

OPTIMALNA REKONFIGURACIJA DISTRIBUTIVNE MREŽE

Darko ŠOŠIĆ, Univerzitet u Beogradu – Elektrotehnički fakultet, Srbija
Predrag STEFANOV, Univerzitet u Beogradu – Elektrotehnički fakultet, Srbija
Đorđe LAZOVIĆ, Univerzitet u Beogradu – Elektrotehnički fakultet, Srbija

KRATAK SADRŽAJ

U ovom radu je predložen novi koncept odabira statusa prekidača mreže u procesu formiranja populacije pri određivanju optimalne rekonfiguracije distributivne mreže. Baziranom na konceptu osnovnih petlji, predloženom procedurom se status uključenosti prekidača određuje na osnovu adaptivnog određivanja petlji i odgovarajućih grana sa prekidačima kandidatima za promenu statusa. Ovakvim izborom se ne dozvoljava formiranje neizvodljivih rešenja, odnosno konfiguracija mreže kod kojih su narušena ograničenja povezanosti i radijalnosti, čime se znatno smanjuje broj pretraga i ubrzava optimizacioni postupak. Na osnovu ovakvih, sukcesivnih pretraga dozvoljenih rešenja takodje je moguće ispraviti neizvodljiva rešenja koja su kreirana u procesu meta-heurističke optimizacije, određivanjem najbližih izvodljivih rešenja, čime se u najvećoj mogućoj meri zadržavaju osnovne karakteristike optimizacionog postupka i ubrzava nalaženje optimalne konfiguracije distributivne mreže. Ovaj koncept je u radu implementiran i testiran na algoritmu za nalaženje optimalne konfiguracije korišćenjem diferencijalne evolucije. Testiranje karakteristika optimizacionog postupka izvršeno je na standardnoj IEEE 33 test distributivnoj mreži.

Ključne reči: rekonfiguracija, diferencijalna evolucija, distributivne mreže

sosic@etf.rs

UVOD

Rekonfiguracija distributivne mreže je jedna od najznačajnijih upravljačkih akcija kojima se optimizuje rad distributivnih mreža. Proračunima optimalne rekonfiguracije se obezbeđuje optimalan status uključenosti rasklopnih uređaja distributivne mreže, sa aspekta različitih, korisnički specificiranih optimizacionih kriterijuma. Ovi optimizacioni kriterijumi obično su vezani za minimizaciju gubitaka aktivne snage u distributivnoj mreži, izbalansiranost opterećenja pojedinih grana mreže, pokazatelje pouzdanosti napajanja potrošača, da bi se sve češće koristili kriterijumi vezani za plasman energije iz distribuiranih izvora, obično iskazani kroz maksimizaciju njihove proizvodnje. Pri tom, pri dobijenoj optimalnoj konfiguraciji mreže moraju da budu zadovoljena pogonska ograničenja vezana za opterećenja pojedinih vodova i transformatora, kao i padove napona na izvodima mreže, odnosno opsege promene napona na krajnjim potrošačima. Pored ovih ograničenja, pri upravljanju distributivnim mrežama, nezavisno od toga da li su u njima uključeni ili ne distribuirani izvori, još uvek se zahteva da se u svim radnim režimima sačuva radijalnost mreže, kao osnova za zadržavanje jednostavne strukture zaštite i upravljanja regulacionim resursima.

Kao takav, gore pomenuti problem se svrstava u mešoviti celobrojno/binarni nelinearni ne diferencijabilni optimizacioni problem. Umesto klasičnih metoda optimizacije (1, 2), za rešavanje ovog problema predložen je niz heurističkih metoda (3-5). Heurističke metode su veoma brzo pronalaze rešenje i mogu se primeniti u okviru automatizacije distributivnih mreža u realnom vremenu. Bez obzira na brzinu proračuna, ovi algoritmi najčešće pronalaze lokalne optimume, zbog čega se poslednjih godina meta-heurističke optimizacione metode sve češće koriste za rešavanje problema rekonfiguracije distributivne mreže: algoritma kolonije mrava (6), roj čestica (7), genetički algoritma (8), itd.

Do sada pomenuti radovi su problem rekonfiguracije distributivne mreže rešavali primenom meta-heurističkih metoda jednokriterijumske optimizacije. U literaturi postoji i manji broj radova koji su predložili višekriterijumsku optimizaciju rekonfiguracije distributivne mreže korišćenjem NSGA 2 za minimizaciju gubitaka aktivne snage, unapređenje naponskih prilika i uspostavljanja izbalansirane potrošnje uz minimalan broj prekidačkih akcija (9, 10), ili više-kriterijumskog gravitacionog algoritma koji je uvažavao i prisustvo distribuirane proizvodnje (11).

Iako ove metode uspevaju da se izbore sa problemima nediferencijabilnosti kriterijumskih funkcija i zaustavljanja proračuna u lokalnim minimumima, one kao kombinatorne metode nailaze na nove probleme. Ti

problemi su u velikoj meri vezani za preveliki broj pretraživanja, generisanjem velikog broja neizvodljivih rešenja koja je potrebno odbaciti, i druga. Ovi problemi otvaraju prostor za dalja unapređenja ovih metoda.

Bez obzira na izabranu optimizacionu metodu, u svim pomenutim radovima koji su koristili meta-heurističke metode za smanjivanje broja pretraga kojima se u početnoj populaciji obezbeđuje da sve razmatrane konfiguracije odgovaraju radijalnoj strukturi distributivnih mreža upotrebljavan je koncept osnovnih petlji (12). Ovaj koncept predstavlja veliki napredak u odnosu na binarno kodiranje (13), ali ima ozbiljan nedostatak koji se ogleda u mogućnosti formiranja nepovezanih delova mreže. S druge strane, ako se u postupku optimizacije kao rezultat primene odgovarajućeg operatora dobije neizvodljivo rešenje, ono se izbacuje i generiše novo sve dokle se ne dobije izvodljivo rešenje. Zbog toga je u ovom radu predložen novi pristup kodiranja grana, nazvan konceptom adaptivnih petlji.

Baziranom na konceptu osnovnih petlji, predloženom procedurom se rešenja određuju na osnovu adaptivnog određivanja petlji, tj. sukcesivnim određivanjem uslova izbora svih prekidača kojima se obezbeđuje radijalnost mreže, bez ostavljanja mogućnosti da rešenje odgovara nepovezanom sistemu. Time se znatno smanjuje broj pretraga i ubrzava postupak generisanja populacije. Primenom predloženog koncepta sukcesivnih pretraga dozvoljenih rešenja moguće je i korigovati neizvodljiva rešenja koja su kreirana u procesu meta-heurističke optimizacije, određivanjem najbližih izvodljivih rešenja, čime se u najvećoj mogućoj meri zadržavaju osnovne karakteristike optimizacionog postupka. Prema tome, upotrebom adaptivnih petlji izbegava se odbacivanje loših međurešenja koja se generišu upotrebom operatora odabrane optimizacione metode. Pomoću ovog koncepta iz svakog lošeg rešenja se izvlači maksimum dobrih podataka, a samo loši podaci se zamenjuju ispravnim podacima. Upravo zbog ovoga se skraćuje vreme proračuna pošto se ispravljeno rešenje direktno poredi sa već postojećim u fazi "nokout" selekcije i samo bolje rešenje prolazi u narednu populaciju.

Ovaj koncept je u radu implementiran i testiran na algoritmu za nalaženje optimalne konfiguracije korišćenjem diferencijalne evolucije (DE), koja se pokazala kao odličan izbor prilikom rešavanja nekih drugih problema u energetici (14). Testiranje karakteristika optimizacionog postupka izvršeno je na standardnoj IEEE 33 test distributivnoj mreži pri čemu je razmatrano smanjenje gubitaka aktivne snage.

Posle uvodnih razmatranja u kojima je izvršen pregled literature i navedna glavna ideja ovog rada, u drugom delu je data opšta formulacija optimizacionog problema rekonfiguracije distributivnih mreža i definisana kasnije u radu korišćena kriterijumska funkcija za koju je vršeno testiranje predloženog algoritma. U trećem delu je objašnjen koncept adaptivnih petlji, definisanjem postupka za kreiranje početne populacije i postupka za ispravku neizvodljivih rešenja. U četvrtom delu dat je kratak opis primenjenog algoritma diferencijalne evolucije, dok je u petom delu dat kompletan prikaz korišćenog optimizacionog algoritma. Rezultati proračuna na test sistemu su prikazani u šestom, dok je na kraju dat zaključak sa daljim smernicama za istraživanja.

OPTIMIZACIONI PROBLEM

Optimalnom rekonfiguracijom se obezbeđuje optimalan status uključenosti rasklopnih uređaja distributivne mreže, sa aspekta različitih, korisnički specificiranih optimizacionih kriterijuma. U ovom postupku se određuje skup upravljačkih promenljivih koji odgovara prekidačkim elementima koje treba otvoriti u distributivnoj mreži u datom radnom režimu (ili u određenom vremenskom periodu) da bi se formirala radijalna konfiguracija povezane distributivne mreže sa optimalnom vrednošću kriterijumskih funkcija. Zbog jednostavnosti prikaza predloženog algoritma u radu je razmatran statički problem, vezan za jedan radni režim sa zadatim opterećenjima u cilju smanjivanja ukupnih gubitaka aktivne snage.

$$F = \min \sum_{i=1}^{N_b} R_i \times I_i^2, \quad (1)$$

gde je sa R_i označena otpornost i -te grane, a I_i je struja koja teče po i -toj grani.

Kreirana konfiguracija distributivne mreže mora da zadovolji topološka i radna ograničenja koja operator distributivne mreže mora da uvaži prilikom modifikovanja trenutnog radnog stanja. Ova ograničenja su povezana sa termalnim ograničenjima elemenata mreže, dozvoljenim padovima napona, povezanošću mreže i radijalnom strukturom. Poslednje pomenuto ograničenje se postavlja zbog tehničkih uslova, koji se odnose na regulatore napona i jednostavniju relejnu zaštitu. Izvodljiva rešenja su ona koja zadovoljavaju sledeći skup ograničenja:

1. Jednačine tokova snaga predstavljaju ograničenja tipa jednakosti,
2. $|I_{i,\min}| \leq |I_i| \leq |I_{i,\max}|, \quad i = 1, \dots, N_b,$
3. $|V_{j,\min}| \leq |V_j| \leq |V_{j,\max}|, \quad j = 1, \dots, N_{bus},$
4. Svi čvorovi mreže moraju da budu pod naponom,
5. Potrebno je sačuvati radijalnu strukturu mreže.

Sa I_i je označena struja koja teče kroz i -tu granu, $I_{i,\min}$ i $I_{i,\max}$ označavaju donju i gornju graničnu vrednost struje koja može da teče preko i -te grane, V_j je napon j -tog čvora, dok su sa $V_{j,\min}$ i $V_{j,\max}$ obeležena donja i gornja granična vrednost napona j -tog čvora.

KONCEPT ADAPTIVNIH PETLJI

U algoritmima optimalne rekonfiguracije distributivnih mreža broj upravljačkih promenljivih se izjednačava sa brojem prekidačkih elemenata čijom se simultanom promenom statusa vrši promena konfiguracije mreže u cilju optimizacije kriterijumskih funkcija. Razvijene heurističke metode uglavnom dozvoljavaju promenu statusa samo jednog para prekidačkih elemenata, (otvaranje grane sa najmanjom strujom u petlji (4) ili na osnovu maksimalne razlike napona na krajevima otvorenih prekidača (15)), čime je rešenje u velikoj meri zavisno od početnog uklopnog stanja distributivne mreže i ne dozvoljava nalaženje globalnog optimuma. Simultano razmatranje većeg broja upravljačkih promenljivih čini zadatak rekonfiguracije izuzetno teškim, što je i bio razlog uvođenja meta-heurističkih metoda u njegovo rešavanje.

Veliki pomak u ovoj oblasti je postignut primenom koncepta osnovnih petlji (12). Uvažavanjem pretpostavke da u svakoj grani mreže postoji rasklopni uređaj kojim se može upravljati i menjati topologija mreže, broj upravljačkih promenljivih se svodi na broj nezavisnih - osnovnih petlji mreže formiranih prema pozicijama normalno otvorenih prekidača u osnovnoj konfiguraciji, a mogućnost promene njihovog statusa se vezuje za analizu pripadnosti ovim petljama. Kako neke grane pripadaju jednoj ili više petlji, izbor upravljačkih promenljivih nije nezavistan, kao što se obično usvaja pri primeni ovog koncepta, čime se ne sprečava mogućnost formiranja ostrva ili izolovanja potrošača. Ujedno, to čini i glavni nedostatak ovog pristupa. Generisana neizvodljiva rešenja se u tom slučaju odbacuju i postupak ponavlja do dobijanja prihvatljivog rešenja.

Kod principa adaptivnih petlji izbegava se mogućnost kreiranja konfiguracije distributivne mreže koja ima petlje ili kod koje neki čvorovi nisu povezani. Izbor upravljačkih promenljivih je vezan za sukcesivnu analizu pripadnosti petljama koje se menjaju u zavisnosti od do tada izabranih promenljivih. Ovaj postupak se može koristiti kako pri formiranju početne populacije, tako i pri ispravci neizvodljivih rešenja dobijenih primenom optimizacionih operatora. Detaljan prikaz neophodnih koraka za ova dva slučaja je dat u narednom izlaganju.

Kreiranje početne populacije

Kao što je navedeno, skupovi dozvoljenih grana koje se isključuju nisu fiksni nego se menjaju sa promenom konfiguracije mreže. Da bi se omogućila što jednostavnija formulacija algoritma adaptivnih petlji, konfiguracije distributivne mreže su prikazane grafovima sa orijentisanim granama. Smer orijentisane grane ide od napojnog čvora ka napajanom čvoru, odnosno poklapa se sa prirodnim tokom snage radijalnih distributivnih mreža i jednoznačno je određena za svaku izvodljivu konfiguraciju. Pored toga, uvedena su dva operatora, „←” i „→”, kojima se zadaje smer pretrage kao uslov za dodavanje grana u odgovarajući skup grana na osnovu kog se vrši definisanje kandidata za pojedine upravljačke promenljive. Prvi operator „←” označava da se pretraga i dodavanje grana u pripadajući skup vrši prema napojnom čvoru, u smeru suprotnom orijentaciji grana mreže, dok drugi operator „→” ukazuje da se pretraga vrši u obrnutom smeru.

Kao što je navedeno, svako rešenje i je definisano vektorom upravljačkih promenljivih t_i , odnosno vektorom sa indeksima grana sa otvorenim prekidačima $t_i = [t_{i1}, t_{i2}, \dots, t_{ik}, \dots, t_{il}]$. Svi ostali prekidači su zatvoreni. Prema uslovu radijalnosti, dužina vektora je određena brojem osnovnih petlji mreže l . Korišćenjem predloženih operatora, algoritam za kreiranje rešenja – elemenata početne populacije sukcesivnim izborom upravljačkih promenljivih primenom koncepta adaptivnih petlji opisuje se sledećim koracima:

- 1) Primenom koncepta osnovnih petlji izabrati jedno, proizvoljno izvodljivo rešenje t_0 . Ovim izborom se vrši inicijalizacija iterativnog postupka.
Skup mogućih vrednosti za k -tu upravljačku promenljivu t_{ik} se definiše kao skup indeksa grana koje je moguće isključiti istovremenim zatvaranjem prekidača u grani t_{0k} . Ovaj skup je označen sa L_{ck} i dobija se polaganjem od analiza povezanosti grane t_{0k} :
- 2) Dodavanjem svih indeksa grana na putanjama dobijenim pretragom od predajnog čvora, odnosno od prijemnog čvora grane t_{0k} , primenom operatora „←” formirati skupove L_{sk} i L_{rk} , respektivno.
- 3) Formirati skup kandidata L_{ck} , primenom relacije

$$L_{ck} = \{t_{0k}\} + (L_{sk} \cup L_{rk}) / (L_{sk} \cap L_{rk}). \quad (2)$$
 Grane koje odgovaraju ovom skupu čine adaptivnu petlju sa granama koje se mogu isključiti.
- 4) Proizvoljno izabrati jednu granu iz skupa L_{ck} koja će se isključiti, odnosno izabrati vrednost t_{ik} .
- 5) Zatvoriti prekidač grane t_{0k} , otvoriti granu t_{ik} i izvršiti ponovnu orijentaciju grana grafa.

- 6) Proračun ponoviti l puta, dok se sukcesivno ne dobiju svi elementi vektora t_i , odnosno jednog rešenja početne populacije.
- 7) Rešenju t_0 pridružiti slučajnim izborom neko od do tog trenutka određenih izvodljivih rešenja i nastaviti sa traženjem narednog rešenja početne populacije.

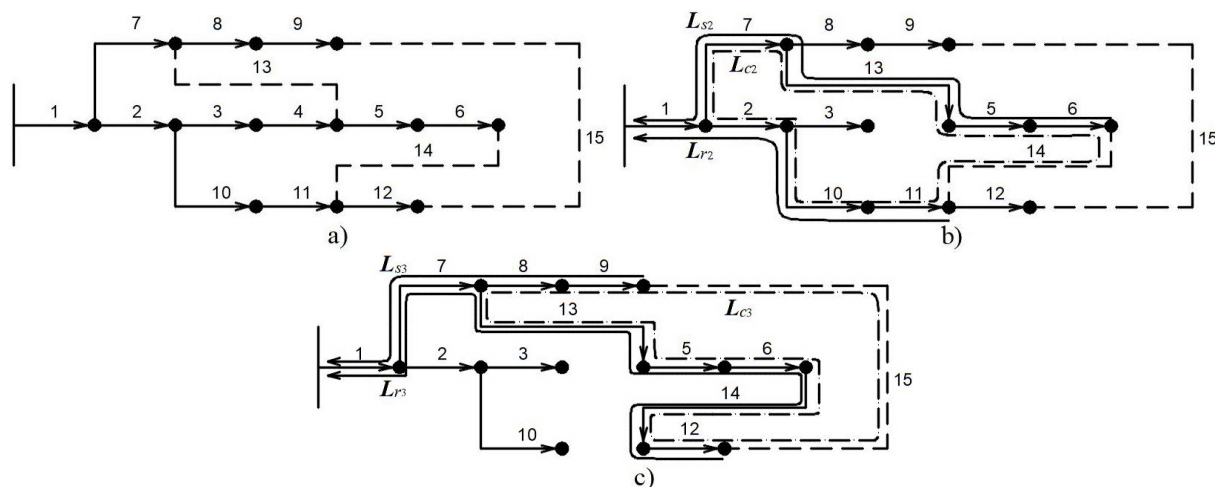
Opisani postupak se ponavlja N_p puta, tj. onoliko puta koliko ima članova populacije. Predloženi algoritam za kreiranje početne populacije je ilustrovan na primeru distributivne mreže prikazane na slici 1 a). Skup t_0 povezanih prekidača označenih na slici isprekidanom linijom, u daljim analizama je usvojen sa sledećim redosledom elemenata $t_0 = [13, 14, 15]$. Uz pretpostavku da se u svakoj grani nalazi rasklopni uređaj, skupovi čiji su elementi grane odgovarajućih osnovnih petlji su $L_1 = \{2, 3, 4, 7, 13\}$, $L_2 = \{3, 4, 5, 6, 10, 11, 14\}$ i $L_3 = \{2, 7, 8, 9, 10, 11, 12, 15\}$. Primećuje se da i u ovom jednostavnom slučaju postoji više grana koje se nalaze u više osnovnih petlji, što kod nezavisne pretrage po skupovima ne sprečava dvostruki izbor iste grane, odnosno formiranje neizvodljivih konfiguracija, sa nepovezanim delovima sistema (npr., ako se izabere treći element, $L_{13} = 4$, za prvu upravljačku promenljivu i prvi element iz drugog skupa, $L_{21} = 3$, kreiralo bi se neizvodljivo rešenje, bez obzira na izbor treće upravljačke promenljive, pošto bi u tom slučaju čvor broj 4 ostao nepovezan).

Neka je u predloženoj proceduri adaptivnih petlji za prvu upravljačku promenljivu izabrana ista vrednost kao u prethodnom slučaju ($t_{i1} = L_{13} = 4$), čime je zatvoren prekidač u grani 4, a otvoren u grani 13. Nakon izvršene preorijentacije grana, za granu $t_{02} = 14$ formiraju se putanje do napojnog čvora, odnosno popunjavaju se skupovi L_{s2} i L_{r2} . Pošto se za ovu pretragu koristi operator „ \leftarrow ” potrebno je pronaći grane preko kojih se napajaju fiktivni ili stvarni potrošači na krajevima (uslovno početnom i krajnjem) ove grane. Polazeći od prvog kraja, dobija se skup grana $L_{s2} = \{6, 5, 13, 7, 1\}$, dok se polazeći od drugog kraja grane 14 dobija skup $L_{r2} = \{11, 10, 2, 1\}$, tako da se primenom relacije (2) dobija skup mogućih vrednosti za drugu upravljačku promenljivu t_{i2} , $L_{c2} = \{2, 5, 6, 7, 10, 11, 13, 14\}$. Da bi mreža ostala povezana potrebno je zatvoriti prekidač $t_{02} = 14$ i otvoriti prekidač na proizvoljno odabranoj grani iz skupa L_{c2} . Neka je za drugu upravljačku promenljivu proizvoljnim izborom odabrana šesta dozvoljena vrednost iz skupa L_{c2} , odnosno neka je usvojeno $t_{i2} = 11$. Nakon preorijentacije grana mreže opisani postupak je potrebno ponoviti i za poslednju upravljačku promenljivu. Elementi skupa L_{s3} su $\{9, 8, 7, 1\}$, dok su elementi skupa $L_{r3} = \{12, 14, 6, 5, 13, 7, 1\}$. Kombinovanjem ova dva skupa dobija se skup L_{c3} čiji su elementi $\{5, 6, 8, 9, 12, 13, 14, 15\}$. Zatvaranjem prekidača u grani 15 i izborom bilo koje dozvoljene vrednosti za treću upravljačku promenljivu kreira se povezana radialna mreža na kojoj se mogu izvršiti dalji proračuni optimizacije.

Kao što je prikazano, upotrebom koncepta adaptivnih petlji neophodno je kreirati novi skup dozvoljenih elemenata za svaku upravljačku promenljivu. Ova dodatna pretraga usporava postupak nalaženja početne populacije, ali se ušteda u pretraživanju ostvaruje time što je svako generisano rešenje izvodljivo, dok se primenom koncepta osnovnih petlji za pojedine vektore upravljačkih promenljivih, odnosno jedinice populacije, više puta prolazi kroz proizvoljnu dodelu vrednosti u procesu kreiranja početne populacije uz odbacivanje generisanih neizvodljivih rešenja.

Ispravka neizvodljivih rešenja

Celobrojni i/ili binarni operatori DE (mutacija i ukrštanje) ili bilo kog drugog meta-heurističkog optimizacionog algoritma mogu da formiraju neizvodljiva rešenja. Da se takva rešenja ne bi odbacivala, potrebno ih je korigovati i u što većoj meri zadržati postavke primenjenog optimizacionog algoritma. Da bi se to ostvarilo u maksimalno



Slika 1. Kreiranje početne populacije: a) primer distributivne mreže, b) adaptivna petlja za drugu upravljačku promenljivu, c) adaptivna petlja za treću upravljačku promenljivu

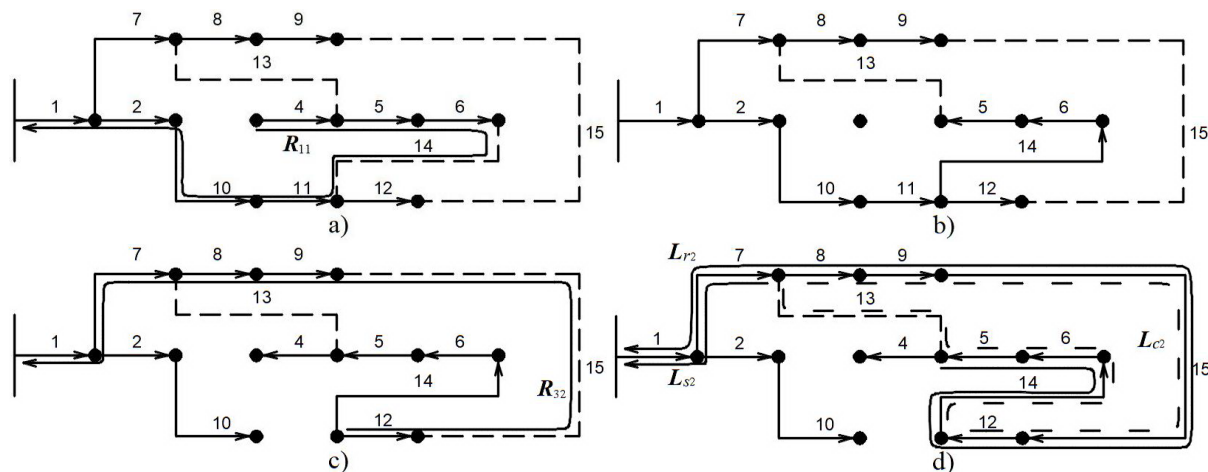
mogućoj meri, pre korekcije je potrebno proveriti koji su elementi vektora rešenja dobri, a koji loši. Za ovu svrhu se ponovo koristi koncept adaptivnih petlji. U nastavku teksta će biti opisan algoritam provere generisanog rešenja $u_i = [u_{i1}, u_{i2}, \dots, u_{ik}, \dots, u_{in}]$ koje se realizuje primenom usvojenih operatora pretrage.

Kao bazno stanje u_0 za proveru izvodljivosti rešenja u_i , na osnovu kog se definiše referentni orijentisani graf, koristi se rešenje roditelja x_{is} odnosno rešenje na osnovu kog je primenom meta-heurističkog optimizacionog algoritma dobijeno ispitivano rešenje. Iterativni postupak ponovo podrazumeva sukcesivnu analizu svake upravljačke promenljive.

- 1) Ako je k -ta upravljačka promenljiva identična nekoj već razmatranoj, takva upravljačka promenljiva se proglašava neprihvatljivom i zahteva se njena zamena, u suprotnom prelazi se na sledeći korak.
- 2) Upotrebom operatora „ \rightarrow ” u vektore putanja R_{km} se upisuju redom grane „nizvodne“ putanje posmatrano od napajanog čvora razmatrane grane. U ovom slučaju, zbog račvanja mreže, postoji više mogućih putanja R_{km} , koje je potrebno ispitati:
 - a. Ako se na kraju bilo koje od putanja upotrebom operatora „ \rightarrow ” dobijena incidentna grana koja odgovara nekoj upravljačkoj promenljivoj j iz vektora u_0 koja odgovara do tada nezatvorenoj grani, pretraga se nastavlja počev od grane u_{0j} upotrebom operatora „ \leftarrow ”. Ako se bilo koja tako formirana putanja završava napojnim čvorom mreže, upravljačka promenljiva se proglašava prihvatljivom, u suprotnom se takva promenljiva proglašava za neprihvatljivu. Kad god se primenom koncepta adaptivnih petlji formira izvodljivo rešenje, neophodno je otvoriti prekidač u_{ik} i zatvoriti prekidač u_{0j} , i izvršiti orijentaciju grafa.
 - b. Ako se na kraju nizvodnog puta, koji je kreiran upotrebom operatora „ \rightarrow ”, ne nalazi neka od grana iz vektora u_0 , takva promenljiva se proglašava za neprihvatljivu.
- 3) Postupak se ponavlja dok se ne ispitaju sve upravljačke promenljive razmatranog rešenja.
- 4) U slučaju postojanja neprihvatljivih promenljivih, njihova zamena se vrši primenom postupka za kreiranje početnog rešenja samo za te promenljive, polazeći od stanja dobijenog nakon provere ispravnosti rešenja završenog korakom 4).

Opisani postupak je i u ovom slučaju najlakše razumeti pomoću primera. Neka je kao rezultat operacija odabranog meta-heurističkog algoritma, primenjen na stanje $u_0 = t_0 = [13, 14, 15]$ kreiran vektor $u_i = [3, 4, 11]$. Kao što se iz ranijih razmatranja može videti, ova kombinacija isključivanja grana je neprihvatljiva. Proces provere započinje sa prvom upravljačkom promenljivom $u_{i1} = 3$. Pošto u ovom rešenju nema identičnih promenljivih provera za svaku promenljivu se svodi na izvršavanje koraka 2) predloženog algoritma. Upotrebom operatora „ \rightarrow ” se popunjavaju vektori putanja R_{km} . U ovom slučaju to su vektori $R_{11} = [4, 5, 6]$, i $R_{12} = [4]$. Kraju putanje određene vektorom R_{11} incidentna je grana $u_{01} = 14$, a putanji R_{12} , $u_{02} = 13$. Nastavljanjem sa pretragom počev od grane u_{02} (proizvoljno izabrano) upotrebom operatora „ \leftarrow ”, vektor R_{12} dobija konačni oblik $R_{11} = [4, 5, 6, 14, 11, 10, 2, 1]$, slika 2 a), odnosno završava se referentnim napojnim čvorom 1, što je dovoljno da se promenljiva u_{i1} proglaši prihvatljivom. Postupak se nastavlja otvaranjem prekidača grane 3 uz zatvaranje prekidača grane 14, i preorientacijom grafa mreže.

Nakon ovoga je moguće preći na proveru ispravnosti druge upravljačke promenljive $u_{i2} = 4$. Pokazuje se da ne postoji ni jedna putanja koja polazi iz napajanog čvora grane 4, slika 2 b), odnosno da se na njenom kraju ne nalazi grana iz vektora u_0 , pa se ova promenljiva proglašava neprihvatljivom i prelazi na proveru poslednje upravljačke promenljive. Vektori dobijeni upotrebom operatora „ \rightarrow ” su sada $R_{31} = [14, 6, 5]$, čija se putanja dalje nastavlja granom $u_{01} = 13$ i $R_{12} = [12]$, koja se nastavlja granom $u_{03} = 15$. Nastavljanjem pretrage počev od grane u_{03} (proizvoljno izabrano) upotrebom operatora „ \leftarrow ”, vektor R_{32} dobija konačni oblik $R_{32} = [12, 15, 9, 8, 7, 1]$, slika 2 c), odnosno završava se u referentnom čvoru, što i ovu promenljivu čini prihvatljivom.



Slika 2. Provera generisanog rešenja: a) provera prve promenljive, b) provera druge promenljive, c) provera treće promenljive, d) korekcija neizvodljive promenljive

Nakon otvaranja prekidača grane 11 uz zatvaranje prekidača grane 15 i odgovarajuće preorijentacije grafa mreže, može se pristupiti zameni promenljive u_{i2} . Polazeći od krajeva grane 13 (odgovara u ovom slučaju jedinoj ne zamenjenoj promenljivoj vektora \mathbf{u}_0), dobijaju se skupovi $L_{s2} = \{7, 1\}$ i $L_{r2} = \{5, 6, 14, 12, 15, 9, 8, 7, 1\}$, što primenom relacije (2) daje skup kandidata za otvaranje prekidača $L_{c2} = \{5, 6, 8, 9, 12, 13, 14, 15\}$, slika 2 d). Pomoću generatora slučajnog celog broja bira se grana koja će se otvoriti. Ako bi se generatorom slučajnih brojeva dobila vrednost 2, odnosno izabrao drugi element skupa, vektor \mathbf{u}_i bi bio [3, 6, 11].

DIFERENCIJALNA EVOLUCIJA

Price i Storm (17) su kreirali pouzdan i lako primenljiv optimizacioni algoritam pod nazivom diferencijalna evolucija (*Differential Evolution*). Kao i sve evolucione metode i DE radi sa populacijom vektora koji predstavljaju potencijalna rešenja posmatranog problema. Ovaj optimizacioni metod ima samo nekoliko kontrolnih promenljivih (mutacioni faktor i faktor ukrstanja) koji mogu da ostanu nepromenjeni tokom celog procesa optimizacije. Kako klasičan DE algoritam radi sa realnim brojevima, za problem koji se razmatra u ovom radu potrebno je raditi sa njegovom modifikovanom verzijom (14) koja radi sa celim brojevima. Osnovni operatori ovog optimizacionog algoritma su: inicijalizacija, mutacija, ukrstanje, selekcija i evaluacija.

Upravljačke promenljive se odmah kodiraju kao celobroje unutar odgovarajućih granica. U procesu kreiranja početne populacije formira se N_p proizvoljnih rešenja (jedinke) koje bi trebalo da ravnomerno prekriju kompletan prostor pretrage. Pomoću sledećeg izraza je moguće kreirati elemente početne populacije

$$\mathbf{X}_i(0) = \text{round}(\mathbf{X}_i^L + r \times (\mathbf{X}_i^U - \mathbf{X}_i^L)), \quad i = 1, \dots, N_p, \quad (3)$$

gde je r proizvoljan broj iz opsega $[0,1]$, \mathbf{X}^L i \mathbf{X}^U su vektori donjih i gornjih granica upravljačkih promenljivih, i round je funkcija koja realni broj zaokružuje na najbliži ceo broj.

Mutirani vektor se kreira upotrebom sledećeg izraza:

$$\mathbf{V}_i(t+1) = \mathbf{X}_{r1}(t) + F \times (\mathbf{X}_{r2}(t) - \mathbf{X}_{r3}(t)), \quad i = 1, \dots, N_p, \quad (4)$$

gde su $r_1, r_2, r_3 \in [1, \dots, N_p]$ proizvoljni celi međusobno različiti brojevi. Faktor mutacije, $F \in [0,1]$, je konstantan i zadaje ga korisnik pre početka proračuna.

Kada raznolikost populacije postane previše mala, novonastali kandidati će se sve više grupisati i neće biti u mogućnosti da dodatno unaprede vrednost razmatrane kriterijumske funkcije. Ovo može prouzrokovati preuranjenu konvergenciju. U svrhu povećavanja raznolikosti populacije uveden je operator ukrstanja. Operacija ukrstanja se izvodi pomoću sledećeg izraza:

$$u_{ij}(t+1) = \begin{cases} x_{ij}(t+1) & r_m > C_r, \\ v_{ij}(t+1) & r_m \leq C_r \vee j = j_{rand}. \end{cases} \quad (5)$$

Gde je C_r faktor ukrstanja ($C_r \in [0,1]$) koji se postavlja na početku procesa optimizacije i ostaje nepromenjen tokom celog proračuna, j_{rand} proizvoljno odabrani parametar iz opsega $[1,n]$, r_m proizvoljan broj iz opsega $[0,1]$. Proces evaluacije se svodi na „nokaut“ selekciju, tj. poređenje kreirane jedinke sa roditeljem (\mathbf{x}_{r1}). U slučaju da je novoformirana jedinka bolja od roditelja ona će zauzeti mesto roditelja u novonastaloj populaciji, i obrnuto. Opisani postupak se nastavlja sve dok se ne zadovolji kriterijum konvergencije.

ALGORITAM

Predloženi optimizacioni algoritam se može opisati sledećim koracima.

- 1 Učitavanje svih relevantnih podataka i podešavanje parametara optimizacionog algoritma.
- 2 **while** $k \leq N_p$ **do**
- 3 Kreiranje k -tog vektora početne populacije primenom koncepta adaptivnih petlji
- 4 Proračun tokova snaga
- 5 Provera ograničenja promenljivih stanja sistema
- 6 **if** ograničenja = 1 **then**
- 7 Izračunavanje vrednosti kriterijumske funkcije k -tog vektora
- 8 $k = k + 1$
- 9 **end if**
- 10 **end while**
- 11 uslov = 1, broj iteracija = 0
- 12 **while** uslov **do**

```

13 broj iteracija = broj iteracija + 1
14 for  $k = 1$  to  $N_p$  do
15     Primenom jednačine (4) kreirati mutirane vektore  $V_k$ 
16 end for
17 for  $k = 1$  to  $N_p$  do
18     Primenom jednačine (5) kreirati ciljne vektore  $U_k$ 
19 end for
20 for  $k = 1$  to  $N_p$  do
21     Provera izvodljivosti ciljnog vektora upotrebom predloženog algoritma
22     if  $U_k$  neizvodljiv then
23         Popravka ciljnog vektora  $U_k$  primenom predloženog algoritma
24     end if
25     Proračun tokova snaga
26     Provera ograničenja
27     Izračunavanje vrednosti kriterijumske funkcije vektora  $U_k$ 
28     if ograničenja =  $0 \vee (f(X_k) < f(U_k))$  then
29         U narednu populaciju prolazi  $X_k$ 
30     else if ograničenja =  $1 \wedge (f(U_k) < f(X_k))$ 
31         U narednu populaciju prolazi  $U_k$ 
32     end if
33 end for
34 if konvergencija = T then
35     uslov = 0
36 end if
37 end while
38 Ispis najboljeg vektora

```

Za kriterijume konvergencije su korišćena tri uslova. Prvi uslov se odnosi na maksimalan broj iteracija. Drugi uslov posmatra samu populaciju. Kada Euklidovo rastojanje upravljačkih promenljivih, koje se računa između najboljeg vektora i svakog drugog vektora, padne ispod unapred zadate granične vrednosti, algoritam se zaustavlja. Za treći kriterijum konvergencije je odabrano ponavljanje najboljeg rešenja.

REZULTATI PRORAČUNA

Za testiranje predloženog algoritma korišćena je standardna IEEE 33 test distributivna mreža, koja je prikazana na slici 3. Razmatrana test mreža ima ukupno 33 čvora i 5 petlji. Ukupna potrošnja aktivne i reaktivne snage je 3.715 MW i 2.3 MVar, respektivno.

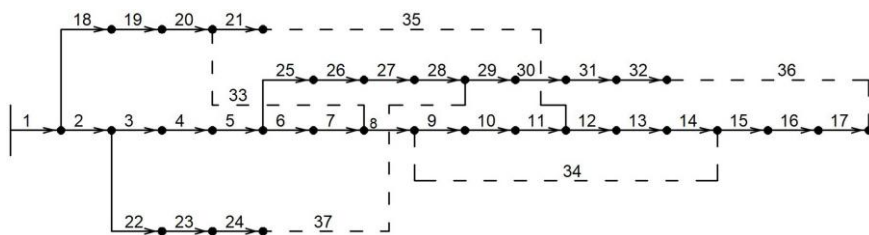
U optimizacionim proračunima, za broj članova populacije usvojena je vrednost $N_p = 50$. Sledeće vrednosti su usvojene za parametre optimizacionog algoritma: $F = 0.9$, $C_r = 0.8$. Za prvi kriterijum konvergencije odabrana je vrednost od 100 iteracija. Za Euklidovo rastojanje korišćena je vrednost od 0.01. Za treći kriterijum konvergencije odabrana je vrednost od 40 iteracija.

Za bazno stanje, prikazano na slici 3, pri naponu referentnog čvora 1 od $V_{ref} = 12.66$ kV, gubici u posmatranoj distributivnoj mreži su 202.68 kW. U tabeli 1 su prikazani uporedni rezultati predloženog koncepta i koncepta osnovnih petlji. U drugoj koloni su prikazane vrednosti upravljačkih promenljivih, tj. redni brojevi grana koje je potrebno otvoriti da bi se dobila optimalna vrednost razmatrane kriterijumske funkcije. Vrednost kriterijumske funkcije je prikazana u trećoj koloni, dok je u četvrtoj koloni prikazan potreban broj iteracija. Peta kolona prikazuje vreme koje je potrebno da se izvrši optimizacioni algoritam.

Proračun tokova snaga je najzahtevniji deo proračuna kada se primenjuje koncept osnovnih petlji. Za jedan proračun tokova snaga potrebno je 0.1 s računarskog vremena, međutim zbog generisanja velikog broja neizvodljivih rešenja broj ovih proračuna drastično raste. Od ukupnog vremena koje je prikazano u tabeli 1, kada se primeni koncept osnovnih petlji, 88% odlazi na proračun tokova snaga. Kao što se iz tabele može videti, predloženi koncept je našao bolje rešenje za manji broj iteracija, ali i za приметно kraće vreme. Dodatna uporedjivanja su vršena sa rešenjima koja su pronađena u literaturi, tabela 1. Prilikom provere svih rešenja korišćena je nominalna vrednost napona napojnog čvora.

Kao što se iz rezultata može zaključiti primenom koncepta osnovnih petlji za jedno bazno stanje mreže, što je najčešći slučaj u literaturi (6, 7, 8, 9, 17), ne vrši se kompletna pretraga prostora rešenja i pronađeno rešenje najčešće predstavlja lokalni optimum. Primenom koncepta adaptivnih petlji ova pretraga se proširuje i pronalaze se bolja rešenja. Druga prednost predloženog algoritma je značajno manji broj iteracija. Na slici 4 je prikazano

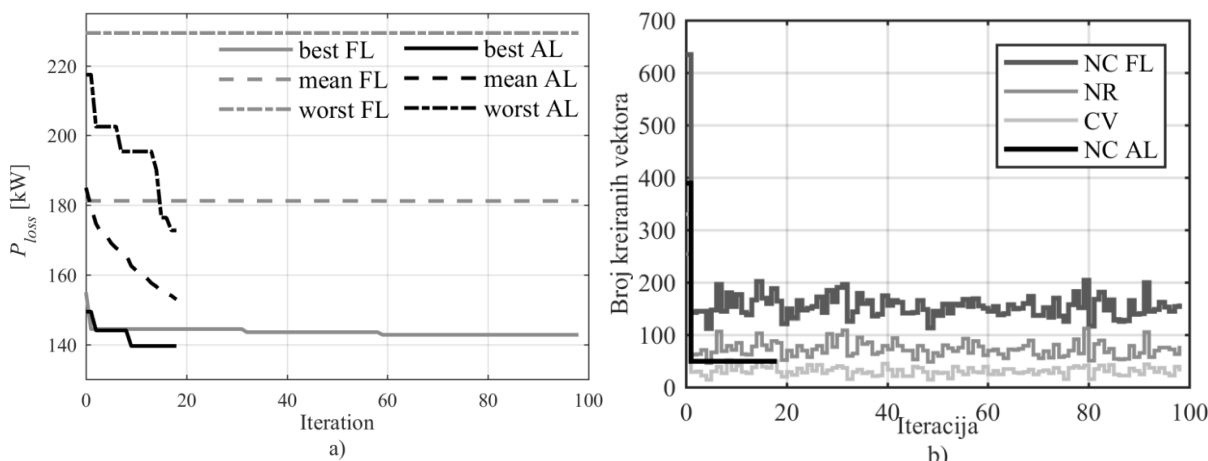
poređenje oba koncepta. Punim linijama je prikazana promena najboljeg rešenja tokom iteracija, isprekidana linija prikazuje promenu srednje vrednosti kriterijumske funkcije cele populacije sa promenom iteracija, dok je



Slika 3. IEEE 33 test mreža

Tabela 1. Upoređivanje rezultata različitih metoda

Metoda	Otvorena grana					P_γ [kW]	NI	Vreme [s]
Osnovne petlje	37	7	34	11	32	142.7588	98	626.6
Adaptivne petlje	37	7	14	9	32	139.5513	18	313.7
Ahmadi (18)	7	9	14	28	32	139.9781	/	/
Souza (19)	7	9	14	32	37	139.5513	/	/
Baran (20)	6	14	7	32	28	163.5722	/	/



Slika 4. Upoređivanje metoda: a) unapređivanje populacije, b) broj kreiranih rešenja.

sa crta tačka crta linijom prikazana promena najgoreg rešenja tokom iteracija. Kao što se sa slike 4 a) može videti srednja vrednost cele populacije i najgore rešenje kada se primenjuje koncept osnovnih petlji se beznačajno menja tokom celog optimizacionog procesa. Ovo praktično znači da je generisan veliki broj rešenja koja su bila gora do već postojećih rešenja, tj. da se populacija veoma sporo menja tokom iteracija. Sa druge strane ovo nije slučaj kada se primenjuje koncept adaptivnih petlji.

Pored toga, primenom osnovnih petlji se generiše daleko veći broj rešenja koja su neprihvatljiva. Na slici 4 b) je prikazan ukupan broj kreiranih jedinki po populacijama primenom obe metode. Sa NC FL je predstavljen ukupan broj formiranih rešenja kod pristupa osnovnih petlji, NR predstavlja broj rešenja kod kojih konfiguracija nije bila radijalna, CV predstavlja broj rešenja koja su narušila neka ograničenja promenljivih stanja sistema, dok je sa NC AL predstavljen ukupan broj formiranih rešenja kod pristupa adaptivnih petlji. Kod oba pristupa najveći broj generisanih vektora se javlja prilikom formiranja početne populacije. Međutim kod koncepta adaptivnih petlji taj broj je skoro dva puta manji. Razlog za ovakvo poboljšanje je činjenica da primenom koncepta adaptivnih petlji nije moguće kreirati konfiguraciju koja ne zadovoljava kriterijum radijalnosti, dok kod koncepta osnovnih petlji to nije slučaj. Ono na šta nije moglo da se utiče je generisanje rešenja koja narušavaju neko ograničenje vezane promenljive stanja, i na početnoj populaciji ovaj broj je približno jednak bez obzira na odabrani koncept.

Velika brzina u nalaženju optimalnog rešenja u slučaju primene adaptivnih petlji je direktna posledica primene predloženog algoritma korekcije neispravnih rešenja. Ako se primenom operacija odabranog optimizacionog algoritma dobije neprihvatljivo rešenje u pogledu radijalnosti i povezanosti mreže ono se koriguje i nije ga potrebno odbacivati. Upravo zbog toga svaki član populacije se obrađuje samo jedanput što nije bio slučaj kod algoritama koji su koristili pristup osnovnih petlji ili binarno kodiranje grana.

ZAKLJUČAK

U ovom radu je predstavljen koncept adaptivnih petlji za kodiranje grana distributivne mreže kod rešavanja optimalne rekonfiguracije distributivne mreže. Kod koncepta adaptivnih petlji svakoj upravljačkoj promenljivoj se sukcesivno kreira dozvoljeni skup rešenja koje može da izabere. Upravo zbog toga izbegava se kreiranje neizvodljive (neradikalne i nepovezane) konfiguracije. Pored upotrebe pri kreiranju početne populacije predložen je algoritam kojim se primenom adaptivnih petlji vrši provera i ispravka neizvodljivih rešenja koja su kreirana operatorima meta-heurističkog algoritma optimizacije. Testiranje karakteristika predloženog koncepta izvršeno je u proračunima na standardnoj IEEE 33 test distributivnoj mreži. Predloženi koncept je pokazao bolje osobine u odnosu na koncept osnovnih petlji u pogledu brzine proračuna i broju iteracija.

LITERATURA

1. López J C, Lavorato M, Franco J F, Rider M J, 2016, 'Robust optimization applied to the reconfiguration of distribution systems with reliability constraints', *IET Generation, Transmission & Distribution*, 10, (4), pp. 917–927
2. Quevedo P M, Contreras J, Rider M J, Allahdadian J, 2015, 'Contingency Assessment and Network Reconfiguration in Distribution Grids Including Wind Power and Energy Storage', *IEEE Transactions on Sustainable Energy*, 6, (4), pp. 1524–1533
3. Goswami S K, Basu S K, 1992, 'A new algorithm for the reconfiguration of distribution feeders for loss minimization', *IEEE Transactions on Power Delivery*, 7, (3), pp. 1484–1491
4. Shirmohammadi D, Hong H W, 1989, 'Reconfiguration of electric distribution networks for resistive line losses reduction', *IEEE Transactions on Power Delivery*, 4, (3), pp. 1492–1498
5. Bayat A, 2013, 'Uniform voltage distribution based constructive algorithm for optimal reconfiguration of electric distribution networks', *Electric Power Systems Research*, 104, pp. 146–155
6. Wu Y-K, Lee C-Y, Liu L-C, Tsai S-H, 2010, 'Study of Reconfiguration for the Distribution System With Distributed Generators', *IEEE Transactions on Power Delivery*, 25, (3), pp. 1678–1685
7. Sayadi F, Esmaili S, Keynia F, 2016, 'Feeder reconfiguration and capacitor allocation in the presence of non-linear loads using new P-PSO algorithm', *IET Generation, Transmission & Distribution*, 10, (10), pp. 2316–2326
8. Khorshid-Ghazani B, Seyedi H, Mohammadi-ivatloo B, et al., 2017, 'Reconfiguration of distribution networks considering coordination of the protective devices', *IET Generation, Transmission & Distribution*, 11, (1), pp. 82–92
9. Eldurssi A M, O'Connell R M, 2015, 'A Fast Nondominated Sorting Guided Genetic Algorithm for Multi-Objective Power Distribution System Reconfiguration Problem', *IEEE Transactions on Power Systems*, 30, (2), pp. 593–601
10. Asrari A, Lotfifard S, Payam M S, 2016, 'Pareto Dominance-Based Multiobjective Optimization Method for Distribution Network Reconfiguration', *IEEE Transactions on Smart Grid*, 7, (3), pp. 1401–1410
11. Mahboubi-Moghaddam E, Narimani M R, Khooban M H, et al., 2016, 'Multi-objective distribution feeder reconfiguration to improve transient stability, and minimize power loss and operation cost using an enhanced evolutionary algorithm at the presence of distributed generations', *Electrical Power and Energy Systems*, 76, pp. 35–43
12. Mendoza J, López R, Morales D, et al., 2006, 'Minimal Loss Reconfiguration Using Genetic Algorithms With Restricted Population and Addressed Operators: Real Application', *IEEE Transactions on Power Systems*, 21, (2), pp. 948–954
13. Huang Y-C, 2002, 'Enhanced genetic algorithm-based fuzzy multi-objective approach to distribution network reconfiguration', *IEEE Proceedings-Generation, Transmission and Distribution*, 149, (5), pp. 615–620
14. Krstivojević J, Šošić D, Savić A, 2016, 'Coordination of directional overcurrent relays by using a heuristic optimization method', *Proc. Int. Conf. Mediterranean Conference on Power Generation, Transmission, Distribution and Energy Conversion, Belgrade, Serbia*
15. Civanlar S, Grainger J J, Yin H, et al., 1988, 'Distribution feeder reconfiguration for loss reduction', *IEEE Transactions on Power Delivery*, 3, (3), pp. 1217–1223
16. Price K, Storm R, Lampinen J, 'Differential Evolution – A practical approach to global optimization', (Springer-Verlag Berlin Heidelberg, 2005, 1st edn.)
19. Li Z, Jazebi S, de León F, 2017, 'Determination of the Optimal Switching Frequency for Distribution System Reconfiguration', *IEEE Transactions on Power Delivery*, 32, pp. 2060–2069
20. Ahmadi H, Martí J R, 2015, 'Minimum-loss network reconfiguration: A minimum spanning tree problem', *Sustainable Energy, Grids and Networks*, 1, pp. 1–9

21. Souza S S, Romero R, Franco J F, 2015, 'Artificial immune networks Copt-aiNet and Opt-aiNet applied to the reconfiguration problem of radial electrical distribution systems', *Electric Power Systems Research*, 119, pp. 304–312
22. Baran M, Wu F, 1989, 'Network reconfiguration in distribution systems for loss reduction and load balancing', *IEEE Transactions on Power Delivery*, 4, (2), pp. 1401-1407