



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



ЗБОРНИК РАДОВА ФАКУЛТЕТА ТЕХНИЧКИХ НАУКА

Едиција: Техничке науке - зборници

Година: XXXVII

Број: 4/2022

Нови Сад

Едиција: „Техничке науке – Зборници“

Година: XXXVII

Свеска: 4

Издавач: Факултет техничких наука Нови Сад

Главни и одговорни уредник: проф. др Срђан Колаковић, декан Факултета техничких наука у Новом Саду

Уредништво:

Проф. др Срђан Колаковић
Проф. др Александар Купусинац
Проф. др Борис Думнић
Проф. др Дарко Стефановић
Проф. др Себастиан Балош
Проф. др Дејан Лукић
Проф. др Јован Дорић
Проф. др Мирослав Кљајић
Проф. др Немања Тасић
Проф. др Дејан Убавин

Проф. др Милан Видаковић
Проф. др Мирјана Дамњановић
Проф. др Јелена Атанацковић Јеличић
Проф. др Игор Пешко
Проф. др Драган Јовановић
Проф. др Небојша Ралевић
Доц. др Сања Ожват
Проф. др Немања Кашиковић
Проф. др Теодор Атанацковић

Редакција:

Проф. др Дарко Стефановић, главни уредник
Проф. др Жељен Трповски, технички
уредник

Проф. др Драгољуб Новаковић
Доц. др Иван Пинђер
Бисерка Милетић

Језичка редакција:

Бисерка Милетић, лектор
Софија Рацков, коректор
Мр Марина Катић, преводилац

Савет за библиотечку и издавачку делатност ФТН,
проф. др Стеван Станковски, 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
Срђан Колаковић. – Год. 7, бр. 9 (1974)-1990/1991, бр.21/22 ; Год. 23, бр 1 (2008)-. – Нови Сад : Факултет
техничких наука, 1974-1991; 2008-. – илустр. ; 30 цм. –(Едиција: Техничке науке – зборници)

Месечно

ISSN 0350-428X

COBISS.SR-ID 58627591

ПРЕДГОВОР

Поштовани читаоци,

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

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

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

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

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

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

У овом броју штампани су радови студената мастер студија, сада већ мастера, који су радове бранили у периоду од 08.10.2021. до 31.10.2021. год., а који се промовишу 22.03.2022. год. То су оригинални прилози студената са главним резултатима њихових мастер радова.

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

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

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

- машинства и
- електротехнике и рачунарства.

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

- грађевинарства,
- саобраћаја,
- графичког инжењерства и дизајна,
- архитектуре,
- мехатронике и
- геодезије и геоматике,

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

- инжењерског менаџмента,
- инжењерства заштите на раду и заштите животне средине,
- управљања ризиком од катастрофалних догађаја и пожара,
- инжењерства информационих система,
- сценске архитектуре и дизајна,
- биомедицинског инжењерства и
- анимације у инжењерству.

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. Nikola Tatarević, UTICAJ HLAĐENJA ALATA ZA INJEKCIONO PRESOVANJE NA OTPRESAK	533-536
2. Весна Кнежевић, РЕГУЛАЦИОНА СТАНИЦА ГЛАВНОГ ГАСНОГ ТРАНСПОРТНОГ ЧВОРИШТА „ГОСПОЋИНЦИ“	537-540
3. Radojka Davinić, UPOREDNA ANALIZA SAVREMENIH SISTEMA ZA GREJANJE I PRIPREMU TOPLE POTROŠNE VODE	541-544
4. Dragana Đekić, ASPEKTI ANALIZE RIZIKA U PREDUZEĆU	545-548
5. Saša Đukić, Branko Štrbac, KVALITET DIMENZIONALNIH I GEOMETRIJSKIH KARAKTERISTIKA PROIZVODA DOBIJENIH BRIZGANJEM PLASTIKE	549-552
6. Darko Blanuša, MODULARNO PROJEKTOVANJE MAŠINA ALATKI ZA OBRADU METODOM BUŠENJA	553-556
7. Kristofer Solar, Sebastian Baloš, UTICAJ PARAMETARA ZAVARIVANJA TRENJEM SA MEŠANJEM SAMOPODESIVIM ALATOM NA EFIKASNOST SPOJA	557-560
8. Miljan Dragičević, Sebastian Baloš, OKSIACETILENSKA METALIZACIJA PRAHOM TANKOZIDNIH ČELIČNIH CEVI	561-564
9. Marina Zahorec, UTICAJ HABANJA ALATA NA SPEKTAR SIGNALA VIBRACIJA I POPREČNI PRESEK STRUGOTINE	565-568
10. Мирко Мијатовић, ПРИМЕНА LEAN МЕТОДОЛОГИЈЕ И ВІМ-А ПРИ ИЗРАДИ ПРОЈЕКТАТА	569-572
11. Дејан Поповић, Драган Ружић, МИКРОКЛИМАТСКИ УСЛОВИ У ПУТНИЧКОМ ПРОСТОРУ АУТОБУСА	573-576
12. Јована Марковић, Сања Бојић, УНАПРЕЂЕЊЕ ПРОИЗВОДНИХ СИСТЕМА ПРИМЕНОМ LEAN АЛАТА	577-580

13. Јелисавета Ковачић, Сања Бојић, ПРИМЕНА LEAN МЕТОДОЛОГИЈЕ У ОДРЖАВАЊУ ПРОИЗВОДНИХ СИСТЕМА	581-584
14. Dejan Lazarević, PROJEKAT BAZENSKE TEHNIKE I SISTEMA GREJANJA I VENTILACIJE ZATVORENOG BAZENA	585-588
15. Jovan Prole, MOGUĆNOST KORIŠĆENJA KISELOG GASA IZ GASNOG LEŽIŠTA „NOVO MILOŠEVO“	589-592
16. Vladan Mitrović, EKSPERIMENTALNO ISPITIVANJE OPTIMALNOG RADA GASNOG GORIONIKA	593-596

Radovi iz oblasti: Elektrotehnika i računarstvo

1. Биљана Татар, Невен Ковачки, ОДРЕЂИВАЊЕ ОПТИМАЛНОГ БРОЈА И ЛОКАЦИЈЕ ДАЉИНСКИ КОНТРОЛИСАНИХ ПРЕКИДАЧА ЗА АУТОМАТИЗАЦИЈУ ДИСТРИБУТИВНИХ МРЕЖА	597-600
2. Nemanja Krajčinović, Dejan Nemec, SISTEM ZA OBRADU PODATAKA PAMETNOG DOMA	601-604
3. Nataša Avramović, Dejan Nemec, PRIMENA MAŠINSKOG UČENJA ZA IDENTIFIKOVANJE LIMFNIH ČVOROVA KOD TIROIDNE ŽLEZDE	605-608
4. Милица Травица, ВИЗУЕЛИЗАЦИЈА И НАВИГАЦИЈА НАД ИСТОРИЈОМ ПРОМЕНА СОФТВЕРСКОГ ПРОЈЕКТА	609-612
5. Nenad Mišić, AUTOMATSKO RASPOREĐIVANJE ELEMENATA DIJAGRAMA UPOTREBOM SUGIYAMA ALGORITMA, LSTM NEURONSKIH MREŽA I ELK BIBLIOTEKE	613-616
6. Miloš Marković, GENERATOR KODA ZA OPTIMALNU IMPLEMENTACIJU BINDCT TRANSFORMACIJE	617-620
7. Stefan Veselinović, SOFTVERSKO REŠENJE ZA PODRŠKU LANCA NABAVKE ORGANSKE HRANE ZASNOVANO NA BLOKČEJNU	621-624
8. Milica Damjanović, GRAMATIKA GRAFIKE ZA VIZUALIZACIJU PODATAKA I NJENE IMPLEMENTACIJE	625-628
9. Ana Tomić, DIZAJNIRANJE IMERSIVNOSTI VIRTUELNE SCENE POMOĆU POGONA IGRE	629-632
10. Jelena Garić, REPRODUKCIJA ZVUKA U VIRTUALNOJ 3D SCENI	633-636
11. Snežana Dupljanin, RAZVOJ APLIKACIJE ZA UPRAVLJANJE ELEKTRIČNOM ENERGIJOM U OKVIRU CLOUD OKRUŽENJA	637-640
12. Boško Jovanović, Boris Dumnić, INDUSTRIJSKI KOMUNIKACIONI PROTOKOLI U TRANSFORMATORSKIM STANICAMA SREDNJEG I VISOKOG NAPONA	641-644
13. Aleksandar Magoč, SOFTVERSKA PLATFORMA ZA ANALIZU PODATAKA O RADU SOLARNIH FOTONAPONSKIH SISTEMA	645-648
14. Marina Vojnović, SISTEM ZA PRAĆENJE FINANSIJSKIH ASPEKATA UGOVORA	649-652
15. Jelena Popov, UPOTREBA ROBOTSKE AUTOMATIZACIJE PROCESA ZA KREIRANJE OBRAZOVNIH UPUTSTAVA	653-656

	STRANA
16. Aleksa Ivković, PLATFORMA ZA SPECIFIKACIJU I INTERPRETACIJU INTERAKTIVNE FIKCIJE	657-660
17. Đorđe Vujić, SIMULACIJA ROBOTA BAZIRANOG NA DIFERENCIJALNOM POGONU KORIŠĆENJEM GAZEBO SIMULATORA I ROS-A	661-665
18. Nemanja Živković, Vladimir A. Katić, KVALITET ELEKTRIČNE ENERGIJE NA NISKONAPONSKOJ STRANI ENERGETSKIH TRANSFORMATORA U POVRŠINSKOM KOPU "DRMNO"	666-669
19. Добривоје Ђурђевић, ПРИМЕНА ETL ПРОЦЕСА У ПОСТУПКУ ТРАНСФОРМАЦИЈЕ РЕЛАЦИОНЕ БАЗЕ ПОДАТАКА У ГРАФОВСКИ ОРИЈЕНТИСАНУ БАЗУ ПОДАТАКА	670-673
20. Željka Zolotić, Vladimir Katić, ISPITIVANJE KVALITETA ELEKTRIČNE ENERGIJE PO ŽALBAMA POTROŠAČA - STUDIJA SLUČAJA IZ PREDGRAĐA SUBOTICE	674-677
21. Andrej Jokić, TRANSFORMACIJE STABALA APSTRAKTNE SINTAKSE U PROGRAMSKI KOD	678-681
22. Goran Mitrović, Neven Kovački, PRIMENE DC MODELA TOKOVA SNAGA U PLANIRANJU I EKSPLOATACIJI ELEKTROENERGETSKIH SISTEMA	682-685
23. Marko Žužić, DETEKCIJA OBLIGACIJA U UGOVORIMA NA ENGLJESKOM JEZIKU	686-689
24. Ana Mutavdžić, DIGITALNI POTPISI I NJIHOVA ULOGA U BLOCKCHAIN TEHNOLOGIJI	690-693
25. Miloš Mihić, Željko Trpovski, PROJEKTOVANJE BAZNIH STANICA	694-697
26. Nikola Pavković, IMPLEMENTACIJA FAZI LOGIKE U UPRAVLJANJU SISTEMOM PRERADE VODE METODOM REVERZNE OSMOZE	698-701
27. Борис Јањић, КЛАСИФИКАЦИЈА ТИПА ГРЕШКЕ СЕНЗОРА КОРИШТЕЊЕМ МЕТОДА МАШИНСКОГ УЧЕЊА	702-705
28. Predrag Kovačević, BESKONTAKTNI SISTEM ZA PRANJE POD VISOKIM PRITISKOM ZA SAMOUSLUŽNU PERIONICU	706-709
29. Ivan Lažetić, Platon Sovilj, PRECIZNOST BIOHEMIJSKIH ANALIZATORA I IMUNOANALIZATORA U MEDICINSKIM LABORATORIJAMA	710-713
30. Јован Славујевић, Предраг Теодоровић, РЕАЛИЗАЦИЈА ТРОСЛОЈНОГ СИСТЕМА ЗА ВИДЕО ПРЕНОС ПО RTSP ПРОТОКОЛУ	714-717
31. Maja Grubor, PRIKAZ DEVOPS PRAKSI U AZURE OKRUŽENJU	718-721

UTICAJ HLADENJA ALATA ZA INJEKCIONO PRESOVANJE NA OTPRESAK INFLUENCE OF COOLING IN AN INJECTION MOLDING TOOL ON WARPAGE OF THE MOLDED PRODUCT

Nikola Tatarević, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – Veliki uspeh injekcionog presovanja (IP) u najvećoj meri se zasniva na mogućnosti proizvodnje visokog broja kompleksnih delova u relativno malom vremenskom periodu. Glavni cilj ovog rada je ispitivanje uticaja hlađenja alata kako bi se procenile kvalitativne i ekonomske prednosti korišćenja umetka proizvedenog aditivnom tehnologijom.

Cljučne reči: Injekciono presovanje plastike, Konformno hlađenje, Krivljenje

Abstract – Highly successful results of injection molding are mainly due to its ability to produce high amount of complex parts in a relatively short time. The main objective of this study was to examine the influence of mold cooling designs in order to evaluate the qualitative and economical benefits of using an additively manufactured core insert.

Keywords: Plastic injection molding, Conformal cooling, Warpage

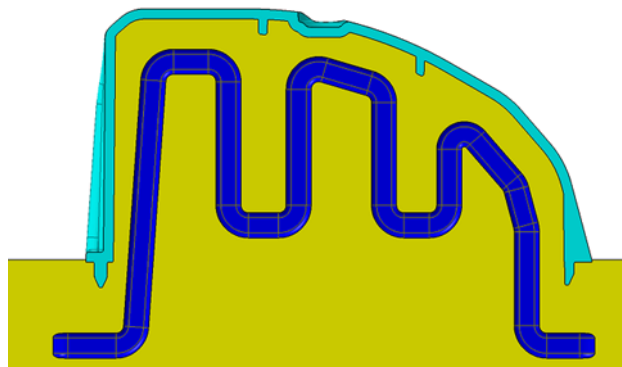
1. UVOD

Osnovni princip injekcionog presovanja jeste da se čvrsti polimer rastopi i ubrizga u šupljinu unutar kalupa i zatim se kao ohlađen deo izbacuje iz alata. Glavne faze u procesu injekcionog presovanja uključuju: punjenje, hlađenje i izbacivanje. Isplativost postupka, između ostalog, zavisi i od vremena provedenog u ciklusu presovanja. Pošto hlađenje predstavlja najduži deo ciklusa, jasno je da vreme hlađenja otpreska u najvećoj meri utiče na dužinu trajanja ciklusa, a samim tim i ekonomičnost proizvodnje. Sistem za hlađenje kod konvencionalnih alata sastoji se od sistema kanala i otvora unutar samog alata, kroz koje cirkuliše medijum za hlađenje (temperiranje) alata. Načini izrade ovih kanala oslanjaju se na konvencionalne metode izrade, npr. bušenje. Međutim, pomoću ovih tehnologija mogu se izraditi samo pravolinijski kanali što predstavlja veliki nedostatak u slučaju otpreska kompleksne geometrije. Konfiguracija kanala treba da prati geometriju otpreska (Slika 1).

U okviru ovog rada izvršena je analiza efikasnosti sistema za hlađenje alata za injekciono presovanje baziranog na primeni konformnih kanala i njegovo poređenje sa konvencionalnim tipom hlađenja. Radi se o alternativnom sistemu kod koga je geometrija kanala za hlađenje prilagođena obliku otpreska - prati liniju kalupne šupljine.

NAPOMENA:

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



Slika 1. Dizajn konformnih kanala na primeru alata kućišta antene

Pri definisanju kanala za hlađenje ovog sistema težilo se tome da kanali budu bliže površini kalupa, kako bi se toplota brže prenela na tečnost za temperiranje. Ovo osigurava da se deo hladi ravnomernije i efikasnije od konvencionalne metode. Analiza sistema hlađenja realizovana je na primeru alata za izradu kućišta antene putničkog automobila. Na osnovu funkcionalnog zahteva da štampana ploča, zbog orijentacije antene, bude postavljena vertikalno, dobijen je složeni oblik kućišta (Slika2) koji je veoma problematičan (netehnološki) za izradu postupkom injekcionog presovanja.



Slika 2. Kućište antene

Dimenzije dela od plastike igraju bitnu ulogu, pogotovo ukoliko će taj deo biti komponenta nekog sklopa. Kritične dimenzije, završni kvalitet površina, ravnost i drugi slični zahtevi moraju biti definisani zajedno sa realnim tolerancijama kako bi se obezbedio kvalitet otpreska. Cena izrade alata umnogome zavisi od veličine tolerancijskih polja. Pored kriterijuma ekonomičnosti procesa, u obzir je uzet i kvalitet proizvedenog dela.

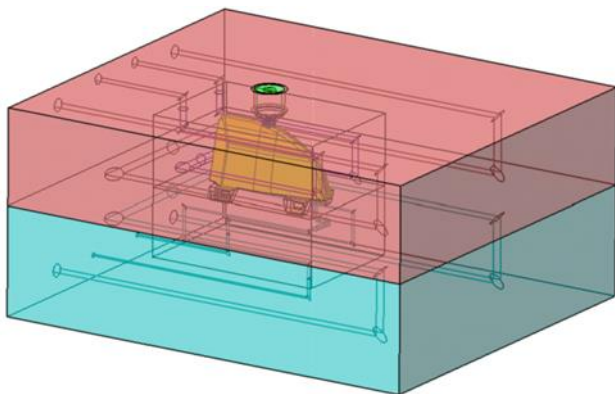
3. PRIPREMA CAE MODELA

Sprovedene analize bazirane su na primeni računarom podržanog inženjeringa (CAE) što je realizovano pomoću programskog paketa *Autodesk Moldflow Insight* u vidu simulacija. Ove simulacije bazirane su na metodi konačnih elemenata. MKE se zasniva na fizičkoj diskretizaciji razmatranog domena [1]. Tako se umesto elemenata diferencijalno malih dimenzija razmatra deo domena konačnih dimenzija tzv. konačni element.

Prvi postupak u analizi je formiranje modela u pripremnoj fazi. Ovim putem se stvara osmišljena, usklađena i povezana grupa konačnih elemenata kojom je opisan ispitivani kontinuum. Postoje četiri faze formiranja modela za analizu: formiranje geometrijskog modela, formiranje idealizovanog modela, formiranje modela zone i formiranje diskretnog modela.

Geometrijski model formira projektant alata. Ovaj model može da sadrži geometrijske elemente koji nemaju značaja za analizu jer ne utiču na ispitivan aspekt otpreska ili alata. Kako bi se smanjila potrebna snaga računara i postigla znatna ušteda vremena trajanja analize, formiraju se idealizovani modeli u kojima su odbačeni nevažni detalji.

Idealizovan model je uprošćen model koji ne mora da predstavlja celinu objekta ukoliko može da se njegovim simetričnim formama predstavi funkcija i način opterećenja celine (Slika 3). Idealizovan model se uvek formira sa zahtevom manjeg obima kontinuuma za analizu. Osnova razvoja racionalnih idealizovanih modela je apstrakcija. Apstrakcija je sagledavanje modela od strane analitičara kojom se postavlja koncept modela, uklanjaju detalji, prepoznaje simetrija, redukuje model i postavlja odgovarajuća opterećenja.



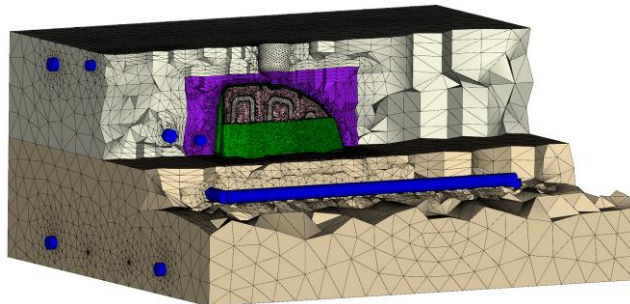
Slika 3. Idealizovan model alata korišćen za analizu

Model zona predstavlja idealizovan model raščlanjen na odgovarajuće segmente - zone koje dozvoljavaju podelu kontinuuma na konačne elemente prema standardnom-poznatom algoritmu generisanja ili preslikavanja.

Diskretni model se razvija na bazi modela zona i usklađenog broja elemenata konkretnih površina zona (Slika 4). On podrazumeva određivanje čvorova, konačnih elemenata, podataka o materijalu, opterećenju i graničnim uslovima. Diskretni model podrazumeva sva potrebna prilagođavanja mreže konačnih elemenata graničnim uslovima. Kao osnova za diskretizaciju u ovom radu izabrana je zapreminska vrsta konačnog elementa. Ova mreža se sastoji od čvrstih tetraedarskih (tetra)

elemenata gde svaki tetra element ima četiri trouglasta lica i šest stranica. Gde je bilo moguće, radi pojednostavljenja diskretizovanog modela, izabrani su elementi vrste grede za diskretizaciju kanala za hlađenje.

Nakon definisanja mreže konačnih elemenata, deo softvera (solver), rešava sistem jednačina koje se formiraju pomoću mreže konačnih elemenata. Nakon završenog proračuna sapost procesorskim operacijama mogu se vizualizovati rezultati analize (Slika 5). Kada je potrebno izdvojiti temperaturno kritične zone objekta (čiji model je rešen solverom) zadaje se kao uslov post - obrade rezultata analize. Na bazi ovog zahteva formira se slika kritičnih temperatura objekta sa obojenim kritičnim zonama.



Slika 4. Presek diskretizovanog modela alata sa integrisanim konformnim kanalima za hlađenje

3.2. Redosled analiza

Postupak dizajniranja kalupa započinje sa šupljinom, ispitivanjem mogućnosti položaja ušća i optimizacijom uslova unutar šupljine. Korišćenjem tog rasporeda ušća unose se korekcije sve dok uslovi unutar šupljine ne budu prihvatljivi.

Kada se optimizuje punjenje šupljine, može se analizirati sistem hlađenja otpreska. Generalno je cilj da se osmisli sistem hlađenja kalupa za ravnomerno izvlačenje toplote iz otpreska. Ovo će smanjiti vreme ciklusa dok se proizvode visokokvalitetni delovi.

Iako su punjenje i pakovanje (naknadni pritisak) usko povezani, pakovanje je najbolje optimizovati nakon analize hlađenja [2]. U fazi pakovanja dominira prenos toplote, dok u fazi punjenja dominira tok rastopa. Analiza hlađenja daje tačnu sliku o tome kako se toplota dela izdvaja, pa je najbolje optimizovati pakovanje dela nakon hlađenja.

Poslednji korak je utvrđivanje vitoperenja dela. Kada se deo pravilno analizira u prethodnim koracima, analiza vitoperenja potvrda je da je optimizacija dela i procesa dobro urađena.

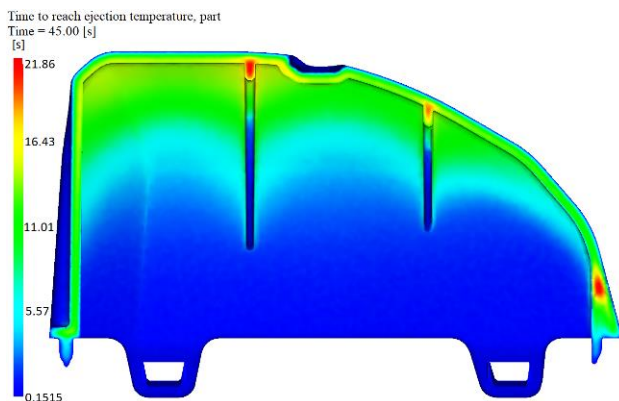
4. REZULTATI CAE ANALIZE

4.1 Analiza hlađenja alata za kućište antene

Kao jedan od rezultata analize hlađenja alata razmotriće se vreme potrebno da se dostigne temperatura izbacivanja (Slika 5 i Slika 6). U poprečnom preseku simuliranog otpreska prikazane su različitim bojama potrebna vremena za očvršćavanje dela.

Kod rezultata simulacije alata sa konvencionalnim sistemom hlađenja bitno je uočiti vreme potrebno za dostizanje temperature izbacivanja na vrhu umetka jezgra (Slika 5). Potrebno je preko 13 sekundi da očvrstne

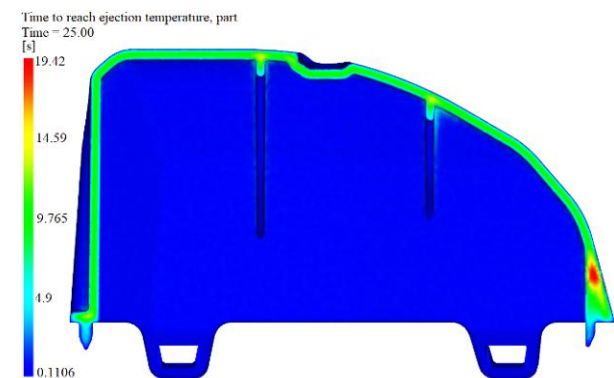
površinski sloj otpreska, dok je sa suprotne strane alata potrebno svega 0,3 sekunde. Na određenim mestima sa povećanom debljinom materijala potrebno je i preko 25 sekundi da otpresak očvrсне. Ove zone nisu glavni problem, jer je generalno moguće izbaciti otpresak nakon očvršćavanja više od 98% zapremine rastopa.



Slika 5. Vreme potrebno za očvršćavanje plastike, poprečni presek konvencionalno hlađenog otpreska

Kod rezultata simulacije alata sa konformnim sistemom hlađenja korišćene su isti polazni uslovi, s tim da je vreme ciklusa 25 s. Vreme ciklusa moguće je smanjiti zbog mnogo efikasnijeg odvođenja toplote koja se u ovoj izvedbi alata mnogo kraće zadržava na vrhu umetka jezgra.

Kao i kod analize prelazne temperature alata sa konvencionalnim hlađenjem, i kod konformno hlađenog alata prikazano je vreme potrebno za dostizanje temperature izbacivanja. Slika 6 ilustruje da je prosečno vreme za očvršćavanje otpreska oko 10 sekundi i da se ono dešava ravnomerno sa obe strane alata. Još uvek je na određenim mestima sa povećanom debljinom materijala potrebno znatno više vremena od proseka (25 sekundi) da otpresak očvrсне.



Slika 6. Vreme potrebno za očvršćavanje plastike, poprečni presek konformno hlađenog otpreska

4.2 Analiza skupljanja i vitoperenja kućišta antene

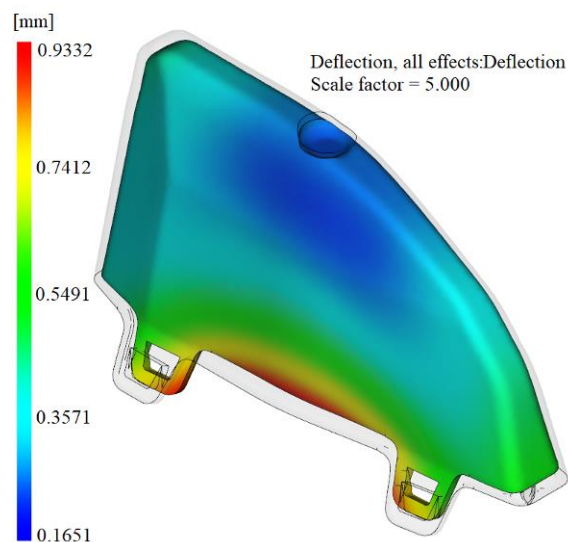
Za ulazne podatke ove analize, pored svojstva materijala preuzetih iz baze podataka programa, korišćene su i vrednosti iz prethodnih analiza. Iz analize *Prelazna temperatura unutar ciklusa* (Eng. *Transient within cycle* [2]) preuzete su temperaturne vrednosti alata, rastopa i okoline, dok su iz analize punjenja uzete vrednosti pritiska, naknadnog pritiska i dužine procesa ubrizgavanja.

Kao rezultat analize dobija se vrednost odstupanja od geometrije alata. Prilikom pripreme simulacije izabrana je opcija dekompozicije odstupanja na svoje uzroke, stoga se ukupno odstupanje može razložiti na vitoperenje usled diferencijalnog skupljanja i na vitoperenje usled diferencijalnog hlađenja.

Kada je u pitanju vitoperenje usled diferencijalnog skupljanja, rezultat prikazuje već sabrano standardno zapreminsko skupljanje sa skupljanjem usled promene geometrije otpreska. Pošto izabrani materijal u sebi ne sadrži nikakve dodatke (poput staklenih ili ugljeničnih vlakana), nije potrebno raditi analizu vitoperenja usled efekta orijentacije.

Slika 7 prikazuje celokupno odstupanje geometrije otpreska u odnosu na geometriju alata sa konvencionalnim sistemom hlađenja. Može se zaključiti da maksimalno odstupanje dostiže vrednost i do 0,93mm na sredini dela, pretežno usled diferencijalnog hlađenja alata.

Mora se proveriti da li dobijene dimenzije ispunjavaju dozvoljene tolerancije zadate od strane konstruktora plastičnog dela. U nekim slučajevima, kada se ove devijacije ne mogu umanjiti, konstruktori alata namerno konstruišu dodatno odstupanje alata u suprotnom smeru od očekivanog vitoperenja kako bi se otpresak nakon proizvodnje deformisao u željeni oblik. Ova metoda se zove kontra-vitoperenje (Eng. Pre-warpage).

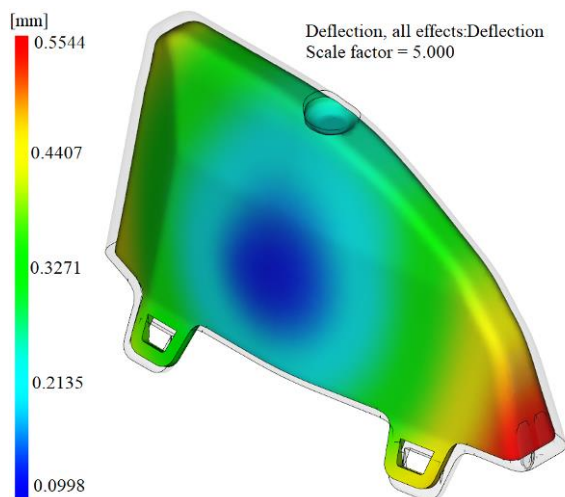


Slika 7. Prikaz svih odstupanja u odnosu na model konvencionalno hlađenog alata

Slika 8 prikazuje celokupno odstupanje geometrije otpreska u odnosu na geometriju alata sa konformnim sistemom hlađenja. Međutim, za razliku od rezultata konvencionalno hlađenog alata, najveće odstupanje se uočava na prednjem delu otpreska.

Razlog je to što se kod ove uvedbe alata gotovo potpuno eliminiše vitoperenje usled diferencijalnog hlađenja.

U tabeli 1 mogu se videti rezultati analize skupljanja i vitoperenja. Može se videti da je u slučaju konformnog hlađenja značajno redukovano vitoperenje otpreska nakon hlađenja i vađenja iz alata čime se obezbeđuje i bolji kvalitet otpreska u odnosu na konvencionalno hlađenje.



Slika 8. Prikaz svih odstupanja u odnosu na model konformno hlađenog alata

Tabela 1. Poređenje rezultata deformacije otpreska

Vrsta analize	Konvencionalno hlađen alat [mm]	Konformno hlađen alat [mm]
Maksimalno odstupanje usled diferencijalnog hlađenja	0,42	0,05
Maksimalno odstupanje usled diferencijalnog skupljanja	0,59	0,55
Ukupno odstupanje	0,93	0,55

5. ZAKLJUČAK

Konformno hlađenje alata za injekciono presovanje može biti atraktivna alternativa za poboljšanje kvaliteta otpreska, smanjenje vremena ciklusa i potrošnje energije. Međutim, upotreba ovakvog sistema za hlađenje je opravdana pri visokoserijskoj proizvodnji zbog visokih troškova izrade.

U radu je prikazan dizajn alata sa konformnim kanalima i upoređen je sa konvencionalnim sistemom hlađenja alata. Analizirana je efikasnost ova dva sistema i razmotreni su rezultati. Bitno je posebno napomenuti sledeće:

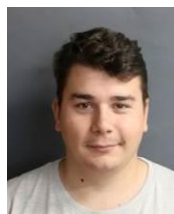
- Prednosti konformnog hlađenja najbolje se ističu kada otpresak ima duboku geometriju jer pravolinijski kanali za hlađenje ne mogu postići adekvatnu razmenu toplote.
- Srednje temperaturne vrednosti alata znatno su niže kod izvedbe alata sa konformnim kanalima za hlađenje.
- Vreme ciklusa je kraće za približno 44% ukoliko se koristi dizajn sa konformnim kanalima. Takođe je potreban manji broj ciklusa od početka proizvodnje da bi se postigla temperaturna stabilnost alata.
- Predložen dizajn alata sa konformnim kanalima za hlađenje pokazao se kao optimalnije rešenje za postizanje visokog kvaliteta otpreska. Celokupna deformacija otpreska smanjena je zbog znatno nižih deformacija uzrokovanih diferencijalnim hlađenjem.

Na osnovu rezultata može se zaključiti da je ova metoda hlađenja alata optimalnije rešenje u pogledu troškova proizvodnje i kvaliteta otpreska. Sledeći korak bio bi izrada ovog alata kako bi se potvrdili rezultati simulacija i dodatno optimizovala simulacija sa realnijim ulaznim podacima.

6. LITERATURA

- [1] Vladić, J., (2011) Kompjutersko projektovanje (skripta), Fakultet tehničkih nauka, Novi Sad
- [2] Shoemaker, J., (2006) Moldflow Design Guide

Kratka biografija:



Nikola Tatarević rođen je u Novom Sadu 1993. god. Diplomski-master rad na Fakultetu tehničkih nauka iz oblasti Mašinstva – Proizvodno mašinstvo odbranio je 2021.god.

**РЕГУЛАЦИОНА СТАНИЦА ГЛАВНОГ ГАСНОГ ТРАНСПОРТНОГ ЧВОРИШТА
„ГОСПОЋИНЦИ“****PRESSURE REGULATING STATION OF MAIN GAS TRANSPORT NODE
„GOSPODINCI“**

Весна Кнежевић, Факултет техничких наука, Нови Сад

Област - МАШИНСТВО

Кратак садржај – Предмет рада јесте регулациона станица у оквиру главног гасног транспортног чворишта „Госпођинци“. Услед велике редукције вредности притиска гаса поједини елементи регулационе станице се греју, тако да регулациона станица садржи и мерно-регулациону станицу за довођење гаса котларници, котларницу и измењивач топлоте. Циљ рада био је да се димензионише и усвоји опрема регулационе станице главног гасног транспортног чворишта „Госпођинци“. Циљ рада реализован је поштовањем законске регулативе Републике Србије. За потребе реализације циља рада коришћен је и Интерни технички правилник за пројектовање и изградњу гасовода и гасоводних објеката на систему ЈП „СРБИЈАГАС“.

Кључне речи: природни гас, регулациона станица, мерно – регулациона станица

Abstract – The subject of the paper is the regulating station of main gas transport node "Gospodinci." Due to the large reduction in the value of gas pressure, some elements of the control station are heated, so that the regulating station also contains a measuring and regulating station for supplying gas to the boiler room, boiler room and heat exchanger. The aim of this work was to dimension and adopt the equipment of the regulating station of the main gas transport node "Gospodinci." The goal of the manuscript was realized by respecting the legal regulations of the Republic of Serbia. For the purposes of realizing the goal of the manuscript, the Internal Technical Rulebook for the design and construction of gas pipelines and gas pipeline facilities on the system of SRBIJAGAS was used.

Keywords: natural gas, regulation station, measuring and regulating station

1. УВОД

Када се природни гас црпљењем доведе на земљину површину, приступа се његовој сепарацији, да би се довео у границе прописаног квалитета. После одређене прераде, он се транспортује преко магистралних гасовода све до потрошача. Системи транспорта гаса могу бити доста сложени јер постоји могућност спајања више гасовода у јединствен систем отпреме гаса. Пут који природни гас прође од производње до потрошача састоји се из неколико

различитих зона притиска. При преласку из једног у друго подручје притисак се регулише у регулационим станицама, а уколико се осим регулације врши и мерење количина протеклог гаса тада се говори о мерно-регулационим станицама. Гасна станица поставља се између магистралног и дистрибутивног гасовода, између дистрибутивног и потрошачког и између потрошачког гасовода и потрошача. Гасна станица раздваја гасовод на две зоне притиска. Притисак испред гасне станице може и да варира, али иза гасне станице притисак је приближно константан.

2. ТЕОРИЈСКЕ ОСНОВЕ**2.1. Гасне станице**

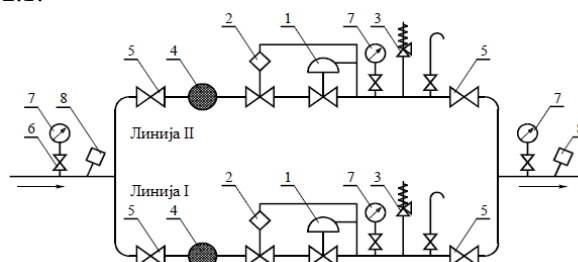
Гасне станице представљају делове гасовода у којима се мери количина предатог гаса и регулише се излазни притисак. На основу намене извршена је подела гасних станица на:

- мерно-регулационе које мере количину (енергију) протеклог гаса и и врше регулацију притиска (МРС),
- регулационе иза којих се регулише само притисак (РС), и
- мерне које мере количину (енергију) протеклог гаса (МС).

Основни делови гасних станица су:

1. сигурносни уређаји (сигурносни вентили и сл.);
2. запорни уређаји (славине, вентили, засуни и сл.);
3. регулишући уређаји;
4. цеви, фазонски комади, спојни елементи и заптивачи;
5. филтери;
6. мерни уређаји и
7. остало (уређаји за одоризацију и одвајачи течности).

Шематски приказ једне гасне станице дат је на слици 2.1.



Слика 2.1. Шема дволинијске гасне станице[1]

НАПОМЕНА:

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

На слици 2.1 приказана је дволинијска гасна станица, линија I представља радну линију, а линија II резервну. Линија I је увек у функцији, док се линија II налази у погонској приправности и у случају квара на радној линији ступа одмах у дејство.

2.2. Регулатор притиска

Основни задатак регулатора притиска је да одржава притисак у унапред задатим вредностима. Принцип рада састоји се у промени пресека кроз који протиче флуид. Радни притисак сме да осцилује само између унапред задатих вредности. Регулатори притиска за гас морају бити отпорни према хемијским утицајима, као и према неким одорантима који се могу наћи у гасу у дозвољеним количинама. Конструкција им мора бити таква, да се за време погона спречи могућност истицања гаса у просторију у коју је постављен. Сви делови регулатора морају бити изграђени од материјала који су намењени за радне услове под којима ће регулатор радити.

3. МАТЕРИЈАЛ И МЕТОД

3.1. Технички опис

Гас у регулациону станицу улази на притиску магистралног гасовода, који је једнак притиску од 74 bar, а затим се редукује на притисак који је једнак транспортном систему ЈП „Србијагаса“ и износи 45 bar.

Снижавање притиска гаса на радни притисак транспортног система ЈП „Србијагаса“ и регулација протока гаса врши се регулаторима „HON 530-E-WG“ произвођача „Honeywell“. Испред регулатора налазе се два блокадна вентила модел „HON 711“ произвођача „Honeywell“. Регулатор и блокадни вентили су називних пречника DN200, а класе притиска Class 600.

Максималан проток гаса у летњем режиму износи 250 000 m³/h за стандардне услове, док је у зимском режиму он 191 667 m³/h. Минималан проток гаса је 41 666 m³/h.

За МРС предвиђени су регулатори модел „HON 200“ произвођача „Honeywell“. Испред регулатора налази се блокадни вентил модел „HON 703“ произвођача „Honeywell“, који у себи садржи два уређаја који блокирају проток гаса у случају када притисак гаса после регулатора притиска порасте изнад дозвољене вредности чиме се спречава могућност квара уређаја који се налазе иза регулатора притиска. Регулатор и блокадни вентил су називног пречника DN25 и класе притиска Class 300. Максималан проток гаса за стандардне услове износи 200 m³/h, док је минималан проток гаса 20 m³/h.

3.2. Прорачун и избор регулатора

Прорачун и избор регулатора урађен је према препорукама стандарда SRPS EN 334:2010 [2]. При минималном притиску гаса на улазу:

$$p_{\min} = p_{ul} - \Delta p_f - \Delta p_{zg} - 2 \cdot \Delta p_{bv}, \quad (1)$$

где су:

p_{ul} – притисак испред регулатора притиска [bar],

Δp_f – пад притиска гаса кроз филтер [bar],

Δp_{zg} – пад притиска гаса на загрејачу [bar] и

Δp_{bv} – пад притиска гаса на блок вентилу [bar],

запремински проток гаса гласи:

$$Q_{\min} = K_G \cdot f \cdot \sqrt{p_{iz} \cdot (p_{ul} - p_{iz})}. \quad (2)$$

Запремински проток гаса при максималном притиску гаса на улазу у регулатор дефинише се као:

$$Q_{\max} = K_G \cdot f \cdot \sqrt{p_{iz} \cdot (p_{ul} - p_{iz})}, \quad (3)$$

где су:

p_{iz} – притисак после регулатора притиска [bar (abs)];

K_G – хидрауличка карактеристика регулатора притиска [m³/h] и

f – фактор конверзије гаса [m³/h].

Брзина гаса кроз регулатор притиска износи:

$$v = \frac{380 \cdot Q_{ulN} \cdot t_2 + 273,15}{D^2 \cdot p_{ul}} \cdot \frac{t_2 + 273,15}{288,15}, \quad (4)$$

где су:

t_2 – температура гаса испред регулатора притиска [°C],

Q_{ulN} – максималан проток гаса кроз регулатор притиска при нормалним условима [m³/h],

K_G – хидрауличка карактеристика регулатора притиска [m³/h] и

D – називни пречник цеви [mm].

Предвиђено је да се регулационе станице испитају према препорукама стандарда SRPS EN 12327 [3].

Предвиђено је испитивање регулационе станице и мерно – регулационе станице на чврстоћу, испитиним притиском који износи:

$$p_{is} = 1,5 \cdot p_{\max} \quad (5)$$

При испитивању гасовода на чврстоћу испитни притисак мора да се одржава минимум 1h.

Испитивање станица на непропусност вршено је испитним притиском који је једнак максималном радном притиску, што за регулациону станицу износи 74 bar, за опрему Class 600. За мерно – регулациону станицу овај притисак износи 50 bar, за опрему Class 300. При испитивању гасовода на непропусност испитни притисак мора да се одржава минимум 24 h.

3.3. Састав природног гаса

Састав природног гаса који је коришћен за прорачун регулатора притиска приказан је у табели 1:

Табела 1. Физичко – хемијске карактеристике природног гаса

метан	95,5 %
етан	2,53%
пропан	0,81%
и-бутан	0,12%
н-бутан	0,09%
и-пентан	0,12%
н-пентан	0,02%
хексан и виши угљоводоници	0,01%
азот	0,62%
угљендиоксид	0,18%

4. ПРОРАЧУН И ИЗБОР ОПРЕМЕ

4.1. Регулатор притиска – регулациона станица

Минималан притисак гаса на улазу у регулатор је:

$$p_{\min} = p_{ul} - \Delta p_f - \Delta p_{zg} - 2 \cdot \Delta p_{bl},$$

$$p_{\min} = 65,7 - 0,5 - 0,5 - 2 \cdot 0,16 = 64,38 \text{ bar (man)}.$$

Провера врсте струјања:

$$p_{iz \max} = 46 \text{ bar (abs)} > p_{ul \min} \cdot 0,5 = 65,39 \cdot 0,5 = 32,7 \text{ bar (abs)}$$

Струјање је подкритично.

Минималан запремински проток гаса кроз регулатор износи:

$$Q_{\min} = K_G \cdot f \cdot \sqrt{p_{iz} \cdot (p_{ul} - p_{iz})}.$$

$$Q_{\min} = 30000 \cdot 1,044 \cdot \sqrt{46 \cdot (65,39 - 46)} = 935383 \text{ Sm}^3/\text{h}.$$

При максималном притиску гаса на улазу у регулатор:

$$p_{iz \min} = 41 \text{ bar (abs)} > p_{ul \max} \cdot 0,5 = 75 \cdot 0,5 = 37,5 \text{ bar (abs)},$$

максималан запремински проток гаса износи:

$$Q_{\max} = K_G \cdot f \cdot \sqrt{p_{iz} \cdot (p_{ul} - p_{iz})},$$

$$Q_{\max} = 30000 \cdot 1,044 \cdot \sqrt{41 \cdot (75 - 41)} = 1169373 \text{ Sm}^3/\text{h}.$$

Брзина струјања гаса кроз регулатор притиска је:

$$v = \frac{380 \cdot Q_{ulN}}{D^2 \cdot p_{ul}} \cdot \frac{t_2 + 273,15}{288,15},$$

$$v = \frac{380 \cdot 256300}{200^2 \cdot 65,39} \cdot \frac{15,2 + 273,15}{288,15},$$

$$v = 37,3 \text{ m/s} < 120 \text{ m/s}.$$

Изабран је регулатор произвођача „Honeywell“, модел „HON 530-E-WG-200/200-200-1-FU“. Карактеристике регулатора дате су у *табели 2*.

Табела 2. Карактеристике изабраног регулатора притиска

Ознака	PCV-RS11 PCV-RS21		
Хидрауличка карактеристика регулатора	K_G	m ³ /h	30 000
Улазни притисак	-	bar	64,38 ÷ 74
Максимални радни притисак	$p_{\max}$	bar	74
Опсег излазног притиска	p_{izl}	bar	40 ÷ 45
Врста	аксијални регулатор са електро погоном	-	-
Погон	актуатор		
Врста актуатора	електро-моторни		

Место уградње: радна и резервна линија.

Притисак који се узима као испитни приликом испитивања регулационе станице на чврстоћу је:

$$p_{is} = 1,5 \cdot p_{\max},$$

$$p_{is} = 1,5 \cdot 75 = 111 \text{ bar},$$

за делове станице урађене у класи опреме Class 600.

Притисак који се узима као испитни приликом испитивања мерно-регулационе станице на чврстоћу је:

$$p_{is} = 1,5 \cdot p_{\max},$$

$$p_{is} = 1,5 \cdot 50 = 75 \text{ bar},$$

за делове станице урађене у класи опреме Class 300.

4.2. Регулатор притиска – мерно – регулациона станица

Минималан притисак гаса на улазу у регулатор је:

$$p_{\min} = p_{ul} - \Delta p_{zg} - 2 \cdot \Delta p_{bv},$$

$$p_{\min} = 40 - 0,4 - 2 \cdot 0,007 = 39,58 \text{ bar (man)},$$

Провера врсте струјања:

$$p_{iz \max} = 1,17 \text{ bar (abs)} < p_{ul \min} \cdot 0,5 = 40,6 \cdot 0,5 = 20,3 \text{ bar (abs)}.$$

Струјање је надкритично.

Минималан запремински проток гаса кроз регулатор износи:

$$Q_{\min} = \frac{K_G \cdot f \cdot p_{ul \min}}{2},$$

$$Q_{\min} = \frac{25 \cdot 1,031 \cdot 40,6}{2} = 523,2 \text{ m}^3/\text{h}.$$

При максималном притиску гаса на улазу у регулатор од:

$$p_{iz \max} = 1,16 \text{ bar (abs)} < p_{ul \min} \cdot 0,5 = 45,6 \cdot 0,5 = 22,8 \text{ bar (abs)},$$

запремински проток гласи:

$$Q_{\max} = \frac{K_G \cdot f \cdot p_{ul \max}}{2},$$

$$Q_{\max} = \frac{25 \cdot 1,031 \cdot 45,6}{2} = 587,7 \text{ m}^3/\text{h}.$$

Брзина гаса кроз регулатор притиска је:

$$v = \frac{380 \cdot Q_{ulN}}{D^2 \cdot p_{ul \min}} \cdot \frac{t_2 + 273,15}{288,15},$$

$$v = \frac{380 \cdot 186,4}{25^2 \cdot 40,6} \cdot \frac{23 + 273,15}{288,15},$$

$$v = 2,9 \text{ m/s} < 50 \text{ m/s}.$$

Изабран је регулатор произвођача „Honeywell“, модел „HON 200-25/25-AL-2/6-610N-703“. Карактеристике регулатора дате су у *табели 3*.

Табела 3. Карактеристике изабраног регулатора притиска

Ознака	PCV-MS11 PCV-MS21		
Хидрауличка карактеристика регулатора	K_G	m ³ /h	25
Улазни притисак	-	bar (man)	39,58 ÷ 45
Максимални радни притисак	p_{max}	bar	50
Опсег излазног притиска	p_{izl}	bar	0,100 ÷ 0,500
Излазни притисак – радна линија	p_{izl}	bar	0,150
Излазни притисак – резервна линија	p_{izl}	bar	0,130
Врста	регулатор са пилотом	-	-

Место уградње: радна и резервна линија.

5. ЗАКЉУЧАК

Предмет рада биле су регулациона станица у оквиру главног гасног чвора „Госпођинци“ и мерно-регулациона станица за котларницу. Циљ рада био је димензионисање станица, као и усвајање и избор опреме. За реализацију циља мастер рада урађен је прорачун у складу са стандардима и прописима који су тренутно на снази у Републици Србији, димензионисане су станице и изабрана је опрема. Регулатор за регулациону станицу је произвођача „Honeywell“, типа „HON 530-E-WG-200/200-200-1-FU“, класе притиска Class 600.

Испитни притисак је 111 bar. Максималан проток гаса кроз регулациону станицу износи 250 000 m³/h. Регулатор мерно – регулационе станице је произвођача „Honeywell“, типа „HON 200-25/25-ал-2/6-610N-703“, класе притиска Class 300. Испитни притисак је 75 bar. Максималан проток гаса кроз мерно – регулациону станицу је 200 m³/h.

6. ЛИТЕРАТУРА

- [1] Бикић, С., Букуров, М., Гасна постројења, Скрипта Факултета техничких наука, Нови Сад, 2018.г.
- [2] SRPS EN 334:2010, Регулатори притиска гаса за улазне притиске од 10 МПа, European Norm
- [3] SRPS EN 12327:2013, Испитивање притиском, поступци за пуштање у рад и искључивање из рада – Функционални захтеви, European Norm

Кратка биографија:



Весна Кнежевић рођена је у Новом Саду 1997. год. Основне студије из области Машинства завршила је на Факултету техничких наука у Новом Саду 2020 год. Мастер рад из области машинства одбранила је на истом факултету 2021 год. Контакт: vesna_1997@hotmail.com.

UPOREDNA ANALIZA SAVREMENIH SISTEMA ZA GREJANJE I PRIPREMU TOPLE POTROŠNE VODE**COMPARATIVE ANALYSIS OF MODERN SYSTEMS FOR HEATING AND SANITARY HOT WATER PREPARATION**Radojka Davinić, *Fakultet tehničkih nauka, Novi Sad***Oblast - TERMOENERGETIKA**

Kratak sadržaj – Ovaj rad prezentuje uputstvo za rad u softveru GeoTSOL i prikazuje uporednu tehnoeкономsku analizu sistema za grejanje i pripremu tople potrošne vode u sistemu sa toplotnom pumpom i solarnim kolektorima.

Ključne reči: Grejanje, sanitarna topla voda, toplotne pumpe, solarni kolektori, uporedna analiza

Abstract – This paper features a guide for working in the software GeoTSOL and displays comparative techno-economical analysis of solar collectors and heat pump systems with different heat sources for heating and sanitary hot water preparation.

Keywords: Heating, sanitary hot water, heat pump, solar collectors, comparative analysis

1. UVOD

Variranje i rast cene konvencionalnih goriva, kao i mala količina preostalih rezervi fosilnih goriva je naterala ljude da se okrenu obnovljivim izvorima energije i pronalaženju adekvatne zamene fosilnim gorivima. Uređaji koji omogućavaju i olakšavaju korišćenje energije koja se nalazi u našem okruženju su toplotne pumpe. U skladu sa tim, cilj rada jeste predstavljanje načina na koji se različiti sistemi sa toplotnom pumpom dimenzionišu, koje su prednosti i nedostaci datih sistema i koja je njihova primenljivost u uslovima Srbije. Analiza sistema sa toplotnom pumpom vazduh - voda, voda - voda, zemlja (geokolektori) - voda i zemlja (geosonde) - voda je izvršena pomoću softvera GeoTSOL [1].

Prva celina predstavlja teorijske osnove o toplotnoj pumpi, izvorima toplote i predstavljanje softvera u kome se radi analiza.

Druga veća celina rada predstavlja unos konkretnih podataka potrebnih kako bi se sistemi definisali i uporedili. Posle unosa podataka, vrši se proces simulacije rada izabranih sistema i porede se dobijeni energetski i finansijski rezultati.

2. PODELA, PREGLED I UPOTREBA TOPLLOTNIH PUMPI RAZLIČITIH IZVORA TOPLOTE

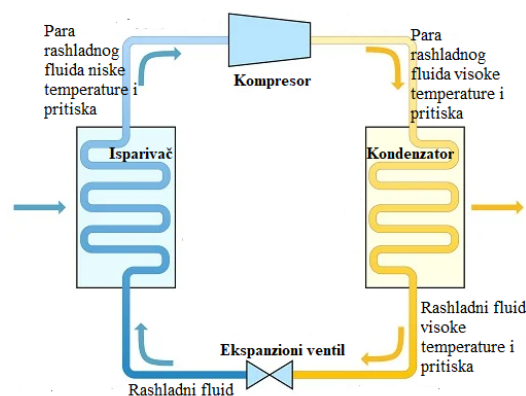
Toplotne pumpe su uređaji koji preuzimaju energiju sa jednog mesta i prebacuju je na drugo mesto na kome je potrebno obezbediti grejanje ili hlađenje.

NAPOMENA:

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

Energija koju svojim radom generišu je nekoliko puta veća od uložene električne energije koja je potrebna za njihov rad.

Osnovni delovi toplotne pumpe su: kompresor, ekspanzioni ventil i dva razmenjivača toplote - isparivač i kondenzator. Princip rada toplotne pumpe je jednostavan i prikazan na slici 1.



Slika 1. Šema rada toplotne pumpe

2.1 Podela toplotnih pumpi prema vrsti toplotnog izvora - Vazduh

Toplotna pumpa vazduh - voda, tokom režima grejanja koristi toplotu okolnog vazduha za zagrevanje primarnog rashladnog medijuma. Ovaj tip toplotne pumpe se ne preporučuje za samostalnu upotrebu tokom hladnih zimskih dana kada je ambijentalna temperatura u minusu.

2.2 Podela toplotnih pumpi prema vrsti toplotnog izvora - Podzemna voda

Toplotna energija se dobija iz bušenih bunara, čija je temperatura vode 10 - 14 °C. Bunar iz koga se ispušćava podzemna voda se naziva crpni, a bunar u koji se podzemna voda vraća se naziva upojni. Sistem toplotne pumpe može da bude sa sekundarnom petljom, što podrazumeva postojanje razmenjivača toplote između toplotnog izvora i toplotne pumpe.

2.3 Podela toplotnih pumpi prema vrsti toplotnog izvora - Zemlja

Sistemi toplotnih pumpi koji koriste energiju zemlje postoje u dve izvedbe - sa vertikalnim cevima (sondama) i sa horizontalnim cevima (kolektorima). Sistem sa geokolektorima se postavlja u zemlju na dubinu do 3 m, dok se vertikalne sonde postavljaju u bušotine koje se nalaze na dubini 40 do 100 m. Bušotina se ispunjava najčešće smešom bentonit gline, peska i vode [2].

Temperatura zemljišta u koje se postavljaju geokolektori varira tokom godine od 2 °C do 20 °C, dok je temperatura na dubini bušotina gotova konstanta oko 10 °C.

3. PROJEKTOVANJE KOMBINOVANOG SISTEMA TOPLOTNE PUMPE I SOLARNIH KOLEKTORA U SOFTVERU GEOTSOL

3.1 Koncept rada softvera GeoTSOL

GeoTSOL je programski alat koji se koristi za planiranje i projektovanje sistema sa toplotnom pumpom. Takođe, simulira godišnji rad sistema sa toplotnom pumpom koji može da koristi i solarne kolektore i konvencionalne kotlove.

3.2 Izbor sistema i parametara neophodnih za simulaciju

- *Izbor lokacije objekta.*
- *Izbor varijante sistema:* od kojih komponenti se sastoji sistem.
- *Parametri sistema za grejanje:* potrebno je definisati temperaturski režim grejanja, toplotno opterećenje objekta, grejnu površinu, temperaturu grejanja i granične temperature.
- *Parametri sistema za pripremu tople potrošne vode (TPV):* unose se podaci o prosečnoj dnevnoj potrošnji tople vode u objektu ili podaci o broju članova i potrošnji tople vode po članu, podatak o temperaturi tople vode i da li postoji recirkulaciona petlja tople vode.
- *Izbor toplotnog generatora:* izbor toplotne pumpe iz baze softvera i ukoliko postoji, definiše se i kotao koji se koristi kao pomoćni sistem u petlji grejanja.
- *Izvor toplote:* kartica izvor toplote je različita za svaki izvor toplote. Za sistem sa toplotnom pumpom vazduh - voda podaci o nominalnoj snazi ventilatora se automatski preuzimaju iz podataka o izabranoj toplotnoj pumpi. Za sistema sa toplotnom pumpom voda - voda potrebno je uneti temperature podzemne vode, kao i podatke o pumpi za podzemnu vodu i cirkulacionu pumpu ukoliko je sistem sa sekundarnom petljom. Za sistem sa geotermalnim kolektorima unose se podaci o površini koju zauzimaju, tipu zemljišta u koje se postavljaju i na koju dubinu.. Zbog mogućeg uticaja podzemnih vode na kolektore, definiše se i dubina njihovog nalaženja i temperatura. Kroz sistem kolektora protiče rastvor glikola i vode i potrebno je definisati snagu cirkulacione pumpe koja pokreće rastvor. Za sistem sa geosondama koje se postavljaju u bušotine, potrebno je definisati njihov tip, zatim prečnik, broj i ispunu bušotina, kao i maksimalnu dubinu bušenja. Cirkulaciona pumpa pokreće rastvor u sondama i potrebno je definisati njenu snagu. Za dubinu na koju se postavljaju geosonde, definiše se i temperatura zemlje. Softver izračunava potrebnu dubinu geosondi. Postoji razlika između dubine geosondi koju softver predlaže (l_{sonde}) i dubine geosondi ($l_{uk,buš}$) koja se dobija kao proizvod broja iskopanih bušotina i dubine jedne bušotine. Zbog toga se uvodi faktor relativnog odstupanja, f:

$$f = \frac{l_{sonde} - l_{uk,buš}}{l_{sonde}} \quad (1)$$

Ukoliko je vrednost faktora f manje od 1, sistem geosondi je optimalno dimenzionisan.

- *Načini rada sistema grejanja:*

1. Monovalentni - sistem samo sa toplotnom pumpom.
2. Alternativni - sistem sa toplotnom pumpom i pomoćnim sistemom grejanja (električnim grejačem ili kotlom). Definiše se granična temperatura do koje radi toplotna pumpa, nakon koje se uključuje kotao i radi samostalno.
3. Paralelan - takođe sistem sa toplotnom pumpom i pomoćnim uređajem. Definiše se temperatura uključivanja pomoćnog sistema, ali se toplotna pumpa ne isključuje i radi zajedno sa pomoćnim sistemom na niskim temperaturama.
4. Delimično paralelan - Pored temperature uključivanja pomoćnog sistema, definiše se i temperatura isključivanja toplotne pumpe. Dakle, sistem toplotne pumpe i pomoćnog uređaja radi uporedo tokom jednog temperaturskog opsega, nakon koga pomoćni sistem radi samostalno.

- *Izbor rezervoara tople vode:* u zavisnosti od varijante sistema koji je izabran potrebno je definisati rezervoar tople potrošne vode, bafer rezervoar za sistem grejanja ili kombinovani rezervoar.

- *Definisanje solarnog sistema:* ukoliko je izabrana varijanta sistema sa solarnih kolektorima, iz baze softvera se bira vrsta solarnih kolektora (pločasti ili vakuumski), njihov broj i položaj na krovu. Definiše se i dužina cevi i snaga solarne cirkulacione pumpe.

- *Finansijska analiza:* da bi softver izvršio finansijsku analizu potrebno je uneti - životni vek sistema, kamatnu stopu (ukoliko se novac pozajmljuje od banke), reinvesticioni povrat (RI), stopu inflacije, cenu investicije i ukoliko postoje, subvencije, troškovi održavanja sistema, porast cene energije na godišnjem nivou. Na osnovu datih podataka softver generiše podatke o tome kolika je cena proizvodnje toplotne energije po kWh, vrednost faktora MIRR i neto sadašnju vrednost. Investicija je isplativa ukoliko je $RI < MIRR$ i ukoliko je neto sadašnja vrednost pozitivna.

- *Rezultati simulacije.*

4. OPIS HIBRIDNIH SISTEMA SA RAZLIČITIM TOPLOTNIM PUMPAMA

U radu su analizirani sistemi za grejanje objekta i pripremu tople potrošne vode sa toplotnom pumpom. Biće analizirana četiri različita slučaja:

1. sistem toplotne pumpe koja koristi energiju vazduha i solarnih kolektora,
2. sistem toplotne pumpe koja koristi energiju podzemne vode i solarnih kolektora,
3. sistem toplotne pumpe sa geotermalnim kolektorima i solarnim kolektorima,
4. sistem toplotne pumpe sa geosondama i solarnim kolektorima.

Ulazni podaci koji su za sva četiri sistema ista su sledeći:

- Lokacija: Novi Sad
- Način grejanja: podno grejanje
- Temperatura petlje grejanja: 45 °C / 33,4 °C
- Toplotno opterećenje: 4,9 kW
- Grejna površina: 102 m²
- Temperatura u objektu: 20 °C
- Limit grejanja: 12 °C
- Proračunska grejna temperatura: -15 °C
- Br. članova domaćinstva: 5
- Dnevna potrošnja vode po članu: 40 l
- Temperatura TPV: 50 °C
- Solarni kolektori: 2 kolektora, ukupne površine 5,1 m², proizvođača Bosch.

4.1 Prikaz sistema sa toplotnom pumpom vazduh - voda

Izabrana je varijanta 4 sistema sa toplotnom pumpom vazduh - voda proizvođača Daikin nominalne snage 3,5 kW. Sistem pored toplotne pumpe i solarnih kolektora sadrži i električni grejač snage 5 kW koji se uključuje na temperaturi od -8 °C. Toplotna pumpa i električni grejač rade u alternativnom režimu rada. Rezervoar u sistemu je kombinovan, zapremine 750 l, Vitocell 340 - M.

4.2 Prikaz sistema sa toplotnom pumpom voda - voda

Izabrana varijanta sistema je 7.1 sa toplotnom pumpom Nilan Compact P Geo 3 nominalne snage 3 kW. Sistem toplotne pumpe je sa sekundarnom petljom, snage cirkulacione pumpe za rastvor od 87 W, a snaga pumpe za podzemnu vodu je 150W. Maksimalna temperatura podzemne vode je u septembru 12 °C, a minimalna u martu 8 °C. Pomoćni sistem grejanja je gasni kotao Bosch snage 7 kW i uključuje se na spoljašnjoj temperaturi od -5 °C. Toplotna pumpa i gasni kotao rade u paralelnom režimu rada. Deo sistema su i dva rezervoara tople vode - rezervoar za toplu potrošnu vodu Ivar Prestige Den 500, zapremine 500 l i bafer rezervoar od 300 l, SWD P1 90.300.

4.3 Prikaz sistema sa toplotnom pumpom sa geotermalnim kolektorima

Izabrana je varijanta sistema 7.1. Toplotna pumpa, gasni kolektor i rezervoari tople vode su isti kao u slučaju sistema sa toplotnom pumpom voda - voda. Toplotna pumpa i gasni kotao rade u paralelnom režimu rada i kotao se uključuje na -5 °C. Sistem se geotermalnih kolektora se postavlja u vlažno peskovito zemljište na dubini od 1,2 m i zauzima površinu od 200 m². Unutar kolektora se nalazi 30 % rastvor glikola koji pokreće cirkulaciona pumpa snage 87 W. Podzemne vode se nalaze na 10 m dubine i imaju temperaturu od 10 °C.

4.4 Prikaz sistema sa toplotnom pumpom sa geotermalnim sondama

I u ovom slučaju izabrana je varijanta sistema 7.1 i iste komponente kao u slučaju sistema sa podzemnom vodom i geotermalnim kolektorima. Razlika u ovom sistemu jeste dimenzionisanje geosondi. Softver definiše potrebnu dužinu sonde koja iznosi 118 m, a maksimalna dubina bušenja je 99 m. U skladu sa potrebnom dužinom sonde i maksimalnom dubinom bušenja, izabrane su dve bušotine na dubini od 58,9 m. Prečnik bušotine iznosi 150 mm, a temperature zemlje na toj dubini je 10 °C. Izabrana je pojedinačna U - sonda u kojoj se nalazi 30% rastvor glikola. Snaga cirkulacione pumpe je 87 W. Faktor relativnog odstupanja je manji od 1 i sistem je optimalno dimenzionisan:

$$f = \frac{|l_{sonde} - l_{uk,buš}|}{l_{sonde}} = \frac{|118 - 117,8|}{118} = 0,002 \quad (1)$$

5. REZULTATI SIMULACIJE

U tabeli 1, prikazani su energetske rezultati simulacije za period od godinu dana. Očekivano, najveći sezonski faktor performanse ima toplotna pumpa sa geosondama 4,33. Ovo se može objasniti kroz tip samog toplotnog izvora koji se koristi. Od svih tipova toplotnih izvora koji se koriste u ovom rada, zemlja sa geosondama predstavlja najstabilniji toplotni izvor i razlika između temperature zemljišta na dubini geosonde i temperature prostora koja treba da se postigne je uvek pogodna za postizanje dobre efikasnosti toplotne pumpe. Sezonski faktor performanse (SPF) sistema toplotne pumpe i solarnih kolektora je veći od SPF-a samo toplotne pumpe (TP) što nam govori o tome da je kombinacija toplotne pumpe i solarnih kolektora odličan izbor.

Tabela 1: Energetske rezultati simulacije

Sistem	4.1	4.2	4.3	4.4
SPF TP	3,05	4,20	4,15	4,33
SPF TP + solarni sistem	3,51	4,46	5,30	5,44
Količina energije koju generiše toplotna pumpa	6035 kWh	4865 kWh	4738 kWh	4835 kWh
Količina energije koju generiše solarni kolektori	2581 kWh	2720 kWh	2816 kWh	2757 kWh
Količina energije koju obezbede solarni kolektori	3140 kWh	3308 kWh	3381 kWh	3338 kWh
Solarna frakcija TPV	51,2 %	42,7 %	43,9 %	43,2 %

Iz tabela 2 se može videti da je količina generisane energije u slučaju toplotnih pumpi koje koriste vodu i energiju zemlje kao toplotne izvore vrlo slična, a da se količina energije koju generiše toplotna pumpa vazduh - voda, značajnije razlikuje. Razlog tome je u načinu povezivanja rada toplotne pumpe sa pomoćnim sistemom grejanja (paralelni i alternativni način). Može se uočiti da je količina energije koju generišu solarni kolektori manja od količine energije koju obezbede kolektori. Razlika između ove dve vrednosti je u gubicima koji postoje u solarnoj petlji - gubici solarne cirkulacione pumpe, gubici solarnog rezervoara, gubici u cevima. Udeo solarne frakcije za pripremu TPV je jedan od važnijih parametara za proveru da li je sistem dobro dimenzionisan. Njegova vrednost treba da je između 50 - 70%.

Finansijska analiza je urađena za životni vek sistema od 22 godine, investicija se finansira iz sopstvenih sredstava - kamatna stopa i RI su nula, cena električne energije je 0,06 \$/kWh i stopa inflacije je 4 %. Za svoj rad toplotna pumpa koristi električnu energiju više tarife 0,06 \$/kWh i niže tarife 0,015 \$/kWh. Cena goriva sistema sa kojim se poredi toplotna pumpa je 0,036 \$/kWh i to je sistem koji koristi gas.

Troškovi održavanja za sisteme sa TP su 50 \$/god, a za sistem sa gasnim kotlom 200 \$/god. Investicioni troškovi

su najveći za sistem sa geosondama 12.000 \$, zatim za sistem sa geokolektorima 11000 \$, 10000 \$ je investicija u sistem sa podzemnom vodom i 6000 \$ je za sistem sa toplotnom pumpom vazduh - voda.

Analiza isplativosti sistema sa toplotnim pumpama je urađena za dva slučaja - postojanje subvencija i ukoliko subvencije ne postoje. Država Srbija tokom 2021. godine dodeljuje subvencije za povećanje energetske efikasnosti, između ostalog za kupovinu toplotnih pumpi tako da je u ovom slučaju, iznos investicije snižen za 50 %. Usporedni prikaz finansijske analize u ovom slučaju prikazan je u tabeli 2.

Tabela 2: Usporedni prikaz rezultata finansijske analize

Sistem	4.1	4.2	4.3	4.4
Cena proizvodnje toplotne energije iz sistema	TP: 0,046 \$/kWh	TP: 0,065 \$/kWh	TP: 0,065 \$/kWh	TP: 0,070 \$/kWh
	Uspoređeni sistem: 0,111 \$/kWh	Uspoređeni sistem: 0,111 \$/kWh	Uspoređeni sistem: 0,111 \$/kWh	Uspoređeni sistem: 0,111 \$/kWh
MIRR	9,83 %	5,80 %	5,43 %	4,68 %
Neto sadašnja vrednost	13 737 \$	9837 \$	9891 \$	8684 \$

Cena proizvodnje energije iz sistema sa toplotnom pumpom je najniža u slučaju toplotne pumpe vazduh - voda. Ovo je očekivano, s obzirom da je sistem sa toplotnom pumpom vazduh - voda značajnije jeftiniji od ostalih. Ipak, sa svakim sistemom toplotne pumpe se vrši ušteda na godišnjem nivou u odnosu na sistem sa kojim se poredi. Modifikovana interna stopa rentabilnosti je najveća u slučaju sistema sa vazduhom 9,83 %, a najmanja u slučaju korišćenja geosondi. Razlog ovome je velika razlika u trošku investicije. Neto sadašnja vrednost, odnosno preostala količina novca na kraju životnog veka je pozitivna za sve toplotne izvore. Ukoliko se analizira slučaj bez subvencija, krajnji finansijski rezultati bi bili nepovoljniji (tabela 3). Međutim, čak i u ovom slučaju gledajući parametre MIRR i neto sadašnju vrednosti, ulaganje u ove sisteme je isplativo.

Tabela 3: Finansijska analiza u slučaju da ne postoji subvencija

Sistem	4.1	4.2	4.3	4.4
Cena proizvodnje toplotne energije iz sistema	TP: 0,061 \$/kWh	TP: 0,088 \$/kWh	TP: 0,091 \$/kWh	TP: 0,095 \$/kWh
	Uspoređeni sistem: 0,111 \$/kWh	Uspoređeni sistem: 0,111 \$/kWh	Uspoređeni sistem: 0,111 \$/kWh	Uspoređeni sistem: 0,111 \$/kWh
MIRR	5,35 %	1,97 %	1,67 %	1,25 %
Neto sadašnja vrednost	10 737 \$	4837 \$	4391 \$	3460 \$

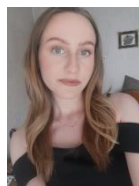
6. ZAKLJUČAK

Softver GeoTSOL predstavlja odličnu prvu stanicu u analizi sistema sa toplotnom pumpom, zato što je jednostavan za upotrebu i sadrži sve neophodne delove za dimenzionisanje sistema. Sistemi sa toplotnim pumpama su se pokazali kao dobro i isplativo rešenje u našim uslovima, bez obzira na visoka početna ulaganja. Njihovo održavanje nije skupo, efikasni su i imaju dugi životni vek. Treba obratiti pažnju na to da svake godine postoje subvencije između ostalog i za kupovinu toplotnih pumpi, što od države ili od kompanija koje se bave prodajom ovih uređaja. Na osnovu toga se može zaključiti da se sistem sa toplotnom pumpom retko kupuje bez posredstva subvencija. Na treba zaboraviti da smo svedoci sve većeg problema sa snabdevanjem konvencionalnim energentima. Čini se da je povećanje energetske efikasnosti, korišćenje obnovljivih izvora energije kroz sisteme sa toplotnim pumpama jedino pravo rešenje.

7. LITERATURA

- [1] <https://valentin-software.com/en/downloads/> Datum pristupa: 30.07.2021.
- [2] <https://www.greenmatch.co.uk/heat-pump/ground-source-heat-pumps-in-the-uk/ground-source-heat-pump-borehole> Datum pristupa: 07.09.2021.

Kratka biografija:



Radojka Davinić rođena je u Zrenjaninu 1996. god. Osnovne studije Čistih energetskih tehnologija na Fakultetu tehničkih nauka završila je 2019. god. Nakon toga upisuje master studije Energetike i procesne tehnike - Termoelektrotehnika koje završava 2021. kontakt: radojkad96@gmail.com

ASPEKTI ANALIZE RIZIKA U PREDUZEĆU**ASPECTS OF RISK ASSESSMENT IN THE COMPANY**

Dragana Đekić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj - Procijena rizika predstavlja način za utvrđivanje faktora rizika koji utiču na zdravlje ljudi, kao i na materijalna dobra. Takođe, određuje prioritete za preduzimanje mjera kako bi se rizik smanjio. Smatra se da su analiza, procijena rizika i adekvatno upravljanje rizicima, od ključnog značaja za postizanje odgovarajućeg kvaliteta funkcionisanja kompleksnih sistema.

Ključne reči: Analiza rizika, Metode za procjenu rizika, Ocjena rizika proizvoda, Upravljanje rizikom od požara.

Abstract – Risk assessment is a way to determine risk factors that affect human health as well as material goods. It also sets priorities for action to reduce risk. Analysis, risk assessment and adequate risk management are considered to be fundamental for achieving adequate quality of complex systems.

Keywords: risk analysis, risk assessment methods, product risk assessment, fire risk management.

1. UVOD

Procijena rizika predstavlja način za utvrđivanje faktora rizika koji utiču na zdravlje ljudi i materijalna dobra. Takođe, određuje prioritete za preduzimanje mjera kako bi se rizik smanjio. Rizik se može posmatrati sa dva aspekta, aspekta uzroka i aspekta posljedice. Dakle, može se reći da je rizik mjera opasnosti, uz sagledavanje njenih posljedica. Ovakve posljedice uvijek nanose štetu, koja može biti ekonomske, socijalne, ekološke prirode i sl. Prema tome, procjena rizika se veže za procijenu štete, što je veća očekivana šteta, to je veći i rizik.

U ovom radu su date osnovne smjernice za upravljanje rizikom i opisane metode za procijenu rizika, sve u cilju lakšeg obavljanja posla procijenitelja rizika.

2. OSNOVE PROCJENE RIZIKA**2.1. Osnovni pojmovi i definicija**

Rizik je vjerovatnoća nastanka povrede, oboljenja ili oštećenja zdravlja zaposlenih uslijed opasnosti [1].

Termin „rizik“, često se zamjenjuje terminom „opasnost“. Opasnost znači potencijalan uslov, stanje ili događaj koji mogu u određenim okolnostima dovesti do gubitka. Opasnost je situacija koja predstavlja stepen opasnosti po život, zdravlje, imovinu ili životnu sredinu. Bez postojanja opasnosti, ne može se govoriti o riziku.

2.2. Metode za procjenu rizika

Metoda procjene rizika definiše algoritme, alate i način sprovođenja procjene, dok procedura sprovođenja procjene rizika definiše niz koraka za sprovođenje procjene rizika u skladu sa zakonima i propisima. S obzirom na vrstu podataka koje koriste, metode za procjenu rizika mogu biti: kvalitativne; kvantitativne i kombinovane.

Kvalitativne metode se baziraju na iskustvu učesnika u procjeni rizika i korišćenju raspoloživih numeričkih podataka. U okviru ovih metoda se najčešće koriste subjektivni kriterijumi, koji se mjere u subjektivnim skalama. Koriste se skale sa tri do sedam kvalitativnih opisa. Metode sa manje od tri kvalitativna opisa nisu zanimljive, a skale sa više od sedam kvalitativnih opisa dovode do značajnih poteškoća, povezanih sa nemogućnošću učesnika u procjeni rizika da precizno prepozna opis faktora rizika.

Procjena rizika ovim metodama podrazumijeva korišćenje kvalitativno opisanih podataka. Ovoj grupi metoda za procjenu rizika pripadaju matrice rizika, u koje spadaju [2]:

- matrica rizika 4x6 (MIL – STD – 882C),
- matrica rizika 5x5 (AS/NZS 4360:2004) i
- matrica rizika 3x3 (OHSAS standard).

Kvantitativna metoda procjene rizika polazi od osnovnog obrasca:

$$\text{Rizik} = \text{Vjerovatnoća događaja} \times \text{Posljedice događaja}$$

Kvantitativna metoda procjene rizika, dakle predstavlja konačnu, tačnu brojnu vrijednost rizika. Kvantitativni kriterijum koristi numeričke vrijednosti kako bi se opisala vjerovatnoća nastanka štetnog događaja i veličine štete – posljedice [3]. Oba činioca se iskazuju kao brojne vrijednosti, zbog čega je neophodno sprovesti dublje analize, posjedovati odgovarajuće statističke podatke i sl., što predstavlja isuviše složen proces za masovnu primjenu. Zbog toga se češće podliježe kvalitativnoj metodi procjene rizika, dok se kvantitativna uglavnom koristi u slučajevima visokog rizika.

Polukvantitativne metode za procjenu rizika, u praksi imaju široku primjenu. Procjenjivanje i rangiranje ovih veličina se zasniva na iskustvu eksperata koji učestvuju u procjeni rizika. Svakoj određenoj kvantitativnoj mjeri rizika pridružuje se kvalitativno tumačenje, odnosno kvalitativni opis i odgovarajući rang [4]. Kvalitativne skale sa određenim brojem kvalitativnih opisa za vjerovatnoću i posljedicu, su osnov za procjenu rizika.

Postoje tri pristupa procjene rizika ovom metodom:

1. matična metoda (5x5, zasnovana na poznatim metodama AUIA i BG),

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dejan Lukić, vanr. prof.

2. tabelarna metoda procjene rizika (*GUARDMASTER* metoda),
3. grafička metoda procjene rizika (Kinney i Wiruth metoda).

3. OCJENA I SMANJENJE RIZIKA PROIZVODA

3.1. Osnove bezbjednosti proizvoda – mašina

Kako bi proizvođač stekao pravo da postavi srpski 3A ili CE znak usaglašenosti na svoj proizvod/mašinu, pomoću koga može da putuje kroz cijelu Srbiju ali i Evropsku uniju, potrebno je sledeće [5]:

- Mašina mora biti u skladu sa svim bitnim zahtjevima za zdravlje i bezbjednost, propisanim pravilnikom;
- Mora biti dostupna tehnička dokumentacija za mašinu, na srpskom, engleskom i jeziku ciljnog tržišta;
- Mora biti dostupno uputstvo za korišćenje mašine i sve dodatne informacije, takođe na ova tri jezika;
- Mora biti sprovedena procedura ocijenjivanja usaglašenosti;
- Mora biti izdata (EC) deklaracija o usaglašenosti i dostupna uz mašinu, kojom proizvođač potvrđuje usaglašenost,
- Mora biti postavljen CE znak, na propisan način, nakon sprovedene odgovarajuće procedure.

3.2. Osnove primene SRPS ISO 12100

Osnovna svrha ovog međunarodnog standarda jeste da projektantima obezbijedi okvirna uputstva za projektovanje mašine koja će biti bezbjedna pri korišćenju u njenoj osnovnoj namjeni. Pored toga, ovaj standard utvrđuje osnovnu terminologiju, principe i metodologiju postizanja bezbjednosti pri projektovanju mašina. Takođe, utvrđuje principe ocijene i smanjenja rizika, koji su zasnovani na znanju i iskustvu, a koji proističu iz projektovanja, korišćenja, incidenata, nezgoda i rizika koji se na mašine i odnose.

U ovom međunarodnom standardu opisane su i procedure za identifikovanje opasnosti i za procjenu rizika koji se javljaju prilikom upotrebe neke mašine, kao i procedure za otklanjanje opasnosti ili odredbe za dovoljno smanjenje rizika [6].

3.3. Strategija ocjene rizika

Projektant mora da preduzme aktivnosti, prema sledećem redoslijedu, da bi uključio ocijenu i smanjenje rizika [6]:

1. odrediti granice mašine, koje uključuju predviđenu upotrebu, ali i bilo koju nepravilnu upotrebu koja se može predvidjeti;
2. identifikovati opasnosti i opasne situacije u vezi s njima;
3. procijeniti rizike za svaku identifikovanu opasnost i opasnu situaciju;
4. vrednovati rizike i preduzeti odluke o potrebi za smanjenjem rizika;
5. eliminisati opasnosti ili smanjiti rizik preduzimanjem zaštitnih mjera.

Aktivnosti od 1 – 4 se odnose na ocijenu rizika. Ocjena rizika predstavlja niz logičnih koraka koji omogućavaju analizu i vrednovanje rizika koji se odnose na mašinu.

Ukoliko je potrebno, ocijena rizika je praćena smanjenjem rizika (aktivnost 5).

3.4. Ocjena rizika proizvoda

Ocjena rizika obuhvata analizu rizika (određivanje granica mašine, identifikaciju opasnosti, procjenu rizika) i vrednovanje rizika. Analiza rizika obezbijeduje informacije koje su potrebne za vrednovanje rizika, čime se odlučuje da li je potrebno smanjiti rizik ili ne [6].

- I. Ocjena rizika započinje određivanjem granica mašine:
 - granice upotrebe uključuju predviđenu upotrebu i nepravilnu upotrebu mašine koja se može predvidjeti;
 - uticaj prostornih ograničenja koji treba uzeti u obzir;
 - uticaj vremenskih ograničenja koji treba uzeti u obzir;
 - druga ograničenja.
- II. Sledeći korak kod ocijene rizika jeste identifikacija opasnosti, opasnih događaja i opasnih događaja tokom cijelog životnog vijeka mašine, tj.:
 - transport, montaža i instaliranje;
 - puštanje u rad;
 - upotreba;
 - demontaža, stavljanje van pogona i prekid rada.
- III. Poslije identifikacije opasnosti, procjena rizika mora da se izvrši za svaku opasnost, određivanjem elemenata rizika:
 - nivoa štete;
 - vjerovatnoće pojavljivanja štete;
 - uticaji koji se razmatraju u toku procjene rizika.
- IV. Vrednovanje rizika – potrebno je odraditi vrednovanje rizika kako bi se odredilo da li je potrebno smanjiti rizik.

3.5. Smanjenje rizika

Smanjenje rizika je moguće postići eliminisanjem opasnosti ili smanjenjem svakog od dva elementa koji određuju rizike, a to su [6]:

- nivo štete nastale uslijed opasnosti koja se razmatra i
- vjerovatnoća nastanka te štete.

Postoje tri osnovna koraka za smanjenje rizika [6]:

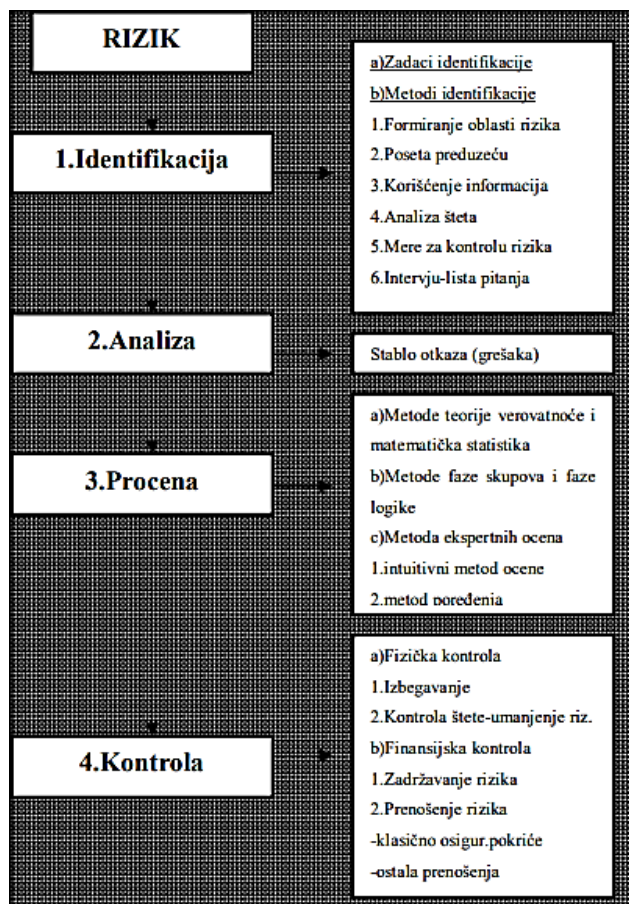
- bezbjednosne mjere ugrađene u projekat;
- bezbjednosna zaštita;
- uputstvo za upotrebu.

Nacrt uputstva za upotrebu je sastavni dio projektne dokumentacije. Ovo uputstvo mora da sadrži sve smjernice za pravilno rukovanje mašinom. Ono se sastoji od niza tekstova, riječi, znakova, signala, simbola i sl. koji sami ili u kombinaciji treba da prenesu informacije korisniku. Uputstvo za upotrebu treba da ukaže na potrebu za obukom, korišćenjem sopstevnih zaštitnih sredstava i moguću potrebu za dodatnim zaštitnicima ili sličnim uređajima. Ovo uputstvo obuhvata transport, montažu, puštanje u rad i upotrebu mašine. Ukoliko je to neophodno, ono sadrži i uputstvo za demontažu, stavljanje van pogona i prekid rada.

4. UPRAVLJANJE RIZIKOM OD POŽARA

4.1. Faze upravljanja rizikom

Osnovni zadatak upravljanja rizikom jeste težnja da se raznim metodama rizici kontrolišu i da se upravlja njima. Na slici 1 su prikazane osnovne faze upravljanja rizikom.



Slika 1. Faze upravljanja rizikom [7]

Identifikacija rizika je veoma složen proces, koji obuhvata dobijanje informacija o izvorima rizika, hazardima, opasnostima i mjestima gdje može doći do štete. Metode za identifikaciju rizika su [7]: ček lista; anketiranje; posjeta preduzeću; matrice vjerovatnoće; intervju.

Analiza rizika predstavlja ključnu fazu upravljanja rizikom, u kojoj se opisuju identifikovani rizici, izoluju uzroci, analizira njihov uticaj na rizik, procijenjuje se vjerovatnoća nastanka štetnog događaja i posljedice tog događaja, ocijenjuju i kvantifikuju rizici, formira se lista prioriteta, predlažu metode za tretman rizika i sl.

Procjena rizika dolazi kao sledeći korak upravljanja rizikom. Potrebno je primjeniti adekvatne metode procjene rizika. Ove metode mogu biti: kvantitativne; kvalitativne i polukvantitativne.

Izbor metode i instrumenata upravljanja, od kojih se izdvajaju: metode fizičke kontrole; metode finansijske kontrole i metode internog smanjenja rizika.

Kontrola rizika je kontinualni proces praćenja i procjene parametara procesa upravljanja rizikom pomoću mjerenja i izveštavanja. Razlikuju se fizička i finansijska kontrola rizika.

4.2. Požar i rizici

„Požar je svako nekontrolisano širenje vatre u prostoru, nanoseći materijalnu štetu, a nerijetko odnoseći i ljudske živote.“ Do nastanka požara najčešće dolazi zbog ljudskog faktora, nepoštovanjem protivpožarnog režima. Požari uglavnom nastaju na jednom mjestu, te se šire u zavisnosti od veličine požara, količine gorivog materijala, prepreka i konstrukcija u cijelom objektu. Četiri osnovne faze odvijanja požara su [7]:

- Faza – rana faza. U ovoj fazi se požar odvija relativno sporo, ali se vremenom razvija i njegova brzina.
- Faza – predstavlja nagli prelaz rastućeg požara u stanje u kome gore svi gorivi materijali.
- Faza – je faza potpunog razvoja požara.
- Faza – je faza potpunog raspada i tada požar postepeno jenjava.

Uslovi požarnih rizika

Osiguranje od požara koje se odnosi na civilna osiguranja, podrazumijeva pokriće pokretnih i nepokretnih stvari tj. imovine privrednih društava i drugih pravnih lica, poljoprivrednih gazdinstva i imovine građana.

Ovo osiguranje se odnosi na sledeće [8]:

- zgrade, namještaj i ostale pokretnosti u zgradama,
- robu i zalihe u trgovinskim radnjama, ugostiteljskim preduzećima i skladištima,
- muzeje, izložbe, pozorišta, bioskope i sl.,
- sajmove i slične priredbe,
- zalihe drva i uglja na slobodnom prostoru,
- niskogradnja (željezničke pruge, avionske piste, kanalizacione mreže i sl.).

Predmet osiguranja mogu biti sve stvari, osim onih koje su isključene uslovima, a to su npr.:

- zemljište, nepopločana dvorišta, nasipi i sl.,
- nepečena cigla i crijep na otvorenom,
- životinje u tekućim i stajaćim vodama,
- usijevi u klasu, plodovi na stablu, rodni i nerodni zasadi osim stabala voćaka u okviru dvorišta.
- motorna vozila, prikolice i samohodne mašine,
- plovni objekti, vazduhoplovi i roba u prevozu i sl.

U pogledu osiguranih opasnosti, za grupu industrijskih osiguranja važe iste odredbe kao i za grupu civilnih osiguranja. Razlika je u nosiocima i predmetu osiguranja, ali su uslovi za osiguranje požara i drugih opasnosti isti.

Rizici

Uslovima osiguranja su pokriveni rizici, koji mogu biti osnovni i dopunski. U osnovne rizike spadaju [8]: požar, udar groma, eksplozija, oluja, grad, udar sopstvenog motor-nog vozila, pad letjelice, manifestacije i demonstracije.

Dopunski rizici obuhvataju sledeće [8]: poplavu, bujicu i visoke vode, izliv vode iz instalacija, zemljotres, klizanje tla i odronjavanje, sniježna lavina, iscurenje tečnosti (lekaža) iz cjevovoda, samozapaljenje zaliha.

4.3. Procjena rizika i osiguranje od požara

Najjednostavniji i najkorišćeniji metod od 70 – ih godina pa sve do danas je metod Euroalarm. Zasniva se na izračunavanju požarnog rizika objekta i požarnog rizika sadržaja objekta.

Požarni rizik objekta zavisi od intenziteta i vremena trajanja požara, kao i od konstrukcije građevinskog objekta. Izračunava se prema obrascu (1) [9]:

$$Ro = \frac{((Po * C) + Pk) * B * L * S}{W * Ri} \quad (1)$$

Gdje su:

Ro – požarni rizik za objekat,
Po – koeficijent požarnog opterećenja sadržaja objekta,
C – koeficijent sagorljivosti sadržaja u objektu,
Pk – koeficijent nepokretnog požarnog opterećenja,
B – koeficijent veličine i položaja požarnog sektora,
L – koeficijent kašnjenja početka gašenja,
S – koeficijent širine požarnog sektora,
W – koeficijent otpornosti na požar noseće konstrukcije,
Ri – koeficijent smanjenja požarnog rizika.

Svi ovi koeficijenti se određuju pomoću različitih tabela, kao i koeficijenti pomoću kojih se izračunava požarni rizik sadržaja objekta.

Požarni rizik za sadržaj objekta se izračunava na osnovu sledećeg obrasca (2) [9]:

$$R_s = H \cdot D \cdot F \quad (2)$$

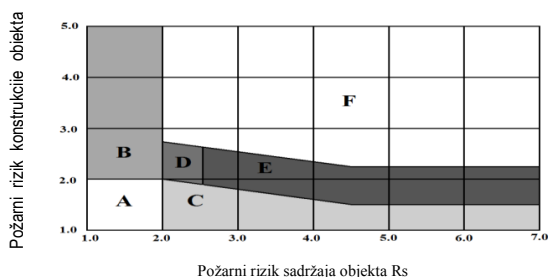
Gdje su:

H – koeficijent opasnosti po ljude;
D – koeficijent rizika imovine;
F – koeficijent djelovanja dima.

Analiza dobijenih rezultata [9]

Na slici 2 je prikazan dijagram za analizu dobijenih rezultata. Na njemu se određuje odgovarajuća tačka. Ako se ona nađe u:

- oblasti A onda nije neophodna ugradnja stabilnih sistema za dojavu i gašenje požara;
- oblasti B onda nije neophodna ugradnja stabilnog sistema za dojavu a preporučuje se ugradnja stabilnog sistema za gašenje požara;
- oblasti C onda nije neophodna ugradnja stabilnog sistema za gašenje a preporučuje se ugradnja stabilnog sistema za dojavu požara;
- oblasti D onda je neophodna ugradnja stabilnog sistema za gašenje a preporučuje se ugradnja stabilnog sistema za dojavu požara;
- oblasti E onda je neophodna ugradnja stabilnog sistema za dojavu a preporučuje se ugradnja stabilnog sistema za gašenje požara;
- oblasti F onda je neophodna ugradnja stabilnog sistema za dojavu i ugradnja stabilnog sistema za gašenje požara.



Slika 2. Dijagram za analizu dobijenih rezultata [9]

Maksimalna moguća šteta (PML)

Procjena maksimalne moguće štete se vrši za svaki rizik koji se daje u reosiguranje. PML se procenjuje podjelom rizika na požarne komplekse, te se za procjenu rizika uzima kompleks sa najvećom vrijednošću. Procjena rizika

se vrši na osnovu scenarija koje postavljaju i dokumentuju preuzimači rizika. Scenario pretpostavlja događaj na riziku pod najnepovoljnijim okolnostima, npr. najgori scenario požara, tj. eksplozije, pod pretpostavkom da su svi protivpožarni sistemi otkazali i da nema efektivne borbe protiv požara. PML se iskazuje kao procentualna vrijednost sume osiguranja.

5. ZAKLJUČAK

U radu su iznijeti osnovni aspekti procjene rizika kod proizvoda, dok je veći akcenat dat na rizicima od požara. Danas postoje brojni faktori koji mogu da prouzrokuju požar, počevši od različitih proizvodnih procesa, nosećih struktura objekata, kao i sirovina koje se čuvaju po skladištima. Iz tog razloga, procijenitelj rizika treba da obrati mnogo veću pažnju o gore navedenim stavkama, i da posebnu pažnju posveti protivpožarnoj zaštiti. Prilikom izrade izvještaja o procjeni rizika, potrebno je da se skrene pažnja na sve faktore koji utiču na nastanak štetnog događaja i da se pismeno dokumentuju. Osiguravajuća kuća daje ponudu za osiguranje imovine u skladu sa procijenjenim nivoom rizika. Ukoliko je veći rizik u pitanju, on povlači za sobom i veću premiju, tj. cijenu osiguranja. Stoga rizik nastanka štete od požara treba da se svede na najmanju moguću mjeru.

6. LITERATURA

- [1] Zakon o bezbjednosti i zdravlju na radu, *Službeni glasnik Republike Srbije* br.101, 21.11.2005.
- [2] Grupa autora: *Bezbjednost i zdravlje na radu*, Monografija: Mašinski fakultet u Kragujevcu, Kragujevac, 2009.
- [3] Adamović, T., Voskresenski, V., Tul, P.: *Održavanje na bazi rizika*, prvo izdanje: Društvo za tehničku dijagnostiku Srbije, Beograd, 2007.
- [4] Kiš, D.: *Znanstveni prisup analizi rizika radnog mjesta u procjeni opasnosti*: Rad Sigur, 6, 2002.
- [5] Stokić, D., Raičević, A.: *Vodič za bezbednost mašina*: Privredna komora Srbije, Beograd, 2016.
- [6] Institut za standardizaciju Srbije: *SRPS EN ISO 12100 – Bezbednost mašina, opšti principi za projektovanje, ocena rizika i smanjenje rizika*, Maj, 2014.
- [7] Purić, P.: *Upravljanje rizikom u požarnom osiguranju*: Master rad, Beograd, 2011.
- [8] *Uslovi za osiguranje od opasnosti požara*: DDOR – osiguravajuće društvo.
- [9] Sekulović, Z., Bogner, M. i Pejović, S.: *Preventivna zaštita od požara*: ETA, Beograd, 2012.

Kratka biografija:



Dragana Đekić je rođena u Derventi 1996. god. Master rad na Fakultetu tehničkih nauka iz oblasti Mašinstva – Proizvodno mašinstvo, smjer Računarom podržane tehnologije odbranila je 2021. god.
kontakt: draganadjekic996@gmail.com

**KVALITET DIMENZIONALNIH I GEOMETRIJSKIH KARAKTERISTIKA
PROIZVODA DOBIJENIH BRIZGANJEM PLASTIKE****QUALITY OF DIMENSIONAL AND GEOMETRIC CHARACTERISTICS OF PLASTIC
INJECTION MOLDED PRODUCTS**Saša Đukić, Branko Štrbac, *Fakultet tehničkih nauka, Novi Sad***Oblast – MAŠINSTVO**

Kratak sadržaj – *Proizvod sa zadatim tolerancijama izrade nemoguće je zamisliti bez upotrebe koordinatnih mjernih sistema u procesu verifikacije u fazi razvoja proizvoda. Prema mjerenjima funkcionalnih dimenzija kao specijalnih karakteristika proizvoda prikazana je sposobnost procesa. Ovaj rad je imao za cilj da utvrdi ocijenu kvaliteta dimenzionalnih i geometrijskih karakteristika proizvoda tokom njegovog životnog ciklusa. Rezultati su pokazali da je koordinatni mjerni sistem, uopšteno i mjerni sistem neizostavan segment životnog ciklusa proizvoda.*

Ključne reči: *Životni ciklus proizvoda, razvoj proizvoda, sposobnost procesa*

Abstract – *The Product with given manufacturing tolerances is impossible to imagine without the use of coordinate measuring systems in the verification process in development phase of the product. Based on the measuring the functional dimensions as special characteristics of the product is shown process capability. This paper aimed to determine and evaluate the quality of dimensional and geometric characteristics of the product during life cycle. The results showed that coordinate measuring system, as well as the measuring system in general, are an unavoidable segment of the product life cycle.*

Keywords: *Product life cycle, Development phase, Capability process*

1. UVOD

Ekonomski uspjeh svake firme zavisi od njihove sposobnosti da identifikuju potrebe kupaca i da stvore proizvode koji zadovoljavaju njihove potrebe. Razvoj proizvoda je skup aktivnosti koje počinju sa percepcijom prilika na tržištu i završavaju se sa proizvodnjom, prodajom i isporukom proizvoda [1].

Razvoj proizvoda i tehnologija u direktnoj je vezi sa istraživanjem tržišta, investicijama i zasnovana je na timskom radu. Dizajniranje je zasnovano na zahtjevima korisnika u pogledu tehničkih, funkcionalnih i estetskih karakteristika proizvoda i odgovarajućim proračunima.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Branko Štrbac.

Dizajn mora jasno da objasni funkciju proizvoda, njegovu primjenu i da bude jasno vidljiv tako da ga korisnik može razumjeti [4].

Razvoj tehnologija podrazumijeva niz aktivnosti vezanih za definisanje načina izrade proizvoda u svim fazama razvoja proizvoda i u serijskoj proizvodnji.

Proizvodi su sve više prisutni u našim životima. Da bismo mogli dalje razrađivati životni ciklus proizvoda je potrebno da znamo šta je to proizvod. Sve što može biti ponuđeno na tržištu, svojim oblikom i svojim karakteristikama zadovoljava određenu potrebu predstavlja proizvod. On prolazi određeni životni ciklus i put na tržištu. Životni ciklus proizvoda je veoma važan koncept, pogotovo u današnje vrijeme kad je konkurencija sve veća. Upravo taj ciklus nam omogućava da imamo uvid u dinamiku konkurentnosti proizvoda. Iz navedenog proizilazi da svaki proizvod ima ograničen vijek trajanja. Prodaja proizvoda prolazi nekoliko specifičnih faza, svaka od njih predstavlja različite izazove, probleme i mogućnosti za prodavca. Proizvodi zahtijevaju različite marketing, finansijske, proizvodne, nabavne i strategije ljudskih resursa u svakoj fazi svog životnog ciklusa. U oblasti projektovanja postupaka i tehnologija koristi se veliki broj postupaka i procedura. Primjenom računara, a posebno CAD/CAM sistema, ne samo da ubrzava aktivnosti razvoja, već stvara pretpostavke za formiranje novih koncepata razvoja. Mjerenje efikasnosti procesa vršimo ocjenom sposobnosti procesa. Da bi utvrdili stabilnost i sposobnost procesa, potrebno je analizirati i utvrditi da li možemo vršiti precizna mjerenja..

2. PREDMET ISTRAŽIVANJA

Za opis predmeta, geometrijskim specifikacijama proizvoda na tehničkim crtežu se definišu: geometrija, dimenzije i površinske osobine predmeta. Na ovaj način bi, s obzirom na određene greške u proizvodnji, trebala biti zagarantovana optimalna funkcionalnost pojedinog dela. Uprkos tome, postoje i predmeti koji su proizvedeni tako da ne zadovoljavaju ove uslove funkcionalnosti pojedinog dijela. Otuda su predmeti mjere kako bi se uporedili sa specifikacijama [2].

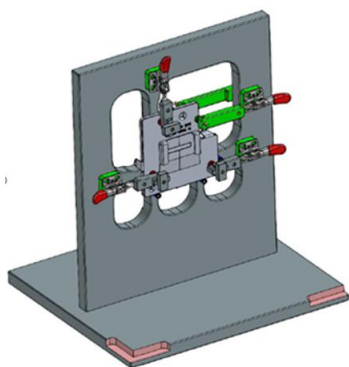
Ako srednja vrijednost određenog broja mjernih rezultata značajno odstupa od stvarne vrijednosti mjerene veličine, ona će imati značajnu grešku. Pri tome se pojedini rezultati mogu samo neznatno razlikovati, tj. mjerni postupak je vrlo precizan. Uzrok tome su stalni uticaji koji se kod mjernih rezultata pojavljuju kao tzv. sistemske greške ako su ponovljena mjerenja izvedena pri jednakim uslovima. Prema tome, preciznost nije garancija za malu

grešku. Obrnuta situacija, tj. velika tačnost bez manje preciznosti manje je vjerovatna. Stabilnost procesa se odnosi na predvidljivost procesa da ostane unutar kontrolnih granica. Sposobnost procesa je mjera sposobnosti procesa da zadovolji specifikacije kupaca. Mjera govori koliko je dobar svaki pojedinačni izlaz. Procjena defektnih dijelova pomoću indeksa je metoda za merenje sposobnosti procesa. Sigma nivo mjeri sposobnost procesa da li proces obavlja poslove bez grešaka. Filozofija Sigma nivoa prepoznaje vezu između broja nedostataka na proizvodu, izgubljenih troškova kao i novi zadovoljstva kupaca. Uobičajni mjerni indeks je defekt po jedinici i može uključivati bilo šta, od komponente, dijela materijala ili linije koda, do administrativnog oblika, vremenskog okvira ili udaljenosti. Sigma vrijednost pokazuje koliko često se mogu javiti defekti. Što je veća sigma vrijednost, manja je verovatnoća da će proces proizvesti nedostatke [3].

Ovaj rad ima za cilj da prikaže životni ciklus poklopca dobijenog brizganjem plastike kao dijela sklopa elektronske kontrolne jedinice sa akcentom na ocjenu kvaliteta dimenzionalnih i geometrijskih karakteristika.

2.1. Mjerni koncept

U cilju ispunjavanja funkcionalnosti i tehničkih karakteristika proizvoda prema zahtjevima kupca tokom faze nastanka uzoraka brizganjem plastike dimenzije su mjerene i verifikovane mjernim sistemima. Sve funkcionalne dimenzije su mjerene koordinatnom mjernom mašinom. Dimenzije koje nije bilo izvodljivo mjeriti sa KMM mjerene su bezkontaktnim mjernim uređajima 3D skenerom i optičkim mikroskopom. Prije nego što je započeto bilo kakvo mjerenje, potrebno je bilo uraditi koncept mjerenje na KMM. Pored mjernog koncepta dobavljač je bio dužan da projektuje i izradi mjerni pribor za KMM.



Slika 1. Mjerni pribor



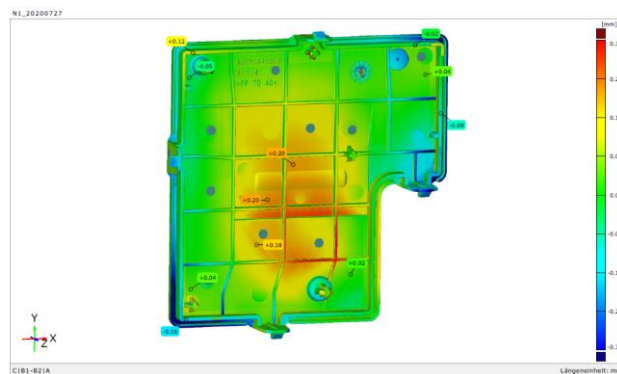
Slika 2. Predmet merenja - poklopac

2.2. Mjerni izvještaji i sposobnost procesa

Mjerni izvještaj sadrži oznaku pozicije na crtezu, redni broj dimenzije, nominalne vrijednosti dimenzije, tolerancijsko polje, izmjerene vrijednosti uzoraka, mjerni uređaj i potvrdu da li je izmjerena vrijednost u tolerancijama ili ne. Ciklus uzorkovanja može da traje sve dok u mjernom izvještaju ne bude svaka dimenzija u tolerancijama prema specifikaciji. Ukoliko u mjernim izvještajima imamo dimenzije koje su izvan tolerancija, razmatra se da li one mogu da se prihvate. Ako tolerancijski proračunom dokažemo da povećanjem tolerancijskog polja proširiti i da nema nikakvog uticaja na funkcionalnost proizvoda tada pristupamo istom.

Skeniranja koja se rade 3D skenerom dobavljač dostavlja uz mjerni izvještaj. Skenovi u IGES formatu se uvoze u Catiju i vrši se poređenje izmjerenih tačaka i nominalnog modela. Odstupanja nam pokazuju koliko i na kojim mjestima imamo krivljenje dijela tokom procesa brizganja. Nezavisno od industrije, kupci imaju zahtjeve kvaliteta i kapaciteta, i ti zahtjevi moraju da budu ispunjeni. Tokom „Run at Rate“ (Run@Rate) procesa dobavljač dokazuje i demonstrira da je njegov proizvodni proces sposoban da proizvodi dijelove prema zahtjevima kupca.

Specijalne karakteristike su funkcionalne karakteristike proizvoda koje treba da budu kontrolisane tokom proizvodnje. U toku „Run at Rate“ faze vrši se mjerenje svih specijalnih karakteristika označenih na crtežu. Kad se proizvod pusti u proizvodnju specijalne karakteristike se kontrolišu mjerenjem određenog broja uzorka na određen broj sati tokom smjene.



Slika 3. Rezultat skeniranja poklopca

PPAP (*eng. Production Part Approval Process*) definiše generičke zahteve za odobrenje proizvodnih delova. To znači da proizvođači dijelova dokumentuju sposobnost da se ispunjavaju specifikacije proizvoda prilikom proizvodnje samog proizvoda na navedenim nivoima proizvodnje. Dokumentacija koju dobavljač mora da dostavi prije puštanja proizvoda u proizvodnju je: Zapise o dizajnu, tok procesa, test čistoće, analiza procesa PFMEA, zapise o održavanju alata, mold flow analiza dizajna, upustvo za rad, specifikacija pakovanja, laboratorijski izvještaji, lista rezervnih dijelova, slike alata, parametri brizganja, IMDS međunarodni sistem podataka o materijalima, rezultati testova materijala, mjerni izvještaj, PSW (*eng. Product Supplier Warranty*), sposobnost procesa, analiza mjernog koncepta.

3. ANALIZA ISTRAŽIVANJA

Analizirali smo rezultate dimenzija specijalnih karakteristika iz mjernog izvještaja i studije sposobnosti procesa.

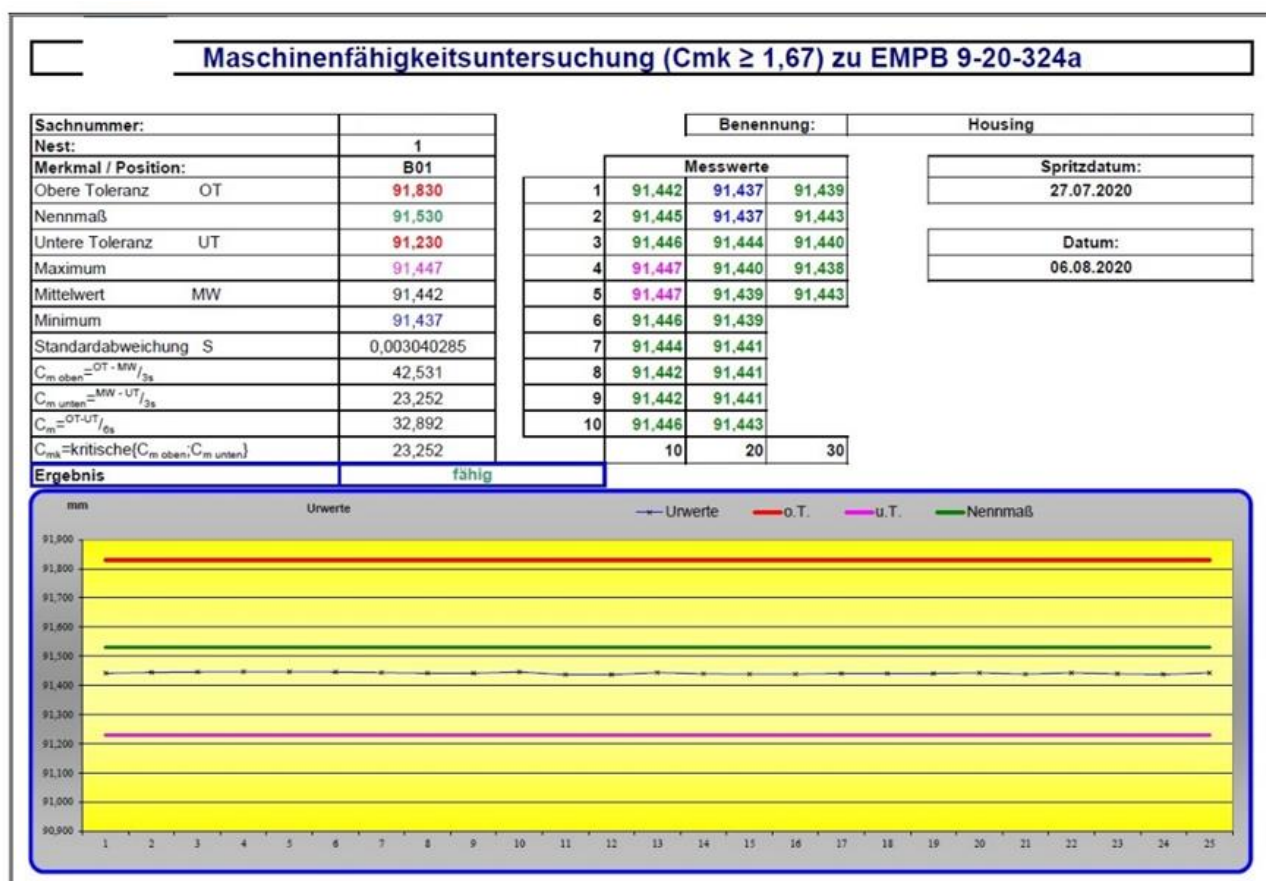
Kao reprezentativnu dimenziju uzeli smo specijalnu karakteristiku B01 čija je vrijednost $91,53 \pm 0,3\text{mm}$ iz šupljine alata N1 (slika5).

Da napomenemo da je studija sposobnosti procesa rađena samo za specijalne karakteristike.

Na 25 uzoraka dobijenih tehnologijom brizganja iz svake šupljine alata dobavljač je radio je studiju sposobnosti procesa sa zatjevanim indeksom sposobnosti $C_{mk} \geq 1.67$.

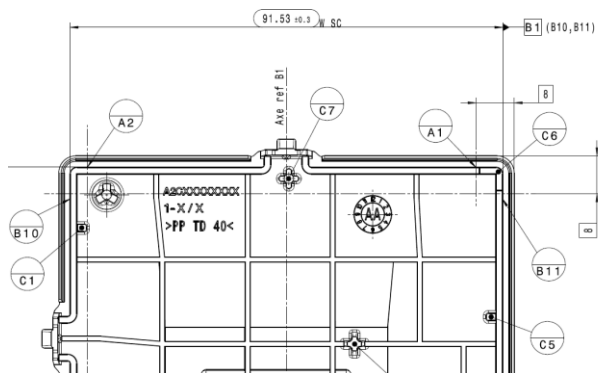
location on drawing	Item #	dimension / specification (incl. limits)	upper Tol.	lower Tol.	cavity 1 / equipment #	Cav. 1 P1 supplier measurement result	Cav. 1 P2 supplier measurement result	Cav. 1 P3 supplier measurement result	Cav. 1 P4 supplier measurement result	Cav. 1 P5 supplier measurement result	measurement equipment	supplier evaluation OK	n. OK	Comment (supplier and/or Customer)	Continental release Yes	No
		Page 1 of 3														
A01	flatness	1.0	0.0	-1.0		0.101	0.061	0.112	0.056	0.031	KMG	X				
B01	W SC	91.53	0.3	-0.3		91.505	91.512	91.504	91.513	91.507	KMG	X				
B02		8.0				7.999	7.998	7.998	7.999	7.999	KMG					
B03	B10	8.0				8.001	8.0	8.0	7.999	8.0	KMG					
B03	B11	8.0				8.001	8.0	8.0	7.999	8.0	KMG					
B04		6.0				6.001	5.999	5.999	5.999	5.999	KMG					
B05		11.5				11.5	11.5	11.5	11.5	11.5	KMG					
B06	W SC	68.53	0.3	-0.3		68.494	68.497	68.496	68.498	68.496	KMG	X				
B07	A2	6.0				5.998	6.0	6.001	6.0	6.0	KMG					
B07	A3	6.0				5.999	6.0	6.001	6.001	6.001	KMG					
B08	B20	6.0				6.0	6.0	6.0	6.0	6.0	KMG					
B08	B21	6.0				6.0	6.0	6.0	5.999	6.0	KMG					
B10	W SC A1-A4	91.53	0.3	-0.3		91.446	91.447	91.443	91.449	91.445	KMG	X				
B10	W SC A2-A3	91.53	0.3	-0.3		91.508	91.508	91.507	91.507	91.504	KMG	X				

Specijalne karakteristike prikazane u mjernom izvještaju



Sposobnost procesa za specijalnu karakteristiku B01 $91,53 \pm 0,3\text{mm}$ poklopca iz šupljine alata 1, $C_{mk} \geq 1,67$

Slika 4. Sposobnost procesa na primjeru jedne specijalne karakteristike



Slika 5. Specijalna karakteristika $91,53 \pm 0,3$ označena na crtežu poklopca

Na slici 4 prikazani su specijalne karakteristike mjernog izvještaja i studija sposobnosti procesa specijalne karakteristike B01. Nakon analiziranih rezultata možemo da zaključimo da je proces stabilan i sposoban. Vrijednost indeksa koja je dobijena kao rezultat mjerenja je mnogo veća od zahtjevanog indeksa sposobnosti. Na isti način rađena je studija sposobnosti za ostale specijalne karakteristike označene na crtežu.

4. ZAKLJUČAK

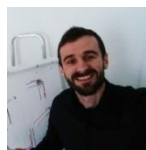
Ocjena kvaliteta dimenzionalnih i geometrijskih karakteristika sa specifikacijskim zahtevima skoro da je nezamisliva bez upotrebe nekog tipa koordinatnog mernog sistema. Zbog visoke tačnosti koordinatna mjerna mašina je idealna za mjerenje funkcionalnih karakteristika nekog proizvoda.

U ovom radu kroz mjerenje funkcionalnih dimenzija na koordinatnoj mjernoj mašini prikazana je studija sposobnosti procesa. Nakon potvrde da su sve dimenzije u tolerancijama i da je proces stabilan i sposoban poklopac je pušten u proizvodnju. Možemo da zaključimo da je ocjena kvaliteta dimenzionalnih i geometrijskih karakteristika jedan od neizostavnih segmenata životnog ciklusa proizvoda. Rad predstavlja uopšten prikaz životnog ciklusa proizvoda jedne kompanije i može da bude osnova za dalja istraživanja i primjenu metodologije Six Sigma.

5. LITERATURA

- [1] Cvetković, D.: Dizajn i razvoj proizvoda, Beograd, 2011.
- [2] Hodolič, J., Budak, I., Bešić, I., Vukelić, Đ., Osanna, H.P., Durakbasa, N.B.: Koordinatne merne mašine i CAD inspekcija, Novi Sad, 2009.
- [3] Paul J. Drake, Jr.: „Dimensioning and Tolerancing“, Handbook, McGraw-Hill, A Division of The McGraw-Hill Companies, 1999, ISBN 0-07-018131-4.
- [4] Burdek, B.E.: „History, Theory and Practice of Product Design“, Basel, 2005, ISBN 3-7643-7029-7.

Kratka biografija:



Saša Đukić rođen je u Slavonskom Brodu 1990. god. Master rad na Fakultetu tehničkih nauka iz oblasti Proizvodno mašinstvo odbranio je 2021 god.

kontakt: sasa.djukic@rocketmail.com



Dr Branko Štrbac rođen je u Novom Sadu 1983. Doktorirao je na Fakultetu tehničkih nauka 2017. god. i od tada je u zvanju docent. Oblast interesovanja su proizvodna metrologija.

MODULARNO PROJEKTOVANJE MAŠINA ALATKI ZA OBRADU METODOM BUŠENJA

MODULAR DESIGN OF MACHINES TOOLS FOR PROCESSING BY DRILLING METHOD

Darko Blanuša, *Fakultet tehničkih nauka, Novi Sad*

Oblast – PROIZVODNO MAŠINSTVO

Kratak sadržaj – Tema rada jeste promena modularnog projektovanja mašina alatki za obradu familije proizvoda metodom bušenja. Predstavljene su osnove i principi modularne gradnje mašina alatki, kao i prednosti koje omogućava ova metoda projektovanja.

Ključne reči: modularno projektovanje, mašine alatke, rekonfigurabilne mašine

Abstract – The topic of this research is the modular design of a machine tool for processing a product family by the drilling method. The basics and principles of modular construction of machine tools are presented, as well as the advantages provided by this design method.

Keywords: modular design, machine tools, reconfigurable machines

1. UVOD

Sa ekonomskim razvojem jedne države dolazi do porasta nacionalnog dohotka po glavi stanovnika, samim tim dolazi do povećanja procenta učešća industrije u ukupnoj privrednoj delatnosti države. U nastavku ovog rada biće prezentovano modularno projektovanje mašine alatke, koja ima svrhu unapređenja procesa obrade na određenoj familiji proizvoda. Takođe navedena mašina alatka ima mogućnost obrade proizvoda različitih geometrijskih oblika i dimenzija, ali istih tehnoloških zahteva.

1. 1. Cilj istraživanja

Cilj ovog rada je unapređenje procesa bušenja i urezivanja navoja na familiji proizvoda koji se proizvode u Fabrici kotrljajućih ležajeva i kardana Temerin (FKL). Istraživanje realizovano u okviru ovog rada je usmereno na pronalaženje produktivnijeg, ekonomičnijeg i fleksibilnijeg postupka obrade familije proizvoda.

2. OSNOVE PROJEKTOVANJA

Projektovanje mašina alatki je složena interdisciplinarna inženjersko razvojna-istraživačka delatnost koja je vrlo značajna za preduzeća koja se bave proizvodnjom mašina alatki kao i preduzeća koja koriste mašine alatke.

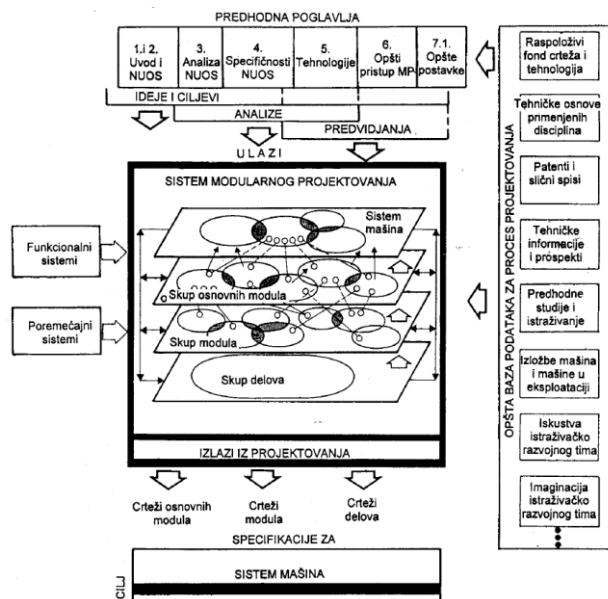
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Zeljković, red.prof.

Projektovanje proizvoda u širem smislu predstavlja dinamički skup svih interdisciplinarnih aktivnosti koje su povezane u jednu celinu. U industriji mašina alatki je posebno zastupljeno i najčešće korišćeno modularno projektovanje mašina alatki.

2.1. Osnovni koncept modularnog projektovanja

Modularno projektovanje je kompleksna aktivnost koja se bazira na velikom broju podataka, analiza i predviđanja koji su međusobno povezani, dopunjuju, ali isto tako ograničavaju jedni druge. Iz svega navedenog stiče se utisak da proces modularnog projektovanja predstavlja stalnu interakciju između navedenih činioaca. Na slici 1 je predstavljena šema kompleksnog sistema modularnog projektovanja.



Slika 1. Kompleksni sistem modularnog projektovanja mašina alatki [4]

Razvoj modularnog projektovanja se veže za početak tridesetih godina dvadesetog veka. Kroz dugi niz godina razvoja principa modularnog projektovanja dolazilo je do modernizacije, usavršavanja, proširivanja samog koncepta, da bi danas za rezultat dobili jednu od najčešće korišćenih metoda za projektovanje proizvoda.

Modularno projektovanje je našlo primenu u mnogim oblastima kao što su: industrija mašina alatki, industriji reznog alata i pribora, fleksibilne tehnološke sisteme i sl.

Osnovni koncept modularnog projektovanja mašina alatki zasniva se na grupi izmenljivih modula, projektuje mašina alatka različitih tehnoloških mogućnosti. Prilikom projektovanja potrebno je izabrati standardizovane module, čime se obezbeđuje zamenljivost modula kao i pravilna funkcija.

Do sredine šezdesetih godina prošlog veka, modularno projektovanje se primenjivalo u oblasti transfer linija pod akronimom projektovanje sistema pomoću blokova. Na osnovu prethodno navedenog, definicija takvog projektovanja glasi:

Mašina alatka sa novim funkcijama i strukturnom konfiguracijom može biti proizvedena biranjem i integrisanjem jedinica, pri čemu se u obzir uzimaju sve specifikacije koje mašina alatka mora da zadovolji.

Standardizovane jedinice definišu sledeće činjenice:

- Svaka jedinica poseduje osnovnu funkciju ili smislenu funkcionalnost
- Svaka jedinica mora imati dimenzionalnu I konfiguracijsku specifikaciju radi mogućnosti spajanja sa drugim jedinicama, čime se obezbeđuje rekonfiguracija.

Danas se umesto standardnih jedinica koriste moduli različitog nivoa složenosti: konstrukcioni, funkcionalni i kinematski. Navedena sistematizacija se bazira na koncepcionom rešenju određene mašine alatke. Na slici 2 predstavljena je hijerarhijska struktura modularno projektovanog obradnog centra.



Slika 2. Hijerarhijska struktura modularno projektovanog obradnog centra [3]

3. DEFINISANJE GRUPE PROIZVODA KOJI SE OBRADUJU METODOM BUŠENJA I POSEDUJU SLIČNE TEHNOLOŠKE ZAHTEVE

Proces transformacije polufabrikata u gotov proizvod u pojedinim situacijama može biti veoma složen, pri čemu u toku procesa obrade dolazi do dejstva elemenata mašine alatke (reznog alata i pribora) na obradak. Osnova za proces projektovanja proizvoda predstavlja najadekvatniji pripremak odnosno polufabrikat, na kojem je potrebno izvršiti najmanje tehnoloških operacija kako bi se dobio gotov proizvod. Proces obrade polufabrikata zavisi od nekoliko bitnih faktora koje je potrebno uzeti u obzir prilikom projektovanja proizvodnje. Faktori koji utiču na proces obrade polufabrikata su:

- raspoloživa tehnološka oprema;
- potrebne količine proizvoda (pojedinačna, maloserijska, srednje serijska ili masovna proizvodnja);
- raspoloživa radna snaga (kvalifikovana ili nekvalifikovana radna snaga);
- raspoloživi rezni alati;
- raspoloživi stezni pribori;

U nastavku rada će biti predstavljeni proizvodi koji imaju iste tehnološke zahteve, ali su različiti po geometrijskom obliku, dimenzijama, oblasti primene kao i materijalu od kog su izrađeni (Slika 3). Navedeni primeri imaju primenu u oblastima kao što su industrija mašina i opreme, industrija poljoprivredne mehanizacije, cevne armature, industrija ležajeva, industrija linearne tehnike, industrija pneumatike.



Slika 3. Izradci koji imaju iste tehnološke zahteve

4. ANALIZA FLEKSIBILNIH TEHNOLOŠKIH MODULA SA SLIČNIM TEHNOLOŠKIM ZAHTEVOM

U cilju unapređenja procesa proizvodnje navedene familije proizvoda, potrebno je modularno projektovati, a zatim konstruisati i izraditi mašinu alatku koja svojim performansama omogućava realizaciju što većeg broja tehnoloških zahvata. Takođe potrebno je izvršiti istraživanje tržišta mašina alatki, kako bi se ustanovilo da li postoje slične ili identične mašine alatke koje mogu da zadovolje potrebe obrade navedene familije proizvoda. Prednost modularnog projektovanja, između ostalog, se ogleda u brzini kojom se omogućava projektovanje i izrada mašine alatke zahtevanih karakteristika.



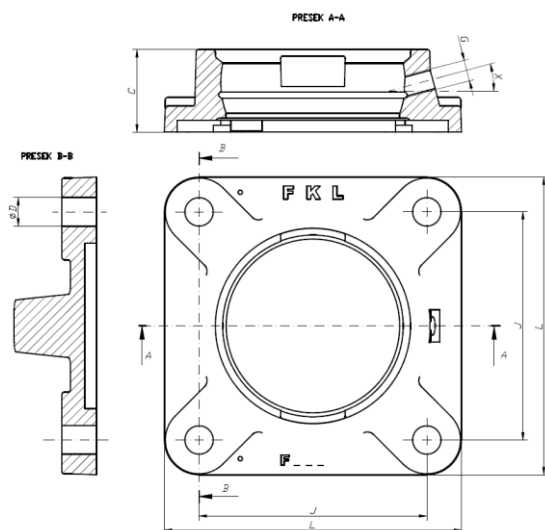
Slika 4. Fleksibilni tehnološki modul za obradu bušenjem i urezivanjem navoja, proizvođača "Patson Machines"

4.1. Definisanje tehnološkog zahteva za obradu familije "F kućišta" metodom bušenja i urezivanja navoja

Familiju "F kućišta" (Slika 5) čine proizvodi, odnosno kućišta koja se razlikuju po veličinama i nose oznaku u rasponu od F203 ÷ F220.

Na slici 6 predstavljena je grupa kućišta iz navedene familije proizvoda, kao skup delova na osnovu kojeg treba modularno projektovati mašinu alatku za njihovu obradu. Na osnovu navedene slike 6 može se zaključiti sledeće:

- gabaritne dimenzije proizvoda su različite po veličinama;
- raspored četiri aksijalna otvora u odnosu na centralni otvor kućišta su različita za svaku veličinu;
- prečnici četiri aksijalna otvora su isti za jednu veličinu kućišta, ali se razlikuju po veličinama kućišta;
- dimenzija i vrsta navoja su iste kod svih veličina kućišta;
- pozicija otvora za navoj se razlikuje u zavisnosti od veličine kućišta;
- ugao pozicioniranja otvora za navoj se razlikuje u zavisnosti od veličine kućišta.



Slika 5. Šematski crtež kućišta F familije

DIMENZIJE F KUĆIŠTA PROIZVODAČA "FKL TEMERIN"							
Redni br.	Oznaka kućišta	Dimenzije kućišta LxL [mm]	Visina kućišta (C) [mm]	Raspored otvora JxJ [mm]	Prečnik otvora (D) [mm]	Ugao otvora (X) [°]	Velicina navoja (G) [T]
1.	F 203	76 x 76	26	54x54	Ø11,5 H13	20	1/4" - 28 UNF
2.	F 204	86 x 86	25,5	64 x 64	Ø12 H13	20	
3.	F205	95 x 95	27	70 x 70	Ø12 H13	20	
4.	F 206	108 x 108	31	83 x 83	Ø12 H13	20	
5.	F 207	118 x 118	34	92 x 92	Ø12 H13	20	
6.	F 208	130 x 130	36	102 x 102	Ø16 H13	20	
7.	F 209	137 x 137	38	105 x 105	Ø16 H13	15	
8.	F 210	143 x 143	40	111 x 111	Ø16 H13	15	
9.	F 211	162 x 162	43	130 x 130	Ø19 H13	15	
10.	F 212	175 x 175	48	143 x 143	Ø19 H13	15	
11.	F 213	187 x 187	50	149 x 149	Ø19 H13	15	
12.	F 214	193 x 193	50,3	152 x 152	Ø19 H13	20	
13.	F 215	200 x 200	53,6	159 x 159	Ø19 H13	20	
14.	F 216	208 x 208	54,5	165 x 165	Ø23 H13	20	
15.	F 217	220 x 220	58,5	175 x 175	Ø23 H13	20	
16.	F 218	235 x 235	63,4	187 x 187	Ø23 H13	15	
17.	F 2120	265 x 265	70	210 x 210	Ø27 H13	20	

Slika 6. Dimenzije F kućišta koje su važne za modularno projektovanje mašine alatke

5. PROJEKTOVANJE MAŠINE ALATKE ZA OBRADU BUŠENJEM OTVORA NA FAMILIJI "F KUĆIŠTA"

U poglavlju 4.1 su predstavljeni svi tehnološki zahtevi koji se odnose na obradu bušenjem i urezivanjem navoja na navedenoj familiji (skupu) proizvoda. Na osnovu analize slika 5 i 6, može se doći do zaključka da navedene zahteve može da ispuni samo mašina alatka izgrađena na modularnom principu. Kako bi se proces proizvodnje u preduzeću FKL unapredio u tehnološkom i ekonomskom smislu, rešenje problema za navedenu familiju proizvoda su modularno projektovane i izgrađene mašine koje služe za obradu navedene familije proizvoda.

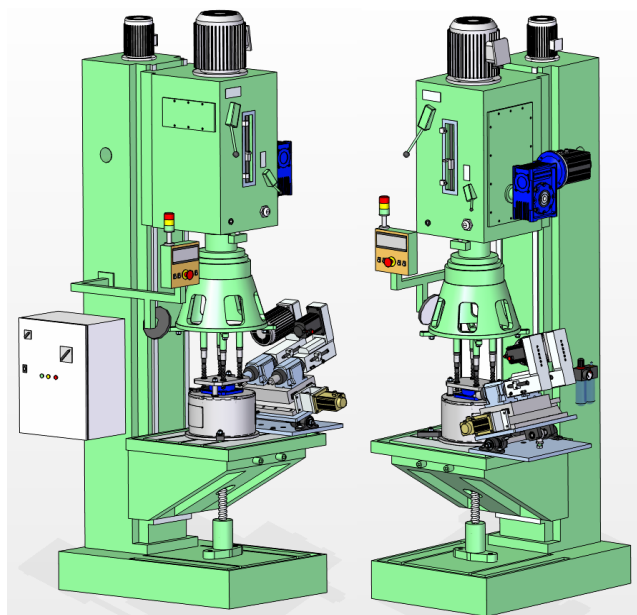
Mašina alatka se zasniva na modularnom principu gradnje mašina alatki, sa napomenom da konkretna mašina alatka nema sve module koje mogu da poseduju savremeni obradni tehnološki moduli. Navedenu mašinu sačinjavaju sledeći moduli:

- noseći sistem mašine
- moduli za glavno kretanje;
- moduli za pomoćno kretanje;
- modul sistema za upravljanje;
- modul za stezanje radnog predmeta /obratka;

Po posebnom zahtevu na mašinu je moguće ugraditi i sledeće module:

- modul za dovod i odvod sredstva za hlađenje i podmazivanje;
- modul za snabdevanje mašine vazduhom pod pritiskom;
- modul za snabdevanje mašine hidrauličnom energijom;

Na slici 7 predstavljena je mašina alatka, projektovana po modularnom principu, a respektujući trenutno raspoloživi modul nosećeg sistema. Namena mašine alatke je obrada prethodno prikazane familije F kućišta.



Slika 7. Modularno projektovana mašina alatka

5.1. Opis rada mašine

U toku procesa obrade radni predmet miruje, odnosno stegnut je i pozicioniran pomoću funkcionalnog modula za stezanje i pozicioniranje obratka. Kinematski modul za bušenje i kinematski modul viševretene glave istovremeno vrše obradu bušenjem u dve ravni. Proces bušenja otvora za navoj traje kraći vremenski period nego bušenje četiri aksijalna otvora. Nakon završetka bušenja, kinematski modul se vraća u prvobitnu poziciju, klizač se pomera i dovodi kinematski modul za navoj u osu sa izbušenim otvorom, nakon čega kinematski modul za navoj počinje da vrši obradu. Proces bušenja četiri aksijalna otvora i bušenje i urezivanje navoja jednog otvora trebali bi da traju približno isti vremenski interval, jer samo na taj način se prikazuju pravi rezultati navedenog rešenja obradnog sistema.

6. ZAKLJUČAK

Projektovana mašina alatke omogućava unapređenje tehnološkog procesa proizvodnje familije „F“ kućišta. Unapređenje procesa se može predstaviti kroz nekoliko činjeica:

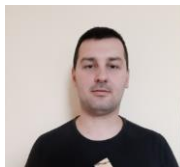
- smanjen utrošak radne snage;
- smanjen broj mašina alatki (u dosadašnjoj koncepciji obrade su potrebne tri mašine alatke);
- povećanje produktivnosti;
- smanjen broj potrebnih pribora za stezanje priprema,
- povećanje rentabilnosti;
- povećan stepen tačnosti izrade radnih predmeta (manji broj pozicioniranja i stezanja).

Razvoj mašine alatke zahteva početna ulaganja, ali za određeni vremenski period njena rentabilnost je znatno veća.

7. LITERATURA

- [1] Zeljković, M., Tabaković, S. Savremeni prilazi u projektovanju proizvoda (Autorizovani rukopis predavanja, školska godina 2014/2015), Fakultet tehničkih nauka, Novi Sad, 2015.
- [2] Gatalo, R., Zeljković, M., Borojev, Lj. Automatski fleksibilni i tehnološki sistemi (Autorizovan rukopis predavanja, školska godina 2013/2014), Fakultet tehničkih nauka, Novi Sad, 2012.
- [3] Yoshimi, I. Modular desing for machine tools, McGraw-Hill companec, Inc., USA, 2008.
- [4] Milojević, M., Lukić, Lj. Modularno projektovanje (Monografija), Lola Institut, Beograd, 1996.
- [5] Zahar, S. Mašine alatke 1, Jugoslovensko društvo za tribologiju, Kragujevac 1993.
- [6] Katalog proizvoda preduzeća "FKL Temerin", Temerin, 2000.

Kratka biografija:



Darko Blanuša rođen je u Somboru, 1995. god. Završio je osnovne studije na Departmanu za proizvodno mašinstvo Fakulteta tehničkih nauka u Novom Sadu 2018. godine, a master rad odbranio na istom fakultetu 2021. godine.

kontakt: darko.blanusa@yahoo.com

**UTICAJ PARAMETARA ZAVARIVANJA TRENJEM SA MEŠANJEM
SAMOPODESIVIM ALATOM NA EFIKASNOST SPOJA****INFLUENCE OF FRICTION STIR WELDING PARAMETERS WITH SELF-REACTIVE
TOOL ON JOINT EFFICIENCY**

Kristofer Solar, Sebastian Baloš, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – U radu je prikazan postupak zavarivanja trenjem sa mešanjem sa samopodesivim alatom. Navedena je problematika korišćenih materijala, geometrija i kinematika alata. Posebna pažnja je poklonjena alatima sa kalemom, odnosno, dvostranim alatima koji igraju i ulogu pribora i vrše zavarivanje osnovnog materijala sa obe strane formirajući i lice i naličje šava. Eksperimentalni rad je izvršen na šest uzoraka, sa dve geometrije alata, tri različite brzine obrtanja alata, uz jednaku brzinu zavarivanja, na tehnički čistom aluminijumu. Rezultati su pokazali da se kvalitetniji zavareni spojevi dobijaju u slučaju upotrebe alata sa većim uglom ramena i sa većom brzinom obrtanja alata.

Ključne reči: zavarivanje trenjem sa mešanjem, samopodesivi alat, mehaničke osobine, mikrostruktura

Abstract – In this work, friction stir welding with self-reacting tool is presented. Materials used for friction stir welding tools are presented, as well as the geometry and tool kinematics. A special attention is paid to self-reacting tools, two sided tools that also act as base plate and weld the base material by forming both the face and the root of the weld. Experimental work was done on six specimens, with two tool geometries, three different rotational speeds and a single welding speed, on technical aluminium. The results shown that a higher weld quality was obtained by applying the tool with a higher shoulder angle and a higher rotational speed.

Keywords: friction stir welding, self-reacting tool, mechanical properties, microstructure

1. UVOD

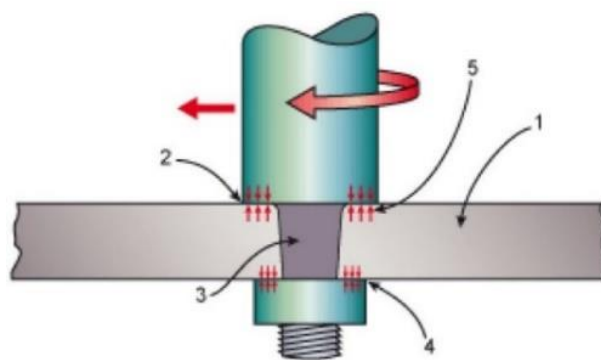
U mašinstvu posebnu pažnju dobijaju razvoj i praktična primena tehnologija, koje omogućavaju spajanje materijala. Toj oblasti pripadaju i različite tehnologije zavarivanja.

„Zavarivanje je proces nerastvorljivog spajanja istorodnih ili raznorodnih materijala, koje stvaraju međuatomske sile“ [1]. Zavarivanje trenjem sa mešanjem (eng. friction stir welding) je relativno nov postupak zavarivanja. Otkriven je 1991. godine na institutu The Welding Institut (TWI). Metoda je postala revolucionarna

za zavarivanje teško zavarljivih materijala, kao što su pojedine legure aluminijuma, magnezijuma i raznorodnih materijala. Trenutno je metoda široko primenjena u raznim industrijama, na primer avio, svemirskoj, železničkoj, automobilske industriji, kao i u brodogradnji [2].

Zavarivanje trenjem sa mešanjem (ZTM) je postupak koji se izvodi pod visokim pritiscima, pri temperaturama od oko $0,8T_{\text{topl}}$, pri čemu je materijal u čvrstom stanju, ali razmekšan. Toplota i zavareni spoj se ostvaruju pomoću namenskog alata. Alat čine čelona površina i trn alata, koji dolaze u kontakt sa osnovnim materijalom (OM). Zavareni spoj se realizuje pomoću toplote, koja se generiše trenjem zbog translatornog i rotirajućeg kretanja alata i mešanjem koji vrši trn alata (slika 1) [3,4,5].

Samopodesivi alat (alat u obliku kalema) sastavljen je od dva čela, gornjeg (sa strane lica šava) i donjeg (sa strane naličja šava). Radne površine su spojene trnom, slika 1 [6].



Slika 1. Šema alata u obliku kalema: 1- osnovni materijal, 2- čelo gornjeg dela, 3-trn, 4- čelo donjeg dela, 5- reaktivne sile [7]

2. EKSPERIMENTALNI DEO

Cilj eksperimentalnog dela je dobijanje kvalitetnog zavarenog spoja bez tunela i drugih grešaka, primenom ZTM sa samopodesivim alatom.

Za istraživanje, upotrebljen je tehnički čist aluminijum ploče debljine 4,9 milimetara. Pomoću hemijske analize postupkom optičke emisije spektrometrije (OES) uređajem ARL3580, potvrđeno je da je materijal tehnički čisti aluminijum 98 %. Obavljeno je ispitivanje zateznih karakteristike OM-a na mehaničkoj kicalici WPM, model ZDM 5/91.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor dr Sebastian Baloš, red.prof.

Tabela 1. Tehnički čist aluminijum 98 % [maseni %]

Cu	Mn	Mg	Si	Fe	Zn	Ti	Al
0,07	0,13	0,59	0,23	0,38	0,055	0,020	98,5

Tabela 2. Rezultati ispitivanja tehnički čistog aluminijuma oznaka Al 98 %

Redni broj epruvete	Rm [MPa]	Rp0,2 [%]	A [%]	Z [%]	HV 0.1
1.	139,9	122,4	20	64,2	41,9
2.	143,5	127,5	16	62,7	42,6
3.	128,2	107	12,2	56,6	42,9
Srednja vrednost	137,3	119	16	61,2	42,4

Cilj je poboljšavanje toka mešanja, kao i kvaliteta šava. Nužno je izabrati najpovoljniji materijal za izradu, kako bi alat bio otporan na višestruka naprezanja kao što su: toplota, dinamičke sile, hemijski efekti itd. Izabrani materijal je alatni čelik za rad na toplo AISI H13 (EN X40CrMoV5-1, JUS Č4753, UTOP MO2).

Korišćeni su alati fiksne koncepcije, koja je sa aspekta izrade jeftina i jednostavna (slika 2). Rastojanje između dva čela, kao i pomeranje po vertikali je nepromenljivo. Čela su konveksna, sa uglom 2° kod alata A2 i 4° kod alata A4. Poprečni presek trna je kvadrati, dimenzija 6 x 6 mm i visine 4,5 mm. Alati su kaljeni u ulju i visokotemperaturno otpušteni.



Slika.2 Prikaz alata u obliku kalema A4 i A2

U eksperimentalnom delu rada korišćena je poluautomatizovana vertikalna glodalica tipa PRVOMAJSKA FSSGVK-3.

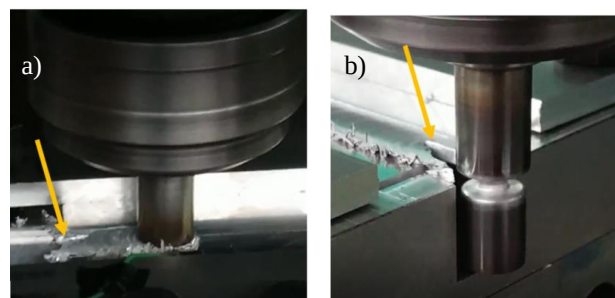
Brojevi obrtaja oba alata su bili 900, 1120, 1400 min⁻¹, sa identičnim translatornom brzinom kretanja. U tabeli 3 je prikazan i sistem označavanja uzoraka brojevima.

Ploče za ispitivanje su isečene na dimenzije 150mm x 60 mm i čelona površine obrađeni na glodalici kako bi dobilo naleganje bez zazora. Takođe, površine ploča su poravnane i debljina im je smanjena na 4,9 mm. Pre ZTM površine ploče su očišćene.

Nakon pripreme, ploče su stavljene i fiksirane u pribor. Potom sledi pozicioniranje alata i zavarivanje. Ulazni i izlazni žljeb je prikazan na slici 3.

Tabela 3. Parametri zavarivanja

Alat sa uglom čelone površine	Redni broj uzorka	Parametri ZTM	
		Broj obrtaja [min ⁻¹]	Brzina zavarivanja (pomak) [mm/s]
A 2°	1	900	20
	2	1120	
	3	1400	
A 4°	4	900	
	5	1120	
	6	1400	



Slika 3. a) ulazni žljeb, b) izlazni žljeb

Na dobijenih uzorka su obavljene sledeće ispitivanje :

- Vizuelna kontrola se odnosi na izgled šava
- Ispitivanje zateznih karakteristika izvedeno je po standardu SRPS EN ISO 5178:2013.
- Ispitivanje na savijanje je urađeno prema standardu SRPS EN ISO 5173:2013.
- Ispitivanje tvrdoće je obavljeno prema standardu SRPS EN ISO 6507-1:2011.
- Metalografska ispitivanja su izvršena na svetlosnom mikroskopu
- Ispitivanje energije udara je urađeno prema standardu SRPS EN ISO 148-1 2016.

Od svakog uzorka, isečene su dve epruvete za ispitivanje zateznih karaktersitika, dve epruvete za ispitivanje energije udara, jedna za savijanje, jedna za metalografska ispitivanja, a na istoj toj je izvršeno i merenje mikrotvrdoće.

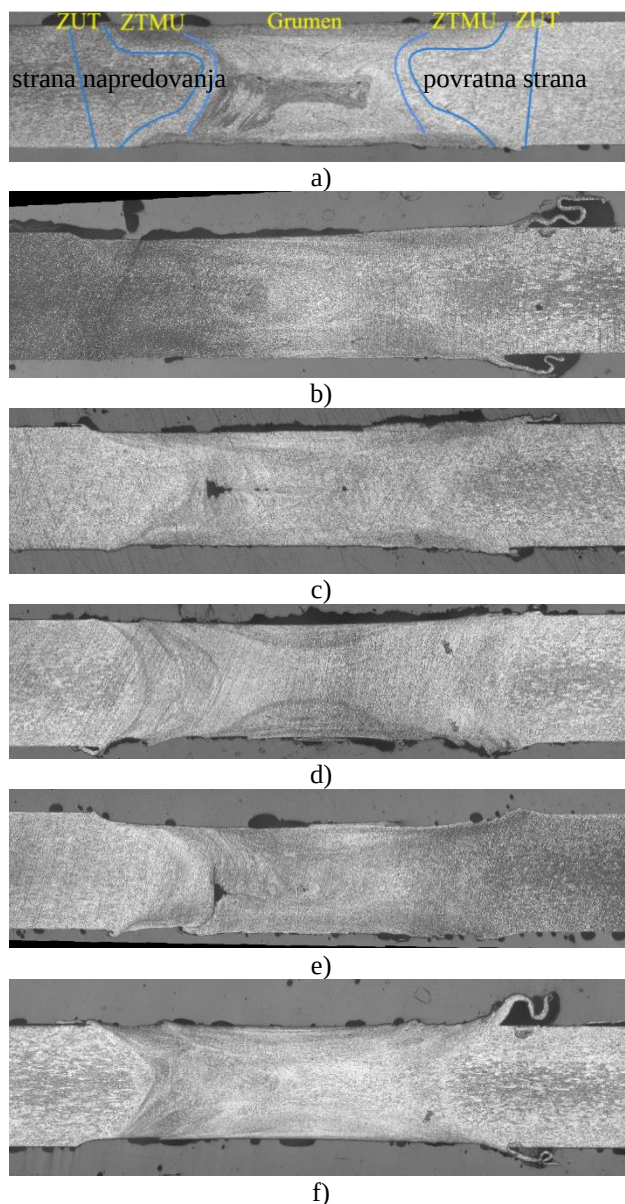
3. REZULTATI I DISKUSIJA

Na slici 4 je prikazan makro izgled šava sa različitim zonama (grumen, ZUT, ZTMU) koji je sličan kod svih uzoraka. Kod uzorka 3 and 5, prisutan je tunel, kod uzoraka 1, 2, i 4 mikrotuneli (svi na strani napredovanja, što je uobičajeno), dok je jedini uzorak bez tunela uzorak 6.

Grumen je uža na sredini i širi prema licu i korenu šava ima oblik peščanog sata. U grumenu je plastična deformacija najznačajnije izražena. Sastoji od reskristalizovane sitnozrne mikrostrukture, prikazano na slici 5a. U ZTMU pored grumena je prisutna mikrostruktura izduženih zrna kao rezultat kombinovanog dejstva toplote i plastične deformacija (slika 5b). U zoni ZUT (slika 5c) je prisutna rekristalizovana krupnozrna struktura.

Poređenjem rezultata zateznih čvrstoća, dobijeni rezultati ukazuju, da između upotrebe dva različita alata nema značajnih statističkih razlika.

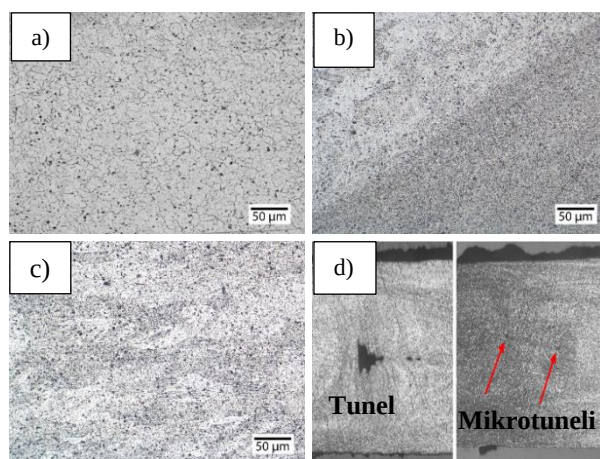
Mesto loma kod svih uzoraka je zona ZUT/ZTMU sa povratne strane (PS) (osim uzorci 3 i 5), koja predstavlja najkritičniji deo šava zbog smanjene brzine prerade materijala (brzina obrtanja i brzina zavarivanja se oduzimaju).



Slika 4. Makro prikaz uzoraka: a) uzorak 1, b) 2, c) 3, d) 4, e) 5, f) 6

Kod uzorka 3 i 5 došlo je do loma blizu sredine, sa napredne strane (NS) gde je primećen tunnel značajnih dimenzija od oko 1 mm. Na osnovu ovoga se konstatuje, da je zatezna čvrstoća u bliskoj vezi sa prisustvom tunela. Najveću srednju vrednost zatezne čvrstoće ima uzorak 5, koji je dobijen sa alatom A4, sa brojem obrtaja 1120 min^{-1} . Iako je primećen najveći tunnel kod uzorka 5, dobijene su više zatezne karakteristike. Jedno od objašnjenje je da je tunnel mestimičan i da kod uzoraka za zatezanje tunela nije bilo. Iako kod uzorka 6 nema tunela, zatezna čvrstoća je manja nego kod uzorka 4 i 5. Najverovatnij uzrok je veća temperatura usled većeg broja obrtaja koja je uzrokovala manju viskoznost tečnog metala i bolje popunjavanje prostora oko trna. U proseku, zatezne čvrstoće kod uzoraka dobijenih alatom A4 su veće nego kod uzoraka

dobijenih alatom A2, verovatno zbog većeg stepena deformacije zbog upotrebe alata sa većim uglom ramena. Pri ispitivanju savijanjem, pet uzoraka je savijano do 180° bez pojave prsline, a kod uzorka 5 se pojavila prsline pri $14,3^\circ$, tabela 5. Ovaj uzorak ima značajan tunnel koji je uzrok prsline, koji se širi prema korenu gde je prisutna greška tipa „kissing bond“, odnosno nepotpuno spajanje materijala., slika 4e. U poređenju sa zavarenim spojevima dobijenim konvencionalnim alatom bez donjeg ramena, gde se koren formira u interakciji trna i podložne ploče, kod samopodesivog alata, sam alat formira koren, što je znatno povoljnije, jer je takva izvedba kruća.



Slika 5. Mikrostrukturne karakteristike: a) grumena, b) ZTMU (dole desno) i ZUT (gore levo), c) osnovni materijal, d) tunela u uzorku 3 i mikrotuneli u uzorku 2

Tabela 4. Rezultati ispitivanja zatezanjem

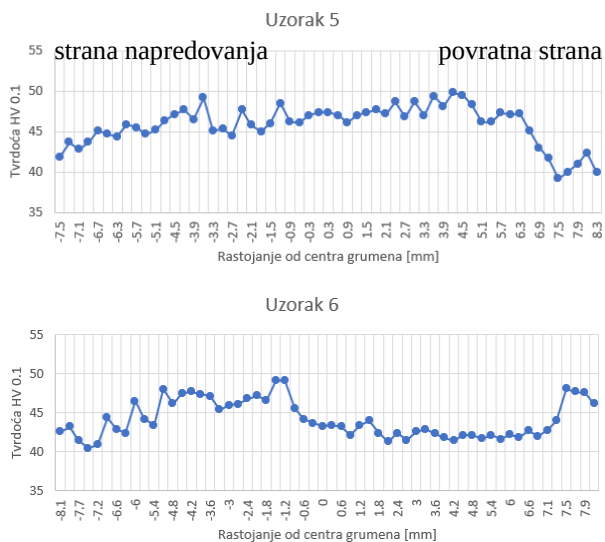
Alat	Uzorak	Srednja vrednost R_m [MPa]	Mesto loma	Strana loma
A 2°	1	112,10	ZUT/ZTMU	Povratna S.
	2	113,99	ZUT/ZTMU	Povratna S.
	3	110,88	grumen/ZTMU	S.Napredovanja
A 4°	4	114,44	ZUT/ZTMU	Povratna S.
	5	114,90	ZUT/ZTMU	Povratna S.
	6	112,36	ZUT/ZTMU	Povratna S.

Tabela 5. Prikaz podataka ispitivanjem savijanjem

Alat	Uzorak	Pojava prsline	Savijanje do 180°
A2	1	Nema prsline	Nema loma
	2	Nema prsline	Nema loma
	3	Nema prsline	Nema loma
A4	4	Nema prsline	Nema loma
	5	Prsline pri $14,3^\circ$	Nema loma
	6	Nema prsline	Nema loma

Na ispitivanim uzorcima, mikrotvrdoća je merena između lica i korena šava. Vrednosti su prikazane od rastojanja centra grumena, gde 0 označava centar grumena. Na uzorku 5, jasno se vidi da je na povratnoj strani tvrdoća nešto veća na povratnoj strani, dok je kod uzorka 6 obrnuto. Uobičajeno je da je tvrdoća veća na strani napredovanja, jer je u toj zoni intenzivnija deformacija zbog sabiranja brzine obrtanja alata i brzine zavarivanja. Međutim, kod uzorka 5 je verovatno prisutan uticaj tunela, odnosno, ivice materijala prema šupljini tunela.

Ista tendencija je primećena i kod uzorka 3, gde je takođe prisutan tunel, dok je kod ostalih uzoraka raspodela kao kod uzorka 6. Takođe, vidi se da su tvrdoće u zonama šava veća u odnosu na osnovni materijal.



Slika 6. Rezultati ispitivanja mikrotvrdoće uzoraka 5 i 6

Šav dobijen sa ZTM poseduje povećanu otpornost na udarno opterećenje, tj. veću žilavost u odnosu na OM. Najveću vrednost ima šav uzorka 4, dobijen korišćenjem alata A4, sa parametrima broja obrtaja 1120 min⁻¹ i pomakom od 20 mm/min. U sledećoj tabeli su prikazane vrednosti energije udara OM i šava. Jasna je tendencija smanjenja energije udara sa povećanjem brzine obrtanja alata, takođe kao posledica povećanog unosa toplote i manjeg stepena usitnjavanja zrna. Izuzetak je uzorak 5, koji ima manju energiju udara u odnosu na uzorak 6, ali se u ovom slučaju može pretpostaviti da je na dobijenu vrednost energije udara uticala pojava tunela i kissing bond greške u korenu. Relativno niska vrednost je dobijena i kod uzorka 3, verovatno iz istog razloga.

Tabela 6. Rezultati ispitivanja energije udara

Alat	Broj uzorka	Energije udara, A [J]	
		OM	Grumen
A2	1	17,59	26,32
	2	19,41	24,55
	3	18,64	20,66
A4	4	18,64	27,79
	5	18,36	24,26
	6	19,60	25,43

4. ZAKLJUČAK

Na osnovu dobijenih rezultata može se zaključiti sledeće:

- Alat A4 se pokazao efikasnijim u odnosu na A2, jer veći ugao ramena izaziva povećan pritisak tokom procesa.
- Tunel se pojavljuje najverovatnije mestimično i detektovan je na svim uzorcima osim uzorka 6 dobijenog najvećom brzinom obrtanja, sa najvećim pritiskom, odnosno, potencijalno najvećom temperaturom u procesu.
- Nema grešaka u korenu šava zahvaljujući dvostranom alatu, osim kod uzorka 5, gde je

dobijena greška tipa kissing bonds, što se manifestovalo pri savijanju.

- Tvrdoća šava je delimično ili po celom preseku veća nego OM, veća je na strani napredovanja, osim ako je prisutan tunel većih dimenzija (uzorci 3 i 5)
- Šav dobijen sa alatom A4 sa brojem obrtaja 1400 min⁻¹ predstavlja optimalno rešenje.

5. ZAHVALNOST

Rezultati prezentovani u ovom radu su realizovani u okviru projekta „Inovativni materijali i tehnologije spajanja“, Departmana za proizvodno mašinstvo, FTN, Novi Sad.

6. Literatura:

- [1] V.M. Rübakov, N.P. Dimitrijević, *Acélszerkezetek hegesztése*, Műszaki könyvkiadó, Budapest 1968.
- [2] S. Baloš, L. Šidanić, D. Rajnović, V. Vučić, *Zavarivanje trenjem sa mešanjem*, XIX Konferencija o međulaboratorijskim ispitivanjima materijala, Užice 2.10.-4.10.2013, str. 66 – 72.
- [3] V. Palić, B. Sabo, *Tehnologija zavarivanja*, Fakultet tehničkih nauka, Univerzitet u Novom Sadu, Novi Sad, 2003.
- [4] S. Kralj, M. Bušić, Z. Kožuh, D. Klobčar, *Utjecaj oblika i dimenzija alata pri zavarivanju trenjem rotirajućim alatom*, Zavarivanje - Welding 2014, str. 296-306.
- [5] R. S. Mishra, M. W. Mahoney, *Friction stir welding and processing*, University of Missouri, Rolla, 2007, str. 6 – 25.
- [6] P. L. Threadgill, M. M. Z. Ahmed, J. P. Martin, J. G. Perrett and B. P. Wynne, *The use of bobbin tools for friction stir welding of aluminium alloys* - Thermec 2009. Berlin, Germany, 25 - 29 August 2009.

Kratka biografija:



Kristofer Solar rođen je u Senti 1995 god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Proizvodno mašinstvo – Tehnologije spajanja odbranio 2018 god.



Sebastian Baloš rođen u Somboru 1974. god. Doktorirao je na Fakultetu tehničkih nauka 2010. god., a 2021. godine je biran u zvanje redovnog profesora iz oblasti Mašinsko inženjerstvo, tj. uže naučne oblasti Materijali i tehnologije spajanja.

OKSIACETILENSKA METALIZACIJA PRAHOM TANKOZIDNIH ČELIČNIH CEVI OXY-ACETYLENE THERMAL COLD SPRAY OF THIN WALLED PIPES

Miljan Dragičević, Sebastian Baloš, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – Prilikom procesa proizvodnje kao i u procesu reparacije, zaštitni/reparacioni sloj se može nanositi različitim metodama. Metode srodne zavarivanju su navarivanje i metalizacija. Metalizacija je postupak koji daje veoma visok kvalitet površine i kao takav u najvećem broju slučajeva ne zahteva naknadnu mašinsku obradu. U ovom radu je prikazana realizacija hladnog postupka gasne metalizacije prahom, gde su nanoseni slojevi volfram karbida na podvrgnuti ispitivanju mikrostrukture i mikrotvrdoća.

Ključne reči: Oksiacetilenska metalizacija prahom, volfram karbid, metalografija, mikrotvrdoća

Abstract – During the production and the repair processes, the protective/reparation layer can be applied by different methods. Welding allied processes are surfacing and thermal spraying. Thermal spraying is a process that gives a very high quality of the surface and as such in most cases does not require subsequent machining. In this paper, cold process of powder gas metallization will be presented, where the applied tungsten carbide layers will be subjected to microstructure and microhardness testing.

Keywords: Thermal metallization, powder flame spraying, tungsten carbide, metallography, microhardness

1. UVOD

Radni vek mašinskih delova, elemenata i opreme u velikoj meri zavisi od načina njihovog eksploatacija. Prilikom izloženosti istih velikim udarnim silama, trenju, habanju ili uticaju spoljašnje sredine, dolazi do oštećenja njihovog površinskog sloja ili razaranja.

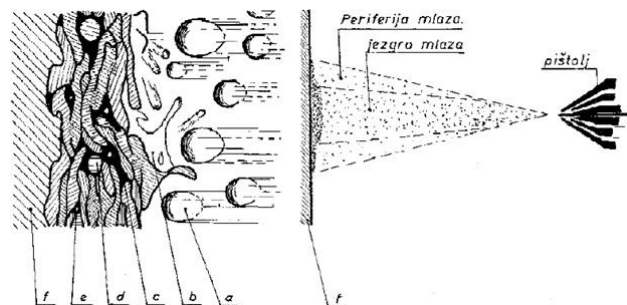
Kako bi se površine tih delova pre svega zaštitile, ojačale ili pak kako bi se oštećenje tih površina saniralo, koriste se metode slične zavarivanju, a to su navarivanje i metalizacija. Kako je poznato da se razvojem tehnologija razvio i veliki broj metoda nanošenja površinskih slojeva, proces gasne metalizacije prahom zauzima relativno visoko pozicionirano mesto zbog pristupačne i jeftine opreme, kao i zbog mogućnosti upotrebe velikog broja različitih dodatnih materijala.

Oksiacetilenska metalizacija prahom predstavlja postupak metalizacije koji se može svrstati u hladne, kao i u tople

postupke. Postupak pronalazi svoju široku primenu u nanošenju površinskih slojeva zbog mogućnosti upotrebe širokog asortimana materijala. Dodatni materijal kako i sam naziv kaže jeste u obliku praha.

Kao dodatni materijal mogu se koristiti sve vrste metala, karbida, polimera i keramike. Materijali koji nemaju afinitet prema kiseoniku predstavljaju veoma pogodna za realizaciju ovog procesa. Hladna metalizacija prahom predstavlja postupak nanošenja dodatnog materijala (istopljeni prah) na već pripremljenu relativno hladnu osnovnu površinu. Osnovna osobina jeste nerastvaranje površinskog sloja osnovnog materijala i mogućnost nanošenja relativno tankih prevlaka od 0,05 do 5 mm. Temperatura osnovnog materijala prilikom samog procesa ne prelazi 250 °C što ga i svrstava u hladni postupak. Sam proces otpočinje u pištolju za gasnu metalizaciju gde se prah kao dodatni materijal topi u plamenu smeše gorivog gasa (acetilen i kiseonik).

Tako istopljene i omekšane čestice putuju u struji plamen do osnovnog materijala gde se lepe za njega i formiraju sloj (slika 1) [1].



Slika 1 Prikaz stvaranja površinskog sloja prilikom postupka metalizacije [1]

Dodatni materijal u obliku praha prolazi kroz pištolj određenim kanalima, dolazi do plamena i topi se. Tako istopljen dodatni materijal u obliku kapljica dolazi do osnovnog materijala (f) i difuzionim silama se vezuje za njega. Prilikom kretanja kapljica pod uticajem temperature oksidiše (a) i nastavlja put do udaranja o podlogu i raspršivanja (b). Kako se već oksidni sloj stvorio oko kapljice, prilikom udara on puca i ostaje zarobljen u prevlaci (c) kao i gasni mehur (e).

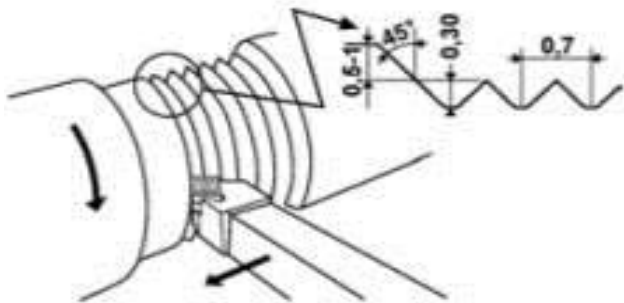
Čestice nakon udara i određenog vremena kreću da očvršćavaju (d). Kako je potrebno ostvariti dobru adheziju, odnosno dobru vezu među dodatnim i osnovnim materijalom, prvenstveno je potrebno pre svega pripremiti površinu osnovnog materijala na određenu hrapavost (f). Prilikom nanošenja prevlaka ovim postupkom dobijaju se slojevi poroznosti od 3 do 15% [1].

NAPOMENA:

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

2. EKSPERIMENTALNI DEO

Pre početka procesa metalizacije potrebno je pripremiti površinu osnovnog materijala prema preporukama koje je na osnovu svojih karakteristika pištolja dao proizvođač "CASTOLIN" (slika 2) [2].



Slika 2 Priprema površine struganjem [2]

Priprema površine osnovnog materijala (čelik S355) polaznog oblika Ø88,9 x 320 mm sastoji se iz površinske obrade struganjem prema sledećim parametrima narezivanja zavojnice [2]:

- korak zavojnice 0,7 mm,
- dubina 0,3 mm,
- ugao strugarskog noža 45° po strani,
- brzina obrtanja cevi 30 m/min.



Slika 3 Obrađen uzorak Ø88,9 x 320 mm

Koristeći prah kao dodatni materijal prvenstveno zahteva poznavanje njegovog hemijskog sastava, te se prilikom eksperimenta koristio prah "Eutalloy RW PE 8902", koji predstavlja leguru na bazi nikla sa 35% volframovog karbida (WC), slika 4. Odlikuje ga visoka otpornost na abrazivno habanje, kao i odlična otpornost na visoke temperature. Specifikacija korištenog praha [3] je sledeća:

- Hemijski sastav: Ni-4,25Si-15,6Cr-3,28B-4Fe+35% WC,
- Tvrdća: 55-63 HRC,
- Maksimalna radna temperatura: 700°,
- Produktivnost-brzina nanošenja: 5,8 kg/h,
- Potrošnja praha: 95 g/mm.

Kako je u samom eksperimentu vršeno nanošenje četiri vrste različitih prevlaka, samo jedna je sadržala 100 % navedenog praha (uzorak 0), mase 12,8 g. Ostala tri sloja nanosena postupkom gasne metalizacije prahom u sebi su

u određenoj količini prema težinskom omjeru sadržala i TiO₂ nanočestice:

- uzorak 1: 12,8 g "Eutalloy RW PE 8902" i 0,01 g 40 nm TiO₂ nanočestica,
- uzorak 2: 12,8 g "Eutalloy RW PE 8902" i 0,05 g 40 nm TiO₂ nanočestica,
- uzorak 3: 12,8 g "Eutalloy RW PE 8902" i 0,09 g 40 nm TiO₂ nanočestica.



Slika 4 Prah "Eutalloy RW PE 8902"

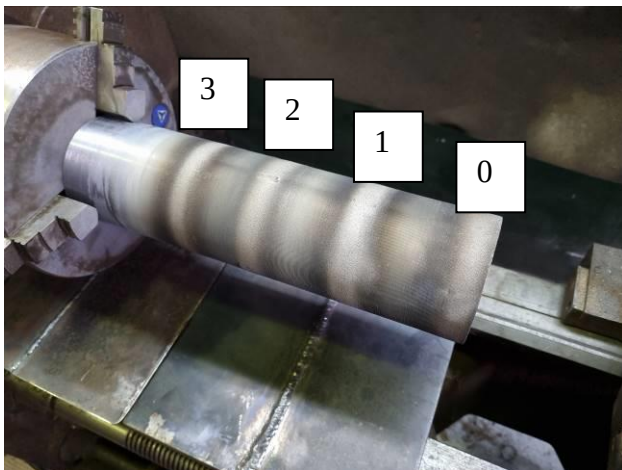
Pre samog početka procesa i nakon provere ispravnosti opreme, potrebno je izvršiti čišćenje i odmašćivanje već pripremljene površine osnovnog materijala. Prilikom izvođenja procesa korištena je oprema proizvođača "CASTOLINE" i praćene su njihove smernice. Prilikom realizacije eksperimenta korišten je pištolj „CASTOLIN - DS 8000“ (slika 5).

Prethodno obrađena cev se postavila na univerzalni strug, sa podešenom brzinom obrtanja 71 o /min, što je u skladu sa preporukama proizvođača uređaja i praha.



Slika 5 Pištolj „CASTOLIN - DS 8000“

Nakon što je završena realizacija postupka metalizacije, pristupa se ispitivanju nanešenih slojeva u cilju dobijanja što preciznih podataka o nanesenim slojevima. Jedna od metoda metalografskih ispitivanja jeste mikroskopska metoda koja je korištena u ovom ispitivanju.



Slika 6 Četiri vrste različitih slojeva (svetle pruge na uzorku, oivičene tamnom bojom usled dejstva toplote), s desna na levo: uzorak 0, 1, 2 i 3

Pre ispitivanja potrebno je pripremiti uzorke koji će se koristiti. Mikroskopsko ispitivanje već pripremljenih uzoraka omogućava da se odredi tip mikrostrukture, veličina zrna i njihova orijentacija, kao i veličina uključaka. Metalografski svetlosni mikroskopi rade na principu odbijanja svetlosti. Svetlosni zraci koji padaju na ravnu površinu odbijaju se i pod istim uglom dolaze do posmatračevog oka. Ta tačka se na mikroskopskom snimku vidi kao bela površina.

Ukoliko se koristi nagrižena površina koja je neravna, svetlosni snop se u veoma maloj meri odbija od te površine i dolazi do posmatračevog oka, gde na mikroskopskom snimku predstavlja crnu površinu [4]. Prilikom ispitivanja korišten je svetlosni mikroskop "Leitz Ortoplan" (slika 7).



Slika 7 Svetlosni mikroskop "Leitz Ortoplan"

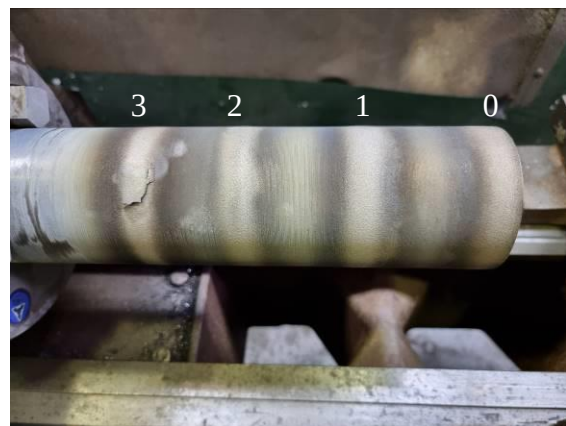
Naneseni slojevi simuliraju prevlake koje će se kasnije koristiti u zaštiti površinskih slojeva, s toga je potrebno izvršiti merenje njihove mikrotvrdoće kako bi se uporedio značaj dodavanja nanočestica u dodatni materijal. Mikrotvrdoća predstavlja precizno određivanje promena u vrednostima tvrdoće na određenim mestima. Prilikom merenja mikrotvrdoće korišten je uređaj „Wilson Tukon 1102“ sa opterećenjem 200 g (HV0.2) (slika 8).



Slika 8 Uređaj za merenje mikrotvrdoće "Wilson Tukon 1102"

3. REZULTATI I DISKUSIJA

Izgled uzorka tankozidne cevi nakon nanošenja četiri metalizirana sloja, dat je na slici 9. Vidi se da je došlo do intenzivnog odvajanja kod uzorka 3, verovatno usled loše athezije.

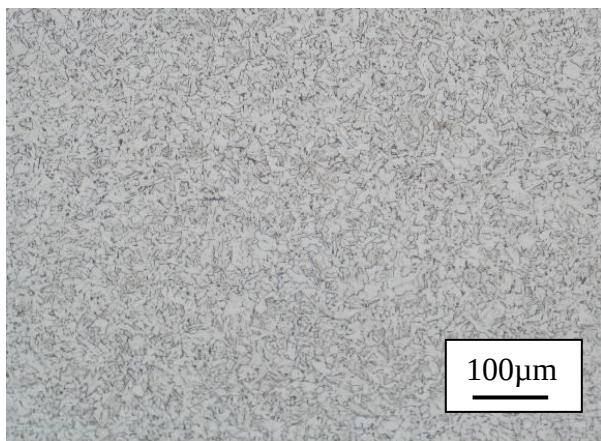


Slika 9 Cev nakon metalizacije

Osnovni materijal (tankozidna cev) ima feritno-perlitnu mikrostrukturu (slika 10), što je u skladu sa relativno niskim sadržajem ugljenika u čeliku. Na slici 11 su dati uzdužni preseki cevi sa vidljivim nanetim slojevima.

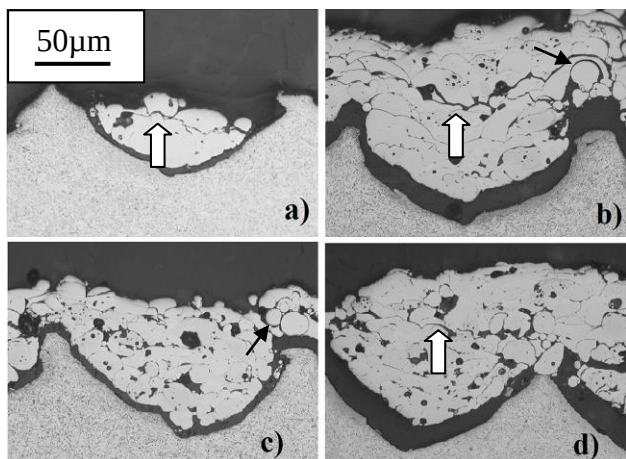
Uzorak 0 je na izgled najtanji i njime nisu popunjeni ni risevi koji vode poreklo od pripreme struganjem. Međutim, sa slike 6 je evidentno da jeste došlo do popune, što ukazuje na konstataciju da je do uklanjanja gornjeg dela prevlake, najverovatnije usled odvajanja pojedinih slojeva tokom metalografske pripreme, koja je uticala na razdvajanje slojeva od osnove.

To se vidi kao tamna granica između prevlake i osnove. Ona je najveća kod uzorka 1 i 3, dok je kod uzoraka 0 i 2 približno jednake širine.



Slika 10 Feritno-perlitna mikrostruktura osnovnog materijala

Najveću debljinu ima sloj 1, potom 3 i konačno, 2. Mikrostruktura sadrži mnoštvo deformisanih zrna, čak i nepretopljenih (označeno strelicom). Pored toga, prisutne su i tamne pore, kao i raslojavanje (bele strelice), koje je najmanje intenzivno kod uzorka 2.



Slika 11 Mikrostruktura nanesenih slojeva
a) uzorak 0, b) uzorak 1, c) uzorak 2, d) uzorak 3

Na osnovu dobijenih rezultata merenja mikrotvrdoće (tabela 1) može se videti da je mikrotvrdoća uzorka 1 nešto manja od kontrolnog uzorka 0. Međutim, na osnovu standardne devijacije, može da se konstatuje da je to unutar statističke ponovljivosti, te se mikrotvrdoća ova dva uzorka može smatrati istom.

Uzorak 2 ima značajno veću tvrdoću u odnosu na ostale uzorke, dok je kod uzorka 3 mikrotvrdoća najmanja. Povećana tvrdoća uzorka 2 može se objasniti stvaranjem prepreka prilikom kretanja dislokacija u leguri nikla (u vidu disperzoida) u osnovi prevlake, koje su nastale unošenjem TiO_2 .

Smanjena tvrdoća uzorka 3 može se povezati sa aglomerisanjem nanočestica (spajanje nanočestica relativno slabim Van der Waalsovima silama). Aglomeracija nanočestica bi takođe moglo da dovede do smanjenja otpornosti na habanje.

Teorija o aglomeraciji je nova u ovoj oblasti, te se mora dokazati skeniranjem ili transmisionim elektronskim mikroskopom.

Tabela 1 Vrednosti mikrotvrdoće $\text{HV}_{0,2}$

	Uzorak 0	Uzorak 1	Uzorak 2	Uzorak 3
Merenje 1	1013,5	990,5	1073,2	748,8
Merenje 2	954,7	935,7	1048,8	687,6
Merenje 3	911,2	917,5	1135,2	839,4
Srednja vrednost	959,8±51,3	947,9±38	1085,7±44,5	758,6±76,4

4. ZAKLJUČAK

Kako je proizvođač dodatnog materijala naveo da se njegova tvrdoća kreće (55-63 HRC, što je približno 596-774 HV_{30}) može se zaključiti da prah u kombinaciji sa nanočesticama pruža mogućnost povećanja tvrdoće što je rezultatima i prikazano. Uzorak 0 i 3 su pokazali najbolje karakteristike, zbog raslojavanja i odvajanja prevlake tokom metalografske pripreme ili neposredno nakon metalizacije. To ukazuje da je za upotrebu ovog dodatnog materijala poželjno koristiti međusloj (bafer sloj) na bazi nikla koji bi povećao adheziju između dodatnog i osnovnog materijala. Povećanje mikrotvrdoće kod uzorka 2 se može objasniti ojačavanjem veziva na bazi legure nikla nanočesticama. Kako je ova oblast nova, potrebno je u perspektivi uložiti napor da se objasne fenomeni kod kojih se došlo u ovom radu.

6. ZAHVALNOST

Rezultati prezentovani u ovom radu su realizovani u okviru projekta „Inovativni materijali i tehnologije spajanja“, Departmana za proizvodno mašinstvo, FTN, Novi Sad.

7. Literatura:

- [1] M. Božović, O. Stojanović O.: „Smanjenje habanja primenom metalizacije“, Stručni rad, Međunarodna konferencija „Stanje i perspektiva zavarivanja na početku novog veka“, (2001), str. 77-82.
- [2] Castolin Eutectic: „Operating manual“, Messer Eutectic Castolin, Gliwice-Poland, str. 1-23.
- [3] <https://www.plasmacentre.ru/produkcziya/poroshki-provoloka-elektrody/castolin-eutectic/eutalloy-rw/eutalloy-rw-pe-8902/> (Pristupio 18.10.2021.)
- [4] D. Ivaniš, „Određivanje tvrdoće materijala Vickersovom metodom uz različita opterećenja“, Završni rad, Sveučilište u Zagrebu fakultet strojarstva i brodogradnje, 2009.

Kratka biografija:



Miljan Dragičević rođen u Bijeljini 1996. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Proizvodno mašinstvo – Materijali i tehnologije spajanja odbranio je 2019.god.



Sebastian Baloš rođen u Somboru 1974. god. Doktorirao je na Fakultetu tehničkih nauka 2010. god., a 2021. godine je biran u zvanje redovnog profesora iz oblasti Mašinsko inženjerstvo, tj. uže naučne oblasti Materijali i tehnologije spajanja

UTICAJ HABANJA ALATA NA SPEKTAR SIGNALA VIBRACIJA I POPREČNI PRESEK STRUGOTINE

INFLUENCE OF TOOL WEAR ON VIBRATION SIGNAL SPECTRUM AND CHIP CROSS SECTION

Marina Zahorec, *Fakultet tehničkih nauka, Novi Sad*

Mašinstvo – PROIZVODNO MAŠINSTVO

Kratak sadržaj – U radu je prikazano eksperimentalno ispitivanje uticaja habanja alata na poprečni presek strugotine i spektar signala vibracija. Mikroskopskom analizom poprečnog preseka strugotine jasno je utvrđen oblik nastale strugotine u korelaciji sa stepenom pohabanosti alata. Tokom procesa obrade praćene su vibracije na nosaču alata, u bliskom području zone rezanja. Analiza generisanih signala vibracija tokom obrade potvrdila je da promene u stepenu habanja alata direktno utiču na oblik strugotine i tip segmentacije.

Ključne reči: *Habanje alata, oblik strugotine, signal vibracija*

Abstract – The paper presents an experimental research of the influence of tool wear on the chip cross section and the spectrum of vibration signals. Microscopic analysis of the chip cross section clearly determined the shape of the resulting chip in correlation with the tool wear. During the machining process, vibrations were monitored on the tool holder, close to the cutting zone. Analysis of the generated vibration signals during machining confirmed that changes in the tool wear directly influence on the chip shape and the type of segmentation.

Keywords: *Tool wear, chip shape, vibration signal*

1. UVOD

Postupak obrade metala rezanjem je proces oblikovanja (promene oblika, dimenzija, hrapavosti obrađene površine i karakteristika površinskog sloja) postepenim uklanjanjem viška materijala mehaničkim dejstvom reznog alata na predmet obrade.

Ovaj postupak se izvodi radi postizanja potrebnih ili predviđenih dimenzija i tačnosti oblika i završne obrade površine, kako bi se proizvodu omogućilo ispunjenje osnovnih funkcionalnih zahteva, poboljšanje performansi, dugi radni vek.

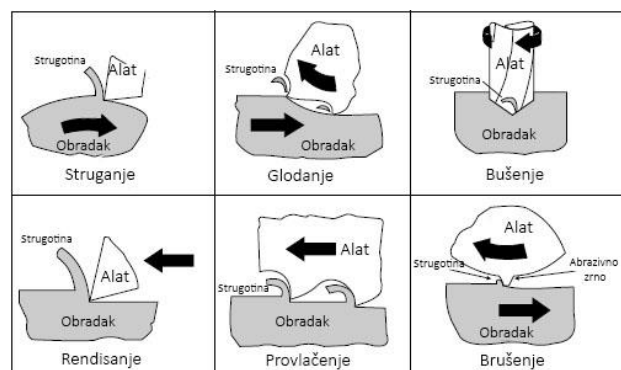
Alat za rezanje odvaja višak materijala od obratka u obliku strugotine, kako bi materijal obrade dobio željeni oblik i veličinu (slika 1). Dostupan je veliki izbor alata za rezanje, od kojih je svaki dizajniran za efikasno obavljanje određenog posla ili grupe operacija rezanja obratka.

Procenjuje se da precizan i pouzdan sistem za identifikaciju i praćenje stanja pohabanosti alata može doprineti: povećanju brzine rezanja za 10-50%, smanjenju zastoja u proizvodnji pravovremenim predviđanjem zamene alata doprinoseći ukupnim uštedama u proizvodnji između 10% i 40% [1].

Oblik (geometrija), materijal i položaj alata u odnosu na obradak imaju značajan uticaj na sposobnost obrade i kvalitet proizvoda. Najvažniji geometrijski elementi, u odnosu na formiranje strugotine, su položaj rezne ivice i orijentacija površine alata u odnosu na obradak i smer rezanja.

Pri obradi materijala rezanjem, međusobnim dejstvom alata i obratka javljaju se vibracije. Jedan od primarnih izvora vibracija u toku obrade je proces formiranja strugotine, tj. lamela strugotine. Drugi tipični izvori vibracija na mašini su rezultat periodičnih ili povremenih sila u sistemu kao što su sila od pogona zupčanika, neravnoteže, neusklađenosti ili labavi spojevi komponenti alatnih mašina itd. [2, 3].

Habanje alata direktno utiče na postojanost, kvalitet i ekonomičnost procesa obrade, a kako su razvoj računarske opreme i automatizacija procesa doveli do postizanja veće fleksibilnosti i produktivnosti u toku procesa obrade rezanjem, kvalitetna informacija o stepenu pohabanosti alata u realnom vremenu predstavlja neophodan uslov za identifikaciju postojanosti alata [4]. Veoma je bitno zameniti alat na vreme, čime se u značajnoj meri podiže kvalitet obradnog procesa.



Slika 1. Skica delovanja alata na obradak, različitim postupcima obrade rezanjem [2]

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Aco Antić, red. prof.

2. EKSPERIMENTALNO ISPITIVANJE

2.1. Postavka eksperimenta

Eksperimentalno ispitivanje izvršeno je na strugu CNC GU 600, proizvođača INDEX, koji je instaliran u laboratorijama Fakulteta tehničkih nauka u Novom Sadu. Tokom procesa rezanja registrovan je signal vibracija na dršci alata. U okviru ovog istraživanja, početna pretpostavka je bila da neki deo frekvencionog spektra sadrži informacije koje se mogu izdvojiti kako bi se omogućilo otkrivanje trenutnog stanja istrošenosti alata za rezanje. U tu svrhu, na držač alata za rezanje pričvršćen je akcelero-metar, koji je korišćen kao sistem senzora i prikupljao je informacije o vibracijama, a nakon filtriranja signala ih prenosio u sistem za akviziciju. Signal vibracija i sila rezanja su uzorkovani frekvencijom od 625 kHz, pomoću A/D konvertora NI625 USB "National Instruments". Tokom eksperimenta, merenja ubrzanja vibracija su praćena pri svakom prolazu sečenja, u trajanju od 1 s. Snimljeni signali su analizirani da bi se procenio uticaj habanja alata na signal vibracija, i promena spektra signala.

Postojale su dve brzine rezanja tokom izvođenja eksperimenata, 180 do 250 m/min i brzina pomoćnog kretanja 0,15 i 0,3 mm/o. Poprečni presek drške alata korišćenog u eksperimentu bio je 20x20 mm. Obrada je izvedena sa reznim pločicama P25 oznake TNMM 110408. Materijal obratka je garantovanih mehaničkih i hemijskih karakteristika oznake 42CrMo4, tvrdoće od 270 HB.

Pored praćenja signala vibracija, pomoću mikroskopa posmatrani su oblici dobijenih strugotina u toku procesa obrade, u cilju upoređivanja tipa formirane strugotine u odnosu na stepen pohabanosti alata.

2.2. Analiza eksperimentalnih rezultata

Rezultati ispitivanja dobijeni su upoređivanjem oblika strugotine na elektronskom mikroskopu i primenom indirektnih merenja senzorskih signala sile i vibracija.

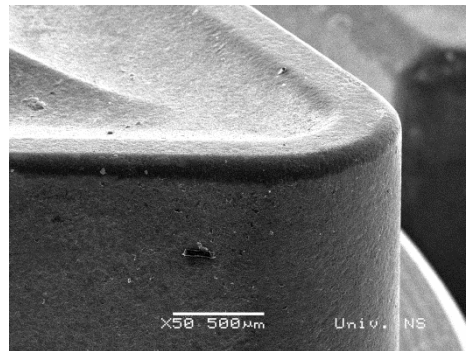
Rezultati dobijeni direktnom mikroskopskom analizom dali su dobar uvid u oblik lamela i formiranu strugotinu u zavisnosti od stepena pohabanosti alata.

U toku obrade je korišćen novi alat, slika 2, pri čemu je formirana kontinualna strugotina, slika 3. Kontinualni oblik strugotine nastaje smicanjem materijala u primarnoj zoni rezanja bez jasno uočljivih granica segmenta u poprečnom preseku i bez izraženih vrhova formiranih segmenata na slobodnoj površini strugotine. Visina segmenata na slobodnoj površini strugotine je veoma mala i odgovara širini pojedinačnog segmenta. Gornja zona strugotine, bliže slobodnoj površini, blago je talasastog oblika sa veoma malim naznakama početka formiranja lamela.

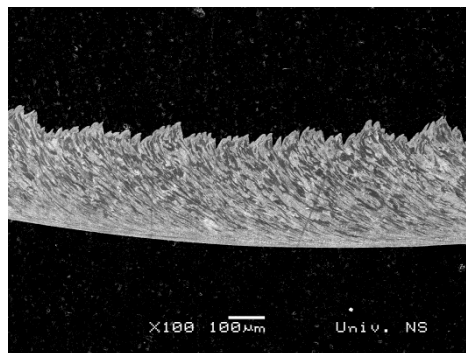
Kako pojas habanja i kratera habanja na grudnoj površini alata postepeno rastu, tj. povećava se stepen degradacije rezne geometrije (slika 4), dolazi do promene tipa formirane strugotine (slika 5). Formirani segmenti su nastali kroz ciklični proces stvaranja (od prvog do poslednjeg segmenta). Nakon određenog vremena obrade, usled promene stepena pohabanosti alata, tj. rezne geometrije, strugotina takođe počinje svoju transformaciju geometrije, menja mehanizam stvaranja i oblik. Sa

makroskopske tačke gledišta, strugotina postaje ravnija i sa izraženijim klizanjem materijala duž osnovne ravni. Ledna površina strugotine je postala valovita, neravna i neujednačena u odnosu na onu dobijenu obradom novim alatom.

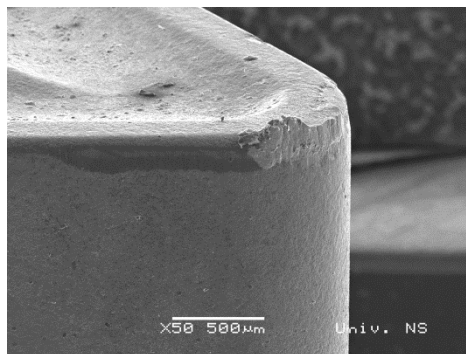
Kako se stepen pohabanosti alata dalje raste (slika 6), stepen deformacije strugotine nastale u procesu rezanja se povećava. Strugotina se sastoji od segmenata, čije se granice mogu jasno uočiti, tj. formira se strugotina testerastog oblika (slika 7).



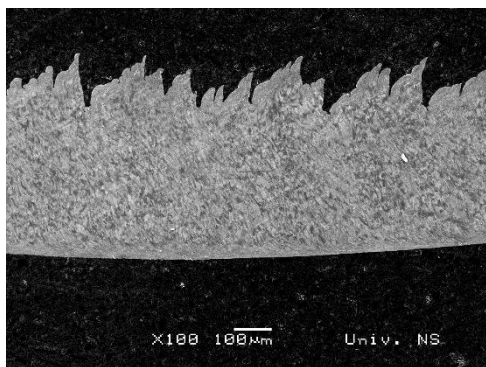
Slika 2. Stanje rezne pločice alata na početku obrade



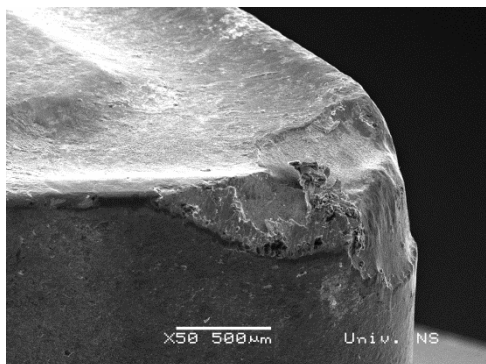
Slika 3. Izgled kontinualne strugotine pod mikroskopom nastale obradom novim alatom



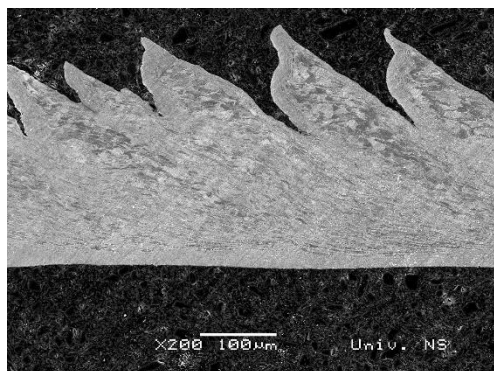
Slika 4. Stanje pohabanosti alata prilikom formiranja izraženijih segmenata na strugotini



Slika 5. Izgled strugotine pod mikroskopom, nakon porasta stepena pohabanosti alata



Slika 6. Izgled degradirane, pohabane, rezne pločice

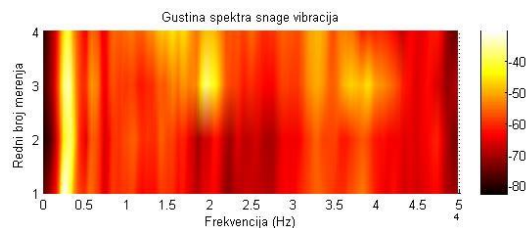


Slika 7. Izgled strugotine testerastog oblika nastala obradom pohabanim alatom

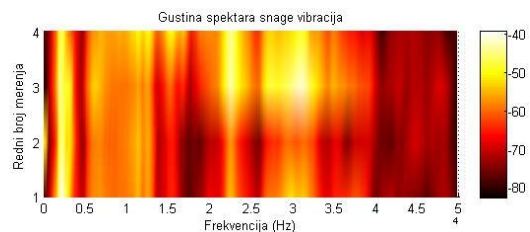
Posmatrajući spektre snage signala vibracija, za više merenja sa različitim stepenom pohabanosti alata, može se uočiti jasna promena spektra, kako je prikazano na slikama 8, 9 i 10. Sve prirodne frekvencije alata i mehaničkih struktura nosača alata nalaze se unutar donjeg spektra vibracija, čija je gornja granica 5 do 10 kHz. Zbog toga su pobude iz zone rezanja u ovom delu spektra izobličene, pa akcelorometar dobija signal jačeg intenziteta, što se odražava na pojačanje na nekim frekvencijama, a potiskuje na drugima. Dakle, iako je u nekim delovima opsega frekvencija signala koji merimo, na frekvencijama ispod 5 kHz prisutan visok intenzitet, on se pripisuje vibracijama obradnog sistema, tj. Same mašine i celokupnog sklopa mašina alat kada nije u zahvatu obrade.

Pretpostavlja se da se pri razmatranim brzinama rezanja segmentacija lamela vrši na većim frekvencijama (preko 15 kHz) i predstavlja jedan od dominantnih izvora vibracija tokom mašinske obrade, i može se primetiti da

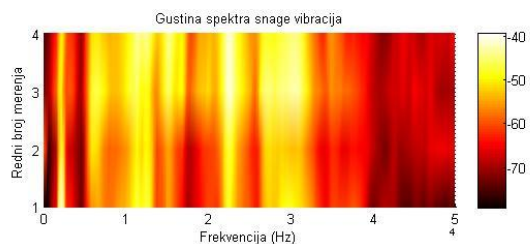
se na višim frekvencijama (preko 15 kHz) stanje rezne pločice više odražava na spektralni sastav signala nego na nižim frekvencijama. Posmatranjem spektra može se primetiti velika razlika u delu od 25 do 35 kHz, što ukazuje na važne promene u procesu formiranja strugotine i promenu geometrije rezne ivice alata. Promena stepena habanja dovodi do povećanog trenja klizanja između površine alata i strugotine koja se evakuše tokom sečenja, što je izazvalo dodatne vibracije sistema delova alatnih mašina.



Slika 8. Promena spektra snage vibracija za različite stepene pohabanosti alata



Slika 9. Promena spektra snage vibracija za različite stepene pohabanosti alata

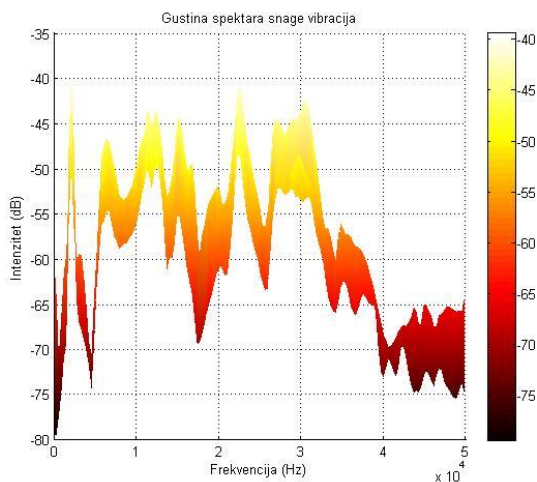


Slika 10. Promena spektra snage vibracija za različite stepene pohabanosti alata

Analizom rezultata eksperimentalnog ispitivanja primećeno je da frekventni odziv obradnog sistema menja karakter u zavisnosti od tipa generisane strugotine.

Slika 11 predstavlja raspodelu energije signala po frekventnoj osi u zavisnosti od stepena pohabanosti alata. Donja deo trake odgovara novoj reznj pločici alata, dok se gornji deo odnosi na istrošeni alat. Na frekvencijama nižim od 5 kHz signal je relativno konstantan, što potvrđuje da je na nižim frekvencijama uticaj obradnog sistema dominantan.

Na frekvencijama većim od 5 kHz stepen pohabanosti pločice se više odražava na spektralni sastav signala, odnosno, u visokofrekventnom delu spektra dolazi do povećanja intenziteta generisanog signala. Na delu frekvencija od 27 kHz do 33 kHz, poređenjem nove i stare rezne pločice, može se primetiti razlika u intenzitetu signala i do 10 dB. Razlika u intenzitetu signala povezana je sa oslobađanjem povišene energije pri formiranju diskontinualnog oblika strugotine i povećanim trenjem na kontaknoj površini između strugotine i alata.



Slika 11. Promena intenziteta spektra snage vibracija za različite stepene pohabanosti alata

3. ZAKLJUČAK

Na osnovu dobijenih rezultata mikroskopske analize formirane strugotine, u početnoj fazi obrade, korišćenjem novog alata, formirana strugotina je ravnog oblika sa glatkom leđnom površinom, koja je bila u punom kontaktu sa grudnom površinom alata, a slobodna površina strugotine je bez izraženih zuba segmenata. Porastom pojasa habanja i kraterskog habanja alata strugotina menja oblik, postaje grublja, valovita i krzana na krajevima, dok se oblik strugotine menja u diskontinualnu sa veoma izraženim zubima na slobodnoj površini strugotine. Daljim trošenjem alata, povećavanjem bočnog habanja, segmentacija strugotine postaje izraženija, testerastog oblika, te je uočena dobra korelacija učestalosti segmentacije strugotine sa stepenom pohabanosti reznog alata. Promenom stepena pohabanosti menja se i oblik segmentacije strugotine, od kontinualne do strugotine sa jako izraženim vrhovima lamela na slobodnoj površini strugotine, tj. diskontinualne strugotine testerastog oblika.

Pored boljeg razumevanja mehanizama formiranja strugotine, morfologija strugotine može biti dobar indikator stanja pohabanosti alata u teškim uslovima obrade. Takođe, znanje morfologije i menjanje oblika strugotine može da poboljša produktivnost proizvodnje, gde habanje alata i dimenzionalna tačnost obratka treba da se prate.

Rezultati spektra snage vibracija u procesu rezanja su u korelaciji sa oblikom segmentacije strugotine i sa razvojem kraterskog i bočnog habanja na alatu. Promena oblika strugotine se odražava na signal vibracija, kvalitet obrađene površine i, posledično, na ukupni utrošak energije pri obradi.

4. LITERATURA

- [1] V. C. Venkatesh, Sudin Izman "Precision Engineering", McGraw-Hill Professional, 2008.
- [2] Serope Kalpakjian, Steven Schmid, "Manufacturing Engineering and Technology", 6th edition, Pearson, 2008.
- [3] H. K. Toenshoff and B. Denkena, "Basics of Cutting and Abrasive Processes" Lecture Notes in Production Engineering, Berlin-Heidelberg, 2013.
- [3] Aco Antić, "Prepoznavanje stanja pohabanosti alata za obradu rezanjem primenom neuro-fazi klasifikatora", doktorska disertacija, Fakultet tehničkih nauka, Novi Sad, 2010.

Kratka biografija:



Marina Zahorec rođena je u Novom Sadu 1995. god. Diplomski rad pod naslovom "Nadzor habanja alata praćenjem otpora rezanja pri obradi struganjem" odbranila je 2019. godine. Nakon toga upisala je Master akademske studije – Računarom podržane tehnologije na Departmanu za proizvodno mašinstvo, Fakulteta tehničkih nauka.

kontakt: zahorecmarina@gmail.com

ПРИМЕНА LEAN МЕТОДОЛОГИЈЕ И BIM-А ПРИ ИЗРАДИ ПРОЈЕКТАТА APPLICATION OF LEAN AND BIM METHODOLOGY IN PROJECT DESIGN

Мирко Мијатовић, *Факултет техничких наука, Нови Сад*

Област - МАШИНСТВО

Кратак садржај – BIM методологија представља иновативно решење при самом приступу изради пројектата. Конципирана је тако да се све одвија у једном централном моделу, коме може свако у сваком тренутку да приступи и осети у реалности како ће неко будуће решење изгледати. Али да би се BIM методологија у потпуности примењивала у процесу израде пројектата, потребно је добро искоординисати и организовати процесе унутар ње. За тај део је задужена LEAN филозофија и њени корисни алати, који су у овом раду примењени и доказано унели побољшање.

Кључне речи: BIM методологија, LEAN филозофија, пројектовање

Abstract – BIM methodology is an innovative solution in the approach to project design. It is designed so that everything takes place in one central model, which anyone can access at any time and feel in reality what a future solution will look like. But in order for the BIM methodology to be fully applied in the project development process, it is necessary to coordinate well and organize the processes within it. The Lean philosophy and its useful tools are in charge of this part, which have been applied and proven to be improved in this paper.

Keywords: BIM methodology, Lean philosophy, design

1. УВОД

Циљ овог рада јесте објаснити концепт пројектовања путем BIM методологије. Међутим рад тачно објашњава како је то потребан, али не и довољан услов да би се пројекат квалитетно завршио. Зато је представљена тренутна проблематика, као и пример предлога побољшања коришћењем Lean филозофије.

Приказан је тренутан пример израде пројектата у једној иностраној компанији, која је желела да остане анонимна. Кроз тај пример уочене су грешке и простор за унапређење. Тек након разумевања читавог процеса настанка пројекта, који се проблеми јављају током животног циклуса једног пројекта, могуће је дати предлог побољшања.

Предлог побољшања представљен је кроз адекватне Lean алате, који су за дати пример показали најбоље унапређење, а то су скраћење укупног времена израде пројекта, као и осигуравање већег квалитета израде.

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је била др Сања Бојић, ванр. проф.

2. LEAN

Тешко је поставити универзалну дефиницију појма Lean. Можемо рећи да заправо разговарамо о неколико појмова одједном. Lean размишљање, Lean приступ, Lean организација, Lean менаџмент итд. Међутим Lean филозофију пословања можемо дефинисати као системски приступ укупном пословању на начин коришћења што мање ресурса, са циљем постизања бољих резултата.

У Lean-у се по први пут појављује појам „отпад“ (енг. Waste), што се у Тоуоти подразумева као:

- Прекомерна производња;
- Чекања;
- Транспорт;
- Прекомерна обрада или корекција;
- Залихе;
- Непотребно кретање;
- Грешке (дефекти);
- Неискоришћена креативност запослених.

„Six sigma“ је скуп алата и метода за побољшавање процеса које се данас користе у многим областима биле оне производне или услужне. Грешке у производним процесима могу имати озбиљан утицај на приходе предузећа, у неким случајевима та бројка иде и до 30% укупних прихода. На тај начин функционису многа предузећа и организације, иако нису свесни да се ти недостаци могу отклонити. „Six sigma“ стратегија фокусира се на континуално побољшање ефикасности и ефективности у свим процесима, задацима и трансакцијама унутар било које гране привреде.

DMAIC (Define-Measure-Analyze-Improve-Control) метода је скраћеница од енглеских речи, а њихов превод гласи: дефинисати, анализирати, побољшати, контролисати. Овом методом побољшавају се постојећи процеси који нису унутар задате спецификације, те их је потребно побољшати.

DMADV (Define-Measure-Analyze-Design-Verify) је скраћеница првих слова енглеских ријечи, а њихов превод гласи: дефинисати, мерити, анализирати, обликовати и проверити. Ова метода се користи када одређени процес не постоји па га је потребно развити. Нпр. ако организација жели понудити потпуно нов производ или услугу.

Уколико је постојећи процес или производ побољшан коришћењем **DMAIC** методе али још увек није задовољно захтеве купаца или постигао очекивани ниво сигме примјењује се **DMADV**.

2.1 LEAN АЛАТИ

У овом раду биће објашњена примена Lean алата у једном средњем предузећу, које не мора бити производно да би имало бенефите од коришћења Lean алата. Такође биће јасно назначено да није довољно само то што нека нова технологија постоји, и да неће она сама по себи донети значајне промене набоље, већ је кључ у њеном правилном разумевању и коришћењу.

Како би целокупан концепт Leana успео, а не остао само на доброј идеји, постоје одговарајући кораци (правила) која се морају испоштовати. Значај тог процеса објашњавали су и идејни творци Lean филозофије Таичи Оно и Шигеоо Шинго. Ти кораци се у Lean-у зову алатима, и поштовање једног представља предуслов за имплементацију следећег алата (корака).

Неколико кључних Lean алата су: Мапирање тока вредности (Value stream mapping - VSM), Пет С (Five S – 5S), Стандардизовани рад (Standardized work – SW), Решавање проблема (Problem solving – PS), Смањење припремног времена (Setup reduction – SUR), Визуелни менаџмент (Material flow – MF), Хеиџунка (Heijunka)...

2.2 Swimlane

Дијаграм Swimlane је дијаграм тока процеса који вам омогућава да визуелно разликујете дужности и одговорности, као и потпроцесе у оквиру ових пословних процеса.

Swimlane дијаграми први пут су се појавили четрдесетих година прошлог века као варијација дијаграма тока који се назива графикони са више колона. Geari Rummler и Alan Brache су их у својој књизи „Improving Performance” (1990) назвали Swimlane дијаграмима. Са рачунарским дијаграмима први пут их је упознао „иГрафк“. Swimlane дијаграми су познати и као „Руммлер-Браче-ови дијаграми” [6].

Као и сваки други дијаграм тока, он визуализује процес од почетка до краја, користећи разне симболе за постављање корака мапирања било вертикално или хоризонтално. Swimlane дијаграм обично се користи за пројекте који се преплићу преко различитих одељења и разликују канале комуникације према одређеном скупу циљева. Организујући одговорности у различитим правцима, може јасно разликовати циљ сваког одељења и појединаца у тиму.

Дијаграми „пливачких стаза“ (буквалан превод) обично се користе у организацијама са више одељења за илустрацију кооперативних пословних модела између одељења. Приказивањем одељења у вертикалној траци и циљева у хоризонталном смеру, или обрнуто.

У суштини, свако одељење или тим представљају своју стазу (симбол). Може се користити онолико симбола колико је потребно да би се илустровао свој циљ. Могу се чак и креирати петље на дијаграмима, или вишеструке стазе „ако ... онда“ или „или“ у случају различитих исхода предложеног циља. Овај део је изузетно важан јер ће нам он открити одређене проблеме и пре њиховог настанка, као и како да реагујемо у таквим ситуацијама.

2.3. SIPOC model

SIPOC модел (Supplier – Input – Process – Output – Customer), или у преводу ИУПИК (Испоручилац – Улаз – Процес – Излаз – Купац). У наставку ћемо користити страну скраћеницу SIPOC јер је далеко распрострањенија и познатија, чак и на нашим просторима. Дијаграм SIPOC је алатка шест сигма која се користи за документовање пословних процеса. Реч SIPOC означава добављаче, улазе, процесе, излазе и купце који чине колоне дијаграма. Дијаграм SIPOC визуелно документује пословни процес од почетка до краја. Будући да дијаграм не садржи много детаља о процесу, често се назива мапом процеса на високом нивоу.

Приликом предузимања било које активности побољшања процеса важно је да прво разумемо процес, и како се уклапа у бизнис (ваш процес је мали део већег система). SIPOC је развијен од стране Петер Сцхолтерс-а као разрада дијаграма система др Деминг-а и одличан је алат који ће вам помоћи да предузме тај корак назад и заиста почнете да разумете процес који желите да побољшате (Петер Р. Сцхолтерс, 1997) [7].

Дијаграм је корисан на више начина. За почетак, људима који нису упознати са пословним процесом даје преглед на високом нивоу. Будући да дијаграм садржи само најосновније информације, он такође добро функционише ако требате брзо попунити нову особу или ажурирати некога ко је у прошлости радио на том процесу.

2.4. Project Constraints and Project Assumptions – Чек Листа Планирања

Ефикасно планирање успеха пројекта значајно зависи од тачне поставке о ограничењима (constraints) и претпоставкама (assumptions). Њихова важност огледа се у томе што постављање одговарајућих пројектних ограничења и претпоставки води ка адекватној анализи ризика, ефикасном планирању пројекта и благовременој испоруци пројекта.

Дакле циљ овог алата је да пружи дефиниције пројектних ограничења и претпоставки пројекта, као и да покаже које кораке је могуће предузети да би се утврдиле кључне карактеристике за успех пројекта.

Пре започињања новог пројекта, руководиоци пројекта и други старији актери који воде главну реч на пројекту, морају се сложити око претпоставки и створити заједничко разумевање успеха пројекта. Ово ће смањити стопу неуспеха пројекта и обезбедити основу за развој стратегија за смањење ризика и планова за непредвиђене случајеве.

Претпоставке пројекта су они догађаји и околности за које се очекује да ће се догодити током животног циклуса пројекта за успешну примену и завршетак. Иако су претпоставке покретачка снага која одређује успех пројекта, обично су изван потпуне контроле пројектног тима. Ово је веома важан корак у планирању неког пројекта, како исти не би био осуђен на пропаст и пре самог почетка. Претпоставке пројекта прихватају се као истините, често без икаквих доказа или демонстрација.

Било којим пројектом се управља под скупом јасних ограничења и потребно их је дефинисати од самог почетка пројекта. Веома је важно да руководиоци пројекта (пројект лидери) јасно разумеју ограничења пројектног окружења (као што су рокови, финансирање, нивои вештина, доступност ресурса итд.) у оквиру којих ће се активности изводити, самим тим они ће учинити бољи посао у обезбеђивању успеха пројекта и развоју доброг стања - основани план примене.

У овом контексту, ограничења пројекта су било који догађаји или околности који могу ограничити или регулисати пројекат. Баш као и претпоставке, типична пројектна ограничења су изван потпуне контроле пројектног тима али се на основу пређашњих искустава могу донекле претпоставити.

2.5. Стандардизовани рад (*Standardized work – sw*)

Стандардизовани рад један је од најснажнијих, али најмање коришћених Lean алата. Документовањем тренутне најбоље праксе, стандардизовани рад чини основу за Каизен или Lean (континуално усавршавање) [1-4]. Како се стандард побољшава, нови стандард постаје основа за даља побољшања итд. Побољшање стандардизованог рада је непрекидан процес.

Овај алат подразумева креирање процедура и радних инструкција које могу садржати стандардизоване кораке за свако радно место, на основу којих оператери знају начин и време трајања операција на сваком радном месту у производном процесу. Овакав начин рада у многима ће олакшати људима када прелазе са једног на друго радно место, да ли услед реорганизације или попуњавања рупа услед недовољног броја људи по операцијама. Овај вид радних инструкција мора поред текстуалног описа како се нека операција извршава садржати и слике са јасно назначеним критичним тачкама које могу утицати на крајњи квалитет производа. Поред тога, оваква радна упутства и даље морају бити довољно једноставна, како радник не би губио превише времена на њихово разумевање. Циљ овако креираних инструкција је фокус на унапређења, а не на строга и рестриктивна правила.

3. BIM (Building Information Modelling)

„Building Information Modelling“ (BIM) у буквалном преводу би значило „Прављење информационог моделирања“ али се у пословном свету употребљава само скраћеница BIM. Концепт BIM-а постоји од 1970-их. Први софтверски алати развијени за моделирање зграда појавили су се крајем 1970-их и почетком 1980-их, а укључивали су производе радних станица као што је „Chuck Eastman's Building Description System“ и GLIDE, RUCAPS, Sonata, Reflek и Gable 4D Series. Ране апликације и хардвер потребан за њихово покретање били су скупи, што је ограничило широко усвајање [8].

Због сложености прикупљања свих релевантних информација током рада са BIM-ом, неке компаније су развиле софтвер дизајниран посебно за рад у BIM оквиру. Ове се апликације разликују од архитектон-

ских алата за израду нацрта, као што је Auto CAD, омогућавањем додавања додатних информација (време, цена, детаљи произвођача, информације о одрживости и одржавању итд.) моделу зграде.

Укратко и поједностављено, BIM се односи на методологију која се ослања на интелигентан, подацима богат модел као темељ за пројектовање, симулацију и сарадњу кроз све фазе пројекта.

BIM (Building Information Modelling) методологија намерава интеграцију свих фаза грађевинског процеса, тј. Интеграцију и промоцију заједничког рада свих дизајнерских дисциплина укључених у фазу пројектовања. Поред тога, подржавају га тродимензионалне апликације за визуелизацију. Велики потенцијал BIM концепта је такође у стандардизацији информација, подржаних, између осталог, стандардизацијом метода за извођење процеса моделирања објеката. На основу овога, потенцијална побољшања у припреми, координацији и ревизији пројектне документације, као и управљање и одржавање изграђеног окружења могу бити изузетно вредна.

Можда више од свега, BIM је алат за управљање пројектима, који пројектним тимовима омогућава тачну процену трошкова, смањење материјалног отпада, оптимизацију распореда, симулацију грађевинских активности и поједностављење операција. Такође је механизам за подршку администрацији уговора, додељивању и праћењу задатака, управљању варијацијама и генерално планирати и извештавати о напретку пројекта.

Софтвери који су се развили са намером да би BIM постао доступан свима су:

- Autodesk Revit
- AutoCAD

4. СТУДИЈА СЛУЧАЈА ПРИМЕНЕ LEAN

МЕТОДОЛОГИЈЕ И ВІМ У РЕАЛИЗАЦИЈИ ПРОЈЕКТА У ПРОЈЕКАНТСКОЈ КУЋИ ЗА МАШИНСКЕ И ЕЛЕКТРО ИНСТАЛАЦИЈЕ

Тренутна студија примене Lean и BIM методологије, рађена је на примеру једне иностране компаније која је у овом случају желела да остане анонимна због разлога очувања пословне тајне.

4.1. Анализа тренутног процеса пројектовања

Анализа тренутног процеса пројектовања рађена је коришћењем посебних Lean алата који су посебно прилагођени за посматрани случај. Кренуло се од постављања тока процеса коришћењем *Swimlane* дијаграма, као и сваког процеса и учесника у њему понаособ.

У посматраном случају *Swimlane* дијаграм састоји се из две битна сегмента којима се жели представити животни циклус једног пројекта, а то су: „Израда пројекта“ и сам процес „Пројектовање“. Како би се дијаграм боље разумео потребно је прво разумети учеснике у процесу, као и њихове улоге, задатке и сам процес реализације истих.

Под учесницима у пројекту подразумевају се:

- Купац/Клијент - трећа страна за коју се израђује пројекат;
- Пројект лидер - Први инжењер на пројекту, особа која води пројекат као и сву комуникацију са купцем;
- Инжењери - тим људи који учествује у изради пројекта, од предпланирања до саме реализације уз координацију Пројект лидера;
- Моделери - тим људи који идеју инжењера преноси са папира у 3Д модел уз помоћ одговарајућег софтвера.

Како је фаза пројектовања временски најзахтевнија фаза (кратки рокови, потреба за координацијом, промена планова...) њој ће бити посвећена посебна пажња, као и могућа унапређења која би могла довести до побољшања радних и пословних услова

4.2. Предлог унапређења

Једна од најважнијих ставки сваког пројекта је уговор између клијента и пројектантског бироа, односно све оно што у њему пише. Сваки уговор састоји се од обавеза уговорених страна које оне имају једна према другој, као и према некој трећој страни. Ту је стриктно дефинисано време потписивања и реализације посла, као и опис по ставкама. Зато уговорене стране морају бити јако опрезне при састављању уговора, зато што све оно што тамо пише су у обавези према закону да испоштују. Из ових разлога, често се дешава да уговорене стране заврше на суду, зато што ни саме нису разумеле уговорено, или су схватиле да им нешто недостаје да би дошло до реализације посла, а то није написано у уговору.

Примена софтвера (у нашем случају DevOps -а) који ће омогућити бољу прегледност, израду „пројект тимелине“-а, као и упростити комуникацију. Софтвер се примењује код пројектантске куће, као интерни алат за расподелу задатака. Садржаће историју израде задатака и коментаре, који могу послужити као „наук“ за следеће пројекте и онемогућити понављање грешака, тј. обезбедиће константан напредак и усавршавање процеса при изради пројекта.

Важна ставка која помаже у скраћењу укупног времена израде пројекта је стандардизација посла (Standardized work). Стандардизација отклања нејасноће, повећава квалитет и продуктивност.

5. ЗАКЉУЧАК

У овом раду могло се видети колико се један пројекат може аутоматизовати и поједноставити коришћењем бројних модерних технологија и решења попут Revit-a, Auto Cad-a, DevOps-a, као и Lean алате који омогућују брз развој и једноставно управљање процесом.

Дакле, суштина је да ВІМ начин управљања пројектом у многоструку олакшава сам процес припреме и израде пројекта, али само уколико се користи на прави начин. Централни ВІМ модел јесте нешто што естетски лепо изгледа, и привлачи купца при куповини услуге који у сваком тренутку може видети како ће финални производ изгледати, али уколико није правилно искоординисан са осталим учесницима у пројекту, може довести до праве катастрофе, која се

огледа у кашњењу предаје пројекта, неправилној комуникацији, и великих трошкова. Ту се у причу уводи Lean филозофија, која тежи да све слабости и неправилности оваквог система, доведе постепено до савршенства. Циљ је да се укаже на грешке, како би се из истих учило и како се не би понављале, зато што је сваки следећи пројекат независан од осталих, али нам је искуство из претходних пројеката неопходно и корисно.

Из разлога што је грађевинска индустрија специфична, нема серијску производњу, већ изискује сваки пут иновативно решење, те је на основу тога захтевно применити Lean алате и укомбиновати са ВІМ методологијом, али је изводљиво што је и доказано кроз овај рад.

6. ЛИТЕРАТУРА

- [1] <https://samoobrazovanje.rs/kaizen-metoda/>;
- [2] Младен Зворц, Lean менаџмент у непроизводљивој организацији;
- [3] <http://project-management-srbija.com/lean-menadzmnt/>;
- [4] Masaaki Imai: Gemba Kaizen: A Commonsense Low-cost Approach to Management, McGraw-Hill, 1997;
- [5] Krunoslav Horvat, SIX SIGMA strategija. <https://zir.nsk.hr/islandora/object/vuka:201/preview>;
- [6] <http://www.worksimp.com/articles/process-improvement-articles.shtml>;
- [7] <http://www.cimlss.rs/sipoc-model/>;
- [8] Eastman, Charles; Fisher, David; Lafue, Gilles; Lividini, Josif; Stoker, Douglas; Lessios, Christos (septembar 1974). Pregled sistema opisa zgrade. Institut za prostorno planiranje, Univerzitet Carnegie-Mellon;
- [9] "Autodesk (2002). Modeliranje informacija o zgradama. San Rafael, Kalifornija, Autodesk, Inc";
- [10] ВІМ Manager: Mark Baldwin.

Кратка биографија:



Мирко Мијатовић рођен у Новом Пазару 1993. године. Основне и мастер студије завршио на Факултету техничких наука у Новом Саду, на студијском програму Механизација и конструкционо машинство.



Сања Бојић рођена је у Карловцу 1981. године. Докторирала је на Факултету техничких наука у Новом Саду 2013. године, а од 2019. године изабрана је у звање ванредног професора. Области истраживања су логистика, складишни системи и симулације токова материјала.

МИКРОКЛИМАТСКИ УСЛОВИ У ПУТНИЧКОМ ПРОСТОРУ АУТОБУСА**MICROCLIMATIC CONDITIONS IN BUS PASSENGERS COMPARTMENT**Дејан Поповић, Драган Ружић, *Факултет техничких наука, Нови Сад***Област – МАШИНСТВО**

Кратак садржај – Тема рада су услови топлотног комфора у кабини аутобуса уз утицај клима уређаја. На почетку рада направљен је преглед развоја клима уређаја у аутобусима, наведена су претходна истраживања за побољшање перформанси система, као и начини и примјер испитивања. Обрађени су и сви параметри топлотног комфора. Урађено је испитивање услова топлотног комфора у кабини аутобуса у стационарном стању. Након испитивања урађено је очитавање резултата, њихова обрада, анализа и поређење са стандардним вриједностима. На крају рада извршена је оцјена резултата, наведени су могући правци даљег истраживања и могућност побољшања.

Кључне речи: аутобус, путничка клима, возачка клима, испитивање топлотног комфора.

Abstract – The topic of the paper is the conditions of thermal comfort in the bus cabin with the influence of air conditioning. At the beginning of the work, an overview of the development of air conditioners in buses was made, previous research to improve system performance was listed, as well as methods and examples of testing. All parameters of thermal comfort are also processed. An examination of the conditions of thermal comfort in the bus cabin in a stationary state was performed. After the examination, the results were read, processed, analyzed and compared with standard values. At the end of the paper, the results were evaluated, possible directions for further research and the possibility of improvement were stated.

Keywords: Bus, microclimate, drivers AC, passengers AC, thermal comfort experiment.

1. УВОД

Аутобуси почињу да играју кључну улогу у транспорту путника, како у туризму тако и у локалном и градском превозу. Због путовања при високим и ниским вањским температурама, јавља се потреба за системом који ће хладити кабину са путницима љети, односно гријати зими. Циљ овог експерименталног истраживања је био да се истраже услови топлотног комфора у путничком простору, у топлим спољашњим условима. Испитивања су рађена на два идентична аутобуса. Приликом првог испитивања била је укључена само возачка клима, а приликом другог испитивања била је укључена и возачка и путничка.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Драган Ружић, ванр. проф.

Идеја је била да се добијени резултати анализирају, да се виде разлике на почетку и на крају мјерења, као и то да се добијени резултати коментаришу да ли задовољавају или не задовољавају услове топлотног комфора.

2. КОНСТРУКТИВНИ ЦИЉЕВИ И ТЕНДЕНЦИЈЕ

Са повећање потребе за топлотним комфором путника у путничким кабинама, али и самог возача, како би се обезбједили услови за вожњу у што комфорнијим топлотним условима, многи произвођачи возила су се укључили у истраживања за развој клима уређаја и система гријања. Поред горе наведених разлога за унапређење постојећих решења клима уређаја, нужно је споменути и два разлога која су чак и битнија, а то су потрошња горива и емисија штетних гасова. На основу прегледа објављених истраживања и тестирања на ову тему као три најчешће кориштена метода за испитивање микроклиматских услова у кабини аутобуса су:

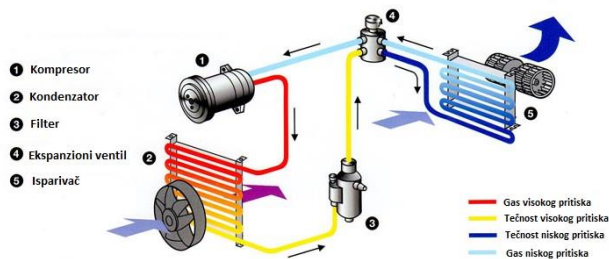
1. испитивање унутар климатске коморе
2. испитивање помоћу софтвера
3. испитивање у току вожње аутобуса

Као што је већ раније речено највећи проблем са клима уређајима возила јесу повећана потрошња горива и већа емисија штетних гасова. Као решење за набројане проблеме јавиле су се климе на електрични погон. Решавају проблем повећане потрошње горива, емисије штетних гасова, али и у могућности су да остваре свој пуни капацитет независно од броја обртаја мотора што је веома битно у условима када возило стоји у мјесту. Даљи напредак и развој ће свакако ићи у овом смијеру.

3. КЛИМАТИЗАЦИЈА АУТОБУСА

Систем за климатизацију аутобуса углавном се састоји од кондензатора, испаривача, компресора, експанзионог вентила, магнетне спојнице, уређаја за свјеж ваздух, уређаја за дување ваздуха, вентилатора, повратне решетке за ваздух, контролне плоче, система цјевовода и електричног система управљања. Клима уређаји аутобуса се могу класификовати у три групе, на основу типа рада компресора [5]:

1. погон компресора помоћним мотором
2. погон компресора главним мотором
3. компресор са електричним погоном



Слика 1. Приказ основних елемената система клима уређаја [7]

4. ТЕОРИЈСКЕ ОСНОВЕ МИКРОКЛИМЕ

Микроклима према дефиницији представља стање природних појава на релативно малом простору. У последње вријеме људи проводе доста времена у моторним возилима, гдје бораве у малом простору, услови у кабини моторног возила могу у великој мјери утицати на њихов радни учинак, задовољство и здравље. Неповољни микроклиматски услови у возилу неповољно утичу на концентрацију, умор и будност, како возача тако и путника [3].

Када се човјек нађе у затвореном простору фактори који утичу на његов осјећај су вриједности везане за ваздух, површине које га окружују и индивидуални фактори самог човјека [1].

Четири основна фактора која утичу на пренос топлоте између човјековог тијела и околине су:

- Температура ваздуха t_a [°C]
- Средња температура зрачења t_{mr} [°C]
- Брзина ваздуха v_a [m/s]
- Влажност ваздуха која се исказује према релативној влажности RH [%] или преко парцијалног притиска водене паре p_a [Pa]

Два индивидуална фактора су:

- Метаболичка активност M [Met, W, W/m²]
- Изолација одјеће I_{cl} [Clo, m²K/W]

Како би се описало стање микроклиме неког простора, нужно је одредити све факторе који утичу на пренос топлоте између човјека и околине. Неопходно је

одредити претходно описане факторе: метаболичку активност, изолацију одјеће, температуре ваздуха, брзину ваздуха, средњу температуре зрачења, влажност ваздуха [4].

Међутим одређивање свих набројаних фактора одједном чини анализу сложену и тешком. Због поједностављења анализе комфора, тежи се ка увођењу једне величине која ће обухватати утицаје појединих параметара. Обједињени показатељи су:

- Оперативна температура t_o
- Еквивалентна температура t_{eq}
- PMV индекс и PPD показатељ

За мјерење обједињених показатеља потребно је и мање мјерне опреме која треба бити постављена у кабини возила, што код аутобуса не представља велики проблем због велике кабине али код возила са мањим кабинама игра значајну улогу [4].

5. ЕКСПЕРИМЕНТАЛНО ИСТРАЖИВАЊЕ

5.1. Спецификације испитиваног возила

За експеримент је кориштен минибус фирме FENIKSBUS d.o.o. Фирма FENIKSBUS је овлаштени бодибилдер компаније IVECO BUS. Бави се производњом свих главних компоненти каросерије аутобуса и подскопова каросерије.

Возило припада категорији М3, класи II, тип аутобуса је школски. Спецификације возила су:

- Модел: FBI 88 S – MY 2021.

- Мотор: IVECO дизел 2998 cm³, 135 kW - Еуро 6 Д.

- Клима: аутобус посједује аутоматску возачку климу са компресором 170 цц. У процесу надоградње уграђена је путничка клима Autoclima PT160, 16kW, компресор TM21, аутоматска контрола.

- Стакла: једнострука зелена, дебљине 5 mm, са 75% пропусности свјетлости, сем предњег вјетробранског стакла дебљине 6.76 mm које има 80%. Аутобус посједује и један кровни отвор са димензијама стакленог дијела 970x630 mm, као и статички одсис – електрични.

- Запремина путничког простора: 32 m³

- Број сједишта је 51 (сједишта за дјецу) + 1 сувозачко сједиште + возачево сједиште.

- Боја каросерије: жута.

- Димензије: 9.000x2.350x3.000mm.

- Максимална дозвољена маса: 7200 kg.

- Материјал костура: нехрђајући челик (ИНОКС-1.4003-феритни).

- Дијелови од полиестера: кров, задња стијена, предња стијена, предњи блатобрани.

- Дијелови од алуминијума: бочне оплате.

- Под: шпер-плоча (бреза) дебљине 12 mm, подна облога Таркет Смарт 121-600 (сива).

- Плафон и бочне облоге: обојени ембосирани лим.

- Изолација: стиропор дебљине 30 mm и 40 mm топлотне проводљивости од 0.37 W/mK, и стаклена вуна у предњем дијелу кабине дебљине 50 mm топлотне проводљивости 0.038 W/mK.



Слика 2. Аутобус FBI 88 S

5.2. Кориштена опрема за мјерење

Параметри који су мјерени током експерименталног истраживања су температура, влажност ваздуха, брзина струјања ваздуха и температура загријаних површина.

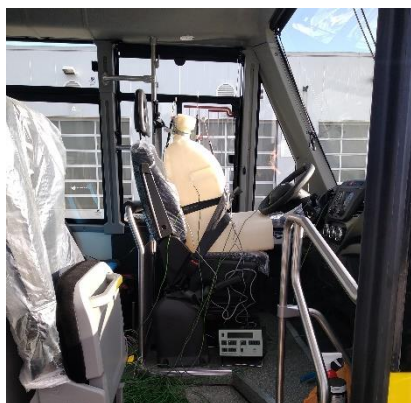
Током експеримента кориштена је следећа мјерна опрема:

- Brüel & Kjær 1213 анализатор
- Сензор температуре ваздуха (Brüel & Kjær MM 0034)
- Сензор брзине ваздуха (Brüel & Kjær MM 0038)
- Сензор температуре зрачења (Brüel & Kjær MM 0036)
- AMI 310 мултифункционални инструмент
- AMI 310 сензор за мјерење температуре ваздуха
- AMI 310 анемометар са лопатицама
- All-Sun EM520B инфрацрвени термометар
- Expert 9520 комуникациони модул
- Термопарови

5.3. Принцип испитивања

Мјерна опрема је постављена унутар кабине возила, након тога се затварају сви прозори и врата, возило је остављено на сунцу и чека се да се кабина загрије. Приликом првог испитивања скоро сва мјерна опрема је укључена тек након загријавања кабине, док је приликом другог испитивања мјерна опрема укључена прије загријавања кабине.

Након временског периода од 25 минута, када се кабина загријала улази се у аутобус и врата се одмах затварају, након тога се укључује клима. Приликом првог испитивања у кабини су биле три особе, а приликом другог двије особе.



Слика 4. Лутка са сензорима на возачком мјесту

5.4. Режим рада аутобуса

Током испитивања аутобус се није кретао. Мотор је радио у режиму празног хода. Број обртаја није прелазео 1000 о/мин. Сви прозори, врата и кровни отвори су били затворени. Клима је била подешена на 16°C што представља најнижу могућу температуру. Брзине вентилатора су биле подешене на највећу брзину. Сви вентилациони отвори су били максимално отворени.

6. АНАЛИЗА ДОБИЈЕНИХ РЕЗУЛТАТА

У овом поглављу рађена је анализа резултата испитивања и њихово међусобно поређење као и поређење са критеријумима по стандарду ISO 7730 за просторе у зградама, с обзиром да стандард не прописује посебно услове за аутобусе.

Tip prostora	Aktivnost W/m ²	Kategorija	Operativna temperatura °C		Maksimalna brzina vazduha m/s	
			Ljeto	Zima	Ljeto	Zima
Kancelarija javni zatvoreni prostor	70	A	24,5 ± 1,0	22,0 ± 1,0	0,12	0,10

Слика 3. Температурни критеријуми за просторе у зградама [2]

6.1. Резултати добијени мјерењем температуре ваздуха

• **Возачко мјесто** - Приликом анализе резултата на возачевом мјесту узети су у обзир резултати добијени са термопарова као и резултати добијени преко Brüel & Kjær анализатора (температуре ваздуха, температуре зрачења, брзина ваздуха) приликом првог испитивања. Приликом првог мјерења разлике вриједности пада температуре нису велике, у просјеку 5.9°C. Чак и температура мјерена Brüel & Kjær 0034 сензором биљежи пад од 4 до 5°C, што такође није много узимајући у обзир висину почетне температуре. Приликом другог испитивања падови температуре су много већи, у просјеку 12.8°C. Разлика у температурама између првог и другог испитивања је очекивана јер је приликом другог испитивања била укључена и путничка клима.

• **Путничко мјесто** - Резултати температуре и брзине ваздуха за путничко мјесто су читавани са AMI мултифункционалног уређаја. Добијене температуре након расхлађивања кабине за прво испитивање су 34.2 и 32.5°C, док су за друго испитивање нешто ниже 28.4 и 26.0°C.

Ако се температурни критеријуми за просторе у зградама са слике узму као мјеродавни и за аутобусе, резултати првог и другог мјерења не задовољавају критеријум, температуре нису биле у опсегу 24.5±1 °C. Као разлог томе може да се узме сама чињеница да су возила током испитивања била у стању мировања, није било струјања ваздуха око кабине које се јавља током вожње, и аутобуси су били у положају у ком су у великој мјери изложени сунцу. Режим рада аутобуса током испитивања је био такав је мотор радио на малом броју обртаја а самим тим ни компресор климе није могао да достигне свој пуни капацитет.

6.2. Резултати добијени мјерењем брзине ваздуха

Резултате добијене мјерењем брзине ваздуха уз помоћ AMI анемометра са лопатицама, приликом првог и другог испитивања није могуће поредити. Приликом првог испитивања брзина ваздуха је мјерена на сва 4 вентилациона отвора са растојања од 100 mm док је приликом другог испитивања брзина мјерена само на два средња отвора и са већег растојања. Иако се резултати не могу упоредити један са други, остаје

могућност да се упореде са прописаним резултатима. Брзина ваздуха мјерена помоћу Brüel & Kjær 0038 сензора код главе возача током првог испитивања након што се усталила је 0.19 m/s што се и подудара са прописаном вриједности.

6.3. Резултати добијени мјерењем температуре зрачења

Температура зрачења је мјерена само приликом првог испитивања. Просјечна вриједност током периода у коме је клима била укључена износи 55°C што је јако неповољно по возача. Оваква вриједност температуре зрачења директно је утицала и на вриједности PMV индекса и PPD показатеља. Вриједности PMV индекса од периода када је укључена клима па до неког периода док је клима још увијек радила је између 6.98 и 7.59. Пошто је према PMV скали доња вриједност за најнеповољни осјећај (врело) 3 [6], може се закључити да је добијени резултат врло неповољан. PPD показатељ у овом случају је очекивано 100%.

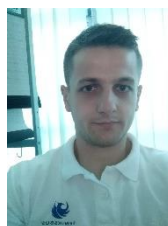
7. ЗАКЉУЧАК

У раду је рађено испитивање микроклиматских услова у аутобусу у топлим условима са два режима рада клима-уређаја. У првом режиму била је укључена само возачка клима а у другом и возачка и путничка клима. Добијени резултати након испитивања су веома неповољни. Далеко одступају од прописаних вриједности. Када се узму у обзир почетни услови испитивања а и сама чињеница да је возило стајало у мјесту било је и за очекивати да ће резултати бити лошији. За даљи рад на испитивању топлотног комфора аутобуса могло би се спровести испитивање приликом вожње, тада би клима остварила свој пуни потенцијал, због зависности од броја обртаја мотора. Струјање ваздуха око кабине би било веће што би резултирало повољнијим резултатима, а и сама изложеност сунцу би се мијењала током вожње. Као конструктивна побољшања на самој кабини би се могло навести додавање елемената за усмјеравање ваздуха на вентилационим отворима у путничком дијелу кабине, као и употреба стакала са вишим степеном тонирања.

8. ЛИТЕРАТУРА

- [1] Хома Д.: Истраживање топлотних услова у кабинама пољопривредних трактора, мастер рад, ФТН, Нови Сад, 2020.
- [2] ISO 7730.: 2005 [E] Ergonomics of the thermal environment — Analytical determination and interpretation of thermal comfort using calculation of the PMV and PPD indices and local thermal comfort criteria
- [3] Ружић Д.: Ергономија моторних возила, ФТН Нови Сад, 2020.
- [4] Ружић Д.: Утицај климатизације на топлотни комфор у путничком аутомобилу, магистарски рад, ФТН Нови Сад, 2006.
- [5] <https://www.guchen.com/industry-news/different-types-of-bus-airconditioning-system-for-body-builders.html> (септембар 2021. године)
- [6] <https://andersonenergy.com.au/how-to-quantify-comfort-with-pmv/>, (октобар 2021. године)
- [7] <https://alenauto.com/air-conditioning-service/>, (октобар 2021. године)

Кратка биографија:



Дејан Поповић рођен у Добоју, Босна и Херцеговина/Република Српска, 1995. год. Дипломирао 2018. год. на Факултету техничких наука, смијер Механизација и конструкционо машинство, на којем исте године уписује мастер студије, смијер Аутомобилско инжењерство.



Драган Ружић рођен у Новом Саду 1973.год. Докторирао је 2013. год. Од 2017. год. је постављен на мјесто директора Департамента за механизацију и конструкционо машинство. Запослен је као ванредни професор на Факултету техничких наука у Новом Саду.

УНАПРЕЂЕЊЕ ПРОИЗВОДНИХ СИСТЕМА ПРИМЕНОМ LEAN АЛАТА OPTIMIZATION OF PRODUCTION SYSTEMS BY APPLICATION OF LEAN TOOLS

Јована Марковић, Сања Бојић, Факултет техничких наука, Нови Сад

Област – МАШИНСТВО

Кратак садржај – У раду су дате теоријске основне LEAN методологије и представљени су неки од основних LEAN алата заједно са њиховом применом у производњи. Посебан акценат је стављен на примену појединих LEAN алата, визуелно управљање, Poka Yoke, Heijunka, у конкретним примерима производних система.

Кључне речи: Оптимизација, производни системи, LEAN, LEAN алати, Poka Yoke, Heijunka, визуелно управљање

Abstract – The Master Theses presents the theoretical foundations of the LEAN methodology and some of the basic LEAN tools together with their application of production. A special emphasis is placed on the application of certain LEAN tools, visual control, Poka Yoke, Heijunka, in concrete examples of production systems.

Keywords: Optimization, production systems, LEAN, LEAN tools, Poka Yoke, Heijunka, visual control

1. УВОД

У данашње време када сведочимо развоју технологија и расту производње огледа се и значај логистичких система где се захтева ефикасност. Сваки појединачни сегмент логистичког ланца је од веома велике важности, па тако унапређење применом LEAN методологије заузима значајно место у ланцу. Битан је цео аспект одржавања и планирања поготово када су већи системи у оптицају.

Акценат у сваком производном процесу свакако треба ставити на непрекидност токова материјала како би систем имао што мање губитке, уз максималну добит, чему се тежи кроз целу LEAN методологију. Потребно је обезбедити флексибилност производње, према потребама купаца. Оно што је потребно елиминисати или свести на минимум јесте сваки облик застоја, било на материјалу, у производњи или од стране радне снаге.

До застоја у производњи, самим тим до престанка непрекидног тока, долази услед великог броја залиха, што сваки систем тежи да избегне. Залихе за произвођача представљају заробљена обртна средства и сматра се губитком. Повећање залиха, како би се уравниотежио производни систем, никад није решење,

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је била др Сања Бојић, ванр. проф.

јер се опет јављају проблеми у току због нагомилавања материјала. Осцилација потражње не сме да буде велика, то је предуслов за непрекидан ток материјала и минималне залихе.

2. LEAN МЕТОДОЛОГИЈА

LEAN је филозофија која користи алате и технике за креирање и промену културе у циљу спровођења добре праксе побољшања процеса која омогућава смањење расипања, побољшање токова, више се фокусира на потребе купаца и подржава поглед на процес. LEAN имплементација би требало идеално да укључи програм континуираног побољшања који је више разноврсан и обухвата промену организације током дужег временског периода.

Према LEAN филозофији суштина целог концепта јесте додавање вредности производу елиминацијом губитака. Како би уопште примена LEAN-а била могућа, потребно је првенствено утврдити све губитке. Губици заправо представљају све кораке који не додају вредност производу.

Како би се губици лакше уочили у систему и лакше елиминисали, груписани су у осам категорија:



Слика 1. LEAN губици [1]

- **Грешке.** Односи се на грешке на производима, сваки квар се сматра грешком и свака поновна дорада одређеног производа.
- **Прекомерна производња.** Односи се на проблем стварања велике количине производа унапред, а да се не зна да ли ће тај производ уопште бити тражен у тој мери. Такође, додатни трошак представља обезбеђивање складишта за све те производе.
- **Чекање.** Односи се на ситуације када информације, запослени, опрема или материјали нису спремни. Сви производи који се у процесу монтаже не обрађују налазе се на чекању.
- **Неискоришћен људски потенцијал.** Одређене послове најбоље обављају запослени који тај

посао раде сваког дана, па обзиром на то треба слушати запослене јер ће они најпре уочити поједине неисправности и пропусте у целом систему.

- **Транспорт.** Представља премештање материјала са једне локације на другу. На самом производу транспорт не врши никакву промену, па самим тим не додаје вредност производу. Поред тога, постоји додатни ризик да се роба оштети, изгуби или закасни са испоруком.
- **Превелик ниво залиха.** Односи се на производе који нису продати купцима. Залихе су увек губитак јер заровљавају обртна средства предузећа. Превелик ниво залиха увек крије и додатне губитке.
- **Непотребна кретања.** Представља неоптимално или непотребно кретање запослених и машина између радних станица или објеката у производним процесима у којима се обављају све операције.
- **Прекомерна обрада.** Односи се на стварање производа прекомерног квалитета услед непознавања захтева и потреба купаца током времена.

Како би LEAN био имплементиран у производни систем, постоји низ LEAN алата који то омогућавају на најлакши могући начин у зависности од стања производног система. Неки од основних LEAN алата су представљени на Слици 2.



Слика 2. Основни LEAN алати [1]

У овом раду акценат је стављен на три LEAN алата, па обзиром на то у наставку ће они бити представљени, како би се лакше разумела њихова примена кроз практичне примере у наставку рада.

Poka Yoke је алат који утиче на спречавање појаве грешке. Циљ овог алата јесте отклањање ненамерних грешака које настају у производном процесу, најчешће од стране људи. Систем функционише тако што прати где им се јављају грешке и предузимају се одређене активности да се те грешке спрече. Превенција како не би дошло до настанка грешке је кључна у сваком производном систему и подразумева да се отклоне сва неразумевања и двоструки избори у процесима.

Heijunka је алат који се користи како би се нивелисале наруџбине купаца тамо где постоји велика осцилација приликом наручивања, на тај начин се постиже стабилност производног система. Овај алат се користи за нивелисање према обиму помоћу фиксних понављајућих распореда производње, али и за нивелисање према типу и миксу производа.

Визуелни менаџмент представља у континуирани процес стварања радног места на којем је елиминисан дефицит информација, тј. процес стварања окружења где су ствари очигледне од првог тренутка када су се нашли у датом подручју. Овај алат служи за доношење одлуке на микро нивоу. Такође, поједностављује сликовито и знаковно помаже запосленима да уштеде време и да на лак начин приме одређену количину информација како би им рад био продуктивнији.

3. КОНКРЕТНА ПРИМЕНА ОДРЕЂЕНИХ LEAN АЛАТА

Анализирана су два производна система где је представљена конкретна примена изабраних LEAN алата.

У првом случају представљена је једна компанија где је приказан систем за комисионирање где никако не сме да дође до грешке приликом узимања производа. Када су велике серије производа у питању, као што је овде случај, где постоје мали делови попут завртњева и сл. велика је шанса да ће доћи до грешке. У овом случају је јако тешко испратити ниво залиха и усагласити стање уколико то није дефинисано посебним системом. Производ за који се врши комисионирање јесте електо орман са пропратном додатном опремом. Умањење процента грешака се може постићи на више начина. Конкретно овде су сви производи и полупроизводи сортирани у одговарајуће кутије како би се њима лакше манипулисало. На свакој кутији постоји индикаторска лампица и количина колико комада производа радник треба да узме из одређене кутије. Индикаторска лампица светли у две боје. Црвена боја значи да је радник већ узео одређени производ из кутије и да нема потребе да се враћа на ту кутију. Зелена лампица засветли изнад одређене кутије када је тај производ на реду за паковање. Када радник узме дефинисан број производа из одређене кутије индикатор опет мења боју у црвену. Такође, како радник случајно не би узео неки производ из кутије, а да није приметио визуелни ефекат, постоји и физичка препрека за узимање производа из кутије у виду врата која се спуштају када се узме одређени број комада и када индикаторска лампица поново буде црвена. На Слици 3. је приказан унапређен систем овог производног система који је изабран као пример за имплементацију Poka Yoke алата.



Слика 3. Унапређен систем применом Poka Yoke алата [4]

У другом случају је приказан производни систем који се бави производњом стандардних путничких и спортских возила. Поставља се питање како на прави начин ускладити и нивелисати ниво производње према типу и миксу у овом систему. Уз помоћ Heijunka кутије и Kanban картица предложено је унапређење постојећег система.

Систем рада је подељен на рад у радним хелијама где ради одређена група радника на више радних станица. Како би им комуникација била олакшана и како би се умањиле грешке и постигли бољи и ефикаснији циљеви у систему потребно је унапредити га и подићи на виши ниво. На првој радној станици код сваке радне хелије постоји држач за Kanban картице са потребним подацима који служе за контролу производње у тој одређеној радној хелији. Сваки држач је напуњен наранџастим и плавим Kanban картицама – наранџасте су за стандардна путничка возила, док је плава картица намењена за спортска возила.

Ове картице сигнализирају радним хелијама које возило је потребно да се прави и по којој поруци је та наредба издата. Супервизор или оператер повлачи Kanban картице које представљају захтеве купаца из Heijunka кутије и ставља их у држаче картица одређене радне хелије, како би се приступило процесу производње. Код сваке радне хелије постоји вођа тима који има задатак да повуче возила из супермаркета производа из претходне радне хелије све док повлачење Kanbana не испуни тражени капацитет. Ова процедура се понавља кроз процес производње. Оператер на првој радној позицији у свакој радној хелији повлачи Kanban картицу из Heijunka кутије, како би почело да се производи возило које је назначено у Kanban картици. Оператерима је једино дозвољено да производе возила назначена од стране Kanban картица. Kanban картица се прослеђује кроз производњу са возилом и даље кроз радну хелију све док се возило не повуче из супермаркета од стране друге радне хелије. Вођа тренутне радне хелије затим ставља Kanban картицу назад у Heijunka кутију, осим у случају последње хелије која ставља Kanban картицу у део преузимања производа. Променом система увођењем Heijunka кутије и Kanban картица, залихе и производње фабрике, може се унапредити време реализације (Lead Time) производа. Пратећи Heijunka кутију и Kanban систем постиже се то да се оствари Just-In-Time производња према захтевима купаца.



Слика 4. Имплементација Heijunka кутије и Kanban картица [4]

Како би се прича заокружила, имплементација визуелног менаџмента је представљена кроз претходна два примера, јер је у оба производна система примењен визуелни менаџмент. Потенцијалних места за унапређење производних система има разних. У сваком случају је могуће додатно унапредити процес и додати ознаке на поду, визуелне ознаке о начину употребе машине ради безбедности итд. Конкретно, када су у питању претходна два случаја визуелни менаџмент видимо у виду светала различитих боја и у виду ознака и места за производ. Углавном су заступљене препознатљиве боје које владају у сваком систему – црвена и зелена.

Када се оне уоче јасно је и недвосмислено шта оне значе. У првом примеру где је рађено комисионирање за производе електро ормана уочава се визуелни менаџмент на местима где имамо индикаторске лампице које светле у две боје. Црвена боја је светлела када је производ већ узет са стања из одређене кутије у одређеној количини.

На тај начин радник јасно визуелно види да је тај производ већ узео јер му црвена боја указује на *стоп, забрану, заустављање*. Визуелна потврда му одмах до знања стави о чему је реч, без да он додатно троши време уколико би били присутни неки други системи потврде. У другом примеру где је представљен систем производње возила такође је заступљен визуелни менаџмент.

На местима где се производ у једној радној хелији ставља на место за прослеђивање возила у другу радну хелију, заједно са Kanban картицом. Та места су означена зеленом и црвеном бојом, што су опет боје које су јасно одређене и недвосмислене. Када их радник види тачно може да уочи да ли је нешто проблем или није. У црвени квадрат се стављају производи са грешком и јасно је описана процедура како се са таквим производом поступа. У зелени квадрат се ставља исправан производ који се прослеђује даље кроз производну линију, када га једна радна хелија заврши.

На Слици 5. а) приказан је случај имплементације визуелног менаџмента у виду индикаторских лампица, док је док је на слици 5б) приказана примена визуелног менаџмента у виду означених места у облику квадратних поља различите боје – за производе са грешком и исправне производе који иду даље у процес производње.



Слика 5. а) Имплементација визуелног менаџмента у производним системима



Слика 5. б) наставак

4. ЗАКЉУЧАК

У данашње време се све више озбиљних компанија и производних система који производе производе или пружају услуге, опредељује управо за примену LEAN концепта и филозофије. Како се тежи унапређењу битно је кренути од начина размишљања и много дубље ући у суштину проблема целокупног система. Дакле, LEAN није само алат којим се постижу жељени резултати, већ је то животна филозофија и примењива је у многим сферама.

LEAN концепт није ни мало једноставан и треба му се посветити са пуном пажњом. Дешава се да компаније не увиде значај примене ове методологије, управо јер не разумеју срж и могућности концепта. Понекад је потребно прво променити начин размишљања, па онда приступити примерни LEAN-а.

Фокус је увек на унапређењу система, уз што мање губитке како би се остварила максимална добит. Како би се остварило управо то што је претходно наведено и како би се постигла оптималност било ког система потребно је да се остваре основе које су темељ за успешност пословања компаније. Основне ствари којима се тежи јесте да се производња одвија по принципу Just-In-Time и да буде остварен непрекидан ток материјала.

Имплементацијом изабрана три алата у конкретним примерима постигнуто је очигледно унапређење у систему у виду уштеде времена и смањења настанка одређених грешака.

Поред претходно наведених унапређења, комуникација у оба система је побољшана и омогућен је бољи ток информација и материјала.

5. ЛИТЕРАТУРА

- [1] Др Бојић С.: Презентација: LEAN алати, ФТН, Нови Сад, 2020.
- [2] Бекер И., Лазаревић М., Морача С., Радловачки В., Рикаловић А., Тешић З., Шевић Д.: LEAN систем, ФТН Издаваштво, Нови Сад, 2017.
- [3] Вомак Џ., Џонс Д.: Филозофија LEAN концепта, ФТН Издаваштво, Нови Сад, 2012.
- [4] <https://www.youtube.com>, датум приступа 19.10.2021.

Кратка биографија:



Јована Марковић рођена у Сомбору 1997. год. Дипломски рад на Факултету техничких наука из области Дизајна – „Основне смернице за пројектовање регалних дизајна“ одбранила је 2020. године.
контакт: markoviicjovana@gmail.com



Сања Бојић рођена је у Карловцу 1981. Докторирала је на Факултету техничких наука 2013. год, а од 2019. је у звању ванредни професор. Област интересовања су логистика, складишта и симулације.

ПРИМЕНА LEAN МЕТОДОЛОГИЈЕ У ОДРЖАВАЊУ ПРОИЗВОДНИХ СИСТЕМА APPLICATION OF LEAN METHODOLOGY IN MAINTENANCE OF PRODUCTION SYSTEM

Јелисавета Ковачић, Сања Бојић, *Факултет техничких наука, Нови Сад*

Област – МАШИНСТВО

Кратак садржај – У раду је описано одржавање производних система, и обрађена концепција превентивног и корективног одржавања. Представљене су теоријске основе Lean методологије, и неких од алата који се користе у производним системима. У фокусу рада је начин функционисања производне линије ротора, као и алата који се примењују на споменутој линији, 5C, Kanban и TPM.

Кључне речи: LEAN филозофија, LEAN алати – 5C, Канбан, ТПМ, одржавање техничких система

Abstract – The paper describes the maintenance of production systems, and discusses the concept of preventive and corrective maintenance. The theoretical foundations of LEAN methodology, and some of the tools used in production systems are presented. The focus of the work is the way the rotor production line functions, as well as the tools used on the line, 5C, Kanban and TPM.

Keywords: LEAN philosophy, LEAN tools – 5S, Kanban, TPM, maintenance of technical systems.

1. УВОД

Тржиште се данас јако брзо мења. Нестабилни, непредвидиви и висококонкуретни услови пословања изискују свакодневно праћење тржишта и захтева. Праћење трошкова је постао императив, а савремено пословно тржиште све више даје на значају друштву.

Другим речима, нова пословна филозофија би гласила “купац у фокусу”. Примена LEAN методологије у производним системима омогућила је континуални процес и регулисање трошкова, кроз индентификацију и елиминисање непотребних активности. Како би се задовољили захтеви тржишта, неопходно је обезбедити висок степен оперативности производних система, што је могуће уз примену Lean методологије и TPM алата.

1.1 Одржавање производних система

Одржавање се може описати као функција која пружа смернице за спровођење активности одржавања, поред остваривања техничке и менаџерске контроле одржавања система.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је била др Сања Бојић, ванр. проф.

Постоје два прилаза методологији одржавања, односно одржавање према поузданости (ОПП) и тотално продуктивно одржавање (ТПО). Одржавање према поузданости (ОПП) је методологија која је заснована на познавању карактеристика поузданости, на основу које се врши прогноза будућих стања, заправо предвиђа се појава отказа.

Примарни задатак овог типа одржавања јесте да обезбеди поузданост и безбедност објеката који се одржавају, у случају појаве кvara или отказа да врати систем на претходни ниво безбедности и поузданости и остваривање свих задатака уз што мање трошкова. Тотално продуктивно одржавање (ТПО) је методологија која се заснива на процени тренутног стања система који се одржава. Основни задатак и циљ ове методологије јесте смањење застоја због изненадних отказа на машинама и потројењима.

1.1.1 Концепција одржавања

Постоје две основне концепције одржавања, а то су превентивно и корективно одржавање. Превентивно одржавање се може описати као сервисирање од стране појединаца и група који су укључени у систем одржавања техничког система. Главни циљеви превентивног одржавања су продужити животни век машине, смањити појаву критичних кварова опреме, омогућити боље планирање и предвиђање активности и смањити губитке у производњи. Превентивно одржавање се може састојати од седам елемената, односно, контроле, сервисирања, калибрације, тестирања, инсталације, подешавања и регулације.

У погледу врсте превентивног одржавања имамо, основно одржавање, превентивну замену и одржавање према стању. Корективно (интервентно) одржавање дефинисано је као спровођење корективних мера услед кvara или откритих недостатака у току превентивног одржавања, све док се технички систем не доведе поново у оптимално стање.

Корективно одржавање врши се тек након што настане застој. Активно време поправке са састоји од шест компоненти, а то су: време припреме, време локације грешке, време које је потребно да се достави резервни део, време корекције грешке, време подешавања и калибрације и време провере.

Још једна битна ствар поред тачно одређеног активног времена поправке јесте и смањење времена самог одржавања. Ово је јако битно како и због саме производње, тако и због трошкова који настају приликом овог одржавања

1.2 Lean методологија

1.2.1 Развој аутомобилске индустрије

Аутомобилска индустрија данас представља најјачу индустријску грану. Круцијалну промену у свету аутомобилске индустрије изазвало је увођење покретне траке. Идеју да покретну траку имплементира у аутомобилску индустрију први је имао Хенри Форд (енг. Henry Ford), амерички предузетник и оснивач Форд Мотор Компаније (енг. Ford Motor Company). Идеологија компаније Форд се огледа у малом броју квалификованих радника, и великом броју неквалификованих радника. Иако је систем уиграм, и зна се тачна расподела посла постоје два проблема. Први је нефлексибилна производња, а други је неправилно одрађена монтажа.

1.2.2 Историја Toyota Production System-a (Lean koncepta)

Почеци развоја Тојете се везују за Сакичи Тојоду, који се често назива и “оцем јапанске индустрије”. Сакичи Тојода 1907.године оснива компанију под називом Toyota Loom Company. Од 1926.године па све до данас, ова компанија је препознатљива под називом Toyota Industries Corporation. Кичиро Тојода, син Сакичи Тојоде заслужан је за активност која се и данас спроводи у фабрикама, познатију по називом “Genchi genbutsu”, што значи иди на лице места и посматрај догађај својим очима. Поред Сакичи Тојоде и Кичиро Тојоде, за развој Тојотине фабрике заслужни су Еиџи Тојода, менаџер фабрике и Таичи Оно, шеф производње.

Табела 1: Студија различитости приступа организације у фабрика

Генерал Моторс, Фраммингем, САД (класична масовна производња)	Тојота, Такаока, Јапан (Lean концепт- TPS)
Велики број радника у линији производње. Непродуктивна производња.	Ласно дефинисан задатак сваког радника.
Нагомилавање залиха, и уоћавање грешке тек пред процес монтаже.	Нема нагомилавања залиха. Врши се директан утовар и транспорт након завршеног производа.
За свако радно место постоји одређена количина залиха, која може бити довољна и за неколико недеља.	Не постоји могућност постајања залиха између радних операција, јер се операције нижу једна за другом.
Искључиво су менаџи овлашћени да зауставе процес производње у случају да дође до грешке.	Сваки радник на радном месту сме да заустави производњу када детектује грешку.
Постојање улазник магацина, и складиштења опреме за неколико недеља.	Потпуно непостојање улазних магацина. Материјале стижу у тачно одређено време.
Конструкциони, производни и радници управљања веома мало проводе времена у фабрики.	Честа присутност радника и инжењера у фабрики.
Недовољно мотивисање радника, самим тим и смањења жеља радника да привређују.	Велико мотивисање радника, самим тим радници су орни за рад, истрајни, знајтељни и радни.

1.2.3 Основни губици према Lean филозофији

Тојотин систем издваја осам Lean губитака:

1. Прекомерна производња (енг. Overproduction) представља основну разлику између масовне и Lean производње.
2. Чекање (енг. Waiting) представља празан ход машине или оператера услед уских грла у процесу, лошег такта или застоја;
3. Транспорт (енг. Transport) смањење непотребног кретања материјала које не додаје вредност производу, нпр. транспорт између радних јединица;

4. Неодговарајућа обрада (енг. Extra processing) узрокована је непотребним повећањем квалитета, које купац није захтевао и које не може да плати.
5. Превелик ниво залиха (енг. Inventory) представља најсакривенији и највидљивији губитак у целом систему.
6. Непотребно кретање (енг. Motion) је губитак који се често сматра небитним, а итекако може да утиче на цео систем.
7. Шкарт (енг. Defects) је јако заступљен у масовној производњи. Одлагање детектовања шкарта може само произвести појаву нових прекривених губитака.
8. Неискоришћен људски потенцијал (енг. Non-utilized talent) је губитак који настаје услед неукључивања радника у систем, као и неусвајањем њихових идеја и вештина.

1.2.4 Основни принципи Lean филозофије

Основни Lean принципи омогућавају компанијама да правилно искористе своје ресурсе, и смање време између почетка креирања производа па све до завршетка. Постоји пет основних Lean принципа:

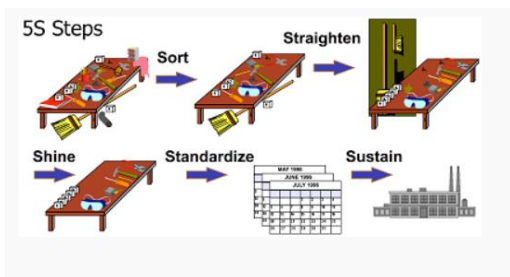
1. Дефинисање вредности (енг. Specify Value) овај принцип дефинише купац.
2. Успостављање тока настајања вредности (енг. Map the Value Stream) Циљ успостављања тока вредности је да се током целог процеса уклоне сви елементи који не доприносе вредности у датом току.
3. Успостављање непрекидног тока (енг. Improve the Flow) има за циљ да процес тече непрекидно.
4. Успостављање pull система (енг. Implement Pull) доприноси да се током целог процеса испоштује такт који диктира производња.
5. Унапређење (енг. Pursue Perfection) у циљу достизања савршенства. Све вредности које не доприносе вредности, морају бити уклоњене.

1.3 Основни алати Lean филозофије

Да би Lean филозофија могла да се имплементира унутар једног предузећа, потребно је увести и спровести одговарајуће методе, односно алате. Правилан редослед коришћења алата омогућава засигурно успешну, ефикасну и сигурну производњу.

5С алат који припада Lean филозофији, и чини камен темељац споменуте филозофије, базира се на одржавању хигијене радног места, тако да ни једне секунде не омета производни процес. Први корак је сортирати ствари, односно одвојити све потребне ствари у радном окружењу од непотребних. Спремити је корак који наступа након сортирања. Подразумева да су ствари које су током процеса сортирања издвојене буду обележене и на свом месту.

Средити ствари је корак у коме сви запослени учествују, односно односи се на проверу након завршетка коришћења машине. Стандардизовање претходна три процеса постиже се уз помоћ израде радних контролних упутстава. Стално одржавати, односно подићи свест код радника да су претходна четири корака јако битна за успешно одржавање.



Слика 1: Кораци током примене 5С алата

Kanban представља систем картица за управљање производњом, у складу са pull производњом. Pull производња функционише по принципу тако што наредно радно место сигнализира претходном шта му је потребно, и на тај начин „повлачи“ шта му је потребно од делова са претходног радног места. Исправном применом Kanban система могуће је унапредити ток материјала, спречити прекомерну производњу, максимално умањити неискоришћеност материјала, повећати флексибилност производње, као и олакшати производњу путем визуелних сигнала. Постоје четири типа канбана:

1. Процесни Kanban (енг. In-process Kanban) се користи као визуелни сигнал који диктира темпо кретања делова у процесима са непрекидним током.
2. Kanban повлачења (енг. Withdrawal Kanban) представља систем потражње производа од претходног процеса на основу наредног.
3. Производни Kanban (енг. Production Kanban) служи да искомунукација наручивање и допуну складишта.
4. Добављачки Kanban (енг. Supplier Kanban) ово су Kanban картице које су намењене за материјале које је потребно направити.

2. ТОТАЛНО ПРОДУКТИВНО ОДРЖАВАЊЕ (Total Productive Maintenance -TPM)

Сврха коришћења овог алата јесте да допринесе елиминацији губитака који се јављају услед неоперативности опреме, и самим тим да омогући бољу, успешнију и сигурнију организацију производње. Могућност потпуног искоришћења опреме и поузданог функционисања система се остварује кроз главне циљеве TPM-а, а то су: изградња свеобухватног система превентивног одржавања, остваривање ефективног и ефикасног коришћења опреме, развија и имплементира активности превентивног одржавања, прати и омогућава одржавање опреме током животног века опреме, пружа помоћ и подршку запосленима, мотивише запослене и формира тимове који су задужени за континуално унапређење. Како би предочени циљеви TPM-а били спроводљиви, дефинисани су основни градивни елементи или стубови TPM-а. Осам највећих стубова TPM-а:

1. 5С,
2. аутономно одржавање,
3. Kaizen,
4. планирање интервенције одржавања,
5. обука,
6. администрација TPM-а,
7. здравље и безбедност на раду и заштита околине.

3. ПРИМЕНА TPM-а У ПРОИЗВОДНОМ СИСТЕМУ

3.1 Увод у коришћење TPM-а у производном систему

Основна полазна тачка при креирању и успостављању TPM-а на линији намеће се наравно, прецизно дефинисање броја оператера, подешавача линије (енг. Setter), вођа линије (енг. Line leader), сменовођа (енг. Shift leader) и техничара одржавања (енг. Maintenance technician). Након дефинисања укупног броја радника који ће вршити активности TPM-а, потребно је дефинисати време трајања TPM активности, као и прецизно дефинисати, ко, када и на који начин ће обављати радње.

3.2 Опис рада производне линије

Погон о коме ће бити речи, део је фабрике која се бави израдом ротора и статора. Дужином линије која је намењена за склапање ротора постоји асемблован аутоматизован транспортни систем, који опслужује све станице у оквиру линије.

Линија почиње станицом распакивања ламинације ротора, потом се у ламинацију утискује осовина ротора. После утискивања осовине у ламинацију, приступа се обради осовине, после чега долази до пресовања комутатора ротора, да би се затим ротор одлагао у станицу за електроиндуктивну изолацију. Када је завршен процес изолације, приступа се намотавању жице на ламинацију.

Након намотавања, жица се заварује за комутатор, следи брушење комутатора, монтирање лежајева на осовину и урезивање пужног навоја на осовину. На крају ових станица имамо готов полупроизвод, у смислу да је ротор, као такав, у потпуности асемблован, па наредне станице све до визуелне инспекције, која иначе представља последњу станицу.



Слика 2: Ротор

3.3 Креирање TPM-а за производну линију

Почетна тачка при креирању TPM-а, пре пуштања у рад производне линије, јесте препорука произвођача машина. Наведене препоруке користе се као основа, и углавном се примењују у форми аутономног одржавања. С друге стране, пракса је показала да се на основу препорука може само градити основни принцип начина одржавања. Веома битан фактор јесте искуствено одржавање тј. примена искуства инжењера одржавања, као и процесних инжењера, у циљу откривања и превенције проблема.

3.4 Аутономно одржавање линије

Највећи акценат у аутономном одржавању на линији ротора, јесте примена Kanban и 5C система. Применом ова два алата, адекватном обуком радника, стандардизацијом активности, увођењем RFID читача који служе за потврду добро одрађене замене (енг. Changeover-a), постигнут је приметан скок у проценту ефикасности, као и пад у проценту губитака, у погледу организације. Обављене активности оператери и сетери конфирмишу потписом на табли која је увек јасно истакнута поред машине.

3.5 Превентивно одржавање линије

Под превентивним одржавањем на линији ротора у поменутом погону, подразумева се одржавање у оквиру ТРМ-а, које се одвија на недељном нивоу, а чије се активности извршавају од стране техничара одржавања. Превентивно одржавање се одвија у оквиру планског застоја. На линији ротора, трајање ТРМ-а је 180 минута. У оквиру наведених 180 минута, радници производње такође раде активности аутономног одржавања које су дефинисане као недељне, док техничари имају списак стандардних активности које добијају у облику налога текуће недеље у којој се одвија ТРМ.

4. ЗАКЉУЧАК

У фокусу Lean-а је купац, то значи да је купац тај који диктира темпо. Водећи се овом филозофијом, сви алати настају у складу са њом. Сврха алата је максимално поједноставити процесе, смањити њихово време, производити квалитетно и увек размишљати како је процес могуће побољшати. Применом Lean алата, 5C, Kanban-а и ТРМ-а, на производној линији омогућен је поузданији и дефинитивно сигурнији процес рада. Споменута два алата омогућавају позитиван исход ТРМ-а. Свакодневна фокусираност на процес, примећивање најситнијих детаља, детектовање и анализирање свих појава које доводе до застоја, усредсређеност на квалитет производа и тежња ка нула броју застоја јесу само једни од критеријума којим ТРМ тежи.

5. ЛИТЕРАТУРА

- [1] Јованчић П.: Одржавање рударских машина. Београд: Рударско – геолошки факултет; 2014.
- [2] Јованчић П.: Техничка дијагностика. Београд: Рударско – геолошки факултет; 2016.
- [3] Ликер Џ., Мајер Д.: The Toyota way Fieldbook. Mc Graw-Hill, 2005.
- [4] Вомак Џ., Џонс Д.: Филозофија Lean концепта. Нови Сад: Факултет техничких наука, 2012.
- [5] Ликер Џ.: The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. Mc Graw-Hill, 2003.
- [6] Монден Ј.: Toyota Production System: An Integrated Approach to Just-In-Time. Productivity Press; 4 edition, 2011.
- [7] Накајима С.: Introduction to TPM: Total Productive Maintenance. Productivity Press, 1998.

Кратка биографија:



Јелисавета Ковачић рођена у Пожаревцу 1995. године. Мастер рад на Факултету техничких наука у Новом Саду, из области Логистика – Примена Lean методологије у одржавању производних система одбранила је 2021. године.



Сања Бојић рођена је у Карловцу 1981. године. Докторирала је на Факултету техничких наука у Новом Саду 2013. године, а од 2019. године изабрана је у звање ванредног професора. Области истраживања су логистика, складишни системи и симулације токова материјала.

**PROJEKAT BAZENSKE TEHNIKE I SISTEMA GREJANJA I VENTILACIJE
ZATVORENOG BAZENA****DESIGN OF POOL AND HVAC EQUIPMENT FOR INDOOR POOL**Dejan Lazarević, *Fakultet tehničkih nauka, Novi Sad***Oblast – MAŠINSTVO**

Kratak sadržaj – U radu je dato tehničko rešenje i kompletan projekat za izvođenje mašinskih instalacija bazenske tehnike kao i grejanja i ventilacije objekta zatvorenog bazena koji je namenjen za sport i rekreaciju. U radu je prikazan kratak pregled standarda za bazensku tehniku kao i proračuni za dimenzionisanje opreme koja je neophodna za pravilno funkcionisanje kompletnog sistema. Pored toga, usvojena je sva oprema prema prethodno urađenom proračunu kao i tehnički opisi svakog od sistema. Na kraju rada nalaze se grafički prikazi sa situacionim planom kao i dispozicijom usvojene opreme te tipskim detaljima za ugradnju.

Glavne reči: Grejanje, ventilacija, gubici toplote, toplotne pumpe, bazenska tehnika

Abstract – The paper presents a technical solution and a complete project for performing mechanical installations of pool equipment as well as heating and ventilation of the indoor pool facility which is intended for sports and recreation. The paper presents a brief overview of the standards for pool technology as well as calculations for sizing the equipment necessary for the proper functioning of the complete system. In addition, all the equipment was adopted according to the previously made calculation, as well as the technical descriptions of each of the systems. At the end of the paper, there are graphical representations with the situation plan as well as the disposition of the adopted equipment and standard details for installation.

Glavne reči: Heating, ventilation, heat loss, heat pumps, pool technology

1. Uvod

Za potrebe zagrevanja i pripreme bazenske vode zatvorenog bazena sa prelivom za sport i rekreaciju, dimenzija 25x12,5x2 m (dužina x širina x dubina) potrebno je izraditi projektno rešenje termotehničkog postrojenja i postrojenja bazenske tehnike.

Pored ovoga potrebno je izraditi projektno rešenje sistema grejanja i ventilacije istog objekta, prema zadatim arhitektonskim osnovama.

Izvor toplote za dati objekat je toplotna podstanica sa priključkom na gradsku vrelodvodnu mrežu. Za potrebe prostora zatvorenog bazena u svrhu odvlaživanja i ventilacije, potrebno je predvideti sistem u skladu sa

zahtevima u standardu DIN 19643 i u smernici VDI 2089. Za potrebe grejanja predvideti sistem podnog i vazdušnog grejanja a za potrebe zagrevanja bazenske vode predvideti sistem sa solarnim kolektorima i dogrevanjem preko toplotne podstanice. Predvideti i pripremu tople sanitarne vode za tuševe.

Tekstualni deo ovog rada sadrži sve neophodne proračune, prikaz usvojene opreme kao i detaljne opise pomenutih termomašinskih sistema dok grafički deo ovog rada sadrži situacioni plan i položaj objekta kao i dispoziciju usvojene opreme.

Objekat se nalazi na području grada Novog Sada.

**2. PREGLED STANDARDA DIN 19643
„BAZENSKA TEHNIKA“**

Nemački standard DIN 19643 „Bazenska tehnika“ podeljen je u tri osnovne celine:

- 1.deo: Opšti zahtevi
- 2.deo: Kombinacije postupaka: adsorpcija – flokulacija – filtriranje – hloriganje
- 3.deo: Kombinacija postupaka: flokulacija – filtriranje – ozoniranje – sorpcijska filtracija – hloriganje

Standard propisuje da voda kojom se puni bazen mora zadovoljavati opšte higijenske uslove karakteristične vodi za piće. Ako se u vodi nalaze nedopuštene količine supstance koje ometaju naknadnu obradu vode one se moraju ukloniti sa za to predviđenim uređajima. U sledećoj tablici su prikazane neke od supstanci kao i njihove dopuštene koncentracije:

Tablica 2.1. - *Dopuštene koncentracije supstanci u bazenskoj vodi [1]*

Supstanca	mg/l	mmol/m ³
Željezo	0,1	1,8
Mangan	0,05	0,9
Amonijak	2	110
Polifosfat	0,005	0,16

Standard DIN 19643 „Bazenska tehnika“ propisuje određene smernice u kojima se naglasak stavlja na blisku saradnju arhitekata, inženjera građevine i inženjera mašinstva prilikom projektovanja i izgradnje bazena. Navedene strane se tokom svih faza projektovanja i izgradnje moraju pridržavati svih važećih zakona i propisa vezanih za izgradnju, odvodnjavanje, higijenske uslove itd. Za održavanje zadovoljavajućih (propisanih) higijenskih uslova potrebno je osigurati ispravan rad svih uređaja

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Đorđije Doder.

bazenske tehnike i vršiti redovne nadzore nad automatizovanim procesima. Uređaji za automatsku dezinfekciju i kontrolu pH vrednosti vode moraju biti u konstantnom pogonu. Dno bazena potrebno je čistiti minimalno dva puta sedmično, a bočne zidove minimalno jednom u dve sedmice, četkama ili vakuumskim uređajima. Bazen se mora obavezno isprazniti jednom godišnje te temeljno očistiti uređajem za pranje pod pritiskom i ribanjem. Takođe je obavezna dezinfekcija dna i zidova bazena. Nakon dezinfekcije potrebno je temeljno ispiranje. Kanali za preliv moraju se čistiti jednom sedmično i to dok cirkulacione pumpe nisu u pogonu. Za pravilno čišćenje potrebno je skinuti zaštitne rešetke. Kompenzacioni bazeni se po potrebi, a minimalno svakih pola godine, moraju isprazniti i temeljno očistiti i dezinfikovati [2].

3. TEHNIČKI OPIS BAZENA SA PRELIVOM

Voda za kupanje u bazenima može biti iz različitih izvora. Pre svega, ona mora biti bistra i čista te pri maksimalnom opterećenju mora ispunjavati sve postavljene zahteve kvaliteta. Kako prilikom kupanja može doći do slučajnog unošenja vode u organizam, ona mora odgovarati i higijenskim zahtevima vode za piće i konzumaciju. Tokom boravka u bazenu vodu zagađuju prisutni kupaći (masti, znoj, kosa, čestice kože, perut, mokraća, mikroorganizmi i sl.) te se zbog toga voda mora stalno menjati čistom ili pročišćavati. Velika većina svih nečistoća se nalazi na površini vode, te se zbog toga preporučuje da na pročišćavanje upravo odlazi voda sa površine bazena.

Za predmetni bazen predviđena je obrada vode u zatvorenom i kontinuiranom toku pri čemu se održava kvalitet propisan pomenutim standardom.

Sistem bazenske tehnike čine:

- Filterska pumpa sa grubim prefilterom
- Sistem za automatsko dopunjavanje vode
- Peščani filter sa baterijom sa 5 ručnih ventila
- Oprema za dezinfekciju vode
- Oprema za snižavanje pH vrednosti vode
- Izmenjivač toplote
- Solarni kolektori za zagrevanje bazenske vode
- Sistem automatskog upravljanja

4. PRORAČUN BAZENA SA PRELIVOM

U sledećoj tablici prikazani su ulazni podaci potrebni za proračun.

Tablica 4.1 *Ulazni podaci za proračun*

Neto površina bazena	312,5	m ²
Zapremina bazena	625	m ³
Dubina bazena	2	m
Korišteni medijum	Voda iz gradskog vodovoda	

Određivanje potrebnog protoka bazenske vode računa se prema sledećem izrazu:

$$Q = \frac{N}{k} = \frac{69,4}{0,5} = 138,9 \text{ [m}^3/\text{h]}$$

Q – protok bazenske vode [m³/h]

N – nazivno opterećenje [1/h]

k – faktor specifičnog opterećenja [m⁻³]

Nazivno opterećenje bazena je projektom bazena utvrđeni broj kupaća u jednom satu rada bazena te se računa prema sledećem izrazu:

$$N = A \cdot \frac{n}{a} = 312,5 \cdot \frac{1}{4,5} = 69,4 \approx 70 \text{ kupaća}$$

n – frekvencija kupaća u satu n=1 [h⁻¹]

A – neto površina bazena [m²]

a – specifična površina vode koju zauzima pojedini kupać; a=4,5 [m²]

Potrebno vreme za izmenu celokupne količine vode se računa pomoću sledeće formule:

$$t_{izmene} = \frac{V}{Q_F} = \frac{625}{157} = 4 \text{ [h]}$$

t_{izmene} – potrebno vreme za izmenu celokupne količine vode [h]

V – ukupna zapremina vode u koritu bazena [m³]

Q_F – ukupan protok vode kroz filter [m³/h]

Iskustvena preporuka je da vreme ismene vode ne prelazi 4h.

Izraz za računanje potrebne zapremine vode kompenzacionog bazena glasi:

$$V_K = V_W + V_V + V_R = 8,12 + 5,25 + 13,08 = 26,45 \text{ [m}^3\text{]}$$

V_K – potrebna zapremina kompenzacionog bazena [m³]

V_W – zapremina istisnute vode zbog uticaja talasa [m³]

V_V – zapremina istisnute vode zbog kupaća [m³]

V_R – zapremina vode potrebne za pranje filtera [m³]

Zapremina vode istisnuta preko prelieva zbog talasa računa se prema sledećem izrazu:

$$V_W = 0,052 \cdot A \cdot \frac{0,144 \cdot Q_F}{L} = 0,052 \cdot 312,5 \cdot 10^{-\frac{0,144 \cdot 157}{75}} = 8,12 \text{ [m}^3\text{]}$$

L – ukupna dužina ivice preko koje se prelieva bazenska voda [m]

Zapremina vode istisnute zbog kupaća se računa prema:

$$V_V = 0,075 \cdot n_{kupača} = 0,075 \cdot 70 = 5,25 \text{ [m}^3\text{]}$$

n_{kupača} – izračunati broj kupaća

$$n_{kupača} = \frac{A}{a} = \frac{312,5}{4,5} = 69,44 \approx 70 \text{ [osoba]}$$

Potrebna količina vode za pranje filtera se računa prema brzini i trajanju pranja. Za velike bazene se koristi brzina pranja filtera od 40-50m/h dok je vreme trajanja pranja filtera od 5 do 8 minuta.

$$V_R = \frac{Q_F}{60} \cdot t_{pranja} = \frac{157}{60} \cdot 5 = 13,08 \text{ [m}^3\text{]}$$

t_{pranja} – vreme ispiranja filtera [min]

Za ubacivanje obrađene vode u bazen ugrađuju se mlaznice u pod bazena. Potreban broj mlaznica određuje se prema površini bazena pri čemu jedna mlaznica pokriva oko 8 m² površine javnih (plivačkih) bazena.

$$N_{mlaznica} = \frac{A}{A_{mlaznica}} = \frac{312,5}{8} = 39,06 \approx 40$$

$N_{mlaznica}$ – broj potrebnih mlaznica; usvojeno 40 [-]

$A_{mlaznica}$ – površina bazena koju pokriva jedna mlaznica; 8 [m²]

Protok obrađene bazenske vode kroz pojedinu mlaznicu računa se prema izrazu:

$$Q_{mlaznice} = \frac{Q_F}{N_{mlaznica}} = \frac{157}{40} = 3,93 \approx 4 \text{ [m}^3/\text{h]}$$

$Q_{mlaznice}$ – protok vode kroz jednu mlaznicu [m³/h]

5. PRORAČUN POTREBNE ENERGIJE ZA ZAGREVANJE BAZENSKE VODE

Količina isparene vode u jedinici vremena računa se prema sledećoj formuli:

$$W = \frac{\varepsilon \cdot (p_s - p_d) \cdot A_{baz}}{3600} = \frac{20 \cdot (33,6 - 22,7) \cdot 312,5}{3600} = 0,01892$$

W – količina isparene vode [kg/s]

ε – koeficijent isparavanja – 20 [kg/m²h] – za vodu sa umerenim pokretanjem [3]

p_s – pritisak zasićenja vodene pare – 33,6 [hPa] [2]

p_d – parcijalni pritisak vodene pare – 22,7 [hPa] [2]

A_{baz} – površina bazena [m²]

Nakon određivanja količine isparene vode u jedinici vremena, potrebno je izračunati toplotni tok odveden isparavanjem preko formule:

$$\phi_\sigma = W \cdot r = 0,01892 \cdot 2439 = 46,15 \text{ [kW]}$$

ϕ_σ – toplotni tok odveden isparavanjem [kW]

r – toplota potrebna za isparavanje vode pri temperaturi graničnog sloja vode - 2439 [kJ/kg] [2]

Toplotni tok odveden transmisijom računa se preko sledeće formule:

$$\phi_T = k \cdot A_{kor} \cdot (\vartheta_w - \vartheta_p) = 2,8$$

ϕ_T – toplotni tok odveden transmisijom toplote [kW]

k – koeficijent prolaza toplote [W/m²K]

A_{kor} – površina korita bazena [m²]

ϑ_w – temperatura bazenske vode [°C]

ϑ_p – temperatura prostorije ispod bazena (pretpostavka) [°C]

Potrebna količina toplote koju je potrebno dovesti svežoj vodi da se zagreje na temperaturu bazenske vode preko formule:

$$\phi_{SV} = m \cdot c_w \cdot (\vartheta_w - \vartheta_{SV}) = 20,65$$

ϕ_{SV} – potrebna količina toplote [kW]

m – protok sveže vode [kg/s]

c_w – specifični toplotni kapacitet vode – 4,187 [kJ/kgK] [2]

ϑ_w – temperatura bazenske vode [°C]

ϑ_{SV} – temperatura sveže vode [°C]

Kapacitet izmenjivača dimenzioniše se prema formuli:

$$\phi_b = V_b \cdot \rho \cdot c_w \cdot (\vartheta_b - \vartheta_w) = 8,68 \cdot 998,2 \cdot 4,187 \cdot (26 - 12) = 141,09$$

Φ_b – količina toplote potrebna za zagrevanje bazenske vode [kW]

ϑ_{gub} – zapreminski protok bazena – 8,68 [m³/h]

ϑ_{gub} – gustina vode – 998,2 [kg/m³]

ϑ_{gub} – toplotni kapacitet vode – 4,187 [kJ/kgK]

ϑ_b – temperatura bazenske vode - 26 [°C]

ϑ_w – temperatura vodovodne vode – 12 [°C]

Proračun solarnih kolektora za zagrevanje bazenske vode:

Ugrađeno je 80 pločastih solarnih kolektora za proračun pokrivanja toplotnih gubitaka bazenske vode što ekvivalentno površini od 188m² kolektora. U tablici 8.5 prikazana je isporučena energija od solarnih kolektora u sistem za zagrevanje bazenske vode.

Tablica 8.5 Isporučena energija solarnog sistema za zagrevanje bazenske vode

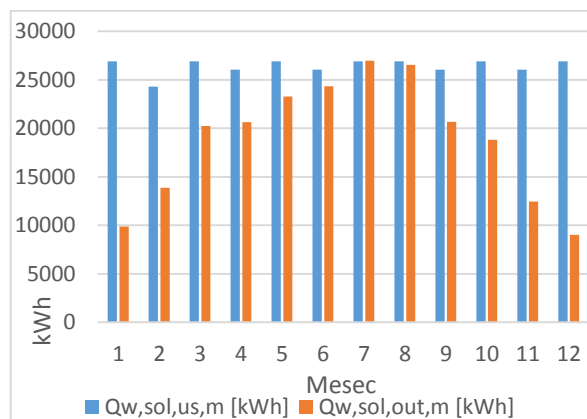
Mesec	Q _{w,sol,us,m} [kWh]	Q _{w,sol,out,m} [kWh]	f sol,m [-]
I	26915	9889	0,367
II	24311	13869	0,570
III	26915	20245	0,752
IV	26047	20658	0,793
V	26915	23288	0,865
VI	26047	24351	0,935
VII	26915	26968	1,002
VIII	26915	26563	0,987
IX	26047	20661	0,793
X	26915	18839	0,700
XI	26047	12456	0,478
XII	26915	9029	0,335

$Q_{w,sol,us,m} = Q_{gub,m}$ – potrebna energija za pripremu bazenske vode [kWh]

$Q_{w,sol,out,m}$ – isporučena energija solarnog kolektora u sistem [kWh]

$f_{sol,m}$ – stopa pokrivanja [-]

Na slici 5. vidljivo je da je dobijena energija od solarnih kolektora dovoljna da se pokriju potrebe zagrevanje bazenske vode za četiri letnja meseca, dok je van letnjeg perioda ostatak energije potrebno nadoknaditi preko toplotne podstanice.



Slika 8.5 - Potrebna i dobijna energija od solarnog sistema za zagrevanje bazenske vode

Minimalna zapremina ekspanzione posude solarnog kruga zagrevanja bazenske vode računa se prema izrazu:

$$V_{min} = (\Delta V + V_D + V_v) \cdot \frac{p_e + 1}{p_e - p_o} = (28,52 + 662,08 + 4,91) \cdot \frac{5,5 + 1}{5,5 - 1,5} = 1065,22$$

V_{min} – minimalna zapremina zatvorene ekspanzione posude [l]

ΔV – promena zapremine vode sa promenom temperature [l]

V_D – preuzimanje sadržaja pare iz kolektora = V_K = 622,08 l

V_v – dodatna zapremina, uzima se kao 0,5% ukupnog zapremina vode u sistemu = 12,1 [l]

p_e – krajnji projektni pritisak, uzima se 0,5 bara ispod pritiska otvaranja sigurnosnog ventila [bar]

p_o – primarni pritisak ekspanzione posude prilikom isporuke – posuda pre isporuke mora biti pod pritiskom od 1,5 bara zbog visine instalacije od 10 - 15 metara [bar]

6. DIMENZIONISANJE PODNOG GREJANJA

Sistem podnog grejanja postavljen je po celom prostoru bazenskog prostora. Dimenzionisanje sistema izvršeno je u programu INTEGRACad. Temperaturni režim podnog grejanja je 45/40 °C. Instalirani toplotni učinak zavisi od otpora podne obloge, razmaku polaganja cevi, temperaturnom režimu grejnog fluida i projektnoj temperaturi vazduha u prostoriji. Broj krugova grejanja se određuje tako da se poštuju ograničenja o dužini cevi od maksimalno 130 metara i padu pritiska kroz jedan grejni krug. Temperatura poda boravišne zone ne sme prelaziti 33°C.

7. PRORAČUN VAZDUŠNOG GREJANJA

Potreba za odsisavanjem vazduha iz prostora javlja se zbog isparavanja vode iz bazena. Kao što je navedeno kod toplotnih gubitaka isparavanjem kod bazena, maseni protok isparene vode iznosi 0,0189 kg/s. Po standardu VDI 2089 računa se potrebni protok vazduha za izbacivanje, prema sledećoj formuli:

Tablica 11.1 - Potreban protok vazduha za izbacivanje

Mesec	xvz [kg/kg]	Vz [m ³ /h]
1.	0,0033	7840
2.	0,0035	7946
3.	0,0044	8712
4.	0,0059	10321
5.	0,0081	14285
6.	0,0102	21851
7.	0,0112	30023
8.	0,0112	30023
9.	0,0091	16950
10.	0,0069	11810
11.	0,0049	9236
12.	0,0037	8163

8. ZAKLJUČAK

U radu su detaljno obrađeni proračuni za odabir adekvatne opreme za pravilno funkcionisanje bazena u higijenskom smislu i u smislu održavanja zadanih parametara termičke ugodnosti. Osim toga, u radu je obrađena problematika održavanja mikroklimatskih parametara u objektima ovog tipa gde je akcenat stavljen na problem odvlaživanja vazduha i održavanja konstantne vlažnosti vazduha unutar prostora.

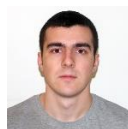
Za predmetni objekat predviđeni su panelni grejači (podno grejanje). Podno grejanje se od ostalih sistema razlikuje po tome što 1/3 toplote odaju konvekcijom, a 2/3 toplote odaju zračenjem kao i vazdušno grejanje preko klimakomore koja ujedno i odvlažuje vazduh.

Kao izvor toplote koriste se daljinsko grejanje i sunčevo zračenje. Kod daljinskog grejanja toplotna energija prenosi se distribucionim medijumom od toplane do podstanice umesto da se proizvodi u svakom objektu zasebno. Sunčevo zračenje ima za prednost to što je taj izvor energije neiscrpan i besplatan, ne zagađuje okolinu, nema potencijalnih opasnosti od korišćenja i koristi se za niskotemperaturne režime grejanja.

9. LITERATURA

- [1] ***, Standard DIN 19463-1 „Bazenska tehnika“
- [2] ***, Standard DIN 19463-2 „Bazenska tehnika“
- [3] Recknagel, Šprenger, Šramek, Čeperković, „Grejanje i klimatizacija“, Vrnjačka Banja, 2004

Kratka biografija:



Dejan Lazarević rođen je u Subotici 1994. god. Master rad na Fakultetu tehničkih nauka iz oblasti Mašinstva – Termoenergetika odbranio je 2021. god.

MOGUĆNOST KORIŠĆENJA KISELOG GASA IZ GASNOG LEŽIŠTA „NOVO MILOŠEVO“

ACID GAS FROM THE „NOVO MILOŠEVO“ GAS WELL USAGE POSSIBILITIES

Jovan Prole, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – Predmet rada je mogućnost korišćenja kiselog gasa iz gasnog nalazišta „Novo Miloševo“ kao gorivo za gasne turbine koje bi proizvodile električnu i toplotnu energiju, kao i proračun emisije ugljen-dioksida koji nastaje pri sagorevanju tog gasa i poređenje dobijenih emisija sa emisijama već postojećih sistema.

Ključne reči: kiseli gas, prirodni gas, gasne turbine, ugljen-dioksid

Abstract – This paper shows possible usage of acid gas from “Novo Miloševo” gas well as a fuel for gas turbines in order to produce electric and heat energy, and calculation process for CO₂ emissions calculation. Those emissions are then compared with the emissions from existing power plants.

Key words: acid gas, gas turbines, natural gas, carbon dioxide

1. UVOD

U radu je prikazana mogućnost korišćenja kiselog gasa u postrojenjima koja bi kao pogonsko gorivo koristilo kiseli gas iz gasnog nalazišta „Novo Miloševo“. Prvo postrojenje se sastoji iz gasne turbine GT OPRA OP16 koja kao gorivo koristi kiseli gas, dok se drugo postrojenje sastoji od gasne turbine PT SOLAR MERCURY 50 koja kao gorivo koristi namešani gas i parnog kotla koji kao gorivo koristi kiseli gas.

Cilj rada jeste da se pokaže uticaj postrojenja u kojima bi se koristio kiseli gas na životnu sredinu. U radu je procenjena količina ugljen-dioksida CO₂ specificirana po kWh isporučene energije koja bi emitovala iz postrojenja. Specificirana emisija upoređena je sa emisijama iz već postojećih sistema.

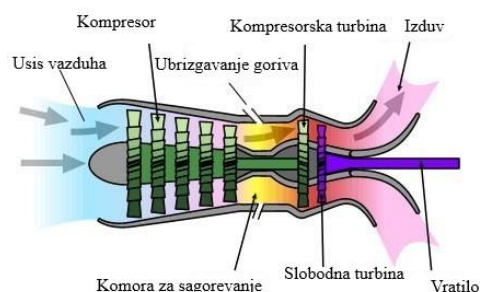
2. TEORIJSKE OSNOVE

Prvu gasnu turbinu patentirao je Džon Barber (John Barber) 1791. godine. Prva turbina zbog ograničenja u materijalima i tehnologijama izrade svog vremena nije mogla da ostvari pozitivan rad na vratilu [1]. Prvu gasnu turbinu koja je mogla da ostvari pozitivan rad patentirao je 1884. godine Džens Viliam Egidus Eling (Jens William Aegidus Elling) [2]. Prva gasna turbina koja je korišćena kao mlazni motor patentirana je 1930. godine od strane Frenka Vitla (Frank Whittle) [5]. Prva gasna turbina sa stepenom efikasnosti iznad 60 % napravljena je 2011. god [5].

NAPOMENA:

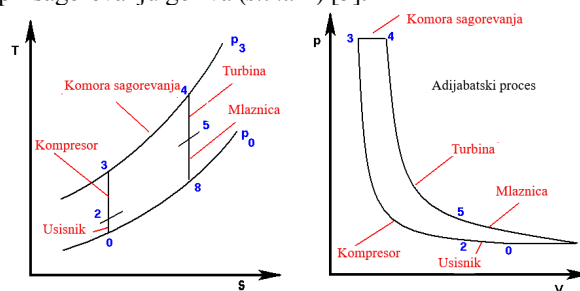
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Siniša Bikić, vanr. prof.

Gasna turbina predstavlja rotacioni motor koji se pokreće pomoću energije toka gasa proizvedenog sagorevanjem goriva. Svaka gasna turbina se sastoji iz tri osnovna dela. Na prednjem delu se nalazi kompresor, nakon njega u telu turbine se nalazi komora za sagorevanje i posle komore za sagorevanje se nalazi i sama turbina. Pored ovih delova većina turbina još poseduje i usis svežeg vazduha koji se često hladi, vratilo koje prenosi mehaničku energiju, otvore na komori za sagorevanje za ubrizgavanje goriva i izdub vrelih gasova (slika 1) [4].



Slika 1. Osnovni delovi gasne turbine [4]

Gasna turbina radi na principu termodinamičkog ciklusa koji se naziva Brajtonov ciklus (Brayton cycle). Tok ciklusa je sledeći: atmosferski vazduh se komprimuje, nakon toga komprimovani vazduh ulazi u komoru za sagorevanje gde se meša sa gorivom i gde dolazi do paljenja nastale smeše i zatim vreli gasovi prolaze kroz proces ekspanzije u turbini, gde fluid oslobađa više energije nego što je potrebno za komprimovanje gasa, zahvaljujući energiji koja se dobija pri sagorevanju goriva (slika 2) [5].



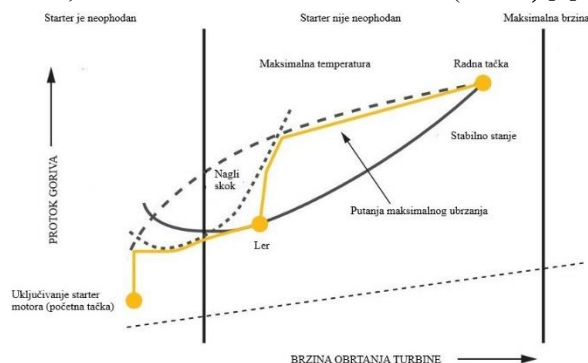
Slika 2. Idealan Brajtonov ciklus [6]

Gasne turbine mogu da imaju otvoren ili zatvoren ciklus. Razlika između ova dva ciklusa se ogleda u rešenju komore za sagorevanje. Kod turbine sa otvorenim ciklusom radni fluid i gorivo se mešaju u komori, dok kod turbine sa zatvorenim ciklusom to nije slučaj. Prednost turbine sa zatvorenim ciklusom jeste ta što je moguće lako kontrolisati kvalitet vazduha i što izbor goriva nije ograničen, takođe lopatice ne dolaze u direktan kontakt sa vrelim izduvnim

gasovima i samim tim mogu biti izrađene od materijala lošijih karakteristika u odnosu na uobičajene materijale. Međutim upravo zbog te niže temperature fluida na lopaticama stepen korisnosti ovih turbina je niži u odnosu na turbine sa otvorenim ciklusom i one se implementiraju samo ukoliko u neposrednoj blizini postoji mogućnost jeftinog eksploataisanja goriva [1].

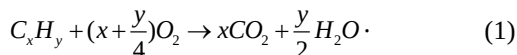
Prilikom rada gasne turbine energija koju gorivo sagorevanjem preda ne može u potpunosti da se pretvori u mehanički rad. Najveći deo energije napušta sistem u vidu toplotne energije. Iz tog razloga se gasne turbine najčešće vezuju u kogenerativni sistem gde se ta toplota, koja bi inače bila puštena u atmosferu iskoristi, pa se samim tim povećava i ukupan stepen efikasnosti sistema, jer za istu ulaznu energiju goriva dobija korisna mehanička energija i korisna toplotna energija [9]. Prilikom kogeneracije stepen iskorišćenja energije ide i do 80 %, dok u sistemima gde kogeneracija nije implementirana stepen iskorišćenja uglavnom bude ispod 30 % [11].

Kako bi se gasna turbina startovala potrebno je započeti ciklus pražnjenja, kako bi se izbegla pojava zapaljivog gasa u izduvnom sistemu. Da bi se to postiglo najpre se uključuje starter motor, koji rotira vratilo turbine niskom brzinom u trajanju koje je prethodno definisano. Nakon završetka ciklusa pražnjenja starter motor se ubrzava i u komoru za sagorevanje se injektuje gorivo i pali se pilot plamen. Starter motor nastavlja da ubrzava sve dok turbina ne bude u stanju da samostalno funkcioniše, na ler brzini. Nakon što sistem dostigne ler brzinu, starter motor se gasi i može se početi sa postepenim opterećivanjem sistema, sve dok se ne dođe do radne tačke (slika 3) [5].



Slika 3. Startovanje turbine

Prilikom sagorevanja prirodnog gasa dolazi do emisije ugljen-dioksida CO_2 u atmosferu. Potrebna količina kiseonika za potpuno sagorevanje se može izračunati pomoću stehiometrijske jednačine sagorevanja:



Kako bi se odredila količina ugljen-dioksida koja se oslobodi u atmosferu prilikom sagorevanja prirodnog gasa neophodno je odrediti mase ugljen-dioksida CO_2 i ugljovodonika koji se sagoreva:

$$m = n \cdot M_r, \quad (2)$$

gde su:

m - masa [kg];

n - broj molova [mol] i

M_r - molarna masa [kg/mol].

Nakon što se odrede mase, neophodno je odrediti i zapreminu ugljovodonika na normalnim uslovima:

$$V = n \cdot V_m, \quad (3)$$

gde su:

V - zapremina [m^3] i

V_m - molarna zapremina [m^3/mol]

Zapreminska koncentracija ugljovodonika u prirodnom gasu glasi:

$$C_v = \frac{V_z}{V_s} \cdot 100 \% \quad (4)$$

gde su:

C_v - zapreminska koncentracija [%];

V_z - zadata zapremina ugljovodonika u prirodnom gasu [m^3] i

V_s - zapremina smeše prirodnog gasa [m^3].

Kada je poznata i zapremina, preko proporcije se može odrediti masa ugljen-dioksida CO_2 koja se oslobodi prilikom sagorevanja kiselog gasa:

$$V_z : x = V : m, \quad (5)$$

gde je:

x - tražena masa ugljen-dioksida [kg].

Prethodno izračunatoj masi ugljen-dioksida neophodno je dodati i masu ugljen-dioksida CO_2 koja se već nalazi u kiselom gasu pre sagorevanja. Kako bi se ta masa izračunala neophodno je prvo proračunati gustinu ugljen-dioksida CO_2 na normalnim uslovima:

$$\rho = \frac{p}{\frac{R_u}{M_r} \cdot T} \quad (6)$$

gde su:

ρ - gustina gasa [kg/m^3];

p - apsolutni pritisak pri normalnim uslovima [Pa];

R_u - univerzalna gasna konstanta [$\text{J}/\text{mol K}$] i

T - temperatura pri normalnim uslovima [K].

Nakon što je izračunata gustina ugljen-dioksida CO_2 neophodno je proračunati njegovu masu:

$$m = V \cdot \rho \quad (7)$$

Ovaj postupak se ponavlja za prva četiri ugljovodonika, jer viši ugljovodonici imaju veoma mali udeo u prirodnom gasu (ispod 0,2%) i ne utiču bitno na tačnost proračuna.

Kako bi se dobila ukupna masa ugljen-dioksida CO_2 koja se emituje prilikom rada sistema neophodno je sabrati masu ugljen-dioksida CO_2 koja se emituje prilikom sagorevanja prva četiri ugljovodonika, kao i masu ugljen-dioksida CO_2 koja je već postojala u gasu i koja se prilikom sagorevanja takođe emituje:

$$m_u(\text{CO}_2) = x_1 + x_2 + x_3 + x_4 + m(\text{CO}_2), \quad (8)$$

gde su:

$m_u(\text{CO}_2)$ - ukupna masa CO_2 koja se emituje prilikom sagorevanja 1 m^3 prirodnog gasa pri normalnim uslovima [kg/m^3];

x_1 - masa CO_2 koja se emituje prilikom sagorevanja metana iz 1 m^3 prirodnog gasa pri normalnim uslovima [kg/m^3];

x_2 - masa CO_2 koja se emituje prilikom sagorevanja etana iz 1 m^3 prirodnog gasa pri normalnim uslovima [kg/m^3];

x_3 - masa CO_2 koja se emituje prilikom sagorevanja propana iz 1 m^3 prirodnog gasa pri normalnim uslovima [kg/m^3];

x_4 - masa CO_2 koja se emituje prilikom sagorevanja butana iz 1 m^3 prirodnog gasa pri normalnim uslovima [kg/m^3] i

$m(\text{CO}_2)$ - masa CO_2 koja je bila prisutna u 1 m^3 prirodnog gasa pri normalnim uslovima koja se emituje prilikom sagorevanja [kg/m^3].

Kako bi se proračunala masa CO₂ koja se godišnje emituje potrebno je prvo proračunati koju količinu prirodnog gasa na normalnim uslovima sistem utroši tokom jednog časa:

$$G_h = \frac{G_D}{24}, \quad (9)$$

gde su:

G_h - satni konzum sistema [m³/h] i

G_D - dnevni konzum sistema [m³].

Nakon što se proračuna satni konzum, neophodno je proračunati i godišnji konzum:

$$G_g = G_h \cdot T_g, \quad (10)$$

gde su:

G_g - godišnji konzum sistema [m³] i

T_g - godišnji broj sati rada sistema [h].

Kada je poznat godišnji konzum sistema, ukupna masa CO₂ koja se godišnje emituje pri radu sistema je:

$$m_g(CO_2) = G_{gO} \cdot m_U(CO_2), \quad (11)$$

gde je:

$m_g(CO_2)$ - godišnja emisija CO₂ u sistemu [kg/ m³].

Ukupna korisna energija koju predaje predloženo postrojenje u zimskom režimu glasi:

$$E_Z = E_{TO} + E_{EO} + E_{ESK} + E_{TSKZ}, \quad (12)$$

gde su:

E_Z - ukupna količina energije koju postrojenje predaje u zimskom režimu [kWh];

E_{TO} - toplotna energija koju predaje gasna turbina OPRA OP16 [kWh];

E_{EO} - električna energija koja se dobija radom gasne turbine OPRA OP16 [kWh];

E_{ESK} - električna energija koja se dobija radom sistem sa gasnom turbinom PT SOLAR MERCURY 50 i parnim kotlom [kWh] i

E_{TSKZ} - toplotna energija koju predaje sistem sa gasnom turbinom PT SOLAR MERCURY 50 i parnim kotlom u zimskom režimu [kWh].

Ukupna korisna energija koju predaje predloženo postrojenje u letnjem režimu glasi:

$$E_L = E_{TO} + E_{EO} + E_{ESK} + E_{TSKL}, \quad (13)$$

E_L - ukupna količina energije koju postrojenje predaje u letnjem režimu [kWh] i

E_{TSKL} - toplotna energija koju predaje sistem sa gasnom turbinom PT SOLAR MERCURY 50 i parnim kotlom u letnjem režimu [kWh].

Ukupna količina CO₂ koja se emituje prilikom rada postrojenja tokom jednog sata ustaljenog režima je:

$$m_h(CO_2) = m_{hO}(CO_2) + m_{hM}(CO_2) + m_{hK}(CO_2) \quad (14)$$

gde su:

$m_h(CO_2)$ - masa CO₂ koju emituje postrojenje za jedan sat rada [kg];

$m_{hO}(CO_2)$ - masa CO₂ koju emituje gasna turbina OPRA OP16 za jedan sat rada [kg];

$m_{hK}(CO_2)$ - masa CO₂ koju emituje parni kotao za jedan sat rada [kg] i

$m_{hS}(CO_2)$ - masa CO₂ koju emituje gasna turbina PT SOLAR MERCURY 50 za jedan sat rada [kg].

Masa ugljen-dioksida koja se emituje tokom jednog sata prilikom rada sistema se računa na sledeći način:

$$m_h(CO_2) = G_{hO} \cdot m_{UO} + G_{hK} \cdot m_{UO} + G_{hS} \cdot m_{US}. \quad (15)$$

Specifična emisija CO₂ u zimskom režimu rada postrojenja proračunava se pomoću sledećeg izraza:

$$m_{ZS}(CO_2) = \frac{m_h(CO_2)}{E_Z}, \quad (16)$$

gde je m_{ZS} - specifična emisija CO₂ u zimskom režimu rada [kg/kWh].

Specifična emisija CO₂ u letnjem režimu rada sistema proračunava se pomoću sledećeg izraza:

$$m_{LS}(CO_2) = \frac{m_h(CO_2)}{E_L}, \quad (17)$$

gde je m_{LS} - specifična emisija CO₂ u zimskom režimu rada [kg/kWh].

Specifična emisija CO₂ u postojećim postrojenjima:

$$e_{PS} = \frac{e_{EA1} + e_{EA2} + e_{EA4} + e_{EA6} + e_{EB1} + e_{EB2}}{7} + e_T, \quad (18)$$

gde su:

e_{PS} - prosečna specifična emisija CO₂ postojećih postrojenja [kg/kWh];

e_{EA1} - specifična emisija CO₂ bloka A1 TE „Nikola Tesla“ [kg/kWh];

e_{EA2} - specifična emisija CO₂ bloka A2 TE „Nikola Tesla“ [kg/kWh];

e_{EA4} - specifična emisija CO₂ bloka A4 TE „Nikola Tesla“ [kg/kWh];

e_{EA6} - specifična emisija CO₂ bloka A6 TE „Nikola Tesla“ [kg/kWh];

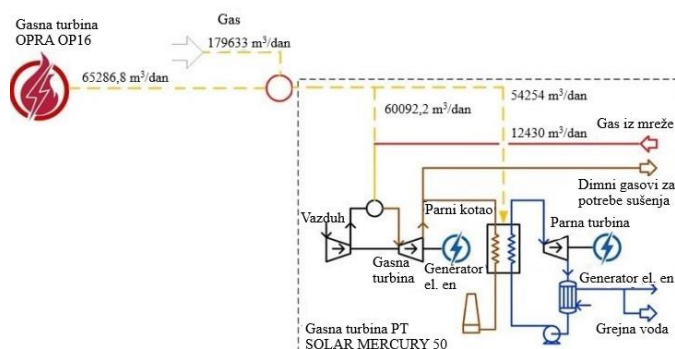
e_{EB1} - specifična emisija CO₂ bloka B1 TE „Nikola Tesla“ [kg/kWh];

e_{EB2} - specifična emisija CO₂ bloka B2 TE „Nikola Tesla“ [kg/kWh] i

e_T - Prosečna specifična emisija CO₂ toplanama u Srbiji [kg/kWh].

3. PRIKAZ REZULTATA

Šema predloženog postrojenja data je na slici 4.



Slika 4. Šema postrojenja

Sastavi kiselog i namešanog gasa prikazani su u tabeli 1 i tabeli 3. Karakteristike sistema sa gasnom turbinom OPRA OP16 i gasnom turbinom PT SOLAR MERCURY 50 date su u tabeli 2 i tabeli 4.

Tabela 1. Zapreminski sastav kiselog gasa

Azot (N ₂)	7,36 %
Ugljen-dioksid (CO ₂)	64,68 %
Metan (CH ₄)	26,73 %
Etan (C ₂ H ₆)	0,54 %
Propan (C ₃ H ₈)	0,13 %
Butan (C ₄ H ₁₀)	0,2 %

Tabela 2. Tehničke karakteristike postrojenja sa GT OPRA OP16

Potrošnja gasa	65 286,8 m ³ /dan
Toplotni konzum	3 200 kW
Električna sanga	1 913 kW

Tabela 3. Zapreminski sastav namešanog gasa

Azot (N ₂)	6,27 %
Ugljen-dioksid (CO ₂)	53,76 %
Metan (CH ₄)	38,63 %
Etan (C ₂ H ₆)	0,72 %
Propan (C ₃ H ₈)	0,2 %
Butan (C ₄ H ₁₀)	0,18 %

Tabela 4. Tehničke karakteristike postrojenja sa PT SOLAR MERCURY 50

Potrošnja namešanog gasa u turbini	75 522,2 m ³ /dan
Potrošnja kiselog gasa u kotlu	54 254 m ³ /dan
Električna snaga gasne turbine	4 845 kW
Električna sanga parnog kotla	3 776 kW
Isporuca grejne vode zimi	6 500 kW
Korisna toplotna snaga leti	2 600 kW

U tabeli 5 date su vrednosti specifičnih emisija već postojećih energetske sistema u Republici Srbiji [12], [13]:

Tabela 5. Specifične emisije CO₂ u postojećim sistemima

Postrojenje	Spec. emisija CO ₂
TE "Nikola Tesla"	1,143 kg/kWh
Toplane u gradovima Srbije	0,235 kg/kWh
Ukupna specifična emisija	1,378 kg/kWh

U tabeli 6 prikazane su vrednosti specifične emisije ugljen-dioksida CO₂ predloženog postrojenja dobijene računskim putem.

Tabela 6. Vrednosti specifične emisije CO₂ predloženog postrojenja

Specifična emisija sistema zimi	0,6483 kg/kWh
Specifična emisija sistema leti	0,9063 kg/kWh

4. ZAKLJUČAK

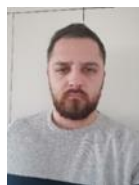
Predmet rada je bio proračun emisije ugljen-dioksida CO₂ koji nastaje sagorjevanjem kiselog i namešanog gasa u predloženom postrojenju sa gasnim turbinama i parnim kotlom. Cilj rada je bio da se pokaže uticaj postrojenja u kojem bi se koristio kiseli gas na životnu sredinu. Planirana je izgradnja postrojenja koji se sastoji iz dve celine na nalazištu kiselog prirodnog gasa „Novo Miloševo“ koji nije isplativo prečišćavati kako bi se dostigao zadovoljavajući kvalitet da se taj gas pusti u gasnu mrežu. Na osnovnu dobijenih vrednosti vidi se da bi izgradnja ovog postrojenja bila ekološki opravdana, s obzirom da bi sistem u poređenju sa već postojećim sistemima koji su u ovom radu uzeti u obzir u letnjem režimu emitovao 35 % manju količinu CO₂ u atmosferu, dok bi u zimskom režimu rada emisija bila čak 53 % niža. U budućnosti da bi se izgradnja postrojenja realizovala neophodno bi bilo tačno definisati mogućnosti upotrebe dobijene toplotne energije. S obzirom da sistem radi u

letnjem i zimskom režimu i da je specifična emisija tokom rada u letnjem režimu gotovo 40 % veća u odnosu na specifičnu emisiju tokom rada u zimskom režimu trebalo bi razmotriti mogućnosti trigeneracije i mogućnost plasmana dobijene rashladne energije kako bi se povećao stepen iskorišćenja predloženog sistema i kako bi se smanjila specifična emisija CO₂ u letnjem režimu rada. Implementacija ove tehnologije bi takođe pozitivno uticala na prost period otplate predloženog sistema.

5. LITERATURA

- [1] Milanović, Z. (2010). „Energetske mašine – Termodinamičke i strujne osnove toplotnih turbomašina“, Univerzitet u Banja Luci ISBN: 978-99938-39-27-9.
- [2] Radosavljević, D, Aćimović, D i Šotra, V (2017). „Istorijat klasifikacija i primena gasnih turbina“, 10th International Scientific Conference “Science and Higher Education in Function of Sustainable Development”, 06 – 07 October 2017, Mečavnik – Drvengrad, Užice, Serbia.
- [3] Brun, K. and Kurz, R. (2019). “Introduction to Gas Turbine Theory”, 3rd Edition, Solar Turbines Incorporated, ISBN: 978-0-578-48386-3
- [4] <https://www.ipieca.org/resources/energy-efficiency-solutions/power-and-heat-generation/open-cycle-gas-turbines/> 14.09.2021.
- [5] <https://www.grc.nasa.gov/www/k-12/airplane/brayton.html> 15.09.2021.;
- [6] Krawczyk, D, A. (2019). Buildings 2020+ - Energy sources, Printing House of Bialystok University of Technology, ISBN 978-83-65596-72-7, <https://doi.org/10.24427/978-83-65596-73-4>
- [7] Kuiken, K. (2016) "Gas and dual fuel engines 1", Target Training Centre, ISBN-10 : 9079104043, ISBN-13 : 978-9079104048.
- [8] Marković, Z., Erić, M., Cvetinović, D., Stefanović, P., Spasojević, V., Škobalj, P. (2015) Proračun specifične emisije ugljen dioksida iz termoelektrana Nikola Tesla A i B, Termotehnika, DOI: <https://doi.org/10.5937/terteh42-11676>.
- [9] Stefanović, P.Q., Marković, Z.J., Bakić, V.V., Cvetinović, D.B., Turanjanin, V.M., Jovanović, M.P. (2011) “Emisija gasova sa efektom staklene baste u toplanama javnih preduzeća daljinskog grejanja u gradovima Srbije”, Termotehnika, 37(2), 183-195, UDK: 504.7:662.71/.74.

Kratka biografija:



Jovan Prole rođen je 16.01.1997. godine u Novom Sadu, Republika Srbija. Završio gimn. „Jovan Jovanović Zmaj“ i nakon toga diplomirao na Fakultetu Tehničkih Nauka na departmanu za energetiku i procesnu tehniku 2020. godine u Novom Sadu sa temom „Gasni motori“.

EKSPERIMENTALNO ISPITIVANJE OPTIMALNOG RADA GASNOG GORIONIKA
EXPERIMENTAL TESTING OF OPTIMAL GAS BURNER OPERATION

Vladan Mitrović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – MAŠINSTVO

Kratak sadržaj – Tema ovog rada jeste sagorevanje prirodnog gasa u gorioniku postavljenom na laboratorijsku komoru za sagorevanje. Cilj rada je ispitivanje uticaja koeficijenta viška vazduha, kao i poziciju deflektora na kvalitet sagorevanja (npr. sastav produkata sagorevanja), kao i analiza dobijenih rezultata.

Ključne reči: Sagorevanje, Gasni gorionik, Energetska efikasnost, Produkti sagorevanja

Abstract – The subject of the paper is natural gas combustion in a gas burner mounted on a laboratory combustion chamber. The aim of this paper is to investigate the effect of excess air coefficient, as well as deflector position on combustion quality (i.e. the contents of the products of combustion), and also to analyze the results.

Keywords: Combustion, Gas burner, Energy efficiency, Combustion products

1. UVOD

Energija je oduvek bila jedna od osnovnih civilizacijskih potreba čovečanstva. Od samog svog postanka, čovekov razvojni put je neodvojiv od pronalaska vatre. U prošlosti je to podrazumevalo vatru za grejanje i pripremu hrane. Međutim, u savremenom dobu, uslovi za život su zavisni od niza tipova energije, i život bez njih bi bio nezamisliv. Jedan od osnovnih faktora životnog standarda upravo jeste pristup centralizovanom sistemu električne i toplotne energije.

Druga polovina dvadesetog veka je okarakterisana neprikosnovenom upotrebom tečnih goriva u skoro svim sferama privrede, što je znalo da dovede do čestih ekonomskih i političkih problema, ukoliko je dolazilo do bilo kakve destabilizacije tržišta nafte u svetu. Kao posledica, većine država, pogotovo zemlje koje većinski uvoze energente su težile da zamene upotrebu tečnih goriva sa što više alternativnih izvora.

Bitno je uzeti u obzir i činjenicu da su ekološki standardi sve strožiji širom sveta. Iz navedenih razloga je na primer u Evropi trend da se sa uglja i nafte sve više prelazi na gasovita goriva. Gasovita goriva generalno obezbeđuju kvalitetnije sagorevanje u odnosu na druge vrste, sa većim stepenom efikasnosti, kao i najmanjim udelom štetnih produkata sagorevanja u odnosu na oslobođenu količinu energije.

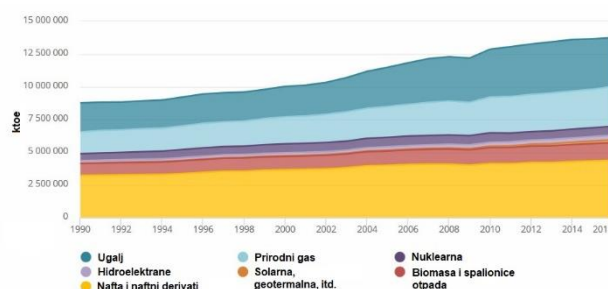
NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bila dr Biljana Miljković, vanr. prof.

2. POTROŠNJA ENERGIJE U SVETU

Naredna slika 1. prikazuje ukupnu svetsku potrošnju primarne energije u periodu od 1990-2016. godine [1]. Sa slike se može uočiti trend porasta za sve izvore, uključujući i prirodni gas. Jedini momenat kada je potrošnja opala jeste 2008. godine, usled ekonomske krize, nakon čega se ubrzo oporavila. Iako se potrošnja svih vidova energetske resursa povećava, prirodni gas je jedini izvor energije čiji se udeo udvostručio od 1990. Razlog za to jesu velike rezerve i visoki kvalitet prirodnog gasa kao energenta, ali prevashodno njegova niska cena u tom periodu.

Veliki udeo tog porasta je usled prelaska sa tečnih na gasna goriva. Preorijentacija ložišta je jednostavan tehnički problem, a investicioni zahtevi nisu značajni u poređenju sa prednostima primene gasovitih goriva.

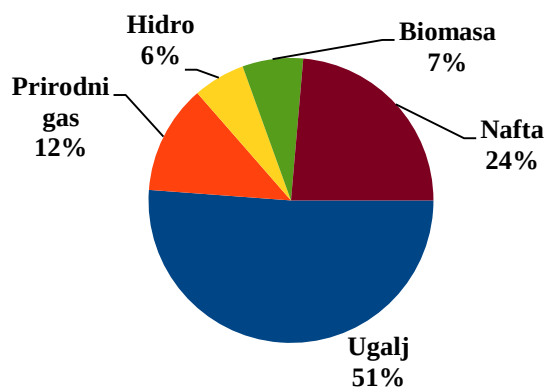


Slika 1. Ukupna potrošnja primarne energije u svetu između 1990-2016. godine [1]

Značajan porast beleže i obnovljivi izvori energije, međutim oni su i dalje iznosili svega 5% ukupne proizvodnje primarne energije, dok je udeo gasa 24% sa trendom rasta. Veliki nedostatak obnovljivih izvora energije, pogotovo kao izvora električne energije, jeste to što količina proizvodnje zavisi od spoljnih faktora kao što su oblačnost za solarnu, prisutnost vetra za vetrogeneratore, itd. To dovodi do situacija gde se vrhunac proizvodnje u toku dana za određene izvore javlja u momentima kada je ukupna potrošnja struje niska. Iz tog razloga je potrebno posedovati rezervne izvore čije se snage lako regulišu (najčešće elektrane na prirodni gas) da bi električna mreža ostala stabilna.

2.1. Potrošnja gasovitih goriva u Srbiji

Ukupna potrošnja primarne energije u Srbiji je iznosila 15440 ktoe u 2016. godini [2]. Ova energija je dobijena iz domaće proizvodnje i uvoza. Udeo korišćenja prirodnog gasa u proizvodnji primarne energije je iznosio 12%, što se nije menjalo nekoliko godina. Razlozi za ovakvo stanje su uglavnom niska cena i velike rezerve lignita u Srbiji, za razliku od prirodnog gasa koji se uglavnom uvozi.



Slika 2. Ukupni udeo izvora primarne energije u Srbiji 2016. godine [2]

3. SAGOREVANJE GASOVITOG GORIVA

Sagorevanje se definiše kao hemijski proces vezivanja gorivih elemenata sa kiseonikom uz intenzivno oslobađanje toplote. To je poseban vid hemijske reakcije koji se dešava u gasnoj fazi [3].

Samim tim, sagorevanje je složen proces koji zavisi od sledećih parametara [4]:

- sastav gasne smeše,
- brzina strujanja gasne smeše u raznim procesima gorionika,
- homogenost mešanja gorive i oksidirajuće komponente,
- početna temperatura goriva i vazduha,
- prisustvo inertnih komponenti u gasnoj smeši,
- način paljenja gasne smeše, itd.

4. GASNI GORIONICI

Gasni gorionici su uređaji koji dovode gas i vazduh do mesta gde se odvija sagorevanje. Dizajnirani su tako da obezbede stabilan plamen, kao i da obezbede laku regulaciju. Još neki od njihovih ciljeva su što racionalnija potrošnja goriva i smanjenje emisije štetnih produkata sagorevanja.

4.1. Gorionici sa prinudnim dovodenjem vazduha

Kod ove vrste gorionika se prinudno dovodi vazduh potreban za potpuno sagorevanje goriva uz pomoć nekog uređaja, najčešće ventilatora ili kompresora što daje mogućnost menjanja koeficijenta viška vazduha, λ . Gasovi se mogu mešati pomoću difuzije u samom cevovodu ili komori za predmešanje, ili uz pomoć deflektora koji se postavljaju u cevovod čija je svrha da pospešuju turbulenciju. Protok vazduha se može regulisati menjanjem brzine obrtanja uređaja za dovod vazduha, dok se protok gasa reguliše uz pomoć ventila.

5. EKSPERIMENTALNA INSTALACIJA ZA ISPITIVANJE SAGOREVANJA GASA

Ispitivanje je vršeno u okviru Erasmus KA1+ mobilnosti na Univerzitetu Klod Bernar 1 u Lionu, u Francuskoj, u periodu od januara do juna 2019. godine.

Eksperimentalna instalacija za ispitivanje sagorevanja se nalazi u Laboratoriji za toplotnu tehniku Departmana za termiku u gradu Burg an Bres nadomak Liona. Laboratorija se bavi obukom studenata univerziteta, kao i

ispitivanjem u komercijalne svrhe. Merenje je sprovedeno pod mentorstvom vanrednog profesora Erika Albina.

Laboratorija je dizajnirana tako da se održava konstantna temperatura vazduha, kao i prirodnog gasa od 25°C i obezbeđena je adekvatna ventilacija da ne bi došlo do kontaminacije vazduha sa ugljen-monoksidom.

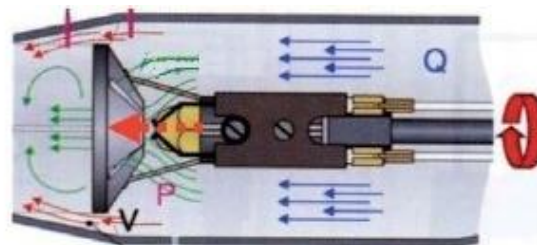
Postrojenje za testiranje se sastoji iz izolovane komore u kojoj se nalazi gorionik, nazivne toplotne snage 50 kW. Cevi kroz koju protiče voda poznatog protoka su direktno izložene plamenu gorionika. Temperatura vode se meri na ulazu i izlazu iz komore. Voda se zatim hladi kroz razmenjivač toplote, čija se toplota odvodi vazdušnim rashladnim uređajem, van laboratorije. Komora za sagorevanje sadrži staklene vizire kroz koje se može posmatrati boja i oblik plamena. Fotografija postrojenja je data na slici 3.



Slika 3. Eksperimentalna instalacija za ispitivanje sagorevanja

5.1. Opis gorionika

Gasni gorionik koji se koristio prilikom ovog istraživanja je gasni gorionik firme "Riello", model BS1 tip 911 T1, toplotne snage od 16 do 52 kW. Sa promenom uslova se direktno utiče i na kvalitet sagorevanja, kao i na promenu koncentracije CO i CO₂. To je moguće učiniti i podešavanjem mešanja pomoću deflektora, prikazanom na slici 4. Deflektor se sastoji iz pločice sa zarezima i podesivog zapornog elementa. Prilikom podešavanja zavrtnja se može povećavati ili smanjivati površina poprečnog preseka, što povećava ili smanjuje brzinu strujanja, kao i koeficijent lokalnog otpora.



Slika 4. Princip podešavanja mešanja pomoću deflektora

5.2. Merna oprema

Gasni analizator proizvođača „Ecom” model „EN2” sa slike 5. se koristi za merenje svojstva produkata sagorevanja. Na izlazu iz komore za sagorevanje se postavlja sonda gasnog analizatora, koja precizno meri temperature produkata sagorevanja, zapreminski udeo O₂, zapreminski

udeo CO_2 , zapreminski udeo CO , zapreminski udeo NO_x i koeficijent viška vazduha.



Slika 5. Gasni analizator

Radi verifikacije rezultata, koristio se gasni analizator proizvođača „Testo“ model „327-2“ koji meri zapreminske udele O_2 , CO_2 i CO .

6. REZULTATI ISPITIVANJA

Proces sagorevanja gasa u nekoj komori za sagorevanje, sasvim izvesno zavisi od vrste goriva, njegovog kvaliteta, geometrije ložišta, ali u najvećoj meri od uslova koji vladaju u ložištu. Tu se prvenstveno misli na koeficijent viška vazduha, temperature ložišta, kao i strujanje u samom ložištu. U cilju ispitivanja nekih od navedenih parametara na proces sagorevanja postavljeno je i eksperimentalno postrojenje, opisano u petom poglavlju. Zadatak mog master rada je bio da ispitam uticaj koeficijenta viška vazduha, kao i položaj deflektora (tj. stepen mešanja) na kvalitet sagorevanja. U prikazanom eksperimentalnom postrojenju glavni pokazatelji kvaliteta sagorevanja svakako su sastav produkata sagorevanja, tako da se posebna pažnja posvetila analizi upravo ovih rezultata.

6.1. Provera merne opreme

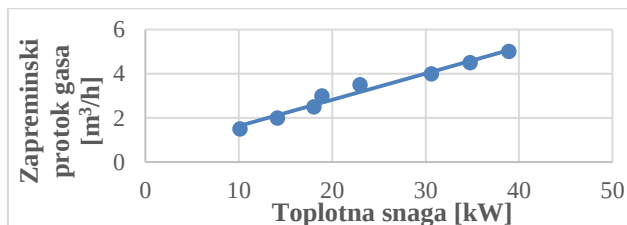
U inženjerskoj praksi je očekivano da se svako merenje ponovi nekoliko puta pri istim uslovima radi provere ispravnosti merenja i mernog uređaja. Nažalost, to nije bilo moguće uraditi zbog vremenskih ograničenja te je urađeno samo nekoliko verifikacionih merenja koja su imala za cilj proveru konzistentnosti izmerenih rezultata. Jedna od tih provera podrazumeva istovremeno merenje sastava produkata sagorevanja pomoću dva različita analizatora gasova opisanih u 5. poglavlju.

Prilikom ove provere utvrđeno je da merenje analizatora „Ecom“ nije adekvatno, s obzirom da je analizator „Testo“ baždaren nekoliko meseci pre merenja, a dolazilo je do značajnih odstupanja u rezultatima.

Da bi se rešio ovaj problem, uređaj „Ecom“ je poslat u sertifikovanu laboratoriju na baždarenje, a nakon toga se merenje ponovo izvelo. Nakon baždarenja su razlike u rezultatima merenja bile veoma niske.

6.2. Eksperimentalni rezultati

Prilikom merenja, protok vazduha je održavan na konstantnih $50 \text{ m}^3/\text{h}$ što se pokazalo kao pogodno za uspešno paljenje uređaja u najširem opsegu vrednosti protoka prirodnog gasa. Kao rezultat proračuna bilansa energije za stvarne uslove sagorevanja datog u prilogu, zavisnost toplotne snage od zapreminskog protoka gasa je prikazana na slici 6.



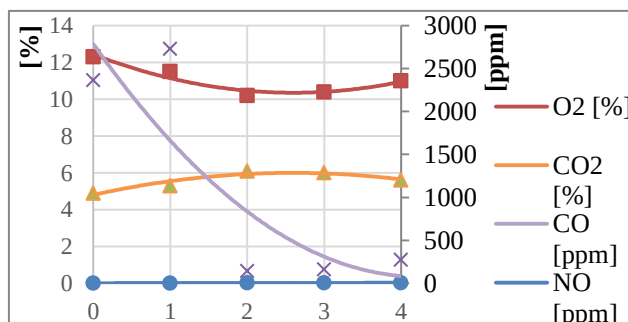
Slika 6. Zavisnost toplotne snage od protoka prirodnog gasa

Rezultat proračuna potvrđuje da je komora za sagorevanje adekvatna za postizanje opsega toplotnih snaga gorionika koje su navedene u proizvođačkoj specifikaciji. Međutim, kvalitet sagorevanja varira te je cilj dalje analize određivanje optimalnih radnih uslova za ovo postrojenje.

6.3. Emisija CO prilikom sagorevanja

Na slici 7. prikazani su udeli CO , CO_2 , NO_x i O_2 na izlazu iz ložišta. U svim slučajevima protok vazduha je održavan na konstantnih $50 \text{ m}^3/\text{h}$, dok se protok gasa menjao od $1,5 \text{ m}^3/\text{h}$ do $5 \text{ m}^3/\text{h}$. Na taj način se praktično varirao koeficijent viška vazduha.

Ukoliko se protok prirodnog gasa smanjio ispod $1,5 \text{ m}^3/\text{h}$ dolazilo je do potpunog gašenja plamena i automatskog isključenja gorionika. Za protoke veće od $5 \text{ m}^3/\text{h}$ i više, sagorevanje je postajalo nepotpuno i udeo CO je rastao značajno iznad opsega gasnog analizatora. Za svaki od ovih protoka gasa, merenje je rađeno za 5 položaja deflektora.

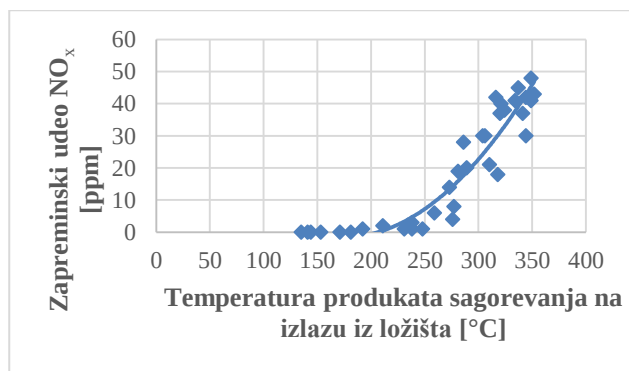


Slika 7. Zapreminski udeli produkata sagorevanja za protok gasa $2,5 \text{ m}^3/\text{h}$ i protok vazduha $50 \text{ m}^3/\text{h}$, nazivna snaga 18 kW

6.4. Emisija NO_x prilikom sagorevanja

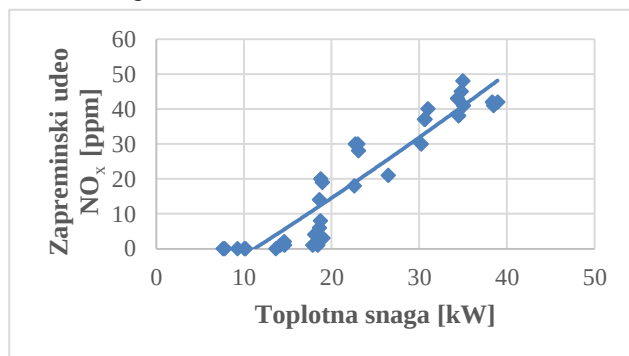
Glavni faktor koji utiče na količinu NO_x koji se formira je temperatura samog plamena. Međutim, pošto nije moguće meriti temperaturu u toj tački, posmatraće se temperatura produkata sagorevanja na izlazu iz komore. Na slici 8. je data zavisnost između temperature produkata sagorevanja na izlazu i zapreminskog udela NO_x . Pokazalo se da ne postoji uticaj deflektora na količinu formiranog NO_x .

Sa dijagrama se može zaključiti da se azotni oksidi formiraju tek nakon što temperatura gasova na izlazu iz ložišta dostigne vrednost od 200°C , a zatim se eksponencijalno povećava. Iz tog razloga bi bilo korisno težiti ka što nižim temperaturama sagorevanja sa stanovišta formiranja azotnih oksida.



Slika 8. Zavisnost udela NO_x od temperature gasova na izlazu iz ložišta

Da bi se bolje uporedila količina formiranja ugljen-monoksida i azotnih oksida korisno je prikazati uticaj promene toplotne snage na formiranje NO_x kao što je prikazano na slici 9. Na njoj se može uočiti linearna zavisnost između toplotne snage gorionika i količine oslobođenog NO_x .



Slika 9. Zavisnost udela NO_x od predatke količine toplote u radnu tečnost

7. ZAKLJUČAK

Osnovni cilj ovog istraživanja bio je da se pronađu optimalni uslovi sagorevanja na gasnom gorioniku „Riello“, model BS1 tip 911 T1, nazivne snage 50 kW, ugrađenom na ispitno postrojenje. U tom cilju, izvršena su merenja u kome se varirao protok goriva pri konstantnom protoku vazduha (čime se menjao koeficijent viška vazduha), kao i položaj deflektora.

Na osnovu eksperimentalnih rezultata, može se zaključiti da se minimalne vrednosti CO, kao glavnog pokazatelja nepotpunog sagorevanja, u produktima sagorevanja javljaju za nazivne snage od 18 do 34 kW (kojima odgovaraju koeficijenti viška vazduha $1,05 \div 1,8$).

Može se primetiti da je uticaj promene položaja deflektora jače izražen pri velikim protocima gasa jer u tim slučajevima intenzitet mešanja ima veći uticaj na kvalitet sagorevanja, a samim tim i na koncentraciju CO. Prilikom rada postrojenja u navedenom opsegu, udeo CO u produktima sagorevanja iznosi manje od 70 ppm, što je nivo koji se smatra bezopasnim po ljudsko zdravlje prilikom dužeg izlaganja [4]. Ovakvo sagorevanje se smatra potpunim.

NO_x se primarno formira kao rezultat javljanja visokih temperatura u komori za sagorevanje. Zbog nemogućnosti merenja temperature u samoj komori za sagorevanje, zapreminski udeo NO_x je doveden u vezu sa toplotnom snagom i izračunato je da počinje da se formira nakon što toplotna snaga pređe 10 kW i raste linearno sa porastom snage.

Na osnovu dobijenih rezultata, pokazalo se da se pri opterećenju od 36-68% javljaju male emisije štetnih produkata sagorevanja. Ipak, to su vrednosti sa dosta velikim koeficijentom viška vazduha, na šta ukazuje veliki udeo O_2 u produktima sagorevanja, a što za posledicu ima povećane gubitke ka okruženju (viši eksergetski nivo). Zbog toga se preporučuje rad uređaja što bliže gornjoj granici opterećenja od 68%, sa adekvatno podešenim mešanjem. Pri tim uslovima se sprečava formiranje velikih količina CO, a temperatura sagorevanja je i dalje dovoljno niska tako da se ne javlja velika količina NO_x .

8. LITERATURA

- [1] [www.iea.org/statistics/?country=WORLD&year=2016&category=Energy supply&indicator=TPESbySource&mode=chart&dataTable=BALANCES](http://www.iea.org/statistics/?country=WORLD&year=2016&category=Energy%20supply&indicator=TPESbySource&mode=chart&dataTable=BALANCES) International Energy Agency (IEA) (pristupljeno u junu 2019.)
- [2] www.iea.org/statistics/?country=SERBIA&year=2016&category=Energy%20supply&indicator=TPESbySource&mode=chart&dataTable=BALANCES (pristupljeno u junu 2019.)
- [3] Boško D. Ćosić, Eksperimentalno fotometrijsko ispitivanje strukture laminarnog plamena nadstehiometrijske smeše propan – butan – vazduh, Mašinski fakultet, Univerzitet u Beogradu, 2013.
- [4] Biljana Miljković, Zbirka rešenih zadataka iz sagorevanja, Fakultet tehničkih nauka, Univerzitet u Novom Sadu, 2018.

Kratka biografija:



Vladan Mitrović rođen je u Novom Sadu 1995. god. Master rad na Fakultetu tehničkih nauka iz oblasti Mašinstva – Energetike i procesne tehnike odbranio je 2021. godine, a od 2021. je u zvanju saradnik u nastavi. Oblast interesovanja su mu primenjena mehanika fluida, računarska dinamika fluida i hidropneumatski sistemi.

kontakt: vladanmit@uns.ac.rs

ОДРЕЂИВАЊЕ ОПТИМАЛНОГ БРОЈА И ЛОКАЦИЈЕ ДАЉИНСКИ КОНТРОЛИСАНИХ ПРЕКИДАЧА ЗА АУТОМАТИЗАЦИЈУ ДИСТРИБУТИВНИХ МРЕЖА**OPTIMAL PLACEMENT OF REMOTE-CONTROLLED SWITCHES IN AUTOMATED DISTRIBUTION NETWORKS**Биљана Татар, Невен Ковачки, *Факултет техничких наука, Нови Сад***Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО**

Кратак садржај – У овом раду представљена је метода заснована на симулираном каљењу за одређивање оптималног броја и локације даљински контролисаних прекидача за аутоматизацију дистрибутивних мрежа. Основни циљ представљене методе јесте унапређење поузданости дистрибутивних мрежа. Представљена метода верификована је на примеру дистрибутивне мреже која садржи 31 чвор, а резултати су приказани и коментарисани у раду.

Кључне речи: дистрибутивна мрежа, поузданост, аутоматизација мреже, даљински контролисани прекидачи, симулирано каљење.

Abstract – This paper presents a method for optimal placement of remote-controlled switches in automated distribution networks which is based on simulated annealing. The aim of the presented method is to improve reliability of distribution network. The presented method is verified on the distribution network that consists of 31 nodes and the results are presented and discussed in the paper.

Keywords: distribution network, reliability, network automation, remote-controlled switches, simulated annealing.

1. УВОД

Основна функција електроенергетског система (ЕЕС) је континуално напајање потрошача електричном енергијом одговарајућег квалитета. У пракси се показује да су најчешћи узрок прекида напајања потрошача кварови у дистрибутивним мрежама (ДМ). Разлози за то су њихова величина, као и опрема лошијег квалитета која се у њих уграђује. Једна од најнефективнијих мера за унапређење поузданости је аутоматизација ДМ, јер се на тај начин смањује како време трајања прекида напајања, тако и број корисника погођених кваром [1].

Одређивање оптималног броја и локације даљински контролисаних (аутоматизованих) прекидача за унапређење поузданости ДМ је комплексан оптимизациони проблем за чије решавање су током година дате различите методе и приступи. У овом раду је за ту сврху развијена метода на бази симулираног каљења, која спада у мета-хеуристичке оптимизационе методе. Као критеријумска функција узета је минимализација укупних трошкова током периода планирања, а који обухватају инвестиционе трошкове даљински контро-

лисаних прекидача и трошкове неиспоручене електричне енергије.

2. ПОУЗДАНОСТ ДИСТРИБУТИВНИХ МРЕЖА

Поузданост рада је својство система да обави задату функцију у задатом обиму при унапред дефинисаним условима рада [2].

2.1. Индекси поузданости

Постоје три основна индекса поузданости [1]:

- интензитет отказа – квара,
- трајања прекида напајања,
- расположивост електричне енергије.

Применом основних индекса није могуће стећи укупну слику о поузданости дистрибутивне мреже (ДМ), јер не узимају у обзир потрошњу електричне енергије и број потрошача који су остали без напајања. Због тога, постоји скуп додатних индекса који се могу користити при оцени поузданости мреже који могу бити оријентисани на [3]: снагу и енергију потрошача, број потрошача погођених кваром и економске ефекте. У овом раду у разматрање је узет индекс очекиване годишње неиспоручене енергије ENSI, који спада у индексе поузданости оријентисане на снагу и енергију потрошача.

2.2. Мере за унапређење поузданости

Поузданост ДМ се може повећати смањењем интензитета и броја кварова, што углавном захтева велике инвестиције у елементе мреже. Такође, поузданост ДМ се може повећати смањењем трајања прекида напајања, што се постиже ефикаснијом идентификацијом места квара, као и бржом променом уклопног стања мреже при изолацији квара и рестаурацији напајања [4]. Наведени циљеви се постижу управом увођењем даљински контролисаних прекидача. Трећи начин за повећање поузданости је смањење броја корисника погођених кваром, што се може постићи одговарајућим планирањем и конфигурацијом ДМ, као и аутоматизацијом ДМ.

2.3 Методологија за анализу поузданости

Алгоритам за прорачун годишње очекиване неиспоручене енергије се базира на прорачуну интензитета квара и трајању прекида напајања након квара. Код прорачуна интензитета кварова потребни подаци се односе на тип разматраног елемента, док је за прорачун трајања прекида напајања потребно располагати са више података: топологијом (тј. конфигурацијом) ДМ, размештајем расклопних уређаја у мрежи, као и временом потребним за приступање појединим деловима мреже [4].

НАПОМЕНА:

Овај чланак произтекао је из мастер рада чији ментор је био доцент др. Невен Ковачки

Алгоритам прорачуна очекиване годишње неиспоручене енергије састоји се из три корака [4]:

Корак 1 – Прорачун интензитета кварова грана разматране дистрибутивне мреже. За гране добијене од кабловских и ваздушних водова, као и за трансформаторске гране. Интензитети кварова (λ_i) су:

$$\lambda_i = \lambda_{i,pod} \cdot l_i, \quad i = 1, 2, \dots, n_{gr}, \quad (1)$$

где су $\lambda_{i,pod}$ подужни интензитет квара гране (i) [kvar/km], l_i дужина гране (i) (за трансформаторске гране $l = 1$ [km]) и n_{gr} укупан број грана разматране дистрибутивне мреже.

Корак 2 – Прорачун очекиваног годишњег трајања прекида напајања чворова услед кварова на гранама мреже:

$$\tau_j = \sum_{i \in \Delta j} \lambda_i \cdot T_{j,i}, \quad j = 1, 2, \dots, n_{cv} \quad (2)$$

где су τ_j очекивано годишње трајање прекида на напајању чвора j , $T_{j,i}$ трајање прекида напајања чвора j , услед квара гране i , Δj скуп индекса грана које се налазе на истом изводу као грана j и n_{cv} укупан број чворова разматране дистрибутивне мреже.

Трајање прекида напајања чвора j услед квара гране i ($T_{j,i}$) се израчунава на следећи начин:

$$T_{j,i} = (T_{j,i})_{loc} + (T_{j,i})_{izo} + (T_{j,i})_{obn}, \quad (3)$$

где су $(T_{j,i})_{loc}$ време трајања прекида напајања чвора j због налажења локације квара на грани i , $(T_{j,i})_{izo}$ време трајања прекида напајања чвора j због изолације квара на грани i , $(T_{j,i})_{obn}$ време трајања прекида напајања чвора j због обнављања напајања (рестаурације) када је квар на грани i .

Корак 3 - Прорачун индекса поузданости- очекиване годишње неиспоручене енергије помоћу следеће релације:

$$ENSI = \sum_{j=1}^{n_{cv}} P_j \cdot \tau_j, \quad (4)$$

где су P_j просечна годишња активна снага потрошње чворова (j), τ_j очекивано годишње трајање прекида напајања чвора (j) које је израчунато у претходном кораку.

3. АУТОМАТИЗАЦИЈА ДИСТРИБУТИВНИХ МРЕЖА

Аутоматизација дистрибутивне мреже (ДМ) је један од најбољих начина којим се може побољшати поузданост ДМ, јер се тиме смањује трајање прекида напајања, као и број потрошача погођених кваром.

Систем за аутоматизацију чине аутоматизовани расклопни уређаји, систем за надзор и прикупљање података у реалном времену (SCADA) и систем за управљање дистрибутивним мрежама (ДМС).

Расклопни уређаји је заједнички термин за опрему која се користи за прекидање и успостављање струјног кола струјног кола [1]. У овом раду, уређаји који су коришћени за унапређење поузданости су: прекидач, реклозер и склопка растављач.

Прекидач је уређај чији је главни задатак да успостави, проводи или прекине радну струју у нормалном режиму рада као и да прекине струју кратког споја.

Реклозер је заштитни уређај који има свој заштитни релеј и систем за аутоматско поновно укључење (АПУ).

Он утиче на смањење броја корисника погођених кваром као и на смањење времена трајања прекида напајања.

Склопка растављач је уређај који може да успостави, проводи и прекине радну струју али не и струју кратког споја. Има важну улогу при изолацији квара.

SCADA (eng *Supervisory Control and Data Acquisition*) је основа система за управљање у реалном времену и важна компонента система за аутоматизацију [1]. Она прикупља податке из система (мерења и сигнализација) што диспечеру омогућава да види стање у систему и на основу тога донесе одлуку о спровођењу управљачке команде. ДМС је софтвер намењен за управљање дистрибутивним мрежама. Овај софтвер обезбеђује многе погодности и пружа разне могућности и функције, од токова снага, естимације стања, анализе поузданости, управљање кваровима и других основних функција па све до напредних као што су реконфигурација мреже, аутоматизације, планирање и развој итд.

4. ОДРЕЂИВАЊЕ ОПТИМАЛНОГ БРОЈА И ЛОКАЦИЈЕ ДАЉИНСКИ КОНТРОЛИСАНИХ ПРЕКИДАЧА

Проблем одређивања оптималног броја и локације даљински контролисаних прекидача може се преставити следећом критеријумском функцијом:

$$\min \{ TC = c_{inv} + \sum_{t=1}^T \frac{1}{(1+d)^t} \cdot c_{ensi} \}, \quad (5)$$

где су c_{inv} инвестициони трошкови даљински контролисаних прекидача, c_{ensi} годишњи очекивани трошкови неиспоручене енергије, d интересна стопа и T период разматрања.

4.1. Оптимизационе методе

За решавање проблема оптималне аутоматизације дистрибутивне мреже могу се користити следеће методе [5]:

- математички засноване методе,
- хеуристички методе,
- мета-хеуристичке методе.

У математички методама највише су заступљене оне које су засноване на мешовитом целобројном линеарном програмирању (МИЛП). Ове методе гарантују глобални оптимум, међутим због своје комплексности и димензиалности нису практичне за примену у реалним дистрибутивним мрежама.

За разлику од њих, хеуристичке методе дају довољно добро решење за мреже реалних димензија уз коришћење мало рачунарских ресурса. Њихова мана је та што с обзиром да траже само локални оптимум зависе претежно од почетног решења, па не можемо знати да ли је то сигурно најоптималније решење.

Да би се унапредио квалитет оваквог решења, пре свега превазилажење проблема локалног оптимума, предложени су алгоритми који користе неку врсту стохастичког претраживања који су засновани на аналогiji са природним процесима и они се називају мета-хеуристички алгоритми [5]. Мета-хеуристичка метода која је изабрана у овом раду је симулирано каљење.

4.2. Метода симулираног каљења

На почетку алгоритма бира се почетно решење и поставља се температура на неку довољно велику вредност да би се обезбедило да сва решења буду прихватљива. Даље се сам процес каљења извршава итеративно. У свакој итерацији се бира ново суседно решење случајним избором и затим се проверава да ли је оно боље или горе од претходног. Уколико је боље оно се прихвата као ново текуће решење, а ако није оно се може прихватити са неком вероватноћом која зависи од контролног параметра T и деградације решења ΔE . На овај начин СК алгоритам дозвољава деградацију решења како би се изашло из локалног оптимума и како би се дошло у глобални оптимум. Вероватноћа да се прихвати лошије решење ће бити велика на великим температурама, а како температура пада ће бити све мања и мања јер би тада требало да се већ налази у околини оптималног решења (као што ће атоми заузимају положај само у стању са минималном енергијом кад је температура мала). Ова вероватноћа генерално прати Болтзманову расподелу [5]:

$$P(\Delta E, T) = e^{-\frac{f(s') - f(s)}{T}},$$

На свакој температури генерише се више суседних решења, а температура се смањује према дефинисаном програму хлађења када се генерише довољан број суседних решења. Алгоритам симулираног каљења представља генеричку методу за решавање било каквог оптимизационог проблема Модификован псеудо код СК коришћен у овом раду гласи:

Улаз: Програм (распоред) хлађења: /* Почетна температура (T_{max}), Крајња температура (T_{min}), Темпо (брзина) хлађења, Број итерација на одређеној температури */

$s = s_0$; /* Генерисање почетног допустивог решења. За почетно решење узима се ДМ без икаквих аутоматизованих уређаја */

$T_k = T_{max}$; /* Почетна температура */

$k = 1$;

Понављати /*Глобална итерација, k */

Понављати /*На одређеној температури T_k , односно у глобалној итерацији k */

Генерисање суседног (блиског) решења s' тако што се дода или склони неки аутоматизован уређај;

$\Delta E = f(s') - f(s)$; /* Разлика циљних функција, тј. остварених укупних трошкова из формуле 4.1 */

Ако је $E \leq 0$ **Онда** $s = s'$; /* Прихватити s' као текуће решење */

У супротном Прихватити s' са вероватноћом $e^{-\Delta E / T_k}$, односно прихватити s' ако важи:

$e^{-\Delta E / T_k} > rand[0,1]$;

Ако је $f(s') - f(s_{best})$ /* Прихватити s' као најбоље решење до сад (s_{best}) */

Док Услов заустављања /* Дефинисан је број итерација $n = 500$ на одређеној температури */

$T_{k+1} = \alpha \cdot T_k$ /* Темпо (брзина) хлађења, где је $\alpha = 0,9$ */

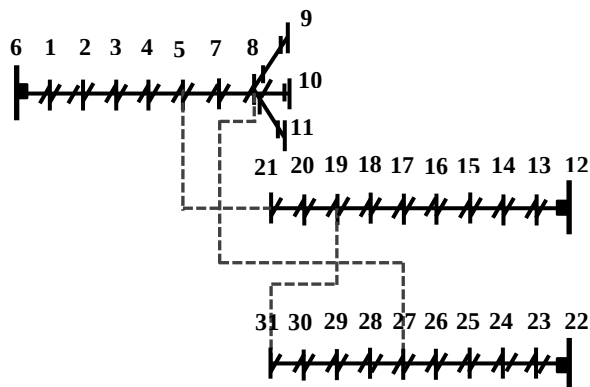
$k = k + 1$;

Док Критеријум заустављања /* нпр $T_k < T_{min}$ */

Изаз: Најбоље нађено решење (s_{best})

5. ВЕРИФИКАЦИЈА АЛГОРИТМА И АНАЛИЗА РЕЗУЛТАТА

У овој глави је извршена верификација алгоритама који су коришћени за прорачун поузданости и за оптималну аутоматизацију дистрибутивне мреже (ДМ). За ову сврху изабрана ја ДМ која се састоји од три средњенапонска извода и приказана је на слици 1.



Слика 1. Приказ тест ДМ [6]

Трансформаторске станице СН/НН међусобно су повезане надземним и подземним кабловима чији су параметри дати у литератури [6]. Ту су такође наведене и вредности активног оптерећења за сваки од чворова ДМ. Ова мрежа има само обичне растављаче на гранама, испрекидане линије представљају нормално отворене растављаче и квадратићи су изводни прекидачи који су једини аутоматизовани.

У овом делу су дати резултати прорачуна оптималне аутоматизације мреже за период од 15 година, за два различита типа потрошача, а то су домаћинства и комерцијални корисници.

Ово је урађено како би се видело колико цена неиспоруцене енергије утиче на број постављених уређаја литературе у мрежи. Параметри као што су цена неиспоруцене електричне енергије и цена уређаја преузети су из [1].

У табели 1 приказани су резултати када се сматра да су потрошачи домаћинства.

Редом су приказане локације на којима је постављен реклозер (Р), секционер (С), аутоматизација НО растављача (НО) (у форми грана (чвор1 – чвор2)), цена неиспоруцене енергије у разматраном периоду T (C_{ENSI}), цена инвестиција (C_{INV}) и укупни трошкови (TC)

У табели 2 приказани су резултати прорачуна оптималне аутоматизације када су потрошачи комерцијални корисници. Резултати су приказани на исти начин као у табели 1.

Из приложених резултата може се видети да се аутоматизацијом ДМ укупни трошкови за разматрани период планирања значајно смањују. За случај домаћинстава то смањење износи 1,7 пута, а за случај комерцијалних корисника чак 7,6 пута

Табела 1. Резултати прорачуна када су потрошачи типа: домаћинства

	P	C	HO	C _{ENSI} [\$]	C _{INV} [\$]	TC [\$]
Почетно решење	/	/	/	78399,2	0	78399,2
Оптимално решење	6(5-7), 15(14 -15)	2(1-2), 23(22- 23), 29(28- 29)	11(5-21), 12(8-27)	18823,74	26600,00	45423.7

Табела 2. Резултати прорачуна када су потрошачи типа: комерцијални корисници

	P	C	HO	C _{ENSI} [\$]	C _{INV} [\$]	TC [\$]
Почетно решење	/	/	/	2438220	0	2438220
Оптимално решење	7(7-8), 20(19 -20)	3(2-3), 5(4-5), 15(14- 15), 18(17- 18), 21(20- 21), 21(20 -21), 25(24- 25), 26(25 , 26), 27(26- 27), 28(27 -28), 30(29 -30), 30(29- 30)	11(5-21), 12(8-27), 22(19-31)	254872,83	64900	319773

6. ЗАКЉУЧАК

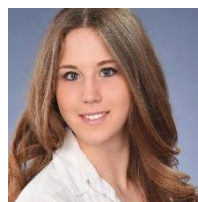
Аутоматизација ДМ је један од најефикаснијих начина за унапређење поузданости ДМ. У овом раду је предложена метода заснована на мета-хеуристици за одређивање оптималног броја и локација аутоматизованих прекидача како би се цена неиспоруцене енергије у неком разматраном периоду смањила. Показано је да се у свим случајевима исплати убацивање аутоматизованих уређаја па чак и када је цена неиспоруцене енергије мала. Такође, може се приметити када је цена неиспоруцене енергије већа исплативо је убацивати више аутоматизованих уређаја него када је цена мања.

7. ЛИТЕРАТУРА

- [1] Б.Брбаклић: *Одређивање оптималног броја, типа и локације уређаја за аутоматизацију електродистрибутивних мрежа*, докторска дисертација; Факултет техничких наука, Нови Сад, 2018.
- [2] М.Д.Нимрихтер: *Електродистрибутивни системи*; ФТН издаваштво, Нови Сад, 2009.
- [3] Г.Швенда: *Специјализовани софтвери у електроенергетици*, скрипта са предавања из истоименог предмета, Факултет техничких наука, Нови Сад, 2019.
- [4] Д.Поповић, Д.Бекут, В.Дабић: *Специјализовани ДМС Алгоритми*; Просвета, Нови Сад, 2011.
- [5] Ж.Поповић: *Планирање развоја дистрибутивних мрежа у присуству неизвесности*; докторска дисертација; Факултет техничких наука, Нови Сад, 2011.

[6] Ching-Tyong Su, Chu-Sheng Lee: Network reconfiguration of distribution systems using improved mixed-integer hybrid differential evolution; *IEEE Trans. on. Power Delivery*, Vol. 18, No. 3, July 2003; pp. 1022-1027; DOI: [10.1109/TPWRD.2003.813641](https://doi.org/10.1109/TPWRD.2003.813641)

Кратка биографија



Биљана Татар рођена је у Новом Саду 1996. године. Дипломски рад на Факултету техничких наука из области Електротехнике и рачунарства – Електроенергетски системи одбранила је 2019. године.



Невен Ковачки рођен је у Зрењанину, 1987. године. Дипломирао је, мастерирао и докторирао на Факултету техничких наука из области Електротехнике и рачунарства – Електроенергетски системи редом 2010, 2011. и 2018. године.

SISTEM ZA OBRADU PODATAKA PAMETNOG DOMA**SMART HOME DATA PROCESSING SYSTEM**Nemanja Krajčinović, Dejan Nemec, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Cilj ovog rada jeste da razvije sistem za obradu velikih količina podataka iz okruženja pametnog doma u cilju izvođenja zaključaka i omogućavanja predviđanja različitih događaja koji pospešuju rad pametnog doma i čine svakodnevni život lakšim.

Ključne reči: internet stvari, pametni dom, obrada velikih količina podataka.

Abstract – This paper aims to develop a system for processing large amounts of data from the smart home environment to draw conclusions and enable the prediction of various events that enhance the work of the smart home and make everyday life easier.

Keywords: internet of things, smart home, big data.

1. UVOD

Računarska tehnika se poslednjih 30 godina razvija u mnogo različitih smerova ali sa jednim ciljem – da poboljša život ljudi. Određene oblasti su već dovedene na visok nivo apstrakcije i kao takve, pomažu u svakodnevnom aktivnostima i olakšavaju život. Naravno, to nije razlog za stagniranje trenda razvoja tih oblasti, kao i otkrivanja novih. Početak 21. veka doneo je nagli razvoj fizičke arhitekture i programske podrške računara što je omogućilo i razvoj mašinskog učenja. Danas, mašinsko učenje se koristi u različitim oblicima na velikom broju mesta i prisutno je čak i bez znanja korisnika.

Ovaj rad predstavlja jedan od primera gde je upotreba različitih tehnika optimizacije velike količine podataka i mašinskog učenja iskorišćena u praktične svrhe. U sprezi sa tehnologijama iz oblasti računarskih komunikacija i fizičke arhitekture računara, kao i programske podrške, cilj ovog rada bio je da se optimizuje baza podataka nastala u jednom privatnom stambenom prostoru i na osnovu te baze kreiranje predikcionih modela za potrošnju električne energije.

U drugom poglavlju iznete su osnove i aspekti koje predstavljaju potporu za ovaj rad. U trećem poglavlju iznet je koncept celokupnog sistema, kao i neki od ključnih aspekata zašto je određena tehnologija izabrana. Detaljan prikaz algoritma predstavljen je u četvrtom poglavlju, kao i specifičnosti koje su se pojavile pri izradi zadatka a takođe i kako su prevaziđene. Postupak testiranja i ostvareni rezultati prikazani su u petom poglavlju, a zaključak u šestom.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Živko Bojović, vanr. prof.

2. OSNOVE RADA**2.1. Internet stvari**

U poslednjoj deceniji, kućni aparati prešli su put od izrade jednostavnih uređaja koji su podržani jeftinim senzorima, do inteligentnih uređaja koji mogu detektovati kretanje čoveka. Pametna kuća predstavlja okruženje u kome se kućni aparati mogu nadgledati i kontrolisati daljinski, na osnovu različitih ugrađenih uređaja kao što su senzori i aktuatori. Neki sistemi pametnih kuća, koji su razvijeni poslednjih godina, mogu da detektuju prisutnost na osnovu senzora pokreta. Pasivni senzori, zasnovani su na infracrvenim i drugim sličnim tehnologijama, i u tu grupu spadaju i senzori pokreta. Nedostatak tih tehnologija je što ne detektuju objekte kada korisnik nije dugo u pokretu. Alternativa za prevazilaženje ovog nedostatka je upotreba neprekidnih tehnologija zasnovanih na kompjuterskoj viziji ili drugih sličnih sveprisutnih tehnologija detekcije objekata. Aktivni senzori prikupljaju značajne informacije, uključujući privatne podatke korisnika, tako da je izazov ispravno odvojiti anonimne podatke. Postoje dva glavna razloga za sprečavanje usvajanja aktivnih i invazivnih senzoriskih tehnologija u pametnim kućama:

- teško generalizovanje modela sa promenljivim uslovima okruženja,
- briga korisnika za uočenu nametljivost.

Ponašanje senzora i aktuatora (nametljivo ili nenametljivo), može značajno varirati u zavisnosti od konteksta objekta (kuće, zgrade). To je moguće precizno postići praćenjem zauzetosti pomoću dostupnih informacija iz nenametljivih izvora podataka kao što su sobna temperatura, rashladni uređaji, potrošnja vode i niz drugih povezanih uređaja [1]-[3].

Predmet ovog istraživanja je inteligentno izračunavanje na osnovu preferencija korisnika pomoću tehnologije velikih količina podataka prikupljenih u IoT okruženju korišćenjem nenametljivog pristupa modelovanja. Uključivanjem personalizacije zasnovane na obrascima ponašanja korisnika IoT okruženja, predloženi model sistema može pružiti pametniji odgovor bez potrebe za intervencijom korisnika, čime postiže nenametljivu automatizaciju pametne kuće.

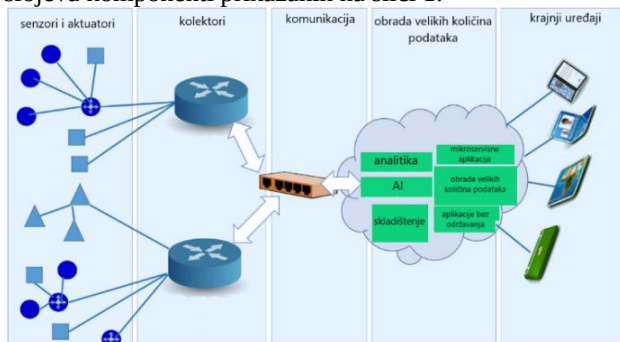
2.2. Obrada velikih količina podataka

Identifikovanje obrazaca ljudskog ponašanja u realnom vremenu povezanih sa pametnom kućom prilika je za modelovanje personalizacije. To se može postići korisničkim profilisanjem podataka u realnom vremenu generisanim sa uređaja i senzora kojima upravlja korisnik.

Osim toga, različiti elementi podataka i atributi iz različitih izvora životne sredine mogu se analizirati kako

bi se razumeli obrasci podataka koji prikazuju ponašanje određenog korisnika i fizičke aktivnosti koje omogućavaju personalizovano profilisanje korisnika u pametnom domu [4]-[6].

Generička IoT arhitektura koja proširuje mogućnosti obrade velikih količina podataka obično se sastoji od pet slojeva komponenti prikazanih na slici 1.



Slika 1. Generička IoT arhitektura

Predloženi model za automatizaciju pametnih kuća zasnovan je na masivnom nenametljivom prikupljanju podataka i generalnom konsenzusu tehnologije velikih količina podataka – četiri V (raznolikost, verodostojnost, brzina, vrednost).

2.3. Mašinsko učenje

Za razvoj prototipa prilagođavaju se parametri pametnog doma, poput temperature i osvetljenja, koristeći regresore. Model namerava da predvidi takve obrasce tokom realnog vremena, tako da se izvrše neophodna prilagođavanja parametara [7].

3. KONCEPT REŠENJA

Prototip rešenja pretpostavljenog sistema delom je razvijen i implementiran u jednom privatnom okruženju. Vodeći računa o aspektima nenametljivosti, senzori i aktuatori su postavljeni prema smernicama iz prethodnog poglavlja. Implementirani sistem obuhvata širok spektar senzora za različite uređaje, od kuhinjskih električnih uređaja, preko prekidača, utičnica, bojlera, pa sve do informativnih stanja mobilnih uređaja koji su se nalazili u domu.

Podaci su prikupljeni 4 meseca i to u periodu od oktobra 2020. godine do februara 2021. godine. Beleženi su u pravilnim vremenskim intervalima – svakog minuta, gde je u svakom minutu zapisivano stanje sa senzora iz okruženja. Takvo okruženje generiše veliku količinu podataka sa mnoštva različitih uređaja, koji se skladište na kolektoru. Ovakvo rešenje pripada konvencionalnim IoT implementacijama pametnog doma, koja se danas u praksi često mogu sresti. Ono što ih karakteriše je nemogućnost predviđanja događaja u budućnosti i automatizacija upravljačkih funkcija. Osnovna ideja ovog sistema koje predstavlja fundament istraživanja je mogućnost prevazilaženja ovih problema, upravo primenom veštačke inteligencije, odnosno tehnologija mašinskog učenja i obrade velikih količina podataka.

Prilikom implementacije predloženog rešenja najveći značaj je pridat ispunjavanju konsenzusa o nenametljivosti sistema pametnog doma, ispunjavanju svih standarda po pitanju komunikacija između uređaja i kolektoru sistema. Međutim, nedostatak ovog sistema je

nepostojanje modela efikasnog skladištenja takvih, velikih količina podataka, kao i njihove obrade. Zato ovaj rad predstavlja koncept i primer implementacije takvog sistema, koji bi omogućio efikasno skladištenje podataka, njihovo prečišćavanje i obogaćivanje, kao i implementaciju modela estimacije ponašanja korisnika i samim tim i personalizacije sistema.

Prečišćavanje i obogaćivanje podataka uvedeno je u cilju da se optimizuje skladišni prostor i obezbedi njegovo efikasno korišćenje. Ovaj proces je izvršen primenom statističkih analiza i izvođenjem odgovarajućih zaključaka na osnovu njih. Zato i najveći doprinos ovog istraživanja, upravo leži u razvoju i implementaciji ovog dela sistema koji obezbeđuje nenametljivu automatizaciju upravljačkih funkcija pametnog doma.

Nakon sređivanja baza podataka u skladištima, primenjeno je mašinsko učenje kao tehnologija, odnosno ključni aspekt koji može obezbediti predikciju ponašanja korisnika sistema, kao i personalizaciju istog. U radu su date određene pretpostavke, modeli koji su isprobani i opisani rezultati njihove primene, koji su potom upoređeni i objašnjeni.

4. PROGRAMSKO REŠENJE

Implementirani sistem ne poseduje potrebne mehanizme za efikasno skladištenje i obradu podataka. Baza podataka je na početku imala 176.830 uzoraka i 250 obeležja.

Raščlanjivanje podataka je prvi deo sistema gde će ovakvi, nestruktuirani podaci, biti prevedeni u struktuirane podatke, u odgovarajućim formatima. Algoritam obrade jedne log datoteke se sastoji iz sledećih koraka:

1. Učitavanje jedne linije te datoteke.
2. Prevođenje ključa linije u format datuma i vremena kada je linija zabeležena.
3. Pribavljanje imena svakog senzora, kao i vrednosti koja je zabeležena o tom senzoru.
4. Zapis takve linije kao jednog uzorka u CSV bazu podataka.

Nakon što su podaci prevedeni u jedinstvenu CSV bazu podataka, izvršeno je prevođenje niza karaktera koji predstavljaju datum i vreme, u pogodan oblik za dalju analizu, a to je poseban zapis vremena, dana, meseca i godine za svaki uzorak iz baze podataka.

U cilju uklanjanja redundantnih podataka prvo se vrši provera broja nedostajućih vrednosti po obeležju. Algoritam je sledeći:

1. sumiranje praznih polja po kolonama,
2. zapis konačnih vrednosti u listu sa imenima obeležja.

Na osnovu ovog algoritma izvučen je jedan veoma zanimljiv zaključak: čak 26 obeležja u ovoj bazi imaju preko 99% nedostajućih podataka, što automatski dovodi do akcije uklanjanja tih obeležja iz baze podataka.

Vizuelnim analiziranjem i testiranjem baze zaključeno je da postoji još jedan problem, a to je nepravilan zapis različitih stanja na nivou cele baze, odnosno, korišćenje različitih ključnih reči i oznaka za jednu te istu vrednost. Tako su nedostajuće vrednosti označavane sa „None“, „unavailable“, „unknown“, „disarmed“. Nakon toga, uzorci koji sadrže nedostajuće podatke su uklonjeni iz baze. Uklonjeni su uzorci sa barem jednim nedostajućim podatkom i broj uzoraka je umanjnjen na 151.779 uzoraka.

Sledeće je trebalo definisati tipove obeležja, kako bi podaci bili iskoristljivi u daljem procesu. Definisanje tipa svakog obeležja je rađeno na osnovu analize histograma i utvrđivanja kakve se vrednosti pojavljuju na nivou jednog obeležja. Ovim postupkom je zaključeno da je 95 obeležja numeričko. Dalje, 92 obeležja su identifikovana kao kategorička obeležja, sa binarnim ishodima.

Analizom je dodatno utvrđeno da postoje obeležja sa tekstualnim informacijama koja u ovakvom sistemu ne mogu biti od koristi. Tim postupkom je uklonjeno 26 obeležja. Konačno, baza je prevedena u pogodan format za dalju upotrebu sa 151.779 uzoraka i 191 obeležjem.

Prečišćavanje i obogaćivanje podataka je proces kojim se može obezbediti redukcija podataka bez gubitka informacija koji se nalaze u podacima.

Prvi korak je analiziranje histograma pojavljivanja vrednosti unutar svakog obeležja i raspodele tih vrednosti. Prilikom vizuelne analize rezultata zapažena je jedna nepravilnost. Veliki broj obeležja imao je raspodelu vrednosti visoko koncentrisanu oko jedne vrednosti, što se moglo videti i na histogramu pojavljivanja vrednosti za to obeležje. Na histogramu je bilo vidljivo i da se jedna vrednost pojavljuje veliki broj puta, skoro pa približan broju uzoraka. Zbog toga je sproveden algoritam za uklanjanje obeležja kod kojih se jedna vrednost ponavlja i to u više od 99% u odnosu na ukupan broj uzoraka. Opravdanje za uklanjanje ovakvih obeležja leži u činjenici da takva obeležja ne nose informaciju za ovaj sistem i samim tim nikako ne mogu doprineti razvoju ovog sistema. Ovim algoritmom je uočeno i uklonjeno čak 49 obeležja iz baze podataka.

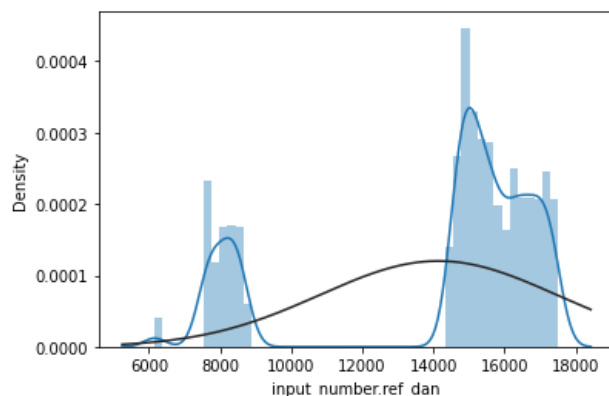
Drugi korak ovog dela analiziranja baze podataka bio je kreiranje i analiza korelacione matrice.

S obzirom da baza podataka i dalje sadrži veliki broj obeležja, nije bilo moguće jasno prikazati matricu sa vidljivim odnosima između obeležja, ali se uočilo da je u nekim delovima korelacija približna maksimumu, što znači da postoje obeležja koja su visoko korelisana i potrebno ih je ukloniti. Detaljnijom analizom ove matrice, utvrđeno je da se visoke korelacije javljaju između obeležja čiji je sufiks imena sličan ili se može pretpostaviti da se odnose na istu ili sličnu stvar, a da se prefiksi razlikuju. Odatle se može izvući sledeći zaključak: postoje obeležja koja su izvedena iz osnovnog skupa obeležja ovog sistema. Detaljnom analizom korelacione matrice utvrđeno je da sva obeležja koja imaju korelacioni koeficijent preko 0,6 sa nekim drugim obeležjem mogu biti uklonjena iz baze podataka. Algoritam uklanjanja suvišnih obeležja izveden je algoritmom korišćenja korelacione matrice i to podataka ispod glavne dijagonale, gde su, idući red po red, odnosno, obeležje po obeležje, uklanjanja ona koja imaju korelacioni koeficijent veći od 0,6 sa analiziranim obeležjem. Ovim postupkom uklonjeno je 70 obeležja iz baze podataka, čime je baza podataka redukovana na 72 obeležja.

Treći korak ovog dela predstavlja analiza ciljnog obeležja za predikciju. Informacija koja je dobijena uz bazu podataka jeste da je obeležje *input_number.ref_dan* ciljano obeležje, da ono predstavlja zbirnu potrošnju električne energije doma na nivou sve tri faze i da je izraženo u kilovat-časovima (kWh). Na osnovu tog podatka, na slici 2

prikazan je histogram pojavljivanja vrednosti unutar tog obeležja u odnosu na normalnu raspodelu.

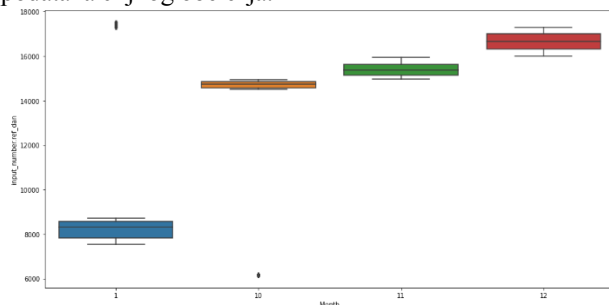
Na osnovu ovog histograma može se jasno uočiti da se ovde vrednosti grupišu oko vrednosti oko 8.000 i 16.000 i da se ne može estimirati normalnom raspodelom na nivou celog uzorka ovog obeležja.



Slika 2. Histogram ciljnog obeležja sa estimacijom u odnosu na normalnu raspodelu

Takođe, ciljno obeležje je statistički analizirano i u odnosu na vremenske podatke.

Na slici 3 može se videti da su raspodele poredane u rastućem poretku u odnosu na mesec, a isti zaključci se mogu uvažiti i za raspodele vrednosti u odnosu na dane u mesecu i vreme u toku jednog dana. Na osnovu tih grafika može se izvući konačan zaključak, koji je potvrđen i vizuelnom analizom ovog obeležja: ciljno obeležje je zapisivano kumulativno, odnosno, vrednosti su akumulisane i zbog toga se na graficima pojavljuju rastući poreci. Takođe, krajem januara 2021. godine vrednost je umanjena i zato se desila promena pika podataka u poslednjem delu baze podataka ciljnog obeležja.



Slika 3. Raspodela uzoraka ciljnog obeležja u odnosu na mesece u godini

Automatizacija pametnog doma i personalizacija takvog sistema leže u primeni veštačke inteligencije na ovakve podatke. Ideja ovakvog sistema svakako nije predviđanje samo jednog obeležja, potrošnje električne energije na nivou celog doma, ali zbog nedostatka informacija o obeležjima, ovde će biti isprobano samo predviđanje potrošnje električne energije.

Baza podataka je prvobitno podeljena na trening i test skup u odnosu 9:1 i to uz prvobitno mešanje uzoraka na slučaj. Modeli koji su testirani jesu:

- opšti model linearne regresije,
- model linearne regresije sa hipotezom koja sadrži interakcije između obeležja,

- opšti model linearne regresije sa standardizovanim obeležjima,
- Ridge regresioni model,
- Lasso regresioni model.

Kao kriterijum pronalaženja optimalnog modela linearne regresije korišćen je RSS. U tabeli 1 predstavljene su RSS vrednosti svakog testiranog modela.

Tabela 1. Rezultati obuke različitih regresionih modela

Regresioni model	RSS
Opšti model linearne regresije	35.698.190.141
Model linearne regresije sa hipotezom koja sadrži interakcije između obeležja	2.575.054.607
Opšti model linearne regresije sa standardizovanim obeležjima	35.698.190.141
Ridge regresioni model	35.698.232.742
Lasso regresioni model	35.697.059.689

Iz tabele rezultata se može zaključiti da su rezultati modela daleko od prihvatljivih rezultata i da ovakvi modeli nisu iskoristivi. Dodatnom analizom su utvrđene dve nepravilnosti u okviru ciljnog obeležja, a to su:

- rezultati potrošnje električne energije su zapisivani kumulativno, gde početak nije jasno označen,
- postoje dva različita pika u podacima i to je uočeno da se dešava zato što je u jednom trenutku, sredinom januara 2021. godine, rezultat umanjnjen za neku vrednost, što ovde nije objašnjivo niti shvatljivo zašto se desilo.

Ove dve nepravilnosti mogle su biti primećene na slici 2. Zbog ovih problema pokušano je da se određenim algoritmima prevaziđu ovi problemi, ali se pokazalo da postoje još neki dodatni problemi zbog kojih nije moguće jednoznačno ukloniti navedene nepravilnosti.

5. REZULTATI

U trećem poglavlju objašnjeno je da je sistem pametnog doma kreiran u privatnom vlasništvu zadržao konstrukciju kolektora i skladišta na nivou jednog uređaja i nije brinuo o efikasnom skladištenju podataka, kao ni o mogućoj obradi tih podataka. Zato su podaci zapisivani u tekstualnom formatu u log fajlove za 4 meseca zauzimali 4,35 GB. Raščlanjivanjem JSON objekata i pakovanjem stanja senzora u CSV bazu, veličina je redukovana na 232,6 MB. Sprovedenim operacijama za uklanjanje neiskoristljivih obeležja, kao i redefinisanjem tipova obeležja u odgovarajući oblik veličina baze podataka redukovana je na 96,94 MB. Redukcijom obeležja sa visokim korelacionim koeficijentom baza je redukovana na veličinu od 43,13 MB. Uzevši u obzir da je originalna veličina podataka bila 4,35 GB, faktor kompresije iznosi 103,58, što znači da je ostvarena ušteda skladišnog prostora 99,03%.

Najbolji model, model linearne regresije sa hipotezom koja sadrži interakcije između obeležja evaluiran je pomoću srednje apsolutne greške (MAE), kroz unakrsnu validaciju, gde je baza podeljena na 10 podskupova sa odgovarajućim parametrima i srednja apsolutna greška je 3315,86. Vizuelnom analizom ciljnog obeležja, zaključeno je da su promene između kumulativnih beleženih vrednosti reda 10^{-2} , a negde čak i 10^{-3} . Iz ovoga se takođe može zaključiti da ovakva srednja apsolutna

greška predstavlja veoma lošu mogućnost za predviđanje potrošnje električne energije.

6. ZAKLJUČAK

U ovom radu prikazano je kako se različite tehnologije danas dostupne mogu praktično upotrebiti za dobijanje informacija koje čovek ne može da pribavi, ili može ali na vrlo težak način. Iako dobijeni rezultati nisu na visokom nivou, uspešno je postignuto izvođenje zaključaka šta je potrebno unaprediti i promeniti na sistemu pametnog doma, kako bi se podaci mogli upotrebiti kao kvalitativni podaci u metodama mašinskog učenja.

7. LITERATURA

- [1] Stojkoska, B.L.R.; Trivodaliev, K.V. [A review of Internet of Things for smart home: Challenges and solutions](#). *J. Clean. Prod.* 2017, *140*, 1454–1464,
- [2] Choi, D.; Choi, H.; Shon, D. [Future changes to smart home based on AAL healthcare service](#). *J. Asian Arch. Build. Eng.* 2019, *18*, 190–199,
- [3] Helal, S.; Bull, C.N. [From Smart Homes to Smart-Ready Homes and Communities](#). *Dement. Geriatr. Cogn. Disord.* 2019, *47*, 157–163,
- [4] Benmansour, A.; Bouchachia, A.; Feham, M. [Multioccupant Activity Recognition in Pervasive Smart Home Environments](#). *ACM Comput. Surv.* 2016, *48*, 1–36,
- [5] Russell, L.; Goubran, R.; Kwamena, F. Personalization Using Sensors for Preliminary Human Detection in an IoT Environment. In Proceedings of the 2015 International Conference on Distributed Computing in Sensor Systems, Fortaleza, Brazil, 10–12 June 2015; pp. 236–241,
- [6] Li, R.Y.M.; Li, H.C.Y.; Mak, C.K.; Tang, T.B. [Sustainable Smart Home and Home Automation: Big Data Analytics Approach](#). *Int. J. Smart Home* 2016, *10*, 177–198,
- [7] Overmars, A.; Venkatraman, S. [Towards a Secure and Scalable IoT Infrastructure: A Pilot Deployment for a Smart Water Monitoring System](#). *Technologies* 2020, *8*, 50.

Kratka biografija:



Nemanja Krajčinović rođen je u Šapcu 1998. godine. Osnovne akademske studije, na smeru Komunikacione tehnologije i obrada signala, završio je 2020. godine, sa odbranom bečelor rada pod naslovom „Praćenje kretanja i zadržavanja ljudi u zatvorenom prostoru primenom kompjuterske vizije”. Do sada je objavio 1 naučni rad na međunarodnom skupu TELFOR.

kontakt: kr.nemanja@gmail.com



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.

Zahvalnica:

Izradu ovog rada pomogao je Fakultet tehničkih nauka u Novom Sadu, Departman za energetiku elektroniku i telekomunikacije, u okviru projekta pod nazivom „Razvoj i primena savremenih metoda u nastavi i istraživačkim aktivnostima na Departmanu za energetiku, elektroniku i telekomunikacije”.

PRIMENA MAŠINSKOG UČENJA ZA IDENTIFIKOVANJE LIMFNIH ČVOROVA KOD TIROIDNE ŽLEZDE**MACHINE LEARNING APPLICATION FOR IDENTIFYING LYMPH NODES IN THE THYROID GLAND**

Nataša Avramović, Dejan Nemec, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Ovaj rad opisuje kreiranje i analizu modela za identifikovanje papilarnog karcinoma tiroidne žlezde. Predmet istraživanja su pacijenti kod kojih klinički i ultrazvučno nisu detekovane patološke pojave, odnosno metastaze limfnih čvorova. Operativni nalaz tih pacijenata pokazao je da se preoperativni rezultati, odnosno ultrazvuk, ne može smatrati pouzdanim, i da dovodi do greške kod oko 48% pacijenata. Cilj rada je dostići senzitivnost i specifičnost modela koji će pružiti dovoljno dobar rezultat.

Ključne reči: mašinsko učenje, klasifikacija

Abstract – The paper describes the creation and analysis of a model for the identification of papillary thyroid cancer. The subject of the research are patients in whom no pathological phenomena or lymph node metastases, have been detected clinically and ultrasound. The surgical outcome findings of these patients point out that preop ultrasound diagnosis could not be considered reliable and that they were wrong in about 40% of patients. The aim of this paper is to achieve the sensitivity and specificity of the model that will provide a good enough result.

Keywords: machine learning, classification

1. UVOD

Kako je era velikih podataka uveliko počela, postoji i izražena potreba za automatizovanim metodama njihove analize i obrade. Dostupnost računara danas i jeftine memorije omogućavaju lak način skladištenja podataka, pa čak i onih nepotrebnih.

Primena obrade velikih podataka u medicini može doprineti veliki značaj. Pružanje najbolje usluge i nega pacijenata dobijaju se kreiranjem modela za personalizovanu, prediktivnu, preventivnu svrhu u medicini, i zasnovani su na korišćenju elektronskih zdravstvenih kartona i generisanju velikih količina podataka.

Analitika velikih podataka obuhvata integraciju heterogenih podataka, analizu, modelovanje, tumačenje i validaciju. Rezultati dobijeni analitikom podataka treba da pruže korist kako pacijentima, tako i lekarima.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Živko Bojović, vanr. prof.

Predmet ovog istraživanja jeste grupa pacijenata sa papilarnim karcinomom štitne žlezde, kod kojih limfni čvorovi u bilo kom delu vrata (centralnom ili lateralnom) klinički (pregled lekara, nalaz krvi) i ultrazvučno nisu opisani kao sumnjive ili patološke pojave. Pacijent je preoperativno išao na ultrazvuk te regije, gde se gledao tumor i procenjivalo se da li su limfni čvorovi maligni ili benigni. Zlatni standard u ovom slučaju je rezultat operacije, odnosno operativni nalaz i on je pokazao stvarno stanje. Kako se ultrazvuk pokazao loše u ovom istraživanju, i pogrešio kod oko 40% pacijenata, potrebno je ispitati koji parametri, dobijeni preoperativnom analizom i operacijom, dovode do metastaza u limfnim čvorovima.

U drugom poglavlju iznete su osnove i aspekti koje predstavljaju potporu za ovaj rad. U trećem poglavlju iznet je koncept celokupnog sistema, kao i neki od ključnih aspekata zašto je određena tehnologija izabrana. Detaljan prikaz algoritma predstavljen je u četvrtom poglavlju, kao i specifičnosti koje su se pojavile pri izradi zadatka a takođe i kako su prevaziđene. Postupak testiranja i ostvareni rezultati prikazani su u petom poglavlju, a zaključak u šestom.

2. OSNOVE RADA

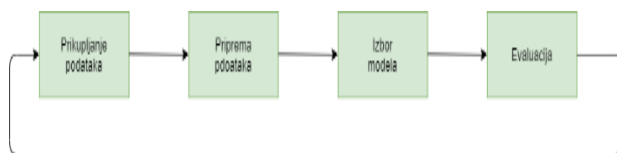
Mašinsko učenje predstavlja jedan od najčešćih oblika veštačke inteligencije, statistike i računarstva, i fokusira se na korišćenje podataka i algoritama tako da imitiraju način učenja kod ljudi. Svakodnevnim prikupljanjem dostupnih podataka omogućava se razvijanje sve većeg broja različitih modela i pametnije donošenje odluka.

Korišćenjem statističkih metoda, ovaj tip učenja obrađuje velike skupove podataka, pronalazi neke obrasce i zakonitosti u njima, pravi predviđanja na osnovu kojih se na kraju donosi odluka. Mašinsko učenje bavi se projektovanjem algoritama koji automatski izvlače bitne informacije iz podataka [1]-[3].

Razvojni ciklus modela mašinskog učenja (Slika 1) je proces izgradnje efikasnog algoritma, tako da se ciklus ponavlja i u svakoj iteraciji dolazi do poboljšanja podataka, modela i bolje procene. Brzina kojom se ciklus ponavlja, predstavlja ono što određuje troškove, a osnovna intervencija je da se izvrši što bolja optimizacija troškova.

U tu svrhu, danas postoje razni dostupni alati. Svrha razvojnog ciklusa je dolazak do rešenja zadatog problema,

tako da ga je najpre potrebno razumeti, jer od toga zavisi krajnji rezultat [4].



Slika 1 Razvojni ciklus mašinskog učenja

Prvi korak pri stvaranju i razvoju modela mašinskog učenja je prikupljanje podataka. On ima za cilj identifikaciju i dobijanje svih podataka koji se odnose na problem koji se rešava. Nakon prikupljanja podataka, iste je potrebno pripremiti za dalju obradu. Priprema podataka za dalju obradu odnosi se na statističku analizu podataka, vizuelnu analizu i transformaciju podataka, koje se sprovode sve dok se ne postigne forma potrebna računaru za razumevanje.

Pri razvoju modela mašinskog učenja mora se voditi računa o tome da je cilj da obučeni model dobro radi na novim, neviđenim podacima. Kako bi se došlo do što objektivnije procene performansi modela, podaci se dele na dva dela:

- skup podataka za evaluaciju i
- skup podataka za validaciju.

U ovoj fazi se pripremljeni i obrađeni podaci koriste u izabranom algoritmu za obuku modela, odnosno, model uči iz pripremljenih i obrađenih podataka. Obuka modela se vrši nad trening podacima, a performanse modela se procenjuju na osnovu skupa za validaciju. Na osnovu informacija dobijenih u ovom koraku, može se završiti sa kreiranjem modela i to predstavljanjem zadovoljavajućih rezultata ili vratiti ponovnom obučavanju modela i podešavanju sve dok se ne dobiju zadovoljavajući rezultati [5]-[6].

3. KONCEPT REŠENJA

Grupa pacijenata koja nas interesuje su pacijenti sa papilarnim karcinomom štitne žlezde kod kojih klinički i ultrazvučno nisu opisani sumnjivi ili patološki limfni čvorovi u bilo kom delu vrata, odnosno identifikovan im je benigni tumor, i nisu otkrivene metastaze.

U ovu studiju uključeni su pacijenti sa patohistološki potvrđenom dijagnozom papilarnog karcinoma štitaste žlezde koji su operisani na na Institutu za onkologiju i radiologiju Srbije u periodu od januara 2015. godine do septembra 2021. godine.

Podaci su podeljeni tako da se u skupu za obuku nalazi 70% pacijenata, a u skupu za testiranje 30% pacijenata. Obuka je vršena metodom unakrsne validacije, kako bi se izbeglo natprilagođenje modela. U ovoj metodi, skup za obuku deli se na manje podskupove, i u ovom slučaju je odlučeno da u svakom podskupu bude približno 10 pacijenata. Postoji onoliko rundi koliko postoji podskupova, i validacija se vrši tako što je u svakoj rundi jedan podskup korišćen za testiranje, a ostali za obuku.

Ukoliko postoji veliki broj obeležja vršena je selekcija obeležja unapred, time su iskorišćena obeležja koja najviše doprinose rezultatu, i uklonjena obeležja koja

ništa ne doprinose, a povećavaju računarsku složenost i vreme izvršavanja. Pored selekcije obeležja, broj obeležja je smanjivan i pomoću rezultata dobijenih iz korelacione matrice, tako da ukoliko su dva obeležja visoko korelisana (pozitivno ili negativno) jedno od njih je uklanjano, jer nose istu informaciju.

Pošto su podaci označeni, odnosno za svakog pacijenta se zna ishod, ova vrsta pravljenja modela spada u nadgledano učenje, u klasifikaciju. Model je testiran na različitim klasifikatorima: *kNN*, *Decision tree*, *Logistic regression*, *Support vector machine*, *Random forest* i *XGBoost*.

Performanse klasifikatora merene su pomoću matrice konfuzije i AUC ROC krive. Iz matrice konfuzije računati su: senzitivnost, specifičnost i tačnost.

4. ANALIZA PODATAKA

4.1. Baza 1

Baza podataka sadrži 278 pacijenta, kod kojih je mereno 29 obeležja, i jedno dodatno obeležje u bazi podataka predstavlja klasu pacijenta. Na osnovu informacija dobijenih o bazi, uočeno je da postoje nedostajući podaci, i da je broj takvih podataka 613. Takođe, kod obeležja *Bilateralno_st* i *Lokacija_tumora_rezanj_recode* nedostaje oko 50% podataka, tako da se ta obeležja uklanjaju. Obeležja *ID* i *RB* se takođe uklanjaju, ona ne donose praktičan značaj rezultatu.

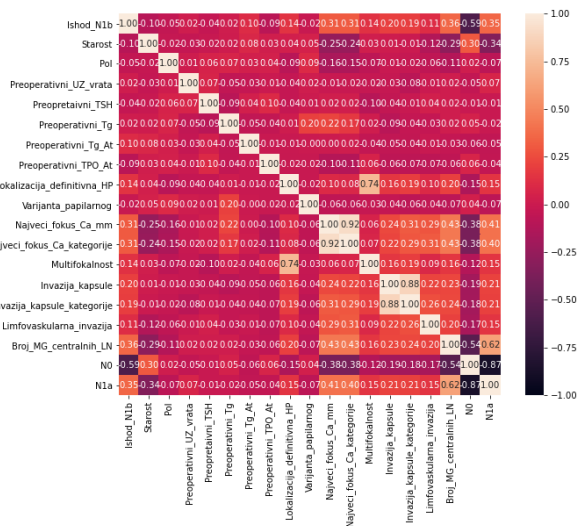
Analizirajući kategorička obeležja iz baze, doneti su dodatni zaključci. Obeležje *Starost* opisuje starost pacijenta, i posmatra se kao numerička varijabla, dok obeležja *Starost_kategorije_55* i *Starost_kategorije_45* opisuju takođe starost, samo se posmatraju kao kategoričke varijable i odnose se na pripadnost određenoj starosnoj grupi. Oдавде, radi analize statističkih vrednosti zdravih i bolesnih pacijenata odlučeno je da se samo obeležje *Starost* zadrži. Dalje, obeležja *Preoperativni_UZ_vrata_nominal* i *Preoperativni_UZ_vrata*, takođe se odnose na isto, tako da će u ovom slučaju samo obeležje *Preoperativni_UZ_vrata* biti zadržano. Obeležja *Multifokalnost*, *Broj_fokusa* i *Broj_fokusa_kategorije* takođe se odnose na isto, odnosno na to da li postoji jedan ili više tumora na žlezdi, tako da se samo obeležje *Multifokalnost* zadržava. Obeležja *Najveci_fokus_Ca_i_micro_Ca* i *Najveci_fokus_Ca_kategorije* se odnose na veličinu tumora, tako da se samo jedno zadržava, u ovom slučaju *Najveci_fokus_Ca_kategorije*.

Međuzavisnost obeležja prikazana je pomoću hitmape na slici 2.

Na osnovu slike 2 uočava se visoka korelacija između obeležja *N1a* i *N0*, tako da se jedno od tih obeležja uklanja, ovde je odabrano da to bude obeležje *N1a*. Takođe, visoka korelacija se primećuje kod obeležja *N0* sa ishodom, tako da se ipak i *N0* uklanja. Korelacija između obeležja *Najveci_fokus_Ca_kategorije* i *Najveci_fokus_Ca_mm* je jasna, jer prva predstavlja kategoričko obeležje, a druga isto to, samo numeričko, tako da se zadržava numeričko obeležje. Obeležje

Lokalizacija_definitivna_HP ima visoku korelaciju sa obeležjem *Multifokalnost*, tako da će i ono biti uklonjeno. Visoka međusobna korelacija postoji i kod obeležja *Invazija_kapsule* i *Invazija_kapsule_kategorije*, tako da se prvo obeležje uklanja.

Nakon uklanjanja visoko korelisanih obeležja, rađene su statističke vrednosti za numerička obeležja.



Slika 2 Korelaciona matrica za bazu 1

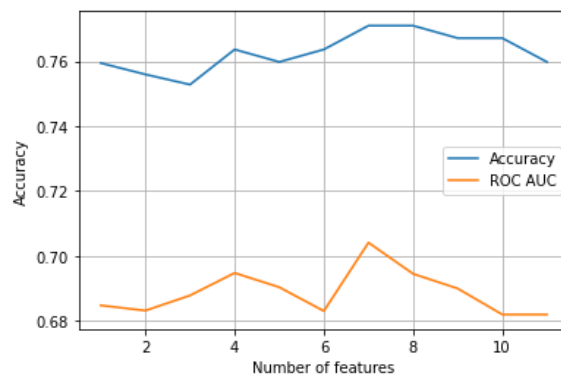
Na osnovu tabele 1, može se zaključiti da su statističke vrednosti zdravih i bolesnih osoba u pogledu ispitivane starosti približne, tako da ovo obeležje ne nosi značajnu informaciju. Obeležje *Preoperativni_TSH* ima duplo veću maksimalnu vrednost kod zdravih osoba, tako da može biti od značaja pri dobijanju rezultata. Obeležje *Preoperativni_Tg* pokazuje približne statističke vrednosti za zdrave i bolesne pacijente, tako da se i ono može isključiti iz daljeg razmatranja.

Obeležje *Preoperativni_Tg_At* pokazuje značajnu razliku u statističkim vrednostima kod zdravih i bolesnih pacijenata, gde pokazuje više od 10 puta veću vrednost maksimuma, i oko 5 puta veću srednju vrednost kod bolesnih pacijenata. Obeležje *Preoperativni_TPO_At* pokazuje 5 puta veću srednju vrednost, i približno duplo veću vrednost kod zdravih pacijenata. Obeležje *Najveci_fokus_Ca_mm* pokazuje razliku u statističkim parametrima, dok obeležje *Broj_MG_centralnih_LN* ne pokazuje značajnu razliku između zdravih i bolesnih pacijenata, tako da se ni ono ne može smatrati bitnim za donošenje rezultata.

Tabela 1 Statističke vrednosti za bazu 1

Obeležje	Zdrave osobe			Bolesne osobe		
	Min	Max	Sr. vr.	Min	Max	Sr. vr.
Starost	19	75	47,8	19	75	44,6
Preoperativni_TSH	0.01	11.93	2.15	0.01	5.56	2.0
Preoperativni_Tg	0.04	780	84.2	0.2	732	88.6
Preoperativni_Tg_At	0.2	1128	159.9	0.1	44061	799.89
Preoperativni_TPO_At	0.1	8663	360.32	0.5	1685.5	172.4
Najveci_fokus_Ca_mm	1	45	11.7	1	65	19.2
Broj_MG_centralnih_LN	0	17	1	0	15	3

Kada se nad preostalim obeležjima izvrši selekcija obeležja dobija se vrednost AUC ROC krive oko 71%, i postiže tačnost od oko 77% pri radu sa samim obeležjem (Slika 3).



Slika 3 Selekcija obeležja

Pravljenjem modela sa obeležjima koja doprinose rezultatu, dobijaju se rezultati prikazani u Tabeli 2.

Tabela 2 Evaluaciona metrika za bazu 1

Klasifikator	Tačnost	Senzitivnost	Specifičnost	Srednja apsolutna greška
kNN	70,24	21,74	88,52	11,31
DT	67,86	34,78	80,33	15,82
LR	72,62	21,74	91,8	4,46
RF	70,24	21,74	88,52	0,01
SVM	73,81	21,74	93,44	16,64
XGBoost	71,43	13,04	93,44	0,01

Na osnovu tabele 2, zaključeno je da nije postignuta odgovarajuća senzitivnost što je bio jedan od uslova.

Analizom dobijenih rezultata sa doktorima koji su radili na prikupljanju podataka, zaključeno je da je došlo do značajnih propusta prilikom pravljenja baze, tako da je ponovna analiza rađena na bazi 2, u kojoj su uočeni problemi rešeni.

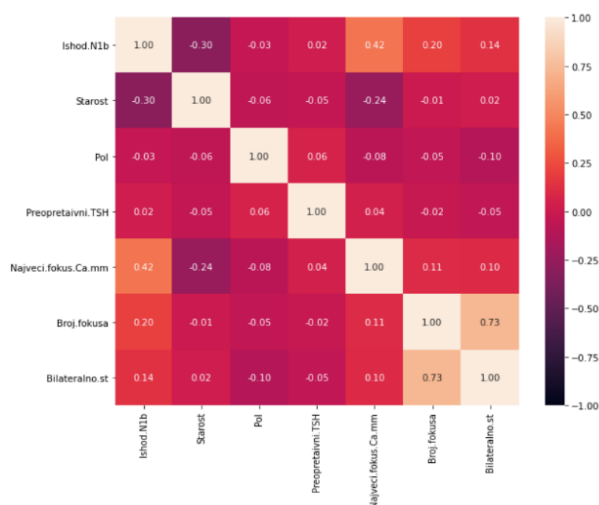
4.2. Baza 2

Baza podataka sadrži 273 pacijenta, kod kojih je mereno 7 obeležja, i jedno dodatno obeležje u bazi podataka predstavlja klasu pacijenta. Osnovnom analizom baze podataka utvrđeno je da svi pacijenti uglavnom imaju sve ispitivane parametre, i da samo u nekim slučajevima parametar *Preoperativni.TSH* nedostaje. Odnosno, nedostajuće vrednosti postoje samo za taj parametar. U zavisnosti od tipa varijable nedostajuće vrednosti se popunjavaju. Kako je parametar *Preoperativni.TSH* numerička varijabla, i kako nedostaje samo 6 vrednosti, što je oko dva posto, odlučeno je da se nedostajuće vrednosti popune metodom srednje vrednošću kolone, jer tako mala količina ubačenih podataka neće mnogo uticati na rezultat.

Međuzavisnost obeležja prikazana je pomoću matrice korelacije na slici 4.

Kao najviše korelisana obeležja mogu se izdvojiti *Broj.fokusa* i *Bilateralno.st*, dok ostala obeležja uglavnom ne pokazuju visoku korelisanoost. Između kategorija *Broj.fokusa* i *Bilateralno.st* postoji veza kada se utvrdi da

je tumor multifokalan, odnosno kada postoji više od jednog tumorskog fokusa, a bilateralan znači da je prisutan u oba režnja.



Slika 4 Matrica korelacije za bazu 2

Statističke vrednosti za numerička obeležja baze 2 date su u tabeli 3.

Tabela 3 Statističke vrednosti za bazu 2

Obeležje	Zdrave osobe			Bolesne osobe		
	Min	Max	Sr. vr.	Min	Max	Sr. vr.
Starost	20	78	51,13	19	77	42,8
Preoperativni .TSH	0.01	11.93	2.07	0.01	10,26	2,15
Najveci.fokus .Ca.mm	1	40	8,85	1	40	15,84
Broj.fokusa	1	7	1	1	6	2

Iz dobijenih rezultata ne uočavaju se značajne razlike između zdravih i bolesnih pacijenata. Selekcija obeležja za bazu 2 nije rađena, pošto je broj obeležja već smanjen na minimum. Rezultati dobijeni testiranjem modela nad različitim klasifikatorima dati su u tabeli 4.

Tabela 4 Evaluaciona metrika za bazu 2

Klasifikator	Tačnost	Senzitivnost	Specifičnost	Srednja apsolutna greška
kNN	67,07	75,0	59,52	24,41
DT	69,51	58,41	82,05	23,24
LR	76,83	76,7	77,08	10,91
RF	79,27	80,95	77,5	0,01
SVM	70,73	72,09	69,23	23,20
XGBoost	75,61	69,44	80,43	0,01

Na osnovu tabele 4 primećuje se da je klasifikator sa najboljim performansama RF, odnosno daje najveću senzitivnost što je najznačajniji ispitivani parametar metrike za ovaj problem.

5. ZAKLJUČAK

Preoperativna ultrazvučna dijagnostika nije dovoljno senzitivna u detekciji metastaza u limfnim čvorovima vrata kod pacijenata sa karcinomom štitaste žlezde, čime je onemogućeno adekvatno stažiranje bolesti. U našem istraživanju je na osnovu postoperativno dobijenog patohistološkog nalaza pokazano da su limfonodalne metastaze bile prisutne kod oko 48% pacijenata kod kojih je preoperativni ultrazvučni nalaz bio uredan.

Rezultat dobijen ovom studijom se pokazuje kao veoma dobar jer pruža senzitivnost oko 80%, sa približnom specifičnošću i tačnošću. Takođe, dobijeni su i demografski i patohistološki parametri koji su se pokazali kao najznačajniji za postizanje rezultata. Veća baza podataka, i univerzalna identifikacija ispitivanih parametara će dovesti do boljih rezultata u budućnosti.

6. LITERATURA

- [1] Andread C. Muller, Sarah Guido, *Introduction to Machine Learning with Python*, O'Reilly Media, Inc. 2016.
- [2] Kevin P. Murphy, *Machine Learning: a Probabilistic Perspective*, MIT Press, 2012.
- [3] Trevor Hastie, Robert Tibshirani, Jeronime Friedman, *The Elements of Statistical Learning*, Springer, 2009.
- [4] Martin Krzowski, Naomi Altman, *Classification and regression trees*, Nature America, Nature Methods, Vol. 14 No. 8, August 2017.
- [5] Jake Lever, Matrin Krzywinski, Naomi Altman, *Model selection and overfitting*, Nature America, Nature Methods, Vol. 13 No. 9, September 2016.
- [6] Candice Bentejac, Anna Csorgo, Gonzalo Matrinez-Munoz, *A Comparative Analysis of XGBoost.*, ResearchGate, November 2019.

Kratka biografija:



Nataša Avramović rođena je u Loznici 1997. godine. Osnovne akademske studije, na smeru Komunikacione tehnologije i obrada signala, završila je 2020. godine, sa odbranom bečelor rada pod naslovom „Praćenje kretanja i zadržavanja ljudi u zatvorenom prostoru primenom kompjuterske vizije”. Do sada je objavila jedan naučni rad na međunarodnom skupu TELFOR.

kontakt: avramovicnatas97@gmail.com



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.

Zahvalnica:

Izradu ovog rada pomogao je Fakultet tehničkih nauka u Novom Sadu, Departman za energetiku elektroniku i telekomunikacije, u okviru projekta pod nazivom „Razvoj i primena savremenih metoda u nastavi i istraživačkim aktivnostima na Departmanu za energetiku, elektroniku i telekomunikacije”.

ВИЗУЕЛИЗАЦИЈА И НАВИГАЦИЈА НАД ИСТОРИЈОМ ПРОМЕНА СОФТВЕРСКОГ ПРОЈЕКТА

SOFTWARE PROJECT CHANGES VISUALIZATION AND NAVIGATION

Милица Травица, Факултет техничких наука, Нови Сад

Област – СОФТВЕРСКО ИНЖЕЊЕРСТВО

Кратак садржај – Истражени и анализирани су *Atom* текстуални едитор, *CodeRibbon* пакет за навигацију (настао на основу *Patchworks* едитора) и визуелизација историје промена над пројектом. Имплементирана је визуелизација историје промена у оквиру пакета *CodeRibbon*.

Кључне речи: управљање конфигурацијом софтвера, *Atom*, *CodeRibbon*, визуелизација, навигација, историја промена

Abstract – The *Atom* text editor, *CodeRibbon* navigation package (based on the *Patchworks* editor) and visualization of project changes history were researched and analyzed. Visualization of project changes history has been implemented within the *CodeRibbon* package.

Keywords: Software Configuration Management, *Atom*, *CodeRibbon*, visualization, navigation, history of change

1. УВОД

Atom [1] је текстуални едитор. Он је *GitHub* пројекат отвореног кода и бесплатан је за коришћење. *Atom* омогућава додавање нових функционалности инсталирањем пакета, односно тема. Одређени пакети и теме се инсталирају приликом инсталирања самог едитора. Пакет који ће бити анализиран у раду је *CodeRibbon* [2].

CodeRibbon је имплементација *Patchworks* [3] пројекта за *Atom*. *Patchworks* је пројекат у оквиру ког је рађено на бржој и једноставнијој навигацији и организацији фајлова у којима се налази програмски код у интегрисаном развојном окружењу. Пројекат је реализован имплементацијом навигационе траке која садржи *Patch*-еве, у којима се налази програмски код.

Приликом развоја софтверског пројекта уз коришћење текстуалних едитора користе се и системи за контролу верзија (енгл. *Version Control System - VCS*). Они се могу користити независно од едитора али могу бити и подржани у њима. Програмери током развоја софтверског пројекта користе *VCS* јер им помаже у управљању изворним кодом, као и чувању сваке верзије пројекта. Такође, омогућава и сарадњу између програмера.

Циљ рада јесте да се истражи и анализира текстуални едитор *Atom* са акцентом на пакету *CodeRibbon*. Овај

пакет биће надограђен са додатном функционалношћу визуелизације и навигације над историјом промена. Конкретније, корисник ће имати могућност увида у личне промене над *Git* пројектима у оквиру *patch*-а, биће му приказан граф промена, као и детаљи о конкретној промени.

2. АНАЛИЗА И СТАЊЕ У ОБЛАСТИ

2.1 *Atom*

Atom текстуални едитор отвореног кода је бесплатан за коришћење. Он је направљен коришћењем *HTML*, *CSS*, *JavaScript* и *Node.js*, а заснован је на *Electron* радном оквиру. Могуће је користити га на *Windows*, *Linux* и *OS X* оперативном систему [1].

Atom нуди проширивост и приступачност [4]. Он је хакабилан и прилагодљив јер даје сваком кориснику могућност да мења сам едитор. Корисник може директно мењати код едитора или неког од његових пакета, како би побољшао своје корисничко искуство. Он даје могућност инсталирања нових пакета и деинсталирања постојећих. Такође, корисник може да креира нове пакете, који могу бити сачувани локално или подељени са осталим корисницима [5].

2.2 *Patchworks* и *CodeRibbon*

Patchworks је текстуални едитор који се разликује од других едитора по начину навигације фајлова. Навигација није традиционална, односно базирана на фајловима, а ни *canvas* базирана. *Patchworks* навигација је базирана на *patch grid*-у и траци. Овакав тип навигације помаже програмерима да брже пролазе кроз фајлове, троше мање времена и праве мање грешака у навигацији [6].

Patch је едитор у ком се налази фрагмент кода или фајл над којим могу да се врше измене. *Patch grid* представља главни видљиви део едитора, где могу да се врше измене кода. Разлика у односу на остале едиторе је што корисник истовремено има у видном опсегу 6 фајлова које може да мења, односно приказани су му као *grid* 3x2. Тренутно видљиви *grid* не приказује све отворене *patch*-еве (под условом да их има више од 6), али их трака садржи, односно *grid* приказује само део траке. Корисник има могућност да се креће кроз траку лево и десно и тако мења *patch grid* приказ. Поред тога има могућност да из птичије перспективе види целу траку или одређени део ње, што зависи од укупног броја *patch*-ева [3, 7].

У раду [6] описано је истраживање у којем се пореди *Patchworks* са едитором базираним на фајловима (*Eclipse*) и *canvas* базираним едитором (*Code Bubbles*).

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био др Игор Дејановић, ванр проф.

Петнаест испитаника, који су имали претходног искуства са *Eclipse* радним окружењем, је приступило овом истраживању. Нико није претходно користио *Code Bubbles* и *Patchworks*. Резултати овог истраживања су показали да: учесник за навигацију утрошио је приближно дупло мање времена коришћењем *Code Bubbles* и *Patchworks* едитора, најмање грешака је прављено коришћењем *Patchworks* едитора и испитаници су утрошили у просеку осам минута више у односу на *Patchworks* за отварање и измену кода ако су користили *Code Bubbles*.

Закључак који је изведен из рада [3] је да ни једна од четири утврђене стратегије не утиче значајно на брзину навигације програмера. Такође, примењено је и да су све стратегије које су примењене приликом навигације над *Patchworks* едитором брже од навигације *Eclipse* радног окружења. За истраживање је коришћен симулатор који је базиран на стварним подацима о навигацији програмера.

Рад [7] говори о истраживању које је извршено поређењем начина навигације 32 *LabVIEW* програмера коришћењем два различита едитора, од којих је први *Patchworks*, а други едитор са картицама. У раду су дошли до закључка да су коришћењем *Patchworks* едитора учесници направили мање кликова по навигацији него коришћењем едитора заснованог на картицама. Такође установили су пет принципа на које је потребно обратити пажњу приликом дизајнирања навигације за едитор кода.

У раду [8] је представљено решење које ће примењивати *Patchworks* начин навигације у другим едиторима. То се постиже креирањем пакета, односно *plugin*-а за већ постојеће, популарне и коришћене едиторе. Пакет допуњава функционалности едитора тако што омогућава да радни простор корисника буде попут радног простора у *Patchworks* едитору. Разлика у односу на *Patchworks* је што је *grid* подесиве величине. Овај пакет тј. *plugin* се зове *CodeRibbon*. Аутори ове идеје су креирали *CodeRibbon* за едиторе *Atom* и *VS Code*.

2.3 Управљање конфигурацијом софтвера

Управљање конфигурацијом (енгл. *Configuration management* - *CM*) је јединствена идентификација, контролисано складиштење, контролисана измена и извештавање о статусу производа, његових компоненти и радних верзија током животног циклуса система [9]. Традиционална дефиниција управљања конфигурацијом (софтвера) подразумева идентификовање конфигурације, управљање променама, праћење статуса и ревизију и верификацију [10]. Управљање конфигурацијом, на начин који се и данас користи, је настало крајем шездесетих година прошлог века. Посматра се као процес, јер је развој софтвера који се посматра као колекција међусобно зависних процеса имао утицаја на *CM*.

2.4 Системи за контролу верзија

Системи за контролу верзија (енгл. *Version control systems* - *VCS*) су део система за управљање конфигурацијом (софтвера). У чланку [11] се истиче да је коришћење система за контролу верзије приликом развоја пројекта једнако важно колико и

коришћење доброг едитора, интерпретера или компјлера. Говори се и о погодностима које доноси коришћење система за контролу верзије, попут измене датотеке без писања кода преко туђег, није потребно уступање датотеке и сл.

Типови приступа системима за контролу верзија су: централизовани систем за контролу верзија (енгл. *Centralized Version Control System* - *CVCS*) и дистрибуирани систем за контролу верзија (енгл. *Distributed Version Control System* - *DVCS*). Основна разлика између дистрибуираних и централизованих система је што дистрибуирани системи не захтевају централни репозиторијум. Сваки корисник има копију репозиторијума локално, на свом рачунару [12].

2.5 Git

Git је дистрибуирани систем за контролу верзија. Настао је 2005. године, а његов творац је *Linus Torvalds*. Настаје као ревизија система који је коришћен за развој *Linux* језгра [13].

У књизи [14] је објашњено да је *Git* имплементиран тако да складишти објекте, у којима се налазе оригиналне датотеке, подаци о аутору, датуми и друго. Постоје четири врсте објекта које могу бити складиштене, а то су: *Blob*, *Tree*, *Commit* и *Tag*.

2.6 Визуелизација *git* историје промена

Бољи преглед и лакше разумевање промена које су се десиле над пројектом омогућава добра визуелизација историје промена. *GitVS* једно је од решења за представљање визуелизације историја промена у *git* пројектима, који је описан у раду [15]. Из рада се може закључити да граф промена заједно са детаљима о *commit*-у представља једно од решења за овај проблем.

Програмерима приликом рада на пројекту може да буде од значаја да виде разлику између тренутне и неке од претходних верзија изабраног фајла. Омогућавање ове функционалности сматра се важним, о чему и говори [16]. Ту је описан *AZURITE plugin*, отвореног кода за *Eclipse* који пружа корисницима увид у историју промена над фајловима.

3. КОРИШЋЕНЕ ТЕХНОЛОГИЈЕ

Atom текстуални едитор је имплементиран коришћењем веб технологија. У почетку код је писан у *CoffeeScript*-у, али је касније већина кода пребачена у *JavaScript* [4]. С обзиром да су творци пакета *CodeRibbon* приликом његове израде користили *JavaScript*, он је коришћен и приликом додавања функционалности визуелизације и навигације над историјом промена. За имплементацију коришћене су и библиотеке *Vis.js* и *Dugite*.

3.1 Vis.js

Vis.js је библиотека за визуелизацију, која је оптимизована да ради са великом количином података. Компоненте које садржи ова библиотека су: *Network*, *Timeline*, *Graph3d*, *Graph2d* и *DataSet* [17].

Подаци се чувају у *DataSet*-у, у облику кључ-вредност. *DataSet* омогућава додавање, брисање и измену података, као и ослушкивање догађаја. Такође, пружа и могућност филтрирања, сортирања и конвертовања података [17].

Network се користи за визуелни приказ мрежа, које могу да садрже чворове и гране. Могуће је подешавати боју, величину, облике, стилове и друго. Може да ради са великим бројем чворова и грана. *Network* је подељен на неколико модула, који су задужени за управљање неким његовим делом. У питању су модули: *configure*, *edges*, *groups*, *interaction*, *layout*, *manipulation*, *nodes* и *physics* [17].

3.2 Dugite

Dugite повезује JavaScript, тј. *Node* пројекте са *git*-ом. Могуће је користити све команде које су доступне у *git*-у, што олакшава рад са *git* репозиторијумом из JavaScript пројекта. Ову библиотеку користи и *GitHub Desktop* апликација. Битни делови, које садржи *Dugite*, су: *GitProcess*, *IGitResult*, *IGitExecutionOptions*, *GitError*, *GitNotFoundErrorCode* и *RepositoryDoesNotExistErrorCode* [18].

4. СПЕЦИФИКАЦИЈА И ИМПЛЕМЕНТАЦИЈА

4.1 Спецификација

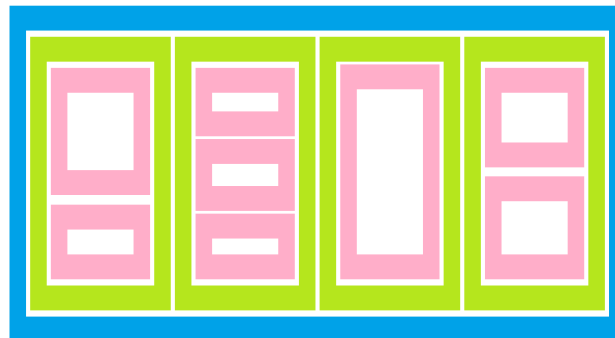
Начин навигације који је имплементиран у пакету *CodeRibbon* према истраживањима доприноси смањењу времена потрошеног за навигацију и мањем броју навигационих грешака. С друге стране системи за контролу верзија се користе на већини пројеката, *git* је један од њих. Јавља се потреба за бележењем стања пројекта у одређеном временском тренутку, као и увид у та стања. С обзиром на претходно, дошло се на идеју да би било пожељно у таквој навигацији имати и увид у историју над *git* репозиторијумом. Узимајући у обзир закључке који су изведени у поглављу Анализа и стање у области, створена је спецификација за функционалност визуелизације и навигације над историјом промена софтверског пројекта у оквиру *CodeRibbon* пакета. Она обухвата:

- Граф приказ историје промена на коме су приказани *commit*-и, везе између њих и референце. Иницијално учитани граф обухвата 50 последњих *commit*-а али постоји могућност учитавања и осталих *commit*-а. Корисник може да види који све *commit*-и спадају под одређену референцу. Такође, доступан је увид у детаље *commit*-а, као и у којим *commit*-ма је одређени фајл измењен.
- Приказ разлике између тренутне верзије изабраног фајла и неке од његових претходних верзија.

4.2 Имплементација

Полазни фајл *CodeRibbon* пакета је *code-ribbon.js* и налази се у *lib* фолдеру. Овај пакет је имплементиран тако да *CodeRibbonManager* представља полазни објекат за приказ навигације. Она садржи *grid* са произвољним бројем *patch*-ева и траком. У њему се налазе сви *patch* објекти који постоје у отвореној апликацији. Односно један од његових атрибута је *CodeRibbonRibbonContainer* објекат који заправо представља траку. Он садржи више *CodeRibbonSingleStrip* објеката који репрезентују део траке, у коме су *patch*-еви приказани један испод другог. Објекат који означава *patch* је *CodeRibbonPatch*. Он може да садржи више ставки али и ни једну. Слика 1. графички приказује начин

имплементације *CodeRibbon* пакета. Плавом бојом је представљен *CodeRibbonRibbonContainer*, зеленом *CodeRibbonSingleStrip*, а са розе *CodeRibbonPatch* објекат.

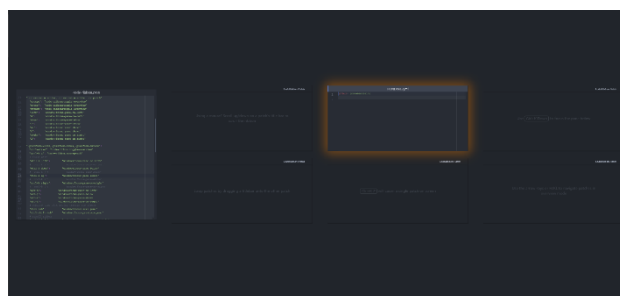


Слика 1. Графички приказ начина имплементације *CodeRibbon* пакета

За развој функционалности визуелизације и навигације над историјом промена креирани су фајлови који се налазе у *visualization* фолдеру. JavaScript фајлови који се у њему налазе су: *commit.js*, *file-diff.js*, *git-utils.js*, *graph-utils.js* и *network-graph.js*. Поред тога додати су одређени делови кода у већ постојећим фајловима. У оквиру *lib* фолдера то су фајлови: *code-ribbon-manager.js*, *code-ribbon-patch.js* и *code-ribbon.js*. У датотеци *git-utils.js* налазе се функције и константе за рад са *Dugite* библиотеком. У *commit.js* фајлу смештена је класа *Commit*. Класа *NetworkGraph* (*network-graph.js*) и класа *FileDiff* (*file-diff.js*) представљају нове ставке које могу бити приказане у *patch*-у.

5. ИЗГЛЕД АПЛИКАЦИЈЕ

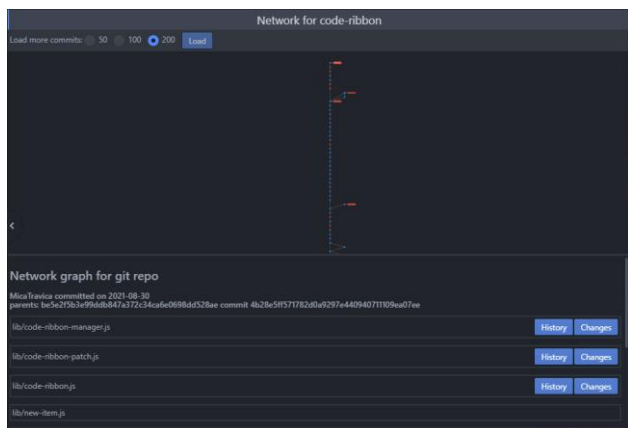
Слика 2. приказује удаљени приказ *CodeRibbon* навигације, односно целу навигациону траку. *Patch* који је тренутно у фокусу издваја се по бојама ивица, уоквирен је наранџастом. Могуће је променити *patch* на којем је фокус, изаћи из приказа целе траке, додавати нове колоне са лева или десна, као и обрисати колону. У *patch*-у који нема ставки могуће је користити брзу претрагу фајлова у оквиру пројекта.



Слика 2. *CodeRibbon* трака

Нова функционалност омогућава увид историје промена над пројектом, учитавање старијих измена, детаље о одређеном *commit*-у, приказ *commit*-а који спадају под одређену референцу, приказ *commit*-а у ком је мењан изабрани фајл и разлике између тренутне и неке од претходних верзија фајла. Слика 3. приказује историју промена за фајл *lib/code-ribbon-manager.js*. Са ње се могу видети детаљи о изабраном

commit-у, као и функционалности које постоје за мењане фајлове.



Слика 3. Визуелни приказ историје промена, историја промене фајла

6. ЗАКЉУЧАК

У раду је анализиран начин навигације у текстуалним едиторима и интегрисаним развојним окружењима. Текстуални едитор на који је стављен акценат је *Atom*, због своје проширивости. Истражен је начин навигације заснован на навигационој траци и *patch grid*-у, који је имплементиран у *Patchworks* едитору. Такође је истражен и *plugin CodeRibbon* који омогућава исту навигацију као и код *Patchworks* едитора. *Plugin* је реализован у едиторима *Atom* и *VSCode*. Изведени су закључци да граф промена заједно са детаљима о *commit*-у омогућава бољи преглед и лакше разумевање промена над пројектом и да увид у разлику између тренутне и неке од претходних верзија изабраног фајла корисници сматрају битним и корисним. На основу истраживања, анализа и донесених закључака утврђена је спецификација и имплементирана је додатна функционалност визуелизације историје промена *CodeRibbon* пакета.

Полазна тачка за даљи развој навигације и визуелизације над историјом промена софтверског пројекта може да буде једна од првобитних идеја. Она подразумева да се приказ разлика између две верзије истог фајла прикажу у једном *patch*-у, а у другом тренутна верзија тог фајла са могућношћу измене. Поред тога, приликом скроловања једног од *patch*-ева и други би био скролован. Како би се реализовала ова идеја потребно је изменити постојећи објекат у коме је имплементирана измена текста фајла или евентуално направити нови.

7. ЛИТЕРАТУРА

- [1] „Atom“, [На мрежи]. Доступно на: <https://atom.io/>. [Последњи приступ Септембар 2021].
- [2] „CodeRibbon“, [На мрежи]. Доступно на: <https://utk-se.github.io/CodeRibbon/>. [Последњи приступ септембар 2021].
- [3] A. Z. Henley, A. Singh, S. D. Fleming и M. V. Luong, „Helping Programmers Navigate Code Faster with Patchworks: A Simulation Study“, у 2014 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 2014.
- [4] „Atom Flight Manual“, [На мрежи]. Доступно на: <https://flight-manual.atom.io/>. [Последњи приступ септембар 2021].
- [5] L. B. a. A. S. M. K Sumangali, „A Comprehensive review on the open source hackable text“, у IOP Conf. Series: Materials Science and Engineering, 2017.
- [6] A. Z. Henley и S. D. Fleming, „The Patchworks Code Editor: Toward Faster Navigation with Less Code Arranging and Fewer Navigation Mistakes“, у Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2014.
- [7] A. Z. Henley, S. D. Fleming и M. V. Luong, „Toward Principles for the Design of Navigation Affordances in Code Editors: An Empirical Investigation“, у Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, 2017.
- [8] B. P. Klein и A. Z. Henley, „CodeRibbon: More Efficient Workspace Management and Navigation for Mainstream Development Environments“, 2021.
- [9] A. M. J. Hass, Configuration Management Principles and Practice, Addison Wesley, 2002.
- [10] B. Aiello и L. Sachs, Configuration Management Best Practices, Addison-Wesley, 2010.
- [11] D. Spinellis, „Version Control Systems“, Tools of the trade, pp. 108-109, 2005.
- [12] S. Otte, „Version Control Systems“, у Computer Systems and Telematics, 2009.
- [13] D. Spinellis, „Git“, Tools of the trade, pp. 100-101, 2012.
- [14] J. Loeliger, Version Control with Git, O'Reilly Media, Inc., 2009.
- [15] K. J. North, A. Sarma и M. B. Cohen, „Understanding Git History: A Multi-Sense View“, у Proceedings of the 8th International Workshop on Social Software Engineering, 2016.
- [16] Y. Yoon, B. A. Myers и S. Koo, „Visualization of Fine-Grained Code Change History“, у 2013 IEEE Symposium on Visual Languages and Human Centric Computing, 2013.
- [17] „Vis.js“, [На мрежи]. Доступно на: <https://visjs.org/>. [Последњи приступ септембар 2021].
- [18] „Dugite, документација“, [На мрежи]. Доступно на: <https://github.com/desktop/dugite/tree/e57cb95c17c411192b0c66397e6e84e19595dce8/docs>. [Последњи приступ септембар 2021].

Кратка биографија:



Милица Травица рођена је у Сомбору 1997. год. Мастер рад на Факултету техничких наука из области Електротехнике и рачунарства – Софтверско инжењерство, одбранила је 2021. год.
контакт: milica.travica@uns.ac.rs

AUTOMATSKO RASPOREĐIVANJE ELEMENATA DIJAGRAMA UPOTREBOM SUGIYAMA ALGORITMA, LSTM NEURONSKIH MREŽA I ELK BIBLIOTEKE**AUTOMATIC GRAPH LAYOUT USING SUGIYAMA ALGORITHM, LSTM NEURAL NETWORKS AND ELK LIBRARY**Nenad Mišić, *Fakultet tehničkih nauka, Novi Sad***Oblast - SOFTVERSKO INŽENJERSTVO I
INFOMACIONE TEHNOLOGIJE**

Kratak sadržaj – Ovaj rad predstavlja specifikaciju i implementaciju sistema za automatsko raspoređivanje elemenata dijagrama. Čvorovi predstavljaju komponente za obradu signala (signal processing) i energetske elektroniku (power electronics), pa je jedan od posebnih zahteva projekta da se raspoređivanje prilagodi tom slučaju korišćenja i na smislen način prikaže date dijagrame.

Ključne reči: vizualizacija, grafovi, automatsko raspoređivanje, sugiyama algoritam, LSTM, ELK

Abstract – This paper presents the specification and implementation of a system for automatic diagram layouting. Nodes are power electronics and digital signal processing components, so one of the special requirements of the project is to adjust the layouting to that use case and display the given diagrams in a meaningful way.

Keywords: visualization, graphs, automatic layout, sugiyama algorithm, LSTM, ELK

1 UVOD

Kompanija Typhoon HIL, lider u domenu energetske elektronike, proizvodi HIL (Hardware-in-the-loop) uređaje za testiranje energetskih postrojenja simulacijom u realnom vremenu. Deo ovog hardversko/softverskog proizvoda predstavlja interaktivni grafički editor (Schematic editor), čija je namena modelovanje kompleksnih energetskih sistema, vizuelno predstavljenih u obliku grafa komponenti.

Procenjeno je da oko 25% vremena pri kreiranju grafova biva potrošeno na manuelno raspoređivanje. Korišćenjem algoritama za automatsko raspoređivanje može se uštedeti vreme, umanjiti napor korisnika, preusmeriti pažnja sa raspoređivanja na modelovanje i znatno povećati čitljivost i razumljivost grafa.

U ovom radu predstavljeno je nekoliko pristupa za automatsko raspoređivanje grafova sa ciljem da se raspored što bolje uklopi u domen obrade signala i energetske elektronike i time rezultuje čitljivom, razumljivom i smislenom vizualizacijom. Prvi pristup, tzv. „naivni“, zasniva se na layer-based algoritmu razvijenom od strane Sugiyama et al. [1]. Drugi pristup koristi LSTM (Long-short term memory) [2] rekurentne neuronske mreže.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Igor Dejanović, vanr. prof.

Poslednji pristup, koji daje najbolje rezultate, oslanja se na ELK (Eclipse Layout Kernel) softverski alat [3].

2 TEORIJSKE OSNOVE**2.1 Grafovi**

Graf (V, E) je uređen par čvorova i grana, gde je grana predstavljena kao par (u, v) čvorova. U zavisnosti od toga da li je par (u, v) uređen ili ne, graf je usmeren ili neusmeren.

Crtež grafa Γ mapira svaki čvor v na jedinstvenu tačku $\Gamma(v)$, a svaku granu (u, v) na krivu $\Gamma(u, v)$ sa početkom u tački $\Gamma(u)$ i krajem u tački $\Gamma(v)$. Svaki graf može biti predstavljen sa više različitih crteža. Shodno tome, način na koji se graf iscrtava utiče na čitljivost, lakoću tumačenja i razumevanja, pa samim tim i na upotrebljivost.

2.2 Kriterijumi dobre vizualizacije grafa

Kako isti graf može imati više različitih crteža, postavlja se pitanje kako odrediti koji crtež je „dobar“ a koji ne.. Neke od glavnih smernica za iscrtavanje estetski ugodnih i čitljivih grafova su [5]:

- Minimizacija broja preklapljenih grana
- Uniformni tok
- Prave linije
- Optimalne dužine grana
- Uniformna gustina čvorova
- Uniformne dužine grana

2.3 Layer-based algoritam raspoređivanja

Algoritmi za automatsko raspoređivanje elemenata dijagrama se, na osnovu načina raspoređivanja čvorova, mogu podeliti u više grupa, kao što su stabla, kružni grafovi, force-directed, layer-based itd.

Raspoređivanje zasnovano na slojevima (layer-based) jedan je od najpopularnijih načina za raspoređivanje dijagrama koji su zasnovani na toku, kao što su dijagrami energetske elektronike ili digitalne obrade signala. Prvi rad koji se bavio ovom temom predstavljen je od strane Sugiyama et al. [1], a takođe, i većina novijih radova na temu layer-based pristupa pri raspoređivanju grafova zasniva se baš na ovom algoritmu. Ono što se srž ovog pristupa jeste da se svi čvorovi rasporede po diskretnim slojevima. Kom sloju će čvor pripadati diktiraju grane koje u njemu počinju ili se završavaju, a cilj je optimizovati raspored tako da se ispune estetski

kriterijumi kao što su uniformni tok podataka, uniformna gustina čvorova i uniformna dužina grana. Kada se svaki čvor smesti, računaju se njihove pozicije unutar sloja i na taj način se minimizuju preklapanja grana [1].

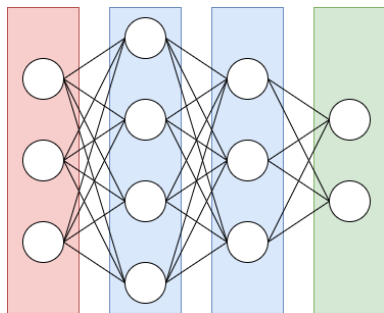
2.4 Neuronske mreže

Svrha neuronskih mreža je da aproksimiraju neku funkciju, koja je najčešće nepoznata ili nepogodna za rad (npr. nema izvod). Neuronske mreže se sastoje iz veštačkih neurona (perceptrona), modelovanih po uzoru na biološke neurone. Mnoštvo neurona, organizovanih po slojevima, funkcioniše zajedno i na taj način formira neuronsku mrežu koja može verodostojno da aproksimira matematičku funkciju.

Perceptron ima n ulaza preko kojih podaci dolaze do tela. Tu se dešava potencijalna aktivacija, tako što se svi ulazni impulsi množe sa korespondentnim težinama i međusobno sumiraju. Dobijena suma provlači se kroz aktivacionu funkciju, čiji je zadatak određivanje pobuđenosti neurona za date ulaze, i rezultat šalje na izlaz.

Neuronske mreže sastoje se iz slojeva veštačkih neurona. Svi slojevi su potpuno povezani (*fully connected*), odnosno ulazne vrednosti neurona sloja k su skup izlaznih vrednosti svih neurona sloja $k - 1$, sa izuzetkom prvog sloja čiji je ulaz ujedno i ulaz čitave neuronske mreže. Izlaz iz poslednjeg sloja predstavlja izlaz cele neuronske mreže. Ova osnovna arhitektura mreže u literaturi se često naziva *višeslojni perceptron*.

Na slici 2.2 prikazan je model višeslojnog perceptrona sa dva skrivena sloja. Neuronska mreža ima tri ulazna i dva izlazna perceptrona.



Slika 2.2. Prikaz modela višeslojnog perceptrona sa dva skrivena sloja.

2.4.1 Rekurentne neuronske mreže

Za razliku od klasičnih neuronskih mreža, odnosno višeslojnih perceptrona, koji obrađuju jedan po jedan ulaz, rekurentne neuronske mreže (Recurrent Neural Network, RNN) obrađuju sekvencu ulaza, usput zadržavajući kontekstne informacije. Ovo ih čini idealnim sredstvom za obradu sekvence podataka varijabilne dužine, kao što su video i audio snimci ili pak grafovi sa varijabilnim brojem čvorova. Rekurentne mreže rade slično poput klasičnih neuronskih mreža, sa razlikom da izlaz obrade elementa sekvence predstavlja jedan od ulaza za obradu sledećeg elementa. Na taj način je omogućeno čuvanje informacija u toku obrade. Glavni problem rekurentnih neuronskih mreža jesu takozvani nestajući i eksplodirajući gradijenti (*vanishing and exploding gradient*). Razlog pojave ovog fenomena je to što računanje greške na izlazu uključuje puno množenja težina sa aktivacionom funkcijom (posebno u slučaju

dugačkih sekvenci), pa se u praksi vrlo često koriste varijante koje rešavaju ovaj problem, kao što je *LSTM*.

Long short-term memory (LSTM) [2] neuronske mreže su specijalne rekurentne neuronske mreže, čija specifičnost leži u tome što, umesto jednog potpuno povezanog sloja, uvodi četiri nove komponente (gates) koje rešavaju pomenuti problem i utiču na izlaz mreže, na sledeći način:

- *Forget gate* – da li da se obriše trenutno stanje
- *Input gate* – da li da se prihvati ulaz
- *Gate gate* – koliko i na koji način treba prihvatiti ulaz
- *Output gate* – da li da se prosledi izlaz sledećoj iteraciji

2.5 Eclipse Layout Kernel

KIELER je istraživački projekat koji se bavi unapređivanjem dizajna složenih sistema zasnovanih na grafičkom modelu. Između ostalog, veliki deo istaživanja posvećen je automatskom raspoređivanju elemenata dijagrama. *Eclipse Layout Kernel (ELK)* predstavlja istraživački projekat baziran na *KIELER* projektu. Ukratko, cilj istaživanja predstavlja proširenje postojećeg layer-based algoritma za raspoređivanje da podržava port ograničenja, hijerarhijske portove i petlje (*self-loops*). *ELK* pruža relevantne implementacije za *Java* i *Javascript* tehnologije, a integracija u postojeći sistem je brza i jednostavna.

3 IMPLEMENTACIJA REŠENJA

Mrežni dijagrami, poput električnih kola, najčešće se iscrtavaju korišćenjem pristupa baziranog na slojevima (*layer-based*) za raspoređivanje čvorova i ortogonalnog pristupa za iscrtavanje grana [4]. Prema tome, prvi pristup u rešavanju problema automatskog raspoređivanja zasniva se na najpoznatijem *layer-based* algoritmu osmišljenom od strane Sugiyama et al. [1]. Drugi pristup koristi model *LSTM* neuronske mreže, dok poslednji, i pristup sa najboljim rezultatima, koristi funkcionalnosti *Eclipse Layout Kernel* biblioteke.

3.1 Pristup korišćenjem Sugiyama algoritma

Dok je originalni *Sugiyama* algoritam podeljen u pet faza, implementirani algoritam rešava problem automatskog raspoređivanja grafova u četiri uzastopne faze, a izostavlja petu koja se tiče iscrtavanja grana.

Uklanjanje ciklusa, kao prva faza algoritma, vrši se privremenim uklanjanjem jednog od čvorova iz ciklusa, za svaki ciklus koji postoji u grafu. Uklonjeni čvor biće iscrtan se u prvom mogućem sloju tako da indeks tog sloja bude veći od indeksa slojeva svih ostalih učesnika u ciklusu.

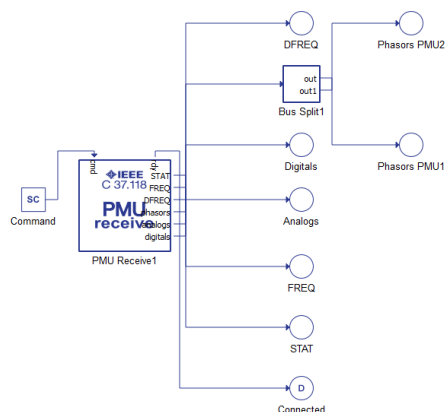
Dodela čvorova slojevima vrši se iterativno, tako što se u svakoj iteraciji pronalaze čvorovi čiji roditelji su već dodeljeni. U prvoj iteraciji, dodeljuju se korenski čvorovi, odnosno čvorovi koji nemaju roditelje, a s obzirom da graf ne sadrži cikluse, takvi čvorovi moraju postojati. Nakon toga, iteracije se nastavljaju sve dok se svim čvorovima ne dodele slojevi.

Minimizacija ukrštanja ispostavlja se kao najteži deo ovog problema, zbog ograničenja portova čiji su pozicija i redosled unapred određeni, zavisno od tipa komponente. Ovo je jedan od uočenih problema *Sugiyama* algoritma, i rešenje tog problema predloženo je u radu od strane

Schulze et al. [3]. Ovo rešenje iskorišćeno je u sklopu *ELK* alata, gde je jedan od glavnih ciljeva bio upravo rešavanje ovog problema.

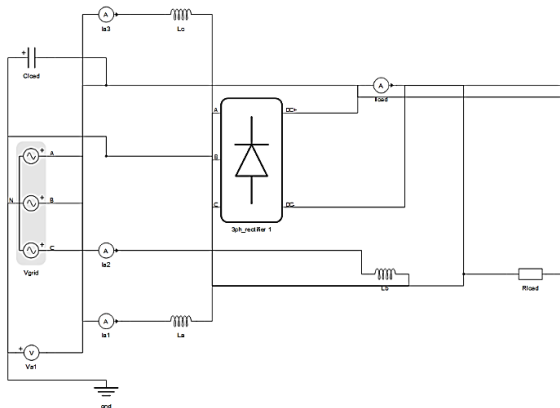
Određivanje pozicije čvorova je faza sa ciljem postizanja što većeg broja pravih linija podešavanjem vertikalnih pozicija čvorova. Algoritam implementiran u ovom pristupu inicijalno postavlja sve čvorove na vertikalnu poziciju koja je jednaka aritmetičkoj sredini vertikalnih pozicija roditeljskih čvorova, a potom iterativno vrši minimalna pomeranja čvorova sve dok postoje preklapanja između čvorova.

Jedan od najjednostavnijih primera za evaluaciju je model digitalne obrade signala prikazan na slici 3.1.



Slika 3.1. Prikaz jednostavnog modela digitalne obrade signala, raspoređenog korišćenjem Sugiyama algoritma

Još jedan od jednostavan primer za evaluaciju je model trofaznog ispravljača, prikazan na slici 3.2.



Slika 3.2. Prikaz modela trofaznog ispravljača, raspoređenog korišćenjem Sugiyama algoritma

Sa slika se može primetiti da su, zbog odsustva treće faze algoritma, čvorovi poređani tako da postoji dosta nepotrebnih preklapanja grana. Sa druge strane, vidi se da je generalni pristup dobar, da se *layer-based* raspoređivanje uklapa u domen problema i da postoji potencijal za kvalitetnu vizualizaciju.

3.2 Pristup korišćenjem LSTM neuronske mreže

Motivacija za ovaj pristup je postojanje skupa ručno raspoređenih grafova koji bi mogle da se iskoriste u svrhu treniranja adekvatnog modela. Taj skup sastoji se od 908 datoteka ručno raspoređenih grafova.

Format razmene podataka sa neuronskim mrežama mora biti u vektorskom, odnosno matričnom obliku, pa je za

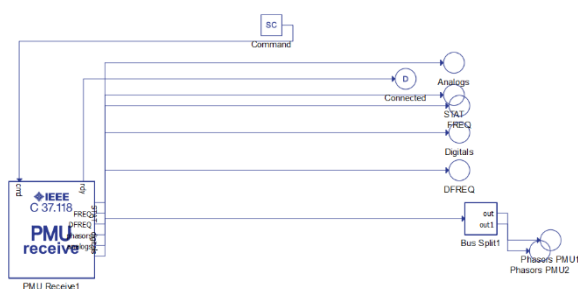
predstavu grafova iskorišćen format *matrice susedstva*. Naime, matrica susedstva grafa koji se sastoji od n čvorova je matrica dimenzije $n \times n$, gde je svaki čvor predstavljen jednim redom matrice. Polja tog reda mogu imati vrednosti 0 i 1, zavisno od toga da li postoji grana koja povezuje čvor na indeksu reda sa čvorom na indeksu kolone.

Pored ovih informacija, redovi matrice prošireni su dodatnim poljima kao što su broj terminala komponente, dimenzije, naziv komponente itd. Na taj način, broj čvorova grafa ograničen je na 100, a svaki red matrice je vektor dužine 135. Na osnovu ovog formata, model neuronske mreže u ulaznom sloju ima 135 neurona. Ulazi se dalje prosleđuju *LSTM* sloju dimenzionalnosti 512. Izlaz iz *LSTM* sloja ide u potpuno povezani (*dense*) sloj sa 2 neurona, kako bi se formirao izlaz iz cele mreže. Funkcija greške koja se koristi je prosečna kvadratna greška (*mean squared error, MSE*), a funkcija aktivacije koja se koristi je *Leaky ReLU*, jedna od standardnih varijanti za probleme regresije.

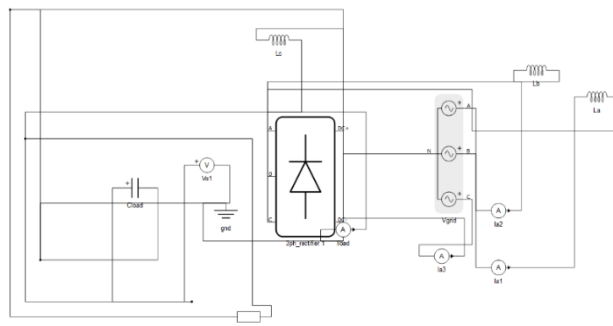
Nakon treniranja modela, jasno se vidi da rezultati nisu zadovoljavajući. Čvorovi grafova su razbacani bez puno smisla i uglavnom postoji ogroman broj preklapljenih grana. Ipak, model uspeva da razvoji komponente tako da nisu sve na sredini, i ta činjenica ulaze nadu da je problem manjak podataka za treniranje na osnovu kojih je mreža mogla da nauči težine.

Drugi potencijalni razlog neuspeha ove metode može biti prevelika dimenzionalnost za *LSTM* model. Ulaz u model je sekvenca vektora, svaki dimenzionalnosti preko 100, što zahteva dubok model kako bi se naučile kompleksne veze koje se kriju u podacima.

Na slikama 3.3 i 3.4 prikazani su isti apstraktni grafovi kao na slikama 3.1 i 3.2, raspoređeni istreniranim *LSTM* modelom.



Slika 3.3. Prikaz jednostavnog modela digitalne obrade signala, raspoređenog korišćenjem LSTM modela



Slika 3.4. Prikaz modela trofaznog ispravljača, raspoređenog korišćenjem LSTM modela

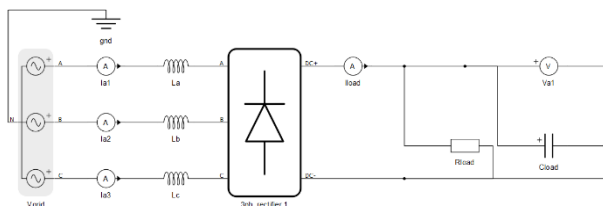
3.3 Pristup korišćenjem ELK biblioteke

ELK je alat za automatsko raspoređivanje grafova u sferi ugrađenog softvera kao i dijagrama električnih kola i ostalih grafova sa port ograničenjima [3]. Pored toga što se slučaj primene poklapa sa našim projektom, *ELK* biblioteka oslanja se na *Sugiyama* algoritam, koji, videli smo, daje solidne rezultate. Kako je ranije objašnjeno, odsustvo port ograničenja u *Sugiyama* algoritmu u velikoj meri je negativno uticalo na rezultate prvog pristupa. Očekivano je da ovaj pristup pruži čitljivije i razumljivije rezultate u odnosu na prvi pristup, sa manje preklapanja grana i više pravih linija.

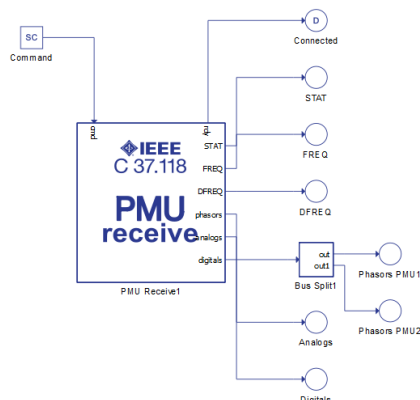
S obzirom da *ELK* nema relevantnu *Python* implementaciju, napravljena je jednostavna *Node.js* aplikacija čiji je zadatak da iščita datoteku u kojoj se nalazi tekstualni zapis grafa u *ElkGraph* formatu. Dobijeni objekat raspoređuje korišćenjem *ELKjs* biblioteke, a potom ga upisuje u izlaznu datoteku, u istom formatu. Odgovornost transformacije iz *Typhoon* formata u *ElkGraph* format prebačena je na *Typhoon HIL* softver, koji pokreće *ELK* servis kao podproces prosleđujući mu potrebne parametre korišćenjem *subprocess* modula. Ovakvom arhitekturom, postignuto je jednostavno upravljanje ali i željena portabilnost (imajući u vidu portabilnost *Javascript* aplikacija). Funkcionalnost automatskog raspoređivanja se vrlo lako može integrisati i u grafički korisnički interfejs.

Očekivano, rezultati su se ispostavili kao najbolji među ova tri pristupa. Raspored u velikoj meri podseća na raspored korišćenjem *Sugiyama* algoritma, ali zbog dodatnih podešavanja i ograničenja algoritam rezultuje kvalitetnijim rasporedom. Kod većine modela teško je uočiti razliku između automatski i ručno raspoređenog grafa.

Na slikama 3.5 i 3.6 prikazani su isti apstraktni grafovi kao na slikama 3.2 i 3.1, raspoređeni uz oslonac na *ELK*.



Slika 3.5. Prikaz modela trofaznog ispravljača, raspoređenog korišćenjem *ELK* biblioteke



Slika 3.6. Prikaz jednostavnog modela digitalne obrade signala, raspoređenog korišćenjem *ELK* biblioteke

4. ZAKLJUČAK

U ovom radu predstavljeno je nekoliko pristupa za automatsko raspoređivanje grafova sa ciljem da se raspored što bolje uklopi u domen obrade signala i energetske elektronike i time rezultuje čitljivom, razumljivom i smislenom vizualizacijom. Najbolje rezultate daje pristup koji se oslanja na *Eclipse Layout Kernel* biblioteku, a rezultujuće vizualizacije u velikoj meri podsećaju na ručno raspoređene grafove, što je i bio cilj istraživanja.

Dalje istraživanje moglo bi se proširiti u pravcu automatskog raspoređivanja čvorova koji predstavljaju komentare, s obzirom da ti čvorovi nisu direktno povezani sa ostatkom grafa. Takođe, dalje istraživanje moglo bi da ide u pravcu neuronskih mreža sa ciljem pronalaženja modela koji daju bolje rezultate u odnosu na *LSTM*.

5. LITERATURA

- [1] Sugiyama, Kozo, Shojiro Tagawa, and Mitsuhiro Toda. "Methods for visual understanding of hierarchical system structures." *IEEE Transactions on Systems, Man, and Cybernetics* 11.2 (1981): 109-125.
- [2] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.
- [3] Schulze, Christoph Daniel, Miro Spönemann, and Reinhard Von Hanxleden. "Drawing layered graphs with port constraints." *Journal of Visual Languages & Computing* 25.2 (2014): 89-106
- [4] Blackwell, Alan, Kim Marriott, and Atsushi Shimojima. "Diagrammatic Representation and Inference." (2004).
- [5] Purchase, Helen C. "Metrics for graph drawing aesthetics." *Journal of Visual Languages & Computing* 13.5 (2002): 501-516.

Kratka biografija:

Nenad Mišić rođen je u Kruševcu 1997. god. Master rad na Fakultetu tehničkih nauka iz oblasti Softverskog inženjerstva i informacionih tehnologija – Inteligentni sistemi odbranio je 2021.god.

kontakt: nenad.misic@uns.ac.rs

GENERATOR KODA ZA OPTIMALNU IMPLEMENTACIJU BINDCT TRANSFORMACIJE**CODE GENERATOR FOR OPTIMAL IMPLEMENTATION OF BINDCT TRANSFORM**Miloš Marković, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu biće predstavljeni DCT, BinDCT i HEVC. Prikazaćemo postupak transformacije leptira u transformaciju od tri koraka podizanja za $N=4$. Izložićemo glavni problem i dati rešenje za novi kod generator, njegove prednosti u odnosu na prethodno rešenje i rezultate njegovog korišćenja.

Ključne reči: DCT, BinDCT kod generator, HEVC, Binarna DCT, three lifting steps transformacije

Abstract – In this paper will be presented DCT, BinDCT, and HEVC. We will show the procedure of butterfly transformation into a transformation of three lifting steps for $N = 4$. We will present the main problem and give a solution for the new code generator, its advantages in relation to the previous solution, and the results of its use.

Keywords: DCT, BinDCT code generator, HEVC, Binary DCT, three lifting steps transformations

1. UVOD

Kompresija je često neophodna kako bi se omogućilo i olakšalo uvođenje novih tehnologija radi smanjenja potrebnog vremena za skladištenje i obradu podataka. U [1] predstavljen je novi kodek za kompresiju slika sa 32 b/p zasnovan na HEVC intra-kodovanju. Rad koristi BinDCT transformaciju [2] sa fleksibilnim veličinama blokova, koje variraju od 4×4 do 32×32 piksela; 12 metoda intra-predikcije, JPEG-XR univerzalnu šemu kvantizacije (sa povećanim rasponom parametara kvantizacije), i malo modifikovan CABAC na kraju, za efikasno entropijsko kodovanje.

Rezultati iz rada [3] značajno nadmašuju rezultate ranije predstavljenih šema kompresije. Analizom radova [2,3,4] došli smo na ideju da napravimo kod generator za binDCT transformaciju. U radu [4] opisana je binDCT transformacija sa *fiksni parametrima*. Motivacija za ovaj rad proističe iz činjenice da u postojećem kodu za BinDCT iz rada [4] nije bilo moguće da parametri binDCT transformacije budu *promenljivi* kako bi korisnik mogao da utiče na njenu implementaciju.

Problem je rešen tako što je napravljen univerzalni *kod generator* (dodatni program), koji generiše potrebni kod i omogućava da unošenjem p , u , p parametara dobijamo različite konfiguracije za binDCT i izaberemo najbolju.

Na taj način moguće je odrediti optimalnu konfiguraciju, što u ranijim implementacijama binDCT nije bio slučaj. Parametri se unose ručno u shifts.xml fajl, proizvoljnim

izborom i na osnovu tih parametara dobijaju se rezultati za tačno potrebnu vrednost parametra određene transformacije, što pre nije bilo omogućeno. Nakon parsiranja .xml fajla i podešavanja stanja, promenljivih i vrednosti promenljivih, kod generator čita template fajl (šablon). Kada pronade ključne reči za generisanje koda i iskoristi vrednosti p , u i p parametara koji su prethodno isparsirani generiše novi C kod odnosno generiše neophodne .cpp i .h fajlove.

Generisani fajlovi se potom mogu koristiti za tehnike kompresije zasnovane na binDCT, kao što je rad [4].

2. HIGH EFFICIENCY VIDEO CODING (HEVC)

HEVC, poznat i kao H.265 i MPEG-H 2 Part 2, trenutno je najnoviji i najnapredniji video standard, koji su zajednički razvili Video Coding Experts Group (VCEG) i Moving Picture Experts Group (MPEG), okupljeni oko jedne grupe Joint Collaborative Team on Video Coding (JCT-VC). U poređenju sa AVC-om, HEVC nudi od 25% do 50% bolju kompresiju podataka pri istom nivou video kvaliteta, ili značajno poboljšan kvalitet videa pri istoj brzini protoka. Podržava rezolucije do 8192×4320 , uključujući 8K UHD. HEVC koristi celobrojne DCT i DST transformacije sa različitim veličinama blokova između 4×4 i 32×32 piksela. Standard HEVC definiše sintaksu ili format komprimovane video sekvence i metod dekodovanja. Stvarni dizajn koda nije standardizovan. Podržava visoko fleksibilno razdvajanje video sekvence.

3. DISCRETE COSINE TRANSFORM

Discrete Cosine Transform ili DCT predstavlja diskretnu transformaciju koja se pokazala kao veoma pogodna za kompaktan zapis i spektralnu analizu signala, tako što očuvava ukupnu energiju signala u malom broju značajnih težinskih koeficijenata. U opštem slučaju definiše se korišćenjem aritmetike sa realnim brojevima. DCT je našla primenu u skoro svim algoritmima kompresije kao što su na primer: JPEG, MPEG, H.26X, MP3.

DCT kompresija je poznata i kao blok kompresija. Koristi blokove raznih veličina piksela počevši od 4×4 pa do 32×32 . Mnogi različiti brzi algoritmi za izračunavanje DCT-a, razvijeni su za primenu na fotografije i video zapise. Neki algoritmi baziraju se na retkim faktorizacijama DCT matrice, i mnogi od njih su rekurzivni. Kada je u pitanju obrada slika i video zapisa, da bi se podaci efikasno komprimovali neizostavno je potrebna kvantizacija. Diskretna kosinusna transformacija diskretne funkcije $f(j)$, $j=0,1,\dots,N-1$ definisana je kao:

$$F(k) = \frac{2c(k)}{N} \sum_{j=0}^{N-1} f(j) \cos \left[\frac{(2j+1)k\pi}{2N} \right]; k = 0, 1, \dots, N-1 \quad (1)$$

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentori su bili: doc. dr Branko Brkljač i dr Miloš Radosavljević.

Slično, inverzna diskretna kosinusna transformacija određena je sa:

$$f(j) = \sum_{k=0}^{N-1} c(k)F(k)\cos\left[\frac{(2j+1)k\pi}{2N}\right]; j = 0, 1, \dots, N-1 \quad (2)$$

$$c(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \\ 1, & k = 1, 2, \dots, N-1 \end{cases}$$

Transformacija poseduje svojstvo visokog energetskog zbijanja (kompakcije). Takođe poseduje i svojstvo linearosti, što omogućava laku primenu teorije linearnih sistema. Brzi računski algoritam za izračunavanje DCT opisan je u radu [3] gde se nalaze sve formule i računski koraci formiranja matrice [F].

4. BINARY DCT (BINDCT)

Diskretna kosinusna transformacija (DCT) uspešno je primenjena na kodovanje slika visoke rezolucije. Konvencionalna metoda implementacije DCT-a koristila je brzu Furijeovu transformaciju dvostruke veličine (2D FFT), koja koristi složenu aritmetiku tokom izračunavanja. Fast Discrete Cosine Transform – FDCT predstavlja efikasniji algoritam koji uključuje samo realne operacije za brzu diskretnu kosinusnu transformaciju skupa od N tačaka. Algoritam se može proširiti na bilo koju željenu vrednost ako je $N = 2^m, m \geq 2$.

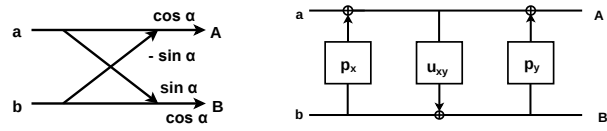
Generalizacija se sastoji u naizmeničnim leptirima matrice kosinusa i sinusa sa binarnim matricama za uređivanje elemenata matrice u oblik koji čuva prepoznatljiv obrnuti redosled bita (engl. bitreverse), raspored (engl. pattern) na svakom drugom čvoru. Generalizacija nije jedinstvena, otkriveno je nekoliko alternativnih metoda ali je metoda koja koristi $\frac{3N}{2}\log_2 N - 1 + 2$ realnih sabiranja i $N\log_2 N - \frac{3N}{2} + 4$ realnih množenja najjednostavnija za razumevanje. Ovakva metoda je približno šest puta brža od konvencionalnog pristupa koji koristi 2D FFT.

5. APROKSIMACIJA BRZOG RAČUNSKOG ALGORITMA

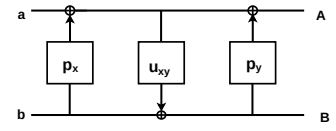
Aproksimacija brzog računskog algoritma pomoću p i u blokova objašnjena je u radu [2]. Na osnovu rada [2] možemo objasniti kako se leptir sa slike 1(a) u radu [2] prevodi u leptir koji je pogodan za transformaciju od tri koraka podizanja odnosno konverziju na p i u blokove. Oblik leptira koji je neophodno da bude ispoštovan kako bi bila dozvoljena transformacija od tri koraka podizanja izveden je iz formule (1) koja se nalazi u radu [2] i prikazan na slici 1. Prikazan leptir sa slike 1, može se konvertovati na sledeći blok dijagram od tri koraka podizanja prikazan na slici 2. Na datom blok dijagramu mogu se primetiti promenljive p_x, u_{xy}, p_y gde x i y predstavljaju indekse p, u i p blokova koji se koriste u transformaciji.

Pored promenljivih mogu se primetiti vrednosti signala a i b koji predstavljaju ulazne signale u blok za transformaciju, kao i signale A i B koji predstavljaju vrednosti izlaznih signala iz bloka za transformaciju.

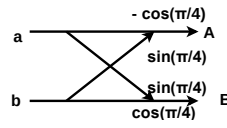
Transformacija od tri koraka podizanja ključna je za binDCT kod generator. Na slici 3 biće prikazan leptir koji se nalazi na slici 1(a) u radu [2], dok se na slici 4 nalazi konvertovan leptir koji je pogodan za transformaciju od tri koraka podizanja.



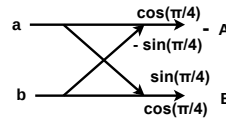
Slika 1 Prikaz osnovnog leptira „three lifting steps“ transformacije



Slika 2 Blok za transformaciju od tri koraka podizanja



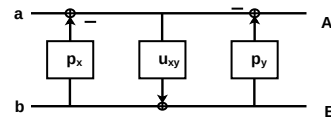
Slika 3 Leptir koji je rezultat postupka DCT transformacije



Slika 4 Konvertovan leptir sa slike 3

Neophodno je da leptir sa slike 3 konvertujemo u leptir oblika kao što je prikazan na slici 1. Konačno, leptir sa slike 4 možemo da prikazemo kao blok dijagram transformacije od tri koraka podizanja. Ovaj blok dijagram može se videti na Fig. 1. (a) iz rada [2].

Elementi p_x, u_x i p_y koji su prikazani na pomenutoj slici iz rada [2] predstavljaju elemente našeg blok dijagrama.

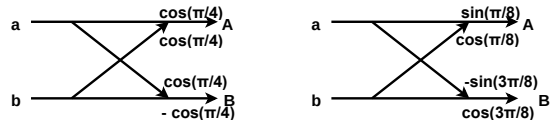


Slika 5 Prikaz transf. od tri koraka podizanja leptira sa slike 4

Prethodno opisan postupak transformacije leptira u transformaciju od tri koraka podizanja ključan je za binDCT kod generator koji je zasnovan na ovoj transformaciji.

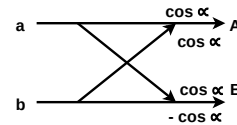
6. TRANSFORMACIJA $N = 4$

Ova transformacija sastoji se od dva leptira kao što je prikazano na slici 2 u radu [3]. Za ovu transformaciju možemo reći da imamo samo jedan prolaz odnosno stanje transformacije. Na narednoj slici biće prikazani leptiri tog stanja.



Slika 6 Prikaz leptira za transformaciju 4×4

Prvi leptir sa slike 6 može se prikazati kao na slici 7:



Slika 7. Prikaz prvog leptira transf. 4×4 u generalnom obliku

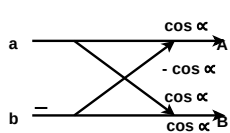
Potrebno je leptir sa slike 7 prevesti u oblik sa slike 1, pogodan za transformaciju od tri koraka podizanja. Leptir sa slike 7 možemo zapisati kao sledeći sistem jednačina

$$\begin{aligned} A &= a \cos \alpha + b \cos \alpha \\ B &= a \cos \alpha + b (-\cos \alpha) \end{aligned} \quad (3)$$

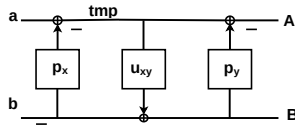
Zbog minusa koji može izaći ispred u drugoj jednačini, menja se izraz u prvoj jednačini na sledeći način:

$$\begin{aligned} A &= a \cos \alpha - b (-\cos \alpha) \\ B &= a \cos \alpha - b \cos \alpha \end{aligned} \quad (4)$$

Ovi izrazi mogu se predstaviti kao leptir sa slike 8.



Slika 8 Prvi leptir transformatije 4x4 u transformacionom obliku



Slika 9 Prikaz transformacije od tri koraka podizanja leptira sa slike 8

Leptir sa slike 8 u potpunosti odgovara leptiru prikazanom na slici 1. Konačno možemo napisati transformaciju od tri koraka podizanja leptira prikazanog na slici 8. Na slici 9 može se primetiti da postoje dva dodatna minusa u dijagramu, usled:

$$p = \frac{\cos \frac{\pi}{4} - 1}{\sin \frac{\pi}{4}} = -0,41421; u = \sin \frac{\pi}{4} = 0,70710 \quad (5)$$

Na osnovu slike 9 možemo napisati potrebne izraze transformacije, odnosno inverzne transformacije.

$$\begin{aligned} tmp &= a + b \gg p_x \\ B &= tmp \gg u_{xy} - b \\ A &= tmp - B \gg p_y \end{aligned} \quad (6)$$

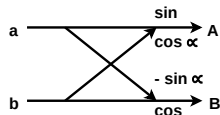
$$\begin{aligned} tmp &= A + B \gg p_y \\ b &= tmp \gg u_{xy} - B \\ a &= tmp - b \gg p_x \end{aligned} \quad (7)$$

U prethodnim izrazima može se primetiti operator $\gg$ koji nije standardni izraz za „pomeranje bitova udesno“ već predstavlja izraz koji kod generator koristi za označavanje zbira levih šiftova „ $<<$ “ pomerenih u desno „ $>>$ “.

Na primer, ako je $p_y = \frac{25}{32}$, *calculate_number_of_shifts* u kod generatoru vratiće: $(4 \sim 3 \sim 0)$ „5“, tj. broj pomeranja koji odgovara izrazu: $(2^4 + 2^3 + 2^0)/2^5$. Pošto u izrazu (7) imamo B koji se pomera za p_y kod generator će liniju u šablonu zameniti sledećom linijom koda:

$$tmp = A + (((B << 4) + (B << 3) + B) >> 5) \quad (8)$$

Preostali (drugi) leptir sa slike 6 za koji je potrebno izvršiti transformaciju možemo prikazati kao:



Slika 10 Prikaz drugog leptira transformatije 4x4 u opštem obliku

Potrebno je leptir sa slike 10 prevesti u oblik sa slike 1 koji je pogodan za transformaciju od tri koraka podizanja. Leptir sa slike 10 možemo zapisati kao:

$$\begin{aligned} A &= a \cos \alpha + b \sin \alpha \\ B &= a (-\sin \alpha) + b \cos \alpha \end{aligned} \quad (9)$$

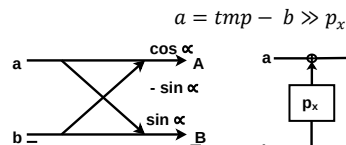
Zbog minusa koji je izašao ispred u drugoj jednačini menja se izraz u prvoj jednačini na sledeći način:

$$\begin{aligned} A &= a \cos \alpha - b (-\sin \alpha) \\ -B &= a \sin \alpha - b \cos \alpha \end{aligned} \quad (10)$$

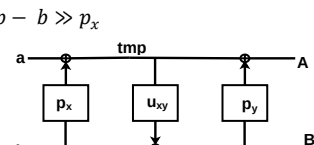
Prethodni izraz odgovara leptiru na slici 11, a koji odgovara leptiru prikazanom na slici 1. Konačno, možemo napisati transformaciju od tri koraka podizanja leptira. Na osnovu slike 12 sledi da su:

$$\begin{aligned} tmp &= a + b \gg p_x \\ B &= b - tmp \gg u_{xy} \\ A &= tmp + B \gg p_y \end{aligned} \quad (11)$$

$$\begin{aligned} tmp &= A - B \gg p_y \\ b &= tmp \gg u_{xy} + B \end{aligned} \quad (12)$$



Slika 11 Drugi leptir transformatije 4x4



Slika 12 Prikaz transformacije od tri koraka podizanja leptira sa slike 11

7. OPIS KOD GENERATORA

Zadatak kod generatora jeste da na osnovu vrednosti parametra p i u elemenata generiše C kod, odnosno da se generiše .cpp i .h fajlovi. Programsko rešenje je pisano u Python programskom jeziku. Konfiguracioni fajl za unos p i u parametara nalazi se u programskom rešenju na sledećoj lokaciji ../Configuration/shifts.xml.

Konfiguracioni fajl se sastoji od sledećih xml elemenata:

- `<shifts>` - predstavlja element u kojem se nalaze vrednosti za p i u elemente određenog stanja za transformacije 4x4
- `<shift_NxN class-name="ShiftNxN">` - predstavlja element određene transformacije gde je $N = 4$
- `<stage value="x">` - predstavlja stanje pojedine transformacije
- `<px>` - predstavlja vrednost p elementa određene transformacije
- `<ux>` - predstavlja vrednost u elementa određene transformacije.

Glavna prednost shifts.xml fajla jeste u tome što korisnik može ručno da unosi željene vrednosti parametara čime dobija rezultate za tačno njemu potrebnu vrednost parametra određene transformacije što mu pre nije bilo omogućeno. Programsko rešenje počinje svoje izvršavanje u fajlu BinDCT_CodeGenerator.py u kojem se nalazi main metoda. Unutar ovog fajla importuju se neophodni moduli koji su pisani unutar ovog programskog rešenja. Moduli koji su importovani jesu: *MyParser as MyP*, *MyCodeGenerator as MyCG*, *MyClass as Class*, *time*. *CalculateShifts* modul ima zadatak da izračuna vrednosti levih i desnih šiftova uz pomoć *calculate_right_shifts* (number) i *calculate_left_shifts* (number) metode. Glavni zadatak ovog modula jeste da u okviru *calculate_number_of_shifts* (value) metode na osnovu vrednosti parametra value vrati vrednosti levih/desnih šiftova za taj parametar što možemo videti u izrazu (8).

CalculateShifts modul ima zadatak da izračuna vrednosti levih i desnih „šiftova“ uz pomoć *calculate_right_shifts* (number) i *calculate_left_shifts*(number) metode. Glavni zadatak ovog modula jeste da u okviru *calculate_number_of_shifts* (value) metode na osnovu vrednosti parametra value vrati vrednosti levih/desnih šiftova za taj parametar što možemo videti u izrazu (8). *MyCodeGenerator.py* modul ima zadatak da pročita šablon koji se nalazi na lokaciji /Template/Template.cpp, čitajući liniju po liniju ovog šablona-a, proveriti da li linija sadrži ključnu reč za generisanje novog koda i da na osnovu ključne reči generiše potreban kod. Nakon generisanja koda, novokreirani kod je potrebno da upiše u fajl TComTrBinDCT.cpp na lokaciji ../Generated.

8. EKSPERIMENTALNA POSTAVKA

Potrebno je napraviti par različitih konfiguracija za BinDCT transformaciju od 4x4 tačke po koracima podizanja i odrediti p , u , p vrednost za svaki korak zadane transformacije. Primer parametara konfiguracije za jedan p parametar dat je u okviru tabele 1. Vrednost ugla

transformacije određujemo tako što na osnovu izračunatog p , u , p parametra tražimo najpribližniju vrednost našeg rezultata formule i beležimo koji ugao pripada tom rezultatu. Tako su uglovi za transformaciju od 4×4 : $\frac{\pi}{4}$ i $\frac{3\pi}{8}$.

TABELA 1 Primer parametara konfiguracije

	Floating-point	C1	C2	C3	C4
P_1	0,41421	$\frac{13}{32}$	$\frac{7}{16}$	$\frac{3}{8}$	$\frac{5}{16}$

Potrebno je dati konačne rezultate i izgled za par nama bitnih konfiguracija. C1 predstavlja najpribližniju konfiguracionu vrednost, dok C4 predstavlja jednu od „najudaljenijih“ konfiguracionih vrednosti. Kod generator na osnovu dobijenih konfiguracija treba da generiše rezultate na osnovu kojih možemo da analiziramo da li postavljeni parametri konfiguracije zadovoljavaju bitne vrednosti performansi i brzine kompresije.

9. REZULTATI

Za dobijanje rezultata iskorišćen je kodek koji je prethodno bio dostupan u okviru rada [1] i u kojem se inače koristi BinDCT transformacija. Pomenuti koder koristio je All-Intra (All-I) konfiguraciju. Performanse kodeka su procenjene korišćenjem standardnih RD krivih. Prosečna razlika između dve RD krive data je u obliku Bjøntegaard delta PSNR mere (BD-PSNR). PSNR predstavlja razliku između dve vertikalno poravnate (interpolirane) tačke iz različitih RD krivih. Relativno vreme kodovanja računa se pomoću izraza:

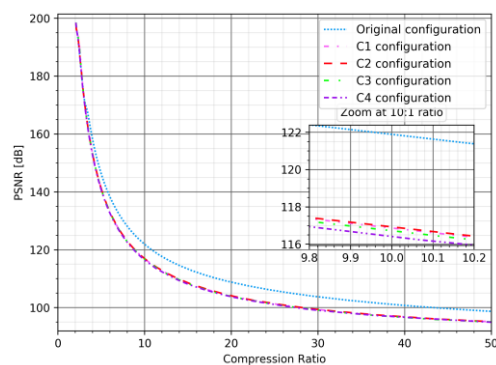
$$\Delta T = \frac{T_{testirano} - T_{referentno}}{T_{referentno}} * 100\%$$

gde je $T_{referentno}$ vreme kodovanja pomoću referentne verzije, a $T_{testirano}$ je vreme kodovanja pomoću konfiguracije čiju složenost želimo da uporedimo sa referentnom konfiguracijom [1]. Za $T_{testirano}$ uzimamo 4 različite konfiguracije (C1, C2, C3, C4), dok za $T_{referentno}$ uzimamo float vrednosti koje su izračunate direktno iz formule $[F] = \frac{2}{N} [A_N][f]$ koja predstavlja DCT transformaciju u matičnom obliku. U okviru tabele 2 možemo videti prosečne vrednosti rezultata konfiguracija za jednu seizmičku sliku.

TABELA 2 Rezultati konfiguracije za jednu sliku 1

Configuration	Encoding Time (EncTime)	Decoding Time (DecTime)	Average BD-PSNR [dB]
C1	-12.55769961	-0.47514177919	-1.8620
C2	-14.30100888	-0.29091856140	-1.9466
C3	-14.63641928	-1.1960734204	-1.8642
C4	-14.9005203	-0.85439783128	-2.5116

Pošto su razlike u parametrima konfiguracije relativno male, što se može videti u tabeli 1, to se takođe reflektovalo i na razlike u rezultatima datih konfiguracija. U okviru EncTime C1 daje najlošiji rezultat ubrzanja od -12.55769961 dok C4 daje najbolji rezultat ubrzanja od -14.9005203. U okviru DecTime C3 daje najbolji rezultat ubrzanja -1.1960734204 dok C2_ daje najlošiji rezultat ubrzanja od -0.29091856140. Što se tiče gubitka u BD-PSNR najbolji rezultat je onaj koji je što bliže 0, međutim plava kriva na slici 13 predstavlja originalnu konfiguraciju i od nje nema bolje. C1 od -1.8620 ima najmanja odstupanja od originalne krive dok C4 od -2.5116 ima najveća odstupanja od originalne krive.



Slika 13 Prosečne „rate-distortion“ (RD) krive za različite konfiguracije koda analizirane u tabeli 2.

Imajući u vidu EncodingTime, DecodingTime i PSNR možemo reći da su rezultati za C3 konfiguraciju najbolji i ona predstavlja traženi optimum. Cilj ovog testiranja nije bio dobiti što bolji PSNR već pokazati funkcionalnost predloženog kod generatora, kroz mogućnost testiranja 4 različite konfiguracije sa 4 različite preciznosti.

10. ZAKLJUČAK

U radu je opisan generator koda koji može da se koristi kao proširenje funkcionalnosti postojećih implementacija binDCT računskog algoritma. Za potrebe eksperimentalne potvrde rezultata kompresije koji se postižu korišćenjem konstruisanog generatora koda korišćen je primer kompresije seizmičkih slika, [2]. Zadatak predloženog kod generatora je bio da na osnovu vrednosti parametra p i u elemenata automatski napiše potreban izvorni kod u programskom jeziku C++, odnosno generiše .cpp i .h fajlove sa potrebnom strukturom i sadržajem, a koji će se koristiti za potrebna izračunavanja u okviru binDCT. Time je omogućeno da parametri binDCT budu promenljivi, što utiče na rezultate kompresije i omogućava izbor optimalne implementacije. Generisani fajlovi se bez dodatnih podešavanja mogu direktno koristiti u standardnim implementacijama algoritama za kompresiju koji koriste binDCT, kao što je npr. [2]. Analizom eksperimentalnih rezultata određeno je koja konfiguracija (generisani kod) daje najbolje rezultate, a koja ima velika odstupanja. Na taj način moguće je odrediti optimalnu konfiguraciju, što u ranijim implementacijama binDCT nije bio slučaj.

11. LITERATURA

- [1] M. Radosavljević, Z. Xiong, L. Lu, D. Hohl, and D. Vukobratović, "High bit-depth image compression with application to seismic data," Proc. VCIP'16, China, 2016.
- [2] J. Liang and T. D. Tran, "Fast multiplierless approximations of the DCT with the lifting scheme," IEEE Trans. Signal Process., vol. 49, no. 12, pp. 3032–3044, Dec. 2001.
- [3] W.-H. Chen, C. Smith and S. Fralick, "A fast computational algorithm for the discrete cosine transform," IEEE Trans. Commun., vol. 25, no. 9, pp. 1004–1009, Sep. 1977.
- [4] M. Radosavljević, A novel algorithm for high bit-depth seismic data compression, PhD thesis, Faculty of Technical Sciences, Novi Sad, 2021.

Kratka biografija:



Miloš Marković rođen je u Rumi 1993. Studije Energetike, elektronike i telekomunikacija, smer Obrada signala, uspešno je završio 2021.god. i stekao uslov za odbranu master rada. Kontakt: markovicnsmilos@gmail.com

**SOFTVERSKO REŠENJE ZA PODRŠKU LANCA NABAVKE ORGANSKE HRANE
ZASNOVANO NA BLOKČEJNU****BLOCKCHAIN BASED SOFTWARE SOLUTION FOR ORGANIC FOOD SUPPLY
CHAIN**

Stefan Veselinović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu će biti predstavljene mogućnosti primene blokčejn tehnologije za podršku lanca nabavke organske hrane i opisan razvoj jednog takvog sistema u vidu aplikacije na Hyperledger Sawtooth platformi. Aplikacija je implementirana na decentralizovan način i omogućava korisnicima da prate organske prehrambene proizvode kroz sve delove lanca nabavke.

Ključne reči: lanac nabavke, blokčejn, decentralizovana aplikacija.

Abstract – This work will present the possibilities for the application of blockchain technology in supporting organic food supply chains and describe the development of one such software solution using the Hyperledger Sawtooth platform. The application is implemented in a decentralized fashion, and it allows users to track organic food products through all parts of the supply chain.

Key words: supply chain, blockchain, decentralized application.

1. UVOD

Tema ovog rada jeste implementacija softverskog rešenja za podršku lancu nabavke organske hrane koje je zasnovano na blokčejn tehnologiji. Ideja je da se predstave prednosti i mane sistema baziranih na blokčejnu kao i razlozi zbog kojih blokčejn tehnologija ima dobru primenu u sistemima koji podržavaju lance nabavke. U daljem tekstu uvoda će biti dat pregled osnova distribuiranih decentralizovanih sistema i blokčejn tehnologije. U drugom poglavlju će biti dat detaljan opis najpoznatijih platformi baziranih na blokčejnu, Bitcoin-a [2] i Ethereum-a [3]. Treće poglavlje se bavi osnovnim konceptima lanca nabavke organske hrane, dok je u četvrtom poglavlju opisana Hyperledger Sawtooth platforma i njena arhitektura. U petom poglavlju su opisani korišćeni alati i tehnologije, dok poslednja dva poglavlja predstavljaju opis rešenja i zaključak.

Iako su centralizovani sistemi tipično jednostavniji za implementaciju nego decentralizovani, u slučaju otkaza bilo kog dela sistema ceo sistem postaje nedostupan, stoga su njihove glavne mane stabilnost i dostupnost. Distribuirani decentralizovani sistemi rešavaju glavne probleme centralizovanih sistema.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dušan Gajić, vanr. prof.

Korisnik distribuiranog sistema mora da ima utisak da koristi jedan koherentni sistem. Jedna od glavnih prednosti distribuiranih sistema jeste da se veoma dobro skaliraju. Vrlo često su delovi distribuiranih sistema na fizički različitim lokacijama što izuzetno poboljšava vreme odziva sistema. Najveći izazov pri dizajniranju distribuiranih sistema je sinhronizacija podataka. CAP [4] teorema govori da distribuirani sistem ne može istovremeno obezbediti dostupnost, toleranciju particija i konzistentnost podataka, usled čega se većina distribuiranih sistema deklarise kao eventualno konzistentna.

Decentralizovani sistemi su privukli veliku pažnju pojavom blokčejn tehnologije koja se prvi put pominje u radu koji opisuje Bitcoin platformu. Ova tehnologija se zasniva na tehnologiji distribuirane knjige (eng. Distributed Ledger Technology – DLT) [5], koja suštinski predstavlja distribuiranu bazu podataka. Svaki od čvorova ovakve baze podataka čuva identičnu kopiju svih podataka. Tri glavne karakteristike tehnologije distribuirane knjige su jezik transakcija, protokol i model podataka. Jezik transakcija definiše format putem kojeg čvorovi u sistemu mogu da vrše promene nad podacima. Protokol definiše pravila pomoću kojih učesnici sistema mogu da postignu konsenzus o tome koje transakcije je potrebno izvršiti i u kom redosledu. Model podataka definiše stanje distribuirane knjige i mora biti definisan pri dizajniranju sistema. Takođe, sigurnost podataka se postiže korišćenjem kriptografije. Svaki podatak koji se čuva u distribuiranoj knjizi sadrži vremenski otisak trenutka u kojem je poslednji put promenjen kao i kriptografski potpis.

Blokčejn predstavlja formu distribuirane knjige koja podatke čuva u lancu blokova, a svaki blok je skup transakcija koje su izvršene. Da bi se formirao lanac blokova, svaki od blokova sadrži kriptografski heš prethodnog bloka koji je smešten u zaglavlju bloka. Kriptografski heš predstavlja vrednost fiksne dužine koja je dobijena primenom heš funkcije nad određenim podacima. Pored ovoga, u zaglavlju bloka nalazi se i vremenski otisak, koren Merkleovog stabla i nounce vrednost. Tri glavne odlike sistema zasnovanih na blokčejnu su transparentnost, decentralizovanost i nepromenljivost podataka. Pod nepromenljivošću podataka se podrazumeva da jednom kada je transakcija izvršena i smeštena na blokčejn gotovo ju je nemoguće poništiti.

Osnovna podela sistema zasnovanih na blokčejn tehnologiji je na privatne i javne. Javnim sistemima može da pristupi bilo ko, dok je pristup privatnim sistemima omogućen samo autorizovanim korisnicima. Takođe, u

privatnim sistemima svi učesnici su poznati jedni drugima. Iako privatni blokčejnovi nude bolje performanse po pitanju brzine izvršavanja transakcija, većina najpoznatijih sistema kao što su Bitcoin i Ethereum se bazira na javnim blokčejn mrežama.

2. BITCOIN I ETHEREUM

Rad pod nazivom „Bitcoin: A Peer-to-Peer Electronic Cash System“ koji je 2008. godine objavljen od strane osobe ili grupe ljudi poznate pod pseudonimom Satoshi Nakamoto opisao je na koji način je moguće rešiti problem dvostruke potrošnje (eng. double spending problem) u sistemima u kojima učesnici nemaju poverenje jedni u druge, kao i način na koji je moguće čuvati izvršene transakcije pomoću blokčejna.

Bitcoin platforma koja je opisana u ovom radu korisnicima omogućava direktnu razmenu digitalne valute pod istim nazivom, bez potrebe za posrednikom.

Bitcoin mreža se održava kroz proces koji se naziva rudarenje (eng. mining). Svaki učesnik, odnosno rudar (eng. miner), sistema se takmiči sa ostalima učesnicima u rešavanju problema baziranog na kriptografiji sa ciljem da doda novi blok u lanac blokova. Takođe, on je taj koji bira koje transakcije iz skupa transakcija koje čekaju izvršavanje će uključiti u novi blok. Da bi rešio kriptografski problem rudar mora da pronađe numeričku vrednost takvu da kada se na ovu vrednost zajedno sa sadržajem bloka primerni heš funkcija, vrednost koja se dobije je numerički manja od zadate vrednosti. Vrednost koju rudar pokušava da pronađe naziva se nounce.

Glavna karakteristika ovog algoritma jeste da je veoma lako dokazati da je rudar pronašao odgovarajuću vrednost, dok pronalaženje iste zahteva veoma puno vremena i procesorske moći. Promenom zadate vrednosti od koje rezultat heš funkcije mora biti numerički manji, moguće je prilagođavanje težine pronalaženja nounce vrednosti, čime se postiže limitiranje broja blokova koji se dodaju u blokčejn. Tehnički gledano rudarenje predstavlja inverznu heš operaciju. Bitcoin mreža ograničava mogućnost dodavanja novog bloka u blokčejn na jedan blok svakih 10 minuta. Učesnik koji uspešno doda novi blok kao nagradu dobija određenu sumu digitalne valute Bitcoin. Opisani algoritam naziva se dokaz posla (eng. Proof of Work).

Dokaz posla je veoma lak za implementaciju, i moguće ga je prilagoditi velikom broju različitih sistema, uprkos tome algoritam ima nekoliko nedostataka koji su se pokazali ključnim. Da bi se mreža održavala kroz proces rudarenja potrebna je velika procesorska moć odnosno troši se velika količina energije, poslednja istraživanja pokazuju da Bitcoin mreža na godišnjem nivou potroši istu količinu električne energije kao Švedska. Osim toga, izvršavanje transakcija uključuje velika kašnjenja, pa je Bitcoin platforma u mogućnosti da obradi tek 4,6 transakcija po sekundi, dok recimo Visa platforma može da obradi preko 1700 transakcija u sekundi.

Sa ciljem rešavanja pomenutih nedostataka dokaza posla definisan je algoritam pod nazivom dokaz uloga (eng. Proof of Stake). Definicija ovog algoritma se prvi put pojavljuje pronalaskom kripto valute pod nazivom Peercoin. Učesnici u blokčejn mreži koja se održava putem dokaza uloga se nazivaju validatori. Svaki učesnik

je u mogućnosti da validira transakcije i uključi ih u blokčejn u vidu novog bloka ukoliko je izabran od strane sistema. Glavni parametar za izbor validatora je količina uloga. Ukoliko je validator A uložio 3 puta više nego validator B, validator A ima tri puta veću šansu da bude izabran. Teoretski, ovo znači da validatori koji ulože najveću količinu će biti uvek izabrani za validaciju. Vremenom bi na ovaj način mreža postala veoma centralizovana. Da bi se izbegla ovakva situacija, različite implementacije ovog algoritma uključuje dodatne parametre. Jedno od mogućih rešenja je da se uz količinu koju je uložio validator takođe posmatra i vreme koje je proteklo od kada je validator napravio ulog.

U radu koji je objavljen 2013. godine pod nazivom „The Ethereum“, Vitalik Buterin je prepoznao da je Bitcoin platforma samo jedna od mogućih primena blokčejn tehnologije. On je opisao sistem u kojem svaki od učesnika u mreži izvršava specijalan softver pod nazivom Ethereum virtuelna mašina (eng. Ethereum Virtual Machine - EVM). Ethereum virtuelna mašina omogućava izvršavanje aplikacija, koje su zbog prirode Ethereum platforme potpuno decentralizovane. Pre ovoga, da bi decentralizovana aplikacija bila implementirana potpuno nova blokčejn mreža je morala biti pokrenuta, dok je korišćenjem Ethereum platforme moguće razviti bezbroj različitih decentralizovanih aplikacija koje se izvršavaju na istoj blokčejn mreži. Ethereum trenutno koristi dokaz posla kao konsenzus algoritam, ali je u planu prelazak na dokaz uloga. Slično Bitcoin-u, Ethereum ima svoju kripto valutu pod nazivom Eter (eng. Ether), koja ima dvojaku ulogu. Ona služi kao valuta kojom je moguće trgovati, dok je programeri aplikacija koje se izvršavaju na Ethereum-u koriste za plaćanje naknade za izvršene transakcije. Dodatno, Ethereum definiše i token pod naziv Gas pomoću koga se meri koja količina računarske moći je potrebna za izvršavanje različitih operacija.

Revolucionarni koncept koji je Ethereum definisao je koncept pametnih ugovora (eng. Smart Contracts). Pametni ugovori predstavljaju programske kodove koji se kompajliraju u EVM bajtkod nakon čega se i izvršavaju na Ethereum virtuelnoj mašini. Koristeći pametne ugovore moguće je definisati operacije koje će se izvršiti kada su zadovoljeni određeni uslovi. Za pisanje pametnih ugovora Ethereum nudi nekoliko Turing kompletnih programskih jezika od kojih je najpoznatiji Solidity.

Korišćenjem pametnih ugovora moguće je definisati koncept decentralizovane autonomne organizacije. Ova vrsta organizacije nema lidera. Pomoću kolekcije pametnih ugovora se definišu struktura i pravila organizacije. Nakon što se pametni ugovori postave na Ethereum platformu, nije moguće menjati ni jedan od definisanih aspekata organizacije. U ekonomskom smislu, ovakva organizacija predstavlja kompaniju.

Decentralizovane finansije (eng. Decentralized finance – DeFi) je još jedan od veoma značajnih koncepta koji je nastao kao posledica Ethereum platforme i pametnih ugovora. Decentralizovane finansije podrazumevaju korišćenje sistema pomoću kojeg finansijski proizvodi postaju javno dostupni svima na korišćenje. Ovo znači da korisnici imaju mogućnost da direktno između sebe izvršavaju transakcije, što u potpunosti isključuje potrebu za posrednikom, kao što je na primer menjačnica kripto valuta. Decentralizovane finansije obezbeđuju korisni-

cima potpunu kontrolu i transparentnost nad svojim novcem, kao i pristup globalnim tržištima.

3. LANACI NABAVKE ORGANSKE HRANE

Lanac nabavke predstavlja system koji se sastoji iz organizacija, resursa, ljudi, aktivnosti i informacija koji su uključeni u snabdevanje potrošača proizvodima ili uslugama. Aktivnosti lanca nabavke uključuju transformaciju prirodnih resursa i sirovina u gotove proizvode, putem proizvodnih procesa, koji se dostavljaju krajnjim korisnicima.

Generički lanac snabdevanja počinje pronalaženjem resursa potrebnih za proizvodnju. Nakon toga resurse preuzimaju učesnici lanca koji predstavljaju dobavljača resursa u lancu nabavke. Sledeći korak je dostava resursa do proizvođača ili grupe proizvođača, koji će kreirati finalne proizvode, od strane dobavljača. Nakon toga distributer preuzima proizvode i dostavlja ih preprodavcima. Potrošači kupuju proizvode od preprodavaca time stvarajući potrebu za novim proizvodima.

Savremeno poslovno okruženje zahteva da se proizvodi izrađuju po željama kupaca, što brže i po što boljoj ceni. Pravilnim upravljanjem i konstantim unapređivanjem lanca nabavke kompanije ostvaruju uštede u vremenu i novcu. Upravljanje lancem nabavke obuhvata planiranje i upravljanje svim aktivnostima u lancu.

Lanci nabavke koji obuhvataju prehrambene proizvode su u velikoj meri transformisani u poslednjih 50 godina. Uprkos tome, moderni lanci nabavke prehrambenih proizvoda su i dalje veoma neefikasni. Prema podacima Ujedinjenih Nacija, više od milijardu tona hrane, na godišnjem nivou, propadne ili se baci. Lanci nabavke prehrambenih proizvoda su mnogo kompleksniji i teži za upravljanje od većine drugih lanaca nabavke jer je potrebno očuvati kvalitet i svežinu hrane, kao i njenu sigurnost.

Najveća tržišta organske hrane nalaze se u Severnoj Americi i Evropi, dok se većina proizvođača nalazi izvan ovih tržišta. Ovo predstavlja jedan od najvećih izazova u lancim nabavke organske hrane. Potreba za organskom hranom raste eksponencijalno iz godine u godinu jer krajnji potrošači pridaju sve veći značaj očuvanju zdravlja i okoline.

Takođe, proces proizvodnje organske hrane je mnogo kompleksniji od proizvodnje neorganske hrane. Većina država zahteva od proizvođača da imaju specijalne sertifikate za proizvodnju organske hrane, da bi mogli da obeležavaju svoje proizvode kao organske. Dodatno, organski proizvodi često imaju veoma kratak rok trajanja u kojem ih je potrebno upotrebiti, stoga je ključno dostaviti ih potrošačima u što kraćem vremenskom intervalu.

Problem koji krajnji potrošači imaju jeste da je, izuzimajući oznaku na pakovanju proizvoda, gotovo nemoguće razlikovati organski proizvod od neorganskog. Zato potrošači moraju da veruju proizvođačima i preprodavcima da proizvod jeste organski.

Upotrebom blokčejn tehnologije potrošači bi imali pristup kompletnoj istoriji proizvoda, koja uključuje način na koji je on napravljen, transportovan i skladišten, čime bi bili potpuno sigurni da je proizvod organskog porekla.

4. HYPERLEDGER SAWTOOTH

Hyperledger Sawtooth je jedan od alata koji je nastao kao rezultat Hyperledger projekta. Ovaj projekat je pokrenut krajem 2015. godine od strane Linux fondacije. Ubrzo nakon toga mnoge svetske kompanije priključile su se projektu. Cilj ovog projekta je obezbeđivanje skupa alata i biblioteka zasnovanih na blokčejn tehnologiji koji će biti korišćeni od strane velikih kompanija za implementaciju softverskih rešenja koja će unaprediti njihova poslovanja. Hyperledger Sawtooth projekat je pokrenut od strane Intel-a i verzija v1.0 je objavljena početkom 2018. godine. Ova blokčejn platforma je privatna i namenjena je za korišćenje od strane korporacija. Glavna karakteristika Hyperledger Sawtooth-a je da u potpunosti razdvaja jezgro sistema od dela sistema namenjenog za razvoj aplikacija. Na ovaj način ostvaruju se veoma dobre performanse sistema i povećava njegova sigurnost. Takođe jezgro sistema omogućava postojanje više aplikacija na blokčejnu u isto vreme.

Hyperledger Sawtooth je modularna platforma i njena arhitektura obuhvata komponente kao što su Validator, Procesor Transakcija, REST API i komponentu za upravljanje konsenzusom. Globalno stanje svih transakcija izvršenih u sistemu se čuva na svim validatorima u mreži. Ukoliko klijent sistema želi da promeni globalno stanje potrebno je da kreira transakciju i pošalje je validatoru. Najmanja jedinica koja može promeniti globalno stanje sistema je serija (eng. batch) transakcija, odnosno skup transakcija kod kojeg su ili sve transakcije iz skupa izvršene ili nijedna od njih nije izvršena. Takođe, korišćenjem serija transakcija pojednostavljeno je održavanje zavisnosti između transakcija, na način da za transakcije koje se nalaze u istoj seriji i jedna od njih zavisi od druge, nije potrebno eksplicitno definisati zavisnost jer će sve transakcije u seriji svakako biti izvršene. Hyperledger Sawtooth platforma podržava serijalno i paralelno izvršavanje transakcija. Takođe, tokom aktivnog rada sistema moguće je menjati način izvršavanja transakcija bez stopiranja sistema.

Hyperledger Sawtooth platforma nudi nekoliko konsenzus algoritama i neki od njih su dokaz o proteklom vremenu (eng. Proof of Elapsed Time - PoET), Sawtooth praktična vizantijska tolerancija grešaka (eng. Sawtooth Practical Byzantine Fault Tolerance – PBFT) i Sawtooth splav (eng. Sawtooth Raft). Konsenzus algoritam dokaz o proteklom vremenu rešava problem vizantijskih generala. Algoritam svakom učesniku dodeljuje nasumično generisanu vrednost koja predstavlja vreme koje je potrebno da sačeka pre nego što proba da validira sledeći blok. Suštinski, učesnik kome je dodeljena najmanja vrednost biva izabran za validaciju. Algoritam uključuje što veći broj učesnika u proces izbora sledećeg validatora i na jednostavan način je u mogućnosti da proveriti da li je svaki od učesnika čekao tačno vreme koje mu je dodeljeno pre nego što je započeo sa dodavanjem novog bloka.

Hyperledger Sawtooth platforma poseduje komponentu koja emituje događaje (eng. events) koji uključuju informacije o operacijama koje su izvršene u sistemu. Na primer, kada je novi blok dodat u blokčejn emituje se događaj i svi klijenti koji su se pretplatili na ovu vrstu

događaja dobijaju informaciju o tome. Dodatno, putem transakcionih priznanica (eng. transaction receipts) Hyperledger Sawtooth obaveštava klijente informacijama koje se tiču izvršavanja transakcija, ali se neće čuvati u globalnom stanju sistema, odnosno na blokčeju.

5. KORIŠĆENE TEHNOLOGIJE I ALATI

Softversko rešenje koje opisuje ovaj rad sastoji se iz nekoliko komponenti i bazira se na Hyperledger Sawtooth platformi. Za potrebe upravljanja zavisnostima između komponenti sistema kao i za jednostavno pokretanje koriste se alati Docker i Docker Compose. Docker je jedna od najpoznatijih platformi putem koje je moguće zapakovati aplikaciju u kontejner. Nakon toga pokretanjem Docker kontejnera na bilo kojoj platformi koja podržava Docker, pokreće se i aplikacija unutar kontejnera. Sve komponente sistema su zapakovane u zasebne kontejnere, i pomoću Docker Compose alata definisane su zavisnosti među njima. Takođe, Docker Compose alat pojednostavljuje upravljanje kontejnerima, odnosno omogućava pokretanje svih kontejnera odjednom, kao i njihovo stopiranje i restartovanje.

Komponenta u kojoj je implementirana poslovna logika aplikacije koristi Rust programski jezik. Rust je jezik visokog nivoa. Ima veoma sličnu sintaksu kao C++, ali za razliku od njega garantuje bezbednost memorije.

Podaci se osim u blokčeju čuvaju i u bazi podataka koju koristi serverska komponenta sa ciljem da ove podatke prikaže korisnicima. Izabrana baza podataka je Rethink. Ova baza podataka korisnicima nudi menadžmentsku konzolu u vidu veb aplikacije pomoću koje je moguće pristupiti podacima u bazi, ali i nadgledati performanse, ulazni i izlazni saobraćaj baze podataka.

Da bi se podaci o izvršenim transakcijama preneli u bazu podataka, implementirana je specijalna komponenta koja je pretplaćena na sve događaje koje emituje Hyperledger Sawtooth platforma. Ova komponenta je implementirana koristeći JavaScript programski jezik.

Serverska komponenta obezbeđuje REST API putem kojeg je moguće pristupiti podacima u bazi podataka. Implementirana je korišćenjem Node.js servera i JavaScript programskog jezika. Node.js je okruženje otvorenog koda, koja korisnicima omogućava da u JavaScript-u pišu alate komande linije kao i skripte koje će se izvršavati na serverskoj strani, odnosno van pretraživača.

Klijentska aplikacija je implementirana kao veb aplikacija korišćenjem JavaScript programskog jezika i Mithril.js biblioteke. Mithril je moderna JavaScript biblioteka namenjena za izradu veb aplikacija koje se sastoje iz jedne stranice (eng. Single Page Application). Ova biblioteka je mala i u poređenju sa drugim poznatim JavaScript bibliotekama veoma brza.

6. REŠENJE

Predstavljeno rešenje sastoji se iz nekoliko komponenti. Ulogu distribuirane baze podataka vrši Hyperledger Sawtooth blokčeju platforma. Sistem sadrži specijalizovanu komponentu koja podatke iz Hyperledger Sawtooth platforme prebacuje u bazu podataka. Serverska aplikacija koristi ovu bazu podataka da bi kroz REST API pružila pristup podacima unutar nje. Takođe, serverska aplikacija omogućava slanje zahteva za izvršavanje transakcija na

blokčeju. Poslovna logika koja vrši obradu ovih zahteva nalazi se u posebnoj komponenti, i implementirana je kao familija transakcija koristeći pametne ugovore.

Klijentska aplikacija komunicira sa serverskom aplikacijom kroz REST API sa ciljem da korisnicima omogući pristup podacima i slanje zahteva za promenu globalnog stanja sistema.

Sistem podržava sledeće funkcionalnosti: pregled liste svih organskih proizvoda, filtriranje liste proizvoda, dodavanje novog proizvoda koji će biti praćen kroz lanac nabavke, pregled podataka pojedinačnog proizvoda, ažuriranje podataka proizvoda koji se prate u lancu nabavke, pregled liste svih agenta prisutnih u sistemu, dodavanje novih i brisanje postojećih agenata.

Sistem podrazumeva upotrebu IoT senzora koji su zakačeni za svaki proizvod koji se prati i direktno koriste REST API da bi ažurirali podatke o proizvodu. Uloga IoT senzora je simulirana koristeći skripte napisane u JavaScript jeziku.

7. ZAKLJUČAK

Tokom izrade opisanog sistema Hyperledger Sawtooth platforma se pokazala kao veoma pogodna za implementaciju aplikacija koje podržavaju lance nabavke. Ona programerima omogućava da se skoncentrišu na implementaciju poslovne logike sistema, koristeći potpuni potencijal blokčeju tehnologije.

Glavna prednost primene sistema baziranog na blokčeju u implementaciji softvera za podršku lancu nabavke je što, ne samo korisnicima, već i svim ostalim učesnicima u lancu pruža potpunu transparentnost nad podacima i garantuje njihovu tačnost i sigurnost. Na ovaj način su korisnici sigurni u ono što kupuju, ali su na primer i preprodavci sigurni da prodaju proverene proizvode.

8. LITERATURA

- [1] Dr Dušan Gajić, *Materijali sa predmeta Paralelni i distribuirani algoritmi i strukture podataka*, dostupno na: <http://www.acs.uns.ac.rs/sr/node/237/4468699>, poslednji pristup oktobar 2021.
- [2] *Bitcoin*, dostupno na: <https://bitcoin.org/en/>, poslednji pristup avgust 2021.
- [3] *Ethereum*, dostupno na: <https://ethereum.org/en/>, poslednji pristup oktobar 2021.
- [4] *CAP*, dostupno na: https://en.wikipedia.org/wiki/CAP_theorem, poslednji pristup jul 2021.
- [5] *Blockchain Consensus*, dostupno na: <https://devopedia.org/blockchain-consensus>, poslednji pristup mart 2020.
- [6] *Distributed Ledger Technology*, dostupno na: https://en.wikipedia.org/wiki/Distributed_ledge, poslednji pristup mart 2020.

Kratka biografija:

Stefan Veselinović rođen je u Novom Sadu, Republika Srbija, 24. novembra 1994. godine. Osnovne akademske studije je upisao na Fakultetu tehničkih nauka Univerziteta u Novom Sadu 2013. godine. Diplomirao je 2017. godine.

GRAMATIKA GRAFIKE ZA VIZUALIZACIJU PODATAKA I NJENE IMPLEMENTACIJE**GRAMMAR OF GRAPHICS FOR DATA VISUALIZATION AND ITS IMPLEMENTATIONS**

Milica Damjanović, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *Grafički prikaz podataka ima značajnu ulogu u njihovoj analizi. U ovom radu su opisani osnovni koncepti gramatike grafike koja se uspešno koristi za vizualizaciju podataka. Predstavljene su komponente slojevite gramatike, slojevi i odgovarajuća preslikavanja podataka. Na primeru, implementiranom u programskom jeziku Python, demonstrirana je efektivna realizacija vizualizacije nad višedimenzionalnim podacima..*

Ključne reči: *gramatika grafike, vizualizacija, Python, analiza podataka*

Abstract – *Graphical data representation has significant role in its analysis. In this theses, basic concepts of grammar of graphics for data visualization are described. Research outline and explain components of layered grammar, layers and related data mappings. Example, implemented in Python programming language, demonstrate developing of efficient visualizations on multidimensional data.*

Keywords: *Grammar of graphics, visualization, Python, data analysis*

1. UVOD

Vizualizacija podataka predstavlja najoptimalniji način sortiranja i predstavljanja kompleksnih podataka. Jednostavan grafički prikaz može da bude koristan i da uštedi sate istraživanja. Grafici su laki za čitanje i interpretaciju, a ako su iz pouzdanih izvora, takođe ih možemo smatrat tačnim. Glavni cilj vizuelizacije podataka nije da čini podatke lepšim, nego pružanje korisnicima dolazak do skrivenih zaključaka u podacima, predstavljajući im ključne aspekte na intuitivniji, smisleniji i koncizniji način.

Vizualizacija podataka nije novina. Istorija vizualizacije potiče još pre 2.500 godina, kad nisu postojali ni računari ni alati za analizu sa vizuelnim rešenjima. Jedna od prvih poznatih vizualizacija je vizuelizacija koja prikazuje izbijanje kolere u ulici Broad Street u Londonu 1854. godine. Istraživač je pokušao da objasni statističkim podacima, a zatim i mapom, povezanost između kvaliteta izvora vode i slučajeva kolere. Mapa je pokazala da se najviše slučajeva kolere dešava u blizini pumpi sa vodom. Smatralo se da se kolera širi vazduhom, ali svojim vizuelnim prikazom podataka je uspeo da prikaže gde zaista nastaje problem.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bila dr Dunja Vrbaški, docent

2. RAZUMEVANJE GRAMATIKE GRAFIKE

U osnovi, gramatika grafike je okvir koji sledi slojevit pristup za opisivanje i konstruisanje vizualizacija ili grafike na strukturiran način. Vizualizacija koja uključuje višedimenzionalne podatke često ima više komponenti ili aspekata, a korišćenje slojevite gramatike grafike pomaže da se opiše i razume svaka komponenta koja je uključena u vizualizaciju u smislu podataka, estetike, razmere i objekata.

2.1. Značajne ličnosti koje su doprinele razvoju gramatike grafike

Originalnu gramatiku grafičkog okvira predložio je Leland Wilkinson [1] koja detaljno pokriva sve glavne aspekte koji se odnose na efikasnu vizualizaciju podataka. Wilkinson je početkom 1980-ih napisao SYSTAT, statistički programski paket. Ovaj je program poznat po svojoj sveobuhvatnoj grafici, uključujući prvu softversku implementaciju zaslona mapa topline (eng. heatmap), koji se danas široko koristi među biolozima. Vilkinsonova gramatika daje sažeta i sveobuhvatna pravila opisivanja objekata statističkih grafika koji uključuju podatke, transformaciju, skalu, koordinate, elemente, vodiče i prikaze kao i kako iscrtati ove objekte pomoću sistema iscrtavanja (slika 2). Njegova knjiga "Gramatika grafike" je takođe poslužila kao temelj za paket R ggplot2. Takođe se koristiti varijanta ovoga okvira, poznata kao slojevita gramatika grafičkog okvira, koju je predložio Hadley Wickham, renomirani istraživač iz oblasti nauke o podacima i osnivač poznatog R paketa za vizualizaciju ggplot2. Gramatika se razlikuje od Vilkinsonove po rasporedu komponenti, razvoju hijerarhije zadatih podešavanja, te zato što je ugrađen u drugi programski jezik.

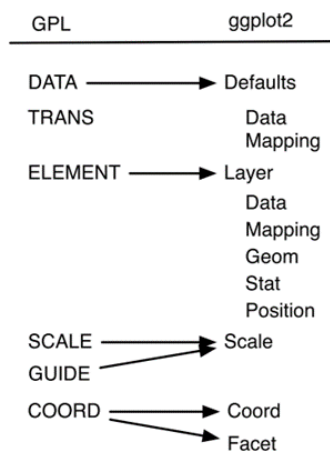
2.2. Komponente slojevite gramatike

Wickhamova slojevita gramatika grafike koristi nekoliko slojevitih komponenti za opis bilo koje grafike ili vizualizacije. Najvažnije je da ima neke varijacije u odnosu na originalnu gramatiku grafike koju je predložio Wilkinson. Preciznije, slojevita gramatika definiše komponente grafika kao skup sledećih elemenata:

- Podrazumevani skup podataka i skup preslikavanja promenljivih u estetiku.
- Jedan ili više slojeva, svaki sastavljen od geometrijskog objekta, statističkih transformacija, i podešavanje položaja, i opciono, skup podataka i preslikavanja estetike.
- Jedna skala za svako korišćeno preslikavanje estetike.
- Koordinatni sistem.
- Specifikacija podele pogleda.

Ove komponente visokog nivoa su prilično slične komponentama iz Vilkinsonove gramatike.

U obe gramatike komponente su nezavisne, što znači da možemo generalno menjati jednu komponentu izolovano. Postoji više razlika unutar pojedinačne komponente, koje će biti opisane u detaljima koji slede.



Slika 1. Mapiranje između komponenti Vilkinsonove gramatike (levo) i slojevite gramatike (desno)

2.3. Slojevi

Slojevi su odgovorni za stvaranje objekata koje opažamo na grafiku. Sloj je sastavljen od četiri dela: mapiranje podataka i estetike, statistička transformacija (stat), geometrijski objekat (geom) i podešavanje položaja.

Obično svi slojevi na grafiku imaju nešto zajedničko, što je tipično jer su oni različiti pogledi na iste podatke. Sloj je ekvivalent Vilkinsonovom elementu. Međutim, parametrizacija je prilično drugačija. U Vilkinsonovoj gramatici svi delovi elementa su isprepleteni, dok su u slojevitoj gramatici oni odvojeni

2.4. Podaci i mapiranje

Podaci su očigledno kritičan deo grafike, ali važno je napomenuti da su nezavisni od ostalih komponenti: može se konstruisati grafik koja se može primeniti na više skupova podataka. Podaci su ono što apstraktnu grafiku pretvara u konkretnu.

Uz podatke, potrebna je specifikacija koje se promenljive preslikavaju na koju estetiku. Izbor dobrog preslikavanja je ključan za generisanje korisne grafike.

2.5. Statistička transformacija

Statistička transformacija (*stat*) transformiše podatke, obično ih sažimajući na neki način. *Stat* uzima skup podataka kao ulaz i vraća skup podataka kao izlaz, pa *stat* može dodati nove promenljive u originalni skup podataka. Moguće je preslikati estetiku na ove nove promenljive. Geometrijski objekti, ili skraćeno geomi, kontrolišu vrstu grafika koji se kreira. Na primer, korišćenje tačkastog geoma će stvoriti raspršeni grafik, dok će upotreba linijskog geoma stvoriti linijski grafik. Geome možemo klasifikovati prema njihovoj dimenzionalnosti:

- 0d: tačka, tekst
- 1d: putanja, linija (uređena putanja)
- 2d: poligon, interval

Ponekad se mora prilagoditi položaj geometrijskih elemenata na grafiku kako u suprotnom ne bi prekrili jedan drugog. Vilkinson naziva ovo modifikatorima sudara.

2.6. Skale

Skala kontroliše preslikavanje podataka na attribute estetike, pa je potrebna jedna skala za svako svojstvo estetike koje se koristi u sloju. Skale su zajedničke po slojevima kako bi se osiguralo konzistentno preslikavanje podataka na estetiku.

Skala slojevite gramatike ekvivalentna je *skali* (*scale*) i *vodiču* (*guide*) Vilkinsonove gramatike. Postoje dve vrste vodiča: vodiči za skale i vodiči za beleške. U slojevitoj gramatici, vodiči se u velikoj meri automatski crtaju na osnovu opcije isporučene na relativnoj skali. Vodiči za beleške nisu neophodni ukoliko mogu biti kreirani sa kreativnom upotrebom geometrijskih objekata ako zavise od podataka, ili ako se osnovim sistemima za crtanje može se direktno pristupiti.

Skale se takođe računaju drugačije ukoliko je moguće preslikavanje promenljive proizvedene statistikom u estetiku. Ovo zahteva dva prolaza skaliranja, pre i posle statističke transformacije.

2.7. Koordinatni sistemi

Koordinatni sistem preslikava položaj objekata na ravan grafika. Položaj je često određen sa dve koordinate (x, y), ali može biti određen bilo kojim koordinatama. Dekartov koordinatni sistem je najčešći koordinatni sistem za dve dimenzije, dok se polarne koordinate i razne projekcije mape rede koriste. Za veće dimenzije postoje paralelne koordinate (projektivna geometrija), mozaičke grafike (hijerarhijski koordinatni sistem) i linearne projekcije na ravan. Koordinatni sistemi utiču na sve promenljive položaja istovremeno i razlikuju se od skala u tome što menjaju i izgled geometrijskih objekata.

2.8. Podela pogleda

Postoji i još jedna značajna komponenta koja se pokazala dovoljno korisnom i koju bi trebalo uključiti u opštem okviru: podela pogleda (aspektovanje, eng. facet). Ovo olakšava stvaranje malih višekratnika različitih podskupova čitavog skupa podataka. Ovo je moćan alat pri istraživanju da li obrasci važe za sve uslove. Specifikacija podele pogleda opisuje koje promenljive treba koristiti za podelu podataka i kako ih treba rasporediti u mrežu. U Vilkinsonovoj gramatici, podela pogleda predstavlja aspekt koordinatnog sistema, sa donekle komplikovanom parametrizacijom: promenljiva podele navedena je unutar komponente *element* i zasebni *coord* parametar specifikira da koordinatni sistem treba da bude podeljen ovom promenljivom. Ovo je pojednostavljeno u slojevitoj gramatici jer se podelapogleda vrši nezavisno od sloja i unutrašnjeg koordinatnog sistema. Manje je fleksibilno, jer se izgled pogleda uvek javlja u Dekartovom koordinatnom sistemu, ali u praksi nije ograničavajući.

2.9. Ugrađena gramatika

Prednosti ugrađivanja gramatike grafike u drugi programski jezik su očigledne: čovek odmah dobija sve postojeće sposobnosti tog jezika. Mogu se koristiti sve mogućnosti

potpunog programskog jezika za automatizaciju ponavljajućih zadataka: petlje za iteraciju preko promenljivih ili podskupova, promenljive za skladištenje često korišćenih komponenti i funkcija za enkapsuliranje i dekompoziciju uobičajenih zadataka. Nedostaci ugrađivanja gramatike su nešto suptilniji i u vezi su gramatičkim ograničenjima koja primenjuje jezik domaćin. Jedna od najvažnijih karakteristika gramatike je njena deklarativna priroda. Za očuvanje ove prirode u paketu ggplot2 koristi se operator + za kreiranje grafika dodavanjem delova zajedno. Funkcija ggplot stvara osnovni objekat, kome se dodaje sve ostalo. Ovaj osnovni objekat nije neophodan u samostalnoj gramatici.

2.10. Implikacije slojevite gramatike

Tri zanimljiva aspekta primene gramatike:

- Histogram, koji preslikava visinu linije na promenljivu koja nije u originalnom skupu podataka, i postavlja pitanja parametrizacije i zadatih vrednosti.
- Polarne koordinate, koje generišu tortne grafike iz trakastih grafika.
- Promenljive transformacije i tri mesta na kojima se mogu dogoditi vrednosti

Jedan od najkorisnijih nacrti za posmatranje jednodimenzionalnih (1-D) distribucija je histogram. Histogram je prilično poseban jer preslikava estetiku (visinu linije) u promenljivu stvorenu statistikom (broj binova) i otvara neka pitanja u vezi sa parametrizacijom i izborom podrazumevanih vrednosti. Jedan koordinatni sistem koji se vrlo često koristi u poslovnoj grafici je polarni koordinatni sistem, koristi se za izradu okruglih dijagrama i radarskih prikaza. Polarni koordinatni sistem parametrisuje dvodimenzionalna ravan u smislu ugla θ i udaljenosti od koordinatnog početka, ili poluprečnika, r . Postoje tri načina za transformaciju vrednosti korišćenjem ggplot2 biblioteke: transformacijom podataka, transformacijom skala i transformacijom koordinatnog sistema. Transformisanje podataka ili skala proizvodi grafike koji izgledaju vrlo slično, ali ose (i linije mreže) su različite: sve ostalo ostaje isto. To je zato što statistika radi na podacima koji su bili transformisani skalom.

3. IMPLEMENTACIJA

Na primeru jednog skupa podataka su prikazane vizualizacije dobijene korišćenjem gramatike grafike za opis prikaza. Sve vizualizacije su rađene u programskom jeziku Python. Plotnine je implementacija gramatike grafike, zasnovana na ggplot2. Gramatika omogućava korisnicima da sastavljaju grafike eksplicitnim preslikavanjem podataka na vizuelne objekte koji čine grafik. Uvek se počinje učitavanjem i posmatranjem skupa podataka koji se želi analizirati i vizualizovati. Koršćen je skup podataka koji sadrži informacije o prodaji računara.

Atributi koji čine skup podataka su:

- Company – tip string, kompanija koja je proizvela računar
- Product – tip string, model i oznaka računara
- TypeName – tip string, kog tipa je računar
- Inches – numeričkog tipa, veličina ekrana
- ScreenResolution – tip string, rezolucija ekrana
- Cpu – tip string, centralna procesorska jedinica
- Ram – tip string, radna memorija

- Memory – tip string, Hard Disk / SSD memorija
- Gpu – niz, grafička procesorska jedinica
- OpSys – tip string, operativni sistem
- Price_euros – numerički tip, cena izražena u eurima

3.1. 2-D vizualizacija podataka

PODACI: Izvor informacija

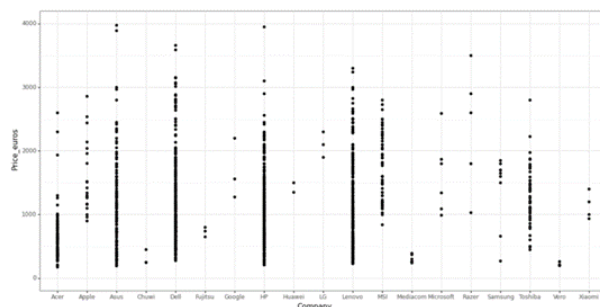
Prvi korak pri kreiranju vizualizacije podataka je određivanje podataka koji će se iscrtavati. Prilikom primene plotnine biblioteke kreira se ggplot objekat i prenosi se skup podataka koji se želi koristiti u konstruktor.

ESTETIKA: Definicija promenljive za svaku osu

Nakon što se navedu podaci koje je potrebno vizualizovati neophodno je definisati promenljive koja se žele koristiti za svaku osu na grafiku. Svaki red u okviru podataka može sadržati mnogo polja, pa se mora navesti koje promenljive se biraju za prikaz. Koristeći objekat ggplot, atribut kompanije se preslikava na horizontalnu grafičku osu, a cena izražena u eurima na vertikalnu osu (slika 2).

GEOMETRIJSKI OBJEKTI: Odabir različite vrste grafika

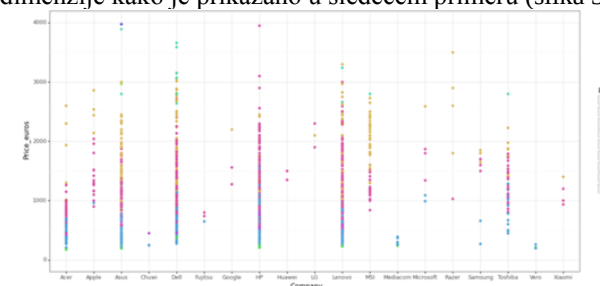
Nakon što se definišu svi podaci i atributi koji se žele koristiti na slici, mora se navesti geometrijski objekat koji će definisati način prikaza, odnosno kako treba iscrtati tačke podataka. Plotnine pruža mnogo geometrijskih objekata koji se mogu koristiti van okvira, poput linija, tačaka, šipki, poligona i još mnogo toga.



Slika 2. Prikaz cene računara po kompanijama

3.2. 3-D vizualizacija podataka

Za vizualizaciju tri dimenzije iz skupa podataka, možemo koristiti boju kao jednu od komponenti estetike za vizualizaciju jedne dodatne dimenzije pored druge dve dimenzije kako je prikazano u sledećem primeru (slika 3).

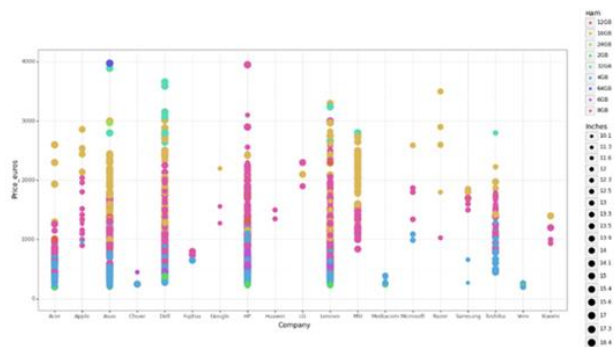


Slika 3. Prikaz cene računara po kompanijama uz dodatnu dimenziju predstavljenu bojom

3.3. 4-D vizualizacija podataka

Vizualizacija pokazuje koliko moćna estetika može biti u tome što pomaže da vizualizujemo više dimenzija

podataka na jednom grafiku. Da bismo vizualizovali četiri dimenzije iz skupa podataka, možemo upotrebiti boju i veličinu, kao dve estetike vizualizacije pored drugih regularnih komponenti. Za veličinu je izabran atribut koji nam daje informacije o veličini ekrana svakog računara u inčima (slika 4). Alternativno, može se takođe koristiti boja i podlea pogleda umesto veličine što je objašnjeno u predstavljanju primera 5-D vizualizacije.



DIZAJNIRANJE IMERSIVNOSTI VIRTUELNE SCENE POMOĆU POGONA IGRE DESIGNING THE IMMERSIVENESS OF A VIRTUAL SCENE USING GAME ENGINE

Ana Tomić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *Imersivnost postiže kod korisnika efekat kao da je zapravo prisutan u virtualnom okruženju čime je omogućeno bolje snalaženje u njemu. U ovom radu predstavljena je scena iz perspektive trećeg lica u pogonu igre Unity i uz pomoć programskog jezika C#, gdje je pokušano stvoriti efekat virtualne scene korištenjem virtualne kamere iz različitih uglova pri obavljanju određenih akcija. Rad takođe prati osnovne koncepte za dizajn imersivnosti scene u igrama. Pored toga, uključene su i generalne informacije o pogonima za igru, animaciji i načinu podešavanja virtualne kamere. Rezultat ovog rada je scena koja može da se nadogradi dodavanjem karakteristika za demonstraciju virtualne realnosti.*

Ključne reči: *Imersivost, VR (virtualna realnost), virtualna kamera, Unity*

Abstract – *Immersiveness achieves the effect of presence in virtual environment for the user, which enables better perceivability of it. This paper presents a scene from the third person perspective in the Unity game engine with the help of the C# programming language, where the attempt was to create the virtual scene effect using virtual camera from the different angles when performing certain actions. The paper also follows the basic concepts for immersiveness design of a virtual scene in games. In addition, general information of Game Engine, animation and the setting up virtual camera is included. The result of this work is a scene that can be upgraded by adding features to demonstrate virtual reality.*

Keywords: *Immersiveness, VR (virtual reality), virtual camera, Unity*

1. UVOD

Jedan od bitnih ciljeva koje svaka igra želi postići je natjerati igrača da zaboravi da drži kontroler u ruci ili da ima virtualni uređaj na glavi. U svrhu dostizanja ovog efekta, potreban je imersivni dizajn igre. Potpuna imerzija označava takvo čulno iskustvo da ne možemo razlikovati stvarni od virtualnog svijeta. Virtualna realnost predstavlja tehnologiju koja dozvoljava korisniku da ima interakciju sa računarski simuliranim okruženjem koja ima učinak na ljudska čula – vizuelna, slušna i taktilna – kako bi se pružilo iskustvo slično stvarnosti.

Prva stvar za stvaranje sistema virtualne realnosti jeste stvaranje virtualne scene. Virtualnu scenu treba izgraditi sa faktorima koji uključuju trodimenzionalni model, 3D

animaciju, video, zvuk, intuitivan korisnički interfejs i percepciju prostora da korisnici steknu utisak realnog okruženja. Ljudska vizija je jako osjetljiva pa za stvaranje ovakve scene i efekat dinamičkog prikaza u realnom vremenu, 3D model je najbolja opcija. Da bi se to postiglo, bitno je obezbijediti teksture, materijale za vegetaciju scene, zatim 3D modelovanje koje zahtijeva prethodnu analizu modela u svrhu korištenja što manje resursa za postizanje što boljeg efekta [1-3].

2. IMERSIVNI DIZAJN IGRE

Razvojem video igara, dizajneri istih kreiraju nove i unaprijeđene tehnike za povećavanje imersivnosti. Takve tehnike su najvažnije pri kreiranju kinematografske i emocionalno rezonantne interaktivne priče. Najbitniji faktori za imersivni dizajn su besprijekoran zvuk, zapanjujući vizuali koji se odnose na grafičku imersivnost kao i vjerodostojna gluma i upečatljiva priča što spada pod *Gameplay* imersivnost, a presudna je i tačka gledišta korisnika za potpuno uranjanje u svijet igre [4].

2.1. Grafička imersivnost

Grafička vjerodostojnost i umjetnički stil igre igraju ogromnu ulogu u imersivnom dizajnu. Iako igra ne mora imati fotorealističnu grafiku da bi bila sveobuhvatna, mora biti bez vidljivih vizuelnih grešaka tokom igranja. Greške kao što su preplitanje predmeta (eng. *Clipping*) igraču umnogome kvare utisak. Animirani umjetnički stil podjednako je sveobuhvatan sve dok postoji velika pažnja posvećena detaljima. Zapravo, igre sa unikatnim jedinstvenim stilovima imaju tendenciju da postanu sve imersivnije kako napreduje grafička tehnologija.

Dok igre ne postignu pravi fotorealizam, animirani umjetnički stilovi će izdržati test vremena. VR, na primjer, nudi jedinstvene grafičke mogućnosti kojima druge platforme ne mogu parirati. Grafička imersivnost sa VR-om isporučuje se slušalicama koje zamjenjuju vidno polje korisnika, omogućujući im da vide samo virtualni svijet oko sebe.

2.2. *Gameplay* imersivnost

Odnosi se na ideju da je stvarni čin igranja igre imersivan sam po sebi. Igra zasnovana na ritmu je imersivna jer se po taktu igre osjeća povezanost sa istom. Druge igre stavljaju akcenat na pokret i radnju koje intenzivno utiču na sam doživljaj. Što se radnje tiče, odlično pripovijedanje može nadoknaditi mnoge nedostajuće aspekte igre ili ih značajno ojačati ako su već dobri. U današnje vrijeme sve je teže doći do jedinstvenih i neviđenih ideja za priču, jer igrače privlače bogati, prostrani svijetovi i ubjedljive priče sa mnogo preokreta. Ispunjavanje ovih aspekata uz održavanje kohezivne priče umnogome poboljšava imersivnost. Likovi takođe

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragan Ivetić, red. prof.

igraju veliku ulogu u povezivanju igre sa igračem. Preko glavnog karaktera se komunicira sa svijetom igre. Vizuelni i audio znaci takođe su važni za imersivnosti, jer igrač mora da vidi i čuje u potpunosti karakter.

Iako većina igara ne sadrži glumce snimljene pokretom ili snimke kamere, ipak se mogu učiniti realističnijim sa dobrom glasovnom glumom. Čitanje tekstualnih dijaloga jednostavno nema isti efekat kao čuti nekoga kako govori riječi ispunjene namjerom i određenim emocijama poput straha, šoka, olakšanja.

2.3 Kreiranje realistične tačke gledišta (POV)

Tačka gledišta (eng. *Point Of View*) je jako bitan faktor za dizajn pri kreiranju digitalne igre. Tradicionalne opcije kamere uključuju publiku, izometriju, priču perspektivu, zadnju kameru, perspektive trećeg i prvog lica.

Mnoge igre danas su dizajnirane tako da igrač, u zavisnosti od ličnih preferencija može da promijeni perspektivu. Ista igra može stvoriti različita iskustva samo u zavisnosti od tačke gledišta kroz koju igrač vidi svijet igre. Perspektiva prvog lica se vjeruje da pruža najveći osjećaj imersivnosti za igrača, jer dozvoljava da igrač posmatra svijet očima karaktera. Posmatrajući izbliza svijet oko sebe, dajući jasan pogled scenografiji, angažman se može poboljšati dodavanjem realističnih detalja poput kretanja vode, pada sijenki itd. Alternativno, perspektiva trećeg lica nije optimalna poput prvog lica, jer izgleda kao da se samo kontroliše i posmatra karakter bez pružanja osjećaja igraču da je on zapravo taj karakter.

Iako ovakvo pozicioniranje kamere daje šire vidno polje okruženja, ipak otežava avataru da precizno izmjeri fokus interesovanja. Češće se ova perspektiva koristi za istraživanje i interakciju, kao i mogućnosti fokusiranja na posljedice sopstvenih akcije, dok je perspektiva prvog lica korisnija kada treba projektivna tačnost, ali se više fokusira samo na akcije izabrane od strane protagoniste. Tačka vizuelnog fokusa je bitna u igrama gdje igrač treba znati tačno gdje ciljati i biti u mogućnosti da fino podesi cilj.

I dalje je predmet rasprave koja od ove dvije perspektive čini igru imersivnijom, ali za potrebe ovog rada odlučeno je postići imersivnost za perspektivu u trećem licu pomoću virtuelne kamere.

3. UNITY POGON IGRE

Pogon igre je kolekcija modularnog koda koji ne upravlja direktno ponašanjem igre, ali obrađuje većinu „pozadinskih“ procedura poput unosa, izlaza i opšte fizike. Pogon igre čita podatke iz različitih vrsta datoteke i pretvara ih u formu spremnu za prikaz na ekranu. Npr., može teksture iz datoteke slike pretvoriti u materijal jer materijal ima referencu teksture.

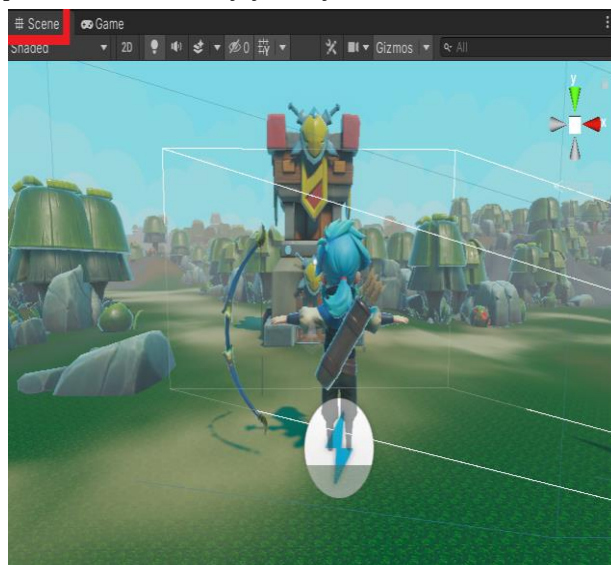
Najčešće korišteni alati za pravljenje igara za virtuelnu realnost su Unity3D i Unreal Engine. Za potrebe ovog rada izabran je Unity zbog jednostavnog IDE okruženja, fleksibilnog i modernog C# programskog jezika i mogućnosti portovanja na druge VR platforme. Pored toga, preporučuje se za razvoj niskobudžetnih video igara i posjeduju veliki broj gotovih rješenja od C# skripti do 3D okruženja i modela. Dizajn igre je prilagođen za desktop VR uređaje jer pružaju viši nivo interakcije i bolju hardversku podršku za razliku od mobilnih platformi. Postoji pet glavnih prikaza (eng. *View*) koji se

koriste u Unity editoru: Prikaz projekta, prikaz scene, prikaz igre, hijerarhijski prikaz i prikaz inspektora.

3.1. Prikaz scene

Prikaz scene (eng. *Scene View*) je jedan od najkorištenijih pogleda gdje se razvija vizuelni aspekt scene kao i mjesto gdje su postavljeni svi elementi igre, nakon čega im se može mijenjati pozicija, veličina ili rotacija pomoću strelica obilježenih oko objekta. Kroz prikaz scene, na jednostavan način organizujemo izgled čitavog nivoa igre. Scene se mogu posmatrati kao nivoi koji sadrže elemente igre – okruženje, karaktere, osvjetljenje, kamere i slično. Pokretanjem igre se zapravo pokreće određena scena.

U ovom prozoru se možemo slobodno kretati i tu se razvija vizuelni dio igre prevlačenjem objekata na scenu i njihovim podešavanjem. Na slici 1. prikaz scene je prikazan na levelu koji je dizajniran u 3D.



Slika 1. Prikaz scene

Funkcije koje se pozivaju kada se učitava scena jednom po svakom objektu u frejmu su: *Awake()* – poziva se odmah po inicijalizaciji prefabrikovanog modela sa svim komponentama osobinama i logikom, i *OnEnable()* – poziva se samo ukoliko je objekat aktivan. U virtuelnoj sceni igre koristik često mora da promijeni ugao observacije scene, kao npr. simulacija pri vožnji kada treba perspektiva i spolja i iznutra i za te potrebe treba podesiti kameru.

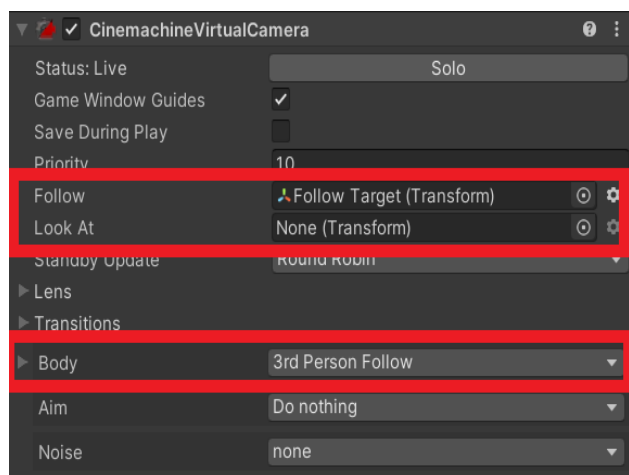
3.2. Virtuelna Kamera

Esencijalni dio predstave svake scene, pogotovo virtuelne u igri je kamera. U virtuelnoj sceni je najčešće postavljeno više kamera kako bi se postigla podjela ekrana, promjena scenarija, pravljenje male mape itd. Kamere mogu biti postavljene da renderuju scenu virtuelne realnosti u bilo kom poretku i u bilo kom uglu. Kretanje kamere i praćenje kamere mogu da naprave veliku razliku u korisnikovom doživljaju. Kamera je postavljena u perspektivnom režimu inicijalno za renderovanje scene i kao takva ima polje pogleda - FOV (eng. *Field Of View*) u obliku piramide. Vizija je jako bitan element za imersivnost, stoga u igrama je kamera preuzela njenu ulogu. Vizuelne informacije dobijene kamerom primjenom određenih alata mogu zavarati čovjeka misliti da je u realnom svijetu. To se postiže

Cinemachine funkcijama kao i pokretima kamere pri animaciji. Funkcija koja je izabrana za potrebe ovog rada je Virtuelna kamera (eng. *Virtual Camera*). Odabirom nje dodaju se dvije stvari: sama virtuelna kamera i komponenta glavne kamere *CinemachineBrain* koji upravlja svim aktivnim virtuelnim kamerama na sceni i omogućuje njihovu zamjenu, kontrolirše pogled kamere i zamućenost pri prelazu.

Za potrebe ovog rada aktivnoj kameri je naknadno dodata virtuelna kamera koja je podešena da prati igrača kao i da pri određenim akcijama zauzima drugačiji ugao u svrhu imersivnosti. Virtuelna kamera ima dvije opcije: „Prati” (eng. *Follow*) i „Pogledaj u” (eng. *Look At*) što možemo vidjeti na Slici 2. Podešavanjem na *Follow* virtuelna kamera prati igrača, a podešavanjem parametara „*Lookahead Time and Smoothing*” određujemo kojom brzom će se prilagoditi kamera igračevom pokretu.

Bitno je podesiti i „*Dead Zone width and Dead Zone Height*” koji igraču pružaju „mrtvi” prostor kojim se može kretati prije nego da kamera počne pratiti. Sa *Look At* moguće je kamerin fokus odrediti i stvoriti efekat zamućenosti za ostalo, takođe se to može postići i prilikom kretanja objekta koje kamera prati [5].



Slika 2. VC Follow/Look At

4. IMPLEMENTACIJA

Ideja ovog rada nije 3D modelovanje, stoga su za kreiranje scene poslužili gotovi elementi koje nudi prodavnica – *Unity Asset Store*. Za objekat glavnog karaktera izabran je besplatni 3D model avatara iz *3D male characters* paketa. Prije pisanja logike i postavljanja karaktera na scenu, poželjno je formirati početnu verziju okruženja u kojoj će se karakter kretati, te se dodao novi prazan objekat sa nazivom *Environment*. U okviru njega su se dodali svi ostali koji zajedno čine okruženje i u centar scene se postavi okruženje. Prazan prostor se popunio kreiranjem nove ravni, tj. *Plane* objekta.

Za kontrolu animacija i stanja objekata u Unity-u koristi se *Animator Controller* komponenta koju prevučemo preko objekta našeg karaktera da bismo mu je pridružili. U okviru modela objekta glavnog lika postavljena su sva njegova moguća stanja kojima se pridružuju odgovarajuće animacije, te je implementirana logika prelaska iz jednog stanja u drugo. Iz razloga što je akcenat na stvaranju virtuelne scene, u nastavku rada dio koji će biti detaljno

obrađen je konfiguracija virtuelne kamere koja umnogome utiče na imersivnost.

4.1. Virtuelna kamera za treće lice

Cinemachine je odličan alat u Unity-u koji obezbeđuje čitav sijaset pomagala i karakteristika koje pomažu u pravljenju funkcionalne i inteligentne kamere za igre i animacije. Kamera za treće lice zahtijevna je za napraviti jer se oslanja na pametnim kontrolama za čije dodatne *gameplay* funkcionalnosti koristimo *Impulse propagation* i *Blending*. U ovoj sceni imamo karaktera koji se kontrolirše tastaturom, cilja i ispaljuje strijelu. Inicijalno je kamera statična i karakter se kreće ispred nje što rezultira neinteresantnim pogledom na igru. Virtuelna kamera je kreirana tačno na istoj poziciji kao pogled na scenu što olakšava zauzimanje kadra. Kreiran je podobjekat *Follow target* koji je pozicioniran tik iza glave karaktera da se bolje uoči ciljanje strijele i njega dodjeljujemo *Follow* svojstvo što možemo da vidimo na Slici 2.

Lako može da se pomjeri distanca kamere i polje pogleda za bolje pozicioniranje sa svojstvom *Side-slider*. Podešavanjem *Z-damping* na *Third person follow* postigao se efekat da kamera juri karakter kada god se on kreće, a rotiranje kamere oko igrača je postignuto pravljenjem skripte koja uzima horizontalni i vertikalni unos i rotira metu koju pratimo, takođe proširen je pogled kamere i gore i dole čime je omogućeno gledanje okolo pri kretanju i rotiranje kamere oko karaktera pri njegovom mirovanju.

Za poboljšavanje imersivnosti moramo uključiti propagaciju impulsa (eng. *Impulse Propagation*) kojom će kamera reagovati na igračevu akciju. Osluškiivače impulsa podesimo na virtuelnu kameru, selektujemo objekat za koji želimo da generišemo impuls događaja, u ovom slučaju da luk i strijela pobude impuls čim ih karakter podigne u zrak i nadodamo *Cinemachine impulse source* komponentu koja emituje vibracioni signal iz tačke na našoj sceni. Efekat zamućenosti i blagog potresa kamere vertikalno i horizontalno pri odapinjanju strijele postizemo konvertovanjem parametara *x* i *y* krivih iz koordinatnog sistema realnog svijeta (eng. *World*) u koordinate kamere. Možemo podesiti amplitudu i frekvenciju događaja i prilagoditi prostorni rang impulsa. Svojstvo *Amplitude gain* utiče na jačinu izvornih impuls vibracija emitovanih od strane „sirovog” (eng. *Raw*) signala, dok svojstvo *Frequency gain* utiče na vrijeme pojave sirovog signala i kontrolirše brzinu vibracija. *Time envelope* kontrolirše vrijeme trajanja impulsa kao i intenzitet sirovog signala tokom tog trajanja. Ovim se kreira sveobuhvatni oblik u koji se signal uklapa. *Spatial range* određuje regiju kroz koju će se kretati impuls. Svaki osluškivač sa ovim rangom biće pogođen impulsom. Preko *impact* radijusa jačina signala polako slabi i neće dalje od toga se raširiti efekat punim intenzitetom amplitude.

Propagation speed kontrolirše koliko brzo događaj odašiljanja impulsa prođe kroz *impact radius*, što manja vrijednost to će postepenije se odvijati događaj. Neka preporučena vrijednost je brzina zvuka, tj. 343 metra u sekundi. Preostaje samo da kažemo impuls izvoru da se generiše svaki put kada strijela se odapne. To možemo postići kreiranjem reference u *ImpulseSource* skripti pozivajući *GenerateImpulse* metod kome je proslijeđen kamerin pravac unaprijed tako da je impuls signal rotiran od koordinata iz realnog svijeta u koordinate kamere. Zbog

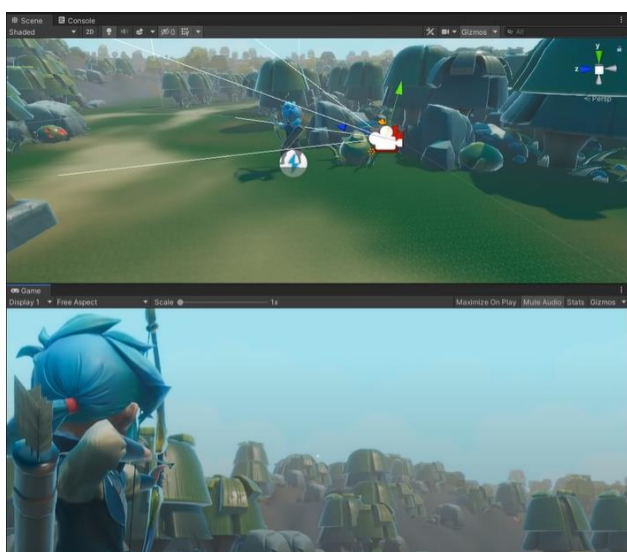
razlike u perspektivi kamere iza karaktera i perspektive gdje on cilja, desi se da strijela ne ode gdje smo mi mislili da hoće.

Nišan poravnavamo sa karakterovim pogledom koristeći *Third person aim* komponentu. Ova komponenta računa niz iz pozicije mete zajedno sa njegovim pravcem i postavlja drugi nišan koji prikazuje realan pravac. Poslednji dodatak koji umnogome dodaje na osjećaju imersivnosti jeste prilagođavanje kamere pri različitim aktivnostima. To postizemo kombinovanjem različitih virtuelnih kamera i sjedinjavanjem prelaza između različitih stanja kamere.

Na primjer, akciju kada karakter nišani, možemo da prikazimo na drugu virtuelnu kameru sa sličnim podešavanjima uz dodatak smanjenog polja pogleda i distance karaktera i napravimo skriptu koja mijenja kamere u zavisnosti od toga da li igrač cilja ili ne. To uradimo tako što provjerimo vrijednost unosa i aktiviramo korespondnu kameru. Kada pritisnemo dugme za ciljanje, uključi se jedna, a isključi druga. Tranzicija je podešena na 2 sekunde. Na slici 3. i 4. možemo da vidimo kako to izgleda:



Slika 3. Prikaz kamere kada igrač ne cilja



Slika 4. Prikaz kamere kada igrač cilja

Na taj način omogućeno je igraču da više uroni u svijet igre i stekne bolji osjećaj i za okolinu, kao i za procjenu preciznosti akcije koju obavlja.

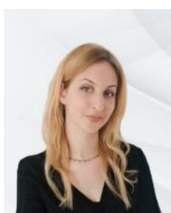
5. ZAKLJUČAK

U ovom radu predstavljeni su neki od osnovnih dizajna imersivnosti virtuelne scene. Upoznali smo se sa pogonom igre Unity i nekim od njegovih osnovnih koncepta. Pokazano je da nužno perspektiva pogleda ne utiče na imersivnost, tj. da perspektiva u trećem licu takođe može izgledati vrlo vjerodostojno dok god se imaju ideje kako približiti korisniku realnu predstavu svijeta. Primjer virtuelne scene u radu se fokusirao na mogućnosti virtuelne kamere i na koji način ona može doprinijeti imersivnosti. Kamera je vrlo važna za stvaranje virtuelne scene, te pravljenjem skripte za povezivanje kamera uvidamo da scena može poprimiti željeni efekat. Rad je podložan nadograđivanju npr. dodavanjem audio zvuka, kao i proširivanjem sa neprijateljima na sceni sa određenim logikama, kao i nivoima scene da bi bilo interaktivnije i podložnije za VR uređaj.

6. LITERATURA

- [1] Jaziar Radianti, "A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda"
- [2] Bowman D.A., McMahan R.P., "Virtual reality: how much immersion is enough?"
- [3] Chavez B., Bayona S. "Virtual reality in the learning process", Trends and advances in information systems and technologies, Springer International Publishing (2018)
- [4] Bruce Shelley (Ensemble Studios), "Guidelines for Developing Successful Games", Game Developers Conference 2001
- [5] WANG Xing jie, "Research and Application of 3D Virtual City Based on Unity3D", Computer Technology and development, Apr'2013.

Kratka biografija:



Ana Tomić rođena je u Vlasenici 1997. god. Osnovne studije završila je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Primijenjene računarске nauke i informatika. Master rad na istom fakultetu upisala je 2020. iz iste oblasti.

Kontakt: tomic.ana36@gmail.com

REPRODUKCIJA ZVUKA U VIRTUALNOJ 3D SCENI**SOUND RENDERING IN VIRTUAL 3D SCENE**Jelena Garić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – Cilj ove studije jeste procjena učinka 3D prostornog zvuka na korisnikov osjećaj imersivnosti u virtualnom okruženju, kao i istraživanje metoda kojima se poboljšava taj uticaj. Prostorni zvuk u 3D virtualnoj sceni treba da uvjerljivo postavi audio u VR prostoru tako da ga korisnik percipira kao da dolazi od stvarnih fizičkih objekata sa scene. Dobro postavljen prostorni zvuk korisniku pruža bolje poznavanje VR prostora simulacije i znatno pomaže u orijentaciji na sceni. Zvuk je jedan od osnovnih alata kojim se postiže osjećaj apsolutnog prisustva u VR-u, i u ovoj studiji to je demonstrirano u posebno dizajniranoj 3D VR sceni u Unity pogonu igre.

Ključne reči: Virtualna realnost, 3D prostorni zvuk, imersivnost, Unity

Abstract – The goal of this study is to evaluate the effect of 3D spatial sound on user's immersion feeling in virtual environment, as well as to explore the methods which improve this effect. Spatial audio should convincingly place sounds in a VR space so that the user perceives the sounds as coming from the real physical objects from the scene. Well placed spatial sound provides user a better knowledge of VR space of the simulation and it considerably helps with the orientation in the scene. Sound is one of the main tools to achieve the feeling of absolute presence in VR, and in this study, that is demonstrated by specially designed 3D VR scene in Unity game engine.

Keywords: Virtual reality, 3D spatial sound, immersiveness, Unity

1. UVOD

Za dobro dizajnirano okruženje virtualne realnosti, „prisustvo“ se smatra bitnim preduslovom. Kvalitetna reprodukcija zvuka je moćan način da se korisnik u potpunosti uroni u 3D VR prostor i da mu se pažnja sasvim usmjeri na dešavanja na sceni.

Ogroman dio naše pažnje može biti fokusiran audio znakovima, ali iskustvo potpune imerzije zahtijeva detaljnu kombinaciju prostora i zvuka, a ne samo znakova koji se dodaju kao naknadna ideja.

Izraz „zvučni pejzaž“ opisuje akustičko okruženje onako kako ga osoba doživljava i razumije (Međunarodna organizacija za standardizaciju, 2014). Zvučni pejzaž za virtualnu stvarnost može se sastojati od (a) pozadinske buke koja se koristi za stvaranje opšte atmosfere

(npr. vjetar, zvukovi saobraćaja, muzika) i (b) predvidivih i impulsivnih zvučnih događaja (npr. koraci, otvaranje vrata) [1]. Za ovu studiju, za pozadinsku buku koriste se zvukovi prirode u VR sceni - žuborenje potoka, ptice i zvuk iskre magičnih prozora scene. Za drugi element korišten je zvuk koraka glavnog karaktera simulacije.

U ovoj studiji procjenjuje se učinak zvuka na osjećaj imerzije korisnika, te na njegovu orijentaciju i snalaženje na VR sceni. Apsolutno prisustvo u VR-u bi značilo da korisnik ima utisak da se zapravo nalazi u virtualnom okruženju i ne shvata da posrednička tehnologija postoji, pa se ponaša kao da nema medija. Upravo iz tog razloga, za najrealnije i najuticajnije iskustvo, prikladno je da dizajn zvuka bude dio kreativnog sadržaja od samog početka jer loš ili pogrešno postavljen audio dizajn i znakovi mogu odvratiti korisnika od željenog, uvjerljivog ishoda VR scene.

2. PROSTORNI ZVUK

Prostorni zvuk čini ono što čujemo uvjerljivim slušnim iskustvom koje odgovara onome što vidimo i doživljavamo. Koncept prostornog zvuka zapravo je prisutan već duže vrijeme i obuhvata niz različitih audio tipova. Ljudski mozak interpretira zvučne signale na specifičan način koji mu omogućava donošenje odluka o okolini. Koristimo svoj sluh, zajedno sa sposobnošću kretanja, za donošenje boljih odluka o položaju audio signala i okruženju koje produkuje zvuk. U najosnovnijem, prostorni zvuk je jednostavno svaki zvuk koji nije mono [2] (tj. proizlazi iz jednog kanala). Stoga se stereo zvuk (zvuk snimljen i pomiješan u odvojenim lijevim i desnim kanalima) može smatrati prostornim zvukom. Drugi oblici zvuka koji se takođe mogu smatrati „prostornim“ u prirodi uključuju okolinski (surround) zvuk i binauralni zvuk. „Surround“ zvuk je način reprodukcije takav da je slušalac okružen zvučnicima, dok je binauralni zvuk snimljen tako da je bitna lokacija izvora zvuka u odnosu na uređaj koji snima, u toku samog snimanja.

Iako ove vrste zvuka zaista povećavaju iskustvo imersivnosti slušaoca u „head-locked“ okruženju, poput bioskopa, to je nedovoljno za potpuno efikasan doživljaj virtualne realnosti. To je zato što se korisnik u virtualnom svijetu kreće i gleda okolo kao u stvarnom svijetu. Ako se čuje ptica koja cvrkuće na drvetu s lijeve strane, trebalo bi da korisnik čuje taj zvuk najjače na lijevom uhu. Ali ako pomakne glavu da gleda direktno u pticu, zvuk bi trebao postati jednako uravnotežen u oba uha. Ovo je izazov s kojim se programeri suočavaju pri stvaranju prostornog zvuka za VR aplikacije. Ako se jačina zvuka i orijentacija ne promijene sa suptilnom promjenom položaja glave, to može duboko uticati na korisničko iskustvo.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragan Ivetić, red. prof.

Prostorni zvuk u virtualnoj realnosti uključuje manipulaciju audio signalima tako da oponašaju akustičko ponašanje u stvarnom svijetu. Precizna zvučna predstava virtualnog svijeta vrlo je moćan način za stvaranje uvjerljivog i sveobuhvatnog iskustva i zato prostorni zvuk ne služi samo kao mehanizam za dovršetak imerzivnog iskustva, već je i vrlo učinkovit kao element korisničkog interfejsa: zvučni znakovi mogu privući pažnju na različite žarišne tačke u naraciji ili privući korisnika da gleda određene dijelove simulacije, na primjer. Prostorni zvuk se najbolje može doživjeti upotrebom običnih slušalica - nisu potrebni posebni zvučnici, hardver ili višekanalne slušalice.

Cilj 3D prostornog zvuka je uvjerljivo smjestiti zvukove u trodimenzionalni prostor tako da korisnik percipira zvukove koji dolaze iz stvarnih fizičkih objekata u njihovom VR doživljaju. Iako se u VR-u samo fizički vidi ono što je direktno ispred korisnika, prostorni zvuk može ukazivati korisniku na ono što se vizualno događa iznad, ispod, iza i sa strane.

U prostornom zvuku, za razliku od stereo zvuka, zvuk je zaključen u prostoru. Ovo korisniku omogućava kretanje po prostoriji, a zvuk će ostati prostorno zaključen u okruženju. Da bi se generisao prostorni zvuk, mora postojati mogućnost kombinovanja zvuka s podacima o koordinatama koji će upozoriti sistem da opazi suptilne pokrete glave.

2.1. Ambisonic zvuk

Ambisonic Audio, iako zapravo nije nova tehnologija, postao je mnogo važniji nakon pojave VR -a i 360° videa. Najbolji način da se to zamisli je neka vrsta audio Skybox -a. U simulaciji (igri) bi Skybox stvorio iluziju 3D neba i udaljenog horizonta, koji se rotira pokretima glave igrača, ali da se nikada ne možete približiti ili udaljiti od njih, a nalazi se iza geometrije "stvarnog" svijeta.

Ambisonic Audio radi na sličan način. Kretanje svijetom neće promijeniti glasnoću zvuka, ali rotiranje glave mijenja ili barem mijenja jačinu četiri (ponekad i više) kanala u datoteci Ambisonic, zavisno u kojem smjeru gleda korisnik.

Rezultat je mnogo realističniji prikaz ambijentalnog zvuka, koji odlično funkcionira za ambijentalne zvučne pejzaže [2]. Jedan od najčešćih formata Ambisonic datoteka, B-Format, koji se ponekad naziva i Ambisonics prvog reda, sastoji se od četiri kanala. Jednostavno rečeno, četiri kanala predstavljaju 3 usmjerena kanala duž osi X, Y i Z i svesmijerni kanal W, koji sadrži snimak svih zvukova u Ambisonic sferi.

Potrebno je prvu vrstu zvuka ukombinovati sa podrškom za 3D zvuk (koji svi pogoni igre pružaju) kako bi se ostvario sasvim realan efekat odašiljanja prostornog zvuka iz izvora.

3. IMPLEMENTACIJA U POGONU IGRE UNITY

Simulacija virtualne realnosti je implementirana u Unity programu (verzija 2020.3.19f). Ona pruža gledaocu hod po prirodi u stilu RPG igrice sa jednim glavnim karakterom. Korisnik upravlja jednostavnom kontrolom kretanjem ovog karaktera (na računaru je to klik miša, što je analogno odgovarajućem kontrolnom dugmetu na svakom VR

headset-u). Komponenta virtualne kamere je definisana kroz objekat čija glavna komponenta jeste XR Rig (XR je spoj VR i AR tehnologija), iz XR Interaction Toolkit paketa. Ovaj paket dodaje Unity pogonu frejmwork koji apstrahuje sistem interakcije za kreiranje VR i AR iskustava, i baziran je na radu sa komponentama.

Ukoliko se simulacija želi testirati na nekom VR headset-u, potrebno je instalirati XR Plugin Management sistem, koji pruža opciju da se doda odgovarajući provajder za uređaj na kom se isprobava simulacija (Oculus, Windows Mixed Reality, ARCore...). XR Rig komponenta zahtijeva standardnu glavnu kameru i Camera Floor Offset objekat (služi za definisanje razmaka između posmatrača i glavne kamere). Na glavnu kameru dodata je Tracked Pose Driver komponenta, koja omogućava da glavna kamera prati pokrete headset-a - odnosno, ukoliko korisnik gleda na desno, kamera će takođe da se okrene u tu stranu.

U slučaju ove simulacije, kamera prati centar očiju na headset-u. Kamera je smještena tako da uvijek gleda između glavnog karaktera simulacije, i na nju je dodata skripta koja uvijek prati likove kretanje. Na početku je pozicija kamere definisana formulom:

$$i(x,y,z) = p(x,y,z) - t(x,y,z) \quad (1)$$

gdje je trodimenzionalni vektor i inicijalni offset VR kamere, p je trenutna pozicija kamere i t je pozicija predmeta koji se prati (*target*), u ovom slučaju karaktera kojim se upravlja u simulaciji. Svako naredno ažuriranje se dešava kretanjem karaktera i definisano je formulom:

$$c(x,y,z) = t(x,y,z) + i(x,y,z) \quad (2)$$

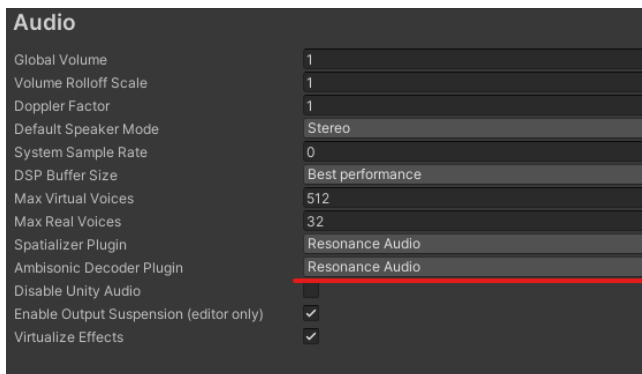
$$p(x,y,z) = t(x,y,z) + (c(x,y,z) - t(x,y,z)) \cdot d$$

gdje je vektor c pozicija kamere u trenutku ažuriranja i računa se kao zbir pozicije praćenog predmeta i inicijalnog offeta kamere. Zatim se pozicija VR kamere postavlja na interpoliranu vrijednost između pozicije praćenog objekta i stare pozicije kamere, a interpolant je d , što je u slučaju ove simulacije proizvod faktora glatкости pokreta kamere i vremena za koje se ažurira fizika i frame-rate u Unity pogonu.

3.1. Resonance Audio Renderer

S obzirom da kamera prati karakter u simulaciji, uz kratko kašnjenje zbog razmaka između kamere i karaktera, korisnik će da čuje sve zvukove koje proizvode audio-izvori kraj kojih karakter prolazi. Ukoliko korisnik naredi karakteru da prođe pored potoka, glasnije će čuti zvuk vode, i to jače na ono uho prema kome je potok u simulaciji bliži. Ukoliko korisnik okrene glavu, i zvuk će se pojačati na suprotnom uhu.

Prema Serafinovoj definiciji [1], ovo je prva komponenta zvučnog pejzaža - okolinski (ambijentalni) zvuk. U Unity pogonu, ovakav uvjerljiv prostorni zvuk omogućen je pomoću Resonance Audio SDK paketa. Resonance Audio komponente poboljšavaju karakteristike Unity ugrađenih audio komponenti. Nakon što se ovaj renderer skine, potrebno je u Edit->Project Settings->Audio meniju postaviti Spatializer Plugin i Ambisonic Decoder Plugin opcije na Resonance Audio.



Slika 1. Dodavanje Resonance Audio Renderera u grupu audio miksera

Dodavanjem Resonance Audio Renderera u grupu audio miksera u Unity-ju se uvodi Ambisonic Audio u opšti audio kanal tako da se može čuti i mogu se primjenjivati efekti na njega.

Resonance audio poboljšava Unity AudioListener (komponenta koja “sluša”) funkcije uvođenjem dodatnih opcionalnih parametara, kao što su šabloni usmjerenja zvuka (koji su istraženi detaljnije u poglavlju 3), globalne maske pojačanja i maske za zaklanjanje izvora. Takođe uključuje Ambisonic rekorder zvučnog polja koji može da ubaci prostorne audio izvore u scenu (u Ambisonic zvučno polje).

Komponenta Resonance Audio Soundfield, predstavlja potpun 360° prostorni zvuk tako što kodira zvučne talase na virtualnu sferu oko slušaoca. Resonance Audio Room, simulira efekte sobe (eho) za određen prostor tako što koristi Transform svojstva (veličina, pozicija i stepen rotacije) objekta u igri i primjenjuje efekte sobe shodno tome (kada je objekat u odgovarajućem dijelu sobe).

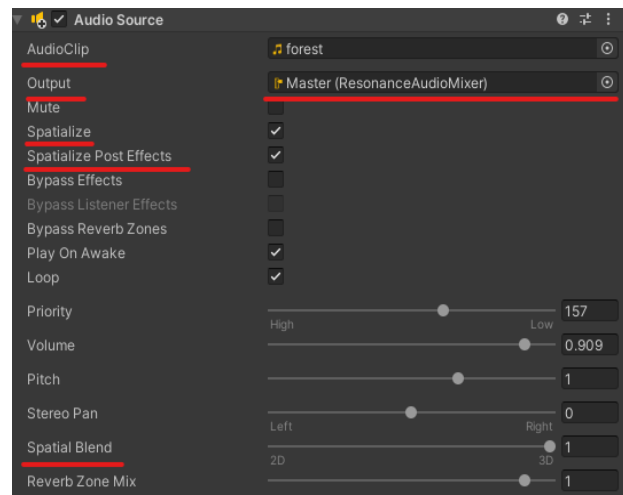
Da bi se omogućio potpun efekat prostornog zvuka, treba da se doda Resonance Audio Listener komponenta u scenu -uobičajeno se stavlja na komponentu glavne kamere.

Dodaje se i AudioSource komponenta na objekat iz simulacije koji je namijenjen da bude izvor zvuka. Za razliku od Ambisonic zvuka, koji se namješta preko audio isječka, prostorni zvuk odabire se na samom izvoru zvuka. Kako bi se dozvolilo Resonance Audio rendereru da načini ovaj objekat izvorom prostornog zvuka, AudioSource svojstva objekta treba da se podese; treba da se odabere audio isječak koji će izvor reprodukovati (u ovom slučaju zvuk potoka, ptica itd.).

Takođe, potrebno je namjestiti izlaz audio mikser grupe na glavnu traku Resonance Audio Mixer-a. Treba da se omogućiti Spatialize i Spatialize Post Effects opcija da bi se primijenili prostorni efekti na izvor zvuka, kao i njihova naknadna dejstva.

Da se kontroliše željeni smijer zvukova u audio Ambisonic datoteci, može se jednostavno rotirati objekat izvora zvuka dok se zvukovi ne poravnaju tamo gdje je željeno (npr. u ambijentu ove simulacije, zvukove potoka koji su zapljuskuju obalu treba uskladiti sa smjerom struje potoka u prizoru).

Većina ostalih postavki na izvoru zvuka, poput glasnoće i visine tona, nastaviće raditi kao i obično.

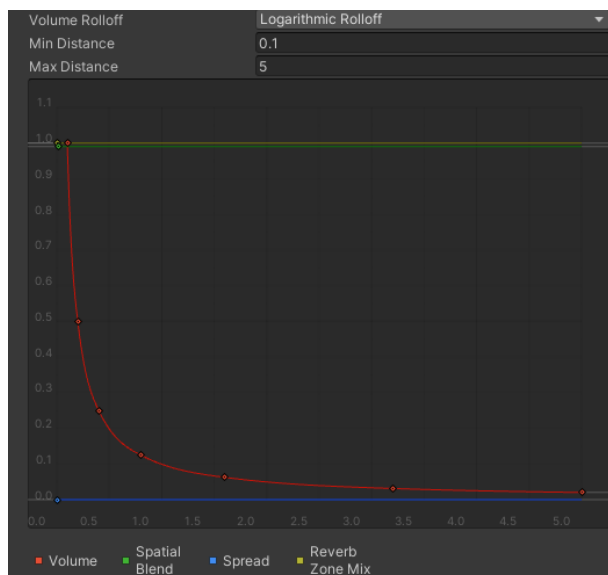


Slika 2. Podešavanje izvora zvuka kako bi se omogućio 3D prostorni zvuk

U pitanju je 3D simulacija, pa će se zvuk pojavljivati iz sva tri smjera objekta audio izvora. To znači da će zvuk zvučati kao da dolazi sa svoje 3D lokacije, ali neće biti tiši dok se od njega udaljava korisnik, baš kao i sa Ambisonic audio isječkom. Razlog tome je način na koji Resonance Audio izračunava prostorni zvuk na razmjerljiv način. Radi tako što kombinuje više prostornih audio izvora u jednu Ambisonic audio sliku koja omogućava da Resonance Audio podržava veliki broj prostornih audio izvora, bez značajnih troškova obrade. Da bi se omogućili 3D efekti – mijenjanje jačine zvuka u odnosu na poziciju u sceni, podešava se Spatial Blend (mapiranje kanala) svojstvo na 3D. Ovo svojstvo je moguće kontinualno podešavati između 2D i 3D krajnosti. U standardnoj Unity AudioSource komponenti, moguće je upravljati funkcijom koja određuje jačinu zvuka u zavisnosti od pozicije slušaoca na izvor. Na sljedećoj slici prikazana je logaritamska kriva koja određuje ovo svojstvo na jednom od izvora simulacije:

Na slici 3 se pored jačine zvuka vide i ostala svojstva izvora -Spatial Blend mapiranje kanala je uvijek postavljeno na 3D zvuk (vrijednost 1). Spread svojstvo je 0 i označava ugao (broj stepeni) širenja zvuka. 3D zvuk je u tom slučaju jako prilijepljen za smijer porijekla. U ovoj simulaciji, ukoliko kamera prolazi kraj drveća direktno sa lijeve strane, zvuk ptica će da bude pušten samo na lijevu slušalicu. Reverb Zone Mix [3] svojstvo radi sa audio isječkom tako što primijeni distorziju na njega u zavisnosti gdje je slušalac lociran u odnosu na podešenu zonu – npr ukoliko karakter ulazi u pećinu postepeno treba da se mijenja ambijentalni efekat.

Kroz komponentu Resonance Audio izvora moguće je fino podešavanje ovih svojstava; npr. smijer za slušaoca ili za izvor se podešava kroz dvije opcije – Alpha i Sharpness. Ovo je šablon tj. oblik koji je povezan sa zaklanjanjem objekta i predstavlja način na koji će zvuk proizilaziti iz izvora u različitim pravcima. Zvuk izvora će se čuti drugačije u zavisnosti od ovog šablona i lokacije kamere u odnosu na izvor.



Slika 3. Logaritamska funkcija odnosa jačine zvuka i rastojanja između slušaoca i izvora

Min i Max Distance parametri određuju minimalnu i maksimalnu udaljenost slušaoca od izvora gdje zvuk počinje odnosno prestaje da se čuje. Najuvjerljiviji zvuk u simulaciji dobijen je kombinacijom parametara kao na slici 3, sa logaritamskim opadanjem.

4. ZAKLJUČAK

Virtualna realnost zahtijeva novi način razmišljanja o obradi zvuka. U dugogodišnjoj istoriji igara i videa na ravnim ekranima, standard realizma za reprodukciju zvuka ostao je relativno nizak, posebno u poređenju sa savremenim napretkom grafike i kinematografskog prikazivanja videa. Iako je sluh i trodimenzionalno čulo, zvuk za igre i kino/video obično je smanjio upotrebu 3D i drugih naprednih tehnologija za reprodukciju zvuka, iz jednostavnog razloga što su sva grafika i video pred korisnikom.

Sa izuzetkom nekih FPS igara u kojima 3D zvuk pruža taktičku pomoć koju igrači nauče koristiti u svoju korist, previše realizma u zvuku za igru ili film s ravnim ekranom ponekad može odvratiti pažnju, pogotovo ako nije u skladu sa vizualnim doživljajem. U svijetu VR-a, prostorni zvuk odličan je alat koji može pomoći pri lociranju i navigaciji za objekte u VR okruženjima.

Rezultati mnogih istraživanja sugerišu da prostorni zvuk može povećati percepciju korisnika u tri dimenzije dok traži skrivene objekte, stoga se može zaključiti da je 3D zvuk vrijedan faktor za obavljanje zadatka za većinu korisnika u VR-u.

Iako je nekad ovo predstavljalo programsku prepreku pri pravljenju simulacija, danas postoje mnoge biblioteke sa priključcima za pogone igara koji apstrahuju i pojednostavljaju čitav proces uvođenja 3D prostornog zvuka, i jedan od načina za postizanje ovog procesa je objašnjen u ovom radu. Osim pomoći korisniku pri orijentaciji, dobar način reprodukcije 3D zvuka u VR-u je neizostavan za poboljšanje osjećaja prisutnosti.

5. LITERATURA

- [1] Serafin, S., i Serafin, G. (2004). "Sound design to enhance presence in photorealistic virtual reality," ICAD 04-Tenth Meeting of the International Conference on Auditory Display (Sydney, NSW).
- [2] Anton Venema, on November 14, 2018, <https://www.frozenmountain.com/developers/blog/archive/spatial-audio-in-vr-why-and-how> (pristupljeno u oktobru, 2021.)
- [3] <https://docs.unity3d.com/Manual/class-AudioSource.html> (pristupljeno u oktobru, 2021.)

Kratka biografija:



Jelena Garić rođena je u Brčkom, BiH, 1997. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Reprodukcijska zvuka u 3D virtualnoj sceni odbranila je 2021.god.

kontakt: nephalem46@gmail.com

RAZVOJ APLIKACIJE ZA UPRAVLJANJE ELEKTRIČNOM ENERGIJOM U OKVIRU CLOUD OKRUŽENJA**DEVELOPMENT OF AN APPLICATION FOR ELECTRICITY MANAGEMENT WITHIN THE CLOUD ENVIRONMENT**

Snežana Dupljanin, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Aplikacija EMS predstavlja sistem upravljanja energijom, koja vrši monitoring potrošnje električne energije u realnom vremenu i teži da zadovolji jednakost između potrošene i proizvedene električne energije, koristeći se optimizacionim genetskim algoritmom. Sistem je implementiran u Cloud okruženju korišćenjem Azure Service Fabric platforme. U radu su izmjerena vremena izvršavanja genetskog algoritma za različit broj klijenata kako za mikroservisnu arhitekturu u Cloud okruženju, tako i za monolitnu aplikaciju.

Ključne reči: Azure, Cloud, CE, stateful i stateless mikroservisi, Azure Storage.

Abstract – EMS (Energy Management System) monitors electricity consumption in real time and strives to satisfy the equality between consumed and produced electricity, using an optimization genetic algorithm. The system environment is Cloud using the Azure Service Fabric platform. The paper measures the execution times of the genetic algorithm for a different numbers of clients, for the architecture of microservices in the Cloud environment and the monolithic application.

Keywords: Azure, Cloud, CE, stateful and stateless microservices, Azure Storage.

1. UVOD**1.1 Azure Service Fabric**

Azure Service Fabric [1] je platforma kao usluga (PaaS), tj. distribuirana servisna platforma open-source koja olakšava pakovanje, primjenu i upravljanje skalabilnim i pouzdanim mikroservisima i kontejnerima. Ona pruža mogućnost upravljanja životnim ciklusom aplikacija zasnovanih na mikroservisima.

Mikroservisi se hostuju unutar kontejnera koji su raspoređeni u klasteru Service Fabric-e. Uvođenjem mikroservisa moguće je skalirati svaku komponentu aplikacije zasebno, što pojednostavljuje proces uvođenja promjena, za razliku od tradicionalne monolitne arhitekture. Azure Service Fabric podržava stateful i stateless mikroservise. Mikroservisi komuniciraju međusobno uz pomoć API-ja.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Srđan Vukmirović, vanr. prof.

1.2 Calculation Engine

Calculation Engine je u okviru monolitičke arhitekture predstavljao jedan kompleksan servis koji je obavljao više funkcionalnosti, pa ga je zbog toga bilo neophodno podijeliti na više mikroservisa. Ta podjela usluga je raspoređena na četiri mikroservisa, a to su: 1. Servis koji se bavi proračunom vezanim za CO2 (emisija/redukcija). 2. Servis koji se bavi proračunom profita i troška. 3. Servis za komunikaciju sa bazom. 4. Servis u kom se primjenjuje genetski algoritam.

Prva tri servisa su stateless jer je njihov zadatak da manipulišu podacima koji se skladište u bazi ili da vrše proračune i dobijene rezultate prosleđuju dalje. Četvrti servis je servis koji izvršava algoritam i on je stateful jer je neophodno da sačuva najbolju jedinku iz prethodnog računanja koja predstavlja prvu jedinku u prvoj populaciji novog računanja.

1.3 Azure Storage

Azure Storage [2] je platforma za skladištenje podataka na cloud platformi. Svi podaci su šifrovani od strane Azure Storage-a koji pruža kontrolu pristupa. Dizajniran je tako da bude skalabilan i zadovolji potrebe za skladištenjem podataka i performansama današnjih aplikacija. On takođe vodi računa o održavanju hardvera, update-a i kritičnih problema. Azure storage platforma nudi više različitih data servisa, od kojih su Azure Tables i Azure Blobs korišćeni u projektu.

2. OPIS KORIŠĆENIH TEHNOLOGIJA I ALATA

Azure Service Fabric je platforma distribuiranih sistema koja olakšava implementaciju i upravljanje skalabilnim i pouzdanim mikroservisima.

Service Fabric Explorer je open-source alat koji omogućava pregled i upravljanje Azure Service Fabric klasterima.

Microsoft Visual Studio je integrisano razvojno okruženje koje omogućava razvoj aplikacija.

Microsoft Azure Storage Explorer je samostalna aplikacija koja olakšava rad sa Azure Storage podacima.

C# je moderan objektno orijentisan programski jezik višeg nivoa, koji je razvila kompanija Microsoft.

.NET Framework je platforma za razvoj softvera koju je Microsoft razvio za izradu i pokretanje aplikacija.

WPF (Windows Presentation Foundation) je tehnologija za pravljenje klijentskih aplikacija na Windows platformi, koja nudi razne opcije za podešavanje izgleda aplikacije.

3. OPIS PROBLEMA

Zadatak ove aplikacije jeste da vrši monitoring potrošnje električne energije u realnom vremenu i teži da zadovolji jednakost između potrošene i proizvedene električne energije, koristeći se optimizacionim gentskim algoritmom. Ovim algoritmom se teži ostvariti minimum potrošnje goriva u konvencionalnim generatorima, tj. povećati profit uzimajući u obzir cijenu goriva kao i proizvedenu električnu energiju iz generatora, koji koriste obnovljive izvore. Pored optimizacije vrše se proračuni za redukciju i emisiju CO₂ izražen u tonama, kao i profit i trošak izražen u dolarima. Mikroservisna arhitektura je sačinjena od skupa nezavisnih servisa od kojih svaki izvršava zasebno svoj zadatak. Glavni cilj jeste da se podijeli aplikacija na što više manjih nezavisnih mikroservisa i da obezbijedi njihovu međusobnu komunikaciju.

4. TEORIJSKE OSNOVE

4.1 Mikroservisna arhitektura

Monolitske aplikacije su dobro rješenje samo za manje aplikacije koje nisu distribuiranje i od kojih se ne zahtjevaju visoke performanse opsluživanja klijenata, ali sa stanovišta održavanja aplikacije i distribuiranosti, kod ovih aplikacija će se uvijek javiti problem kako sistem održati stabilnim.

Mikroservisna arhitektura sa druge strane omogućava tu pouzdanost, skalabilnost i proširivost na brži i jednostavniji način. Čitava poenta je razbijanje aplikacije na mikroservise kao zasebne cjeline u sistemu koje odraduju neki svoj specifičan dio posla. Mikroservisna arhitektura sve više dobija na popularnosti, sa sve više kompanija koje dijele pozitivna iskustva u vezi sa njenom primjenom. Servisi unutar aplikacije su izdjeljeni na stateless i stateful.

4.1.1 Stateless mikroservisi

Stateless mikroservisi [4] obezbjeđuju zahtjeve samo na osnovu informacija prenijetih uz svaki zahtjev i ne oslanjaju se na informacije iz ranijih zahjeva. To znači da servisi ne moraju da drže informacije o stanju između zatjeva. Stateless servisi ne čuvaju lokalno promjenjiva stanja. Uglavnom se potrebni podaci za rad čuvaju eksterno u bazi podataka.

Instance stateless servisa se raspoređuju po čvorovima i u slučaju otkaza bilo koje instance ili čvora na kom se stanica nalazi, se vrši ponovno startovanje na ispravnom čvoru. Stateless servisi su jednostavniji od stateful servisa, samim tim što ne moraju da paze na kopiranje podataka prilikom spuštanja servisa. Samim tim što ne čuvaju podatke, servisi ovog tipa komuniciraju sa stateful servisima radi dobavljanja podataka. Na podacima koje su primili, obave određene proračune i vrte ih servisima koji ih mogu čuvati ili ih prosto proslijede da bi se prikazali korisniku.

4.1.2 Stateful mikroservisi

Stateful servisi [4] održavaju promjenljivo stanje. Programski kod se izvršava samo na jednom čvoru koji se naziva primarni, dok su ostali čvorovi određenog servisa pasivni. U toku rada, primarni čvor konstantno replicira podatke na sekundarne čvorove i na taj način održava konzistentno stanje sistema. Odnosno, u slučaju otkaza

primarnog čvora, jedan od sekundarnih preuzima posao i nastavlja njegovo izvršavanje. Takođe, vrši se oporavak čvora koji je otkazao, pri čemu se on formira kao sekundarni čvor.

5. IMPLEMENTACIJA RJEŠENJA

5.1 Arhitektura rješenja

Tokom izrade projekta korišćeno je razvojno okruženje (Local cluster) koje je isto kao produkciono okruženje, pa tako implementirane aplikacije na lokalnom klasteru mogu lako biti primjenjene na nekom drugom okruženju. Aplikacija u Cloud okruženju je na local host-u i sastoji se od jedne Service Fabric-e, koja ima 14 mikroservisa. Slika 1.

Calculation Engine Microservice je servis zadužen za izvršavanje genetskog algoritma i on je stateful jer je neophodno da sačuva najbolju jedinku iz prethodnog računanja koja predstavlja prvu jedinku u prvoj populaciji novog računanja.

Emission and Reduction CO₂ Microservice je zadužen za računanje emisije CO₂ pri čemu se ukupna snaga dobijena iz generatora pomnoži sa emisijom faktorom koji je različit za svako gorivo. Redukcija CO₂ se računa na način da se pretpostavi koliko bi bilo emitovano CO₂ da se snaga dobijena iz generatora na obnovljivi izvor energije, proizvodila iz konvencionanih generatora i da se pri tome uzima najveći emisijni faktor.

Cost and Profit CE Microservice je zadužen za računanje profita tako što snagu iz generatora na obnovljive izvore energije pomnožimo sa najjeftinijim gorivom. Cost se računa tako što se uzme suma svih snaga generatora pomnožena sa cijenom za to gorivo koje generator troši.

Repository CE Microservice je servis zadužen za komunikaciju sa bazom, tj. za čuvanje svih izračunatih vrijednosti na CE to su: profit, cost, emisija i redukcija. Izračunati podaci se čuvaju u Azure Table Storage-u.

Calculation Engine PubSub Microservice Komunikacija između Calculation Engine-a i UI-a je ostvarena preko Publish – Subscribe je paterna. Nakon što genetski algoritam završi optimizaciju, trenutna proizvodnja generatora, profit i trošak izražen u dolarima, emisija i redukcija CO₂ se prosleđuju na UI. U ovom slučaju CE predstavlja Publisher-a, dok UI predstavlja Subscriber-a.

SCADA Collecting Microservice je zadužen za prikupljanje podataka iz polja. Simulator korišćen za simuliranje podataka je EasyModbus, koji koristi Modbus protokol za komunikaciju. Scada uspostavlja konekciju sa simulatorom koristeći ModbusClient klasu. Nakon uspostavljene komunikacije, počinje prikupljanje podataka sa simulatora koje se vrši na svake 4 sekunde.

SCADA Processing Microservice Nakon što se podaci pročitaju sa simulatora, oni se šalju na SCADA Processing mikroservis. On je zadužen za procesiranje podataka, tj. vrši se konverzija promjenljivih iz sirovog formata u format inženjerskih jedinica (EGU values).

SCADA Commanding Microservice ima zadatak da prima komande i omogući njihovo dalje prosleđivanje u polje. Komandovanje digitalnim i analognim vrijednostima moguće je manuelno ili automatski.

Putem grafičkog korisničkog interfejsa, korisnik može da uključi/isključi generator i da zadaje vrijednosti generatora. Automatsko komandovanje je od strane genetskog algoritma koji zadaje vrijednosti u zavisnosti od potreba sistema.

Alarms Events Microservice Procesirane vrijednosti sa SCADA Processing mikroservisa se šalju na Alarms Event mikroservis kako bi se provjerilo da li su vrijednosti u opsegu alarmnih vrijednosti. Ukoliko se desi alarm na nekoj od tačaka, ovaj mikroservis upisuje podatke u bazu podataka u Azure Table Storage.

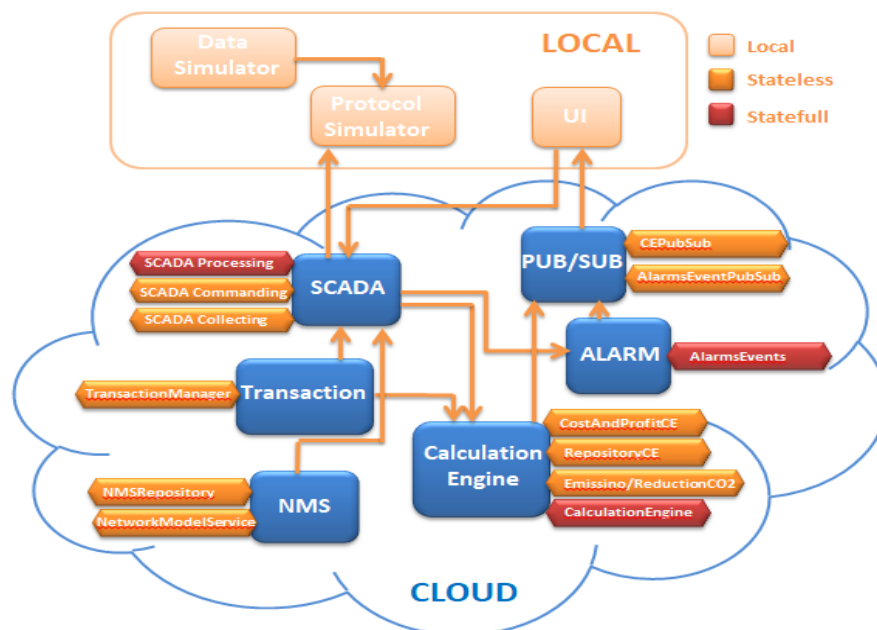
Alarms Events PubSub Microservice Publish – Subscribe patern je iskorišćen i u komunikaciji Alarm Service-a sa UI-om. Uz pomoć ovog mikroservisa UI

dobija sva alarmna stanja analognih i diskretnih signala, kako bi se omogućio korisniku uvid u iste.

Transaction Manager Microservice je servis koji kontroliše koordinaciju transakcije (promjena NMS modela) i odgovoran je za stvaranje jednog ili više transakcionih objekata i njihovo prosleđivanje na servise kako bi se očuvala konzistentnost modela.

NMS Repository Microservice je servis koji je zadužen za komunikaciju sa bazom, tj. služi za čitanje i upisivanje Delte u Azure Blob Storage.

Network Model Microservice je servis koji je isključivo zadužen za rad sa NMS modelom, vrši učitavanje i konverziju modela, kao i prosleđivanje konvertovanog modela ostalim servisima.



Slika 1. Arhitektura rješenja

5.2 Service Fabric Explorer

Service Fabric Explorer (SFX) je open-source alat koji omogućava pregled, razvoj i upravljanje Azure Service Fabric klasterima.

Pokretanje SFX u bilo kom web pretraživaču ostvaruje se putem linka <https://localhost:19080/Explorer>. ServiceFabric ima dva mode-a simulaciju sa jednim ili pet čvorova. Ukoliko se koristi mode sa pet čvorova, moguće je deaktivirati pojedine čvorove. Omogućeno je ponovno pokretanje čvorova ili njihovo deaktiviranje pa se na taj način može lako testirati failover. Za testiranje većeg broja mikroservisa, simulator troši dosta procesorske moći i radne memorije. Praćenje stanja servisa i upravljanje njima vrši se kroz SFX, a u njemu je omogućen i prikaz svih aplikacija i node-ova, kao i pranje zdravlja istih.

5.3 Testiranje

Aplikacija je testirana za brzinu izvršavanja genetskog algoritma prije i poslije njene implementacije u Cloud okruženju. Na samu brzinu izvršavanja algoritma utiču vrijednosti svih ulaznih parametara, a to su: broj iteracija, broj populacija, elitism i mutation rate. Default-ne

vrijednosti su ostavljene prilikom testiranja: broj iteracija= 200, broj populacija = 100, elitism = 5, mutation rate = 1. Sve ove vrijednosti je moguće komandovati i preko UI.

Opšti koraci genetskog algoritma su prikazani na slici 2. Teorijski postoje različiti uslovi zaustavljanja genetskog algoritma, u ovom projektu taj uslov je dok se ne postigne određen broj iteracija tj. 200.

```
Genetski algoritam{
    generisanje pocetne populacije P;
    za svako rjesenje iz P izracunati osobine;

    ponavljaj(uslov)
    {
        Odabir roditelja iz P;
        dijete = ukrstanjeGena(roditelj1,roditelj2);
        mutacija(dijete);
        izracunavanje osobine djeteta;
    }
    Ispisi rjesenje;
}
```

Slika 1. Genetski algoritam

Tabela 1. Rezultati testiranja

Br. klijenata	Monolitna aplikacija [s]	Aplikacija u <i>Cloud</i> okruženju [s]
10	1	1
50	2	1
100	3	2
300	9	6
600	19	12
1000	33	21

U Tabeli 1 prikazana su vremena izvršavanja akcije. Prva kolona predstavlja broj klijenata koji zahtijevaju postizanje elergetskog balansa između proizvedene i potrošene energije, koji se dobija putem genetskog algoritma. U drugoj koloni se nalaze vremena izvršavanja akcije za monolitnu aplikaciju, a u trećoj koloni vremena vezana za aplikaciju implementiranu u Cloud okruženju.

Vrijednosti u tabeli predstavljene su u sekundama i zao-kružene su na cjelobrojne vrijednosti. Iz tabele se jasno može zaključiti da su vremena izvršavanja genetskog algoritma za mali broj korisnika približna, dok se sa porastom broja klijenata vrijednosti međusobno razlikuju. Sa povećanjem broja korisnika, monolitnoj aplikaciji treba više vremena za izvršenje u odnosu na aplikaciju implementiranu u Cloud okruženju i na osnovu toga se može zaključiti da je rad monolitne aplikacije sporiji u odnosu na aplikaciju koja je implementirana u Cloud okruženju.

6. ZAKLJUČAK

Zbog pojave nekih vrsta problema poput: opadanja performansi, stabilnosti sistema i sporog razvojnog ciklusa, mikroservisa arhitektura je bolji odabir u odnosu na tradicionalnu monolitnu arhitekturu. U mikroservisnoj arhitekturi promjene na jednom dijelu sistema ne utiču na neki drugi dio sistema iako nisu logički povezani.

Veliki problem nastaje kada uvođenje bilo kakave promjene u aplikaciju počne da oduzima previše vremena.

Mikroservisna arhitektura omogućava pouzdanost, skalabilnost i proširivost na brži i jednostavniji način razbijanjem aplikacije na mikroservise kao zasebne cjeline.

7. LITERATURA

- [1] <https://www.c-sharpcorner.com/article/understand-azure-service-fabric-in-three-minutes/>
- [2] <https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction>
- [3] <https://docs.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices>
- [4] <https://www.xenonstack.com/insights/stateful-and-stateless-applications>
- [5] <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-visualizing-your-cluster>

Kratka biografija:



Snežana Dupljanin rođena je 27.02.1997. godine u Sokocu. Diplomirala je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Primenjeno softversko inženjerstvo 2019. godine.

INDUSTRIJSKI KOMUNIKACIONI PROTOKOLI U TRANSFORMATORSKIM STANICAMA SREDNJEG I VISOKOG NAPONA**INDUSTRIAL COMMUNICATION PROTOCOLS IN MEDIUM AND HIGH VOLTAGE SUBSTATIONS**Boško Jovanović, Boris Dumnić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U radu je navedena i opisana oprema koja se nalazi u transformatorskoj stanici, sa akcentom na zaštitne releje i SCADA sistem. Opisan je princip komunikacije između uređa na samoj stanici i prema SCADA sistemu. Komunikacija je zasnovana na IEC 61850 protokolu. Opisani su MMS, GOOSE i Sampled Values principi komunikacije. Praktičnim radom obrađena je zaštita od otkaza prekidača i predstavljen GOOSE princip komunikacije između tri releja.

Ključne reči: Relejna zaštita, Mikroprocesorski relej, SCADA, Komunikacija, Transformatorska stanica

Abstract – This paper discusses the equipment located in the substation, with emphasis on protection relays and SCADA system. Communication principle between relays on substation level and communication between relays and SCADA system were described. Communication is based on the IEC 61850 protocol. MMS, GOOSE and Sampled Values principle of communication were described. By practical work, the circuit breaker failure protection was adjusted and GOOSE principle of communication was tested.

Keywords: Relay protection, Microprocessor relay, SCADA, Communication, Substation

1. UVOD

Elektroenergetika predstavlja osnovu privrednog i društvenog razvoja na globalnom nivou. Razvoj elektroenergetike praćen je intenzivnim razvojem elektroenergetskog sistema. Elektroenergetski sistem sastoji se u suštini od četiri osnovna podsistema:

- Podsystem za proizvodnju električne energije;
- Podsystem za prenos električne energije;
- Podsystem za distribuciju električne energije i
- Podsystem za potrošnju električne energije.

Najvažniji elementi elektroenergetskog sistema su generatori, transformatori i vodovi. S obzirom da se u elektroenergetskom sistemu najčešće radi o velikim snagama i naponima potrebno je čvorne tačke sistema pažljivo izvesti. Zbog toga se čvorne tačke izvode kao razvodna postrojenja.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Boris Dumnić, van. prof.

Ova postrojenja mogu biti sa ili bez transformacije napona. S obzirom na namenu za koju se koriste, razvodna postrojenja mogu se podeliti na razdelne i transformatorske stanice. Razdelne stanice su postrojenja u kojima se stiču vodovi istog naponskog nivoa i za cilj imaju raspodelu električne energije na priključne vodove. U transformatorskim stanicama, pored raspodele energije na priključne vodove, vrši se i transformacija električne energije sa jednog naponskog nivoa na drugi. Transformatorska stanica je razvodno postrojenje sa jednim ili više energetske transformatora. Transformatorske stanice po svojoj konstrukciji su veoma kompleksni objekti koji moraju da ispune brojne složene i važne zadatke [1]. Osnovni elementi transformatorskih stanica su: sabirnice, izolatori, rastavljači, prekidači, osigurači, energetski transformatori, merni transformatori, energetski kablovi, prigušnice, odvodnici prenapona i zaštitni uređaji i dr.

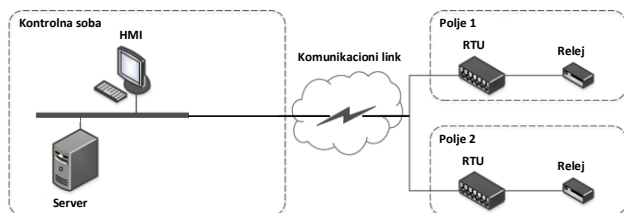
2. OPIS POSTROJENJA

Pri projektovanju transformatorskih stanica mora se voditi računa o izboru primarne opreme i opreme za zaštitu, jer su transformatorske stanice jedan od najskupljih elemenata u čitavom elektroenergetskom sistemu. Oprema u transformatorskoj stanici izložena je raznim naprezanjima kao što su: električna, termička, mehanička, ambijentalni uslovi i slično. Takođe, bitni faktori u radu stanice su i starost opreme, vreme korišćenja i način skladištenja. U radu su opisani svi glavni elementi transformatorske stanice.

Najvažniji zaštitni elementi u transformatorskim stanicama su zaštitni mikroprocesorski releji. Njihova uloga je da brzim reagovanjem isključe elemente sa mreže kada se pojavi struja kvara ili struja preopterećenja. Na taj način se oprema dovodi u beznaponsko stanje i izbegavaju se havarijske posledice do kojih bi došlo da su elementi ostali priključeni na mrežu pri pojavi struje kvara ili preopterećenja. Svaki relej štiti određeni element ili grupu elemenata, tako da na nivou transformatorske stanice postoji veliki broj releja. Zbog logike i selektivnosti zaštite releji moraju međusobno komunicirati. Takođe, releji komuniciraju i sa SCADA sistemom.

Arhitektura mreže između releja i SCADA sistema može biti realizovana na više različitih načina. Arhitektura mreže koja se danas najviše koristi jeste da je određena grupa releja preko ethernet sviča povezana na RTU. Između ostalog, uloga RTU-a u sistemu je da jedan komunikacioni protokol pretvori u drugi. Komunikacija između releja i RTU-a realizovana je preko IEC 61850 protokola, a dalje od RTU-a do SCADA-e preko IEC

60870-5-104 protokola. Arhitektura mreže sa RTU-om prikazana je na slici 1.



Slika 1. Arhitektura mreže od releja do SCADA sistema

3. MIKROPROCESORSKI RELEJI I SCADA SISTEM

3.1. Mikroprocesorski relej kao zaštitni element

U prošlosti su bile korišćene razne tehnologije za detekciju kvara u sistemu i zaštitu elemenata. Pre pojave mikroprocesorskih relaja, kao zaštita sistema koristili su se elektromehanički releji. Oni su radili na principu elektromagnetne indukcije. Sa razvojem tehnologije dolazi do razvoja i zaštitnih uređaja u transformatorskim stanicama. Prvi mikroprocesorski releji pojavili su se 1985. godine. Masovnja upotreba počela je 1990. godine. Danas se zaštita transformatorskih stanica ne može zamisliti bez mikroprocesorskih relaja. Postali su nezamenljivi elementi svake stanice zbog svoje pouzdanosti, preciznosti i brzine delovanja. Još jedna velika prednost u odnosu na prethodne izvedbe relaja jeste programabilnost. Time je omogućena upotreba relaja u mnogo složenijim procesima nego ranije. U ovom radu obrađivaće se samo mikroprocesorski releji.

Mikroprocesorski relej je zaštitni uređaj koji koristi mikroprocesor za analizu prikupljenih analognih i digitalnih signala. Oni imaju ugrađen konvertor za konverziju analognih signala, kao što su napon i struja, u digitalne i koriste mikroprocesor za određivanje logike i izdavanje komandi. Princip rada se kreće od prostih i jednostavnih do veoma složenih i zahtevnih operacija. Ovi releji imaju napredne programabilne funkcije koje omogućavaju veliku fleksibilnost u praćenu sistema i izboru karakteristike krive za proradu određene zaštite. Mikroprocesorski relej istovremeno upravlja sa nekoliko zaštitnih funkcija, obavlja komunikaciju sa ostalim uređajima, vrši nadgledanje i snimanje promena u sistemu. Upotrebom mikroprocesorskih relaja smanjuje se potreba za uređajima za lokaciju greške, snimanje događaja, kontrolu, nadzor i merenje, jer je sada sve to objedinjeno u jedan uređaj.

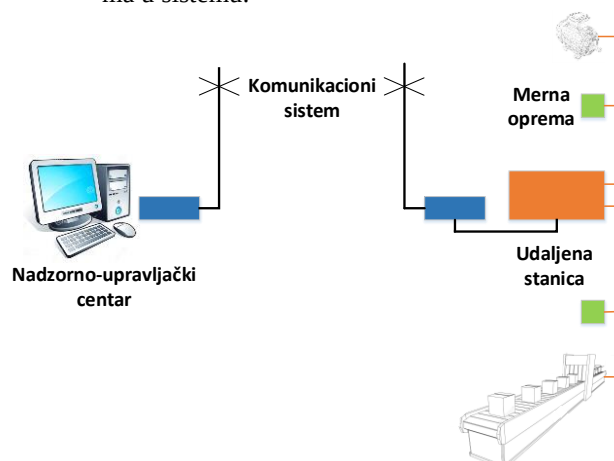
3.2. SCADA sistem

U današnjem savremenom društvu postoji veliki broj procesa za koje je potreban nadzor i upravljanje. Ti procesi su opremljeni uređajima za merenje, senzorima, raznim detektorima i drugom opremom neophodnom za nadzor i upravljanje sistemom. Kada je proces opremljen ovim uređajima on može biti kontrolisan određenim sistemom. Danas, najzastupljeniji sistem za tu namenu jeste SCADA sistem. Sa SCADA sistemom vrši se nadzor procesa i daljinsko upravljanje. Svi elementi procesa, nad kojima se vrši nadzor i upravljanje sa SCADA sistemom, su grafički prikazani na ekranima u kontrolnoj sobi. SCADA sistem ima veoma široku primenu u raznim sistemima i procesima. Najčešće se primenjuje u sistemima kao što su proizvodnja i distribucija električne i

toplote energije, petrohemijska industrija, građevinska industrija, prehrambena industrija, industrija plastičnih masa i slično.

Glavni ciljevi SCADA sistema su:

- Vršanje nadzora sistema;
- Upravljanje sistemom;
- Prikupljanje i skladištenje pristiglih podataka iz sistema;
- Smanjenje angažovanja radne snage primenom automatizacije i
- Izdavanje upozorenja o detektovanim problemima u sistemu.



Slika 2. Primer SCADA sistema

Osnovni elementi SCADA sistema su:

- Nadzorno-upravljački centar;
- Komunikacioni sistem;
- Udaljena stanica i
- Merna oprema.

4. OPIS KOMUNIKACIONIH PROTOKOLA

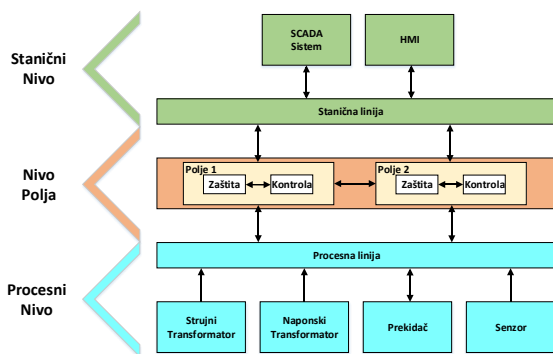
IEC 61850 je međunarodni standard primenljiv za komunikaciju u transformatorskim stanicama. Pre objavljivanja standarda IEC 61850 različiti proizvođači opreme koristili su različite protokole za komunikaciju. Tako je u transformatorskim stanicama u kojima se koristila oprema različitih proizvođača bilo neophodno informacije pretvoriti iz jednog protokola u drugi. Pretvaranje jednog protokola u drugi unosilo je određeno kašnjenje u sam proces. Sa porastom količine informacija to vreme je postalo sve duže i to je postao potencijalni problem. Cilj IEC 61850 standarda jeste da reši taj problem jednim standardizovanim protokolom za transformatorske stanice. Time je omogućena velika interoperabilnost između uređaja različitih proizvođača i objedinjeni različiti nivoi u okviru postrojenja [2].

Struktura IEC 61850 protokola je hijerarhijska i definisana je na tri tipična nivoa:

- Stanični nivo (računar, operatorske stanice...);
- Nivo polja (kontrolni, nadzorni i zaštitni uređaji) i
- Procesni nivo (merna i rasklopna oprema) [3].

Standard IEC 61850 definiše komunikaciju koristeći OSI model. OSI model je internacionalni standardizovani model koji opisuje interakciju uređaja, programa, softvera i protokola tokom komunikacije. Takođe, daje spisak

funkcija, servisa i protokola koji funkcionišu na svakom nivou. Za razliku od nekih prethodnih komunikacionih standarda čija se semantika procesnih podataka temelji na signalima (engl. *Signal-oriented*), IEC 61850 uvodi načelo opisivanja energetskeg uređaja na principu objektno orijentisanog modelovanja [4].



Slika 3. Arhitektura IEC 61850 protokola na nivou transformatorske stanice

Tri najistaknutija protokola su:

- Client/Server (MMS);
- GOOSE i
- Sampled Values.

Princip klijent-server omogućava razmenu informacija koje nisu vremenski kritične između klijenta i servera. Primjenjuje se u aplikacijama za automatizaciju postrojenja kao što su kontrola i nadzor opreme u transformatorskoj stanici, sinhronizacija vremena, prenos komandi i dobijanje izveštaja iz sistema [5]. Klijent – server komunikacija je vertikalna komunikacija a primer je protokol proizvođačkih specifikacionih poruka (engl. *Manufacturing Message Specification - MMS*).

Za prenos informacija koje su vremenski kritične koristi se izdavač – pretplatnik princip. Prenos informacija je u okviru milisekundi. Dve metode koje se koriste kod principa izdavač – pretplatnik su opšti objektno orijentisani događaji trafostanice (engl. *Generic Object Oriented Substation Event - GOOSE*) i prenos uzorkovanih vrednosti (engl. *Sampled Values - SV*).

5. GOOSE KOMUNIKACIJA NA PRIMERU TRI RELEJA

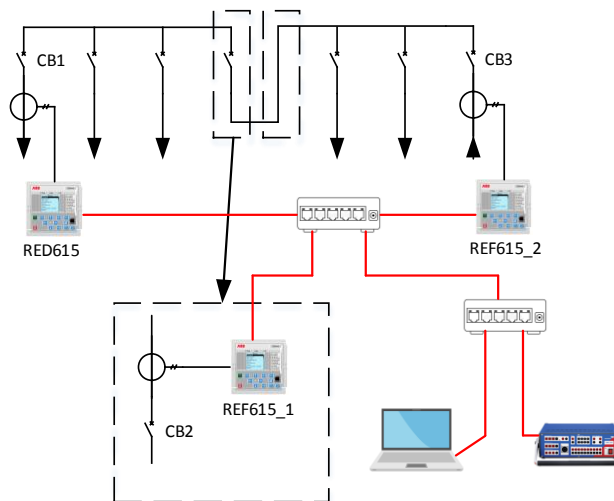
PCM600 (engl. *Protection and Control IED Manager*) softver predstavlja okruženje za podešavanje i ispitivanje releja proizvođača ABB. On je usklađen sa IEC61850 protokolom i pojednostavljuje inženjering releja i omogućava razmenu informacija sa drugim uređajima koji su takođe usklađeni sa IEC 61850 protokolom.

Osnovna uloga GOOSE komunikacije jeste prenos informacija između releja na nivou jedne stanice. Za potrebe ovog rada koriste se 3 releja proizvođača ABB. Koristi se jedan RED615 i dva REF615 releja.

RED615 je mikroprocesorski relej namenjen prvenstveno za podužnu diferencijalnu zaštitu nadzemnih vodova i kablovskih izvoda.

REF615 je mikroprocesorski relej za zaštitu nadzemnih vodova i kablovskih izvoda. Kao zaštitna funkcija koristi se zaštita od otkaza prekidača (CCBRBRF).

Nakon pojave struje kvara u polju u kojem se nalazi prekidač CB1, relej RED615 šalje komandu za otvaranje prekidača CB1. Nakon izdavanja komande za otvaranje prekidača, kontakti prekidača su ostali zatvoreni. Zatim, potrebno je da relej ponovo pošalje komandu za otvaranje prekidača (Re-Trip) i ukoliko prekidač i dalje bude zatvoren potrebno je preko GOOSE komunikacije proslediti informaciju releju REF615_1 o otkazu prekidača CB1. Zatim, relej REF615_1, koji štiti sabirnice, treba, takođe preko GOOSE komunikacije, da prosledi informaciju svim relejima koji se nalaze na sabirnici da izdaju komandu za otvaranje prekidača da bi se izolovao prekidač CB1 koji je u kvaru.

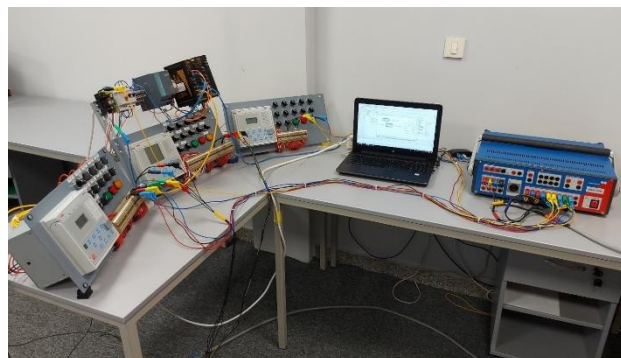


Slika 4. Pozicije prekidača i releja u mreži

Za potrebe praktičnog rada korištene su prekostrujna zaštita (PHHPTOC) i zaštita od otkaza prekidača (CCBRBRF). Nakon završenog podešenja zaštita potrebno je podesiti parametre GOOSE komunikacije u sva tri releja.

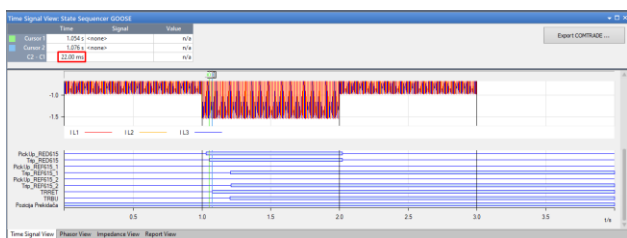
5.1. Ispitivanje GOOSE komunikacije

Sva tri releja su povezana na upravljivi svič RuggedSwitch RS900. On je zatim povezan na drugi manji neupravljivi svič D-Link DES-1005D. Na njega su takođe povezani računar i test set CMC 356, proizvođača OMICRON, koji je korišćen za injektovanje struja i simulaciju kvara. Za potrebe ispitivanja GOOSE komunikacije u sva tri releja injektovane su struje samo u prvoj fazi. To je dovoljan uslov za prorađu zaštite.



Slika 5. Izgled testnog okruženja za ispitivanje GOOSE komunikacije

Za prvi test, zaštita od otkaza prekidača podešena je tako da prati samo poziciju prekidača. To znači da će izlaziti iz bloka zaštite za Re-Trip i BackUp-Trip ostati aktivni sve dok je prekidač zatvoren, bez obzira na vrednost struje. Vreme Re-Trip signala podešeno je na 20ms, a vreme BackUp-Trip signala na 150ms. Na slici 6. prikazani su odzivi vremenskih signala.



Slika 6. Prikaz vremenskih signala

Za drugi test, zaštita od otkaza prekidača podešena je tako da prati samo vrednost struje. To znači da će izlaziti iz bloka zaštite za Re-Trip i BackUp-Trip ostati aktivni sve dok je vrednost struje veća od podešene vrednosti u koloni *Current Value*, bez obzira na poziciju prekidača. Vreme Re-Trip signala podešeno je na 20ms, a vreme BackUp-Trip signala na 150ms.

Za treći test, zaštita od otkaza prekidača podešena je tako da prati vrednost struje i poziciju prekidača. To znači da će izlaziti iz bloka zaštite za Re-Trip i BackUp-Trip ostati aktivni sve dok vrednost struje ne padne ispod podešene vrednosti u koloni *Current Value* ili dok prekidač ne otvori svoje kontakte. Vreme Re-Trip signala podešeno je na 20ms, a vreme BackUp-Trip signala na 150ms.

Snimanje i praćenje paketa podataka koji se šalju kroz mrežu odrađeno je *Wireshark* softverom. *Wireshark* je najrasprostranjeniji softver za analizu mrežnih protokola. Omogućava praćenje dešavanja u mreži sa veoma velikom preciznošću. *Wireshark* softver ima širok set funkcija, neke od najbitnijih su: snimanje paketa na mreži, filtriranje paketa, vizuelizacija podataka.

6. ZAKLJUČAK

U ovom radu opisana je komunikacija između uređaja na nivou transformatorske stanice i komunikacija prema SCADA sistemu. Opisana su tri osnovna principa komunikacije IEC61850 protokola. To su MMS komunikacija, GOOSE komunikacije i Sampled Values komunikacija. Pažnja je posvećena modelovanju podataka u okviru IEC61850 protokola.

Praktičan deo rada obuhvatio je detaljan opis softvera za podešavanja ABB releja – PCM600, opis i podešavanje zaštite od otkaza prekidača i podešavanje GOOSE komunikacije. Dobijeni rezultati testiranjem GOOSE komunikacije pokazuju da nakon otkaza prekidača, relej šalje Re-Trip komandu za ponovno otvaranje prekidača.

Nakon novog neuspešnog pokušaja otvaranja prekidača, relej preko GOOSE komunikacije šalje releju hijerarhski iznad sebe informaciju o otkazu prekidača. Taj relej nakon prijema informacije, preko GOOSE poruke šalje komadu svim ostalim relejima, na dovodnim i odvodnim poljima, da izdaju komandu za otvaranje prekidača i na taj način izoluju kvar.

7. LITERATURA

- [1] S. Škuletić, Z. Sekulić, „Proizvodnja električne energije,” Zavod za udžbenike i nastavna sredstva, Podgorica, 2019.
- [2] Dajana Cvijić, „GOOSE Komunikacija,” Master rad, Fakultet tehničkih nauka Novi Sad, 2020.
- [3] Shawn Nick, „An investigation approach to test Protection Intelligent Electronic Devices (IEDs) in IEC 61850 based Substation Automation Systems (SAS) at Station level,” Science and Engineering Faculty, Brisbane, 2014.
- [4] S. Sučić, A. Martinić, H. Keserica, „Utjecaj drugog izdanja IEC 61850 standarda na razvoj komunikacijske arhitekture elektroenergetskog sustava,” Hrvatski ogranak međunarodne elektrodistribucijske konferencije, Umag, Hrvatska 2010.
- [5] Matija Skelo, „Pregled komunikacijskih sustava u pametnim elektroenergetskim mrežama,” Završni rad, Osijek, 2017.

Kratka biografija:



Boško Jovanović rođen je u Nürnberg-u 1996. godine. Osnovne akademske studije uspio je 2015. godine na Fakultetu tehničkih nauka, studijski program Energetika, elektronika i telekomunikacije. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Električne mašine odbranio je 2019. god. Master studije upisao je 2019. god. Kontakt: bosko.jovanovic.bnb@gmail.com



Boris Dumnić rođen je 1976. godine. Zaposlen je na Fakultetu tehničkih nauka, Departmanu za energetiku, elektroniku i telekomunikacije, Katedri za energetsku elektroniku i pretvarače. Oblasti interesovanja su mu električne mašine, pogoni, energetska elektronika i obnovljivi izvori električne energije.

SOFTVERSKA PLATFORMA ZA ANALIZU PODATAKA O RADU SOLARNIH FOTONAPONSKIH SISTEMA**SOFTWARE PLATFORM FOR ANALYSIS OF DATA ON THE OPERATION OF SOLAR PHOTOVOLTAIC SYSTEMS**

Aleksandar Magoč, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je opisan značaj obnovljivih izvora energije, sa akcentom na solarnu energiju. Način na koji se pretvara solarna energija u električnu sa detaljima konfiguracije fotonaponskih elektrana je predstavljen. Opisan je sistem nadzora fotonaponskih elektrana. Razvijena je softverska platforma čijom primenom se pojednostavljuje detaljna analiza podataka sa invertora fotonaponske elektrane.

Ključne reči: Analiza podataka, obnovljivi izvori, fotonaponske elektrane, softver platforma.

Abstract – This paper describes importance of renewable power sources, with focus on the solar power. Principle of transforming solar power to electrical power with details of configuration of solar power plants are presented. System for supervision of solar power plants is described. Software platform, which simplifies the detailed analysis of data from the solar inverter, is developed.

Keywords: Data analysis, renewable power sources, photovoltaic power plant, software platform.

1. UVOD

Obnovljivim izvorima energije smatraju se nekonvencionalni izvori energije koje nam je priroda dala u izobilju. Oni kao takvi, ne mogu da se potroše, odnosno konstantno se obnavljaju. Zbog trenutne energetske situacije u svetu povećava se interesovanje za energijom iz obnovljivih izvora. Broj stanovnika u na Zemlji raste, a samim tim i potreba za energijom. Dve milijardi ljudi ni dan danas nema pristup električnoj energiji, iako je dvadeset prvi vek.

Solarna energija je ekološki čist i sasvim obnovljiv izvor energije. Za razliku od fosilnih goriva solarna energija je gotovo svuda u svetu dostupna i najvažnije je da je besplatna. Energija sunčevog zračenja predstavlja veoma pogodan izvor energije za grejanje potrošne vode, u domaćinstvu ali i za dobijanje električne energije. Prednosti solarne energije su višestruke: Sunce je neiscrpan izvor energije, korišćenje solarne energije ne zagađuje (hemijski, radioaktivno, ili toplotno) okolinu.

Globalno, ponderisani prosečni troškovi električne energije fotonaponskih sistema opali su za 85% između 2010 i

2020 godine. Sa 0.381 USD po kilovat satu (kWh) na 0.057 USD po kilovat satu, te je godišnje smanjenje bilo oko 7%. Iako su mnoge energetske kompanije i dalje u krhkom finansijskom stanju, postoje znakovi da programeri koriste period predviđen prilagodljivom monetarnom politikom i podrškom vlade za planiranje razvoja infrastrukture i ulaganja u nove projekte.

Investicije u obnovljive izvore napredovale su na tržištima sa dobro uspostavljenim lancima snabdevanja ge su niži troškovi praćeni regulatornim okvirima koji omogućavaju vidljivost novčanog toka. Potražnja korporativnog sektora za čistom električnom energijom radi postizanja ciljeva održivosti takođe je odigrala svoju ulogu [1].

2. TIPIČNE KONFIGURACIJE SOLARNIH ELEKTRANA

Pomoću fotonaponskih panela energija Sunca se pretvara u električnu energiju putem fotonaponskog efekta. Fotonaponski paneli se sastoje solarnih fotonaponskih ćelija koje predstavljaju osnovne elemente svih fotonaponskih sistema. Solarne ćelije se danas uglavnom proizvode od monokristalnog i polikristalnog silicijuma. Savremeni fotonaponski paneli imaju visoku efikasnost, stabilnost i trajnost. Asortiman snage fotonaponskih panela od 10 - 500 Wp zadovoljava najširi opseg i služi za najrazličitije potrebe [2].

Zavisno o načinu rada u odnosu na energetska mrežu, postoje dve vrste fotonaponskih sistema:

1. Samostalni (autonomni),
2. Mrežni

Podela solarnih elektrana se vrši i prema tipu invertora koji se koristi te shodno tome postoje elektrane:

1. Centralizovane koncepcije,
2. String koncepcije i
3. Hibridne koncepcije.

Centralizovana koncepcija je česti izbor prilikom izgradnje fotonaponske elektrane. Formira se fotonaponski niz redno vezanih solarnih panela, time se postiže visoki napon. Ukoliko napon nije dovoljno visok potreban je transformator ili podizač napona. Bitno je voditi računa i o nivou snage, fotonaponski nizovi se povezuju paralelno pomoću string dioda. Centralizovana koncepcija pruža izlaznu snagu sistema i do nekoliko MW. Prednosti ove koncepcije je niža cena i jednostavno održavanje, ali ovakav sistem ima ograničenja. Nedostaci

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Boris Dumnić, van. prof.

su gubici koji se pojavljuju u string diodama, gubici u fotonaponskim modulima, gubici usled ne slaganja napona i gubici snage usled centralizovanog MPPT.

String koncepcija čini jedan fotonaponski niz sa jednim pretvaračem. Svaki niz ima svoj MPPT i time se povećava prinos energije tako što se smanjuje neusklađenost i gubici usled senke. Nije potreban podizač napona niti transformator jer je ulazni napon dovoljno visok. Ova koncepcija je pouzdanija od centralizovane koncepcije. Cena je niža jer je moguća masovna proizvodnja. Efikasnost ovog sistema je 1-3% veća u poređenju sa centralizovanim, nedostatak je povećani trošak zbog broja pretvarača.

Najnovija koncepcija fotonaponskih elektrana je hibridna koncepcija. Ona se primenjuje kod string invertora većih snaga (preko 50kW). Čvorišta se formiraju i na AC i na DC strani invertora. Ova koncepcija predstavlja dalji razvoj string invertora. Kod ove koncepcije nekoliko nizova su povezani sa sopstvenim DC/DC pretvaračem u zajednički DC/AC inverter. Svaki niz se može kontrolisati pojedinačno što je efikasnije nego kod centralizovanog sistema. Hibridni sistem kombinuje prednosti centralizovane i string tehnologije [3].

3. NADZOR NAD RADOM SOLARNIH FOTONAPONSKIH ELEKTRANA

Aurora vision je portal za upravljanje sa postrojenjima koji omogućava:

- Pregled elektrana koje se posmatraju,
- Pregled njihovog trenutnog stanja,
- Pregled zbirne statistike proizvodnje,
- Pregled evidencije događaja
- Detaljne informacije o greškama.

Primarni alat ovog softvera je Menadžer elektrane. Pomoću njega se napravi nalog i dodaju željene elektrane. Postoji mogućnost da informacije o postrojenju budu javne ili privatne.

Menadžer elektrane pruža stalnu vidljivost performansi elektrane. Mogućnost posmatranja više elektrana sa detaljima o njihovoj lokaciji, veličini sistema i geografske lokacije. Tačna lokacija se vrši pomoću Google mapa.



Slika 1. Menadžer elektrane, pregled elektrana

Aurora Vision portal se pokazao kao vrlo koristan alat inženjerima koji nadziru jednu ili više elektrana. U svakom trenutku može se proveriti proizvodnja kao i stanje sistema. Ukoliko se nešto desi i elektrana prestane

sa radom, *Aurora* informiše službu održavanja ili vlasnika. Svaki događaj se memoriše, te se može pregledati i istorija događaja što je vrlo korisno ukoliko se neke greške ponavljaju [4].

4. REALIZACIJA SOFTVERSKJE PLATFORME ZA ANALIZU PODATAKA

Aurora portal se koristi i u našoj zemlji. Inženjeri koriste portal za praćenje rada kako fotonaponskih tako i drugih elektrana. Podaci sa invertora elektrane se snimaju na svakih 15 minuta, što znači da u toku jednog dana memorišu se 96 puta informacije kao što su generisana snaga, generisana energija, napon i struja na AC ili na DC strani i druge. Kako se snimanje vrši svaki dan baza podataka je po prilično velikog formata. Iz tog razloga zgodno je napraviti sopstveni softver pomoću kojeg će se moći analizirati podaci o radu elektrane i na osnovu njih izvlačiti odgovarajući zaključci i uočavati problemi u radu fotonaponskih sistema.

Tema rada jeste kreirati softver pomoću kojeg će se baza podataka preuzeta sa *Aurora* portala u vidu Excel dokumenta analizirati, tj softver će memorisane podatke predstaviti u vidu grafičkog prikaza. Prikazivaće se razni podaci u periodu vremena od jednog dana, mesec dana, više meseci, cela godina i mogućnost prikaza više godina. Korisnik će moći:

- Da se kreće po iscrtanom grafiku i vidi tačne vrednosti u željenim tačkama,
- Imaće mogućnost zumiranja određenog perioda radi uočavanja raznih promena,
- Mogućnost snimanja željenog grafičkog prikaza.

Za izradu ovog softvera koristiće se programski jezik Pajton (eng. *Python*).

4.1. Programski jezik Python

Programski jezik *Python* nastao je početkom devedesetih godina, njegov autor je Gvido van Rosum. *Python* je razvijen u Holandiji na Nacionalnom institutu za matematiku i kompjuterske nauke, njegova sintaksa omogućava pisanje veoma preglednih programa i vrlo je lak za razumevanje. Vrlo brzo programski jezik stekao je veliku popularnost, njegova autorska prava drži neprofitna organizacija tako da se on može koristiti besplatno u bilo koje svrhe.

Koristi za automatizaciju i skriptovanje. Automatizuje se interakcija sa veb pretraživačima ili GUI-jevima (GUI-Graphical user interface), koristi se za opšte programiranje aplikacija, primenjuje se u nauci o podacima za mašinsko obučavanje, koristi se za metaprogramiranje. Treba pomenuti da postoje i zadaci za koje *Python* nije pogodan, njime nije pogodno programirati sistem izbegavati upravljačke programe uređaja ili jezgro operativnog sistema, nije pogodan u situacijama kada se pozivaju binarne platforme, a najveća mana *Python*-a jeste njegova brzina izvršavanja aplikacije (u tom slučaju koriste se C ili C++). Količina vremena potrebnog da se izvrši program napisan u *Python*-u je nadoknađena vremenom ušteđenim u razvojnom procesu programa pomoću *Python*-a. Može pisati raznim tekst

editorima kao što su: Notepad, Notepad++, PyCharm, VS Code, Sublime text editor [5].

4.2. Realizacija alata za analizu podataka

Alat za analizu podataka sa solarne elektrane je razvijen u programskom jeziku *Phyton*. *Phyton* je trenutno jedan od najpogodnijih programskih jezika za analizu podataka. Pored toga što je relativno jednostavan, a tako praktičan i rasprostranjen to je jedan od razloga zašto je ovaj alat pisan u programskom jeziku *Phyton*.

Kao podrška za vizuelni prikaz i sam korisnički interfejs korišćen je PyQt5 paket za *Phyton* koji je pored PySide jedna od besplatnih verzija Qt programa, sa malim razlikama u korisničkoj licenci.

Alat je pravljen za specifičan format baze podataka prikupljen sa Aurora portala u cilju mogućnosti primene alata nad bilo kojom bazom podataka prikupljene pomoću Aurora portala. Ukoliko bi se bilo koja baza podataka svela na isti format kao sa Aurora portala, ovim alatom bi se mogla izvršiti željena grafička analiza. Za razvoj alata koristi se biblioteka Pandas koja pruža visoke performanse u procesu analize baza podataka na pojednostavljen način. Kôd za grafički korisnički interfejs (GUI) pisan u *Phyton* programskom jeziku uz biblioteku PyQt se poziva kao klasa. Metode te klase predstavljaju funkcije samog korisničkog interfejsa.

5. ANALIZA PODATAKA POMOĆU RAZVIJENOG SOFTVERA

Fotonaponska elektrana čije informacije su bile na Aurora portalu nalazi se na teritoriji opštine Kikinda u selu Sajan. Snaga elektrane je 536kW sa položajem severne geografske širine 45° 50' i istočne geografske dužine 20° 17'.



Slika 2. Fotonaponska elektrana Sajan

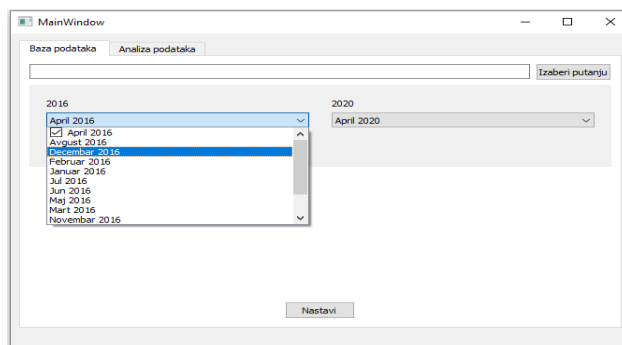
Elektrana poseduje transformator sa svom zaštitnom i sklopnom opremom radi priključivanja na distributivni sistem električne energije. Tokom dana elektrana snabdeva distributivnu mrežu sa energijom i napaja domaćinstva.

5.1. Opis rada softvera za analizu

Grafički korisnički interfejs (GUI) je osmišljen i realizovan tako da bude što jednostavniji za upotrebu krajnjem korisniku, slika 3.

Prozor sadrži dve kartice, „Bazu podataka” koja je inicijalno otvorena i „Analiza podataka”. Prvo što je potrebno uraditi jeste da se učitaju podaci, odnosno da se učita baza podataka. Klikom na dugme „Izaberi putanju” otvoriće se prozor pomoću kojeg je potrebno doći do

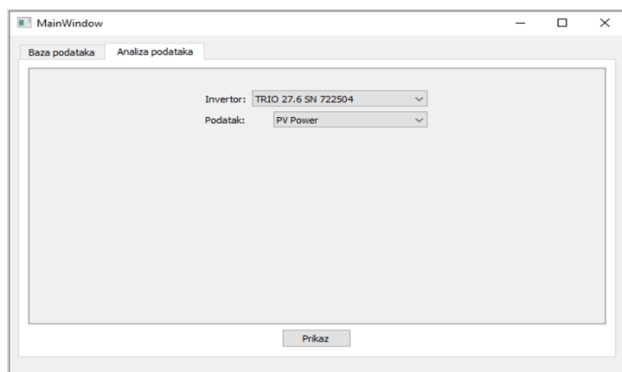
željene lokacije gde su locirani Excel dokumenti koji su prethodno snimljeni sa Aurora portala. Kako bi se počela analiza, na padajućem meniju izvrši se selekcija meseci za koje želimo da uradimo analizu, levim klikom miša pojavice se kvačica koja nam potvrđuje odabir željenih meseci. Izbor može biti jedan mesec, svi ili više meseci.



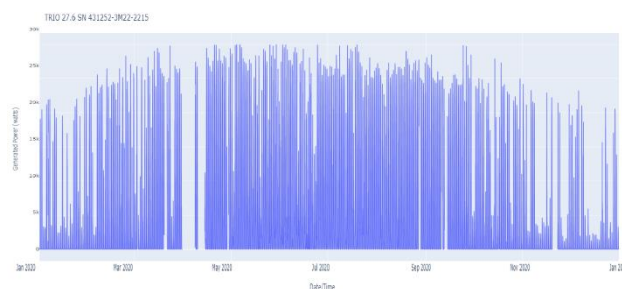
Slika 3. Glavni prozor nakon učitane baze podataka

Kada je odabran željeni mesec sa kojim će se početi analiza, klikom na dugme „Nastavi” prelazi se na karticu „Analiza podataka”. Zbog velikog obima baze podataka biće potrebno malo vremena da se podaci obrade, pogotovo ako se izabere posmatranje cele godine. Na brzinu učitavanja baze utiču i performanse računara na kojem se pokreće program, slika 4. i slika 5.

Program uđe u Excel tabelu, proveriti sve kartice koje zapravo predstavljaju različite invertore i na svakoj kartici od svakog invertora preuzme sve snimljene podatke. Na ovaj način postoji mogućnost odabira invertora čije podatke želimo da analiziramo i koji podatak od izabranog invertora želimo da analiziramo. Ove dve opcije su takođe predstavljene pomoću padajućeg menija, biramo šta želimo da analiziramo i na kojem invertoru.



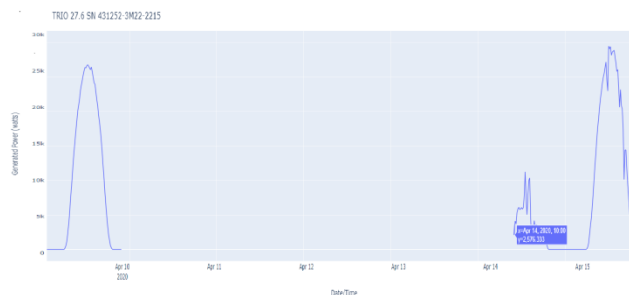
Slika 4. Prikaz kartice analize podataka



Slika 5. Grafički prikaz generisane snage za celu 2020 godinu

Kada se prikaže jedan mesec po X osi prvo će se videti dani u mesecu, a kada se zumira jedan poseban dan videće se prikaz u minutama. Na Y osi biće prikazani željeni podatak analize. Koje sve podatke možemo posmatrati zavisi šta se sve prati i memoriše sa invertora.

Kretanjem kroz grafik uz pomoć uveličavanja željenog perioda može se detaljnije videti kada je proizvodnja stala i kada se vratila. Kada se kursom miša dotakne linija podataka, pojaviće se pravougaonik sa detaljima u toj tački.



Slika 6. Grafički prikaz nestanka proizvodnje u aprilu sa povratkom proizvodnje

Alat je vrlo jednostavan za korišćenje i praktičan. Funkcionalan je za sve podatke koje *Aurora portal* memoriše. Vrlo bitna karakteristika jeste to što nije ograničen brojem podataka koji se analiziraju kao na primer Excel.

6. ZAKLJUČAK

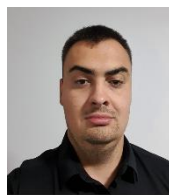
Ideja ovog rada je nastala na master studijama kada je bilo potrebno da se analiziraju podaci dobijeni sa fotonaponske elektrane. Baza podataka nije bila velika i uz pomoć Excela zadatak analize je bio ispunjen. Ozbiljne analize zahtevaju duži period posmatranja rada elektrane, a time se utiče i na veličinu baze podataka. Glavni doprinos ovog rada je kreiran softver pomoću kojeg se uspešno dolazi do kvalitetnih podataka u različitim vremenskim okvirima (godišnjem, sezonskom, mesečnom i dnevnom nivou), neophodnih za donošenje zaključaka o eventualnim nastalim zastojevima na radu panela. Ova platforma omogućuje bolji pristup i analizu podataka od komercijalna rešenja, jer je prilagodljiva za bilo koju elektranu. Kada se baza učitava za svega par trenutaka mogu se predstaviti grafički prikazi raznih podataka i početi analiza radi otkrivanja potencijalnih grešaka i problema u radu elektrane.

Sam eksperiment će sigurno uticati na poboljšanje rada prikazanog softvera, potencijalnog razvoja i daljeg usavršavanja. Sa ovim softverom moguće je u budućem periodu stvoriti i dodatne funkcije koje bi uticale na poboljšanje rada elektrane, a samim tim bi se smanjio i trošak njihovog održavanja.

7. LITERATURA

- [1] „Energija za budućnost.” 2017, dostupno na: <http://www.astronomija.org.rs/nauka/fizika/2427-energija-za-budunost>
- [2] E. Romero-Cadaval, G. Spagnuolo, L. G. Franquelo, C. A. Ramos-Paja, T. Suntio, and W. M. Xiao, „Grid-Connected Photovoltaic Generation Plants: Components and Operation,” *IEEE Ind. Electron. Mag.*, vol. 7, no. 3, pp. 6–20, Sep. 2013, doi: 10.1109/MIE.2013.2264540.
- [3] L. Hassaine, E. OLias, J. Quintero, and V. Salas, „Overview of power inverter topologies and control structures for grid connected photovoltaic systems,” *Renew. Sustain. Energy Rev.*, vol. 30, pp. 796–807, Feb. 2014, doi: 10.1016/j.rser.2013.11.005.
- [4] „Plant Portfolio Manager- Aurora Vision platform” 2021, dostupno na: <https://www.fimer.com/aurora-vision-plant-management-platform/plant-portfolio-manager>
- [5] „Python” 2021, dostupno na: [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))

Kratka biografija:



Aleksandar Magoč rođen je u Vrbasu 1992. godine. Osnovne akademske studije uspio je 2011 godine na Fakultetu tehničkih nauka, studijski program Energetika, elektronika i telekomunikacije. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi odbranio 2017. godine. Master studije upisao je 2017. godine.
Kontakt: alexmagic53@gmail.com



Boris Dumnić rođen je 1976. godine. Zaposlen je na Fakultetu tehničkih nauka, Departmanu za energetiku, elektroniku i telekomunikacije, Katedri za energetsku elektroniku i pretvarače. Oblasti interesovanja su mu električne mašine, pogoni, energetska elektronika i obnovljivi izvori električne energije.

SISTEM ZA PRAĆENJE FINANSIJSKIH ASPEKATA UGOVORA**SYSTEM FOR TRACKING FINANCIAL ELEMENTS OF CONTRACTS**Marina Vojnović, *Fakultet tehničkih nauka, Novi Sad***Oblast – SOFTVERSKO INŽENJERSTVO I INFORMACIONE TEHNOLOGIJE**

Kratak sadržaj – U ovom radu predstavljeni su značaj i primena sistema za upravljanje dokumentima, kao i zahtevi koji se stavljaju pred njih, sa posebnim osvrtom na sisteme za upravljanje ugovorima. Analizom sprovedenih istraživanja i rešenja iz oblasti, uz osluškivanje trenutnih zahteva tržišta, specificirano je i implementirano rešenje za praćenje finansijskih aspekata ugovora, koje je dokumentovano u ovom radu.

Ključne reči: upravljanje ugovorima, ugovori, fakture, plaćanja

Abstract – This paper presents the importance and usage of document management systems, as well as the requirements set in front of them, with special reference to contract management systems. Analysing conducted researches and solutions in the field, while listening to the current market demands, has led to specification and implementation of a solution for tracking the financial elements of contracts, documented in the paper.

Keywords: contract management, contract, invoice, payment

1. UVOD

Ugovori predstavljaju obavezujući odnos između preduzeća i njihovih dobavljača, poslovnih partnera ili klijenata i obuhvataju suštinski sporazum priznat zakonom o ključnim uslovima i odredbama trgovine ili usluga, poput cena, plaćanja, odgovornosti, rokova i sl. [1].

Ugovori pored toga što su sredstva pravnog obezbeđenja firme po pitanju potraživanja i dugovanja, takođe su i izvor, pa i garancija, većine njihovih prihoda. Ipak, s obzirom na tradicionalne papirne formate za njihovo prezentovanje i čuvanje, a neophodnost svakodnevnog rukovanja istima, koje rezultuje značajnim oduzimanjem vremena, oni zapravo velikim delom potencijalno mogu predstavljati ozbiljan rashod poslovanja preduzeća.

Iz toga proizilazi da automatizacija nekih zadataka upravljanja ugovorima tokom njihovog životnog ciklusa predstavlja značajnu mogućnost stvaranja vrednosti za preduzeće, što i jeste bila suština sistema za upravljanje ugovorima, detaljnije opisanog u ovom radu.

1.1. Stanje u oblasti

Sprovedena su brojna istraživanja u oblasti i identifikovani su različiti benefiti. Nekoliko istraživačkih kompanija dalo je čak procene o budućim prihodima koji proističu iz primene rešenja. Prema međunarodnoj kompaniji za istraživanje tržišta *AMR Research*, implementacije sistema za upravljanje ugovorima traju dva do tri meseca i postižu otplatu od 150% do 250% u roku od jedne godine.

Međunarodna marketinška kompanija usmerena na prikupljanje i analiziranje podataka o ponašanju kupaca, *Aberden Group*, izvestila je da je povraćaj ulaganja veći od 3 puta već u prvoj godini za organizaciju sa 750 miliona dolara u godišnjim prihodima. Procenjuje se da organizacija vredna milijardu dolara sa godišnjim kupovinama od ukupno 500 miliona dolara, od čega 400 miliona je pod ugovorom, gubi 18 miliona svake godine zbog nedostatka odgovarajućeg sistema za upravljanje ugovorima sa dobavljačima.

Dalje, vreme, trud i troškovi za ručno upravljanje ugovorima koštaju dodatnih 12 miliona dolara godišnje. Iako će se procene razlikovati zavisno od organizacije i prema pojedinačnim pokazateljima, očito je da se istraživačke grupe i organizacije slažu da implementacija sistema za upravljanje finansijskim aspektima ugovora, ako se izvrši temeljno, može pomoći u smanjenju troškova, maksimiziranju prihoda i minimiziranju rizika za svaku organizaciju [2,3].

Sistem za upravljanje ugovorima značajno doprinosi smanjenju faktora rizika i nepredvidivosti za posao, ali i klijente [4,5,6]. Garantuje uspeh pri dobavljanju, jer zahteva performanse ugovarača prema definisanom redosledu, vremenu i specifikaciji definisanoj u ugovoru. [7,8,9]. Pored pojedinačnih kompanija i organizacija, primena bi našla mesto i na nivou makroekonomije.

Različite vlade širom svetske ekonomije izgubile su značajne iznose usled slabe prakse upravljanja sistemima ugovora. Takve prakse mogu biti preispitane i obnovljene kroz regulatorne propise koje definišu relevantne vlasti ili odeljenja u regionu. Među različitim pitanjima, ključni pokazatelji pokrivaju rešavanje problema sa kupcima, isporuku projekata, faktor kvaliteta, usklađenost budžeta i zadovoljstvo korisnika [10,11,12].

Takođe, ispitani su troškovi uključeni u ugovore o nabavkama i objašnjena je njihova važnost u javnim nabavkama za uzimanje vrednosti [13,14]. Osim toga, poboljšanje zdravstvenog sistema putem upravljanja ugovorima je takođe sagledano i empirijski opravdano [15,16,17].

Osnovni zahtevi za koje se očekuje da ovi sistemi ispune su [2,18]:

NAPOMENA:

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

- Obezbeđenje skladištenja, pretraživanja, sortiranja, upravljanja i izveštavanja o ugovorima i povezanim dokumentima
- Pretraživanja na osnovu ključnih polja kao što su broj dobavljača, datum početka i završetka, ime nosioca ugovora
- Pružanje funkcionalnosti za povezivanje dokumenata ili stvaranje hijerarhije podređenih roditelja među dokumentima
- Analitika i alati za reviziju
- Mehanizam za izveštavanje
- Mogućnost integrisanja postojećih mehanizama za izveštavanje organizacija
- Izveštavanje zasnovano na ulogama i kontrolne table

2. SPECIFIKACIJA SISTEMA

Detaljnou analizom, uočavanjem i sagledavanjem sličnih rešenja i različitih aspekata upravljanja ugovorima, kao i njihovim daljim prilagođavanjem kroz direktnu saradnju sa klijentima, formirana je specifikacija sistema, koja na visokom nivou zadovoljava sve identifikovane funkcionalne i nefunkcionalne zahteve, zarad postizanja sveobuhvatnog najboljeg korisničkog iskustva. Za prezentaciju i opis korišćen je *Unified Modelling Language (UML)*¹.

2.1. Dijagram slučajeva korišćenja

Sistem za upravljanje ugovorima predviđen je za korišćenje od strane dve vrste korisnika: administratora kompletnog sistema i prijavljenog korisnika na nivou preduzeća.

Korisnik ima na raspolaganju sledeće funkcionalnosti:

- Dodavanje ugovora, faktura, tendera, komitenata, plaćanja, izvoda, zaposlenih, dokumenata zaposlenih
- Izmena ugovora, faktura, tendera, komitenata, plaćanja, izvoda, zaposlenih, dokumenata zaposlenih
- Brisanje ugovora, faktura, tendera, komitenata, plaćanja, izvoda, zaposlenih, dokumenata zaposlenih
- Pregled statistike
- Kreiranje naloga zaposlenim licima u preduzeću

Funkcionalnosti administratora obuhvataju sve funkcionalnosti redovnog korisnika sistema, uz dodatak sledećih:

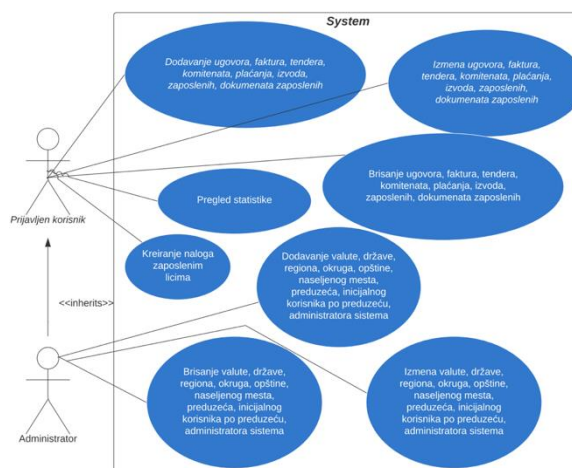
- Dodavanje valute, države, regiona, okruga, opštine, naseljenog mesta, preduzeća, inicijalnog korisnika po preduzeću, administratora sistema
- Izmena valute, države, regiona, okruga, opštine, naseljenog mesta, preduzeća, inicijalnog korisnika po preduzeću, administratora sistema
- Brisanje valute, države, regiona, okruga, opštine, naseljenog mesta, preduzeća, inicijalnog korisnika po preduzeću, administratora sistema

Dijagram slučajeva korišćenja prikazan je u nastavku na slici 1.

2.3. Dijagram klasa

Dijagram klasa predstavljen je na slici 2.

Sivom bojom označene su klase čijim objektima upravlja administrator sistema i to su:



Slika 1. Dijagram slučajeva korišćenja.

- Radnik – zaposleno lice u preduzeću
- Dokument radnika – *upload*-ovani dokumenti koji pripadaju ili se odnose na radnika preduzeća
- Tip identifikacije – služi za grupisanje dokumenata u kategorije
- Preduzeće – organizacija koja je pretplaćena i koja je korisnik sistema
- Komitent – poslovni saradnik preduzeća
- Organizaciona jedinica – sektor unutar preduzeća
- Naseljeno mesto – sedište preduzeća
- Opština – opština kojoj pripada naseljeno mesto
- Okrug – okrug kome pripada opština
- Region – region kome pripada okrug
- Država – država kojoj pripada region
- Valuta – monetarna jedinica razmene

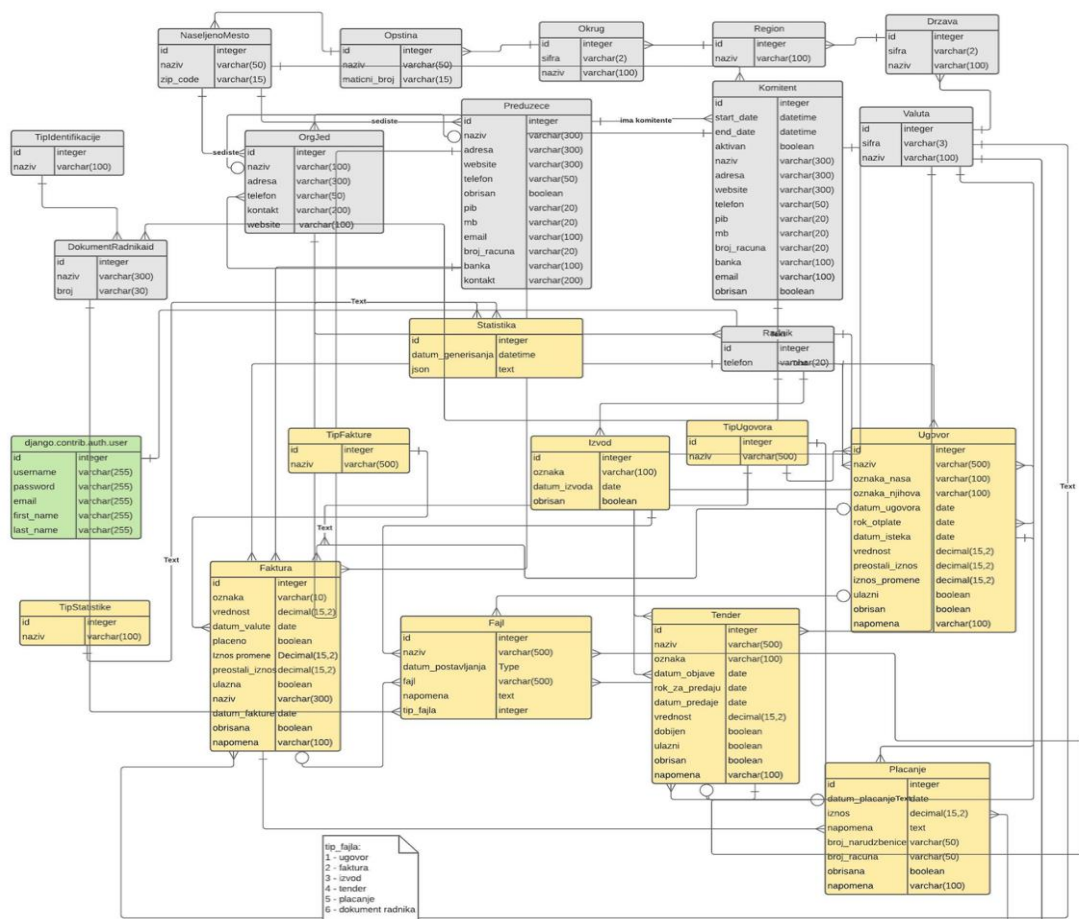
Žutom bojom su označene klase čijim objektima, pored administratora, upravljaju i redovni korisnici sistema i to su:

- Ugovor – pravno sredstvo obezbeđenja po nekom predmetu između dve ili više ugovornih strana
- Tip ugovora – vrsta ugovora, odnosno da li je ulazni ili izlazni
- Tender – specifikacija ponude koja se iznosi na javnom nadmetanju
- Plaćanje – izmirenje dela obaveza po ugovoru ili fakturi
- Faktura – serijski vid realizacija obaveza po ugovoru
- Tip fakture – vrsta fakture, odnosno da li je ulazna ili izlazna
- Izvod – zvanični izveštaj banke o stanju na računu tokom meseca
- Fajl – *upload*-ovani dokument
- Statistika – izveštaji o poslovanju preduzeća
- Tip statistike – vid kategorizacije statističkih izveštaja

Zelenom bojom označena je klasa *Django REST Framework*-a, koja se koristi za autentifikaciju korisnika i to je:

- `django.contrib.auth.User`

¹ <https://www.tutorialspoint.com/uml/index.htm>



Slika 2. Dijagram klasa.

3. IMPLEMENTACIJA

Serverska aplikacija implementirana je u *Python*² programskom jeziku u *Django REST Framework*-u³, dok je klijentska u *JavaScript*-u⁴ pomoću *Angular*⁵ radnog okvira.

3.1. Klijentska aplikacija

Prateći konvenciju strukturiranja sistema pomoću *NgModules*, *Angular* aplikacija sastavljena je iz više modula, koji imaju ulogu kontejnera koji sadrže komponente, templejte, servise i druge datoteke posvećene određenom domenu aplikacije.

Kako su gotovo sve funkcionalnosti aplikacije ograničene na prijavljene korisnike, te je ta informacija neophodna kroz kompletnu upotrebu aplikacije, za upravljanje korisničkim nalogima korišćen je *Redux*, konkretno *ngxs* biblioteka.

Token koji se na taj način skladišti se pomoću interfejsa *HttpInterceptor* prilikom svakog slanja zahteva ka serverskoj strani dodaje u zaglavlje.

3.2. Serverska aplikacija

Sledeći konvenciju *Django REST Framework*-a, projekat *mojiugovori* razdvojen je na dve *Django* aplikacije – *ugovori* i *orgsema*. Aplikacija *orgsema* napravljena je sa ciljem efikasnog strukturiranja radnika po organizacionim jedinicama preduzeća, te preduzeća po naseljenim mestima u okviru opština, okruga, regiona i naposljetku države. Kroz ovu aplikaciju vrši se dodavanje i izmena radnika i komitenata, kao i pretraživanje i prikaz prethodno spomenutih entiteta. Aplikacija *ugovori* namenjena je za upravljanje ugovorima, tenderima, fakturama, izvodima i njihovim fajlovima, kao i za formiranje statistika. Autentifikacija i autorizacija su regulisane *Django auth* modulom. Za skladištenje podataka korišćena je *mysql* baza.

4. ZAKLJUČAK

U ovom radu najpre je predstavljeno stanje u oblasti sistema elektronskog upravljanja ugovorima, zahtevi koji se stavljaju pred njih, benefiti koji se korišćenjem ostvaruju, kao i konkretni statistički pokazatelji opravdanosti ulaganja i povraćaja investicije. Potom je predstavljena specifikacija našeg rešenja, predstavljene su tehnologije u kojima je realizovana i okvir arhitekture klijentskog i serverskog dela. Cilj rešenja je olakšanje uvida u finansijsko stanje preduzeća i obezbeđenje likvidnosti, praćenja trendova i rezultata prethodno donetih odluka, formiranje strategija daljeg upravljanja,

² <https://www.python.org/>

³ <https://www.django-rest-framework.org/>

⁴ <https://www.javascript.com/>

⁵ <https://angular.io/>

razvoja i skaliranja, blagovremeno reagovanje i ispunjenje obaveza plaćanja, kao i direktan uvid u transakcije bitne za računovodstvo. Eliminisan je potencijal ljudskih grešaka i narušavanja politika privatnosti i poverljivosti podataka.

Kako primena ovog softvera direktno utiče na povećanje efikasnosti i produktivnosti unutar preduzeća, prirodno je očekivati napretke i promene u organizacionoj kulturi, širenje domena i/ili obima poslovanja, te postoji mogućnost za pojavljivanjem novih zahteva koje bi sistem trebao da podrži kako bi se ove promene ispratili. Iz tog razloga vođeno je računa o tehničkoj implementaciji i mogućnosti njenog proširivanja i prilagođavanja. Kao jedan od pravaca daljeg razvoja jeste podrška za ekstrakciju informacija iz samih ugovora i faktura i automatsko popunjavanje baze podataka korišćenjem tehnika veštačke inteligencije.

5. LITERATURA

- [1] Don Lavoie, Emily Chamlee-Wright, „Culture and Enterprise, The development, representation and morality of business“, Cato Institute, 2000.
- [2] Anuj Saxena, Enterprise contract management, A practical Guide to Successfully Implementing an ECM Solution, 2008
- [3] Turner, J. R., Simister, S. J. Project contract management and a theory of organization. International Journal of Project Management, 2001
- [4] Haapio, H., & Siedel, G. J. A short guide to contract risk. Routledge, 2017
- [5] Mo, B., Gjelsvik, A., Grundt, A. Integrated risk management of hydro power scheduling and contract management. IEEE Transactions on Power Systems, 2001
- [6] Wiengarten, F., Humphreys, P., Gimenez, C., & McIvor, R. Risk, risk management practices, and the success of supply chain integration. International Journal of Production Economics, 2016
- [7] Ajayi, S. O., Oyedele, L. O., Bilal, M., Akinade, O. O., Alaka, H. A., Owolabi, H. A., & Kadir, K. O. Waste effectiveness of the construction industry: Understanding the impediments and requisites for improvements. Resources, Conservation and Recycling, 2015
- [8] Bayliss, R., Cheung, S.-O., Suen, H. C., & Wong, S.-P. (2004). Effective partnering tools in construction: a case study on MTRC TKE contract 604 in Hong Kong. International Journal of Project Management, 2004
- [9] Kartam, N. A., & Kartam, S. A. Risk and its management in the Kuwaiti construction industry: a contractors' perspective. International Journal of Project Management, 2001
- [10] Oberlender, G. D., & Oberlender, G. D. *Project management for engineering and construction* (Vol. 2). McGraw-Hill New York, 1993
- [11] Klijn, E. H., & Koppenjan, J. The impact of contract characteristics on the performance of public– private partnerships (PPPs). Public Money & Management, 2016
- [12] Santhi, N. S., & Gurunathan, K. B. Fama-French three factors model in Indian mutual fund market. Asian Journal of Economics and Empirical Research, 2014
- [13] Anyanwu, J. O., Okoroji, L. I., Ezewoko, O. F., & Nwaobilor, C. A. The impact of training and development on workers performance in Imo State. Global Journal of Social Sciences Studies, 2016
- [14] Rasheli, G. A. Procurement contract management in the local government authorities (LGAs) in Tanzania: A transaction cost approach. International Journal of Public Sector Management, 2016
- [15] Alexander, J. A., & Morrissey, M. A. A resource-dependence model of hospital contract management. Health Services Research, 1989
- [16] Goddard, B. L. Termination of a contract to implement an enterprise electronic medical record system. Journal of the American Medical Informatics association, 2000
- [17] Hyvönen, T., & Järvinen, J. Contract-based budgeting in health care: a study of the institutional processes of accounting change. European Accounting Review, 2006
- [18] Mchopa, A. Integrating contract management practices into the achievement of value for money in tanzania public procurement: Evidence from selected procuring entities in moshi municipality. Journal of Public Procurement, 2015

Kratka biografija:



Marina Vojnović rođena je u Novom Sadu 1997. god. Osnovne akademske studije završila je na Fakultetu tehničkih nauka 2020. godine. Master rad iz oblasti Elektrotehnike i računarstva – Softversko inženjerstvo i informacione tehnologije odbranila je 2021. god.

kontakt:

marina.vojnovic1997@gmail.com

**UPOTREBA ROBOTSKE AUTOMATIZACIJE PROCESA ZA KREIRANJE
OBRAZOVNIH UPUTSTAVA****USE OF ROBOTIC PROCESS AUTOMATION IN CREATING EDUCATIONAL
TUTORIALS**

Jelena Popov, *Fakultet tehničkih nauka, Novi Sad*

**Oblast – SAVREMENE OBRAZOVNE
TEHNOLOGIJE I STANDARDI**

Kratak sadržaj – U ovom radu objašnjeno je šta robotska automatizacija procesa predstavlja, šta su RPA alati i koji su najpoznatiji. Kreiran je primer uputstva sa ciljem analize upotrebe RPA za kreiranje obrazovnog materijala u obliku video uputstava.

Ključne reči: RPA, RPA alati, UiPath, interaktivna uputstva, obrazovni materijali

Abstract – This paper explains what is robotic process automation, what are RPA tools and what are the most common. An example of tutorial was created with the aim of analyzing the use of RPA to create educational material in the form of video tutorials.

Keywords: RPA, RPA Tools, UiPath, interactive tutorials, educational materials

1. UVOD

Zadatak ovog rada jeste analiza upotrebe robotske automatizacije procesa za kreiranje obrazovnog materijala u obliku video uputstava. Robotska automatizacija procesa (eng. *Robotic Process Automation*, skr. RPA) je softverska tehnologija koja olakšava izradu, primenu i upravljanje softverskim robotima koji oponašaju ljudske radnje u interakciji sa digitalnim sistemima i softverima [1].

Sa razvojem digitalnih tehnologija, povećana je potreba za kreiranjem obrazovnog materijala u obliku video uputstava. Kreiranje ovih materijala je vremenski zahtevan posao koji podrazumeva kreiranje primera, snimanje i montiranje video snimka. Pored toga, njihova izmena zahteva ponovno snimanje pojedinih delova i ponovno montiranje video snimka. Automatizacijom kreiranja uputstava znatno bi bio olakšan posao kreiranja, ali i izmene ovih materijala. Stoga je, u ovom radu predložen metod kreiranja video uputstava upotrebom RPA alata sa nazivom *UiPath*. Ovaj metod je ilustrovan na studiji slučaja kreiranja uputstva za učenje osnova programiranja u Java programskom jeziku.

2. ROBOTSKA AUTOMATIZACIJA PROCESA

Robotska automatizacija procesa je tehnologija koja koristi unapred definisanu poslovnu logiku, utvrđena pravila i strukturirane podatke za automatizaciju poslovnih procesa [2]. Ova tehnologija oponaša način na koji ljudi

komuniciraju sa softverom radi obavljanja često ponavljanih zadataka zasnovanih na pravilima.

U organizacijama koje koriste RPA tehnologije, manuelne i često ponavljane zadatke ne obavljaju ljudi, nego softverski roboti. Softverski roboti obavljaju monotone zadatke mnogo brže, tačnije i efikasnije od ljudi, jer ljudima tokom obavljanja tih zadataka veoma brzo opadne koncentracija i samim tim postaju neefikasni [3]. Uvođenjem RPA tehnologije organizacije smanjuju troškove i potrebu za povećanjem broja zaposlenih, a ujedno povećavaju efikasnost.

2.1. Prednosti i mane

Kao tehnologija koja je još uvek u razvoju, robotska automatizacija procesa donosi mnogobrojne prednosti. U računarskom sistemu, glavni pristup automatizaciji posla je putem pisanja programa, što zahteva poznavanje programskog jezika i kodiranja. Najveća prednost korišćenja robotske automatizacije procesa je to što omogućava automatizaciju bez pisanja programa. Ovo omogućava šire usvajanje tehnologije i lako korišćenje.

Komponenta za snimanje ekrana, koja je ugrađena u sve RPA alate, dodatno olakšava njihovu upotrebu. Snimanjem ekrana se registruje šta je korisnik odradio i na osnovu tih akcija se kreira bot koji će obavljati sav posao umesto korisnika. Bot je softverski program koji obavlja automatizovane, unapred definisane, ponavljajuće zadatke [4]. Obavljanje dosadnih, često ponavljanih zadataka od strane botova ostavlja zaposlenima više vremena da se fokusiraju na složenije poslove koji zahtevaju inteligenciju [3].

Robotska automatizacija procesa može lako da se integriše u postojeće softvere. To znači da organizacije ne moraju da kupuju novi softver, nadograđuju infrastrukturu ili prekvalifikuju zaposlene. RPA ne utiče na postojeće aplikacije i procese, nego se oslanja na već dostupne funkcionalnosti i na dosadašnji način rada zaposlenih [5].

RPA nudi više opcija za postavljanje u radno računarsko okruženje (eng. *deployment*), poput virtuelnih mašina i računara u oblaku (eng. *cloud*). RPA sistemi mogu automatski da *deploy*-uju robote u grupama od po nekoliko stotina. Uz to, više botova može da se koristi za različite zadatke u okviru istog procesa prilikom procesiranja velike količine podataka [5].

Iako RPA u velikoj meri poboljšava poslovanje organizacija, ova tehnologija ima svoje mane. Jedna od glavnih mana je što je i dalje u fazi razvoja, stoga se mnoge organizacije dvoume da li da izvrše automatizaciju

NAPOMENA:

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

sada ili da sačekaju da se ova tehnologija još razvije [5]. Iako su RPA alati veoma jednostavni za korišćenje i ne zahtevaju veštine programiranja, problem je što skaliranje i upravljanje nisu uvek jednostavni. Pored toga, skup zadataka koji se mogu automatizovati upotrebom RPA alata je ograničen. Organizacije ih obično koriste samo za automatizaciju često ponavljanih zadataka zasnovanih na pravilima, jer je još uvek veoma teško automatizovati procese koji zahtevaju ljudsko rasuđivanje.

2.2. Oblasti primene

U današnje vreme mnoge multinacionalne kompanije koriste RPA tehnologiju za automatizaciju svojih svakodnevnih zadataka [6]. To ovakvim kompanijama omogućava da dobijaju tačne, pouzdane i dosledne rezultate.

RPA alati se mogu koristiti za automatizaciju rada osoblja za korisničku podršku. *Chatbots* su inteligentni programi, koji koriste mašinsko učenje zajedno sa obradom prirodnog jezika za proučavanje ljudskog govora ili teksta [6]. Ovi programi se koriste za automatizaciju rada osoblja za korisničku podršku, čime se može poboljšati korisničko iskustvo i znatno smanjiti troškovi u organizaciji.

Robotsku automatizaciju procesa može da koristi i osoblje za IT podršku kako bi daljinski nadgledali i upravljali IT infrastrukturom. Na taj način bi rutinski problemi bili rešavani bez potrebe za ljudskom intervencijom čime bi osoblje imalo više vremena za rešavanje složenijih problema.

Pored toga, RPA se veoma često koristi u računovodstvu za automatizaciju osnovnih zadataka kao što je, na primer, izračunavanje poreza i sastavljanje poreskih prijava [8].

Kompanije u industriji finansijskih usluga koriste RPA za aktivnosti kao što su obrada plaćanja u devizama, automatizacija otvaranja i zatvaranja računa, upravljanje zahtevima za proveru poreza, obrada zahteva za osiguranje itd. [9].

Pored pobrajanih postoje i druge oblasti primene RPA sistema što govori u prilog tome koliko je njihov opseg primene širok, a potencijal za dalji razvoj velik.

3. VIDEO UPUTSTVA

Danas su, pored klasičnih tekstualnih uputstava, u velikoj meri zastupljena i video uputstva. Softverski i hardverski alati za kreiranje, snimanje i montažu video snimaka dostupni su svima po povoljnim cenama ili, čak, besplatno. Uz to je širenju ovog vida edukacije doprineo i razvoj pametnih telefona koji poseduju kvalitetne kamere i dostupni su velikom broju ljudi. Zbog činjenice da većina učenika provodi mnogo vremena na internetu gledajući razne video snimke, video uputstva su preferirani izvor informacija u odnosu na prezentacije, knjige ili tekstualna uputstva.

Glavna prednost video uputstava, u odnosu na tekstualna, je mogućnost praćenja celog postupka realizacije nekog zadatka. Pored toga, video uputstva su pogodna i za dodatne materijale jer omogućavaju učenicima da gledaju nastavnika kako objašnjava to gradivo. Dodatne prednosti video materijala, koje učenici ističu u odnosu na klasično obrazovanje, su mogućnost ponovnog gledanja, premotavanje, pauziranje, usporavanje i ubrzavanje po potrebi [10].

Iako su video uputstva veoma pogodna za učenike, za nastavnike je kreiranje video materijala vremenski zahtevan posao, jer često podrazumeva ponovno snimanje određenih delova usled grešaka ili nepredviđenih opstrukcija i njihovu kasniju montažu u finalni video snimak. Takođe, usled potrebe za ažuriranjem delova video uputstava potrebno je ponoviti netrivialan postupak snimanja i montaže video materijala. Ove mane mogu nastavnike odbiti od želje za snimanjem video materijala i pored svih benefita koje oni donose njihovim učenicima.

Veliki broj trenutno dostupnih obrazovnih materijala u obliku video uputstava namenjen je programerima. Ovaj vid edukacije je za njih pogodan jer, za razliku od tekstualnih uputstava koja sadrže samo teorijska objašnjenja, sadrži i celokupan postupak kreiranja projekta. Programiranje podrazumeva pisanje koda u određenom redosledu, što se u tekstualnom uputstvu ne može prikazati. Takođe, programski kod se najčešće ne piše linearno, nego se postepeno dopunjuje novim linijama programskog koda između već postojećih linija. Kreiranje ovakvih video uputstava podrazumeva pripremu teorijskog materijala, osmišljavanje projekta za demonstraciju prethodno uvedenih teorijskih koncepata, snimanje i montiranje video snimka. Iz prethodno opisanog može se zaključiti da su video uputstva veoma pogodna za edukaciju programera, ali su istovremeno i zahtevna za kreiranje zbog prethodno opisanih složenih koraka koji se moraju ponoviti u slučaju greške.

3.1. Primena RPA u obrazovanju

Pored oblasti navedenih u poglavlju 2.2, robotska automatizacija procesa se veoma često koristi i u obrazovanju. Fokus rada je analiza upotrebe ove tehnologije u obrazovanju, te će ova oblast primene biti detaljnije objašnjena u nastavku poglavlja.

Učitelji i nastavnici, bilo da predaju nižim ili višim razredima, imaju zadatak da prenesu znanje učenicima na najbolji način. Pored toga, oni imaju veliku količinu administrativnih poslova koji im oduzimaju dosta vremena. Do sada je robotska automatizacija procesa najčešće korišćena za automatizaciju takvih poslova sa ciljem da nastavno osoblje ima što više vremena za edukaciju učenika.

Početkom svake godine, obrazovne institucije troše mnogo vremena na upis učenika. Umesto da učitelji proveravaju da li su učenici i roditelji dobro popunili sve obrasce, da li su učenici zadovoljili sve kriterijume za upis i slično, ovaj monoton posao se može poveriti softverskim robotima [11].

U današnje vreme se nastava veoma često obavlja onlajn. Onlajn nastava sa sobom donosi brojne zadatke koje nastavno osoblje mora da obavi. Ti zadaci uključuju upravljanje prisustvom, proveru domaćih zadataka, kačenje nastavnog materijala, praćenje napretka kursa itd. RPA može da upravlja sadržajem na svakoj platformi, prati i beleži prisustvo učenika, automatski obaveštava roditelje učenika koji često ne prisustvuju na časovima, automatizuje proveru domaćih zadataka i slično, tako da nastavnici mogu da se fokusiraju na sam obrazovni proces [12].

Iz navedenog se zaključuje da je RPA veoma pogodan za automatizaciju administrativnih poslova, koji nepotrebno oduzimaju vreme nastavnom osoblju. To je razlog zbog

kog sve više obrazovnih institucija u svetu koristi ovu tehnologiju za automatizuju poslova.

3.2. Kreiranje video uputstava korišćenjem RPA

Postoji mnoštvo radova u kojima su prikazane prednosti upotrebe RPA u obrazovanju. Međutim, ni u jednom radu nije analizirano kako bi ova tehnologija mogla da pomogne nastavnicima u samom procesu edukacije.

Sve češće se javlja potreba za kreiranjem uputstava u obliku video materijala sa gradivom koje učenici treba da obrade. Kao što je već izloženo u poglavlju 3, iako je ovaj oblik edukacije veoma pogodan za učenike, predstavlja vremenski zahtevan posao za nastavnike. Upravo se robotska automatizacija procesa može primeniti kao alternativni način za pravljenje uputstava koji, pritom, mogu biti i interaktivni. U ovom slučaju, uputstva se kreiraju kao niz individualnih koraka koji zajedno omogućavaju obavljanje kompleksnijih zadataka. Iako je kreiranje prvog takvog uputstva vremenski zahtevno, kasnije se svodi na ponovnu primenu istih ili sličnih koncepata. Zbog segmentiranosti ove vrste materijala, ažuriranje se svodi na izmenu jednog ili više koraka, što je jednostavnije i vremenski efikasnije. Slično kao i kod video materijala, i u ovom slučaju je moguće da učenik prilagodi tempo napretka svojim mogućnostima.

Stoga je cilj ovog rada analiza upotrebe robotske automatizacije procesa za kreiranje obrazovnog materijala u obliku video uputstava sa interaktivnim elementima.

4. RPA ALATI

RPA tehnologija je razvijana sa ciljem da se omogući automatizacija procesa u različitim oblastima. Da bi to bilo moguće, razvijeni su različiti RPA alati koji nude različite funkcionalnosti korisnicima. Na taj način je organizacijama omogućeno da odaberu alat koji je najpogodniji za automatizaciju njihovih procesa. Neki od najpopularnijih RPA alata su: *Blue Prism*, *Automation Anywhere*, *Pega*, *Jacada*, *Nice System* i *UiPath* [13].

UiPath je popularan alat za automatizaciju robotskih procesa koji pomaže u automatizaciji aplikacija, kao što su desktop i veb aplikacije. Pruža platformu otvorenog izvornog koda koja promovise saradnju i automatizaciju ponavljajućih zadataka [14]. Pored toga, nudi više opcija *hosting-a*, kao što su *Cloud* okruženja i virtuelne mašine i omogućava upotrebu *scrapping* rešenja koja se mogu primeniti na .NET i Java aplikacije, ali i na PDF dokumenta. Glavne prednosti ovog alata su to što povećava produktivnost organizacije, podržava internu reviziju koja pomaže organizacijama da se pridržavaju industrijskih propisa, poseduje sigurnosne funkcije koje štite procese od različitih oblika napada i poboljšava korisničko iskustvo.

Pored toga, ovaj alat omogućava kreiranje dijagrama, pokretanje automatizacije i izmenu postojećih dijagrama na jednostavan način. Upotrebom *UiPath Studio* korisnik može da kreira prazan projekat ili da iskoristi neki od ugrađenih šablona. Šabloni su kreirani sa ciljem da korisniku olakšaju automatizaciju procesa. Nakon kreiranja projekta, korisnik definiše niz aktivnosti koje se izvršavaju sekvencijalno ili paralelno. *UiPath Studio* omogućava upotrebu jednostavnih aktivnosti kao što su klik na dugme i unos teksta u polje, ali i komplikovanih kao što

su pokretanje aplikacija i upotreba *screen scaping* tehnologije za ekstrakovanje podataka iz tabela ili pokrenutih aplikacija. Ovaj alat omogućava i korišćenje snimača, koji snima ekran i korisnikove akcije pretvara u niz aktivnosti. Na taj način se početnicima znatno olakšava kreiranje automatizovanih procesa. U ovom alatu postoje i aktivnosti koje su namenjene iskusnim korisnicima i programerima, kao što su kreiranje CSV datoteka, pokretanje JavaScript datoteka, upotreba naredbi grananja, petlji, metoda za manipulaciju stringovima i pronalaženje grešaka.

Upravo zbog svih izloženih mogućnosti i prednosti, ovaj alat je upotrebljen za izradu studije slučaja izložene u narednom poglavlju.

5. STUDIJA SLUČAJA

Cilj ove studije slučaja je analiza upotrebe RPA alata za kreiranje obrazovnog materijala u formi video uputstava. Automatizacijom kreiranja video uputstava nastavnom osoblju bi bio olakšan posao kreiranja, ali i izmene ovih materijala. Kreiranje prvog ovakvog uputstva je vremenski zahtevan posao, jer zahteva od nastavnog osoblja da se upoznaju sa samim RPA alatom i da zatim kreiraju uputstvo. Sa druge strane, postupci izmene i kreiranja novih uputstava su veoma jednostavni. Kreiranje, u tom slučaju, podrazumeva upotrebu i izmenu postojećih uputstava. Takođe, zahvaljujući RPA alatima nastavno osoblje može da od interaktivnih kreira video uputstva prostim snimanjem izvršavanja interaktivnog uputstva. Interaktivna uputstva podrazumevaju pokretanje samog uputstva na učenikovom računaru, pri čemu bi on u realnom vremenu gledao kako softverski robot klikće i obavlja aktivnosti na njegovom računaru. Ovaj način edukacije je često interesantniji od video uputstava, pa samim tim duže drži pažnju učenika.

Za potrebe ovog rada kreirano je uputstvo za učenje osnova programiranja u Java programskom jeziku upotrebom *UiPath* alata. Ovo uputstvo se sastoji iz nekoliko primera:

1. Prvi Java program
2. Promenljive i tipovi podataka
3. Ugrađene metode za rad sa stringovima
4. Operatori
5. Nizovi
6. Naredbe grananja
7. Petlje
8. Završni primer

Primeri su kreirani upotrebom *Eclipse* razvojnog okruženja. U svakom primeru je kreiran po jedan Java projekat u kom je prikazano kako se koriste elementi koji su prethodno teorijski objašnjeni u tom primeru. Iako jednostavni, kreirani primeri prikazuju kako treba pristupiti rešavanju programerskih zadataka i koji su koraci neophodni da se to postigne. Svaki primer je kreiran kao zaseban *UiPath* projekat i pokreće se nezavisno od ostalih. Ovo uputstvo je dostupno na *GitHub* repozitorijumu sa nazivom „RPA-Uputstva“ [15].

Za potrebe kreiranja ovog uputstva prvo je kreiran šablon koji sadrži dijagrame za pokretanje *Eclipse* alata, kreiranje projekta i klase, i izvršavanje kreiranog projekta. Pošto su ovi elementi neophodni u svim primerima,

namena šablona je da omogući njihovu ponovnu upotrebu bez višestrukog kreiranja istih dijagrama.

Kreirani primeri demonstriraju upotrebu *UiPath* alata za kreiranje relativno složenog interaktivnog uputstva koji može biti korišćen u edukativne svrhe. Iako su mogućnosti samog alata daleko veće od prikazanih, kroz primere je demonstrirano da se svrsishodna uputstva mogu kreirati upotrebom svega nekoliko tipova aktivnosti. Takođe, moderan i intuitivan grafički korisnički interfejs ovog alata olakšava pomenuti posao i omogućava upotrebu od strane korisnika koji nisu programerski edukovani, već predstavljaju tipične korisnike računara kakvih je, zapravo, i najviše. Samim tim, ne postoji prepreka da ovaj alat bude naširoko korišćen u oblasti obrazovanja.

5.1. Od interaktivnog do video uputstva

Uputstva kreirana upotrebom *UiPath* alata su interaktivna, što znači da učenik pokreće uputstvo na svom računaru i u realnom vremenu posmatra kako softverski robot na njemu obavlja zadatke. Ovakva uputstva podrazumevaju učešće i interakciju učenika, što povećava njihov stepen pažnje i zainteresovanosti za samo gradivo. Pored toga, ova uputstva omogućavaju učeniku da prilagođava brzinu izvršavanja i da ga po potrebi zaustavlja. Ipak, ovakva uputstva nisu uvek jednostavna za pokretanje, jer je neophodno instalirati *UiPath* alat, kao i sve alate korišćene u samom uputstvu. Može se desiti i da različite verzije instaliranih alata ili različita konfiguracija računara izazovu probleme prilikom pokretanja uputstva. Stoga je pogodno da osoba koja je kreirala uputstvo, uz interaktivne, ostavi svojim učenicima i video materijale. Na taj način će učenici imati mogućnost da pokrenu video uputstvo ukoliko imaju problema sa pokretanjem interaktivnog ili da, prosto, odaberu tip uputstva koji im je najpogodniji za usvajanje novog gradiva.

Kreiranje video materijala na osnovu *UiPath* projekta zahteva posedovanje programa za snimanje ekrana. Danas postoji mnoštvo ovakvih programa koji su besplatni, pa korisnik samo treba da pronađe i preuzme alat koji želi da koristi. Zatim je potrebno da pokrene snimanje, a nakon toga i kreirani *UiPath* projekat. Rezultat snimanja ekrana biće video materijal nastao od kreiranog interaktivnog uputstva. Zahvaljujući *UiPath* alatu osoba koja je kreirala uputstvo će mnogo lakše menjati interaktivno, ali i video uputstvo. Izmena podrazumeva izmenu pojedinih segmenata kreiranog uputstva i ponovno snimanje ekrana. Ovaj postupak je mnogo jednostavniji od ponovnog snimanja pojedinih delova i montiranja video snimka.

6. ZAKLJUČAK

Kreiranjem prikazanog uputstva pokazano je da je robotska automatizacija procesa pogodna za kreiranje ovakve vrste obrazovnog materijala. Iako je kreiranje prvog primera vremenski zahtevan posao, jer zahteva od korisnika i upoznavanje sa samim RPA alatom, kreiranje narednih primera i izmena postojećih je znatno olakšana. Mnogobrojni alati za ovu namenu su prilagođeni tipičnim krajnim korisnicima računara, pa pored mogućnosti programiranja, nude i mogućnost kreiranja dijagrama prevlačenjem raspoloživih grafičkih objekata.

Pokretanjem snimanja ekrana, na jednostavan način se od interaktivnog uputstva može kreirati video uputstvo. Samim tim, osoba koja je kreirala uputstvo može svojim učenicima da ostavi i video materijale i interaktivna uputstva i da im mogućnost da odaberu tip materijala koji im je najpogodniji za usvajanje novog gradiva. Ono što korisnike može odbiti jeste potreba za ulaganjem dodatnog vremena i napora za istraživanje ove tehnologije. Potrebno je istražiti dostupne alate i među njima odabrati najpogodniji za konkretnu oblast primene. Zatim je neophodno ovladati korišćenjem odabranog alata, pre nego što se upotrebi u željene svrhe.

7. LITERATURA

- [1] <https://www.uipath.com/rpa/robotic-process-automation> (pristupljeno u septembru 2021.)
- [2] <https://digitalworkforce.com/rpa-robotic-process-automation> (pristupljeno u septembru 2021.)
- [3] <https://www.clariontech.com/platform-blog/8-unparalleled-benefits-of-rpa-that-will-power-up-your-business> (pristupljeno u septembru 2021.)
- [4] <https://www.kaspersky.com/resource-center/definitions/what-are-bots> (pristupljeno u septembru 2021.)
- [5] <https://www.javatpoint.com/advantages-and-disadvantages-of-rpa> (pristupljeno u septembru 2021.)
- [6] <https://searchcio.techtarget.com/definition/RPA> (pristupljeno u septembru 2021.)
- [7] <https://www.newgenapps.com/sr/blogs/what-is-robotic-process-automation-uses-application-benefits> (pristupljeno u septembru 2021.)
- [8] <https://www.cpajournal.com/2018/07/02/how-robotic-process-automation-is-transforming-accounting-and-auditing> (pristupljeno u septembru 2021.)
- [9] <https://searchcio.techtarget.com/definition/RPA>, (pristupljeno u septembru 2021.)
- [10] <https://www.dominknow.com/blog/video-based-learning-an-evolution> (pristupljeno u septembru 2021.)
- [11] <https://www.neevista.com/articles/9-rpa-use-cases-that-can-take-your-educational-institution-to-the-next-level> (pristupljeno u septembru 2021.)
- [12] <https://www.robomotion.io/blog/use-of-rpa-in-education> (pristupljeno u septembru 2021.)
- [13] <https://www.itbusinessedge.com/applications/robotic-process-automation-rpa-tools> (pristupljeno u septembru 2021.)
- [14] <https://www.section.io/engineering-education/introduction-to-uipath-in-rpa/> (pristupljeno u septembru 2021.)
- [15] <https://github.com/JelenaPopov/RPA-Uputstva> (pristupljeno u septembru 2021.)

Kratka biografija:

Jelena Popov rođena je u Zrenjaninu 1997. god. Osnovne akademske studije je upisala na Fakultetu tehničkih nauka Univerziteta u Novom Sadu 2016. godine. Diplomirala je 2021. godine.

PLATFORMA ZA SPECIFIKACIJU I INTERPRETACIJU INTERAKTIVNE FIKCIJE**PLATFORM FOR INTERACTIVE FICTION SPECIFICATION AND IMPLEMENTATION**

Aleksa Ivković, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Ovaj rad predstavlja implementaciju jezika specifičnog za domen za potrebe kreiranja igri iz žanra interaktivne fikcije. Ovaj projekat je realizovan pomoću biblioteke textX. U okviru rada istražene su karakteristike ovog žanra, kao i neka slična rešenja iz domena.

Ključne reči: Jezici specifični za domen, interpreter, interaktivna fikcija

Abstract – This paper presents implementation of a domain specific language for creating interactive fiction games. This project was done using textX library. Also this paper explores characteristics of this genre, as well as some similar solutions in the domain.

Keywords: Domain specific languages, interpreter, interactive fiction

1. UVOD

Personalni računari nisu u masovnoj upotrebi ni 50 godina, ali napredak koji su oni dostigli za taj period je neverovatan. Takav napredak nije zabeležen do sada u istoriji čovečanstva. 70-ih godina prošlog veka pojavili su se personalni računari, a na njihovu revolucionarnu pojavu je dodatno uticao momenat njihove svrhe i upotrebe - ljudi su dobili priliku da ih koriste u udobnosti svog doma. U tom periodu računari su imali svega nekoliko kilobajta memorije i softveri su morali da budu optimizovani kako bi mogli da izvuku njihov maksimum. Samim tim igre sa grafičkim interfejsom nisu ni bile moguće. Shodno tome, na tržištu su se pojavile igre koje su pripadale žanru interaktivne fikcije. One su bile dosta kompleksne i pružale su veliki broj sati zabave igračima [1]. U okviru ovog rada će biti istražene same igre i njihov istorijat, ali će veći fokus biti na samom načinu njihove implementacije.

2. INTERAKTIVNA FIKCIJA

Interaktivna fikcija (u daljem tekstu IF) u svom najširem smislu može predstavljati bilo koji vid pripovedanja u kome čitalac ili slušalac aktivno učestvuje. U okviru ovog rada će se posmatrati jedan određeni tip IF, koji je ujedno i najpopularniji, a to jeste u vidu igri za računare. Neki vid definicije ovih igara je dat od same zajednice koja je pomogla da ovaj vid zabave ne izumre: „IF predstavlja

jedinstveni vid kompjuterskog pripovedanja koji stavlja igrača u ulogu lika u simuliranom svetu, koji karakteriše njegovo oslanjanje na tekst kao glavni vid opisa, a koristi fleksibilni jezik, sličan ljudskom, za izdavanje komandi, koje obrađuje parser“ [2].

2.1. Počeci

Začetak ovog žanra predstavlja igra pod nazivom Hunt the Wumpus. Ova igra je nastala 1972. godine od strane Gregory Yob – a. Ono što nju razlikuje od njenih naslednika je to što nije sadržala parser koji je mogao da tumači kompleksne rečenice, već je igrač imao ograničen skup akcija koje je mogao da izvrši. Ove akcije su se sadržale od dve reči, glagola i objekta.

Sredinom 1970 – ih, nastala je igra Adventure, prva ovog žanra koja se odlikuje svim karakteristikama modernog IF – a. Njen kreator je Will Crowther sa Stanford – a, inspirisan, tada novom društvenom igrom, Dungeons and Dragons. Adventure je igra napisana u Fortran – u i bila je kreirana za PDP-10. Veliki korak je napravljen ovom igrom, jer je Will kreirao parser koji je mogao da obrađuje naredbe od dve reci, sa mnogo većim vokabularom nego u igri Hunt the Wumpus.

2.2. Komercijalna era

1978. godine, Scott Adams, pod uticajem igre Adventure, odlučuje da je modifikuje, kako bi mogla da se izvršava na kućnim računarima, koji su imali znatno manje memorije. Tako je nastala igra poznata pod nazivom Adventureland. Kako bi postigao ovakav cilj Scott je kreirao svoj jezik i interpreter, kako bi sama igra bila nezavisna od platforme na kojoj se izvršava. Sa ovakvim rešenjem, Scott je postigao veliki uspeh.

Paralelno sa razvitkom Adventureland – a, nastaje i igra pod nazivom Zork. Naime, u trenutku kada je Adventure pušten u ARPANET, grupa studenata sa MIT – a je pronašla igru i odlučili su da naprave bolju igru od nje. Od 1975. do 1979. su razvijali ovu igru. Igra Zork ne predstavlja komercijalni uspeh, već tehnološki, njen parser je bio u mogućnosti da prepozna složenije fraze, i imao je vokabular od oko 900 reči. Svi moderni parseri koriste sličnu paradigmu kao i ovaj parser.

1979. godine su asistent sa MIT – a i neki od početnika Zork igre odlučili da osnuju svoju kompaniju Infocom, primarni fokus kompanije je bio prilagoditi igru Zork kućnim računarima. Zaposleni u kompaniji su shvatili da mogu da iskoriste novu moć personalnih kompjutera, tako što će samo izvršavati igru na računarima. Ovo su postigli pomoću kompjuterskog čipa Z-machine koji su kreirali.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Igor Dejanović, vanr. prof.

Na ovom čipu bi igra mogla da se izvršava, ali da bi zapravo mogla da se izvršava na računaru, svaki uređaj je morao da ima ZIP (Z-machine Interpreter Program). Ovaj interpreter bi omogućio da se igra izvršava isto kao i na Z-machine. Ali kako bi ovo sve bilo moguće, bilo je neophodno kreirati jezik koji je dovoljno kompaktan kako bi kompleksna igra mogla da stane, a i da se izvršava na uređajima tog vremena. Tako je nastao jezik Zork Implemented Language (ZIL). Igra Zork je donela veliki uspeh kompaniji, tako da su nastala dva naslednika Zork II i Zork III, kao i oko 30 drugih naslova.

2.3. Kraj ere i trenutno stanje

Nakon uspešne decenije, početkom '90 – ih godina prošlog veka, IF žanru počinje da opada popularnost. Naime, razvitkom računara, i većom snagom i kapacitetom koji su oni doneli, same računarske igre su mogle biti zahtevnije. U tom periodu se pojavljuje novi tip avanturističkih igara, koji se u potpunosti oslanja na vizualni prikaz i kliktanje mišem. Pojavile su se dve nove kompanije koje su dominirale tržištem avanturističkih igara baziranih na igri pomoću miša, a to su bile Sierra i Lucasarts. Ljudi su bili očarani novim grafičkim dostignućima i samim tim IF igre im nisu bile više primamljive.

Tokom 1990 – ih i 2000 – ih bilo je više pokušaja da se oživi IF žanr u komercijalnom smislu, ali je to bilo bez nekih većih uspeha. Do danas žanr održava njegova mala grupa vernih obožavaoca, koji razvijaju svoje naslove i većinski ih besplatno distribuiraju na internetu. Postoji nekolicina foruma i blogova koji omogućavaju istomišljenicima da razmenjuju ideje i pomažu jedni drugima da sačuvaju ovaj žanr od zaborava.

3. SLIČNA REŠENJA

3.1 Alan

Alan predstavlja jezik specifičan za domen baziran na karakteristikama objektno orijentisanog programiranja. Prva verzija ovog jezika je nastala 1985. godine kada je ovaj žanr bio u svom vrhuncu. Tokom godina je unapređen i trenutno je aktuelna treća iteracija ovog softvera. Ovaj jezik je namenjen autorima IF igara, tako da je njegova sintaksa približena više ne tehničkim licima. Glavna ideja tokom razvoja ovog jezika jeste jednostavnost, ali time nije uskraćena implementacija kompleksnijih ideja. Sintaksa ovog jezika se može posmatrati više kao skup opisa koji nalikuju rečenicama [3].

3.2 Adventuron

Adventuron predstavlja softversko rešenje koje svojim korisnicima pruža online editor za tekst, i kompajler koji može da izvršava igre u pretraživaču, na računari, mobilnim telefonima... Za razliku od prethodnog jezika, Adventuron je okrenutiji programerima svojom sintaksom. Ona najviše podseća na hibrid JSON formata i JavaScript jezika [4].

4. JEZICI SPECIFIČNI ZA DOMEN

Jezik je sistem znakova, simbola, gestova i pravila koji se koriste u komunikaciji. U kontekstu računara, jezik predstavlja način komunikacije između korisnika i računara, ili između dva računara. Programski jezici se

sastoje od skupova stringova koji se kompajliraju u komande koje računar može da izvrši. Jezike možemo podeliti na jezike opšte namene (engl. *general purpose languages*) i jezike specifične za domen (engl. *domain specific languages*). Neki od poznatih jezika opšte namene su: C, C#, Java, Python, JavaScript. Jezici opšte namene su definisani na način koji im omogućava da se prilagode velikom broju domena, samim tim i način na koji se može doći do rešenja je velik. Ova karakteristika nosi i lošu posledicu sa sobom, a to jeste, povećana šansa za greškama[5].

Da bismo mogli da razumemo namenu jezika specifičnih za domen, neophodno je da razumemo šta domen predstavlja. Domen predstavlja sfere delovanja, interesa ili funkcije. Jezici specifični za domen su prilagođeni jednom domenu i oni mogu da se koriste vokabularom i notacijama tog domena. Iz razloga što su ovi jezici usko vezani za njihove domene, drugi naziv im je i mali jezici.

Odabirom nekog jezika specifičnog za domen za realizaciju softvera, čini se prvi korak ka efektivnijem rešenju. Naime, studije su pokazale da se korišćenjem ovakvih jezika produktivnost može uvećati i do 10 puta. Pored toga, ovi jezici su koncizniji, a takođe mogu da se maksimalno prilagode potrebama korisnika. Kada se pristupa rešavanju problema, neophodno je kreirati konceptualni model za koncepte iz domena problema. Najveći problem ovog procesa jeste ručno mapiranje konceptata, na koncepte domena rešenja, pri korišćenju jezika opšte namene [5].

5. TEXTX

TextX predstavlja alat koji je izgrađen na Arpeggio – vom parseru, ali je njegova gramatika jednostavnija, konciznija i ekspresivnija [6][7]. Rad textX – a je baziran na sledećem toku, prvo se parsira sama gramatika jezika. TextX dinamički kreira meta – model i Arpeggio parser. Parser će dalje parsirati sam model i kreirati graf Python objekata u odnosu na meta – model. Model se dalje može koristiti u svrhe interpretacije ili za generisanje fajlova.

6. OPIS REŠENJA

6.1. Jezik

Prvo će biti opisan sam jezik, njegove osobine i sama struktura. Pre toga, potrebno je napomenuti da je sama sintaksa bila inspirisana Python jezikom i sintaksom online alata Adventuron, koji je bio opisan u trećem poglavlju. Zamisao ovog rada jeste bila stvoriti jezik koji pruža korisniku veliku fleksibilnost pri pisanju igre, i time mu omogućiti da pomoću nekih elemenata koje sam definiše, bude u mogućnosti da postigne željene rezultate. Glavne komponente koje omogućavaju funkcionalnost ovog softverskog rešenja jesu: lokacije, veze, objekti, stanja, akcije i glagoli. U daljem tekstu će biti detaljno opisano svako pravilo.

Kao prvo pravilo, a samim tim i prvi objekat koji se generiše jeste klasa Game. Ovo pravilo, predstavlja skup svih drugih pravila, i u njemu se nalaze svi neophodni elementi za igru. Naslov i uvod su obična string polja koja se ispisuju pri pokretanju igre. Sledeće polje klase Game jeste početak koji označava sa koje lokacije će se igra započeti. Korisnik mora izabrati neku od lokacija koje je pre toga definisao, a identifikuje je njenim nazivom.

Obeležje `end` je definisano pravilom `GameEnd`. Ono sadrži uslove (engl. *conditions*) i poruku u slučaju da je ispunjen uslov za kraj igre, koje je opisano pravilom `Message`. Za opis kraja igre su odabrana ova polja, jer kako će biti opisano u daljem tekstu, uslovima se mogu proveriti i stanja i lokacije, što je sasvim dovoljno da se postavi cilj koji treba postići u igri.

Pravilo `Conditions` sadrži listu uslova, koji su definisani pravilom `Condition`. Samo pravilo može biti ili `StateCondition` ili `LocationCondition`. Prva vrsta uslova se bazira na stanjima, koja će biti objašnjena kasnije detaljno, i proverava njegovu vrednost prostim logičkim poređenjem jednakosti. Dok pravilo provere uslova lokacije, proverava da li se objekat nalazi na određenoj lokaciji ili u određenom objektu. Za oba ova pravila je iskorišćeno referenciranje. Kako možemo primetiti u oba uslova postoji opcioni parametar, poruka. Ona je namenjena ako želimo da ispišemo određeni tekst u slučaju da taj uslov nije zadovoljen.

Pravilo `Message` definiše samu poruku, i njeno obeležje jeste `text`, čija je vrednost tipa `string`. Za potrebe ovog pravila, odlučeno je da bude svedeno na minimum kako bi korisniku olakšali definisanje same igre.

Pravilo `CID`, prilagođava `ID` tip ovom jeziku, tako što sprečava korisnika da za ime izabere neku od rezervisanih reči koja se koristi u ostatku sistema. Ono je definisano pomoću negativnog predviđanja (engl. *negative lookahead*). Ovim putem se obezbeđuje da nijedna od rezervisanih reči ne može biti iskorišćena kao `ID`. Pravilo `Keywords` (ključne reči) definiše skup reči pomoću regex izraza. U rezervisane reči spadaju predefinisane akcije, lokacije, objekti, i reči korištene u ostatku jezike.

Opis je definisan pomoću pravila `Description`, koje u sebi sadrži samo obeležje `text`, koje je običan `string`.

Veze su opisane pravilom `Connection`. Ovo pravilo referencira početnu (engl. *from location*) i odredišnu (engl. *to location*) lokaciju, kao i smer, koji je određen pravilom `Direction`.

Smer (engl. *direction*) određuje smer kretanja, u odnosu na stranu sveta. Pravilo `Direction` je definisano kao uređena lista `stringova`.

Pravilo za objekat definiše njegov naziv (koji bi trebalo da bude jedinstven radi identifikovanja), opis i referencira njegovu lokaciju, koja može biti neka lokacija (definisana `Location` pravilom) ili može biti neki drugi objekat. Takođe sadrži skup uslova koji definišu šta sve mora biti ispunjeno kako bi taj objekat bio vidljiv, tj. da li će biti ispisivan pri ispisu lokacije i da li može da se istraži objekat. Više o samim akcijama i njihovoj konkretnoj implementaciji će biti kasnije u radu.

Sledeće pravilo omogućava da se odrede stanja. Stanje definišu ime, koje predstavlja identifikator, samo stanje, koje koristimo kao `flag`, i njegova vrednost može biti tačno ili netačno. Pored toga je neophodno definisati i prioritet, koji je tipa celog broja. Prioritet služi pri ispisu objekta, ako želimo da ispišemo neku poruku kada je stanje tačno, potrebno je da definišemo `when_true` poruku, ili u suprotnom `when_false`. Ove poruke nisu obavezne, ali je bitno napomenuti da će biti ispisana samo prva poruka za koju je zadovoljen uslov (iz tog razloga je bio potreban prioritet).

Postoje dve vrste akcija, jedne koje vrše neke promene - opisane pravilom `StateAction` ili one koje vrše opisivanje nekih objekata - opisane `DescribeAction` pravilom. Prvo će biti opisano `StateAction` pravilo. Njega definiše naziv, objekat/lokacija na koju se odnosi, skup uslova pod kojima može da se izvrši radnja, poruka u slučaju da je uspešno izvršena akcija i skup promena. Promene su definisane pravilom `Change` koje će biti opisano u narednom poglavlju. Druga vrsta akcija je opisana pravilom `DescribeAction`. One služe kao dodatak već ugrađenoj funkcionalnosti istraživanja (engl. *exam*). Prvo je potrebno definisati na koji objekat se odnosi, a zatim je potrebno definisati uslove, koji moraju biti ispunjeni kako bi se izvršila željena akcija. Pored toga je moguće definisati poruku i/ili `print_condition`, kojim se definiše da li se ispisuju objekti koji su sadržani u tom objektu koji se istražuje.

Izmene su definisane pravilom `Change`, koje mogu biti ili promene stanja (`StateValueChange`) ili mogu biti promene lokacije (`LocationChange`). Pri promeni stanja neophodno je referencirati neko stanje i dodeliti mu novu logičku vrednost. Pravilo za lokaciju je slično, neophodno je referencirati objekat i dodeliti mu novu vrednost za lokaciju, takođe referencirajući neku postojeću lokaciju ili objekat.

Poslednje pravilo definiše skup glagola, koji se koriste kao reference u pravilu `StateAction`. Njih definiše jedinstveno ime i opis.

6.2. Interpreter i parser

Kako za ovakav tip igara nije neophodno kreirati generator, kreiran je interpreter. On u vremenu izvršavanja kreira modele, dodatne metode i obrađuje zahteve koje igrač unosi. Kako bi funkcionalnosti samog igranja bile omogućene, klasa `Game` je proširena metodama koje to omogućavaju. Pored klase `Game`, definisana je i klasa `GamePlay` koja predstavlja naš parser i ona obrađuje unos od strane. Pored toga se iterira kroz listu glagola i kreiraju se metode kao što je malopre naznačeno. Ugrađene akcije su: *go, look, exam, inventory, new i exit*.

6.3. Generator

Druge komande koje je moguće izvršiti su *save* i *load*. Ove akcije predstavljaju proces čuvanja stanja igre na nekom medijumu, a *load*, akciju koja omogućava da igrač nastavi tamo gde je stao u trenutku kada je sačuvao. Za generisanje sačuvanog dokumenta koriste se Jinja šabloni.

Jinja predstavlja obrađivač šablona (engl. *template engine*) koji koristi semantiku sličnu Python – u i zasniva se na „bušenju rupa“ gde je potrebno kasnije umetnuti podatke. Za svaki šablon se proslede podaci koji se posle samo smeštaju na adekvatne pozicije. Iako većina logike treba da bude izmeštena van šablona, veliki deo posla i dalje može da se uradi unutar njih [8].

7. STUDIJA SLUČAJA

U daljem tekstu će biti prikazana jedna igra i opisani koraci koje je igrač učinio kako bi došao do cilja.

Nakon pokretanja igre, igrača prvo dočeka ekran na kome je ispisan naziv igre, i opcija da otpočne novu igru, ili nastavi prethodno sačuvanu. Za potrebe ove demonstracije biće otpočeta nova igra.

Na početku igre se ispiše uvodni tekst, a zatim sledi i opis trenutne lokacije. U prostoriji vidi kamen, seno i vrata od ćelije u kojoj se nalazi. Pored toga, iz prostorije postoji samo jedan izlaz. Prvo što igrač čini jeste da pokuša da izađe iz prostorije. Ovaj pokušaj je neuspešan i ispisuje mu se poruka da su vrata zaključana. Kako bi otkrio više o vratima on ih istražuje, a opis koji dobija mu kaže da mu je neophodan srebrni ključ. Zatim on pokušava da pomeri kamen, i ova akcija prolazi uspešno. Nakon toga, igrač odlučuje da ponovo osmotri sobu, nakon čega primećuje novi objekat, a to jeste srebrni ključ. Pre nego što otključa vrata, odlučuje da istraži i poslednji predmet u sobi a to jeste seno, koje daje beznačajan opis. Nakon uspešnog otključavanja vrata, igrač prelazi na novu lokaciju.

Pri ulasku u sledeću prostoriju, igrača dočekuje novi opis. Ovog puta, jedini objekat koji vidi jeste lampa i ima samo dva izlaza iz sobe, od čega je jedan kroz koji je ušao. Nakon istraživanja lampe, ne otkriva ništa specifično. Potom je uzima, i nastavlja u sledeću sobu.

Ovog puta, igrača dočekuje prostorija koja sadrži vrata, sanduk i ima dva izlaza iz nje. Igrač kao i u prvoj prostoriji pokušava da izađe, ali ne uspeva. Nakon istraživanja vrata, otkriva da mu je neophodan magični ključ. Zatim pokušava da otvori sanduk, ali je on zaključan. Nakon pokušaja otključavanja, otkriva da mu je potreban mesingani ključ. Nakon pretraživanja inventara, otkriva da jedino nije lampu iskoristio, nakon čega pokušava da je slomi. Ova akcija uspešno prolazi. Nakon istraživanja lampe, u njoj pronalazi mesingani ključ, i uzima ga. Zatim otključava i otvara sanduk, u kom pronalazi zlatni ključ. Da bi otključao vrata koja vode ka izlazu, mora da pronađe magični ključ. Kako nema više šta da istraži, pokušava da ga napravi. Ova akcija uspešno prolazi, uništava tri ključa koja je do sada skupio, i u inventar mu smešta magični ključ. Sa ovim ključem u inventaru, otključava vrata i napušta sobu. Pri napuštanju sobe, dobija poruku da je uspešno prešao igru.

8. ZAKLJUČAK

Kao i u svakom softverskom rešenju, tako i u ovom, uvek postoji prostor za napredak. Neke od ključnih stvari od kojih bi korisnici imali veliki benefit su:

1) Mogućnost ulančavanja proveru stanja ili lokacija u veći logički izraz

Rezultat: omogućeno korišćenje I/ILI logičkih operatora

2) Mogućnost definisanja sinonima za glagole

Rezultat: veća fleksibilnost pri kreiranju igre

3) Mogućnost definisanja osnovnog dela poruke, na koje bi se samo lepilo ime objekta, i ovakve poruke bi se vezivale za glagole

Rezultat: manje linija koda, bio bi čitljiviji, a i kreatorima igara bi uštedelo vreme

4) Mogućnost poboljšanja definisanja poruka kod uslova

Treba napomenuti da i sa uvođenjem svih ovih izmena sama funkcionalnost ne bi bila drastično izmenjena.

9. LITERATURA

- [1] Roger Firth i Sonja Kesserich, „The Inform Beginner’s Guide“, The Interactive Fiction Library, August 2002
- [2] Kevin Jackson-Mead i J. Robinson Wheeler, „IF Theory Reader“, April 2011
- [3] <https://alanif.se/downloads/documentation/manual.pdf> (pristupljeno 10.10.2021.)
- [4] <https://adventuron.io/> (pristupljeno 10.10.2021.)
- [5] I. Dejanović, „Prilog metodama brzog razvoja softvera na bazi proširivih jezičkih specifikacija. PhD thesis, Faculty of Technical Sciences, University of Novi Sad, January 2012”
- [6] I. Dejanović, R. Vaderna, G. Milosavljević, Z. Vuković, „textX: A Python tool for Domain-Specific Languages Implementation“, Faculty of Technical Sciences, University of Novi Sad
- [7] Dejanović I., Milosavljević G., Vaderna R.: Arpeggio: A flexible PEG parser for Python, Knowledge-Based Systems, 2016, 95, 71 - 74, doi:10.1016/j.knsys.2015.12.004
- [8] <https://jinja.palletsprojects.com/en/3.0.x/intro/> (pristupljeno 19.10.2021.)

Kratka biografija:



Aleksa Ivković rođen je u Novom Sadu 1997. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – računarstvo i automatika, odbranio je 2021.god.
kontakt: aleksa.ivk@gmail.com

**SIMULACIJA ROBOTA BAZIRANOG NA DIFERENCIJALNOM POGONU
KORIŠĆENJEM GAZEBO SIMULATORA I ROS-A****SIMULATION OF THE ROBOT BASED ON DIFFERENTIAL DRIVE BY USING
GAZEBO SIMULATOR AND ROS**

Dorđe Vujić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U ovom radu razvijen je simulator robota sa diferencijalnim pogonom, uz oslonac na Gazebo platformu za simulaciju robotskih aplikacija i robotski operativnisistem (ROS). Razvijenim simulatorom se posebno razmatra sprega ova dva alata. Analiza rezultata vršena je korišćenjem raznih alata kompatibilnih sa robotskim operativnim sistemom.

Ključne reči: Simulacija, PID, ROS, Gazebo, diferencijalni pogon, kretanje, robot

Abstract – In this paper, simulator of the differential drive robot is developed, which relies on Gazebo platform for simulation of robot applications and robot operating system (ROS). Simulator which is developed provides a common analysis between Gazebo and ROS. Analysis of the results is performed by using different tools which are compatible with ROS.

Keywords: Simulation, PID, ROS, Gazebo, differential drive, movement, robot

1. UVOD

ROS (*engl. Robot Operating System*) je set softverskih biblioteka i alata koji pomažu u pravljenju robotskih aplikacija. ROS olakšava razvoj kompleksnih i robusnih robotskih platformi različitih tipova.

Zbog toga što je pravljenje robusnog softvera za generalnu upotrebu veoma teško, rodila se ideja za kolaboracijom. ROS je rezultat kolaboracije različitih pojedinaca, laboratorija i institucija, gde svako razvija ono za šta je specijalizovan.

Na primer, jedna laboratorija ima eksperte iz oblasti mapiranja unutrašnjeg okruženja i može da učestvuje u sistemima za pravljenje mapa. Druga grupa možda ima eksperte u korišćenju mapa radi vršenja navigacije, a treća grupa će razvijati korišćenje kompjuterske vizije za detekciju objekata u toku kretanja.

ROS je dizajniran specijalno za ovakve grupe kako bi svaka mogla da vrši razvoj i dostiže svoje ciljeve uz pomoć produkata rada neke druge grupe.

2. KONCEPTI**2.1. Čvorovi**

Čvorovi (*eng. nodes*) su procesi koji vrše izračunavanja. ROS je dizajniran da bude modularan, tako jedan robotski sistem uglavnom sadrži mnogo čvorova. Na primer, jedan čvor kontroliše laserski senzor, drugi čvor kontroliše točkove na motorima, a treći čvor vrši lokalizaciju.

2.1. Poruke

Čvorovi komuniciraju jedni sa drugima prosleđujući poruke. Poruka je jednostavna struktura podataka, sa tipiziranim poljima.

2.2. Topici

Čvorovi šalju poruke objavljujući ih topicima (*eng. topic*). Topik je ime koje se koristi za identifikaciju sadržaja poruke. Čvor koji je zainteresovan za odrađenu vrstu podataka, pretplatiće se na odgovarajući topik.

2.3. Servisi

Servisi su par struktura poruka: jedna za zahtev, a druga za odgovor. Čvorovi koji obezbeđuju servise, nude ih pod određenim imenom, a klijenti ih koriste šaljući zahtev u vidu poruke i čekaju odgovor.

3. KOMUNIKACIONA INFRASTRUKTURA**3.1. Objavi/pretplati se**

Poruke su rutirane kroz transportni sistem putem objavi/pretplati se metodologije. ROS sadrži ugrađen i dobro testiran sistem razmene poruka između čvorova, korišćenjem topika. Ovaj sistem štedi vreme razvoja jer nije potrebno obraćati pažnju na tehničku pozadinu implementacije pri korišćenju objavi/pretplati se mehanizma između anonimnih čvorova.

Druga korist korišćenja sistema prosleđivanja poruka jeste da korisnika primorava na implementaciju čistog interfejsa između čvorova u sistemu, i tako poboljšava enkapsulaciju i ponovno korišćenje već napisanog koda gde god je to moguće.

Struktura interfejsa poruka je definisana standardom message IDL (*eng. Interface Description Language*) [1].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Rapaić, red. prof.

3.2. Snimanje i reprodukcija poruka

Zbog toga što je objavi/pretplati se sistem asinhron i anoniman, podaci mogu jednostavno biti snimljeni i reprodukovani bez bilo kakvih izmena u kodu. Za primer se može uzeti komunikacija između Čvora A i Čvora B. Neka Čvor A čita podatke sa senzora, a razvija se Čvor B koji će da procesira podatke koje proizvede Čvor A. ROS čini jednostavnim da se podaci koje je Čvor A objavio čuvaju u datoteku.

Apstrakcija prosleđivanja poruka omogućava da Čvor B nema potrebu za poznavanjem stvarnog izvora podataka, što bi mogao biti Čvor A ili log fajl u ovom slučaju. Ovakav dizajn omogućava da se smanji napor uložen u razvoj, i povećavaju fleksibilnost i modularnost sistema.

3.3. Pozivi udaljenih procedura

Asinhrona priroda objavi/pretplati se sistema poruka radi za mnoge komunikacione potrebe u robotskim aplikacijama, ali su ponekad potrebne i sinhroni zahtev/odgovor (eng. request/response) interakcije između procesa. ROS kao srednji sloj (eng. middleware) omogućava korišćenje servisa. Podaci poslani između procesa u pozivu servisa su definisani u istom, jednostavnom opisu poruke (message IDL).

3.4. Jezik za opis robota – URDF

ROS omogućava set alata za opisivanje i modeliranje robota koji je kompatibilan sa ostatkom sistema. Format za opisivanje robota zove se URDF (eng. *Unified Robot Description Format*) koji se bazira na XML notaciji u kojoj se opisuju fizičke osobine robota, kao što su dužina, pozicija, boja i slične.

4. ALATI

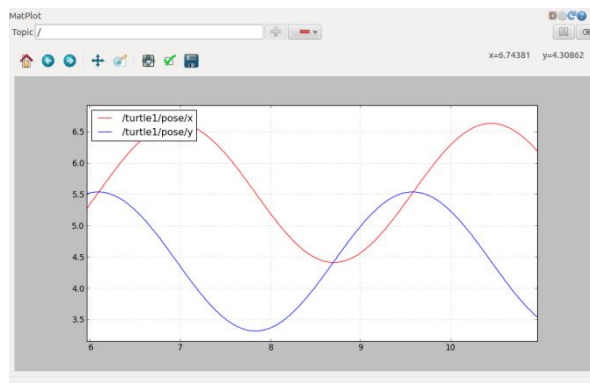
4.1. Alati iz komandne linije

ROS može biti korišćen u potpunosti bez grafičkog korisničkog interfejsa. Sve ključne funkcionalnosti su dostupne kroz više od 45 alata iz komandne linije. Tu su komande za pokretanje grupa čvorova, proveru topika i servisa, pravljenje snimaka i njihova reprodukcija kao i ostale situacije. Ukoliko korisnik preferira grafičke alate, [2] i *rqt* [3] omogućuju sličnu i proširenu funkcionalnost. Više detalja o ovim alatima dajemo u nastavku.

4.2. Rqt

ROS sadrži *rqt* [3], Qt [4] bazirani alat za razvoj grafičkih interfejsa za robota. Moguće je kreirati interfejse specifične za samu aplikaciju, kao i same dodatke za *rqt*. *rqt_graph* [5] dodatak omogućava proveru i vizualizaciju ROS sistema, prikazivajući čvorove i veze između njih, omogućavajući da se jednostavan pregled trenutnog sistema.

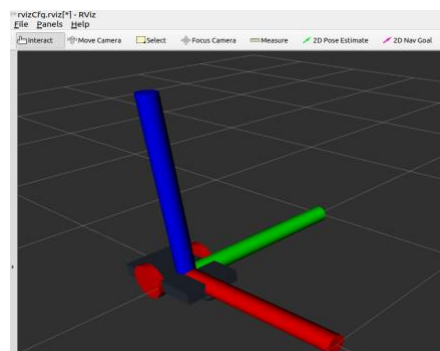
rqt_plot [6] dodatak omogućava monitoring raznih signala kao što su enkoderski signali, naponi, ili bilo koji drugi signali. Ovaj dodatak će crtati grafike izabranih signala. Slika 4.1 - Primer izgleda *rqt_plot* grafičkog interfejsa ilustruje izgled grafičkog interfejsa *rqt_plot* alatke.



Slika 4.1 - Primer izgleda *rqt_plot* grafičkog interfejsa

4.3. Rviz

Rviz omogućava 3-dimenzionu vizualizaciju mnogih tipova senzorskih podataka i robota opisanih u URDF formatu. Rviz može da prikaže mnoge tipove poruka ugrađenih u ROS, kao što su poruke sa laserskih skenera, trodimenzionih oblaka i slika sa kamera. On takođe koristi informaciju iz tf [7] biblioteke da bi prikazao sve podatke sa senzora u zajedničkom koordinatnom sistemu od korisnikovog izbora, zajedno sa 3D modelom robota. Vizualizacija svih podataka u istoj aplikaciji omogućava pregled toga što robot „vidi”, i da se uvide problemi kao što su neravnomernosti kod senzora ili nepravilnosti u modelu robota. Primer robota u rviz okruženju prikazan je na Slika 4.2 - Primer rviz korisničkog interfejsa Slika 4.2:



Slika 4.2 - Primer rviz korisničkog interfejsa

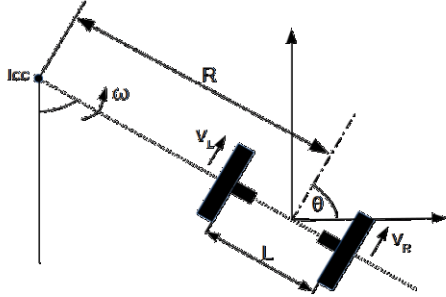
5. GAZEBO

Gazebo je programski paket otvorenog koda koji se koristi za simulaciju robotskih sistema i njihovog okruženja. Od 2004. do 2011. Gazebo je integrisao fizički podsistem za simulaciju dinamičkih sistema (opisanih diferencijalnim jednačinama – ODE engine), OpenGL renderovanje, kao i podršku za simulaciju senzora i kontrolu aktuatora.

Gazebo može da koristi više fizičkih podsistema za simulaciju, visokih performansi, kao što su ODE, Bullet i slični. Omogućava realistično renderovanje okruženja uključujući visoko kvalitetno osvetljenje, senke, i texture. Može da modeluje senzore koji „vide” simulirana okruženja, kao što su laserski senzori, kamere, radari i slični [8].

6. KINEMATIKA DIFERENCIJALNOG POGONA

Pošto točkovi mogu da se okreću različitim brzinama jedan u odnosu na drugi, može se uzeti slučaj da se centar robota kreće konstantno po zamišljenoj kružnici. U tom slučaju brzine točkova su različite jedna u odnosu na drugu, ali su konstantne. Tačka oko koje se kreću nazvana je ICC, i prikazana na Slika 6.1 u odnosu na centar diferencijalnog pogona.



Slika 6.1 - Kinematika diferencijalnog pogona

Promenom brzina točkova, moguće je varirati trajektoriju po kojoj se robot kreće. Pošto ugaona brzina rotacije ω mora biti ista za oba točka u slučaju kretanja oko tačke ICC, moguće je napisati sledeće jednačine:

$$\omega \left(R + \frac{L}{2} \right) = V_R \quad (1)$$

$$\omega \left(R - \frac{L}{2} \right) = V_L \quad (2)$$

gde je L distanca između centara točkova, V_R i V_L brzine desnog i levog točka respektivno, u odnosu na nepokretnu podlogu, i R oznaka za razdaljinu središnje tačke robota u odnosu na ICC. U bilo kom trenutku moguće je izraziti R i ω kao:

$$R = \frac{L V_L + V_R}{2 V_R - V_L}; \omega = \frac{V_R - V_L}{L}; \quad (3)$$

Tri slučaja su od posebnog interesa:

- 1) Ukoliko je $V_L = V_R$, onda je to pravolinijsko kretanje konstantnom brzinom. R postaje beskonačno, i nema rotacije, što znači da je $\omega = 0$.
- 2) Ukoliko je $V_L = -V_R$, tada je $R = 0$, i imamo rotaciju oko centralne tačke – rotacija u mestu.
- 3) Ukoliko je $V_L = 0$ i $V_R \neq 0$, tada imamo rotaciju oko levog točka. U tom slučaju, $R = \frac{L}{2}$. Ista logika važi i u obrnutom slučaju kad je $V_R = 0$ a $V_L \neq 0$.

6.1. DIREKTNI KINEMATSKI PROBLEM

Pitanje: kako će se robot kretati u prostoru ako znamo kretanje pojedinačnih točkova robota? Na Slika 6.1, treba primetiti da je robot na poziciji (x, y) , orijentisan pod uglom θ u odnosu na X osu.

Manipulacijom parametara V_L , V_R robot može biti pomeran na različite pozicije i biti različito orijentisan. Poznajući V_L i V_R , i koristeći jednačinu 3, može se naći sledeća lokacija tačke ICC:

$$ICC = [x - R \sin(\theta), y + R \cos(\theta)] \quad (4)$$

i u vremenu $t + \delta t$ pozicija robota biće:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\omega \delta t) & -\sin(\omega \delta t) & 0 \\ \sin(\omega \delta t) & \cos(\omega \delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICCx \\ y - ICCy \\ \theta \end{bmatrix} + \begin{bmatrix} ICCx \\ ICCy \\ \omega \delta t \end{bmatrix} \quad (5)$$

Ova jednačina jednostavno opisuje kretanje robota na distanci R oko ICC tačke, ugaonom brzinom ω .

6.2. INVERZNI KINEMATSKI PROBLEM

Pitanje: ako je zadato željeno kretanje mobilnog robota u prostoru, kako treba da se kreću pojedinačno točkovi, da bi se to kretanje i ostvarilo? Generalno, pozicija robota koji se kreće u smeru $\theta(t)$ brzinom $v(t)$ može se opisati kao:

$$x(t) = \int_0^t v(t) \cos(\theta(t)) dt \quad (6)$$

$$y(t) = \int_0^t v(t) \sin(\theta(t)) dt \quad (7)$$

$$\theta(t) = \int_0^t \omega(t) dt \quad (8)$$

Radi uprošćavanja putanje kretanja robota sa diferencijalnim pogonom, biće razmotrena dva slučaja:

- 1) Robot se kreće pravolinijski ($v_L = v_R = v$). Jednačina kretanja postaje:

$$\begin{bmatrix} X' \\ Y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x + v \cos(\theta) \delta t \\ y + v \sin(\theta) \delta t \\ \theta \end{bmatrix} \quad (9)$$

- 2) Robot se rotira u mestu ($v_R = -v_L = v$). Jednačina kretanja postaje:

$$\begin{bmatrix} X' \\ Y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta + 2v \delta t / L \end{bmatrix} \quad (10)$$

Ovaj pristup motivisan je strategijom da se robot kreće pravolinijski, zatim rotira u mestu, i zatim ponovo nastavlja pravolinijski po strategiji navigacije.

7. SOFTVERSKA ARHITEKTURA SISTEMA BAZIRANOG NA ROS_CONTROL PAKETU

Paket *ros_control* dobija stanja zglobova kao ulaze. Koristi generičku upravljačku petlju sa povratnom spregom, tipično PID regulator, da bi upravljao izlazima. Upotreba paketa se dodatno komplikuje pri upotrebi nad realnim sistemima koji nemaju jedan prema jedan mapiranje očitanih i realnih pozicija zglobova. Takvi problemi prevazilaze se korišćenjem transmisije.

Transmisije su tu da oponašaju mehaničke prenosnike, poput reduktora sa datim prenosnim odnosom n , tako što mapiraju promenljive ulaznog momenta na promenljive izlaznog momenta pri čemu snaga ostaje nepromenjena.

U nastavku sledi objašnjenje pojedinih delova ovakvog softvera. Controller Manager je komponenta zadužena za regulatore iz *ros_control* paketa. Najbitnije naredbe su one za učitavanje i uklanjanje regulatora (*load_controller* i *unload_controller*). Sledeća je naredba za promenu regulatora (*switch_controller*) i naredba za izlistavanje aktivnih regulatora (*list_controllers*). Funkcija *update()*

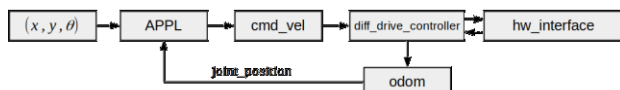
poziva se periodično i vrši ažuriranje raznih parametara koje koristi regulator, takođe i proračun upravljanja zavisno od tipa regulatora koji se koristi. Sledeća komponenta je komponenta zadužena za spregu sa spoljašnjim svetom u pogledu hardvera. Shodno ovoj komponenti, komunikacija će ići u pravcu simulacije ili realnog hardvera. Ova komponenta stanja zglobova dostavlja Controller Manager komponenti u radijanim, a od nje u ovom primeru preuzima željeni obrtni moment. Odatle se podaci mapiraju na klasu `hardware_interface`. U slučaju simulacije, metode ove klase će da generišu samo upravljanje ka objektu simulacije, a takođe će iz njega da čitaju stanja simuliranih hardverskih komponenti. Treba još napomenuti da je u ovom slučaju robot opisan kroz URDF alat, i da je potrebno učitati Gazebo dodatak nazvan `gazebo_ros_control`. Kada je realni hardver u pitanju, metode klase `hardware_interface` imaju zadatak da vrše njegovu apstrakciju. One su zadužene da realnom regulatoru prenesu naredbu. U ovom primeru to je moment koji je prethodno izračunat, a može da bude i pozicija ili brzina. Realni regulator može da bude mikrokontroler na kojem je implementiran PID regulator koji vrši regulaciju po parametru dobijenom iz hardverskog interfejsa. On dalje vrši upravljanje nad aktuatorima, i čita stanja sa senzora.

8. PRIMER: APLIKACIJA ROS_DIFF_DRIVE

8.1. UVOD

Aplikacija `ros_diff_drive` pisana je za potrebe ovog rada. Ideja je da se kreira robot koji će biti simuliran u Gazebo okruženju. Robot je pogonjen diferencijalnim pogonom. Opisan je korišćenjem URDF alata. Kompletan programski paket ove aplikacije nalazi se na Github platformi, na sledećoj web stranici: https://github.com/djordjevujić/ros_diff_drive.

Uprošćeni dijagram sistema softvera prikazan je na Slika 8.1:



Slika 8.1 - Uprošćeni dijagram softvera `ros_diff_drive` sistema

Pojednostavljeno objašnjenje segmenata dijagrama arhitekture sistema sledi u nastavku:

- x , y , θ – Željena pozicija robota u globalnom koordinatnom sistemu.
- APPL – Aplikativni sloj softvera. Donosi odluke o kretanju robota, u vidu koordinata tačke ka kojoj se robot kreće, željene rotacije, linearne i rotacione brzine
- `cmd_vel` – Komanda koja predstavlja željenu brzinu. U ROS sistemu, to je zaseban topik.
- `diff_drive_controller` – Niži sloj softvera koji vrši proračun brzine točkova. U odnosu na trenutnu, proračunava željene brzine točkova koje prosleđuje hardverskom interfejsu. Ova softverska biblioteka je deo standardnih ROS biblioteka.

- `hw_interface` – Hardverski interfejs. Služi kao apstrakcija hardvera u odnosu na ostatak sistema. Omogućava da se u odnosu na okruženje, u sistemu koristi realan hardver ili simulacija.

- `odom` – Topik koji predstavlja stanje odometrije diferencijalnog pogona. Vrednosti bivaju osvežene od strane sloja nazvanog „`diff_drive_controller`” koji vrši regulaciju brzine točkova.

8.2. INKREMENTALNI PID REGULATOR

Inkrementalni PID regulator implementiran je u datoteci `regulator.py`, u okviru klase `Regulator()`. Metoda za proračun upravljanja navedena je u metodi `pid_incremental`:

```

def pid_incremental(self, error):
    # Calculate control
    dUp = self.KP * (error - self.err_prev)
    dUi = self.KI * self.T * error
    dUd = self.KDT * (error - 2 * self.err_prev + self.err_p_prev)
    self.u = self.u + dUp + dUi + dUd

    # Anti wind-up
    if self.u > self.u_limit:
        self.u = self.u_limit
    elif self.u < -self.u_limit:
        self.u = -self.u_limit

    # Backup values needed for the next iteration
    self.error_p_prev = self.err_prev
    self.err_prev = error

    return self.u
  
```

8.3. MAŠINA STANJA

Procesiranje naredbe o željenoj tački, od čekanja naredbe, do kretanja i dostizanja željene tačke, implementirano je kroz mašinu stanja. Ova mašina ima tri stanja:

- 1) Čekanje naredbe,
- 2) Rotacija ka željenoj tački,
- 3) Pravolinijsko kretanje ka željenoj tački.

Graf sa slike Slika 8.2 predstavlja dozvoljene tranzicije stanja tokom izvršavanja naredbe:

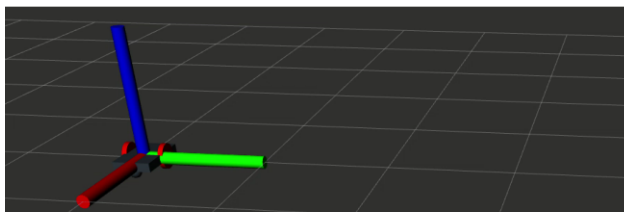


Slika 8.2 - Graf tranzicije stanja

Ova mašina stanja vezana je za kretanje ka željenoj tački. Ukoliko bi bilo željeno vezati više naredbi, spoljna mašina stanja bi mogla biti implementirana, i ona bi izvršavala trenutno opisanu mašinu stanja kada je to potrebno.

8.4. REZULTATI TESTA KRETANJA KA ZADATOJ TAČKI

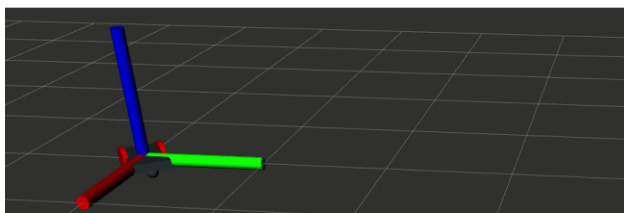
U nastavku je dat primer kretanja iz tačke (0.0), ka tački (5,5), pri čemu je početna orijentacija robota ugao od 0 stepeni.



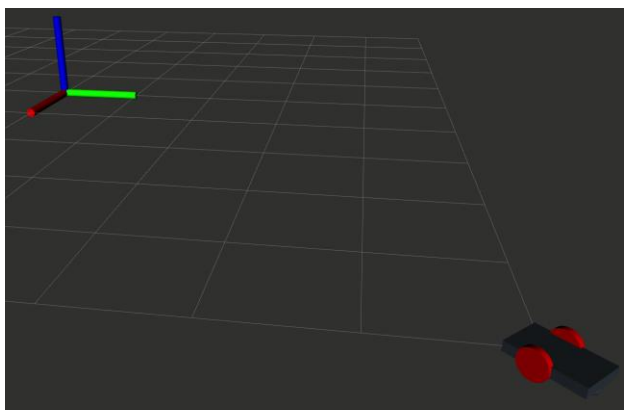
Slika 8.3- Inicijalna pozicija - čekanje naredbe

```
/target_position/position Goal: (5.0, 5.0)
[INFO] [1632763493.653956, 45.640000]: Robot M2XR: Switching state to FsmStates.Rotating
[IDL] New goal: (5.0,5.0)
[ROT] Angle to goal: 44.97359397681263, theta: 0.05565307042002554
[ROT] Angle to goal: 44.97359397681263, theta: 0.05565307042002554
[ROT] Angle to goal: 44.973571004170175, theta: 0.09213355976872485
```

Slika 8.4 – Konzola: ulazak u stanje rotacije



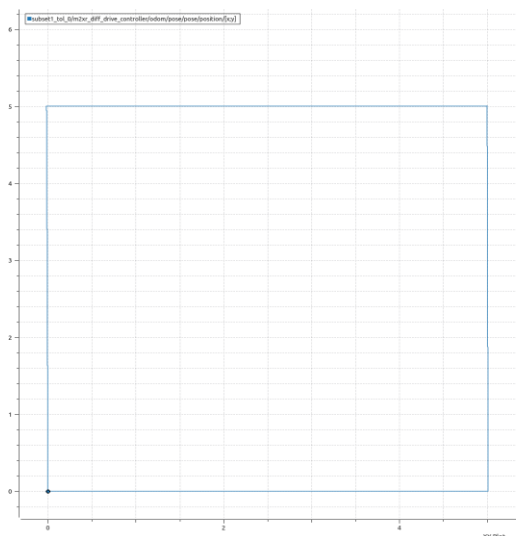
Slika 8.5 - Završetak rotacije



Slika 8.6- Završetak pravolinijskog kretanja

```
Point(5.009458512728726, 5.002450450578338) Dist to goal: 7.073810360365403, Current: 7.082235035141147
Point(5.023210310780652, 5.015949330185043) Dist to goal: 7.073810360365403, Current: 7.082235035141147
[1632763512.654600, 50.501000]: Robot M2XR: Switching state to FsmStates.Idle
Point(5.023210310780652, 5.015949330185043) Dist to goal: 7.073810360365403, Current: 7.101504362495414
```

Slika 8.7- Konzola: završetak pravolinijskog kretanja



Slika 8.8 - Rezultati kretanja trajektorijom kvadrnog oblika

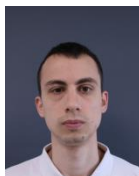
8.5. GRAFIČKI PRIKAZ TRAJEKTORIJE KRETANJA ROBOTA

Zadavane su komande tako da se robot kreće trajektorijom kvadratnog oblika, sa početkom u (0, 0), i dostižući sledeće tačke: (5, 0), (5, 5), (0, 5), i na kraju opet početnu(0, 0). Rezultati kretanja prikazani su na Slika 8.8, gde je prikazana trajektorija u XY koordinatnom sistemu.

9. LITERATURA

- [1] Object Management Group®. (2021) OMG - Standards Development Organization. [Online]. HYPERLINK "<https://www.omg.org/spec/IDL/>" <https://www.omg.org/spec/IDL/>
- [2] David Gossow, Josh Faust, William Woodall Dave Hersberger. (2018) ROS Wiki. [Online]. HYPERLINK "<http://wiki.ros.org/rviz>" <http://wiki.ros.org/rviz>
- [3] Dorian Scholz, Aaron Blasdel Thomas. (2016, Aug.) ROS Wiki. [Online]. HYPERLINK "<http://wiki.ros.org/rqt>" <http://wiki.ros.org/rqt>
- [4] The Qt Company. (2021) Qt. [Online]. HYPERLINK "<https://www.qt.io/>" <https://www.qt.io/>
- [5] Aaron Blasdel Dirk Thomas. (2018, Sep.) ROS Wiki. [Online]. HYPERLINK "http://wiki.ros.org/rqt_graph" http://wiki.ros.org/rqt_graph
- [6] Dirk Thomas Dorian Scholz. (2018, May) ROS Wiki. [Online]. HYPERLINK "http://wiki.ros.org/rqt_plot" http://wiki.ros.org/rqt_plot
- [7] Eitan Marder-Eppstein, Wim Meeussen Tully Foote. (2017, Oct.) ROS.org/tf. [Online]. HYPERLINK "<http://wiki.ros.org/tf>" <http://wiki.ros.org/tf>
- [8] Open Source Robotics Foundation. Gazebo. [Online]. HYPERLINK "<https://gazebo.org/>" <https://gazebo.org/>

Kratka biografija:



Đorđe Vujić rođen je u Novom Sadu 1995. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Adaptivno i napredno upravljanje odbranio je 2021. god.

Kontakt: djordjevujic@yahoo.com

**KVALITET ELEKTRIČNE ENERGIJE NA NISKONAPONSKOJ STRANI
ENERGETSKIH TRANSFORMATORA U POVRŠINSKOM KOPU "DRMNO"****ELECTRICITY QUALITY ON THE LOW VOLTAGE SIDE OF POWER
TRANSFORMERS IN THE OPEN PIT MINE "DRMNO"**Nemanja Živković, Vladimir A. Katić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu opisano je merenje i analiza kvaliteta električne energije na 0,4 kV strani energetskih transformatora. Merenja su vršena na trafostanici Rudnik 4 i Pogonskoj stanici B-2000. Prikazani su rezultati merenja, njihova analiza i upoređivanje sa datim standardima i propisima.

Ključne reči: Kvalitet električne energije, energetski transformator, pogonska stanica B-2000, površinski kop Drmno.

Abstract– The paper describes the measurements and analysis of electric power quality at the 0.4 kV side of a power transformers. Measurements were done at the substation Rudnik 4 and the power station B-2000. The results of measurements, their analysis and comparison with the given standards and regulations are presented.

Keywords: Power quality, power transformer, Belt conveyor B-2000, open pit mine "Drmno".

1. UVOD

Zbog svoga značaja u životu savremenog sveta, električna energija, podleže određenim normama, standardima ili preporukama u pogledu raspoloživosti i pouzdanosti napajanja, kao i stabilnosti osnovnih parametara. Polazeći od toga, dobija se veoma široka postavka pitanja, koja su objedinjena pod pojmom kvaliteta električne energije, odnosno može se govoriti o:

1. Upotrebom kvaliteta,
2. Ekološkom kvalitetu i
3. Tehničkom kvalitetu električne energije [1].

Najčešće kada se pominje kvalitet električne energije, podrazumeva se tehnički kvalitet. Tehnički kvalitet ima prvorazredni značaj i pažnju, zbog toga što je jedan od važnijih parametara, koji određuje cenu i atraktivnost proizvoda ili usluge. Stoga, on obuhvata kvalitet isporuke, kvalitet isporučene električne energije i komercijalni kvalitet (kvalitet tržišnih odnosa).

Kvalitet isporuke uključuje probleme pouzdanosti, efikasnosti i sigurnosti napajanja, koji se rešavaju u postupku planiranja, razvoja i tokom redovnog održavanja mreže.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Vladimir Katić, red. prof.

Kvalitet isporučene električne energije generalno je vezan za kvalitet napona na priključcima potrošača i vezan je za rad potrošača, poremećaje u radu sistema, kao i međusobne uticaje.

Komercijalni kvalitet definiše uređenost odnosa snabdevač (distribucija) i potrošač i njihovu interakciju.

Ukoliko kvalitet električne energije nije zadovoljavajući može se javiti niz problema. Među najznačajnijim problemima je pojava viših harmonika, koji imaju za posledicu:

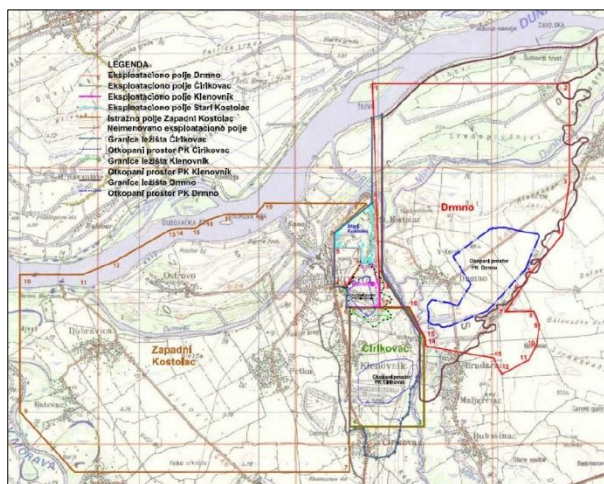
- Prevelika struja nultog provodnika,
- Pregrevanje transformatora,
- Pregrevanje i nepravilan rad električnih motora,
- Uticaj harmonika na prekidače i zaštitne releje,
- Uticaj na elektronsku opremu i komunikacije,
- Uticaj na rad statičkih energetskih pretvarača,
- Pojava rezonancije u mreži,
- Naprezanje kondenzatorskih baterija za kompenzaciju reaktivne energije,
- Izobličenje mrežnog napona usled toka harmonijskih struja i dr.

U ovom radu razmatraće se kvalitet isporučene električne energije na niskonaponskoj (NN) strani energetskih transformatora sa koje se snabdevaju električnom energijom potrošači u površinskom kopu "Drmno", kod Kostolca. Cilj je da se utvrdi stanje kvaliteta električne energije i poređenjem sa relevantnim standardima da njegoja ocean.

2. O POVRŠINSKOM KOPU "DRMNO"

Kostolac se nalazi u peri-panonskom delu severo istočne Srbije, na udaljenosti od Beograda od 90 km, gde se Požarevačka greda (pobrđe istočno od Velike Morave) spušta na obale Dunava. Nalazi se zapadno od Velike Morave, a periferni delovi grada prostiru se do desne obale Dunava, na obalama Dunavca.

Istočno od Kostolca, na desnoj obali Velike Morave na površini od oko 3074 ha nalazi se površinski kop (PK) "Drmno". Na slici 1 prikazan je kompletan kostolački rudarsko-energetski basen sa označenim kopom "Drmno" [2]. U jugoistočnom delu prilikom otvaranja i razrade površinskog kopa formirano je spoljašnje odlagalište, koje zauzima prostor od oko 196 ha. Nakon toga formirano je unutrašnje odlagalište sa pet etažih nivoleta. Takođe, prema termoelektrani "Kostolac A" nalazi se transportni sistem za prevoz uglja koji spada u osnovnu opremu kopa "Drmno".



Sl. 1. Kostolački rudarsko-energetski basen [2]

2.1. Napajanje PK “Drmno”

Površinski kop “Drmno” raspolaže sa transformatorskom stanicom (TS) „Rudnik 1“ TS 110/6 kV, 2 x 16 MVA, smeštenoj na istočnoj strani kopa, zatim „Rudnik 3“ TS 110/6 kV, 2 x 16 MVA, smeštenoj na istočnoj strani kopa, te TS 110/6 kV, 1 x 16 MVA, „Rudnik 2“ smeštenoj na zapadnoj strani kopa, u selu Drmno, TS 35/6 kV, 6,3 MVA, MO1 i mobilnom trafostanicom MO1 smeštenoj na zapadnoj strani kopa. Na osnovu raspoloživog kapaciteta se može reći da je njihova instalirana snaga dovoljna za snabdevanje energijom kopa proizvodnje od $9 \cdot 10^6$ tona uglja ali nedovoljna za predviđeno povećanje kapaciteta od $12 \cdot 10^6$ tona uglja na godišnjem nivou. U okviru snabdevanja kopa energijom nalaze se i 110 kV dalekovodi:

- dalekovod br. 1196 Veliko Gradište - Rudnik 1, dužina 22,9 km,
- dalekovod br. 1128/1, TE Kostolac A – TS Rudnik 1, dužina 12,074 km,
- dalekovod br. 1128/2, TS Rudnik 1 – TS Rudnik 2, dužina 6,735 km,
- dalekovod br. 1160 TS Rudnik 2 – RP Drmno, dužine 2,572 km i
- dalekovod br. 1159 RP Drmno - TE Kostolac A, dužine 9,9 km.

Takođe, kop poseduje i nadzemnu kablovsku 6 kV mrežu i trafo stanice 6/0,4 kV.

2.2. Radni pogon PK “Drmno”

Na PK „Drmno“ otkopavanje otkrivke vrši se u okviru šest BTO (bager-traka-odlagač) sistema, dok se transport iste obavlja dvokrilno po istočnoj i zapadnoj strani kopa. Eksploatacija uglja vrši se duž dva sistema etažnih transportera, sa zajedničkim izvoznim transporterima do lokacije raspodelnog bunkera.

Otkopavanje i transport uglja obavlja se pomoću BTĐ (bager-traka-deponija) sistema, koji obuhvata dva sistema etažnih transportera, pri čemu na prvoj etaži rade rotorni bageri u tandemu sa samohodnim transporterom u visinskom radu, dok je na drugoj etaži pozicioniran bager vedričar koji radi dubinski.

3. POGONSKA STANICA B-2000

Za pogon transportne trake koristi se pogonska stanica B-

2000 koja ima četiri pogona za okretanje transportne trake. Elektromotori su predviđeni za rad sa frekventnim pretvaračima, odabrani prema mašinskoj specifikaciji, Minimalna snaga motora pri radu sa frekventnim pretvaračem je 1000 kW (snaga bez pretvarača 1300 kW), nominalne brzine 1000 ob/min.

Brzina kretanja trake je oko 5 m/s. Ugrađene su četiri jedinice frekventnih pretvarača. Ova pogonska stanica napaja se iz TS “Rudnik 4” 6,9/0,4 kV.

Većina razvijenih zemalja sveta, kao i najznačajnije međunarodne oragnizacije su donele tehničke propise – standard sa navođenjem graničnih vrednosti osnovnih parametara kvaliteta električne energije od kojih zavisi stabilnost rada EES-a (efektivna vrednost napona, frekvencija i sl.). Ostali parametric spominjani ranije su definisani pojedinačno ili u sklopu sveobuhvatnijih dokumenata o kvalitetu električne energije. Ovi propisi su u obliku ili obavezujućih ili neobavezujućih preporuka.

4. MERENJE KVALITETA ELEKTRIČNE ENERGIJE

4.1. Merni sistem

Za merenje je korišćen mrežni analizator prenosnog tipa, Dranetz Power Guide 4400, prenosivi, ručni, osmokanalni profesionalni merni system (slika 2) [3]. On može da nadgleda, snima i prikazuje podatke na četiri naponska kanala i četiri strujna kanala istovremeno. Dizajniran je da zadovolji kako IEEE 1159 tako i IEC 61000-4-30 standarde za zahtevanu tačnost i merenja. On takođe može pratiti usklađenost na osnovu EN (evropskih) standard, EN 50160, koji je identičan sa srpskim standardom SRPS EN 50160:2012 [4].



Sl. 2. Mrežni analizator Dranetz Power Guide 4400 [3]

4.2. Merno mesto 1 (MM1)

Mrežni analizator je postavljen 11.08.2021. na niskonaponskom razvodnom ormanu i izvršena su merenja parametara i kvaliteta električnog napajanja na objektu „Pogonska stanica B-2000“ u trajanju od 7 dana. Merni instrument je postavljen na ormanu niskog napona trafo polja na Pogonskoj stanici B-2000 kao što se može videti na slici 3.

4.3. Merno mesto 2 (MM2)

Mrežni analizator je postavljen 18.08.2021. na niskonaponskom razvodnom ormanu i izvršena su merenja parametara i kvaliteta električnog napajanja na objektu TS “Rudnik 4” 6,9/0,4 kV u trajanju od 7 dana. Merni instrument je postavljen na ormanu niskog napona trafo polja u trafostanici, kao što se može videti na slici 4.

4.4. Standard SRPS EN 50160

Za analizu mernih rezultata primenjene su odredbe srpskog standarda SRPS EN 50160:2012, kao i uslovi merenja [4].



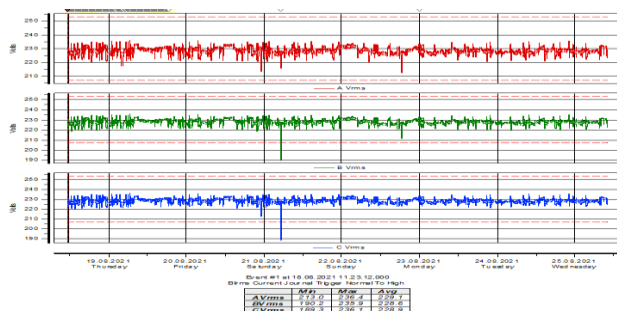
Sl. 3. Postavka mrežnog analizatora – MM1 (levo) i MM2

5. REZULTATI MERENJA

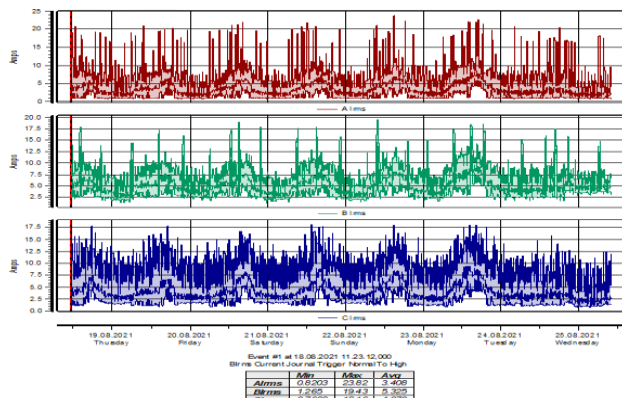
Na slikama 4 i 5 se nalaze dijagrami promene efektivnih vrednosti faznih napona i struja u sve tri faze u posmatranom periodu.

Efektivne vrednosti napona su uglavnom stabilne (slika 4) i kreću se oko $\pm 5\%$ nazivnog napona uz pojedina kratkotrajna smanjenja na od -10% (faza A) do -18% (C). Vidi se da se efektivne vrednosti faznih struja (slika 5) kreću se u granicama od 23,82 A do 0,7602 A u zavisnosti od doba dana i opterećenja mreže. Sa grafika se može videti da u toku svih dana u nedelji postoje jednake oscilacije opterećenja, uključujući i vikend. Treća faza je manje opterećena od prve dve.

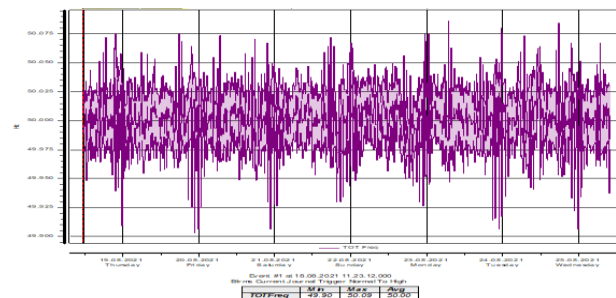
Nominalna frekvencija napona napajanja je 50 Hz. Pri normalnim pogonskim uslovima srednja vrednost osnovne frekvencije od deset sekundi u nekoj distributivnoj mreži mora biti u sledećim opsezima 50 Hz $\pm 1\%$ (tj. od 49,5 Hz do 50,5 Hz) tokom 95% nedelje, odnosno 50 Hz $-6\% / +4\%$ (tj. od 47 Hz do 52 Hz) tokom preostalih 5% nedelje, prema standardu SRPS EN 50160 [4]. Na slici 6 prikazano je kretanje frekvencije u periodu merenja i može se videti da joj se vrednosti kreću od 49,90 Hz do 50,09 Hz, što znači da je zadovoljen kriterijum iz pomenutog standarda.



Sl. 4. Efektivne vrednosti faznih napona

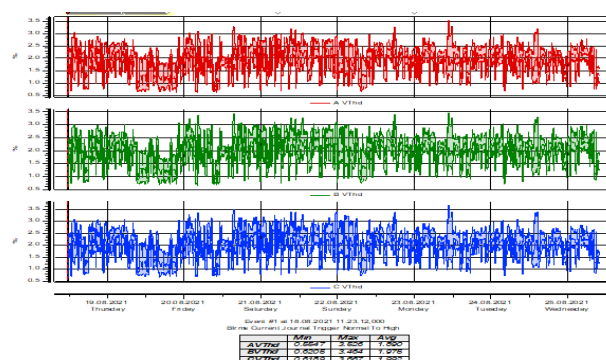


Sl. 5. Efektivne vrednosti faznih struja



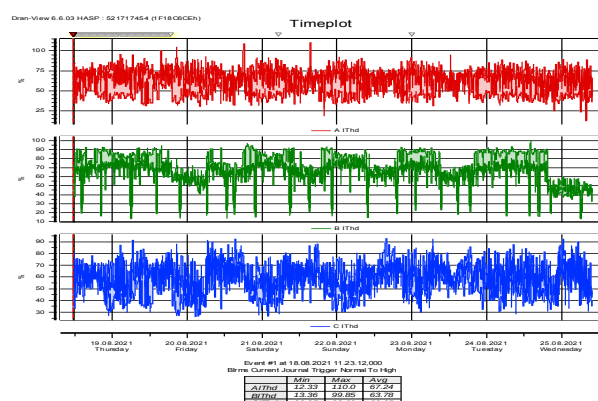
Sl. 6. Frekvencija

Viši harmonici su dobijeni u obliku ukupne harmonijske distorzije (THD) nakon FFT analize snimljenih talasnih oblika napona i struje. Na slici 7 je prikazan THD napona, koji ne prelazi vrednost od 3,67%. Može se zaključiti da ne postoji povećan THD napona, s obzirom na 8% granicu koja je predviđena srpskim standardom.



Sl. 7. Promena THDU tokom perioda merenja

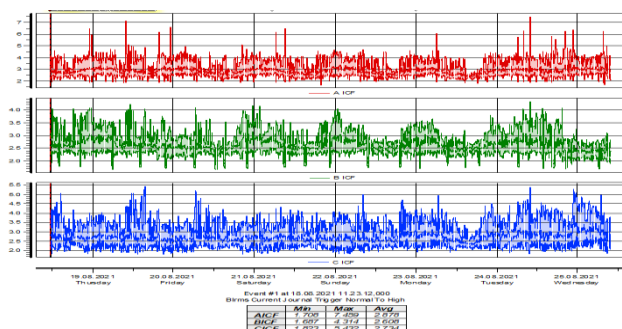
U pogledu totalnog harmonijskog izobličenja struje (THDI) može se zaključiti da je on povećan. Vrednosti THDI tokom vremena merenja prikazane na su slici 8. Može se uočiti da su dobijeni iznosi veći u odnosu na vrednosti predviđene standardima (THDI<20%). Maksimalna vrednost THDI za period merenja od 7 dana je čak 110%, što ukazuje na moguće pojave problema.



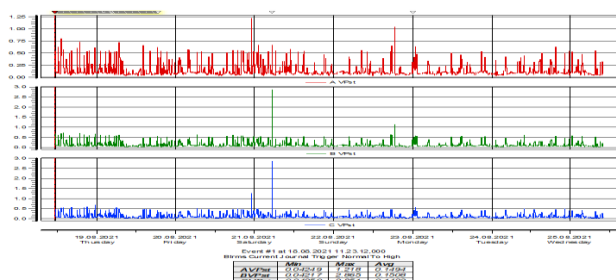
Sl. 8. Promena THDI tokom perioda merenja

Rezultati snimanja promene krest faktora struje prikazani su na slici 9. Sa slike se može zaključiti da i krest faktor struje prelazi očekivane vrednosti, što potvrđuje izobličenje talasnog oblika struje.

Na slici 10 prikazane su promene flikera u posmatranom periodu. Na osnovu prikazanih grafika uočljivo je nekoliko velikih pikova u sve tri posmatrane faze.

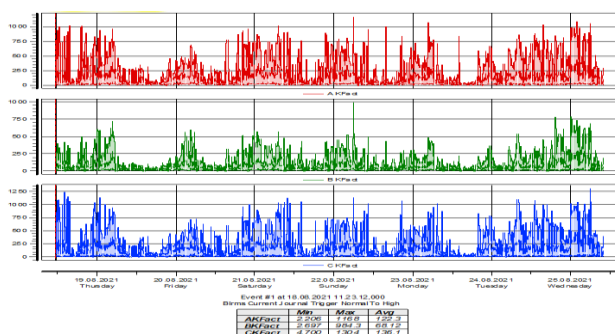


Sl. 9. Promena krest faktora struje u toku 7 dana



Sl. 10. Promena flikera napona u toku 7 dana

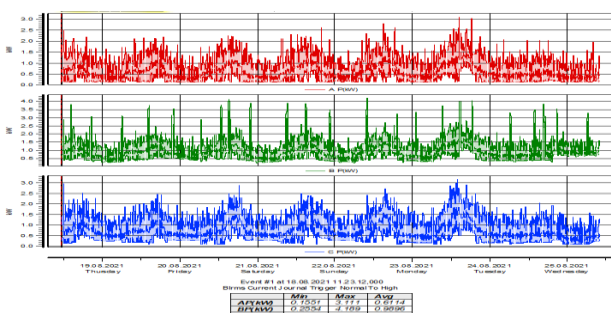
Na slici 11 prikazanu su rezultati merenja K-faktora. Što je veći K faktor to su veći gubici u transformatoru izazvani povećanim nivoim viših harmonica, pa se sa slike 11 može zaključiti da K faktor može predstavljati problem. Olakšavajuća okolnost je ta što kada je K faktor veliki, efektivna vrednost struje je mala pa se transformator neće pregrevati.



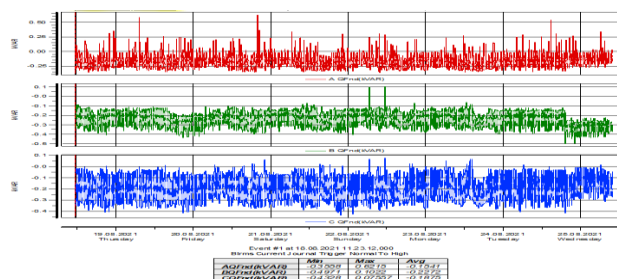
Sl. 11. Promena K faktora u toku 7 dana

Na slikama 12-14 prikazani su nedeljni dijagrami aktivne, reaktivne i prividne snage. Jasno se sa prikazanih grafika vidi da je potrošnja kako aktivne tako i reaktivne snage ujednačena svih 7 dana u nedelji.

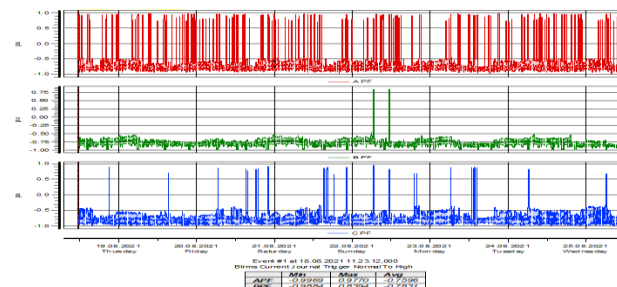
U pogledu vrednosti faktora snage treba napomenuti da se isti nalazio u opsegu očekivanih vrednosti (od -1 do +1).



Sl. 12. Dijagram aktivne snage u toku nedelju dana



Sl. 13. Dijagram reaktivne snage u toku nedelju dana



Sl. 14. Promena faktora snage tokom merenja

6. ZAKLJUČAK

Upoređujući dobijene rezultate merenja, sa aspekta kvaliteta napona može se zaključiti da su neki indikatori kvaliteta u skladu sa nacionalnim standardom, ali i da ima odstupanja. To se odnosi na vrednosti THDI i krest faktora struje. Ovo može potencijalno predstavljati problem. Takođe, K faktor je u pojedinim slučajevima bio veliki, što može biti problem za transformator, koji se tada dodatno greje, a izolacija brže stari.

7. LITERATURA

- [1] V. Katić, A. Tokić, T. Konjić, „Kvalitet električne energije“, TEMPUS-CEFES, FTN, Novi Sad, 2007.
- [2] V. Bulatović, M. Trifunović, „Razvojne perspektive površinske eksploatacije uglja u kostolačkom basenu“, XIII međunarodna konferencija OMC 2018, Zlatibor, 17-20. 10. 2018., pp.6-13.
- [3] Power guide 4400, User's guide, Dranetz, New Jersey, 2005.
- [4] SRPS EN 50160:2012, „Karakteristike napona isporučene električne energije iz javnih električnih mreža“, Institut za standard. Srbije, Beograd, 2012.

Kratka biografija:



Nemanja Živković rođen je u Banja Luci 1992. godine. Osnovne studije završio je na Fakultetu tehničkih nauka 2018. god. iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine. Master rad odbranio je 2021. godine, na istom fakultetu.



Vladimir Katić rođen je u Novom Sadu 1954. Doktorirao je na Univerzitetu u Beogradu 1991. god., a od 2002 je u zvanju redovnog profesora Univerziteta u Novom Sadu. Oblasti interesovanja su energetska elektronika, obnovljivi izvori električne energije, kvalitet električne energije i električna vozila.

ПРИМЕНА ETL ПРОЦЕСА У ПОСТУПКУ ТРАНСФОРМАЦИЈЕ РЕЛАЦИОНЕ БАЗЕ ПОДАТАКА У ГРАФОВСКИ ОРИЈЕНТИСАНУ БАЗУ ПОДАТАКА**APPLICATION OF ETL PROCESS IN THE TRANSFORMATION PROCEDURE OF A RELATIONAL DATABASE INTO GRAPH-ORIENTED DATABASE**

Добривоје Ђурђевић, *Факултет техничких наука, Нови Сад*

Област – ПРИМЕЊЕНЕ РАЧУНАРСКЕ НАУКЕ И ИНФОРМАТИКА

Кратак садржај – У овом раду представљено је коришћење ETL (Extract Transform Load) процеса у циљу аквизиције података за апликацију музичке енциклопедије коју је могуће претраживати постављањем упита на природном језику. Ови процеси представљају унапређење система који је у ту сврху у својој првој верзији користио web crawler. За пружање одговора на постављено питање систем користи технике машинског учења *sequence to sequence*, ради преводјења питања са природног на упитни језик графовске базе и пружање одговора. За имплементацију мапирања базе података коришћена су два ETL процеса. Први ETL процес мапирање из релационе у релациону базу података и реализован је у ODI (Oracle Data Integrator) алату. Док је други ETL процес мапирања из релационе у графовски оријентисану базу података, реализован коришћењем Neo4j ETL алата.

Кључне речи: релациона база података, графовски оријентисана база података, ETL процес, трансформација, моделовање

Abstract – This paper will present the use of the ETL (Extract Transform Load) process, used in order to acquire data for the application of the music encyclopedia that can be searched by querying in natural language. These processes represent an improvement on a system that used a web crawler for data acquiring in its first version. To provide an answer to the question, the system uses *sequence to sequence machine learning techniques*, in order to translate the question from natural to query language of the graph database and answer provisioning. Two ETL processes were used to implement database translation from relational to graph model. The first is mapping from a relational to a relational database and is implemented in the ODI (Oracle Data Integrator) tool. While the second ETL is a mapping process that maps a relational to a graph-oriented database, using the Neo4j ETL tool.

Keywords: Relational database, Graph-Oriented database, ETL process, transformation, modeling

1. УВОД

Информационе технологије чине неизоставни део модерног друштва, могу се пронаћи у практично сваком аспекту свакодневног живота, било да представљају примарни или алтернативни начин за извршавање неког задатка. Позитивни ефекти које носе са собом су многи, олакшавају велики број активности и задатака који су пред нама, пружају различите начине за комуникацију, аутоматизују послове и производњу, убрзавају банкарске услуге и пословање, олакшавају едукацију и обезбеђују доступност информација на длану. У основи скоро сваког информационог система постоји и база података задужена да одржи интегритет и доступност података који су од интереса.

Различити начини коришћења и конзумације информација довели су до експанзије и креирања нових типова база података. Поред класичних и свакако најраспрострањенијих релационих база података, које већ дуго предстаљају стандард у индустрији, све су више заступљене NoSQL, Document, Key-Value, хијерархијске, графовски оријентисане и друге базе података, доступне за интеграцију у DW (data warehouse) и cloud оријентисаним системима. Поред тога у последњих двадесет година настали су и концепти попут Big Data и Blockchain технологија који се заснивају на информацијама које анализирају, структурирају и чине их доступним у новом облику.

Основа за писање овог дипломског рада представља наставак развоја дигиталне музичке енциклопедије, односно концепта који је замишљен и започет као тема за бечелор дипломски рад аутора [1]. У тренутном облику ова музичка енциклопедија садржи малу количину информација из домена, приближно 5000 чворова и 5000 веза смештених у графовски оријентисану базу података. Подаци су прикупљени уз помоћ web scraping техника. Како би се ова апликација од концепта претворила у корисну енциклопедију која садржи велику количину података и која може да пружи одговор на бројна питања из домена музике, потребно је обезбедити те податке и структурирати их. Управо то представља мотивацију за писање овог мастер рада, овакав задатак пред себе поставља могућност истраживања и коришћења различитих технологија за аквизицију података.

Поменути систем који се ослања на web scraping технике како би обезбедио податке за енциклопедију ограничен је и тежак за одржавање, стога главни циљ овог пројекта је заменити овај систем ефикаснијим,

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији је ментор био др Милан Челиковић, доцент.

робуснијим и системом који ће моћи да обезбеди велику количину података. Циљ овог рада је истраживање и имплементација *ETL* [2] процеса. Многи информациони системи користе *ETL* процесе приликом реорганизације података из различитих разлога, попут миграције базе података у неки нови облик, било да то диктира модернизација система, потреба за променом основног концепта, скалирање система, интеграција различитих извора података у један или подела једног система на више мањих уже специјализованих. Циљ пројекта представља преузимање података из бесплатне релационе базе податка и њихова трансформација у графовски оријентисану базу податка. Реализација овог задатка заснива се на коришћењу *ETL* процеса при интеграцији података кроз *ODI* [3] платформу и *Neo4j ETL* алата.

2. ПРЕГЛЕД ТРЕНУТНОГ СТАЊА У ОБЛАСТИ

У оквиру овог поглавља описан је концепт *ETL* процеса (због своје основне намене да се подаци из једног облика и извора складиштења трансформишу и пребаце у други облик и у други тип складишта) током времена и према потреби корисника, креирани су различити алати и приступи за моделовање и реализацију оваквих процеса. Данас их има много и тешко их је категорисати и приказати све. У овом поглављу ће бити приказан основни ток једног *ETL* процеса, могућа варијација основном концепту *ETL* процеса, са прегледом најпопуларнијих алата који се користе. Поред овога користе се различити приступи при моделовању и оптимизацији поменутих процеса, а они место проналазе у различитим окружењима и технологијама попут нпр. у анализи *Blockchain*-а [4]. *DW* системи се заснивају на прикупљању, управљању и анализи великих сетова података. Као ефикасан начин за прикупљање, трансформацију и континуалну интеграцију различитих извора *ETL* процеси често чине саставни део *DW* система [5].

2.1 *ETL* Процес

ETL процес [6] представља процедуру која за циљ има копирање податка из једног или више извора у одредишни систем који репрезентује податке у другачијем контексту у односу на почетни систем. Овај процес укључује преузимање *extract* података из хомогених или хетерогених извора, трансформацију *transform* података према потребама система који их преузима у циљу складиштења, анализе и претраге. Последњи корак у овом процесу представља перзистенција података *load* у жељену базу податка. *Extract* као почетни корак представља увод у овај процес, како би подаци били даље процесирани, битно је да буду преузети успешно и да се при томе очува њихов интегритет, ово може посебно да буде изазовно уколико се преузимање врши из више различитих извора. Такође у овом кораку често се ради са различитим моделима, сваки извор података практично може бити имплементиран са другом технологијом и према другом формату података, које је потребно ускладити. У склопу преузимања података потребно је извршити и валидацију, тај корак може додатно да филтрира и обезбеди тачније податке у циљаној бази података.

Transform - трансформација податка као следећи корак, према дефинисаном сету правила припрема податке за циљани модел и њихово уписивање у циљану базу података. Још један циљ овог корака је *data cleansing* – чишћење података које додатно обезбеђује да само жељени подаци буду прослеђени финалном кораку и буду сачувани у новој бази, ово такође представља изазов када је више извора из којих се подаци преузимају.

Load – уписивање података у *flat file*, базу податка релационе или било које друге парадигме или *DW*, са инкременталним или *overwrite* приступом, ово варира од конкретне примене, али неопходно је да се трансформисани подаци сачувају и да буду доступни у новом облику за даље коришћење.

3. ОПИС СИСТЕМА

Основна идеја коју има имплементирана апликација, односно музичка енциклопедија, је да пружи брз одговор кориснику на различита питања из домена музике и да то оствари пруживши кориснику једноставан начин за интеракцију са системом. Замисао је да се питања попут: које инструменте одређени извођач свира, из које државе или града он потиче, да ли је члан неке групе, ко су остали чланови те групе, листа свих нумера које изводи група или албуми које је издала, одакле потиче одређени инструмент и како је настао и сл., постављају на једноставан начин како би се добио тражени одговор. Добро познати системи који су послужили као инспирација и који пружају овакве функционалности су персонални асистенти базирани на техникама вештачке интелигенције, међу којима су вероватно најпознатији „Google Assistant“, „Alexa“ и „Siri“. Овакви системи пружају велику базу знања и одговор на скоро свако питање са једноставним и интуитивним начином интеракције. Управо те функционалности захтевају коришћење различитих техника и решавање разноврсних проблема, што њихову имплементацију чини веома комплексном и изазовном. Циљ овог пројекта је био имплементирање једног система који извршава сличан задатак и истраживање техника које би могле у ту сврху да послуже. Као домен је изабрана музика, јер је са овим ограничењем лакше обезбедити базу знања коју систем може да претражује.

3.1. Архитектура система

Основна архитектура система приказана је на слици 1. Систем се састоји од три компоненте од којих је прва софтверски агент са дуалном улогом – пружа начин интеракције корисника са системом и омогућава му претрагу базе података. Као лак и интуитиван начин интеракције софтверски агент кориснику омогућава да постави систему питање на природном језику, односно на енглеском језику. Затим се постављено питање преводи на упитни језик графовски оријентисане базе, односно у овом случају *Cypher* језик. Овако преведено питање систем користи за претраживање базе података, при чему се добија повратна информација као одговор.



Слика 1: Архитектура предложеног система

Друга компонента система је графовски оријентисана база података. Овакав модел је изабран због могућности једноставног претраживања веза између типова ентитета, без потребе прорачунавања веза приликом извршавања упита и интуитивности коју пружа при анализи ових веза. Такође због својих карактеристика постоји могућност једноставних измена и додавања нових веза.

Трећа компонента је систем за аквизицију података, а њен задатак је прикупљање и упис података од интереса у базу података. Систем се ослања на претраживање отворене и бесплатне базе података *MusicBrainz*.

3.2. Унапређење система

Идеја за унапређење система у овој фази је прибављање што већег броја података. Како је *MusicBrainz* база података отвореног типа, постоји могућност да се креира локална копија ове базе. Локална копија пружа одличне могућности за преузимање свих или само неких података који су од интереса. Сама база садржи велики број података који је већ прикупљен и структуриран, што у великој мери олакшава задатак.

С обзиром да музичка енциклопедија користи графовски оријентисану базу података, а да је *MusicBrainz* база релационог типа, ипак је потребно извршити одређена прилагођавања како би било могуће преузимање података. Унапређење система обухвата замену подсистема за аквизицију података базираног на *web scraping* техникама са *ETL* процесима у циљу прикупљања и трансформације података у жељени облик.

4. СИСТЕМ ЗА АКВИЗИЦИЈУ ПОДАТАКА

У овом поглављу описан је систем за аквизицију података, ова подкомпонента система представљена је у првој верзији овог пројекта, као основни начин за аквизицију података.

Систем за аквизицију података је замишљен као аутоматизована софтверска компонента, што значи да након покретања од стране корисника све своје функционалности обавља самостално. За циљ има претрагу и прикупљање података од значаја за систем и њихово уписивање у базу података. Систем је имплементиран као *web scraper* коришћењем *.Net framework-a*, *C#* програмског језика и *Abot Web Crawler framework-a*.

5. ПРЕВОЂЕЊЕ РЕЛАЦИОНЕ У ГРАФОВСКИ ОРИЈЕНТИСАНУ БАЗУ ПОДАТАКА

Релационе и графовски оријентисане базе се у великој мери концептуално разликују. Ипак основна намена им је у суштини иста – да обезбеде перзистенцију података и релација међу њима и да обезбеде интегритет и доступност података структурираних према шеми базе.

Једна од очигледних разлика је начин на који су подаци повезани. Основна компонента графа је веза

између два чвора, која има подједнаку важност као и сами подаци које повезује. Ово граф чини интуитивним, лаким за интерпретацију и анализу веза између података. Везе као и чворови у графу могу имати атрибуте који их додатно описују. За разлику од графа у релационом моделу, везе између ентитета се реализују референцирањем на страни кључ, због чега су везе у релационом моделу мање очигледне, теже уочљиве поготово код комплексних модела.

Повезивање ентитета се прорачунава приликом извршавања *SQL (Structured Query Language)* упита коришћењем *JOIN* клаузуле и ослања се на подударање примарног и страног кључа. Ове операције могу бити веома захтевне за системске ресурсе у зависности од упита и количине података која се претражује и повезује. Приликом повезивања ентитета са кардиналитетом *N:N (many-to-many)* неопходно је увести *JOIN* табелу која садржи стране кључеве оба типа ентитета и тиме даље повећава захтеве за системским ресурсима.

Све ово релациони модел чини мање пожељним ако је анализа веза један од примарних захтева система, као што је у овом случају пружање одговора на питања попут „*Who are members of <Artist group>?*“.

Основне компоненте које чине релациони модел података присутне су и у графу као моделу. У циљу лакшег превођења релационог у графовски модел

[7], дате су следеће смернице:

1. Табела у релационом моделу еквивалентна је лабели у графовском моделу.
2. Сваки ред у табели релационог модела представља један чвор у графу.
3. Колона у табели релационог модела представља атрибут чвора у графу.
4. Страни кључ у релационом моделу замењује се везом између чворова у графу и није га потребно после чувати као атрибут чвора.
5. *JOIN* табела релационог модела трансформише се у везу између чворова, а колоне ове табеле постају атрибути те везе.

5.3 SQL TO SQL, први ETL процес

Први процес за основни циљ има преузимање података од интереса из релационе базе података *MusicBrainz* и њихову трансформацију у нови релациони модел који се затим уписује у *Oracle DW* базу података. Поред тога у овом кораку се врши припрема за наредни *ETL* процес који представља трансформацију новог релационог модела у модел графовски оријентисане базе. Мапирање се врши уз помоћ *ODI* алата, који на лак начин може да обезбеди интеграцију више различитих извора у одредишну базу података, па стога он практично заузима централно место приликом трансформације података.

5.4. SQL TO GRAPH, други ETL процес

Neo4j desktop поред *DBMS (Database Management System)* за графовски оријентисане базе података садржи и скуп алата који омогућавају различите операције. Један од тих алата је *Neo4j ETL* алат, који долази са графичким интерфејсом али и као *CLI (Command-line interface)* алат. За овај алат се

дефинише изворна база податка, у овом случају то је *Oracle DW* из претходног *ETL* процеса, затим се преузимају потребне информације и креира се графовски модел. При реализацији овог процеса коришћен је алат са графичким интерфејсом, јер су биле потребне само мање измене, приликом ревидирања графа попут кориговања назива повезника између чворова.

6. ЗАКЉУЧАК

Унапређење пројекта завршено је успешно, систем за аквизицију реализован као *web crawler* замењен је *ETL* процесом који омогућава да подаци буду преузети директно из изворне *MusiscBrainz* базе података, при чему је графовска база података проширена са новим лавеллама, везама и великом количином чворова које садржи.

Систем за аквизицију података имплементиран као *web crawler* представљао је добру почетну тачку при развоју концепта ове апликације. Овакав приступ омогућава имплементацију једноставног система који прикупља само основне информације и чини почетни сет податка. Он такође пружа могућност да се креира потпуно аутоматизован систем, који би прикупљао податке и инкрементално их додавао у базу података у сваком пролазу, који је могуће периодично пуштати или према заказаном распореду, и да при томе води рачуна о интегритету података. Ипак имплементација оваког система доста зависи и од циљаног извора који се претражује и парсира, односно његове комплексности, али и учесталости и типа измена које се дешавају над њим. Ови аспекти могу драстично да повећају комплексност приликом имплементације и одржавања *web crawler*-а и парсирања информација од значаја. Уколико се неки делови или комплетно све *html* странице динамички генеришу, парсирање података са таквих страница може да постане практично немогуће или јако тешко изводљиво.

Када је реч о реализацији *ETL* процеса приликом имплементације овог пројекта у почетку је потребно доста труда док се конфигуришу сви изабрани алати, базе података, виртуелне машине, односно сав потребан софтвер. Након овог корака сам процес је доста једноставан, пружа већу контролу над подацима али може да буде временски захтеван. Уколико би циљ био само да се подаци пребаце из изворне *MusiscBrainz* базе у одредишну графовски оријентисану базу, у овом случају цео први *ETL* процес је могао бити изостављен, уз директне измене на изворној бази и покретање *SQL to Graph ETL* процеса. Ипак овакав приступ би ограничио евентуално проширивање и могућност интеграције са другим базама података, па је из тог разлога је коришћено релационо мапирање и *ODI*. Такође не би било могуће итеративно ажурирање базе података које *ODI* пружа као могућност приликом креирања мапирања. Поред тога локална копија изворне базе података била би измењена и тиме би било онемогућено конфигурисати репликацију између локалне копије базе и *MusiscBrainz* сервера. Ова функционалност и тренутна архитектура целог система омогућава и потпуну аутоматизацију овог процеса. Имплементација ове функционалности била би

одлично унапређење овог система. Реализација би захтевала конфигурисање репликације, екстерне окидаче или скрипт за покретање мапирања која чине први *ETL* процес у *ODI* алату, коришћење *Neo4j ETL CLI* алата за аутоматизацију другог *ETL* процеса и скрипт који дефинише кораке за овај процес. На овај начин било би могуће преузимати све измене и нове податке унете у *MusiscBrainz* базу потпуно аутоматски. Још једно од могућих унапређења било би интегрисање више различитих извора података. На овај начин било би могуће још више проширити скуп података са додатним информацијама. Поред овога, како би се даље проширио скуп података, могуће је имплементирати и засебан систем који корисницима дозвољава да уносе нове податке, који нису прикупљени интеграцијом из других извора података. Имајући у виду све до сада наведено, у ситуацији када је цела база података доступна, нема пуно смисла користити *web crawler*, *ETL* приступ је много боља опција за аквизицију податка у сваком аспект.

7. ЛИТЕРАТУРА

- [1] Ђурђевић, Д., 2019. *Развој музичке енциклопедије, засноване на графовски оријентисаној бази податка и софтверским агентима*, Нови Сад: Факултет Техничких Наука.
- [2] Simitsis, A. Vassiliadis, P. & Sellis, T., 2005. *Extraction-Transformation-Loading Processes*. [Online] Research Gate Available at: https://www.researchgate.net/publication/239638567_Extracton-Transformation-Loading_Processes [Accessed 29 September 2021].
- [3] Oracle, 2021. *Oracle Data Integrator 12.2.1.4.0*. [Online] Oracle Available at: <https://docs.oracle.com/en/middleware/fusion-middleware/data-integrator/12.2.1.4/> [Accessed 9 August 2021].
- [4] Galici, R. Ordile, L. Marchesi, M. Pinna, A. & Tonelli, R., 2020. *Applying the ETL Process to Blockchain Data. Prospect and Findings*. [Online] MDPI Available at: <https://doi.org/10.3390/info11040204> [Accessed 10 August 2021].
- [5] Santos, R. & Bernardino, J., 2008. *Real-time data warehouse loading methodology*. [Online] Research Gate Available at: https://www.researchgate.net/publication/221524861_Real-time_data_warehouse_loading_methodology [Accessed 20 August 2021].
- [6] Wikipedia, 2021. *Extract, transform, load*. [Online] Available at: https://en.wikipedia.org/wiki/Extract,_transform,_load [Accessed 21 August 2021].
- [7] Neo4j, 2021. *Model: Relational to Graph* [Online] Available at: <https://neo4j.com/developer/relational-to-graph-modeling/> [Accessed 23 August 2021].

Кратка биографија:



Добривоје Ђурђевић рођен је 1990. год. у Београду. Бечелор рад из области Рачунарских наука и информатике одбранио је 2019. године. Исте године уписује мастер студије на Факултету техничких наука.

**ISPITIVANJE KVALITETA ELEKTRIČNE ENERGIJE PO ŽALBAMA POTROŠAČA -
STUDIJA SLUČAJA IZ PREDGRADA SUBOTICE****ELECTRIC POWER QUALITY TESTING UPON CONSUMER COMPLAINTS - A CASE
STUDY FROM THE SUBURBS OF SUBOTICA**

Željka Zolotić, Vladimir Katić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Tema ovog istraživačkog rada bavi se merenjem i analizom kvaliteta električne energije u domaćinstvima na području predgrada Subotice.

Ključne reči: Analiza kvaliteta napona, kvalitet električne energije, nesimetrija napona

Abstract – The topic of this research paper deals with measuring and analyzing quality of electricity in households in the suburbs of Subotica.

Keywords: Voltage quality analysis, quality of electricity, voltage unbalance

1. UVOD

Kvalitet isporučene električne energije, sa aspekta elektrodistribucije, podrazumeva kvalitet isporučenog napona, odnosno kvalitet napona na sabirnici na koju se priključuju potrošači – mesto zajedničkog priključenja (*Point of Common Coupling* – PCC). To znači da se osnovne karakteristike napona, kao što su efektivna vrednost, frekvencija, simetričnost i talasni oblik (sinusoidálnost) drže u zadanim okvirima tolerancije oko nominalnih vrednosti pri ustaljenom stanju elektroenergetskog sistema [1]. Ove vrednosti i granice tolerancije definisane su odgovarajućim nacionalnim ili međunarodnim standardima, u ovom slučaju srpskim standardom SRPS EN 50160 [2].

Na karakteristike napona značajan uticaj imaju sami potrošači, naročito grupa nelinearnih potrošača (elektronski potrošači, kao što su TV, računari i računarska oprema, audio uređaji i sl.), zatim velike varijacije opterećenja na perifernim delovima, komutacije, atmosferski prenaponi, kvarovi u sistemu, rad sistema na granicama mogućnosti (naponske redukcije, “slaba” mreža i sl.), te drugi faktori [1].

Problem kvaliteta električne energije u Srbiji je stalno aktuelan. Veoma pomno se prate i koriguju varijacije osnovnih parametara, koje su ipak neki put veće od dozvoljenih. Odgovarajuće studije i snimanja stanja traju već više od 25 godina, s tim da se poklanja dosta pažnje kvalitetu napona, odnosno pojavi viših harmonika i propada napona [3, 4]. Mogu se definisati dve grupe činilaca, koje određuju kvalitet korišćenog napona: osnovni parametri i deformacije talasnih oblika:

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Vladimir Katić, red. prof.

- varijacija efektivne vrednosti napona, frekvencije i pojava nesimetrije u mreži, i
- deformacije tranzijentne prirode, deformacije zbog prelaznih režima i deformacije u ustaljenom stanju.
U slučaju odstupanja od propisanih vrednosti javljaju se negativni efekti kod potrošača, odnosno nepravilnosti u radu njegovih električnih uređaja. Ove nepravilnosti uočavaju se na bazi prijave samih potrošača ili tokom redovnih merenja u elektrodistributivnoj mreži.

Cilj ovog rada je da razmotri stanje kvaliteta napona na bazi žalbe potrošača lociranog u predgrađu Subotice.

**2. POJMOVI I DEFINICIJE PARAMETARA
KVALITETA**

Kvalitet isporučene električne energije se definiše nizom parametara. Oni su podeljeni u osnovne parametre i deformacije talasnih oblika, koje se mogu javiti tokom tranzijentnih pojava, u prelaznim režimima ili u ustaljenom stanju [1]. U ovom poglavlju oni će biti ukratko detaljnije opisane.

2.1. Osnovni parametri napona

U osnovne parametre napona, odnosno prvu grupu činilaca kvaliteta spadaju:

- varijacija efektivne vrednosti napona,
- varijacija frekvencije i
- pojava nesimetrije u mreži.

Ovi parametri definišu pravilan rad sistema i njihova odstupanja su definisana standardima. Kvalitet je ugrožen, ako varijacije premaše dozvoljene limite [1].

2.2. Deformacije talasnog oblika napona

Druga grupa činilaca kvaliteta podrazumeva parametre, koji utiču na sinusoidalni oblik napona na sabirnicama i varijacije njegove efektivne vrednosti, kao i na talasni oblik struje, koju potrošači konzumiraju. Oni se mogu podeliti na tri podgrupe na osnovu dužine perioda nastajanja, odnosno trajanja:

- deformacije tokom tranzijenata,
- deformacije u prelaznim stanjima sistema (kratkog i dugog trajanja) i
- deformacije u ustaljenom stanju sistema.

Ova grupa činilaca direktno definiše kvalitet isporučene električne energije, odnosno kvalitet napajanja. Njoj se poslednjih godina poklanja sve veća pažnja, jer je uočen niz negativnih efekata, pre svega kod modernih računara i računarske opreme, sofisticiranih mikro-elektronskih uređaja, ali i standardne električne opreme, te drugih tkzv. osetljivih potrošača. Naročito predstavljaju problem

kratkotrajne varijacije napona (propadi i poskoci napona), zatim viši harmonici, fliker i slične deformacije [1].

2.3. Tranzijenti

Tranzijent označava pojavu ili veličinu koja se menja između dva uzastopna ustaljena stanja tokom vremenskog intervala, koji je kratak u poređenju sa razmatranim vremenskim opsegom. Tranzijent može biti jednosmerni impuls bilo kog polariteta ili prigušen oscilatorni talas sa prvim maksimumom ma kog polariteta. Tako se tranzijenti mogu podeliti u dve kategorije, impulsne i oscilatorne. Ovi termini odražavaju talasni oblik tranzijenta struje ili napona [1].

2.4. Prelazni režimi rada

U prelazne režime spadaju propadi, premašenja, beznaponske pauze, podnaponi, prenaponi i beznaponske pauze duže od 1 minuta. Prelazni režimi sistema se mogu, shodno trajanju, podeliti na tri kategorije:

- trenutni (0,5-30perioda),
- prolazni (30perioda-3s),
- privremeni (3s-1min).

Imajući u vidu ovu podelu, propadi, premašenja i beznaponske pauze takođe mogu biti trenutni, prolazni ili privremeni, a u skladu sa svojom dužinom trajanja [1].

2.5. Ustaljeno stanje sistema

Deformacije u ustaljenom stanju obuhvataju DC offset (jednosmernu komponentu), više harmonike, interharmonike, ureze (rupe), fluktuaciju napona i šum [1].

3. STANDARDI I PREPORUKE (SRPS EN 50160)

Standard SRPS EN 50160 daje definicije i opisuje bitne osobine distributivnog napona na mestu predaje potrošaču u javnim niskonaponskim (NN) i srednjenaponskim (SN) mrežama pri normalnim pogonskim uslovima. Cilj standarda je da se utvrde karakteristike distributivnog napona s obzirom na frekvenciju, veličinu, talasni oblik i simetriju trofaznog napona [2].

Pri normalnim radnim uslovima, izuzimajući situacije koje nastaju usled kvara ili prekida u napajanju, tokom svakog perioda jedne nedelje 95% desetominutnih srednjih efektivnih vrednosti napona napajanja moraju biti u opsegu od $U_n \pm 10\%$, odnosno 100% desetominutnih srednjih efektivnih vrednosti napona napajanja moraju biti u opsegu $U_n + 10\% / -15\%$.

4. UTICAJ POTROŠAČA U DOMAĆINSTVU NA KVALITET ELEKTRIČNE ENERGIJE

U domaćinstvima, kao jednoj od najrasprostranjenijih kategorija potrošača u elektrodistributivnoj mreži, električni uređaji se mogu podeliti u 3 grupe:

- Rezistivni (R-tip), kod kojih dominira otporna karakteristika (grejalice, štednjaci, električni bojleri, sijalice sa užarenim vlaknom i sl.)
- Rezistivno-induktivni (RL-tip), koji se karakterišu radom električnih motora ili drugih uređaja sa otporno-induktivnom karakteristikom (mašine za pranje veša i sudova, frižideri i klima uređaji, fluo rasveta i sl.)
- Kapacitivni ili nelinearni (C-tip), kod kojih dominira uticaj kapacitivnosti, ali i njihova nelinearna

karakteristika (elektronski uređaji sa diodnim ispravljačem naponskog tipa)

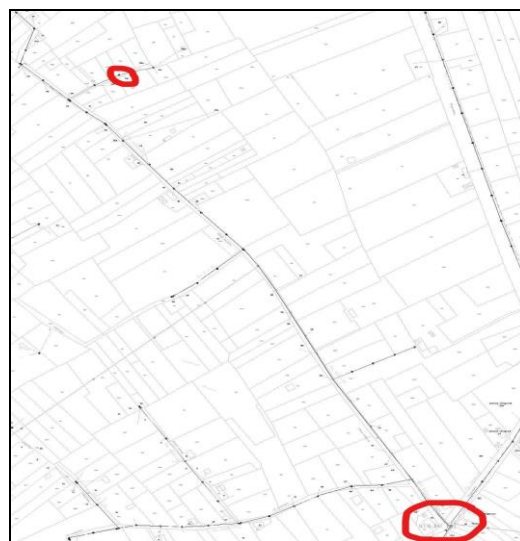
Prva grupa potrošača se karakteriše intenzivnim strujama i kao takva izaziva najznačajnije padove napona na radikalnoj mreži, posebno na dugim vodovima. To je naročito izraženo na perifernim delovima distributivne mreže, u slučajevima „slabe mreže“, kao i u ruralnim mrežama. Rezultat je nizak nivo (efektivna vrednost) napona na potrošačima usled kojeg oni ne rade pravilno, preopterećuju se ili dolazi do prekida u radu.

Na kvalitet mrežnog napona najznačajniji uticaj imaju druga i treća vrsta. Kod RL potrošača, nelinearna karakteristika jezgra (gvožđa) električne mašine ili prigušnice fluo-rasvete izazivaju deformacije struje. Sličan efekat ima i rad diodnog naponskog ispravljača sa C filterom u trećoj grupi potrošača. Njihova upotreba je posebno raširena kod elektronskih, mikroelektronskih i računarskih uređaja i druge opreme malih snaga ($I_n < 16A$). Pomenuti nelinearni potrošači (računari, štampači, fluorescentne svetiljke, TV, video i audio uređaji) su izvori viših harmonika, ali istovremeno i uređaji na koje izobličenja u talasnom obliku napona napajanja mogu imati izrazito negativan uticaj [5].

5. ELEKTRO MREŽA U PREDGRADU SUBOTICE

Elektrodistributivna mreža ED Subotice pokriva potrošače i gradskoj zoni Subotice, ali i u predgrađu i okolnim selima. Ima 687 trafo stanica 20/0,4kV, 290 km nadzemnih vodova i kablova na 20 kV naponskom nivou, kao i 296 km na 0,4 kV naponskom nivou. Karakterišu je dobra stabilnost i sigurno napajanje potrošača, kao i kvalitetno održavanje.

Za potrošače u predgrađu, elektrodistributivna mreža je uglavnom radikalne, zvezdaste konfiguracije. Primer mreže u predgrađu Subotice, odnosno naselju Makova sedmica dat je na slici 1. Vidi se da od glavne transformatorske stanice (TS) TS 187 (zaokružena u dnu slike) vodovi idu zvezdasto sa dugim trasama.



Sl.1. Distributivna mreža u naselju Makova sedmica

6. POSTUPAK MERENJA I MERNO MESTO

Potrošači u naselju Makova sedmica u predgrađu Subotice uputili su žalbu ED Subotice na slab kvalitet

napona, odnosno na njegove niske efektivne vrednosti. Na osnovu toga pokrenut je postupak merenja kvaliteta električne energije. Cilj je bio proveriti stanje naponskih prilika kod udaljenih potrošača, otkrivanje uzroka i posledica lošeg kvaliteta električne energije, i predlaganje mera za rešenje problema i unapređenje snabdevanja.

6.1. Kratak opis mernih mesta

Na slici 2 pokazane su lokacije dva domaćinstva, kod kojih je meren kvalitet električne energije, kao i pozicija TS 187. Potrošači su u ulicama Jelene Četković 41b i Isidore Sekulić 36. Obe lokacije se napajaju iz iste TS 187, samo iz različitih NN izvoda, ali obe imaju probleme sa kvalitetom napajanja i startovanjem kućnih aparata. Crvenom bojom su označene obe lokacije i sama TS, pa se može uočiti da se konkretno obe lokacije nalaze prilično udaljeno od TS, što može da bude jedan od razloga zbog kojeg je narušen kvalitet napona. Zbog ograničenog prostora, ovde će biti prikazani samo rezultati provere kod potrošača u ulici Jelene Četković 41b.



Sl. 2. Pozicija TS 187 i lokacija mernih mesta 1 i 2

6.2. Lokacija mernog mesta 1

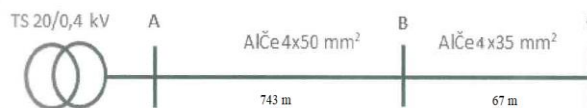
Potrošač u ulici Jelene Četković br.41b, koji se žalio na kvalitet napona, napaja se iz TS 187, čiji naponski odnos je 20/0,4 kV, sa NN izvoda „Jelena Četković“. Na tom izvodu nalazi se 34 potrošača, od kojih 16 potrošača ima trofazni priključak, među kojima i potrošač sa broja 36. Preostalih 18 potrošača ima samo monofazni priključak.

Pošto se očekuje da je uzrok problema nedozvoljeni pad napona prvi korak je da se razmotre preseki i tip provodnika duž NN voda, broj potrošača i mesta priključenja istih. Takođe, potrebno je izmeriti rastojanje od TS do svakog potrošača. Sa ovim ulaznim podacima može se izvršiti proračun tokova snaga i naponskih prilika.

Na slici 3 prikazan je tip provodnika i debljina kabla, kao i tačna dužina trase do samog mernog mesta. Potrebno je naglasiti da je za priključak od stuba NN linije (ABS 9m) do krovnog nosača na kući korišćen kabl tipa Al/Če 4x16 mm².

Za merenje je korišćen profesionalni merni uređaj „Analizator kvaliteta električne energije“ oznake CA 8435, čiji širi opis je dat u sledećem poglavlju. Za mesto priključenja uređaja izabrana je garaža objekta, jer je u njoj izvedena trofazna utičnica. Merenje struje za

konkretno ovaj slučaj nije bilo potrebno, pa shodno tome nisu povezivana strujna klešta. Napajanje mernog uređaja izvršeno je sa posebnog priključka u garaži.



Sl. 3. - Konfiguracija NN mreže za proračun naponskih prilika

6.3. Zadatak merenja

Potrošač se žalio na slab napon i nemogućnost funkcionisanja osnovnih uređaja, kao što su frižider, veš mašina, računar, tako da zadato da se izvrši merenje naponskih prilika. Pored efektivne vrednosti napona, merenje je obuhvatilo fliker i ukupno harmonijsko izobličenje.

7. MERNI UREĐAJI I RUKOVANJE

Merni uređaj CA 8435 je profesionalni trofazni analizator kvaliteta električne energije, koji je kompaktan, otporan na udarce i pogodan za merenje na terenu [6]. Koristi se prilikom merenja kvaliteta električne energije na zahtev potrošača, ako se oni žale na kvalitet energije ili ukoliko se uočavaju neke smetnje ili dešavaju kvarovi. Ergonomski dizajn ovih uređaja i jednostavnost njihovog korisničkog interfejsa, čine njihovu upotrebu prijatnom i intuitivnom. Oni ne samo da omogućavaju korisniku da dobije trenutnu sliku o osnovnim karakteristikama mreže, već i da prate njihove varijacije tokom određenog vremenskog perioda (merenje traje bar 7 dana).

7.1. Rukovanje sa CA 8435

Pre priključenja sondi i kablova na merne tačke, merni uređaj mora biti ispravno podešen. Mod za podešavanje uređaja se bira tipkom „KONFIGURACIJA“. Pritiskom na tu tipku, na ekranu se pojavljuju parametri za podešavanje [6]:

- datum i vreme (*date/time*);
- način proračuna (*calculation parameters*);
- način priključenja (*connection*);
- tip strujnog senzora (*current sensor*);
- odabir mernih vrednosti za zapis (*recording*) i
- nominalna frekvencija (*rated frequency*).

7.2. Način priključenja

Postoje četiri načina priključenja uređaja na merno mesto, u zavisnosti od potrebne merne veličine i načina priključenja i to: monofazno, dvofazno, trofazno trožično ili trofazno četvorožično [6]. U konkretnom slučaju, merenje je vršeno četvorožično.

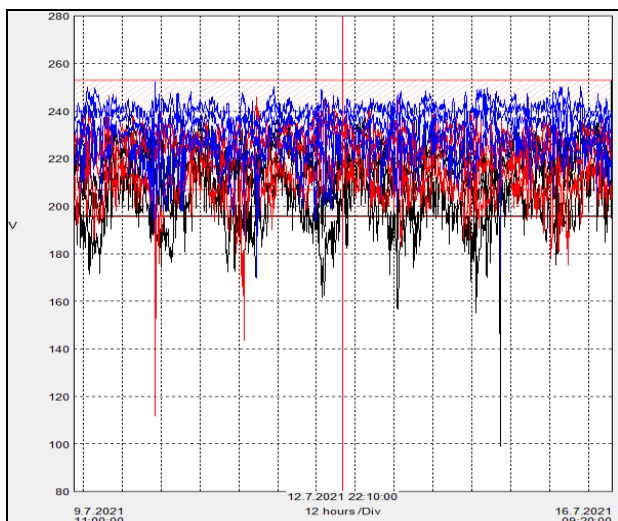
8. ANALIZA REZULTATA MERENJA

Merni uređaj CA 8435 je postavljen od strane stručnog tima na mernom mestu dana 09.07.2021. u 11 časova, a merenje je trajalo 7 dana, tj. do 16.07.2021., a u skladu sa SRPS EN 50160 [2]. U ovom periodu mereni su i zabeleženi sledeći parametri:

- varijacije napona,
- brze promene napona,
- dugi i kratki prekidi napona,
- propadi napona i poskoci napona,
- talasni oblik napona sve tri faze,

- pojava flikera P_{lt} ,
- ukupno harmonijsko izobličenje napona THDU,
- neravnoteža odnosno nesimetrija napona U_{unb} .

Na slici 4 dat je grafički prikaz snimljenog napona u sve tri faze. Označena je gornja i donja granica prema standardu. Po standardu SRPS EN 50160 nominalni fazni napon iznosi 230V sa dozvoljenim odstupanjem $\pm 10\%$ nominalne vrednosti. U tabeli 1 je dat skraćeni prikaz svih merenih rezultata sa oznakom da li je vrednost unutar dozvoljenih granica (*Test passed*) ili ima odstupanja (*Test failed*). U tabeli 2 je izveštaj o propadima napona.



Sl. 4. Grafički prikaz varijacije faznih napona sa označenim granicama

Rezultati merenja ukazuju na značajna odstupanja od graničnih vrednosti. Vide se problemi u niskoj vrednosti napona, čija 95% vrednost je u jednoj fazi (L1) bila 83,5% nominalnog napona ili 192,5 V (minimalna vrednost je 207 V). Uočene su i pojave propada napona (Tabela 2), kao i odstupanja flikera (P_{lt} , Tabela 1). Očigledno je da su žalbe potrošača bile umerne.

Tabela 1: Izveštaj sa merenja sa indikacijom odstupanja

Parameter	Phase	Nominal	Threshold Low %	Threshold Low	Threshold High %	Threshold High	Aggregat...	% of aggregation periods needed	Test results	% of aggregation periods within limits
V rms	L1	230,0 V	-15,0%	195,5 V	+10,0%	253 V	10 min	100,0	Test failed	97,8
V rms	L2	230,0 V	-15,0%	195,5 V	+10,0%	253 V	10 min	100,0	Test passed	100,0
V rms	L3	230,0 V	-15,0%	195,5 V	+10,0%	253 V	10 min	100,0	Test passed	100,0
V rms	L1	230,0 V	-10,0%	207 V	+10,0%	253 V	10 min	95,0	Test failed	83,5
V rms	L2	230,0 V	-10,0%	207 V	+10,0%	253 V	10 min	95,0	Test passed	99,1
V rms	L3	230,0 V	-10,0%	207 V	+10,0%	253 V	10 min	95,0	Test passed	100,0
U rms	L1	398,4 V	-15,0%	338,6 V	+10,0%	438,2 V	10 min	100,0	Test passed	100,0
U rms	L2	398,4 V	-15,0%	338,6 V	+10,0%	438,2 V	10 min	100,0	Test passed	100,0
U rms	L3	398,4 V	-15,0%	338,6 V	+10,0%	438,2 V	10 min	100,0	Test passed	100,0
U rms	L1	398,4 V	-10,0%	358,5 V	+10,0%	438,2 V	10 min	95,0	Test passed	99,9
U rms	L2	398,4 V	-10,0%	358,5 V	+10,0%	438,2 V	10 min	95,0	Test passed	100,0
U rms	L3	398,4 V	-10,0%	358,5 V	+10,0%	438,2 V	10 min	95,0	Test passed	100,0
Pf	L1					1	2 h	95,0	Test failed	0,0
Pf	L2					1	2 h	95,0	Test failed	2,4
Pf	L3					1	2 h	95,0	Test failed	20,7
U THDF	L1		8 %	10 min	95,0				Test passed	100,0
U THDF	L2		8 %	10 min	95,0				Test passed	100,0
U THDF	L3		8 %	10 min	95,0				Test passed	100,0
U unbal (u2)			2 %	10 min	95,0				Test failed	71,3
V THDF	L1		8 %	10 min	95,0				Test passed	100,0
V THDF	L2		8 %	10 min	95,0				Test passed	100,0
V THDF	L3		8 %	10 min	95,0				Test passed	100,0

9. ZAKLJUČAK

Rezultati provere stanja kvaliteta električne energije po žalbi potrošača pokazali su da su primedbe bile opravdane. Uočeno je niz vremenskih perioda sa efektivnim vrednostima napona nižim od propisane, pojave propada napona i nedozvoljenog flikera. Uzrok tim pojavama je nađen u brzom širenju elektrodistributivne mreže, koja je izgrađena u prošlom veku, odnosno veliki

broj potrošača na TS, koja nije bila predviđena za napajanje tolikog broja. Kao mere za otklanjanje ovog problema predviđena je izgradnja nove TS 20/0,4 kV/kV sa minimalno 3 NN izvoda, čime bi se, prema analizi stručnog tima inženjera ED Subotica značajno poboljšale naponske prilike u domaćinstvima u tom području.

Tabela 2: Izveštaj o izmerenim propadima napona

Date Started	Time Started	Duration	MAX ΔV % of nominal ΔV/s
9.7.2021	11:30:00	12 h 50 min 1 s	+34,2%
10.7.2021	00:40:00	2 h 20 min 1 s	+19,3%
10.7.2021	03:20:00	20 h 10 min 1 s	+61,0%
11.7.2021	01:40:00	50 min 1 s	+18,1%
11.7.2021	03:10:00	45 h 30 min 1 s	+46,5%
13.7.2021	01:00:00	2 h 10 min 1 s	+18,3%
13.7.2021	03:40:00	25 h 40 min 1 s	+40,0%
14.7.2021	05:40:00	20 h 1 s	+65,7%
15.7.2021	02:10:00	20 min 1 s	+17,1%
15.7.2021	03:20:00	21 h 40 min 1 s	+32,7%
16.7.2021	01:20:00	1 h 20 min 1 s	+19,1%
16.7.2021	03:10:00	40 min 1 s	+14,7%
16.7.2021	04:10:00	5 h 10 min 1 s	+27,1%
11.7.2021	01:10:00	10 min 1 s	+13,8%
15.7.2021	02:50:00	10 min 1 s	+20,4%
11.7.2021	00:40:00	1 s	+18,3%

10. LITERATURA

- [1] V. Katić, A. Tokić, T. Konjić, „Kvalitet električne energije“, TEMPUS-CEFES, FTN, Novi Sad, 2007.
- [2] SRPS EN 50160:2012, „Karakteristike napona isporučene električne energije iz javnih električnih mreža“, Institut za standardaciju Srbije, Beograd, 2012.
- [3] V. Katic: "Energetski elektronski pretvarači kao izvori viših harmonika u distributivnoj mreži – I deo", Studija za SOUR Elektrovojvodina, Novi Sad, mart 1990.
- [4] Studija: "Kvalitet električne energije u elektro-distributivnim mrežama", (Nosilac: V. Katić), Studija za JP Elektroprivreda Srbije, Beograd, 1997-2000.
- [5] V. Katić, B. Dumnić, S. Mujović, J. Radović: "Effects of Low Power Electronics & Computer Equipment on Power Quality at Distribution Grid – Measurements and Forecast", IEEE – Int. Conf. on Industrial Technology – ICIT'04, Hammamet (Tunis), Dec. 8-10, 2004, Vol. 2, pp. 585-589.
- [6] Chauvin Arnoux CA 8345 Original User Manual.

Kratka biografija:



Željka Zolotić rođena je u Subotici 1992. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Distribuirani elektroenergetski resursi odbranila je 2021.god.

kontakt: zeljka.zolotic92@gmail.com



Vladimir Katić rođen je u Novom Sadu 1954. Doktorirao je na Univerzitetu u Beogradu 1991. god., a od 2002 je u zvanju redovnog profesora Univerziteta u Novom Sadu. Oblasni interesovanja su energetska elektronika, obnovljivi izvori električne energije, kvalitet električne energije i električna vozila.

TRANSFORMACIJE STABALA APSTRAKTNE SINTAKSE U PROGRAMSKI KOD**TRANSFORMATIONS OF ABSTRACT SYNTAX TREES INTO SOURCE CODE**Andrej Jokić, *Fakultet tehničkih nauka, Novi Sad***Oblast – RAČUNARSTVO I AUTOMATIKA**

Kratak sadržaj – Mnogi napredni alati koji automatizuju izmjene programskog koda, kao što su alati za automatsko refaktorisanje, zasnivaju se na izmjenama stabla apstraktne sintakse. Izmjene izvršene na stablu je zatim potrebno primijeniti na originalni izvorni kod, što se može postići transformacijom stabla nazad u tekstualni oblik. U ovom radu su istražena i implementirana rješenja za implementaciju ovakvih transformacija, uz očuvanje nesemantičkih elemenata originalnog izvornog koda, kao što su komentari i ručno formatiranje.

Cljučne reči: *Stablo apstraktne sintakse, jezici specifični za domen, refaktorisanje*

Abstract – Most of the advanced tools for automated source code modifications, such as automated refactoring tools, are based on modifications of the abstract syntax tree. The modifications which are done on the abstract syntax tree must be applied to the original source code, which can be done by transforming the tree back into a textual representation. In this paper, solutions for such transformations are researched and implemented, focusing on the problems of preserving the non-semantic elements of the original source code, such as comments and manual formatting.

Keywords: *Abstract syntax tree, domain specific languages, refactoring*

1. UVOD

Razvoj računarskih tehnologija napreduje velikom brzinom, pružajući korisnicima sve više softverskih funkcionalnosti. Obim poslova koji se obavljaju na računaru je iz godine u godinu sve veći, a samim tim rastu mogućnosti i kompleksnost softvera. Pri tome, zahtjevi klijenata se veoma često mijenjaju pa je potrebno neprekidno održavati i nadograđivati softver. Uz porast kompleksnosti softvera raste i cijena njegovog razvoja i održavanja. Zbog toga, dosta pažnje se posvećuje istraživanju i razvoju tehnologija koje programerima olakšavaju posao i povećavaju produktivnost.

U savremenom programiranju koristi se mnoštvo alata koji automatizuju mnoge radnje koje programeri često obavljaju.

Jedan primjer ovakvih alata su alati za automatsko refaktorisanje koda. Refaktorisanje podrazumijeva izmjene programskog koda radi poboljšanja njegove

interne strukture, ali bez promjene spoljašnjeg ponašanja koda. Naime, neprestane izmjene i nadogradnje softvera dovode do povećanja kompleksnosti i zamršenosti koda, što ga vremenom čini sve težim za održavanje i proširivanje. Da bi se kod održavao kvalitetnim i čistim, potrebno je neprestano refaktorisati djelove koda koji mogu biti implementirani na jednostavniji i jasniji način [1].

Za razvoj preciznih alata za refaktorisanje neophodno je da alati vrše dublju analizu izvornog koda i da izmjene vrše uzimajući u obzir semantiku elemenata koda. Zbog toga, mnogi napredni alati za refaktorisanje ne vrše izmjene direktno nad tekstualnim oblikom koda, već za to koriste neku vrstu apstraktne reprezentacije koda. Stablo apstraktne sintakse (eng. Abstract Syntax Tree, skraćeno AST) je konačno usmjereno stablo koje sadrži samo suštinu semantike programa i dobija se iz izvornog koda postupkom koji se zove parsiranje ili sintaksa analiza. Stablo apstraktne sintakse je veoma jednostavno za analizu od strane računarskih programa i omogućava kreiranje veoma preciznih alata za automatske izmjene izvornog koda.

Nakon izvršenih izmjena na stablu apstraktne sintakse, potrebno je date izmjene primijeniti i na originalni izvorni kod. Ovo se može uraditi postupkom koji je suprotan od parsiranja, tj. transformisanjem stabla apstraktne sintakse nazad u tekstualni oblik. Međutim, kako stablo apstraktne sintakse sadrži samo suštinu značenja programa, mnogi elementi koda koji nemaju semantičkog značaja se gube tokom parsiranja, pa stablo ne sadrži sve informacije koje su potrebne za rekonstrukciju originalnog izvornog koda. Zbog toga, implementacija ovakvih transformacija nije jednostavna i zahtijeva pokrivanje mnogih graničnih slučajeva kako bi se dobio kvalitetan rezultat [2].

Cilj ovog rada jeste implementacija rješenja za transformaciju stabala apstraktne sintakse u programski kod u okviru alata za razvoj jezika specifičnih za domen TextX. U radu će biti istraženi problemi koji se susrijeću prilikom implementacije ovakvog postupka, kao i moguća rješenja datih problema. Na kraju će biti predstavljena implementacija rješenja i prikaz dobijenih rezultata.

2. PROBLEMI**2.1. Ključne riječi**

Prvi problem na koji se nailazi pri generisanju teksta iz stabla apstraktne sintakse je nedostatak nekih elemenata izvornog koda. Zgrade, znaci interpunkcije i ključne riječi ne doprinose semantici programa već služe za prepoznavanje različitih struktura u kodu prilikom parsiranja kao i za bolju čitljivost koda. Prema tome, ovi

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Igor Dejanović, vanr. prof.

elementi su neophodni za generisanje ispravnog koda i potrebno je pronaći način da budu sačuvani.

Jedno rješenje ovog problema podrazumijeva čuvanje datih elemenata u samom stablu apstraktne sintakse. Za ovo je potrebno modifikovati parser tako da ove elemente takođe uključi u stablo. Kako ne bi bila narušena struktura stabla, i kako bi ostali programi koji koriste ovo stablo nastavili da rade kako je očekivano, dodatne podatke je poželjno sačuvati kao atribut postojećih čvorova stabla. Kada su svi neophodni elementi sačuvani na ovaj način, generisanje izvornog koda se svodi na prolazak kroz svaki čvor stabla redom i štampanje sačuvanih podataka [2].

Alternativni pristup se ne oslanja na čuvanje podataka u stablu, već na generisanje nedostajućih podataka prilikom transformacije stabla u tekst. Naime, ukoliko je poznat oblik različitih struktura u jeziku, na osnovu podataka iz stabla apstraktne sintakse mogu se generisati elementi koji su neophodni da bi rezultujući kod odgovarao sintaksi jezika. Prednost ovog pristupa je što nije potrebno modifikovati postojeći parser i dodavati podatke u stablo, ali je sa druge strane potrebno pisati logiku za štampanje svakog elementa jezika [3].

2.2. Formatiranje

Formatiranje se odnosi na vizuelnu strukturu i izgled koda. Dok formatiranje nije neophodno za generisanje ispravnog koda, ono je veoma značajno za preglednost i čitljivost. Vizuelnu strukturu koda najčešće definišu prazni karakteri - razmaci, prelomi redova i prazni redovi. Slično kao kod ključnih riječi, prazni karakteri se mogu dobiti na dva osnovna načina, čuvanjem dodatnih atributa u stablu ili generisanjem novog formatiranja. Međutim, u ovom slučaju je potrebno obratiti pažnju na još neke slučajeve.

Ukoliko se prazni karakteri čuvaju u stablu, bitno je obratiti pažnju kom čvoru se dodjeljuju prazni karakteri, u suprotnom može doći do toga da refaktorisanjem elemenata stabla prazni karakteri završe na pogrešnom mjestu. Formatiranje nije formalno definisano u jeziku već programer ima slobodu da ga koristi kako želi, pa je veoma teško definisati pravila za tretiranje formatiranja, i potrebno je uvoditi određene pretpostavke [4].

Sa druge strane, generisanje formatiranja, poznato pod nazivom *pretty-printing*, takođe može proizvesti dobre rezultate. Međutim, ponovno generisanje formatiranja predstavlja problem ukoliko stil generisanog koda odstupa od stila kojim je formatiran originalni kod, što je gotovo uvijek slučaj osim ako je programer od samog početka koristio isti alat za automatsko formatiranje koda. Za alate kao što su alati za refaktorisanje, poželjno je da rezultujući kod bude u potpunosti konzistentan sa originalnim, pa je poželjno koristiti pristupe koji u što većoj mjeri čuvaju formatiranje originalnog koda.

2.3. Komentari

Komentari u kodu su veoma slični formatiranju, u tome što uglavnom nemaju formalnu strukturu i programer ima slobodu da ih koristi kako želi. Kao kod formatiranja, neophodno je obraćati pažnju na povezivanje komentara sa odgovarajućim elementima koda. Međutim, problem povezivanja komentara sa kodom je mnogo ozbiljniji u

odnosu na formatiranje. Dok pogrešno formatiranje može izazvati vizuelne nekonzistentnosti, komentari koji se nalaze na pogrešnoj lokaciji mogu u potpunosti izgubiti značenje, ili čak dobiti potpuno drugo značenje u zavisnosti od konteksta.

Određivanje veza između komentara i čvorova stabla se najčešće vrši upotrebom raznih heuristika, tj. jednostavnih pravila koja u praksi uglavnom daju dobre rezultate. Na primjer, može se pretpostaviti da se komentari koji se nalaze samostalno u jednoj liniji odnose na liniju koda nakon sebe, dok se komentari u istoj liniji sa kodom odnose na kod koji im prethodi. Ovo jednostavno pravilo će u većini slučajeva dati dobre rezultate, ali to nije zagarantovano jer programeri imaju potpunu slobodu da koriste komentare kako žele. Za precizne rezultate potrebna je mnogo dublja analiza, uključujući vizuelni izgled koda, položaj komentara u odnosu na kod i druge komentare, pa čak i sadržaj samih komentara i koda koji se nalazi u blizini. Čak i sa naprednom analizom, bez prepoznavanja značenja komentara nije moguće dobiti rezultate koji su u potpunosti ispravni [5].

2.4. Novogenerisani AST čvorovi

Čuvanje originalnog formatiranja koda je veoma bitno kod alata koji proizvode kod namijenjen za dalji razvoj. Međutim, Ovo nije uvijek moguće uraditi. Naime, prilikom refaktorisanja koda nije rijedak slučaj da se kreiraju novi elementi koda koji nisu postojali u originalnom kodu. Na primjer, kod *Extract Method* refaktorisanja, potrebno je kreirati novu funkciju, kopirati dio koda u nju, i zatim kreirati naredbu koja poziva datu funkciju na mjestu gdje se prethodno nalazio dati kod. U ovakvim slučajevima poželjno je kombinovati tehnike čuvanja originalnog formatiranja sa *pretty printing*-om tj. generisanjem novog formatiranja. Na taj način će svi postojeći elementi zadržati originalno formatiranje, a novi čvorovi će imati novo formatiranje. Napredniji oblik ovog pristupa podrazumijeva prepoznavanje stila originalnog formatiranja i primjena datog stila i na nove čvorove kako bi bili konzistentni sa ostatkom koda.

3. IMPLEMENTACIJA

U nastavku će biti predstavljena praktična implementacija koncepata koji su istraženi u radu. Rješenje će biti implementirano kao proširenje alata za razvoj jezika specifičnih za domen TextX i testirano koristeći primjere programa napisanih u jednom novom jeziku, kao i implementacijom jednostavnog alata za refaktorisanje koji se oslanja na dato rješenje. Sve izvršene modifikacije, kao i primjeri koji su korišćeni za testiranje, dostupni su u repozitorijumu <https://github.com/ajokic1/pprint-textx>.

3.1. TextX

TextX [6] je alat za razvoj jezika specifičnih za domen napisan u Python-u. Dizajniran je za veoma brz i jednostavan razvoj jezika specifičnih za domen. TextX iz jedinstvene definicije gramatike automatski generiše meta-model i parser, koji se zatim mogu koristiti za parsiranje programa i generisanje modela, koji predstavlja stablo apstraktne sintakse programa. Pri tome, TextX funkcioniše kao interpreter, što ga čini veoma dinamičnim i pogodnim za jednostavan razvoj jezika koji su podložni

velikim promjenama, što je veoma značajno u ranim fazama razvoja. TextX je zasnovan na Arpeggio [7] parseru, rekurzivnom silaznom parseru sa vraćanjem i memoizacijom, zasnovanom na PEG gramatikama.

Pri kreiranju novog jezika, TextX prvo parsira gramatiku i na osnovu nje generiše parser i meta-model (apstraktnu reprezentaciju gramatike jezika). Meta-modelu se zatim može proslijediti izvorni kod programa u datom jeziku, pri čemu TextX prvo poziva generisani parser koji kreira stablo parsiranja, a zatim se na osnovu datog stabla konstruiše sami model programa.

3.2. Štampanje

Kako bi se omogućilo pretvaranje modela nazad u izvorni kod, prvo je potrebno obezbijediti da model sadrži sve neophodne elemente koda. Dok stablo parsiranja sadrži sve terminale iz originalnog koda, model sadrži samo one elemente koji su bitni za semantiku koda. Prema tome, ovi elementi se gube prilikom konstruisanja TextX modela, pa je potrebno izvršiti modifikacije u logici za kreiranje modela.

Ovo je postignuto dodavanjem logike koja svaki terminal i neterminal koji se obradi prilikom kreiranja jednog čvora modela čuva u privremenoj promjenljivoj. Pri konačnom kreiranju čvora modela, sadržaj ove promjenljive se čuva kao atribut čvora modela pod nazivom `_tx_pprint_data`.

Nakon dopunjavanja modela na ovaj način, model se može pretvoriti u tekst jednostavnim prolaskom kroz stablo i štampanjem sadržaja pomenutog atributa, ili rekurzivnim pozivanjem funkcije za štampanje nad ostalim čvorovima. Na slici 1. prikazana je osnovna implementacija funkcije za štampanje, koja će kasnije biti proširena

```
def process_node(self, node):
    for n in node._tx_pprint_data:
        n_class = n.__class__.__name__

        if isinstance(n, Terminal):
            self._pprint_terminal(n)
        else:
            self.process_node(n)

    def _pprint_terminal(self, node):
        self.append(' ')
        self.append(node)
```

Slika 1. Osnovna funkcija za štampanje modela

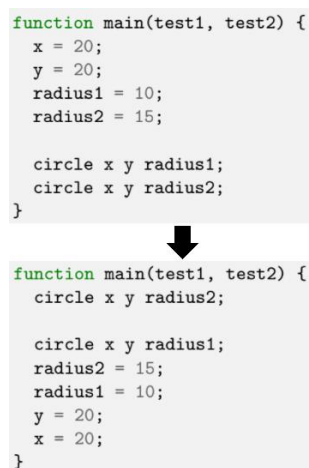
3.3. Čuvanje formatiranja

Rezultat prethodnog proširenja je stablo koje u većini slučajeva može generisati ispravan kod, ali bez bilo kakvog formatiranja - svi terminali će biti odštampani redom u jednoj liniji. Sada je potrebno obezbijediti da pored odbačenih terminala budu sačuvani i prazni karakteri, tj. formatiranje koda. U ovom slučaju, odbacivanje praznih karaktera se vrši na nivou Arpeggio parsera, prilikom konstruisanja samog stabla parsiranja.

Prema tome, potrebno je izvršiti modifikacije u samom parseru. U podrazumijevanoj konfiguraciji, Arpeggio u potpunosti preskače prazne karaktere prilikom parsiranja.

Za čuvanje formatiranja je potrebno obezbijediti da oni ostanu sačuvani i vezani za čvorove stabla. Ovo je moguće uraditi prikupljanjem praznih karaktera u privremenu promjenljivo tokom parsiranja, i njihovim vezivanjem za čvor koji predstavlja naredni terminal. Nakon što su prazni karakteri za svaki terminal sačuvani u vidu atributa čvorova na stablu parsiranja, ovi podaci će biti dostupni u atributu `_tx_pprint_data` iz prethodnog koraka, čime će biti omogućeno štampanje koda sa originalnim formatiranjem.

Međutim, ukoliko se svi prazni karakteri između dva terminala vežu direktno za naredni terminal, ovo može dovesti do neželjenog ponašanja u pojedinim situacijama, jer ovakva veza nije uvijek logički opravdana. Ovo se može uočiti na primjeru na slici 2.



```
function main(test1, test2) {
    x = 20;
    y = 20;
    radius1 = 10;
    radius2 = 15;

    circle x y radius1;
    circle x y radius2;
}

function main(test1, test2) {
    circle x y radius2;

    circle x y radius1;
    radius2 = 15;
    radius1 = 10;
    y = 20;
    x = 20;
}
```

Slika 2. Pogrešno čuvanje formatiranja prilikom refaktorisanja

Nakon zamjene pozicija određenih linija koda, prazna linija koja razdvaja djelove koda ostaje vezana za pogrešan čvor, što dovodi do pogrešnog rezultata. Ovaj problem se može riješiti dodatnom analizom i obradom sačuvanih podataka, tako da se prazne linije tretiraju kao zasebni entiteti i da ne budu vezane direktno za naredni čvor.

3.4. Čuvanje komentara

Komentari se u Arpeggio parseru tretiraju na sličan način kao prazni karakteri. Prema tome, rješenje za čuvanje komentara je relativno slično rješenju za formatiranje - komentari se prikupljaju zajedno sa svojim pozicijama i vezuju se za atribute čvorova u stablu. Međutim, kod komentara je potrebno pokriti neke granične slučajeve kako bi se obezbijedilo čuvanje komentara na odgovarajućim pozicijama u odnosu na kod. Primjer ovakvog slučaja dat je na slici 3. Iz datog primjera može se uočiti da drugi komentar sa tekstom "Second circle" nije vezan za čvor na koji se zapravo odnosi, pa nakon refaktorisanja ostaje na pogrešnoj poziciji. Problem je nastao zato što se svi komentari vezuju za naredni terminal, što u ovom slučaju znači da komentar biva vezan za vitičastu zagradu koja označava završetak funkcije. Ovo se može riješiti uvođenjem pravila koje će komentare koji se nalaze u istoj liniji sa dijelom koda vezati za taj kod. Uz ovu modifikaciju, nakon refaktorisanja komentar ostaje na ispravnoj poziciji.

```
function main(test1, test2) {
  x = 20;
  y = 20;
  radius1 = 10;
  radius2 = 15;

  // Draw two circles with the same center
  circle x y radius1;
  circle x y radius2; // Second circle
}

function main(test1, test2) {
  // Draw two circles with the same center
  circle x y radius1;
  circle x y radius2;
  x = 20;
  y = 20;
  radius1 = 10;
  radius2 = 15;

  // Second circle
}
```

Slika 3. Pogrešno čuvanje komentara prilikom refaktorisanja

3.5. Novogenerisani čvorovi

Kao što je pomenuto, prilikom refaktorisanja može doći i do kreiranja čvorova u stablu koji nisu postojali u originalnom kodu. Za ove čvorove nije moguće sačuvati originalno formatiranje već se ono mora generisati na osnovu određenih pravila. Ovo se može postići dizajniranjem proširive klase za štampanje modela koja će pored štampanja sačuvanog formatiranja omogućiti korisniku da definiše dodatna pravila za štampanje elemenata stabla koji ne posjeduju sačuvano formatiranje. Ovo je implementirano tako da korisnik kreira funkciju za obradu određenog tipa čvora u stablu, a zatim ga dinamički registruje u klasi za štampanje.

Radi demonstracije ove funkcionalnosti, implementiran je jednostavan alat za automatsko vršenje "Extract method" transformacije refaktorisanja, koja podrazumijeva izdvajanje dijela koda u zasebnu funkciju. Pri ovoj transformaciji, generiše se nova funkcija, kao i poziv date funkcije, koji predstavljaju novogenerisane čvorove u stablu i formatiraju se korišćenjem definisanih pravila. Rezultat ove transformacije može se vidjeti na slici 4.

4. ZAKLJUČAK

Dok osnovna implementacija transformacije stabla apstraktne sintakse u programski kod nije pretjerano kompleksna, za dobijanje kvalitetnih rezultata neophodno je pokriti veliki broj graničnih slučajeva. Potrebno je vršiti dublju analizu koda i uvoditi pretpostavke o načinu upotrebe nesemantičkih elemenata u većini programa. Rješenja koja su implementirana kao proširenje alata za razvoj jezika poput TextX-a značajno olakšavaju implementaciju funkcionalnosti koje se oslanjaju na ovakve transformacije kod jezika specifičnih za domen, jer će jezici kreirani u ovom alatu već imati podršku za ovakve transformacije. Pri tome, predstavljeno rješenje se može dodatno proširivati dodatnim pravilima za prepoznavanje veza između čvorova stabla i nesemantičkih elemenata koda, čime se mogu dobiti još bolji rezultati.

```
function main(test1, test2) {
  x = 20;
  y = 20;
  radius1 = 10;
  radius2 = 15;

  print_circles(x, y, radius1, radius2);
}

function print_circles (x, y, radius1, radius2) {
  // Draw two circles with the same center
  // but different radiuses
  circle x y radius1;
  circle x y radius2; // Second circle
}
```

Slika 4. Extract Method refaktorisanje

4. LITERATURA

- [1] Kent Beck, Cynthia Andres, "Extreme Programming Explained: Embrace Change (2nd Edition)", Addison-Wesley Professional, 2004.
- [2] J. Overbey, R. Johnson, "Generating rewritable abstract syntax trees", *SLE*, 2008.
- [3] M. de Jonge, "Pretty-printing for software reengineering", *International Conference on Software Maintenance*, 2002. Proceedings, pp. 550–559, 2002.
- [4] M. de Jonge, E. Visser, "An algorithm for layout preservation in refactoring transformations.", *SLE*, 2011.
- [5] M. Van De Vanter, "Preserving the Documentary Structure of Source Code in Language-Based Transformation Tools", *SCAM*, pp. 133-143, 2011.
- [6] I. Dejanović, R. Vaderna, G. Milosavljević, Ž. Vuković, "TextX: A python tool for domain-specific languages implementation", *Knowledge-Based Systems*, 115:1-4, 2017.
- [7] I. Dejanović, G. Milosavljević, R. Vaderna, "Arpeggio: A flexible PEG parser for Python", *Knowledge-Based Systems*, issn:0950-7051, vol.95, p.71–74, DOI:10.1016/j.knosys.2015.12.004, 2016.

Kratka biografija:



Andrej Jokić rođen je u Kotoru 1996. god. Osnovne studije je završio na Elektrotehničkom fakultetu u Podgorici 2019. godine. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – računarstvo i automatika odbranio je 2021.god.

Kontakt: aj.jokic@gmail.com

PRIMENE DC MODELA TOKOVA SNAGA U PLANIRANJU I EKSPLOATACIJI ELEKTROENERGETSKIH SISTEMA

APPLICATIONS OF DC LOAD FLOW MODEL IN PLANNING AND OPERATIONS OF POWER SYSTEMS

Goran Mitrović, Neven Kovački, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Cilj ovog rada je da prikaže model jednosmernih (DC) tokova snaga, kao i neke od najčešćih primena ovog modela u planiranju i eksploataciji elektroenergetskih sistema (EES). Najpre je prikazan DC model tokova snaga, kao i njegovo izvođenje iz klasičnog modela tokova snaga. Zatim su prikazane primene ovog modela u proračunu faktora osetljivosti prenosnih snaga, analizi sigurnosti, problemu optimalnog odsecanja opterećenja i problemu optimalne preraspodele generatorske proizvodnje. Navedene primene DC modela tokova snaga demonstrirane su na primeru realne prenosne mreže, a rezultati su prikazani i komentarisani u radu.

Ključne reči: jednosmerni (DC) model tokova snaga, faktori osetljivosti prenosnih snaga, analiza sigurnosti, optimalno odsecanje potrošnje, preraspodela generatorske proizvodnje

Abstract – The aim of this paper is to present direct current (DC) load flow model and some of the most common applications of this model in planning and operations of power systems. First, derivation of DC load flow model is presented, based on classical load flow model. Then, the paper presents applications of this model in the calculation of power transfer distribution factors, power system contingency analysis, load shedding and re-scheduling of generation powers. All the presented applications of DC load flow model are demonstrated on the example of real transmission network and the results are presented and discussed in the paper.

Keywords: direct current (DC) load flow model, power transfer distribution factors, contingency analysis, load shedding, generation re-scheduling

1. UVOD

Većina algoritama za proračun tokova snaga u prenosnim mrežama zasnovana je na nelinearnim matematičkim modelima i iterativnom postupku rešavanja (npr. Newton–Raphson-ov iterativni postupak) [1]. Uprkos visokoj tačnosti ovih proračuna, u različitim analizama koje se sprovode u okviru planiranja i eksploatacije EES-a (npr. analiza sigurnosti) potrebno je uraditi veliki broj ovih proračuna, što zahteva značajno računarsko vreme i veliko zauzeće memorije računara [2]. Pored toga, u procesima planiranja i eksploatacije EES-a često se rešavaju različiti optimizacioni problemi.

NAPOMENA:

Ovaj članak proistekao je iz master rada čiji mentor je bio docent dr Neven Kovački.

Ukoliko bi se oni zasnivali na klasičnom (nelinearnom) modelu tokova snaga, dobijeni optimizacioni modeli bi bili bazirani na nelinearnom programiranju i njihovo rešavanje bi bilo veoma složeno. Iz navedenih razloga, u mnogim proračunima u planiranju i eksploataciji EES-a umesto klasičnog modela tokova snaga koriste se razni približni, najčešće raspregnuti modeli tokova snaga, među kojima posebno mesto zauzima linearni model jednosmernih (DC) tokova snaga.

Cilj ovog rada jeste da prikaže neke od najčešćih primena DC modela tokova snaga u planiranju i eksploataciji EES-a. U ovom radu obrađene su primene DC modela tokova snaga u proračunima faktora osetljivosti prenosnih snaga, analizi sigurnosti, kao i u rešavanju problema minimalnog odsecanja opterećenja i problema preraspodele generatorske proizvodnje.

2. DC MODEL TOKOVA SNAGA

U ovoj glavi je prikazano izvođenje DC modela tokova snaga [2]. Polazi se od opšteg izraza za bilans aktivnih snaga u čvorovima EES-a iz klasičnog modela tokova snaga:

$$P_i = P_{gi} - P_{pi} = G_{ii}V_i^2 + V_i \sum_{j \in \alpha_i} V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}), \quad (1)$$

$$i = 1, \dots, N,$$

gde su sa P_{gi} i P_{pi} označene proizvodnja i potrošnja aktivne snage u čvoru (i), sa G_{ij} i B_{ij} su označeni realni i imaginarni delovi elemenata matrice admitansi $\underline{Y}_{bus} = \{\underline{Y}_{ij}\} = \{G_{ij} + jB_{ij}\}$ na poziciji “ ij ”, dok N predstavlja broj čvorova u EES-u. Ako se u izrazima za proračun matrice admitansi $\underline{Y}_{bus}$ zanemare sve otočne admitanse i uvedu sledeće pretpostavke:

1. $G_{ij} \approx 0$;
2. $V_i \approx V_j \approx 1.00$ [r. j.],
3. $a_{ij} = 1$ (a_{ij} je odnos transformacije realnih transformatora),
4. $\cos \theta_{ij} \approx 1.00$; $\sin \theta_{ij} \approx \theta_{ij}$,

jednačina (1) postaje:

$$P_i = \sum_{j \in \alpha_i} B_{ij} \theta_{ij} = -B_{ii} \theta_i - \sum_{j \in \alpha_i} B_{ij} \theta_j, \quad (2)$$

$$i = 1, 2, \dots, N,$$

pri čemu je:

$$B_{ii} = - \sum_{j \in \alpha_i} B_{ij} = \sum_{j \in \alpha_i} B_{ij}^g; \quad B_{ij}^g = - \frac{1}{X_{ij}^g}, \quad (3)$$

gde je B_{ij}^g susceptansa, a X_{ij}^g reaktansa grane ($i-j$).
U matricnoj formi jednačina (2) postaje:

$$\mathbf{P} = -\mathbf{B}'\boldsymbol{\theta}, \quad (4)$$

gde je matrica $\mathbf{B}'$ ($N \times N$)-dimenziona matrica susceptansi nezavisnih čvorova mreže, $\mathbf{P}$ je N -dimenzioni vektor aktivnih injehtiranja, a $\boldsymbol{\theta}$ je N -dimenzioni vektor faznih stavova fazora napona u nezavisnim čvorovima.

Matrica $\mathbf{B}'$ u jednačini injehtiranja (4) je singularna matrica. Međutim, ukoliko se unapred izabere fazni stav jednog čvora (na primer, $i = 1, \theta_1 = 0$), tada se sistem jednačina (4) svodi na $(N-1)$ -dimenzioni sistem jednačina:

$$\mathbf{P}_r = -\mathbf{B}'_r\boldsymbol{\theta}_r, \quad (5)$$

gde je $\mathbf{B}'_r$ nesingularna, kvadratna, redukovana matrica susceptansi, reda $(N-1)$ dobijena odstranjivanjem vrste i kolone iz matrice $\mathbf{B}'$ koje odgovaraju referentnom čvoru (ovde $i = 1$), dok $\mathbf{P}_r$ i $\boldsymbol{\theta}_r$ predstavljaju redom redukovani vektor aktivnih injehtiranja i vektor nepoznatih faznih uglova u nezavisnim čvorovima, dimenzija $(N-1)$.

Linearni model jednosmernih (DC) tokova snaga (4) daje izraz za fazne uglove napona u PQ i PV čvorovima, tj. u svim čvorovima izuzev referentnog čvora ($i = 1$), proračunom vektora:

$$\boldsymbol{\theta}_r = -[\mathbf{B}'_r]^{-1}\mathbf{P}_r; \boldsymbol{\theta} = [0 \ \boldsymbol{\theta}_r]^T. \quad (6)$$

Tokovi snaga po granama EES-a se prema DC modelu tokova snaga računaju na sledeći način:

$$P_{ij} = b_{ij}(\theta_i - \theta_j) = -b_{ij}^g(\theta_i - \theta_j) = \frac{\theta_i - \theta_j}{X_{ij}^g} [\text{r.j.}], i, j = 1, 2, \dots, N \quad (7)$$

gde je b_{ij} odgovarajući element redukovane matrice susceptansi $\mathbf{B}'_r$.

3. PRORAČUN FAKTORA OSETLJIVOSTI PRENOSNIH SNAGA

Faktori osetljivosti prenosnih snaga (*eng. power transfer distribution factors – PTDFs*) definišu se kao odnos malog priraštaja toka aktivne prenosne snage po nekoj grani (l) i priraštaja veličine neke od promenljivih, ili parametara od kojih snaga zavisi (npr. moduli napona (V_i), fazni uglovi napona (θ_i), tokovi snaga po granama itd. Ovi faktori se vrlo efikasno proračunavaju upravo primenom DC modela tokova snaga. Osnovni PTDF faktori osetljivosti su faktori preraspodele za ispade generatora i distribicioni faktori za ispade grana [2].

3.1. Određivanje faktora preraspodele pri ispadu generatora

Ovi faktori definišu se kao:

$$a_l^i = \frac{\Delta P_l}{\Delta P_i} = \frac{1}{X_{ij}^g} \left(\frac{\Delta \theta_M}{\Delta P_i} - \frac{\Delta \theta_K}{\Delta P_i} \right), \quad (8)$$

gde je ΔP_l mala promena toka aktivne snage u grani (l), ΔP_i mala promena aktivnog injehtiranja u čvoru (i), a sa $\Delta \theta_M$ i $\Delta \theta_K$ označene su promene faznih stavova fazora napona u čvorovima koje povezuje grana (l).

Pozitivne vrednosti a_l^i pokazuju da porast snage injehtiranja u čvoru i izaziva porast toka snage u grani (l), dok smanjenje snage u istom čvoru ima suprotan efekat.

3.2. Određivanje distribucionih faktora pri ispadu grana

Ovi faktori definišu se kao:

$$d_l^k = \frac{\Delta P_l}{P_k^0} = \frac{P_l - P_l^0}{P_k^0}, k, l = 1, 2, \dots, L; l \neq k. \quad (9)$$

gde je P_l^0 tok aktivne snage po grani (l) pre ispada grane (k), a P_k^0 je tok aktivne snage po grani (k) pre ispada.

4. ANALIZA SIGURNOSTI EES

Cilj analize sigurnosti EES-a je da se proceni da li je došlo do narušavanja pogonskih ograničenja posle pojave poremećaja u sistemu [2]. Problem analize velikog broja mogućih poremećaja u nekom EES-u zahteva da se svi proračuni sprovedu u što kraćem mogućem roku, čime se omogućuje da, ako je sigurnost sistema ugrožena, operatori preduzmu odgovarajuće akcije. Jedan od postupaka koji omogućava ove brze proračune, u prvom redu one čiji je cilj otkrivanje pojava preopterećenja, jeste proračun faktora osetljivosti prenosnih snaga koji su objašnjeni u prethodnom delu. Pomoću njih, moguće je proračunati nove vrednosti tokova snaga preko sledećih izraza:

$$P_l' = P_l^0 + \Delta P_l = P_l^0 + a_l^i \Delta P_i, l = 1, 2, \dots, L, \quad (10)$$

$$P_l' = P_l^0 + \Delta P_l = P_l^0 + d_l^k P_k^0, l = 1, 2, \dots, L. \quad (11)$$

Proračunatim vrednostima tokova aktivnih snaga na osnovu DC modela tokova snaga može se na brz način proveriti da li u sistemu postoji preopterećenje. Naime, tok snage po svakoj grani može se dalje uporediti sa graničnom snagom te grane, i ako je ta granica prevaziđena, daje se upozorenje operatoru da preduzme akcije kojima će se tokovi snaga vratiti u dozvoljene granice. U tu svrhu najčešće se sprovedu odsecanje dela opterećenja i preraspodela generatorске proizvodnje, koji su opisani u sledećim glavama.

5. PROBLEM MINIMALNOG ODSECANJA OPTEREĆENJA

Problem minimalnog odsecanja opterećenja predstavlja optimizacioni problem kojim se određuje minimalno opterećenje koje je potrebno odseći kako bi bila zadovoljena pogonska ograničenja u EES-u [3]. Ovaj problem je značajan jer se njegovim rešavanjem dobija uvid u promene snaga potrošnji nakon nekog poremećaja u mreži, tj. nakon ispada nekog od generatora ili grane. Pogonska ograničenja EES-a odnose se na proizvodne mogućnosti generatora u pogledu aktivnih snaga, na prenosne sposobnosti svih grana prenosne mreže izraženih preko tokova aktivnih snaga, kao i na maksimalno dozvoljene vrednosti odsecanja potrošnje u čvorovima sistema. Primenom DC modela tokova snaga navedeni problem moguće je formulisati u domenu linearnog programiranja na sledeći način:

$$\min \left\{ z = \sum_{i=1}^N R_i \right\}, \quad (12)$$

gde je R_i odsečena (neisporučena) aktivna potrošnja u čvoru (i). Jednačina (13) predstavlja optimizacioni kriterijum problema minimalnog odsecanja opterećenja i njome se izražava minimizacija zbira odsečenih (neisporučenih) potrošnji u sistemu.

U datom optimizacionom modelu važe sledeća ograničenja:

1. Ograničenje bilansa snaga u svim čvorovima mreže osim referentnog, $i = 2, 3, \dots, N$:

$$-\sum_{j \in \alpha_i} B_{ij} \theta_{ij} + P_{gi} + R_i = P_{pi}, \quad (13)$$

2. Ograničenje ukupnog bilansa snaga u EES-u:

$$\sum_{i=1}^N P_{gi} + \sum_{i=1}^N R_i = \sum_{i=1}^N P_{pi}. \quad (14)$$

3. Ograničenje opsega proizvodnih jedinica u svim čvorovima, $i = 1, 2, \dots, N$:

$$P_{gi}^{min} \leq P_{gi} \leq P_{gi}^{max}, \quad (15)$$

gde su sa P_{gi}^{min} i P_{gi}^{max} , predstavljene donja i gornja granica proizvodnje aktivne snage u čvoru (i), respektivno.

4. Ograničenje potrošnje u svim čvorovima, $i = 1, 2, \dots, N$:

$$0 \leq R_i \leq P_{pi}. \quad (16)$$

5. Ograničenje maksimalne prenosne sposobnosti svih grana sistema, $l = 1, 2, \dots, L$:

$$-\bar{P}_l \leq b_{ij} \cdot (\theta_i - \theta_j) \leq \bar{P}_l \quad (17)$$

gde je sa $\bar{P}_l$ predstavljena maksimalna dozvoljena vrednost toka aktivne snage po grani (l).

6. PROBLEM PRERASPODELE GENERATORSKE PROIZVODNJE

Problem preraspodele generatorske proizvodnje predstavlja optimizacioni problem kojim se određuje režim u kojem je zbir troškova promene aktivnih generatorskih proizvodnji u svim čvorovima minimalan, pri čemu su zadovoljena sva pogonska ograničenja [3]. Ovaj problem je značajan jer se njegovim rešavanjem dobija uvid u preraspodelu generatorskih proizvodnji nakon nekog poremećaja. Primenom DC modela tokova snaga navedeni problem moguće je formulisati u domenu linearnog programiranja na sledeći način:

$$\min \left\{ z = \sum_{i=1}^N c_i(P_{gi}^0) \cdot \Delta P_{gi} \right\}, \quad (18)$$

gde su sa $c_i(P_{gi}^0)$ označeni proizvodni troškovi pri promeni generatorske proizvodnje u tački linearizacije definisanoj preko vrednosti bazne proizvodnje P_{gi}^0 , u [N.J./r.j. MW], a sa ΔP_{gi} predstavljena je promena proizvodnje aktivne snage u čvoru (i) u odnosu na baznu vrednost P_{gi}^0 . U ovom optimizacionom modelu važe sledeća ograničenja:

1. Bilansna jednačina aktivnih snaga u sistemu:

$$\sum_{i=1}^N \Delta P_{gi} = 0, \quad (19)$$

2. Ograničenje promene aktivnih generatorskih aktivnih snaga $i = 1, 2, \dots, N$:

$$P_{gi}^{min} - P_{gi}^0 \leq \Delta P_{gi} \leq P_{gi}^{max} - P_{gi}^0, \quad (20)$$

3. Ograničenje promene tokova aktivnih snaga po granama sistema, $l = 1, 2, \dots, L$:

$$-\bar{P}_l - P_l^0 \leq \sum_{n=2}^N a_l^n \cdot \Delta P_{gn} \leq \bar{P}_l - P_l^0, \quad (21)$$

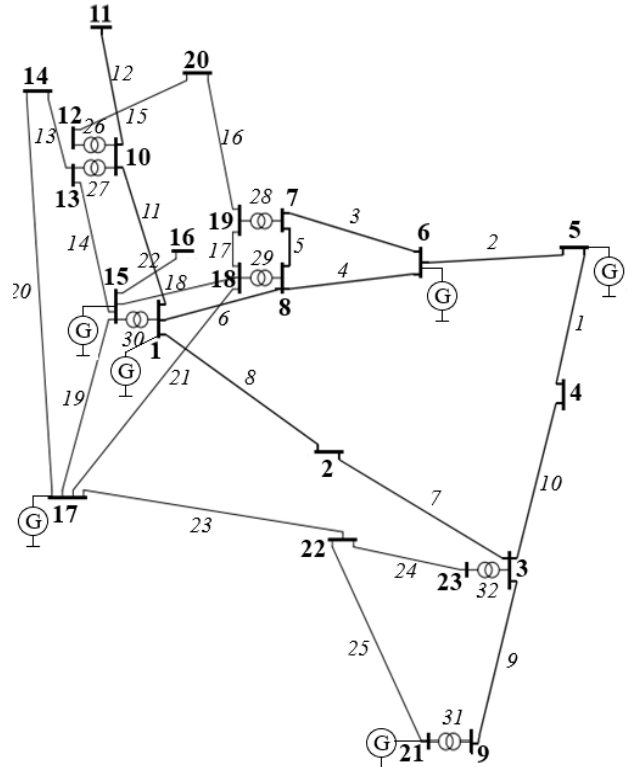
gde je a_l^n je faktor promene toka snage u grani (l) pri jediničnoj promeni aktivne generatorske proizvodnje u čvoru (n).

7. PRIMENE DC MODELA TOKOVA SNAGA

Model DC tokova snaga primenjen je u proračunima analize sigurnosti, faktora osetljivosti, a zatim i u rešavanju optimizacionih problema odsecanja potrošnje i preraspodele generatorske proizvodnje. U ovoj glavi su za svaku primenu prikazani dobijeni rezultati proračuna.

7.1. Opis prenosne test mreže

Svi proračuni vršeni su na uprošćenoj prenosnoj test mreži prikazanoj na slici 1. Ovim sistemom je obuhvaćeno 11 čvorova 400 kV naponskog nivoa i 12 čvorova 220 kV nivoa realnog sistema, dok su svi čvorovi niženaponskih nivoa zanemareni. Podaci o test mreži dati su u [4]. Na slici 1 indeksi čvorova su podebljani (bold), dok su indeksi grana označeni kurzivom (italic).



Slika 1. Prikaz prenosne test mreže [4]

7.2. Proračun DC tokova snaga

Nakon proračuna tokova snaga baziranih na DC modelu, najpribližniji rezultat proračuna dobijen je za granu (1) i pri tome relativna greška iznosi 0.18 %, dok se najveće odstupanje javlja za granu (16) i pri tome relativna greška iznosi 4.96 %.

7.3. Analiza sigurnosti i proračuni faktora osetljivosti

U slučaju analize sigurnosti za ispade generatora, dolazi se do zaključka da se najkritičnije vrednosti u pogledu prekoračenja dozvoljenih tokova snaga dobijaju u slučaju ispada generatora u čvoru (5).

Do najvećeg prekoračenja dolazi u grani (5), gde ono čak za 97.2% premašuje dozvoljenu granicu toka aktivne snage za posmatranu granu. U ovom slučaju, faktor preraspodele toka aktivne snage za granu sa preopterećenjem iznosi $a_5^5 = 0.304$.

Za slučaj ispada grana, najnepovoljniji slučaj dešava se kada dođe do ispada grane (3), pri tome najveće preopterećenje je u grani (5) i u tom slučaju je dozvoljena granica premašena za 97.1 %.

Vrednost distribucionog faktora pri ispadu grane (3) za granu sa preopterećenjem iznosi $d_3^3 = -0.921$.

7.4. Minimalno odsecanje opterećenja

Problem minimalnog odsecanja opterećenja u ovom radu, razmatran je za najkritičniji slučaj ispada grana, tj. za ispad grane (3). Navedeni optimizacioni model je implementiran u programskom jeziku AMPL i rešavan na on-line NEOS serveru [5].

Nakon proračuna, dobija se da opterećenje koje je potrebno odseći iznosi 230.962 MW.

U tabeli 1 prikazani su čvorovi u kojima je došlo do redukcije potrošnje, kao i njihove vrednosti pre i posle razmatranog poremećaja.

7.5. Preraspodela generatorske proizvodnje

Problem optimalne preraspodele generatorske proizvodnje takođe je razmatran za najkritičniji slučaj ispada grane, tj. za ispad grane (3). Navedeni optimizacioni model takođe je implementiran u programskom jeziku AMPL i rešavan na on-line NEOS serveru.

Cene proizvodnih troškova određuju se na sledeći način:

$$c_i = 3.385 \cdot P_{gi} + 0.007 \cdot P_{gi}^2, \quad i = 1, \dots, N. \quad (22)$$

Nakon proračuna dobija se da zbir proizvodnih troškova iznosi 28.855 [N.J./MW]

Vrednosti snaga proizvodnje po čvorovima, pre i nakon razmatranog poremećaja, prikazane su u tabeli 2.

8. ZAKLJUČAK

U ovom radu opisan je DC model tokova snaga, a cilj rada je upravo bio da pokaže efikasnost i primene ovog modela u eksploataciji i planiranju EES-a. Na osnovu proračunatih vrednosti tokova aktivnih snaga, u vrlo kratkom roku se može analizirati sigurnost EES-a poređenjem vrednosti tokova snaga sa unapred definisanim granicama maksimalnih tokova snaga u slučaju nekog od poremećaja sa liste specifikovanih kvarova.

Sa druge strane, pomoću proračunatih faktora osetljivosti prenosnih snaga, vrlo brzo se može utvrditi postojanje preopterećenja za slučaj pretpostavljenih ispada svih generatora i grana u sistemu.

Takođe, pokazuje se da je DC model tokova snaga vrlo efikasan i pri rešavanju optimizacionih problema koji postoje u planiranju i eksploataciji EES-a.

Tabela 1. Redukcija potrošnje u čvorovima EES-a

Indeks čvora	Stara vrednost snage potrošnje [MW]	Nova vrednost snage potrošnje [MW]
7	24.00	0.00
18	540.00	376.75
19	167.00	0.00
20	130.50	2.728

Tabela 2. Promena proizvodnje u čvorovima EES-a

Indeks čvora	Stara vrednost snage proizvodnje [MW]	Nova vrednost snage proizvodnje [MW]
1	1184.40	1379.80
5	675.00	264.10
6	349.00	355.32
15	553.00	663.60
17	1111.00	1333.20
21	436.00	312.37

9. LITERATURA

- [1] V. C. Strezoski: *Osnovni proračuni elektroenergetskih sistema. Tom II, Tokovi snaga i kratki spojevi*; FTN Izdavaštvo, Novi Sad, 2017.
- [2] M. S. Čalović, A. T. Sarić, P. Č. Stefanov: *Eksploatacija elektroenergetskih sistema u uslovima slobodnog tržišta*; Srbija: Čačak, Tehnički fakultet, 2005.
- [3] V. Levi, D. Bekut: *Primena računarskih metoda u elektroenergetici*; Stylos art, Novi Sad, 1997.
- [4] N. Rajaković, M. Čalović, P. Stefanov, A. Savić: *100 rešenih zadataka iz analize elektroenergetskih sistema*; Elektrotehnički fakultet Beograd, Beograd, 2002.
- [5] <https://neos-server.org>

Kratka biografija



Goran Mitrović rođen u Novom Sadu, 1996. god. Diplomirao je na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi 2019. godine.



Neven Kovački rođen u Zrenjaninu, 1987. god. Diplomirao je, masterirao i doktorirao na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Elektroenergetski sistemi redom 2010, 2011. i 2018. godine.

DETEKCIJA OBLIGACIJA U UGOVORIMA NA ENGLJESKOM JEZIKU**OBLIGATION DETECTION IN ENGLISH CONTRACTS**Marko Žužić, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu predstavljen je sistem za detekciju obligacija u okviru ugovora napisanih na engleskom jeziku. Klasifikator obligacija kao ulaz prima rečenice iz ugovora, a kao izlaz daje informaciju da li su rečenice obligacije, ili ne.

Ključne reči: Detekcija obligacija, Pravni domen, Jezički modeli, NLP, Klasifikacija teksta, BERT

Abstract – This paper presents a system for obligation detection in english contracts. The classifier trained to solve the problem accepts sentences from contracts as an input and outputs information on whether they are considered obligations, or not.

Keywords: Obligation detection, Legal domain, Language models, NLP, Text classification, BERT

1. UVOD

Pisani ugovori su sastavni deo današnjeg poslovnog sveta. Prilikom stupanja u bilo kakav poslovni odnos sklapa se ugovor kojim se definiše odnos između ugovornih strana. Pregledanje ugovora treba da obavi pravno obučeno lice i ono iziskuje dosta vremena, a samim tim i novca. Jasna je motivacija za pravljenje sistema koji bi delimično, ili u potpunosti, automatizovao, a samim tim i olakšao i ubrzao, proces analize pravnih dokumenata. Obligacije, kao osnova svakog pravnog dokumenta, predstavljaju bitan segment ovakvog sistema. Stoga je u ovom radu ponuđeno rešenje problema automatske detekcije obligacija.

Skup podataka koji je korišćen u radu sačinjen je od ručno anotiranih rečenica (“jeste obligacija”, ili “nije obligacija”) iz javno dostupnih ugovora.

Algoritmi koji su korišćeni za rešavanje problema detekcije obligacija birani su na osnovu dostupnih rezultata iz literature koja je pisana na istu ili sličnu temu. Isprobani su algoritmi za vektorizaciju rečenica, poput *bag of words* i *TFIDF*, kao i složeniji, poput *FastText*-a. Nakon vektorizacije, trenirano je nekoliko različitih binarnih klasifikatora. Kao osnovni klasifikator isproban je *Naive Bayes*, a potom i napredniji klasifikatori, poput *SVM* i *XGBoost*-a. Isprobane su i tehnike dubokog učenja, poput konvolutivnih neuronskih mreža i *BERT* transformer modela.

Dobijeni rezultati potvrdili su performanse algoritama vektorizacije teksta i klasifikatora koje su dobijene u

prethodnim radovima. Najbolje rezultate dale su tehnike dubokog učenja (88% F1 skor prilikom korišćenja *LegalBERT* modela), dok su kontekstualne tehnike vektorizacije rečenica u kombinaciji sa jednostavnijim klasifikatorima dale slabije rezultate (82% F1 skor za *XGBoost* klasifikator).

U narednom poglavlju biće analizirana relevantna literatura za problem detekcije obligacija koji se analizira. U trećem poglavlju biće opisana predložena metodologija za rešavanje problema, dok četvrto poglavlje nudi uvid u rezultate i njihovu diskusiju. U petom poglavlju dat je zaključak o samom radu i problemu koji je rešavan.

2. PRETHODNA REŠENJA

Klasifikacija delova ugovora, poput klauzula, ili obligacija, nije obimno istražena tema u literaturi.

U radu [1] vršena je analiza tehnika za klasifikaciju sudskih procesa na različite pravne domene. Autori su implementirali ansambl sistem, sastavljen od nekoliko *SVM* klasifikatora sa linearnim kernelom i dostigli F1 skor od 96% nad skupom automatski označenih pravnih dokumenata kojih je bilo 126,865. Skup obeležja koji su iskoristili bio je jednostavan: *bag of words* sa unigramima i *bag of bigrams*, gde su gledali samo broj pojavljivanja svakog unigrama, ili bigrama u celom korpusu reči.

U radu [2] rađena je *multilabel* klasifikacija nad 57,000 pravnih legislativa Evropske Unije na 4,300 različitih klasa. Autori ovog rada su kao osnovnu arhitekturu koristili algoritam logističke regresije sa *TF-IDF* vektorima ngrama reči u korpusu. Ova tehnika vektorizacije isprobana je i u ovom radu.

Kao napredne arhitekture u radu [2] korišćene su kompleksne neuronske mreže, poput *CNN-LWAN* i *BIGRU*. *CNN-LWAN* mreža podrazumeva neuronsku mrežu, gde su obeležja dokumenta (rečenice) su enkodovane preko *CNN* sloja mreže. Druga arhitektura koju su isprobali bila je *BIGRU* mreža, u okviru koje je dokument predstavljen kao niz vektora pojedinačnih reči u njemu, koji dalje prolazi kroz *BIGRU* sloj (specijalna vrsta rekurentne neuronske mreže). Arhitektura koja je dala najbolji rezultat u ovom radu bila je kombinacija *CNN-LWAN* i *BIGRU* arhitekture. Stoga je odlučeno da se i u ovom radu isprobaju neke specijalne tehnike kreiranja vektora, poput konvolutivne neuronske mreže koju koristi biblioteka *spaCy*. Pored pomenutih arhitekture, autori su isprobali da koriste i *BERT* transformer algoritam. On je dao dobre rezultate, te je stoga isproban kao jedan od algoritama u ovom radu.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji je mentor bio prof. dr Aleksandar Kovačević.

Rad [3] i drugi radovi na temu klasifikacije rečenica često isprobavaju kako druge tehnike vektorizacije utiču na rezultate. Jedna od tehnika vektorizacije je kreiranje *FastText* vektora, koja je korišćena u radovima [1], [2] i [3]. U radu [1] autori su pomoću jednostavnog linearnog klasifikatora i *FastText* vektora postigli rezultate slične onima dobijenim korišćenjem tehnika dubokog učenja (konvolutivne mreže). *FastText* vektorizacija je istražena i u ovom radu.

Pored *FastText* vektorizacije, autori radova na sličnu temu koristili su i druge algoritme koji uzimaju u obzir kontekst oko pojedinačnih reči u rečenici, poput *GloVe* i *Word2Vec* algoritma [1], [3]. Ove tehnike isprobane su i u ovom radu.

3. METODOLOGIJA I ALATI

Ovo poglavlje posvećeno je prikazu primenjene metodologije za detekciju obligacija: korišćenog skupa podataka, tehnika vektorizacije teksta i odabranih klasifikatora.

3.1. Formiranje skupa podataka

Skup podataka je dobijen manuelnim anotiranjem rečenica koja su izvučene iz javno dostupnih ugovora. Rečenice su anotirala pravna lica. Svaka rečenica označavana je jednom od dve klase: "jeste obligacija" (1), ili "nije obligacija" (0). Klase su prilično izbalansirane, imajući 7944 pozitivnih primera i 7001 negativnih primera. Trenig i test skup su dobijeni deljenjem skupa podataka u odnosu 80:20, prilikom čega je očuvana distribucija pozitivnih i negativnih primera.

Skup podataka je formiran iterativno. Početni, manji skup podataka, formiran je korišćenjem jednostavnih pravila (korišćenjem frazi "*should*", "*should not*", "*is obliged*", "*is not obliged*", "*is obligated*", "*is not obligated*"). On je dat pravicima, koji su formirali skup od 200 rečenica koje su zaista bile obligacije. Od ostatka skupa uzeto je još 200 rečenica koje nisu obligacije, kako bi se dobio balansirani skup podataka za treniranje početnog algoritma.

Za dalje proširivanje skupa podataka najpre je istreniran početni algoritam nad 400 obligacija. Ovaj model je dalje korišćen za inicijalno obeležavanje preostalog skupa podataka koji je sakupljen pomoću pravila, pomenutih u pasusima iznad.

Ovako je skup formiran iterativno. Pretpostavka je bila da će algoritam davati bolje rezultate sa većom količinom podataka.

3.2. Tehnike preprocesiranja teksta

Iz formalnih rečenica koje su sačinjavale skup podataka su izbačeni znakovi interpunkcije, specijalni znakovi i stop reči (engl. *stop words*), dok je ostatak teksta prebačen u mala slova [7].

3.3. Vektorizacija teksta

Odabrano je i isprobano nekoliko tehnika vektorizacije teksta, po ugledu na prethodnu literaturu na sličnu temu. Sve tehnike opisane su u nastavku teksta.

Bag of words je pojednostavljena reprezentacija teksta u okviru koje se delovi teksta predstavljaju frekvencijom reči koje taj tekst čine [8]. Ova tehnika je istražena u ovom radu kao osnovna tehnika (engl. *baseline*), kao i u radu [1].

Kod *TF-IDF* tehnike broj pojavljivanja termina deli se ukupnim brojem termina koji su prisutni u tom dokumentu. Kako bi se termini razlikovali po važnosti, uvodi se "*Inverse Document Frequency*", koji dodeljuje veću težinu terminima koji su retki. Što je veći broj pojavljivanja termina u dokumentu, veći je njegov *term frequency*, a što se manje pojavljuje u drugim dokumentima, raste njegova važnost [9]. Očekivano je da ova tehnika vektorizacije da bolje rezultate kada se koristi za klasifikaciju teksta [1] [2].

GloVe (engl. *Global Vectors*) metod pretpostavlja da se veza između reči može zaključiti proučavanjem verovatnoće njihovog zajedničkog pojavljivanja. Stoga, *GloVe* reprezentacijom moguće je iskazati i semantičke informacije [10]. U slučaju ovog rada, vektorizovala se cela rečenica, tako što su se uzimali *GloVe* vektori pojedinačnih reči u njoj i na kraju računala srednja vrednost vektora svih vektora reči u rečenici [1] [2] [3].

FastText algoritam vektorizacije uzima u obzir internu strukturu reči i njenu okolinu [11]. Ovaj algoritam pretpostavlja da se svaka reč sastoji iz *n*-grama karaktera, gde dužina *n* može varirati od 1 do dužine reči [10]. Slično kao i kod *GloVe* reprezentacije, vektorizuje se cela rečenica tako što se uzima *FastText* reprezentacija svih pojedinačnih reči u rečenici i na kraju računa srednja vrednost vektora za celu rečenicu [1] [2] [3].

3.4. Klasifikacija teksta

U ovom radu isprobani su klasični algoritmi mašinskog učenja, kao i algoritmi dubokog učenja. Svi korišćeni algoritmi opisani su u nastavku teksta.

Naive Bayes je algoritam mašinskog učenja zasnovan na Bajesovoj teoremi [13]. Ovaj algoritam, međutim, podrazumeva međusobnu uslovnu nezavisnost atributa u okviru jedne klase. U prethodnoj literaturi je pokazao prilično dobre rezultate za probleme binarne klasifikacije teksta [1].

SVM je algoritam mašinskog učenja razvijen za binarnu klasifikaciju [12]. Ovaj algoritam je zasnovan na ideji pronalaženja hiperravni (engl. *hyperplane*) koja najbolje deli skup podataka u dve kategorije. U prethodnim radovima na sličnu temu, *SVM* klasifikator davao je bolje rezultate nego *Naive Bayes*, te je stoga odlučeno da se isproba i u ovom radu [1] [3].

Extreme Gradient Boosting (XGBoost) je algoritam iz grupe *boosting* algoritama, koji kombinuju slabe "učenike", tj. jednostavna stabla odlučivanja, koja nemaju veliku preciznost prilikom klasifikacije, u precizniji algoritam, iterativno. *XGBoost* obično daje najbolje metrike od svih klasičnih algoritama, koji ne spadaju u kategoriju neuronskih mreža [14]. Stoga je odlučeno da se isproba i u ovom radu.

SpaCy je biblioteka koja nudi različite pretrenirane modele nad generalnim skupom podataka. Arhitektura

koja je u ovom radu korišćena kao osnova za treniranje tekstualnog klasifikatora bila je neuronska mreža, gde se vektori pojedinačnih tokena računaju koristeći jednostavnu konvolutivnu neuronsku mrežu (*CNN*). Više detalja o specifičnoj implementaciji može se pronaći na zvaničnoj stranici *spaCy* biblioteke [6]. Ovaj model je isproban zbog toga što je u literaturi pronađeno da vektori dobijeni kroz konvolutivne mreže doprinose dobrim metrikama prilikom tekstualne klasifikacije [1].

BERT (*Bidirectional Encoder Representations from Transformers*) je transformer model, koji koristi bidirekciono treniranje i *attention*¹ mehanizam kako bi napravio model jezika (engl. *Language model*) [4]. *BERT* koristi transformer i *attention* mehanizam, koji ima sposobnost da nauči kontekstualne odnose između reči (ili delova reči) u tekstu. U slučaju ovog rada korišćen je *BERT* specijalizovan za pravni domen, koji se zove *LegalBERT* [5]. Korpus koji je korišćen za treniranje ovakvog jezičkog modela je uključivao tekst iz različitih pravnih dokumenata (legislativa, sudskih dokumenata, presuda, ugovora).

4. EKSPERIMENTALNI REZULTATI I DISKUSIJA

Eksperimentalna evaluacija izvršena je radi određivanja i kvantifikovanja performansi svih pojedinačnih algoritama.

Svaki klasifikator obučen je pomoću skupa koji se sastojao od približno 10500 parova (rečenica, binarna labela), gde je binarna labela imala vrednost 1 ili 0, u zavisnosti od toga da li je data rečenica obligacija, ili ne. Za svaki klasifikator optimizovani su hiperparametri. Metrike klasifikatora su određene uz pomoć testnog skupa, koji je činilo 4.500 rečenica, na isti način anotiranih kao i kod obučavajućeg skupa.

Metrike koje su korišćene u radu za evaluaciju svih klasifikatora su preciznost (engl. *precision*), odziv (engl. *recall*) i F1 mera. Odziv je posebno bitna metrika u radu, jer je bitnije da se u ugovoru prepoznaju i označe sve obligacije.

4.1. Evaluacija sistema

Prvi model koji je istreniran i evaluiran bio je multinomijalni *Naive Bayes*. Kod ovog modela trenirane su tri varijante (sram vektorizacije): *bag of words* i *TFIDF* (sa tehnikama preprocesiranja i bez njih). Najbolje rezultate (F1 skor od 73%) dalo je korišćenje *TFIDF* i tehnika preprocesiranja teksta, što je u skladu sa prethodnom literaturom na sličnu temu [1].

Kod *SVM* modela istrenirano je nekoliko varijanti (sram vektorizacije): *bag of words*, *TFIDF*, *GloVe* i *FastText* nad generalnim domenom. Svaka varijanta modela trenirana je korišćenjem tehnika preprocesiranja. Najbolje rezultate dalo je korišćenje *TFIDF* i tehnika preprocesiranja teksta (F1 skor od 82%). Tehnike vektorizacije uz pomoć *FastText*-a i *GloVe*-a nisu dali očekivano visoke rezultate, jer su to vektori koji se formiraju nad generalnim domenom teksta, a ne nad domenom ugovora.

XGBoost model je treniran u nekoliko varijanti (sram vektorizacije): *bag of words*, *TFIDF*, *GloVe* i *FastText* vektora nad pravnim domenom. Najbolja varijanta ovog modela dobijena je korišćenjem *BoW* vektorizacije (F1 skor od 82%).

Poslednja dva modela koja su istrenirana i evaluirana bile su duboke neuronske mreže: konvolutivna neuronska mreža kroz biblioteku *SpaCy*, kao i transformer model *LegalBERT*.

U tabeli 4.1 mogu se videti objedinjeni rezultati najboljih istreniranih varijanti svih isprobanih modela.

Model	Preciznost	Odziv	F1
NB TFIDF pp	76%	75%	74%
SVM TFIDF	82%	82%	82%
XGB BoW	82%	82%	82%
SpaCy CNN	86%	86%	86%
LegalBERT	88%	88%	88%

Tabela 4.1: Rezultati najboljih varijanti svih modela

Može se primetiti da je *SVM* ostvario bolje rezultate od *Naive Bayes* modela. U prethodnoj literaturi je to takođe bio slučaj [1] [3].

Takođe, vidimo da *XGBoost* daje slične rezultate kao *SVM* model, ali bolje od *Naive Bayes*-a. Ovo potvrđuje pretpostavke iz literature [17].

Metrike *CNN* su za par procenata bolje od najboljeg “klasičnog” modela istreniranog odabranim tehnikama vektorizacije rečenica. Odziv nam je naročito bitan za detekciju obligacija, jer želimo da ih sve pronađemo i prikazemo pravnim licima koji analiziraju ugovore. Ovde je odziv iznosio 86%, što je za 4% bolje od najboljeg odziva “klasičnih” modela. Ovo je u skladu i sa prethodnim istraživanjima na sličnu temu [3].

LegalBERT daje daje najbolje rezultate do sada: prosečan odziv je 88%, a i F1 skor iznosi 88%. Pretreniranje *BERT*-a je veoma značajno, jer je u ovom slučaju korišćen transformer model treniran baš nad domenom nad kojim treniramo i sam klasifikator. Ovi rezultati su u skladu i sa rezultatima iz prethodnih radova [3] [5].

4.2. Analiza grešaka

Pored evaluacije, analizirane su greške osnovnog modela (*BoW* u kombinaciji sa multinomijalnim *Naive Bayes*-om), konvolutivne neuronske mreže, kao i modela baziranog na transformer arhitekturi, koji je dao najbolje rezultate (*LegalBERT*).

Baseline model, *Naive Bayes*, grešio je u dosta slučajeva gde su rečenice bile dugačke. Takve rečenice obično su sadržale reči koje nisu u kontekstu obligacija, nego “uobičajene” reči, poput “software”, ili “books”, i ne javljaju se mnogo u skupu podataka u sklopu obligacija.

¹ selektivno fokusiranje na određene delove vektora koji su relevantni

Konvolutivna neuronska mreža greši kod sličnih rečenica, jer nije uspjela da dobro poveže kontekst određenih reči, koje se pojavljuju u dugačkim rečenicama.

LegalBERT model uglavnom greši kod klasifikacije nestandardnih struktura rečenica i prilikom korišćenja fraza koje se ne javljaju često u rečenicama koje nisu obligacije, kao npr. “*agrees to sign*”.

Pored ovoga, *LegalBERT* greši i u sledećim tipovima primera:

“*This Section 18.0 shall not delay or excuse Client's payment obligations.*”

Ovakva rečenica nije obligacija. Međutim, u primeru se pojavljuje fraza “*Section 18.0*”, koju je *LegalBERT* verovatno tretirao kao entitet u samom dokumentu (ugovornu stranu) i stoga pogrešno klasifikovao ovu rečenicu kao obligaciju.

5. ZAKLJUČAK

U okviru ovog rada implementiran je algoritam za detekciju obligacija u ugovorima napisanim na engleskom jeziku. Ulaz u algoritam je rečenica iz ugovora, a izlaz je odgovor da li je rečenica obligacija, ili ne.

Za potrebe rada trenirano je i evaluirano 5 različitih tipova klasifikatora: *Naive Bayes*, *SVM*, *XGBoost*, *SpaCy CNN* i *LegalBERT*. Isprobane su i različite tehnike vektorizacije teksta: od jednostavnijih, kao što su *bag of words* i *TF-IDF*, do naprednih tehnika koje se baziraju na vektorima koje enkoduju okolinu pojedinačnih reči u rečenici: *GloVe* i *FastText*.

Motivacija za rešavanje problema detekcije obligacija leži u omogućavanju automatizacije procesa čitanja dugačkih ugovora na engleskom jeziku. Implementirani su algoritmi koji klasifikuju rečenice po tome da li predstavljaju obligaciju, ili ne.

Skup podataka je napravljen u saradnji sa advokatima koji govore engleski jezik. Najpre je sakupljena (engl. *scrape*²) velika količina ugovora sa sajta *Edgar*³, koja je zatim data advokatima na anotaciju. Oni su najpre anotirali tipove klauzula u dokumentu, a zatim i pojedinačne rečenice iz klauzula u dva tipa: da li one jesu ili nisu obligacija. Skup je izbalansiran i podeljen na trening i test skup.

Za evaluaciju klasifikatora korišćene su metrike preciznosti, odziva i F1 mere. Jako bitan je bio odziv svakog klasifikatora, zato što želimo da pravniku koji čita ugovore prikazemo sve obligacije koje u njemu postoje. Dobijeni rezultati klasifikatora su u skladu sa prethodnim rešenjima na istu, ili sličnu temu. Najbolje rezultate dao je klasifikator baziran na pretreniranom *LegalBERT* transformer modelom [4][5].

Osim sakupljanja veće količine podataka, mogu se probati i druge arhitekture dubokih neuronskih mreža u kombinaciji sa složenijim vektorskim reprezentacijama

teksta ili pretreniranje nekog drugog jezičkog modela, baziranog na transformerima, na pravnom domenu.

6. LITERATURA

- [1] Octavia-Maria Sulea, Marcos Zampieri, Shervin Malmasi, Mihaela Vela, Liviu P. Dinu, Josef van Genabith, “Exploring the Use of Text Classification in the Legal Domain”, ASAIL, oktobar 2017.
- [2] Vladimir Zolotov, David Kung, “Analysis and Optimization of FastText Linear Text Classifier”, IBM Watson Research
- [3] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, “Large-Scale Multi-Label Text Classification on EU Legislation”, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, jul 2019
- [4] Asaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Lilion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, “Attention Is All You Need”, NIPS 2017, Long Beach, CA, USA
- [5] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, Ion Androutsopoulos, “LEGAL-BERT: The Muppets straight out of Law School”, oktobar 2020.
- [6] TextCat Ensemble Model, spaCy: <https://spacy.io/api/architectures#TextCatEnsemble>
- [7] V. Srividhya, R. Anitha, “Evaluating Preprocessing Techniques in Text Categorization”
- [8] Somuya George K, Shibily Joseph, “Text Classification by Augmenting Bag of Words (BOW) Representation with Co-Occurrence Feature”, IOSR/JCE, volume 16, Issue 1, Jan 2014.
- [9] Shahzad Qaiser, Ramsha Ali, “Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents”, International Journal of Computer Applications, Volume 181, Jul 2018.
- [10] Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov, “Enriching Word Vectors with Subword Information”, Transactions of the Association for Computational Linguistics, 2017.
- [11] Birol Kuyumcu, Cuneyt Aksakalli, Selman Delil, “An automated new approach in fast text classification (FastText): A case study for Turkish text classification without preprocessing”, NLP19, jun 2019.
- [12] Corinna Cortes, Vladimir Vapnik, “Support-Vector Networks”, Machine Learning, 20, 273-297, 1995.
- [13] I. Rish, “An Empirical Study of the Naive Bayes Classifier”, jan. 2001.
- [14] Ashish Chaturvedi, Santosh Yadav, Mohd. Abuzar Mohd. Haroon Ansari, Mahendra Kanojia, “Comparative Multinomial Text Classification Analysis of Naïve Bayes and XGBoost with SMOTE on Imbalanced Dataset”, septembar 2021.

Kratka biografija:



Marko Žužić rođen je u Subotici 28. aprila 1994. godine. Master rad na Fakultetu tehničkih nauka, oblast Elektrotehnika i računarstvo – Računarske nauke i informatika odbranio je 2021. godine.

Kontakt: markozuzic@nordnet.rs

² korišćenje specijalnih programa za preuzimanje sadržaja sa nekog veb-sajta

³ <https://www.sec.gov/edgar.shtml>, otvorena baza ugovora na engleskom jeziku

DIGITALNI POTPISI I NJIHOVA ULOGA U BLOCKCHAIN TEHNOLOGIJI**DIGITAL SIGNATURES AND THEIR ROLE IN BLOCKCHAIN TECHNOLOGY**

Ana Mutavdžić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – KRIPTOZAŠTITA INFORMACIJA

Kratak sadržaj – Digitalni potpis postao je jedan od najvažnijih kriptografskih alata koji je danas u širokoj upotrebi. Njegova osnovna uloga je da potvrdi identitet pošiljaoca poruke, kao i da obezbedi dokaz da je originalni sadržaj poruke ostao nepromenjen. U ovom radu dat je opis nastanka i funkcionisanja digitalnih potpisa, kao i njihova primena u blockchain tehnologiji.

Ključne reči: Kriptografija, Digitalni potpis, Blockchain tehnologija

Abstract – Digital signature has become one of the most important and broadly used contemporary cryptographic tools. Its main purpose is to verify the identity of the sender of the document, as well as to provide evidence that the original content of the sent message is unchanged. This paper describes the origin and basic principles of digital signatures, as well as their application in blockchain technology.

Keywords: Cryptography, Digital signature, Blockchain technology

1. UVOD

Potreba za komunikacijom je jedna od glavnih sociogenih potreba čoveka. Veći deo čovečanstva dnevno ulazi u komunikaciju sa drugim ljudima. Neki od njih provode sate razgovarajući i razmenjujući informacije, dok neki to rade samo par minuta dnevno. Jako je bitno znati ko šalje poruku i biti siguran da tokom slanja poruke niko nije uspeo da promeni njen sadržaj. Ovo se postiže autentifikacijom. Jedan od najvažnijih kriptografskih alata koji se koristi da potvrdi identitet pošiljaoca poruke kao i činjenicu da je poslata poruka nepromenjena naziva se **digitalni potpis**.

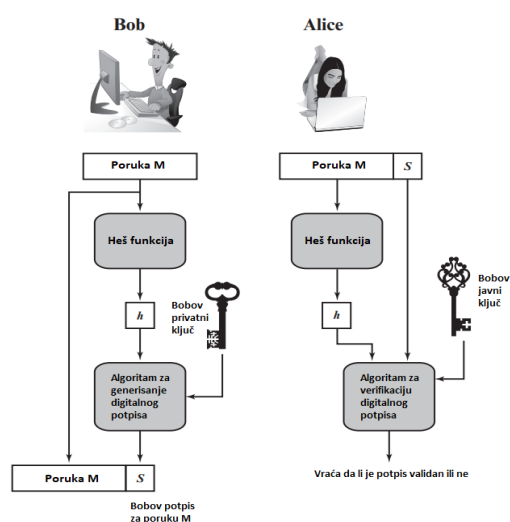
2. DIGITALNI POTPIS

Digitalni potpis je jedan od najvažnijih kriptografskih alata koji danas ima široku primenu. U mnogim slučajevima, oni pružaju sloj provere valjanosti i sigurnosti porukama poslatim putem nebezbednog kanala. Pravilno sproveden, digitalni potpis daje primaocu razlog da veruje da je poruku poslao pošiljalac koji se potpisao. Digitalni potpisi su u mnogim aspektima ekvivalentni tradicionalnim ručno napisanim potpisima, ali je pravilno primenjene digitalne potpise teže falsifikovati nego ručno pisane.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Mladen Kovačević.

Na Slici 1. predstavljen je generički model procesa izrade i upotrebe digitalnih potpisa. Pretpostavimo da komuniciraju dve strane koje nazivamo Alice i Bob. Ukoliko Bob želi da pošalje poruku Alice, on želi da Alice bude sigurna da je poruka zaista od njega. Algoritam potpisa je funkcija Bobovog privatnog ključa te pod pretpostavkom da on zaista drži svoj ključ privatnim, jedino Bob može potpisati poruku M u svoje ime. Poruka M predstavlja ulaz u algoritam za potpis u cilju njenog povezivanja sa potpisom. Bob najpre formira sažetak poruke h korišćenjem određenog algoritma za heširanje. Zatim se sažetak poruke, koji je obično dužine 128 ili 256 bita, šifrue korišćenjem Bobovog privatnog ključa i tako nastaje digitalni potpis S . Digitalni potpis se dodaje na kraj poruke pa se par (M, S) šalje Alice. Važno je napomenuti da digitalni potpis sam po sebi nije od koristi osim ako nije praćen porukom. Potpis je koristan samo ako postoji način da Alice proveri je li važeći ili ne. Za ovo je potrebna funkcija verifikacije koja uzima sažetak poruke i potpis kao ulaze. Da bi se potpis povezao sa Bobom, funkcija takođe zahteva njegov javni ključ. Njen jedini izlaz je binarni izraz *true* ili *false*. Ako je poruka potpisana privatnim ključem koji odgovara javnom ključu za verifikaciju, izlaz je *true*, u suprotnom je *false*.



Slika 1. Generički model procesa izrade i upotrebe digitalnih potpisa

Najčešće korišćeni digitalni potpisi u *blockchain* tehnologiji su digitalni potpisi bazirani na eliptičnim krivama ECDSA (eng. *Elliptic Curve Digital Signature Algorithm*) i Schnorovi potpisi čiji će način rada biti objašnjen u taljem tekstu.

2.1. Eliptične krive

Eliptične krive imaju nekoliko prednosti u odnosu na druge vrste digitalnih potpisa. Konkretno, u krypto-sistemima baziranim na eliptičnim krivama, ključevi dužine 160–256 bita pružaju sigurnost ekvivalentnu 1024–3072-bitnim RSA i šemama baziranim na problemu diskretnog logaritma. Kraća dužina često rezultira kraćim vremenom obrade i kraćim potpisima. Iz ovih razloga, algoritam digitalnog potpisa eliptične krive standardizovan je u SAD od strane Američkog instituta za standarde (ANSI) 1998. godine.

ECDSA standard zasnovan je na problemu diskretnog logaritma konstruisanog u grupi eliptične krive. Generisanje ključeva za ECDSA izgleda ovako:

1. Koristiti eliptičnu krivu E sa modulom p , koeficijentima a i b , i tačkom A koja generiše cikličnu grupu osnovnog reda q .
2. Odabrati slučajni ceo broj d takav da je $0 < d < q$.
3. Izračunati $B = dA$.

Ključevi su sada $k_{pub} = (p, a, b, q, A, B)$ i $k_{pr} = (d)$. ECDSA potpis se sastoji od para celih brojeva (r, s) . Svaka vrednost ima istu bitsku dužinu kao q , što čini prilično kompaktne potpise. Koristeći javni i privatni ključ, potpis za poruku x izračunava se na sledeći način:

1. Odabrati ceo broj kao slučajni efemeralni ključ k_E takav da važi $0 < k_E < q$.
2. Izračunati $R = k_E A$.
3. Neka je $r = x_R$.
4. Izračunati $s \equiv (h(k) + d \cdot r) k_E^{-1} \bmod q$.

Poruka x mora biti heširana pomoću funkcije h da bi se izračunalo s . Izlazna dužina heš funkcije mora biti najmanje iste dužine kao q , a proces verifikacije potpisa je sledeći:

1. Izračunati pomoćnu vrednost $w \equiv s^{-1} \bmod q$.
2. Izračunati pomoćne vrednosti $u_1 \equiv w \cdot h(x) \bmod q$ i $u_2 \equiv w \cdot r \bmod q$.
3. Izračunati $P = u_1 A + u_2 B$.
4. Verifikacija $ver_{k_{pub}}(x, (r, s))$ sledi iz:
$$x_P \begin{cases} = r \bmod p \implies \text{validan potpis} \\ \neq r \bmod p \implies \text{invalidan potpis} \end{cases}$$

Glavni analitički napad na ECDSA pokušava da reši problem diskretnog logaritma eliptične krive. Kada bi napadač bio sposoban za to, mogao bi izračunati privatni ključ d i/ili efemeralni ključ. Takođe, na početku verifikacije mora se proveriti da li su $r, s \in \{1, 2, \dots, q\}$ kako bi se sprečio određeni napad. Ponovna upotreba efemeralnog ključa takođe se mora sprečiti.

2.2. Schnorr digitalni potpis

Schnorr šema potpisa zasnovana je na problemu diskretnog logaritma. Ona minimizuje broj potrebnih izračunavanja pri generisanju potpisa. Prvi deo ove šeme jeste generisanje privatnog i javnog ključa, koje se sastoji od sledećih koraka:

1. Odabrati proste brojeve p i q , tako da važi $p - 1 \equiv 0 \bmod q$.
2. Odabrati ceo broj a , tako da je $a^q = 1 \bmod q$.

3. Odabrati slučajni ceo broj s tako da je $0 < s < q$.
4. Izračunati $v = a^{-s} \bmod p$. Ovo je javni ključ korisnika.
5. Vrednosti a , p i q predstavljaju javni ključ koji može biti zajednički za grupu korisnika. Privatni ključ korisnika je s a javni ključ korisnika je v .

Za vrednosti p i q obično se uzima $p \approx 2^{1024}$ i $q \approx 2^{160}$. Dakle, p je 1024-bitni broj, a q je 160-bitni broj, što je ujedno i dužina SHA-1 heš vrednosti. Korisnik pomoću privatnog ključa s generiše potpis na sledeći način:

1. Odabrati slučajni ceo broj r tako da je $0 < r < q$ i izračunati $x = a^r \bmod p$. Ovo izračunavanje je faza predprocesiranja nezavisna od poruke M koju treba potpisati.
2. Spojiti poruku M sa x i heširati rezultat kako bi se izračunala vrednost $e = H(M \parallel x)$.
3. Izračunati $y = (r + se) \bmod q$. Potpis se sastoji od para (e, y) .

Svaki korisnik može da verifikuje potpis na sledeći način koristeći javni ključ.

1. Izračunati $x' = a^y v^e \bmod p$.
2. Proveriti da li je $e = H(M \parallel x')$. Treba primetiti da važi sledeće

$$x' \equiv a^y v^e \equiv a^y a^{-se} \equiv a^{y-se} \equiv x \bmod p$$

Dakle, $H(M \parallel x') = H(M \parallel x)$.

Schnorovi potpisi izgledaju kao slučajni brojevi. Konkretno, vrednosti r i s su slučajni brojevi. $H(M \parallel x)$ takođe treba da bude slučajan broj jer predstavlja izlaz heš funkcije, a spajanje više slučajnih brojeva daje nasumičan broj. Verovatnoća vrednosti potpisa jednaka je verovatnoći nasumice izabranog broja, bez obzira na sadržaj poruke i na to ko je potpisuje. Ukoliko potpišemo dve različite poruke istim ključevima, procuriće ceo privatni ključ. Ukoliko imamo dve poruke M_1 i M_2 potpisane privatnim ključem s i istim vrednostima r i x tada napadač može da oduzme dve vrednosti i nasumične izmene koje skrivaju privatni ključ će se poništiti. Iz tog razloga je ponovna upotreba vrednosti r i x zabranjena.

Sada kada imamo bolji uvid u način funkcionisanja digitalnih potpisa u daljem tekstu biće reči o revolucionarnoj *blockchain* tehnologiji i primeni digitalnih potpisa u ovoj oblasti.

3. BLOCKCHAIN TEHNOLOGIJA

Blockchain predstavlja revolucionarnu tehnologiju koja je u velikoj meri uticala na različite industrije. Na tržištu je predstavljena svojom prvom modernom aplikacijom *bitcoin*. *Blockchain* tehnologija omogućava svim učesnicima u mreži da postignu dogovor, opšte poznat kao konsenzus. Svi podaci uskladišteni na *blockchain-u* snimaju se digitalno i imaju zajedničku istoriju koja je dostupna svim učesnicima na mreži. Na ovaj način se eliminišu šanse za bilo kakvu lažnu aktivnost ili dupliranje transakcija bez potrebe za trećom stranom.

Postoje tri vrste *blockchaina* a to su javni, privatni i hibridni *blockchain*. Javni *blockchain* o kom će se govoriti u ovom radu je otvorena, decentralizovana mreža računara dostupna svima koji žele da zatraže ili potvrde transakciju. Sve transakcije koje se odvijaju na javnom

blockchainu su potpuno transparentne, što znači da svako može pregledati pojedinosti transakcije. Javni *blockchain* koristi mehanizme konsenzusa dokaza o radu (*eng. proof-of-work*) ili dokaza o ulogu (*eng. proof-of-stake*) i oni koji potvrđuju transakcije dobijaju nagrade. Dva uobičajena primera javnog *blockchaina* su *bitcoin* i *ethereum*.

Blockchain se kao što mu samo ime govori sastoji od blokova. Blok je struktura podataka u kojoj su zapisane digitalne informacije koje se dele putem *blockchaina*. Blokovi su organizovani linearno, a nove transakcije se konstantno procesuiraju od strane rudara (*eng. miner*) u nove blokove koji se nadodaju na kraj lanca. Blok takođe sadrži podatke o svojoj veličini, zapise koji prikazuju sadržaj podataka skladištenih u datom bloku i njihov broj. U strukturi bloka takođe se nalazi i zaglavlje, koje sadrži metapodatke.

3.1. Sistem ravnopravnih partnera

Sistem ravnopravnih partnera, odnosno sistem građen prema modelu ravnopravnih partnera (*eng. peer-to-peer*) sastoji se od velikog broja procesa, takozvanih partnera (*eng. peer*). Partneri obavljaju zadatke prema potrebama svojih korisnika. U javnom *blockchainu* partneri se razlikuju prema funkcijama koje obavljaju. Vrste partnera su:

- Potpuni partner
- Rudar
- Novčanik
- Blockchain partner

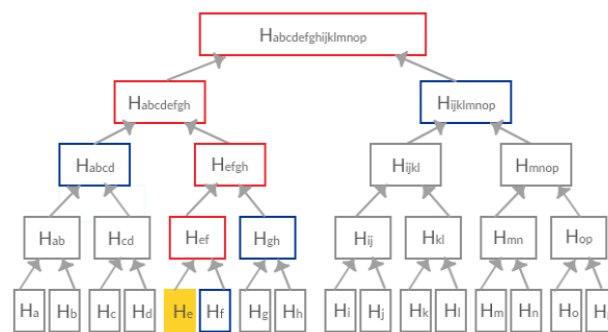
3.1.1. Novčanik

U javnim sistemima koji koriste *blockchain* zbog velike količine podataka nema svaki korisnik mogućnost da skladišti podatke u čitav *blockchain*. Takav korisnik tada u sistemu sudeluje kao jednostavan novčanik.

Glavni zadatak koji jednostavni novčanik obavlja jeste kreiranje novih zapisa u skladu sa protokolom koji propisuje sistem. U svrhu potvrde vlasništva nad digitalnim novcem ili nekom drugom digitalnom informacijom čiji integritet želimo zaštititi stavljanjem na *blockchain*, novčanici skladište parove javnih i privatnih kriptografskih ključeva. Budući da nemaju punu sliku transakcija koje su se dogodile pre one koju žele da validiraju oslanjaju se na ostale partnere koji im na njihov zahtev mogu pružiti uvid u deo *blockchaina*. Jednostavni novčanici proveravaju dubinu transakcija. Binarno heš stablo pruža partnerima novčanici proveru bloku pripada određena transakcija, to je prikazano na Slici 2.

U blok čije binarno heš stablo vidimo na slici, zapisano je 16 transakcija i jednostavni novčanik želi proveriti pripada li tom bloku transakcija *e*. Budući da poznaje sadržaj transakcije *e* partner lako može odrediti heš H_e . Kako bi proverio pripadnost transakcije bloku potrebno je da primi od nekog od svojih susednih partnera koji održavaju celu kopiju *blockchaina* heš vrednosti H_f , H_{gh} , H_{abcd} i $H_{ijklmnop}$. Ove heš vrednosti na Slici 2. označene su plavim pravougaonikima i nazivaju se *autentifikacijski put*. Nakon što jednostavni novčanik poseduje autentifikacijski put on može izračunati heš vrednosti H_{ef} , H_{efgh} , $H_{abcde fgh}$ i konačno koren binarnog heš stabla $H_{abcde fghijklmnop}$ označen crvenim pravougaonikom. Izračunat koren stabla partner tada upoređuje sa korenom

binarnog heš stabla zapisanog u zaglavlju bloka, ako se ti podaci podudaraju transakcija *e* pripada bloku sa Slike 2.



Slika 2. Proces potvrde pripadnosti transakcije određenom bloku

3.1.2. Blockchain partner

Glavna funkcija blockchain partnera je da održava ceo lanac sa podacima u njemu, krenuvši od prvog do poslednjeg bloka u *blockchainu* koji u tom određenom trenutku postoji. Ukoliko je potrebno validirati neku određenu transakciju ili proveriti verodostojnost neke transakcije koja je zapisana u *blockchainu*, tada *blockchain* partner može proveriti jesu li sredstva koja su deo transakcije stvarno od tog korisnika.

3.1.3. Rudar

Funkcija rudara je da prihvata nove zapise koji su napravljeni od strane novčanika, kreira blokove zapisa i smešta ih u *blockchain*. Na primeru *bitcoina*, način kako se upisuju novi zapisi u lanac potrebno je korišćenje računarskih resursa. Pod korišćenjem računarskih resursa, misli se na rešavanje algoritma koji se zove *proof-of-stake*. Dakle, novi blokovi se dodaju u *blockchain* pomoću rešavanja algoritma, kada se izvrši validacija transkripcije rudar koji je obavio taj zadatak i utrošio svoje resurse na izvršenje iste, biva nagrađen sa određenim udelom *bitcoina*.

Treba napomenuti da potpuni partner može obavljati funkcije svih prethodno navedenih partnera.

3.2. Uloga digitalnog potpisa u blockchain tehnologiji

Digitalni potpisi su temelji *blockchain* tehnologije, pružaju ključne prednosti skladištenja i prenošenja i garantuju integritet. Tehnički, poslati podaci mogu se proveriti da da to ne učini haker. Čak i da se to dogodi, u slučaju podataka koje prati digitalni potpis, potpis bi podrazumevano postao nevažeći. Prema tome, digitalno potpisani podaci, koji su šifrovani, ne samo da su sigurni već će i u slučaju da su podaci falsifikovani to otkriti i učvrstiti njihovu koruptivnost.

Kada korisnici podnose transakcije, moraju dokazati svakom čvoru u sistemu da su ovlašćeni da troše ta sredstva, dok sprečavaju i druge korisnike da troše ta sredstva. Svaki čvor u mreži će proveriti uslove poslate transakcije i proveriti rad svih ostalih čvorova kako bi se dogovorili o ispravnom stanju. Ovo se postiže u okviru tri osnovne faze, to su heširanje, potpis i verifikacija.

Faza 1: Pre svega, *blockchain* hešira poruku ili digitalne podatke putem algoritma heširanja. Algoritam pomaže u generisanju heš vrednosti kako bi različite poruke imale

heš vrednosti istih dužina. Kao što već znamo, ovo je osnovna osobina heš funkcije i pokazuje jasan uticaj na digitalne potpise. Heširanje je obavezno u većini *blockchain* aplikacija zbog fleksibilnosti u korišćenju sažetka poruke fiksne dužine.

Faza 2: Sledeći korak u radu digitalnog potpisa u *blockchainu* odnosi se na potpisivanje. Nakon heširanja informacija pošiljalac poruke mora da je potpiše. U ovom trenutku procesa, kriptografija javnog ključa igra ključnu ulogu. Mnogi algoritmi za digitalni potpis nude jedinstvene mehanizme, iako sa jedinstvenim pristupom asimetrične kriptografije. Pošto su digitalni potpisi direktno povezani sa sadržajem svake poruke, digitalno potpisane poruke će imati različite digitalne potpise.

Faza 3: Poslednji korak u korišćenju digitalnog potpisa odnosi se na verifikaciju. Primaoci su lako mogli da provere valjanost digitalnih potpisa upotrebom javnog ključa. Potpis bi mogao funkcionisati kao jedinstveni digitalni otisak date poruke. Međutim, takođe je važno obratiti pažnju na sigurno skladištenje i upravljanje ključevima kako bi se izbegle neželjene okolnosti.

Dakle, ako Alice želi da pošalje jedan *bitcoin* Bobu, mora potpisati transakciju koristeći svoj privatni ključ i poslati ga čvorovima na mreži. Rudari, koji znaju njen javni ključ, zatim će proveriti uslove transakcije i potvrditi autentičnost potpisa. Nakon što se potvrdi valjanost, blok koji sadrži tu transakciju biće spreman za finalizaciju od strane validatora/rudara.

3.3. Šeme potpisa koje se koriste u blockchainu

Šema potpisa koja se trenutno koristi u *bitcoinu* je ECDSA. U poređenju sa RSA, ECDSA koristi kraće ključeve i ima manje računarskih zahteva, a da pritom održava jaku bezbednost. Zamislimo grupu eliptične krive kao konačnu grupu tačaka na krivnoj gde je neku operaciju lako izvesti u jednom smeru, ali je teško izvesti u drugom smeru. Osim toga, ECDSA se oslanja na problem diskretnog logaritma umesto na teškoću primarne faktorizacije radi sigurnosti. *Bitcoin* je imao problema sa implementacijom ECDSA u kojoj su identifikatori transakcije mogli biti izmenjeni promenom potpisa transakcije, što je kasnije popravljeno. Implementacija ECDSA takođe može biti sklona raznim drugim napadima.

Uprkos problemima, ECDSA je godinama generalno dobro služio *bitcoinu*. Međutim, ECDSA nema ključno poželjno svojstvo: ne postoji efikasan način za komprimovanje i proveru potpisa zajedno. Poslednjih godina došlo je do pritiska za prelazak na novu šemu potpisa radi poboljšanja skalabilnosti, efikasnosti i privatnosti *bitcoina*: *Schnorr* potpisi.

Svojstvo linearnosti koje poseduju *Schnorrovi* potpisi ima dve prednosti za *bitcoin*. Prvo, linearnost u transakciji sa više potpisa omogućava potpisnicima da kombinuju svoje javne ključeve u jedan agregatni ključ (agregacija ključeva). *Schnorrovi* potpisi omogućavaju da se lista i broj učesnika sakriju objedinjavanjem javnih ključeva i stvaranjem jedinstvenog objedinjenog potpisa koji se ne razlikuje od normalnog potpisa. Ovo svojstvo bi dodatno smanjilo opterećenje blokova i povećalo privatnost omogućavanjem izgradnje pametnog ugovora, tehnike koja čini složene skripte nerazlučivim od normalnih transakcija.

Schnorrovi potpisi dodatno omogućavaju agregaciju unakrsnog unosa. *Bitcoin* transakcije često imaju mnogo ulaza za koje su potrebni pojedinačni potpisi i mogu zauzeti veliku količinu prostora u bloku. Omogućavaju da se pojedinačni potpisi u transakciji objedine i stoga dopuštaju da se svi ulazi u transakciji predstave jednim potpisom. Ova mogućnost kombinovanja potpisa ostavlja više prostora za podatke o transakcijama u blokovima i procenjuje se da će povećati kapacitet za 20-40%.

4. ZAKLJUČAK

Karakteristike šeme digitalnog potpisa imaju kaskadne efekte na funkcionalnost blockchaina. Kao takav, odabir šeme je fundamentalni deo procesa razvoja protokola. Postoje kompromisi i nijedno rešenje neće ostati zlatni standard. Nove optimizacije za poboljšanje upotrebljivosti potpisa se stalno dešavaju kako bi se poboljšala veličina potpisa, vreme verifikacije ili bezbednost. Gore opisane šeme potpisa postoje decenijama i verovatno će biti prisutne u doglednoj budućnosti. Ipak, neizbežno je da će ih na kraju zameniti novije šeme razvijene iz istraživanja koja se danas sprovode.

5. LITERATURA

- [1] C. Paar and J. Pelzl, “*Understanding Cryptography*”, Springer, 2010.
- [2] W. Stallings, “*Cryptography and Network Security*”, 7th ed., Pearson, 2017.
- [3] S. Vaudenay, “*A Classical Introduction to Cryptography*”, Springer, 2006.
- [4] <https://suredbits.com/introduction-to-schnorr-signatures/>
- [5] <https://bisontrails.co/digital-signatures/>
- [6] <https://101blockchains.com/hashing-and-digital-signature-in-blockchain/>
- [7] <https://www.cryptomathic.com/news-events/blog/how-digital-signatures-and-blockchains-can-work-together>
- [8] <https://www.investopedia.com/terms/b/blockchain.asp>
- [9] A. Lewis, “*The Basics of Bitcoins and Blockchains*”, Mango Publishing, 2018.
- [10] D. Tapscott and A. Tapscott, “*Blockchain Revolution*”, Penguin, 2016.

Kratka biografija:



Ana Mutavdžić rođena je u Šapcu 1997. godine. Osnovne studije na Fakultetu tehničkih nauka, odsek Biomedicinsko inženjerstvo, završila je 2020. godine, a master studije na istom fakultetu, odsek Energetika, elektronika i telekomunikacije, studijski modul Obrada signala, 2021. godine. kontakt: ana.mutavdzic.97@gmail.com

PROJEKTOVANJE BAZNIH STANICA

BASE STATION DESIGN

Miloš Mihic, Željens Trpovski, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Tema ovog rada jeste da poveže teorij-ske osnove radio komunikacija i propagaciju elektromag-netnog talasa u slobodnom prostoru kroz primenu na projektovanje i instalaciju baznih stanica.

Ključne reči: *propagacija, bazna stanica, antena*

Abstract – The topic for this research is to connect teoretical basis of radio communications and propagation of electromagnetic radiation in free space used in design and installation of base station.

Keywords: *propagation, base station, antenna*

1. UVOD

Živimo u vremenu u kojem je svet nemoguće zamisliti bez komunikacije. Od davnina, ljudi zarad boljitka svog života i okruženja u kojem se nalaze stvaraju neophodne mehanizme kako bi obezbedili komunikaciju sa nekim. Kako bi komunikacija bila uspešna, potrebno je da imamo dve strane koje komuniciraju i kanal kojim će da se odvija komunikacija. Nekada je kanal komunikacije bio sam čovek, koji je „prevoznim sredstvom“ karakterističnim za period tog vremena, prenosio informaciju. Taj prenos mogao je da traje danima pa i mesecima.

Poznata je legenda o atinskom maratonskom trkaču Filipidu iz petog veka pre nove ere čiji je zadatak bio da prenese poruku Atinjanima da je njihova vojska savladala Persijance. Komunikacija je obavljena, ali ispostavilo se da ovaj kanal za prenos poruke nije najbolje rešenje. Tehnologije za prenos informacija sve su više napredovale i sada živimo u vremenu kad gotovo trenutno možemo da stupimo u komunikaciju sa nekim ko je na najudaljenijem kraju sveta. Komunikacija može da se odvija žičanim putem i bežično [2,3], što je osnov ovog rada.

2. BEŽIČNA KOMUNIKACIJA

Pod bežičnom komunikacijom podrazumevamo razmenu podataka i informacija između dva ili više entiteta pri čemu je medijum za prenos podataka vazduh. Informacije se kroz vazduh prenose putem elektromagnetnih talasa, koje još nazivamo i radio talasi.

Da bi došlo do komunikacije, potreban je izvor koji emituje signale (predajnik), antena koja predaje emitovane talase i uređaj koji registruje primljene radio talase (prijemnik). Da bi se mogla ostvariti komunikacija između prijemnika i predajnika u bežičnom sistemu

prenosa podataka, prijemnik, predajnik i antene moraju raditi u istom frekventijskom opsegu. Nije moguća komunikacija fizičkih elemenata u sistemu ako nisu u istom frekventijskom opsegu.

2.1. Predajnik

Predajnik je deo sistema koji emituje informaciju. Ovaj deo sistema često se naziva Tx modul ili Tx. To je elektronski uređaj koji stvara električne oscilacije visoke frekvencije, pojačava ih i moduliše korisnim signalom informacije, i preko predajne antene ih zrači u slobodan prostor.

2.2. Prijemnik

Prijemnik je deo sistema koji prima emitovanu informaciju. Ovaj deo sistema se često naziva Rx modul ili Rx. Njegov zadatak jeste da primi od antene poruku prenesenu preko nekog medijuma, demoduliše signal i preda demodulirani signal ka korisniku.

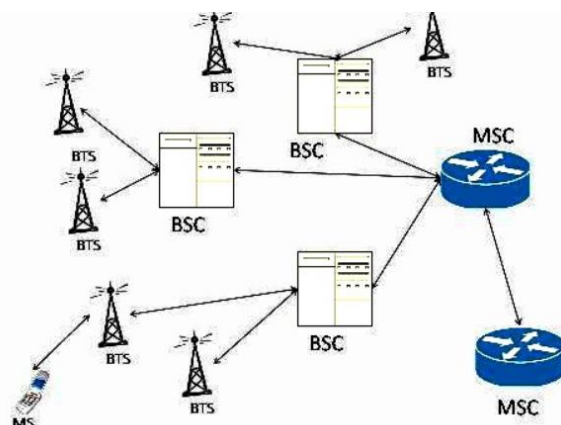
Prijemnici se sastoje od pojačavača, oscilatora, oscilator-nih kola, filtera, demodulatora, itd.

Originalni signal se obično demodulira preko demodu-latora, a koristan signal može biti zvuk kod radio prijem-nika, slika i zvuk kod televizijskog prijemnika i binarni kodovi kod računara.

Radio prijemnik ima ulogu da pojača deo signala koji sadrži željenu informaciju. Električni signal koji iz antene stiže u prijemnik, veoma je male snage i treba ga pojačati i do nekoliko stotina hiljada puta kako bi se dobila snaga dovoljna za njegovu dalju obradu.

2.3. Mobilna mreža

Mobilnu mrežu (Slika 1) karakteriše skup resursa (delova opreme) u okviru operaterove mreže koji omogućavaju komunikaciju krajnjih korisnika usluga, kako između sebe, tako i sa drugim entitetima u mreži (web serveri, internet, razni uređaji itd).



Slika 1. Šema bežične mobilne mreže [2]

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Željens Trpovski, vanr.prof.

Mobilnu mrežu čine korisnici (engl. *User element*), pristupni deo mreže (engl. *radio access network*) i jezgro mreže (engl. *Core network*).

Pristupni deo mreže omogućava korisnicima pristup radio mreži operatera i povezuje krajnjeg korisnika sa ostatkom mreže.

Jezgro mreže povezuje sve delove mobilne mreže i omogućava protok informacija između njih.

3. RADIO TALAS

Radio talasi su vrsta elektromagnetskih talasa koji putuju kroz prostor brzinom svetlosti. Osnovna karakteristika radio talasa jeste frekvencija, koja se meri u hercima (Hz, kHz, MHz i GHz). Područje radijskog spektra obuhvata talasne dužine od milimetra do kilometra odnosno frekvencije od 3 Hz do 300 GHz. Raspodela frekvencija i odgovarajućih opsega, kao i njihova upotreba prikazana je u Tabeli 1 [4].

Radio talasi se veštački generišu pomoću transmitera i primaju se radio prijemnicima, koristeći antene. Različite frekvencije radio talasa imaju različite propagacione karakteristike u Zemljinoj atmosferi; dugi talasi mogu da budu difraktovani oko prepreka, kao što su planine i visoki objekti i slede konturu terena (površinski talasi), kratki talasi mogu da budu reflektovani od jonosfere i vraćaju se na zemlju izvan horizonta, dok se talasi mnogo kraćih talasnih dužina veoma malo savijaju ili bivaju difraktovani i putuju duž vidnog polja, tako da su njihova propagaciona rastojanja uglavnom ograničena na vizuelni horizont [4].

4. PROPAGACIONI MEHANIZMI U VHF I UHF [1]

4.1 RASTOJANJE

Svaki radio talas gubi snagu što se više odvaja od svog izvora i ti gubici su poznati kao gubici u slobodnom prostoru. U praksi, jako retko imamo samo gubitke koji nastaju u slobodnom prostoru, a to je slučaj jedino između jako usmerenih antena sa čistom putanjom bez prepreka u blizini linka. Uglavnom su ti gubici praćeni i dodatnim slabljenjima koja nastaju od brojnih refleksija i prepreka koje se nalaze na tom putu.

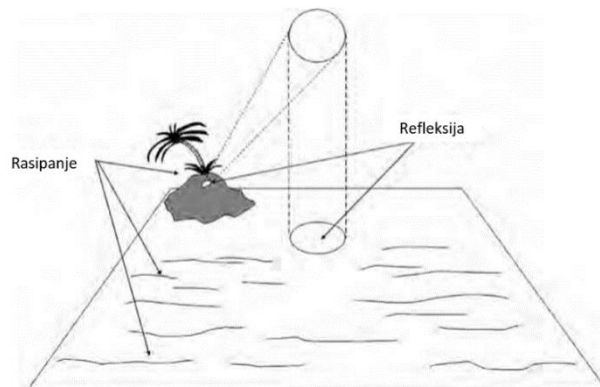
4.2 REFLEKSIJA

Refleksija je pojava koja nastaje kada talas naiđe na glatku ravnu površ. U većini slučajeva površina nije ravna, ali zbog mnogih faktora može da se aproksimira kao ravna površina. Najidealniji primer je transmisija talasa preko mirnog mora, ali postoje i brojne druge situacije. Refleksija stvara drugu putanju između predajnika i prijemnika i ta dva talasa će dovesti do interferencije na prijemu.

4.3 RASIPANJE

Rasipanje nastaje isto kao i refleksija, tačnije i koristi se izraz refleksija, samo talas nailazi na površ koja nije glatka pa se refleksija ne odvija organizovano i u određenom pravcu, nego se rasipa na više strana. Mnogo češća pojava nego sama refleksija, i možemo je vizuelno ilustrovati. Ako bismo posmatrali sunce koje zalazi iza jezera mogli bismo da vidimo jasnu refleksiju njega o

glatku površinu jezera. Međutim iz drugog ugla ta refleksija može da stigne do našeg oka i preko talasa na vodi, raznih glatkih predmeta u okolini jezera i mnogih drugih površina koje ga okružuju kao što je prikazano na Slici 2. Slično se dešava i sa signalom, a njegova interpolacija na prijemu je potpuno nepredvidiva i stohastička i model propagacije je izuzetno kompleksan.

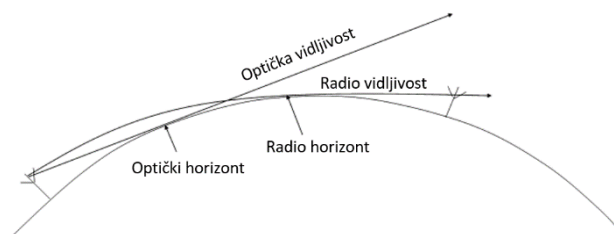


Slika 2. Ilustracija refleksije i rasipanja [1]

4.4 REFRAKCIJA

Kao rezultat ovog propagacionog fenomena zvanog refrakcija, dolazi do zakrivljenja elektromagnetnog talasa. Nastaje zbog varijacija u indeksu refrakcije na različitim nivoima atmosfere.

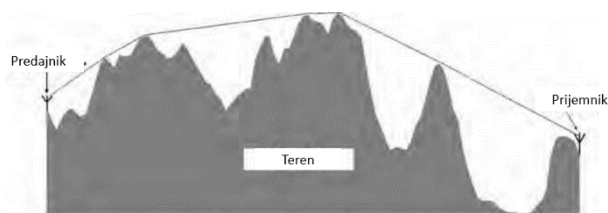
Za zemaljske mobilne komunikacije, najvažnije područje jeste ono koje je relativno blizu površine Zemlje. Indeks refrakcije se smanjuje pri povećanju visine zbog niskog atmosferskog pritiska. Upravo ovaj indeks dovodi do zakrivljenja radio talasa na dole, omogućavajući prijem signala čak i preko prividnog horizonta, ne oslanjajući se na efekte difrakcije (Slika 3).



Slika 3. Refrakcija radio talasa [1]

4.5 DIFRAKCIJA

U većini situacija putanja kojom putuje signal od bazne stanice do mobilnog prijemnika je ispresecana brojnim neravninama i uzvišenjima tako da nemamo optičku vidljivost od predajnika do prijemnika. U tim situacijama signal stiže do prijemnika difraktovan od objekata na tom putu kao što je prikazano na slici 4.



Slika 4. Difrakcija nastala usled neravnog terena [1]

Tabela 1. Elektromagnetski spektar radio talasa [4]

Naziv opsega	Skraćenica (engleska)	Frekvencija	Talasna dužina	Upotreba
		< 3 Hz	>100 000 km	vojna upotreba
Ekstremno niske frekvencije	ELF	3-30 Hz	100 000km - 10 000 km	komunikacije sa podmornicama
Super niske frekvencije	SLF	30-300 Hz	10 000 km – 1000 km	komunikacije sa podmornicama
Ultra niske frekvencije	ULF	300-3000 Hz	1000 km – 100 km	podzemne komunikacije - rudnik
Vrlo niske frekvencije	VLF	3-30 kHz	100 km – 10 km	podmornice, geofizika, nadzor medicinskih uređaja
Niske frekvencije	LF	30-300 kHz	10 km – 1 km	navigacija, AM radio
Srednje frekvencije	MF	300-3000 kHz	1 km – 100m	AM radio
Visoke frekvencije	HF	3-30 MHz	100 m - 10 m	radio - amateri
Vrlo visoke frekvencije	VHF	30-300 MHz	10 m – 1 m	FM radio, televizija, avioni
Ultra visoke frekvencije	UHF	300-3000MHz	1 m – 100 mm	televizija, mobilni telefoni, avijacija
Super visoke frekvencije	SHF	3-30 GHz	100 mm – 10 mm	Mikrotalasna peć, avijacija, radar

Gubici koji nastaju prilikom difrakcije, mogu znatno da se razlikuju od signala koji putuje u slobodnom prostoru i veoma su važni za određivanje propagacionog modela.

4.6 APSORPCIJA

Brojni građevinski objekti, vegetacija, saobraćajna vozila i ljudi apsorbiraju određen deo radio energije. Što je više objekata prisutno na propagacionom putu talasa to su uticaji tih objekata na radio talas veći.

5. LOKACIJA

Lokacija ili sajt obuhvata jednu, ili više baznih stanica koje su postavljene u neposrednoj blizini. Mogu da koriste istu prostoriju ili deo zgrade, koje su montirane u iste montažne ormane, ili kontejnere i koriste isti antenski stub. Po vrsti bazne stanice lokacije delimo na *outdoor lokacije*, u kojima se kabinet bazne stanice i antenski sistem nalaze u spoljašnjem prostoru i *indoor lokacije*, u kojima se ove komponente nalaze u zatvorenom prostoru.

Osnovni elementi svake lokacije su:

- Antenski stub sa gromobranskom zaštitom,
- Bazna stanica (RBS / BTS / Node B),
- Antenski sistem (antene sa antemskim kablovima i ostalim delovima sistema),
- Sistem za napajanje strujom (AC/DC struja, ispravljači, baterijski backup, napajački kablovi),
- Sistem prenosa (link, optika, LAN kablovi),
- Instalacioni materijali (čelični nosači, rostovi, kanalice).

Antenski stub predstavlja čelični element na svakoj lokaciji i koristi se za montažu antena, radio modula i ostalih delova antenskog sistema. Osnovne vrste stubova koje se sreću na lokacijama su rešetkasti stubovi, igla stubovi, cevasti stubovi i “pauk” stubovi.

Pored ovih postoji i posebna vrsta stubova koji se namenski prave, kako ne bi narušili izgled okoline, a zovu se maskirani stubovi. Svaki tip stuba ima odgovarajuću primenu u zavisnosti od toga šta želi da se postigne njegovom upotrebom, bilo to veća visina, nosivost, itd.

Jedna od najbitnijih komponenti na antenskom stubu jeste gromobranska zaštita. Ovi stubovi su najčešće najviša tačka u okolini i zbog toga su izloženi čestim udarima groma, koji bi mogli da unište opremu i ugroze živote ljudi na lokaciji. Njihova prvenstvena uloga je da zaštite ljude na lokaciji u slučaju udara groma, a zatim da zaštite opremu.

Bazna stanica (RBS, BTS) je deo RAN sistema čija je funkcija da obezbedi komunikaciju korisničke opreme (mobilnih telefona, laptopova, tableta itd.) sa ostatkom mreže. Njena uloga je da primi radio signal koji stiže od korisnika, pretvori ga u digitalni signal i prosledi ga ostalim delovima RAN-a, i obrnuto. Bazna stanica opslužuje jednu ili više ćelija (sektora). Osnovni funkcionalni delovi bazne stanice su kabinet, radio moduli, sistemski moduli, kablovi, napajački deo i pomoćna oprema.

Antene su pasivni elementi sistema čiji je zadatak da primaju i predaju radio signal. Karakteristika antene je ugao zračenja, koji određuje pod kojim uglom u odnosu na osu zračenja antene ona emituje signal koji nije slabiji od 3 dB u odnosu na najjači signal antene. Antene koje se najčešće koriste su omni, sektorske i usmerene antene. Omni antene imaju kružno zračenje po horizontalnoj ravni.

Zračenje je jednako usmereno u svim pravcima, tj. ravnomerno emituju signal u krugu od 360 stepeni oko sebe. S obzirom da mogu da prime signal iz svih pravaca, koriste se za pristupne tačke u mreži. Sektorske antene, ili panel antene, imaju horizontalni ugao zračenja od 180, 120, 90 i 75 stepeni. Iako pokrivaju užu sektor od omni antena, on je i dalje dovoljno širok i može da prihvati veliki broj klijenata.

Usmerene (direkcionane) antene zrače radio talas u veoma uskom spektru i tako postižu mnogo veću dobit u odnosu na prethodne antene. Imaju cilj da spoje klijenta sa jednom jedinom tačkom, a to je pristupni čvor mreže. Što je užu ugao zračenja, to joj je veća dobit i signal je bolji, ali pokrivenost je mala što je glavna mana ovog tipa antene. Isto važi i za vertikalni ugao zračenja.

Antenski kabl povezuje antenu sa baznom stanicom. Kraći tipovi kabla koji se koriste kada je radio modul montiran blizu antene zovu se džamperi, a u situacijama u kojima se koriste duži kablovi, fideri. To je slučaj kada je radio modul montiran pored kabineta. Fider prečnika 1/2" se koristi za trase kraće od 10m; 7/8" za trase do 60m, a 5/4" za trase duže od 60m.

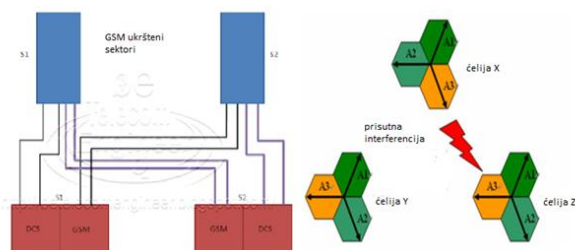
Konektori se prave na krajevima fidera i omogućavaju spajanje fidera sa kablom koji je direktno povezan sa antenom ili radio modulom (džamper). Najveći gubici u antenskom sistemu nastaju upravo zbog loše napravljenih konektora, tako da je njihova izrada veoma važna za kvalitet antenskog sistema.

5.1 MEHANIČKI PROBLEMI U ANTENSKOM SISTEMU

Ove probleme povezujemo sa osnovnim elementima antenskog sistema: antena, kablovi i konektori.

Problemi sa antenskim kablom najčešće nastaju prilikom loše montaže, bilo pričvršćivanjem za hengere ili povezivanjem sa konektorom. Uglavnom je to veza džamper-konektor. Korozija kablova zbog loše hidroizolacije ili oštećenja džampera (fabrički ili na lokaciji) dovode do gubitaka u sistemu.

Najčešći tip greške kod instalacije jeste ukrštanje antenskih kablova. Nastaje isključivo nepažnjom montera, i nekad ja veoma teško ustanoviti zato što sistem radi, ali ne daje željene i očekivane rezultate. Ovaj problem nazivamo ukršteni sektori, i pod njim podrazumevamo situaciju u kojoj su antenski kablovi (fider ili optički kablovi) jednog radio sektora, vezani za antenu drugog sektora. Ukrštanje sektora dovodi do interferencije između susednih ćelija, slab ili nikakav prijem signala u sektorima i loš hendover između susednih ćelija što je prikazano na slici 5.



Slika 5. Ukršteni sektori nastali usled loše povezanih kablova

6. ZAKLJUČAK

Kroz ovaj rad predstavljen je osnovni koncept bežične komunikacije. Sve ono što je neophodno da bi komunikacija bila uspešna su predajnik, prijemnik i kanal. Kako je kanal u slučaju ovog tipa komunikacija slobodan prostor, signal koji putuje kroz njega je često degradiran i oslabljen propagacionim mehanizmima koji se javljaju na tom putu.

Slabljenja nastaju usled većeg rastojanja prijemnika i predajnika, gde signal slabi sa povećanjem, ne samo rastojanja, nego i frekvencije. Refleksije, rasipanje, refrakcija, difrakcija i apsorpcija su prateće pojave koje dodatno utiču na slabljenje signala.

U sistemima radio komunikacija, od izuzetnog značaja je način projektovanja baznih stanica. Prvo se kroz rad na terenu upoznaje sa opremom koja se koristi, načinom montiranja i ukazuje se na greške koje mogu da se dese pri instalaciji bazne stanice.

7. LITERATURA

- [1] Mobile Radio Network Design in VHF and UHF Bands, Adrian W. Graham, Nicholas C. Kirkman and Peter M. Paul
- [2] Modern Wireless Communication, Simon Haykin and Michael Moher
- [3] Principles of Electronic Communication Systems, Louis E. Frenzel, Glencoe/McGraw-Hill. ISBN 9780-02-800409-9
- [4] [https://en.wikipedia.org/wiki/Radio_wave#:~:text=Radio%20waves%20are%20a%20type,gigahertz%20\(GHz\)%20and%20below.](https://en.wikipedia.org/wiki/Radio_wave#:~:text=Radio%20waves%20are%20a%20type,gigahertz%20(GHz)%20and%20below.)

Kratka biografija:



Miloš Mihić rođen je u Novom Sadu 1996. god. Srednje obrazovanje stekao u Gimnaziji "Isidora Sekulić" na prirodno-matematičkom smeru. Osnovne akademske studije iz oblasti Energetike, elektronike i telekomunikacija upisao 2015. godine, a diplomirao 2019. Iste godine upisuje master studije iz oblasti Telekomunikacionih sistema.

kontakt: mmihic4869@gmail.com



Željko Trpovski rođen je u Rijeci 1957. godine. Doktorirao je na Fakultetu tehničkih nauka 1998. god. Oblast interesovanja su telekomunikacije i obrada signala.

Kontakt: zeljen@uns.ac.rs

Zahvalnica:

Izradu ovog rada pomogao je Fakultet tehničkih nauka u Novom Sadu, Departman za energetiku elektroniku i telekomunikacije, u okviru projekta pod nazivom: "Istraživanja u oblasti energetike, elektronike, telekomunikacija i primenjenih informacionih sistema u cilju modernizacije studijskih programa".

**IMPLEMENTACIJA FAZI LOGIKE U UPRAVLJANJU SISTEMOM PRERADE VODE
METODOM REVERZNE OSMOZE****IMPLEMENTING A FUZZY LOGIC CONTROLLER IN A REVERSE OSMOSIS WATER
TREATMENT PLANT CONTROL**

Nikola Pavković, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – Tema ovog rada jeste implementacija fazi regulatora u upravljanju procesom prerade vode.

Ključne reči: *Reverzna osmoza, fazi regulator, sistemi automatskog upravljanja.*

Abstract – *The topic of this paper is implementation of fuzzy logic algorithm for water treatment plant process control.*

Keywords: *Reverse osmosis, fuzzy controller, automatic control systems.*

1. UVOD

Voda je osnovni preduslov za život sveg živog sveta na Zemlji. Zbog izuzetnih osobina voda je bila, i biće, predmet izučavanja mnogih nauka. Poslednjih godina Zemlja – 'Plava planeta' se nažalost suočava sa problemom ograničenosti resursa pijaćih voda, te će takvi resursi imati odlučujući uticaj u budućem razvoju ljudskog društva. U savremenoj industriji gotovo da nema procesa u kom se ne koristi voda [1].

Da bi se izbegla kriza nedostatka vode u budućnosti, potrebno je racionalnije i efikasnije iskorišćavanje iste, kao i ulaganje u razvoj novih metoda za preradu i prečišćavanje vode. Rešenje za ovakav problem jeste filtracija, kako pijaćih voda tako i otpadnih voda koje se vraćaju u prirodu.

U ovom radu biće opisana tehnologija i načini upravljanja automatskim sistemima fabrike za preradu vode koja radi po jednom od najnaprednijih principa filtriranja – reverzne osmoze, proizvodeći vodu za potrebe hlađenja termoelektrane.

2. PROCES REVERZNE OSMOZE

Reverzna osmoza je jedan od najefikasnijih i najboljih načina za prečišćavanje vode. Reverzna osmoza je proces prerade vode koristeći polupropusne membrane pomoću kojih se iz vode odstranjuju neorganske materije, minerali, soli i ostale nečistoće u cilju poboljšanja izgleda, ukusa i ostalih svojstava vode. Na taj način se dobija kvalitetna voda za piće koja ispunjava sve standarde. Sam postupak osmoze pominje se prvi put još 1748. godine, a

opisao ga je Jean-Antoine Nollet. Narednih 200 godina ovaj postupak se radio samo u laboratorijama, da bi 1949. godine krenula istraživanja na polju filtracije vode. U početku ovakvi sistemi su bili dosta neefikasni. Najveća ograničenja dolazila su iz pravca materijala i načina proizvodnje samih membrana, da bi kasnije njihovim razvojem ovi sistemi ispunili svoj pun potencijal. Početkom 90-ih ova tehnologija nalazi svoju primenu i prave se mnoga postrojenja za preradu vode bazirana na reverznoj osmozi [2].

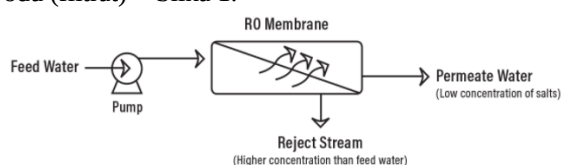
Osmoza je prirodni proces i predstavlja difuziju molekula rastvarača kroz polupropusnu membranu koja propušta molekule rastvarača, ali ne propušta molekule rastvorene supstance. Osmoza se javlja kada su koncentracije rastvora sa dve strane membrane različite.

Pošto polupropusna membrana propušta samo molekule rastvarača, ali ne i rastvorene supstance, dolazi do difuzije rastvarača kroz membranu tj. molekuli rastvarača prolaze kroz membranu kako bi se izjednačile koncentracije sa obe strane membrane. Molekuli rastvarača se kreću iz sredine sa manjom koncentracijom u sredinu sa većom koncentracijom rastvorenih supstanci.

Pri difuziji molekula rastvarača kroz membranu dolazi do porasta nivoa tečnosti na jednoj strani membrane, onoj u kojoj je veća koncentracija rastvorenih supstanci i smanjenja nivoa tečnosti na drugoj strani. Porast nivoa tečnosti na jednoj strani stvara hidrostatički pritisak koji se suprotstavlja osmozi.

Kada se ova dva pritiska izjednače proces difuzije se zaustavlja tj. neće doći do podizanja nivoa tečnosti. Pritisak pri kome se uspostavlja ova ravnoteža naziva se osmozni pritisak [3].

Postupak filtriranja vode reverznom osmozom funkcioniše na sličan način ali sa kretanjem vode u suprotnom smeru. Neprečišćenu vodu treba propustiti kroz osmozne polupropusne membrane pod velikim pritiskom, kako bi se savladao osmozni pritisak. Prljavština, soli, bakterije i virusi u tom postupku ostaju (koncentrat) i zadržavaju se na membranama dok molekuli čiste vode prolaze i na izlazu dobijamo čistu vodu (filtrat) - Slika 1.



Slika 1. Princip reverzne osmoze

NAPOMENA:

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

2. DELOVI UPRAVLJAČKOG SISTEMA

Više ne postoji ni jedna grana industrije koja nije automatizovana. Kao primarni cilj, industrijski sistemi automatskog upravljanja stvaraju mogućnost povećanja obima proizvodnje uz smanjenje troškova i poboljšanje kvaliteta proizvoda i mogućnost povećanja efikasnosti proizvodnje.

2.1. PLC – Programabilni logički kontroler

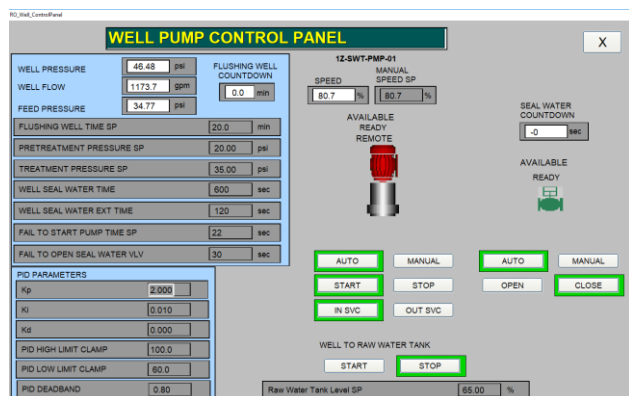
U srcu svakog automatski upravljanog procesa nalazi se PLC kontroler (programabilni logički kontroler). On predstavlja centralni deo upravljačkih automatskih sistema u industriji. PLC je digitalni računar. Njegov program se izvršava ciklično i sastoji se od 3 faze: Čitanje ulaznih promenljivih (čitanje stanja i vrednosti sa senzora), izvršavanje programskog koda i određivanje stanja na izlazu (slanje komandi uređajima poput kontrolera pumpe, ventila, itd.).

2.2. SCADA sistem

SCADA (engl. Supervisory Control And Data Acquisition) predstavlja sistem za praćenje i kontrolu industrijskih sistema. Koristi se za prikupljanje podataka sa senzora i instrumenata lociranih na udaljenim stanicama i za prenos i prikazivanje tih podataka u centralnoj stanici u svrhu nadzora ili upravljanja. Prikupljeni podaci se obično posmatraju na jednom ili više SCADA računara u centralnoj stanici. SCADA sistem u realnosti može da prati i upravlja i do stotinama hiljada ulazno-izlaznih vrednosti.

Uobičajeni analogni signali koje SCADA sistem nadzire jesu nivoi, temperature, pritisci, brzine protoka, brzine motora i slično. Uobičajeni digitalni signali koje SCADA sistem nadzire jesu prekidači nivoa, prekidači pritiska, stanja releja, signali sa motora tj. motornog upravljačkog uređaja, bio to starter ili varijabilni frekventni regulator. SCADA nema potpunu kontrolu nad sistemom, već PLC. Uloga SCADA sistema jeste nadgledanje i nadziranje i predstavlja sponu između PLC-a i operatera koji nadgleda rad fabrike. SCADA sistem obično prezentuje informacije operateru u obliku mimičkih dijagrama.

To znači da operater može da vidi šematsko predstavljanje fabrike ili nekog dela sistema koji kontroliše – slika 2.



Slika 2. SCADA ekran za upravljanje pumpom bunara

3. ALGORITMI UPRAVLJANJA

Postoje razni kriterijumi za klasifikaciju algoritama upravljanja. Ovde ćemo posmatrati klasifikaciju u odnosu na prirodu obrade raznih signala sa ciljem da se dobije upravljački signal.

3.1. PID regulator

Linearni zakoni upravljanja predstavljaju jednu od najstarijih i najrasprostranjenijih upravljačkih strategija. Razlog tome jeste što se pomoću njih može rešiti bar 90% svih upravljačkih zadataka. Tehnike podešavanja parametara ovih zakona upravljanja su razrađene i veoma jednostavne za praktičnu primenu [4].

Linearnim zakonima upravljanja ostvaruju se tri dejstva: proporcionalno (P), integralno (I) i diferencijalno (D), pa se zbog toga i sam regulator naziva PID regulator. Zakon upravljanja PID regulatora se može predstaviti jednačinom (1).

$$U(t) = K_p + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} = K_p(e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de(t)}{dt}) \quad (1)$$

PID regulator ima tri podesiva parametra: pojačanje **K_p**, integralnu vremensku konstantu **T_i**, i konstantu diferenciranja **T_d**. Prisustvo proporcionalnog, integralnog i diferencijalnog dejstva u ovom regulatoru omogućuje dobijanje željenih performansi kao što su: stabilnost, brzina reagovanja, tačnost rada i vreme trajanja prelaznog procesa. Na taj način PID regulator sjedinjuje sva dobra svojstva osnovnih regulatora:

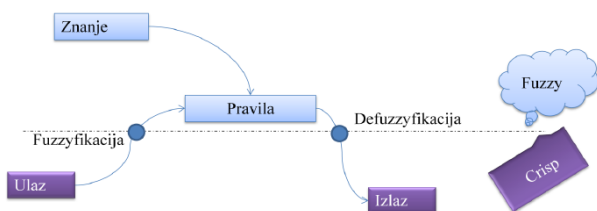
- D – delovanje obezbeđuje brzo reagovanje
- P – delovanje obezbeđuje konstantno dovoljno jako pojačanje
- I – delovanje obezbeđuje tačnost i potpuno otklanja grešku

3.2. Fazi regulator

Fazi algoritam upravljanja je jedan od savremenih metoda upravljanja blizak čovekovom pristupu regulaciji. Fazi upravljanje obezbeđuje formalnu metodologiju za predstavljanje, manipulaciju i implementaciju ljudskog heurističkog predznanja o tome kako kontrolisati jedan, određeni sistem. Ovo ne isključuje razvoj modela procesa jer nam je ovaj model u svakom slučaju potreban za detaljnu simulaciju ponašanja kontrolera u cilju ispitivanja zadovoljenja performansi, stabilnosti sistema kao i za ispitivanje krajnjih ograničenja samog dizajna.

U našem konkretnom slučaju model sistema za razvijanje kontrolera nije potreban, zahvaljujući već postojanju samog sistema, te je regulator pravljen i testiran na realnom sistemu.

Cilj fazi pristupa jeste da, umesto da jezikom matematike pokuša da što bolje reši problem upravljanja sistemom, omogući implementaciju inženjerskog iskustva o procesu u sam algoritam kontrolera. Kako bi izgledala najprostije objašnjeno struktura fazi sistem – slika 3.



Slika 3. Struktura fazi sistema [5]

Fazifikacija modifikuje signale ulaza tako da mogu biti pravilno protumačeni i upoređeni sa pravilima u bazi pravila. Krip (Crisp) signal pretvaramo u adekvatan fazi oblik.

Zaključivanje na osnovu pravila je mehanizam za procenjivanje koja kontrolna pravila su relevantna za trenutno stanje sistema i odlučuje logičkim sklopom kakav će biti upravljački signal, tj. ulaz u proces.

Defazifikacija transformiše fazi oblik u crisp oblik signala, koji je razumljiv procesu.

4. PRIMENA FAZI LOGIKE U UPRAVLJANJU SISTEMOM ZA PRERADU VODE METODOM REVERZNE OSMOZE

Za ovaj konkretan primer, algoritam upravljanja nastao je po želji klijenta da fabrika vode bude apsolutno automatizovana, tačnije da ljudski faktor bude skroz isključen iz procesa upravljanja i donošenja odluka. Naravno ljudski faktor uvek mora da postoji, jer i za potpuno automatizovan proces, u slučaju nepredviđene okolnosti kao što je kvar na sistemu, kontrolu mora da preuzme operater. Praksa je da se od operatera koji nadgleda i upravlja fabrikom očekuje, da zavisno od potrebe za vodom, bira koliko će filtera biti aktivno. Fabrika u ovom primeru sastoji se od tri filtera. Ideja je da se napravi regulator koji će umesto operatera odlučivati, na osnovu nekoliko različitih veličina, kolika je potrebna proizvodnja vode u datom momentu i na osnovu toga koliko filtera je potrebno da bude aktivno.

Dobijena filtrirana voda naravno mora negde da se skladišti. U našem primeru sva dobijena filtrirana voda se šalje u rezevoar čiste vode, čije nam dimenzije nisu poznate, te tako ni zapremina samog rezervoara. Od podataka koje dobijamo jeste stalno merenje nivoa vode u rezervoaru. Iz rezervoara se voda cevima šalje u rashladni toranj elektrane i na taj način se vrši hlađenje tornja. Veličina koja se zadaje i koja se održava jeste nivo vode u rashladnom tornju. Ukoliko elektrana radi većim kapacitetom, nivo vode u rashladnom tornju mora da bude veći, dok ukoliko radi smanjenim kapacitetom nivo vode u tornju treba da bude nešto niži. Sam zahtevani nivo zadaje PLC koji pripada upravljanoj elektrani i on pripada potpuno odvojenom sistemu, nad kojim mi nemamo kontrolu. Na osnovu zadatog nivoa vode u rashladnom tornju, voda se pušta iz rezevoara filtrirane vode kroz modulišući ventil ka tornju i na taj način se održava nivo rashladne vode. Kompletna kontrola održavanja nivoa rashladne vode se obavlja pomoću tog modulišućeg ventila između rezervoara filtrirane vode i rashladnog tornja. Kako u ovom slučaju govorimo u ogromnim

površinama koje se hlade vodom i u kojim se nalazi voda, čak i spoljašnji vremenski uslovi imaju velik uticaj. Leti je potrebno mnogo više vode kako bi se nivo u rashladnom tornju održao. Voda iz rezervoara filtrirane vode ne koristi samo za rashladni toranj, nego se po potrebi koristi i u protiv požarnom sistemu.

Zadatak našeg upravljanja jeste da zavisno o potrebi za filtriranom vodom, odluči koliko je filtera potrebno aktivirati kako bi se zahtev za vodom ispunio i nivo u rezervoaru filtrirane vode održao.

4.1. Definisanje fazi regulatora

Osnovni koraci pri definisanju novog fazi regulatora jesu definisanje ulaza i izlaza, fazifikacija ulaza tj. prevođenje ulaza tako da ih regulator razume, defazifikacija izlaza tj. prevođenje izlaza tako da ih sistem upravljanja razume i kreiranje same tabele pravila po kojoj će regulator raditi tj. donositi odluke.

Tabela pravila pomoću koje ćemo probati da objasnimo arhitekturu našeg regulatora – Tabela 1.

Tabela 1. Baza fazi pravila

Nivo rezervoara	Brzina potrošnje	Stepen aktivnih filtera
Skoro maksimum	Normalna	Stage 0
Visok	Normalna	Stage 1
Srednji	Normalna	Stage 2
Nizak	Normalna	Stage 3
Skoro maksimum	Velika (ekstremna)	Stage 3
Visok	Velika (ekstremna)	Stage 3
Srednji	Velika (ekstremna)	Stage 3
Nizak	Velika (ekstremna)	Stage 3

„Stage“ jeste veličina koja obeležava koliko će aktivnih filtera biti i može da bude u vrednostima 0, 1, 2 ili 3. Stage 0 odgovara slučaju kada ni jedan filter nije aktivan, odnosno nema potrebe za proizvodnjom nove vode. Stage 1 odgovara slučaju kada je potrebno da jedan filter bude aktiviran tj. potrebna količina nove vode može da se postigne koristeći samo jedan filter. Stage 2 ukoliko je potrebno aktivirati dva filtera i stage 3 ukoliko je potrebno aktivirati 3 filtera da bi se postigla dovoljna količina proizvodnje nove vode. Sledeći korak pri kreiranju fazi regulatora jeste definisanje ulaza i izlaza i njihovo prevođenje (fazifikacija) u oblik koji će regulator razumeti.

Ulazni parametri našeg regulatora jesu:

- trenutni nivo rezervoara filtrirane vode – merenje koje dobijamo sa ultrazvučnog mernog instrumenta nivoa,
- granični nivoi vode u rezervoaru – definisanje visine vode u rezervoaru za koje odgovaraju veličine sa kojima radi fazi regulator (maksimu, visok, srednji, nizak),
- brzina rasta ili opadanja nivoa filtrirane vode – u slučaju naglih poremećaja, primera radi požara, dešava se naglo opadanje nivoa vode u rezervoaru i zbog toga algoritam mora da

prepozna ovakav momenat i aktivira sva tri filtera tj. maksimalnu proizvodnju vode,

- vreme dodavanja filtera (stage up time) i vreme oduzimanja filtera (stage down time) – objašnjenje u nastavku teksta.

Kao izlazni parametar našeg regulatora jeste:

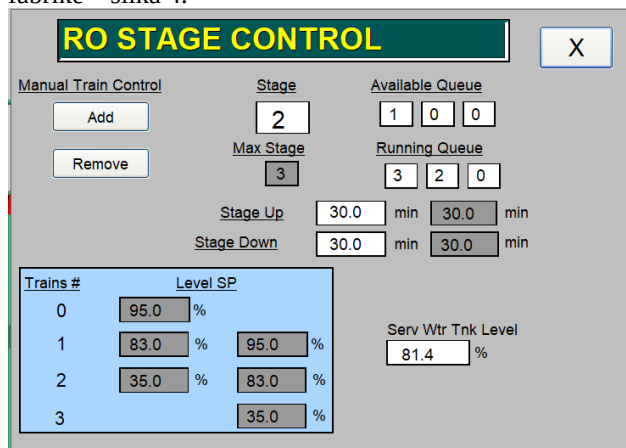
- Odluka koliko će aktivnih filtera da bude, da li ni jedan, da li samo jedan, da li dva ili tri filtera (stage 0, stage 1, stage 2, stage 3)

Pri pokretanju filtera postoji određeno vreme koje je potrebno da se filter pripremi za rad (kvašenje filtera), te vreme za koje će filter početi da proizvodi vodu u svom punom kapacitetu.

Takođe i nakon gašenja filtera potrebno je određeno vreme (ispiranje filtera), te određeno vreme za koje će se filter moći opet startovati. Ovde vidimo da je i sam sistem filtera zapravo izuzetno inertan sistem i našem regulatoru je potrebno da zna ovo. Ukoliko ova inertnost filtera ne bi bila uračunata, desilo bi se da regulator zatraži dodavanje još jednog filtera (stage up), to se ne desi istog trenutka, i zbog toga on zatraži dodavanje još jednog, pa još jednog, a isto tako i za oduzimanje filtera (stage down) i na taj način bi dobili upravljanje koje osciluje i aktivira ili ni jedan ili sva tri filtera.

Jedino se u slučaju naglog opadanja nivoa vode, usled neke vanredne situacije, aktiviraju sva tri filtera istog momenta, kako bi se obezbedila potrebna količina vode. U samom algoritmu fazi kontrolera postoji deo logike koji prati razliku izmerenog nivoa vode kroz vreme i na taj način prepoznaje atipičan slučaj, vanredne situacije potrošnje veće količine vode nego uobičajeno.

Kontrola fazi regulatora i unošenje parametara potrebnih za njegov rad realizovana je pomoću SCADA sistema za upravljanje fabrikom kao odvojen prozor koji se poziva iz glavnog prozora sa parametrima za pokretanje cele fabrike – slika 4.



Slika 4. SCADA prozor za kontrolu fazi regulatora

„Available queue“ predstavlja red dostupnih filtera za naredno aktiviranje kada se stepen uveća. Da bi filter bio dostupan potrebno je da svaki deo njegovog sistema (ventili, pumpa visokog pritiska, itd.) budu u automatskom režimu, da nema signala grešaka i da je sekvenca ispiranja filtera posle rada završena.

„Running queue“ predstavlja red aktivnih filtera, odnosno onih koji se trenutno koriste. Kada se stepen filtera smanji

onaj koji je prvi u redu aktivnih filtera, odnosno onaj koji je najduže aktivan će se ugastiti. U slučaju da se pojavi bilo kakva greška na sistemu filtera ili se neki deo sistema prebaci u ručni režim kontrole, a ne automatski, to će rezultirati da taj filter ne bude više dostupan logici za kontrolu, a samim tim će se i izbaciti iz reda aktivnih filtera i njega će zameniti prvi sledeći dostupan filter, ukoliko je to potrebno i red će biti presložen.

Na ovaj način pomoću sortiranja reda dostupnih i aktivnih filtera omogućeno je da se sekvenca nastavi i ukoliko se desi problem na nekom od tri sistema filtera. Takođe još vrlo bitna stvar jeste da se na ovaj način filteri rotiraju, tačnije postiže se podjednako korišćenje sva tri filtera, što je izuzetno važno kako bi se osmotske membrane podjednako trošile na sva tri sistema filtera. Sortiranje se radi pomoću algoritma „bubble sort“.

Računanje brzine gubitka vode, koje se prosleđuje kao ulazni podatak u regulator i posmatra da ukoliko se poveća iznad određene granice regulator prepozna kao ekstremni slučaj potrošnje vode, nije moguće dodatno podešavati kao parametar preko SCADA sistema zbog sigurnosti.

6. ZAKLJUČAK

Složeni sistemi se ponašaju različito u vremenu i zbog toga je često potrebno koristi više od tradicionalnih načina upravljanja. Fazi upravljanje iako na prvi pogled deluje dosta jednostavno, daje odlične rezultate, pogotovo u upravljanju nelinearnim sistemima. Posebno dobra osobina fazi regulatora jeste što se zapravo ljudska iskustva i razumevanja, u našem slučaju operatera i inženjera o nekom sistemu, mogu preneti u oblik svojstven razumevanju algoritma, pomoću kojih će on upravljati zadatim sistemom.

7. LITERATURA

- [1] “Voda kao resurs budućnosti” – Milica Pajčin, *Fakultet tehničkih nauka Čačak*
- [2] “The early history of reverse osmosis membrane development” – Julius Glater, *Department of Civil and Environmental Engineering, University of California*.
- [3] “Osmosis” - <https://en.wikipedia.org/wiki/Osmosis>
- [4] “Sistemi automatskog upravljanja” – Prof. dr Dušan Petrovački
- [5] “Fazi sistemi kao podrška odlučivanju” – FON, BG

Kratka biografija:



Nikola Pavković rođen je 27.05.1994. godine u Sremskoj Mitrovici. Master rad na Fakultetu tehničkih nauka iz oblasti elektrotehnika i računarstvo odbranio je 2021. godine.

Kontakt:
nikolapavkovic@rocketmail.com

КЛАСИФИКАЦИЈА ТИПА ГРЕШКЕ СЕНЗОРА КОРИШТЕЊЕМ МЕТОДА
МАШИНСКОГ УЧЕЊА

SENSOR FAULT CLASSIFICATION USING MACHINE LEARNING METHODS

Борис Јањић, Факултет техничких наука, Нови Сад

Област – РАЧУНАРСТВО И АУТОМАТИКА

Кратак садржај – У овом раду описани су сензори температуре и најчешће грешке које настају на сензорима. Такође су описане три методе машинског учења које су кориштене при класификацији типа грешке сензора. Кориштене методе су: *k* најближих сусједа, метод потпорних вектора и неуронска мрежа. Приказана је анализа поменутих метода и дато поређење добијених резултата.

Кључне речи: Грешке сензора, машинско учење, класификација, *k* најближих сусједа, метод потпорних вектора, неуронске мреже

Abstract – This paper describes temperature sensors and the most common faults on sensors. Three machine learning methods for classification are also described. Methods that have been used are: *K* Nearest Neighbors, Support Vector Machines and Neural Networks. The analysis of the mentioned classification methods of the sensor fault is presented and a comparison of the obtained results is given.

Keywords: Sensor fault, Classification, Support Vector Machines, *K* Nearest Neighbors, Neural Networks, Machine Learning

1. УВОД

Мјерење је настало као потреба да са природне појаве опишу квалитативним карактеристикама и резултат је опажања и поређења тих појава. Појавом управљања и аутоматизације процеса јавља се потреба да се мјерења величина која нису електрична некако пребаце у електрични облик. Тај проблем се рјешава уз помоћ сензора. Нагли развој сензора почиње у периоду аутоматизације и у компјутерској ери. Данас постоји широк дијапазон сензора од најпростијих давача и биметала до крајње софистицираних сензора који се користе за најпрецизнија мјерења. Због велике употребе сензора велика штета може да настане ако се благовремено не детектује квар на сензору. У овом раду биће ријечи о најчешћим типовима грешака које се могу десити на сензорима.

Такође ће се говорити и о машинском учењу и класификационим методама. Биће ријечи о прикупљању, припреми и обради података са грешкама сензора. Биће приказане анализе претходно поменутих метода машинског учења и резултати њиховом примјеном на улазне податке.

НАПОМЕНА:

Овај рад произтекао је из мастер рада чији ментор је био др Велимир Чонградац, ред. проф.

2. СЕНЗОРИ

Сензор је уређај који претвара мјерену физичку величину у електрични сигнал, одликује се малим димензијама, изузетним техничким карактеристикама и способношћу обраде сигнала [1].

Значај сензора се огледа у прикупљању података о објекту или процесу зарад одређених анализа, дијагностике, пројектовања, управљања. Примјена им је веома широка у свим инжењерским и многим другим областима.

2.1. Сензори температуре

Температура је једна од основних седам физичких величина SI (међународни систем јединица) система. Сензори температуре према изведби могу бити: термоелементи, експанзиони, отпорнички, полупроводнички, сензори инфрацрвеног зрачења.

2.2. Грешке сензора

Одступања од очекиваног понашања на излазу сензора се сматрају грешкама. Због утицаја многобројних фактора, чије се дејство не може контролисати, услови у којима раде системи се непрекидно мијењају. Зато резултат мјерења увијек садржи грешку.

Аутори у радовима [2], [3] и [4] наводе сљедеће грешке сензора као најчешће: грешка дрифта, грешка биаса, грешка деградације прецизности, грешка пика и грешка заглављења.

Нека је $s(t) = h(t) + \eta$ очекивано понашање сензора без присуства грешке, гдје је $h(t)$ излазна вриједност сензора у времену t , и $\eta \sim N(0, \delta_\eta^2)$ је шум. На основу ове једначине моделоваћемо једначине горе наведених грешака.

Дрифт је промјена статичке карактеристике сензора у условима употребе у току дужег временског периода

$$S(t) = h(t) + \eta + b(t), b(t) = b(t-1) + \vartheta, \vartheta = \text{const} \quad (1)$$

Грешка биаса (eng. *Bias Fault*) се моделује тако што се на излаз сензора дода нека константа. Ова грешка се јавља због хардверских проблема као што су кратак спој или грешка калибрисања. Моделује се:

$$S(t) = h(t) + \eta + \vartheta, \vartheta = \text{const} \quad (2)$$

Грешка деградације прецизности (eng. *Precision Degradation Fault*) се јавља услед високо фреквентних шума или физичког оштећења сензора. Математички модел:

$$S(t) = h(t) + \eta + \vartheta \vartheta \sim N(0, \delta_\vartheta^2) \delta_\vartheta^2 \gg \delta_\eta^2 \quad (3)$$

δ_y^2 представља варијансу високо фреквентног шума и показује да она мора бити вишеструко већа од δ_n^2 варијансе шума сензора који смо у стању да коригујемо.

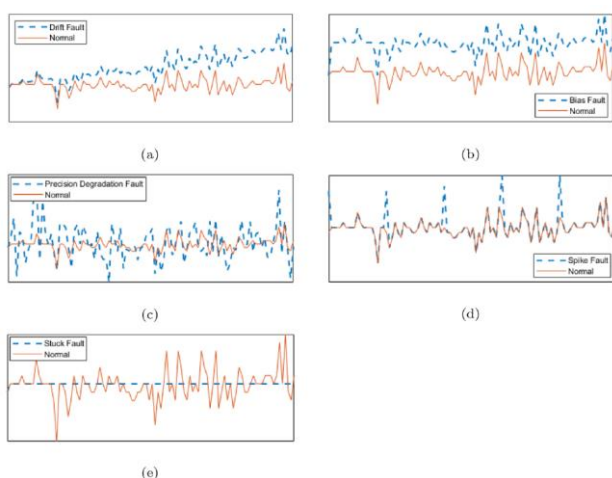
Грешка пика (eng. *Spike Fault*) је грешка гдје се на излазу сензора, у константним интервалима, појављују пикови велике амплитуде. Математички је описујемо:

$$S(t) = \begin{cases} h(t) + \eta + \vartheta(t); & \forall t = v \times \tau, v = \{1, 2, \dots\}, \tau \geq 2, \\ h(t) + \eta; & \text{otherwise} \end{cases} \quad (4)$$

Заглављење (eng. *Stuck Fault*) како само име каже манифестује се тако што на излазу сензора даје константну вриједност. Може да буде привремена или трајна. Моделује се на следећи начин:

$$S(t) = \vartheta, \vartheta = \text{const} \quad (5)$$

На слици 1 је приказано свих 5 грешака сензора које су описане. Поменуте грешке ће бити кориштене у експерименталном дијелу задатка.



Слика 1. Типови грешке сензора: (а) дрифт, (б) биас, (ц) деград. прецизности, (д) пик, (е) заглављење [3]

3. МАШИНСКО УЧЕЊЕ

Машинско учење је област вјештачке интелигенције која се бави изградњом рачунарских система који уче из искуства без експлицитног програмирања.

За компјутерски програм се каже да учи из искуства E (*Experience*), везаног за задатак T (*Task*), и мјеру перформанси P (*Performance*), уколико се његове перформансе на задатку T , мјерене метрикама P , унапређују са искуством E [5].

Алгоритме машинског учења дијелимо на: надгледано учење, ненадгледано учење и учење условљавањем.

Надгледано учење је облик машинског учења који као скуп параметара за тренирање прима улазне податке и жељене очекиване вриједности на излазу. Циљ је добити резултате за необиљежене податке [5].

3.1. Алгоритми класификације

Класификација представља једну од двије основне врсте надгледаног учења.

Класификација је проблем предвиђања категоричке циљне промјенљиве. Категоричким се сматрају промјенљиве које узимају коначан број вриједности међу којима нема уређења [5]. Конкретно у овом раду

је ријеч о класификовању типова грешке сензора. Скуп грешака је дат у поглављу о грешкама, а резултати ће бити изнесени у поглављу везаном за имплементацију.

Када излаз модела може имати више од двије класе онда говоримо о *multiclass* класификацији.

У наставку ће бити више ријечи о алгоритмима који су кориштени у практичном дијелу рада: k најближих сусједа, методу потпорних вектора и неуронским мрежама.

3.1.1. К најближих сусједа

К најближих сусједа (eng. *K Nearest Neighbors* - KNN) је вјероватно најједноставнији алгоритам машинског учења који се може користити и за класификацију и за регресију. Основна претпоставка овог алгоритма је постојање растојања над простором атрибута. Најчешће се претпоставља векторска репрезентација инстанци и еуклидско растојање, али могуће су и општије претпоставке [6].

3.1.2. Метод потпорних вектора

Метод потпорних вектора (eng. *Support Vector Machine* - SVM) је један од важнијих метода машинског учења. SVM је иницијално креиран као класификациони проблем двије класе. Ова метода се проширује на проблем више класа (eng. *multiclass*), а самим тим постаје све занимљивија тема истраживања [7].

Заснована на геометријској интуицији, ова метода тражи најбољу раздвајајућу хиперраван [6].

3.1.3. Неуронске мреже

Неуронске мреже (*neural networks* - NN) се убрајају међу најпопуларније и најпримјењеније методе машинског учења. Њихове примјене су многобројне и помјерају домете вјештачке интелигенције, рачунарства и примјењене математике. Неке од њихових примјена су класификација текста и слика, аутономна возња, препознавање говора, и слично. Неуронске мреже заправо представљају параметризовану репрезентацију која може да послужи за апроксимацију других функција. Проналажење одговарајућих параметара врши се математичком оптимизацијом неког критеријума квалитета апроксимације [6].

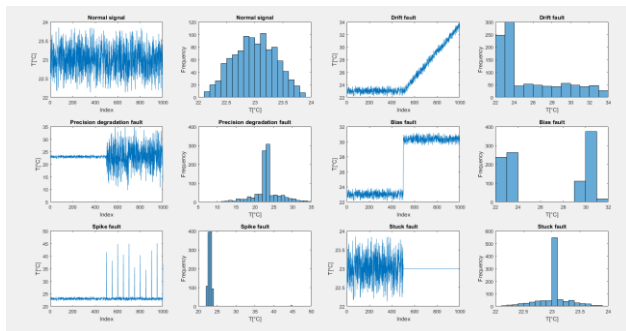
4. КЛАСИФИКАЦИЈА ГРЕШКЕ СЕНЗОРА

Идеја овог задатка јесте да се симулирају најчешћи типови грешака на сензорима и да се класификују. Циљ је да се подаци обраде и прикажу анализе методама машинског учења. Конкретно ће бити обрађене методе KNN, SVM и NN и биће приказани резултати.

4.1. Прикупљање и припрема података

Процес прикупљања података и припреме за класификационе методе је веома сличан приступу у радовима [2], [3] и [4]. Мјерења су изведена користећи *Arduino Uno* и сензор температуре и влажности DHT22, с тим да је акценат био на мјерењу температуре. Мјерења су прикупљена користећи серијску комуникацију између рачунара и *Arduino*

Uno микроконтролера. Извршено је 100 мјерења по 1000 тачака на сензору који нема никакве грешке у раду. Мјерења су била прилично чиста, пошто у просторији гдје је вршено мјерење није било неке веће промјене температуре. Ради симулирања шума у реалним системима, додат је шум у нормалан сигнал. Наредни корак је био да се додају грешке у нормалан сигнал. Додате су грешке описане једначинама (1), (2), (3), (4) и (5) и примјери сваког сигнала су дати на слици 14. Као што се види на сликама, у првих 500 тачака сигнал је нормалан сигнал и није модификован, док је у других 500 тачака додата грешка.



Слика 2. Прикази грешака и њихових хистограма

Послије проширења скупа са уметнутим грешкама, гдје је сваки тип грешке представљен једном класом читав скуп података износи 600x1000. Пошто је скуп од 1000 тачака прилично велик, на основу тих тачака креирано је 12 карактеристика које ће бити кориштене као улазни параметри у класификаторе, па су димензије сета података редуковане на 600x12. једначине које су кориштене су приказане у табели 1, а средње вриједности карактеристика су приказане у табели 2.

Табела 1. Списак једначина коришћених у одређивању карактеристика

Назив	Дефиниција	Назив	Дефиниција
Средња вриједност	$\mu = \frac{1}{N} \sum_{i=1}^N x_i$	Стандардна девијација	$Y_{STD} = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$
Средња квадратна вриједност	$Y_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}$	Варијанса	$Y_{VAR} = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$
Кв. коријен амплитуде	$Y_{SRA} = \left(\frac{1}{N} \sum_{i=1}^N \sqrt{ x_i } \right)^2$	Оштрина врха	$Y_{KV} = \frac{1}{N} \sum_{i=1}^N \left(\frac{x_i - \mu}{Y_{STD}} \right)^4$
Накривљеност	$Y_{SV} = \frac{1}{N} \sum_{i=1}^N \left(\frac{x_i - \mu}{Y_{STD}} \right)^3$	Фактор гребена	$Y_{CF} = \max \left(\frac{ x_i }{Y_{RMS}} \right)$
Фактор импулса	$Y_{IF} = \max \left(\frac{ x_i }{\frac{1}{N} \sum_{i=1}^N x_i } \right)$	Гранични фактор	$Y_{MF} = \max \left(\frac{ x_i }{Y_{SRA}} \right)$
Фактор форме	$Y_{FF} = \frac{Y_{RMS}}{\mu}$	Од пика до пика	$Y_{PPV} = \max(x_i) - \min(x_i)$

4.2. Методологија евалуације

Модел је увијек потребно процијенити на новим и непознатим подацима. У фази обуке моделу дајемо улазне податке и очекивану вриједност излаза и на основу тога се модел обучава. У фази евалуације модела користи се други сет података, а као улаз у модел користе се само улазни подаци. На основу тих података модел одређује излаз система. Добијене вриједности се пореде са очекиваним и добија се

тачност модела. У овом раду кориштене су следеће метрике перформанси: прецизност (*Precision*), одзив (*Recall*), Ф1 скор (*F1 score*) и тачност (*Accuracy*).

Табела 2. Средње вриједности карактеристика за сваки тип грешке

Карактеристика	Нормални	Дрифт	Деградиација прецизности	Бисас	Пик	Заглављење
μ	22.9994	29.0187	23.0059	26.6089	23.1924	22.9998
Y_{STD}	0.4030	7.7988	3.5599	3.6331	1.9266	0.2850
Y_{RMS}	23.0029	30.2908	23.2796	26.8875	23.2725	23.0015
Y_{VAR}	0.1625	85.0076	12.6816	15.5133	3.7278	0.0813
Y_{SRA}	22.9976	28.4289	22.8627	26.4674	23.1627	22.9989
Y_{KV}	2.5210	2.4035	5.0826	1.1041	98.0002	5.0326
Y_{SV}	-0.0003	0.8623	0.0065	-0.0036	9.4265	-0.0038
Y_{CF}	1.0460	1.5278	1.5495	1.1590	2.0129	1.0447
Y_{IF}	1.0461	1.5964	1.5679	1.1710	2.0199	1.0448
Y_{MF}	1.0462	1.6314	1.5778	1.1772	2.0225	1.0448
Y_{FF}	1.0002	1.0397	1.0119	1.0101	1.0035	1.0001
Y_{PPV}	2.1263	25.5666	26.0597	9.2622	24.9120	2.0593

4.3. Анализа експеримента

За имплементацију класификатора кориштен је програмски језик *python 3.7* верзија и кориштена је библиотека *scikit-learn*. Циљ је да се упореде резултати класификације следећих метода KNN, SVM и NN. Ради вјеродостојности резултата за сваку метода користи се исти скуп за тренирање и исти скуп за тестирање. Извршена су три експеримента над подацима:

- Први експеримент је био над чистим подацима, гдје су улази у сваки од класификатора били сирови подаци. Сваки класификатор имао је 1000 улазних параметара.
- Други експеримент је био извлачење најбољих параметара од 1000 задатих. За одабир 10 и 50 најбољих представника од 1000 је кориштен *SelectKBest* селектор. Улаз у класификаторе је био величине 10 и 50 карактеристика.
- За трећи експеримент су употријебљене претходно припремљене карактеристике. Једначине карактеристика су дате у табели 2. У овом случају је узиман варијабилни број карактеристика од 5 до 12.

Скалирање података доприноси томе да нека од карактеристика не буде значајно доминантнија од друге па су подаци прескалирани користећи *MaxAbsScaler*. Од укупног броја података (600) приказаћемо анализе гдје је 20%, 40% и 60% укупног броја података одвојено за тестирање модела.

4.3.1. К најближих сусједа

Имплементиран користећи *KNeighborsClassifier*. Метрике су рађене на основу параметра К и величине тестног скупа. Експериментално је утврђено да је најбољи избор за параметар К=9.

3.1.1. Метод потпорних вектора

За имплементацију ове методе кориштен је *SVC*. За потребе ових експеримената биће кориштени *linear*,

poly и *rbf* типови *kernela*. Пошто се ради о *multiclass* класификацији изабрана је и *one-vs-one* стратегија, која дијели *multiclass* проблем у један бинарни задатак класификације за сваки пар класа.

4.3.3. Неуронске мреже

За моделовање неуронске мреже кориштен је *MLPClassifier*. Употријебљени *solver* је *adam*, а као активациона функција је узета *relu*. Неуронска мреже је састављена из три слоја, улазног, излазног и скривеног слоја. Улазни слој неуронске мреже зависи од броја карактеристика, што ће у првом експерименту бити 1000, у другом 10, а у трећем редом 5, 8, 10 и 12. Излазни слој једнак је броју класа и у свим експериментима износи 6. Кориштен је предефинисани број неурона у скривеном слоју и он износи 100.

4.4. Резултати

Кренувши од првог експеримента, који је као улазне параметре узимао сирове податке, видимо да је тачност за KNN 52-55% у зависности од параметра K, за SVM је 55-80% у зависности који *kernel* је кориштен, а за NN класификатор је тачност 70-72% у зависности од величине скупа за обучавање. Видимо да сваки од класификатора са прецизношћу од готово 100% класификује грешке дрефта, пика и биаса. Битна ствар за први експеримент је и то што сваки класификатор има проблем да класификује три класе нормалан сигнал, грешку деградације прецизности и грешку заглављења. Једино је SVM *rbf* успио да ријеша проблем грешке деградације прецизности, док су друге двије класе остале неријешене.

Због осредње тачности и проблема класификације првенствено горе наведене три проблематичне класе, предложен је други експеримент гдје је узето 50 и 10 најбољих карактеристика из сировог сигнала, а остале су одбачене. Над KNN класификатором је показано како је класификатор са 10 тачака показао боље перформансе. Када упоредимо резултате KNN, SVM и NN класификатора можемо видјети да се тачност повећала.

Тачност KNN класификатора је скочила са 55% на 84%, а дјелимично је ријешен проблем класификовања проблематичних класа, док су за остале 3 класе перформансе остале исте. За SVM класификатор је слична ситуација као за KNN класификатор, перформансе су поправљене, посебно за *kernel poly* гдје је тачност скочила са 55% на 91-93%. Класификација по критичним класама се знатно поправила у односу на претходни експеримент. Код NN класификатора у овом експерименту није било пуно напретка, тачност је скочила са 72% на 78%, с тим да проблем класификовања нормалног сигнала и закочења није ријешен.

Пошто је у другом експерименту одбачен огроман број тачака, од 1000 тачака сигнала кориштено је само 10, тако је огроман потенцијал који имају остале тачке остао неискориштен. Због тога су кориштене математичке једначине из табеле 1 да се направи 12 карактеристика на основу сировог сигнала.

У трећем експерименту се показало да се са таквим карактеристикама, небитно да ли их узмемо само 5 или 12, добија тачност од 99-100%. Такође је битно напоменути да повећање обучавајућег скупа са 40% на 80% не утиче битно на перформансе добијених резултата.

5. ЗАКЉУЧАК

У првом експерименту је показано је да су метрике које се добију из чистих података задовољавајуће за неке класе грешка као што су дрефт, биас и пик, док за друге грешке то није случај. Показано је и да се перформансе могу поправити ако се из цијелог скупа података извуку најбољи представници, као што је урађено у другом експерименту. Тај приступ се показао добрим јер су перформансе на свим свим методама скочиле.

Најбоље се показао метод потпорних вектора са тачношћу од 93%. У трећем експерименту је показано да је могуће искористити све вриједности сигнала за израчунавање одређених карактеристика на основу којих ће се обучаваати класификатори, а као резултат тога перформансе свих класификатора износе 99-100%.

Потенцијална унапређења и даљи кораци би били да се користе прави подаци са реалним грешкама, и да се размотре перформансе постојећих модела. Још један приједлог за даљи развој био би да се смањи број карактеристика за обучавање модела са задржавањем истих или побољшањем перформанси.

4. ЛИТЕРАТУРА

- [1] M. Popović „Senzori u robotici“, Viša elektrotehnička škola, Beograd, 1996
- [2] S. U. Jan, I. Koo, „A Novel Feature Selection Scheme and a Diversified-Input SVM-Based Classifier for Sensor Fault Classification“
- [3] S. U. Jan, Y. D. Lee, I. S. Koo, „A distributed sensor-fault detection and diagnosis framework using machine learning“
- [4] S. U. Jan, Y.-D. Lee, J. Shin, and I. Koo, „Sensor fault classification based on support vector machine and statistical time-domain features“
- [5] B. Nikolić, D. Drašković „Nastavni materijali iz predmeta Inteligentni sistemi“, ETF, Beograd 2021.
- [6] M. Nikolić, A. Zečević „Mašinsko učenje“, Matematički fakultet, Beograd 2019.
- [7] Y. Zhang „Support Vector Machine Classification Algorithm and Its Application“

Кратка биографија:



Борис Јањић рођен је у Требињу 1994. године. Факултет техничких наука уписао је 2013. године. Дипломски рад из области Електротехника и рачунарство – Аутоматика и управљање системима одбранио је 2017. године.

BESKONTAKTNI SISTEM ZA PRANJE POD VISOKIM PRITISKOM ZA SAMOSLUŽNU PERIONICU

CONTACTLESS HIGH-PRESSURE WASHING SYSTEM FOR SELF-SERVICE WASHING

Predrag Kovačević, *Fakultet tehničkih nauka, Novi Sad*

Oblast: PRIMENJENA ELEKTRONIKA

Kratak sadržaj: U ovom radu predstavljeno je tehničko rešenje i realizacija projekta beskontaktnog sistema za pranje pod visokim pritiskom za samouslužnu perionicu.

Ključne reči: PLC, beskontaktni sistem, automatizacija, senzori, akuatori, upravljanje, perionica

Abstract: This paper presents the technical solution and implementation of the project of contactless high-pressure washing system for self-service washing.

Keywords: PLC, contactless system, automation, sensors, accumulators, control, washing

1. UVOD

U ovom radu dato je tehničko rešenje i realizacija projekta beskontaktnog sistema za pranje pod visokim pritiskom za samouslužnu perionicu. Glavna motivacija za izradu ovog rada bila je praktične i finansijske prirode. Firma Telix, u kojoj je projekat realizovan, uvidela je potrebu za samostalnom samouslužnom perionicom radi uštede novca i mogućnosti namenskog prilagođavanja sistema svojim potrebama. Sva postojeća komercijalna rešenja su koncipirana tako da rade po principu tokena, njihovo postrojenje zauzima mnogo prostora i vrlo ih je teško kofigurisati prema specifičnim korisničkim potrebama jer su uglavnom namenjeni za pranje automobila. Firmi Telix je ovo postrojenje prevashodno bilo potrebno za pranje mašina, konvejera i službenih automobila. Dostupno je malo literature o ovoj konkretnoj temi, zbog čega je izučavana literatura o zasebnim komponentama projekta (PLC, senzori, aktuatori i dr), što je omogućilo uspešnu realizaciju ovog projekta. U daljem tekstu biće prikazano rešenja projekta po celinama: projektni zadatak, radni ciklus perionice, realizacija sistema i zaključak.

2. PROJEKTI ZADATAK

Sistem treba da se sastoji od: ormana za kontrolu i upravljanje sistemom, korisničkog interfejsa, pumpe visokog pritiska, pumpe za vodu, rezervoara za skladištenje vode sa sapunicom, tj. deterdženta, daljinske komande, uređaja za zagrevanje vode (protočni bojler), izlaza za sudoperu i odgovarajućih ventila i propratne opreme.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Jovan Bajić, vanr. prof.

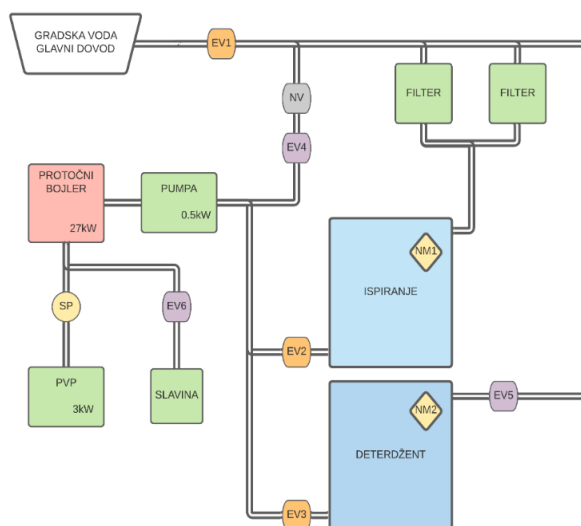
Korisnički interfejs sistema treba da sadrži određene tastere za upravljanje sistemom: „glavni ventil“, „dopuna ispiranje“, „dopuna deterdžent“, „stop dopuna“, „bojler“, „gradska voda“, „deterdžent“, „ispiranje“, „sudopera“, „pauza“ i „stop“, čija funkcija je detaljno opisana u radu. Svi tasteri treba da imaju i odgovarajuću svetlosnu indikaciju, da bi korisnik znao kada je taster aktivan.

Treba da postoje četiri režima rada sistema: pranje gradskom vodom, pranje deterdžentom, ispiranje i „sudopera“. Predviđen je sledeći način pranja:

1. Skidanje krupne prljavštine koristeći pumpu visokog pritiska i gradsku vodu,
2. Natapanje objekta koji se pere deterdžentom koristeći pumpu visokog pritiska,
3. Pauza od 3-4 minuta dok deterdžent za bezkontaktno pranje nagrize prljavštinu, i
4. Spiranje sapunice filtriranom vodom (ispiranje).

Potrebno je programirati automatiku sistema koja će obezbediti rad svih komponenti na željeni način u odgovarajućim režimima rada.

Blok šema sistema je prikazana na slici 1, a detaljniji opis rada režima, kao i pojedinačnih tastera dat je u radu.



Slika 1. Blok šema projektovanoq sistema

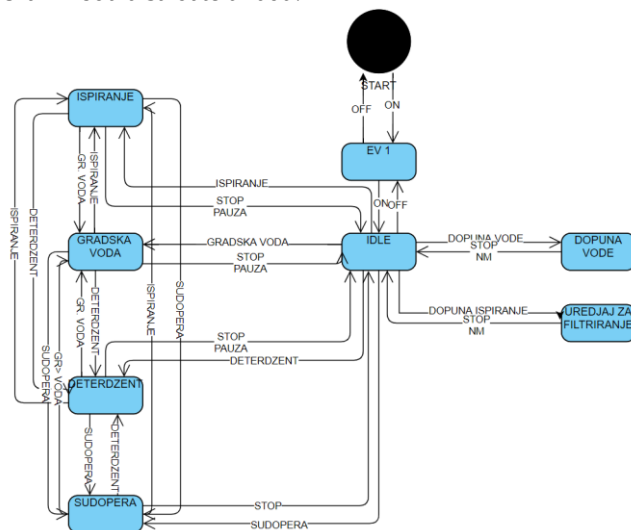
3. RADNI CIKLUS PERIONICE

Da bi se odredile vrste i potreban broj senzora, kao i napisao program za PLC, mora se definisati radni ciklus, tj. način kontrole ciklusa [1,2,3].

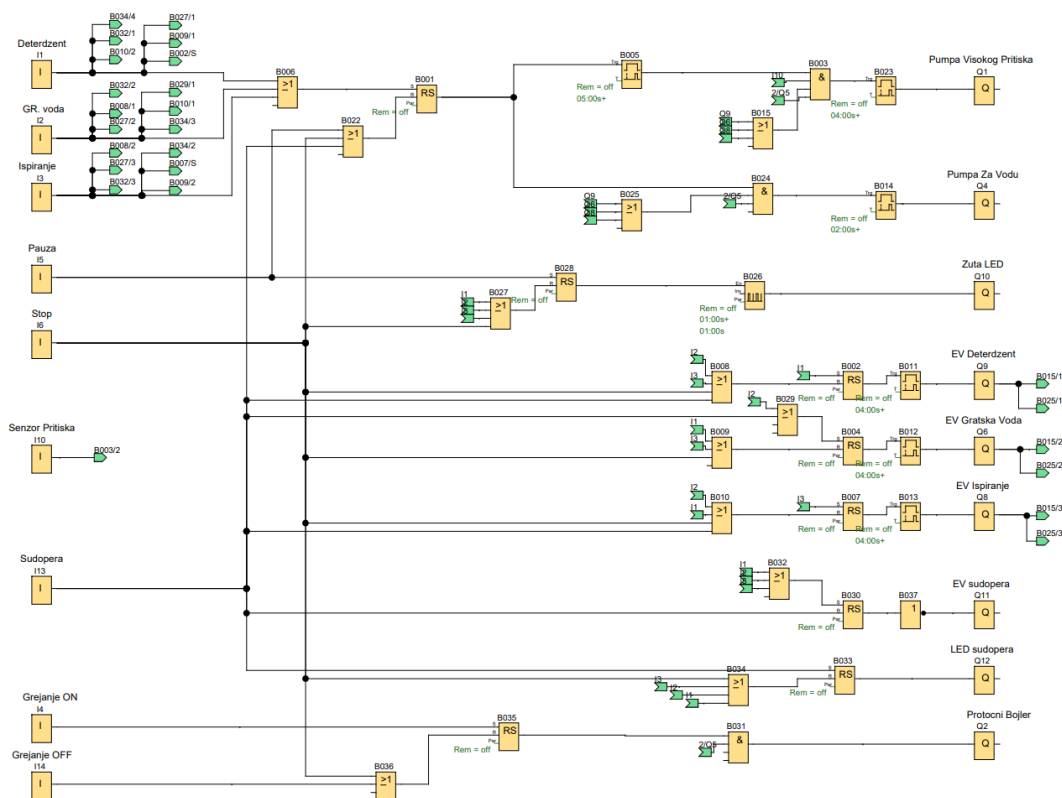
Radni ciklus se sastoji iz više režima rada koji su pomenuti u prethodnom poglavlju, a pritiskanjem odgovarajućih tastera, sistem prelazi iz jednog režima rada u drugi. Na sl. 2 prikazan je dijagram mašine stanja [4,5]. U radu je takođe dat detaljan pregled stanja aktuatora i senzora za odgovarajuće modove rada.

Zaključeno je da je PLC najbolje rešenje za upravljačku jedinicu u projektu. Izabran je PLC proizvođača Siemens, oznake LOGO PLC 12/24, sa osam ulaza i četiri izlaza, zbog dostupnosti modela na lageru firme Telix, u kojoj je projekat realizovan. Zbog nedostatka dovoljnog broja ulaza, dodata su dva modula oznake Siemens LOGO DM8 12/24P, sa četiri ulaza i četiri izlaza. Osnovne karakteristike korišćenog PLC-a i modula su date u radu.

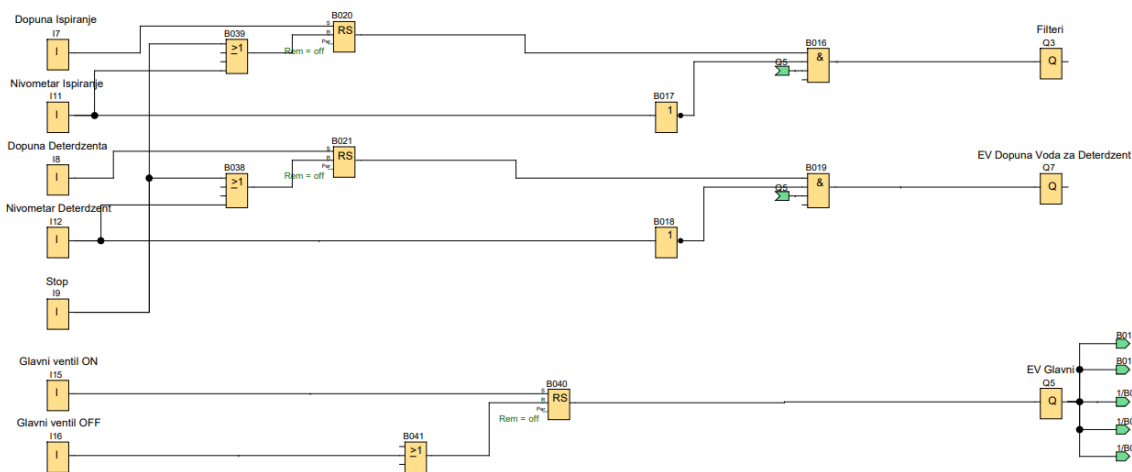
Softver koji služi za pisanje programa za Siemens LOGO familiju programabilnih logičkih kontrolera je LOGO! Soft Comfort. Softver podržava dva načina pisanja programa: funkcijski blok dijagram i lestvičast dijagram, a postoji i opcija automatskog prebacivanja jednog oblika dijagrama u drugi. U rešavanju ovog projekta korišćeni su funkcijski blokovi za kreiranje algoritma rada. Na slikama 3 i 4 prikazan je funkcijski blok dijagram programa. Blokovi od I1 do I6 su ulazni signali, a blokovi Q1 do Q12 izlazni signali. Svi blokovi funkcijskog dijagrama, kao i logika rada detaljno su opisani u radu.



Slika 2. Dijagram mašine stanja



Slika 3. Funkcijski blok dijagram

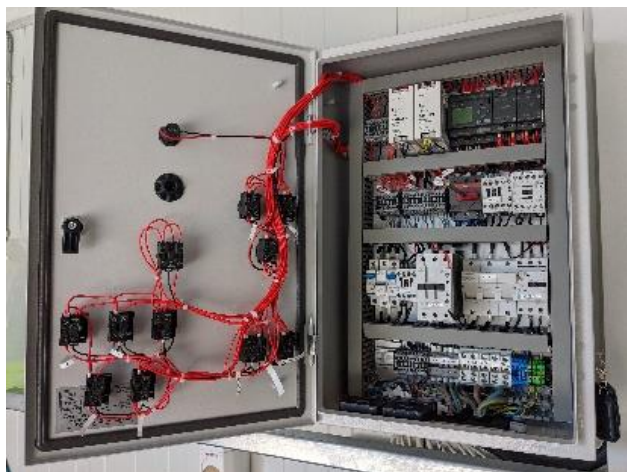


Slika 4. Funkcijski blok dijagram

4. REALIZACIJA SISTEMA

Da bi se obezbedila robusnost, vodonepropusnost i jednostavnost instalacije, sva elektronika je smeštena u elektroorman (slika 5). Orman sadrži upravljački i energetski deo. U upravljački deo spadaju: PLC, pomoćni releji, tasteri, sklopke, svetla za indicaciju, RF prijemnik... Energetski deo čine: ispravljači, osigurači, releji, kontaktori, elektroventili...

Na vratima ormana se nalaze indikacione sijalice, tasteri za kontrolu sistema i grebenasta sklopka. Tasteri su jasno označeni i lako pristupni korisniku. Svi potrebni signali sa senzora i upravljački signali za aktuatora dovode se do ili od elektroormana i priključuju na PLC i odgovarajuća napajanja.



Slika 5. Unutrašnjost ormana

Bilo je potrebno implementirati i osigurače na svakoj fazi protočnog bojlera, prekidače za ispravljače, kao i prekidač sa motornom zaštitom, za zaštitu trofaznog motova pumpe visokog pritiska. Takođe su implementirani i različiti kontaktori za: uređaj za filtriranje vode i pumpu za vodu; pumpu visokog pritiska; i protočni bojler. Za napajanje senzora, indikacionih sijalica i PLC-a, korišćeni su ispravljači AC-DC. Konkretni modeli svih korišćenih komponenti, kao i njihove karakteristike navedene su u radu.

Korisnički interfejs, tasteri i svetlosna indicacija, su predviđeni da stoje na ormanu sa elektronikom. Korišćeni su tasteri sa integrisanim LED indikatorima. Raspored tastera i njihove funkcije mogu se videti na slici 6.



Slika 6. Korisnički interfejs

Takođe je obezbeđena i mogućnost daljinskog upravljanja sistemom, pomoću RF prijemnika sa daljinskim upravljanjem. U radu je data električna šema sistema.

Za rešavanje ovog projektnog zadatka korišćeni su i senzori za merenje pritiska i nivometri. Za izradu samouslužne perionice bilo je neophodno odrediti pritisak na ulaznoj cevi pumpe visokog pritiska. Izabran je senzor LF703 proizvođača LEFOO zbog opsega merenja i dostupnosti. Dodatno, po dva nivometra su ugrađena u svaki od rezervoara, radi kontrole nivoa tečnosti u rezervoarima. Senzori su vezani paralelno i nalaze se jedan iznad drugog, na rastojanju od 3cm, čime se obezbeđuje robusnost sistema. U slučaju otkazivanja jednog od senzora, drugi će obezbediti da ne dođe do prelivanja tečnosti van rezervoara. Odabrani su senzori oznake NIVH-PF, proizvođača Nivostat.

Implementirani su sledeći aktuatori u okviru sistema: uređaj za filtriranje vode, pumpa visokog pritiska, pumpa

za povećanje pritiska vode, protočni bojler i elektroventili.

Uređaj za filtriranje vode implementiran je radi poboljšanja čistoće vode i, posledično, poboljšanja kvaliteta pranja. Prisustvo kamenca u gradskoj vodi (kalcijum i magnezijum karbonata i bikarbonata, kao i drugih soli rastvorenih u vodi) predstavlja problem za pranje i naročito je primetno pri pranju automobila u samouslužnim perionicama, kada se nakon pranja pojave bele fleke na površini. Iz tog razloga, implementirana su dva uređaja za filtriranje vode, oznake RO-400G, proizvođača Amanda.

Pumpa visokog pritiska služi za proizvodjenje visokog pritiska sa malim protokom na pištolju za pranje, a odabrana je pumpa proizvođača Karcher.

Pumpa za povećanje pritiska vode implementirana je radi povećanja pritiska tečnosti iz rezervoara na ulazu u pumpu visokog pritiska, jer pritisak na ulazu u pumpu visokog pritiska mora biti minimum dva bara, kako bi se obezbedio kontinualan, neometan rad. Odabrana je pumpa firme Grundfos, oznake Scala 2.

Protočni bojler služi za brzo grejanje velikih količina tečnosti, a topla voda u sistemu je potrebna jer prilikom bilo kakvog odstranjivanja prljavštine, svi deterdženti i rastvarači imaju bolje performanse u toploj vodi. Pranje toplom vodom pogotovo je korisno zimi. Odabran je protočni bojler firme Clage, oznake DCX next.

Elektroventili korišćeni su zbog kontrole protoka različitih tečnosti u sistemu. U projektu su implementirane dve vrste elektroventila: soleniodni elektroventili i električni kugličasti ventili.

Principi rada korišćenih aktuatora, kao i tehničke specifikacije odabranih komponenti, nalaze se u radu.

5. ZAKLJUČAK

Zadatak ovog rada bio je projektovanje i realizacija beskontaktnog sistema za pranje pod visokim pritiskom za samouslužnu perionicu. Problem je detaljno razmotren i upoređena su različita rešenja i implementacije. Na kraju je izloženo konačno rešenje, kao i problemi sa kojima se autor susreo prilikom rešavanja. Za rešenje je iskorišćen Siemens PLC serije LOGO. Autor je iskoristio šest elektroventila, dva nivometra, senzor pritiska, pumpu za vodu, pumpu visokog pritiska, kao i svu prpratnu elektroniku za realizaciju elektroormana, koja je opisana u radu. Elektroorman, koji je napravljen prilikom realizacije ovog rada, ima mogućnost lake zamene i dodavanja novih elemenata.

Glavno unapređenje ovakvog rešenja bi bilo rešavanje problema sporog ispiranja sapunice iz creva i cevi kada se sistem prebaci iz moda rada sa deterdžentom u mod rada ispiranja ili gradskom vodom.

Trenutno je potrebno dvadesetak sekundi da bi se sapunica isprala novom, „čistom“ vodom iz sistema (cevi i creva).

6. LITERATURA

[1] Ljiljana Živanov, „*Primena senzora i aktuatora deo senzori*“, UNS, Novi Sad, 2008.

[2] Документација LOGO PLC

<http://w3.siemens.com/mcms/programmable-logic-controller/en/logic-module-logo/logo-software/pages/default.aspx>

[3] Dr Božo Ilić „*Senzori i Aktuatori*“, Visoko tehnička škola strukovnih studija, Novi Sad 2019.

[4] Reversi osmoza <https://quenchwater.com/blog/what-is-reverse-osmosis-and-how-does-it-work/>

[5] <https://pressurewashr.com/pressure-washer-pumps/>

Kratka biografija:



Predrag Kovačević je rođen u Šapcu 1994. godine. Master rad iz oblasti Elektrotehnike i računarstva – Energetska elektronika i električne mašine odbranio je 2018. godine na Fakultetu tehničkih nauka. Zaposlen je u firmi Telix od 2019. godine.
Kontakt: pedjawork@gmail.com

**PRECIZNOST BIOHEMIJSKIH ANALIZATORA I IMUNOANALIZATORA U
MEDICINSKIM LABORATORIJAMA****THE ACCURACY OF BIOCHEMICAL ANALYZERS AND IMMUNOANALYZERS IN
MEDICAL LABORATORIES**Ivan Lažetić, Platon Sovilj, *Fakultet tehničkih nauka, Novi Sad***Oblast – ELEKTROTEHNIKA I RAČUNARSTVO**

Kratak sadržaj – U ovom radu je prikazan postupak provere preciznosti rada laboratorijskih analizatora u okviru automatizovanog procesa ispitivanja humanog biološkog materijala u laboratorijskoj dijagnostici. Radi postizanja maksimalne pouzdanosti rezultata ispitivanja, neophodno je nadgledanje svih aktivnosti i elemenata analitičkog procesa.

Ključne reči: laboratorijski analizator, preciznost, pouzdanost rezultata

Abstract – This paper presents the procedure for establishing the accuracy of laboratory analyzers within the automated process of testing human biological materials in laboratory diagnostics. To achieve the maximum reliability of test results, it is necessary to monitor all activities and elements of the analysis process.

Keywords: laboratory analyzer, precision, reliability of the results

1. UVOD

Laboratorije su nezamenjiv deo medicinske dijagnostike i kliničke medicine. Nezavisno od toga, koja vrsta laboratorijske analize je u pitanju, za nesmetano funkcionisanje celokupnog, veoma složenog sistema, neophodna je potpuna usklađenost svih resursa.

Nadzor sistema vrše, pored stručnjaka medicinske struke i eksperti iz raznih nemedicinskih profesija, uglavnom inženjera biomedicine, biohemije, elektrotehnike i računarstva.

Sinhronizacija svih segmenata postupka se postiže potpunom automatizacijom analize humanog biološkog materijala, uz upotrebu fabrički izrađenih reagenasa i kalibratora koji obezbeđuju veliku tačnost i pouzdanost. Nove tehnike molekularne dijagnostike, pojava biočipova, kao i nove generacije sekvencioniranja, takođe doprinose efikasnom radu laboratorija. Korišćenjem moderne, automatizovane opreme, dijagnostičke procedure postaju brže, a kvalitet pružanja usluga ujednačeniji.

Kontinualna provera ispravnosti rada analizatora, tačnosti i preciznosti merenja, kao i ispravnog tumačenja rezultata su osnovni preduslovi za nesmetan rad celokupnog

laboratorijskog sistema i donošenja ispravnih odluka pri lečenju pacijenata.

Cilj ovog rada jeste da se predstavi postupak koji se primenjuje pri proveru preciznosti rada laboratorijskih analizatora u procesu analize humanog biološkog materijala u laboratorijskoj dijagnostici, koja se sprovodi u potpuno automatizovanom okruženju. Ispravnost rada svakog segmenta automatizovane linije unutar laboratorije doprinosi produktivnosti rada, kvalitetu izvođenja i tumačenja testova, a time i smanjenju pogrešno donetih dijagnoza.

U okviru rada, na konkretnom primeru je prikazana procedura testiranja analizatora sa aspekta preciznosti. Testiranje je izvršeno na integrisanom sistemu biohemijske analize i imunoanalize, pod nazivom Alinity ci, kompanije Abbott Laboratories.

2. LABORATORIJSKE ANALIZE

U mnogim složenim zdravstvenim institucijama se u okviru jedne integrisane laboratorije odvijaju različite dijagnostičke procedure, od hematoloških, seroloških, bakterioloških, hemostaznih do virusoloških, endokrinoloških, imunometrijskih, radioizotopnih analiza sve do klasičnih metoda medicinske biohemije, koje su ujedno i najzastupljenije. Da bi se dijagnostički proces nesmetano odvijao, od zahteva lekara prema laboratoriji do dobijanja traženog izveštaja o izvršenim merenjima, neophodno je obezbediti nesmetan rad laboratorijskih uređaja i efikasnu komunikaciju sa različitim jedinicama Klinika i Instituta preko savremenih informacionih sistema [1].

Efikasnost rada laboratorija se naglo povećala, uvođenjem laboratorijske automatizacije, pri kojem jedan ili više laboratorijskih analizatora izvršava određen broj analiza, pri čemu je broj laboratorijskog osoblja sveden na minimum. Laboratorijska automatizacija se postepeno usavršavala razvojem tehnologije i korak po korak preuzimala pojedine faze kompletne laboratorijske analize. U laboratorijskoj automatizaciji se desio veliki iskorak u trenutku kada su proizvođači laboratorijske opreme, pojedine standardizovane delove analizatora međusobno povezali pomoću pokretnih traka. Priključivanje kompletnog sistema na računarsku mrežu je uokvirio ceo sistem u jednu zatvorenu, funkcionalnu celinu.

Laboratorijski uređaji, nezavisno od vrste uzoraka koje analiziraju, su veoma kompleksne celine, sastavljene od većeg broja različitih podsistema. Mnogi od njih treba da budu priključeni na kompletnu infrastrukturu (električnu

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Platon Sovilj, red. prof.

energiju, vodu, kanalizaciju) ali i da predstavljaju deo modernog informatičkog sistema. Da bi laboratorijske analize mogle nesmetano da se izvršavaju, timovi inženjera raznih profila neprestano nadgledaju i prate proces rada svakog segmenta celokupnog sistema i po potrebi intervenišu u otklanjanju njihovih disfunkcija.

Elektroinženjeri su zaduženi za instalaciju i osposobljavanje novih analizatora za rad, obezbeđivanje ispravnosti rada elektromehaničkih celina, senzora, aktuatora, napajanja kao i besprekornog funkcionisanja mrežne komunikacije. Takođe, njihov zadatak je da isprate održavanje kvaliteta analitičkih postupaka, kao i ispravnost rada korišćenih metoda i analizatora.

Svaka faza automatizovane laboratorijske analize je značajna, pri čemu je centralni deo svakako očitavanje uzoraka. U procesu automatske analize uzoraka, koristi se nekoliko tehnika očitavanja uzoraka, od kojih su najrasprostranjenije *fotometrijska, potenciometrijska i hemijsko-svetlosna analiza* [2].

3. MERENJA U MEDICINSKOJ BIOHEMIJI

Osnovni zadatak svih metoda medicinske biohemije je da se odredi sastav ili koncentracija određenih materijala u biološkim uzorcima, sa ciljem da se otkriju poremećaji koji se javljaju kao posledica disfunkcije organizma, tj. bolesti. Hemijski sastav zdravog organizma je stalan, pošto ga mnogi regulacioni mehanizmi drže u homeostaznim okvirima. Kada organizam oboli, skoro pri svim bolestima dolazi do promena u hemijskom sastavu telesnih tečnosti, koje se raznim metodama medicinske biohemije, relativno lako mogu detektovati.

U svakodnevnoj praksi se sastojci biološkog uzorka određuju i mere uz pomoć određenih reagenasa bez da se izvrši izolacija merenog sastojka (MS). Shematski prikaz analitičkog postupka prikazan je na slici 1.



Slika 1. Shematski prikaz hemijske reakcije reagens - analiziran uzorak

Biološki uzorak, najčešće je to uzorak krvnog seruma koji sadrži veoma velik broj hemijskih sastojaka (MS₁ - MS_n) se dovodi u kontakt sa reagensom koji se spaja samo sa određenim sastojkom (MS_i), dovodeći do određenih promena (najčešće promene boje reagensa) koje se mogu meriti. Ako postoji proporcionalnost između veličine merene promene i količine merene materije taj reagens može da se iskoristi za određivanje njene koncentracije u biološkim uzorcima.

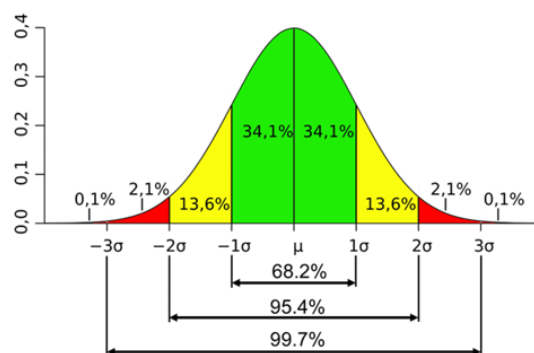
3.1. Interpretacija rezultata biohemijskih merenja

Najveći deo dijagnostičkih testova spadaju u biohemijske testove, koji nakon primene odgovarajućih analitičkih metoda za analizu hemijskog sastava bioloških uzoraka izračunava njihovu koncentraciju i uz primenu kontrole kvaliteta, odgovarajućim statističkim metodama procenjuje dobijeni rezultat.

Osnovno pitanje nakon kvantitativnog merenja i dobijanja rezultata biohemijskih analiza je: da li dobijen rezultat odgovara zdravoj ili bolesnoj osobi? Da bi se dobio odgovor na dato pitanje, vrši se određivanje koncentracije

ispitivane materije istom metodom i sa istim reagensom na velikoj grupi sigurno zdravih osoba. Naravno, skoro nikad se ne može dobiti ista vrednost, pošto se meri pojava koja podleže biološkoj varijabilnosti.

Statistička obrada izmerenih vrednosti grupe zdravih osoba ukazuje da rezultati podležu Gausovoj raspodeli (najveći broj osoba ima vrednosti oko aritmetičke sredine, $\bar{x}$ određene za grupu, a broj pojedinaca sa nižim i višim vrednostima opada). Raspon referentnih vrednosti (RV) predstavlja interval poverenja od tri standardne devijacije, $3\sigma = 3SD$, što znači da se oko 99.7% dobijenih vrednosti nalazi u intervalu od plus-minus tri standardne devijacije, kao što je prikazano na slici 2 [3]. Drugim rečima, dozvoljene su samo tri do četiri greške u procesu rada na milion uzoraka. Tako strog kriterijum provere ispravnosti je rasprostranjen pod nazivom metoda šest sigma (six sigma method) i obezbeđuje visok kvalitet pruženih usluga u svim sferama u kojima se primenjuje [4].



Slika 2. Raspodela rezultata merenja po Gausovoj raspodeli

Srednja vrednost, standardna devijacija i referentna vrednost se izračunavaju pomoću sledećih formula:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \sigma = \left(\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1} \right)^{0.5}$$

$$RV = \bar{x} \pm 3\sigma. \quad (1)$$

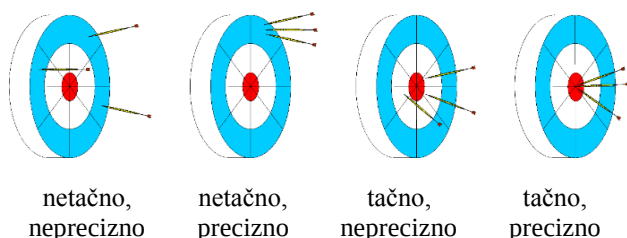
Svaka laboratorija ima definisan raspon referentnih vrednosti kao i pokazatelje *merne nesigurnosti, tačnosti i preciznosti* korišćenih biohemijskih metoda. Merna nesigurnost predstavlja kvantifikaciju sumnje u tačnost rezultata merenja. Što je merna nesigurnost manja, kvalitet mernog rezultata je veći. Tačnost, kao kvalitativan pojam za bliskost slaganja rezultata merenja i stvarne vrednosti nije isti pojam kao i preciznost.

Preciznost ukazuje npr. na varijaciju nekog merenja pri ponovnim određivanjima istog uzorka i predstavlja važan pokazatelj za svakog lekara, pošto mu pokazuje sa kojom sigurnošću može primenom određene metode da prati razvoj patoloških promena ili ozdravljenje pacijenata. Pri maloj preciznosti, porast neke vrednosti može da se tumači kao napredovanje patološkog procesa ali i kao promena do koje dovodi nepreciznost metode.

Preciznost laboratorijske metode se određuje izvršavanjem iste analize istom metodom na istom uzorku. U medicinskoj biohemiji se koriste samo one metode, čija je preciznost ispod 5%.

Tačnost predstavlja meru odstupanja izmerene vrednosti od stvarne vrednosti. Tačnost se u biohemijskim analizama određuje tako, što se izmeri koncentracija biološkog uzorka koji ima deklarisanu koncentraciju tzv. kontrolnog seruma, pa se izrazi odstupanje dobijenog merenja od deklarisanе vrednosti, kao njegov procenat. Tačnost može da ima oba znaka i ne sme da pređe vrednost $\pm 10\%$ [2].

Razlika u pojmovima tačnost i preciznost se najbolje uočava posmatranjem pogodaka strelca u metu, prikazanu na slici 3:



Slika 3. *Tačnost nije preciznost*

Postoje još dva parametra koja ukazuju na kvalitet i upotrebljivost neke biohemijske analize. To su *reproducibilnost*, koja predstavlja meru rasipanja rezultata merenja istog uzorka u različitim terminima, npr. testovi danas-sutra. *Osetljivost* metode ukazuje na najmanju koncentraciju neke materije koju data metoda može da detektuje, a da je ne izjednači sa nulom, sa verovatnoćom većom od 0,95.

3.2. Analizatori iz familije Abbott Laboratories

U svetu postoji velik broj proizvođača medicinske laboratorijske opreme. Jedan od njih je Abbott Laboratories, koji proizvodi širok spektar medicinskih uređaja, u koji spadaju i dijagnostički uređaji pod zajedničkim nazivom ALINITY (ALignment, INnovation, UnITY). Potreba za analizatorima velikih kapaciteta i širokog dijapazona analiza je dovela do pojave kombinovanih analizatora, kao što je i model ALINITY *ci* koji predstavlja integrisan sistem kliničko-hemijske (biohemijske) analize i imunoanalize.

Prilikom instalacije pojedinih modula i njihovom spajanju, treba ispratiti unapred definisane procedure, opisane u internim dokumentima ISA 210-001 za Alinity *c* modul i ISA 205-001 za Alinity *i* modul. Predviđeno vreme kompletne instalacije jednog modula je 21,5 čas.

Pre puštanja u rad analizatora, neophodno je izvršiti sve kalibracije u okviru sistema. Prvo se vrši kalibracija glavnog robota RSM (Reagent Sample Manger) koji prenosi nosače sa uzorcima između aparata i koji dozira i izbacuje reagens. Nakon toga se kalibrišu pipete koje uzimaju deo uzoraka iz epruveta i pipete koje dodaju reagens. U sledećem koraku se vrši kalibracija mešača na Alinity *c* modulu, kalibracija ICT (Integrated Chip Technology) modula i kalibracija zona za pranje i ispiranje testova u Alinity *i* modulu. Neophodno je proveriti temperaturu u hladnjaku za reagens kao i temperaturu u delu gde se vrše testovi u inkubacionom periodu. Za ispravan rad sistema treba da se izvrši i provera ispravnosti pumpi za pranje pipeta i špriceva za aspiraciju reagenasa, a zatim i da se inicira dnevno čišćenje aparata. Potrebno je instalirati i testove za različite analize, podesiti vrednosti kalibratora i

kontrolnih jedinica za analize, postaviti sve module u mod za analizu testova, u takozvani running status, i dodati reagens, koji se očitavaju putem bar koda. Na kraju postupka treba nasuti i kalibracione tečnosti.

Proces kompletne analize na *c* modulu po uzorku traje oko 20 minuta, a na *i* modulu oko 40 minuta. Nakon završene kalibracije, rade se testovi za preciznost; oko 20 uzoraka po svakom nivou kontrolne supstance (niske, normalne i visoke vrednosti). Nakon izvršenih kontrolnih testova i na osnovu dobijenih rezultata, proračunavaju se srednje vrednosti, kao i odstupanja. Na osnovu datih vrednosti, odlučuje se da li je validacija kompletirana, tj. da li je aparat spreman za korisnika [5].

Celokupnim sistemom upravlja softver (Alinity PRO), razvijen u kompaniji Abbott koji povezuje analizator, preko računarske mreže bolnice sa medicinskim osobljem koji tumači rezultate analiza u samoj bolnici ili u nekom drugom udaljenom medicinskom centru. Dat program upravlja i nesmetanim protokom reagenasa, njihovim doziranjem kao i automatskom kalibracijom u unapred definisanim vremenskim razmacima. Ovaj softver omogućava i monitoring rada uređaja preko interne Abbott mreže, dobijajući adekvatne informacije o radu sistema i predikciju potencijalnih zastoja i kvarova.

Ceo softverski upravljački sistem predstavlja deo jedinstvenog IT sistema bolnice, i podleže ISO sertifikatom za bezbedan protok poverljivih medicinskih informacija.

4. REZULTATI

Najvažniji cilj svakog laboratorijskog ispitivanja jeste, da svi dobijeni rezultati budu pouzdani, u skladu sa svim propisanim normativima. Poverenje u kvalitet rezultata je preduslov za stabilan rad svake laboratorije.

Kvalitet analitičke prakse predstavlja skup karakteristika jedne procedure, tj. postupka koji kao izlaznu veličinu daje analitički rezultat koji pokriva sve očekivane i propisane zahteve korisnika tih rezultata. Da bi se proverio kvalitet nekog postupka ili rezultata, neophodno je stalno ponavljanje datog postupka i dodatna matematičko statistička obrada rezultata [6].

Postupak kojim se proveravaju tehničke karakteristike nekog instrumenta, da bi se dobili pouzdani analitički podaci, naziva se *validacija*. Validacija se sprovodi uvek prilikom instalacije novog sistema, nakon svakog servisa ali i na početku svakog radnog dana. Najčešće se proveravaju tačnost i preciznost metode ispitivanja.

Kao primer provere preciznosti rada uređaja iz Alinity *ci* serije, biće prikazan standardni postupak koji se primenjuje u te svrhe. Po protokolu, na početku radnog vremena, za proveru preciznosti rada uređaja vrši se testiranje nad kontrolnim uzorkom. U aparat se postavlja po 20 uzoraka, za sva tri definisana nivoa rezultata (donja granica intervala, srednja vrednost i gornja granica intervala), posebno za biohemijske analize, posebno za imunoanalize i posebno za proveru elektrolita natrijuma, kalijuma i hlora, tj 3×20 provera po tipu analize. Na osnovu rezultata merenja, određuje se srednja vrednost i standardna devijacija po svakom uzorku. Izmerene vrednosti se unose u tabelu (označene oznakom +). Podatke o očekivanoj srednjoj vrednosti i standardnoj

devijaciji obezbeđuje proizvođač uređaja. Za prikazan test, te vrednosti iznose

$$\mu = 25,0000 \text{ mIU/mL} \text{ i } \sigma = 4,5000 \text{ mIU/mL}.$$

Izmerene vrednosti, za jedan set uzoraka, pri proveru donjeg nivoa intervala rezultata, za imunoanalize, prikazane su u tabeli 1.

Tabela 1. Rezultati merenja

Rezultat mIU/mL	-3 σ	-2 σ	-1 σ	μ	+1 σ	+2 σ	+3 σ
25,57				+			
26,12				+			
26,32				+			
26,26				+			
26,51				+			
25,65				+			
26,52				+			
25,24				+			
25,58				+			
25,78				+			
25,23				+			
25,45				+			
25,84				+			
26,18				+			
26,70				+			
25,57				+			
25,26				+			
25,43				+			
25,06				+			
26,30				+			

Srednja vrednost dobijenih podataka, navedenih u tabeli 1 iznosi $\bar{x} = 25,8285 \text{ mIU/mL}$, dok standardna devijacija, ima vrednost $SD = 0,49794 \text{ mIU/mL}$, računata na osnovu formula iz (1).

Na osnovu položaja izmerenih vrednosti (oznake + u tabeli) se vidi, da su sva odstupanja sa istim znakom, u okviru odstupanja do +1 σ , na osnovu čega se donosi odluka a analizator precizno radi i da je spreman za analiziranje pravog biološkog materijala za taj dan.

Svakodnevno praćenje izmerenih vrednosti i tendencije odstupanja obezbeđuje blagovremeno reagovanje na otklanjanje neispravnosti rada analizatora i obezbeđivanje ispravnosti rezultata laboratorijskih analiza.

5. ZAKLJUČAK

Današnja savremena medicinska istraživanja se velikim delom oslanjaju na rezultate analiza sprovedenih na automatizovanim sistemima laboratorijskih analizatora. Osnovni i najbitniji parametar svakog laboratorijskog analizatora je njegova sposobnost da daje pouzdane rezultate, koji treba da budu transparentni i dostupni širem krugu zainteresovanih strana.

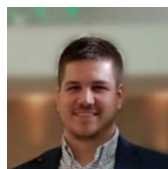
Opšte prihvaćena metoda provere performansi laboratorijskih uređaja je široko prihvaćena šest sigma metoda, koja predstavlja metodologiju kontrole kvaliteta dugi niz godina.

Na primeru provere rada analizatora Alinity ci, prikazan je postupak donošenja odluke o preciznosti rada uređaja i njegovog bezbednog korišćenja pri sprovođenju imunoloških analiza, koji se na isti način može primeniti i na drugim testovima iz posmatrane grupe analizatora.

6. LITERATURA

- [1] F. Dujmović, M. Đerić, Z. Stošić, "Laboratorijska medicina danas", *Med Pregl* 2007, LX (11-12), pp. 543-548, 2007.
- [2] J. Borota, "Praktikum medicinske biohemije i hemije", Novi Sad, Univerzitet u Novom Sadu, Medicinski fakultet, 2011.
- [3] I. Lažetić, "Probabilističke osnove uputstva za izražavanje nesigurnosti u merenju (GUM)", Diplomski rad, Fakultet tehničkih nauka Novi Sad, 2017.
- [4] *Six Sigma-based quality control – Learning guide series*, Abbott Diagnostics, Corelaboratory.Abbott, 2018.
- [5] S. Westgard, V. Petrides, Sh. Schneider and al. "Assessing precision, bias and sigma-metrics of 53 measurands of the Alinity ci system", *Clinical Biochemistry*, 50, Vol. 18, pp. 1216-1221, 2017.
- [6] M.B. Rajković, M. Mitrović, S. Antić-Mladenović, "Obezbeđenje poverenja u kvalitet rezultata hemijskih metoda ispitivanja", *Zaštita materijala* 60, Vol. 4, pp. 342-359, 2019.

Kratka biografija:



Ivan Lažetić rođen je 1992. god u Novom Sadu. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Merni sistemi odbranio je 2021.god.
kontakt: lazeticivan@gmail.com



Platon Sovilj je doktorirao 2010. godine na Fakultetu tehničkih nauka. Redovni je profesor i šef Katedre za električna merenja. Oblast interesovanja su biomedicinska merenja, metrologija i merno-informacioni sistemi.

РЕАЛИЗАЦИЈА ТРОСЛОЈНОГ СИСТЕМА ЗА ВИДЕО ПРЕНОС ПО RTSP ПРОТОКОЛУ

REALIZATION OF THREE-LAYER RTSP VIDEO STREAMING SYSTEM

Јован Славујевић, Предраг Теодоровић, *Факултет техничких наука, Нови Сад*

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – У овом чланку описана је реализација и функционисање трослојног система за видео пренос по RTSP протоколу. Омогућено је вишезначно видео преношење уживо са једне или више камера, као и једнозначно приказивање по захтеву раније снимљених видео записа у временским размацама од 15 минута. Систем је имплементиран на личној машини - i7@3GHz са 16GB RAM и Windows 10 OS. Систем је подржан и на Linux OS, како би се омогућила имплементација на популарним ембедед платформама као што је нпр. Raspberry Pi.

Кључне речи: RTSP, Qt, GUI, SQL, криптографија

Abstract – This article describes realization and functioning of three-layer video streaming system according to RTSP protocol. Here is provided live multicast video streaming from one or more cameras and unicast On Demand video streaming of earlier recorded video files in 15 minutes time slots. System was implemented on personal computer - i7@3GHz with 16GB RAM and Windows 10 OS. System has supported Linux OS, in order to provide implementation on popular embedded platforms such as Raspberry Pi.

Keywords: RTSP, Qt, GUI, SQL, cryptography

1. УВОД

Видео пренос представља незаобилазну појаву данашњице, а у највећој мери предњачи интернет видео пренос. Није изненађујуће што је моћ директности коју са собом носи телевизија пронашла свој пут до интернета. Online видео постаје императив, па је тако све више различитих платформи које омогућавају живи видео пренос.

Постоје два различита типа живог преноса. У првом начину, видео садржај је доступан директно на сајту и користи софтвер који је већ интегрисан у сам сајт. Код другог начина, гледаоци могу да прате садржај само кроз самостални (енг. *Standalone*) програм – рачунарску или мобилну апликацију, где софтвер омогућава приказивање садржаја.

У оквиру овог рада реализован је софтверски пакет који обухвата видео пренос – серверску страну, али и његово приказивање – клијентску страну. Ток видео

преноса диктиран је по специјалном стандарду видео преноса какав је RTSP [1]. Поред тога, приступ видео преносима лимитиран је евидентирањем корисника у бази података [2] и приступањем уз помоћ јединствених корисничких креденцијала.

Видео пренос представља пренос директних снимака са камере. Број камера је конфигурабилан, па их тако може бити једна или више. Камере морају бити физички повезане на уређају на којем је покренут сервер. Систем подржава и својеврсно „враћање у назад“ по захтеву у виду 4 могућа преноса која константно пружају снимак од последњих сат времена у размацама од по 15 минута.

Корисничка апликација поседује графички кориснички интерфејс – GUI, са менијем усмереним ка кориснику (енг. *User-friendly*). Цео систем је имплементиран и тестиран на локалној мрежи, али би уз извесне измене могао да функционише и на глобалној мрежи.

2. Концепт решења

У овом поглављу описан је и илустрован концепт решења, идеја пројекта, као и архитектура и функционалност појединачних модула система.

2.1 Библиотека мрежне утичнице у објектно оријентисаној форми

Интерфејс за програмирање апликације - API мрежне утичнице (енг. *Socket*) представља срце комуникације међу процесима или уређајима, старајући се о ниским нивоима комуникације као што су интернет протокол или ниже од тога, остављајући кориснику могућност да се стара и размишља само о комуникацији на апликативном нивоу.

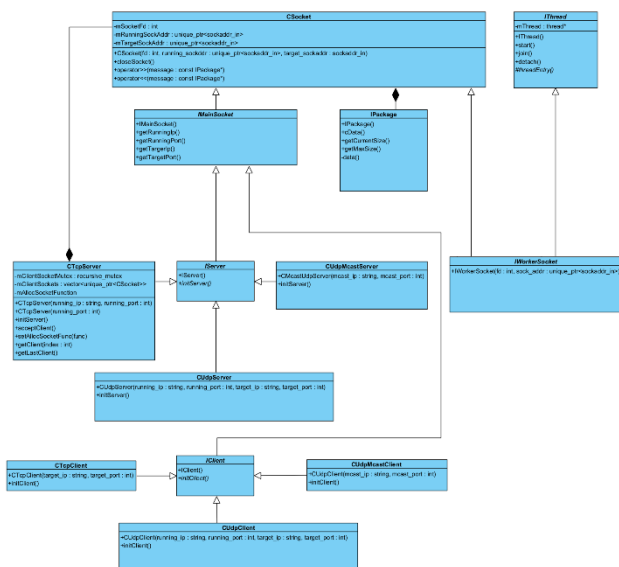
Иако веома неопходан и готово свеприсутан, овај API написан је у програмском језику C, те га је, као таквог, некомфорно користити. Идеја је била прилагодити библиотеку објектно-оријентисаној парадигми паковањем њених функција и структура у класе познатије као *Wrapper* класе. На тај начин, поред сажетог и читког кода, добија се заједнички интерфејс за оба оперативна система, који се лако може користити у оквиру било које апликације.

Интерфејса садржи 3 пара класа које је могуће користити (инстанцирати). У питању су парови – клијент и сервер, а 3 пара покривају TCP, затим једнозначни и вишезначни UDP IP [3]. Поред наведених постоје и апстрактне класе које је могуће наследити. Једна од њих је класа *IPackage* која се користи за слање и пријем порука. У оквиру изведене класе потребно је

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др. Предраг Теодоровић, доцент.

дефинисати (*override*) чисте (*pure*) виртуалне метода које заправо представљају тзв. *Getter* методе за максималну и искоришћену величину, као и стартну адресу поруке. На следећој слици (слика 1.) приказан је *UML* дијаграм класа.



Слика 1. *UML* класни дијаграм *Wrapper* библиотеке

2.2 Архитектура, подела и улога модула система

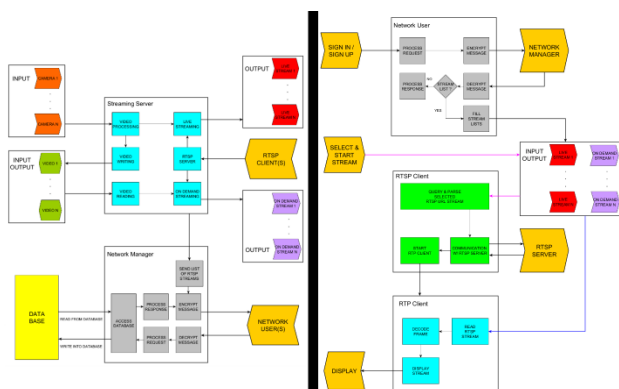
Серверска апликација садржи два велика модула:

1. Сервер за видео пренос - модул задужен за сву видео обраду, интеракцију и дистрибуцију преноса.
2. Мрежни менаџер - модул задужен за евиденцију корисника, одржавање базе података и заштиту личних података корисника.

Клијентска апликација садржи 3 мања модула:

1. *RTSP* клијент - модул задужен за потраживање, прекид или пребацивање на други видео пренос.
2. *RTP* клијент - модул задужен за пријем, обраду и приказивање видео преноса по налогу *RTSP* клијента.
3. Мрежни корисник - модул задужен за приступ мрежи система и набавку листе могућих преноса.

На следећој слици (слика 2.) приказане су идејне блок шеме за обе апликације у оквиру система.



Слика 2. Идејне блок шеме серверске и клијентске апликације у оквиру трослојног система

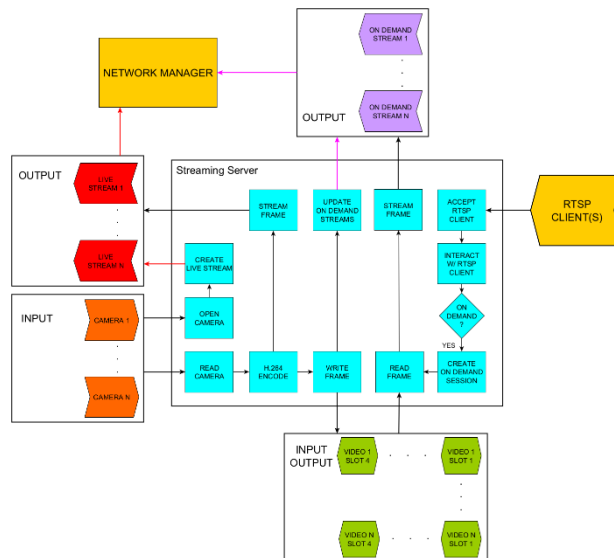
3. ПРОГРАМСКО РЕШЕЊЕ

У овом поглављу описано је програмско решење пројекта - рад појединачних модула.

3.1 Програмско решење серверске апликације

3.1.1 Сервер за видео пренос

Главна улога овог модула јесте видео пренос. Међутим његове активности превазилазе те оквире, па је тако задужен за целокупну видео обраду, интеракцију са клијентима преноса и услуживање видео преноса – живо и по захтеву. На следећој слици (слика 3.) приказана је детаљнија блок шема сервера за видео пренос.



Слика 3. Детаљна блок шема сервера за видео пренос

Читање са камере и видео записивање

Серверска апликација при покретању зна којим бројем камера ће управљати. Да би управљала камером, апликација користи објекте из *OpenCV* библиотеке. Након што је камера успешно повезана, креира се и покреће посебна нит у којој се извршава петља читања све до краја рада серверске апликације или физичког прекида везе са камером.

Слика са камере се преузима у сировом формату, где као таква досеже огромне величине. За слику резолуције 640x480 добија се 307.200 пиксела, што множењем са бројем основних боја из *RGB* система резултује у 921.600 бајта, односно 900 *kB*. Очигледно је да слика у оваквом формату није практична за манипулисање или пренос путем мреже.

Због тога се након читања слика компресује по *H.264* [4] кодеку. Овај механизам реализује се уз помоћ библиотеке *FFmpeg*. Након што је сирова слика камере компресована добија се *frame* који је знатно мањи, а његова величина варира у зависности од промена које је ухватила камера, али највећа величина до које досеже је око 70 *kB*.

Новонастали *frame* се прослеђује на видео записивање. Генерисање видео функционише тако да сваки фајл не треба бити дужи од 15 минута. У моменту када то време истекне, ажурира се листа видео преноса по захтеву. Евиденција о протеклом

времену од 15 минута извршава се у другој нити која мери протекло време и обавештава када оно истекне, да би се видео записивач пребацио на нови видео.

Креирање новог видео записа функционише по принципу кружног попуњавања постојећих фајлова када постану застарели. Како би се избегла ситуација да видео записивач покуша да пресними видео који се користи, уведена су два фајла која се наизменично користе за уписивање. На тај начин постоји 8 видео записа од по 15 минута за 4 могућа видео преноса по захтеву који се циклично попуњавају. Уписани видео *frame* се даље прослеђује на пренос уживо.

RTSP интеракција са клијентима

Сервер за видео пренос пружа пренос искључиво по *RTSP* протоколу. Протокол захтева од корисника да, поред утичнице за добављање видео садржаја (*RTP*), поседује и једну за интеракцију са *RTSP* сервером.

RTSP сервер се може илустровано представити као телевизор, а клијент као даљински управљач, где се притиском бројева на управљачу добија комбинација на основу које телевизор зна на који канал треба усмерити гледаоца. Комбинацију, у *RTSP* протоколу, представља *URL* линк који клијент шаље приликом интеракције, док усмеравање гледаоца представља прослеђивање *IP* адресе, порта и протокола видео преноса, односно *RTP* сервера, али и друге важне информације као што је кодек који се користи.

RTP вишезначни видео пренос

Раније је утврђено да *RTSP* и *RTP* представљају уређени пар серверских утичница за видео пренос, где један без другог не могу. *RTSP* сервер мора да има пар који ће да пласира видео садржај, док *RTP* сервер у већини случаја не постоји или није започео пренос све док овај други није сигнализирао.

Вишезначни видео пренос је, у оквиру овог система, намењен искључиво за пренос уживо, што има смисла, јер живи пренос пласира идентичан садржај ка свим корисницима истовремено. *RTP* вишезначни сервер *H.264 frame* шаље у *RTP* формату. *RTP* пакет поседује максималну величину од 1460 бајта за *payload*, па се велики *frame*-ови шаљу у сегментима. На почетку сваког пакета умиће се заглавље - *RTP Header*, које има 12 *byte*-а у оквиру којих су садржани подаци неопходни за композицију и коришћење.

RTP једнозначни видео пренос

Видео пренос по захтеву је тип преноса који ради у „1 на 1“ односу сервера и клијента. За сваког пристиглог клијента репродукује се раније снимљени видео садржај од почетка. На пример, један корисник може гледати пети минут видео снимка, док други гледа почетак.

Да би тако нешто било изводљиво сервер за видео пренос, након *RTSP* интеракције, мора инстанцирати посебан једнозначни *RTP* сервер, као и читач видео записа који ће видео запис пласирати ка клијенту у оквиру засебне сесије (нити). Важно је напоменути да сваки *RTP* сервер мора поседовати јединствени порт на којем извршава видео пренос по захтеву. Овај део

је веома осетљив, јер овакав подухват приликом повезивања више клијената резултује већом потрошњом *CPU* и *RAM* серверског уређаја.

3.1.2 Мрежни менаџер

Мрежни менаџер је модул који је, у оквиру овог система, задужен за интеракцију са мрежним корисницима, одржавање и приступање бази података, као и заштиту личних података корисника.

Интеракција са мрежним корисницима

Мрежни корисници, попут *RTSP* клијената у интеракцији са сервером, морају испоштовати стриктан протокол односно начин на који комуницирају са менаџером. Протокол не представља неки од познатих стандарда, већ специфичан вид паковања и формирања захтева и одговора који је посебно осмишљен за потребе овог модула.

Комуникација се може поделити у два режима:

1. Иницијални – одвија се приликом корисничког приступања мрежи.
2. Устаљени – одвија се од тренутка када је иницијални део комуникације завршен све до изласка мрежног корисника.

Током иницијалног режима, мрежни корисник може послати захтев за пријаву и/или регистрацију. Уколико је корисник већ регистрован, тај корак прескаче и шаље захтев за пријаву. Мрежни менаџер комуникацију са корисником држи у иницијалном режиму све док захтев за пријаву није успешно обрађен, а потом се прелази у устаљени режим.

У устаљеном режиму комуникације, мрежни менаџер благовремено ажурира корисника о листи постојећих *RTSP* видео преноса. Поред тога, уколико корисник жели да користи неки од понуђених видео преноса по захтеву мрежни менаџера врши проверу заузетости таквих преноса. Број активних корисника видео преноса по захтеву је ограничен на 5.

За евиденцију корисника користи се база података помоћу *MySQL* сервиса. За регистрацију нових корисника ради се унос нових података у базу, а за пријаву читање са исте. Током акција за регистрацију и пријаву корисника кроз мрежу се у форми поруке преносе приватни подаци корисника – корисничко име, *e-mail* адреса и лозинка. У интернет мрежи је могуће те поруке пресрести и видети њихов садржај, чиме се нарушава приватност.

Мрежни менаџер чува корисничке податке преко криптографије – енкрипције података. У питању је *AES* симетрична енкрипција, у оквиру које и мрежни менаџер и корисник знају јавни кључ и вектор иницијализације – *IV*. За реализацију енкрипције података користи се библиотека *Crypto++*. Принцип рада овог подмодула је једноставан – пре слања поруке, корисник или менаџер ураде енкрипцију података, а након пријема исте ураде декрипцију примљеног садржаја.

3.2 Програмско решење клијентске апликације

3.2.1 RTSP клијент

Након што је корисник успешно евидентиран у мрежи, добија листу активних видео преноса, где одабиром неког од њих RTSP започиње комуникацију са сервером.

Приликом комуникације са сервером – клијент шаље следеће RTSP захтеве: *Options*, *Describe*, *Setup* и на крају *Play*. У оквиру ових порука налазе се, између осталог, и информације као што су IP адреса, порт и протокол RTP сервера жељеног видео преноса, али и порт који сервер очекује од RTP клијента, у случају преноса по захтеву. Након успешно обрађеног *Setup* захтева, могуће је иницијализовати RTP клијента, а након успешно обрађеног *Play* захтева, могуће је започети његов рад. Уколико корисник жели прекид преноса шаље се *Teardown RTSP* захтев.

3.2.2 RTP клијент

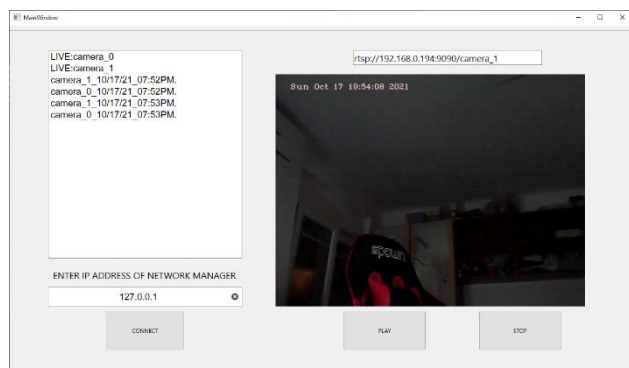
Након што је RTSP клијент успешно одрадио свој део посла, овај подмодул започиње примање и колекцију RTP пакета, H.264 декомпресију и приказивање преноса на главном UI прозору апликације.

Постоји могућност да декодер у некој итерацији не успе да добије слику. То је честа појава код живог видео преноса јер не може успешно да одради предвиђања. За неуспелу декомпресију се примљени материјал брише и наставља се колекција следећег. Ова појава се дешава при самом почетку репродуковања, где је потребно да се декодер „уигра“, а након првог успешног декодовања видео пренос даље тече у најбољем реду.

3.2.3 Мрежни корисник

Да би корисник могао да гледа било који од видео преноса у оквиру овог система неопходно је да буде евидентиран у мрежи са следећим подацима: e-mail адреса, корисничко име и лозинка. При првом приступању у мрежи корисник се региструје, а након тога само пријављује.

Притиском на дугме за повезивање покреће се процес конекције са мрежним менаџером. Уколико је конекција успешна отвара се додатни прозор за пријаву/регистрацију корисника, слика 4.



Слика 4. Успешно приказивање живог видео преноса

Након успешне пријаве, мрежни корисник попуњава табелу на менију главног прозора на основу поруке коју благовремено добија од мрежног менаџера. Интеракција са мрежним менаџером одиграва се у главној нити све до краја иницијалног дела комуникације, док се устаљени режим врши у позадинској нити.

4. ЗАКЉУЧАК

Коначно тестирање показало је да су сви елементи система успешно интегрисани у синхронизовану целину која успешно обавља тражени задатак и обезбеђен је поуздан рад свих модула система.

Предлози за побољшање:

- Омогућити подршку за паузирање и пуштање насумичних делова за видео пренос по захтеву
- Раздвојити модуле серверске апликације у више апликација, тако да се рад подели у више уређаја у систему:
 - Апликација која пласира живи видео пренос са камере и врши записивање видео снимака на самба путањи.
 - Апликација која врши услуживање видео преноса по захтеву.
 - Апликација која врши евиденцију корисника система, управља базом података и преусмерава на видео пренос.
- Имплементирати *Watchdog* сервис, који би поновно покретао неактивне делове система.

5. ЛИТЕРАТУРА

- [1] Henning Schulzrinne, Anup Rao, and Robert Lanphier. *Real time streaming protocol (RTSP)*. (1998).
- [2] Алемпије Вељовић, и Мирољуб Захорјански. *Увод у базе података*. СЕТ, 2014.
- [3] Милојко Јевтовић. *Комуникациони протоколи интернета*. Академска мисао, 2015.
- [4] Richardson, Iain E. *The H. 264 advanced video compression standard*. John Wiley & Sons, 2011.

Кратка биографија:



Јован Славујевић рођен је у Београду 1996. год. Дипломски рад на Факултету техничких наука из области Електротехника и рачунарство – Примењена електроника одбранио је 2020.год.

Области интересовања су му развој софтвера у програмским језицима C++, C# и Java.
контакт: slavujevic.jovan.96@gmail.com

PRIKAZ DEVOPS PRAKSI U AZURE OKRUŽENJU**AZURE DEVOPS IN PRACTICE**Maja Grubor, *Fakultet tehničkih nauka, Novi Sad***Oblast – RAČUNARSTVO I AUTOMATIKA**

Kratak sadržaj – Ovaj rad se bavi DevOps-om, sa fokusom na Azure DevOps i svim servisima koje on nudi. Opisuje se i još neki DevOps alati. Srž rada jeste deployment više verzija jednog softvera upotrebom Azure DevOps-a.

Ključne reči: DevOps, Azure, Azure DevOps, build, release, pipeline, deployment

Abstract – Main goal of this paper is to present DevOps with focus on Azure DevOps and services which it offers. Here is described other Dev Ops tools. The core of the work is deployment more then one software versions via Azure DevOps.

Keywords: DevOps, Azure, Azure DevOps, build, release, pipeline, deployment

1. UVOD

DevOps je skup praksi koje rade na automatizaciji i integraciji procesa između razvoja softvera i IT timova, tako da oni mogu brže i pouzdanije graditi, testirati i isporučivati softver. Termin “DevOps” nastao je kombinovanjem reči “development” (razvoj) i “operations” (operacije) te označava praksu razvoja softvera koja objedinjuje razvoj i IT operacije kako bi omogućila kontinuiranu isporuku korisnicima. DevOps uključuje osnovne prakse kao što su planiranje, praćenje, razvoj, izgradnja, testiranje, isporuka i nadgledanje. DevOps prakse održavaju ideju stalnog poboljšanja i automatizacije. Mnoge prakse fokusiraju se na jednu ili više faza razvojnog ciklusa. Agilni razvoj je značajan deo DevOps-a i imao je značajan uticaj na njegovo stvaranje. Nije uobočajeno a takođe se ni ne preporučuje koristiti agilne metodologije bez upotrebe DevOps-a.

2. DEVOPS I ŽIVOTNI CIKLUS APLIKACIJE

DevOps utiče na životni ciklus aplikacije tokom celog njenog plana, razvoja, isporuke i rada. Svaka faza se odnosi na ostale faze, a one nisu specifične za uloge. U pravoj DevOps kulturi svaka uloga je donekle uključena u svaku fazu [1].

- Plan - U fazi planiranja DevOps timovi osmišljavaju, opisuju i definišu karakteristike i mogućnosti aplikacija i sistema koje grade. Oni prate napredak na niskom i visokom nivou detaljnosti – od zadataka pojedinačnih proizvoda do zadataka koji obuhvataju više proizvoda [2].

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Željko Vuković, docent.

- Develop - Faza razvoja uključuje sve aspekte kodiranja – pisanje, testiranje, pregled i integraciju koda svih članova tima. DevOps timovi nastoje da brzo unose inovacije bez smanjenja kvaliteta, stabilnosti i produktivnosti [2].

- Deliver - Isporuka predstavlja proces postavljanja aplikacija u proizvodna okruženja na dosledan i pouzdan način. Faza isporuke takođe uključuje primenu i konfigurisanje infrastrukture. U fazi isporuke timovi definišu proces upravljanja sa jasnim fazama [2].

- Operate - Faza rada uključuje održavanje, nadgledanje i rešavanje problema u aplikacijama u proizvodnom okruženju. Timovi rade na tome da obezbede pouzdanost sistema, veliku dostupnost i teže ka tome da imaju nulti zastoj uz jačanje bezbednosti i upravljanja sistema. Timovi imaju u cilju da identifikuju probleme i ublaže ih ili otklone u potpunosti pre nego što na njih naiđe korisnik [2].

3. DEVOPS ALATI

Bez odgovarajućih alata i tehnologija DevOps nije održiv model. Oni omogućavaju automatizaciju, kontinuiranu integraciju i isporuku, upravljanje konfiguracijom, testiranje, puštanje u produkciju i monitoring. U narednom tekstu biće navedeni neki od alata.

3.1. Jenkins

Jenkins je platforma otvorenog koda za kontinuiranu integraciju i kontinuiranu isporuku koja se može koristiti za automatizaciju svih vrsta zadataka u vezi sa izgradnjom, testiranjem i isporukom ili primenom softvera [3].

3.2. Kubernetes

Kubernetes je alatka otvorenog koda za upravljanje kontejnerima koja se naziva i alatka za orkestraciju kontejnera. Koristi se za stvari poput automatizacije postavljanja kontejnera, skaliranja i uklanjanja istih.

To nije platforma za kontejnerizaciju već alat koji se koristi na toj platformi. Kubernetes klasteri mogu obuhvatati hostove u lokalnom, javnom, privatnom ili hibridnom cloud-u. Iz tog razloga, Kubernetes je idealna platforma za hosting aplikacija zasnovanih na cloud-u koje zahtevaju brzo skaliranje, poput strimova podataka u realnom vremenu [4].

3.3. Docker

Docker je softverska platforma koja omogućava brzo pravljenje, testiranje i primenu aplikacija. On pakuje softver u standardizovane jedinice (kontejnere) koji imaju sve što je potrebno softveru za pokretanje, uključujući biblioteke, sistemse alate i kod. Pomoću Docker-a mogu

se brzo deploy-ovati i skalirati aplikacije u bilo koje okruženje i znati da će se kod pokrenuti [5].

3.4. Git

Git je distribuirani SCM (source code management) alat koji koriste udaljeni timovi i korisnici otvorenog koda. Git omogućava praćenje i napredak rada kao i čuvanje različitih verzija izvornog koda i samim tim i vraćanje na neke od prethodnih verzija [6].

3.5. Azure DevOps

Azure DevOps pruža usluge timovima za planiranje rada, saradnju na razvoju koda i izradu i primenu aplikacija. Podržava skup procesa koji drži na okupu programere, menadžere projekta i saradnike da bi se završio razvoj projekta. Omogućava organizacijama da stvaraju i poboljšavaju proizvode brže nego što mogu sa tradicionalnim pristupima razvoju softvera [7].

4. AZURE

Azure je platforma kompanije Microsoft koja nudi širok spektar cloud servisa koji uključuju računarstvo, analitiku, skladištenje i umrežavanje. Ova platforma ima za cilj da pomogne raznim kompanijama da upravljaju izazovima i ispunе svoje organizacione ciljeve.

4.1. Azure DevOps

Azure DevOps je Microsoft-ova platforma koja pruža end-to-end lanac alata za razvoj i primenu softvera. Takođe, integriše se sa većinom alata na tržištu i odlična je opcija za orkestriranje DevOps lanca alata. Za izradu aplikacije odabrana su razvojna okruženja Visual Studio i Visual Studio Code koja imaju integraciju sa Azure DevOps-om. Back-end deo rađen je u .NET Core tehnologiji i korišćen je C# programski jezik, dok je za front-end deo korišćen Angular i TypeScript. Ovo je aplikacija za menadžovanje zaposlenix u kompaniji. Za izradu aplikacije odabrana su razvojna okruženja Visual Studio i Visual Studio Code koja imaju integraciju sa Azure DevOps-om. Back-end deo rađen je u .NET Core tehnologiji i korišćen je C# programski jezik, dok je za front-end deo korišćen Angular i TypeScript. Ovo je aplikacija za menadžovanje zaposlenix u kompaniji.

4.1.1. Azure Boards

Pomoću Azure Boards veb usluge timovi mogu upravljati svojim softverskim projektima. Pruža bogat skzp mogućnosti uključujući izvornu podršku za Scrum i Kanban, prilagodljive kontrolne table i integrisano izveštavanje. Možemo brzo i jednostavno započeti praćenje user story-a (korisničkih priča), feature-a, backlog item-a i bug-ova vezanih za naš projekat [8].

4.1.2. Azure Repos

Azure Repos je skup alata za kontrolu verzija koje možemo koristiti za upravljanje kodom. Bez obzira da li se radi o malom ili velikom softevru, upotreba kontrole verzija je jako dobra ideja. Sistemi za kontrolu verzija su softveri koji nam pomažu da pratimo promene koje unesemo u kod tokom vremena [9].

4.1.3. Azure Pipelines

Azure Pipeline automatski build-uje i testira kod projekta kako bi ih učinili dostupnim drugima. Kombinuje kontinuiranu integraciju i kontinuiranu isporuku za

testiranje i izradu koda i njeovo slanje na bilo koji target [10].

Continuous Integration (CI) – praksa koju koriste razvojni timovi za automatizaciju spajanja i testiranja koda. Implementacija CI pomaže u hvatanju grešaka u ranom razvojnom ciklusu, što ih čini jeftinijim za ispravljanje [10, 13].

Continuous Delivery (CD) – proces pomoću kog se kod gradi, testira i primenjuje u jedno ili više testnih ili proizvodnih okruženja. Primenom i testiranjem u više okruženja povećava se kvalitet. CI sistemi proizvode artifakte koji se mogu deploy-ovati, uključujući infrastrukturu i aplikacije [10, 13].

Continuous Testing (CT) – upotreba automatskih build-deploy-test tokova lokalno ili u cloud-u, sa izborom tehnologija i framework-a koji kontinuirano testiraju naše promene na brz, skalabilan i efikasan način [10].

4.1.4. Azure Test Plans

Azure DevOps pruža bogate i moćne alate koje svima u timu mogu koristiti za poboljšanje kvaliteta i saradnje tokom celog procesa. Rešenje za upravljanje testovima u pregledaču pruža sve mogućnosti potrebne za planiranje ručnog testiranja, istraživačkog testiranja i prikupljanja povratnih informacija od stakeholder-a [11].

4.1.5. Azure Artifacts

Omogućava programerima da dele i koriste pakete iz različitih izvora i javnih registara. Paketi se mogu deliti unutar istog tima, iste organizacije ili javno [12].

5. OPIS PROBLEMA I IMPLEMENTACIJA REŠENJA

Cilj ovog rada jeste ispratiti deployment nekog softvera, odnosno više verzija tog softvera. U narednim poglavljima biće detaljno objašnjeno kako je aplikacija implementirana i kako je uz pomoć Azure DevOps-a ona deploy-ovana u produkciono okruženje.

5.1. Opis korišćenih alata

Za izradu aplikacije odabrana su razvojna okruženja Visual Studio i Visual Studio Code koja imaju integraciju sa Azure DevOps-om. Back-end deo rađen je u .NET Core tehnologiji i korišćen je C# programski jezik, dok je za front-end deo korišćen Angular i TypeScript. Ovo je aplikacija za menadžovanje zaposlenix u kompaniji.

5.2. Kreiranje aplikacije u Azure DevOps-u

Pre svega, treba da imamo kreiran nalog na Azure DevOps-u kao i organizaciju u sklopu koje ćemo da napravimo projekat. Samo kreiranje projekta je prilično jednostavan korak. Potrebno je uneti ime projekta i označiti da li želimo da projekat bude javno dostupan ili privatan. Postoje još dva dodatna podešavanja koja se mogu odabrati, a to je da odaberemo Version Control agilni pristup želimo da koristimo.

5.3. Planiranje i kreiranje taskova upotrebom Azure Boards

Ono što bi sledeće trebalo biti urađeno kada se koristi ovaj pristup jeste da se isplanira posao, kreiraju zadaci i dodele programerima, testerima i drugim učesnicima.

U sklopu kartice Boards postoji nekoliko pod-kartica: backlog u kojoj možemo da vidimo sve kreirane taskove, sprint gde su prikazani samo taskovi koji spadaju u selektovanu iteraciju i još neke.

5.4. Postavljanje koda u repozitorijum

Sledeći korak jeste da se projekat koji postoji kreiran lokalno postavi u Azure DevOps repozitorijum. To može da se uradi na nekoliko načina a jedan od njih je da se to uradi kroz Visual Studio ili Visual Studio Code jer kao što je ranije spomenuto oni su integrisani sa Azure DevOps-om i Git-om. Ovde je prvo postavljanje koda rađeno kroz Visual Studio. Za početak, u VS treba biti ulogovan sa nalogom na kom je prethodno kreiran repozitorijum, zatim treba kliknuti desni klik na lokalni repozitorijum pa "Create Git repository" i otvoriće se prozor gde se može da kreirati novi repozitorijum ili iskoristiti postojeći. Kako ovde već postoji kreiran, potrebno je samo uneti njegov url.

5.5. Kreiranje Build Pipeline-a

Sledeći korak jeste konfigurisanje build pipeline-a. Potrebno je iskonfigurisati build proces za projekat koji je rađen u .NET Core-u i projekat rađen u Angular-u. Taj build može da se pokreće svaki put nakon što je dodat neki novi kod direktno na glavnu granu ali to baš i nije dobra praksa jer nismo sigurni da li naša izmena može prouzrokovati greške. Da bi se koknfigurisao build, treba otići na karticu Pipelines koja se nalazi u sklopu sekcije Repos i kliknuti na "Create pipeline".

Nakon što smo rekli da želimo da kreiramo pipeline treba da se odabere izvor gde se nalazi kod, projekat i repozitorijum koji želimo kao i granu za koju kreiramo konfiguraciju. Inicijalno se dobija jedan Agent Job u koji se dodaju taskovi koji će se redom izvršavati. Ovaj Agent Job će se koristiti za konfigurisanje .NET Core projekta i samim tim će se dodavati taskovi u skladu sa njegovim potrebama. Prvi task koji je dodat je task za instaliranje Nuget paketa. Potrebno je da se odabere verzija Nuget-a koju želimo da instaliramo. Ovde je odabrano da svaki put želimo da se proveriti da li postoji novija verzija i ukoliko postoji da se ona instalira.

Naredni task koji je dodat jeste Nuget restore. Nakon što su dodati prethodni taskovi, može da se uradi build solution-a koristeći komandu .net build. Iako u ovoj fazi nisu implementirani testovi, dodati je zadatak za to jer je planirano dodavanje testova u nekoj od narednih iteracija. Sledeći korak je izvršavanje dotnet publish-a kako bi se dobili konačni rezultati build-a. Poslednji korak u konfiguraciji serverske strane predstavlja publish-ovanje build artifakta, tako da release pipeline može da vidi šta je rezultat ovog build procesa.

Kada su svi taskovi potrebni za serverski deo dodati, može se preći na klijentski deo aplikacije. Postoje dve mogućnosti u ovoj fazi a to je da se u sklopu ovog pipeline-a doda novi Agent Job i dodaju taskovi potrebni za build Angular aplikacije ili da se kreira novi pipeline koji bi imao jedan Agent Job. Ovde nije kreiran novi pipeline već je dodat još jedan Agent Job u sklopu postojećeg pipeline-a.

Prvi task koji je dodat je npm install koji služi za instaliranje paketa. Sledeći dodat task je npm build.

Nakon toga, kao i kod serverskog dela, dodat je task za publish-ovanje artifakta. Putanja koja je ovde konfigurirana je ista ona putanja iz prethodnog taska koja predstavlja njegov rezultat.

Sledeći korak bi bio kreiranje release pipeline-a uz pomoć kreiranih artifakta. Odlučeno je da se kreira AppService na Azure Portal-u i na njega hostuje naša aplikacija prilikom release-a.

5.6. Kreiranje App Service-a

Azure Portal je konzola bazirana na vebu koja pruža alternativu command-line alatima. Pomoću Azure Portal-a možemo upravlјati Azure subscription-ima pomoću grafičkog korisničkog interfejsa. Možemo da kreiramo, upravljamo, nadgledamo sve od najjednostavnije veb aplikacije do složenih cloud implementacija.

Azure App Service je HTTP servis za hostovanje veb aplikacija, REST api-a i back end dela mobilnih aplikacija.

Tokom kreiranja AppService-a treba da se unesu Subscription i Resource Group – predstavlja kontejner koji sadrži povezane resurse solution-a. Takođe, treba uneseti naziv veb aplikacije, da označiti da li se publishuje kod ili Docker kontejner, selektovati programski jezik ili framework, region i još neke opcione podatke. Nakon što je kreiran AppService, dobija se url na kome će se nalaziti naša aplikacija nakon hostovanja [14].

5.7. Kreiranje Release Pipeline-a

Na Azure DevOps-u, u okviru sekcije Pipelines postoji karticu Release, klikom na nju dobija se stranica gde su prikazani trenutno postojeći release-ovi. Template koji je izabran u momentu kreiranja jeste Azure App Service Deployment koji se koristi za preuzimanje artifakta i postavljanja na Azure. Ovde će i klijentski i serverski deo biti deploy-ovani na isti server, ali je moguće da budu deploy-ovani na različite servere.

Prvi artefakt koji je dodat je serverski kod, pa je kao izvor izabrana build definiciju koja je dobijena iz artifakta kao izlaz.

Nakon što je kreiran prvi artefakt, klikom na "1 job, 1 task" može se otići na stranicu gde se može detaljnije videti i podesiti šta treba da se uradi sa artefatom. U tom dijalogu, treba da se odabere subscription i ime App Service-a na koji će se deploy-ovati naš kod.

Kao drugi artefakt odabran je dist artefakt iz klijentskog koda koji se proizvodi u drugoj fazi build-a. Što se tiče ovog artifakta, sva podešavanja su ostala ista kao i kod prvog osim izvora, koji mora biti drugačiji. Postoji mogućnost da se nakon svakog uspešno završenog build-a automatski pokrene release.

Nakon što je pokrenut kreirani build pipeline, a nakon njega i release, kada se ode na url App Service-a, trebalo bi da se tamo vidi naša aplikacija.

5.8. Re-deploy aplikacije

Da bi se uradio re-deploy aplikacije, potrebno je da postoje neke izmene u kodu, odnosno da postoji nova verzija build-a. Kada je iz lokala push-ovana grana i kod koji je dodat, na stranici Pull Requests biće prikazana ta grana kao i mogućnost kreiranja Pull Request-a za tu

granu. Nakon što se klikne na dugme “Create a pull request” dobija se stranica gde se mogu videti svi fajlovi koji su izmenjeni, koliko i koji commit-ovi su napravljeni. Takođe, može se povezati preko id-a sa feature-om ili bug-om na koji se odnosi. Bitna stvar u ovom koraku jeste odabrati granu u koju će se spojiti izmene nakon što se PR complete-uje. Nakon što je odrađen review, svi komentari su ispravljeni, dovoljan broj review-era je approve-ovao PR, on može biti complete-ovan. Takođe, može se podesiti da se uradi build feature grane čim se kreira PR ili se desi neka izmena na PR-u, da kreator PR-a može da zna ukoliko build nije prošao uspešno da ne sme tu granu spojiti sa glavnom granom da ne bi došlo do grešaka.

U našem slučaju kao i u prethodnom poglavlju pokrenuće se Build Pipeline koji će nakon što se uspešno završi pokrenuti release i deploy-ovati novu verziju naše aplikacije. Može se automatizovati da svaki put kada se dogodi izmena na grani koju deploy-ujemo, da se pokrene build i tako svaki put dobijemo novu verziju softvera dostupnu.

6. ZAKLJUČAK

Cilj DevOps-a jeste da skрати životni ciklus razvoja softvera i da obezbedi kontinuiranu integraciju i kontinuiranu isporuku sa visokim kvalitetom softvera. Osnovni koncept DevOps-a CI/CD se sastoji iz kontinuirane integracije (CI), kontinuirane isporuke (CD) i kontinuiranog deployment-a (CD). Ova tri procesa zajedno automatizuju izgradnju, testiranje i primenu tako da DevOps timovi mogu brže i pouzdanije isporučiti kod. CI/CD procesi su fokusirani na automatizaciju isporuke softvera dok se DevOps takođe fokusira na organizacione promene kako bi podržao saradnju između svih uključenih funkcija.

Uz pomoć Azure DevOps-a možemo da ispratimo razvoj softvera od ideje, preko planiranja, organizacije, implementacije, automatizacije procesa do samog deployment-a softvera u produkciju. Takođe ono što je vrlo bitno jeste da on ima integraciju sa velikim brojem alata, čeka i nekih ranije pomenutih.

Rešenje prikazano u ovom radu, moglo bi se unaprediti u nekim aspektima. Prva stvar bi bila prilikom kreiranja build pipeline-a, da se umesto jednog kreiranja, za svaki projekat kreira zaseban build pipeline jer bismo smanjili vreme izvršavanja. Često se dešava da se menja samo jedan projekat, te nema potrebe za pokretanje build-a svih projekata. Takođe i u slučaju da imamo više od dva projekta u repozitorijumu, bilo bi dobro da se za svaki projekat našravi poseban build pipeline. Nešto slično bi se moglo uraditi i za release, gde bismo razdvojili release klijentskog i serverskog dela.

7. LITERATURA

[1] <https://azure.microsoft.com/en-us/overview/what-is-devops/#devops-overview> (pristupljeno u junu 2021.)

[2] <https://azure.microsoft.com/en-us/overview/what-is-devops/#culture> (pristupljeno u junu 2021.)

[3] jenkins.io (pristupljeno u julu 2021.)

[4] <https://kubernetes.io/> (pristupljeno u julu 2021.)

[5] <https://aws.amazon.com/docker/> (pristupljeno u julu 2021.)

[6] <https://git-scm.com/> (pristupljeno u julu 2021.)

[7] <https://docs.microsoft.com/> (pristupljeno u julu 2021.)

[8] <https://docs.microsoft.com/en-us/azure/devops/boards/get-started/what-is-azure-boards?view=azure-devops&tabs=agile-process> (pristupljeno u junu 2021.)

[9] <https://docs.microsoft.com/en-us/azure/devops/repos/get-started/what-is-repos?view=azure-devops> (pristupljeno u junu 2021.)

[10] <https://docs.microsoft.com/en-us/azure/devops/pipelines/get-started/what-is-azure-pipelines?view=azure-devops> (pristupljeno u junu 2021.)

[11] <https://docs.microsoft.com/en-us/azure/devops/test/overview?view=azure-devops> (pristupljeno u junu 2021.)

[12] <https://docs.microsoft.com/en-us/azure/devops/artifacts/start-using-azure-artifacts?view=azure-devops> (pristupljeno u junu 2021.)

[13] AZ-400 (Handbook) Implementing Continuous Integration, Implementing Continuous Delivery

[14] AZ-203 (Handbook) Develop Azure Platform as a Service compute solutions

Kratka biografija:



Maja Grubor rođena je 27.04.1995. god. u Drvaru. Završila je osnovnu školu „Sveti Sava” u Bačkoj Palanci kao odličan učenik. Gimnaziju „20. Oktobar” završava 2014. godine kao odličan učenik, tokom čijeg pohađanja je učestvovala na takmičenjima iz matematike. Iste te godine upisuje Fakultet tehničkih nauka, smer Elektroenergetski softverski inženjering. U septembru 2018. završava osnovne studije, nakon čega upisuje master studije takođe na Fakultetu Tehničkih nauka, smer Računarstvo i Automatika, Elektronsko poslovanje.

U realizaciji Zbornika radova Fakulteta tehničkih nauka u toku 2021. godine učestvovali su sledeći recenzenti:

Aco Antić	Duško Bekut	Maša Bukurov	Relja Strezoski
Aleksandar Erdeljan	Đorđe Ćosić	Matija Stipić	Slavica Mitrović
Aleksandar Kovačević	Đorđe Lađinović	Milan Čeliković	Slavko Đurić
Aleksandar	Đorđe Obradović	Milan Mirković	Slobodan Dudić
Kupusinac	Đorđe Vukelić	Milan Rapajić	Slobodan Krnjetin
Aleksandar Ristić	Đula Fabian	Milan Segedinac	Slobodan Morača
Bato Kamberović	Đura Oros	Milan Simeunović	Sonja Ristić
Biljana Njegovan	Đurđica Stojanović	Milan Trifković	Srđan Kolaković
Bogdan Kuzmanović	Filip Kulić	Milan Trivunić	Srđan Popov
Bojan Batinić	Goran Sladić	Milan Vidaković	Srđan Vukmirović
Bojan Lalić	Goran Švenda	Milena Krklješ	Staniša Dautović
Bojan Tepavčević	Gordana	Milica Kostreš	Stevan Gostojić
Bojana Beronja	Milosavljević	Milica Miličić	Stevan Milisavljević
Branislav Atlagić	Gordana Ostojić	Mijodrag Milošević	Stevan Stankovski
Branislav Nerandžić	Igor Budak	Milovan Lazarević	Strahil Gušavac
Branka Nakomčić	Igor Dejanović	Miodrag Hadžistević	Svetlana Bačkalić
Branko Milosavljević	Igor Karlović	Miodrag Zuković	Svetlana Nikoličić
Branko Škorić	Igor Peško	Mirjana Damnjanović	Tanja Kočetov
Damir Đaković	Ivan Beker	Mirjana Malešev	Tatjana Lončar -
Danijela Ćirić	Igor Maraš	Miroslava Radeka	Turukalo
Danijela Gračanin	Ivan Mezei	Mirko Borisov	Uroš Nedeljković
Danijela Lalić	Ivan Todorović	Miro Govedarica	Valentina Basarić
Darko Čapko	Ivana Katić	Miroslav Hajduković	Velimir Čongradec
Darko Marčetić	Ivana Kovačić	Miroslav Kljajić	Veran Vasić
Darko Reba	Ivana Maraš	Miroslav Popović	Veselin Perović
Dejan Ecet	Ivana Miškeljin	Miroslav Zarić	Višnja Žugić
Dejan Jerkan	Jasmina Dražić	Mitar Jcanović	Vladimir Katić
Dejan Ubavin	Jelena Atanacković	Mitar Đogo	Vladimir Mučenski
Dejana Nedučin	Jeličić	Mladen Kovačević	Vladimir Strezoski
Dragan Ivanović	Jelena Borocki	Mladen Tomić	Vlado Delić
Dragan Jovanović	Jelena Demko Rihter	Mladen Radišić	Vlastimir Radonjanin
Dragan Ivetić	Jelena Radonić	Nebojša Brkljač	Vojin Ilić
Dragan Jovanović	Jelena Slivka	Neda Milić Keresteš	Vuk Bogdanović
Dragan Kukolj	Jelena Spajić	Nemanja	Zdravko Tešić
Dragan Pejić	Jovan Petrović	Stanisavljević	Zoran Anišić
Dragan Šešlija	Lazar Kovačević	Nemanja Sremčev	Zoran Brujić
Dragana Bajić	Leposava Grubić	Nikola Đurić	Zoran Čepić
Dragana	Nešić	Nikola Jorgovanović	Zoran Jeličić
Konstantinović	Livija Cvetičanin	Nikola Radaković	Zoran Mitrović
Dragana Šarac	Ljiljana Vukajlov	Ninoslav Zuber	Zoran Papić
Dragana Štrbac	Ljiljana Cvetković	Ognjen Lužanin	Željen Trpovski
Dragoljub Šević	Ljubica Duđak	Peđa Atanasković	Željko Jakšić
Dubravka Bojanić	Maja Turk Sekulić	Petar Malešev	
Dušan Dobromirov	Marinko Maslarić	Platon Sovilj	
Dušan Gvozdenac	Marko Marković	Radivoje Dinulović	
Dušan Kovačević	Marko Todorov	Radomir Kojić	
Dušan Uzelac	Marko Vekić	Radovan Štulić	