

# Particle Swarm Optimization algoritam sa primjerom primjene u upravljanju sistemom automatske regulacije napona

Amina Bralić

Odsjek za automatiku i elektroniku

Elektrotehički fakultet

Univerzitet u Sarajevu

Sarajevo, Bosna i Hercegovina

abralic1@etf.unsa.ba

**Abstract**—Pojam i problem regulacije u sistemima automatskog upravljanja, te pronalazak parametara PID regulatora koji bi osigurali optimalno ponašanje sistema uvijek su zahtijevali posebno inženjersko umijeće. Veliki broj naučnih radova bavi se ovom problematikom, a s ciljem da se ovaj proces učini jedinstvenim i primjenjivim na sve sisteme. U ovom radu primjenjuje se algoritam optimizacije rojem čestica (PSO algoritam) koji se koristi za dobivanje optimalnih parametara regulatora za sistem automatske regulacije napona. Algoritam je testiran na raznim kriterijalnim funkcijama, od koji su pojedine osmišljene i/ili modifikovane od strane autora. Utvrđeno je da ovakav algoritam za relativno mali broj iteracija i pogodno odabrane parametre u većini slučajeva daje optimalno rješenje ili rješenje koje je vrlo blizu optimalnog.

## I. UVOD

### A. Opis problema

Problem automatske regulacije javlja se u svim elektroenergetskim sistemima. Prilikom puštanja sistema u pogon za pravilno funkcionisanje istog potrebno je napon na trafostanicama i vodovima dalekovoda održati na zadanom nivou i pored smetnji i kvarova koji se javljaju na mreži. Naravno, potrebno je napraviti balans između aktivne i reaktivne snage na mreži, koji su u direktnoj vezi sa frekvencijom i naponom, respektivno [3].

Za regulaciju napona (samim tim i reaktivne snage) u EE sistemima koristi se automatski regulator napona koji djeluje u zatvorenom. Sistem regulacije napona (AVR sistem) je linearni SISO sistem zatvoren negativnom povratnom spregom [8]. Posmatranjem blokovskog prikaza na slici 1 možemo uočiti karakteristične dijelove sistema. Nakon sumatora na kojem se vrši oduzimanje referentnog signala od signala izlaza dobivamo signal greške. Ovaj signal biće osnova za formiranje funkcije kriterija jer se na osnovu njega procjenjuje kvaliteta upravljanja sistemom. Signal se dovodi na regulator, pri čemu se uobičajeno koristi linearni PID regulator, jer daje najbolje performanse. Blok PID regulatora je u serijskoj vezi sa blokom sistema upravljanja u širem smislu, koji se sastoji od pojačavača, ekscitatora i generatora. Senzorom se detektuje signal sa objekta upravljanja koji se spregom dovodi na sumator, te je na taj način sistem zatvoren.

PSO algoritam u ovom slučaju se primjenjuje u svrhu traženja optimalnih parametara PID regulatora, pri čemu

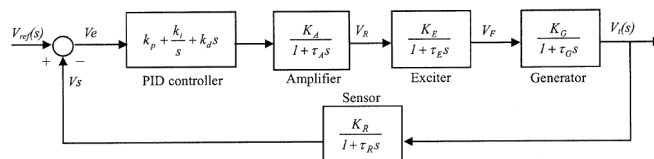


Fig. 1. Blok shema sistema automatske regulacije napona (preuzeto iz [8])

se može koristiti više različitih kriterija u ovisnosti šta je primarni cilj regulacije: brži odziv (kraće vrijeme smirivanja oko stacionarnog stanja i/ili kraće vrijeme porasta napona), minimalna vrijednost prvog preskoka, minimalno upravljanje, itd.

### B. Pregled literature vezane za problem automatske regulacije

Upravljanje sistemom, bilo linearnim ili nelinearnim, uvijek je predstavljalo svojevrsan izazov inženjerima, iako je tehnika i industrija tokom godina jako napredovala. S tim u vezi, nastoje se osmisliti brzi i efikasni algoritmi i metode traženja parametara regulatora radi preciznijeg i/ili robusnijeg upravljanja.

Najnapredniji algoritmi koje se danas koriste primjenjuju metode neuronskih mreža [5], adaptivnog upravljanja ili fuzzy logiku [4]. Međutim, ovakvi napredni algoritmi su rijetko zastupljeni, obzirom da je njihova implementacija kompleksna i da se u polju automatike primjenjuju tek odnedavno. Najzastupljeniji tip regulatora u industriji je ipak regulator proporcionalno-integralno-derivativnog dejstva ili skraćeno PID regulator. U prilog tome govori i činjenica da je PID regulator lahko implementirati, a jednačine koje ga opisuju su intuitivne i jednostavne za razumijevanje. Ipak, mana je u tome što je traženje optimalnih parametara PID regulatora mukotrpan posao.

Prva metoda kojom se ovaj proces pokušao unificirati je Ziegler-Nichols metoda [2], koja opet nije značajno olakšala postupak. Kao alternativa ovim pristupima dolaze do izražaja heuristički algoritmi čiji je zadatak da u konačnom vremenu i sa ograničenim resursima pronađu rješenje dovoljno blisko optimalnom rješenju. Neki od takvih algoritama jesu algoritam simuliranog hlađenja i genetički algoritam [6] koji, pored

svojih specifičnih mehanizama pretraživanja problemskog prostora u određenoj mjeri imaju i svojstva slučajnosti. Pokazuje se da ovakvi algoritmi uglavnom daju zadovoljavajuća rješenja problema upravljanja na koji se primjenjuju [8], [9], [10].

Još jedan heuristički algoritam o kome će biti riječi u ovom radu jeste i algoritam optimizacije rojenjem čestica (Particle Swarm Optimization algorithm). PSO algoritam se pojavljuje tek na kraju dvadesetog stoljeća, ali je svejedno njegov razvoj jako napredovao iako u vrlo kratkom vremenu. Takođe posjeduje svojstva slučajnosti (kao GA i SA algoritam), a spada među jednostavnije heurističke algoritme u pogledu implementacije. Ovaj algoritam takođe ima i više stepeni slobode, čijim pravilnim odabirom se mogu poboljšati njegove performanse.

Obzirom da je algoritam organiziran tako da simulira roj čestica koje istovremeno traže optimum, te da se pri tome kreću određenom brzinom, različite varijacije algoritma su upravo napravljene na temu računanja brzine čestice. Jedna od varijanti ovako modifikovanog PSO algoritma koristi metode mapiranja problemskog prostora (haotični PSO) ili skaliranja proračunate brzine faktorom koji ovisi od parametara algoritma [7]. Poređenjem ove dvije varijante, haotični algoritam osigurava odziv bez preskoka, ali nešto veće vrijeme kašnjenja odziva za razliku od skaliranog PSO. Što se tiče konvergencije, skalirana varijanta brže konvergira u odnosu na haotičnu. PSO sa ubrzanjem skalira samo jednu od tri komponente brzine, i to brzinu čestice iz prethodne iteracije. Ubrzani PSO mnogo brže konvergira u odnosu na haotični i skalirani, te je i vrijeme odziva kraće, ali zato se javlja izvjestan preskok u odzivu [11]. Ipak, postojanje i veličina prvog preskoka mogu se izbjeći pogodnim odabirom kriterija koji treba zadovoljiti odabrano rješenje, što otvara vrata za dalje modifikacije algoritma. Tako možemo koristiti funkcije kriterija koje podrazumijevaju integral (sumu) kvadrata greške [8], [9] ili apsolutne greške [8], te u kombinaciji sa varijablom vremena kao umnožak pod integralom ili sumom [8], [10]. Koriste se i funkcije kriterija koje osim greške odziva koriste i informaciju o prvom preskoku, vremenu porasta i/ili vremenu smirenja [8], itd.

Zaključujemo da se samo na osnovu ova dva segmenta PSO algoritma može napraviti više varijanti algoritma, a uzimajući u obzir i ostale stepene slobode, algoritam je moguće podesiti tako da dobiveno rješenje bude odgovarajuće u primjeni.

## II. PARTICLE SWARM OPTIMIZATION ALGORITAM

### A. Opis rada PSO algoritma

Particle Swarm Optimization algoritam je osmišljen 1995. godine od strane tri autora: Kennedy, Eberhart i Shi. Inspiracija za kreiranje ovakvog algoritma je pronađena u ponašanju životinja u velikim skupinama (roj, jato, itd.). Naime, kada se životinjske vrste kreću u velikim skupinama u potrazi za hranom imaju većeg izgleda za uspjeh. Takođe, prilikom napada grabežljivaca, veće su šanse da se jedinke zaštite kada su u skupini nego kada se kreću samostalno. Ukoliko neka

jedinica iz skupine pronađe put ka hrani, ona informiše ostale jedinke, koje bi se kretale u tom pravcu. Na taj način se jedinke usmjeravaju ka područjima koja obećavaju uspjeh, pa tako mogu pronaći i druga mjesta na kojima ima hrane.

PSO algoritam na izvjestan način imitira ponašanje jedinki u roju. Kao prvo, algoritam ima veliki broj jedinki koje čine populaciju. Kretanje svake jedinke je složeno i određeno je pozicijom i brzinom. Brzina kretanja se sastoji od determinističke i stohastičke komponente. Deterministička komponenta je vezana za tekuću najbolju poziciju (poziciju koja daje najbolju vrijednost kriterija), a stohastičke komponente su vezane za tekuću brzinu i tekuću najbolju poziciju najboljeg informanta [1], [11]. Informanti predstavljaju jedinke koje omogućavaju da se korisna informacija o problemskom prostoru prenese do svake jedinke u roju. Na slici 2 prikazani su mogući smjerovi kretanja čestice  $i$  koji zavise od globalnog rješenja  $g^*$  i najboljeg rješenja  $x_i^*$  za česticu  $i$ .

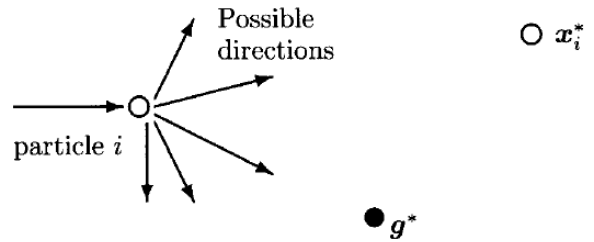


Fig. 2. Shematski prikaz kretanja čestice roja u PSO algoritmu, pri čemu je  $g^*$  globalno rješenje, a  $x_i^*$  najbolje rješenje za svaku pojedinačnu česticu (preuzeto iz [11])

Standardni PSO algoritam koristi relacije (1) i (2) za ažuriranje pozicije i brzine čestice respektivno [1], [11] ( $i$  je indeks čestice, a  $d$  razmatrana dimenzija):

$$x_{i,d}(n+1) = x_{i,d}(n) + v_{i,d}(n+1) \quad (1)$$

$$v_{i,d}(n+1) = c_0 \cdot v_{i,d}(n) + c_1 \cdot \epsilon_1 \cdot [particle\_best_{i,d} - x_{i,d}(n)] + c_2 \cdot \epsilon_2 \cdot [global\_best(n) - x_{i,d}(n)] \quad (2)$$

gdje  $\epsilon_1$  i  $\epsilon_2$  uzimaju nasumične vrijednosti između 0 i 1, ali sa uniformnom raspodjelom, a koeficijenti  $c_0$ ,  $c_1$  i  $c_2$  utiču na ubrzanje napredovanja algoritma. Ovi koeficijenti predstavljaju parametre algoritma i obično se  $c_1$  i  $c_2$  postavljaju na vrijednosti 2. U slučaju da su vrijednosti  $c_1$  i  $c_2$  jednake nuli, čestica se kreće po istom pravcu na svim iteracijama. Ukoliko se ne uzima u obzir trenutna brzina i jedan od parametara  $c_1$  ili  $c_2$  je nenulti, a drugi jednak nuli, tada se gubi na raznolikosti roja i dolazi do stagniranja ili preuranjene konvergencije algoritma. Takođe, parametri algoritma su i veličina roja, te broj informanata.

PSO algoritam ima sličnosti sa genetičkim algoritmom, obzirom na to da i ovdje imamo više jedinki koje čine populaciju kao kod GA. Međutim, razlika je u tome što broj čestica ovdje nije toliko veliki, već je optimalan broj između 20 i 40 jedinki. Ono po čemu se, između ostalog,

PSO algoritam razlikuje od drugih algoritama jeste postojanje informanata. Informanata može najviše biti onoliko koliko ima čestica u roju. Tokom razvijanja ovog algoritma i različitih istraživanja pokazalo se da je taj broj prevelik, te da je dovoljno uzeti od 3 do 5 informanata [1].

Organizacija PSO algoritma prikazana je na slici 3. Prilikom inicijalizacije početnih brzina čestica, preporučuje se da se brzine inicijaliziraju na slučajne vrijednosti u okviru nekog konačnog opsega, da bi se onemogućilo da se odmah raspše u problemskog prostoru. Informanti se nasumično biraju, nakon čega se vrši ažuriranje brzine i pozicije svake čestice, na prethodno opisani način. Evaluacijom kriterija se odaberu najbolje pozicije čestica, a nakon toga se provjeravaju uslovi zaustavljanja. Obično se koristi maksimalan broj iteracija i/ili mjera napredovanja u prostoru vrijednosti.

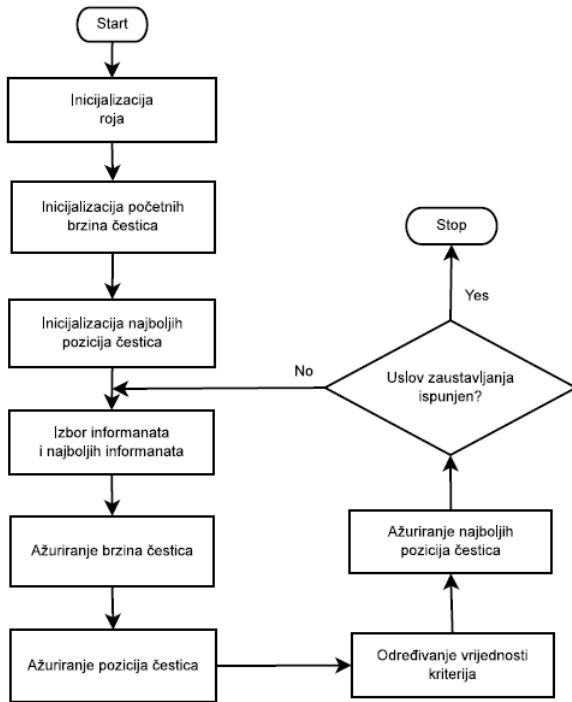


Fig. 3. Organizacija osnovnog algoritma PSO (preuzeto iz [1])

### B. Svođenje problema automatske regulacije napona u formu PSO algoritma

Blokovski prikaz AVR sistema predstavljen u prethodnom poglavlju (slika 1) sadrži opće parametre. Na slici 4 prikazane su vrijednosti pojedinih parametara (pojačanja i vremenskih konstanti) za gradivne elemente korištenog sistema.

Odziv sistema bez regulatora je kvazioscillatoran i kao takav može uzrokovati mnoge probleme unutar EE sistema. Na slici 5 je prikazan odziv AVR sistema u zatvorenom, ali bez PID bloka. Odziv ne prati zadanu vrijednost, prvi preskok je prevelike vrijednosti, te je vrijeme smirivanja jako veliko. Ovakve oscilacije mogu sistem učiniti nestabilnim (u najboljem slučaju dovesti na granicu stabilnosti).

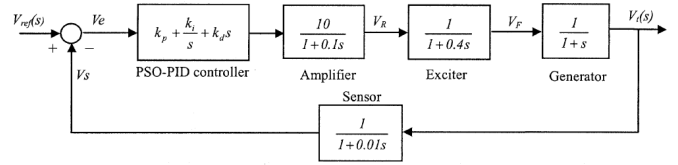


Fig. 4. Blok shema sistema automatske regulacije napona (preuzeto iz [8])

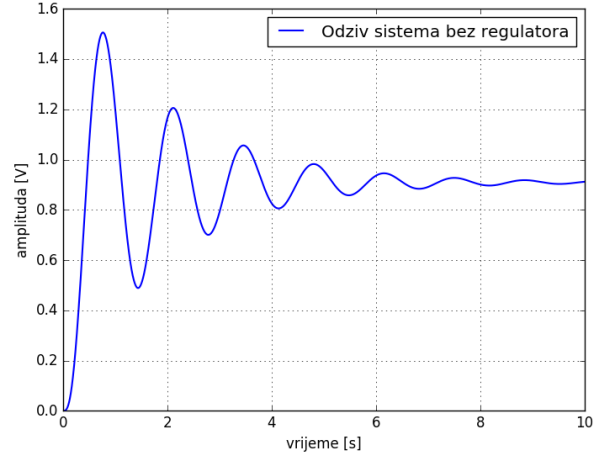


Fig. 5. Odziv sistema zatvorenog negativnom povratnom spregom bez regulatora

Već je napomenuto da PSO algoritmom tražimo parametre PID regulatora koji će omogućiti optimalan odziv sistema na pobudu. U tom slučaju problemski prostor ima tri dimenzije i to:  $K_p$ ,  $K_i$  i  $K_d$ . Ovi parametri regulatora su u PSO zapravo koordinate pozicije čestice. Ograničenja koja treba uvažiti tokom pretraživanja jesu sljedeća [8]:

$$\begin{aligned} 0 &\leq K_p \leq 1.5 \\ 0 &\leq K_i \leq 1.0 \\ 0 &\leq K_d \leq 1.0 \end{aligned} \quad (3)$$

Uzeta su u obzir na način da se prilikom ažuriranja pozicije čestice provjere navedena ograničenja. Prilikom inicijalizacije čestica takođe se vodi računa o ovim ograničenjima, da algoritam ne bi generisao nelegalna rješenja. Prema relaciji (1) se ažurira pozicija čestice. Inicijalne brzine čestica se generišu u opsegu od  $-0.5$  do  $0.5$ , a u narednim iteracijama se ažuriraju prema relaciji (2). Za parametre  $c_1$  i  $c_2$  su uzete preporučene vrijednosti. Veličina roja je varirana u opsegu od 30 do 100 čestica, a maksimalan broj iteracija od 50 do 80.

## III. SIMULACIJSKI REZULTATI

### A. Postavka simulacija

Nakon što je definisan problem, te izvršeno svođenje problema regulacije u formu PSO algoritma, slijedi implementacija. Algoritam koji rješava zadani problem implementiran je u programskom jeziku *Python*, uz korištenje biblioteka *control* i *scipy*. Funkcija koja implementira PSO algoritam,

pored veličine populacije, maksimalnog broja iteracija, broja dimenzija problemskog prostora, te parametara  $c_1$  i  $c_2$ , prima i prenosnu funkciju (Laplaceovu transformaciju) sistema u otvorenom. Prenosnu funkciju prosljeđenu ovoj funkciji čini serijska veza pojačavača, ekscitatora, generatora i senzora. Rezultat pretrage je vektor parametara PID regulatora.

### B. Rezultati simulacija

Prilikom testiranja algoritma bilo je potrebno postaviti parametre algoritma na određene vrijednosti. Obzirom da broj čestica direktno utiče na uzorkovanje problemskog prostora za veličinu roja odabrano je 100 čestica. Broj iteracija je postavljen na 50, a parametri  $c_1$  i  $c_2$  su postavljeni na preporučene vrijednosti 2. Parametar  $c_0$  je postavljen na 0.5. Prvobitno je ova vrijednost iznosila 1, ali uočena su značajna poboljšanja prilikom promjene vrijednosti na 0.5, tako da je ista vrijednost i zadržana. Ispitivanja su izvršena za 10 različitih kriterijalnih funkcija, a u nastavku su navedene korištene funkcije, kao i odgovarajući dijagrami. Za referentnu vrijednost odabran je napon od 1 volt, a odziv se posmatra unutar vremena od 10 sekundi. Svaki dobiveni odziv se posmatra u odnosu na odziv određen setom parametara iz [10]. Oznakom  $e(i)$  označena je vrijednost greške, odnosno razlika ulaza u sistem i izlaza iz sistema (vrijednosti koja povratnom spregom dolazi na sumator sa znakom minus). Oznaka  $OS$  predstavlja vrijednost prvog preskoka u sistemu. Pod pretpostavkom da vektori izlaza i ulaza imaju  $N$  elemenata prva kriterijalna funkcija je oblika:

$$J = \sum_{i=0}^N (i+1) \cdot e(i) + 10 \cdot OS \quad (4)$$

Rezultati dobiveni algoritmom prikazani su u tabeli I:

TABLE I  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (4))

$K_p$	0.92808926
$K_i$	0.86329009
$K_d$	0.33425985

Veličina prvog preskoka je u opsegu dozvoljenih granica (+20% od referentne vrijednosti), vrijeme uspona i vrijeme smirenja su takođe jako povoljni, te odziv dostiže referentnu vrijednost (slika 6).

Druga funkcija kriterija je oblika:

$$J = \sum_{i=0}^N (i+1) \cdot e(i) \quad (5)$$

Rezultati dobiveni algoritmom prikazani su u tabeli II:

Analizom ovakvog odziva (slika 7) zaključujemo da je u odnosu na prethodni odziv, vrijeme porasta puno manje. S druge strane ovo plaćamo neznatno većom vrijednošću prvog preskoka, ali i dužim vremenom smirenja (obzirom da postoji podbačaj u odzivu).

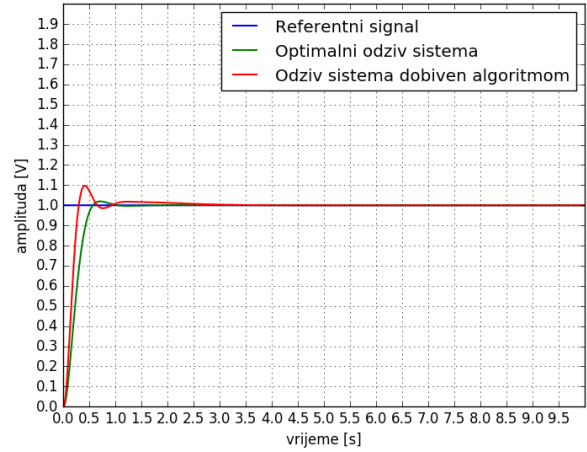


Fig. 6. Odziv sistema prema parametrima PID regulatora I

TABLE II  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (5))

$K_p$	0.66175838
$K_i$	0.53608159
$K_d$	0.73166126

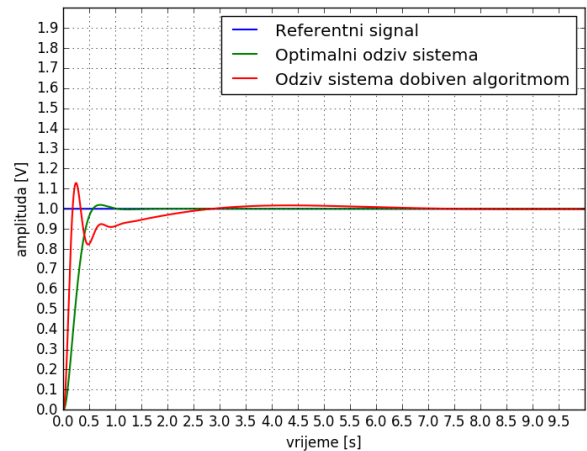


Fig. 7. Odziv sistema prema parametrima PID regulatora II

Treća funkcija kriterija je suma apsolutnih vrijednosti signala greške u odgovarajućim vremenskim trenucima:

$$J = \sum_{i=0}^N |e(i)| \quad (6)$$

Rezultati dobiveni algoritmom prikazani su u tabeli III:

Kao što se na slici 8 može vidjeti, ovdje je vrijeme smirenja jako dugo, obzirom da se referentna vrijednost ne dostiže čak ni za 10 sekundi, već postoji malo odstupanje. Vrijeme porasta je malo, prvi preskok je i dalje u dozvoljenom opsegu, ali vrijeme smirenja je neprihvatljivo veliko.

TABLE III  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA  
FUNKCIJA (6))

$K_p$	0.95085358
$K_i$	0.19173412
$K_d$	0.67655164

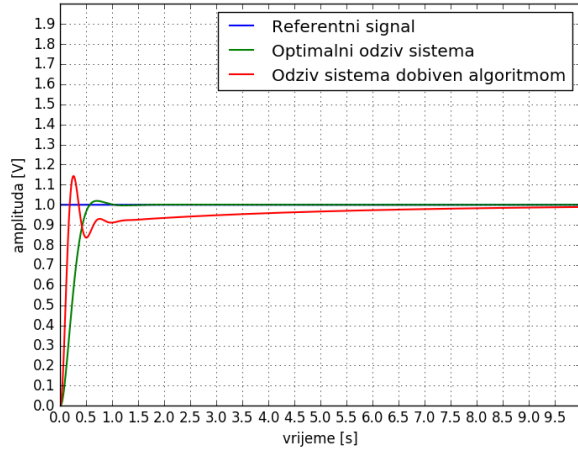


Fig. 8. Odziv sistema prema parametrima PID regulatora III

Za četvrtu funkciju kriterija odabrana je suma kvadrata greške u određenim vremenskim trenucima:

$$J = \sum_{i=0}^N [e(i)]^2 \quad (7)$$

Rezultati dobiveni algoritmom prikazani su u tabeli IV:

TABLE IV  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA  
FUNKCIJA (7))

$K_p$	0.7552782
$K_i$	0.93381122
$K_d$	0.41435291

Primjećuje se da je u ovom slučaju najmanja vrijednost prvog preskoka, mada sistem ima jednu kvazioscilaciju prije nego dosegne ravnotežno stanje, što rezultira većim vremenom smirenja. Ova blaga oscilacija u odzivu bi mogla predstavljati problem, pa bi prije implementacije ovih parametara trebalo ispitati odziv na brze promjene ulaza u sistem (slika 9).

Funkcije koje se često koriste kao kriterijalne podrazumijevaju i integriranje funkcije greške u raznim oblicima, npr. integriranje kvadrata greške [10] (Integral Square Error - ISE):

$$J = \int_0^T (e(t))^2 dt \quad (8)$$

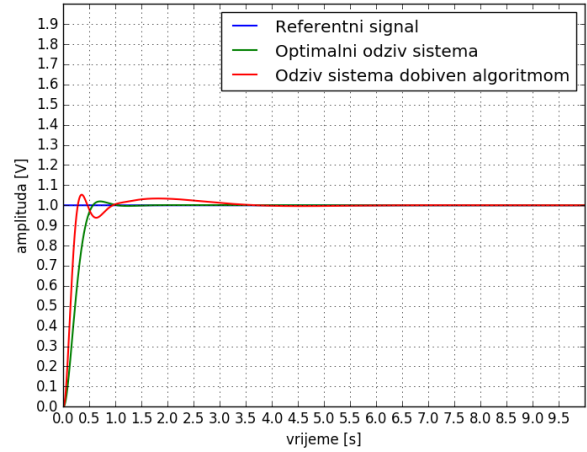


Fig. 9. Odziv sistema prema parametrima PID regulatora IV

Rezultati dobiveni algoritmom prikazani su u tabeli V:

TABLE V  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA  
FUNKCIJA (8))

$K_p$	0.94455497
$K_i$	0.10328197
$K_d$	0.85158134

Odziv dobiven sa ovim parametrima regulatora ne bi se mogao prihvatiti kao optimalan, što je očito ukoliko pogledamo grafik (slika 10). Referentna vrijednost nije dostignuta, a prvi preskok je prevelike amplitude.

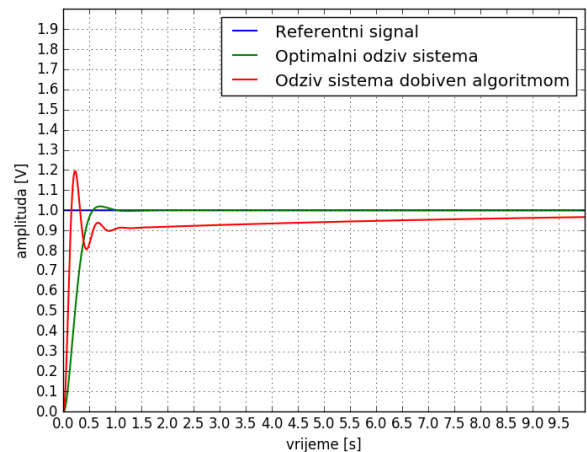


Fig. 10. Odziv sistema prema parametrima PID regulatora V

Prethodno dobiveni odziv (slika 10) je imao preveliku amplitudu prvog preskoka, pa je naredna funkcija kriterija dobivena tako što je, pored kvadrata greške uzeta u obzir i

maksimalna amplituda, upravo s ciljem da se dobije odziv sa manjim preskocom:

$$J = \int_0^T (e(t))^2 dt + 10 \cdot OS \quad (9)$$

Rezultati dobiveni algoritmom prikazani su u tabeli VI:

TABLE VI  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (9))

$K_p$	0.98352487
$K_i$	0.83102129
$K_d$	0.93790068

Primarni cilj ovog kriterija nije postignut, jer je vrijednost preskoka čak i nešto veća nego vrijednost u prethodnom slučaju, ali je omogućeno da sistem dostigne ravnotežno stanje (slika 11). Za dalje korekcije željenog odziva bi se moglo ispitati ponašanje algoritma u ovisnosti od faktora koji množi vrijednost prvog preskoka.

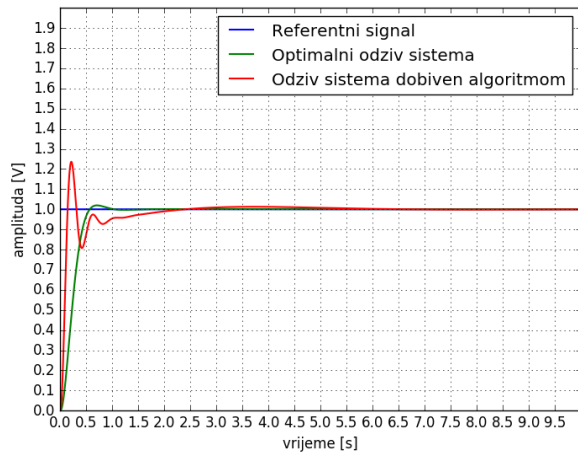


Fig. 11. Odziv sistema prema parametrima PID regulatora VI

Integral apsolutne greške [10] (Integral absolute error - IAE) definiše se na sljedeći način:

$$J = \int_0^T |e(t)| dt \quad (10)$$

Rezultati dobiveni algoritmom prikazani su u tabeli VII:

TABLE VII  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (10))

$K_p$	0.58885207
$K_i$	0.4182031
$K_d$	0.58818611

Nakon početne perturbacije odziv se otprilike od prve do treće sekunde prema skoro linearnom zakonu penje do referentne vrijednosti, neznatno je premaši, a zatim konvergira do kraja posmatranog intervala. Potrebno je relativno puno vremena da se dostigne zadani napon, ali se sistem ne dovodi u situaciju gdje bi zbog reakcije regulatora moglo doći do oštećenja izvršnih organa (slika 12).

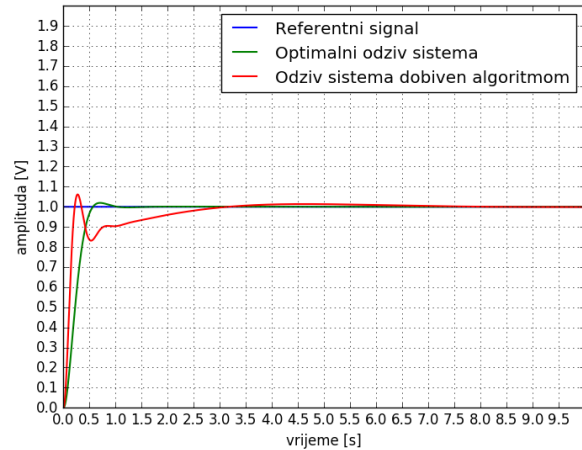


Fig. 12. Odziv sistema prema parametrima PID regulatora VII

Nešto bolja vrijednost odziva postignuta je upotrebom kriterijalne funkcije koja integrira umnožak vremena i apsolutne greške [10] (Integral time and absolute error - ITAE):

$$J = \int_0^T t \cdot |e(t)| dt \quad (11)$$

Rezultati dobiveni algoritmom prikazani su u tabeli VIII.

TABLE VIII  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (11))

$K_p$	0.77251278
$K_i$	0.39405533
$K_d$	0.45253815

Kao što je vidljivo sa dijagrama (slika 13) sve karakteristike odziva (u odnosu na prethodni grafik 12) su poboljšane osim vremena smirivanja oko ravnotežnog stanja, koje je u ovom slučaju nešto veće.

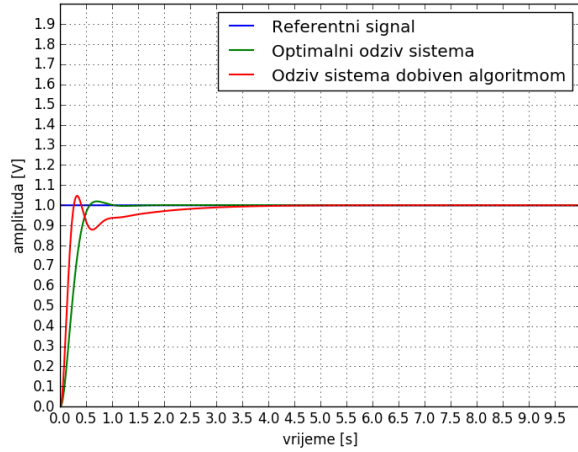


Fig. 13. Odziv sistema prema parametrima PID regulatora VIII

Sljedeća korištena funkcija kriterija je integral umnoška vremena i kvadrata greške [10] (Integral Square Time Error - ITSE) oblika:

$$J = \int_0^T t \cdot (e(t))^2 dt \quad (12)$$

Rezultati dobiveni algoritmom prikazani su u tabeli IX.

TABLE IX  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (12))

$K_p$	0.40258078
$K_i$	0.1382108
$K_d$	0.83299094

Iako je vrijeme porasta jako malo, prvi preskok nije tako velik, te bi se mogao uzeti kao rješenje (slika 14), ali javlja se drugi problem, a to je dostizanje stacionarnog stanja. Ovdje to vrijeme iznosi oko osam sekundi, te je ovakav sistem trom i mogao bi dovesti u pitanje stabilnost ostalih dijelova EE sistema, pa ovi parametri ne mogu biti rješenje problema.

Kao zadnji korišten kriterij upotrijebljen je izraz koji zavisi od vremena porasta ( $t_{rise}$ ), vremena smirenja oko referentne vrijednosti ( $t_{settling}$ ), vrijednosti prvog preskoka ( $M_p$ ), te apsolutne greške ( $E_{ss}$ ). Izraz je oblika [8]:

$$J = (1 - e^{-\beta}) \cdot (M_p + E_{ss}) + e^{-\beta} \cdot (t_s + t_r) \quad (13)$$

Ovdje je  $\beta$  težinski faktor i u ovisnosti od odabira parametra  $\beta$  akcenat stavljamo na to da smanjimo prvi preskok i grešku u odzivu ili da smanjimo vrijeme porasta i smirivanja. Algoritam je testiran više puta za razne vrijednosti faktora  $\beta$  u opsegu od 0.8 do 1, a najbolji rezultati se dobiju za vrijednosti 0.82 i 0.83.

Rezultati dobiveni algoritmom prikazani su u tabeli X:

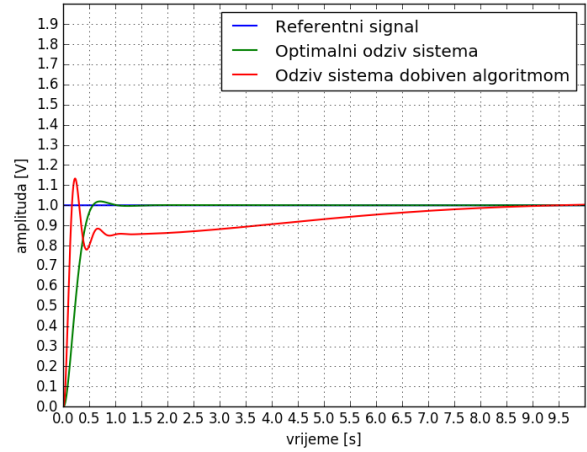


Fig. 14. Odziv sistema prema parametrima PID regulatora IX

TABLE X  
VRIJEDNOSTI PARAMETARA PID REGULATORA (KRITERIJALNA FUNKCIJA (13))

$K_p$	0.69386626
$K_i$	0.6433866
$K_d$	0.32823495

Posmatrajući naredni grafik (slika 15) zaključujemo da je ovakav odziv zadovoljavajući, obzirom na preskok, vrijeme smirenja i brzinu odziva. Istina, brzina odziva bi se mogla povećati, ali bi to vjerovatno rezultiralo višim preskokom. Ipak, vrijeme porasta napona nije unutar kritičnih vrijednosti, pa bi i ovakav odziv bio rješenje zadanog problema.

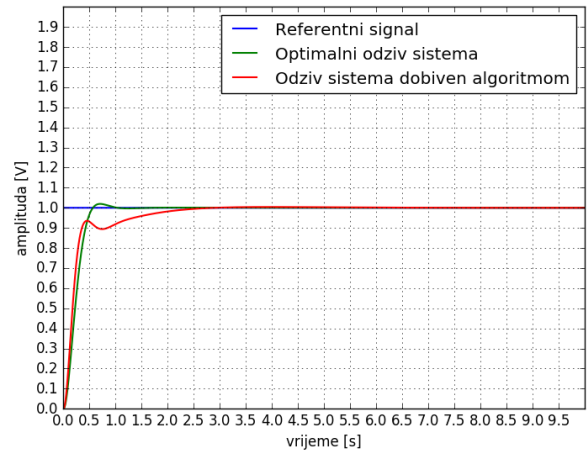


Fig. 15. Odziv sistema prema parametrima PID regulatora X

### C. Zaključak

Nakon provedenih testiranja algoritma za različite kriterijalne funkcije i analiziranja dobivenih rezultata možemo zaključiti da je pojam optimalnog odziva relativan i jako

osjetljiv. U ovisnosti od toga da li je bitnije imati brz odziv ili odziv u kojem nema preskoka referentne vrijednosti odabiremo različita rješenja. Rješenje koje je optimalno po svim parametrima je nemoguće pronaći, jer su zahtjevi koje postavljamo na odziv često suprotstavljeni. Ukoliko povećamo brzinu odziva, povećava se i prvi preskok, što može dovesti do visokih prenapona. Prenaponi u elektroenergetskom sistemu su jako nepoželjni, obzirom da mogu oštetiti sistem, ali i dovesti do ljudskih žrtava u krajnjem slučaju. S druge strane, tromost odziva radi smanjenja preskoka u sistemu može dovesti do pada frekvencije i raspada sistema. Prema tome, potrebno je postići određeni balans i na izvjestan način napraviti kompromis među ovim zahtjevima i ograničenjima sistema. Od svih navedenih rezultata, rezultati koji su u velikoj mjeri zadovoljili sva ograničenja koriste funkcije kriterija 4, 7 i 13. Naravno, ovaj zaključak varira u ovisnosti od toga šta je primarni cilj odziva, o čemu je prethodno bilo riječi.

#### IV. ZAKLJUČAK I DISKUSIJA

U ovom radu je prezentiran i primijenjen PSO algoritam u svrhu traženja optimalnih parametara PID regulatora. Primjena ovog algoritma izvršena je na sistemu automatske regulacije napona, koji je lineariziran da bi primjena linearnog PID regulatora imala smisla. Bilo je potrebno pronaći optimalnu funkciju i vrijednost kriterija, koja rezultira optimalnim setom parametara. Izvršeno je ispitivanje na raznovrsnim funkcijama kriterija i utvrđeno ponašanje algoritma. Algoritam je takođe ispitan za različite vrijednosti parametara, te je utvrđeno da su preporučene vrijednosti ujedno i optimalne vrijednosti u pogledu konvergencije algoritma. Dakle, PSO omogućava brz i efikasan pronalazak optimalnog rješenja problema uz korištenje malog obima resursa. Takođe, algoritam je fleksibilan i jednostavan za implementaciju, te je kao takav poželjan i u primjeni na složenije probleme.

#### REFERENCES

- [1] Samim Konjicija, Predavanja na predmetu: Optimizacija resursa, Elektrotehnički fakultet, Univerzitet u Sarajevu, Š/K 2015/2016
- [2] Mujo Hebibović, Predavanja na predmetu: Linearni sistemi automatskog upravljanja, Elektrotehnički fakultet, Univerzitet u Sarajevu, Š/K 2015/2016
- [3] Sead Kreso, Predavanja na predmetu: Strukture i režimi rada elektroenergetskih sistema, Elektrotehnički fakultet, Univerzitet u Sarajevu, Š/K 2015/2016
- [4] A. Visioli, Tuning of PID controllers with fuzzy logic, Proc. Inst. Elect. Eng. Contr. Theory Applicat., vol. 148, no. 1, pp. 1–8, Jan. 2001.
- [5] T. L. Seng, M. B. Khalid, and R. Yusof, Tuning of a neuro-fuzzy controller by genetic algorithm, IEEE Trans. Syst., Man, Cybern. B, vol. 29, pp. 226–236, Apr. 1999.
- [6] Y. Mitsukura, T. Yamamoto, and M. Kaneda, A design of self-tuning PID controllers using a genetic algorithm, in Proc. Amer. Contr. Conf., San Diego, CA, June 1999, pp. 1361–1365.
- [7] H. I. Kang, M. W. Kwon, H. G. Bae, Comparative Study of PID Controller Designs Using Particle Swarm Optimizations for Automatic Voltage Regulators, IEEE conference on Information Science And Applications, pp. 1-5, 2011.
- [8] Zwe-Lee Gaing, A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System, IEEE Transactions on Energy Conversion, Vol. 19, No. 2, pp.384-391, June 2004.

- [9] Mohammad Sadegh Rahimian and Kaamran Raahemifar, Optimal PID Controller Design for AVR System using Particle Swarm Optimization Algorithm, Electrical and Computer Engineer Department, Ryerson University Toronto, Ontario, Canada. IEEE CCECE 2011.
- [10] Panda, S., Sahu, B. and Mohanty, P., Design and performance analysis of pid controller for an automatic voltage regulator system using simplified particle swarm optimization, Journal of the Franklin Institute, Vol. 349, No. 8, (2012), 2609-2625.
- [11] X.S. Yang, Engineering Optimization: An Introduction with Metaheuristic Applications, John Wiley & Sons, 2010.