeno od properties, odnosno svojstva). Props-i služe za prosleđivanje podataka i svojstava od jedne komponente do druge. Obično roditelj komponenta prosleđuje svojstvo ili podatke dete komponenti. Oni omogućuju dinamiku i prilagodljivost svakog prikaza. Stanje komponente predstavlja objekat koji određuje kako će se neka komponenta ponašati ili šta će neka komponenta prikazivati. Samo klasne komponente mogu imati stanje. Ono uvodi dinamiku u komponente. Stanje jedne komponente može da pristupa samo komponenta koja je njen vlasnik i ona može da je menja.

NAPOMENA:

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

Ukoliko je vrednost iz stanja potrebno proslediti drugoj komponenti to se može učiniti tako što se ta vrednost prosledi kao prop, komponenti koja je u hijerarhijskom poretku dete komponente iz koje se stanje prosleđuje. Svakom promenom stanja ili props-a komponenta ponovo osvežava svoj prikaz. Što je jedna od karakteristika React biblioteke i React Native radnog okvira.

2.4. Metode životnog ciklusa komponente

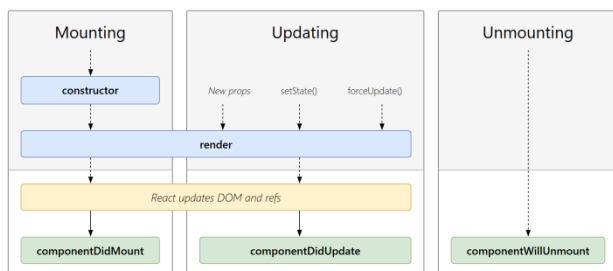
Jedna komponenta ima svoj životni ciklus od kreiranja pa do brisanja.

U različitim fazama životnog ciklusa u kojima se komponenta može naći pozivaju se različite metode.

Ove metode omogućuju da se na racionalniji način koriste resursi na mobilnom telefonu, da se kreira kvalitetnija aplikacija kao i da se isplanira šta i kako će se odraditi u određenoj fazi. Razlikujemo tri faze u kojima se komponenta može naći (Slika 1).

To su:

- Mounting,
- Ažuriranje,
- Unmounting.



Slika 1. Životni ciklus komponente i metode koje se pozivaju [5].

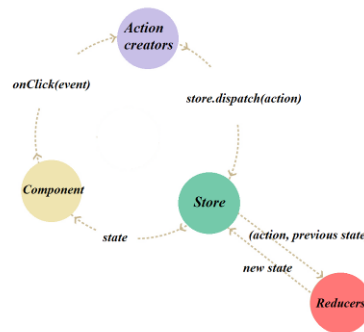
U svakoj od ovih faza pozivaju se odgovarajuće funkcije životnog ciklusa. One omogućuju pravilnu manipulaciju podacima. Na slici 1 prikazan je životni ciklus React komponente za veb aplikacije. Iste faze životnog ciklusa primenjive su i na React Native komponentu. U prvoj fazi se inicijalizuje komponenta. U ovom periodu komponenta još uvek nije prikazana. Nakon inicijalizacije poziva se `componentWillMount()` metoda koja se poziva samo jednom pre metode `render()` i služi najčešće za dobavljanje podataka sa nekog eksternog servisa. `render()` je naredna metoda koja se pozove.

Ova metoda prikazuje elemente korisničkog interfejsa. Ona mora da vrati ili React Native komponentu, odnosno element ili ništa. Sledeća metoda koja se poziva jeste `componentDidMount()`, ona se isto kao i `componentWillMount()` poziva jednom ali razlika je u tome što se ona poziva tek nakon što je komponenta završila sa prikazivanjem elemenata korisničkog interfejsa. Druga faza je faza ažuriranja. Metode iz ove faze pozivaju se samo ukoliko je došlo do izmene stanja ili props-a.

Ukoliko dođe do ove izmene ponovo se poziva metoda `render()` koja sada prikazuje izmenjeno stanje i izmenjen props. `componentDidUpdate()` poziva se nakon što se elementi korisničkog interfejsa ponovo prikažu. Poslednja metoda koja se poziva jeste `componentWillUnmount()`. Ona se poziva pre nego što se komponenta uklanja iz prikaza i uništava.

2.5. Redux

Redux je JavaScript biblioteka koja manipuliše stanjem aplikacije i kontroliše tok podataka kroz aplikaciju [6]. Bitno je znati kada je treba koristiti. Ukoliko ima više podataka koji se menjaju tokom vremena i potreban je jedan izvor podataka za koji se uvek zna da je istinit onda treba koristiti Redux. Ako je aplikacija jednostavna nema potrebe za upotrebom Redux-a. Sve što je potrebno jeste da se određeni podaci čuvaju u stanju komponente na višem hijerarhijskom nivou. Na slici 2 prikazan je tok podataka Redux-a sa metodama koje se pozivaju.



Slika 2. Tok podataka u Redux-u.

Skladište (Store) predstavlja jezgro Redux-a koje povezuje akcije i reduktore. Postoji samo jedno skladište koje sadrži stanje aplikacije. Reduktor (Reducer) je čista funkcija koja prihvata prethodno stanje i akciju, obrađuje dobijene podatke i na osnovu te obrade vraća novo izmenjeno stanje ili trenutno stanje.

Promene stanja moraju biti immutable (nepromenljive) što znači da se objekat stanja ne sme menjati nakon što je kreiran, nego se uvek kreira novi objekat sa izmenjenim svojstvima. Akcije predstavljaju jedini izvor informacija za skladište. One su običan JavaScript objekat.

Akcije samo opisuju šta se dogodilo ne kako se stanje promenilo. Da bi se kreirala akcija koriste se funkcije za kreiranje akcija koje se nazivaju Action creators. Da bi se stanje izmenilo poziva se `dispatch()` metoda iz skladišta koja okida odgovarajuću akciju. Ta akcija prosleđuje se reduktoru koji je obrađuje i ažurira stanje.

3. REST

REST je arhitektonski stil koji obezbeđuje standardizaciju u komunikaciji između različitih sistema [7]. On uvodi ograničenja koja doprinose skalabilnosti i boljim performansama sistema u kom se isti koristi. REST arhitektonski stil definiše određena ograničenja koja se moraju poštovati. Adresabilnost je jedno od njih. Ovo ograničenje definiše da svaki resurs u sistemu mora imati jedinstveni identifikator, mora postojati uniformisani interfejs koji pojednostavljuje i odvaja arhitekturu.

Zatim, sve poruke moraju biti samoopisive poruke što znači da svaka poruka treba da sadrži dovoljno informacija o tome kako je obraditi.

Komunikacija treba da bude bez čuvanja stanja. Serverska strana ne čuva klijentsku sesiju, samo čuva stanje resursa koje nudi. Ovaj stil odvaja klijentsku od serverske strane.

3. IMPLEMENTACIJA

Predmet ovog rada je softversko rešenje predstavljeno u vidu mobilne aplikacije. Mobilna aplikacija pod nazivom Travelimg nudi korisniku prikaz i pretragu znamenitosti u različitim gradovima uz oslonac na servise za lokaciju. Aplikacija zahteva pravljenje naloga i prijavu na isti. Korisniku takođe stoje na raspolaganju funkcionalnosti za prikaz, obeležavanje i pretragu znamenitosti. Pored toga korisnik ima mogućnost da vidi detalje konkretne znamenitosti i da izmeni podatke korisničkog naloga. Mobilna aplikacija implementirana je pomoću React Native radnog okvira, a server sa kojim ova aplikacija komunicira implementiran je pomoću Spring Boot radnog okvira.

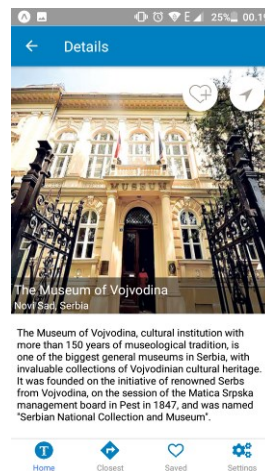
Nakon što korisnik instalira aplikaciju prvi ekran koji mu se prikaže jeste ekran za prijavu na sistem. Da bi se korisnik uspešno prijavio na sistem mora se prethodno registrovati. Za prijavu na sistem potrebno je uneti korisničko ime i lozinku. Da bi se korisnik uspešno registrovao potrebno je da unese korisničko ime, e-mail adresa i lozinka. Nakon što unese odgovarajuće podatke korisnik se preusmerava na početnu stranicu aplikacije. Sva polja za unos u okviru aplikacije se validiraju pre nego što se podaci pošalju eksternom servisu. Podaci koje korisnik unese šalju se serveru na proveru pomoću axios biblioteke [8]. Axios biblioteka koristi se za kreiranje HTTP zahteve i na taj način omogućuje komunikaciju sa REST API-jem. Pomoću ove biblioteke moguće je dobavljati i slati podatke koristeći GET, POST, PUT i DELETE zahteve iz aplikacije. Axios se zasniva na takozvanom Promise objektu. U Java Script-u Promise je objekat koji reprezentuje kompletno izvršavanje ili prekid asinhronih operacija.



Slika 3. Prikaz početnog ekrana mobilne aplikacije.

Početni ekran, prikazan na slici 3, sadrži listu znamenitosti sa nazivom svake znamenitosti. U okviru ovog ekrana može se vršiti pretraga znamenitosti po nazivu ili lokaciji znamenitosti, odnosno po gradu i državi. Podaci koji se prikazuju u listi preuzimaju se iz skladišta podataka. To skladište kreira se upotrebom Redux biblioteke koja je opisana u prethodnom poglavlju. Redux je implementiran tako što je cela aplikacija izdvojena u domene. Ti domeni su na primer domen znamenitosti ili domen korisnika. Za svaki od ovih domena kreirani su posebni tipovi, akcije kao i reduktori.

Za svaku akciju definisan je tip. Ovi tipovi koriste se u okviru akcije kao i reduktora. Akcije predstavljaju jedini izvor informacija koje se čuvaju u okviru stanja u skladištu. Reduktori definišu kako će se stanje aplikacije menjati u zavisnosti od akcije. Akcije samo opisuju šta se desilo, a reduktori opisuju promene stanja skladišta. Pritiskom na konkretnu znamenitost na početnom ekranu, ekranu sa preporučenim ili obeleženim znamenitostima odlazi se na ekran sa prikazom detalja znamenitosti prikazan na slici 4.



Slika 4. Prikaz detalja znamenitosti.

Ekran sa mapom znamenitosti sadrži prikaz mape na kom se nalaze najbliže znamenitosti. Najbliže znamenitosti definišu se tako što se preuzme trenutna lokacija mobilnog telefona i prikažu se znamenitosti koje su u radijusu od distance koju je korisnik uneo u svojim korisničkim podešavanjima. Inicijalna vrednost te distance je 500 metara. Za prikaz mape koristi se React Native Maps biblioteka [9]. Da bi se ova biblioteka koristila potrebna su dodatna podešavanja. S obzirom na to da se prikaz radijusa koristi trenutna lokacija istoj se mora pristupiti putem telefona. Za to je potrebno da se definišu dozvole kao i prilikom implementacije nativnih aplikacija.



Slika 5. Prikaz mape sa znamenitostima.

Za prikaz mape koristi se Google mapa. Da bi se ista prikazala potrebno je u okviru aplikacije definisati API ključ koji se dobija od Google-a i koji omogućuje upotrebu mapa.

Bez istog se Google mapa ne bi mogla biti prikazana.

Pored prikaza znamenitosti na početnom ekranu korisnik ima mogućnost da vidi preporučene znamenitosti kao i one znamenitosti koje je sačuvao preko ekrana za prikaz detalja znamenitosti.

Preporučene znamenitosti su lista onih znamenitosti koje su korisnici najviše dodavali u svoju listu sačuvanih znamenitosti. Prikaz liste preporučenih znamenitosti ne samo po podacima koji se prikazuju već i po tome što se sačuvane znamenitosti grupišu po gradovima u kojima se znamenitost nalazi. Još jedna od razlika jeste to što u stavci znamenitosti ne prikazuje samo naziv znamenitosti nego i trenutna udaljenost od te znamenitosti. Pored ovih funkcionalnosti korisnik može da izmeni podatke svog korisničkog naloga kao i distancu koja se koristi prilikom prikaza mape sa najbližim znamenitostima.

Serverska strana implementirana je uz pomoć Java programskog jezika u vidu REST API-ja. API omogućuje komunikaciju između dva softvera. U okviru implementirane aplikacije svaki zahtev koji se šalje ka serveru proizvodi odgovarajući odgovor. Kada se zahteva određeni resurs taj odgovor formatiran je u JSON formatu.

Pored toga u odgovoru se može naći i informacija o uspešnosti izvršavanja određene akcije. REST je stateless što znači da ne čuva stanje aplikacije. U okviru rešenja ovaj stil se koristi preko HTTP protokola. Potrebno je navesti i način na koji se kreira šema baze podataka. Za to se koristi Hibernate biblioteka [10]. Pomoću ove biblioteke stavljaju se anotacije iznad polja određenog modela i na osnovu tih anotacija kreiraju se tabele u bazi sa definisanim kolonama.

4. ZAKLJUČAK

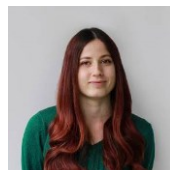
U ovom radu predstavljen je opis softverskog rešenja za prikaz i pretragu znamenitosti. Opisane su funkcionalnosti klijentske strane odnosno mobilne aplikacije. Za implementaciju klijentske strane korišćen je React Native radni okvir i JavaScript programski jezik. Serverska strana implementirana je pomoću Spring Boot radnog okvira i Java programskog jezika. Podaci su modelovani pomoću Java programskog jezika i Hibernate biblioteke na sistemu za upravljanje bazama podataka MySQL.

Softversko rešenje predstavljeno u ovom radu moglo bi se proširiti na više načina. Jedan od njih su obaveštenja koja bi stizala korisniku kada se određeno obeležje nalazi u njegovoj blizini. Pored toga potrebno je dodati mogućnost dodavanja komentara i prikaz istih u okviru stranice sa detaljima znamenitosti, kao i podršku za izmenu lozinke u slučaju zaboravljene lozinke. Da bi aplikaciju mogla više da se upotrebljava potrebno je odraditi lokalizaciju. Još jedna od mogućnosti jeste dodavanje audio zapisa za svako obeležje tako da korisnik ne mora da čita opis obeležja već može da sluša o istom putem tog zapisa.

5. LITERATURA

- [1] React Native – A framework for building native apps using React, <https://facebook.github.io/react-native/> (pristupljeno u aprilu 2019.)
- [2] React - A JavaScript library for building user interfaces, <https://reactjs.org/> (pristupljeno u aprilu 2019.)
- [3] React Components and Props, <https://reactjs.org/docs/components-and-props.html> (pristupljeno u aprilu 2019.)
- [4] JSX, <https://reactjs.org/docs/introducing-jsx.html> (pristupljeno u aprilu 2019.)
- [5] Metode životnog ciklusa komponente, <http://projects.wojtekmaj.pl/react-lifecycle-methods-diagram/> (pristupljeno u aprilu 2018.)
- [6] Redux biblioteka, <https://redux.js.org/> (pristupljeno u aprilu 2019.)
- [7] RESTful Java with JAX-RS 2.0, Bill Burke
- [8] Axios biblioteka, <https://github.com/axios/axios> (pristupljeno u aprilu 2019.)
- [9] React Native Maps библиотека, <https://github.com/react-native-community/react-native-maps>
- [10] Hibernate, Java persistence with Hibernate, Christian Bauer and Gavin King

Kratka biografija:



Nataša Subić rođena je 02.01.1994. godine u Novom Sadu. Završila je srednju ekonomsku školu „Svetozar Miletić“ u Novom Sadu 2013. godine. Fakultet tehničkih nauka u Novom Sadu upisala je 2013. godine, odsek Računarstvo i automatika. Osnovne studije završila je 2017. godine i iste se upisala na master studije, takođe na Fakultetu tehničkih nauka. Ispunila je sve obaveze i položila je sve ispite predviđene studijskim programom.

POSLOVNI INTEGRACIONI ŠABLONI I NJIHOVA UPOTREBA U APACHE CAMEL OKRUŽENJU**ENTERPRISE INTEGRATION PATTERNS AND THEIR USE WITHIN APACHE CAMEL FRAMEWORK**Marko Kljajić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Rad objašnjava šta predstavlja pojam integracije sistema i koje tipove integracije je moguće primeniti. Najveći deo rada posvećen je razmeni poruka kao jednom od načina integracije.

Ključne reči: Integracija sistema, Razmena poruka

Abstract – This paper discusses what system integration is and which types of integration exist. The major focus is on integration using messaging systems.

Keywords: System integration, Messaging

1. UVOD

Programska rešenja se retko izvršavaju izolovano, česta potreba da se prilikom izrade nekog poslovnog sistema uspostavi komunikacija sa drugim sistemima radi razmene potrebnih informacija uvodi potrebu za integracijom sistema [1]. Sistemi koji u svom radu vrše komunikaciju sa drugim sistemima kako bi postigli određenu funkcionalnost iz specifikacije samog sistema nazivaju se integracionim sistemima.

-Prilikom njihove izrade nailazi se na nekoliko osnovnih problema kao što su nepouzdanost i spore mreže u poređenju sa sistemima koji se izvršavaju na jednom računaru, razlike u operativnim sistemima, programskim jezicima, pa i samoj predstavi podataka, česte promene u sistemima što izaziva promene u svim sistemima koji učestvuju u komunikaciji.

Vremenom su nastajala različita rešenja koja su doprinosila da se osnovni problemi prebrode:

- Prenos fajlova – jedan sistem beleži podatke u fajl, drugi sistem čita zabeležene podatke. Potrebno je postići dogovor oko naziva fajla, njegovog formata i lokaciji, vremenu kada će biti pisan i čitan i na kraju ko će vršiti uklanjanje istog.
- Deljenje baze podataka – potrebno je postići dogovor oko šeme baze podataka
- Udaljenje procedure – sistem izlaže procedure kojima je moguće pristupiti iz druge aplikacije (*web servisi*)
- Razmena poruka – informacije se dele upotrebom sistema za razmenu poruka navođenjem komunikacionog kanala preko kojeg se vrši prenos. Potrebno je postići dogovor oko naziva komunikacionog kanala i formata poruke.

NAPOMENA:

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

2. INTEGRACIJA RAZMENOM PORUKA

Integracija razmenom poruka je najčešće korišćen tip integracije iz razloga što omogućava objavljivanje poruka i ukoliko drugi sistem nije u mogućnosti da ih čita. Iste bivaju dostupne drugom sistemu kada postane spreman. Komunikacija je nezavisna od platforme i programskog jezika u kom su sistemi pisani, dok asinhrona razmena omogućava da sistem koji proizvodi poruke ne mora čekati na odgovor primaoca, već može neometano nastaviti sa radom i proizvodnjom narednih poruka.

2.1. Komunikacioni kanali

Nakon što se grupi aplikacija stavi na raspolaganje sistem za razmenu poruka, potrebno je definisati komunikacione kanale koji će biti korišćeni za razmenu poruka. Kao što je već rečeno potrebno je postići dogovor o njihovom nazivu. Pored naziva potrebno je definisati i njegov tip.

Komunikacija tačka-tačka omogućava da tačno jedan primalac konzumira jednu poruku iz kanala. Ukoliko je na kanalu prisutno više od jednog primaoca samo jednom od njih će biti dostavljena poruka.

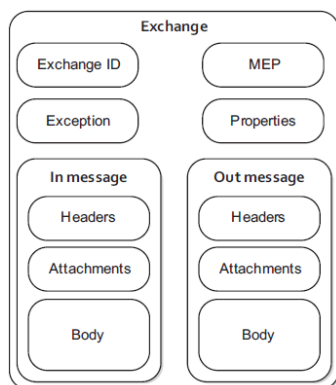
Kod *publish-subscribe* tipa komunikacije naziv kanala predstavlja temu na koju se zainteresovane strane moraju prijaviti radi primanja poruka. Prilikom slanja poruke na određenu temu sve zainteresovane strane dobijaju svoju kopiju poruke.

2.2. Poruka

Poruka predstavlja nedeljiv skup informacija koje je potrebno preneti od pošiljaoca do primaoca. Sastoje se iz dva osnovna dela [2]:

- Zaglavlje - služi da opiše njeno poreklo, tip i destinaciju...
- Telo – sam sadržaj koji se prenosi

Postoje različiti tipovi poruka kao što su komandne poruke koje pošiljalac šalje kada ima potrebu da inicira funkcionalnost na strani primaoca. Poruke sa dokumentom primaocima samo prosleđuju sadržaj. Poruke sa događajem obaveštavaju primaoca o nastalim promenama. Zahtev-odgovor poruke se koriste ukoliko je potrebno da pošiljalac na svaki svoj zahtev dobije određen odgovor. U *Apache Camel* okruženju svi navedeni tipovi poruka mogu se predstaviti instancom *Exchange* klase. Ulazna poruka je obavezna za sve tipove poruka i predstavlja primljenu poruku sa komunikacionog kanala. Polje *MEP* označava o kom tipu komunikacije se radi i sadrži jednu od vrednosti *InOnly* i *InOut*. Izlazna poruka se koristi ukoliko je potrebno poslati odgovor za primljenu poruku i u tom slučaju *MEP* ima vrednost *InOut*.



Slika 1. Sadržaj poruke u *Apache Camel* okruženju [3]

Identifikator razmene jednoznačno određuje poruku, ukoliko nije unapred naveden on biva automatski generisan. *Exception* polje nosi sa sobom informaciju o grešci ukoliko je do nje došlo prilikom obrade. Zaglavlje poruka predstavlja mapu vrednosti gde je ključ tipa *String*, a sama vrednost je tipa *Object*. Ovo znači da *Camel* ne zahteva određen tip za elemente zaglavlja. *Properties* polje je slično kao i zaglavlje, s tom razlikom što postoji dok god postoji objekat razmene. Poruka takođe može posedovati i priloge koji se uglavnom koriste za *web* servise i email komponente. Telo poruke je tipa *Object*, što korisnicima daje slobodu za tip poruke koji će biti prenet od pošiljaoca ka primaocu.

2.3. Filteri i cevi

U većini poslovnih scenarija jedan događaj može inicirati niz akcija koje je potrebno izvršiti s ciljem uspešne obrade. Korišćenje cevi i filtera omogućava podelu složene poslovne logike na manje celine koje mogu biti ponovo iskorišćene. Celine moraju biti nezavisne i izvršavaju se u redosledu navođenja. Akcije su međusobno povezane cevima.

Svaka akcija mora da implementira isti interfejs čime se garantuje da će čitati poruke sa ulazne cevi, obraditi ih na proizvoljno definisan način i postaviti ih na izlaznu cev.

```
public class MyProcessor implements Processor {
    public void process(Exchange exchange) throws Exception {
        // do something...
    }
}
```

Slika 2. Filter u *Apache Camel* okruženju [4]

Za definisanje akcija u *Apache Camel* okruženju potrebno je implementirati *Process* interfejs koji sadrži jednu metodu čiji je argument referenca na instancu *Exchange* klase i nema povratni tip. Dakle sve izmene se vrši nad prosleđenim objektom razmene. Samim tim što svaka od akcija implementira isti interfejs moguće je zameniti redosled izvršavanja ukoliko je potrebno.

2.4. Rutiranje poruka

Rutiranje poruka se bavi problemom kako proslediti poruku jednom od mogućih filtera u zavisnosti od definisanog uslova. Razlika između filtera i rutera je u tome što se izlaz rutera povezuje na nekoliko izlaznih kanala. Zahvaljujući arhitekturi filtera i cevi komponente koje okružuju ruter poruka nisu o njemu svesne. Sam

ruter nema pravo da menja sadržaj poruke, njegov zadatak je samo prosleđivanje poruke na jedan od mogućih izlaza. Jedan od najčešće korišćenih rutera je ruter na osnovu sadržaja. Ruter prihvata ulaznu poruku na osnovu čijeg sadržaja utvrđuje kojem od postojećih filtera se prosleđuje poruka, odnosno na koju izlaznu cev se ona postavlja.

```
from("seda:a")
    .choice()
        .when(header("foo").isEqualTo("bar"))
            .to("seda:b")
        .when(header("foo").isEqualTo("cheese"))
            .to("seda:c")
        .otherwise()
            .to("seda:d");
```

Slika 3. Rutiranje na osnovu sadržaja poruke [4]

Filter poruka je specijalni tip rutera, čiji zadatak je da eliminiše one poruke za čiji sadržaj sistem nije zainteresovan. Sadrži samo jednu izlaznu cev u koju se smeštaju poruke koje zadovoljavaju kriterijum zadat filterom. Poruke koje nisu zadovoljile kriterijum bivaju odbačene.

```
from("seda:a")
    .filter(header("foo").isEqualTo("bar"))
        .to("seda:b");
```

Slika 4. Filter poruka [4]

Dinamički ruter obezbeđuje nezavisnost samog sistema od uslova rutiranja, ovo znači da kod samog sistema neće u sebi sadržati uslove rutiranja. Za postizanje ove funkcionalnosti potrebno je da ruter poseduje kolekciju koja će sadržati destinacije kao i definisane uslove za rutiranje. Zainteresovani sistemi da bi primali poruke moraju specijalnim komunikacionim kanalom dostaviti uslov rutiranja.

U pojedinim slučajevima postoji potreba da se jedna poruka dostavi do više primaoca. Prvo je potrebno utvrditi listu primaoca kojima se sve poruka prosleđuje. Najčešće se lista utvrđuje dinamički odnosno predstavlja rezultat neke obrade, ali može biti i statička. Drugi korak predstavlja samo postavljanje u komunikacioni kanal svakog od primaoca. Lista primaoca ne menja sadržaj poruke.

```
from("seda:a")
    .multicast().to("seda:b", "seda:c", "seda:d");
```

Slika 5. Statička lista primaoca [4]

Dinamička lista primaoca koristi kolekciju u kojoj se čuvaju informacije o destinaciji i uslovi za primanje poruka. Svaki od zainteresovanih primaoca putem kontrolnog kanala dostavlja željeni uslov.

```
from("direct:a").recipientList(
    header("recipientListHeader").tokenize(","));
```

Slika 6. Dinamička lista primaoca [4]

Mnoge poruke koje prolaze kroz integracione sisteme sačinjene su od nekoliko elemenata. Primer je porudžbenica koja sadrži stavke. Za svaku od stavki potrebno je izvršiti proveru dostupnosti gde same stavke ne moraju biti sadržane unutar istog sistema. Svaki od sistema treba da primi samo onu stavku za koju treba da odradi proveru.

Za obradu ovakvih poruka moguće je razdvojiti glavnu poruku u niz manjih poruka (po stavki porudžbenice). Dakle poruke koje poseduju ponavljajuće elemente mogu biti podeljene u niz manjih poruka gde će svaka poruka sadržati jedan element iz niza.

```
from("seda:a")
    .split(body(String.class).tokenize("\n"))
    .to("seda:b");
```

Slika 7. Podela ulazne poruke [4]

Poruke koje su u određenoj zavisnosti ponekad je potrebno grupisati u jednu poruku koja se zatim postavlja na izlazni kanal. Dakle, od jedne ili više ulaznih poruka potrebno je napraviti jednu izlaznu poruku. Predstavlja suprotnu operaciju razdvajanja. Problem predstavlja utvrđivanje tačnog broja poruka koje je potrebno spojiti. Za uspešno grupisanje poruka koriste se tri osnovne komponente:

- Identifikatora zavisnosti – jedinstveno identifikuje poruke koje su u međusobnoj zavisnosti
- Stanje kompletnosti – kada je potrebno objaviti poruku na osnovu prikupljenih podataka
- Algoritam sakupljanja – opisuje način spajanja poruka

2.6. Transformacija poruka

Poruke koje se razmenjuju između različitih sistema često nisu u željenom formatu za svaki od sistema, prilikom čega je potrebno obezbediti mehanizme transformacije s ciljem prilagođavanja formata.

Zahtevanjem svih sistema, koji komunikaciju obavljaju preko sistema za razmenu poruka, da koriste isti format poruke vrši se „vezivanje“ jednog sistema za drugi čime je jedna od glavnih odlika integracije narušena. Samim tim upotreba istog formata poruke od svih sistema onemogućava svakom od sistema da vrši nezavisne izmene.

U slučaju da sistem koji prima poruke zahteva više informacija nego što se nalazi u samoj poruci moguće je upotrebiti mehanizme za proširivanje sadržaja. Potreban sadržaj moguće je dobiti izračunavanjem, gde se nedostajeći sadržaj izračunava na osnovu postojećih informacija.

Nedostajeći sadržaj je moguće dobiti iz okruženja, kao npr. sistemsko vreme. Najčešće korišćeni pristup predstavlja dobavljanje informacija od drugih sistema, npr. baza podataka, datoteka, ili čak manuelni unos.

Potrebno da se sadržaj poruke umanji, odnosno da se pojedina polja uklone vrši se filterom sadržaja. Ovaj mehanizam uveden je prevashodno kako bi se što efikasnije iskoristila mreža, kao i iz bezbednosnih razloga gde pojedini sistemi nemaju prava pristupa poverljivim informacijama.

U određenim scenarijima neophodno je samo privremeno ukloniti određeni sadržaj iz poruke. Npr. primljena poruka poseduje polja koja nisu neophodna svim komponentama sistema za uspešnu obradu ali određenom malom broju jesu.

Kako bi se smanjila veličina poruke i povećala efikasnost sistema, pojedina polja se privremeno čuvaju u trajnom

skladištu i uklanjaju iz same poruke. Komponenta kojoj je potreban uklonjen sadržaj vraćanje istog vrši proširivanjem sadržaja.

2.7. Krajnje tačke poruka

Ovom grupom šablona rešava se problem povezivanja sistema sa samim komunikacionim kanalima. Sistem za razmenu poruka i sistemi koji komuniciraju razmenom poruka su zasebni sistemi. Jedan sistem rešava domenski problem korisnika, dok drugi brine o komunikacionim kanalima kojima se razmenjuju poruke kao i o samoj razmeni. Iz razloga što su to nezavisni sistemi potrebno je obezbediti način na koji će se uspostaviti njihova komunikacija. Šablon definiše potreban API kako bi same korisničke aplikacije izbegle implementaciju dodatne logike potrebne da se izvrši uspešna konekcija sa sistemom za razmenu poruka.

Ukoliko sistem ima potrebu da kontroliše trenutak u kom će primiti sledeću poruku na raspolaganju mu je primalac baziran na prozivanju. Pošiljalac dostavlja poruke proizvoljnom učestalošću, dok primalac diktira tempo njihovog preuzimanja. Prisutnost poruka se utvrđuje proverom komunikacionog kanala, ukoliko se utvrdi prisutnost poruke ista biva prosleđena primaocu. U suprotnom se nastavlja provera. Ovaj proces se naziva prozivanjem. Dakle primalac proziva sledeću poruku kada je spreman da je obradi.

```
Endpoint endpoint = context.getEndpoint("activemq:my.queue");
PollingConsumer consumer = endpoint.createPollingConsumer();
Exchange exchange = consumer.receive();
```

Slika 8. Prozivanje poruka [4]

Za sisteme koji zahtevaju da poruka bude obrađena čim je pre moguće, postoje primaoci bazirani na događajima. Umesto učestale provere na pristigle poruke, komunikacionom kanalu je omogućeno da objavi događaj o pristizanju poruke. Dakle, umesto da primalac vrši konstantnu proveru novih poruka, komunikacioni kanal prosleđuje poruku primaocu čim je ona postavljena.

```
from("file://local/router/messages/foo")
    .to("jms:queue:foo");
```

Slika 9. Primaoc baziran na događaju [4]

Sistem iz komunikacionog kanala ponekad ne želi da preuzme sve postavljene poruke, potrebno je omogućiti primaocu način na koji će naglasiti za kakve poruke je zainteresovan. Za jedan komunikacioni kanal koji osluškuje jedan korisnički sistem podrazumevano je preuzimanje svih postavljenih poruka.

Poželjno je da za svaki tip poruke postoji zaseban kanal kojim će se vršiti njen prenos do sistema primaoca. Ovim pristupom se sistema primaoca lišava pisanje logike koja određuje kojim tokom će poruka biti prosleđena. Kada je broj tipova koji se razmenjuju prevelik ovo nije moguće postići.

Dakle, više različitih tipova poruka moguće je postaviti na isti komunikacioni kanal. U ovom slučaju potrebno je koristi šablon otparnika poruka koji će za svaku pristiglu poruku utvrditi tok kojim će ona biti obrađena.

3. ZAKLJUČAK

Sve veći broj sistema vrši razmenu poruka kao način obavljanja komunikacije, uzrokujući stručna lica da usavršavaju postojeća znanja i šablone kao i definisanje novih. Vremenom se na svakom od šablona vrši manji broj izmena, dok razvoj implementacije ne prestaje kako bi se krajnjim korisnicima što više olakšala njihova upotreba i omogućio što brži razvoj sistema. Integracija sistema je veoma važna zbog postojanja kompleksnih sistema čija se rešenja mogu nalaziti na različitim platformama i u različitim tehnologijama.

Šabloni predstavljaju odličnu pomoć obezbeđujući fleksibilnu strukturu rešenja. Oblast integracije je veoma opširna i ovim radom nisu obuhvaćeni svi njeni aspekti i šabloni.

4. LITERATURA

- [1] Gregor Hohpe, Bobby Woolf, "Enterprise Integration Patterns", 2004.
- [2] Li Da Xu, "*Enterprise Integration and Information Architecture*", 2011.
- [3] Claus Ibsen, Jonathan Anstey, "*Camel in Action*", 2011.
- [4] <http://camel.apache.org/manual/camel-manual-2.24.0.html> (pristupljeno u maju 2019.)

Kratka biografija:

Marko Kljajić rođen je u Bačkoj Topoli 1994. god. Odbranio je diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektro-tehnike i računarstva na temu „Implementacija bezbednosnih mehanizama u sistemu platnog prometa“ u septembru 2017. godine.

kontakt: kljajic77@gmail.com

PROJEKTOVANJE OPTIČKE MREŽE U NASELJU SREDNJE VELIČINE OPTICAL NETWORK DESIGN FOR MEDIUM SIZED CITY

Aleksandra Kerezović, Željen Trpovski, Dejan Nemec, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Rad se bavi pasivnim optičkim mrežama i njihovim podelama. Posebno se razmatra Gigabit pasivna optička mreža (GPON) i njene prednosti. Pored toga rad se bavi izgradnjom pristupne optičke mreže u Rumi, gde se razmatra polaganje optičkog kabla i povezivanje miniIPAN/IPAN uređaja u jednom delu naselja.

Ključne reči: Optika, optičko vlakno, FTTx, GPON, miniIPAN/IPAN

Abstract – The paper deals with passive optical networks and their types. The Gigabit Passive Optical Network (GPON) and its benefits are specifically considered. In addition, the paper deals with the construction of an optical access network in Ruma, where the construction of an optical cable and the connection of miniIPAN / IPAN devices in the part of city are discussed.

Keywords: Optics, fiber, FTTx, GPON, passive optical networks, miniIPAN/IPAN devices

1. UVOD

U ovom radu opisan je postupak projektovanja optičke mreže u naselju srednje veličine. Dat je konkretan primer izgradnje trase optičkog kabla kroz kratku ulicu u Rumi. Prikazan je osnovni princip arhitekture, topologije i realizacije FTTx (*Fiber To The x*) mreža. Opisane su pasivne optičke arhitekture i princip planiranja pristupne mreže, GPON (*Gigabit Passive Optical Network*), kao i povezivanje miniIPAN i IPAN (*Internet Protocol Access Node*) uređaja. Na kraju su dati podaci o pokrivenosti domaćinstava kao „home passed“ ili „home connected“.

2. FTTX (*Fiber To The X*) KONCEPTI I APLIKACIJE

Fiber to the X (FTTx) je opšti naziv za bilo koju širokopojasnu mrežnu arhitekturu koja koristi optička vlakna kao zamenu bakarnih lokalnih petlji koje se koriste za telekomunikacije u poslednjem kilometru mreže, do korisnika.

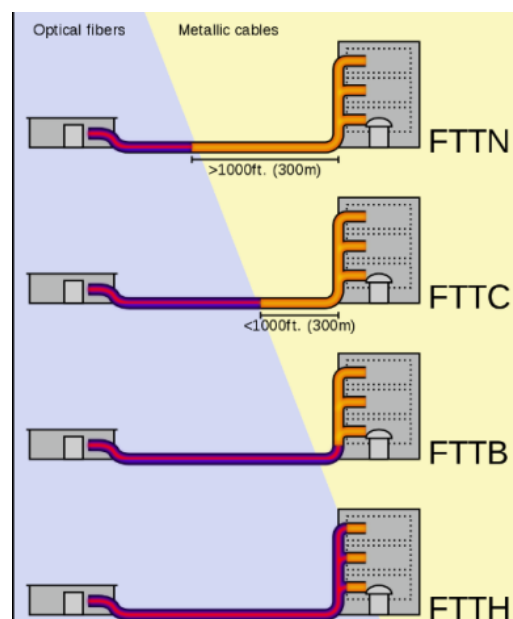
Generički termin je bio prvobitna generalizacija nekoliko različitih konfiguracija optičke realizacije (FTTN, FTTC, FTTB, FTTH itd.)

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Željen Trpovski.

Početna tri slova FTT uvek su bila ista, a konfiguracije su se razlikovale po poslednjem slovu, koje se menjalo u zavisnosti od generacije sistema. Neki primeri tih konfiguracija prikazani su na slici 1:

- FTTN (*Fiber To The Node*) – optika do izdvojenog stepena.
- FTTC (*Fiber To The Curb/Cabinet*) – optika do telefonskog stuba ili kabineta/ormana.
- FTTB (*Fiber To The Building/Basement*) – optika do objekta koji se povezuje na mrežu.
- FTTH (*Fiber To The Home*) – optika do kuće.
- FTTD (*Fiber To The Desk*) – optika do stola/računara.
- FTTP (*Fiber To The Premises*) – optika do prostorije.
- FTU (*Fiber To The User*) – optika do korisnika [1].



Slika 1. Varijante FTTx arhitekture [1]

3. PASIVNE OPTIČKE MREŽE (PON/GPON)

Za razliku od konvencionalnih mreža, pasivne optičke mreže PON (*Passive Optical Network*) nemaju aktivnih komponenta između glavnog komutacionog centra i korisničkih prostorija. Koriste se samo potpuno pasivne optičke komponente, kako bi vodile informacione signale sadržane u pojedinim talasnim dužinama.

Pasivne optičke mreže (PON) postale su alternativa u rešavanju problema optičkog pristupa do krajnjeg

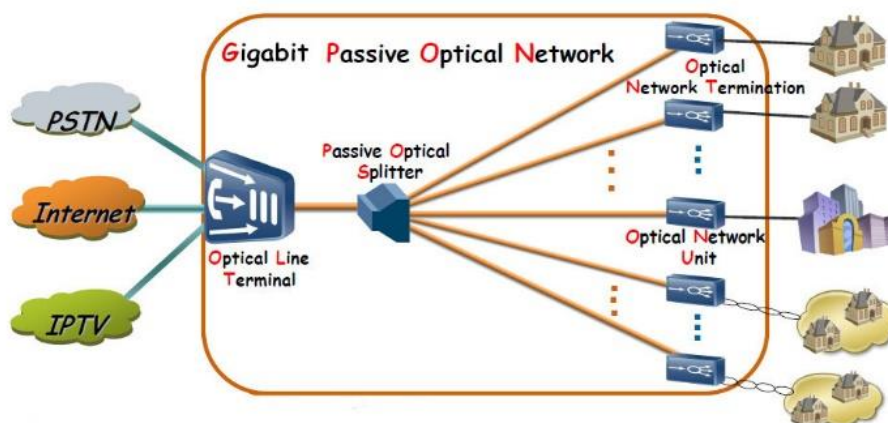
Mogućnost sabtendovanja IPAN uređaja, kao i potencijalna izgradnja novih agregacionih čvorišta bliže korisnicima, ne treba da budu restriktivni kriterijum prilikom projektovanja optičke mreže. U zavisnosti od topologije mreže, može se predvideti sabtend do dva pristupna čvora na određeni IPAN/miniIPAN uređaj. Odabir AN za sabtendovanje treba vršiti po pre svega prostornim kriterijumima područja, vodeći pri tome računa o dužini i optimalnom korišćenju vlakana raspoložive optičke mreže [3].

5. miniIPAN DRVARSKA („home passed“ ili „home connected“)

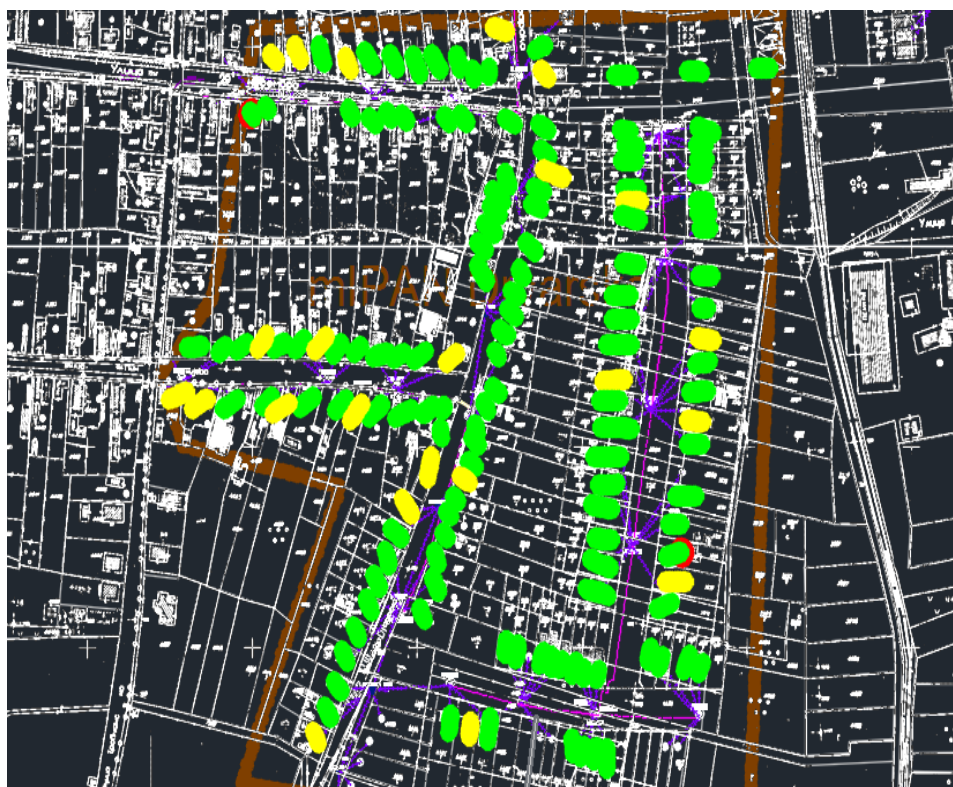
Kao što je već rečeno, široka implementacija GPON rešenja treba da se oslanja na „home passed“ model. To znači da se okosnica GPON mreže planira tako da zadovolji kompletno područje, a sami pojedinačni elementi GPON mreže lociraju i dimenzionišu na osnovu

procenjenog broja korisnika za konekciju u prvoj fazi. Povezani korisnici će biti „home connected“, dok će oni kod kojih je povezivanje na GPON mrežu omogućeno naknadnom izgradnjom kratkih optičkih privoda i/ili optičke instalacije i eventualnog dodavanja splitera, imati status „home passed“ korisnika.

Na slici 5. prikazan je deo grada Rume koji obuhvata privodni optički kabl miniIPAN Drvarska. Deo sa slike obuhvata četiri ulice, a to su Atanasije Stojkovića, Pinkijeva, Bore Gavrilovića i Drvarska. Za ove četiri ulice obeleženo je svako domaćinstvo koje je pokriveno kao „home passed“, žutom bojom, i svako domaćinstvo pokriveno kao „home connected“, zelenom bojom. Pokrivenost „home passed“ je 16% od ukupnog broja domaćinstava koje obuhvata miniIPAN Drvarska. „Home passed“ menja bakar za optiku i pruža korisnicima mnogo bolji signal, bolji protok, jednom rečju bolju uslugu.



Slika 4. Arhitektura GPON mreže [4]



Slika 5. Prikaz miniIPAN Drvarska [3]

6. ZAKLJUČAK

Danas je svuda zastupljena digitalna tehnika i optički prenos podataka što je uticalo na formiranje novih tehnika prenosa, sa kojim se započela nova faza u evoluciji telekomunikacionih mreža. Ovaj napredak će dovesti do značajnih mogućnosti pružanja novih servisa znatno većih binarnih protoka.

To dovodi do razvoja novih optičkih mreža koje već sada omogućavaju prenos ogromne količine podataka daleko većim brzinama, nego što je to bio slučaj pre deset godina.

Optičke mreže, bazirane na pojavljivanju optičkog sloja u transportnim mrežama, obezbeđuju veće kapacitete i smanjuju troškove novim aplikacijama kao što su internet, video i multimedijalna interakcija i moderni digitalni servisi [5].

Osnovni cilj razvoja savremenih telekomunikacionih mreža jeste skraćivanje pretplatničke petlje i približavanje širokopojasne mreže korisnicima usluga. Na području Rume u ulici Drvarska izvršena je izgradnja miniIPAN Drvarska i obezbeđeni su potrebni brojevi telefonskih priključaka za korisnike kojima su priključci bili neophodni. Izgradnjom ovog optičkog kabla "Telekom Srbija" dobija nove pretplatnike, korisnike usluga, a samim tim i bolju zaradu.

7. LITERATURA

[1] <https://networks.nokia.com/products/7330-isam-fttn-ansi>, Februar 2019.

[2]

https://en.wikipedia.org/wiki/Passive_optical_network, Februar 2019.

[3] Telekom Srbija, primeri projektne dokumentacije za izgradnju optičkih mreža, 2014.

[4] <https://www.fs.com/overview-of-gpon-technology-aid-500.html>, Februar 2019.

[5] Nemanja Andrić, Optičko kabliranje, Fakultet tehničkih nauka, Čačak, 2016.

Kratka biografija:



Aleksandra Kerezović rođena je u Rumi 1992. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Komunikacione tehnologije – Telakomunikacioni sistemi odbranila je 2019. god.



Željko Trpovski rođen je u Rijeci 1957. godine. Doktorirao je na Fakultetu tehničkih nauka 1998. god. Oblast interesovanja su telekomunikacije i obrada signala.



Dejan Nemec rođen je 1972. god. Diplomirao, specijalizirao i magistrirao je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva. Oblast interesovanja su telekomunikacije i obrada signala.

TEHNOLOGIJE I SOFTVER U 3D ŠTAMPI SA PRIMEROM PRIMENE U PČELARSTVU
TECHNOLOGIES AND SOFTWARE IN 3D PRINTING WITH EXAMPLE OF USAGE IN
APICULTUREStojan Mitrić, *Fakultet tehničkih nauka, Novi Sad***Field – ELECTRICAL ENGINEERING AND COMPUTING**

Abstract – *Main contribution of this paper is comparison of 3D printing technologies with detailed description of their work processes, usage of technologies in different applications and their advantages and disadvantages. This paper also describes software that are most commonly used in 3D printing world, and example of usage of 3D printing in Apiculture for creating system of rollers for making beeswax foundations.*

Keywords: *3D printing, 3D modeling, 3D software, CAD*

Kratak sadržaj – *MGlavni doprinos ovog rada jeste poređenje 3D tehnologija štampe, sa detaljnim opisom radnog postupka, primenen, prednosti i nedostataka. Rad opisuje najčešće korišćen softver u oblasti 3D štampe, kao i primere primene 3D štampe u pčelarstvu, za izradu sistema valjaka za izradu osnova košnica.*

Ključne reči: *3D štampa, 3D modelovanje, 3D softver, CAD*

1. INTRODUCTION

3D printing is a category of additive manufacturing in which from digital 3D model real 3D object is created by the method of joining thin layers of materials that represent horizontal cross-sections of the model being printed.

Additive manufacturing is a transformative approach to industrial production that enables the creation of lighter, more powerful parts and systems. It represents another technological advancement that is made possible by switching from analogue to digital processes [1].

Technology was created in the 80s of the 20th century and was used primarily for the production of prototypes. Since then, 3D printing has begun to be used as a technique for creating products in almost all industries. Lately, 3D printers and their ability to quickly make complicated prototypes find their place in many areas - in science, technology, medicine, or entertainment.

There are printers that can print from materials such as sugar, plastic, metal, nylon, and even human cells. As technology progresses, 3D printers cease to be expensive exotic technology and move into everyday life of average consumers [2].

NOTE:

This paper came from a MSc thesis under supervision of Dr. Dragan Dinu, assistant professor.

2. 3D PRINTING TECHNOLOGIES

American scientist Chuck Hal printed the first three-dimensional object in 1986. It was a plastic cup made with a stereolithography technique, curing a liquid plastic resin under UV light. Hal patented this technique and founded 3D Systems, which is still one of the leading companies in the field.

The company also developed the first file format for 3D printing, STL (from the term stereolithography). On the other hand, in the late eighties, Scott Kramp set the basis for modeling by joining coatings or FDM (Fused Deposition Modeling).

Kramp patented this technique and founded Stratasys, dedicated to the development and use of this 3D printing technique. It turned out to be one of the most important techniques for the expansion of 3D printing in the general public.

Since 2010, the American Society for Testing and Materials, the group "ASTM F42 - Additive Production", has developed a set of standards that classify the additive production processes in several categories according to the standard terminology for additive production technologies, which are: SLA/DLP, FDM, SLS, Material Jetting/DOD, Binder Jetting, DMLS/SLM/EBM [2].

2.1. SLA

Stereolithography (SLA) is an additive manufacturing process that belongs to the Vat Photopolymerization family. In SLA, an object is created by selectively curing a polymer resin layer-by-layer using an ultraviolet (UV) laser beam. The materials used in SLA are photosensitive thermoset polymers that come in a liquid form.

SLA is famous for being the first 3D Printing technology: its inventor patented the technology back in 1986. If parts of very high accuracy or smooth surface finish are needed, SLA is the most cost-effective 3D printing technology available.

Work process:

The build platform is first positioned in the tank of liquid photopolymer, at a distance of one layer height for the surface of the liquid. Then a UV laser creates the next layer by selectively curing and solidifying the photopolymer resin. The laser beam is focused in the predetermined path using a set of mirrors, called galvos. The whole cross sectional area of the model is scanned, so the produced part is fully solid. When a layer is finished, the platform moves at a safe distance and the sweeper blade re-coats the surface.

The process then repeats until the part is complete. After printing, the part is in a green, no-fully-cured state and requires further post processing under UV light if very high mechanical and thermal properties are required.

The liquid resin is solidified through a process called photopolymerization: during solidification, the monomer carbon chains that compose the liquid resin are activated by the light of the UV laser and become solid, creating strong unbreakable bonds between each other.

The photopolymerization process is irreversible and there is no way to convert the SLA parts back to their liquid form: when heated, they will burn instead of melting. This is because the materials that are produced with SLA are made of thermoset polymers, as opposed to the thermoplastics that FDM uses [3].

2.2. SLS

Selective Laser Sintering (SLS) is an Additive Manufacturing process that belongs to the Powder Bed Fusion family. In SLS, a laser selectively sinters the particles of a polymer powder, fusing them together and building a part layer-by-layer.

The materials used in SLS are thermoplastic polymers that come in a granular form.

SLS 3D Printing is used for both prototyping of functional polymer components and for small production runs, as it offers a very high design freedom, high accuracy and produces parts with good and consistent mechanical properties, unlike FDM or SLA.

Work process:

The powder bin and the build area are first heated just below the melting temperature of the polymer and a recoating blade spreads a thin layer of powder over the build platform. CO₂ laser then scans the contour of the next layer and selectively sinters (fuses together) the particles of the polymer powder. The entire cross section of the component is scanned, so the part is built solid. When the layer is complete, the build platform moves downwards and the blade re-coats the surface.

The process then repeats until the whole part is complete. After printing, the parts are fully encapsulated in the unsintered powder and the powder bin has to cool down before the parts can be unpacked.

This can take a considerable amount of time (up to 12 hours). The parts are then cleaned with compressed air or other blasting media and are ready to use or further post process [3].

2.3. MJ

Material Jetting (MJ) is an additive manufacturing process that operates in a similar fashion to 2D printers. In material jetting, a printhead (similar to the printheads used for standard inkjet printing) dispenses droplets of a photosensitive material that solidifies under ultraviolet (UV) light, building a part layer-by-layer. The materials used in MJ are thermoset photopolymers (acrylics) that come in a liquid form.

MJ 3D Printing creates parts of high dimensional accuracy with a very smooth surface finish. Multi-material printing and a wide range of materials (such as ABS-like, rubber-like and fully transparent materials) are available in Material Jetting.

These characteristics make MJ a very attractive option for both visual prototypes and tooling manufacturing. A variation of the MJ process uses Drop-On-Demand (DOD) printheads to dispense viscous liquids and create wax-like parts.

Work process:

First, the liquid resin is heated to 30-60°C to achieve optimal viscosity for printing. Then the printhead travels over the build platform and hundreds of tiny droplets of photopolymer are jetted/deposited to the desired locations. UV light source that is attached to the printhead cures the deposited material, solidifying it and creating the first layer of the part. After the layer is complete, the build platform moves downwards one layer height and the process repeats until the whole part is complete [3].

2.4. BJ

In Binder Jetting (BJ) a binder is selectively deposited onto the powder bed, bonding these areas together to form a solid part one layer at a time. The materials commonly used in Binder Jetting are metals, sand, and ceramics that come in a granular form. Binder Jetting is used in various applications, including the fabrication of full-color prototypes, the production of large sand casting cores and molds and the manufacture of low-cost 3D printed metal parts.

Work process:

First, a recoating blade spreads a thin layer of powder over the build platform. Then, a carriage with inkjet nozzles (which are similar to the nozzles used in desktop 2D printers) passes over the bed, selectively depositing droplets of a binding agent (glue) that bond the powder particles together. In full-color Binder Jetting, the colored ink is also deposited during this step. The size of each drop is approximately 80 µm in diameter, so good resolution can be achieved.

When the layer is complete, the build platform moves downwards and the blade re-coats the surface. The process then repeats until the whole part is complete. After printing, the part is encapsulated in the powder and is left to cure and gain strength. Then the part is removed from the powder bin and the unbound, excess powder is cleaned via pressurized air [3].

2.5. SLM/DMLS

Selective Laser Melting (SLM) and Direct Metal Laser Sintering (DMLS) are two metal additive manufacturing processes that belong to the powder bed fusion 3D printing family. The two technologies have a lot of similarities: both use a laser to scan and selectively fuse (or melt) the metal powder particles, bonding them together and building a part layer-by-layer. Also, the materials used in both processes are metals that come in a granular form.

The differences between SLM and DMLS come down to the fundamentals of the particle bonding process (and also patents): SLM uses metal powders with a single melting temperature and fully melts the particles, while in DMLS the powder is composed of materials with variable melting points that fuse on a molecular level at elevated temperatures.

Work process:

The build chamber is first filled with inert gas (e.g. argon) to minimize the oxidation of the metal powder and then it is heated to the optimal build temperature. A thin layer of metal powder is spread over the build platform and a high power laser scans the cross-section of the component, melting (or fusing) the metal particles together and creating the next layer. The entire area of the model is scanned, so the part is built fully solid. When the scanning process is complete, the build platform moves downwards by one layer thickness and the recoater spreads another thin layer of metal powder. The process is repeated until the whole part is complete [3].

2.6. FDM

Fused Deposition Modeling (FDM), or Fused Filament Fabrication (FFF), is an additive manufacturing process that belongs to the material extrusion family. In FDM, an object is built by selectively depositing melted material in a pre-determined path layer-by-layer. The materials used are thermoplastic polymers and come in a filament form. FDM is the most widely used 3D Printing technology: it represents the largest installed base of 3D printers globally and is often the first technology people are exposed to. In this article, the basic principles and the key aspects of the technology are presented.

Work process:

A spool of thermoplastic filament is first loaded into the printer. Once the nozzle has reached the desired temperature, the filament is fed to the extrusion head and in the nozzle where it melts. The extrusion head is attached to a 3-axis system that allows it to move in the X, Y and Z directions. The melted material is extruded in thin strands and is deposited layer-by-layer in predetermined locations, where it cools and solidifies. Sometimes the cooling of the material is accelerated through the use of cooling fans attached on the extrusion head. To fill an area, multiple passes are required (similar to coloring a rectangle with a marker). When a layer is finished, the build platform moves down (or in other machine setups, the extrusion head moves up) and a new layer is deposited. This process is repeated until the part is complete [3].

3. SOFTWARE

In the world of 3D printing, CAD software are used mostly for creating 3D printing models such as OpenScad, Fusion 360, Simplify 3D, and software that generate 3D-readable G-code from 3D models, of which the most popular is Ultimaker Cura [6]. Cura prepares a 3D model for printing. It is possible to print according to the recommended mode or customized mode in which more than 300 settings for maximum control can be configured. It has support in the form of CAD software add-ons and optimized material profiles. Every model that is designed for printing must be translated into the G code, or in the instructions that the 3D printer understands. Cura does this by dividing the model vertically into thin layers and creating a ready-to-print file [4]. Prior to the printing process, the Cura offers an

animation option for the 3D model to look after printing and the ability to view the models in layers of printing. Image 1 represents a overview of the 3D model by layers in Cura.

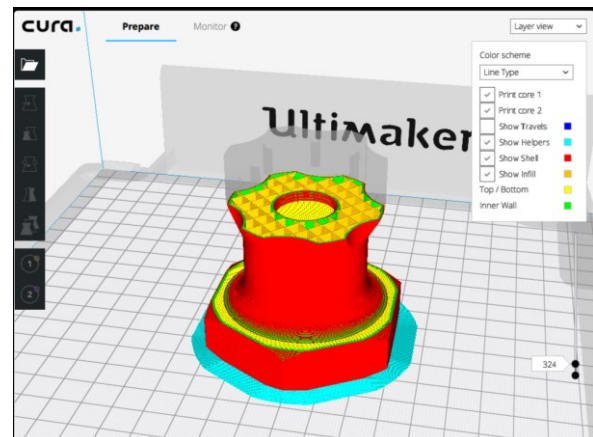


Image 1. 3D model view by layers in Cura

4. EXAMPLE OF USAGE IN APICULTURE

Traditional manufacturing of system of rollers for making beeswax foundations requires usage of expensive machines such as CNC machine. This process requires a lot of precision and time to build, and the amount of waste after CNC milling can be high. 3D printing can do this process with much less time to build, nearly the same precision and without any waste. Image 2 shows rollers for making beeswax foundations printed on FDM 3D printer.



Image 2. Printed rollers on FDM 3D printer

5. COMPARISON OF TECHNOLOGIES

3D printers use different techniques for printing 3D models. While some use a thermoplastic filament, others use a conventional light source or a laser source to harden the molecules of the material to print the model. Some technologies use materials based on plastic polymer for printing while other technologies use metals and alloys of metals. Comparison of 3D printing technology of plastic materials is given in table 1.

Table 1. Comparison of 3D printing technology of plastic materials

	SLA	SLS	MJ	FDM
<i>Usage</i>	<i>Ideal for full 3D printing. Best waterproof waterproofing models. Use in the production of jewelry making patterns, use in dentistry [3,5].</i>	<i>Production of functional mechanical models with complex details. Production of temperature and chemically resistant models [5].</i>	<i>The most common use in 3D prototype prints, which requires the use of multiple colors and multiple different materials.</i>	<i>Production of prototypes and making decorative objects, molds and templates for casting concrete or silicone.</i>
<i>Advantages</i>	<i>The technology that first appeared in 3D printing. Smooth finishing surface.</i>	<i>Does not require a support structure, the ability to print more robust functional models with excellent mechanical properties. A wide range of colors [3].</i>	<i>Multiple color printing, very detailed print of small parts of the model. Homogeneous mechanical and thermal properties of materials [3].</i>	<i>Good finishing surface, wide range of colors and the ability to print one model from several different materials. Recycling and reuse of filaments. the minimum amount of waste after printing [3].</i>
<i>Disadvantages</i>	<i>Prints are fragile, not intended for the production of mechanical parts. It is impossible to recycle material after printing [3].</i>	<i>The porous, granular finish of the model, printing is more expensive compared to the other technologies [3]</i>	<i>MJ prints have poor mechanical properties, are easily fragile, materials are photosensitive, degrade over time [3].</i>	<i>ABS and PLA presses are fragile. Very strong filaments such as Nylon or Ultem material [3].</i>

6. CONCLUSION

SLA is the most popular 3D printing technology. Today it is widely used because it provides accuracy along with excellent finishing surface.

It also allows printing of objects using various materials. However, it can be quite expensive and is mainly used in the prototype manufacturing industry.

DLP is very similar to SLA. The main difference between these two technologies is the type of light source they use.

SLS is a good choice for building objects with complex shapes and geometries. No additional structural support is required as for other 3D printing technologies, and no further processing is required. The final object may not be very precise and the finish surface is not quite smooth.

MJ offers a wide range of possibilities in terms of 3D printing, which consists of different colors and materials, which is a key advantage of this technology compared to others. It also supports two different 3D printing settings regarding the final surface of the printed model, matte and shine, which can be useful in emphasizing the aesthetic details of the model.

FDM is the cheapest process compared to all other 3D printing methods. This makes it suitable for home use. The level of accuracy and finishing of the surface is good. Materials used by FDM are limited mainly to thermoplastics [5].

7. REFERENCES

- [1] Additive Manufacturing Technologies, Rapid Prototyping to Direct Digital Manufacturing, Ian Gibson, David W. Rosen, Brent Stucker, Springer, 2014
- [2] Svet kompjutera, Dragan Kosovac, 3D štampači, <https://www.sk.rs/2013/05/skpr01.html>, (last time accessed 03.05.2019.)
- [3] 3D Hubs, Alkaios Bournias Varotsis, Knowledge Base, Manufacturing Processes Explained <https://www.3dhubs.com/knowledge-base/>, (last time accessed 03.05.2019.)
- [4] Ultimaker, Ultimaker Cura software, <https://ultimaker.com/en/products/ultimaker-cura-software> (last time accessed 03.05.2019.)
- [5] Comparison of Different Types of 3D Printing Technologies, Shiwpursad Jasveer, Xue Jianbin International Journal of Scientific and Research Publications, Volume 8, Issue 4, 2018
- [6] MatterHackers, Anatomy of a 3D Printer: How Does a 3D Printer Work?, <https://www.matterhackers.com/>, (last time accessed 03.05.2019.)

Short biography:



Stojan Mitrić is born in Novi Sad in 1992. He defended MSc thesis on Faculty of technology sciences in field Electrical engineering and computing in 2019. Contact: stojan007mitric@gmail.com

ПРИМЕНА СОФТ СТАРТЕРА SIEMENS 3RW4426-1BC44 У ИНДУСТРИЈСКОЈ ПУМПНОЈ СТАНИЦИ DANKHAN - MONGOLIA

SOFT STARTER APPLICATION SIEMENS 3RW4426-1BC44 IN INDUSTRIAL PUMP STATION DANKHAN - MONGOLIA

Дарко Китановић, Факултет техничких наука, Нови Сад

Област – Електротехника и рачунарство

Кратак садржај – Предмет овог рада је анализа софт старт уређаја Siemens 3RW4426-1BC44, примењеног у реализацији индустријске пумпне станице. Најпре биће детаљно приказан опис система и постављени критеријуми обухваћени пројектом. Посебан акценат биће на софт старт уређају, његовој анализи и примени. Софт старт уређај представља главни извршни елемент у управљању пумпи. Затим ће бити приказана детаљна анализа рада енергетских претварача, утицај на хармонијска изобличења, таласни облици напона и струја фазних регулатора у зависности од угла укључења и резултати практичних мерења за монофазни фазни регулатор са једностраном фазном регулацијом. На крају, у делу практичне реализације биће приказани екрани који приказују начин подешавања и параметрисања софт старт уређаја Siemens 3RW4426-1BC44.

Кључне речи: софт стартер, параметри, анализа

Abstract – The subject of this study is an analysis of the soft start of the Siemens 3RW4426-1BC44 device, implemented in the realization of an industrial pump station. First of all, a detailed description of the system and the set criteria included in the project will be presented. A special emphasis will be on the soft start device, its analysis and application. Soft start device is the main executive element in pump management. A detailed analysis of the operation of energy converters, the effect on harmonic distortion, waveforms and current of phase regulators depending on the angle of inclusion and the results of practical measurements for a single phase phase regulator with one-phase phase control will be shown. Finally, a part of the practical realization will show screens that show the way of setting and parameterizing the soft start of the Siemens 3RW4426-1BC44.

Keywords: soft starter, analysis, parameters

1. УВОД

Постоје укупно 3 пумпне станице које допремају отпадну воду до WWTP укључујући и пумпну станицу у самом постројењу за прераду воде.

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био проф. др Дарко Марчетић.

Фекалне воде из индустријског подручја у WWTP се допремају у две фазе, док се отпадне воде из старог и новог Dankhan-а допремају у једној фази из секундарне пумпне станице (слика 1).



Слика 1. Графички приказ снабдевања WWTP отпадним водама

Кроз овај пројекат су обезбеђена средства за аутоматизацију и проширење индустријске пумпне станице чиме је у многоме решен проблем отпадних вода у индустријској зони овог града.

Пројекат се изводи у две фазе, а њен завршетак пројектован је за 2040. годину када би пумпна станица радила жељеним капацитетом.

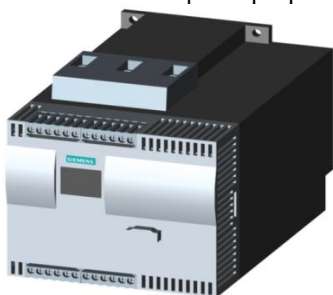
У табели 1.1 приказани су критеријуми који морају бити испуњени кроз поменуте две фазе.

Description		Waste Water PS 1 Industrial Area
Pumping Station Capacity at 2030	[l/s]	97.2 [350 m ³ /h]
Total Number of Pumps	[-]	3
Number of operating pumps	[-]	2
Number of stand-by pumps	[-]	1
Each Pump Capacity	[l/s]	75
Pumping Station Capacity at 2040	[l/s]	150 [540 m ³ /h]
Total Number of Pumps	[-]	3
Number of operating pumps	[-]	2
Number of stand-by pumps	[-]	1
Each Pump Capacity	[l/s]	75
Screening Plant		
Capacity at 2030 [Q_PS_2030]	[l/s]	97.2 [350 m ³ /h]
Capacity at 2040 [Q_PS]	[l/s]	150 [540 m ³ /h]
Total Number of Screening Units	[-]	2
Number of operating units	[-]	1
Number of stand-by units	[-]	1
Each Screening Unit Capacity	[l/s]	150
Number of containers at 2030	[-]	4
Number of containers at 2040	[-]	4

2. СОФТ СТАРТЕР Siemens 3RW44

2.1 Опис уређаја

Siemens SS 3RW44 се као алтернативно решење може искористити код мотора који захтевају рад преко упуштача звезда-троугао или фреквентног претварача. Главне предности су регулисано покретање и заустављање разних врста погона (soft starting, soft stop) и прекидачко коло које врши комутацију без штетних струјних пикова који могу утицати на квалитет мреже. Такође, додатна предност овог модела је његова компактност која долази до изражаја приликом пројектовања места монтаже унутар разводног ормана. На слици 2 приказан је изглед **Siemens 3RW4426-1BC44** софт стартера.

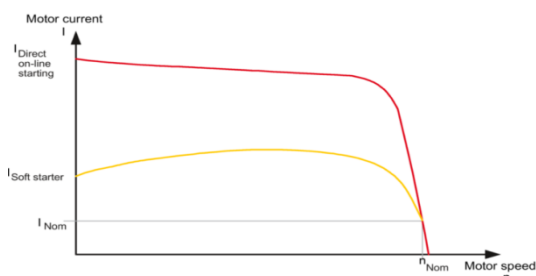


Слика 2. Графички приказ софт стартера Siemens 3RW4426-1BC44

2.2 Принцип рада

Софт стартер Siemens 3RW44 има антипаралелну везу тиристора по свакој фази, један тиристор за позитивну, а други за негативну полупериоду. Регулацијом анти-паралелне везе тиристора могуће је регулисати ефективну вредност напона на мотору од минималне до номиналне вредности.

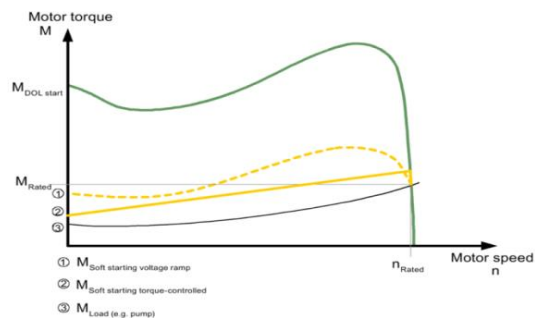
Струја се пропорционално мења са променом напона на мотору, што значи да се полазна струја смањује са смањењем напона који се примењује на мотор. Обртни момент се мења квадратно са напонам мотора, што значи да се полазни момент квадратно смањује са смањењем напона на мотору. Из ове две дефиниције закључујемо да је напон кључна величина за контролу струје и момента мотора.



Слика 3. Карактеристика полазне струје мотора управљаног са софт стартером Siemens 3RW44

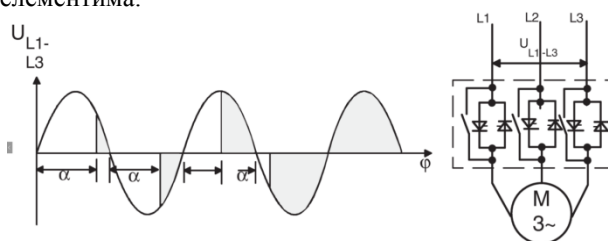
Са графика полазне струје и момента мотора можемо закључити да уколико имамо могућност контроле напона мотора, можемо вршити регулацију полазног момента и струје. Исти принцип се примењује приликом заустављања мотора, у том случају генерисани момент у мотору се постепено смањује уз помоћ smooth down-ramp на софт стартеру. Током процеса фреквенција софт стартера остаје једнака

мрежној фреквенцији што није случај код старт/стоп процеса фреквентног претварача.



Слика 4. Карактеристика полазног момента мотора управљаног са софт стартером Siemens 3RW44

Приликом исправног старта мотора, тиристори софт стартера су у потпуности искоришћени, што значи да се целокупан мрежни напон преноси на напојне терминале мотора. Уколико није потребно контролисати напон на мотору, могуће је преко by-pass контактора преспојити контакторе (слика 5) и на тај начин минимизовати топлотне губитке током процеса, насталих на тиристорима као прекидачким елементима.



Слика 5. Фазна регулација и графички приказ софт стартера Siemens 3RW44

2.3 Начини покретања (Startup mode)

Захваљујући великим могућностима, овај уређај нуди различите функције покретања. Према томе покретање мотора се може прилагодити свакој појединачној примени.

Siemens 3RW44 нуди следеће начине покретања:

1. Напонска рампа (Voltage ramp)
2. Контрола момента (Torque control)
3. Пулс одвајања у комбинацији са напонском рампом и контролом момента
4. Ограничење струје у комбинацији са напонском рампом и контролом момента
5. Директан старт (Direct start)
6. Загревање мотора (Motor heating)

1. Напонска рампа (Voltage ramp)

Најједноставнији тип софт старта се добија помоћу напонске рампе. Вредност почетне вредности напона се подешава у сету параметара софт стартера, као и одговарајуће време које је потребно да напон на мотору достигне номиналну вредност.

Време трајања старта одређује време током којег се напон мотора повећава од параметрисаног стартног напона до номиналног напона. Ово има утицај на момент убрзања мотора који покреће оптерећење. Време је потребно изабрати тако да се достигне номинална вредност напона напајања мотора.

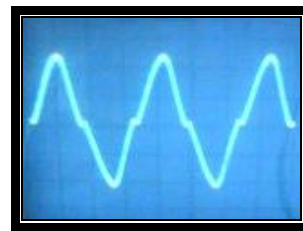
Унутар софтвера постоји опција "run-up recognition". Уколико уређај препозна да је залетање завршено, помоћу ове опције се извршава кратко спајање (премоштавање) тиристора помоћу bypass контактора. Уколико се ова опција изврши пре завршетка залетања, рампа се отказује, излазни напон на мотору се аутоматски подиже на 100% номиналног напона пре него што се изврши кратко затварање bypass контактора.



3.1 Резултати практичних мерења за моноfazни fazni regulator sa једностраном fazном регулацијом

The schematic diagram shows the electrical connections for the test stand. It includes a power source with lines L1 and N. A switch S1 is connected to L1. The 'Fazni regulator' is connected to L1 and N. An ammeter (AM 1f~) is connected in series with the regulator's output. A metering unit (MK) is connected to the N line. A digital multimeter is connected to the output of the regulator.

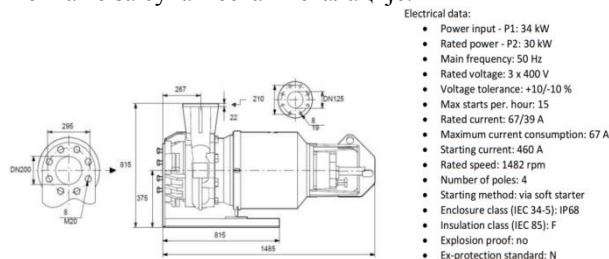
Слика 8. Таласни облик струје мреже за фазни регулатор са омско-индуктивним оптерећењем за различите углове укључења



Слика 9. Таласни облик струје фазног регулатора за омско-индуктивно оптерећење за угао укључења близак нули

4. ПРАКТИЧНА РЕАЛИЗАЦИЈА ИНДУСТРИЈСКЕ ПУМПНЕ СТАНИЦЕ СА СОФТ СТАРТЕРОМ Siemens 3RW4426-1BC44

За потребе пројекта аутоматизација индустријске пумпне станице било је потребно изградити хардверски и софтверски део. Хардверски део пројекта је обухватао израду 5 разодних ормана који заједно чине нисконапонски енергетски блок за управљање пумпном станицом. На критеријумима постављеним од стране инвеститора видели смо да постоје укупно 3 пумпе које су инсталиране у пумпној станици чије се управљање извршава преко софт старта **Siemens 3RW4426-1BC44**. Пумпе су утопне, хоризонталне монтаже за сува места инсталације.



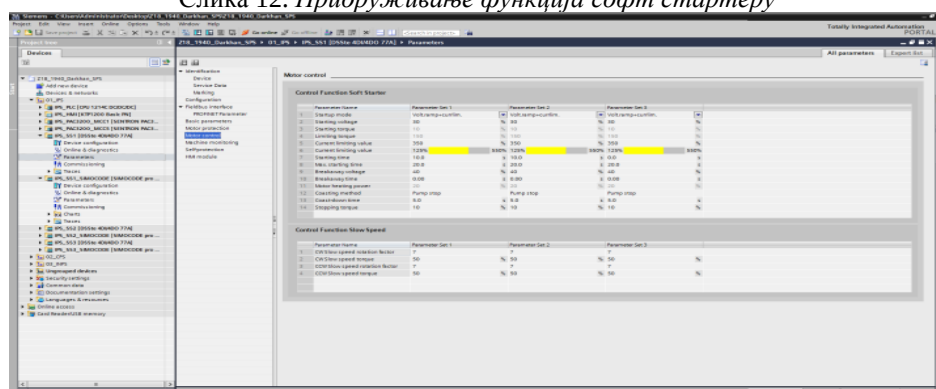
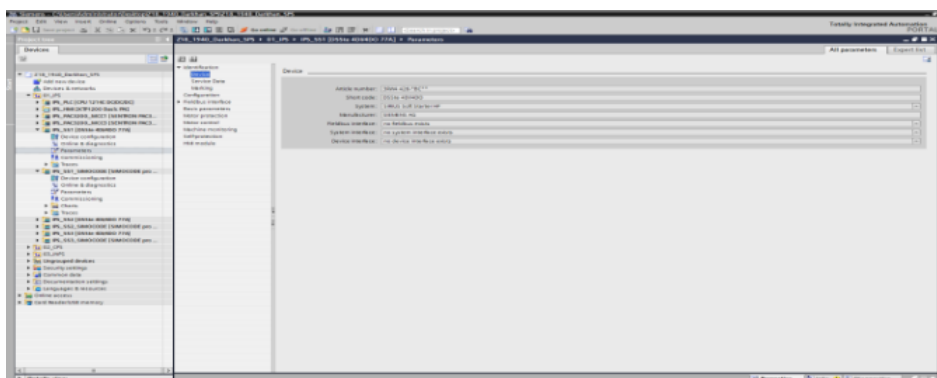
Слика 10. Графички цртеж пумпе са општим подацима

У наредном делу су приказани екрани са подешавањима уређаја који су посебно битни за тему овог мастер рада. Прво је дефинисан софтверни старт уређаја, тако што се прво додељују функције софтверног стартера, а затим и параметри дефинисани техничким подацима пумпе.

5. ЗАКЉУЧАК

Као што је већ поменуто, задатак мастер рада је била примена софтвера стартера Siemens 3RW4426-1BC44 у индустријској пумпној станици. Током израде целокупног пројекта наилазило се на велики број потешкоћа и проблема, који су решавани у ходу пројекта. Овакви пројекти управо и захтевају. Посебан акценат је стављен на софтвер стартера као главни извршни елемент у управљању пумпама. Велики број захтева је уврштен и испоштован кроз овај пројекат почевши од пројектовања хардвера преко софтвера и на крају пуштање у рад система.

Због специфичних захтева инвеститора по први пут сам имао изазов упаривања два уређаја сличних карактеристика како би се постигао јединствен резултат, а то је управљање пумпама.



Имао сам прилику да учествујем у сваком делу стварања пројекта што је за мене лично представљало велики изазов. Најзанимљивији део пројета је био пројектовање система у софтверском пакету ePlan Electric, затим део који се односио на израду електричних ормана, интеракција са радионициом, и на крају подешавање и пуштање софтверског старта у рад. Генерални утисак овог пројекта јесте да је одличан пример примене теоретског знања стеченог током школовања. Такође, указала ми се идеална прилика да кроз пројектовање система проширим знање о софтверским стартама и њихову примену у реалним системима.

6. ЛИТЕРАТУРА

- [1]Darko Marčetić, Marko Gecić, Boris Marčetić, "Programabilni logički kontroleri i komunikacioni protokoli u elektroenergetici", ISBN 978-86-7892-590-0, FTN izdavaštvo, Novi Sad, 2016.
- [2]Miloš Nedeljković, "Mrežom vođeni pretvarači", Akademska misao, Beograd, 2007.
- [3]Miloš Nedeljković, "Mrežom vođeni pretvarači", Akademska misao, Beograd, 2007.

- [4]Radoje Radetić, "Tiristorski pretvarači", Bor 2012.
- [5]Tomislav Brodić, "Osnove energetske elektronike", Školska knjiga Zagreb, 2002.
- [6]Vladislav Teodorović, "Električne pogonske mašine", Naučna knjiga Beograd, 1967.
- [7]Slobodan Vukosavić, "Električne mašine", Akademska misao Beograd, 2010.
- [8]Gojko Dotlić, "Elektroenergetika kroz standarde, zakone, pravilnike i tehničke prepouruke"
- [9]INTERNATIONAL STANDARD IEC 60909-0.

Кратка биографија:



**KOMBINATORNI ALGORITMI- PROBLEM TRGOVAČKOG PUTNIKA I
RASPOREĐIVANJE PROCESA PROCESORA****COMBINATORIAL ALGORITHMS- TRAVELLING SALESMAN PROBLEM AND
SCHEDULING CPU PROCESSES**Nevena Dovićin, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Problem trgovačkog putnika spada u veliku klasu problema koji se nazivaju kombinatorni problemi optimizacije. Ovo je jedan od najviše proučavanih problema optimizacije, a njegova rešenja i primene će biti opisane u ovom radu. S druge strane raspoređivanje procesa u procesoru je zadatak u kom se bira proces iz reda čekanja kom se dodeljuje procesor nakon toga (od strane dispečera). Postoji više različitih algoritama za biranje procesa koji će se izvršavati, od kojih će najznačajniji biti opisani.

Ključne reči: kombinatorni algoritmi, problem trgovačkog putnika, raspoređivanje procesa

Abstract – The travelling salesman problem is one of the problems called combinatorial optimization problems. This is one of the most studied optimization problems, and its solutions and applications will be described in this paper. On the other hand, scheduling a process in a processor is the task in which the process is selected from the queue to which processor is assigned after that (by dispatcher). There are several different algorithms for selecting the process to be executed, and most significant will be described.

Keywords: combinatorial algorithms, travelling salesman problem, scheduling processes

1. UVOD

Razvojem računarstva i informatike i povećanjem procesne moći računara, ljudi su počeli rešavati kompleksne probleme koji su do tada bili nerešivi. S obzirom da su danas računari ekstremno brzi, nije retka situacija da se njihovom upotrebom u sekundi mogu istražiti milioni potencijalnih rešenja, i time vrlo brzo pronaći ono najbolje.

Postoji čitav niz problema koje srećemo u svakodnevnom životu, a jedan od primera je problem trgovačkog putnika. U ovom radu biće opisani neki od algoritama koji rešavaju ovaj problem.

U informatici, raspoređivanje je metoda po kojoj se procesima i tokovima podataka daje pristup vremenskim resursima. Potreba za algoritmima raspoređivanja procesa je proizašla iz potrebe modernih sistema da u isto vreme obavljaju više poslova.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Aleksandar Kupusinac, red. prof.

U tipičnim slučajevima procesor i periferije nisu ravnomerno iskorišćeni. Takođe u poslednjim poglavljima biće opisani algoritmi koji se bave problemima rasporedom procesa u procesoru.

**2. ISTORIJA PROBLEMA TRGOVAČKOG
PUTNIKA**

Problem trgovačkog putnika (TSP) je dugo privlačio pažnje sa svih strana i jedan je od najintenzivnije proučavanih, ako ne i najproučavaniji problem iz područja matematike. Tačan izvor problema je nepoznat, kao i informacija ko je ovaj problem imenovao tako. Problem se prvi put spominje 1832. godine u priručniku jednog nemačkog trgovačkog putnika. Mnoge trgovce tog doba je zanimalo strateško planiranje što ekonomičnije rute do područja svojih klijenata. Zvanično je prvi put formulisao 1930. godine.

3. DEFINICIJA I PREDSTAVLJANJE

Problem trgovačkog putnika je predstavnik velike klase problema poznatih kao kombinatorni problemi optimizacije. U regularnom obliku TSP-a, mapa gradova je dodeljena prodavcu i on mora da poseti sve gradove samo jednom kako bi završio obilazak tako da je dužina turneje najkraća među svim mogućim putovanjima za ovu mapu. S obzirom na skup gradova i troškova putovanja između svakog mogućeg para, cilj je da se pronađe najbolji način posete svih gradovima i vraćanje na početnu tačku, s najmanjim mogućim troškovima putovanja.

4. EVOLUTIVNI ALGORITMI

U prirodi preživljavaju jedinke koje se prilagođavaju novim uslovima i okolnostima. Jedinke sa karakteristikama kao što su bolje prilagođavanje okolini imaju veće šanse za preživljavanje i povećanje svoje vrste, dok jedinke koje se sporije prilagođavaju izumiru, uzrokujući tako izumiranje čitavih vrsta. Na taj način se u prirodi sprovodi prirodna selekcija i osigurava da geni bolje prilagođenih jedinki imaju veće šanse za preživljavanje od onih koje su se slabije prilagodile. Upotrebom tog osnovnog biološkog modela, nastali su evolutivni algoritmi (*Evolutionary Algorithms*) čiji su deo genetski algoritmi. Kada kažemo evolutivni algoritmi mislimo na one algoritme koju podrazumevaju procese biološke evolucije. Ove metode pretraživanja su našle inspiraciju u prirodi i biološkom svetu. U svakoj iteraciji ovakvog algoritma postoji konkurentni izbor među svim

rešenjima u populaciji; rezultat je najbolji od njih, a loša rešenja se brišu iz populacije. Vrš se rekombinacija i formira se novo rešenje koje je bolje od prethodnih. Rekombinacija i mutacija se koriste u evoluciji populacije.

5. GENETSKI ALGORITMI

Genetski algoritmi (*Genetic Algorithms, GA*) su najpopularnija vrsta evolutivnih algoritama, a predstavljaju model optimizacije čije ponašanje potiče iz procesa evolucije koji se odvija u prirodi. Osnovni cilj genetskih algoritama je pronalaženje rešenja nekog problema koje tradicionalne determinističke metode ne mogu rešiti. Za razliku od većine determinističkih algoritama, genetski algoritmi pretragu ne započinju od jedne tačke nego od celog niza potencijalnih rešenja. U ovim algoritmima razlikujemo ukrštanje i mutaciju. Ukrštanje se često naziva i rekombinacija. Ovo je proces kojim iz dve jedinke roditelja nastaje nova jedinka dete. Najvažnije karakteristika ukrštanja je da deca nasleđuju osobine roditelja. Novonastalo dete poseduje genetski materijal oba roditelja i ima jednak broj gena kao i oni. Mutacija je slučajna promena jednog ili više gena u cilju dobijanja genetske raznolikosti.

5.1. Rešavanje TSP-a korišćenjem genetskih algoritama

Postoji veliki broj algoritama i mogućih rešenja, a GA se smatraju još boljim pristupom, naročito kada su praćeni pažljivo osmišljenim genetskim operaterima. Garantuju veliki prostor rešenja za najbolji odgovor, a operateri im pomažu da postanu brži. Poslednjih godina je ovo privuklo veliku pažnju. Spovedene su mnoge studije i istraživači pokušavaju da doprinesu različitim delovima procesa razmišljanja i rešavanja.

6. OPTIMIZACIJA KOLONIJOM MRAVA

Algoritam optimizacije kolonijom mrava proučava veštačke sisteme koji su našli inspiraciju po ugledu na ponašanje pravih mravljih kolonija, koji se koriste za rešavanje optimizacijskih problema, najčešće pronalaženja najkraćeg puta u zadatom grafu. Ovo je heuristički metod koji je inspirisan stvarnim kretanjem i ponašanjem mrava u pronalaženju najkraće putanje između gnezda i hrane.

Ovo se postiže supstancom koja se naziva feromon koja pokazuje trag mrava. U svom traganju, mravi koriste heurističke informacije koje je već njihovo znanje odakle miris hrane dolazi.

6.1. Ponašanje veštačkih mrava

Za objašnjenje ponašanje veštačkih mrava i izbor nekog od mogućih puteva koristi se jednačina :

$$p_{ij}^k(t) = \left\{ \begin{array}{l} \frac{[\tau_{ij}(t)]^\alpha * [\mu_{ij}]^\beta}{\sum_{k \in \text{dopusteno}_k} [\tau_{ik}(t)]^\alpha * [\mu_{ik}]^\beta} \text{ ako } (j \in \text{dopusteno}_k) \\ 0 \text{ inace} \end{array} \right\}$$

P predstavlja koeficijent isparavanja feromona, τ predstavlja feromon.

6.2. Mravlji algoritam za rešavanje TSP-a

Veštački mrav je agent koji se kreće po grafu TSP. Njegova putna strategija se zasniva na probabilističkoj funkciji koja uvažava dve stvari. Prvo, agent beleži puteve kojima je prošao idući od grada do grada i sabira njihove dužine. Drugo, on oseća miris feromona koje su iza sebe ostavili drugi agenti. Svaki agent menja svoje okruženje na dva različita načina:

- Lokalna promena traga: Dok se kreće između gradova, mrav menja količinu feromona na svakoj pređenoj deonici između dva grada.

- Globalna promena traga: Kada svi mravi završe putovanje, mrav koji je prešao najkraći put dodaje količinu feromona na svim deonicama kojima je prošao. Svrha lokalnog pravila promene traga je da se izbegne da svaki mrav izabere deonice na kojima je položeno mnogo feromona.

Na taj način se povećava istraživačka sposobnost mrava, a u suštinskom smislu izbegava da se proces traženja najboljeg puta okonča u nekom lokalnom minimumu. Globalno pravilo promene traga daje jače osveženje feromonskog traga na najkraćem pređenom putu, odnosno svaka njegova deonica dobija jači miris pri sledećim iteracijama procesa.

7. OPTIMIZACIJA ROJEM ČESTICA

Optimizacija rojem čestica pripada skupu algoritama koji koriste ponašanje inspirisano jatom ptica, riba ili pčela. To je u suštini stohastički algoritam čiji su temelj sociološko-psihološki principi i pružaju uvid u sociološka ponašanja te pomoću njih doprinose rešavanju inženjerskih problema i aplikacija.

7.1. Rešavanje problema TSP-a korišćenjem optimizacije roja čestica

Smatra se da je jedan od odličnih algoritama koji se zasniva na PSO za rešavanje TSP objasnjen u *Pang et al 2004*. U ovom radu je predložen algoritam koji koristi *fuzzi* matrice za vektore brzine i položaja. Pored toga, oni koriste operatore *fuzzi* množenja i sabiranja za formule brzine i položaja.

Pomenuti PSO u prethodnim odeljcima modifikovan je na algoritam koji radi na osnovu *fuzzi* sredstava. U svakoj iteraciji položaj generisanog rešenja se koristi da bi se odredio trošak (vrednost) pojedinca. Ovaj trošak će biti korišćen za ažuriranje najboljeg lokalnog položaja.

8. OPTIMIZACIJA KOLONIJOM PČELA

Kao i svi ostali algoritmi inspirisani prirodom, i ovi algoritmi su zasnovani na kolektivnoj inteligenciji, kao što je PSO preuzet iz života ptica, ACO zasnovan na društvenom životu kolonije mrava -ovo je još jedna vrsta sistema veštačke inteligencije koji može biti koristan u rešavanju mnogih problema i aplikacija što se tiče inženjeringa, upravljanja, kontrole i računskih problema.

Algoritam inspirisan pčelinjim kolonijama u prirodi je jedan od interesantnih metaheurističkih algoritama koji predstavljaju drugi pravac u oblasti inteligencije roja. Zasnovana je na biološkoj potrebi jedinki da se udruže radi ostvarenja zajedničkih ciljeva.

8.1. BCO algoritam

Osnovna ideja je konstruisanje sistema od više agenata što znači da na početku svaka pčela pripada koloniji koja traži hranu pojedinačno.

Ovakva kolonija (kolonija veštačkih pčela) se sastoji od manjeg broja individua – što ne narušava osnovne koncepte modelovanja zasnovane na ponašanju pčela u prirodi. Kada pčela iz kolonije pronađe hranu, ona plesom obaveštava druge pčele. Ostale pčele sakupljaju i prenose hranu u košnicu.

U ovom algoritmu postoje dve faze – let unapred i let unazad. Ove dve faze se smenjuju NC puta u toku jedne iteracije. Smenjuju se dok se ne izgenerišu konstruktivna rešenja ili dok se početna rešenja ne poprave. Tokom leta unapred, sve pčele se bave istraživanjem okolnog prostora.

Po analogiji sa prirodnim procesima, (parcijalno) rešenje se može poistovetiti sa količinom sakupljenog nektara i/ili sa udaljenosti košnice od izvora hrane, a proces sakupljanja nektara u prirodi odgovara fazi leta unapred u BCO algoritmu.

Nakon generisanja svojih (parcijalnih) rešenja veštačke pčele započinju drugu fazu, let unazad. U fazi leta unazad veštačke pčele razmenjuju informacije o kvalitetu svojih pronađenih rešenja.

9. PRIMENA TSP-A I VEZE SA DRUGIM PROBLEMIMA

Osim što predstavlja veliki izazov za rešavanje, problem trgovačkog putnika je široko primenjiv na različite probleme usmeravanja i upoređivanja. Direktno primene pronađene su u problemima bušenja štamparskih ploča, postavljanje lopatica na turbine aviona, analize strukture jednog kristala, izbora naloga u skladištima, planiranje misije u vojsci.

Ovaj problem ima mnogo primena, a ovde su navedene samo neke od njih. Postoji još niz primera u kojima se problem trgovačkog putnika pojavljuje kao potproblem ili se drugi problemi mogu svesti na njega.

10. RASPOREĐIVANJE PROCESA PROCESORA

Pored problema trgovačkog putnika, algoritmi koji su takođe bili zanimljivi i proučavani su algoritmi raspoređivanja procesa u procesoru. Raspoređivanje procesa procesora se bavi problemom odlučivanja- u smislu koji procesi u redu čekanja će biti dodeljeni procesoru. Postoji mnogo različitih algoritama za raspoređivanje procesora. U ovom odeljku SU opisani neki od njih.

10.1. First come, first served algoritam

Jedan od najjednostavnijih algoritama raspoređivanja je upravo ovaj, *first-come, first-served* algoritam raspoređivanja (u bukvalnom prevodu *prvi došao, prvi uslužen*). Sa ovom shemom, proces koji procesor prvo zahteva, prvo je i dodeljen procesoru.

Implementacija sa FCFS algoritmom se lako upravlja pomoću FIFO reda. Kada proces uđe u red čekanja, njegova PCB je povezana na rep reda. Pokrenut proces se zatim uklanja iz reda čekanja. Kod za FCFS algoritam raspoređivanja je jednostavan za pisanje i razumevanje.

Sa druge strane, prosečno vreme čekanja kod FCFS je često veliko.

10.2. Shortest job first algoritam

Drugačiji pristup raspoređivanju rada procesora od FCFS, je tzv. *shortest-job-first* (u bukvalnom prevodu-prvo najkraći posao, u daljem tekstu SJF). Ovaj algoritam povezuje sa svakim procesom dužinu narednog procesa procesora. Kada je procesor slobodan, dodeljen je procesu koji ima najkraće vreme izvršavanja. Ukoliko postoje dva procesa sa istim vremenom izvršavanja, FCFS algoritam je tu da odluči koji će se izvršiti sledeći.

10.3. Raspoređivanje s prioritetom

SJF algoritam je ustvari specijalan slučaj tzv. *priority scheduling* (raspoređivanje sa prioritetom). Svakom procesu je zadat prioritet, a procesor je dodeljen procesu sa najvećim prioritetom. Procesi sa jednakim prioritetom su dodeljeni po FCFS algoritmu raspoređivanja. SJF spada u algoritme raspoređivanja sa prioritetom, gde je prioritet inverzija predviđenog sledećeg vremena izvršavanja. Što je veće vreme izvršavanja, manji je prioritet i obrnuto. U ovom tipu raspoređivanja kao što je već navedeno, postoje procesi niskog i visokog prioriteta. To obično bude opseg brojeva.

10.4. Round robin raspoređivanje

Ovaj algoritam je dizajniran za *time-sharing systeme*. Vrlo je sličan FCFS algoritmu, ali je *preemption* (pretpražnjenje) dodata da omogućiti da se sistem premešta sa procesa na proces. U ovom algoritmu postoji nešto specifično, a to je mala jedinica vremena koja je definisana. Zove se *time quantum* ili *time -slice*. Obično traje od 10 do 100 milisekundi. Spreman red se tretira kao kružni red. *CPU scheduler* ide oko spremnog reda i dodeljuje procesor svakom procesu za vremenski interval do 1 vremenski kvant. Da bismo implementirali RR planiranje, ponovo tretiramo red spremnosti kao FIFO red čekanja procesa. Novi procesi se dodaju u rep spremnog reda. Raspoređivač procesa u procesoru bira prvi proces iz reda spremnosti, postavlja tajmer da se prekine nakon 1 vremenskog kvanta, i raspodeli proces.

11. ZAKLJUČAK

Kao što je već navedeno, TSP je jedan od najproučavanijih problema u istoriji problema optimizacije. Naime, zadatak ne zvuci teško. Najjednostavniji postupak bi generalno bio da se korišćenjem iscrpne pretrage prebroje sve moguće putanje i na kraju da se odabere najoptimalnija. Moglo bi se reći da se vrlo trivijalnim putem stiže do rešenja. Takođe postoji metoda čija je ideja da se krene iz bilo kog grada i da se bira uvek najbliži – što nije najpouzdaniji algoritam. Pored toga, ovde su opisani evolucionari algoritmi, algoritmi inspirisani prirodom i slično. Ove vrste algoritama daju nešto malo manje lošije rešenje od optimalnog. Ali nijedno od ovih rešenja ne predstavlja univerzalno koje garantuje najbolji rezultat. Razlog je složenost algoritma, jer povećanjem broja gradova raste ukupan broj putanja ogromnom brzinom. Zato se ovaj zadatak svrstava u nerešive u opštem smislu takoreći, odnosno u zadatke NP klase. Ovaj problem ima mnogo primena, a ovde su navedene neke od njih. Postoji

još niz primerau kojima se problem trgovačkog putnika pojavljuje kao potproblem ili se drugi problemi mogu svesti na njega. S druge strane raspoređivanje procesa u procesoru je zadatak u kom se bira proces iz reda čekanja kom se dodeljuje procesor nakon toga (od strane dispečera).

FCFS je najjednostavniji algoritam za raspoređivanje, ali može da uzrokuje da kratki procesi čekaju duge procese da se izvrše. Za SJF algoritam bi se moglo reći da je optimalan, međutim njegova implementacija je dosta teška jer treba pretpostaviti i predvideti dužinu sledećeg procesa.

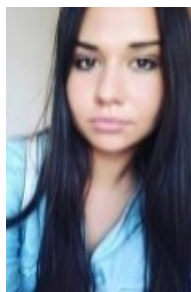
SJF algoritam je poseban slučaj algoritama sa prioritetom, koji alocira procesoru proces sa najvećim prioritetom. RR raspoređivanje je pogodno za interaktivne sisteme, gde postoji vremenska jedinica koja ima najveći značaj u ovom algoritmu. Postoji širok opseg algoritama za raspoređivanje, a ovo su samo neke od njih. Analitičke metode koriste matematičke algoritme da odrede učinak algoritma.

Simulacione metode određuju performanse tako što naprave imitaciju algoritma po uzorku procesa i izračunavaju i programiraju performanse. Simulacijom se najbolje aproksimiraju performanse stvarnog sistema. Jedina pouzdana tehnika za procenu algoritma raspoređivanja je da se primeni algoritam na stvarnom sistemu i da se nadgleda njegovo funkcionisanje u realnom okruženju.

LITERATURA

- [1] U-Turning Ant Colony Algorithm for Solving Symmetric Traveling Salesman Problem- Saman M Al-Mufti
- [2] Programming of processes, Scheduling Algorithms - Uldis Sukovskis
- [3] Wikipedia, Regular expression, 2019, Wikipedia
- [4] Mohammad Reza Bonyadi, Mostafa Rahimi Azghadi, and Hamed Shah-Hosseini (2008). Population-Based Optimization Algorithms for Solving the Travelling Salesman Problem,
- [5] Travelling Salesman Problem –Jose Guevara Ilatoma

Kratka biografija:



Nevena Dovičin rođena je 27.09. 1994. godine u Novom Sadu. Završila je Gimnaziju “20. oktobar” u Bačkoj Palanci, a osnovne četvorogodišnje studije na odseku elektrotehnika, smeru računarstvo i automatika 2017. godine. Ispunila je sve obaveze i položila je sve ispite predviđene studijskim programom.

ROBUSTNI KONTROLER MAKSIMALNE EFIKASNOSTI U POGONU AM ZA ELEKTRIČNA VOZILA

ROBUST EFFICIENCY CONTROLLER IN ELECTRIC VEHICLE DRIVES

Emir Murtezić, Vladimir Popović, Darko Marčetić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu je prikazan sistematičan prilaz analizi performansi kontrolera energetske efikasnosti pogona sa asinhronim motorom u električnim vozilima. Važni aspekti upravljačkog algoritma za maksimizaciju energetske efikasnosti AM kao što su matematički model AM sa gubicima u gvožđu, LMC metode i algoritmi pretraživanja su teorijski analizirani. Izvršena je simulaciona analiza rada predloženog kontrolera na matematičkom modelu pogona AM sa gubicima u gvožđu za sve karakteristične radne režime pogona električnog vozila. Verifikacija rezultata pretraživanja optimalnih trajektorija za unapređenje efikasnosti prikazana je tabelarno i grafički za celokupan opseg brzina i momenata pogona.

Ključne reči: energetska efikasnost, AM, LMC, kontroler pretrage

Abstract – Systematic approach for performance analysis of efficiency controller in induction motor electric vehicle drive is given. Important aspect of this algorithm including IM modelling with incorporated iron losses, loss model controller – LMC and search controller are theoretically analysed. Extensive simulations on IM model including iron losses are performed to demonstrate the effectiveness of the proposed efficiency algorithm. Verification is given through presentation of results which demonstrate the optimal trajectories over all operating drive range.

Ključne reči: power efficiency, AM, LMC, search controller

1. UVOD

Upotreba vozila na električni pogon postaje sve veći trend u svetu. U najvećem broju tih pogona, dominantu poziciju zauzima trofazna AM naizmenične struje sa kratko spojenim kaveznom rotorom. Moderne tehnologije izrade baterija zasnovane na litijum-jonskim ćelijama, još uvek ne pružaju specifične gustine snaga, dovoljne za potrebe električnog transporta na velikim udaljenostima. S tim u vezi danas se u svetu posvećuje velika pažnja projektovanju energetski efikasnih pogona, usled čega se povećava iskorišćenost baterija i sam domet električnog automobila. Opšte je poznato u teoriji električnih pogona, da se odgovarajućim izborom nivoa fluksa u funkciji od momenta opterećenja u AM, može postići maksimalna efikasnost pogona i stoga i optimalna kontrola AM sa stanovišta minimizacije gubitaka pogona.

NAPOMENA:

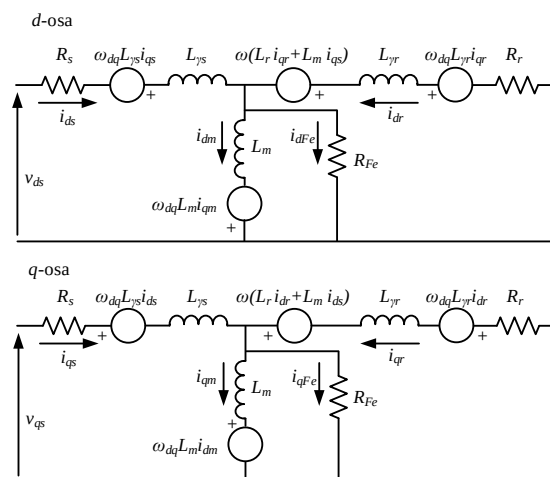
Ovaj rad proistekao je iz master rada, čiji mentor je bio dr Darko Marčetić, red.prof.

U literaturi je predstavljen veliki broj metoda za poboljšanje energetske efikasnosti elektromotornog pogona [1-6]. Metode se mogu klasifikovati u dve osnovne kategorije. Prva kategorija predstavlja LMC kontrolu (eng. Loss Model Controller) [1-4]. Ova metoda bazirana je na matematičkom modelu AM. Optimalni nivo fluksa motora se bira na osnovu izračunavanja minimuma gubitaka motora, predstavljanih putem analitičkih jednačina istog matematičkog modela. Drugu grupu predstavljaju metode, zasnovane na pretraživanju, tzv. SC metode (eng. Search Controller) [5-6]. Ove metode vrše pretragu maksimuma efikasnosti motora, merenjem ulazne snage i promenom reference fluksa u mašini. Utiskivanjem odgovarajućeg vektora napona statora, za konstantnu izlaznu snagu, nastoji se postići minimum ulazne snage, čime se postiže i minimizacija gubitaka tj. maksimizacija efikasnosti u elektromotornom pogonu.

U ovom radu je predložena jednostavna, ali efikasna strategija zasnovana na SC kontroleru, koja objedinjuje dobre performanse oba dijametralno suprotna pristupa. Na osnovu ove strategije, dobija se energetski optimalan pogon u celokupnom režimu rada.

2. MATEMATIČKI MODEL AM SA GUBICIMA U GVOŽĐU

Gubici u gvožđu su gubici u feromagnetnom jezgru mašine, koji se sastoje od gubitaka izazvanih vrtložnim strujama i gubitaka usled histerezisa. Čest prilaz modelovanju efekta gubitaka u gvožđu, jeste dodavanje otpora, paralelnog sa granom magnetčenja, R_{Fe} , u dinamičku ekvivalentnu šemu, kao što je prikazano na Slici 1.



Slika 1. Dinamička ekvivalentna šema AM sa uvaženim gubicima u gvožđu

Jednačine koje opisuju ovaj model su date u nastavku, gde su sve veličine i parametri koji figurišu u njima opisani u [1]:

- Jednačina naponske ravnoteže statora:

$$u_{ds} = R_s i_{ds} + L_{\gamma s} \frac{di_{ds}}{dt} + L_m \frac{di_{dm}}{dt} - \omega_{dq} (L_{\gamma s} i_{qs} + L_m i_{qm}) \quad (1)$$

$$u_{qs} = R_s i_{qs} + L_{\gamma s} \frac{di_{qs}}{dt} + L_m \frac{di_{qm}}{dt} + \omega_{dq} (L_{\gamma s} i_{ds} + L_m i_{dm}) \quad (2)$$

- Jednačina naponske ravnoteže rotora:

$$0 = R_r i_{dr} + L_{\gamma r} \frac{di_{dr}}{dt} + L_m \frac{di_{dm}}{dt} - (\omega_{dq} - \omega) (L_{\gamma r} i_{qr} + L_m i_{qm}) \quad (3)$$

$$0 = R_r i_{qr} + L_{\gamma r} \frac{di_{qr}}{dt} + L_m \frac{di_{qm}}{dt} + (\omega_{dq} - \omega) (L_{\gamma r} i_{dr} + L_m i_{dm}) \quad (4)$$

- Dinamička jednačina grane magnetenja

$$i_{ds} + i_{dr} = \frac{1}{R_{Fe}} \left(L_m \frac{di_{dm}}{dt} - \omega_{dq} L_m i_{qm} \right) + i_{dm} \quad (5)$$

$$i_{qs} + i_{qr} = \frac{1}{R_{Fe}} \left(L_m \frac{di_{qm}}{dt} + \omega_{dq} L_m i_{dm} \right) + i_{qm} \quad (6)$$

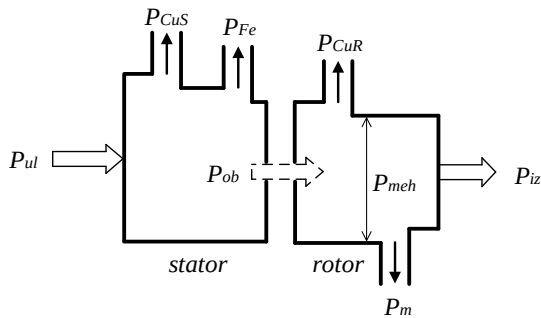
Ove jednačine će biti iskorišćene u okviru simulacionih analiza za verifikaciju algoritma pretrage optimalnih gubitaka pogona AM.

2.1. Energetski bilans AM

Ukupni gubici u AM, P_{gub} , se sastoje od gubitaka u bakru statora i rotora P_{cu} , gubitaka u gvožđu P_{Fe} i mehaničkih gubitaka P_m .

$$P_{gub} = P_{cu} + P_{Fe} + P_m \quad (7)$$

Tok energije kroz mašinu je prikazan na Slici 2:



Slika 2. Energetski bilans asinhronog motora

- Gubici u bakru mašine se dele na gubitke u statorskom P_{cus} i gubitke u rotorskom namotaju P_{cur} i dati su kao:

$$P_{cu} = P_{cus} + P_{cur} = R_s (i_{ds}^2 + i_{qs}^2) + R_r (i_{dr}^2 + i_{qr}^2) \quad (8)$$

- Gubici u gvožđu se sastoje od gubitaka usled histereze i vrtložnih struja:

$$P_{Fe} = k_h \omega \psi_m^2 + k_v \omega^2 \psi_m^2 = (\omega \psi_m)^2 / \left(1 / \left(\frac{k_h}{\omega} + k_v \right) \right) = E^2 / R_{Fe}(\omega) \quad (9)$$

- Mehanički gubici usled trenja i ventilacije:

$$P_m = k_m \omega_r \quad (10)$$

Prve dve komponente gubitaka, zavise od električnih veličina AM, te se mogu kontrolisati putem promene vrednosti tih veličina.

Sa druge strane, mehanički gubici zavise od brzine, koja je referenca u sistemu regulisanim po brzini i momentu, te se na njih ne može uticati.

Osnovni cilj upravljačkog optimalnog algoritma je da minimizuje ove gubitke u cilju poboljšanja stepena iskorišćenja pogona pri svim režimima rada.

2.2. Optimalni zakoni kontrole bazirani na LMC pristupu

Optimalna kontrola zasnovana na LMC pristupu, koristi aproksimativni matematički model pogona i gubitaka u cilju izvođenje zakona optimalnog upravljanja. Na osnovu prethodno navedenog energetskog bilansa AM i aproksimacija modela AM iz [2–4], izvedeni su optimalni zakoni:

- Uprošćeni model AM sa zanemarenim gubicima u gvožđu [2]:

$$\frac{i_{ds}}{i_{qs}} = \sqrt{1 + \frac{R_r}{R_s} \left(\frac{L_m}{L_r} \right)^2} \quad (11)$$

- Uprošćeni model AM sa uvažanim i raspregnutim gubicima u gvožđu [3]:

$$\frac{i_{ds}}{i_{qs}} = \sqrt{1 + \frac{R_r}{R_s} \left(\frac{L_m}{L_r} \right)^2} \sqrt{\frac{1}{1 + \frac{\omega^2 L_m^2}{R_s R_{Fe}}}} \quad (12)$$

- Uprošćeni model AM sa zanemarenim rasipanjem [4]:

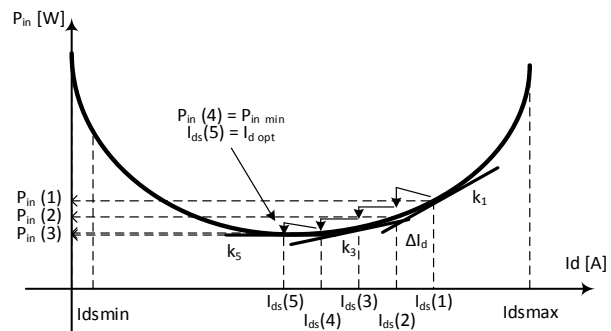
$$\frac{i_{ds}}{i_{qs}} = \sqrt{1 + \frac{R_r}{R_s} \left(\frac{R_{Fe}}{R_{Fe} + R_r} \right)} \sqrt{\frac{1}{1 + \frac{\omega^2 L_m^2}{R_s (R_{Fe} + R_r)}}} \quad (13)$$

Ove relacije će biti upoređene sa optimalnim trajektorijama pogona dobijenim na osnovu nadograđenog algoritma pretrage minimalnih gubitaka i biće komentarisani rasponi greške koja se time postiže.

3. KONTROLER PRETRAGE MINIMALNIH GUBITAKA AM

Standardni pristupi zasnovani na LMC algoritmima zavise od parametara i/ili zahtevaju ispravnu orijentaciju polja, kako bi se osigurala optimalna efikasnost. Ove nesigurnosti su glavni razlog zašto su potrebne dodatne metodologije na bazi pretraživanja, kako bi se osigurala robustnost procesa efikasnosti. Razmatranja u prethodnim paragrafima, apostrofiraju neophodnost nadogradnje LMC pristupa sa dodatnim mehanizmom pretraživanja, kako bi se suzbili nedostaci LMC-a. Kombinovanjem LMC-a sa dodatnim algoritmom pretraživanja, može se dobiti brza, precizna i robustna identifikacija karakteristike optimalne efikasnosti u odnosu na radne uslove u pogonu.

Predloženi algoritam pretraživanja je vrsta gradijentnog optimizatora i prikazan je na Slici 3.



Slika 3. Algoritam za pretraživanje minimalne ulazne snage u odnosu na referencu fluksa

Smer promene fluksa određuje se prema znaku nagiba funkcije $P_{in} = f(I_{ds})$.

Nagib funkcije se računa kao:

$$k(nT) = (P_{in}(nT) - P_{in}(nT - T)) / \Delta I_d \quad (14)$$

Ako je nagib (14) negativan, referenca fluksa se povećava i obrnuto. Strategija promene fluksa se analitički predstavlja kao:

$$k(nT) < 0 \rightarrow I_{ds}(nT + T) = I_{ds}(nT) \pm \Delta I_d \quad (15)$$

gde za znak < odgovara povećanje reference za korak ΔI_d ; + predznak, i obrnuto.

Strategija opisana u (14) se zasniva na metodi opadanja gradijenta i osigurava da apsolutna vrednost nagiba k konvergira ka nuli u vremenu. Optimalna tačka je osigurana u pogonu kada nagib dostigne nultu vrednost. Gore predstavljena metoda je jednostavna metoda za dobijanje optimalnih uslova vožnje. Međutim, ova metoda ima neke praktične nedostatke, koji se moraju preduprediti na odgovarajući način, kako bi se osigurala efikasna i robustna optimizacija.

3.1. Kontroler pretrage – praktični aspekti

Metoda prikazana u prethodnom poglavlju ima određene nedostatke, koji mogu ograničiti opseg njene primene. Samo sa istovremenim ispunjavanjem ovih uslova, može se održati efikasna i efektivna optimizacija unutar pogona. Ovi nedostaci se mogu klasifikovati kao:

- Greška u proceni snage, ukoliko merenje iste nije dostupno unutar pogona
- Nagle promene uslova vožnje (momenat/brzina) u vezi sa procesom pretraživanja
- Problemi sa logikom detekcije stacionarnog stanja i svojstvima konvergencije algoritma

Merenje aktivne snage motora je u većini slučajeva izostavljeno, kako bi se smanjila ukupna cena pogona. Međutim, kada se koriste algoritmi za pretraživanje zasnovani na snazi, za praćenje optimalnih uslova, potrebno je imati informaciju o ulaznoj snazi. U tim slučajevima, procena ulazne snage je osnovni preduslov.

$$P_{el} = \frac{3}{2} (u_{ds} i_{ds} + u_{qs} i_{qs}) \quad (16)$$

Referenca [5] diskutuje problem procesuiranja napona i struja statora za procenu snage. Ovi problemi su uglavnom vezani za procesuiranje signala napona, kada se propuštaju kroz filtarske blokove za potiskivanje viših PWM indukovanih harmonika. Demonstrirano je da je, naročito u slučajevima malih opterećenja, neophodno kompenzovati faznu grešku prvog harmonika napona u cilju precizne estimacije snage [6]. Procesuiranje struje se obično vrši putem kvalitetnih senzora visokopropusnog opsega, dok se mikroprocesorskim odabiranjem na polovini PWM ciklusa može otkloniti sistemski PWM šum.

Drugi problem se rešava umetanjem blokova za promenu reference fluksa, putem rampa funkcija (17) i usrednjavanjem ulazne snage (18). Zbirno gledano, smanjuju se oscilacije momenta i brzine u pogonu i smanjuje mogućnost neželjenih promena fluksa u slučaju oscilovanja trenutne vrednosti snage.

$$i_{ds}(t) = I_{ds}(nT) + k_{ramp} t \rightarrow k_{ramp} = \frac{\Delta I_d}{T} \quad (17)$$

$$P_{in_avg}(nT) = \sum_{j=1}^m P_{in}(jT_{PWM}) / T \quad (18)$$

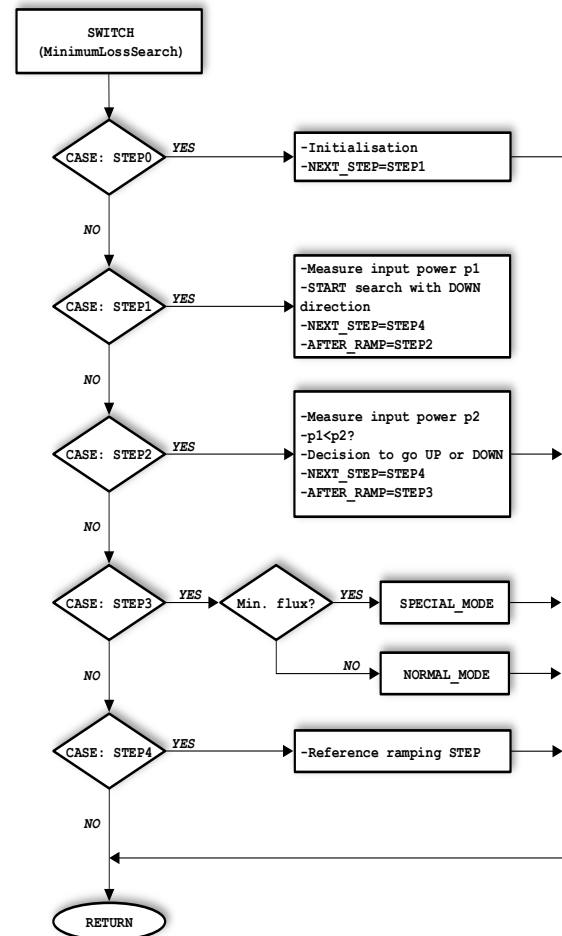
gde su:

- k_{ramp} označen parametar brzine promene fluksa uz već definisan period trajanja rampe fluksa.
- T sastoji od m uzastopnih PWM odabiraka $T = mT_{PWM}$.

Poslednji problem je inherentan kontrolerima pretrage i rešava se postepenim smanjivanjem promene fluksa, kao i efikasnom detekcijom stacionarnog stanja, koja uvažava male promene u varijablama stanja sistema [5].

3.2. Kontroler pretrage – realizacija algoritma

Algoritam mašine stanja kontrolera traženja minimalne snage je prikazan na Slici 4 i sastoji se od pet koraka. Prvi korak – STEP0 služi za inicijalizaciju algoritma pretrage, nakon čega se prelazi na drugi korak. U drugom koraku – STEP1 vrši se merenje prvog odbirka ulazne snage i početak traženja sa smerom prema dole, nakon čega se prelazi u peti korak – STEP4 gde se promena reference fluksa vrši po linearnoj rampa funkciji. Nakon obavljene rampe sledi treći korak – STEP2 gde se vrši merenje odbirka ulazne snage sa smanjenom vrednošću reference fluksa i donosi odluka o daljem smeru traženja i zadavanje nove reference fluksa. Zadana referenca se ponovo menja po linearnoj rampa funkciji u koraku STEP4, nakon čega se prelazi u četvrti korak – STEP3 koji je podeljen na dva radna režima pogona: normalni režim rada (sa referencom fluksa u radnom opsegu) i specijalni režim rada (sa minimalnom referencom fluksa).



Slika 4. Blok dijagram algoritma mašine stanja kontrolera minimalne snage

4. PARAMETRI POGONA ELEKTRIČNOG VOZILA ZASNOVANOG NA AM

U cilju validacije optimalnih algoritama zasnovanih na minimizaciji gubitaka pogona, (LMC i SC metode) opisanih u prethodna dva poglavlja, neophodno je upoznati se sa karakteristikama samog pogona korišćenog za potrebe električne vuče.

Električni i mehanički podaci sa natpisne pločice mašine kao i termičke osobine izolacije su prikazane:

$$U_{sn} = 48 \text{ V}, I_{sn} = 80 \text{ A}, f_n = 170 \text{ Hz}, P_n = 5 \text{ kW}$$

$$n_n = 5000 \text{ o/min}, S2 - 60 \text{ min}$$

Parametri 5kW pogona AM YDQ 6730 su dobijeni putem implementacije metoda za samopodešavanje pogona AM. Kao ulazni parametri u algoritam su korišćeni podaci sa natpisne pločice i dobijeni su sledeći parametri:

$$\text{Otpornosti namotaja: } R_{sn} = 4.5 \text{ m}\Omega, R_{rn} = 5 \text{ m}\Omega$$

$$\text{Induktivnosti rasipanja namotaja: } L_{ys} = L_{yr} = 28 \mu\text{H}$$

$$\text{Nazivna induktivnost magnećenja: } L_{mn} = 0.94 \text{ mH}$$

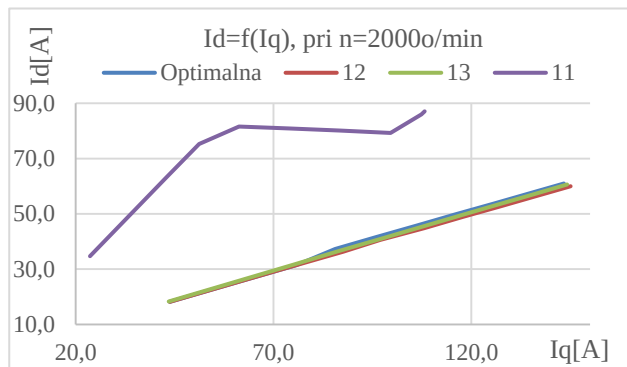
$$\text{Nazivna vremenska konstanta rotora: } T_{rn} = 255 \text{ ms}$$

$$\text{Otpornosti gubitaka u gvožđu: } R_{Fe} = 3.2 \Omega$$

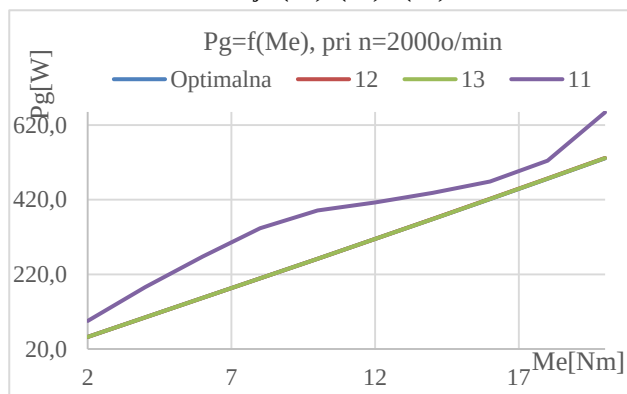
Parametar otpornosti koja modeluje gubitke u gvožđu R_{Fe} ima veliku parametarsku nesigurnost, ako se ne procesuiru putem preciznog merenja snage. Ako se doda i njegova promenljivost sa frekvencijom napajanja (9), nije moguće očekivati njegovu preciznu estimaciju.

5. SIMULACIONA ANALIZA OPTIMALNOG KONTROLERA NA MODELU AM

Simulacije rada LMC metoda zasnovanih na optimalnim relacijama (11), (12) i (13) vršene su u Matlab Simulink modelu. Rezultati simulacije dati su u nastavku.



Slika 5. Funkcija zavisnosti $I_d = f(I_q)$ za optimalne relacije (11), (12) i (13)



Slika 6. Funkcija zavisnosti $P_g = f(M_e)$ za optimalne relacije (11), (12) i (13)

Analizom rezultata simulacije, dolazi se do zaključka da se korišćenjem otpimalnih relacija (12) i (13) pravi greška manja od 0.2% u odnosu na stvarnu optimalnu relaciju, dobijenu nadogradnjom LMC relacije (12) mehanizmom pretrage. Sa druge strane, korišćenjem optimalne relacije (11) može se praviti greška i do 60%. Dakle, ukoliko se dobro poznaju parametri pogona, moguće je koristiti LMC metode, zasnovane na relacijama (12) i (13), čineći beznačajnu grešku. Međutim, u prethodnom poglavlju je već bilo reči o parametarskoj nesigurnosti LMC metoda, te je, usled nepoznavanja parametra R_{Fe} , neophodna nadogradnja LMC metoda mehanizmom pretraživanja, kako bi se dobile prave optimalne trajektorije, i samim tim postigla maksimalna efikasnost električnog pogona.

6. ZAKLJUČAK

LMC metode zbog parametarske osetljivosti, pokazuju dobre performanse pri malim brzinama i opterećenjima gde uticaj nepoznavanja stvarnih vrednosti parametara pogona nije u velikoj meri izražen. Sa druge strane, SC metode u slučaju malih opterećenja (mali faktori snage), značajno povećavaju osetljivost procene snage, te se dobre performanse postižu tek na većim brzinama i momentima, kada je faktor snage značajniji.

7. LITERATURA

- [1] Levi E., "Impact of Iron Loss on Behavior of Vector Controlled Induction Machines", IEEE Transactions on Industry Applications, vol. 31, no. 6, pp. 1287 – 1296, 1995.
- [2] Kioskeridis I., Margaritis N., "Loss Minimization in Induction Motor Adjustable – Speed Drives", IEEE Transactions on Industrial Electronics, vol. 43, no. 1, pp. 226 – 231, 1996.
- [3] Lim S., Nam K., "Loss-minimising control scheme for induction motors", IEEE Proc. Electr. Power Appl. vol. 151, no 4, pp. 385–397, 2004.
- [4] Garcia O. G., Mendes Luis C. J., Stephan M. R., Watanabe H. E., "An Efficient Controller for an Adjustable Speed Induction Motor Drive", IEEE Transactions on Industrial Electronics, vol. 41, no. 5, pp. 533 – 539, 1994.
- [5] Kirschen S. D., Novotny W. D., Lipo A. T., "Optimal Efficiency Control of an Induction Motor Drive", IEEE Power Engineering Review, pp. 70 – 76, 1987.
- [6] Raj C. T., Srivastava S. P., Agarwal P., "Energy Efficient Control of Three-Phase Induction Motor – A Review", International Journal of Computer and Electrical Engineering vol. 1, no. 1, pp. 1793-1800, 2009.

Kratka biografija:



Emir Murtezić rođen je u Novom Pazaru 1994. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva - Energetska elektronika i električne mašine odbranio je 2019.god.

РАЗВОЈ JavaScript БИБЛИОТЕКЕ ЗА РАД СА КОМПОНЕНТАМА КОРИСНИЧКОГ ИНТЕРФЕЈСА**DEVELOPMENT OF JavaScript LIBRARY FOR CREATING USER INTERFACE COMPONENTS**

Ивана Савин, Милан Видаковић, Факултет техничких наука, Нови Сад

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај: У овом раду представљено је креирање JavaScript библиотеке која ће омогућити једноставно креирање компоненти корисничког интерфејса. За демонстрацију рада ове библиотеке написана је једноставна веб апликација за управљање електронском поштом. Клијентски део апликације је развијен помоћу претходно креиране JavaScript библиотеке. Серверски део апликације је развијен у програмском језику Јава, коришћењем Spring окружења. Апликација омогућује основни скуп операција, попут управљања електронском поштом, креирања и слања поште.

Кључне речи: JavaScript, HTML, DOM, Gmail API, WEB, REST.

Abstract: This paper deals with the implementation of JavaScript library for creating user interface components. For demonstration how this library works, a simple web application for email management will be created. The client part of an application will be implemented using the previously created JavaScript library. The server part will be implemented using Java programming language and Spring framework. The application will provide basic managing features like listing, reading, composing and sending email messages.

Key words: JavaScript, HTML, DOM, Gmail API, WEB, REST.

1. УВОД

У овом раду биће описана JavaScript [1] библиотека која омогућује једноставно креирање компоненти корисничког интерфејса, као и примена те библиотеке за развој једноставних веб апликација. Као пример за примену библиотеке креирана је апликација за управљање електронском поштом, инспирисана постојећим решењем (Google Mail [3]).

Будући да је било потребно развити веб апликацију, посебно су развијени клијентски и серверски део. Клијентски део апликације развијен је коришћењем претходно имплементираних библиотека док је серверски део апликације Spring апликација која комуницира са Gmail API-јем.

НАПОМЕНА:

Овај рад је произтекао из мастер рада чији ментор је био проф. др Милан Видаковић.

2. КОРИШЋЕНЕ ТЕХНОЛОГИЈЕ

За имплементацију клијентског дела репрезентативне веб апликације коришћена је *OldSchoolComponents* [2]. JavaScript библиотека за креирање компоненти корисничког интерфејса, jQuery библиотека за управљање елементима DOM стабла [4], Bootstrap библиотека за дизајн компоненти [5] и axios библиотека за комуникацију са сервером [6].

ECMAScript 6

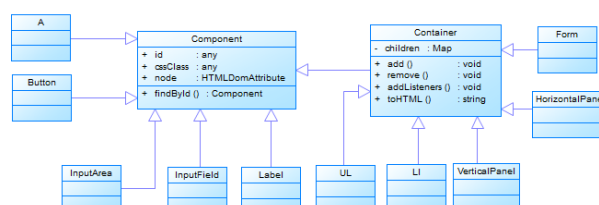
ECMAScript 6 [1] је верзија JavaScript-а која је увела значајна унапређења и олакшања у раду. Нека од значајнијих унапређења су објектно-оријентисани приступ раду и креирање JavaScript класа. ECMAScript 6 класе нуде објектно-оријентисани модел наслеђивања иако је у позадини JavaScript прототипско наслеђивање. На листингу број 1 се може видети креирање JavaScript класа.

```
class Rectangle {
  constructor(height, width) {
    this.height = height;
    this.width = width;
  }
}
```

Листинг 1. Креирање класе помоћу class декларације

OldSchoolComponents

OldSchoolComponents библиотека [2] је JavaScript библиотека намењена раду са компонентама корисничког интерфејса. Она пружа програмеру могућност да веб апликацију прави програмски, употребом кода, тако што се свака компонента на крају претвара у HTML, који се потом уграђује у основну страницу.



Слика 1. Дијаграм класа

Компоненте

Као што се може видети на слици 1, хијерархија графичких компоненти почиње са компонентом *Component* и она је база осталим компонентама.

Основна компонента садржи атрибуте потребне за графичко исцртавање једне HTML компоненте а то су *id* атрибут за идентификацију, *CSSclass* атрибут за приказ компоненте, као и *node* атрибут који представља везу са DOM репрезентацијом компоненте.

constructor метода

Специјална метода *constructor* је задужена за креирање и иницијализацију објеката креираних кључном речју *class*. Приликом креирања објекта класе *Component* и свих објеката који наслеђују ову класу, креира се објекат који има идентификатор и *css* атрибут. Такође, приликом креирања објектне репрезентације компоненте креира се и привремени елемент DOM стабла. На листингу 2 може се видети имплементација *constructor* методе у класи *Component*.

```
constructor(id, CSSclass) {
    this.id = id;
    this.CSSclass = CSSclass;
    this.node
    document.createElement("div");
    this.node.id = id;
    this.node.component = this;
    document.body
    .appendChild(this.node);
}
```

Листинг 2. Имплементација *constructor* методе

Остале компоненте које наслеђују основну компоненту имају могућност додавања специфичних атрибута и метода.

Контејнери

Поред основних компоненти приказа постоје и компоненте које унутар себе могу садржати неку другу компоненту. *Container* објекат се разликује од *Component* објекта по томе што може садржати друге *Component* објекте. Приликом креирања *Container* објекта поставља се атрибут *children* који представља мапу свих објеката које тај контејнер садржи.

Да би се нека компонента приказала потребно је додати је у DOM стабло. Компонента не може постојати уколико није додата у неки *Container* објекат. Из тог разлога постоји метода *add* која додаје компоненту у одређени контејнер на нивоу објектне репрезентације и на нивоу DOM стабла.

Уколико желимо динамичан приказ компоненти, без поновног читавања комплетног интерфејса можемо неку компоненту обрисати и на њено место уметнути нову. Из тог разлога креирана је *remove* метода која истовремено брише објектну репрезентацију компоненте и брише компоненту из DOM стабла.

Приликом додавања компоненти позива се *tohtml* метода која враћа *string* репрезентацију компоненте. Приликом исцртавања контејнера исцртавају се и све компоненте које он садржи.

Google Gmail API

Google Gmail API [7] је скуп функција развијен од стране Google-а како би омогућио другим апликацијама да комуницирају са *Google* сервисима и да их интегришу у своја решења како би их побољшале или

прошириле. Имплементације овог сервиса постоје у програмским језицима Java, JavaScript, .NET. За већину веб апликација *Gmail API* је најбољи избор јер нуди ауторизован приступ корисниковим *Gmail* подацима. У *Google Gmail API*-ју свака функционалност представља посебан ресурс и тиме *Gmail API* нуди *RESTful* приступ функционалностима. Неке од функционалности које нуди *Gmail API* су:

- преглед електронске поште,
- манипулисање електронском поштом,
- креирање нове поште,
- слање поште и
- управљање лабелама.

Google API користи OAuth 2.0 протокол [8] за аутентикацију. OAuth 2.0 је једноставан протокол за аутентикацију а апликација која користи API мора да спецификује *scope* стрингове који служе за идентификовање ресурса које је могуће користити.

Gmail API је веб сервис који користи *RESTful* начин комуникације а ресурси су представљени у JSON формату. Google Gmail API нуди неколико различитих ресурса:

- *message* – ресурс који представља поруку. Порука може бити креирана или обрисана али се ни једно својство поруке не може изменити,
- *draft* – ресурс који представља поруку која је креирана али није послата. *Draft* ресурс је везан за један *message* ресурс који може бити измењен. Након слања, *draft* ресурс се брише и ресурсу се додаје системска лабела SENT,
- *label* – ресурс који служи као средство за категоризацију порука. Има many-to-many [9] везу са порукама. На једну поруку може бити примењено више лабела и једна лабела може бити примењена на више порука,
- *history* – ресурс који представља колекцију измењених порука у хронолошком редоследу,
- *thread* – ресурс који представља колекцију порука које представља конверзацију,
- *settings* – ресурс који пружа могућност контроле над Google налогом.

```
{
  "id": string,
  "name": string,
  "messageListVisibility": string,
  "labelListVisibility": string,
  "type": string,
  "messagesTotal": integer,
  "messagesUnread": integer,
  "threadsTotal": integer,
  "threadsUnread": integer,
  "color": {
    "textColor": string,
    "backgroundColor": string
  }
}
```

Листинг 3 – JSON репрезентација *label* ресурса

Типичан процес рада са *Gmail API* сервисом се састоји из следећих корака:

- аутентикација,
- позив API методе и

- процесирање ресурса добијених у одговору.

Метод *Gmail API*-ја је могуће позивати путем HTTP метода, међутим постоје клијентске библиотеке које то олакшавају. У развоју апликације коришћена је Јава библиотека.

3. СПЕЦИФИКАЦИЈА АПЛИКАЦИЈЕ

Задатак обухвата израду *JavaScript* библиотеке за рад са компонентама корисничког интерфејса – *OldSchoolComponents*. За демонстрацију ове библиотеке написана је веб апликација за управљање електронском поштом. Будући да апликација комуницира са *Gmail API*-јем, корисник има могућност да користи основни скуп функционалности које оригинална апликација нуди. Апликација омогућава прегледање електронске поште, руковање поштом и њено ажурирање. Такође, омогућава креирање нових порука, слање порука и одговарање на постојеће.

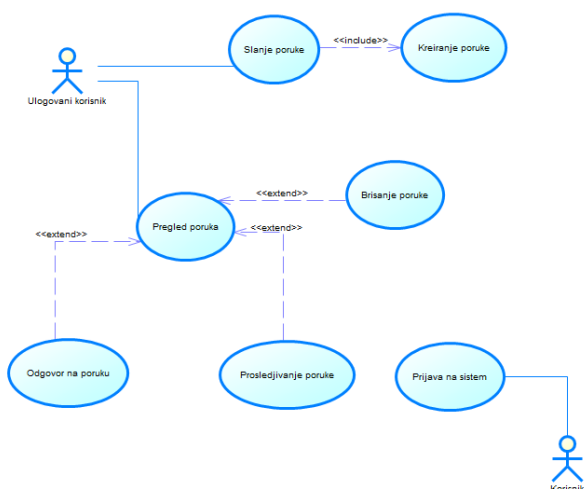
Приликом доласка на веб страницу апликације кориснику је омогућена пријава на систем. Пријава на систем се врши помоћу *Google*-овог система за аутентикацију.

Након успешне пријаве на систем кориснику се приказује главна страница апликације на којој корисник може да прегледа своју електронску пошту по лабелама. Корисник има могућност да креира нову поруку, одговори на неку од примљених порука или да поруку проследи другом кориснику. Поред главне странице са приказом примљене поште постоје још и странице са приказом поште из сваке лабеле.

На страници са поштом за брисање корисник може да прегледа пошту коју је пребацио у канту за брисање и уколико се предомисли да опозове ту акцију или да пошту директно обрише.

На страници са поштом припремљеном за слање корисник има могућност да поруку пошаље.

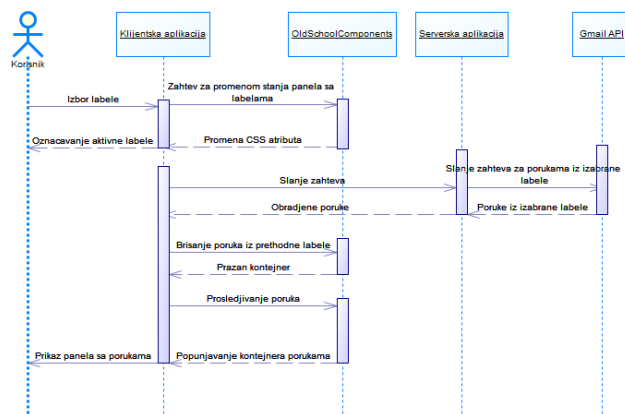
На слици 2 је приказан дијаграм случајева коришћења.



Слика 2 - Дијаграм случајева коришћења

На слици 3 је приказан дијаграм секвенце за приказ порука из селектоване лабеле. Секвенца догађаја

почиње корисниковим избором жељене лабеле из панела са лабелама. Након што корисник изабере једну лабелу њој се промени CSS атрибут и она постаје активна а све остале лабеле имају CSS атрибут постављен на неактиван. Затим клијентска апликација шаље захтев серверској апликацији са називом изабране лабеле која потом упућује захтев *Gmail API*-ју за добављање свих порука из изабране лабеле. *Gmail API* враћа поруке серверској апликацији која потом те податке форматира и шаље клијентској апликацији. Након што пристигну подаци на клијентску апликацију, контејнер са порукама из претходне лабеле се бришу и потом се исцртавају нове компоненте за подацима.



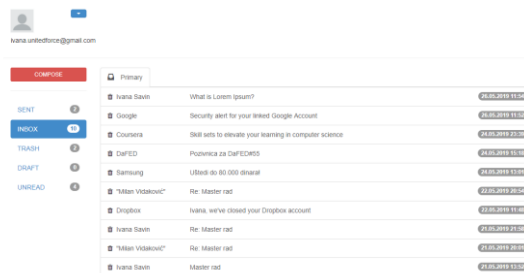
Слика 3. Дијаграм секвенце

4. ИМПЛЕМЕНТАЦИЈА

Апликацију чине две целине, серверски и клијентски део. Серверски део апликације је имплементиран у програмском језику *Java* [10], верзије 1.8 коришћењем *Spring* окружења. За имплементацију клијентског дела апликације коришћена је претходно креирана *OldSchoolComponents* библиотека.

Клијентска апликација

Клијентски део апликације представља *view* компоненту у архитектури ове апликације. Клијентски део апликације чине методе за исцртавање компоненти и методе за комуникацију са серверским делом апликације и добављање порука. Почетно стање апликације је главни прозор са приказом примљене поште и може се видети на слици 4.



Слика 4. Главни прозор апликације

Серверска апликација

Серверски део апликације је креиран зарад лакше комуникације са *Google API*-јем. На серверској страни

се налазе слој *controller*-а и *service*-а. Будући да слој *controller*-а садржи само позиве ка Gmail API-ју сви позиви су груписани у једну класу. Сервисни слој се састоји од три класе у којима се налазе методе за процесирање података потребних контролеру. Као што је споменуто, класа из слоја *controller* садржи методе за аутентикацију, методе за прикупљање корисникових лабела или порука, креирање и слање порука. Сервисни слој садржи методе које трансформишу потребне податке у одговарајући JSON објекат.

5. ЗАКЉУЧАК

Основни задатак овог рада је био развој и демонстрација JavaScript библиотеке за рад са компонентама графичког корисничког интерфејса – OldSchoolComponents. Основна идеја је била да се компоненте корисничког интерфејса могу декларативно направити и додати на екран, како се иначе ради у програмским језицима који се не користе за развој веб апликација.

За демонстрацију OldSchoolComponents библиотеке, развијена је једноставна апликација за управљање електронском поштом, инспирисана већ постојећим решењем. Досадашњи рад обухвата основни скуп операција за управљање електронском поштом, попут прегледања поште, манипулисања поштом, креирања нове поште и одговарања на постојећу.

Апликација је развијана у тренутно актуелним технологијама и има добру основу за даљи развој. Библиотека која је коришћена за креирање компоненти корисничког интерфејса је у процесу развоја тако да су проширења могућа.

6. ЛИТЕРАТУРА

- [1] JavaScript, <https://en.wikipedia.org/wiki/JavaScript>
- [2] OldSchoolComponents, <https://github.com/iv17/OldSchoolComponents>
- [3] Google Mail, <https://mail.google.com/mail/>
- [4] jQuery, <https://api.jquery.com/>
- [5] Bootstrap, <https://getbootstrap.com/>
- [6] axios, <https://github.com/axios/axios>
- [7] Google Mail API, <https://developers.google.com/gmail/api/v1/reference/>
- [8] OAuth 2.0, <https://en.wikipedia.org/wiki/OAuth>
- [9] many-to-many, [https://en.wikipedia.org/wiki/Many-to-many_\(data_model\)](https://en.wikipedia.org/wiki/Many-to-many_(data_model))
- [10] Java, [https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))

Кратка биографија:

Ивана Савин, рођена 29.01.1995. у Зрењанину. Завршила Основну школу „Бранко Радичевић” у Александрову и Зрењанинску гимназију у Зрењанину. Завршила основне академске студије на Факултету техничких наука у Новом Саду, смер Софтверско инжењерство и информационе технологије. Након завршених основних студија уписала је мастер академске студије на истом факултету, смер Софтверско инжењерство и информационе технологије, модул Електронско пословање. Положила је све испите предвиђене планом и програмом.

Милан Видаковић је рођен у Новом Саду 1971. године. На Факултету техничких наука у Новом Саду завршио је докторске студије 2003. године. На истом факултету је 2014. године изабран за редовног професора из области *Примењене рачунарске науке и информатика*.

**ANALIZA POTREBA ZA IZGRADNJOM SKLADIŠTA ZA VOĆE I POVRĆE U
REPUBLICI SRBIJI****NEEDS ANALYSIS FOR THE CONSTRUCTION OF A FRUIT AND VEGETABLE
WAREHOUSES IN REPUBLIC OF SERBIA***Mihajlo Blagojević, Fakultet tehničkih nauka, Novi Sad***Oblast - SAOBRAĆAJ**

Kratak sadržaj – Ovaj rad ima za cilj da ispita potrebe za dodatnim skladišnim kapacitetima za voće i povrće u Srbiji. U radu su distributivni centri za voće i povrće posmatrani kao logistički centri sa specifičnim karakteristikama. Analiza strukture i obima proizvodnje voća i povrća u Srbiji, predstavljala je osnov za proračun potrebnog kapaciteta za skladištenje voća i povrća. Detaljna analiza uslova skladištenja i vremena dozreivanja urađena je za 5 vrsta voća (jabuke, šljive, grožđe, višnje i maline) i 5 vrsta povrća (krompir, paradajz, dinje i lubenice, kupus i kelj i paprike) jer ove vrste čine više od 80% ukupne proizvodnje voća i povrća. Obim uvoza i proizvodnje u julu (kao merodavnom mesecu), predstavljao je osnov za utvrđivanje potrebnih skladišnih kapaciteta. Komparacija sa raspoloživim kapacitetima (za tri definisane varijante) pokazuje razlike između potrebnih i postojećih kapaciteta.

Ključne reči: Skladište, hladnjača, voće, povrće.

Abstract – This paper aims to examine the need for additional scaling capacities for fruits and vegetables in Serbia. The paper first presents the theoretical basics of the concept of logistics centers, and then more detailed distribution centers for fruits and vegetables. After that, the fruit and vegetable production calculation in Serbia was started as the basic parameter for calculating the basic problem, the necessary storage capacity for fruits and vegetables. The detailed analysis of the storage conditions and the time of ripening is done for 5 types of fruits (apples, plums, grapes, sour cherries and raspberry) and 5 types of vegetables (potatoes, tomatoes, melons and watermelons, cabbage and kale and peppers) as these species account for more than 80% production of fruits and vegetables. On the basis of imports and production in July, the month in which all 10 fruits and vegetables were seasoned, the basic assumption about the needs for storage capacities was obtained. By comparison with available data on existing capacities, three variants have been obtained showing the differences between the required and existing capacities.

Key words: Storage, cold storage, fruits, vegetables.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bila dr Svetlana Nikolić, vanredni profesor.

1. UVOD

Skladištenje je samo jedna faza u prometu robe od pošiljaoca do primaoca i zajedno sa ostalim logističkim procesima treba da omogući da roba u nepromenjenom stanju prođe ceo svoj put do krajnjeg potrošača. Skladišta podrazumevaju posebno uređene i tehnički opremljene prostore koji služe za prihvatanje, čuvanje, doradu, oplemenjivanje i primopredaju robe u prometu ili procesu proizvodnje [1].

U okviru skladišnog podsistema primenjuju se različite skladišne tehnologije kojima se, shodno vrsti robe i njenim karakteristikama, obezbeđuju različiti uslovi skladištenja. Posebnu grupu u prometu proizvoda čine voće i povrće. Dokazano je da se povišenjem temperature svi procesi unutar ploda odvijaju brže, a snižavanjem temperature svi procesi u plodu se usporavaju, te je neposredno nakon berbe potrebno rashladiti ubrane plodove. Lanac snabdevanja voća i povrća se svrstava u hladni lanac jer u svim procesima, bilo da se radi o transportu ili mirovanju, mora postojati hlađenje. Najčešće je vremenski najduži proces upravo skladištenje, tako da je osnovni uslov za očuvanje kvaliteta voća i povrća od proizvođača do potrošača postojanje hladnjača.

2. LOGISTIČKI CENTRI

Logistički centar je čvorište u određenoj oblasti gde se, na komercijalnoj osnovi od strane različitih operatera, obavljaju sve aktivnosti koje se odnose na transport, logistiku i distribuciju robe na nacionalnom i međunarodnom tržištu [2]. Logistički centri se formiraju sa ciljem da se pruži kompletna logistička usluga, uvedu savremene logističke tehnologije, povećaju ekonomičnost i efikasnost transporta, optimizuje koncentracija robnog rada i logističkih aktivnosti, smanji vezani kapital, kreiraju uslovi za efikasan razvoj i sl.

2.1. Distributivni centri

Skladišta i distributivni centri imaju iste građevinske elemente: zidove odnosno granice područja, podove, krovnu konstrukciju, dokove za pristajanje kamiona. Međutim, razlika ipak postoji. Suštinske razlike su u procesima i aktivnostima koje se obavljaju. Distributivni centar se može definisati kao objekat čija je osnovna funkcija protok proizvoda i njihova dalja isporuka veleprodajama i maloprodajama nasuprot skladištu koje služi za čuvanje zaliha i čija je funkcija u odnosu na distributivne centre, više statička [3]. Distributivni centri ne podrazumevaju samo uskladištenje robe, već pružaju i druge usluge: pretovar robe, etiketiranje, pakovanje,

pripremu naloga na osnovu porudžbine, organizacija transporta i špedicije, prijem i obrada u slučaju vraćene robe. Dok su skladišta orijentisana na što efikasnije skladištenje robe, uz što bolje iskorišćenje prostora i što manje troškove, distributivni centri su u potpunosti okrenuti klijentima i različitim uslugama koje mogu da im pruže [4].

2.2. Lociranje logističkih centara

Robni tokovi su uzročno – posledični faktor stalnog porasta prostornih, vremenskih i količinskih transformacija u neprekidnoj smenljivosti aktivnosti pakovanja, utovara, transporta, skladištenja, ponovnog pretovaranja, transporta, istovara, skladištenja, isporuke itd. Ova područja se neprekidno istražuju i analiziraju u pogledu mogućnosti racionalizacije, ubrzanja protoka robe, povećanja efikasnosti logističkih sistema, harmonizacije logističkih procesa i kooperacije učesnika u logističkim lancima. Veliki privredni regioni, industrijske zone i urbane sredine, oduvek su bili centri izvora i ponora značajnih robnih tokova i njihove transformacije iz makro u mikro distribuciju i obrnuto. Na mestima sučeljavanja tokova makro i mikro distribucije danas se postavljaju zahtevi za koherentnom logističkom uslugom. Ovim zahtevima najviše odgovara forma robno – transportnog centra kao najkompleksnijeg oblika logističkog centra [5].

3. DISTRIBUTIVNI CENTRI ZA VOĆE I POVRĆE

Distributivni centar za voće i povrće je objekat, gde se vrši sakupljanje voća i povrća od proizvođača, veleprodaja i uvoznika i gde se dodatno vrši razvrstavanje, sortiranje, pranje, sečenje, pakovanje i slično a potom isporuka maloprodaji, prerađivačima, institucijama i direktnim kupcima prema naložima. Njegova uloga je da povećava promet i dobit i da donese konkurentnost kompaniji tako što zadovoljava potrebe klijenata na visokom nivou. To je najvažnija veza između farmera, odnosno dobavljača i potrošača prilikom isporuke proizvoda sa optimalnim troškovima i odgovarajućeg (potrebnog) kvaliteta usluge isporuke.

3.1. Potrebni uslovi prilikom skladištenja voća i povrća

U distributivnom centru za voće i povrće važno je usaglasiti zahteve koje postavljaju ovi proizvodi. Adekvatni uslovi skladištenja, kao što su relativna vlažnost i temperatura, minimalno rukovanje i izbegavanje interakcija proizvoda mogu smanjiti gubitak kvaliteta.

U zemljama u razvoju gubici svežeg povrća i voća čine 40% njihove ukupne proizvodnje, zbog toga što ne postoji adekvatan sistem hlađenja. Tačnije, veliki gubici se javljaju zbog nedostatka skladišta hladnjača, loših logističkih sistema, neefikasnih objekata hladnog lanka i loše infrastrukturne podrške. Za poređenje, u razvijenim zemljama ti gubici iznose samo 15%. U tropskim oblastima ili u toplim letnjim mesecima, usled visokih temperatura i vlažnosti okoline stopa dekompozicije (stepen kvarenja) svežeg voća i povrća je veoma visoka bez hlađenja. Moderni i standardizovani distributivni sistemi su uglavnom primenjeni u razvijenim zemljama, odnosno proces distribucije od dobavljača do potrošača se

vrši u kontrolisanim temperaturnim uslovima u skladu sa karakteristikama proizvoda. U slučaju voća i povrća, temperaturu treba kontrolisati u rasponu od 0 do 5 °C [6].

3.2. Hladnjače sa kontrolisanom atmosferom (CA hladnjače)

Vreme od berbe do rashlađivanja voća i povrća treba da bude što kraće. Preporuka je da se obrani plodovi rashlade istog dana. Brzo rashlađivanje plodova nakon berbe veoma je važno, jer se na taj način usporava proces zrenja i smanjuje pojava pojedinih fizioloških oboljenja.

Hladnjače s kontrolisanom atmosferom imaju sposobnost kontrole, osim temperature i sastava atmosfere unutar komora. To se postiže snižavanjem koncentracije O₂ i povećanjem koncentracije CO₂. U kontrolisanoj atmosferi vrlo je važno vreme potrebno da se uspostavi tzv. ULO (*Ultra Low Oxygen*) režim. Za većinu sorti potrebno je da to vreme bude što kraće, najviše 3 dana. Da bi se kiseonik spustio sa standardnih 21% koliko ga ima u vazduhu, potrebno je ubaciti čist azot. Količina CO₂ takođe je važna, pošto dolazi do oštećenja plodova ako je njegova koncentracija previsoka. U komorama se održava visoka relativna vlažnost vazduha, najčešće od 90 do 95%. Ako bi vlažnost vazduha bila niža, došlo bi do jače transpiracije i plodovi bi postali smežurani i bez tržišne vrednosti.

3.3. Ostale usluge u distributivnom centru

Osim funkcije skadištenja u distributivnom centru za voće i povrće se obavljaju i drugi procesi, podproces i aktivnosti koje karakterišu ovu vrstu logističkih centara. Važno je napomenuti da se svi procesi i aktivnosti sa voćem i povrćem moraju obavljati u skladu sa uslovima koji se postavljaju u pogledu kvarljivosti. Osim skladištenja, distributivni centar vrši organizovanje dopreme, unutrašnji transport, prerađne aktivnosti (kao što su npr. sečenje, vađenje koštica), dozrevanje, pranje, komisioniranje, pakovanje, isporuka itd. Ove aktivnosti prethode distribuciji proizvoda ostalim učesnicima u transportnom lancu kao što su maloprodajni objekti, prerađivačke industrije itd. Distributivni centar za voće i povrće je veza između farmera odnosno dobavljača i kupaca i obezbeđuje isporuku proizvoda klijentu sa optimalnim troškovima, na potrebnom nivou usluge i u dogovorenom vremenu.

4. VOĆE I POVRĆE KAO EKONOMSKI POTENCIJAL SRBIJE

4.1. Karakteristike poljoprivrednih gazdinstava

Učešće poljoprivrede u ukupnoj zaposlenosti u Srbiji je visoko, čak i u poređenju sa drugim evropskim tranzicionim zemljama. Prema poslednjim zvaničnim podacima, čak 21,4% zaposlenih radi u poljoprivredi, što čini oko 18% aktivnog stanovništva. Osnovni razlozi visoke zavisnosti od poljoprivrede su smanjene mogućnosti zapošljavanja i niska investiciona aktivnost, pogotovo u ruralnim područjima [7].

Tokom tranzicionog perioda, vlasnička struktura poljoprivrednog zemljišta u Srbiji je postala vrlo složena: od mikrofarmi u vlasništvu siromašnih seljaka ili naslednika vraćenog zemljišta, preko malih porodičnih poljoprivrednih i mešovityh farmi, do velikih poseda,

porodičnih ili u vlasništvu pripadnika tranzicione elite. Popis iz 2002. godine potvrđuje da najveći broj gazdinstava proizvodi za sopstvene potrebe (75%), dok je samo 20% orijentisano na proizvodnju za tržište. Po podacima Geodetskog zavoda (maj 2009), u privatnoj svojini je 67% zemljišta, u državnoj 30%, društvenoj svojini oko 2%, dok je 1% u ostaloj vrsti svojine (zadružna, meštovita i dr.). U posedovnoj strukturi preovlađuju mala gazdinstva sa znatnim procentom neobrađenog zemljišta. Dualna struktura farmi postoji u severnom delu zemlje gde je i tržište zemljištem, posebno tržište zakupa, mnogo aktivnije [4].

4.2. Uticaj demografskih promena na potencijal proizvodnje voća i povrća

Srbija je za deset godina samo po osnovu negativnog priraštaja izgubila 353 000 stanovnika. Na osnovu podataka OECD Srbija je među vodećim zemljama po broju emigranata u odnosu na broj stanovnika. Prosečan broj emigranata iz Srbije je za period 2004-2014 bio oko 31.000 godišnje. Problem je i što sve veći broj mladih odlazi u inostranstvo i tamo dobija potomstvo [8]. U Srbiji danas relativno veliki deo stanovništva ostvaruje dohodak od poljoprivredne aktivnosti. Na taj način, i pored činjenice da se aktivno poljoprivredno stanovništvo smanjilo za 64,1% u 1991. godini u odnosu na 1948. godinu, odnosno za 37,4% u 2002. godini u odnosu na 1991. godinu (bez Kosova i Metohije), deo koji pripada aktivnom poljoprivrednom stanovništvu od 15,6% jeste dosta veliki, što odražava smanjen nivo ukupne ekonomske razvijenosti. U ovom kontekstu, poljoprivreda se afirmiše kao jedna od osnovnih grana materijalne proizvodnje u privredi Srbije [8].

Usitnjenost poljoprivrednih gazdinstava predstavlja veliku prepreku razvoju intenzivne poljoprivredne proizvodnje. Prema podacima Popisa poljoprivrede iz 2012. godine, u Srbiji je bilo 631.555 gazdinstva, koja koriste 3,437 miliona hektara poljoprivrednog zemljišta i imaju prosečnu veličinu poseda od 5,44 hektara. Među popisanim gazdinstvima 99,6% su gazdinstva fizičkih lica i koriste 84% površina. Prosečna veličina poseda gazdinstva fizičkih lica je prilično mala, iznosi 4,5 hektara i značajno varira po regionima – od 2,1 hektara u Jablaničkoj oblasti do 10,0 hektara u Srednjobanatskoj oblasti. Preostalih 0,4% gazdinstava su u posedu pravnih lica, koriste 16% površina i imaju prosečnu veličinu poseda od 210 hektara po gazdinstvu.

4.3. Proizvodnja voća i povrća u Srbiji

Poslednjih godina se ne zapaža smanjenje površina pod povrćem sa proizvodnjom koja se održava na sličnom ili višem nivou, odnosno primećuju se ujednačeni prinosi po jedinici površine za većinu povrtarskih kultura, sa neznatnim trendom rasta. To je posledica većeg prinosa po hektaru, kao i primena značajnijih investicija u opremu, mehanizaciju kao i sve većoj proizvodnji povrća u zaštićenom prostoru. Osim rasta prinosa, evidentna je i šira ponuda i uvođenje novih vrsta povrća u proizvodni asortiman, kao i sve veći plasman ranog povrća. Proizvodnja povrća na godišnjem nivou iznosi između 1,7-1,8 miliona tona, u vrednosti oko 630 miliona dolara. U voćarskoj proizvodnji angažovano je oko 15% stanovništva, koje zbog nestabilnih - nesigurnih uslova

privređivanja na tržištu i veoma skromnih podsticajnih subvencija, nije potpuno specijalizovano za voćarsku proizvodnju. Ukupna proizvodnja voća na godišnjem nivou iznosi između 1,3-1,4 miliona tona, u vrednosti oko 550 miliona dolara.

U 2017. Godini u Srbiji je proizvedeno 1,33 miliona tona voća i 1,55 miliona tona povrća. Međutim, postoje velike razlike u količini proizvodnje različitih vrsta. Pojedine vrste voća i povrća se proizvode u znatno većim količinama dok su druge znatno više zastupljene u ukupnoj proizvodnji [9].

Pet vrsta voća (jabuke, šljive, grožđe, maline, višnje) čine oko 82% ukupne proizvodnje voća i pet vrsta povrća (krompir, paradajz, kupus i kelj, paprika i dinje i lubenice) predstavlja oko 88% ukupne proizvodnje povrća. S obzirom da izdvojene vrste voća i povrća dominiraju u ukupnoj proizvodnji, shodno definisanoj temi rada, detaljnije su analizirane [9].

5. ANALIZA POTREBA ZA SKLADIŠNIM KAPACITETIMA

Proizvodnja voća i povrća u Srbiji zauzima značajno mesto u ukupnoj vrednosti poljoprivredne proizvodnje i veliki deo proizvedenog voća i povrća se izvozi, pa se samim tim zahtevaju i veliki kapaciteti i uvođenje novih tehnologija skladištenja. Savremene tehnologije koje obezbeđuju sigurno i produženo skladištenje bioproizvoda su skladišta sa kontrolisanom atmosferom CA (*controlled atmosphere*) i veoma malom koncentracijom kiseonika ULO (*ultra low oxygen*) hladnjače.

5.1. Postojeće stanje hladnog lanca u Srbiji

Veći deo proizvodnje svežeg voća i povrća dolazi sa farmi male do srednje veličine, koje ne proizvode velike količine proizvoda. Nepostojanje zajedničkog nastupa pri prodaji obranih proizvoda, što je moguće kroz zadruge i asocijacije, stvara okruženje u kojem svaki farmer prodaje svoj proizvod direktno prerađivačima, što umanjuje njihovu prodajnu moć koja se zasniva na efikasnoj proizvodnji i zajedničkom nastupu prilikom prodaje [10]. Nepoštovanje standarda, osustvo inspektora za bezbednost hrane ili loše iskorišćenje kapaciteta hladnjača samo su neki od problema koji dovode do gubitaka u hladnom lancu.

5.2. Rapoloživi kapaciteti za skladištenje voća i povrća u Srbiji

Kako se procenjuje, u Srbiji, trenutno ima više od 370 hladnjača, u kojima može da se uskladišti oko 600 000 tona voća i povrća [11]. Najvećim delom su to „specijalizovane” hladnjače, u kojima se skladišti i duboko zamrzava jagodičasto voće, višnje i šljive. Onih, namenjenih za rashlađivanje, skladištenje, a zatim i pakovanje drugih vrsta voća ima desetak, dok je hladnjača u kojima se može uskladištiti povrće još manje, i sve su smeštene u Vojvodini. Srbija raspolaže sa oko 100 hladnjača kapaciteta preko 3.000 tona i oko 150 hladnjača kapaciteta preko 1.000 tona. Međutim, precizniji podaci o kategorijama hladnjača i njihovoj lokaciji nisu dostupni. [9].

5.3. Potrebni kapaciteti za skladištenje voća i povrća

U ukupnom obimu proizvodnje voća, pet vrsta (jabuke, šljive, grožđe, maline i višnje) učestvuju sa oko 80%. U ukupnoj proizvodnji povrća pet vrsta (krompir, paradajz, paprika, kupus i kelj i dinje i lubenice) učestvuju u ukupnoj proizvodnji sa oko 90%. Izdvojene vrste se mogu smatrati relevantnim za dalju analizu pri utvrđivanju potrebnih skladišnih kapaciteta.

Da bi se odredio neophodan kapacitet hladnjača za voće i povrće u Srbiji, potrebno je, između ostalog, uzeti u obzir doba godine kada sazreva voće i povrće kao i potencijalnu dužinu njihovog skladištenja. Period godine u kome postoje najveće potrebe za skladišnim kapacitetima posmatran je kao merodavan za utvrđivanje kapaciteta. Analizom vremena dozrevanja i osobina voća i povrća, dobijeno je da u julu sve izdvojene vrste voća i povrća opterećuju skladišne kapacitete. Zbog toga je količina proizvedenog voća i povrća u julu uzeta kao merodavna prilikom izračunavanja kapaciteta i iznosi 527 825 tona. U Srbiju se u poslednjih 10 godina prosečno uveze 350 000 tona voća i povrća. Ukoliko se usvoji da se voće i povrće ravnomerno uvozi tokom godine, količina uvoza u julu iznosi 29 167t. Za ostale vrste voća i povrća, na osnovu podataka o njihovoj proizvodnji, potrebno je u julu obezbediti skladišne kapacitete za 144 786 tona.

Ovakvim proračunom je dobijeno da količina proizvodnje i uvoza iznosi 701 778 tona voća i povrća. Na osnovu ovih rezultata analizirane su 3 moguće varijante koje određuju potreban kapacitet skladišta u Srbiji.

U prvoj varijanti je uzeto da 30% (210 533 tone) dobijene proizvodnje u julu ne ide u skladište. Pretpostavljeno je da zbog velike usitnjenosti poljoprivrednog zemljišta deo prinosa ne odlazi u skladište, već se odmah konzumira ili otprema do mesta prodaje. U ovom slučaju potreban kapacitet skladištenja voća i povrća u Srbiji iznosi 491 245 tona voća i povrća što je za 108 755 tona manje od raspoloživog kapaciteta. Ova varijanta pokazuje da potrebni kapaciteti odgovaraju zahtevima.

U drugoj varijanti je pretpostavljeno da 10% (70 178 tona) voća i povrća ne ide u skladište iz istog razloga kao i u prethodnoj varijanti. U ovoj varijanti potreban kapacitet iznosi 631 600 tona voća i povrća što znači da potrebe premašuju postojeće kapacitete (600 000 tona) za 31 600 tona. Ova varijanta pokazuje da je potrebno povećati skladišne kapacitete za oko 5%.

Treća varijanta podrazumeva da celokopnu proizvodnju i uvoz treba skladištiti. U ovom slučaju se može uočiti da skladišni kapacitet za 701 778 tona premašuje postojeće kapacitete za oko 20%.

6. ZAKLJUČNA RAZMATRANJA

Polazne pretpostavke zasnivaju se na osnovnim zahtevima koje treba da ispuni distributivni centar za voće i povrće. Obuhvaćene su osnovne preporuke u vezi sa uslovima skladištenja voća, povrća i cveća. Savremene hladnjače sa kontrolisanom atmosferom koje su sastavni deo modernih skladišta za voće i povrće treba primeniti za većinu sorta voća i povrća koje se proizvodi u Srbiji.

U skladu sa podacima dobijenim analizom pretpostavljene su tri varijante.

U prvoj varijanti analizirana je potreba za skladišnim kapacitetom ukoliko 30% proizvoda ne postavlja zahtev za skladištenjem. Ova varijanta pokazuje da ne postoji potreba za dodatnim skladišnim kapacitetima za voće i povrće jer popunjenost iznosi 82% postojećih kapaciteta. Druga varijanta podrazumeva da se 10% voća i povrća odmah konzumira ili prodaje. Ova varijanta ukazuje na potrebu za dodatnim kapacitetima jer potrebe premašuju postojeće skladišne kapacitete za 5%.

Treća varijanta uključuje skladištenje celokupnog prinosa voća i povrća. U ovoj varijanti je pokazano da postoje značajne razlike između postojećih i potrebnih kapaciteta i da za oko 20% ukupnog prinosa nema skladišnog prostora u hladnjačama.

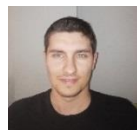
Rad obuhvata proračun potrebnog kapaciteta hladnjača za skladištenje voća i povrća, ali ne zalazi u prostorni razmeštaj hladnjača na teritoriji zemlje. Optimalna prostorna raspodela bi uvećala konkurenske prednosti proizvodnje i distribucije voća i povrća.

Rezultati dobijeni u radu najvećim delom uključuju skladištenje voća i povrća u uslovima kontrolisane atmosfere i skladištenje u komorama za duboko smrzavanje. Rasčlanjivanjem količine voća i povrća na ono namenjeno za duboko smrzavanje i ono koje je namenjeno za korišćenje u svežem stanju mogu se odvojiti potrebni skladišni kapaciteti za sveže i smrznuto voće i povrće.

7. LITERATURA

- [1] Stojanović, L., „Unutrašnji transport i skladištenje” Završni rad, Sveučilište Sjever, pp. 11, 2016.
- [2] Nikolić, S., Skripta sa predavanja, Fakultet tehničkih nauka, Novi Sad, 2014.
- [3] Ivačković, A., Kostandinović, N., Organizacija i optimizacija skladišta, Fakultet tehničkih nauka, Novi Sad 2014.
- [4] <https://www.industrija.rs/vesti/clanak/skladiste-distribucioni-centri-i-voice-pick-tehnologija>.
- [5] Ministarstvo poljoprivrede, šumarstva i vodoprivrede, *Nacrt strategije ruralnog razvoja 2010-2013*, Republika Srbija, 2009.
- [6] <http://www.ilsclusters.com/pdf/report%204.pdf>.
- [7] Milićević, D., „Demografska slika Srbije i Evrope”, Makroekonomija, 2018.
- [8] <https://www.tehnologijahrane.com/knjiga/preradjivaci-voca-i-povrca-u-srbiji>.
- [9] <http://www.stat.gov.rs/>.
- [10] Neel, S., Bonar, H. II, „Strategija lanca rashlade za Srbiju”, USAID, 2009.
- [11] <http://angbalkan.rs/poljoprivredni-sektori/sektor-proizvodnje-voca-i-povrca.php>

Kratka biografija:



Mihajlo Blagojević je rođen u Boru 1992. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Saobraćaja odbranio je 2016. god.

УТИЦАЈ ЕЛЕКТРОНСКОГ СИСТЕМА НАПЛАТЕ НА ДИНАМИЧКЕ ЕЛЕМЕНТЕ ЛИНИЈЕ**THE INFLUENCE OF AN ELECTRONIC PAYMENT SYSTEM ON DYNAMIC ELEMENTS OF LINE**Слободан Нинков, Павле Питка, *Факултет техничких наука, Нови Сад***Област – САОБРАЋАЈ**

Кратак садржај – У овом раду је извршена процена уштеде времена укрцавања путника у возило и утицај на динамичке елементе линија за случај замене класичног система наплате карата електронским системом у јавном градском превозу путника у Новом Саду.

Кључне речи: Систем наплате, измена путника, јавни превоз путника

Abstract – This paper evaluates reduction in boarding time of passenger and influence on dynamic elements of line for the requirements of substituting the classical ticketing system with the electronic one in public passenger transport Novi Sad.

Keywords: Payment system, passenger substitution, public passenger transport

1. УВОД

У систему ЈГПП-а у Новом Саду се тренутно користи класичан систем наплате. Класичан систем наплате (КСН) и поред великог броја позитивних страна представља застарелу технологију која је у већини развијених европских градова замењена електронским системима наплате (ЕСН).

Тренутни систем наплате у Новом Саду не пружа довољну информациону основу помоћу које би управљање системом било ефикасно. Поред тога што тренутни систем наплате утиче на ефикасност менаџмента, он такође утиче на брзину измене путника. Измена путника се реализује на стајалиштима док се возило налази у стању мировања. Бржи системи наплате омогућавају краће задржавање возила на стајалиштима самим тим скраћују време обрта возила. Смањењем времена обрта у постојећем систему ЈГПП-а у Новом Саду би могло имати значајан утицај и на остале динамичке елементе линија.

На територији Новог Сада дневно се оствари и до 180 000 возњи путника [1], што представља значајан потенцијал за оптимизацију времена задржавања возила на стајалишту.

Предмет овог рада је да се установи време задржавања возила на стајалиштима ради измене путника и да ли се и у којој мери може утицати на смањење тог времена кроз имплементацију електронског система наплате. Циљ рада је да се покаже како смањење времена задржавања возила на стајалишту може утицати на динамичке елементе линија.

2. МЕТОДОЛОГИЈА ИСТРАЖИВАЊА

Утврђивање просечног времена потребног за улазак путника у возило је извршено у систему ЈГПП-а у Новом Саду. Истраживање је спроведено на појединим стајалиштима система, која имају велику фреквенцију возила. Стајалишта на којима је вршено истраживање: Булевар Ослобођења - Лутрија Војводине; Булевар Михајла Пупина – РК “Базар”; Стражиловска улица - Факултети. Истраживање је извршено, у понедељак 16.07.2018. године, у периодима од по 2 сата на сваком од наведених стајалишта.

Истраживање је обухватило мерење времена задржавања возила на стајалишту, као и бројање путника који су ушли у возило. Истраживање је вршено мануелно помоћу истраживача - бројача. За мерење времена коришћен је мобилни телефон са инсталираном апликацијом Stopwatch, док је бројање путника који су ушли у возило вршено мануелно.

3. РЕЗУЛТАТИ ИСТРАЖИВАЊА

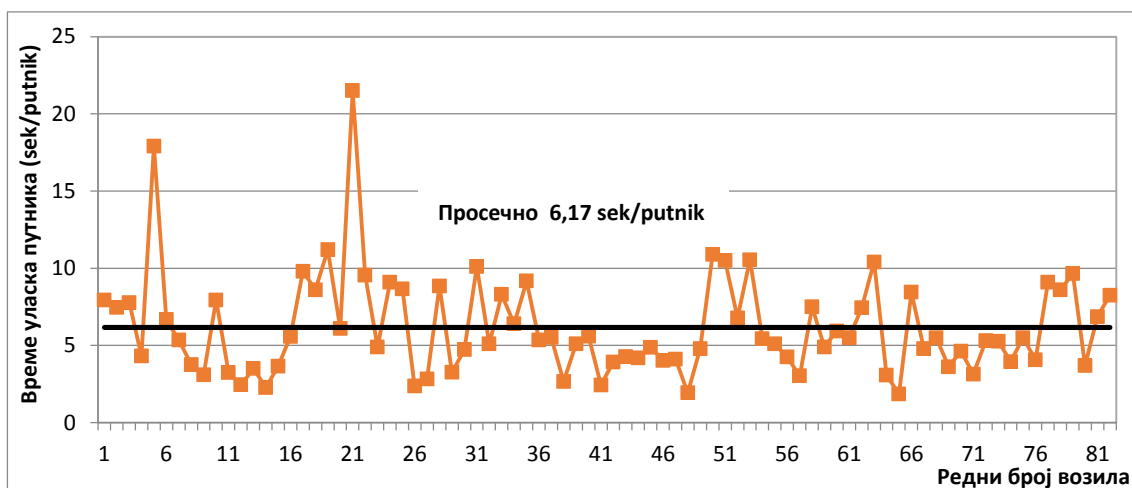
Истраживање на градским линијама је обухватило 81 измену путника, односно 386 улазака путника у возило. Резултати истраживања су представљени графички на слици 1. На основу резултата истраживања је добијено да просечно време уласка једног путника у возило на градским линијама износи 6,17 секунди, уз просечну стандардну девијацију од 3,28.

Линије које су обухваћене овим истраживањем су: Линија 1, Линија 3, Линија 4, Линија 6, Линија 8, Линија 9, Линија 11А и Линија 12. Резултати истраживања по обухваћеним линијама су дати на слици 2.

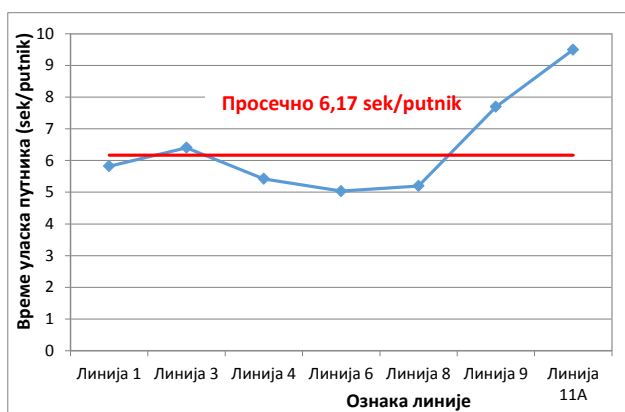
Као што се може приметити време од 6.17 секунди по путнику представља неприхватљиву величину гледано са аспекта модерног јавног превоза европских земаља. Потребно је тежити његовом смањењу, поготово у подручјима где влада повећани интензитет захтева за превозом.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Павле Питка, доцент.



Слика 1. Резултати истраживања



Слика 2. Резултати истраживања по линијама

Међутим ако се узме у обзир да у литератури за систем наплате код возача просечно време уласка износи 11 секунди по путнику, онда резултати добијени приликом истраживања у Новом Саду нису лоши. Већа брзина уласка путника постиже се захваљујући великим уделом месечне (показне карте) у структури карата.

4. ЕЛЕКТРОНСКИ СИСТЕМ НАПЛАТЕ

Полазећи од тога да је субјект наплате путник, може се направити основна класификација према којој постоје три система наплате [2]: Систем наплате преко кондуктера; Систем наплате преко возача; Систем наплате самоуслуживањем.

Електронски систем наплате (ЕСН) спада у систем наплате самоуслуживањем што значајно убрзава измену путника на стајалиштима (Табела 1).

На тржишту постоје три врсте безконтактних картица. Подела картица је извршена према растојању које је потребно да би се извршила размена података: веома кратког домета; кратког домета; већег домета.

Табела 1. Време уласка путника у зависности од система наплате [3]

Куповина карте	Време уласка путника (секунди/путнику)
У возилу – кеш	11.06
У возилу – магнетна трака	3.78
У возилу – безконтактна картица	2.93
Ван возила (1 врата)	2.35
Ван возила (2 врата)	1.74
Ван возила (3 врата)	1.32

Безконтактне картице веома кратког домета се морају поставити на површину читача. Иако изгледа да је ипак потребан контакт, чињеница је да у контактне картице спадају оне које се морају уметнути у читач.

Картице кратког домета су најчешће присутне у пракси. Растојање потребно да се читавање оствари је до 10 cm. Размена података се остварује за 0,1 - 0,3 sek.

Картице које могу бити прочитане и на растојању до 1 m су картице већег домета. Оне имају највећу практичност, јер не доприносе стварању застоја приликом валидације.

Безконтактне картице функционишу помоћу RFID технологије (Radio Frequency Identification) која помоћу високе фреквенције омогућава пренос података без потребе за контактом металних делова.

5. ДИНАМИЧКИ ЕЛЕМЕНТИ ПРЕ И ПОСЛЕ УВОЂЕЊА ЕСН

Динамички елементи линије се утврђују редом вожње који се мења периодично, у складу са променама превозних захтева на линији. Динамички елементи истовремено представљају и значајне параметре квалитета ЈГПП-а [4].

Динамички елементи линија су:

- Број возила на линији,
- Време обрта,
- Брзина обрта,
- Интервал слеђења возила на линији,
- Фреквенција,
- Превозна способност линије.

Образац који нам приказује зависност између времена обрта, интервала слеђења и броја возила на линији:

$$i = \frac{T_o}{N_r} \quad (1)$$

Где је:

- i – Интервал слеђења [min],
- T_o – Време обрта [min],
- N_r – Број возила на линији [возила].

Смањењем времена измене путника скраћује се време обрта, што може даље да утиче на интервал слеђења и број возила на раду и ту разликујемо три случаја.

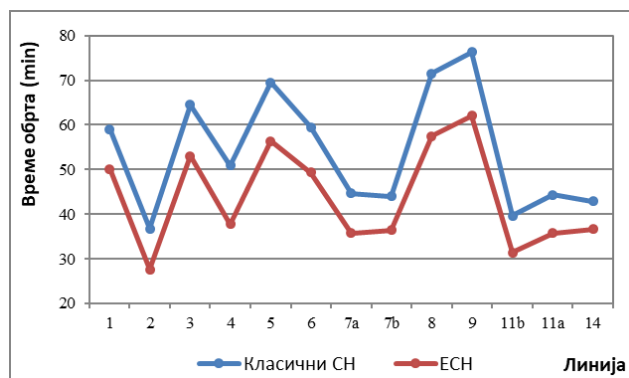
Први случај је да је смањење времена обрта мање од интервала слеђења и тада се задржава исти број возила на линији, а интервал слеђења се смањује. Тада се са постојећим ресурсима у систему добија бољи квалитет услуге, јер смањени интервал слеђења возила значи бољу временску приступачност путницима и већу превозну способност. Повећање превозне способности позитивно утиче на комфор.

Други случај се јавља када су уштеде у времену обрта веће или једнаке пројектованом интервалу слеђења. Тада је могуће смањити број возила и задржати постојећи пројектовани интервал, односно могуће је направити уштеде у систему ЈПП-а, а задржати претходни квалитет услуге (интервал слеђења, превозна способност и комфор остају непромењени).

Код трећег случаја, баш као и код другог уштеде у времену обрта су веће или једнаке пројектованом интервалу слеђења, али се не смањује број возила на раду већ се као код првог случаја смањује интервал слеђења и са постојећим ресурсима значајно повећава квалитет услуге. У овом раду као опција за оптимизацију система биће разматран само први или други случај.

На слици 3 се могу видети времена обрта у оба случаја: Класичан систем наплате и електронски систем наплате.

ЕСН омогућава да се време обрта смањи на свакој линији без изузетака.

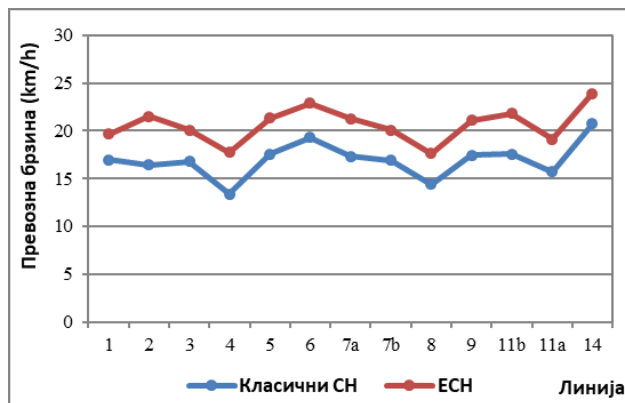


Слика 3. Времена обрта за КСН и ЕСН

На линијама 8 и 9 су остварене највеће уштеде у временима обрта. На линији 8 је након симулације установљена уштеда од 14 минута, док је на линији број 9 остварена уштеда од 14,3 минута.

На те две линије актуелне вредности интервала слеђења износе 8,5 и 10,3 минута респективно, што даље значи да је могуће смањење броја возила на раду за једну јединицу.

Имплементација система електронске наплате ће имати позитиван утицај на превозну брзину кроз смањење временских губитака која су актуелна на стајалиштима. На слици 4 је дата превозна брзина по линијама за случај постојећег стања и унапређеног система наплате.



Слика 4. Превозна брзина по линијама у вршном сату

На претходној слици се види да је повећање превозне брзине значајно на свакој линији и на нивоу целокупног система за анализирани вршни час износи просечно 3,7 km/h. Највеће вредности које су израчунате односе се на линију број 2 и линију број 4 и износе 5,1 и 4,4 km/h. Повећање брзине превоза директно зависи од броја превезених путника и дужине линије. Највећи ефекти на повећање брзине улед бржег система наплате је на линијама са мањим дужинама трасе, а са већом изменом путника.

Када уштеде у времену нису веће од просечног интервала слеђења, тада не постоји могућност смањења броја возила на раду, јер би у супротном интервал слеђења био повећан што би директно утицало на смањење квалитета услуге. Тада се позитивни ефекти могу наћи у повећању квалитета услуге кроз смањење вредности интервала слеђења, односно повећање фреквенције возила на посматраној линији.



Слика 5. Однос уштеде у времену обрта и интервала

На слици 5 може се уочити да се од 13 посматраних линија на 8 може приметити уштеда у времену већа од интервала слеђења што омогућава смањење броја возила на раду за једну јединицу. На 5 посматраних линија (6, 7b, 11b, 11a и 14) уштеда у времену је мања од вредности интервала слеђења и могућност за смањење броја возила на раду не постоји. На тих 5 линија ипак постоји могућност смањења интервала слеђења, односно повећања фреквенције возила без увођења додатних возила на линији.

6. ЗАКЉУЧАК

Имплементацијом ЕСН пре свега информациони систем јавног превоза бива значајно унапређен. Међутим, увођењем електронског система наплате смањује се просечно време уласка путника у возило. Смањењем задржавања возила на стајалишту смањује се време обрта, што може имати позитиван утицај на друге динамичке елементе линија. Пре свега, долази до повећања превозне брзине. Превозна брзина представља просечну брзину коју путник остварује приликом кретања између два терминала.

У оквиру овог рада извршено је истраживање потребног времена уласка путника у возило ЈГПП-а у Новом Саду, где је у примени класичан систем наплате без кондуктера у возилу. Истраживање је извршено на стајалиштима ЈГПП-а, при чему је за просечно време уласка путника у возило добијено 6,17 секунди.

Прегледом доступне литературе дошло се до закључка да увођењем електронског система наплате (бесконтактне смарт картице) у систем ЈГПП-а у Новом Саду просечно време уласка путника у возило ЈГПП-а може да се смањи на 2,93 секунде.

За познате расподеле улазака путника у возило (систематско бројање путника 2010. год) и претпостављене уштеде времена при уласку путника у возило (последња примена ЕСН) извршена је симулација времена обрта возила на линији. На основу нових (смањених) времена обрта израчунати су и други динамички елементи на свим градским линијама, уз претпоставку да је време задржавања возила на стајалишту детерминисано временом уласка путника у возило.

Добијени резултати су показали да увођењем ЕСН уз задржавање истог броја возила на раду интервал слеђења може да се смањи у распону од 1 минута до 14,3 минута у зависности од линије и периода у току дана. Док је број возила на раду могуће смањити за 1 возило на линијама 2, 3, 4, 5, 7а, 8, и 9.

Из наведеног истраживања и извршене симулације може се закључити да се увођењем ЕСН у систему ЈГПП-а у Новом Саду могу постићи значајна побољшања квалитета услуге.

7. ЛИТЕРАТУРА

- [1] *Системско-генерално бројање и анкета путника у јавном градском и приградском превозу путника на подручју Новог Сада*, ЈП "Урбанизам" завод за урбанизам, Нови Сад, 2010.
- [2] П. Гладовић, В. Поповић, М. Симеуновић, „*Информациони системи у друмском транспорту*“, Нови Сад, ФТН Издаваштво, 2014.
- [3] Tirachini, Alejandro, i David A. Hensher, „Bus congestion, optimal infrastructure investment and the choice of a fare collection system in dedicated bus corridors“. *Transportation Research Part B: Methodological* 45 (5), 2011, Elsevier Ltd: 828–44. doi:10.1016/j.trb.2011.02.006.
- [4] Р. Банковић, „*Организација и технологија јавног градског путничког превоза*“, Београд, Саобраћајни факултет, 1994.

Кратка биографија:



Слободан Нинков рођен у Сомбору 1991. год. Bachelor рад на Факултету техничких наука из области Експертизе саобраћајних незгода – Временско-просторна анализа саобраћајних незгода са учешћем пешака одбранио је 2016. год.

контакт: ninkov91@gmail.com



Павле Питка рођен је у Шашинцима 1983. Докторирао је на Факултету техничких наука 2016. год., а од 2017. је звању доцент. Област интересовања су системи јавног превоза.

IMPLEMENTACIJA STANDARDA ENERGETSKE EFIKASNOSTI ISO 50001 U PROIZVODNOM SISTEMU**IMPLEMENTATION OF THE ENERGY EFFICIENCY STANDARD ISO 50001 IN THE PRODUCTION SYSTEM**

Miloš Kovačević, *Fakultet tehničkih nauka, Novi Sad*

Oblast - MEHATRONIKA

Kratak sadržaj – U ovom radu opisan je način uvođenja ISO 50001 standarda u kompaniju JC. Standard ne propisuje samo primenu modernih ili novih tehničkih rešenja, već i potrebu za kvalitetnim upravljanjem energijom i osposobljenost osoblja da na adekvatan energetska efikasan način upravlja kako opremom tako i energijom. U ovom radu su predstavljene prakse i rešenja prepoznati u okviru kompanije i njihova potpuna implementacija na tehničkom i organizacionom nivou.

Ključne reči: ISO 50001, menadžment energije, energetska efikasnost, prakse.

Abstract – This paper describes how to introduce the ISO 50001 standard in the company JS. The standard does not prescribe only the application of modern or new technical solutions, but also the need for quality energy management and the ability of personnel to manage in an energy efficient manner both equipment and energy. In this paper, presentation of practices and solutions recognized within the company and their full implementation at the technical and organizational level is given.

Key words: ISO 50001, energy management, energy efficiency, practice.

1. UVOD

Prema standardu ISO 50001, koji pruža zahteve za sisteme menadžmenta energije, "sistem menadžmenta energije" je definisan kao skup međusobno povezanih i međusobno delujućih elemenata, koji imaju za cilj da se uspostavi energetska politika i energetska opšti ciljevi, uključujući i procese i procedure za dostizanje tih opštih ciljeva [1]. Kroz okvir menadžmenta energijom u organizacijama, kako preporučuje međunarodni standard ISO 50001, organizacije treba uspostaviti programe uštede energije kojima se ostvaruju energetska efikasnost i konkurentnije poslovanje.

Ovi programi treba da istaknu prioritetna ulaganja u projekte energetske efikasnosti, koje jedno preduzeće ima za cilj da realizuje [2]. Međunarodna organizacija za standardizaciju (ISO) je identifikovala energetska menadžment kao jednu od pet najznačajnijih oblasti koja zahteva razvoj i promociju međunarodnih standarda. Zasnovan na široj primeni kroz nacionalne ekonomske sektore, budući standard može da utiče na do 60% svetskih potreba za energijom.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milovan Lazarević, vanr. prof.

2. PLAN IMPLEMENTACIJE

Iako je prvenstveno namenjen industriji, standard je primenljiv na bilo koji tip organizacije koja želi da uspešno upravlja korišćenjem energije i energetska efikasnošću [3].

Kao multinacionalna kompanija Johnson Controls je standardizovao korake prilagođene za sve svoje proizvodne jedinice i oni su [4]:

- 1) Izabrati koordinatora programa. Koordinator je odgovoran za vođenje tima za implementaciju standarda.
- 2) Sastaviti tim. Tim je potrebno obučiti za korišćenje alata za implementaciju, praksi i procesa. Tim za EM sastoji se od članova iz proizvodnog održavanja i inženjeringa, upravljanja proizvodnjom i radne snage generalne proizvodnje. Preporučuje se uključivanje upravljanja postrojenjima i centralnih funkcija kao što su ljudski resursi ili finansije.
- 3) Prikupiti i analizirati podatke. Sakupljaju se informacije o korišćenju energije u preduzeću. Pregledaju se postojeći projekti energetske poboljšanja postrojenja, trendovi potrošnje energije i ciljevi smanjenja proizvodnje. Pored toga, pregledi prošlih računa za energetiku su neophodni kako bi se identifikovale mogućnosti za projekte za uštedu energije. Članovi tima je potrebno da planiraju akcije zajedno kako bi identifikovali mogućnosti ili vrste postupaka upravljanja energijom kojima je potrebno pružiti najveću pažnju.
- 4) Uspostaviti plan komunikacije u kompaniji kako bi se podstakli zaposleni da postanu aktivni učesnici u identifikovanju novih mogućnosti poboljšanja i preduzimanju akcija za smanjenje potrošnje energije.
- 5) Proceniti svaku primenu upravljanja energijom. Praksa upravljanja energijom koja je najrelevantnija za energetska menadžment se bira na osnovu procene kompanije. Kalkulator štednje energije koristi se za procenu potencijalne uštede različitih aktivnosti i mera. Prema završenom rasporedu, sva područja prakse upravljanja energijom trebaju biti pokrivena jednom u kvartalu.
- 6) Unositi projekte u CI-TRAC. Jednom kada se implementiraju prakse za energetska menadžment koji je kvalifikovan i njegov prioritet uspostavljen, one se unose u CI TRAC.
- 7) Praviti mesečne preseke. Kao deo normalne mesečne operativne nadogradnje, tim se sastaje mesečno kako bi pratio napredak u merama za poboljšanje objekta zabeleženih u CI TRAC-u i energetska aktivnostima održavanja. Tim daje pismene izveštaje o napretku sa mesečnim ažuriranjima projekta kako bi obavestio rukovodstvo o tome šta je postignuto tokom proteklog meseca i šta je planirano za naredni mesec.

8) Ponoviti proces. Program za uštedu energije je ireverzibilni i kontinuirani proces koji zahteva podršku putem česte komunikacije i prepoznavanja aktivnosti i dostignuća programa EM. Pored toga, potrebno je ohrabrujuće aktivno učešće celog osoblja postrojenja.

3. UPRAVLJANJE TROŠKOVIMA ENERGIJE

Pod upravljanjem troškovima energijom nalazi se proračun potrebne energije, rukovanje procesima i potrošnjom sa minimalnim gubicima, korišćenje opreme koja je efikasna [5]. Oprema koja se koristi u kompaniji je visokoefikasna, planirana prema standardu i zahtevima JC-a. Jedini način da se utiče je redovno održavanje što se i čini. Veoma važan činilac u upravljanju energijom, korekcija faktora snage, je postavljena u startu na visok nivo od 0,99. Pregledom računa potrošnje energije, ugovorene snage maksigrafa i očitanih vrednosti uočena su četiri meseca u kojima je maksigraf prekoračio dogovorenu isporuku. Prekoračenja ugovorene snage prouzrokuju visoke gubitke jer je u tom slučaju cena električne energije dva puta viša. Nakon pokretanja proizvodnje istovremenim uključivanjem mašina i opreme uz grejanje ili hlađenje prostorija trenutna potrošnja će preći ugovorenu tim više ukoliko su razlike spoljne i radne temperature veće. Iz ovoga se zaključilo da se pokretanje uređaja, mašina i sistema za grejanje i ventilaciju mora pokretati planski i sekvencijalno. Plan i procedura su pripremljeni i standardizovani. Smenovođe i tim održavanja su uključeni u pravilno i koordinisano pokretanje proizvodnje, sistema grejanja, rasvete i sistema vazduha pod pritiskom. Na ovaj način izbegava se istovremeno priključenje svih uređaja na elektrodistributivnu mrežu i prekoračenje ugovorene snage.

4. KOMPRIMOVANI VAZDUH

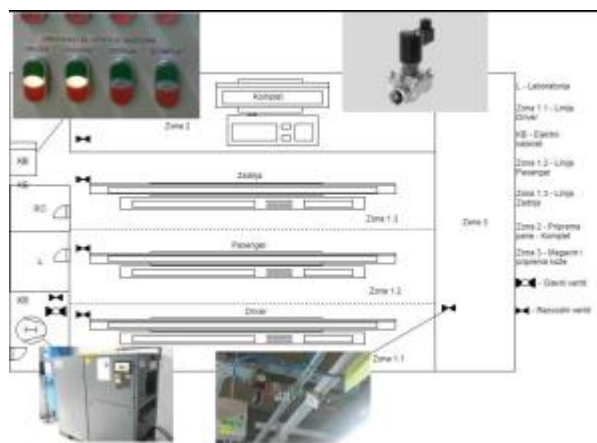
Ušteda energije zbog poboljšanja sistema vazduha može da iznosi od 20 do 50 % ili čak i više. Prva faza provere sistema je detekcija mesta curenja i provera filtera. Zaprljani filteri izazivaju padove vazduha pod pritiskom čime se rad kompresora povećava.

Provera je urađena pomoću dva nezavisna manometra, jedan manometar je povezan na vod vazduha pod pritiskom pre komponente kako bi se očitala vrednost pritiska pre ulaza u određeni element - merač ulaznog pritiska, dok je drugi manometar postavljen na vod vazduha pod pritiskom nakon komponente - merač izlaznog pritiska. Pad pritiska predstavlja razliku dva merena pritiska i značajniji pad je ustanovljen na sedam vodova[5]. Vodovi su obeleženi NOK labelom i uvršteni u plan održavanja.

Problem koji se u sistemima vazduha pod pritiskom povezuje sa značajnim uštedama koje se mogu postići je problem curenja. Procene mogućih ušteda kreću se od 30% do 60%. Standard kompanije Johnson Controls preporučuje proveru curenja ultrazvučnom metodom. Predstavništvo kompanije SMC iz Beograda je uradilo proveru ultrazvučnim uređajem čime je izvršena lokalizacija curenja. Pronađeno je preko 30 curenja u sistemu na crevima, mestima spajanja i ventilima. Tim održavanja vodi evidenciju o mestima curenja, planu popravke i obeležavanju. Pored pomenute provere izvršena je i detekcija curenja starom metodom ispuštanjem mehurića koja je dala još približan broj mesta

curenja jer ultrazvučna metoda nije pouzdana prilikom velikog šuma koji je prisutan u postrojenjima gde se vrši detekcija. Za ovu primenu potreban je deterđent ili sapun pomešan sa vodom čime se oblažu sastavnice i razdelnici sistema vazduha pod pritiskom. Sistem provere curenja je uveden u plan održavanja kao redovna procedura.

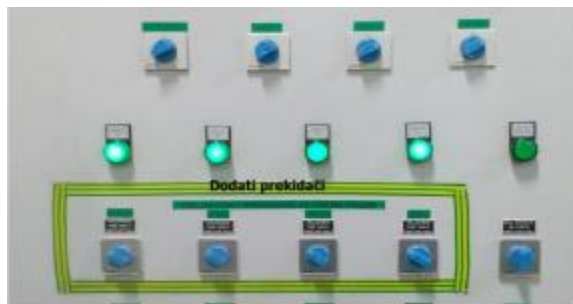
Prepoznata je mogućnost zoniranja proizvodne hale, opreme i prostorija gde se koristi vazduh pod pritiskom (slika 1). Ugradnjom ventila sistem je podeljen tako da je u momentu uključena samo potrebna zona. Odgovorne osobe su zadužene za kontrolu uključivanja-isključivanja sistema prema potrebama. Smanjenjem radnog pritiska u sistemu na optimalnu vrednost postiže se ušteda bez uticaja na performanse rada uređaja. Naime povećani radni pritisak povećava gubitke curenja, kompresor radi većim kapacitetom, a performanse sistema su iste. Kako većina uređaja radi na 6 bara, to se i uzelo kao optimalna mera pritiska u sistemu. Kompresor je podešen na radni pritisak u granicama od 5.8 do 6.7 bara. Uređaji koji rade na manje od šest bara su grupisani i ograničeni regulatorima na odgovarajući nivo pritiska.



isključenja rasvete na radnim mestima koja nisu u upotrebi u datom momentu. Za ovo je bilo potrebno omogućiti zonsko uključenje/isključenje dodatnim prekidačima (slika 2).

Prostor za uštede je prepoznat i kod mogućeg smanjenja nivoa osvetljenosti. Kako je nivo propisanog osvetljenja minimum 700 luksa(lx) izvršeno je merenje intenziteta osvetljenja u kompaniji.

Na mestima gde je nivo svetlosti bio iznad definisanog osvetljenja je redukovano na potrebni nivo redukovanjem broja lampi. Definisane su procedure uključivanja-isključenja delova rasvete u skladu sa potrebama, a odgovorne osobe su imenovane za izvršenje.



Slika 2. Dodati prekidači

6. HVAC

Sistemi grejanja, ventilacije i klimatizacije (HVAC) kontrolišu temperaturu, vlažnost i kvalitet vazduha kako bi osigurali ugodnu okolinu za ljude i pogodnu okolinu za rad opreme. U JC-u HVAC sistem je decentralizovan što daje mogućnosti za prilagođavanje potrebama svakog radnog mesta. Ovo nam govori da kontrolu nad sistemom treba ograničiti na isključivo odgovorne osobe.

Dogovorena su tri koraka kao mogućnost ušteda: ograničenje radne temperature i isključenje tokom neradnih dana, redovan pregled i održavanje sistema i uključivanje svih zaposlenih u prakse čuvanja energije.

Mogućnost ograničavanja radne temperature. S obzirom da je propisana radna temperatura između 18 i 25 stepeni celzijusa regulatori grejanja su ograničeni na 21 stepen celzijusa što je optimalna radna temperatura tako da je izbegnuto subjektivno podešavanje radnih prostorija.

Osoba zadužena za kontrolu je menadžer zaštite na radu kao osoba odgovorna za kontrolu i implementaciju dogovorenog. Svaka promena radnih uslova se tako može uraditi samo uz njegovu saglasnost.

S obzirom da kompanija posluje u tri smene vođa projekta uštede HVAC sistema je dužan da obuču odgovornu osobu za svaku smenu o praćenju poštovanja dogovorenih praksi.

Čišćenje i pregled sistema grejanja kako bi se izbegli dodatni gubici je obuhvaćeno preventivnim održavanjem ali kako pukotine ili oštećenja u zidu i na fasadi zgrade mogu prouzrokovati povećane troškove energije, procedure preventivnog i prediktivnog održavanja zgrade su poboljšane. Takođe kako vrata ili prozori s pukotinom veličine 3 mm mogu propustiti vazduha koliko i rupa u zidu veličine cigle proveru navedenog je ubačena u akcije održavanja. U okviru programa podizanja svesti i aktivnog uključivanja svih zaposlenih prezentovane su

mere zaposlenima, a baneri su postavljeni na potrebnim mestima.

7. SVEST I ANGAŽMAN ZAPOSLENIH

Prakse koje mogu pomoći za efikasan program podizanja svesti zaposlenih i koje su implementirane i postavljene kao deo standardnih radnih aktivnosti su:

- Postavljajte pitanje o svesti na svakom mesečnom sastanku,
- Koristi medije kao što su monitori za podršku programima,
- Koristi postere i banere,
- Pratiti napredak i deliti rezultate,
- Prepoznati i nagraditi dobro ponašanje.

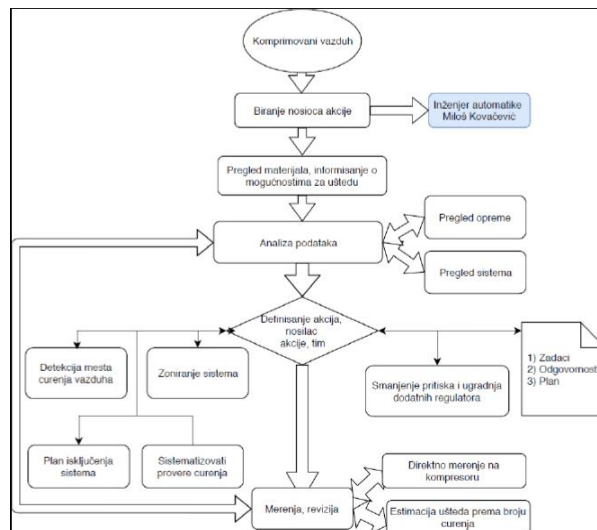
8. MERENJE I PRAĆENJE

Sva merenja potrošnje su merena kroz merače električne energije i gasa na taj način što su grupisane određene zone i na njima se direktno vršilo potrebno merenje potrošnje i uštede. Svi rezultati su praćeni kroz CI TRACK procedure koje su neophodne u procesu implementacije standarda za uštedu energije [6]. Sve izmerene i proračunate vrednosti potrošnje električne energije po tipu uređaja i zonama je potrebno uneti u tabelu(dodatak 1) za evidenciju potrošača kako bi se izračunali ukupni troškovi potrošnje energije. Pomenuta tabela je i parametar za merenje, praćenje i evaluaciju preduzetih koraka u menadžmentu uštede energije. Tabela se pravi iznova svake godine nakon čega imamo jasnu sliku potrošnje energije.

Merenje i praćenje se obavlja pre implementacije praksi gde je to moguće i nakon implementacije, rezultati se zatim upoređuju i oni će služiti za analizu, kako bi se ocenile efikasnosti ugrađenih praksi.

9. REZULTATI I ANALIZA

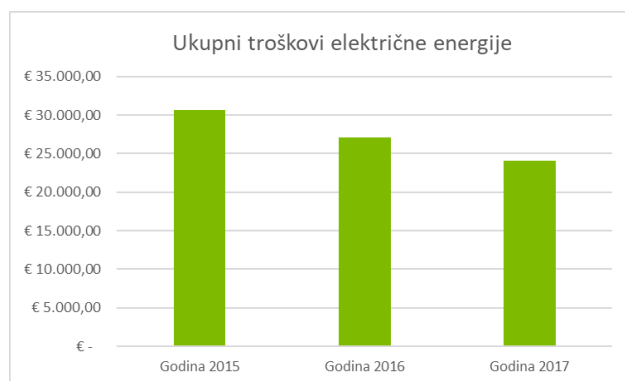
Nakon primene akcija i merenja pristupa se reviziji rezultata i korekciji praksi ukoliko je to potrebno. Na ovaj način se proverava efikasnost primenjenih mera i doslednost u izvršavanju dogovorenog. Na slici 3 je grafik procesa implementacije komprimovanog vazduha odakle se vidi da je proces analize i korekcije dvosmeran.



Slika 3. Grafik implementacije praksi ušteda u sistemu komprimovanog vazduha

Na priloženom grafiku (slika 4) prikazano je koliko je smanjenje potrošnje električne energije u toku 2016 i 2017 godine nakon primene EnMS u odnosu na 2015 godinu pre primene mera štednje. Takođe preduzete mere i definisane procedure za plansko pokretanje opreme tokom 2016 i 2017 godine su rezultovale održavanje maksimalne snage u definisanim granicama, prekoračenja ugovorene snage nije bilo. Energetski pokazatelji zavisni od broja radnih dana i obima proizvodnje su najrealniji pokazatelj efikasnosti primenjenih akcija.

EP su izračunati na osnovu potrošnje energije po danu proizvodnje i po jedinici proizvoda. Oni pokazuju tendenciju smanjenja energije tokom 2016 i 2017 godine što govori u prilog uspešnosti praktične implementacije ISO 50001 standarda.



Slika 5. Rezultati pre i nakon primene EnMS

10. ZAKLJUČAK

Nema potrebe govoriti o važnosti energije, jer je poznato da je energija najveći ulaz u sve sisteme, ne samo industrijske proizvodne sisteme. Industrija je najvažniji sektor, koji utiče na čitavu ekonomiju. Potencijal za poboljšanje energetske efikasnosti kroz usvajanje praksi menadžmenta energije zavisi od veličine firme, vrste proizvodnje, energetskog intenziteta, stepena automatizacije proizvodnje, itd.

Standard ISO 50001 specificira zahteve za implementaciju sistema menadžmenta energije. Namena standarda je da omogući organizacijama da uspostave procese neophodne za poboljšanje energetske performanse. Energetska performansa obuhvata energetske efikasnost, korišćenje energije ili neki drugi indikator. Primena standarda vodi ka smanjenju uticaja na životnu sredinu, ali i troškova za energiju. Može se primeniti u svim vrstama organizacija i u organizacijama svih veličina. Dakle, standard je po prirodi generički, kao i ostali standardi za sisteme menadžmenta, sa kojima ga je moguće integrisati. Može se reći da predstavlja razradu jednog specifičnog i važnog aspekta životne sredine, a to je svakako energija.

Cilj analize potrošnje energije u predstavljenoj studiji slučaja je utvrđivanje indikativnih pokazatelja koji predstavljaju polaznu osnovu za preuzimanje odgovarajućih mera na smanjenju troškova električne energije kao i dobra podloga za kreiranje strategije energetskog razvoja. Strategija u energetskom razvoju JC ogleda se u prepoznavanju sopstvenih propusta i praksi koje su najprimenjivije u okviru kompanije.

Kao prioritetan operativni zadatak nameće se smanjenje troškova električne energije kroz eliminaciju gubitaka kako pomoću tehničkih rešenja ali i organizacionih koje mogu čak i premašiti uštede tehničkim sredstvima. Na kraju cilj je da svaki zaposlen ima svest o važnosti ušteda u energiji i da aktivno doprinosi programu.

Prema istraživanjima sistemski pristup upravljanja energetskom efikasnošću u većini kompanija nije primenjen, i u većini preduzeća postoje samo njegovi obrisi. Energetska politika se uglavnom ne definiše, korišćenje energije se planira sporadično, regulativa je ali standardi ređe. U kompaniji JC proces uspostavljanja praksi za upravljanje energijom je trajao godinu dana uz aktivno angažovanje menadžmenta i nosioca akcija. To nam kazuje da je implementaciju ovog standarda potrebno shvatiti ozbiljno i da samim postavljanjem sistema i promenama se završena samo jedna faza.

Sistem je potrebno unapređivati i održavati, za sve propisane i dogovorene akcije je potrebno imati odgovarajuću dokumentaciju i pisane procedure koje su deo obaveznih radnih rutina zaposlenih jer samo obučeni zaposleni sa tačno propisanim instrukcijama mogu biti garant da će se izvršavati propisane akcije.

Treba napomenuti da proces implementacije nije trajan, to znači da energetski lov treba da bude ugrađen u normalne operativne rutine i kulturu. To je tekući, iterativni proces i uvek postoji prostor za uštedu.

Konačan cilj jeste uključiti svakog zaposlenog da stekne svest o važnosti uštede energije i poštovanju procedura kako bi cela zajednica imala višestruku korist.

11. LITERATURA

- [1] Jovanović, B. (2016). Sistem menadžmenta energije u prerađivačkoj industriji u Srbiji (doktorska disertacija). Beograd: Fakultet organizacionih nauka, str. 13.
- [2] Gordić, D. et al. (2010). Development of energy management system - Case study of Serbian car manufacturer. *Energy Conversion and Management*, 51(12), 2783-2790.
- [3] (2012). *Standard SRPS ISO 50001:2012*. "Službeni Glasnik RS", br. 80/12.
- [4] Johnson Controls document "AE-LOS-FR-151 - Evaluation_of_Energy_Aspects"
- [5] Milenković I. (2014). Kvalitet vazduha pod pritiskom u funkciji održive proizvodnje (doktorska disertacija)
- [6] Johnson Controls document "Energy Hunt - Master Calculator (002) "

Kratka biografija:



Miloš Kovačević rođen je u Loznici 1990. godine. Diplomski-master rad odbranio je 2018. godine na Fakultetu tehničkih nauka iz oblasti mehatronika - robotika.

IMPLEMENTACIJA BIKONA U POSLOVANJE KOMPANIJE IMPLEMENTATION OF BEACONS IN COMPANY'S BUSINESS

Sandra Veljkov, *Fakultet tehničkih nauka, Novi Sad*

Oblast – INŽENJERSTVO INFORMACIONIH SISTEMA

Kratak sadržaj – Cilj ovog master rada jeste istraživanje, modelovanje i implementacija bikona u poslovanje kompanije.

Ključne reči: Veb programiranje, marketing, bikon, API, modelovanje

Abstract – The aim of this master thesis is to model and apply beacons to the company's business, using Google API.

Keywords: Web development, marketing, beacons, API

1. UVOD

Informacione tehnologije se nalaze u svim sferama čovekovog života i skoro je pa nemoguće zamisliti svakodnevnicu bez upotrebe računara ili pametnog telefona.

Kako je u današnjem društvu prisutan svakodnevni rast i razvoj tehnologije, biti kompjuterski nepismen je skoro pa neprihvatljivo; takva osoba nije u mogućnosti da učestvuje u modernom društvu i vrlo često da obavlja bitne životne aktivnosti. Tehnologija nam omogućava da brže i lakše dođemo do potrebnih informacija, štedi vreme i vrlo često i novac.

Upravo zbog ovoga je sve veći akcenat stavljen na moguće načine dolaska do novih potrošača, njihova animacija i prelazak iz grupe "potencijalnih" u grupu "aktivnih", kao i zadržavanje postojećih.

Tema ovog master rada i jeste online marketing, pogotovo jedan njegov alat – bikon (engl. *Beacon*). Fokus istraživanja predstavlja dostupnost informacija na internetu, istraživanje dostupnih tehnologija, projektovanje, testiranje, realizacija i na kraju predstavljanje gotovog projekta klijentima i njihova edukacija.

Na kraju rada je izvršena zaključna analiza, sa benefitima korišćenja bikona, kao i njihovim nedostacima i predlozima za unapređenje.

2. TEORIJSKE OSNOVE

2.1. Pojam i značaj informaciono – komunikacionih tehnologija

Informacione tehnologije se definišu kao dizajn, razvoj izučavanje, implementacija i podrška ili upravljanje računarskim informacionim sistemima, softverskim aplikacijama i hardverom, kako je to definisano od strane Američke asocijacije za informacionu tehnologiju [1].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Darko Stefanović, vanr. prof.

Najvažniju komponentu svakako predstavljaju računari. Informaciona tehnologija (IT) je pojam kojim se opisuje hardver i softver koji omogućavaju pristupanje, preuzimanje, organizovanje i predstavljanje informacija elektronskim putem. Komunikaciona tehnologija je pojam kojim se opisuje telekomunikaciona oprema pomoću koje se informacije šalju, primaju, pretražuju i pristupa im se.

2.2. Pojam i značaj e – marketinga

Upotreba elektronskog poslovanja omogućava konkurentnije poslovanje i povećava šansu opstanka na tržištu, mikromarketing tj. marketing koji se obraća ciljnim segmentima tržišta. Uspostavlja se bolji odnos sa kupcima i razvija njihova lojalnost. Korišćenjem interneta omogućava se jeftini globalni marketing sa velikom bazom potencijalnih kupaca.

Elektronski marketing ili e – marketing predstavlja sve online ili elektronski bazirane aktivnosti koje omogućavaju proizvođačima roba i usluga da zadovolje potrebe i želje svojih kupaca na brz i efikasan način.

Internet danas na poslovnom planu nudi nove puteve komuniciranja sa tržištem. Jedna od prednosti jeste ta što omogućava korisnicima da sami biraju vreme kada će pristupiti veb stranicama, dok na primer, kod televizije postoji vreme emitovanja. To nije beznačajno, jer kada korisnik odluči da poseti određene stranice, velika je verovatnoća da će lakše prihvatiti informacije, čak i one koje mu se nude, a ne samo one koje traži.

2.3. Tehnike e – marketinga

Prilikom izbora tehnike e – marketinga, treba voditi računa o tome šta je cilj nastupa u odnosu na mogućnosti organizacije i odabrati one tehnike online marketinga koje se mogu, prilagođene, iskoristiti za ostvarenje zadatog cilja.

Tehnike online marketinga se dele na one tehnike promocije u sklopu vlastite internet stranice i one izvan vlastite internet stranice.

Jedna od najvažnijih tehnika online marketinga bi svakako bila marketing sadržajem (engl. *Content Marketing*). Kreiranjem sadržaja formira se poruka koja se šalje kroz određeni kanal komunikacije i postaje vidljiva korišćenjem određenih tehnika promocije. Od kvaliteta formirane poruke zavisi razumljivost poruke ciljnoj grupi i samim tim, efekat određene tehnike promocije. Ovo je marketinška tehnika stvaranja i distribucije korisnog, relevantnog i ujednačenog sadržaja s ciljem privlačenja i zadržavanja jasno definisane publike, a koji može neposredno doneti korist u vidu povećanja profita [2].

Ovo je takav tip marketinga koji se zasniva na ideji neometanja korisnika. Proizvodi i usluge se stavljaju u

drugi plan, a glavni fokus je davanje dodatne vrednosti za korisnike kroz sadržaj. Dobar marketing sadržajem jednostavno privlači pažnju ljudi da oni zastanu, čitaju, razmišljaju i ponašaju se drugačije.

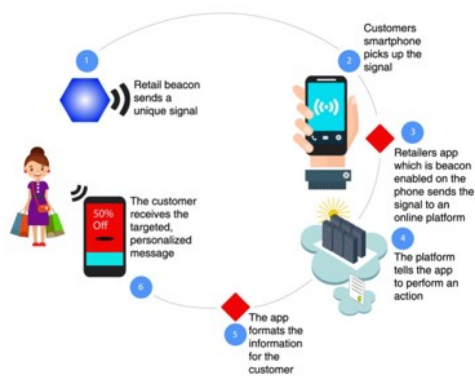
Upravo je bikon takva tehnologija da kompanija svakodnevno može da menja sadržaj bikon poruke, bez puno napora i troškova, u direktnoj interakciji sa korisnikom.

2.4. Uvođenje pojma BIKON

Danas se čini da je najlakše privući potrošače preko kanala koji im je najbitniji i najbliži, a to je mobilni telefon. Potrebno je da se na ekranu mobilnog telefona potrošača pojavi relevantna informacija, koja u pravom momentu i na pravom mestu, izaziva pažnju i poziva na akciju.

Najjednostavnije rečeno, lokacijski marketing (engl. *Proximity marketing*) je poruka poslata korisniku u odnosu na to gde se on nalazi. Funkcioniše tako što locira mobilne telefone putem geofesinga, internet konekcije i *bluetooth* odašiljača. Upotreba je veoma široka, od velikih prodajnih objekata, muzeja, sportskih objekata, aerodroma itd.

Međutim, svoju najveću zastupljenost su našli u maloprodaji. Tako kupac ulaskom u prodavnicu može dobiti kupon popust, informaciju o proizvodima na akciji ili bilo koju poruku koju prodavac želi da mu prikaže. Primer upotrebe bikona u maloprodaji je dat na slici 1.



Slika 1. Princip korišćenja bikona u maloprodaji [3]

3. KORIŠĆENE TEHNOLOGIJE

3.1. Google Bikon platforma (engl. *Google Beacon Platform*)

Interfejs za programiranje aplikacija (engl. *Application Programming Interface – API*) je spisak komandi i njihovih formata koje jedan program razmenjuje sa drugim. Kompanije danas koriste API da dele sadržaj putem kanala (telefoni, kompjuteri, aplikacije itd.) i obezbede interakciju između usluga i poslovnih procesa.

Google API je način da se koriste Google-ovi proizvodi na olakšan način. Postoje Mape API (engl. *Maps API*), Pretraživanje API (engl. *Google Search API*), Bikon API (engl. *Google Beacon API*) itd. U ovom radu je poseban akcenat stavljen na Google Bikon API.

Google Bikon platforma omogućava korisnicima bolje korišćenje lokacije i korisničkog iskustva, preko jakog signala koji stiže do njihovih pametnih mobilnih telefona preko *bluetooth* niske energije (engl. *Bluetooth low energy – BLE*) bikona sa Edistona (engl. *Eddystone*), otvorene bikon platforme sa Googla. Google bikon platforma omogućava da se bikonima upravlja daljinski, integriše se sa Google uslugama i pomoću uređaja korisnicima se otkriva sadržaj i funkcionalnost aplikacije i veba [4].

Lokacijski bikon API (engl. *Proximity Beacon API*) je servis koji omogućava upravljanje BLE bikonima i podacima povezanim sa njima preko *Representational State Transfer (REST)* interfejsa. REST je skup pravila koje programer koristi kada kreira svoj API. Korišćenjem bikon platforme, uz korišćenje Google Proximity Beacon API i Eddystone bikona, korisnik lako može da omogući aplikaciji da komunicira sa korisnikovim telefonom preko bikon poruke (engl. *attachment*), nadgleda status koristeći dijagnostički alat, i koristi prednosti fizikal veba (engl. *Physical Web*).

Bikoni se mogu postaviti na raznim pozicijama, fiksnim ili ne, i jednom kada se postave, treba ih registrovati preko Google Proximity bikona kako bi se dobili kompletni benefiti od Google bikon platforme. Ova platforma čini poveznicu između bikon hardvera i poruka koje su povezane sa njim.

Bikon poruke su povezane sa konkretnim prostorom i tipom (engl. *Namespace and Type*). Ove poruke se čuvaju na klauđu (engl. *Cloud*) i Google Proximity Beacon API omogućava lak pristup i menjanje ovih poruka. Ovo omogućava korisnicima bikona da uvek imaju dostupne nove informacije i eliminiše potrebu za manuelnim, ručnim podešavanjem izmena.

3.1. PHP, JavaScript i PostgreSQL

U ovom projektu su servisi pisani u *Hypertext Preprocessor* – PHP programskom jeziku, a za korisnički interfejs je korišćen JavaScript, *HyperText Markup Language* – HTML, *Cascading Style Sheets* – CSS i JQuery biblioteka. Baza podataka je PostgreSQL.

PHP je servisni skriptni jezik i veoma moćan alat za izradu dinamičkih i interaktivnih veb stranica. PHP je naširoko korišćen, skriptni jezik otvorenog koda. Izvršava se na serveru i besplatan je za korišćenje. PHP fajlovi mogu sadržati tekst, HTML, CSS, JavaScript, PHP kod.

PHP je dovoljno moćan da bude osnova najvećeg bloging sistema na vebu, WordPress i najveće socijalne mreže današnjice, Facebook [5].

JavaScript je objektno zasnovan skriptni jezik. Skriptni je jer se sastoji od serije komandi koje se učitavaju u interpreteru, a da se prethodno ne kompajlira sadržaj, odnosno ne prevodi se u mašinski jezik. Zbog ove karakteristike JavaScript se direktno izvršava na klijentskoj strani. JavaScript je potreban svaki put kada se učita veb stranica koja ne prikazuje samo statički sadržaj. Naravno, JavaScript ima i druge primene, kao npr. detekcija pretraživača (postoji mogućnost prikaza određenog sadržaja posebno po pretraživaču), kontrola prozora, veb kolačići (engl. *Web cookies*), provera sadržaja upisanog od strane korisnika.

CSS daje stilove HTML dokumentu, sa posebnim naglaskom na stil. Dok se HTML koristi da da strukturu veb dokumentu (definiše stvari poput zaglavlja i paragrafa), CSS ga prožima i daje stilove dokumentu, dimenzije stranice, boje, fontove. Prednosti CSS – a su mnogobrojne: štedi vreme, stranice se brže učitavaju, lakše je kasnije održavanje, kompatibilnost sa više različitih uređaja, globalni veb standardi itd.

HTML je opisni jezik pomoću koga se kreiraju sajtovi, tačnije funkcionalnost elemenata jedne veb stranice. Svi sajtovi su kreirani uz pomoć HTML-a i upravo zbog toga se ovaj opisni jezik često smatra da danas predstavlja osnovu kompletnog veba. HTML tagovi predstavljaju određene delove dokumenta.

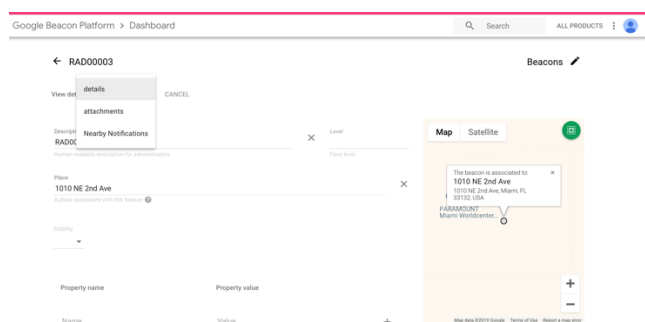
Za dodavanje, pristupanje i procesiranje podataka u bazi podataka potreban je sistem za upravljanje bazama podataka kao što je to PostgreSQL. Ovo je sistem za upravljanje relacijskim bazama podataka. Objektno relacijska baza podataka je ona baza podataka koja podatke čuva u odvojenim tabelama, što daje brzinu i fleksibilnost bazama podataka. Tabele su povezane definisanim relacijama što omogućava njihovo kombinovanje. PostgreSQL je utemeljen na modelu klijent – server.

4. KORIŠĆENE TEHNOLOGIJE

4.1. Opis rada Google Bikon Platforme

Nakon istraživanja opcija, rešeno je da se implementacija bikona odvija preko Google Bikon platforme i njihovog API-a. Nakon pribavljanja potrebnih bikona, potrebno je izvršiti registraciju i to pre neke od aplikacija koje mogu da se nađu u prodavnicama za mobilne telefone (za Android i za iOS) koja se poveže sa Google Bikon platformom, odabere se željeni projekat i izaberu bikoni koji treba da se registruju.

Registrovani bikoni se potom prikazu na Google Bikon platformi, tačnije na njenom *dashboard*-u. Klikom na određeni bikon se se prikazuju informacije za taj bikon, kao što su lokacija, baterija, status, lista njegovih poruka, sa mogućnostima dodavanja novih, podešavanja vremena emitovanja poruka itd. Takođe, postoji mogućnost i podešavanja sprata na kom se bikon nalazi. Prikaz informacija za jedan bikon u Googlovom dashboardu je dat na slici 2.



Slika 2. Prikaz informacija za jedan bikon

Nakon kreiranja poruke, i ukoliko je bikon aktivan, poruka će biti vidljiva na telefonu. Da bi poruka bila vidljiva na telefonu, potrebno je da bluetooth bude uključen, mobilni telefon bude konektovan na internet

(wifi ili mobilni podaci), bude uključena lokacija i obaveštenja u blizini (engl. *Nearby Notifications*).

Kako se kompanija za koju se i pravio ovaj projekat zapravo kompanija koja se bavi digitalnim marketingom, izuzetno je nepraktično klijentima davati pristup Googlovoj bikon platformi, koji zapravo i nije korisnički prilagođen interfejs za osobe koje nemaju programersko obrazovanje. Iz ovog razloga je omogućen pristup bikonima, kreiranje i brisanje poruka omogućen preko aplikacije same kompanije, kojoj svi njeni klijenti imaju pristup i profil. Opis modelovanja je dat u nastavku.

4.2. Opis rada aplikacije

Najpre se pristupilo pravljenju stranice, tzv. Master admin, na kojoj je prikaz svih bikona, kao i mogućnosti dodeljivanja bikona klijentima. Klijent, preko svog pristupa aplikaciji nema pristup ovoj stranici, on je omogućen samo sa master loginom. Prikaz stranice je dat na slici 3.

Device name	Device ID	Assigned account	Account Manager	Status	Battery level	Traffic	Clicks	Adopt
M01	beacons/311	Account1		Active	100	0	0	Adopted
M02	beacons/12	Account2		Active	100	199	0	Adopted
M03	beacons/456	Account3		Active	100	485	0	Adopted
M04	beacons/890	Account3		Active	100	0	0	Adopted
M05	beacons/789	Account3		Active	100	1028	0	Adopted
M06	beacons/985	Account3		Active	100	150	0	Adopted
M07	beacons/197	Account4		Active	100	80	0	Adopted
M08	beacons/222	Account33		Active	100	0	0	Adopted

Slika 3. Bikon admin stranica

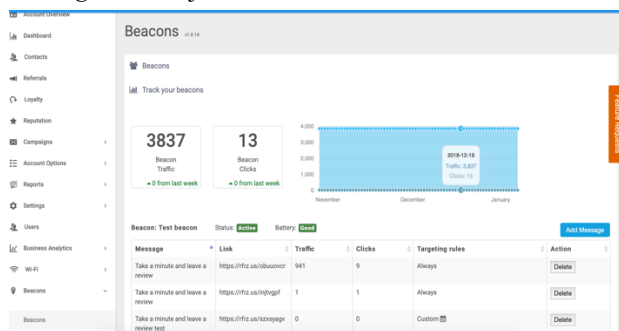
Kada je omogućeno dugme *Adopt*, taj bikon može da se dodeli klijentu preko popup prozora sa listom svih klijenata.

Svaki klijent kojem je dodeljen bikon može njemu da pristupi izborom opcije Bikon u meniju. Otvara se stranica sa statističkim podacima, grafikonom koji prikazuje broj klikova i proteklog saobraćaja, za sve bikone tog klijenta. Svaka poruka koja je sačuvana za bikon, sistem Google proveriti taj link da nije *spam* i ovo se prikaže kao podatak proteklog saobraćaja preko bikona.

Svaki bikon koji je u posedu tog klijenta se prikazan u zasebnoj tabeli; iznad tabele su prikazani podaci o statusu bikona i stanju baterije, a u tabeli podaci o porukama koje postoje za taj bikon. Ti podaci podrazumevaju naslov poruke, link, broj klikova i proteklog saobraćaja, pregled vremenskog okvira za prikaz poruke (ukoliko on postoji, ukoliko ne postoji, biće postavljen na Uvek (engl. *Always*)) i mogućnost brisanja poruke. Ne postoji mogućnost izmene poruke jer to nije moguće ni u Google bikon platformi. Prikaz bikon stranice je dat na slici 4.

Dodavanje nove poruke se vrši klikom na dugme *Add Message*. Otvara se nova stranica za dodavanje poruke. Programirano je tako da su odmah ponuđene 3 opcije za poruku, a to su: Otvaranje vaučera (koji taj klijent ima već kreiran), potom otvaranje stranice za ostavljanje sugestija, primedbi, komentara i treća opcija je proizvoljni link sa tekstualnim poljem za upis tog linka. Novo pravilo za vremenski prikaz se podešava klikom na dugme *Set new targeting rule*.

Čuvanje poruke se vrši klikom na dugme *Add Beacon Message*. Prikaz je dat na slici 5.



Slika 4. Bikon stranica

Slika 5. Dodavanje nove poruke

4.3. Poteškoće pri modelovanju

Ceo proces modelovanja je protekao bez velikih poteškoća. Google API dokumentacija je dosta dobro objašnjena, sa spiskom funkcija, ulaznih parametara i tipova podataka koje vraća kao rezultat. Prilikom čuvanja nove poruke bikona bilo je potrebno pozvati funkcije `json_encode()` i `base64_encode()`, pre poziva API funkcije `create()`, kako bi niz od podataka koji čine poruku mogao da se sačuva.

Takođe, u početku je bilo nejasno na koji način Google vrši redirekciju linkova u porukama, pa se broj proteklog saobraćaja zapravo predstavljao kao broj klikova. Međutim, kako je pravilo kompanije da sve čuva kao kratki link (engl. *Short link*), ubačen je još jedan parametar (bcn parametar), koji nakon redirekcije sa short linka, upisuje upitom u bazu da je postojao klik na tu poruku. Googlevo proveravanje validacije linkova proverava samo short link, ne čeka da se uradi cela redirekcija.

5. ZNAČAJ IMPLEMENTACIJE BIKONA

Poruke koje bikon šalje se šalju preko Nearby obaveštenja. Ova obaveštenja se prikazuju kada korisnik Android telefona prevuče po ekranu ka dole. Izgledaju kao redovno obaveštenje, ali su samo vidljiva kada je telefon u dometu bikona koji emituje poruku. Ove poruke su predviđene da postoje na Android telefonima. Poruke bikona se ne smatraju *spamom* jer su deo Android operativnog sistema.

Važno je napomenuti da i telefon neće zvoniti ni vibrirati kada primi ovakvu poruku. Ona će jedino biti vidljiva kada se telefon otključa i prevuče se prstom po ekranu.

Statistika pokazuje da prosečan korisnik ovo uradi 40 puta na dan [6].

Jasna ograničenja su odmah vidljiva, jer, kao što je već napomenuto, Nearby obaveštenja nije moguće realizovati na iOS operativnim sistemima. Postoje neki bikoni koji podržavaju i iOS, ali ovim projektom nisu obuhvaćani. Takođe, kako bluetooth mora stalno da bude upaljen, ovo značajno smanjuje trajanje baterije mobilnog telefona.

Dosta korisnika misli i da bikoni nisu pouzdani i da imaju pristup informacijama koja se nalaze u telefonu. Međutim, ovo naravno nije istina. Jedina interakcija bikona sa telefonom je preko Nearby aplikacije.

Treba i posebnu pažnju posvetiti pozicioniranju bikona u prostoru; armature, cigla, fontane, skulpture, mogu značajno da utiču na smanjenje performansi bikona. Neka od preporuka je da budu na 3 metra visine.

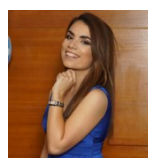
6. ZAKLJUČAK

Za doba u kojem živimo je karakteristična brza promena okruženja, uslova, proizvoda i navika, i svakodnevno se dolazi do novih načina pridobijanja tržišta, pa tako se pojavio i bikon. Bikon je mali radio transmitter koji se pozicionira u prostoru i emituje poruke koje su vidljive osobama koje imaju pametne mobilne telefone sa Android operativnim sistemom, uključen Bluetooth, internet konekciju i Nearby obaveštenja; kada se nađu u dometu bikona. Benefiti koje bikoni donose u poslovanju su mnogobrojni, od mogućnosti direktnog kontakta sa potrošačem, praćenje učestalosti poseta, efikasnost marketing kampanja, povratne informacije klijenata, aktuelnost kupovine itd. Glavni nedostatak ovih bikona je što vrše interakciju samo sa Android operativnim sistemima, što ostavlja veliki deo tržišta nepokriven. Takođe, moraju da se ispune svi uslovi da bi bikon bio vidljiv na telefonu. Iz svega ovoga se može zaključiti da bikoni jesu inovacija, imaju mnoge benefite, ali ne garantuju uspeh u 100% slučajeva.

7. LITERATURA

- [1] <https://techspirited.com/what-is-information-technology> (pristupljeno u decembru 2018.)
- [2] <http://www.draganvaragic.com/blog/najvazniji-kanali-komunikacije-i-tehnike-promocije> (pristupljeno u decembru 2018.)
- [3] <https://www.happiestminds.com/Insights/Beacons-technology> (pristupljeno u decembru 2018.)
- [4] <https://developers.google.com/beacons/> (pristupljeno u januaru 2019.)
- [5] <https://www.w3schools.com> (Pristupljeno u decembru 2018)
- [6] <http://support.referrizer.com> (Pristupljeno u januaru 2019)

Kratka biografija:



Sandra Veljkov rođena je u Novom Sadu 1989.godine. Master rad na Fakultetu tehničkih nauka iz oblasti Inženjerstvo informacionih sistema odbranila je 2019.godine.
kontakt: sveljkovs@gmail.com

SISTEM ZA AUTOMATIZACIJU PROCESA IZRADE LIČNIH DOKUMENATA**SYSTEM FOR AUTOMATIZATION PROCESS OF ISSUING PERSONAL DOCUMENTS**Dordo Kalaković, *Fakultet tehničkih nauka, Novi Sad***Oblast: INŽENJERSTVO INFORMACIONIH SISTEMA**

Kratak sadržaj - U modernom dobu jedan od najbitnijih resursa postalo je vreme. Mnoge tehnologije, alati, procedure uvođeni su ne bi li se uspelo uštedeti na vremenu. Ušteda ovog resursa bila je osnovni motiv nastanka aplikacije opisane u daljem radu. Aplikacija je napisana u programskom jeziku C# i napravljena je da omogući jednostavnije apliciranje za neki od ličnih dokumenata. Korisnik aplikacije bi mogao uvek da ima uvid na vreme čekanja u nekoj od ustanova, kao i na vreme čekanja na zahtev, za koji je prethodno aplicirao.

Ključne reči – Visual Studio; Microsoft SQL Server; MVC; Twilio; WCF service; Windows service;

Abstract - In recent years, time has become our most valuable resource. Many technologies, tools and procedures have been implemented in order to save certain amount of time. The same motive has been the main catalyst behind the software application described in this paper. This software has been written in C# programming language and it is developed to simplify the process of application for obtaining personal documents. By using the application, user can have insight into waiting time in the appropriate institution, as well as the waiting time for the document request which user previously made.

Keywords – Visual Studio; Microsoft SQL Server; MVC; Twilio; WCF service; Windows service

1. UVOD

Danas je C# jedan od najpopularnijih programskih jezika. Koristi se od strane programera širom sveta. C# je razvijen od strane Microsoft kompanije, kao moderna platforma za razvoj .NET aplikacija.

C# jezik se distribuira zajedno sa *Common Language Runtime* (CLR), specijalnim okruženjem na kojem se izvršava. Ovo okruženje je deo platforme .NET Framework (eng. okvir) koji uključuje CLR, skup standardnih biblioteka, kompajler, debugger i ostale razvojne alate [1].

ASP.NET MVC predstavlja jednostavan framework za razvoj Web aplikacija u Microsoft .NET okruženju. Kada ljudi upotrebe akronim MVC najčešće misle na design pattern (eng. dizajn obrasce). Model-View-Controller je dizajn obrazac koji omogućava razdvajanje prikaza, podataka, kao i kontrole i tokove upravljanja. View predstavlja prikaz na ekranu i ono što korisnik vidi.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor dr Srđan Sladojević, docent.

Controller upravlja unesenim podacima na View. Model predstavlja strukturu podataka koja se prenosi između prethodna dva navedena. Dolazi se do zaključka da obrazac pomaže razdvajanju prikaza, interakcije i podataka.

U ASP.NET MVC obrazac je malo drugačiji. Svaki zahtev uslužuje Controller. Zatim Controller vrši neku izmenu nad modelom i aktivira View za prikaz. U ASP.NET MVC View predstavlja .aspx stranice koje sadrže HTML elemente [2]. Model mogu biti podaci iz baze podataka, što je i slučaj u aplikaciji koja se upisuje u radu.

Aplikacija ima podršku logovanja. Logovanje se startuje odmah prilikom startovanja aplikacije i kreira se na zadatoj lokaciji, a ime log fajla se kreira po sledećem obrascu {Log}{dan}{mesec}{godina}{milisekunde}. Kako bi se fajl kreirao samo jednom i sve vreme se radilo sa istim fajlom upotrebljen je i Singleton obrazac.

Singleton obrazac se koristi kada treba da postoji samo jedna instanca klase i mora biti pristupačna klijentu kroz poznatu tačku pristupa. Takođe, obrazac se koristi kada instanca klase trebalo da bude proširiva, a klijenti bi trebali biti u mogućnosti da koriste proširenu instancu bez izmene sopstvenog koda [3].

2. PRIMENJENE SOFTVERSKJE TEHNOLOGIJE

U radu su se koristile sledeće tehnologije:

.Net razvojni okvir predstavlja razvojnu programsku platformu opšte namene. Microsoft kompanija je započela razvoj ove platforme krajem 1990-te godine, a prva beta verzija .Net 1.0 je puštena krajem 2000-te godine. Od početka razvoja, pa sve do danas Microsoft je usavršavao i otpuštao različite verzije .Net framework-a. Ovaj framework se koristi isključivo na Windows operativnim sistemima, a minimalna verzija je Windows 7 Service Pack 1. Glavne karakteristike .Net framework-a su da podržava više programskih jezika, asinhrono i konkurentno programiranje modela i interoperabilnost koja omogućava širok spektar scenarija kroz više platformi [4].

Microsoft SQL Server je sistem za upravljanje relacionim bazama podataka u vlasništvu kompanije Microsoft. Kao server za baze podataka, ovo je softverski proizvod sa primarnom namenom za skladištenje i konzumiranje podataka preko zahteva, najčešće upućenih od strane drugih aplikacija, koje se mogu nalaziti na istom računaru ili na drugom računaru, kada se interakcija odvija putem Interneta. SQL Server Management Studio je alat za korisničku interakciju sa Microsoft SQL Server-om. Služi za konfigurisanje, upravljanje i administraciju svih komponenti unutar Microsoft SQL Server-a. Ovaj alat ima ugrađen editor za skripte i grafičke alate koji rade sa objektima i osobinama koje nudi server.

Windows Communication Foundation (WCF) predstavlja deo .NET softverskog okruženja koji je namenjen za razvoj i upravljanje servisima. WCF je u osnovi Microsoft-ova implementacija skupa industrijskih standarda koji definišu interakciju između servisa, konverziju tipova, kao i upravljanje različitim protokolima što omogućava WCF-u da podrži interoperabilnost između servisa. Kako bi WCF servis bio funkcionalan i pristupačan neophodno je da bude hostovan u odgovarajućem okruženju, koje je najčešće realizovano u okviru nekog procesa [5].

Windows service predstavlja osnovnu komponentu Microsoft Windows operativnog sistema i omogućava kreiranje i upravljanje dugotrajnim procesima. Za razliku od običnih programa koji su pokrenuti od strane korisnika i jedino se izvršavaju kada je korisnik ulogovan, Windows servisi se mogu startovati bez intervencije korisnika i mogu trajati mnogo duže nakon što se korisnik izloguje. Servis radi u pozadini i obično se startuju kada se mašina pokrene. Usluge upravljaju širokom paletom funkcija uključujući mrežu, zvukove, boje, praviti rezervu podataka i mnoge druge [6].

U aplikaciji windows servis služi za slanje sms poruka ukoliko vreme čekanja na neki zahtev bude manje od 20 minuta. Servis će iz baze pretražiti vreme apliciranja svakog neizvršenog zahteva, uporediti da li je vreme čekanja na zadati zahtev manje od 20 minuta. Ukoliko je vreme čekanja manje od 20 minuta servis će poslati poruku obaveštenja na broj telefona podnosioca zahteva. U bazi podataka će se polje *IsNotify* podesiti na true, kako taj broj više ne bi bio obaveštavan. Servis nakon toga pravi minut pauze.

Twilio predstavlja razvojnu platformu za komuniciranje. Softverski timovi koriste Twilio za dodavanje zvuka, videa, razmene poruka u njihovim aplikacijama. U pozadini Twilio Api-a je Super Network, softverski sloj koji povezuje i optimizuje komunikacione mreže širom sveta. Ovo omogućava korisnicima da pouzdano pozovu i šalju poruke bilo gde u svetu [7].

Kako bi iskoristili pogodnosti slanja poruka, potrebno je prvo registrovati se na sajtu ove platforme. Nakon registracije se unosi broj telefona, na koji će stići kod za potvrdu. Potrebno je podesiti zemlju u kojoj se dati broj nalazi kao i pozivni broj. Dobija se broj koji će se koristiti u aplikaciji kao broj telefona, API i token. API i token izgledaju kao *hash* (mešavina), a služe kako bi se aplikacija povezala sa profilom koji je registrovan i kreiran na sajtu.

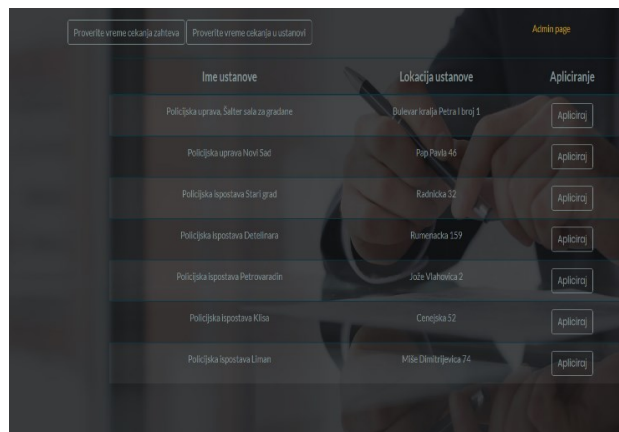
3. FUNKCIONALNOSTI

Osnovna funkcionalnost aplikacije jeste da korisnik preko Web-a aplicira za izradu nekog od ličnih dokumenata u određenoj ustanovi. Na početnoj stranici korisniku su navedene sve ustanove u kojima može da aplicira. Korisnik ima mogućnost da vidi tabelu u kojoj je navedeno ime ustanove kao i njena lokacija. Poslednja kolona sadrži dugme koje će klikom korisnika odvesti na stranicu za dodavanje zahteva.

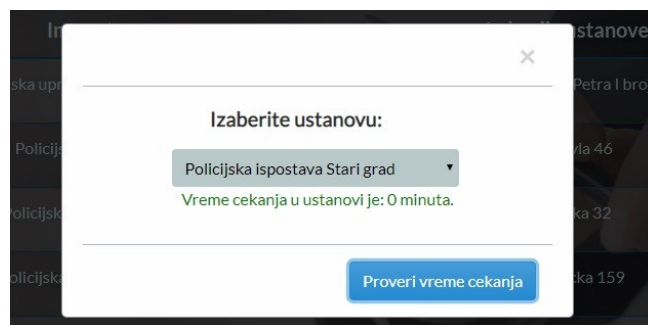
Početna stranica se može videti na slici 1. Pre nego što kreira zahtev korisnik može da proveri da li u ustanovi, u kojoj želi da aplicira ima gužve i da li postoji određeno

čekanje. Ovim se želelo postići da korisnik ne dođe u situaciju da aplicira a da je pritom odmah sledeći na redu. Na početnoj stranici postoji dugme sa nazivom „proverite vreme čekanja u ustanovi”.

Klikom na ovo dugme pojaviće se modal na kom korisnik ima mogućnost da izabere ustanovu u kojoj želi da proveri vreme čekanja. Klikom na dugme „proveri vreme čekanja” pojaviće se labela sa ispisom vremena čekanja. Prikaz na slici 2.



Slika 1: Index stranica



Slika 2: Provera vremena čekanja u ustanovi

Kada korisnik izabere ustanovu u kojoj želi da aplicira prikazaće se stranica za dodavanje zahteva. Korisnik popunjava svoje lične podatke na formi. Kada korisnik pritisne dugme za kreiranje zahteva automatski će se u bazu podataka, u posebnu tabelu dodati i korisnik sa svojim podacima, gde će JMBG biti identifikator. Identifikator zahteva biće identifikator izabranog dokumenta kao i JMBG korisnika.

Ovim se želelo postići da jedan korisnik ne može da aplicira za jedan dokument više puta. Uvek postoji mogućnost gubitka nekog od dokumenata tako da korisnik ima pravo da aplicira na neki dokument ponovo, ali tek nakon što je aplicirani zahtev izvršen. U svakom trenutku, pre nego što kreira zahtev, korisnik može da se vrati na prethodnu stranicu.

Obavezna polja koja korisnik treba da popuni su obojena crvenom bojom. Ukoliko ne unese neki podatak, ili neki unos nije validan, prikazaće se određene validacione poruke. Slika forme za dodavanje zahteva nalazi se na slici 3. Na slici 4. su prikazane validacione poruke.

Slika 3: Forma za dodavanje zahteva

Slika 4: Validacija na formi za dodavanje zahteva

Nakon što je zahtev kreiran, na broj telefona i na email, ukoliko ga je korisnik popunio, treba da stigne poruka obaveštenja. U poruci se nalazi identifikacion broj zahteva, a poruka sadrži i informacije gde je aplicirano i za koji dokument. Primer poruke koja je u formatu sms poruke se nalazi na slici 5.

Postovani Djordjo,
Uspesno ste kreirali zahtev u ustanovi SUP Zmaj Ognjena Vuka, za izradu dokumenta: Vozacka dozvola. Identifikacioni broj Vaseg zahteva je [23757171](#). Na sajtu mozete proveriti potencijalno vreme cekanja za Vas zahtev. Srdacan pozdrav, Adminstracija

Slika 5: SMS obaveštenje za kreiran zahtev

Nakon dobijanja identifikacionog zahteva korisnik u svakom trenutku može da ima uvid u vreme čekanja na dati zahtev. Na početnoj stranici se nalazi dugme pod nazivom „proverite vreme čekanja zahteva”. Klikom na pomenuto dugme pojavljuje se modal gde korisnik ima mogućnost da unese identifikacioni broj zahteva. Validan unos zahteva će prikazati labelu sa ispisom vremena čekanja na zahtev kao i broj osoba na čekanju. Nevalidan unos će prikazati validacionu poruku da je potrebno uneti postojeći zahtev. Unos identifikacionog broja zahteva koji je izvršen će prikazati poruku da je zahtev već izvršen. Prikaz validne poruke na slici 6.

Slika 6: Prikaz vremena čekanja na zahtev

Aplikacija je kreirana kako bi pružila podršku i ustanovama za obradu zahteva. Na početnoj stranici se nalazi dugme sa nazivom „admin page”. Klikom na pomenuto dugme pojavljuje se modal koji služi za login (eng. prijavu).

Slika modala za prijavu može se videti na slici 7. Korisnik unosi korisničko ime i šifru nakon čeka potvrđuje unos. Nakon toga se prikazuje stranica za obradu zahteva. Korisnik je prijavljen kao radnik u samo jednoj ustanovi i može da obrađuje zahteve samo iz te ustanove.

U desnom uglu je prikazana notifikacija dobrodošlice prijavljenom korisniku. Notifikacija se isključuje klikom bilo gde na formi. Istog momenta kako se korisnik prijavio tada je izvršen (promenjen status u bazi podataka na izvršen) prvi zahtev koji je bio na čekanju. Informacije o zahtevu se nalaze na formi. Forma za obradu zahteva se može videti na slici 8.

Ukoliko nema zahteva prikazaće se poruka da nema zahteva na čekanju. Dugme na kome je pisalo „admin page” nakon prijave ima prikazano ime prijavljenog zaposlenog, sve dok se isti ne odjavi. Odjavljivanje se vrši tako što korisnik klikne na dugme „logout” na stranici za obradu zahteva.

Nakon toga biće preusmeren na početnu stranicu. Forma za obradu zahteva, gde je ispisano da nema zahteva na čekanju, se može videti na slici 9.

Slika 7: Prijava za obradu zahteva

Slika 8: Prikaz informacija o zahtevu na formi

Slika 9: Prikaz stranice za obradu zahteva kada nema zahteva na čekanju

4. ZAKLJUČAK

Svima se bar jednom desilo da se našao u gužvama prilikom vađenja ličnih dokumenata. Aplikacija može biti alat putem kog ćemo ovakve probleme rešiti. Sličan alat postoji, gde je moguće zakazati izradu nekog dokumenta nekoliko dana unapred, ali smatram da je ovim softverskim rešenjem obuhvaćena veća celina. Aplikacija pruža dosta benefita. Od prikaza koliki su redovi čekanja u određenim ustanovama, apliciranja za određena dokumenta u ustanovama, do toga da pruža podršku zaposlenima u ustanovama, kako bi izvršavali zahteve po redu, i imali informacije o zahtevu.

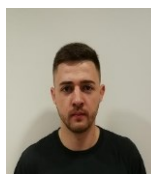
Prilikom prijavljivanja zaposlenog u ustanovi pristupa se stranici za obradu zahteva. Ovo znači da zaposleni može od kuće, tačnije van posla, pristupiti sajtu i obraditi zahtev. Ovo predstavlja jedan od nedostataka aplikacije. Potrebno je uvesti jedan nivo verifikacije, gde će se proveravati sa koje lokacije se zaposleni prijavljuje, kako ne bi bilo dozvoljeno obraditi zahtev van radnog mesta. Svakim pristupom stranici za obradu zahteva obrađuje se novi zahtev koji je na redu. Može se desiti da zaposleni nekada ne želi odmah nakon prijavljivanja startovati obradu zahteva, tako da i ovo predstavlja polje gde bi moglo da se izvrši unapređenje.

Ideja izbegavanja čekanja u redovima se može primeniti i na neke druge grane. Aplikaciju je moguće unaprediti tako da se može aplicati u zdravstvu, kako bi se izbegla ogromna čekanja kod lekara. Veliki redovi su problem u katastru, kao i u nekim drugim ustanovama. Primenom aplikacije se u navedenim institucijama mogu rešiti pomenuti problemi.

5. LITERATURA

- [1] Svetlin Nakov, Veselin Kolev & Co. (2013, Sofia), "Fundamentals of computer programming with C#".
- [2] Jonathan McCracken, The Pragmatic Bookshelf Raleigh „Test-Drive ASP.NET MVC", North Carolina Dallas, Texas.
- [3] Erich Gamma, Richard Helm, Ralph Johnson, John Vissedes, "Design Patterns Elements of Reusable Object-Oriented Software".
- [4] Sempf B., Sphar C., Davis S.R., (2013), "C# 5.0 All - In-One For Dummies", John Wiley & Sons, Inc., Hoboken, NJ, USA.
- [5] Stopper, Markus, and Bernd Gastermann. "Service-oriented communication concept based on WCF .NET for industrial applications." International multi conference of engineers and computer scientists. 2010.
- [6] Stackify , What are Windows Services? How Windows Services Work, Examples, Tutorials and More, dostupno na: <https://stackify.com/what-are-windows-services> [pristupljeno 07. decembar 2018]
- [7] Twilio, dostupno na: <https://www.twilio.com/learn/twilio-101/what-is-twilio> [datum pristupa 07.12.2018]

Kratka biografija:



Dordo Kalaković rođen je 1990. godine u Novom Sadu. Srednju tehničku školu "Mileva Marić" završava 2009. godine, nakon čega svoje obrazovanje nastavlja na Fakultetu tehničkih nauka u Novom Sadu, smer Inženjerski menadžment. Nakon završenih osnovnih akademskih studija 2015. godine, upisuje master akademske studije, program Inženjerstvo informacionih sistema.

U realizaciji Zbornika radova Fakulteta tehničkih nauka u toku 2018. godine učestvovali su sledeći recenzenti:

Aco Antić	Dušan Uzelac	Matija Stipić	Slobodan Dudić
Aleksandar Erdeljan	Duško Bekut	Milan Rapajić	Slobodan Krnjetin
Aleksandar Ristić	Đorđe Ćosić	Milan Simeunović	Slobodan Morača
Bato Kamberović	Đorđe Lađinović	Milan Trifković	Sonja Ristić
Biljana Njegovan	Đorđe Obradović	Milan Trivunić	Srđan Kolaković
Bogdan	Đorđe Vukelić	Milan Vidaković	Srđan Popov
Kuzmanović	Đula Fabian	Milena Krklješ	Srđan Vukmirović
Bojan Batinić	Đura Oros	Milica Kostreš	Staniša Dautović
Bojan Lalić	Đurđica Stojanović	Milica Miličić	Stevan Gostojić
Bojan Tepavčević	Filip Kulić	Mijodrag Milošević	Stevan Milisavljević
Bojana Beronja	Goran Sladić	Milovan Lazarević	Stevan Stankovski
Branislav Atlagić	Goran Švenda	Miodrag	Strahil Gušavac
Branislav Nerandžić	Gordana	Hadžistević	Svetlana Nikoličić
Branislav Veselinov	Milosavljević	Miodrag Zuković	Tanja Kočetov
Branislava Kostić	Gordana Ostojić	Mirjana	Tatjana Lončar -
Branislava	Igor Budak	Damnjanović	Turukalo
Novaković	Igor Dejanović	Mirjana Malešev	Toša Ninkov
Branka Nakomčić	Igor Karlović	Mirjana Radeka	Uroš Nedeljковиć
Branko Milosavljević	Ivan Beker	Mirko Borisov	Valentina Basarić
Branko Škorić	Ivana Katić	Miro Govedarica	Velimir Čongradec
Damir Đaković	Ivana Kovačić	Miroslav	Veran Vasić
Danijela Lalić	Ivana Miškeljin	Hajduković	Veselin Perović
Darko Čapko	Jasmina Dražić	Miroslav Popović	Vladimir Katić
Darko Marčetić	Jelena Atanacković	Mitar Jocanović	Vladimir Strezoski
Darko Reba	Jeličić	Mladen Kovačević	Vlado Delić
Dejan Ubavin	Jelena Borocki	Mladen Radišić	Vlastimir Radonjanin
Dejana Nedučin	Jelena Kiurski	Momčilo Kujačić	Vuk Bogdanović
Dragan Ivanović	Jelena Radonić	Nemanja	Zdravko Tešić
Dragan Ivetić	Jovan Petrović	Stanisavljević	Zoran Anišić
Dragan Jovanović	Jovanka Pantović	Nemanja Sremčev	Zoran Brujić
Dragan Kukolj	Ksenija Hiel	Nikola Đurić	Zoran Jeličić
Dragan Mrkšić	Laslo Nađ	Nikola	Zoran Mitrović
Dragan Pejić	Lazar Kovačević	Jorgovanović	Zoran Papić
Dragan Šešlija	Leposava Grubić	Nikola Radaković	Željen Trpovski
Dragana Bajić	Nešić	Ninoslav Zuber	Željko Jakšić
Dragana	Livija Cvetičanin	Ognjen Lužanin	
Konstantinović	Ljiljana Vukajlov	Pavel Kovač	
Dragana Šarac	Ljiljana Cvetković	Peđa Atanasković	
Dragana Štrbac	Ljubica Duđak	Petar Malešev	
Dragoljub Šević	Maja Turk Sekulić	Predrag Šiđanin	
Dubravka Bojanić	Marko Todorov	Radivoje Dinulović	
Dušan Dobromirov	Marko Vekić	Radovan Štulić	
Dušan Gvozdenac	Maša Bukurov	Slavica Mitrović	
Dušan Kovačević		Slavko Đurić	