

# Optimiranje proizvodnje upravljivog distribuiranog izvora u distributivnoj mreži s neupravljivim distribuiranim izvorima

---

Šostarko, Goran

Master's thesis / Diplomski rad

2018

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:200:998442>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-07-11**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I**

**INFORMACIJSKIH TEHNOLOGIJA OSIJEK**

**Sveučilišni diplomski studij**

**OPTIMIRANJE PROIZVODNJE UPRAVLJIVOG  
DISTRIBUIRANOG IZVORA U DISTRIBUTIVNOJ  
MREŽI S NEUPRAVLJIVIM DISTRIBUIRANIM  
IZVORIMA**

**Diplomski rad**

**Goran Šostarko**

**Osijek, 2018.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za obranu diplomskog rada

Osijek, 17.09.2018.

Odboru za završne i diplomske ispite

**Imenovanje Povjerenstva za obranu diplomskog rada**

<b>Ime i prezime studenta:</b>	Goran Šostarko
<b>Studij, smjer:</b>	Diplomski sveučilišni studij Elektrotehnika
<b>Mat. br. studenta, godina upisa:</b>	D 1007, 20.09.2017.
<b>OIB studenta:</b>	13920193816
<b>Mentor:</b>	Izv. prof. dr. sc. Marinko Barukčić
<b>Sumentor:</b>	Dr. sc. Marko Vukobratović
<b>Sumentor iz tvrtke:</b>	Dalibor Vlajsavljević
<b>Predsjednik Povjerenstva:</b>	Doc.dr.sc. Danijel Topić
<b>Član Povjerenstva:</b>	Dr. sc. Marko Vukobratović
<b>Naslov diplomskog rada:</b>	Optimiranje proizvodnje upravljivog distribuiranog izvora u distributivnoj mreži s neupravljivim distribuiranim izvorima
<b>Znanstvena grana rada:</b>	<b>Elektroenergetika (zn. polje elektrotehnika)</b>
<b>Zadatak diplomskog rada:</b>	U radu je potrebno opisati ukratko tehnologije izvora distribuirane proizvodnje električne energije koje se koriste u distributivnim mrežama. Pri naglasiti za svaku tehnologiju da li se s obzirom na vremensko promjenjivost izvora energije radi o upravljivom ili neupravljivom distribuiranom izvoru. Na primjeru srednje naponske 20 (10) kV distributivne mreže s instaliranim upravljivim i neupravljivim izvorima potrebno je simulacijama na računalu odrediti optimalnu proizvodnju upravljivog distribuiranog izvora. Optimizaciju obaviti za zadane dnevne profile opterećenja i intenziteta energetske izvora neupravljive distribuirane proizvodnje. Kao simulacijski alat koristiti OpenDSS ili (i) DSSim računalni program, a za provedbu optimizacije koristiti gotove alate dostupne u programskim paketima u PYTHON (preporuka) ili MATLAB okruženju. (Rezervirano: Goran & Scaron;ostarko) (sumentor: Dalibor Vlajsavljević,
<b>Prijedlog ocjene pismenog dijela ispita (diplomskog rada):</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene mentora:</b>	17.09.2018.
Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 08.10.2018.

**Ime i  
prezime  
studenta:**

Goran Šostarko

**Studij:**

Diplomski sveučilišni studij Elektrotehnika

**Mat. br.  
studenta,  
godina  
upisa:**

D 1007, 20.09.2017.

**Ephorus  
podudaranje  
[%]:**

3

Ovom izjavom izjavljujem da je rad pod nazivom: **Optimiranje proizvodnje upravljivog distribuiranog izvora u distributivnoj mreži s neupravljivim distribuiranim izvorima**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Marinko Barukčić

i sumentora Dr. sc. Marko Vukobratović

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

## Sadržaj

1. UVOD .....	1
1.1. Zadatak i organizacija rada.....	2
2. DISTRIBUIRANA PROIZVODNJA ELEKTRIČNE ENERGIJE.....	4
2.1. Tradicionalni generatori s unutarnjim izgaranjem.....	5
2.1.1. Mikro turbine.....	5
2.2. Netradicionalni generatori .....	6
2.2.1. Elektrokemijski uređaji – gorivne ćelije .....	6
2.2.2. Fotonaponski sustavi .....	8
2.2.3. Vjetrogeneratori .....	8
2.3. Klasifikacije distribuiranih izvora .....	9
2.4. Prednosti distribuiranih izvora.....	11
2.5. Opći problemi u proizvodnji električne energije pomoću distribuiranih izvora .....	12
2.6. Tehnički utjecaji distribuiranih izvora na distribucijski sustav .....	15
2.7. Rješenja tehničkih utjecaja distribuiranih izvora na distribucijski sustav .....	18
3. EVOLUCIJSKI ALGORITMI I PROGRAMSKI PAKETI ZA OPTIMIZACIJU .....	20
3.1. Evolucijski algoritmi .....	20
3.1.1. Značajke evolucijskih algoritama .....	23
3.1.2. Osnovni ciklus evolucijskih algoritama.....	24
3.1.3. Standardni diferencijalni evolucijski algoritam .....	26
3.2. Programski paket za analizu distribucijske mreže – OpenDSS.....	27
3.2.1. Elementi za distribuciju i konverziju energije u OpenDSS-u .....	28
3.2.1. OpenDSS sintaksa .....	32
3.2.2. Definiranje objekata i njihovih parametara .....	32
3.3. Programski jezik Python.....	35
3.3.1. Diferencijalni evolucijski algoritam u programskom jeziku Python .....	35
4. MATEMATIČKI ZAPIS PROBLEMA I RJEŠENJE OPTIMIZACIJE DISTRIBUTIVNE MREŽE .....	38
4.1. Matematički zapis problema.....	38
4.2. Definiranje distribuiranih izvora u OpenDSS-u .....	39
4.3. Rezultati simulacije mreže i analiza rezultata .....	43
5. ZAKLJUČAK .....	54
LITERATURA.....	55
POPIS SLIKA .....	58

POPIS TABLICA.....	60
SAŽETAK.....	61
ABSTRACT .....	62
ŽIVOTOPIS .....	63
PRILOZI.....	64
PRILOG I – Popis trošila, lokacija, spoj, naponi i snage .....	64
PRILOG II – Python kod za simuliranje sustava i optimizaciju.....	67

## 1. UVOD

Većina suvremenih elektroenergetskih sustava izgrađena je u zadnjih 50-ak godina. Glavna svrha tih sustava bila je postojanje centraliziranih generatora koji šalju električnu energiju u prijenosni i distribucijski sustav do krajnjeg korisnika. U posljednje vrijeme pojavilo se veliko zanimanje za priključenjem proizvodnih objekata na distribucijsku mrežu što je poznato pod nazivom distribuirana proizvodnja električne energije (engl. distributed generation).

Konvencionalni elektroenergetski sustavi imaju veliki broj prednosti kao što su: veće i učinkovitije proizvodne jedinice, relativno manji broj pogonskog osoblja te omogućen ulazak u pogon u najučinkovitijem proizvodnom objektu u svakom trenutku. Distribucijske mreže projektirane su i dimenzionirane za tok snage u jednom smjeru. U posljednje vrijeme javlja se interes za distribuiranu proizvodnju.

Utjecaj na okoliš jedan je od značajnijih čimbenika prilikom razmatranja potencijalnog priključivanja novog proizvodnog objekta na mrežu. Kako na temelju Kyoto protokola, mnoge zemlje imaju obvezu smanjiti emisiju CO<sub>2</sub> kako bi se smanjio utjecaj na klimatske promjene gdje do izražaja dolaze prednosti obnovljivih izvora energije. Razvijaju se razni programi iskorištavanja obnovljivih izvora koji uključuju vjetroelektrane, male hidroelektrane, fotonaponske izvore, zemni plin te energiju iz otpada te iz biomase. Obnovljivi izvori imaju znatno manju energetska vrijednost u usporedbi s fosilnim gorivima zbog čega su takve elektrane manje veličine i geografski su široko raspoređene. Distribuirani izvori, koji kao pogonsko gorivo koriste energiju sunčevog zračenja ili energiju vjetra, zbog vremenskih promjena i drugih faktora imaju vrlo male mogućnosti upravljanja količinom proizvedene energije. S druge strane, distribuirana proizvodnja koja za pogon koristi biopljin ili biomasu zbog konstantnog intenziteta pogonskog goriva ima veliku mogućnost upravljanja proizvodnjom i djelatne i jalove snage.

Distribuirani izvori u velikom broju slučajeva su u privatnom vlasništvu, a njihovi vlasnici imaju u interesu što veću proizvodnju električne energije zbog financijskih benefita investitora i vlasnika. S druge strane, performanse distributera mogu biti smanjenje zbog prevelike količine energije (povećani gubici, prekoračenja naponskih granica te druge strujno-naponske i energetske prilike u mreži). Iz tog razloga je nastala optimizacija dnevne proizvodnje električne energije u distribuiranim izvorima zbog promjena u opterećenju mreže i cijeni električne energije.

Ovim radom analiziran je optimalni pogon upravljivog distribuiranog izvora u paralelnom radu s neupravljivim distribuiranim izvorima, uz promjenjivi profil opterećenja i promjenjive izvore neupravljivih distribuiranih izvora spojenih na distribucijsku mrežu.

Prema izvještaju IRENA-e [20] iz priloženih podataka može se vidjeti kako se kroz godine broj obnovljivih izvora povećava u većini zemalja, pa tako i u Hrvatskoj. Iz tog razloga je ovima radom opisan utjecaj obnovljivih izvora na postojeći elektroenergetski sustav te je prikazan način na koji je moguće optimizirati proizvodnju rada upravljivog distribuiranog izvora.

## **1.1. Zadatak i organizacija rada**

Ovim radom se opisuju tehnologije izvora distributivne proizvodnje električne energije koje se koriste u distributivnim mrežama. Prilikom opisivanja tehnologija bilo je potrebno staviti naglasak da li se s obzirom na vremensku promjenjivost izvora energije radi o upravljivom ili neupravljivom distribuiranom izvoru za svaku tehnologiju. Na primjeru srednje naponske distributivne mreže s instaliranim upravljivim ili neupravljivim izvorima potrebno je računalnim simulacijama odrediti optimalnu proizvodnju upravljivog distribuiranog izvora. Optimizaciju je bilo potrebno obaviti za zadane dnevne profile opterećenja i intenziteta energetskih izvora neupravljive distribuirane proizvodnje. Kao simulacijski alat korišten je OpenDSS računalni program, a za provedbu optimizacije korišteni su gotovi alati koji su dostupni u Python okruženju.

Prvi korak u izradi ovog rada bio je upoznavanje s računalnim programom OpenDSS te upoznavanje s programskim okruženjem Python te savladavanje njihovih osnova. Nakon svladavanja osnova ovih računalnih alata, sintaksi njihovog programskog koda te principa rada bilo je potrebno istražiti evolucijske algoritme, što su i na koji način funkcioniraju kako bi mogli biti iskorišteni prilikom implementacije ovoga rada za optimizaciju proizvodnje. Treći korak bio je izrada programskog koda za optimizaciju zadane distributivne mreže te analiza dobivenih podataka i fizikalno tumačenje istih.

Rad je podijeljen u pet poglavlja. U prvom poglavlju dana je uvodna riječ te značajke distribuirane proizvodnje. Opisan je zadatak rada i koraci koji su poduzeti prilikom izrade rada. U drugom poglavlju opisani su distribuirani izvori, njihove podjele, karakteristike, prednosti i mane. U trećem poglavlju opisani su evolucijski algoritmi, njihov princip rada te je posebno opisan i standardni diferencijalni evolucijski algoritam korišten pri izradi ovog rada kroz Python programsko okruženje. U četvrtom poglavlju opisan je matematički zapis problema te predstavljen optimizacijski problem i koraci njegovog rješavanja kroz ovaj rad te su prikazana rješenja



optimizacije i analizirani su i fizikalno protumačeni dobiveni simulacijski rezultati. U zadnjem, petom poglavlju dan je osvrt na cijeli rad, opisani su problemi nastali prilikom izrade rada te navedeni osvrti autora na sam rad.

## 2. DISTRIBUIRANA PROIZVODNJA ELEKTRIČNE ENERGIJE

Pojam distribuirane proizvodnje odnosi se na proizvodnju električne energije u blizini potrošača. Resursi korišteni u distribuiranoj proizvodnji su obnovljiva energija i kogeneracija (istovremena proizvodnja toplinske i električne energije). Obnovljiva energija je energija iz prirodnih resursa kao što su vjetar, energija sunca, plima, valovi, geotermalna toplina, biomasa, itd.

Elektroenergetski sustavi koji su konvencionalno ustrojani imaju više prednosti. Velike konvencionalne jedinice su učinkovitije i u slučaju da su automatizirane potreba za prisutnim tehničkim osobljem je vrlo mala. Visokonaponske mreže kojima je prijenosni sustav povezan, omogućuju manje zahtjeve za snagom pričuve generatora. Također se omogućuje priključenje najučinkovitije elektrane na sustav u bilo kojem trenutku. Velike snage moguće je prenositi na velike udaljenosti putem prijenosne mreže i to uz male gubitke. U tom slučaju moguće je projektirati distributivne mreže za jednosmjerne tokove snaga te ih je moguće projektirati samo za potrebe potrošača. S druge strane, u zadnjih nekoliko godina pojavilo se sve veće zanimanje za distribuiranu proizvodnju električne energije. Zanimanje je povećano zbog smanjenja emisija CO<sub>2</sub> prilikom distribuirane proizvodnje, veće energetske učinkovitosti, povećanja se racionalnog korištenja električne energije, interesa za samoodrživost nacionalnih energetske sustava, itd.

Prema [4], jedan od najznačajnijih faktora u razmatranju priključivanja novih proizvodnih jedinica na postojeću mrežu je utjecaj na okoliš. Uz brigu o emisijama štetnih plinova iz konvencionalnih elektrana (elektrane na fosilna goriva kao što su ugljen, nafta, zemni plin, itd.), obnovljivi izvori imaju svoje prednosti. Kako je to propisano Kyoto protokolom, mnoge zemlje imaju obvezu smanjiti emisije CO<sub>2</sub> zbog smanjenja utjecaja na klimatske promjene. Sve se više potiče distribuirana proizvodnja te nastaju razni programi iskorištavanja obnovljivih izvora energije kao što su vjetrogeneratori, hidroelektrane malih snaga, fotonaponski izvori, zemni plin, energija iz otpada i iz biomase. Obnovljive izvore moguće je koristiti prilikom kogeneracije gdje se otpadna toplina koristi za industrijske procese ili za grijanje te se na taj način povećava ukupna energetska učinkovitost. Kako je energetska vrijednost obnovljivih izvora puno manja elektrane koje rade na tom principu su puno manjih veličina i geografski su raspodijeljene što nije slučaj kod konvencionalnih elektrana.

Implementacija distribuiranih izvora u distributivnu mrežu, prema [15], utječe na napone u čvorovima mreže, gubitke snage, mrežni kapacitet, cijenu energije i druge čimbenike. Jačina

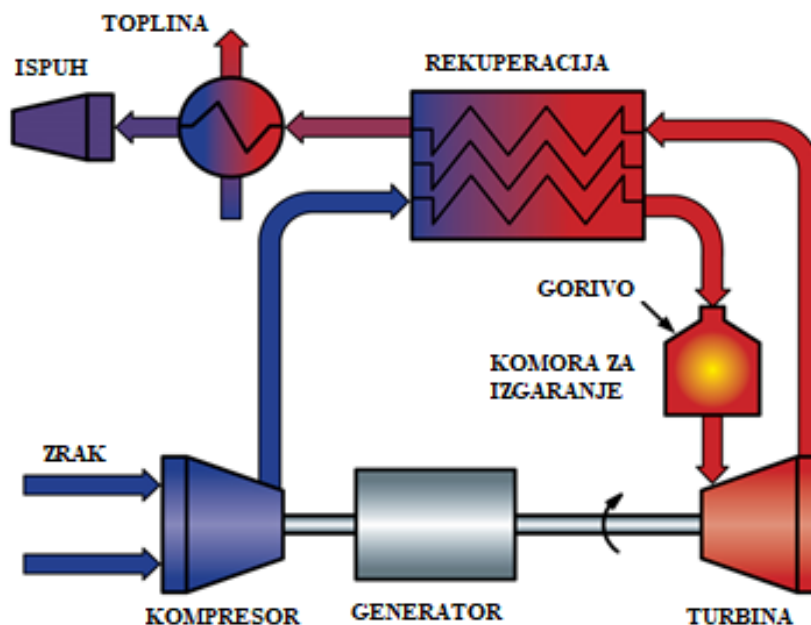
utjecaja distribuiranih izvora na mrežu ovisi o konfiguraciji mreže, parametrima mreže te parametrima tereta. Ovisno o vrsti potrošača koji su priključeni na mrežu, dolazi do promjena opterećenja distribucijskih mreža tijekom dana. Budući da opterećenja mijenjaju količinu proizvedene električne energije, distribuirani izvori neće imati isti učinak na mrežu tijekom dana. Razvoj tržišta električne energije uključuje i promjene cijene električne energije tijekom dana.

Distribuirani izvori često su u vlasništvu i financirani od strane privatnog sektora. Vlasnici distribuiranih izvora imaju interes za proizvodnjom što veće količine energije. S druge strane, proizvodnja prevelike količine električne energije u distribuiranim izvorima, umjesto povećanja, može izazvati smanjenje performansi distributera električne energije (povećani gubici, prekoračenje naponskih granica i sl.). Iz tog razloga razvila se potreba za optimizacijom dnevne proizvodnje električne energije u distribuiranim izvorima jer postoje promjene u opterećenju mreže i cijeni električne energije. Postoje različiti tipovi distribuirane proizvodnje s konstrukcijskih i tehnoloških stajališta. Kako bi odabir bio najprikladniji za određenu situaciju, potrebno je usporediti ove tipove.

## **2.1. Tradicionalni generatori s unutarnjim izgaranjem**

### **2.1.1. Mikro turbine**

Za tehnologiju mikro turbina smatra se da ima perspektivnu budućnost. To su turbine s unutarnjim izgaranjem malog kapaciteta čije pogonsko gorivo može biti benzin, propan te lož ulje. Jedan od oblika obnovljivih izvora energije i ekološki prikladniji od benzina i nafte je bio-plin. Prema pojednostavljenoj shemi (slika 2.1.) sastoje se od kompresora, komore izgaranja, sustava rekuperacije, male turbine i generatora. Mikro turbine se proizvode u rasponu snage od 20 do 500 kW te rasponu volumena od 0.4 do 1 m<sup>3</sup> te rade pri nižim temperaturama i tlakovima ali pri većim brzinama (do 100 000 o/min) za razliku od normalnih turbina sa unutarnjim izgaranjem. Neki postojeći primjerci mikro turbina imaju dobru pouzdanost, postižu velike brzine te imaju nisku cijenu. Mikro turbine su upravljivi distribuirani izvori.



Slika 2.1. Princip rada mikro turbine. [11]

Prednosti mikro turbine:

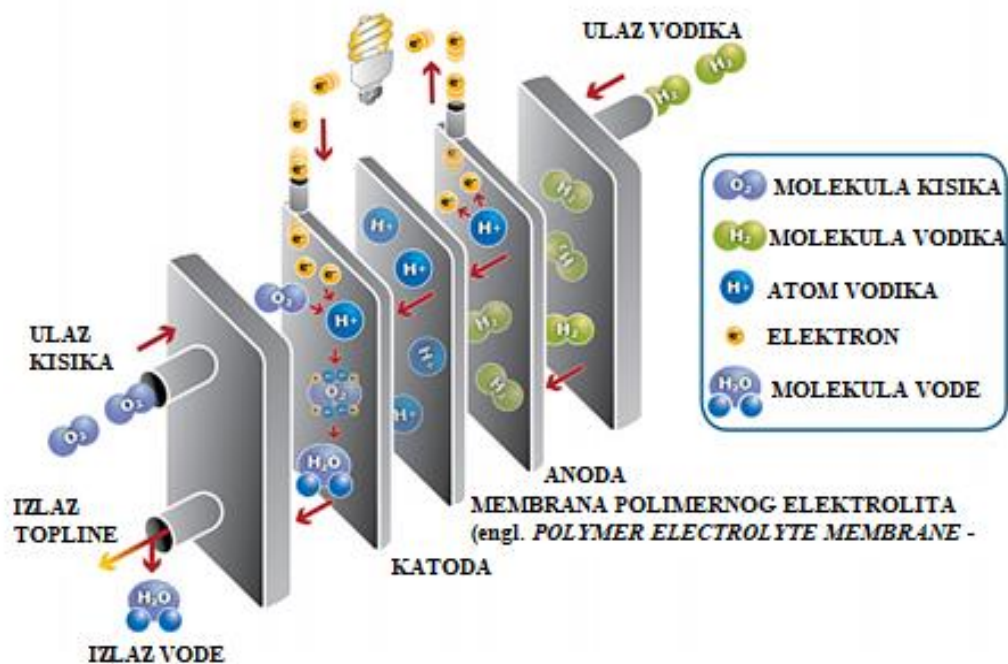
- Mogućnost instalacije na licu mjesta, pogotovo ako postoje prostorna ograničenja. Male su veličine i težine u odnosu na tradicionalne turbine.
- Iznimno visoka efikasnost rada (više od 80%) te znatno smanjene emisije štetnih plinova (10ppm NO<sub>x</sub>) u odnosu na velike turbine.
- Tehnologija prisutna već duži niz godina, brzo pokretanje sustava te lakše održavanje zbog jednostavnog dizajna.
- Povećanje fleksibilnosti efikasnijeg upravljanja pomoću sklopova energetske elektronike.

## 2.2. Netradicionalni generatori

### 2.2.1. Elektrokemijski uređaji – gorivne ćelije

Gorivna ćelija je uređaj koji se koristi za proizvodnju električne i toplinske energije iz kemijske energije putem elektrokemijskih procesa. Gorivnu ćeliju moguće je pojednostavljeno prikazati (slika 2.2.) kao bateriju kojoj se konstantno dovodi gorivo. Ali za razliku od baterije, nije ju potrebno puniti. Tehnologija gorivnih ćelija je poznata još od 1960-ih godina. Gorivne ćelije se proizvode za snage u kW (za prenosive jedinice) do MW (za stacionarne jedinice). Vodik koji se koristi kao gorivo se dobiva iz ugljikovodika ili putem elektrolize iz vode. Moguće je koristiti niz

goriva koja su obogaćena vodikom kao što su prirodni plin, benzin, biopljin ili propan. Gorivne ćelije su upravljivi distribuirani izvori.



Slika 2.2. Princip rada gorivne ćelije. [12]

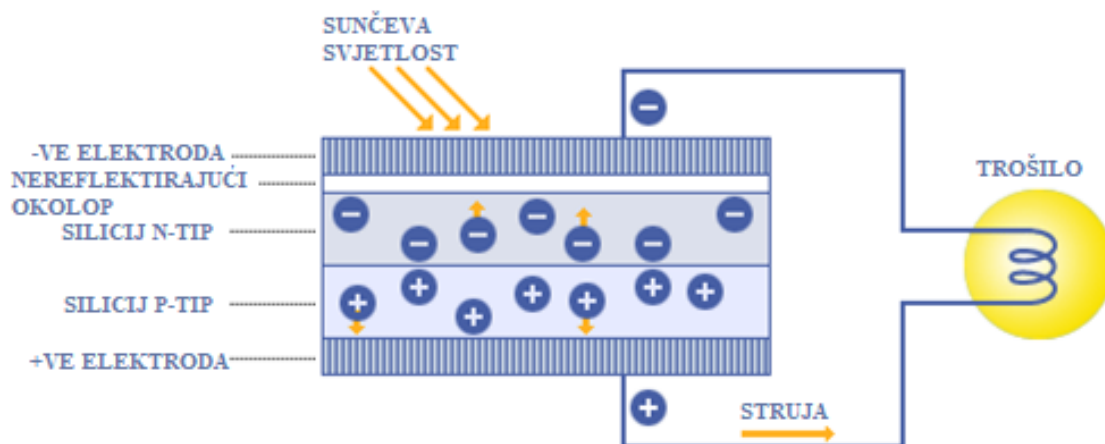
Prednosti gorivnih ćelija:

- Gorivne ćelije pretvaraju kemijsku energiju u električnu uz efikasnost od 60%.
- Budući da nemaju pokretnih dijelova (osim ventilatora za dovod  $O_2$  i  $H_2$  i/ili pumpi za vodu) razina buke koju proizvode je vrlo niska.
- Budući da nema izgaranja ugljika, emisije  $CO_2$  su zanemarive.
- Mogućnost spajanja više ćelija kako bi se zadovoljila proizvodnja električne energije i to uz vrlo nisku cijenu.

S vremenom dolazi do porasta unutarnje impedancije pa je potrebno ugraditi komponente energetske elektronike kako bi se regulirao izlazni napon, što je nedostatak gorivnih ćelija.

### 2.2.2. Fotonaponski sustavi

Osnovna jedinica fotonaponskog sustava (FN sustav) je ćelija (slika 2.3.) koja može biti okruglog ili kvadratnog oblika, izrađena od dopiranih kristala silicija. Ćelije se spajaju na način da se formiraju moduli ili paneli dok se paneli spajaju kako bi se formirale mreže s ciljem proizvodnje željene električne energije. Ćelije stvaraju električnu energiju na način da apsorbiraju sunčevu svjetlost. Protoni iz sunčeve svjetlosti potiču protjecanje elektrona. Ovisno o veličini, svaka ćelija daje 2 do 4A i izlazni napon od 0.5V. U nekoj praktičnoj primjeni mreže (ćelije spojene u seriju) daju izlazni napon od 12V. Fotonaponski sustavi imaju nekoliko ograničenja kao što su: mala snaga na izlazu, (u prosjeku se proizvede oko 150kW po hektaru zemlje) te ograničenja u pogledu geografskih i meteoroloških karakteristika područja na kojemu su instalirani. Fotonaponski sustavi su neupravljivi distribuirani izvori.



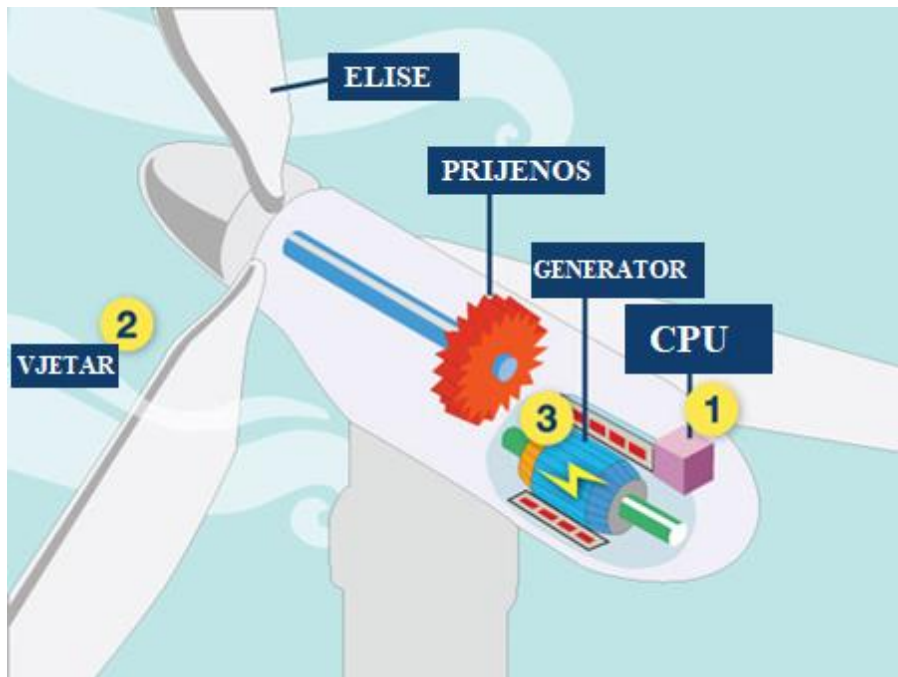
Slika 2.3. Princip rada fotonaponske ćelije. [13]

### 2.2.3. Vjetrogeneratori

Energija vjetra se koristi već dugi niz godina. Vjetrogenerator (slika 2.4.) se sastoji od rotora, lopatica, generatora, prijenosa, osovine i glave turbine koja sadrži pogon generatora i prijenos. Vjetrogeneratori mogu proizvoditi čistu električnu energiju, samostalno ili u sklopu vjetro parka. Lopatice vjetrogeneratora su dužine od 10 do 30 metara te se na rotoru nalazi dvije ili tri lopatice, ovisno o konstrukciji.

Vjetrogeneratori rade na način da vjetar udara lopatice koje se nalazi na rotoru te na taj način ostvaruje zakretanje osovine spojene na generator, koji u konačnici proizvodi električnu energiju. Veliki vjetro parkovi rade uglavnom kao centralizirani izvori električne energije te u takvim sustavima mali vjetrogeneratori (koji rade kao moduli) mogu raditi u kombinaciji s fotonaponskim

sustavima i baterijama, kako bi proizveli 25 do 100kW električne energije. Vjetrogeneratori su neupravljivi distribuirani izvori električne energije.



Slika 2.4. Princip rada vjetrogeneratora. [14]

Prednosti vjetrogeneratora:

- Čist način proizvodnje električne energije.
- Pridonose globalnoj sigurnosti, za razliku od elektrana na nuklearno gorivo.
- Ne koriste gorivo, pokreće ga strujanje vjetra koje pogoni lopatice.
- Porast cijene goriva na svjetskom tržištu ne utječe na cijenu električne energije proizvedene na ovaj način.

### 2.3. Klasifikacije distribuiranih izvora

Prema [1] distribuirani izvori se uglavnom razvrstavaju prema različitim tipovima i prema različitim tehnologijama koje koriste prilikom svog rada. Prikladniji način razvrstavanja bio bi s elektrotehničkog stajališta, kako bi se proučio njihov utjecaj na elektroenergetski sustav. Distribuirani izvori se razvrstavaju prema njihovoj aplikaciji u energetsom sustavu, prema trajanju opskrbe, tipu proizvedene energije i prema obnovljivosti energije (obnovljiva ili neobnovljiva).

Različite tehnologije distribuiranih izvora su implementirane kako bi ispunile potrebe širokog spektra. Potrebe se razlikuju od slučaja, do slučaja ovisno o teretu i potrebama mreže. Prema [1] neka područja primjene su:

- **Rezervni izvor (engl. Standby)** – ovakav distribuirani izvor može se koristiti kao pomoćni, odnosno rezervni izvor napajanja osjetljivih potrošača, kao što su osjetljivi procesi u industriji i bolnice u slučaju nestanka električne energije u mreži.
- **Samostalni (engl. Stand alone)** – uglavnom se koriste prilikom opskrbe potrošača u izoliranim ili zabačenim područjima. Koriste se kada postoje geografske prepreke koje spajanje potrošača na mrežu čine skupim.
- **Ravnanje vršnog opterećenja (engl. Peak load shaving)** – cijena električne energije mijenja se ovisno o krivulji potrošnje električne energije te prema dostupnosti proizvodnje energije. Distribuirani izvori se stoga mogu koristiti pri opskrbi vršnih opterećenja, kako bi se smanjio financijski udarac na velike industrijske potrošače.
- **Opskrba ruralnih područja** – u ovoj primjeni, za distribuciju grijanja, rasvjete, hlađenje, komunikaciju, napajanje malih industrijskih i poslovnih objekata u ruralnom području.
- **Kogeneracija** – distribuirani izvori mogu pružiti potrošačima kombinaciju grijanja i električne energije. Proizvedena toplinska energija, pretvorbom kemijske energije goriva izgaranjem u električnu energiju, koristi se za grijanje ili proizvodnju pare za npr. bolnice te procesnu industriju .
- **Stalno opterećenje** – distribuirani izvor ponaša se kao temeljna elektrana s ciljem pokrivanja proizvodnje električne energije za stalno opterećenje sustava. Ovo najčešće koriste distributeri električne energije ili komunalna poduzeća koji posjeduju svoje vlastite distribuirane izvore.

Zaključno se podjela tipova distribuiranih izvora, prema [1], može prikazati tablicom 2.1. koja prikazuje osnove tipove distribuiranih izvora i njihovu glavnu primjenu.

*Tablica 2.1. Usporedba često korištenih tipova distribuirane energije. [1]*

Tip distribuiranog izvora	Glavna primjena
Mikro turbina	Koristi se za ravnanje vršnih opterećenja, kogeneraciju, proizvodnja energije za stalno opterećenja. Komercijalno su dostupne kao male jedinice (30-75 kW).
Gorivne ćelije	Prikladne su za kogeneraciju. Velike jedinice su prikladne za proizvodnju energije za stalno opterećenje.



	Komercijalno su dostupne kao male jedinice (3-250 kW), ali se modularnim spajanjem mogu koristiti za napajanje većih potrošača.
Fotonaponski sustav	Koriste se kao samostalni izvori i za proizvodnju energije za stalno opterećenje uz kombinaciju s baterijskim sustavima. Sustav koji se minimalno održava, a koristi se u telekomunikacijama, uličnoj rasvjeti te reklamnim sustavima
Vjetrogeneratori	Primjenjuje se kod kuća, gospodarskih objekata i procesne industrije u zabačenim područjima.
Tradicionalni agregati s unutarnjim izgaranjem (dizel motor)	U upotrebi su već dugi niz godina no imaju visoke razine emisije štetnih plinova, rukovanje i održavanje je skupo te postoji opasnost od biološkog onečišćenja prilikom transporta dizel goriva do potrošača na zabačenim lokacijama. Najčešće se koriste kao sustavi u slučaju nužde (pouzdati su) no nisu za kontinuiranu upotrebu.

## 2.4. Prednosti distribuiranih izvora

Implementacija distribuiranih izvora u elektroenergetski sustav ima velik broj prednosti. Prema [1] neke od prednosti implementacije distribuiranih izvora su:

### 1. S ekonomskog stajališta:

- Distribuirani izvori mogu podmiriti potrebe lokalnog povećanja tereta instalacijom na određene lokacije u mreži. S time se može smanjiti ili izbjeći potreba za izgradnjom novih prijenosnih i distribucijskih vodova, poboljšanje postojećeg energetskog sustava te smanjiti opterećenje prijenosne i distribucijske mreže u fazi planiranja.
- Distribuirani izvori su modularni, što znači da ih je lako montirati i instalirati na bilo kojoj lokaciji te stoga imaju veliki broj prednosti:
  - a) Mogu se instalirati u jako kratkom vremenu i na bilo kojoj lokaciji. Svaki modul odmah nakon instalacije može započeti s radom te nezavisno o ostalim modulima (ukoliko dođe do kvara na jednom modulu, ostali moduli mogu nesmetano nastaviti s radom).

b) Ukupni kapacitet proizvodnje moguće je mijenjati na način da se dodaju ili oduzimaju pojedini moduli.

- Nisu ograničeni centralizacijom energetskeg sustava jer se mogu instalirati bilo gdje. Fleksibilnost pri odabiru lokacije za instalaciju distribuiranih izvora ima pozitivan učinak na cijenu električne energije. Obnovljivi distribuirani izvori, kao što su hidro, vjetro i solarne jedinice, moraju ispuniti određene geografske uvjete.
- Distribuirani izvori su projektirani tako da se instaliranjem dodatnih modula u malim inkrementima pruža dodatna snaga koju kupac zahtjeva.
- Cijena električne energije energetskeg sustava može se smanjiti priključenjem distribuiranih izvora na mrežu te samim time se smanjuje potrebna snaga koju bi trebala proizvesti centralizirana elektrana.

**2. Sa stajališta vođenja:** distribuirani izvori imaju pozitivan učinak na naponski profil distribucijskeg sustava te na probleme vezane sa snagom.

- Distribuirani izvori mogu smanjiti gubitke u elektroenergetskom sustavu, smanjiti tokove snaga u prijenosnoj mreži kako bi se poboljšao naponski profil sustava.
- Pomoću distribuiranih izvora moguće je pokriti vršna opterećenja.
- Povećavaju pouzdanost energetskeg sustava jer postoji veći broj izvora u mreži, a ne samo jedan kao što je to kod centraliziranih sustava.
- U slučaju npr. elementarnih nepogoda moguće je instalirati distribuirane izvore na mjestima koja su pogođena nepogodom zbog ispada energetske mreže iz pogona.
- Što se tiče utjecaja na okoliš i društvo, obnovljivi distribuirani izvori eliminiraju ili smanjuju štetne emisije.

## **2.5. Opći problemi u proizvodnji električne energije pomoću distribuiranih izvora**

Elektroenergetski sustav postaje manje upravljiv ako se na postojeću mrežu počne priključivati sve više obnovljivih izvora (npr. vjetroelektrane, fotonaponski sustavi). Pomoću vremenske prognoze moguće je predvidjeti srednje brzine vjetra u kratkom periodu ali je nemoguće odrediti oscilacije u brzini vjetra koje se naglo događaju. Promjene u snazi koje vjetrotgenerator preda mreži nastaju zbog naglih promjena brzine vjetra koje zakreće elise te se tako narušava kvaliteta isporučene električne energije koja se isporučuje krajnjem kupcu (dolazi do poteškoća u regulaciji napona i frekvencije).

Prema [4] postoji više problema vezanih za priključenje neupravljivih distribuiranih izvora na postojeću mrežu:

- **Regulacija napona i kompenzacija jalove snage.** Prilikom distribucije električne energije krajnjim potrošačima potrebno je održati naponsku razinu u zadanim granicama propisanim normom. U slučaju priključenja neupravljivog distribuiranog izvora na mrežu dolazi do promjene tokova snage i promjenom naponskog profila ovisno o stanju opterećenja u mreži. Kako bi bilo moguće regulirati napon u čvorovima distributivne mreže potrebno je koristiti elemente kojima bi se ostvarila kompenzacija jalove snage.
- **Upravljanje snagom u mreži i reguliranje frekvencije.** U slučaju da dođe do otočnog režima rada, dolazi do problema s frekvencijom. U slučaju da u ostatku mreže u kojoj je došlo do otočnog režima rada nema ni jedna proizvodna jedinica reguliranog tipa, nije moguće održati frekvenciju u zadanim okvirima. Do otočnog načina rada dijela sustava može doći u slučaju da transformator koji spaja taj dio sustava na visokonaponsku mrežu iz nekog razloga prestane raditi.
- **Kut i stabilnost napona.** U slučaju kvara u mreži i u slučaju da distribuirani generator ispadne, izgubljena je lokalna proizvodnja u kratkom periodu koja će se vrlo brzo uspostaviti određenim mehanizmima. Prijelazne pojave imaju veliki utjecaj na stabilnost elektroenergetskog sustava ako je distribuirani generator bitan u sustavu. Ako se radi o asinkronom generatoru kod vjetrolektrana doći će do njegovog ubrzanja prilikom nestabilnog stanja te će generator iz mreže povlačiti jako veliku jalovu snagu te se tako napon snižava.
- **Kvaliteta električne energije.** Procjena kvalitete električne energije se vrši prema normi koja zadaje dopuštene prijelazne pojave napona i utjecaj viših harmonika na isporučeni napon. Do smanjenja kvalitete napona kod potrošača u distributivnoj mreži može doći ovisno o karakteristikama mreže i neupravljivih distribuiranih izvora.
- **Zaštita.** Kako u konvencionalnoj mreži postoji samo jedan smjer kretanja energije, kvar se može otkloniti ukoliko proradi samo jedan relej koji štiti mrežu (serijski nadstrujni relej). U slučaju priključenja neupravljivog distribuiranog izvora na postojeću distributivnu mrežu dolazi do tokova snage u oba smjera te postojeći zaštitni uređaji neće biti dovoljni i neće reagirati u slučaju nekih kvarova što nije dopustivo.
- **Raspoloživost i pouzdanost sustava.** Glavni motiv ulaganja u distribuirane izvore jest sigurnost da će se ulaganja brzo isplatiti. Distribuirani izvori u velikoj mjeri utječu na pouzdanost opskrbe električne energije ali nemaju toliko veliki utjecaj na pouzdanost

elektroenergetskog sustava. U nekim slučajevima distributeri imaju korist od distribuirane proizvodnje električne energije ako zbog implementacije takvih izvora uspiju smanjiti troškove električne energije ili uspiju odgoditi financijska ulaganja u mrežu.

Prilikom projektiranja distribucijskih mreža koriste se dvije načelne ideje: da se isporuči električna energija krajnjim potrošačima i to uz prihvatljivu kvalitetu i da se zaštiti mreže ako dođe do kvara. Analiza mreže zasniva se na proračunima kratkog spoja, stabilnosti i tokova snaga.

Napon u cijeloj distribucijskoj mreži potrebno je održavati u dozvoljenim granicama prilikom svih vrsta opterećenja jer su električni uređaji koje kupci priključuju na mrežu osjetljivi na velike promjene napona. Normalno pogonsko stanje elektroenergetskog sustava moguće je provjeriti pomoću proračuna tokova snaga. Pomoću proračuna tokova snaga moguće je znati napone u svim čvorištima u sustavu uz ulazne parametre kojima se zadaju snage proizvodnje i opterećenja u sustavu. Rezultati proračuna prikazuju iznos i kut napona te tokove snaga u svim elementima mreže. Pomoću tih proračuna moguće je odrediti stanje mreže pri različitim iznosima snaga opterećenja i pri raznim režimima proizvodnje.

Na oštećenja u distribucijskoj mreži može utjecati jako puno čimbenika. Prilikom izgradnje distribucijske mreže vodiči se odvajaju od zemlje i međusobno se razmiču pomoću raznih vrsta izolatora. U nekim slučajevima dolazi do pucanja izolatora što dovodi do stvaranja kratkog spoja između dva ili više vodiča. Kvarovi kao što je kratki spoj ne samo da ugrožavaju djelatnike pogona nego i oštećuju opremu u slučaju da se javi struja kratkog spoja velikog iznosa. Kako pogreške prilikom otkrivanja kvara čine neprihvatljivi rizik u sigurnosti potrebno je projektirati distribucijski sustav tako da struja kvara bude dovoljno velika te da ju elementi zaštite mogu prepoznati kao kvar i da ju sustav zaštite može uočiti u svim pogonskim uvjetima.

Prema [4] prilikom definiranja kratkih spojeva potrebno je naglasiti postojanje simetričnih i nesimetričnih kratkih spojeva. Spojevi kod kojih između struje i napona ne dolazi do izmjene simetrije nazivaju se simetrični kratki spojevi te takvi spojevi na sličan način obuhvaćaju sve tri faze. Jednofazni model mreže najčešće se koristi prilikom proračuna takvih kvarova. Kod nesimetričnih kratkih spojeva javlja se nesimetričnost u mreži i potrebna je puno kompleksnija analiza.

Prilikom izrade proračuna tokova snaga i struja kvarova uzima se pretpostavka da se sustav prije nastanka kvara nalazio u ravnotežnom stacionarnom stanju. Prema [4] ukoliko je sustav u stanju ravnoteže, napon je takav da protok snage ide u smjeru prema čvoru kod kojega je potrošnja

energije veća od proizvodnje. Svaki čvor je u konačnici u stanju ravnoteže u odnosu na snage proizvodnje, potrošnje i snage prenesene na druge čvorove.

## 2.6. Tehnički utjecaji distribuiranih izvora na distribucijski sustav

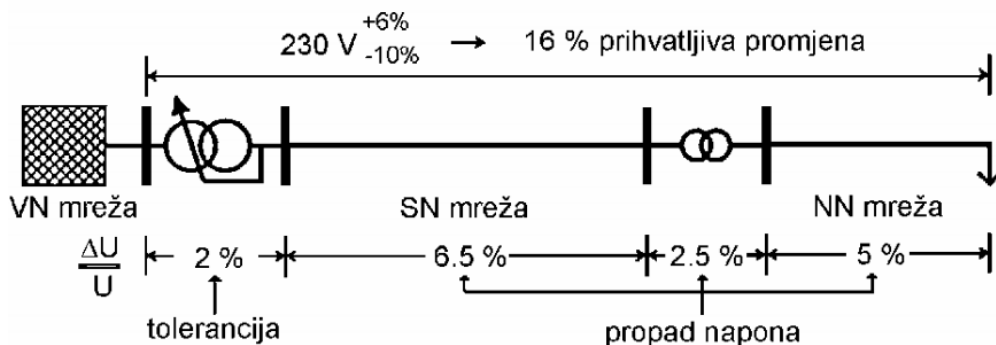
U slučaju da postoje distribuirani izvori spojeni na mrežu može se primijetiti njihov utjecaj na tehnička obilježja kao što su snaga kratkog spoja i zaštitni sustav kao i na kvalitetu napona. Prema [18] uglavnom nema promjena u pouzdanosti opskrbe radi spajanja distribuiranih izvora koji nisu povezani sa upravljačkim sustavom iz najvišeg središta u hijerarhiji jer nisu povezani s opterećenjem. Korištenjem lokalnih mjera moguće je odstraniti neke od nastalih utjecaja na mrežu, dok ostala obilježja uglavnom ovise o mrežnoj strukturi pa se zato gledaju sa stajališta dugoročnog planiranja sustava.

Ovisnost između distribuiranih izvora i pogona mreže može se razmatrati kroz dva aspekta:

- **Mrežna moć** je ograničena problemom stabilnosti napona i struje opteretivosti opreme. U slučaju da se želi postići veća mrežna moć to je moguće postići u određenom rasponu optimiranjem postojeće opreme ili kroz ulaganje u obnovu primarne opreme.
- **Priključni kriteriji** dolaze pod utjecaj tehnologija kojom je distribuirani izvor izveden te ostalih lokalnih mjera. Priključni kriteriji primjenjuju se na flikere, velika odstupanja u naponu, struju kratkog spoja te se primjenjuju na selektivnost zaštitnog sustava.

Jako je bitno pažljivo razmotriti napone u stacionarnom i prijelaznom stanju u mrežama srednjeg i niskog napona. Kolebanje snage na izlazu koja nije u direktnoj vezi s električnim teretom jedno je od osnovnih obilježja distribuiranih izvora kojima se ne može upravljati. Ukupno kolebanje napona se superponira na kolebanje koje već postoji u mreži te koje je uzrokovano s primjenom električnog tereta koji je priključen na mrežu. To superponiranje fluktuacije može dovesti do proširenja raspona napona prilikom normalnog pogona, a u tom slučaju se troši pričuva u mreži te se povećava neraspoloživost mreže.

Prema [4] svi potrošači priključeni na niskonaponsku mrežu primaju napon od 230 V +6%/-10% dok potrošači koji su priključeni na sredjenaponsku mrežu primaju napon  $U_n \pm 10\%$ .



Slika 2.5. Propad napona na mreži srednjeg napona..[4]

Na slici 2.5. prikazan je primjer izračuna za propad napona koji je prihvatljiv za mrežu srednjeg napona. Sa slike se uočava da nakon što se oduzme najveće očekivani propad napona u NN mreži od 5% i na transformatoru instaliranom na lokalnu mrežu od 2.5% te nakon uračunavanja tolerancije regulatora napona za transformator između VN i SN mreže od 2%, ograničenje raspona napona u SN mreži može iznositi 6.5%. Ako je u NN mreži propad napona manji, to znači da se povećava moć srednjenaponske mreže i obratno. Prijenosna moć mreže može se povećati pomoću dodatnih komponenti koje služe za regulaciju napona te su priključene na raznim lokacijama u takvoj mreži.

Kako bi se smanjila potreba za širenjem naponskih raspona zbog injektirane snage koja dolazi iz distribuiranog izvora te kako bi se obavilo optimiziranje tokova jalove snage na VN razinama u mreži, može se raditi kompenzacija jalove snage distribuiranih izvora. Samim tim nije dobro kod distribuiranog izvora postaviti faktor snage na konstantan iznos koji je uglavnom iznosa jedan, nego je potrebno dozvoliti određeni stupanj slobode kako bi se zadovoljili lokalni zahtjevi.

Prilikom priključenja distribuiranih izvora na mrežu za svaki pojedini slučaj priključivanja potrebno je izvršiti provjeru snage kratkog spoja te u slučaju ako se pokaže da je nužno, potrebno je i zadržati iznos snage kratkog spoja u dozvoljenom rasponu korištenja prikladnih mjera u elektrani kao što su konvertersko odvajanje ili na način da se primjeni ograničenje struje kratkog spoja. Razlog tome je što distribuirani izvori koji su priključeni na mrežu i imaju izravno spojene sinkrone ili asinkrone generatore mogu povećati snagu kratkog spoja iznad dozvoljenih granica odnosno iznad sposobnosti elemenata koji su ugrađeni u mrežu. Snaga kratkog spoja je obično najveća u čvoru VN/SN transformatora. Ovisno o tome kako je mreža konfigurirana, moguće je da snaga kratkog spoja dođe u gornju granicu u tom čvoru iako u srednjenaponskoj mreži nema dodatnog izvora snage.

Kada se priča o zaštiti distribuiranih izvora ne postoje posebne poteškoće prilikom realizacije zaštite proizvodnih jedinica jer se zaštita izvodi pomoću standardne oprema. Glavni problem se nalazi u projektiranju zaštite i njenoj koordinaciji jer su danas sustavi srednjenaponske mreže zasnovani na radijalnoj topologiji. Kako se ne gleda usmjerenost kvara doći će do prorade samo zaštitnih uređaja koji nadgledaju samo relevantne grane mreže dok bi preostali dio mreže ostao u pogonu. Ako zaštitni sustav ne uspije detektirati kvar, nezavisni nadstrujni releji s vremenski zatezanjem mogli bi neselektivno isključiti granu ili grane koje nisu pogođene kvarom. Također je potrebno omogućiti pouzdanije otkrivanje malih struja kvara, pogotovo ako mreža ima veliki broj grana jer postoji mogućnost da nastane problem vezan uz sustave zaštite te je nekad potrebno djelovati i sa strožim mjerama kao što je npr. ugradnja jednosmjernog nezavisnog nadstrujnog releja koji ima vremensko zatezanje.

Prilikom projektiranja potrebno je preventivno djelovati da ne bi došlo do pojava kod distribuiranih izvora koje bi utjecale na pouzdanost napajanja. Do toga bi moglo doći u slučaju da selektivnost zaštitnog sustava postane ugrožena, učinak automatskog ponovnog uklopa nije više zajamčena ili dolazi do pojave opasnosti od preopterećenja kabela. Ako se distribuirani izvor nalazi u stacionarnom stanju, ne smije doći do prelaska napona izvan ograničenja. Ako do toga dođe nužno je distribuirani izvor odvojiti s mreže ili je potrebno instalirati opremu na distribuirani izvor koja će pomoću automatskih mehanizama ograničiti napon.

Promjene kod tokova snaga ovise o promjenama gubitaka u promatranoj mreži. Gubici u mreži se povećavaju ako distribuirani izvor u velikoj količini prekoračuje iznos opterećenja. Gubici u mreži nisu sastavi dio pogonskog ograničenja ali zbog ekonomskih razloga potrebno ih je dovesti do minimuma. U slučaju da se neupravljivi distribuirani izvori priključuju na mrežu dolazi do povećanja troškova. Njihovo priključenje uzrokuje smanjenje mrežne moći koja je potrebna za napajanje dodatnih potrošača jer se granice očekivanih napona povećavaju.

## 2.7. Rješenja tehničkih utjecaja distribuiranih izvora na distribucijski sustav

U današnje vrijeme postoje dva pristupa prilikom planiranja priključenja distribuiranih izvora na postojeću mrežu [18]. Kod oba primjera pristup se svodi na praćenje norme EN 50160 kojom se ograničava kvaliteta napona koji se isporučuje potrošaču. Različitosti u pristupu su:

- O mogućnosti priključenja distribuiranih izvora na postojeću mrežu su značajni samo zahtjevi potrošača. Mrežni operator provjerava mogući nastanak smetnje u svakom pojedinom slučaju. Velika Britanija najviše koristi ovakav pristup. Norme koje opisuju kvalitetu električne energije definiraju inženjeri na temelju svoje stručne preporuke.
- Drugi pristup je donošenje posebnih pravila o priključenju velikog broja distribuiranih izvora kako bi se postupak što više pojednostavio. Pravila priključenja u Njemačkoj nastaju prema standardima koji opisuju pogon te također ovise o zahtjevima potrošača.

Prema [4], Njemačka, Danska, Španjolska i Nizozemska odstupanja napona u stacionarnom stanju drže puno ograničenije kada bi se to usporedilo sa vrijednostima koje su definirane normom EN 50160. Ta stroga ograničenja dolaze od pretpostavke da se ukupno odstupanje vrijednosti koje je prihvatljivo od strane potrošača ne bi smjelo predvidjeti za samo jedan distribuirani izvor, nego bi se trebao raspodijeliti između distribuiranog izvora i potrošača.

Kada je govor o potrošnji jalove snage koja se koristi za smanjenje iznosa povišenih napona u stacionarnom stanju vide se razlike u stajalištima Europskih zemalja. Nizozemski distributeri ne smatraju da je injektiranje jalove snage korisno dok se injektiranje jalove snage u mrežama drugih zemalja koristi samo pri posebnim uvjetima. Prema [4], niti jedna država nema definirane modele za naplatu potrošene jalove snage ako je ispunjen uvjet da je  $\cos\varphi > 0,9$ . Ako je  $\cos\varphi < 0,9$ , koriste se tarifni modeli za naplatu koji iznose od 0 do 0,015 €/kVAh u Njemačkoj, dok se cijene u Španjolskoj kreću od 0 do 4% od cijene kWh.

Prema [19], potrebno je osmisliti i koristiti nove matematičke modele za analiziranje sustava prilikom razmatranja distribuiranog izvora prilikom planiranja i pogona elektroenergetskog sustava. Potrebno je uvesti dodatne elemente povezane iz različite neizvjesnosti jer izlazna snaga distribuiranog izvora ima stohastičku ovisnosti. Kod distribuiranih izvora pojavljuju se sljedeće neizvjesnosti:

- **Predviđanje proizvodnje i opterećenja.** Prema [19] provedba uvrštavanja jedinice za proizvodnju u plan proizvodnje vrši se na temelju izrađenog plana opterećenja koje kod konvencionalnih sustava ima uglavnom neizvjesnost od 3%. Ovisno o tehnologiji



distribuiranog izvora potrebno je osmisliti nove i točnije modele pomoću kojih će se predviđati proizvodnja. Prikupljanje i skladištenje mjernih podataka je temeljni ključ u ovom procesu. Npr. u Danskoj sve konvencionalne proizvodne jedinice snage veće od 2 MW i gotovo sve velike vjetroelektrane imaju uređaje za mjerenje u stvarnom vremenu, dok se u manjim jedinicama problem mjerenja rješava pomoću snimača (engl. Recorder). U slučaju da postoji veliki broj distribuiranih izvora potrebno je posvetiti veliku pozornost mjerenju i pohrani mjernih podataka.

- **Prilagođavanje zahtjeva s aspekta snage pričuve i instaliranog kapaciteta.** Fluktuaciji proizvodnje doprinose proizvodne jedinice sa stohastičkim vladanjem. Iz tog razloga je potrebno proračunati i analizirati kolika je vjerojatnost da će doći do ispadanja određene snage proizvodnje. Kolebanja i ispadanja jedinica u proizvodnji imaju vrlo veliki utjecaj na iznos snage pričuve, kako primarne tako i sekundarne. Ovisno o vrsti i broju distribuiranih izvora koji su priključeni na mrežu potrebno je prilagođavati pričuvu kako bi zadovoljila zadane zahtjeve.
- **Planiranje prijenosne mreže (VN).** Veliki utjecaj na izradu plana visokonaponske prijenosne mreže imaju distribuirani izvori jer u pojedinim situacijama mogu biti potencijalna rješenja uobičajenom razvoju mreže. Raspoloživost mreže može biti povezana s trenutnim stanjem opterećenja mreže i može biti vremenski ovisna što ju čini predmetom planiranja. Kod preostalih ograničenja, mrežni operator može poduzeti rješenja u domeni okoline lokalne mreže (npr. za slučaj kada vod nije raspoloživ), također se mogu poduzeti rješenja u domeni generatora.
- **Planiranje distribucijske mreže (SN i NN).** Već postojeće mreže moraju se koristiti učinkovitije na način da se primjenjuju novi modeli u kojima su uključeni distribuirani izvori. Od velike je važnosti izraditi dobar proračun utjecaja distribuiranih izvora na distributivnu mrežu. Analizom slučajeva iz Njemačke i Nizozemske, moguće je uočiti da prilikom uporabe novih modela postoji mogućnost odgode ili izbjegavanja dodatnih investicija u mreži čime se povećava ukupna ekonomičnost [4].

### **3. EVOLUCIJSKI ALGORITMI I PROGRAMSKI PAKETI ZA OPTIMIZACIJU**

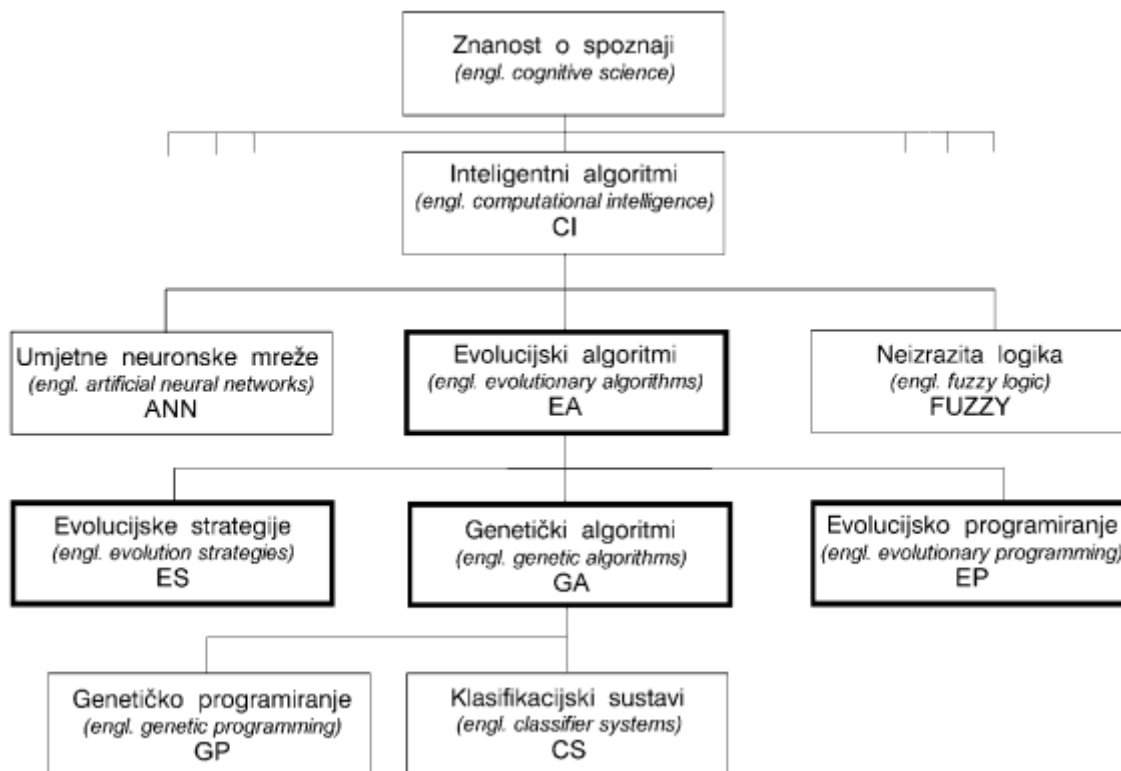
U ovom poglavlju radi se osvrst na osnovne pojmove koji opisuju evolucijske algoritme, njihov nastanak, način funkcioniranja te njihova primjena u tehničke svrhe. Budući da se evolucijski algoritmi koriste prilikom optimizacije tehničkih sustava, potrebno je dobiti podatke iz simulacije, stoga je opisan programski paket OpenDSS koji služi, između ostaloga, za simulaciju distribucijske mreže elektroenergetskog sustava. Također je opisan i programski jezik Python koji služi za analitičku obradu podataka, grafički prikaz podataka dvodimenzionalnim grafovima i za implementaciju diferencijalnog evolucijskog algoritma pomoću kojega se distribucijska mreža optimizira.

#### **3.1. Evolucijski algoritmi**

Kako razvojem tehnologije dolazi do sve složenijih problema inženjeri imaju sve veću potrebu za rješenjima prilagodljivim pojedinim zadacima. Jedan od pristupa rješavanju kompleksnih problema je imitiranje procesa evolucije u prirodi. Evolucijski algoritmi su skup stohastičkih metoda koje efikasno pretražuju prostor rješenja.

Križanjem kvalitetnijih rješenja i upotrebom mutacija početni skup nasumično odabranih rješenja kroz određeni broj iteracija evoluiraju te algoritam pronalazi sve bolja i bolja rješenja. Osnovna prednost ovih algoritama nad drugima je preslikavanje rješenja problema koji su prikazani kao eksponencijalne funkcije u polinomijalne funkcije. Budući da se ne pretraži čitav prostor rješenja, optimalno rješenje nije garantirano. Međutim, pronađena rješenja su u većini slučajeva optimalna ili vrlo blizu optimalnih.

Prema [9], evolucijski algoritmi (engl. evolutionary algorithms, evolutionary computing) su postupci optimiranja, učenja i modeliranja temeljeni na mehanizmima evolucije koji se mogu pronaći u prirodi. Ovakvi sustavi pokušavaju biti istog oblika kao i prirodna evolucija. Nastanak evolucijskih algoritama svodi se na dvije ideje: ideja da se prirodna evolucija bolje i detaljnije počne razumijevati i ideja za pokušajem primjenjivanja načela koje diktira prirodna evolucija za rješavanje različitih problema. Pedesetih godina prošlog stoljeća pojavljuju se neke ideje i primjeni evolucijskih algoritama, ali kako zbog nemogućnosti izvede tadašnjih računala odustaje se od te zamisli. Prema [9], tijekom šezdesetih godina prošlog stoljeća neovisno su se razvijala tri postupka koja su bila zasnovana na načelima biološke evolucije.



Slika 3.1. Podjela inteligentnih algoritama. [9]

Na slici 3.1. prikazana je podjela inteligentnih algoritama među kojima se nalaze i evolucijski algoritmi koji se koriste prilikom izrade simulacija potrebnih za ovaj rad.

Teorija biološke evolucije temelji se na nekoliko opće prihvaćenih pretpostavki [9]:

- a) Postoji naraštaj odnosno populacija (engl. *population*) koja može stvoriti potomke. S aspekta biologije naraštaj je skup živih organizama koji su iste vrste te borave na ograničenom području u određenom trenutku. Moraju postojati naraštaji od više jedinki koje će se razmnožavati kako bi mehanizam evolucije mogao funkcionirati. Postojanje naraštaja s više jedinki različitih značajki je nužno jer se evolucija kao mehanizam temelji na prirodnom izboru. Populaciju kod evolucijskih algoritama predstavljaju skupovi objekata koji su matematički neovisni, kao što su: realni brojevi, vektori, binarni brojevi i itd.
- b) Postojeći naraštaj se razmnožava, odnosno stvaraju se potomci (rekombinacija, engl. *recombine*). Sa gledišta biologije novi naraštaji nastaju iz postojećeg naraštaja postupkom razmnožavanja, odnosno rekombinacijom. Pri tom novi naraštaj nasljeđuje značajke svojih prethodnika. Rekombinacija se kod evolucijskih algoritama obavlja tako da se stvaraju

novi matematički objekti koji se temelje na postojećim objektima tako da se koriste prikladnih matematički operatori.

- c) Događaju se slučajne nejednakosti među jedinkama, odnosno mutacije (engl. *mutatuion*), tj. nastaju varijacije koje su uzrokovane slučajnim promjenama značajki jedinki. Nasljedne značajke se u biologiji mogu mijenjati jedino pomoću procesa mutacije, odnosno pomoću slučajne promjene koja se dogodi prilikom nasljeđivanja. Moguće je da zbog mutacije jedan od potomaka ima značajke koje nema ni jedan prethodnik. Mutacija je kod evolucijskih algoritama slučajna promjena matematičkog objekta nastala stohastičkom uporabom odgovarajućeg matematičkog operatora na taj matematički objekt.
- d) Samo neke jedinke mogu se razmnožavati i prenositi svoje značajke nasljeđivanjem. U prirodi se ovo zove prirodni odabir (engl. *natural selection*) tj. preživljavanje najsposobnijih jedinki (engl. *survival of the fittest*). Okoliš dopušta preživljavanje samo dijela, a ne svih jedinki naraštaja. Pretpostavlja se da će preživjeti one koje su bolje prilagođene okolišu. Pod preživljavanjem se podrazumijeva prosljeđivanje nasljednih značajki u sljedeći naraštaj. Proces izbacivanja pojedinih matematičkih objekata populacije se kod evolucijskih algoritama naziva izbor. Kod evolucijskih algoritama izbor je odbacivanje nekih matematičkih objekata populacije.
- e) Proces nastajanja novih naraštaja se ponavlja, odnosno proces je iterativan. Na slici 3.2 prikazan je shematski prikaz evolucije, tj. opetovanje postupka razmnožavanja i prirodnog izbora. Dok se ne postignu zadovoljavajuća rješenja postavljene zadaće ili se ne zadovolje neki drugi kriteriji za završetak evolucijskog algoritma postupci rekombinacije, mutacije i izbora se ponavljaju.



Slika 3.2. Shematski prikaz evolucije. [9]

Mišljenje da je evolucija napredak je pogrešno mišljenje jer evolucija stvara naraštaje koji bi po svojim karakteristikama bili bolji ili lošiji od prijašnjih naraštaja nego su novi naraštaji više prilagođeni svom okolišu.

Prema [5], evolucijski algoritmi su populacijsko orijentirani matematički optimizacijski algoritmi koji koriste biologijom inspirirane mehanizme kao što su mutacije, križanje, prirodni odabir, opstanak najsnažnijih s ciljem pronalaženja rješenja iterativnim putem.

Prednost evolucijskih algoritama u usporedbi sa ostalim optimizacijskim metodama je njihov karakter „crne kutije“ kod kojega je potrebno samo nekoliko pretpostavki za rješavanje problema. Definiranje funkcija za rješavanje problema obično zahtijeva manji uvid u strukturu samog problema nego što je to potrebno prilikom ručne konstrukcije prihvatljivog rješenja.

### **3.1.1. Značajke evolucijskih algoritama**

Prema [9] evolucijski algoritmi su dio šire grupe inteligentnih algoritama. Neka od svojstava evolucijskih algoritama su:

- a) Implicitno su paralelni jer evolucijski algoritmi odjednom pretražuju prostor rješenja za sve parametre.
- b) Evolucijski algoritmi su stohastičke naravi jer prilikom pronalaženja rješenja koriste stohastičke, a ne determinističke metode. Na temelju postojećeg rješenja uporabom operatora izbora, mutacije i rekombinacije, rabeći pri tome stohastičke metode, evolucijski algoritmi iterativno traže novo ili potencijalno bolje rješenje.
- c) Jedna od temeljnih odlika evolucijskih algoritama je svojstvo pronalaska globalnog ekstrema neovisno o vrsti zadatka koji se rješava.
- d) Evolucijski algoritmi se vrlo jednostavno implementiraju u postojeće matematičke modele, simulacije i stvarne procese te ih je vrlo jednostavno proširi i nadograditi ukoliko je to potrebno.
- e) Postupak stvaranja hibridnog sustava za optimizaciju vrlo je jednostavno izvediv tako što se evolucijski algoritam koristi za pronalazak približnog globalnog rješenja a nakon toga se pronalazak točnog rješenja problema izvodi pomoću neke determinističke metode.
- f) Prilikom korištenja evolucijskih algoritama koriste se izvorni oblici funkcije direktno, a ne izvedena svojstva funkcije. Prilikom korištenja determinističkih metoda često se koriste neka od svojstava funkcije kao što su npr. derivacija funkcije, što nije izvorni oblik funkcije.

- g) Gotovo svaki zadatak koji se može formulirati kao optimiranje moguće je riješiti pomoću evolucijskog algoritma te je upravo to jedan od razloga zašto su evolucijski algoritmi toliko popularni kod znanstvenika i inženjera.
- h) Evolucijski algoritmi pronalaze samo približno rješenje i nema garancije da će algoritam pronaći najbolje rješenje, odnosno apsolutni ekstrem. Pronalazak približnog rješenja je posljedica stohastičke prirode algoritma. U praksi je ovo bitno jer ako druge metode ne mogu uopće pronaći rješenje problema onda je približno rješenje evolucijskog algoritma najbolje rješenje s kojim se raspolaze.

### 3.1.2. Osnovni ciklus evolucijskih algoritama

U praksi postoji veliki broj inačica evolucijskog algoritma ali sve sve one mogu prikazati, prema [10], općim jednostavnim pseudokodom:

```

1. t: = 0;
2. initialize P(t);
3. evaluate P(t);
4. while not terminate do
5.     P'(t): = variation [P(t)];
6.     evaluate [P'(t)];
7. P(t+1): = select[P'(t) union Q(t)];
8. t: = t+1
9. od

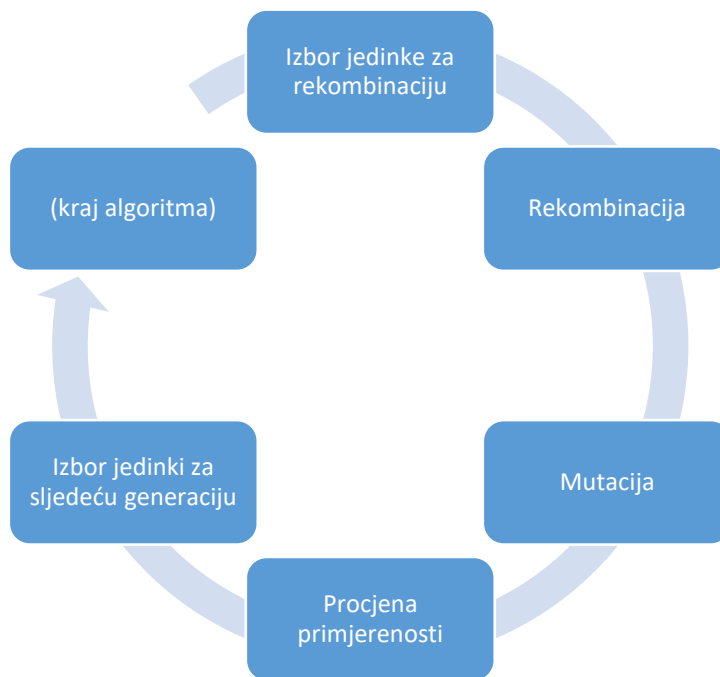
```

gdje je:

t	varijabla koja predstavlja generaciju populacije, tj. broj iteracija, $t \in N$ ;
P(t)	skup od $\mu$ jedinki u broju ponavljanja $t$ , drugim riječima ovo je skup roditelja;
P'(t)	skup od $\lambda$ jedinki u broju ponavljanja $t$ , odnosno skup potomaka koji su proizašli iz populacije P(t);
Q(t)	odabrani skup jedinki roditelja koji idu u sljedeću generaciju. U sljedeću generaciju populacije P(t+1) ne mogu doći svi članovi iz sadašnje generacije populacije P(t) već samo oni koji su odabrani. Matemački je Q(t) podskup skupa P(t);
Initialize	operand koji diktira početak stvaranja populacije;
Evaluate	operand koji procjenjuje primjerenost jedinke u populaciji;

- Terminate     operand koji diktira završetka algoritma;
- Variation     operand koji diktira promjene u genotipu kod jedinke;
- Select         operand koji je zadužen za izbor jedinki za sljedeću generaciju.

Evolucijski algoritam moguće je prikazati pomoću iteracijske petlje u kojoj je operator „variation“ rasčlanjen na operatore rekombinacije, mutacije i operator izbora jedinki, kao što je prikazano na slici 3.3.



*Slika 3.3. Iteracijska petlja evolucijskog algoritma.*

### 3.1.3. Standardni diferencijalni evolucijski algoritam

Općenito diferencijalni evolucijski algoritam ima pet koraka. Prema [17] tih pet koraka prikazano je pseudo kodom koji prikazuje jednostavnu strukturu:

```
1. import veličina populacije, generacija, broj varijabli
2. while uvjet
3. {
4.   for i=0 to NP, i++
5.   {
6.     Odabir tri vektora iz populacije nasumično
7.     U_(i(mut))^(G)=U_a^(G)+SF(U_b^(G))-U_c^(G) )
8.     for j=1 to VN; j++
9.     {
10.      if rand(0,1)<CR
11.        U_(ji(cross))=U_mut
12.      else
13.        U_(ji(cross))=U_ji
14.      } %kraj operacije križanja
15.      % početak operacije selekcije
16.      if f(U_(i(cros))) ≤f(U_i)
17.        U_i=U_(i(cros))
18.      else
19.        U_i=U_i
20.      % kraj operacije selekcije
21.    }
22.  } % kraj while petlje
```

**Inicijalizacija:** Ovaj algoritam se zasniva na populaciji, prema izrazu (3-1) unosi se početna populacija kako bi počeo proces optimizacije:

$$\bar{Z}_{i,k}^G = \bar{Z}_L^{MIN} + rand \cdot (\bar{Z}_L^{MAX} - \bar{Z}_L^{MIN}) \quad (3-1)$$

gdje je  $i \in [1, P]$ ,  $L \in [1, V]$ . Dimenzija diferencijalnog evolucijskog algoritma ovisi o veličini populacije  $P$  i varijabli  $V$ .  $\bar{Z}_L^{MIN}$  i  $\bar{Z}_L^{MAX}$  su donja i gornja granica odabrane ovisno o tipu problema. Član *rand* generira nasumičnu vrijednost u rasponu  $[1,0]$ .

**Mutacija:** Početna populacija je mutirana prema izrazu (3-2). Operacija mutacije pomaže algoritmu da se makne od lokalnih minimuma. Odabiru se nasumično tri vektora iz početne populacije  $Z_1$ ,  $Z_2$  i  $Z_3$ . Glavni kriterij za stvaranje mutirane matrice je faktor skaliranja,  $F$ , koji je u rasponu  $[0,2]$ .

$$\bar{Z}_{m,i}^{G+1} = \bar{Z}_1^G + F(\bar{Z}_2^G - \bar{Z}_3^G) \quad (3-2)$$

**Križanje:** Prilikom procesa križanja od prethodne populacije proizvodi se slijedeća populacija. Operacija križanja se ne izvodi na cijeloj populaciji već se gleda parametar  $CR$  koji predstavlja stopu križanja. Parametar  $CR$  ima realnu vrijednost  $[0,1]$ .

$$\bar{Z}_{c,ji}^G = \begin{cases} \bar{Z}_i^{G+1}, & \text{ako je } j \in J \\ \bar{Z}_i^G, & \text{u suprotnom} \end{cases} \quad (3-3)$$



gdje se  $\bar{Z}_{c,ji}^G$  odnosi na  $j$ -ti element vektora  $\bar{Z}_i^G$ , a  $J$  je skup elemenata točaka križanja.

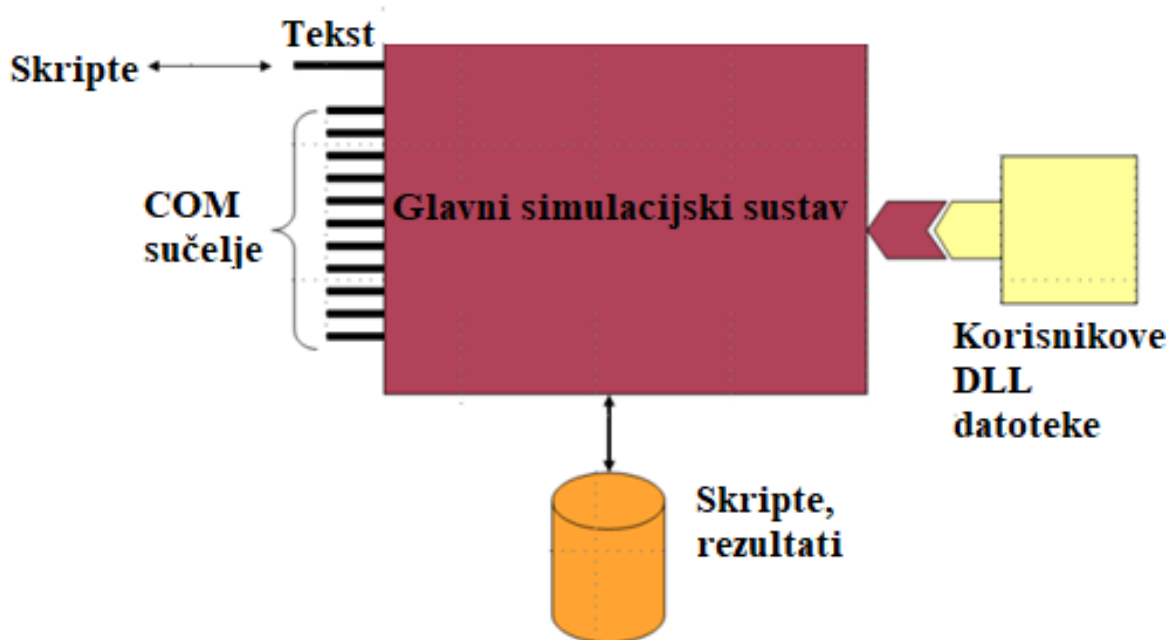
**Odabir:** u ovoj fazi algoritam koristi operator kako bi odabrao optimalno rješenje. Ako je stopa križanja veća od nasumične vrijednosti uzimaju se vektori iz koraka mutacije, u suprotnom slučaju odabir se vrši iz početne populacije. Drugim riječima operator odabira odlučuje između početne matrice  $Z_i$  i matrice križanja  $Z_{ij}$ .

$$\bar{Z}_{s,ji}^G = \begin{cases} \bar{Z}_{c,i}^G, & \text{ako je } \alpha_j \leq CR \text{ ili } j = \gamma \\ \bar{Z}_i^G, & \text{u suprotnom} \end{cases} \quad (3-4)$$

gdje je  $i = 1, \dots, P, j = 1, \dots, V, \alpha_j$  i  $\gamma$  se biraju nasumično u opsegu  $[1, \dots, V]$ .

### 3.2. Programski paket za analizu distribucijske mreže – OpenDSS

Prema [6] OpenDSS (engl. Open Distribution System Simulator) je sveobuhvatni alat za simulaciju distribucijskih sustava. Program je otvorenog koda (engl. Open Source). OpenDSS je zamišljen kao samostalni izvršni program i kao COM (engl. Component Object Mode) poslužitelj koji se može pokretati i upravljati pomoću raznih postojećih softverskih platformi. Izvršna inačica ima osnovno korisničko sučelje temeljeno na tekstualnom načinu unosa naredbi. Program podržava gotovo sve RMS analize (npr. frekvencijska analiza) koje se često koriste za planiranje i analizu distribucijskih sustava. Osim toga, OpenDSS podržava i mnoge nove vrste analize koje su dizajnirane za zadovoljavanje budućih potreba, od kojih su mnoge diktirane deregulacijom komunalnih usluga širom svijeta i pojavom pametnih mreža (engl. Smart Grid). Prvobitna namjena većine značajki programa bila je analiza distribuirane proizvodnje. Ostale značajke podržavaju analizu energetske učinkovitosti isporuke električne energije, analizu pametnih mreža, te analizu harmonika. OpenDSS je dizajniran da bude neograničeno proširiv tako da ga je moguće jednostavno modificirati i nadograditi kako bi se zadovoljile buduće potrebe.



Slika 3.4. Struktura OpenDSS-a. [6]

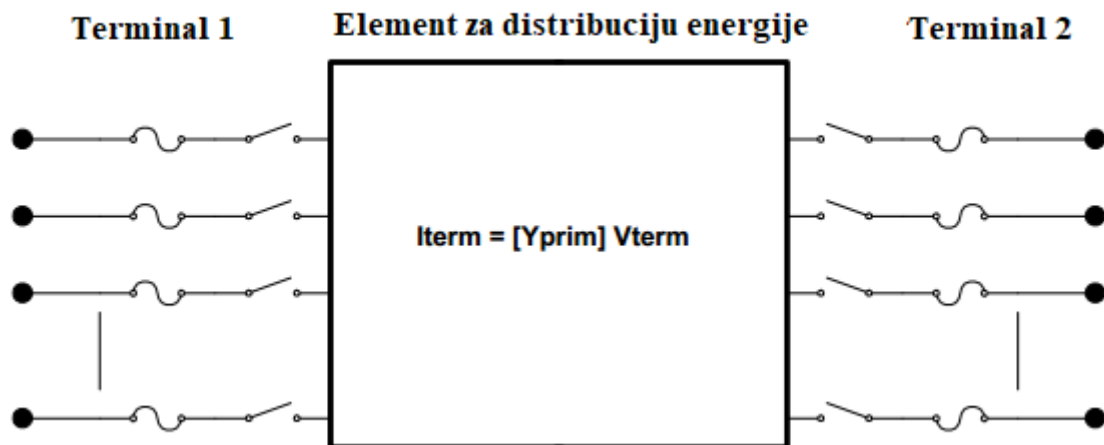
Kroz COM sučelje, korisnik je u mogućnosti dizajnirati i izvršavati prilagođene naredbe za rješavanje problema iz vanjskog programa i obavljati funkcije simulatora, uključujući definiranje podataka modela. Dakle prema [6], DSS se može implementirati putem bilo kojeg programskog paketa koji podržava upravljanje putem COM-a. Korisnici obično OpenDSS-om upravljaju pomoću MATLAB-a, Pythona, C#, R ili nekog drugog programskog jezika. Ovaj način upravljanja omogućava vanjsku analitičku obradu podataka te grafički prikaz rezultata. COM sučelje također pruža izravan pristup sučelju za tekstualne naredbe kao te pristup raznim metodama i mnogim svojstvima modela koji se simulira.

Putem tekstualnog sučelja korisnik može generirati skriptu za izvršavanje simulacije. Unos se može preusmjeriti i na tekstualnu datoteku kako bi se ostvario isti učinak kao i na mikronaredbe u nekom programskom okruženju. Mnogi rezultati mogu se dohvatiti putem COM sučelja, kao i iz vanjskih datoteka u kojima su spremljeni neki rezultati. Mnogi rezultati su zapisani u CSV formatu (engl. Comma-Separated Value) koji se vrlo lako može uvesti u druge programske pakete kao što su Microsoft Excel ili MATLAB za daljnju analitičku obradu.

### 3.2.1. Elementi za distribuciju i konverziju energije u OpenDSS-u

Elementi za distribuciju energije (Slika 3.5.) obično se sastoje od dva ili više višefaznih terminala. Njihova osnovna funkcija je isporuka energije iz jedne točke u drugu. U elektroenergetskom

sustavu najčešći elementi za distribuciju energije su vodovi i transformatori. Dakle, ti elementi obično imaju više od jednog terminala (kondenzatori i zavojnice mogu biti iznimka). Elementi za isporuku energije su uobičajeni električni elementi potpuno definirani s efektivnim vrijednostima u stabilnom stanju pomoću njihovih impedancija. Prema tome, mogu se u potpunosti prikazati pomoću primitivne Y matrice ( $Y_{prim}$ ).



Slika 3.5. Primjer elementa za distribuciju energije. [6]

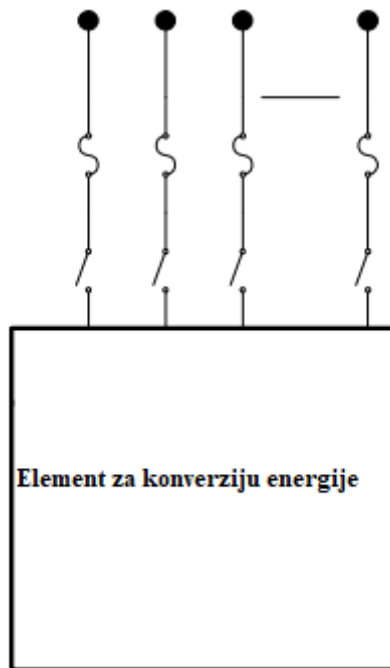
Elementi za konverziju energije (Slika 3.6.) pretvaraju električnu energiju u neki drugi oblik energije. Neki elementi mogu privremeno pohraniti energiju, a zatim ju vratiti natrag, kao što je slučaj kod reaktivnih elemenata. Većina će imati samo jednu vezu sa elektroenergetskim sustavom i stoga samo jedan višefazni terminal. Opis mehaničke ili termalne strane pretvorbe energije nalazi se u modelu crne kutije. Prema [6] opis može biti jednostavna impedancija ili komplicirani skup diferencijalnih jednadžbi koje daju jednadžbu ubrizgavanja struje:

$$I_{Term} = F(V_{Term}, [Stanje], t) \quad (3-4)$$

Funkcija  $F$  će se mijenjati ovisno o vrsti simulacije koja se izvodi. Elementi za konverziju energije također moraju biti sposobni raditi i sa parcijalnim matricama kada je to potrebno:

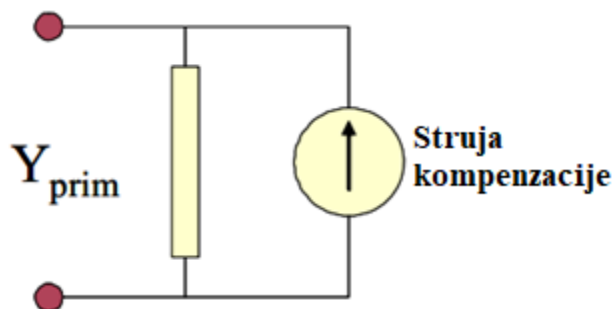
$$\frac{\partial F}{\partial V} \quad (3-5)$$

U jednostavnim slučajevima to će jednostavno biti primitivna Y matrica (admitancija), odnosno Y matrica samo za ovaj element.

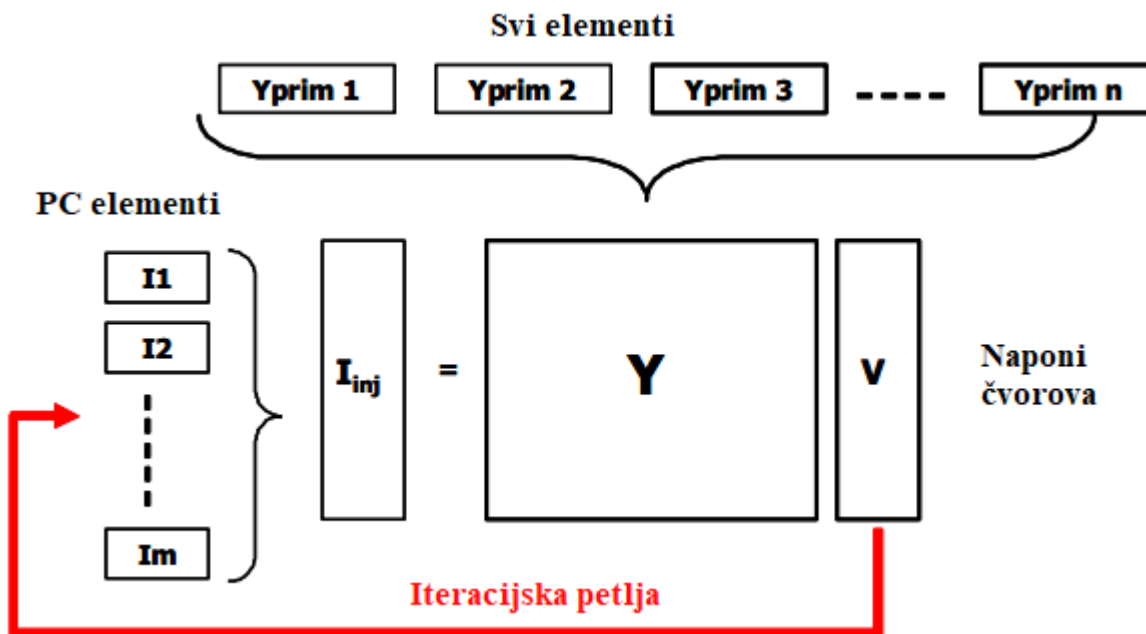


Slika 3.6. Element za konverziju energije. [6]

U okviru OpenDSS-a, tipična implementacija elemenata za konverziju energije prikazana je na slici 3.7. Nelinearni elementi, posebno elementi opterećenja i generatori, tretiraju se kao Nortonovi elementi s konstantom primitivnom  $Y$  matricom i kompenzacijskom strujom ili strujom ubrizgavanja koja kompenzira nelinearni dio. Ovo funkcionira dobro za većinu distribucijskih opterećenja i omogućuje široki raspon modela opterećenja. Dobro se konvergira za veliku većinu tipičnih uvjeta distribucijskih sustava. Primitivna  $Y$  matrica se općenito održava konstantnom zbog efikasnosti računanja, iako OpenDSS to ne zahtijeva.



Slika 3.7. Model elementa za konverziju energije sa kompenzacijom struje. [6]



Slika 3.8. Petlja za rješavanje problema u OpenDSS-u. [6]

U osnovi, struktura OpenDSS-a (Slika 3.8.) upravlja stvaranjem i modificira primitivne Y matrice za svaki element u mreži. Isto tako upravlja sa listom sabirnica, sakupljanjem rezultata kroz mjerne elemente i izvršenjem kontrolnih elemenata.

Prvi rezultati napona dobiju se tako što se obavlja simulacija toka snaga bez opterećenja. To znači da su svi elementi osim elemenata za distribuciju odvojeni s mreže. Na taj način se dobiju svi fazni kutovi i amplitude napona koje su u odgovarajućem odnosu. Ovo je bitno jer je OpenDSS napravljen za rješavanje n-faznih mreža u kojima mogu postojati razne vrste transformatorskih omjera i spojeva.

Ciklus iteracije počinje prikupljanjem struja ubrizgavanja na svim elementima za konverziju energije (PC) u sustavu i dodavanjem u odgovarajuće polje u  $I_{inj}$  vektoru. Ciklus se ponavlja dok naponi ne konvergiraju na vrijednost od 0.0001 p.u. Sustavna Y matrica se uglavnom ne obnavlja tijekom ovog procesa pa stoga iteracije idu brže.

Utvrđeno je da su ova jednostavna iterativna rješenja dobro konvergirala za većinu distribucijskih sustava koji imaju odgovarajući kapacitet izvora za napajanje trošila. Ključ je imati dominantni izvor napajanja, što je slučaj kod većine distribucijskih sustava. U OpenDSS-u ovo se rješava pomoću normalnog algoritma. Također postoji i Newtonov algoritam za teže sustave (Nije isto što i tipična Newton-Raphsonova metoda za tokove snaga).

Prilikom izvođenja dnevnih ili godišnjih simulacija, rješenje u trenutnom vremenskom koraku koristi se kao polazište za rješenje u sljedećem vremenskom koraku. Ako postoji velika promjena u opterećenju rješenje će obično konvergirati u dvije iteracije – jedna za rješenje a druga za provjeru je li konvergirala. Dakle, OpenDSS je u stanju izvršiti takve izračune vrlo učinkovito. Kontrolne iteracije izvode se izvan ove petlje. Odnosno, postiže se konvergirano rješenje prije nego što se izvrši provjera potrebe za kontrolnim radnjama.

### 3.2.1. OpenDSS sintaksa

U najosnovnijem obliku naredba u OpenDSS-u ima sljedeći oblik:

```
1. Command parm1 parm2 parm3 parm4 ...
```

gdje je *Command* naredba koja se želi izvršiti, dok su *parm1*, *parm2* itd. parametri koji opisuju naredbu. Parametri su odvojeni zarezom ili razmakom (obični razmak ili tabulatorom). Ako parametar sadrži delimiter potrebno ga je zatvoriti u:

- Dvostruke navodnike (npr. „...“),
- Jednostruke navodnike (npr. '...'),
- Oblu zagradu (npr. (...)),
- Uglatu zagradu (npr. [... ]),
- Vitičastu zagradu (npr. {...}).

Dok će svaki od ovih načina zadovoljiti sintaksu programskog jezika preporuča se zatvaranje s jednostrukim navodnicima jer je taj način najprikladniji za *string* oblik podataka, dok su uglate zagrade prikladne za polja i matrice, a vitičaste zagrade za neke matematičke izraze.

### 3.2.2. Definiranje objekata i njihovih parametara

Na temelju primjera prikazano je kako se u OpenDSS-u stvaraju novi objekti i kako se definiraju njihovi parametri. Prikazani su objekti koji se najčešće koriste te su osnovni dijelovi neke mreže.

Kako bi se stvorio novi objekt koji predstavlja izvor ili nadomjesnu mrežu na koju je spojen definirani dio mreže, potrebno je napisati naredbu oblika:

```
1. New object=circuit.ieee123  
2. ~ basekv=4.16 Bus1=150 pu=1.00 R1=0 X1=0.0001 R0=0 X0=0.001
```

gdje se definira novi objekt odnosno izvor („*circuit*“) pod nazivom „circuit.ieee123“ koji ima bazni napon od 4,16 kV. Izvor je spojen na sabirnicu „150“ te je vrijednost izvora postavljena na 1 p.u. (engl. per-unit). Nakon toga su definirani otpori i reaktancije koji opisuju izvor.

Kako bi se definirao vod koji spaja dvije sabirnice potrebi je upisati naredbu:

```
1. New Line.L3 Phases=3 Bus1=1.1.2.3 bus2=7.1.2.3 LineCode=1 Lenght=0.3
```

gdje je definiran novi objekt voda („Line“) pod nazivom „L3“. Broj faza je tri, a s jedne strane je spojen na sabirnicu „1.1.2.3“ dok je s druge strane povezan na sabirnicu „7.1.2.3“. Duljina voda je 0.3 km te se putem naredbe „LineCode“ poziva na dio koda koji opisuje parametre voda:

```
1. New LineCode.1 nphases=3 BaseFreq=50
2. ~ rmatrix = [0.08666 | 0.02954 0.08837 | 0.0290 0.02992 0.08740]
3. ~ xmatrix = [0.20416 | 0.09501 0.19852 | 0.0728 0.08022 0.20172]
4. ~ cmatrix = [2.85171 | -0.9202 3.0046 | -0.356 -0.585 2.71132]
```

Naredbom „LineCode“ stvara se popis parametara voda te mu se dodjeljuje ime „1“. Broj faza je tri dok je bazna frekvencija 50 Hz. Ispod toga se definiraju matrice otpora, reaktancija i kapaciteta koje opisuju svojstva voda. Ako zasebno nisu definirane jedinice one iznose: za otpor i reaktanciju  $\Omega/\text{km}$  dok za kapacitet iznose nF/km.

Kako bi korisnik stvorio transformator u OpenDSS-u potrebno je kreirati objekt:

```
1. New Transformer.XFM1 Phases=3 Windings=2 Xh1=2.72
2. ~ wdg=1 bus=61s conn=Delta kv=4.16 kva=150 %r=0.635
3. ~ wdg=2 bus=610 conn=Delta kv=0.48 kva=150 %r=0.635
```

Naredbom „Transformer“ kreira se objekt transformator pod nazivom „XFM1“ s tri faze i dva namota. Parametar Xh1 (ili X12) definira reaktanciju u postotku između prvog i drugog namota transformatora. Potom je definirano da je prvi namotaj spojen u spoju trokut na sabirnicu pod nazivom „61s“, bazni napon mu je 4.16 kV, snaga 150 kVA te je otpor namota pri nazivnoj snazi izražen u postotku i iznosi 0.635. Drugi namotaj se definira na isti način kao i prvi.

Objekt trošila definira se na sljedeći način:

```
1. New Load.S1a Bus1=1.1 Phases=1 Conn=Wye Model=1 kV=2.4 kW=40.0 kvar=20.0 dai
ly=teret
```

Naredbom „Load“ kreira se objekt tereta, odnosno trošila pod nazivom „S1a“ koji je spojen na sabirnicu 1.1.. Parametar „Phases“ označava da se radi o monofaznom trošilu (u slučaju da je ovaj parametar iznosa 3, značilo bi da se radi o trofaznom trošilu. Parametar „Conn“ označava naziv spoja koji je u ovom slučaju zvijezda spoj. Također, potrebno je zadati nazivni napon te radnu i jalovu snagu trošila. Parametar „daily“ definira krivulju dnevnog opterećenja koja se u ovom slučaju zove „teret“. Prema [6], moguće je zadati različite modele tereta pomoću parametra „Model“ kao što je prikazano u tablici 3.1.:

Tablica 3.1. Različiti tipovi modela opterećenja. [6]

Tip modela	Opis modela opterećenja
1	<b>Konstantan <math>P</math> i konstantan <math>Q</math>:</b> općenito se naziva model konstantnog opterećenja. Ovaj model se najčešće koristi prilikom analiza tokova snaga. Može doći do konvergencije u slučaju ako znatno odstupa previše od nazivnih vrijednosti.
2	<b>Konstantan <math>Z</math> (konstantna impedancija):</b> $P$ i $Q$ ovise o kvadratu napona. Ovaj model opterećenja obično jamči konvergenciju u bilo kojem stanju terećenja. Model je uglavnom linearan.
3	<b>Konstantan <math>P</math> i kvadratni <math>Q</math>:</b> reaktivna snaga, $Q$ , ovisi o kvadratu napona, dok radna snaga $P$ , ne ovisi o naponu.
4	<b>Eksponencijalni model opterećenja:</b> ovisnost $P$ i $Q$ o naponu definira se eksponencijalno. Ovaj model opterećenja se uglavnom koristi prilikom provođenja studija smanjenja napona. Također se koristi u slučajevima kada nije poznato kako se opterećenje točno ponaša.
5	<b>Konstantan <math>I</math> (konstantan iznos struje):</b> snage $P$ i $Q$ variraju s promjenom iznosa napona dok je iznos struje opterećenja konstantan. Ovo je uobičajeno prilikom analize distribucijskih sustava.
6	<b>Konstantan <math>P</math> i fiksni <math>Q</math></b> (ako je snaga definirana kao konstantna tada ju krivulja opterećenja može modificirati pomoću multiplikatora dok snaga koja je definirana kao fiksna ne može biti promijenjena, odnosno u ovom slučaju snaga $Q$ je neovisna o vremenu i naponu.
7	<b>Konstantan <math>P</math> i kvadratni <math>Q</math>:</b> reaktivna snaga $Q$ ovisi o kvadratu napona.
8	<b>ZIP model opterećenja:</b> snage $P$ i $Q$ opisane su mješavinom konstantne snage, konstantnog iznosa struje i konstantne impedancija.



### 3.3. Programski jezik Python

Python je programski jezik opće namjene, interpretiran i visoke razine. Stvorio ga je Guido van Rossum 1990. godine. Prema [7] Python dopušta programeru korištenje nekoliko stilova programiranja kao što su objektno orijentirano, strukturno i aspektno orijentirano programiranje. Programski jezik Python moguće je nadograditi s raznim dodacima kao što je npr. paket *SciPy* koji služi za rješavanje matematičkih, znanstvenih i inženjerskih problema. U ovom dodatku nalaze se neki korisni paketi kao što je *NumPy* koji služi prvenstveno za rješavanje n-dimenzionalnih matrica, Fourierovu transformaciju te je sposoban generirati nasumične brojeve. *Matplotlib* omogućuje korisniku prikazivanje obrađenih podataka na dvodimenzionalnim grafovima. Paket *scipy.optimize* pruža korisniku mogućnost raznih vrsta optimizacije, između ostaloga nudi i diferencijalni evolucijski algoritam.

#### 3.3.1. Diferencijalni evolucijski algoritam u programskom jeziku Python

Instalacijom dodatka *SciPy* dobiva se mogućnost korištenja optimizacijskih alata. Prema [8], diferencijalni evolucijski algoritam u ovom slučaju pronalazi globalni minimum funkcije s više varijacija. Diferencijalni evolucijski algoritam je stohastičke prirode (ne koristi metode gradijenta) te se temeljni na populaciji podataka i koristan je za globalnu optimizaciju problema. Algoritam radi na način da prilikom svakog prolaženja kroz populaciju (podatke) mutira svako rješenje kandidata na način da ga miješa s ostalim rješenjima kandidata kako bi stvorio probnog kandidata. Nakon što je probni kandidat stvoren procjenjuje se njegova sposobnost. Ako probni kandidat bolje odgovara željenim postavkama onda se originalni kandidat odbacuje a probni kandidat zauzima njegovo mjesto. Kako bi se povećale šanse za pronalazak globalnog minimuma potrebno je koristiti što veću populaciju s više mutacija ali s manje rekombinacija. Ovim se proširuje skup podataka koji se koristi u optimizacije, odnosno proširuje se radijus pretraživanja ali se usporava konvergencija. Prema [8], diferencijalni evolucijski algoritam se u Python programskom jeziku definira pomoću naredbe:

1. `scipy.optimize.differential_evolution(func, bounds, args=(), strategy='best1bin', maxiter=None, popsize=15, tol= 0.01, mutation=(0.5, 1), recombination= 0.7, seed=None, callback=None, disp=False, polish=True, init='latinhypercube')`

Pojedini parametri opisani su u tablici 3.2.:

Tablica 3.2. Opis svih mogućih parametara za funkciju optimizacije diferencijalnim evolucijskim algoritmom u Python-u. [8]

Parametar	Tip	Opis
<b>func</b>	<i>callable</i>	Funkcija cilja koja se minimizira. Mora biti zapisana u formi $f(x, *args)$ , gdje je $x$ argument iz jednodimenzionalnog polja a $args$ je n-torka dodatnih fiksnih parametara koji su potrebni kako bi se definirala funkcija.
<b>bounds</b>	<i>sequence</i>	Granica za varijable. $(min, max)$ parovi za svaki element u $x$ , definira donju i gornju granicu za argumente funkcije cilja <i>func</i> . Uvjet $len(bounds) == len(x)$ mora biti ispunjen. <b><i>Len(bounds)</i></b> služi za određivanje broja parametara u $x$ .
<b>args</b>	<i>tuple</i> (nije obavezno)	Svaki dodatni fiksni parametar koji je potreban za definiranje funkcije cilja.
<b>strategy</b>	<i>str</i> (nije obavezno)	Strategija diferencijalnog evolucijskog algoritma koja se koristi. Sve su nabrojane u literaturi [8]. Zadana strategija je <b>'best1bin'</b> .
<b>maxiter</b>	<i>int</i> (nije obavezno)	Maksimalan broj generacije kroz koje je cijela populacija evoluirala. Maksimalan broj evaluacija funkcije definiran je: <b><math>(maxiter + 1) * popsize * len(x)</math></b>
<b>popsize</b>	<i>int</i> (nije obavezno)	Multiplikator za postavljanje ukupne veličine populacije. Populacija ima <b><math>popsize * len(x)</math></b> individua.
<b>tol</b>	<i>float</i> (nije obavezno)	Kada srednja vrijednost populacije pomnožena sa <b><i>tol</i></b> i podijeljena sa standardnom devijacijom populacije ima iznos veći od 1 tada se proces optimizacije prekida. <b><math>Convergence = mean(pop) * tol / stdev(pop) &gt; 1</math></b>
<b>mutation</b>	<i>float</i> <i>tuple(float, float)</i> (nije obavezno)	Konstanta mutacije. U nekoj literaturi se ova konstanta naziva i diferencijalna težina i označava sa <b>F</b> . Ovaj parametar bi trebao biti u rasponu [0, 2].
<b>recombination</b>	<i>float</i> (nije obavezno)	Konstanta rekombinacije koja bi trebala biti u rasponu [0,1]. U literaturi se često naziva vjerojatnost križanja i

		označava se sa <i>CR</i> . Povećanjem ove konstante omogućava se većem broju mutanata da prijeđu u iduću generaciju na štetu stabilnosti populacije.
<i>seed</i>	<i>int</i> (nije obavezno)	Ako parametar <i>seed</i> nije definiran, uzima se nasumičan broj. Parametar se koristi za ponovljive optimizacije.
<i>disp</i>	<i>bool</i> (nije obavezno)	Prikazivanje poruke o statusu.
<i>callback</i>	<i>callable</i> (nije obavezno)	Funkcija koja prati napredovanje minimizacije. Ako funkcija vrati stanje <i>True</i> minimizacija se zaustavlja.
<i>polish</i>	<i>bool</i> (nije obavezno)	Zadana vrijednost je <i>True</i> . Koristi se <i>scipy.optimize.minimize</i> s <i>L-BFGS-B</i> (Broyden-Fletcher-Goldfarb-Shannon) metodom kako bi se u konačnici dobili najbolji članovi populacije. Ovaj proces može poboljšati rezultate optimizacije.
<i>init</i>	<i>string</i> (nije obavezno)	Zadano je ' <i>latinhypercube</i> '. Ovaj parametar pokušava uzorkovanjem povećati pokrivenost dostupnih parametara.

Funkcija nakon optimizacije vraća nazad rezultat *res*. Rezultat je predstavljen kao objekt naziva *OptimizeResult*. Bitni atributi objekta su: *x* koji predstavlja jednodimenzionalno polje rješenja, *success* indikator koji govori da li je optimizacija uspješno završena i *message* koji opisuje razlog prekida optimizacije.

Najjednostavniji primjer pozivanja diferencijalnog evolucijskog algoritma u Python-u glasio bi:

```

1. from scipy.optimize import differential_evolution
2.
3. def func:
4.     .
5.     .
6.     .
7.     return
8.
9. granice = [(.,.), (.,.), (.,.), (.,.)]
10. rezultat = differential_evolution(func, granice)

```

Prvo se poziva i učitava paket „*differential\_evolution*“ koji omogućava implementaciju evolucijskog algoritma u Python skriptu. Zatim se definira funkcija koja se obrađuje, a nakon toga se zadaju granice funkcije. Rezultat optimizacije sprema se u varijablu pod nazivom „rezultat“.

## 4. MATEMATIČKI ZAPIS PROBLEMA I RJEŠENJE OPTIMIZACIJE DISTRIBUTIVNE MREŽE

### 4.1. Matematički zapis problema

Prema [16] izraz (4-1) opisuje funkciju cilja koja za cilj ima minimizirati gubitke djelatne snage:

$$O = \sum_{i,j \in k} (P_i - P_j) = \sum_{i=1}^n \sum_{j=1}^n \left( \frac{R_{ij} \cos(\delta_i - \delta_j)}{V_i V_j} (P_i P_j + Q_i Q_j) + \frac{R_{ij} \sin(\delta_i - \delta_j)}{V_i V_j} (Q_i P_j - P_i Q_j) \right) \quad (4-1)$$

gdje je  $n$  broj čvorova,  $R_{ij}$  je otpor voda između čvorova „ $i$ “ i „ $j$ “,  $P_i$  i  $Q_i$  su radna i jalova snaga injektirane u čvor „ $i$ “, a  $P_j$  i  $Q_j$  su radna i jalova snaga injektirane u čvor „ $j$ “.  $V_i$  i  $\delta_i$  su razina napona i fazni kut na čvoru „ $i$ “, dok su  $V_j$  i  $\delta_j$  razina napona i fazni kut na čvoru „ $j$ “.

Prema [17], optimizacijski problem je riješen uz neka ograničenja koja su prikazana sljedećim izrazima.

Kako bi se ostvarila ravnoteža opterećenja na svakoj sabirnici radi zadovoljavanja potražnje i opskrbe potrebno je zadovoljiti izraz (4-2):

$$P_K + \sum_{i=1}^n P_{DG_i} = \sum_{i=1}^n P_{D_i} + P_L \quad (4-2)$$

gdje je  $P_K$  snaga kompenzacije,  $P_{DG_i}$  je snaga distribuiranog izvora injektirana u čvor „ $i$ “,  $P_{D_i}$  je snaga distribucijske mreže injektirana u čvor „ $i$ “ dok je  $P_L$  snaga trošila odnosno snaga opterećenja. Funkcija cilja se računa proračunom tokova snaga koji radi program OpenDSS stoga je ograničenje (4-2) automatski zadovoljeno te ga nije potrebno dodatno provjeravati u optimizacijskom problemu.

Kako bi se osigurale naponske razine propisane normom EN 50160 potrebno je zadati minimalne i maksimalne naponske granice kao što je prikazano izrazom (4-3):

$$|U_i|^{min} \leq |U_i| \leq |U_i|^{max} \quad (4-3)$$

gdje je minimalni napon  $|U_i|^{min} = 0.9 p. u.$ , a maksimalni napon  $|U_i|^{max} = 1.1 p. u.$

Kako bi se odredila veličina distribuiranih izvora treba postojati i niz raspoloživih veličina pa se stoga zadaju granice djelatne ( $P$ ) i jalove ( $Q$ ) snage:

$$Q_{DG_i}^{min} \leq Q_{DG_i} \leq Q_{DG_i}^{max} \quad (4-4)$$

$$P_{DG_i}^{min} \leq P_{DG_i} \leq P_{DG_i}^{max} \quad (4-5)$$

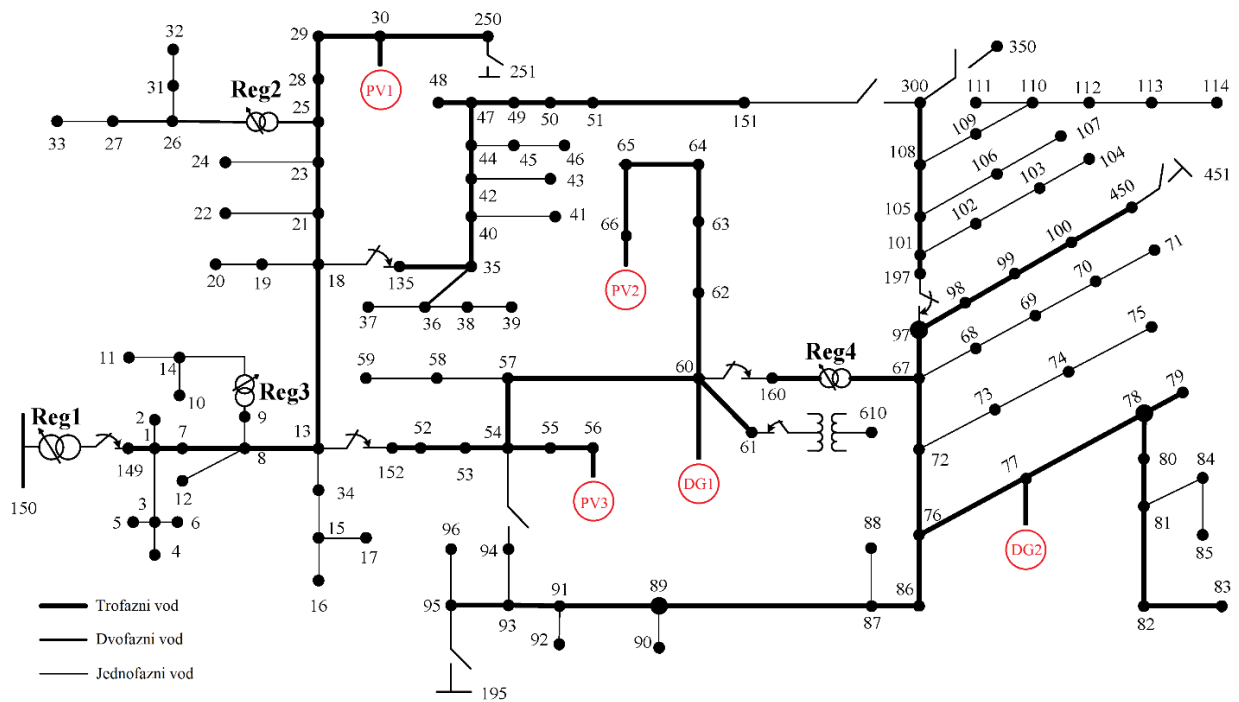
Gdje su  $Q_{DG_i}^{min}$  i  $P_{DG_i}^{min}$  minimalna jalova i djelatna snaga distribuiranog izvora dok su  $Q_{DG_i}^{max}$  i  $P_{DG_i}^{max}$  maksimalna reaktivna i radna snaga distribuiranog izvora.

## 4.2. Definiranje distribuiranih izvora u OpenDSS-u

Simulacija se izvodila na testnoj distributivnoj mreži IEEE123Bus. Mreža se sastoji od 123 čvora te je morala biti modificirana na način da su dodana tri neupravljiva distribuirana izvora koji su u ovom slučaju tri fotonaponska sustava te su dodana 2 upravljiva distribuirana sustava koji su u ovom slučaju dva bio-plinska postrojenja. Dodani distribuirani izvori označeni su crveno bojom na slici 4.1. Tablicom 4.1. prikazane su karakteristike korištenih distribuiranih izvora.

Tablica 4.1. Popis distribuiranih izvora spojenih na mrežu i njihove karakteristike.

Tip izvora	Naziv izvora	Lokacija (čvor u mreži)	Broj faza	P [kW]	Q [kvar]
Neupravljivi distribuirani izvor	PV1	30	3	200	0
	PV2	66	3	200	0
	PV3	56	3	200	0
Upravljivi distribuirani izvor	DG1	60	3	1745	480
	DG2	77	3	1745	480



Slika 4.1. IEEE123Bus testna mreža sa dodanim distribuiranim izvorima.

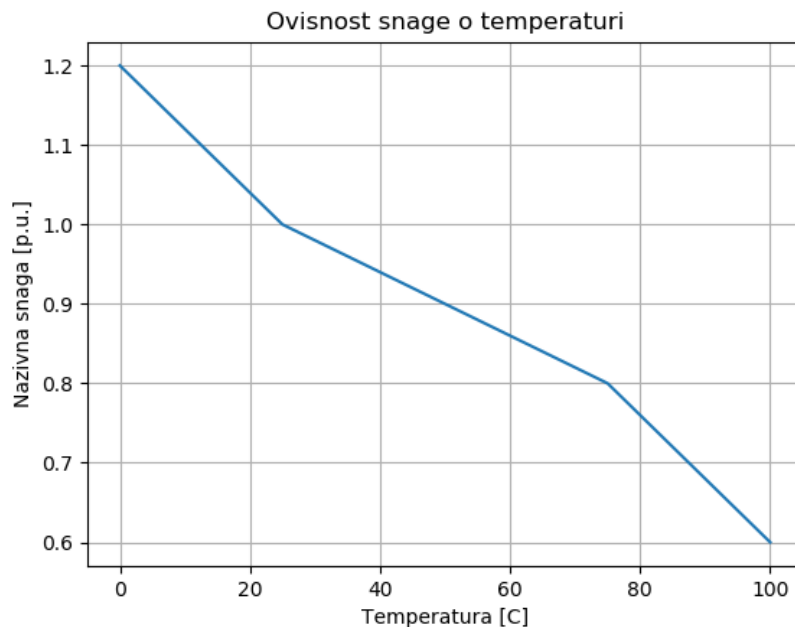
Kako bi se ovi izvori dodali bilo je potrebno modificirati OpenDSS kod:

```

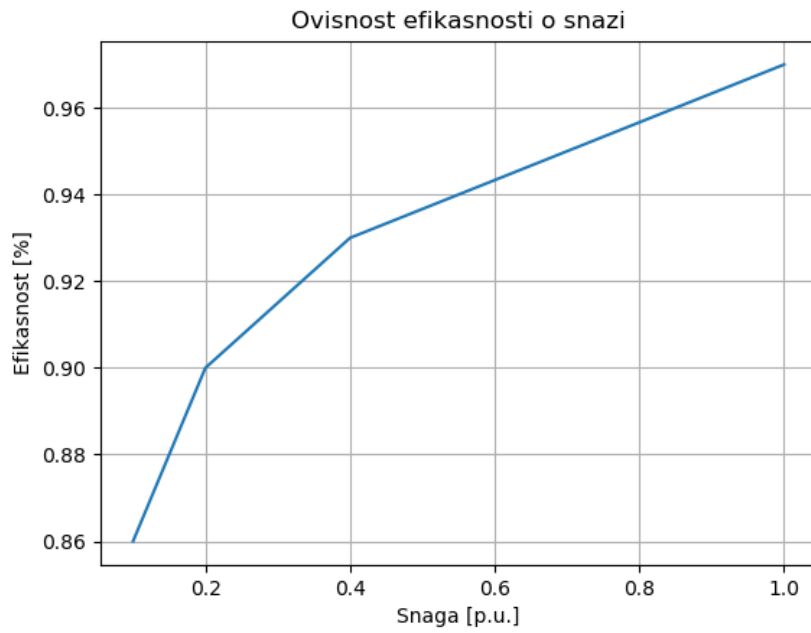
1. //PV krivulje
2. New XYcurve.MyPvsT npts=4 xarray=[0 25 75 100] yarray=[1.2 1 .8 .6]
3. New XYcurve.MyEff npts=4 xarray=[.1 .2 .4 1] yarray=[.86 .9 .93 .97]
4. New LoadShape.zracenje npts=24 interval=1
5. ~ mult=[0 0 0 0 0 0 .1 .2 .3 .5 .8 .9 1.0 1.0 .99 .9 .7 .4 .1 0 0 0 0 0]
6. New Tshape.temperature npts=24 interval=1
7. ~ temp=[15 15 15 15 15 15 15 15 20 25 27 30 32 35 35 30 27 25 22 20 18 15 15 15]
8. //PV sustavi
9. New PVSystem.PV1 phases=3 bus1=30 kv=4.16 irrads=0.98 pmp=200 temperature=25 pf=1
10. ~ %cutin=0.1 %cutout=0.1 effcurve=MyEff P-
    tCurve=MyPvst Daily=zracenje Tdaily=temperature
11. New PVSystem.PV2 phases=3 bus1=66 kv=4.16 irrads=0.98 pmp=200 temperature=25 pf=1
12. ~ %cutin=0.1 %cutout=0.1 effcurve=MyEff P-
    tCurve=MyPvst Daily=zracenje Tdaily=temperature
13. New PVSystem.PV3 phases=3 bus1=56 kv=4.16 irrads=0.98 pmp=200 temperature=25 pf=1
14. ~ %cutin=0.1 %cutout=0.1 effcurve=MyEff P-
    tCurve=MyPvst Daily=zracenje Tdaily=temperature
15. //Distribuirani izvori
16. New Generator.DG1 phases=3 bus1=60 kv=4.16 kw=150 kvar=15
17. New Generator.DG2 phases=3 bus1=77 kv=4.16 kw=150 kvar=15

```

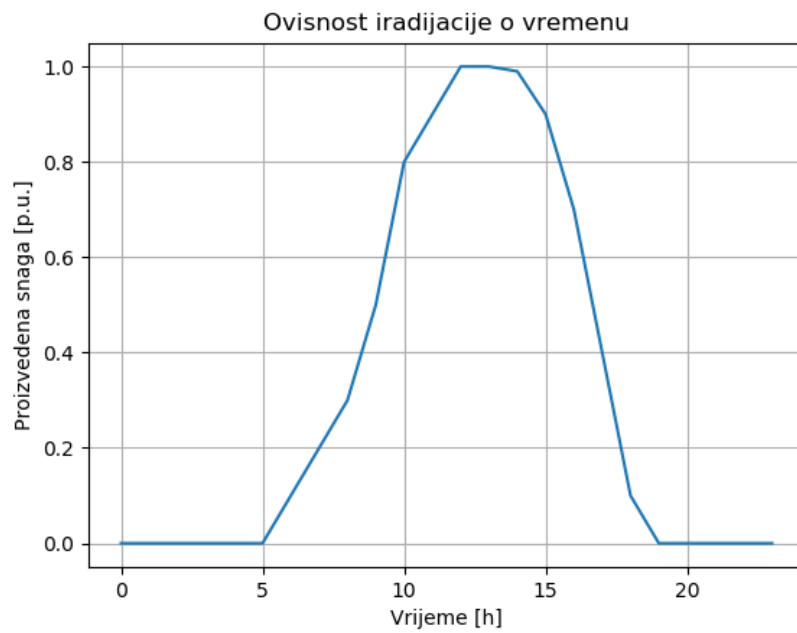
Prilikom zadavanja parametara fotonaponskog sustava potrebno je definirati kako će se fotonaponski sustav ponašati s obzirom na vremenske uvjete, pa je tako u kodu zadana krivulja ovisnosti proizvedene snage o temperaturi koja je prikazana na slici 4.2. te je u kodu označena kao *XYcurve.MyPvsT*. Zatim se zadaje krivulja koja opisuje ovisnost efikasnosti o proizvedenoj snazi koja je prikazana na slici 4.3. i označena u kodu kao *XYcurve.MyEff*. Potrebno je zadati i krivulju koja opisuje proizvodnju, odnosno iradijaciju, u odnosu na vrijeme u danu, kao što je prikazano na slici 4.4. Podaci su zadani u p.u. sustavu (engl. per-unit system) zbog jednostavnosti koda. U slučaju da se dva fotonaponska sustava nalaze na istom geografskom području a različitih su snaga, OpenDSS će definirati vrijednost 1 u p.u. kao maksimalnu snagu pojedinog fotonaponskog sustava. Na kraju je potrebno definirati dnevne temperature, odnosno ovisnost temperature o vremenu kako je prikazano na slici 4.5.



Slika 4.2. Ovisnost nazivne snage fotonaponskog sustava temperaturi ćelije.

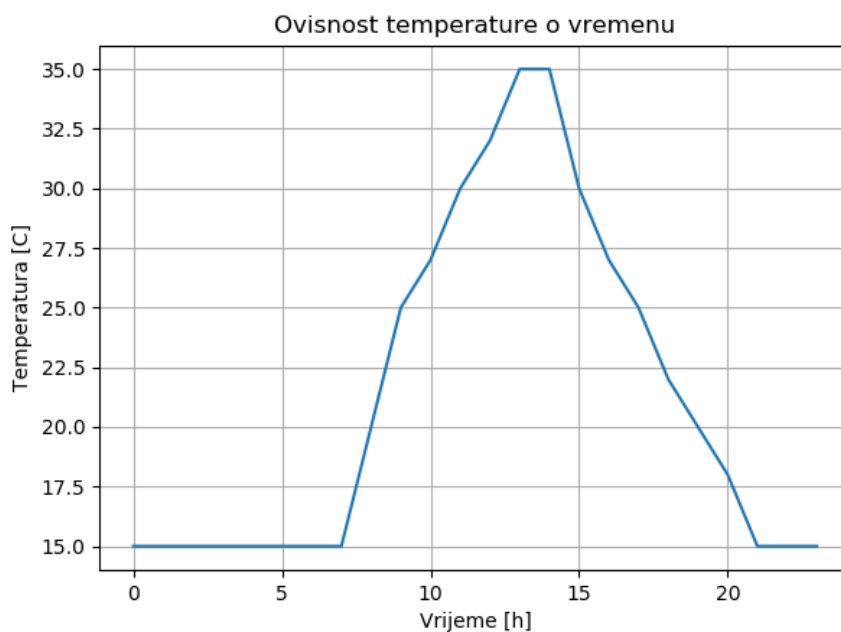


Slika 4.3. Ovisnost efikasnosti fotonaponskog sustava o proizvedenoj snazi.



Slika 4.4. Ovisnost iradijacije (proizvodnje snage) o vremenu.

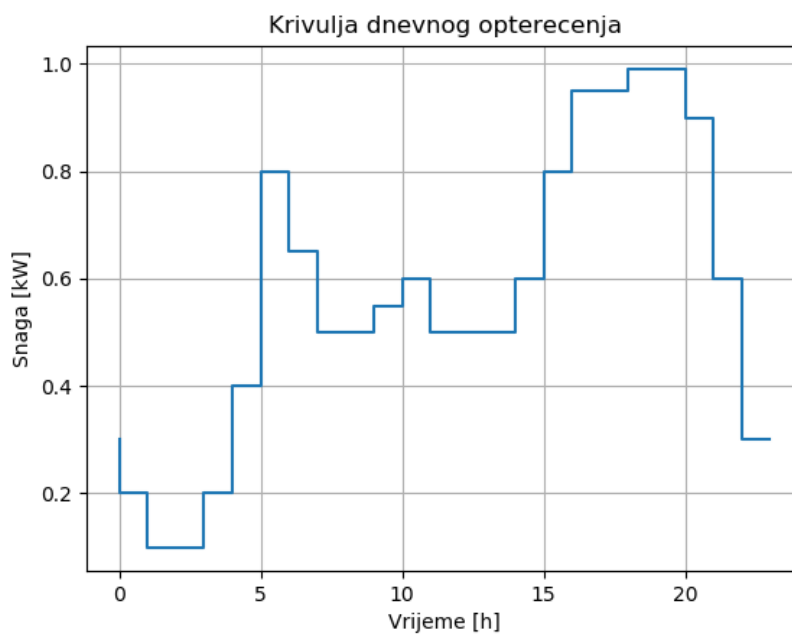




Slika 4.5. Ovisnost dnevne temperature o vremenu.

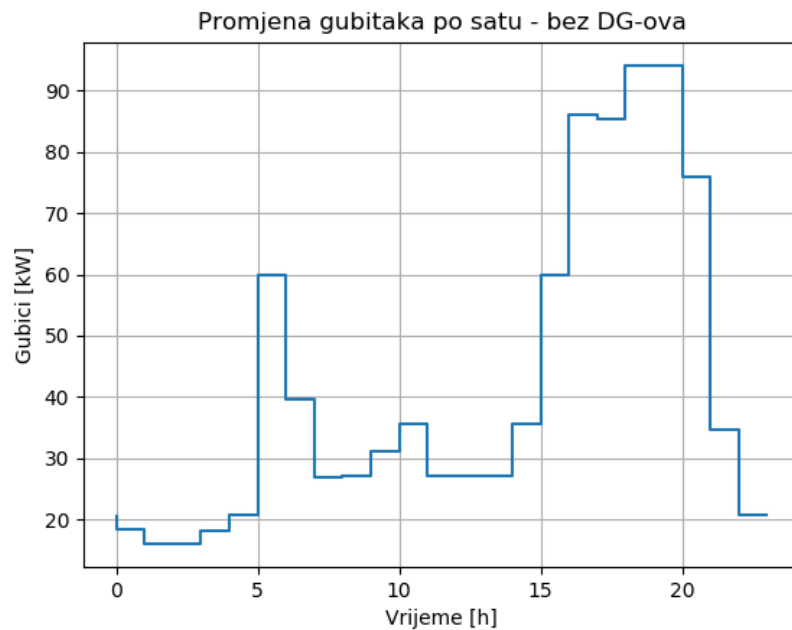
### 4.3. Rezultati simulacije mreže i analiza rezultata

Na slici 4.6. prikazana je krivulja dnevnog opterećenja koja opisuje ponašanje trošila spojenih u mrežu. Prilogom II. definirani su iznosi snaga trošila, vrsta spoja i mjesta priključenja. Kako je opisano u potpoglavlju 3.2.2. potrebno je zadati parametar „daily“ koji definira izgled krivulje opterećenja.

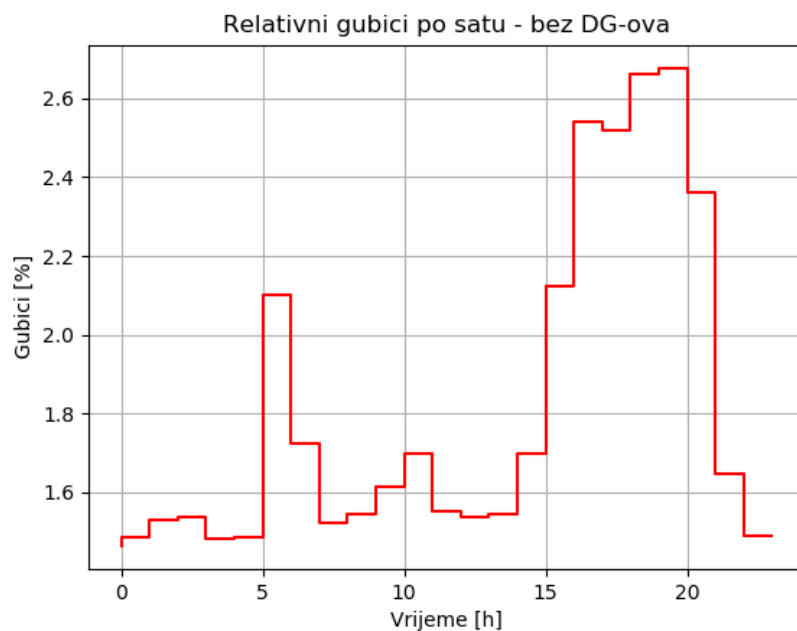


Slika 4.6. Krivulja opterećenja koja opisuje trošila spojena na mrežu.

Simulacija mreže prvo je odrađena za slučaj kada na nju nije spojen ni jedan distribuirani izvor kako bi se utvrdilo početno stanje mreže.

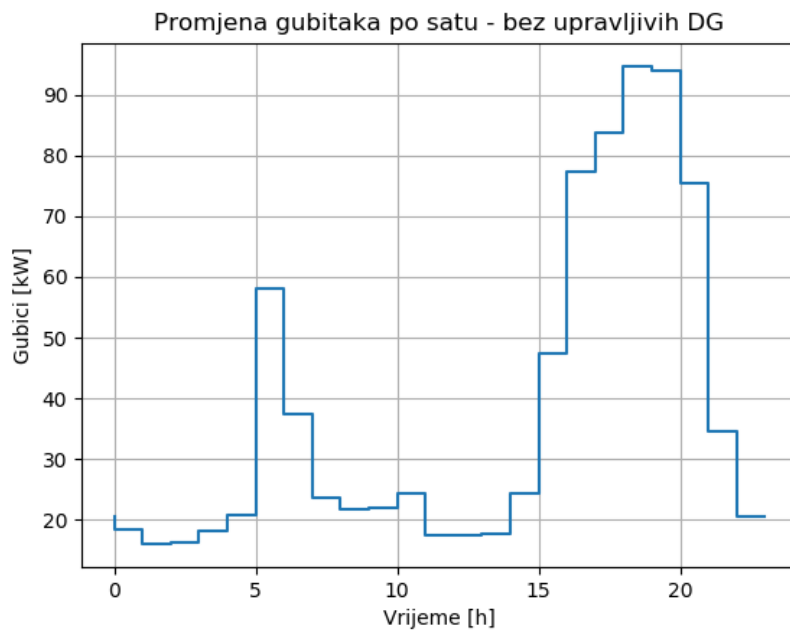


Slika 4.7. Promjena gubitaka po satu za slučaj kada distribuirani izvori nisu spojeni na mrežu.

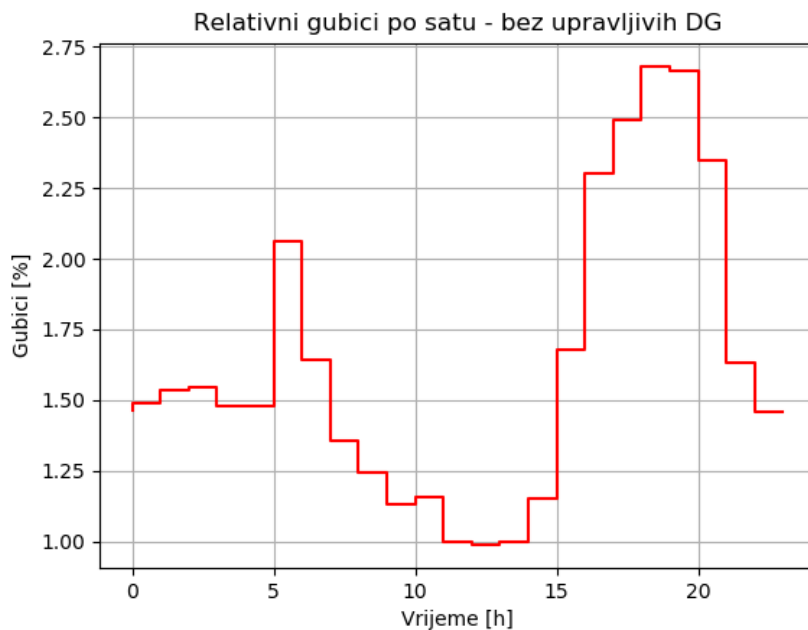


Slika 4.8. Relativni gubici po satu za slučaj kada distribuirani izvori nisu spojeni na mrežu.

Simulacija mreže zatim je odrađena s promjenjivim opterećenjem koje je zadano slikom 4.6 i fotonaponskim distribuiranim izvorima spojenim na mrežu bez upravljivih distribuiranih izvora (vrijednost djelatne snage je nula).

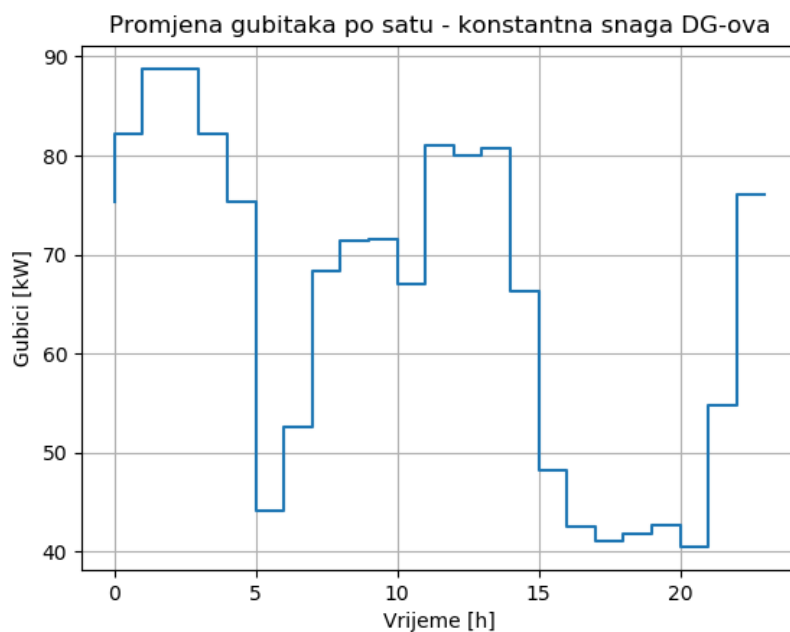


Slika 4.9. Promjena gubitaka po satu za slučaj kada upravljivi distribuirani izvori nisu spojeni na mrežu.



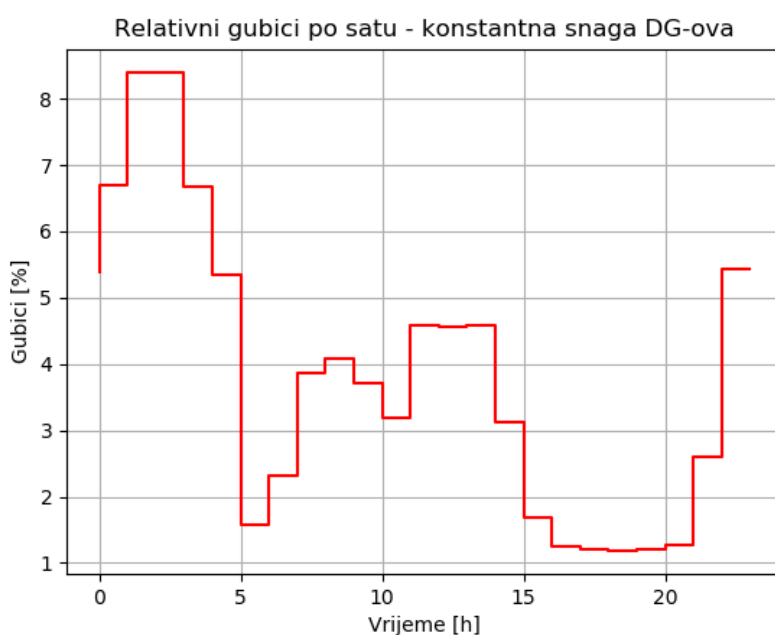
Slika 4.10. Relativni gubici po satu za slučaj kada upravljivi distribuirani izvori nisu spojeni na mrežu.

Zatim se simulirala mreža s promjenjivim opterećenjem, fotonaponskim distribuiranim izvorima te upravljivim distribuiranim izvorima sa konstantnom snagom. Slika 4.11. prikazuje promjenu gubitaka djelatne snage po satu za konstantnu snagu distribuiranih izvora koja se kreće od 40,6 do 88,7 kW.



Slika 4.11. Promjena gubitaka po satu za konstantne snage upravljivih distribuiranih izvora.

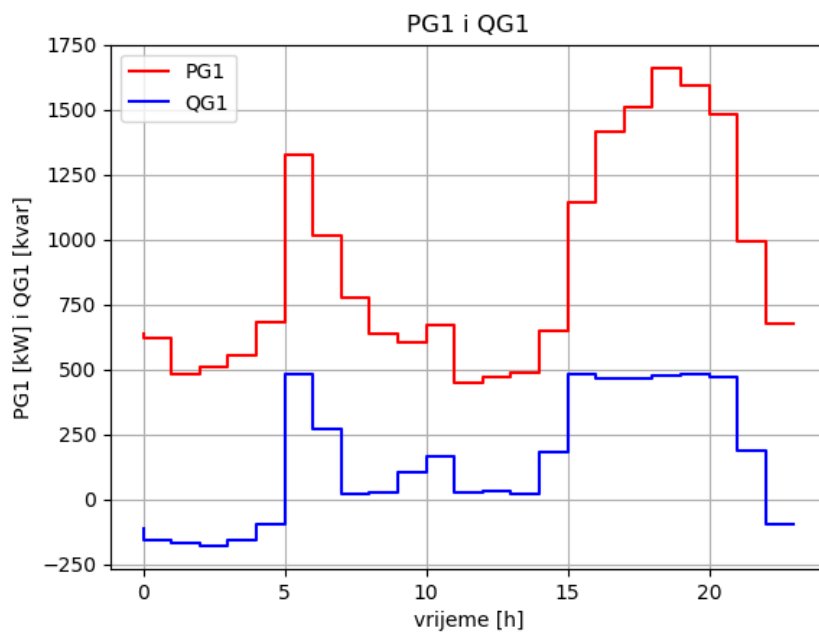
Slika 4.12. prikazuje relativne gubitke po satu za konstantne snage distribuiranih izvora koji iznose od 1,2 do 8,4 %.



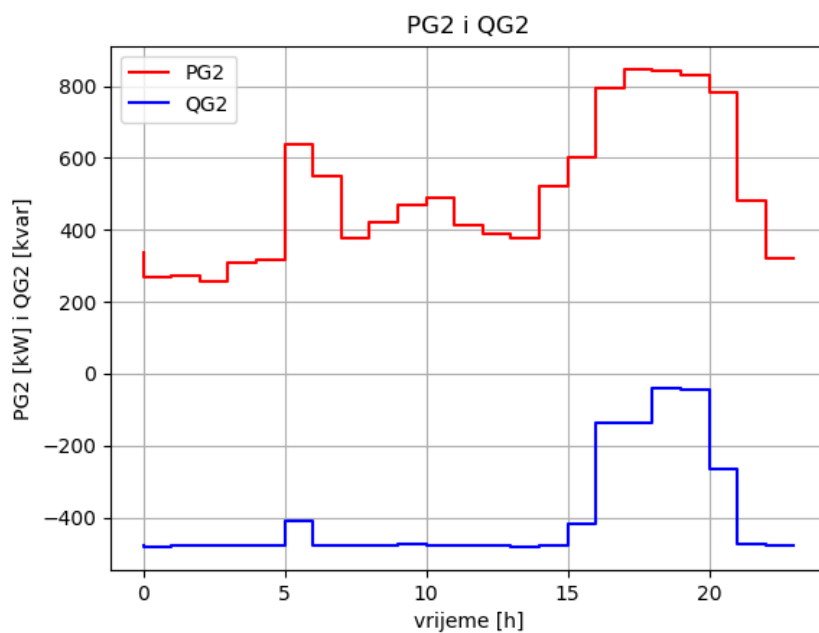
Slika 4.12. Relativni gubici po satu za konstantne snage upravljivih distribuiranih izvora.

Optimizacija za upravljanje distribuiranih izvora odrađena je pomoću Python koda (Prilog II.). Slika 4.13. prikazuje djelatnu i jalovu snagu prvog distribuiranog izvora nakon optimizacije. Na grafu je vidljivo kako snaga izvora prati oblik krivulje dnevnog opterećenja (Slika 4.6) što je jedan

od znakova da je optimizacija uspješno obavljena. Slika 4.14. prikazuje snage drugog distribuiranog izvora nakon optimizacije.

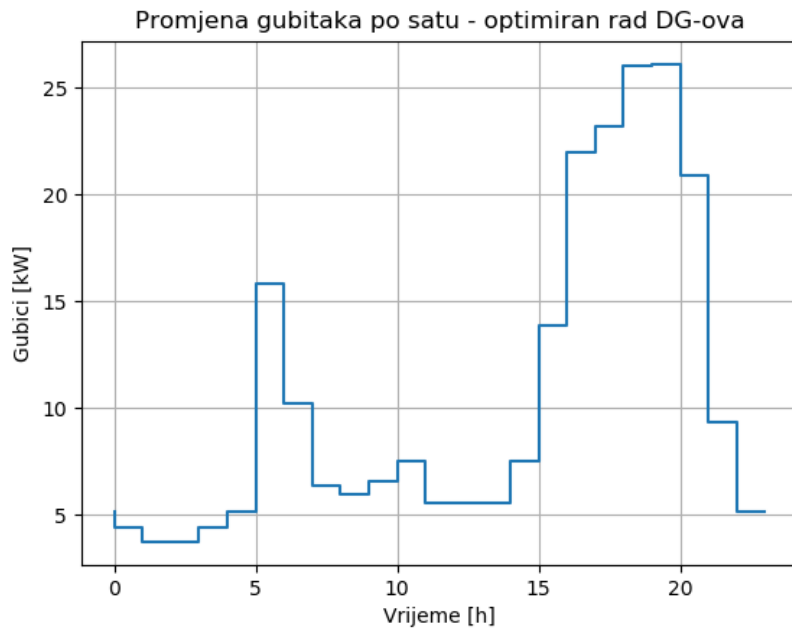


Slika 4.13. Djelatna i jalova snaga prvog upravljivog distribuiranog izvora.



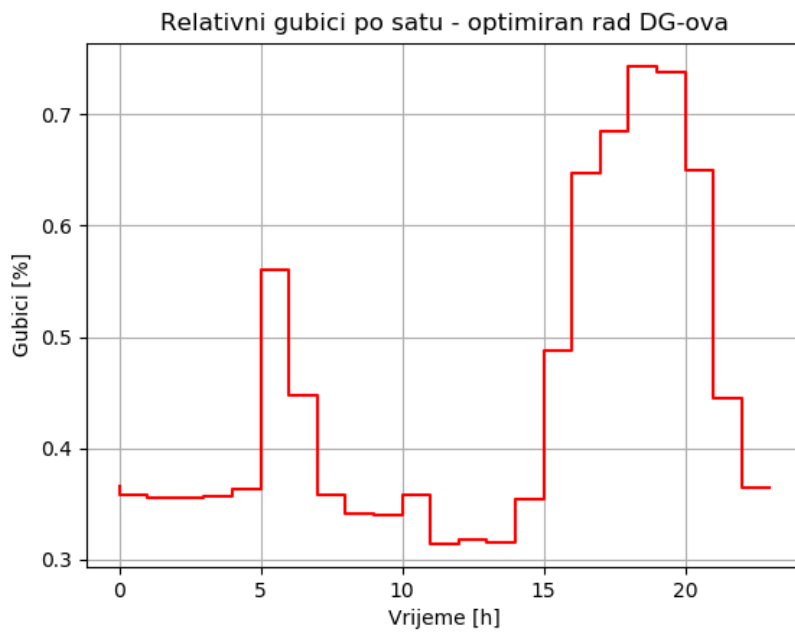
Slika 4.14. Djelatna i jalova snaga drugog upravljivog distribuiranog izvora.

Slika 4.15. prikazuje promjenu gubitaka djelatne snage po satu za optimirani rad upravljivih distribuiranih izvora koji iznose od 3,7 do 26,1 kW.

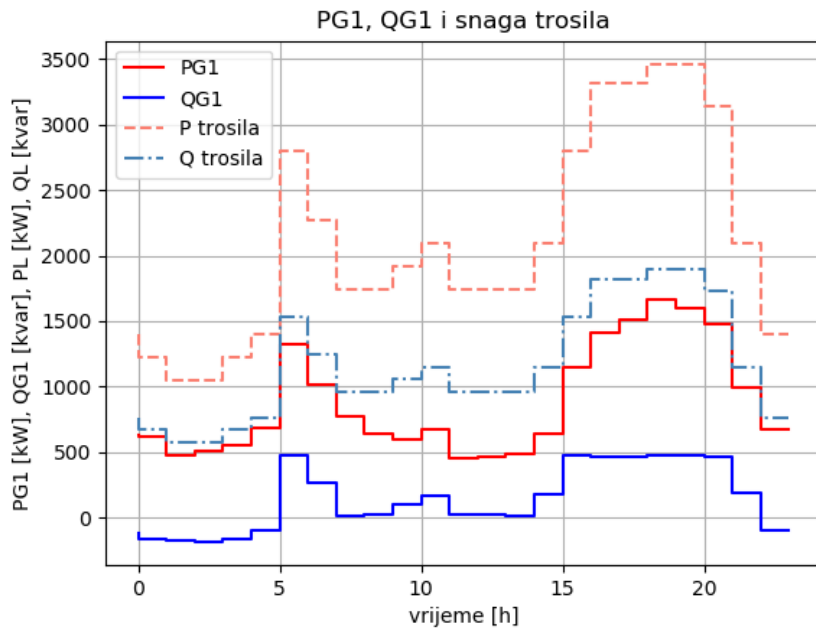


Slika 4.15. Promjena gubitaka po satu za optimirani rad upravljivih distribuiranih izvora.

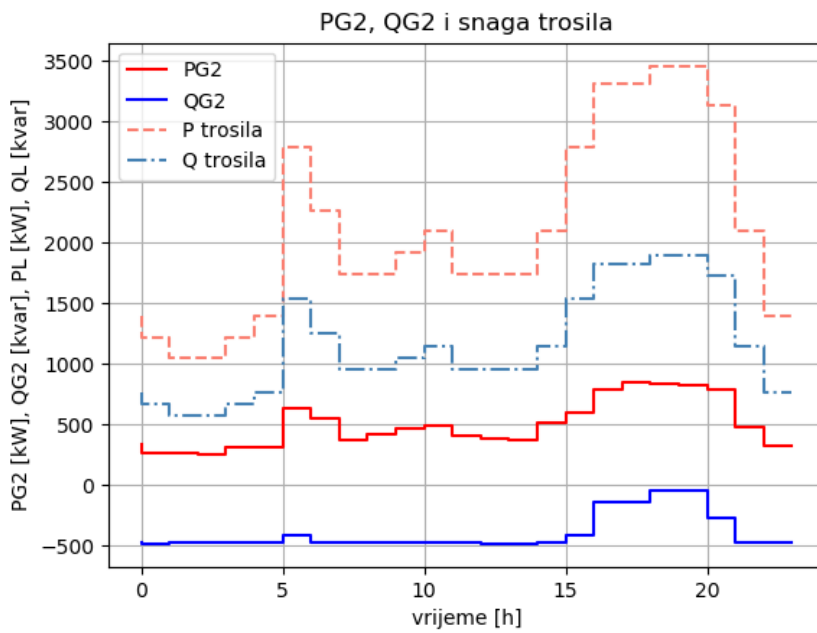
Slika 4.16. prikazuje relativnu promjenu gubitaka po satu za optimirani rad upravljivih distribuiranih izvora koji iznose od 0,31 do 0,74 %.



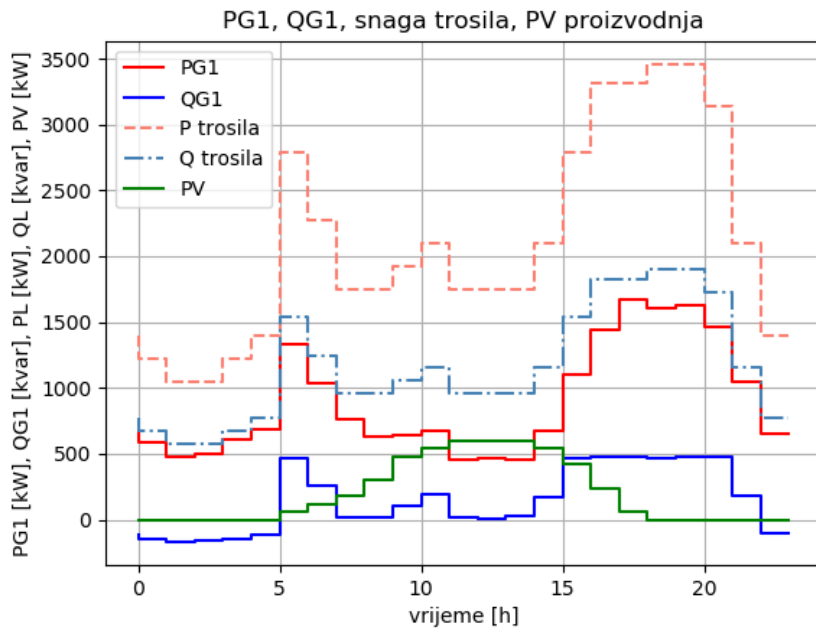
Slika 4.16. Relativni gubici po satu za optimirani rad upravljivih distribuiranih izvora.



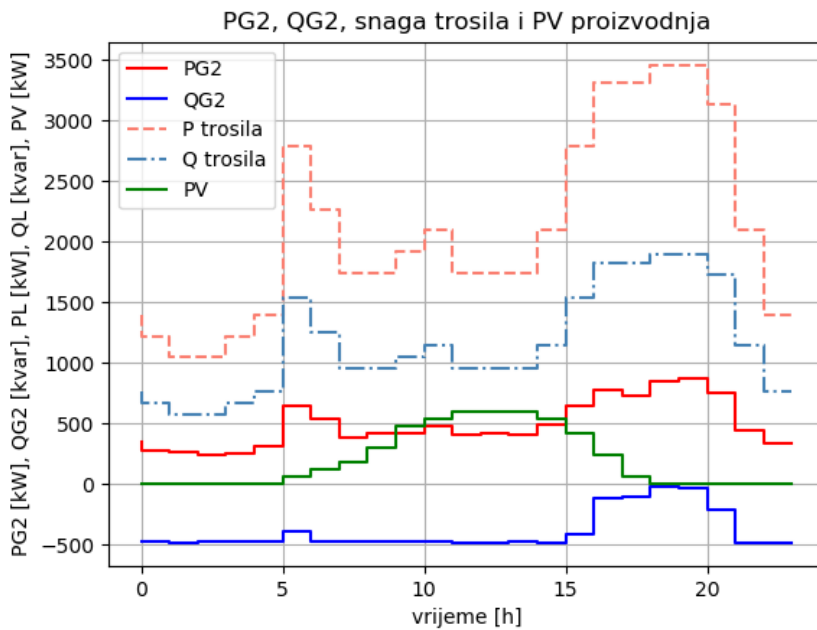
Slika 4.17. Djelatne i jalove snage prvog distribuiranog izvora i trošila nakon optimizacije.



Slika 4.18. Djelatne i jalove snage drugog distribuiranog izvora i trošila nakon optimizacije.



Slika 4.19. Djelatne i jalove snage prvog distribuiranog izvora i trošila te profil proizvodnje fotonaponskih elektrana.



Slika 4.20. Djelatne i jalove snage drugog distribuiranog izvora i trošila te profil proizvodnje fotonaponskih elektrana.

Kada se na slike 4.13. i 4.14. doda profil proizvodnje svih fotonaponskih elektrana (oznaka PV na grafu) spojenih na mrežu dobiju se slike 4.19. i 4.20. Na tim slikama se može vidjeti kako upravljivi distribuirani izvori pokrivaju proizvodnju u slučaju kada nema sunčevog zračenja.



Odnosno, kada je proizvodnja fotonaponskih elektrana smanjena povećava se proizvodnja na upravljivim distribuiranim izvorima kako bi se opskrblila trošila spojena na mrežu uz minimalne gubitke djelatne snage.

Tablica 4.2. prikazuje gubitke za različite slučajeve simulacije. U prvom redu prikazani su gubici kada su na mrežu spojeni samo neupravljivi distribuirani izvori, dok su u drugom redu tablice prikazani gubici za slučaj kada su na mrežu spojeni i upravljivi (konstantne snage) i neupravljivi distribuirani izvori. Treći red prikazuje rezultate, odnosno gubitke nakon optimizacije.

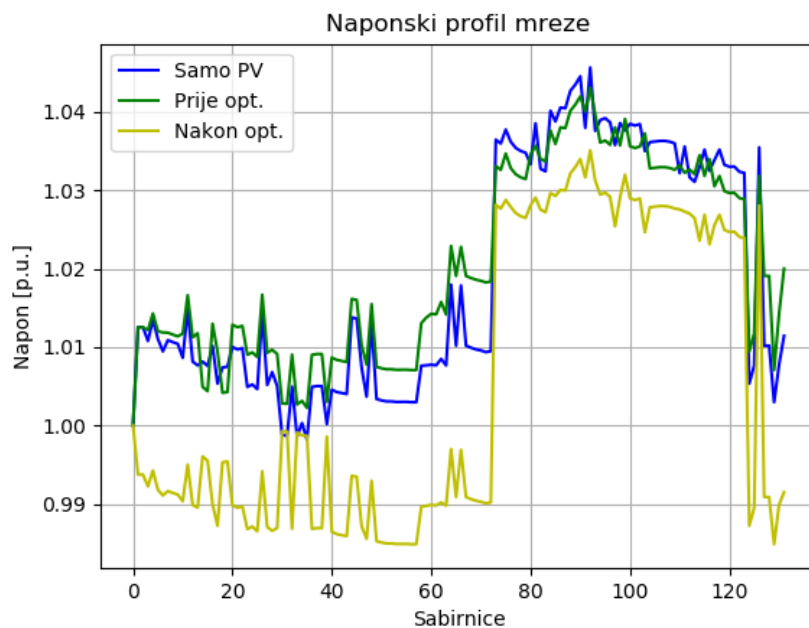
*Tablica 4.2. Tablični prikaz gubitaka za različite slučajeve simulacija.*

	Gubici snage				Dnevni gubici energije	
	[kW]		[%]		[kWh]	[%]
	Min.	Max.	Min.	Max.		
Mreža bez distribuiranih izvora	16,1	94,0	1,5	2,7	998,1	2,0
Neupravljivi distribuirani izvori	16,2	94,6	1	2,7	902,4	1,8
Neupravljivi distribuirani izvori i konstantna snaga upravljivih izvora	40,6	88,7	1,2	8,4	1563,7	3,1
Nakon optimizacije	3,7	26,1	0,31	0,74	249,5	0,5

Iz rezultata (Tablica 4.2) može se vidjeti kako u slučaju kada su na mrežu spojeni samo neupravljivi distribuirani izvori dnevni gubici energije iznose 902,4 kWh (satni gubici energije kreću se od 16,2 do 94,6 kWh tokom dana). Nakon priključenja upravljivih distribuiranih izvora

na mrežu dnevni gubici energije se povećavaju sa 902,4 na 1563,7 kWh za slučaj kada distribuirani izvori imaju konstantnu djelatnu snagu. Ovo je čest slučaj u praksi jer investitori drže snagu konstantnom odnosno maksimalnom jer žele da se njihova investicija što prije isplati. Kako to nije pogodno za mrežu i distributere jer se gubici u mreži povećavaju izvršena je optimizacija upravljivih distribuiranih izvora te su se gubici energije smanjili na 249,5 kWh, odnosno smanjili su se sa 1,2 do 8,4% na 0,31 do 0,74%.

Prilikom simuliranja svih slučajeva snimljeni su i naponi na svim sabirnicama (Slika 4.21). Na slici je vidljivo kako u slučaju kada se na mrežu na kojoj su priključeni neupravljivi distribuirani izvori priključe upravljivi ali s konstantnom djelatnom snagom napon se na nekim sabirnicama povećava dok je na nekima malo niži nego u slučaju kada su priključeni samo neupravljivi distribuirani izvori.



Slika 4.21. Naponski profili mreže za tri simulirana slučaja.

Slika 4.21. također prikazuje kako se naponi spuštaju nakon optimizacije na svim sabirnicama te se samim time poboljšava naponski profil mreže. Tablica 4.3. prikazuje vrijednosti minimalnih i maksimalnih vrijednosti napona u p.u. za sva tri simulirana slučaja. Iz tablice se vidi kako se nakon optimizacije naponsko odstupanje kreće oko baznog napon od 1 p.u.

Tablica 4.3. Tablične vrijednosti minimalnih i maksimalnih napona u mreži za tri simulirana slučaja.

	Napon [p.u.]		Odstupanje napona od 1 p.u. [p.u.]	
	Min.	Max.	Min.	Max.
Mreža bez distribuiranih izvora	0,9971	1,0453	-0,0029	+0,0453
Neupravljivi distribuirani izvori	0,9982	1,0456	-0,0018	+0,0456
Neupravljivi distribuirani izvori i konstantna snaga upravljivih izvora	1,0000	1,0431	0,0000	+0,0431
Nakon optimizacije	0,9849	1,0351	-0,0151	+0,0351

## 5. ZAKLJUČAK

Distribuirani izvori koji kao pogonsko gorivo koriste sunčevo zračenje ili vjetar nemaju mogućnosti skladištenja goriva. Gorivo je izgubljeno u slučaju ako je ovakav izvor isključen za vrijeme kada je gorivo dostupno. Kako su ova pogonska goriva nepredvidljiva, ekonomičnost i isplativost takvog izvora postaje upitna. S druge strane distribuirani izvori koji kao pogonsko gorivo koriste bioplin ili biomasu imaju mogućnost skladištenje goriva. Isključenje izvora ili smanjenjem snage nadoknađuje se onda kada bude potrebe za povećanom potrošnjom. Investitori zbog financijskih razloga žele da distribuirani izvori proizvode što više i samo djelatnu snagu, a to nije uvijek i tehnički najbolje rješenje s obzirom na strujno-naponske i energetske prilike u mreži. Glavni uzrok tome je vremenska promjenjivost i opterećenja i intenziteta promjenjivih energenata distribuiranih izvora. Tehnologije distribuirane proizvodnje električne energije promijenile su mnoge aspekte rada, dizajna i implementacije distribucijskih sustava. Povećanjem broja decentraliziranih sustava s manjim proizvodnim jedinicama koje su izravno povezane na distribucijsku mrežu u blizini grupe trošila distribucijske tvrtke smanjuju gubitke u svojim mrežama.

Cilj ovog rada bio je korištenje evolucijskog algoritma za optimizaciju dnevne proizvodnje distribuiranih izvora spojenih na distribucijsku mrežu. Simulacija sustava je odrađena prvo za slučaj kada su na mrežu spojeni samo neupravljivi distribuirani izvori. Zatim se simulacija radila za slučaj kada se dodaju upravljivi distribuirani izvori ali imaju konstantnu djelatnu snagu. Na kraju se radila optimizacija sustava kojom su se gubici snage smanjili na minimalnu vrijednost od 0,31% i maksimalnu vrijednost od 0,74%. Dok se nije obavila optimizacija snaga gubitaka iznosili su 1,2% do 8,4%. Iz rezultata dobivenih simulacijom može se vidjeti kako su se gubici energije uvelike smanjili te su se smanjili na dnevnoj razini sa 3,1% na 0,5%.

Također prilikom analize naponskog profila mreže može se vidjeti kako su se nakon optimizacije naponi na svim sabirnicama smanjili te se naponsko odstupanje kreće oko baznog napon od 1 p.u. te iznosi  $-0,0151/+0,0351$  p.u.. Ovo se postiglo optimalnim upravljanjem snage upravljivih distribuiranih izvora.

## LITERATURA

- [1] W. El-Khattam, M. M. A. Salama, *Distributed generation technologies, definitions and benefits*, Electric Power System Research, br. 71, str. 119-128, 2004.
- [2] T. Ackermann, G. Andersson, L. Soder, *Distributed Generation: a definition*, Electric Power System Research, br. 57, str. 195-204, 2001.
- [3] L. I. Dulau, M. Abrudean, D. Bica, *Distributed generation technologies and optimization*, Procedia Technology, br. 12, str. 687-692, 2014.
- [4] N. Dizdarević, M. Majstorović, S. Žutobradić, *Distribuirana proizvodnja električne energije*, Energetski institut 'Hrvoje Požar', Zagreb, 2003.
- [5] T. Weise, *Global Optimization Algorithms – Theory and Application*,  
(url: <http://www.it-weise.de/projects/book.pdf>, pristup: 20.11.2017.)
- [6] R. C. Dugan, *The Open Distribution System Simulator (OpenDSS) – Reference Guide*, Electric Power Research Institute, 2016.  
(url: <https://sourceforge.net/projects/electricdss/files/OpenDSS/OpenDSSManual.pdf>, pristup: 20.1.2017.)
- [7] Službena stranica programskog jezika Python  
(url: <https://www.python.org/about/> , pristup: 06.04.2018.)
- [8] Službena stranica SciPy.org – *Differential Evolution*,  
(url: [https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential\\_evolution.html#scipy.optimize.differential\\_evolution](https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential_evolution.html#scipy.optimize.differential_evolution) , pristup: 06.04.2018.)
- [9] D. Grundler, *Evolucijski algoritmi (I) – Pobude i načela*, Sveučilište u Zagrebu, Tekstilno tehnološki fakultet, Zagreb, 2001.  
(url: <https://hrcak.srce.hr/file/8743> , pristup: 06.04.2018.)
- [10] T. Black, U. Hammel, H. P. Schwefel, *Evolutionary Computation: Comments on the History and Current State*, IEEE Trans. on Evolutionary Computation, Vol. 1. No. 1, pp. 3-17, 1997.

- [11] Ener Twin, *The EnerTwin micro turbine: reliable technology*  
(url: <http://www.enertwin.com/enertwin-en/the-micro-turbine-technology> , pristup: 06.04.2018.)
- [12] Energy Design Resources, *Fule Cells Fuel Alternative Energy Options*, e-News No. 90., 2013.  
(url: <https://energydesignresources.com/resources/e-news/e-news-90-fuel-cells.aspx> , pristup: 06.04.2018.)
- [13] Mitsubishi Electrical, *Frequently Asked Questions about Solar Power*  
(url: <http://www.mitsubishielectric.com/bu/solar/faq/index.html> , pristup: 06.04.2018.)
- [14] Tianjin Anson International Co. Ltd. (TICO), *Things You Don't Know About Wind Turbines*  
(url: <http://www.industrycrane.com/blog/How-wind-turbines-work.html> , pristup: 06.04.2018.)
- [15] M. Barukčić, Ž. Hederić, K. Miklošević, *Multi Objective Optimization of Energiy Production of Distributed Generation in Distribution Feeder*, ENERGYCON 2014, str.13-16, Dubrovnik, Hrvatska, 2014.
- [16] I. Pisica, C. Bulac, M. Eremia, *Optimal distributed Generation Location and Sizing using Genetic Algorithms*, ISAP 2009, IEEE, Curitiba, Brazil, 2009.
- [17] H. Manafi, N. Ghadimi, M. Ojaroudi, P. Farhadi, *Optimal Placement of Distributed Generations in Radial Distribution System Using Various PSO and DE Algorithms*, Elektronika IR Elektrotehnika, vol. 19, br. 10., str. 53-57, 2013.
- [18] A. Schweer, J. Tzschope, A. Sorensen i ostali, *CIGRE – Impact of Increasing Contribution of Dispersed Generation on the Power System*, WG 37.23, 1999,  
(url: <https://e-cigre.org/publication/137-impact-of-increasing-contribution-of-dispersed-generation-on-the-power-system>, pristup: 25.05.2018)
- [19] N. Hatziargyrio, M. Donnelly, S. Papathanassiou i ostali, *CIGRE – Modeling New Forms of Generatrion and Storage*, TF 38.01.10, 2010,

(url: <http://fglongatt.org/OLD/Archivos/Archivos/SistGD/CIGRE-TF-380110.pdf>,  
pristup: 25.05.2018.)

[20] International Energy Agency (IRENA) – Renewable Energy Statistics 2017.

(url: [www.irena.org/Publications](http://www.irena.org/Publications) , pristup: 08.06.2018.)

## POPIS SLIKA

Slika 2.1. Princip rada mikro turbine. [11].....	6
Slika 2.2. Princip rada gorivne ćelije. [12].....	7
Slika 2.3. Princip rada fotonaponske ćelije. [13] .....	8
Slika 2.4. Princip rada vjetrogeneratora. [14] .....	9
Slika 2.5. Propad napona na mreži srednjeg napona..[4] .....	16
Slika 3.1. Podjela inteligentnih algoritama. [9].....	21
Slika 3.2. Shematski prikaz evolucije. [9].....	22
Slika 3.3. Iteracijska petlja evolucijskog algoritma. ....	25
Slika 3.4. Struktura OpenDSS-a. [6] .....	28
Slika 3.5. Primjer elementa za distribuciju energije. [6] .....	29
Slika 3.6. Element za konverziju energije. [6] .....	30
Slika 3.7. Model elementa za konverziju energije sa kompenzacijom struje. [6].....	30
Slika 3.8. Petlja za rješavanje problema u OpenDSS-u. [6].....	31
Slika 4.1. IEEE123Bus testna mreža sa dodanim distribuiranim izvorima. ....	40
Slika 4.2. Ovisnost nazivne snage fotonaponskog sustava temperaturi ćelije. ....	41
Slika 4.3. Ovisnost efikasnosti fotonaponskog sustava o proizvedenoj snazi. ....	42
Slika 4.4. Ovisnost iradijacije (proizvodnje snage) o vremenu.....	42
Slika 4.5. Ovisnost dnevne temperature o vremenu.....	43
Slika 4.6. Krivulja opterećenja koja opisuje trošila spojena na mrežu. ....	43
Slika 4.7. Promjena gubitaka po satu za slučaj kada distribuirani izvori nisu spojeni na mrežu..	44
Slika 4.8. Relativni gubici po satu za slučaj kada distribuirani izvori nisu spojeni na mrežu. ....	44
Slika 4.9. Promjena gubitaka po satu za slučaj kada upravljivi distribuirani izvori nisu spojeni na mrežu. ....	45
Slika 4.10. Relativni gubici po satu za slučaj kada upravljivi distribuirani izvori nisu spojeni na mrežu. ....	45
Slika 4.11. Promjena gubitaka po satu za konstantne snage upravljivih distribuiranih izvora. ....	46
Slika 4.12. Relativni gubici po satu za konstantne snage upravljivih distribuiranih izvora. ....	46
Slika 4.13. Djelatna i jalova snaga prvog upravljivog distribuiranog izvora. ....	47
Slika 4.14. Djelatna i jalova snaga drugog upravljivog distribuiranog izvora. ....	47
Slika 4.15. Promjena gubitaka po satu za optimirani rad upravljivih distribuiranih izvora.....	48
Slika 4.16. Relativni gubici po satu za optimirani rad upravljivih distribuiranih izvora. ....	48
Slika 4.17. Djelatne i jalove snage prvog distribuiranog izvora i trošila nakon optimizacije.....	49
Slika 4.18. Djelatne i jalove snage drugog distribuiranog izvora i trošila nakon optimizacije.....	49



Slika 4.19. Djelatne i jalove snage prvog distribuiranog izvora i trošila te profil proizvodnje fotonaponskih elektrana. ....	50
Slika 4.20. Djelatne i jalove snage drugog distribuiranog izvora i trošila te profil proizvodnje fotonaponskih elektrana. ....	50
Slika 4.21. Naponski profili mreže za tri simulirana slučaja. ....	52

## POPIS TABLICA

Tablica 2.1. Usporedba često korištenih tipova distribuirane energije. [1].....	10
Tablica 3.1. Različiti tipovi modela opterećenja. [6] .....	34
Tablica 3.2. Opis svih mogućih parametara za funkciju optimizacije diferencijalnim evolucijskim algoritmom u Python-u. [8].....	36
Tablica 4.1. Popis distribuiranih izvora spojenih na mrežu i njihove karakteristike. ....	39
Tablica 4.2. Tablični prikaz gubitaka za različite slučajeve simulacija. ....	51
Tablica 4.3. Tablične vrijednosti minimalnih i maksimalnih napona u mreži za tri simulirana slučaja.....	53

## SAŽETAK

Ovaj rad opisuje postupak optimizacije proizvodnje upravljivog distribuiranog izvora koji je priključen na srednje naponsku distributivnu mrežu na kojoj su instalirani upravljivi i neupravljivi distribuirani izvori. Za optimizaciju je korišten diferencijalni evolucijski algoritam i optimizacija je odrađena za zadane dnevne profile opterećenja i intenzitet energetskih izvora neupravljive distribuirane proizvodnje. Kao simulacijski alat korišten je OpenDSS računalni program, a za provedbu optimizacije korišteni su gotove biblioteke u Python okruženju. Simulacija je rađena za period od jednog dana u intervalima po sat vremena. Rezultati pokazuju kako su se gubici optimizacijom smanjili te se naponski profil sustava poboljšao.

**Ključne riječi:** optimizacija, distribuirani izvori, evolucijski algoritam, Python, OpenDSS

## **ABSTRACT**

This paper describes the process of optimizing the production of controllable distributed power source that is connected to a medium voltage distribution network on which there are controllable and uncontrollable distributed power sources. Differential evolutionary algorithm was used for the optimization and optimization was performed for the given daily load profiles and the intensity of energy sources of the uncontrollable distributed production. OpenDSS computer program was used as simulation tool, and for the implementation of the optimization the ready-made libraries in the Python environment were used. The simulation was done for a period of one day in intervals of one hour. The results show that the optimization has reduced the losses and the systems voltage profile improved.

**Keywords:** optimization, distributed power sources, evolution algorithm, Python, OpenDSS

## ŽIVOTOPIS

Goran Šostarko rođen je 31. srpnja 1990. godine u Osijeku. Nakon završetka osnovne škole upisuje Strojarsku tehničku školu Osijek, smjer: računalni tehničar za strojarstvo. Dvije godine je predstavljao Strojarsku tehničku školu Osijek na državnom natjecanju u strojarstvu u organizaciji Agencije za strukovno obrazovanje i obrazovanje odraslih te je obje godine osvojio drugo mjesto u crtanju tehničkih nacrtu u AutoCAD-u.

Po završetku srednje tehničke škole, 2010. godine, zapošljava se u privatnoj tvrtki u Kneževim Vinogradima kao strojarski tehničar. Nakon godinu dana napreduje na mjesto zamjenika voditelja proizvodnje uređaja za pročišćavanje otpadnih voda.

2012. godine upisuje stručni studija elektrotehnike na Elektrotehničkom fakultetu u Osijeku, smjer: Automatika. Po završetku stručnog studija upisuje razlikovnu godinu na istom fakultetu.

Po završetku razlikovne godine, 2016. godine, upisuje sveučilišni diplomski studij elektrotehnike, smjer elektroenergetika (izborni blok: Industrijska elektroenergetika) na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku.

Goran Šostarko

---

## PRILOZI

### PRILOG I – Popis trošila, lokacija, spoj, naponi i snage

Naziv trošila	Lokacija priključenja	Broj faza	Tip spoja	Model trošila	Napon [kV]	Djelatna snaga [kW]	Jalova snaga [kvar]
S1a	1.1	1	Y	1	2.4	40.0	20.0
S2b	2.2	1	Y	1	2.4	20.0	10.0
S4c	4.3	1	Y	1	2.4	40.0	20.0
S5c	5.3	1	Y	5	2.4	20.0	10.0
S6c	6.3	1	Y	2	2.4	40.0	20.0
S7a	7.1	1	Y	1	2.4	20.0	10.0
S9a	9.1	1	Y	1	2.4	40.0	20.0
S10a	10.1	1	Y	5	2.4	20.0	10.0
S11a	11.1	1	Y	2	2.4	40.0	20.0
S12b	12.2	1	Y	1	2.4	20.0	10.0
S16c	16.3	1	Y	1	2.4	40.0	20.0
S17c	17.3	1	Y	1	2.4	20.0	10.0
S19a	19.1	1	Y	1	2.4	40.0	20.0
S20a	20.1	1	Y	5	2.4	40.0	20.0
S22b	22.2	1	Y	2	2.4	40.0	20.0
S24c	24.3	1	Y	1	2.4	40.0	20.0
S28a	28.1	1	Y	5	2.4	40.0	20.0
S29a	29.1	1	Y	2	2.4	40.0	20.0
S30c	30.3	1	Y	1	2.4	40.0	20.0
S31c	31.3	1	Y	1	2.4	20.0	10.0
S32c	32.3	1	Y	1	2.4	20.0	10.0
S33a	33.1	1	Y	5	2.4	40.0	20.0
S34c	34.3	1	Y	2	2.4	40.0	20.0
S35a	35.1.2	1	D	1	4.160	40.0	20.0
S37a	37.1	1	Y	2	2.4	40.0	20.0
S38b	38.2	1	Y	5	2.4	20.0	10.0
S39b	39.2	1	Y	1	2.4	20.0	10.0
S41c	41.3	1	Y	1	2.4	20.0	10.0
S42a	42.1	1	Y	1	2.4	20.0	10.0
S43b	43.2	1	Y	2	2.4	40.0	20.0
S45a	45.1	1	Y	5	2.4	20.0	10.0
S46a	46.1	1	Y	1	2.4	20.0	10.0
S47	47	3	Y	5	4.160	105.0	75.0
S48	48	3	Y	2	4.160	210.0	150.0
S49a	49.1	1	Y	1	2.4	35.0	25.0
S49b	49.2	1	Y	1	2.4	70.0	50.0
S49c	49.3	1	Y	1	2.4	35.0	20.0
S50c	50.3	1	Y	1	2.4	40.0	20.0
S51a	51.1	1	Y	1	2.4	20.0	10.0
S52a	52.1	1	Y	1	2.4	40.0	20.0
S53a	53.1	1	Y	1	2.4	40.0	20.0

S55a	55.1	1	Y	2	2.4	20.0	10.0
S56b	56.2	1	Y	1	2.4	20.0	10.0
S58b	58.2	1	Y	5	2.4	20.0	10.0
S59b	59.2	1	Y	1	2.4	20.0	10.0
S60a	60.1	1	Y	1	2.4	20.0	10.0
S62c	62.3	1	Y	2	2.4	40.0	20.0
S63a	63.1	1	Y	1	2.4	40.0	20.0
S64b	64.2	1	Y	5	2.4	75.0	35.0
S65a	65.1.2	1	D	2	4.160	35.0	25.0
S65b	65.2.3	1	D	2	4.160	35.0	25.0
S65c	65.3.1	1	D	2	4.160	70.0	50.0
S66c	66.3	1	Y	1	2.4	75.0	35.0
S68a	68.1	1	Y	1	2.4	20.0	10.0
S69a	69.1	1	Y	1	2.4	40.0	20.0
S70a	70.1	1	Y	1	2.4	20.0	10.0
S71a	71.1	1	Y	1	2.4	40.0	20.0
S73c	73.3	1	Y	1	2.4	40.0	20.0
S74c	74.3	1	Y	2	2.4	40.0	20.0
S75c	75.3	1	Y	1	2.4	40.0	20.0
S76a	76.1.2	1	D	5	4.160	105.0	80.0
S76b	76.2.3	1	D	5	4.160	70.0	50.0
S76c	76.3.1	1	D	5	4.160	70.0	50.0
S77b	77.2	1	Y	1	2.4	40.0	20.0
S79a	79.1	1	Y	2	2.4	40.0	20.0
S80b	80.2	1	Y	1	2.4	40.0	20.0
S82a	82.1	1	Y	1	2.4	40.0	20.0
S83c	83.3	1	Y	1	2.4	20.0	10.0
S84c	84.3	1	Y	1	2.4	20.0	10.0
S85c	85.3	1	Y	1	2.4	40.0	20.0
S86b	86.2	1	Y	1	2.4	20.0	10.0
S87b	87.2	1	Y	1	2.4	40.0	20.0
S88a	88.1	1	Y	1	2.4	40.0	20.0
S90b	90.2	1	Y	5	2.4	40.0	20.0
S92c	92.3	1	Y	1	2.4	40.0	20.0
S94a	94.1	1	Y	1	2.4	40.0	20.0
S95b	95.2	1	Y	1	2.4	20.0	10.0
S96b	96.2	1	Y	1	2.4	20.0	10.0
S98a	98.1	1	Y	1	2.4	40.0	20.0
S99b	99.2	1	Y	1	2.4	40.0	20.0
S100c	100.3	1	Y	2	2.4	40.0	20.0
S102c	102.3	1	Y	1	2.4	20.0	10.0
S103c	103.3	1	Y	1	2.4	40.0	20.0
S104c	104.3	1	Y	1	2.4	40.0	20.0
S106b	106.2	1	Y	1	2.4	40.0	20.0
S107b	107.2	1	Y	1	2.4	40.0	20.0
S109a	109.1	1	Y	1	2.4	40.0	20.0

S111a	111.1	1	Y	1	2.4	20.0	10.0
S112a	112.1	1	Y	5	2.4	20.0	10.0
S113a	113.1	1	Y	2	2.4	40.0	20.0
S114a	114.1	1	Y	1	2.4	20.0	10.0
Total:						3490.0	1920.0



## PRILOG II – Python kod za simuliranje sustava i optimizaciju

```
1. # -*- coding: utf-8 -*-
2. """
3. Created on Tue Apr 03 12:10:19 2018
4.
5. @author: Goran Šostarko
6. """
7.
8. import win32com.client
9. import matplotlib.pyplot as plt
10. import numpy as np
11. from scipy.optimize import differential_evolution as DE
12. from numpy import array
13. import time
14.
15.
16.
17. start = time.time()
18.
19. dssObj = win32com.client.Dispatch("OpenDSSEngine.DSS")
20.
21. #UNOS IMENA DATOTEKE KOJU POKRECEMO
22. fileName = 'IEEE123Master.dss' # 'IEEE123Master.dss'
23. if dssObj.Start(0) == False:
24.     print ("DSS Failed to Start")
25. else:
26.     dssText = dssObj.Text
27.     dssCircuit = dssObj.ActiveCircuit
28.     dssSolution = dssCircuit.Solution
29.     dssObj.ClearAll()
30.
31.
32.
33. #Izvođenje naredbe "compile"
34. dssText.Command = "compile " + fileName
35. #dssText.Command = r"Compile 'E:\OneDrive - etfos.hr\NASTAVA\ZAVRSNI_DIPLOMSKI\2017_18\
    D01_Sostarko\IEEE123Master.dss'"
36. #dnevno opterećenje
37. teret = [0.4, 0.35, 0.30, 0.30, 0.35, 0.4, 0.8, 0.65, 0.5, 0.5, 0.55, 0.6, 0.5, 0.5, 0.
    5, 0.6, 0.8, 0.95, 0.95, 0.99, 0.99, 0.9, 0.6, 0.4]
38. iradijacija = [0,0,0,0,0,0, 0.1, 0.2, 0.3, 0.5, 0.8, 0.9, 1, 1, 0.99, 0.9, 0.7, 0.4, 0.
    1, 0, 0, 0, 0, 0]
39.
40. gubici=[]
41.
42. dssSolution.Solve()
43. BusNames = dssCircuit.AllBusNames
44.
45. # SA ind JE U POZIVU FUNKCIJE OZNACENA INDIVIDUA U DE
46. def funkcija(ind, *var):
47.
48. # DEKODIRANJE SNAGA DG IZ DE INDIVIDUE
49.     Pg1=ind[0]
50.     Qg1=ind[1]
51.     Pg2=ind[2]
52.     Qg2=ind[3]
53. # SLANJE DE JEDINKE U OpenDSS
54.     dssText.Command = "Edit Generator.DG1 kw= " +str(Pg1) + " kvar= " + str(Qg1)
55.     dssText.Command = "Edit Generator.DG2 kw= " +str(Pg2) + " kvar= " + str(Qg2)
56. # POKRETANJE SIMULACIJE U OpenDSS
57.     dssSolution.Solve()
58.
59.
60. # OCITAVANJE IZRACUNATIH GUBITAKA
```

```

61.     los= dssCircuit.Losses # dssObj.ActiveCircuit.Losses
62.     gub=array(los)/1000
63.
64. #PROVJERA NAPONA I PENALIZACIJA
65.     V_PU=dssObj.ActiveCircuit.AllBusVmagPU
66.     Naponi_PU=array(V_PU)
67.     for i in range (0, 278):
68.         if (Naponi_PU[i] < 0.9):
69.             gub = gub*10
70.         if (Naponi_PU[i] > 1.1):
71.             gub = gub*10
72.
73. # SLANJE VRIJEDNOSTI FUNKCIJE CILJA U DE
74.     return gub[0]
75.
76. V_prije_op=[]
77. V_nakon_op=[]
78.
79. rezultat=[]
80. optimgubici=[]
81. gubicinakon=[]
82. ukupna_snaga=[]
83. sourceP=[]
84. PV_snaga=[]
85.
86. #naponi
87. V_p=[]
88. V_n=[]
89.
90. # ISCITAVANJE
91. Tp = []
92. Tq = []
93. iLoad = dssCircuit.Loads.First
94. while iLoad:
95.     #Skaliram teret za vrijednost iz polja teret
96.     Tp.append(dssCircuit.Loads.kw)
97.     Tq.append(dssCircuit.Loads.kvar)
98.     #Prolazim na iduci teret
99.     iLoad = dssCircuit.Loads.Next
100.     Tlp = sum(Tp) # Ukupna djelatna snaga trosila
101.     TLq = sum(Tq) # Ukupna jalova snaga trosila
102.
103.     #granice za DE Pg1(0-50%), Qg1(+/-2,5%), Pg2(0-50%), Qg2(+/-2,5%)
104.     granice = [(0,0.5*Tlp),(-0.25*TLq, 0.25*TLq),(0,0.5*Tlp),(-0.25*TLq,0.25*TLq)]
105.
106.     for t in range(0,24):
107.         #Prolazim kroz svaki teret i skaliram ga
108.         iLoads = dssCircuit.Loads.First
109.         while iLoads:
110.             #Skaliram teret za vrijednost iz polja teret
111.             dssCircuit.Loads.kw = dssCircuit.Loads.kw * teret[t]
112.             dssCircuit.Loads.kvar = dssCircuit.Loads.kvar * teret[t]
113.             #Prolazim na iduci teret
114.             iLoads = dssCircuit.Loads.Next
115.
116.             dssText.Command = "Edit PvSystem.PV1 irrad=" +str(iradijacija[t])
117.             dssText.Command = "Edit PvSystem.PV2 irrad=" +str(iradijacija[t])
118.             dssText.Command = "Edit PvSystem.PV3 irrad=" +str(iradijacija[t])
119.             # POSTAVLJANJE PROIZVODNJE DG-ova na MAX uz faktor snage DG-
120.             ova jednak 1 (ne proizvode jalovnu snagu)
121.             dssText.Command = "Edit Generator.DG1 kw= " +str(0.5*Tlp) + " kvar= " + str(
122.                 0.0*TLq)
123.             dssText.Command = "Edit Generator.DG2 kw= " +str(0.5*Tlp) + " kvar= " + str(
124.                 0.0*TLq)
125.
126.         dssSolution.Solve()

```

```

124.         los1= dssCircuit.Losses # dssObj.ActiveCircuit.Losses # ili dssCircuit.Losse
    s
125.         gub1=array(los1[0])/1000
126.         gubici.append(gub1)
127.
128.
129.         for i in range(0,len(BusNames)):
130.             dssCircuit.SetActivebus(BusNames[i])
131.             V_p.append(dssObj.ActiveCircuit.ActiveBus.PuVoltages)
132.
133.             #izračun napona
134.             naponi_p=[]
135.             for i in range(0,len(BusNames)):
136.                 naponi_p.append(np.sqrt((V_p[i][0])**2+(V_p[i][1])**2))
137.
138.             nap_pr=[]
139.             nap_pr.append(naponi_p)
140.
141.
142.         result = DE(funkcija,granice,popsize=100, maxiter=1000, mutation=(0.7, 1), r
    ecombination=0.8)
143.
144.
145.         #snaga izvora po satu
146.         sourceP.append(dssCircuit.TotalPower[0])
147.
148.         #snaga na PV
149.         PV_snaga.append(dssCircuit.PVsystems.kw)
150.
151.         # los2=dssObj.ActiveCircuit.Losses
152.         # gub2=array(los2)/1000
153.         # gubicinakon.append(gub2)
154.
155.         rezultat.append(result.x)
156.         optimgubici.append(result.fun)
157.
158.
159.         for i in range(0,len(BusNames)):
160.             dssCircuit.SetActivebus(BusNames[i])
161.             V_n.append( dssObj.ActiveCircuit.ActiveBus.PuVoltages)
162.
163.             #izračun napona
164.             naponi_n=[]
165.             for i in range(0,len(BusNames)):
166.                 naponi_n.append(np.sqrt((V_n[i][0])**2+(V_n[i][1])**2))
167.
168.             nap_nak=[]
169.             nap_nak.append(naponi_n)
170.
171.
172.             #Prolazim kroz svaki teret i skaliram ga
173.             iLoad = dssCircuit.Loads.First
174.             while iLoad:
175.                 #Skaliram teret za vrijednost iz polja teret
176.                 dssCircuit.Loads.kw = dssCircuit.Loads.kw / teret[t]
177.                 dssCircuit.Loads.kvar = dssCircuit.Loads.kvar / teret[t]
178.                 #Prolazim na iduci teret
179.                 iLoad = dssCircuit.Loads.Next
180.
181.             #kontrola gdje se petlja nalazi trenutno
182.             print('trenutni sat: ' + str(t+1))
183.
184.
185.         #izračun i ispis vremena trajanja simulacije
186.         end = time.time()
187.         print('vrijeme trajanja simulacije: ' +str(end-start) + ' s')

```

```

188.
189.
190.     #izvlačenje podataka za crtanje grafova
191.     PG_1=[]
192.     QG_1=[]
193.     PG_2=[]
194.     QG_2=[]
195.
196.     for i in range (0,24):
197.         PG_1.append(rezultat[i][0])
198.         QG_1.append(rezultat[i][1])
199.         PG_2.append(rezultat[i][2])
200.         QG_2.append(rezultat[i][3])
201.
202.     vrijeme = range(0,24)
203.
204.     #crtanje grafova
205.     plt.figure(1)
206.     plt.step(vrijeme,PG_1, 'r', label= 'PG1')
207.     plt.step(vrijeme, QG_1, 'b', label='QG1')
208.     plt.title('PG1 i QG1')
209.     plt.xlabel('vrijeme [h]')
210.     plt.ylabel('PG1 [kW] i QG1 [kvar]')
211.     plt.grid(True)
212.     plt.legend()
213.     plt.show()
214.
215.     plt.figure(2)
216.     plt.step(vrijeme,PG_2, 'r',label= 'PG2')
217.     plt.step(vrijeme, QG_2, 'b', label= 'QG2')
218.     plt.title('PG2 i QG2')
219.     plt.xlabel('vrijeme [h]')
220.     plt.ylabel('PG2 [kW] i QG2 [kvar]')
221.     plt.grid(True)
222.     plt.legend()
223.     plt.show()
224.
225.     plt.figure(3)
226.     plt.step(vrijeme, optimgubici)
227.     plt.title('Promjena gubitaka po satu - optimiran rad DG-ova')
228.     plt.xlabel('Vrijeme [h]')
229.     plt.ylabel('Gubici [kW]')
230.     plt.grid(True)
231.     plt.show()
232.
233.     plt.figure(4)
234.     plt.step(vrijeme, gubici)
235.     plt.title('Promjena gubitaka po satu - konstantna snaga DG-ova')
236.     plt.xlabel('Vrijeme [h]')
237.     plt.ylabel('Gubici [kW]')
238.     plt.grid(True)
239.     plt.show()
240.
241.     #ukupna snaga sustava (snaga izvora + PDG1 + PDG2 + 3xP_PV)
242.     for a in range(0,24):
243.         ukupnaP=((sourceP[a]*(-1))+rezultat[a][0]+rezultat[a][2]+3*PV_snaga[a])
244.         ukupna_snaga.append(ukupnaP)
245.
246.     relativno=[]
247.     relat=[]
248.     for v in range (0,24):
249.         postotak=optimgubici[v]/ukupna_snaga[v]*100
250.         relativno.append(postotak)
251.
252.     for b in range(0,24):
253.         post=gubici[b]/ukupna_snaga[b]*100

```

```

254.         relat.append(post)
255.
256.     plt.figure(5)
257.     plt.step(vrijeme, relativno, 'r',label= 'nakon')
258.     plt.title('Relativni gubici po satu - optimiran rad DG-ova')
259.     plt.xlabel('Vrijeme [h]')
260.     plt.ylabel('Gubici [%]')
261.     plt.grid(True)
262.     plt.show()
263.
264.     plt.figure(6)
265.     plt.step(vrijeme, relat, 'r',label= 'nakon')
266.     plt.title('Relativni gubici po satu - konstantna snaga DG-ova')
267.     plt.xlabel('Vrijeme [h]')
268.     plt.ylabel('Gubici [%]')
269.     plt.grid(True)
270.     plt.show()
271.
272.     plt.figure(7)
273.     plt.step(vrijeme, ukupna_snaga)
274.     plt.xlabel('Vrijeme [h]')
275.     plt.ylabel('Ukupna snaga [kW]')
276.     plt.grid(True)
277.     plt.show()
278.
279.
280.     #izračun gubitaka
281.
282.     gub_prije=0
283.     gub_nakon=0
284.
285.     gub_prije=sum(gubici)
286.     gub_nakon=sum(optimgubici)
287.
288.     #
289.     #for i in range(0,24):
290.     #     gub_prije+=gubici[i]
291.     print('gubici prije optimizacije= ' + str(gub_prije) + ' kW' )
292.     #
293.     #for i in range(0,24):
294.     #     gub_nakon+=optimgubici[i]
295.     print('gubici nakon optimizacije= ' + str(gub_nakon) + ' kW' )
296.
297.
298.     #crtanje grafova
299.     PL = []
300.     QL = []
301.     for t in teret:
302.         PL.append(t*TLp)
303.         QL.append(t*TLq)
304.     plt.figure(8)
305.     plt.step(vrijeme,PG_1, 'r', label= 'PG1')
306.     plt.step(vrijeme, QG_1, 'b', label='QG1')
307.     plt.step(vrijeme,PL, 'salmon', label= 'P trosila', linestyle = '--')
308.     plt.step(vrijeme,QL, 'steelblue', label= 'Q trosila', linestyle = '-.')
309.     plt.title('PG1, QG1 i snaga trosila')
310.     plt.xlabel('vrijeme [h]')
311.     plt.ylabel('PG1 [kW], QG1 [kvar], PL [kW], QL [kvar]')
312.     plt.grid(True)
313.     plt.legend()
314.     plt.show()
315.
316.     plt.figure(9)
317.     plt.step(vrijeme,PG_2, 'r',label= 'PG2')
318.     plt.step(vrijeme, QG_2, 'b', label= 'QG2')
319.     plt.step(vrijeme,PL, 'salmon', label= 'P trosila', linestyle = '--')

```

```
320.     plt.step(vrijeme,QL, 'steelblue', label= 'Q trosila', linestyle = '-.')
321.     plt.title('PG2, QG2 i snaga trosila')
322.     plt.xlabel('vrijeme [h]')
323.     plt.ylabel('PG2 [kW], QG2 [kvar], PL [kW], QL [kvar]')
324.     plt.grid(True)
325.     plt.legend()
326.     plt.show()
327.
328.
329.
330.     plt.figure(11)
331.     plt.plot(naponi_p[0:132], 'r', label= 'Prije opt.')
332.     plt.plot(naponi_n[0:132], 'b', label= 'Nakon opt.')
333.
334.     plt.title('Naponski profil')
335.     plt.xlabel('Sabirnice')
336.     plt.ylabel('Napon [p.u.]')
337.     plt.legend()
338.     plt.grid(True)
339.     plt.show()
340.
341.
342.
343.
```