

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Lucija Matulin

**Razvoj i implementacija simulatora
SCADA sustava laboratorijske
mikromreže**

Zagreb, 2021.

Ovaj rad izrađen je u Smart Grid laboratoriju (SGLab), u Zavodu za visoki napon i energetiku, pod vodstvom izv. prof. dr. sc. Tomislava Capudera i predan je na natječaj za dodjelu Rektorove nagrade u akademskoj godini 2020./2021.

POPIS OZNAKA I KRATICA

- ADS** engl. *Active Distribution System*. 3
- API** engl. *Application Programming Interface*. 11
- DER** engl. *Distributed Energy Resources*. 2
- DG** engl. *Distributed Generation*. 2
- EES** Elektroenergetski sustav. 1, 2
- ESS** engl. *Energy Storage System*. 2
- HMI** engl. *Human-Machine Interface*. 6
- ICT** engl. *Information and Communication Technology*. 1, 7
- MTU** engl. *Master Terminal Unit*. 6
- PC** engl. *Point of Coupling*. 2
- PCC** engl. *Point of Common Coupling*. 2
- PLC** engl. *Programmable Logic Controller*. 6
- RTU** engl. *Remote Terminal Unit*. 6
- SCADA** engl. *Supervisory, Control and Data Acquisition*. 1, 6
- SG** engl. *Smart Grid*. 2, 3
- SGLab** engl. *Smart Grid Laboratory*. 2, 9
- SoC** engl. *State of Charge*. 22
- VPP** engl. *Virtual Power Plant*. 3

Sadržaj

1. Uvod	1
1.1. Formulacija problema	1
1.2. Mikromreža	2
1.3. Napredna elektroenergetska mreža	5
1.4. SCADA sustavi	6
1.5. Simulacije u SG okruženju	7
1.6. SGLab	9
2. Ciljevi rada	12
3. Arhitektura sustava i modeli komponenata	14
3.1. Konfiguracijska datoteka	15
3.2. GatewayHandler	18
3.3. RoutingHandler i DeviceRegistry	20
3.4. GenericWorker	20
4. Rezultati	25
5. Rasprava	30
6. Zaključak	32
Literatura	34
Popis slika	37
Popis tablica	37

1. Uvod

1.1. Formulacija problema

Povećanjem integracije elektroenergetskog sustava (EES) i informacijsko-komunikacijskih tehnologija (ICT) koji pokreću evoluciju naprednih elektroenergetskih mreža, nameće se potreba za odgovarajućim alatima za simulaciju procesa iz obje domene. S obzirom na vrlo složenu prirodu EES-a i na činjenicu da se sastoji od vrlo skupih komponenti velikih snaga (poput generatora, vodova, transformatora), eksperimenti na stvarnom sustavu gotovo su nemogući, nepraktični, skupi i opasni za ljude i opremu. Iz tog razloga često se koriste alati i platforme za simulaciju kojima se oponaša rad i ponašanje stvarnih sustava i njegovih komponenata.

Cilj ovoga rada razviti je novo rješenje - implementaciju simulatora SCADA sustava (engl. *Supervisory, Control and Data Acquisition*) kakav se koristi u laboratorijskoj mikromreži, kako bi se stvorilo sučelje koje povezuje obje domene - za testiranje, razvoj i istraživanje ne samo u području SCADA sustava i mikromreža, nego i korak više prema naprednim elektroenergetskim mrežama i složenim simulacijama u tom okruženju. Razvoj i implementacija simulatora podrazumijevaju:

- modeliranje komponenti mikromreže,
- implementaciju industrijskog komunikacijskog protokola,
- generiranje mjerenja i slanja odgovora na upravljačke signale,

što su izvorni doprinosi ovoga rada.

Izvorni kod cijelog projekta dostupan je na:

<https://github.com/lucijamatulin/SGLab-SCADA-simulator>.

Najbolji prikaz rezultata projekta, dostupan je u obliku kratkog videozapisa na sljedećoj poveznici:

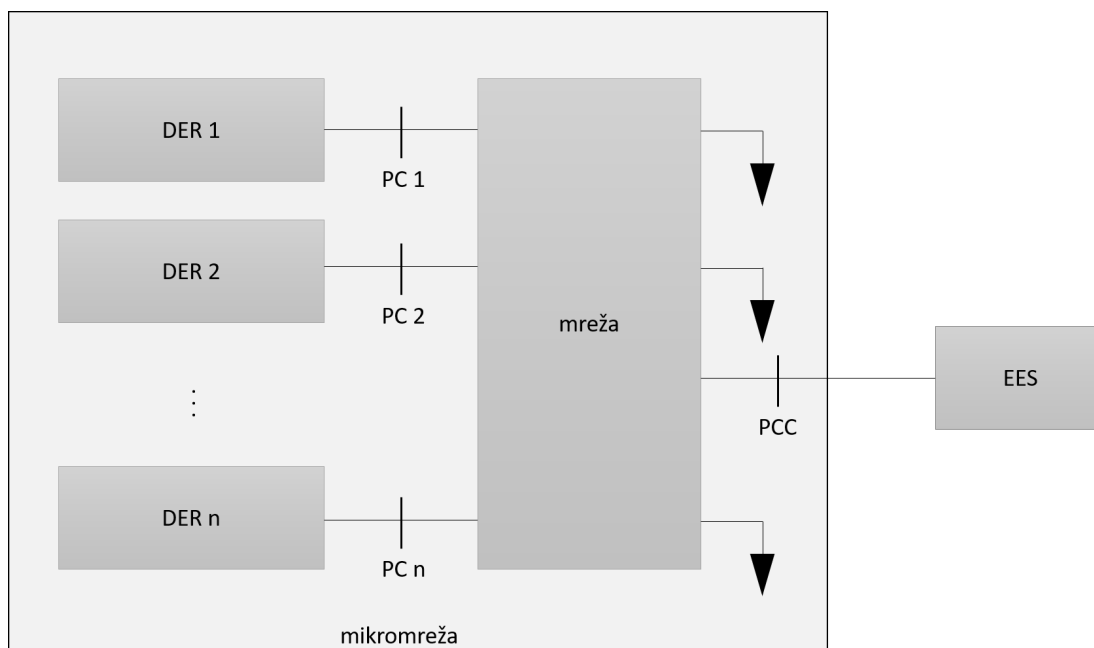
https://github.com/lucijamatulin/SGLab-SCADA-simulator/blob/main/assets/demo_video2.mp4.

Rad je organiziran kako slijedi: u 1. poglavlju dan je teoretski uvod u pojmove i koncepte koji se odnose na mikromreže, napredne elektroenergetske mreže i SCADA sustave. Dan je i kratak pregled problematike simulacija u SG (engl. *Smart Grid*) okruženju s naglaskom na simulacije mikromreža i SCADA sustava. Opisano je laboratorijsko okruženje Laboratorija za napredne elektroenergetske mreže (engl. *Smart Grid Laboratory* (SGLab)) - odnosno laboratorijska mikromreža i dijelovi od kojih se sastoji, koji su integrirani u SCADA sustav - što će biti predmetom za izradu modela, razvoj i parametriranje simulatora SCADA sustava kakav se koristi u spomenutoj laboratorijskoj mikromreži. U 2. poglavlju prezentirani su ciljevi rada. U 3. poglavlju detaljno je opisana arhitektura razvijenog sustava, postupak modeliranja komponenti mikromreže, implementacije komunikacijskog protokola i način generiranja mjerenja i slanja odgovora na upravljačke signale. U 4. poglavlju navedeni su rezultati rada. 5. poglavlje donosi raspravu i diskusiju dobivenih rezultata uz prijedlog budućeg rada. Na kraju su, u 6. poglavlju izneseni zaključci i najvažnije zaključne tvrdnje.

1.2. Mikromreža

Mikromreža se sa stajališta EES-a, odnosno mreže na koju je spojena, može definirati kao jedinstveni entitet - skup jedinica distribuirane proizvodnje (engl. *Distributed Generation*, DG), tereta i spremnika energije (engl. *Energy Storage System*, ESS) - koji odgovaraju na upravljačke signale [1]. Taj klaster elemenata djeluje koordinirano u pogonu s ciljem pouzdane opskrbe električnom energijom. Mikromreža je na EES (najčešće na mrežu distribucijske razine) spojena jednim spojnim mjestom (engl. *Point of Common Coupling*, PCC). Koncept mikromreže se u literaturi pojavio kao rješenje pouzdane integracije distribuiranih izvora energije (engl. *Distributed Energy Resources*, DER) koji osim obnovljivih izvora energije uključuju i sustave pohrane električne energije (ESS) i upravljive terete [1]. Prihvatanje mikromreže kao paradigme za masovnu integraciju distribuirane proizvodnje, omogućuje rješavanje tehničkih problema na decentraliziran način i uz smanjenje potreba za kompleksnom centralnom koordinacijom [1].

Mikromreža općenito može imati proizvoljnu konfiguraciju kao što je prikazano na slici 1.1. Više distribuiranih izvora energije, spojeno je na mrežu spojnim mjestom (engl. *Point of Coupling*, PC), a cijela mikromreža je spojena na EES samo jednim, zajedničkim spojnim mjestom (PCC). Sastoji se od proizvoljnog broja distribuiranih izvora energije i tereta.



Slika 1.1: Konfiguracija mikromreže, prema [1]

Mikromreža i slični entiteti poput aktivnog distribucijskog sustava (engl. *Active Distribution System*, ADS), kognitivne mikromreže ili virtualne elektrane (engl. *Virtual Power Plant*, VPP) smatraju se i mogu se koristiti kao glavni gradivni dijelovi za napredne elektroenergetske mreže (SG) [1]. U pogonu, mikromreža može biti u jednom od dva različita režima rada:

- spojena na krutu mrežu (engl. *grid-connected*) - paralelni pogon s EES-om,
- autonoman rad (engl. *stand-alone*) - otočni režim rada.

Kada je mikromreža spojena na glavnu mrežu deficit električne energije može biti nadoknađen iz proizvodnje u ostatku elektroenergetskog sustava dok suficit može biti prodan u isti - odnosno mikromreža može aktivno sudjelovati na tržištima električne energije i pružati pomoćne usluge. Za razliku od toga, u otočnom režimu rada radna i jalova snaga moraju biti u ravnoteži. Postoje i mikromreže bez spoja na glavnu mrežu - izolirane mikromreže u slučaju udaljenih zajednica ili mjesta kojima treba osigurati napajanje električnom energijom, a interkonekcija na glavnu mrežu nije moguća ili nije tehnički ili ekonomski opravdana [1]. Takve mikromreže stalno i trajno rade u autonomnom režimu rada, a male autonomne mikromreže postoje već desetljećima u odvojenim i udaljenim zajednicama. Poželjna karakteristika mikromreže je rad u oba režima i stabilan prijelaz između njih. Za svaki od režima rada, mogu se definirati različite upravljačke strategije [1].

S obzirom na arhitekturu upravljačkog sustava postoje dva potpuno suprotna pristupa: centralizirani i decentralizirani [1]. Centralizirani pristup se oslanja na podatke prikupljene preko centralnog kontrolera koji izvodi potrebne izračune i određuje upravljačke akcije za sve jedinice na jednom mjestu. U tom je pristupu potreban kompleksan i pouzdan komunikacijski sustav između centralnog kontrolera i svih jedinica kojima se upravlja. S druge strane, decentralizirani pristup se temelji na upravljanju svakom jedinicom putem zasebnog, lokalnog kontrolera koji prima lokalne informacije, nisu mu poznate vrijednosti svih varijabli u sustavu ni dostupne akcije svih drugih kontrolera. Oba ova pristupa imaju svoje nedostatke:

- centralizirani - potreban kompleksan komunikacijski sustav, velika računalna složenost potrebnih izračuna,
- decentralizirani - ako postoji povezanost između operacija različitih jedinica u sustavu, minimalna razina koordinacije ne može biti izvedena samo lokalnim varijablama.

Hijerarhijski pristup je kompromis između centraliziranog i decentraliziranog pristupa, a sastoji se od 3 razine - primarne, sekundarne, tercijarne. Na te 3 upravljačke razine razlika je u vremenu odziva i vremenskom okviru unutar kojeg djeluju, kao i u infrastrukturnim zahtjevima (npr. komunikacijski zahtjevi). Hijerarhijski pristup ima prednosti ako se gledaju različite vremenske konstante koje su prisutne u sustavu, uključujući brzu dinamiku izlaznih varijabli i sporiju u ekonomskom dispečiranju [1].

S obzirom na arhitekturu sustava i naponske karakteristike, mikromreže se mogu klasificirati u 3 kategorije [2]: izmjenične (AC), istosmjerne (DC) i hibridne (AC/DC).

AC mikromreža je konvencionalni tip mreže u kojem su distribuirani izvori energije (npr. vjetrene elektrane, fotonaponski paneli, mikroturbine, gorivne ćelije) integrirani preko uređaja učinske elektronike. Ovakav tip mikromreže zahtjeva minimalne modifikacije prilikom integracije u postojeću mrežu i spaja se uglavnom na srednjenaponsku ili niskonaponsku razinu distribucijske mreže.

Razvojem učinske elektronike, koristi se velik broj DC tereta, pretvarača, spremnika energije i proizvodnih jedinica. Otprilike 30% snage prolazi kroz uređaje energetske elektronike prije nego što se koristi [2]. U tom slučaju može se koristiti DC mikromreža čije su glavne prednosti limitirani broj razina konverzije i nema tokova jalove snage.

Hibridne AC/DC mikromreže kombiniraju prednosti oba svijeta za povećanu pouzdanost, efikasnost i ekonomičnost operacija. Moguća je direktna integracija i AC

i DC baziranih proizvodnih jedinica, spremnika, tereta u postojeći distribucijski sustav. Ovakve mreže smanjuju broj potrebnih pretvarača i smanjuju gubitke. Nedostatak hibridnih AC/DC mikromreža je kompleksnost mreže, potrebno je koordinirano upravljanje individualnim AC i DC dijelovima i kompenzacija jalove snage [2].

1.3. Napredna elektroenergetska mreža

Napredna elektroenergetska mreža (SG) naziv je za kompleksni sustav sastavljen od komponenata elektroenergetskog sustava (EES) zajedno s komponentama komunikacijskih mreža - omogućuje distribuirano upravljanje EES-om putem kontrolera, senzora i aktuatora [3]. Razvoj SG ključan je faktor u koordinaciji različitih vrsta sustava, tehnologija i infrastruktura poput mikromreža, električnih vozila i obnovljivih izvora energije.

Cilj SG je omogućavanje pristupačnog, pouzdanog i održivog načina opskrbe električnom energijom uz inteligentnu integraciju akcija svih korisnika - od proizvodnih jedinica, krajnjih potrošača i ostalih sudionika [4], [5], [6]. Iz tih razloga pojavile su se inicijative diljem svijeta poput SAD-a (*IntelliGrid* inicijativa instituta EPRI) i Europe (*SmartGrids*) s ciljem formuliranja i promoviranja vizije razvoja budućih naprednih elektroenergetskih mreža [4].

Izazovi i potrebe za razvojem SG mogu se sažeti u 4 glavna aspekta [4]:

1. okolišni izazovi:

- tradicionalna proizvodnja električne energije velik je izvor emisije stakleničkih plinova, potrebne su velike promjene u proizvodnji kako bi se smanjili učinci klimatskih promjena,
- limitirana količina fosilnih goriva i njihov mogući nedostatak u budućnosti,
- prostorno proširenje EES - dostupan i prigodan prostor se drastično smanjuje, a uz stalno rastuće potrebe za električnom energijom potrebno je širenje kapaciteta,

2. potrebe tržišta i krajnjih korisnika:

- transparentnost i kompetitivnost u uvjetima liberaliziranog tržišta,
- povećanje razine korisničkog zadovoljstva u opskrbi - visok omjer kvalitete i cijene električne energije,

- omogućiti slobodu interakcije korisnika s mrežom,
3. infrastrukturni izazovi:
 - postojeća infrastruktura sadrži komponente koje zastarijevaju i nema dovoljno investicija za poboljšanje,
 - stalnim povećanjem potreba za električnom energijom sve je veći pritisak na mrežu i mogu se pojaviti zagušenja u mreži,
 - potrebni su brzi alati za online analizu sustava, sustavi nadgledanja, mjerenja i upravljanja, brza i točna zaštita - kako bi se poboljšala pouzdanost mreže,
 4. inovativne tehnologije - od novih materijala, naprednih uređaja energetske elektronike do informacijsko-komunikacijske tehnologije - s jedne strane tehnologije nisu dovoljno sazrele ili nisu komercijalno dostupne kako bi se dogodila revolucija, a s druge strane postojećoj mreži nedostaje kompatibilnosti za prihvatanje novih tehnologija.

1.4. SCADA sustavi

SCADA sustavi (engl. *Supervisory, Control and Data Acquisition System*) dio su industrijskog okruženja i primarna im je namjena automatizirani nadzor i upravljanje komponentama sustava [6]. Koriste se u različitim industrijama i za nadgledanje i upravljanje različitom infrastrukturom. Sastoje se od mjernih instrumenata, logičkih kontrolera poput PLC-a (engl. *Programmable Logic Controller*) ili RTU-a (engl. *Remote Terminal Unit*), glavne jedinice MTU (engl. *Master Terminal Unit*), komunikacijske mreže i sučelja čovjek-stroj (engl. *Human-Machine Interface*, HMI) [6].

Mjerni instrumenti se sastoje od senzora koji prate fizikalna mjerenja poput temperature, tlaka i napona [6]. Logički kontroleri su uglavnom odgovorni za prikupljanje podataka s mjernih instrumenata, detekciju abnormalnog ponašanja, aktiviranja ili deaktiviranja tehničkih komponenata. MTU je centralna komponenta kojom operator sustava šalje upravljačke signale na logičke kontrolere i prima podatke. Interakcija između MTU-a i logičkih kontrolera ostvarena je putem komunikacijske mreže. Komunikacijske se mreže baziraju na industrijskim protokolima poput Modbus-a [7], DNP3 (engl. *Distributed Network Protocol 3*) [8], EtherNet/IP [9], IEC 60870-5-104 [10]. HMI je programski paket s grafičkim mogućnostima instaliran na MTU i ostvaruje interakciju između MTU-a i logičkih kontrolera [6].

Moderni SCADA sustavi imaju mnoštvo značajki koje znatno povećavaju kompleksnost sustava i teške su za održavanje [11]. Neke od tih značajki uključuju upravljačku logiku, komunikacijske protokole, korisnička sučelja, sigurnost. Mnoge organizacije i institucije koje koriste SCADA sustave ne toleriraju kašnjenje ili gubitak podataka - sustav mora raditi kontinuirano i unutar uskog vremenskog okvira.

SCADA sustav se sastoji od entiteta organiziranih u hijerarhijsku strukturu [11]: integrirani sustavi za dohvat podataka, prijenos podataka i sučelja čovjek-stroj koji povezuje čovjeka s uređajem, vizualizira podatke i omogućuje nadzor ulaznih i izlaznih veličine.

SCADA sustavi imaju neku od 4 tipične vrste arhitekture [11]:

1. monolitska - kontrolne jedinice i MTU povezani žicom s RTU-om,
2. distribuirana - MTU i RTU komuniciraju jedino preko komunikacijskih protokola i servera, ne dopuštaju internetsku konekciju,
3. umrežena - koriste se vanjske mreže poput interneta,
4. web-bazirana - u novije vrijeme, korisnici mogu pristupiti putem web preglednika ili mobilnih uređaja.

Evolucija i razvoj SCADA sustava dovode do povećane kompleksnosti, a za to su odgovorne neke od sljedećih značajki: nove komponente (računala, stanice, komunikacijski serveri), povećanje količine izmjenjivanih podataka između jedinica, povećan broj komponenata sustava i interakcija između njih, korištenje vatrozida i antivirusa koji usporavaju cijeli sustav, smanjuju računalnu moć i na kraju dovode do kašnjenja u prijenosu podataka [11].

1.5. Simulacije u SG okruženju

U tranziciji prema SG-u, EES postaje snažno isprepleten infrastrukturom informacijsko-komunikacijske tehnologije (ICT) i međusobna ovisnost ovih dviju domena zahtijeva kombiniranu analizu fizikalnih i ICT procesa [12]. Simuliranje fenomena iz obje domene zajedno može predstavljati velike izazove obzirom na fundamentalno različite koncepte modeliranja i simuliranja, zato je neophodan cjelovit pristup.

Modeliranje i simuliranje ključne su metode u rješavanju inženjerskih problema, a danas se koriste u gotovo svim domenama tehnologije [12]. Moderni EES uvodi DG, DER, inteligentne terete, tržišta u stvarnom vremenu - sve je to omogućeno i osnaženo korištenjem ICT tehnologija - što omogućava EES-u da bude što fleksibilniji i efikasniji. Međutim, prije svega je potrebno razumjeti kompleksnu interakciju između digitalnog svijeta i fizičke infrastrukture. ICT u jednom predstavlja i rješenje i problem istovremeno: omogućuje kompleksno upravljanje u okruženju SG uvodeći pritom dodatnu kompleksnost, nove izvore kvarova i sigurnosnih prijetnji. Sve to zahtijeva i novu razinu detalja u simulacijama EES-a. U ovom području, zajedničkih simulacija EES-a i ICT sustava, mnoga se rješenja još istražuju i razvijaju.

Kada se promatraju mogućnosti i izazovi u modelima i simulacijama, pogled na EES sustav mora biti širi, uključujući višestruke fizikalne domene i detaljne upravljačke sheme bazirane na ICT rješenjima. Rješenja drugih industrijskih sektora (poput automobilske) koji su u prošlosti imali sličan razvojni put kao i elektroenergetski sektor, nisu u potpunosti primjenjivi. Elektroenergetski sustavi su veliki sustavi s velikim brojem komponenata koje djeluju na različitim vremenskim skalama [12].

Važno je napomenuti da glavne poteškoće u razvoju kombiniranih modela komunikacija i EES-a ne proizlaze iz nedostatka primjenjivih matematičkih i numeričkih metoda, kako su ti fundamentalni problemi već ranije razmatrani i razriješeni u nekoliko različitih konteksta prigodnih za simulaciju i modeliranje pametnih EES-a. Problem je u troškovima i kompleksnosti izrade novih modela unutar novih simulacijskih programskih okvira za hibridne sustave što čini proces teškim [12]. Postoji snažna inicijativa da se iskoriste postojeći simulacijski alati i modeli unutar njih, ali uvijek postoji problem kompromisa i ravnoteže između performansi. Kompromis se može prikazati kao izbor između 3 glavne opcije [12]:

1. iskoristiti postojeće modele - modeli komunikacijskih sustava i EES-a postoje u različitim, odvojenim, ali afirmiranim simulacijskim paketima,
2. točne simulacije interakcija između ICT sustava i EES-a,
3. brza izvedba kombinirane simulacije.

Kakva će kombinacija biti odabrana, ovisi o procjeni vrijednosti ponovnog iskorištenja komponenata, točnosti i brzine izvođenja unutar specifičnog konteksta određene simulacijske studije [12].

Testiranje kompleksnih mikromreža zahtijeva dostupnost specijaliziranih laboratorija i međusobno povezanih uređaja [13]. Napredne simulacijske metode i tehnike poput HIL (engl. *Hardware-In-The-Loop*) se koriste kako bi se izbjeglo uništavanje vrlo skupe opreme, jamči sigurnost i nudi mnogo različitih mogućnosti testnih uvjeta. Takve metode omogućavaju konekciju fizičke komponente (npr. ispravljača ili sličnog uređaja učinske elektronike) na simuliranu mrežu kako bi se promatralo ponašanje i rad stvarne opreme [14]. U EES-u se takve simulacije najčešće koriste za promatranje zaštitnih sustava i opreme - komercijalni releji se mogu testirati provođenjem izazovnih testova poput kvarova u sigurnom okruženju, u stvarnom vremenu posredstvom simulacijskog okruženja, bez štetnih posljedica, efekata i utjecaja na opremu [14].

SCADA sustavi su često dijelom industrijske i kritične infrastrukture i stoga je nepraktično vršiti eksperimente na živim sustavima zbog troškova implementacije zasebnih testnih sustava, potencijalnih rizika i troškova uzrokovanih jer je kontrolirani proces ili sustav za to vrijeme izvan pogona [15]. Simulacije SCADA sustava moraju imati podršku za integraciju različitih vanjskih uređaja i aplikacija. Takvi simulacijski alati i okruženja mogu se koristiti i za edukaciju korisnika kako provoditi upravljačke operacije [16].

1.6. SGLab

Laboratorij za napredne elektroenergetske mreže (engl. *Smart Grid Laboratory*, SGLab) napredni je laboratorij za elektroenergetske sustave, ustrojen 2015. godine [17]. Okuplja istraživače *Zavoda za visoki napon i energetiku* [18], *Fakulteta elektrotehnike i računarstva*, *Sveučilišta u Zagrebu* s ciljem znanstvenog istraživanja, profesionalnog rada i edukacije u polju naprednih elektroenergetskih sustava. Primarni je fokus na istraživanju utjecaja obnovljivih izvora energije i distribuirane proizvodnje na EES, s povećanom potražnjom za uslugama fleksibilnosti sustava zbog integracije i korištenja tehnologija poput: sustava za pohranu - spremnika energije, električnih vozila i postaja za punjenje električnih vozila, sinkrofazora, agregacije odziva potrošnje, mikromreže, višegeneracijskih sustava. Umreženost i upravljanje velikim brojem tehnologija i uređaja postignuto je primjenom ICT rješenja.

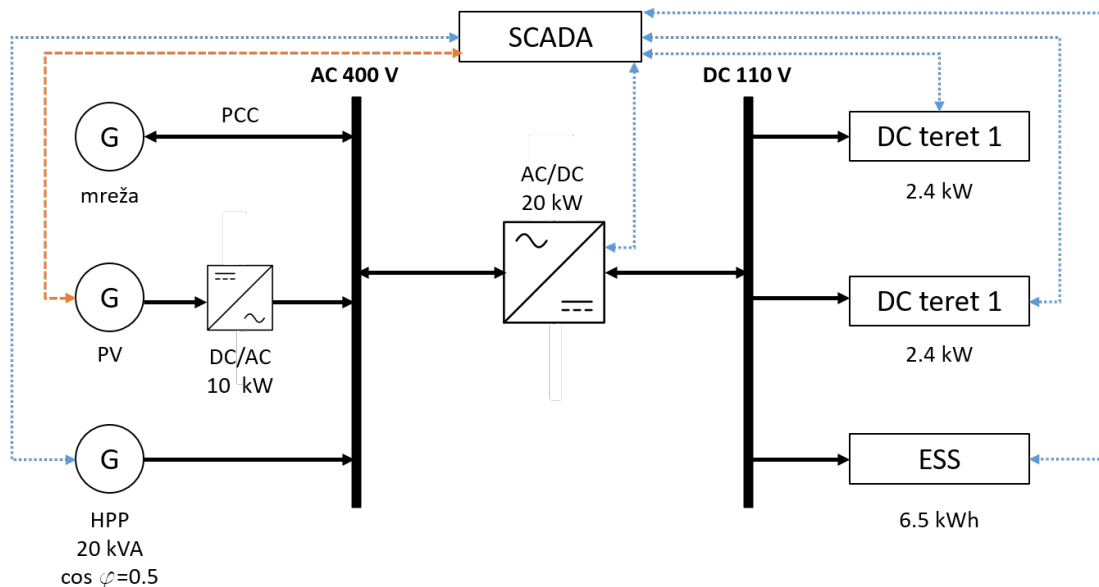
Laboratorij posjeduje mnoštvo uređaja i opreme, a najvažnije za spomenuti je laboratorijska mikromreža koja se sastoji od:

- modela hidroelektrane s Peltonovom turbinom,

- 2 upravljiva DC tereta,
- dvosmjernog AC/DC pretvarača za spajanje AC i DC dijelova mikromreže,
- fotonaponskih panela,
- baterijskih spremnika energije.

Detaljniji opis komponenti koje se koriste u procesu modeliranja sustava i popis njihovih parametara, dan je u poglavlju 3. Laboratorijska mikromreža se može koristiti i za simuliranje rada niskonaponske distribucijske mreže.

Sve komponente integrirane su u SCADA sustav PROZA NET [19], koji podržava velik broj komunikacijskih protokola kao što su IEC 61850 [20], IEC 60870-5-104 [10], Modbus TCP/IP [7], DNP3 [8] i drugi. U ovom postavu mikromreže, koriste se OPC UA (engl. *Unified Architecture*, UA) [21] i Modbus TCP/IP [7] protokoli za komunikaciju među komponentama mikromreže [22]. Shematski prikaz komponenata mikromreže i komunikacijskih protokola, može se vidjeti na slici 1.2: osim spomenutih komponenata, prikazani su i tokovi električne energije (crnom punom linijom) i informacija (obojene isprekidane linije). Crnom punom linijom prikazani su spojevi komponenti na električnu mrežu, AC i DC dijelove mikromreže. Isprekidanom linijom su prikazani komunikacijski protokoli: narančasta boja označava Modbus TCP/IP, a plava boja OPC UA.



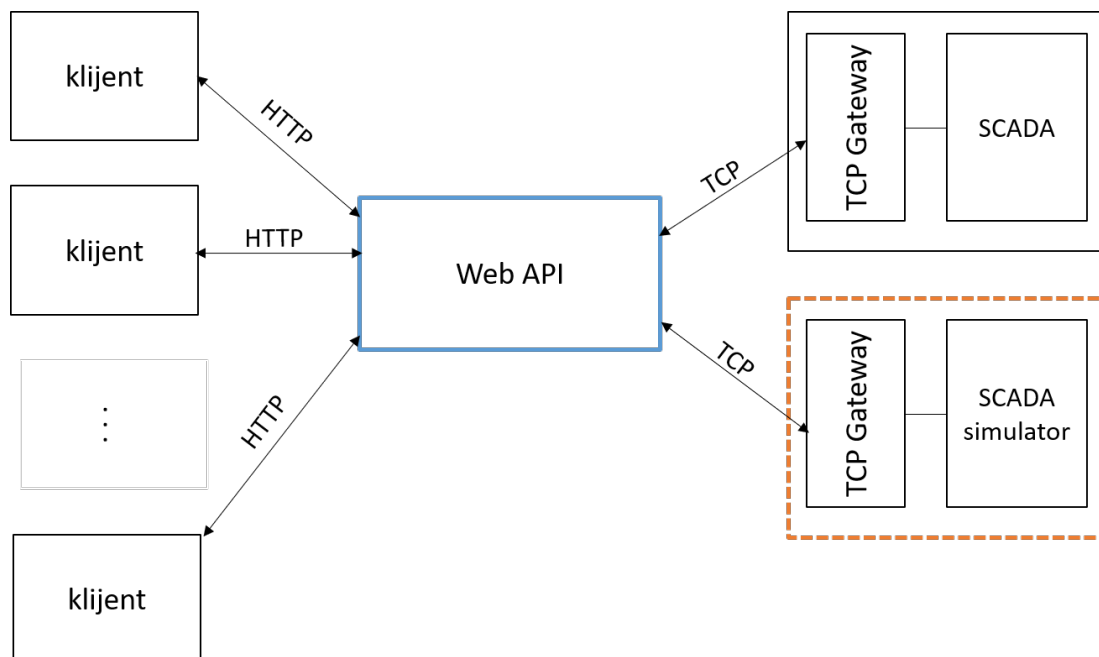
Slika 1.2: Shematski prikaz laboratorijske mikromreže, SGLab - prema [22]

Razvijene klijentske aplikacije koje se nadovezuju na rad SCADA sustava, sa SCADA sustavom komuniciraju posredstvom aplikacije SCADA Web API (engl. *Application Programming Interface*, API) koja se prilagođenim protokolom implementiranim na TCP/IP mrežnom sloju spaja na SCADA sustav. U toj komunikaciji SCADA sustav predstavlja poslužitelja (engl. *Server*), a SCADA Web API predstavlja posrednog poslužitelja (engl. *Proxy*) za klijenta (engl. *Client*).

2. Ciljevi rada

U posljednje vrijeme, razvijaju se klijentske aplikacije koje se nadovezuju na rad SCADA sustava, odnosno dohvaćaju podatke u obliku mjerenja pojedinih uređaja, vizualiziraju podatke putem HMI-a, šalju referentne vrijednosti određenih varijabli sustava i naredbe. Razvoj takvih aplikacija iterativni je proces koji zahtijeva stalno testiranje - stoga bi bilo od iznimne važnosti razviti sustav koji bi oponašao rad SCADA sustava koji se koristi. Na taj način ubrzao bi se i olakšao proces razvoja aplikacija na udaljen način, bez odlaska u laboratorij što je od velike važnosti posebice u situacijama poput "lockdown"-a - kada fizički pristup laboratoriju nije moguć.

Simulator SCADA sustava zamišljen je da simulira SCADA sustav laboratorijske mikromreže, opisane u prethodnom poglavlju. Arhitektura budućeg sustava, opisanog u prethodnom poglavlju, a dopunjena simulatorom, ilustrirana je slikom 2.1.



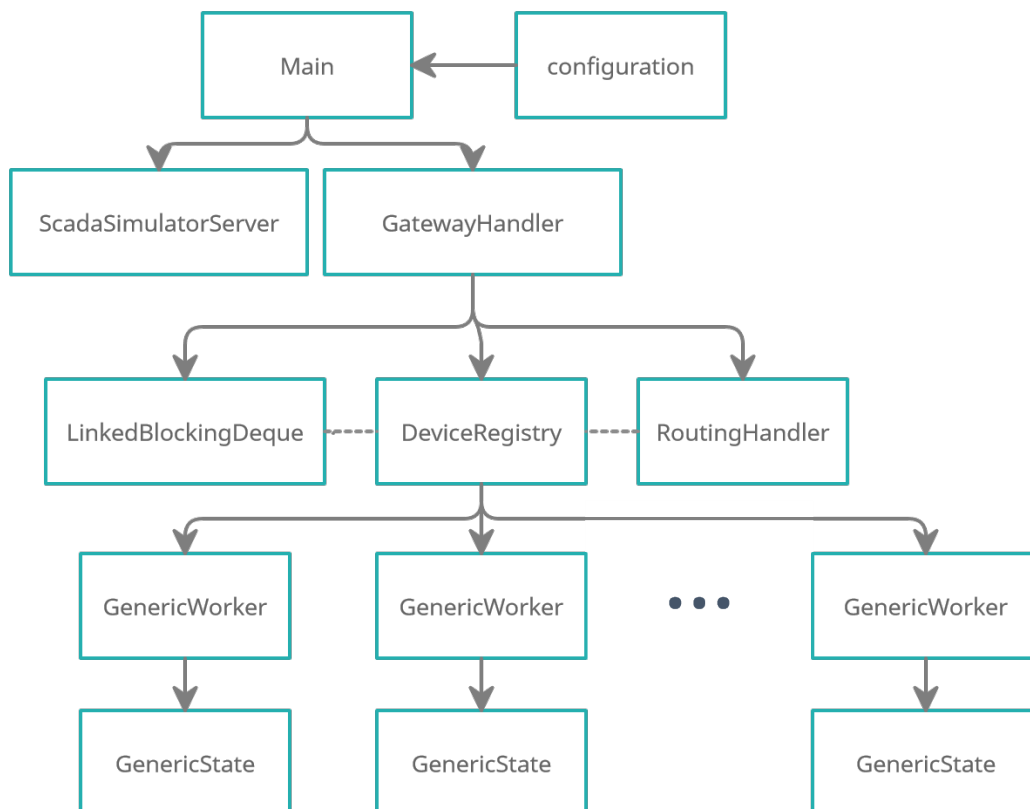
Slika 2.1: Arhitektura sustava: simulator ilustriran narančastim isprekidanim pravokutnikom

Razvoj simulatora uključuje konfiguriranje simulatora, strojeva, mjerenja i šumova u sustavu i podacima, kao i implementaciju komunikacijskih sučelja koji su preslika sučelja kakvo se koristi. Simulator je potrebno parametrirati i modificirati njegovo ponašanje ovisno o poslanim referentnim vrijednostima i naredbama. Za to će se koristiti konfiguracijska datoteka u kojoj će biti navedeni svi željeni parametri i u skladu s time, bit će ostvareno konfiguriranje simulatora. Osim toga potrebno je i implementirati korišteni industrijski komunikacijski protokol za razmjenu podataka na TCP/IP mrežnom sloju u skladu sa zadanom specifikacijom.

Ključni dio ovoga rada je generiranje mjerenja s uređaja na pametan način, odnosno na način da se reagira na određeni zadani upravljački signal. Drugim riječima, mjerenja će se generirati ovisno o naredbama i poslanim vrijednostima iz vanjskog izvora - klijentskih aplikacija (npr. simulator počinje generirati mjerenja tek nakon što je poslana naredba *start* ili sličan ekvivalent). Potrebno je definirati šum u sustavu i podacima. U konfiguracijskoj datoteci definiraju se i vrste signala (kontinuirani ili diskretni), kao i raspon parametara i mjerenja - parametri ne mogu poprimiti vrijednosti izvan zadanih raspona.

3. Arhitektura sustava i modeli komponenata

Aplikacija *SGLab SCADA simulator* pisana je u programskom jeziku *Java* (Java SE 11 LTS) u razvojnom okruženju *IntelliJ IDEA*. Arhitektura aplikacije prikazana je na slici 3.1, gdje je moguće vidjeti sve glavne module. Ostale komponente koje predstavljaju modele vezane za konfiguraciju (paket *config*), poruke (paket *models/messages*), vrijednosti parametara poruka (paket *models/values*) ovdje nisu prikazani.



Slika 3.1: Arhitektura aplikacije *SGLab SCADA simulator*

Main glavni je modul (engl. *Class*) i predstavlja ulaznu točku aplikacije. Čita konfiguracijsku datoteku *configuration.yml*, stvara instance konfiguracijskog modela (*Configuration Model*), *ScadaSimulatorServer*-a i *GatewayHandler*-a. U nastavku će osim spomenutih modula, biti opisane i ostale komponente arhitekture.

Configuration Model modul je koji preslikava strukture iz konfiguracijske datoteke u varijable koje se dalje koriste na mjestima u programu gdje su potrebne vrijednosti tih varijabli (npr. konfiguracijski model koriste moduli *ScadaSimulatorServer* i *GatewayHandler*).

ScadaSimulatorServer modul je koji je zaslužan za pokretanje web-sučelja aplikacije, za što su korišteni web razvojni okviri *Javalin* i *Angular*.

3.1. Konfiguracijska datoteka

Konfiguracijska datoteka *configuration.yml*, pisana je u tzv. *markup* jeziku *YAML*, jeziku za serijalizaciju podataka i uobičajeno se koristi za konfiguracijske datoteke. Konfiguracijska datoteka se koristi za sve parametre simulatora (verzija, ime, podaci za autentifikaciju - korisničko ime i lozinka, podaci vezani za komunikacijski protokol - mrežna vrata, duljina poruke u byte-ovima), kao i podatke o komponentama mikromreže - kao što je prikazano u nastavku, isječkom programskog koda 1. Dijelovi mikromreže navedeni su u listi, a opisani su svojim identifikacijskim brojem, podacima poput imena, opisa, zastavice radi li se o upravljivoj komponenti ili ne, i 2 mape (*measurements*, *commands*) u kojima su navedena dostupna mjerenja i naredbe određene elektrane ili uređaja.

Isječkom programskog koda 2 i 3, u dijelu dani su primjeri za hidroelektranu, pripadno mjerenje radne snage (isječak 2) i naredba *start* (isječak 3).

U mapi mjerenja ključ je naziv mjerenja (*activePower*), a vrijednost se sastoji od strukture u kojoj se nalaze ključ za određeno mjerenje, kojeg je tipa vrijednost mjerenja, granice - minimalna i maksimalna vrijednost parametra, kao i iznosi za koliko se vrijednost može promijeniti svake sekunde.

U mapi naredbe ključ je naziv naredbe (*start*), a vrijednost se sastoji od strukture u kojoj se nalaze ključ za određenu naredbu, kojeg je tipa poruka kojom se šalje naredba i tip vrijednosti koji se šalje.

```

version: int
info:
  name: String
authentication:
  username: String
  password: String
communication:
  port: int
  messageLength: int
machines:
- id: int
  data:
    name: String
    description: String
    controllableUnit: boolean
  measurements: Map
  commands: Map

```

Listing 1: Isječak koda konfiguracijske datoteke, *configuration.yml*

Za svaku komponentu mikromreže napravljena je preslika komunikacijskih sučelja iz stvarnog SCADA sustava na način da su navedena mjerenja i naredbe, a iznosi parametara koriste se u postupku generiranja mjerenja i odgovaranja na naredbe kao upravljačke signale. U tablici 3.1 u nastavku navedeni su iznosi parametara pojedinih uređaja, a u tablici 3.2 dostupna mjerenja i naredbe po uređajima.

Tablica 3.1: Parametri komponenata mikromreže

Hidroelektrana	
cos ϕ	0.5
radna snaga [kW]	[0.8, 11.8]
iznos porasta snage [kW/s]	0.8
iznos smanjenja snage [kW/s]	0.7
Fotonaponski paneli	
cos ϕ	0.95
prosječna dnevna izlazna snaga [kWh/h]	0, 0, 0, 0, 0, 3, 9, 19, 29, 38, 43, 45, 44, 40, 34, 26, 17, 8, 3, 0, 0, 0, 0, 0
prosječni dnevni izlaz [MWh/dan]	0.259
prosječni godišnji izlaz [MWh/dan]	94.676
AC/DC pretvarač	

radna snaga [kW]	[0, 20]
iznos porasta snage [kW/s]	10
iznos smanjenja snage [kW/s]	10
napon [V]	[0, 400]
struja [A]	[0, 63]
DC teret 1	
radna snaga [kW]	[0, 2.4]
iznos porasta snage [kW/s]	1
iznos smanjenja snage [kW/s]	1
napon [V]	[0, 750]
struja [A]	[0, 40]
DC teret 2	
radna snaga [kW]	[0, 2.4]
iznos porasta snage [kW/s]	1
iznos smanjenja snage [kW/s]	1
napon [V]	[0, 400]
struja [A]	[0, 50]
Baterijski spremnici	
nazivni kapacitet [kWh]	6.5
$\cos \phi$	0.95
radna snaga [kW]	[0, 2.5]
iznos porasta snage [kW/s]	2
iznos smanjenja snage [kW/s]	2
iznos napunjenosti [%]	[0, 100]

Tablica 3.2: Mjerenja i naredbe komponenta mikromreže

komponenta	mjerenja	naredbe
hidroelektrana	radna snaga, <i>activePower</i> jalova snaga, <i>reactivePower</i> prividna snaga, <i>apparentPower</i>	<i>start</i> <i>stop</i> <i>pref</i>
fotonaponski paneli	radna snaga, <i>activePower</i> jalova snaga, <i>reactivePower</i> prividna snaga, <i>apparentPower</i> ograničenje radne snage,	-

	<i>activePowerLimitation</i> ukupni iznos dobivene energije, <i>totalEnergyFed</i> ukupni dnevni iznos dobivene energije, <i>currentDayTotalEnergyFed</i>	
pretvarač	radna snaga, <i>activePower</i> napon, <i>voltage</i> struja, <i>current</i>	<i>activate</i> <i>uref</i> <i>pref</i> <i>discharge</i>
DC tereti	radna snaga, <i>activePower</i> napon, <i>voltage</i> struja, <i>current</i>	<i>activate</i> <i>iref</i> <i>pref</i>
baterijski spremnici	radna snaga, <i>activePower</i> prividna snaga, <i>apparentPower</i> stanje napunjenosti, <i>stateOfCharge</i>	<i>activate</i> <i>discharge</i> <i>pref</i>

3.2. GatewayHandler

GatewayHandler modul je u kojem je implementiran komunikacijski protokol na TCP/IP mrežnom sloju, tzv. *PNETGateway* komunikacijski protokol. Najprije se otvara priključnica poslužitelja (engl. *Server Socket*) i ako postoji klijent, uspostavlja se komunikacija između njih i dohvaćaju se ulazni i izlazni tokovi podataka. Ulazni tok podataka služi za dohvat podataka koje šalje klijent, a izlazni za slanje podataka klijentu. Istovremeno se u 2 različite dretve procesiraju poruke:

1. zahtjevi: *AuthenticationRequest*, *ChangeDataRequest*, *CommandRequest* - to su poruke koje šalje klijent sa zahtjevom za autentifikaciju (na početku uspostavljene komunikacije), za promjenom vrijednosti nekog podatka ili za izvršenjem neke naredbe, a poslužitelj na njih odgovara porukama *AuthenticationResponse*, *ChangeDataResponse*, *CommandResponse*,
2. tzv. *unsolicited* poruke: *DataChangedUnsolicited*, *StatusChangedUnsolicited* - poruke koje server šalje klijentu periodički ili nakon promjene neke vrijednosti.

Poslužitelj obrađuje zahtjeve na sljedeći način - čita poruke koje pristižu s ulaznog toka podataka, priprema odgovor i šalje ga klijentu preko izlaznog toka podataka.

```
measurements:
  activePower:
    key: String
    dataType: DOUBLE
    bounds:
      min: 0.8 #[kW]
      max: 11.8 #[kW]
    rampUp: 0.8 #[kW/s]
    rampDown: 0.7 #[kW/s]
```

Listing 2: Isječak programskog koda konfiguracijske datoteke, mjerenje *activePower*

```
commands:
  start:
    key: String
    messageType: COMMAND
    dataType: BOOLEAN
```

Listing 3: Isječak programskog koda konfiguracijske datoteke, naredba *start*

Poruke su u *JSON* (engl. *JavaScript Object Notation*) formatu, kodirane standardom *UTF-8*. Programskim isječkom 4 u nastavku, prikazana je općenita struktura poruke koja se sastoji od tipa i tijela poruke. *JSON paketi* se šalju u byte-ovima i to na način da je prva 4 byte-a paketa zapisana duljina poruke N , a ostatak paketa čini poruka duljine N .

```
{
  "type": MESSAGE_TYPE,
  "body": Object
}
```

Listing 4: Općenita struktura poruke

Odgovor poslužitelja na zahtjev ovisi o tipu poruke, a osim pripreme odgovora, za određeni tip poruka usmjeren je poziva i tzv. *Worker* koji je zaslužan za obradu zahtjeva za promjenom određene vrijednosti, odnosno naredbe. Odgovor poslužitelja u *JSON* formatu se također mora zapisati u byte-ove prije slanja putem izlaznog toka podataka.

Obrada tzv. *unsolicited* poruka nešto je drugačija budući da ih poslužitelj šalje klijentu bez zahtjeva. Postupak generiranja *unsolicited* poruka bit će opisan kasnije, a njihova obrada se sastoji od uzimanja generirane poruke s reda (engl. *queue*) i zapisivanja u odgovarajući zapis prije slanja putem izlaznog toka podataka, na ranije opisani način kao za ostale poruke.

3.3. RoutingHandler i DeviceRegistry

RoutingHandler modul je koji predstavlja implementaciju usmjeritelja - služi da bi određeni zahtjev za promjenom vrijednosti ili naredbu, usmjerio prema tzv. *Worker*-u odgovarajuće elektrane ili uređaja radi daljnjeg procesuiranja. Ima interne mape *measurementToDevice* i *commandToDevice* u kojima su ključevi vrijednosti ključeva mjerenja ili naredbi, a vrijednosti su identifikacijski brojevi uređaja. Pomoću tih mapa može odrediti koji je uređaj u pitanju kada dobije ključ mjerenja ili naredbe, a zatim i vratiti instancu traženog tzv. *Worker*-a, kako bi se mogla nad njime pozvati metoda za obradu naredbe.

DeviceRegistry modul je koji vodi računa o *Worker*-ima. Svaka elektrana ili uređaj ima svoju instancu *Worker*-a koji radi u zasebnoj dretvi, paralelno s ostalima. Najprije se stvaraju nove instance *Worker*-a, predaju se parametri uređaja i započinju se odvijati paralelni procesi svih uređaja u zasebnim dretvama.

3.4. GenericWorker

GenericWorker modul je koji predstavlja implementaciju sučelja *WorkerInterface*, a služi za generiranje mjerenja i *unsolicited* poruka, promjenu referentnih vrijednosti i obradu naredbi. Svaki uređaj prima svoj dio konfiguracijske datoteke za inicijalizaciju parametara, pomoću kojih određuje svoje stanje predstavljeno modulom *GenericState*. *GenericState* sastoji se od varijabli koje predstavljaju ranije nabrojana mjerenja uređaja i status - je li uređaj u pogonu, ili nije, ili je u procesu izlaska iz pogona.

Inicijalizacija stanja ovisi o uređaju - ona mjerenja koja posjeduje uređaj, inicijalizirat će se na svoju početnu vrijednost koja je u slučaju upravljivih komponenata (sve komponente predmetne mikromreže osim fotonaponskih panela) jednaka 0.0, obzirom da na početku uređaji nisu u pogonu. Zatim se mapiraju vrijednosti ključeva naredbi s imenima naredbi, što se koristi za obradu naredbi po imenu.

Za svaki uređaj, svake sekunde se u red stavlja poruka tipa *DataChangedUnsolicited*, koja služi za osvježavanje vrijednosti mjerenja. U slučaju da je uređaj izvan pogona, šalje se 0.0, a inače se na željenu vrijednost dodaje i generirana vrijednost šuma u podacima. Vrijednost šuma u podacima i u sustavu, modelira se kao slučajna varijabla s jediničnom normalnom razdiobom - vrijednost je moguće aproksimirati na ovaj način iz razloga što šum nastaje uslijed mnogih nesavršenosti i faktora (zajednički doprinos svih mjernih, mehaničkih, regulacijskih i komunikacijskih komponenti sustava), a normalna se razdioba javlja kao granična u svim situacijama kada je slučajna varijabla dobivena kao zbroj velikog broja međusobno nezavisnih pribrojnika [23]. Slučajna varijabla X ima normalnu razdiobu s parametrima $a \in \mathbf{R}$ i $\sigma^2 > 0$, ako je X neprekinuta slučajna varijabla s gustoćom [23]:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-a)^2}{2\sigma^2}\right)$$

Izaberu li se parametri $a = 0$ i $\sigma = 1$, dobije se slučajna varijabla s jediničnom normalnom razdiobom [23] čija gustoća iznosi:

$$\phi(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}$$

Opća normalna razdioba može se dobiti iz jedinične linearnom transformacijom [23]:

$$Y = a + \sigma X$$

Parametar a predstavlja željenu referentnu vrijednost, a za σ se uzima proizvoljan broj malog iznosa (npr. 0.02).

Proces generiranja mjerenja ovisi o elektrani ili uređaju, kako svaki uređaj posjeduje različita, ranije navedena mjerenja. Postupak generiranja mjerenja za svaki uređaj, odnosno elektranu prikazan je slikom 3.2. U slučaju da elektrana ili uređaj tek ulaze u pogon referentna vrijednost, odnosno *setPoint* će biti minimalna moguća vrijednost radne snage, P prema učitanoj konfiguraciji. Hidroelektrana ima zadan faktor snage, $\cos\phi$ u konfiguracijskoj datoteci, a za fotonaponske (engl. *Photovoltaic*, PV) panele i baterijske spremnika, taj se iznos postavlja nasumično između 0.95 i 1.0. Preko faktora snage, može se dobiti prividna snaga, S :

$$S = \frac{P}{\cos(\phi)}$$

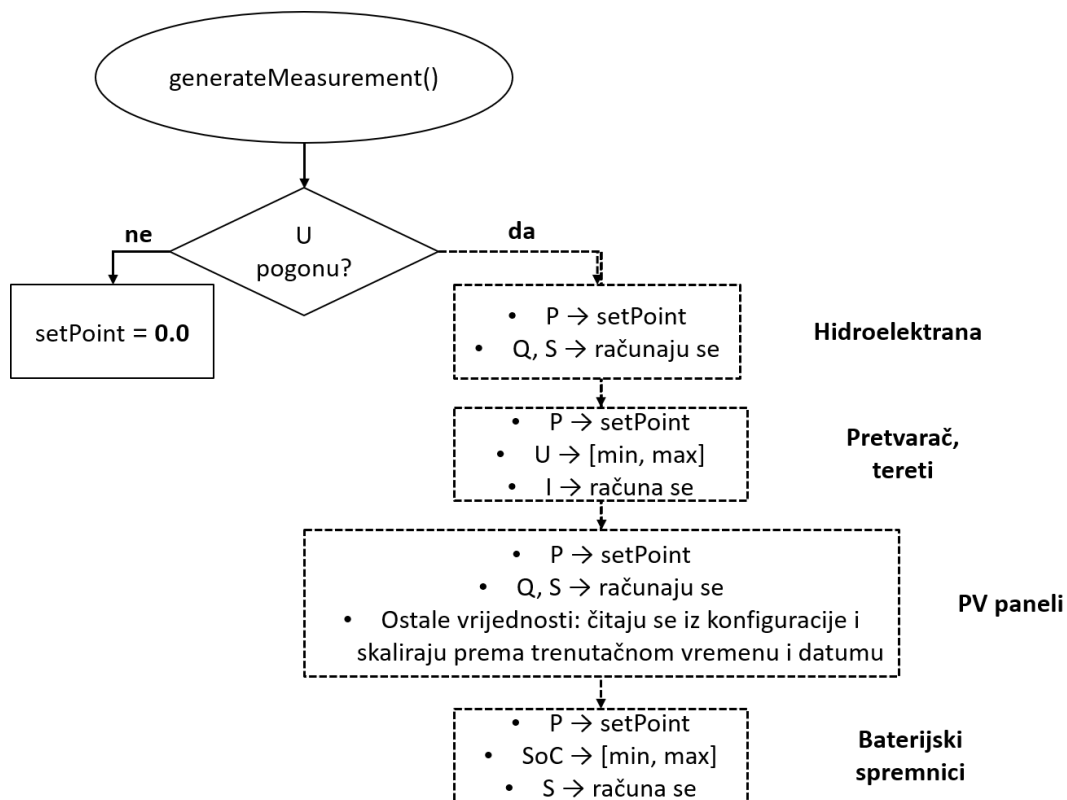
Jalova snaga, Q dobije se iz preostalih iznosa snaga, sljedećim izrazom:

$$Q = \sqrt{S^2 - P^2}$$

Za pretvarač i upravljive DC terete, iznos napona, U se postavlja na nasumičnu vrijednost unutar intervala zadanog konfiguracijom, a iznos struje, I se može dobiti kao:

$$I = \frac{P}{U}$$

Za fotonaponske panele, dohvaćaju se sat u danu i mjesec, kako bi se vrijednosti ukupnog iznosa i ukupnog dnevnog iznosa dobivene energije skalirali. Za baterijske spremnike energije, postoji još vrijednost stanja napunjenosti, (engl. *State of Charge*, SoC) koja se postavlja na nasumičan broj unutar danog raspona.



Slika 3.2: Proces generiranja mjerenja

Obrada naredbe odvija se prema imenu, pa će u nastavku biti opisan postupak za pojedinu naredbu.

Naredba *start* koristi se za pokretanje hidroelektrane - kada pristigne naredba s vrijednošću *true*:

- postavi se status u *ON* - što označava da je hidroelektrana u pogonu,
- postavi se vrijednost radne snage, P na minimalnu moguću vrijednost,
- prema prethodno opisanom postupku računaju se i postavljaju vrijednosti prividne snage, S i jalove snage, Q .

Naredba *stop* koristi se za kraj pogona hidroelektrane - kada pristigne naredba s vrijednošću *true*:

- postavi se status u *TURNING_OFF* - što označava da hidroelektrana izlazi iz pogona,
- postavi se vrijednost radne snage, P na 0.0,
- prema prethodno opisanom postupku računaju se i postavljaju vrijednosti prividne snage, S i jalove snage, Q .
- kada sve vrijednosti padnu na 0.0, postavi se status u *OFF* - označava da je hidroelektrana izvan pogona.

Naredba *activate* služi za pokretanje pretvarača, upravljivih DC tereta i baterijskih spremnika. To se postiže slanjem vrijednosti *true* - uređaj je u pogonu, a u slučaju da se pošalje vrijednost *false* - uređaj koji je prethodno bio u pogonu, izlazi iz pogona na način da mu se referentne vrijednosti postave u 0.0 i čeka se dok ne padnu postupno, u skladu sa svojim pogonskim parametrima na postavljenu vrijednost 0.0. U slučaju pretvarača i upravljivih DC tereta:

- postavi se status u *ON* - označava da je uređaj u pogonu,
- postavi se vrijednost radne snage, P na minimalnu moguću vrijednost,
- postavi se vrijednost napona, U na nasumičnu vrijednost unutar dozvoljenog raspona,
- prema prethodno opisanom postupku računa se i postavlja vrijednost struje, I .

U slučaju da se radi o baterijskim spremnicima:

- postavi se status u *ON* - označava da je uređaj u pogonu,
- postavi se vrijednost radne snage, P na minimalnu moguću vrijednost,
- postavi se vrijednost iznosa napunjenosti, SoC na nasumičnu vrijednost unutar dozvoljenog raspona,
- prema prethodno opisanom postupku računa se i postavlja vrijednost prividne snage, S .

Naredba *discharge* služi za pokretanje procesa pražnjenja baterijskih spremnika energije i to na sljedeći način: kada pristigne vrijednost *true*, trenutačnoj vrijednosti stanja napunjenosti, SoC oduzima se postotak energije koja odlazi na pražnjenje svake sekunde, određen pomoću trenutačne vrijednosti radne snage, P .

Naredbe *pref*, *uref* i *iref* služe za postavljanje referentnih vrijednosti radne snage, napona i struje: poziva se odgovarajuća metoda za promjenu željene vrijednosti i postavlja se poslana vrijednost u skladu s pogonskim parametrima elektrane (npr. vrijednost radne snage ne mijenja se trenutno nego postupno, prema iznosu promjene snage u određenom vremenskom intervalu). Na ranije opisani način, računaju se i ostala mjerenja koja ovise o referentnim vrijednostima.

4. Rezultati

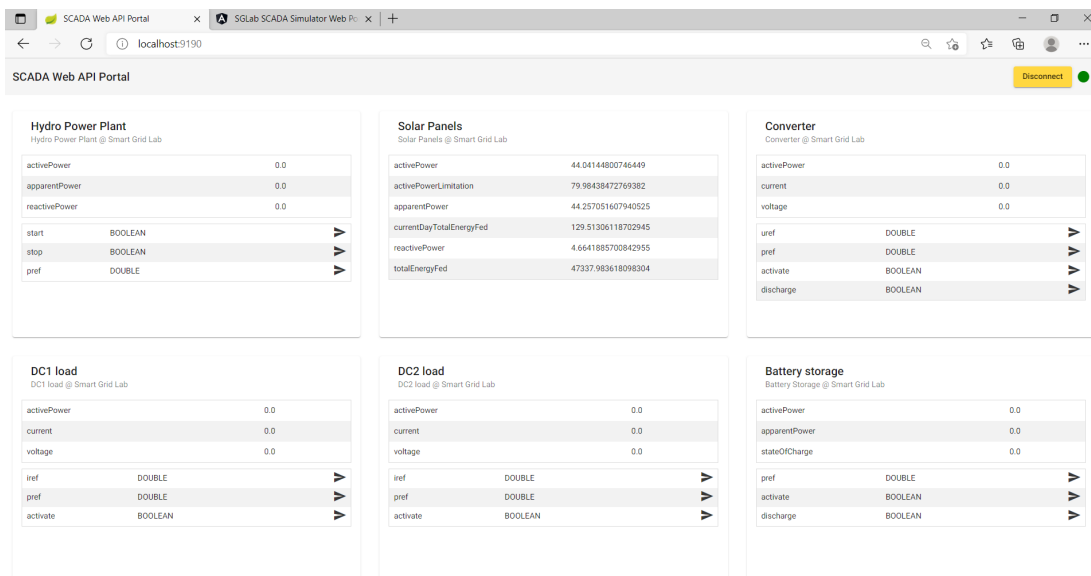
Testiranje razvijene implementacije simulatora provodi se korištenjem ranije spomenute aplikacije *SCADA Web API* koja se u stvarnom, laboratorijskom okruženju koristi za spajanje na stvarni SCADA sustav. Prikaz web sučelja aplikacije *SCADA Web API* moguće je vidjeti na slici 4.1, a aplikacije *SGLab SCADA Simulator* na slici 4.2. Tablično su prikazane komponente mikromreže, sa svim dostupnim mjerenjima koja se dinamički osvježavaju, a moguće je i poslati naredbe, odnosno referentne vrijednosti - što se vidi na slici 4.1. Web sučelje aplikacije *SGLab SCADA Simulator* vidljivo na slici 4.2, daje prikaz parametara pojedine komponente mikromreže kako su navedeni u konfiguracijskoj datoteci. Na slici 4.3 dan je ispis iz naredbenog retka simulatora - prikazan je:

- početak i uspješni završetak učitavanja konfiguracijske datoteke,
- pokretanje *Javalin* aplikacije za web sučelje,
- početak rada poslužitelja,
- prihvatanje konekcije klijenta kao i primjeri poruka pristiglih od klijenta.

Konfiguriranje simulatora, komponenti mikromreže (elektrana i uređaja), mjerenja - uspješno je ostvareno pomoću konfiguracijske datoteke. Šumovi u sustavu i podacima, dodani su na vrijednosti mjerenja što se vidi na slici 4.4 koja prikazuje mjerenja fotonaponskih panela.

Pokazatelj uspješne implementacije industrijskog komunikacijskog protokola na TCP/IP mrežnom sloju je uspješno spajanje simulatora kao poslužitelja s aplikacijom *SCADA Web API* kao klijentom. Na slici 4.5 je prikazan ispis iz terminala aplikacije *SCADA Web API*, gdje je moguće vidjeti primljene *unsolicited* poruke od poslužitelja.

Na primjeru sučelja hidroelektrane prikazanog na slici 4.6, moguće je vidjeti da se mjerenja ne generiraju samim pokretanjem aplikacije - odnosno mjerenja se generiraju

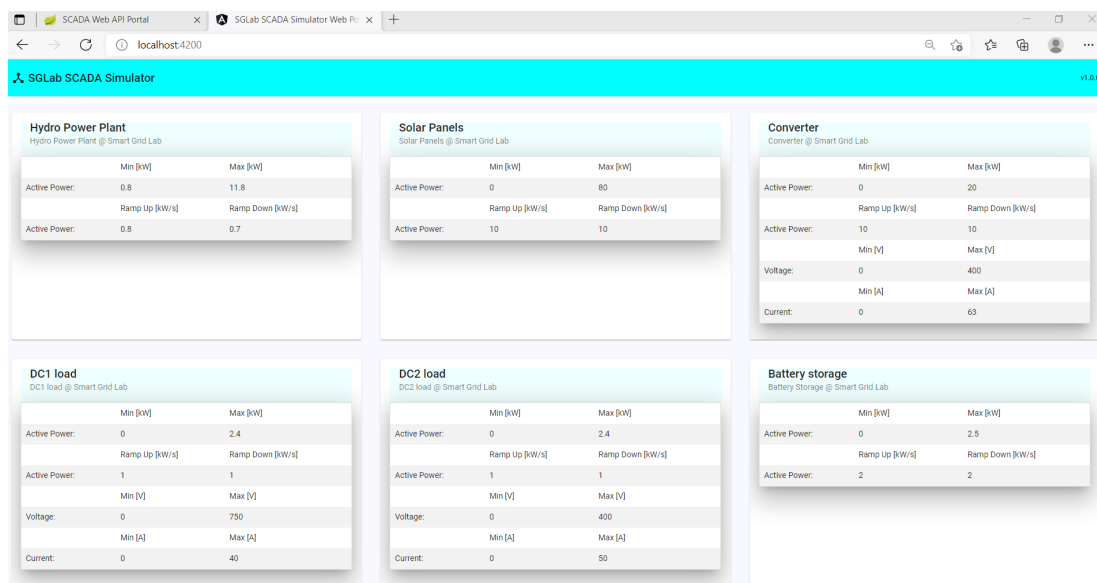


Slika 4.1: Web sučelje aplikacije *SCADA Web API*

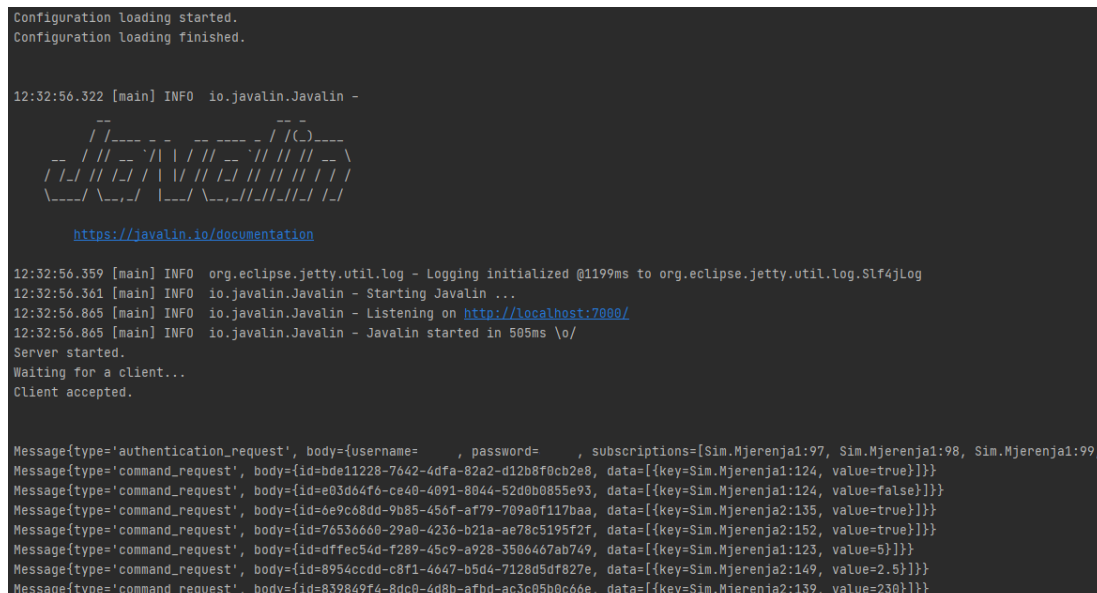
na pametan način, odgovarajući na naredbe i poslane referentne vrijednosti iz klijentske aplikacije.

Na slici 4.7, prikazana su generirana sva mjerenja što je pokazatelj uspješne implementacije komunikacijskog sučelja koje je preslika sučelja stvarnog SCADA sustava kakav se koristi u laboratorijskoj mikromreži. Moguće je vidjeti i da su sva mjerenja za koje postoje definirani rasponi konfiguracijskom datotekom, unutar zadanih raspona.

Osim opisa i slikovnog prikaza dobivenih rezultata, kao prilog ovome radu uključen je i videozapis koji prikazuje ranije navedene rezultate.



Slika 4.2: Web sučelje aplikacije *SGLab SCADA Simulator*



Slika 4.3: Ispis iz naredbenog retka, *SGLab SCADA Simulator*

Solar Panels	
Solar Panels @ Smart Grid Lab	
activePower	7.997655103945307
activePowerLimitation	80.02371899938043
apparentPower	8.292336911423886
currentDayTotalEnergyFed	183.4667610825572
reactivePower	2.1039486424760776
totalEnergyFed	47338.00007159617

Slika 4.4: Sučelje fotonaponskih panela, SCADA Web API

```

17:45:19.716 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - Read 459 bytes
17:45:19.716 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - {"type": "data_changed_undefined", "body": {"data": [{"key": "Sim.Mjerenja1:97", "value": "0.0", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.433"}, {"type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja1:98", "value": "0.0", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja1:99", "value": "0.0", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}]}}
17:45:19.761 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - Read 584 bytes
17:45:19.799 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - {"type": "data_changed_undefined", "body": {"data": [{"key": "Sim.Mjerenja3:158", "value": "7.961866319640913", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja3:165", "value": "2.0667597346872695", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja3:169", "value": "8.249329838885537", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}]}}
17:45:19.825 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - Read 462 bytes
17:45:19.825 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - {"type": "data_changed_undefined", "body": {"data": [{"key": "Sim.Mjerenja4:176", "value": "0.0", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.434", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja4:177", "value": "0.0", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja4:178", "value": "0.0", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}]}}
17:45:19.827 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - Read 583 bytes
17:45:19.827 [main] DEBUG h.f.z.s.a.s.PNETGatewayHandler - {"type": "data_changed_undefined", "body": {"data": [{"key": "Sim.Mjerenja3:155", "value": "88.00169326944216", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja3:156", "value": "47337.90294848919", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.469", "type": "EVENT", "source": "APPLICATION"}, {"key": "Sim.Mjerenja3:157", "value": "183.4284892390719", "quality": "GOOD", "timestamp": "2021-06-25 17:45:19.47", "type": "EVENT", "source": "APPLICATION"}]}}

```

Slika 4.5: Ispis iz terminala, SCADA Web API

Hydro Power Plant	
Hydro Power Plant @ Smart Grid Lab	
activePower	0.0
apparentPower	0.0
reactivePower	0.0
start	BOOLEAN
stop	BOOLEAN
pref	DOUBLE

Slika 4.6: Sučelje hidroelektrane, SCADA Web API

SCADA Web API Portal	
Disconnect	
Hydro Power Plant Hydro Power Plant @ Smart Grid Lab	
activePower	7.306209929475745
apparentPower	14.590577222711607
reactivePower	12.636503293351833
start	BOOLEAN
stop	BOOLEAN
pref	DOUBLE
Solar Panels Solar Panels @ Smart Grid Lab	
activePower	2.949627755310256
activePowerLimitation	79.98297833105849
apparentPower	3.144803027307305
currentDayTotalEnergyFed	194.2516733142827
reactivePower	0.9446841709995228
totalEnergyFed	47337.99914721962
Converter Converter @ Smart Grid Lab	
activePower	2.50250094022366
current	15.478386561782976
voltage	161.61758718506331
uref	DOUBLE
pref	DOUBLE
activate	BOOLEAN
discharge	BOOLEAN
DC1 load DC1 load @ Smart Grid Lab	
activePower	2.37401482639959
current	4.146457738402988
voltage	577.5997297144387
iref	DOUBLE
pref	DOUBLE
activate	BOOLEAN
DC2 load DC2 load @ Smart Grid Lab	
activePower	2.4194129426030115
current	17.220885254438947
voltage	138.90703762885002
iref	DOUBLE
pref	DOUBLE
activate	BOOLEAN
Battery storage Battery Storage @ Smart Grid Lab	
activePower	2.5012198059462434
apparentPower	2.5394513997701176
stateOfCharge	61.46699178594256
pref	DOUBLE
activate	BOOLEAN
discharge	BOOLEAN

Slika 4.7: Generirana mjerenja, SCADA Web API

5. Rasprava

Uspješnim završetkom implementacije i dobivenim rezultatima koji su prikazani u prethodnom poglavlju, pokazano je da se rad stvarnog SCADA sustava kakav se koristi u laboratorijskoj mikromreži opisanoj u poglavlju 1, može simulirati. Pokazano je i da se jednostavnim modelom, uz mali broj parametara određene komponente mikromreže, mogu generirati smisljena mjerenja koja mogu poslužiti u razvoju drugih klijentskih aplikacija koje se nadovezuju na rad SCADA sustava.

Razvijeni sustav je modularan što znači da se na jednostavan način može dodati komponenta mikromreže ili neki parametar, dodavanjem u konfiguracijsku datoteku s minimalnim promjenama u samom kodu. Također, ne postoji ni ograničenje na stvarni sustav - mogu se dodati elektrane ili uređaji koje nisu dio stvarnog postava mikromreže i ispitati međusobne interakcije s ostalim komponentama. Kako su i dijelovi koji se odnose na komunikacijski protokol i specifikaciju podataka modelirani zasebnim modulima, moguće je i na jednostavan način i uz minimalne promjene dodati podršku za neki drugi komunikacijski protokol.

Moguće je i dodavanje sučelja prema nekom postojećem programskom alatu (poput *Matlaba* i *Simulinka*) - na taj način moglo bi se omogućiti i dinamičko modeliranje komponenta mikromreže, ili korištenje nekih gotovih modela uz parametre zadane unutar simulatora. Taj korak bi omogućio cjelovit pristup modeliranju i simuliranju ostalih razina upravljanja osim lokalnog upravljanja (primarne razine) pojedinim komponentama mikromreže.

Kako bi se što preciznije generirala mjerenja, uz korištenje povijesnih podataka iz stvarnog SCADA sustava potrebno je identificirati koji su parametri ključni i koji su rasponi vrijednosti koje poprimaju. Na taj način moglo bi se i preciznije definirati šumove i greške prisutne u podacima i u radu stvarnog SCADA sustava. Za sada se šum modelira na jednostavan način, jednako za sve komponente mikromreže. Kako

šum obuhvaća nepreciznosti mnogih dijelova sustava poput senzora, mjernih pretvornika, turbinskog regulatora, komunikacijskog kanala - potrebno je identificirati koliko odstupaju podaci od zadanih referentnih vrijednosti i modelirati to na odgovarajući način.

U budućnosti se planira proširenje sustava i dodatnim električnim parametrima mreže, kako bi se mogli simulirati tokovi snaga i slične dodatne funkcije SCADA sustava. Planira se i proširenje postojećeg grafičkog sučelja kako bi se dobila potpuna web arhitektura SCADA sustava uz intuitivnije korisničko sučelje.

6. Zaključak

U okviru ovoga rada razvijena je implementacija simulatora SCADA sustava kakav se koristi za nadzor, upravljanje i dohvaćanje podataka u laboratorijskoj mikromreži Laboratorija za napredne elektroenergetske mreže.

Analizom područja, utvrđeno je da simulacije u SG okruženju zahtijevaju cjelovit pristup kako se radi o fenomenima iz domena elektroenergetskih sustava i informacijsko-komunikacijskih tehnologija, što povećava kompleksnost i nameće dodatne zahtjeve u simulacijama međusobne interakcije.

Uspješnom implementacijom komunikacijski protokol omogućena je komunikacija s klijentskim aplikacijama koje se razvijaju i koriste u istraživačkom i profesionalnom radu SGLab Laboratorija. Simulator je korištenjem konfiguracijske datoteke parametriran vrijednostima koje odgovaraju rasponu parametara modeliranih komponenata (elektrana i uređaja) stvarne mikromreže. Na pametan način ostvareno je generiranje mjerenja: generiraju se smislene vrijednosti i to na način da odgovaraju na upravljačke signale - naredbe i poslane referentne vrijednosti. Vrijednosti su unutar zadanih raspona.

Rezultati prikazani u ovom radu potvrđuju ispravnu funkcionalnost razvijene aplikacije, što je vidljivo i u replici komunikacijskih sučelja stvarnog sustava i generiranim mjerenjima. Implementacija je modularna, omogućena je nadogradnja sustava na jednostavan način, a opisane su i dodatne mogućnosti razvoja i budućeg rada.

ZAHVALA

Zahvaljujem izv. prof. dr. sc. Tomislavu Capuderu i Alenu Hrgi, mag. ing., na omogućenoj prilici za rad na ovom projektu, prenesenim znanjima, savjetima i usmjeravanju.

LITERATURA

- [1] D. E. Olivares, A. Mehrizi-Sani, A. H. Etemadi, C. A. Cañizares, R. Iravani, M. Kazerani, A. H. Hajimiragha, O. Gomis-Bellmunt, M. Saeedifard, R. Palma-Behnke, G. A. Jiménez-Estévez, and N. D. Hatziargyriou, “Trends in Microgrid Control,” *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1905–1919, Jul. 2014, conference Name: IEEE Transactions on Smart Grid.
- [2] M. Ahmed, L. Meegahapola, A. Vahidnia, and M. Datta, “Stability and Control Aspects of Microgrid Architectures—A Comprehensive Review,” *IEEE Access*, vol. 8, pp. 144 730–144 766, 2020, conference Name: IEEE Access.
- [3] H. Palahalli, E. Ragaini, and G. Gruosso, “Smart Grid Simulation Including Communication Network: A Hardware in the Loop Approach,” *IEEE Access*, vol. 7, pp. 90 171–90 179, 2019, conference Name: IEEE Access.
- [4] F. Li, W. Qiao, H. Sun, H. Wan, J. Wang, Y. Xia, Z. Xu, and P. Zhang, “Smart Transmission Grid: Vision and Framework,” *IEEE Transactions on Smart Grid*, vol. 1, no. 2, pp. 168–177, Sep. 2010, conference Name: IEEE Transactions on Smart Grid.
- [5] P. Arboleya, C. Gonzalez-Moran, M. Coto, M. C. Falvo, L. Martirano, D. Sbordone, I. Bertini, and B. D. Pietra, “Efficient Energy Management in Smart Micro-Grids: ZERO Grid Impact Buildings,” *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 1055–1063, Mar. 2015, conference Name: IEEE Transactions on Smart Grid.
- [6] P. I. Radoglou-Grammatikis and P. G. Sarigiannidis, “Securing the Smart Grid: A Comprehensive Compilation of Intrusion Detection and Prevention Systems,” *IEEE Access*, vol. 7, pp. 46 595–46 620, 2019, conference Name: IEEE Access.
- [7] “The Modbus Organization.” [Online]. Available: <https://modbus.org/>

- [8] “Overview of DNP3 Protocol.” [Online]. Available: <https://www.dnp.org/About/Overview-of-DNP3-Protocol>
- [9] P. Brooks, “Ethernet/IP-industrial protocol,” in *ETFA 2001. 8th International Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No.01TH8597)*, vol. 2. Antibes-Juan les Pins, France: IEEE, 2001, pp. 505–514. [Online]. Available: <http://ieeexplore.ieee.org/document/997725/>
- [10] “IEC 60870-5-104:2006+AMD1:2016 CSV | IEC Webstore.” [Online]. Available: <https://webstore.iec.ch/publication/25035>
- [11] S. Ghosh and S. Sampalli, “A Survey of Security in SCADA Networks: Current Issues and Future Challenges,” *IEEE Access*, vol. 7, pp. 135 812–135 831, 2019, conference Name: IEEE Access.
- [12] I. T. F. o. I. T. f. S. Tools, S. C. Müller, H. Georg, J. J. Nutaro, E. Widl, Y. Deng, P. Palensky, M. U. Awais, M. Chenine, M. Küch, M. Stifter, H. Lin, S. K. Shukla, C. Wietfeld, C. Rehtanz, C. Dufour, X. Wang, V. Dinavahi, M. O. Faruque, W. Meng, S. Liu, A. Monti, M. Ni, A. Davoudi, and A. Mehrizi-Sani, “Interfacing Power System and ICT Simulators: Challenges, State-of-the-Art, and Case Studies,” *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 14–24, Jan. 2018, conference Name: IEEE Transactions on Smart Grid.
- [13] Y. Huo and G. Grusso, “Hardware-in-the-Loop Framework for Validation of Ancillary Service in Microgrids: Feasibility, Problems and Improvement,” *IEEE Access*, vol. 7, pp. 58 104–58 112, 2019, conference Name: IEEE Access.
- [14] P. C. Kotsampopoulos, V. A. Kleftakis, and N. D. Hatziargyriou, “Laboratory Education of Modern Power Systems Using PHIL Simulation,” *IEEE Transactions on Power Systems*, vol. 32, no. 5, pp. 3992–4001, Sep. 2017, conference Name: IEEE Transactions on Power Systems.
- [15] C. Queiroz, A. Mahmood, and Z. Tari, “SCADASim—A Framework for Building SCADA Simulations,” *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 589–597, Dec. 2011, conference Name: IEEE Transactions on Smart Grid.
- [16] S. Carullo and C. Nwankpa, “Interconnected power systems laboratory: a computer automated instructional facility for power system experiments,” *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 215–222, May 2002, conference Name: IEEE Transactions on Power Systems.

- [17] “Laboratory - Zavod za visoki napon i energetiku.” [Online]. Available: https://www.fer.unizg.hr/zvne_old/research/research_labs/sglab/laboratory
- [18] “Home - ZVNE.” [Online]. Available: <https://www.fer.unizg.hr/zvne>
- [19] “Programske platforme i projekti.” [Online]. Available: <https://www.koncar-ket.hr/upravljacki-centri/it-rjesenja/>
- [20] “IEC 61850:2021 SER | IEC Webstore | LVDC.” [Online]. Available: <https://webstore.iec.ch/publication/6028>
- [21] “Unified Architecture.” [Online]. Available: <https://opcfoundation.org/about/opc-technologies/opc-ua/>
- [22] M. Beus, F. Banis, H. Pandžić, and N. K. Poulsen, “Three-level hierarchical microgrid control—model development and laboratory implementation,” *Electric Power Systems Research*, vol. 189, p. 106758, Dec. 2020. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0378779620305617>
- [23] N. Elezovic, *Vjerojatnost i statistika. 2. Slučajne varijable*. Zagreb: Element, 2010.

Popis slika

1.1. Konfiguracija mikromreže, prema [1]	3
1.2. Shematski prikaz laboratorijske mikromreže, SGLab - prema [22] . . .	10
2.1. Arhitektura sustava: simulator ilustriran narančastim isprekidanim pravokutnikom	12
3.1. Arhitektura aplikacije <i>SGLab SCADA simulator</i>	14
3.2. Proces generiranja mjerenja	22
4.1. Web sučelje aplikacije <i>SCADA Web API</i>	26
4.2. Web sučelje aplikacije <i>SGLab SCADA Simulator</i>	27
4.3. Ispis iz naredbenog retka, <i>SGLab SCADA Simulator</i>	27
4.4. Sučelje fotonaponskih panela, <i>SCADA Web API</i>	28
4.5. Ispis iz terminala, <i>SCADA Web API</i>	28
4.6. Sučelje hidroelektrane, <i>SCADA Web API</i>	29
4.7. Generirana mjerenja, <i>SCADA Web API</i>	29

Popis tablica

3.1. Parametri komponenata mikromreže	16
3.2. Mjerenja i naredbe komponenata mikromreže	17

SAŽETAK

Naslov: *Razvoj i implementacija simulatora SCADA sustava laboratorijske mikromreže*

Autor: *Lucija Matulin*

Cilj je ovoga rada razviti i implementirati novi simulacijski alat koji oponaša SCADA sustav laboratorijske mikromreže SGLab. Razvoj osim konfiguracije simulatora, strojeva, mjerenja, šumova, uključuje i implementaciju komunikacijskih sučelja, industrijskog komunikacijskog protokola, parametrisanje prema danoj konfiguraciji i modifikaciju ponašanja ovisno o zadanim referentnim vrijednostima i naredbama uz generiranje mjerenja unutar zadanih raspona i na pametan način - na određeni upravljački signal. U sklopu rada, dan je teorijski uvod u pojmove i koncepte - mikromreže, napredne elektroenergetske mreže, SCADA sustave - odnosno svojevrsno stanje područja i pregled problematike simulacija u spomenutom okruženju. Opisano je laboratorijsko okruženje SGLab laboratorija - mikromreža i njezine komponente integrirane u SCADA sustav - koje su predmet modeliranja i simuliranja ovoga rada. Implementacija SCADA simulatora ostvarena je u programskom jeziku Java, a kao dodatak radu dostupan je potpuni izvorni kod razvijenog rješenja. Detaljno je opisana arhitektura razvijenog sustava, način modeliranja i parametrisanja komponenata mikromreže i generiranja mjerenja. Rezultati testiranja provedenog pomoću postojeće aplikacije SCADA Web API koja se u laboratorijskom okruženju koristi za komunikaciju sa stvarnim SCADA sustavom, pokazuje uspješno parametrisanje, prihvatanje konekcije i uspostavljanje komunikacije, uz generiranje smislenih mjerenja i odgovora na poslana naredbe i referentne vrijednosti. Navedeni su i diskutirani rezultati, a dan je i prijedlog za budući rad koji se sastoji od mogućnosti nadogradnje razvijenog rješenja.

Ključne riječi: mikromreža, napredna elektroenergetska mreža, simulator, SCADA sustav

ABSTRACT

Title: *Development and implementation of SCADA system simulator for microgrid laboratory*

Author: *Lucija Matulin*

The main objective of this paper is the new simulation tool, developed and utilized to simulate the SCADA system used in the SGLab microgrid laboratory. Besides configuring the simulator, machines, measurements, noises - development includes communication interfaces and communication protocol implementation, simulator parametrization in regards to the given configuration, system behavior depending on given reference points and commands, along with smart measurement generating within given parameter ranges and responding to control inputs. Within the scope of this paper, a theoretical introduction to definitions and concepts is presented - including microgrid, smart grid, SCADA system - serving as state of the art in the field of simulations in the smart grid environment. The SGLab laboratory is described - including the microgrid and its components integrated into the SCADA system - the subject of modeling and simulation in this paper. SCADA simulator implementation is written using the programming language Java, the source code of the developed project is available in addition to this paper. System's architecture detailed description, microgrid's components modeling and parameterization, and generating measurement process are presented. Testing conducted with the existing application SCADA Web API used in the laboratory for communication with the SCADA system shows successful parametrization and communication, measurement generating, and meaningful response to commands and reference points. The results are shown and discussed. The future work guidelines consisted of possible system extensions and upgrades are given.

Keywords: microgrid, SCADA system, simulator, smart grid