

# Optimizacija proceduralno generisanog drveća

Darko Brdareski, Milan Tomić

*Sadržaj* – Lindenmajerovi sistemi (L-sistemi) jesu metod proceduralne generacije sadržaja koji se koristi sa uspehom već dugi niz godina. Mana L-sistema jeste što se njihova pravila i parametri moraju ručno sastavljati. Genetski algoritmi (GA) su metod optimizacije postojećih sistema. Kombinovanjem L-sistema za generisanje sadržaja i GA za optimizaciju sistema, moguće je lako doći do novog i kvalitetnog sadržaja, bez potrebe da se provede velika količina vremena za podešavanje pravila i parametara.

*Ključne reči* – Genetski algoritmi, L-sistemi, Proceduralno generisanje sadržaja

## I. UVOD

U mnogim oblastima računarstva često se ukazuje potreba za velikom količinom kvalitetnog sadržaja. U slučaju potrebe za vizuelnim sadržajem (npr. u video igrama), ovo mogu biti 3D i 2D modeli, mape terena, teksture i slično. Tradicionalno, ove zadatke je rešavao čovek, tako što je ručno konstruisao, dizajnirao i crtao sav sadržaj. Međutim, sa porastom procesorske moći, sve su više počele da se primenjuju tehnike proceduralne generacije sadržaja. Korišćenjem ovih tehnika, lako se može dobiti velika količina kvalitetnog vizuelnog sadržaja za samo delić vremena koji bi bio potreban čoveku da ga stvori.

*Lindenmajerovi sistemi* (dalje: L-sistemi) su sistemi koji se uglavnom koriste za proceduralnu generaciju sadržaja. L-sistem se sastoji iz alfabeta simbola koji se mogu koristiti za kreiranje stringova, skupa produkcionih pravila koji pretvaraju jedan string u drugi, i početnog stringa koji se naziva *aksiom*. Rad L-sistema počinje od aksioma. U svakoj iteraciji sistema, na trenutni string se paralelno (u istom trenutku) primenjuju sva moguća pravila produkcije. Dobijeni

Darko Brdareski, Računarski fakultet, Knez Mihailova 6, 11000 Beograd, Srbija (e-mail: dbrdareski13@raf.edu.rs).

Milan Tomić, Računarski fakultet, Knez Mihailova 6, 11000 Beograd, Srbija (e-mail: mtomic@raf.edu.rs).

string se zatim koristi u sledećoj iteraciji, i proces se nastavlja dok se sistem ne zaustavi (npr. dostigne se određen broj iteracija, ili je korisnik zadovoljan trenutnim rezultatima).

Jedna od čestih primena L-sistema jeste stvaranje vizuelnog sadržaja, kao i primena u drugim naučnim oblastima, poput genetike [1] i botanike [2].

Iako su L-sistemi veoma efikasan način proceduralnog generisanja sadržaja (pošto se ne mora čuvati sam sadržaj, već samo pravila kojima se generiše), oni ipak poseduju nekoliko mana. Prvo, ne mogu se koristiti za generisanje sadržaja koji nema određen nivo samsličnosti, što ograničava domen njihove primene. Druga mana jeste što se pravila moraju ručno pisati, analizirati njihovi rezultati, i dalje doradivati.

Cilj ovog rada je da pokaže kako se genetski algoritmi mogu efikasno iskoristiti za unapređivanje unapred zadatih pravila produkcije, čime se veći deo posla analiziranja i doradivanja prepušta algoritmu, a čoveku ostaje samo da ocenjuje rezultate i na taj način „vodi” algoritam ka dobrim pravilima produkcije. Eksperiment je urađen na pravilima za generisanje drveta, a subjektivna ocena posmatrača dodeljuje se na osnovu grafičkog prikaza drveta koje nastaje primenom njemu pripadajućih pravila produkcije određeni broj puta.

Drugi autori su već predstavljali slična rešenja: neograničenom mutacijom i evolucijom L-sistema [3], ili korišćenjem GA da se dobije L-sistem koji opisuje-oponaša željenu strukturu [4].

## II. KORIŠĆENE METODE

### A. *L-sistem*

Struktura L-sistema korišćenog u ovome radu se zasniva na osobinama L-sistema koje su opisane u [5]. Aksiom se sastoji iz 5-10 simbola, koji modeliraju „izdanak”. Alfabet se po potrebi sastoji iz proizvoljnog broja simbola. Osnovne simbole koristi tzv. *kornjača*. Kornjača je uobičajen naziv [5] za parser L-sistema, zbog načina parsiranja komandi. String se parsira simbol po simbol i kornjača izvršava odgovarajuće naredbe. Izvršavanjem naredbi, kornjača ostavlja trag na radnoj površini – iscrtava oblike, odnosno grafičku predstavu generisanog stabla. Osnovni simboli predstavljaju veoma

proste komande koje se koriste da bi se iscrtavali složeniji oblici poput cvetova, grana i sl.

Pored alfabeta osnovnih simbola, koji predstavljaju jedinične komande kornjače, L-sistem, zajedno sa GA, koristi i produženi alfabet koji predstavlja organe. Ovaj alfabet predstavlja složenije oblike. Na primer, jedan cvet ili grana se mogu sastojati iz više desetina osnovnih simbola. Tokom parsiranja složenih simbola, oni će biti zamenjeni nizom osnovnih simbola koje u stvari predstavljaju, da bi se iscrtali. Ovaj pristup je odabran jer nam obezbeđuje dva važna svojstva. Prvo je da se složeni elementi jedne jedinke uvek na isti način iscrtavaju (npr. svi cvetovi kod jednog drveta su isti), a drugo je da različite jedinke mogu imati različite listove i cvetove, bez komplikovanja samog genetskog algoritma u domenu ukrštanja. Još jedan razlog je što se na taj način lakše prati struktura stabla. Kada listove i cvetove (kao složene elemente) apstrahujemo, možemo pratiti samo redosled simbola koji obrazuju grane, a na mestima listova i cvetova može se naći bilo koji oblik, čime obezbeđujemo veću čitljivost i upotrebljivost generisanog sistema.

TABELA 1: OSNOVNI SIMBOLI L-SISTEMA

F	Pomeranje kornjače napred
B	Pomeranje kornjače nazad
+	Rotiranje kornjače desno
-	Rotiranje kornjače levo
C	Crtanje kruga počev iz trenutne pozicije kornjače
!	Promena debljine linije kojom kornjača crta
L	Promena boje kojom kornjača crta
[	Čuvanje stanja kornjače na steku
]	Vraćanje stanja kornjače sa steka

TABELA 2: PRODUŽENA TABELA SIMBOLA L-SISTEMA

a	Deblo drveta (najveća, centralna grana)
b	Grana drveta
l	List
f	Cvet

Celokupan skup simbola (bez zagrada za operacije na steku) može biti numerički parametrizovan, čime se kornjači prosleđuju instrukcije na koji način da izvede određenu operaciju. Parametri uz simbole pišu Vol. 7, 2015.

se unutar zagrada, na primer  $F(40)$  - idi napred 40 koraka,  $L(1,0,0)$  - promeni boju iscrtavanja u crvenu (RGB model), i tome slično.

Korišćena pravila produkcije su kontekstno-slobodna, odnosno leva strana produkcije ne sadrži više od jednog simbola (sa 0 ili više pripadajućih numeričkih parametara). Sama produkcija, međutim, može biti uslovljena vrednošću nekog od tih parametara, pa se na osnovu toga za isti simbol mogu primeniti različite produkcije. Jedan od parametara koji određuju koje će se pravilo primeniti je parametar koji određuje vreme pre nego što se cvetovi pojave na mestu nove grane. Rezultat primene pravila na simbol takođe može zavisiti od parametara početnog simbola, što se izvodi ugradnjom jednostavnog matematičkog izraza u rezultat, na primer:  $F(x) \rightarrow F(x \cdot 1.17)$ . Takvi su parametri koji određuju fizičku predstavu simbola. Primer ove vrste parametara jeste parametar koji se nalazi u granama stabla, i određuje njihovu dužinu. Kada izraste nova grana, ovaj parametar ima malu vrednost, te je i grana mala. Kako stablo raste u visinu, tako rastu i vrednosti parametara svih organa, pa samim tim i grane dobijaju na veličini. Treća vrsta parametara služi samo za vizuelnu reprezentaciju stabla; ovde spadaju parametri poput boje listova i cvetova. Ukupno među aksiomima i pravilima produkcije postoji 17 parametara koji mogu određivati izgled jedne jedinke. Koji će se od njih koristiti, zavisi od aksioma i seta pravila produkcije koje jedinka dobije. Svi oni, međutim, ulaze u ukrštanje u genetskom algoritmu, tako da je svakoj jedinki obezbeđena vrednost za svaki od parametara, čime se obezbeđuje da različiti setovi pravila produkcije mogu nesmetano da saraduju.

Da bi se izbegla jednoličnost biljaka generisanih pomoću L-sistema, može se koristiti stohastički L-sistem [5]. Međutim, zbog potrebe da se stabla ocene sa većom sigurnošću u njihovu dobru formu, ova mogućnost nije u potpunosti iskorišćena. Korišćeni L-sistem je stohastički samo u domenu rasta cvetova, dok su sva ostala pravila deterministička. Ova odluka je doneta nakon prvobitnih eksperimenata sa stohastičkim pravilima (npr. dodeljivanje proizvoljne verovatnoće za rast grane ili listova). Proizvoljne verovatnoće za rast cvetova su ipak ostale prisutne, da bi se sprečila pojava prekomernog cvetanja, gde je celo drvo prekriveno cvetovima.

### *B. Genetski algoritam*

Genetski algoritam je metod optimizacije koji počiva na teoriji evolucije i prirodne selekcije. Potencijalna rešenja problema su u genetskom algoritmu predstavljena kao jedinke, koje se odabiraju, ukrštaju i mutiraju. Na ovaj način se kroz generacije, odnosno iteracije algoritma, dolazi do sve boljih i boljih mogućih rešenja. Više detalja o osnovama genetskog algoritma može se naći u [6]. Genetski algoritmi su najpopularniji evolutivni metod optimizacije [4].

Genetski algoritam korišćen u radu je specifično prilagođen strukturi jedinke. Hromozomi su podeljeni u celine koje se međusobno dopunjuju formirajući kompleksan genotip L-sistema (drveta) koji opisuju: aksiom, formalna pravila ekspanzije, fiksne karakteristike drveta (oblici listova i cvetova, takođe zadati preko L-stringova) i parametri koji podešavaju specifične vrednosti u svim navedenim celinama. Svaka od ovih celina podleže genetskom operatoru ukrštanja, a jedna podleže i mutaciji. Ocena hromozoma je subjektivna i promenljiva, relativna u odnosu na pripadnike iste generacije. Ukoliko hromozom preživi odbacivanje i pređe u sledeću generaciju, ocena se traži ponovo od ocenjivača. Takođe, redosled ocenjivanja je nasumičan, tako da ocenjivač ima mogućnost da stekne uvid u neke druge jedinke pre nego što mu se desi ponovljena, te da ocenu prilagodi onome što je video. Na ovaj način, postiže se da se uz mali skup vrednosti ocena (0-9) trenutno bolji hromozomi lakše izdvoje od onih koji su u odnosu na njih inferiorni.

Da bi se bolje razumelo ukrštanje, detaljnije ćemo objasniti strukturu jedinke.

Osnovna jedinica građe svakog dela jedinke koji nije čist numerički parametar je ćelija, koja je sposobna da se razvija (po pravilima ekspanzije), može da mutira i da se replicira u ćeliju potomka. Od ćelija su sastavljeni delovi jedinke izvedeni pomoću L-sistema. Svaka ćelija predstavlja jedan simbol iz osnovnog alfabeta i u sebi može da sadrži i jedan ili više parametara koji je bliže opisuju. Takođe, ćelije se mogu grupisati u organe, a organi su u suštini grupacije ćelija između para uglastih zagrada (koje u L-sistemu predstavljaju čuvanje trenutnog stanja kornjače na steku).

Svaka jedinica sadrži aksiom, koji predstavlja početno stanje drveta, iz koga se ono dalje razvija pomoću ekspanzija. To stanje je

sačuvano u nizu ćelija, koji se u metodi ekspanzije širi po odgovarajućim pravilima. Pravila ekspanzije implementirana su pomoću stringova. Svako pravilo sadrži simbol na koji se odnosi i string u koji se taj simbol može proširiti. Pri širenju, zamenjuju se odgovarajući numerički parametri i od rezultujućeg stringa se kreiraju nove ćelije. Svaka jedinka sadrži po jedno pravilo za svaku postojeću kombinaciju simbola i parametara, i ova pravila se prosleđuju svim ćelijama te jedinke za ekspanziju.

Numerički parametri su u okviru jedinke smešteni u posebnu mapu. Njihove vrednosti se uzimaju kada su potrebne za ekspanziju. Postoji varijabilan broj numeričkih parametara koji se koriste u pravilima ukrštanja (po potrebi se mogu definisati novi parametri). Svaki od njih ima definisanu minimalnu i maksimalnu vrednost, i za svaku jedinku može uzeti vrednost iz tog opsega.

Karakteristike lista i cveta se definišu pri nastanku jedinke. One predstavljaju neki od fiksno zadatih oblika lista ili cveta i služe za olakšano iscertavanje. Oblici su zadati u formi nizova simbola L-sistema, ali se pri kreiranju jedinke pretvaraju u organ. Svaki put kada se u procesu iscertavanja dođe do simbola organa, iscertava se taj organ. Na taj način štedi se na memorijskom prostoru i obezbeđuje se da svi listovi i cvetovi jedne jedinke budu identični. List i cvet ne podležu ekspanziji (rastu), ali podležu parametrizaciji (boje).

Ukrštanje se izvodi na više nivoa, a rezultat je jedna nova jedinka (ne dve komplementarne, kao što je to često uobičajeno, da bi se postigao veći diverzitet na manjim populacijama).

Ukrštanje na nivou aksioma je u biti uniformno. Izvodi se tako što se od aksioma oca i aksioma majke redom uzimaju ćelije i nasumično se bira koja će biti nasleđena. Izuzetak od toga su ćelije koje predstavljaju čuvanje stanja na steku ili skidanje sa steka, koje se direktno prepisuju od majke, a preskaču kod oca. Takođe, ako je aksiom majke duži, ostatak od trenutka kada se stiglo do kraja očeve jedinke prepisuje se od majke. Ovo je jednostavna odluka doneta u fazi dizajniranja, jer se na taj način lako obezbeđuje da stanja ostanu konzistentna (tj. da ne postoji operacija slanja stanja na stek bez odgovarajuće operacije skidanja stanja sa steka, jer to nije gramatički korektno).

Ukrštanje na nivou pravila ekspanzije izvodi se slično. Kroz obe jedinke se prolazi paralelno i nasumično se od roditelja bira po jedno pravilo za svaki od simbola, po uniformnoj raspodeli. S obzirom na to da su pravila ekspanzije opisana tako da svaka jedinka ima po jedno pravilo za svaki definisani simbol (i odgovarajuće varijante gde je simbol dodatno parametrizovan), ovo ukrštanje je jednostavnije.

Ukrštanje na nivou karakteristika lista i cveta izvode se na isti način.

Ukrštanje na nivou parametara je malo složenije. Za svaki od parametara proverava se kog je tipa. Ako je u pitanju složeni parametar (kao što je npr. boja koja se sastoji iz 3 vrednosti – intenzitet crvene, zelene i plave), nad svakom vrednošću sprovodi se linearno stapanje sa prethodno nasumično izabranim parametrom  $a \in [0, 1)$ .

$$d_i = (1 - a) \cdot o_i + a \cdot m_i$$

Ako je u pitanju jednostavan parametar, ukrštanje se izvodi modifikovanom linearnom ekstrapolacijom: nasumično se uniformno bira jedna od tri formule za ukrštanje (1), (2), (3), a zatim se proverava da li je dobijena vrednost dozvoljena za taj parametar, i ako nije, postavlja se na bliži ekstrem, (4).

$$b_i = (1 - a) \cdot o_i + a \cdot m_i \quad (1)$$

$$b_i = (1 + a) \cdot m_i - (1 - a) \cdot o_i \quad (2)$$

$$b_i = (1 + a) \cdot o_i - (1 - a) \cdot m_i \quad (3)$$

$$d_i = \max(\min(b_i, M_i), m_i) \quad (4)$$

gde su  $M_i$  i  $m_i$  dozvoljeni maksimum i minimum  $i$ -tog parametra, respektivno. Po izvršenom ukrštanju, nova jedinka se dodaje u postojeću populaciju (iz koje su već izbačene neprilagođene).

Nakon faze ukrštanja, nastupa mutacija. Mutacija je takođe specifična u odnosu na neke tradicionalne primene genetskog algoritma, jer joj podleže samo jedan deo genoma – numerički parametri. Svaka jedinka sa unapred podešenom verovatnoćom podleže izvesnom (takođe podešenom) broju mutacija, od čega se može izvesti i manje od tog broja, što je rezultat još jednog nasumičnog odabira. Zadati broj parametara se onda nasumično bira, i jedan po jedan se ponovo generišu iz skupa svojih dozvoljenih vrednosti. U algoritmu gde je ocena subjektivna i podložna promeni iz

generacije u generaciju, isprobali smo dva pristupa – sa i bez elitizma, odnosno očuvanja najbolje jedinke. Postojanje elitizma može pozitivno uticati na krajnji rezultat, ali treba imati u vidu da algoritam nema konačni završetak, tj. ako je neka jedinka jako dobra, ocenjivač može lako prepisati njene parametre pre nego što je oceni ili isključi algoritam. S druge strane, nedostatak elitizma može povećati diverzitet jer omogućuje jednom drvetu više da se promeni i uvede novi genetski materijal u populaciju.

### III. REZULTATI

Izvedeno je osam eksperimenata sa različitim podešavanjima genetskog algoritma. Broj generacija svakog izvođenja bio je 20, uz populaciju od 20 jedinki, od kojih je 10 sa najnižim ocenama odbacivano pre selekcije. Broj mutacija po jedinki je varijabilan (između 5 i 10), a primenjuje se ukoliko je jedinka odabrana za mutaciju. Za verovatnoću mutacije jedne jedinke uzete su dve vrednosti – 0.05 (niska) i 0.2 (visoka). Za selekciju su korišćena dva algoritma – Ruletski točak ponderisan ocenom prilagođenosti i Takmičarska selekcija 3 jedinke.

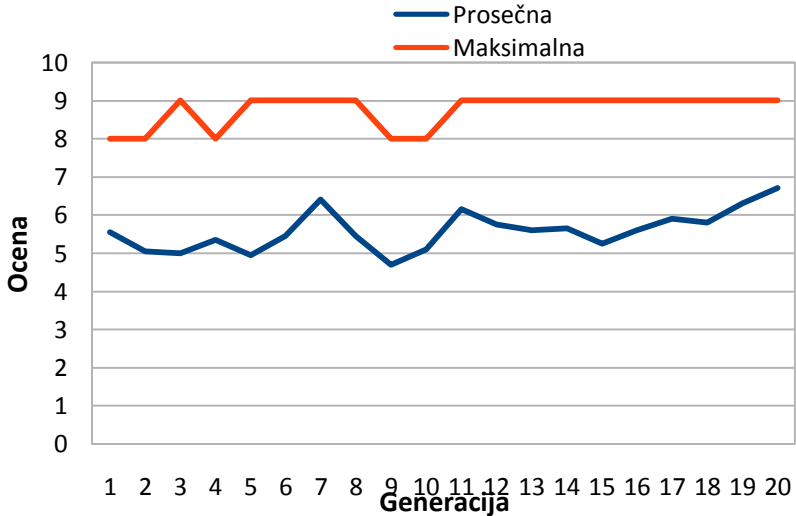
Rezultati svakog izvođenja sačuvani su kao niz ocena po generaciji, tako da se iz njih može izvući kretanje prosečne i maksimalne ocene po generacijama. Na ovaj način možemo zaključiti kakav je uticaj postavke genetskog algoritma na optimalni razvoj. Pri tumačenju ovih rezultata trebalo bi uzeti u obzir i zamor i subjektivnost ocenjivača, jer je proces evolucije prilično dugotrajan i može postati zamoran.

Na graficima su prikazana kretanja prosečne i maksimalne ocene po generacijama za svaku kombinaciju parametara genetskog algoritma.

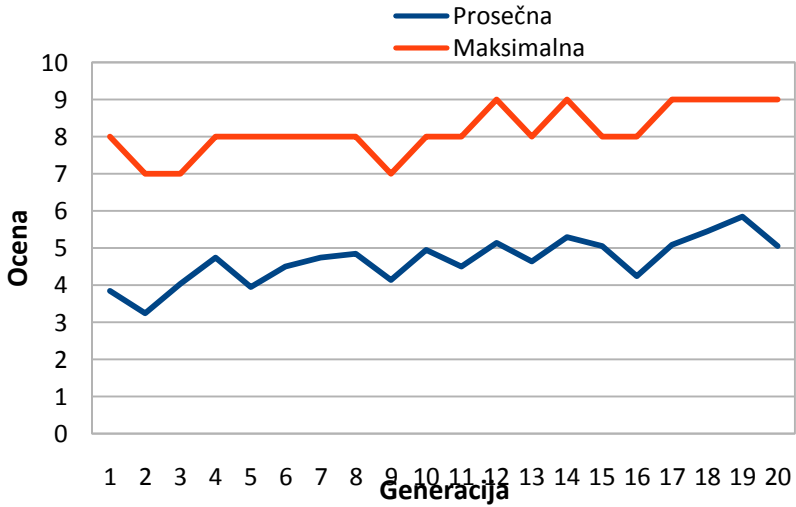
Na osnovu pregleda grafika možemo zaključiti da prosečna ocena u svakom testu varira, negde manje, negde više. Izuzetno, kod visokog iznosa mutacije, sa primenjenim elitizmom i selekcijom metodom ruletskog točka, vidimo da prosečna ocena uglavnom postepeno raste, dok je kod ostalih slučajeva uglavnom nestabilnija, na šta ukazuje testerast oblik krive prosečne ocene. Uglavnom se i maksimalna ocena u toku generacije testerasto menja, ali sa više platoa. Platoi ukazuju na održavanje kvaliteta jedinki kroz više generacija, a testeraste promene



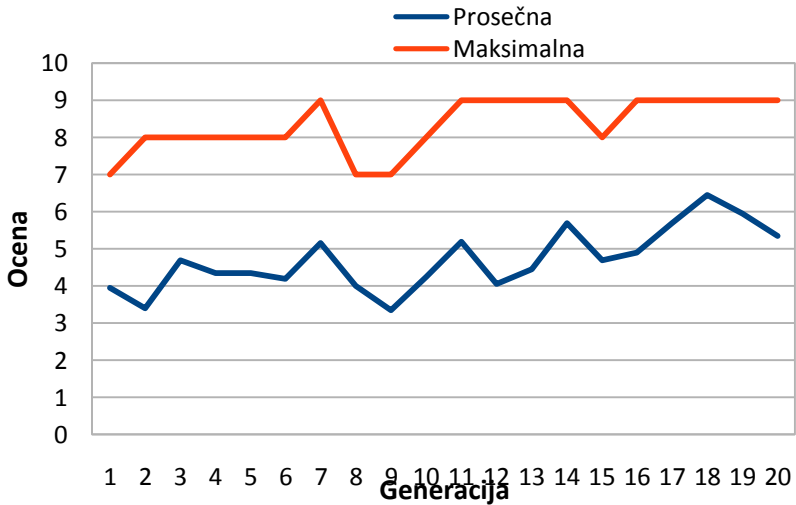
na promene kvaliteta čitave generacije, objektivno ili subjektivno. Naime, kako broj generacija raste, od algoritma se očekuje više sličnih rezultata, kod kojih detalji igraju presudnu ulogu. Samim tim i kriterijumi postaju oštriji, pa i prosečna ocena može opasti, kao i maksimalna, jer se očekuje neka nova i prilagođenija jedinka u generaciji, koja uprkos svemu nije nastala.



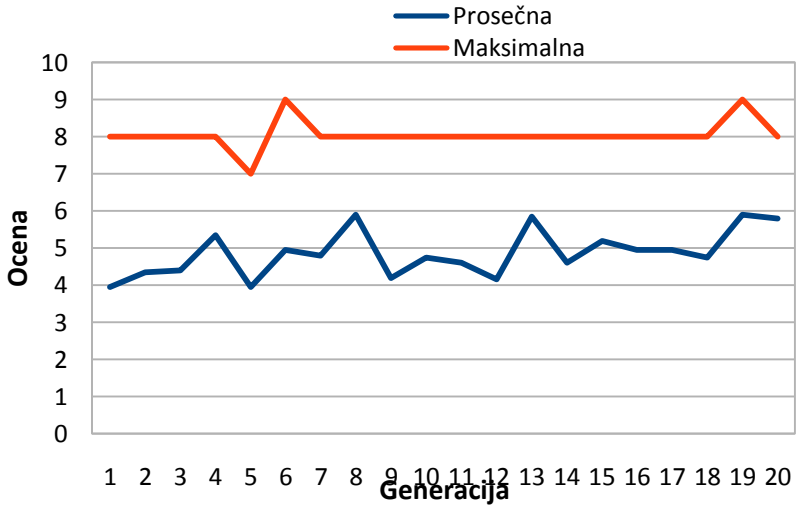
Sl. 1 Ruletski točak, bez elitizma,  $r=0.05$



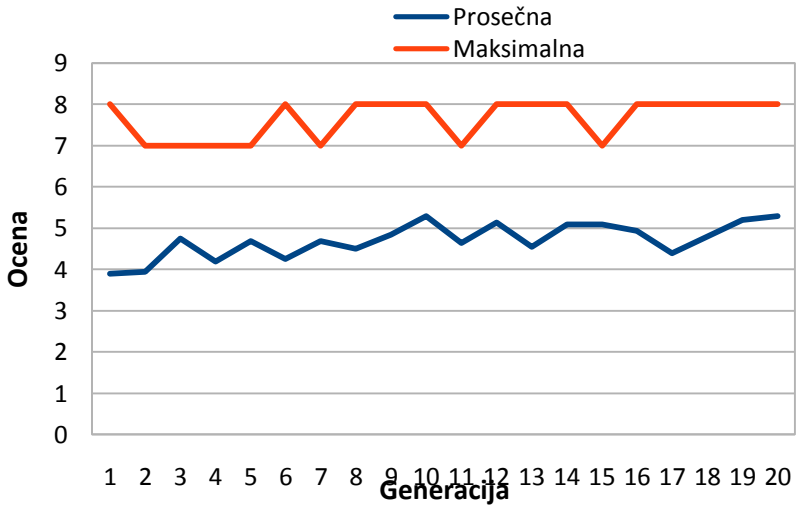
Sl. 2 Ruletski točak, bez elitizma,  $r=0.2$



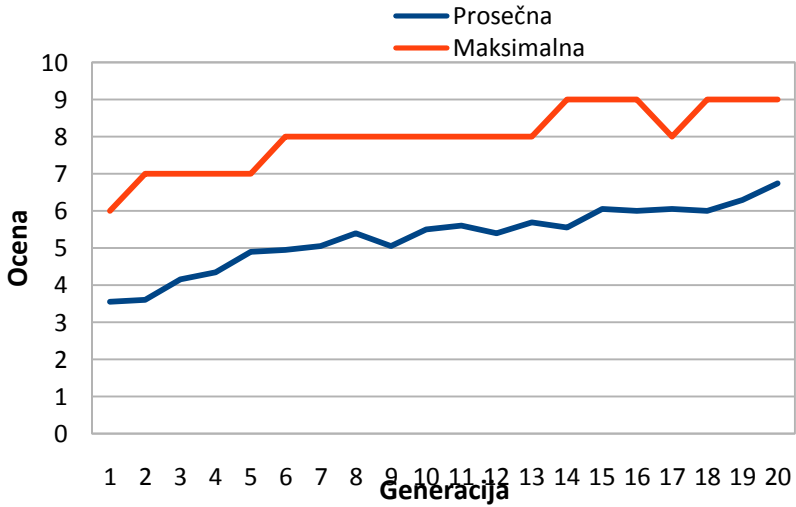
Sl. 3 Takmičarska selekcija, bez elitizma,  $r=0.05$



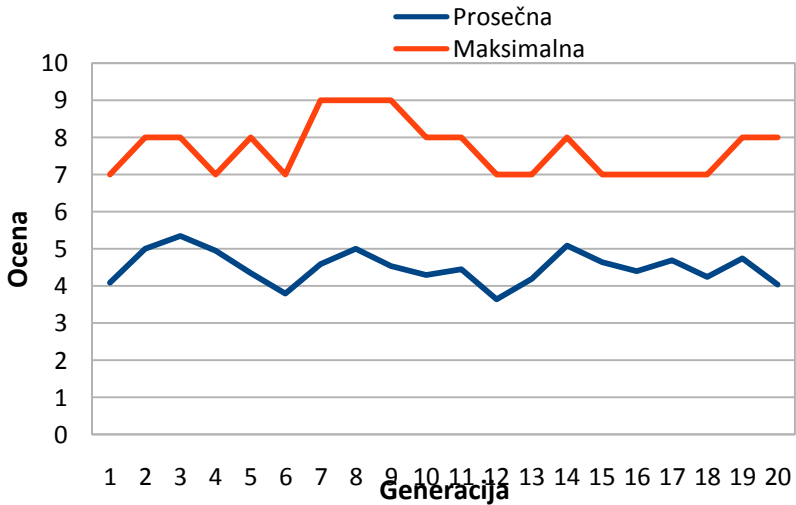
Sl. 4 Takmičarska selekcija, bez elitizma,  $r=0.2$



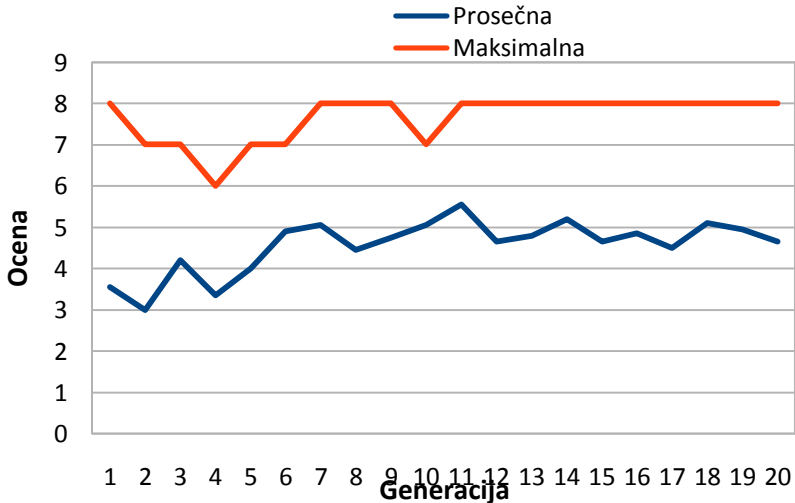
Sl. 5 Ruletski točak, sa elitizmom,  $r=0.05$



Sl. 6 Ruletski točak, sa elitizmom,  $r=0.2$



Sl. 7 Takmičarska selekcija, sa elitizmom,  $r=0.05$



Sl. 8 Takmičarska selekcija, sa elitizmom,  $r=0.2$

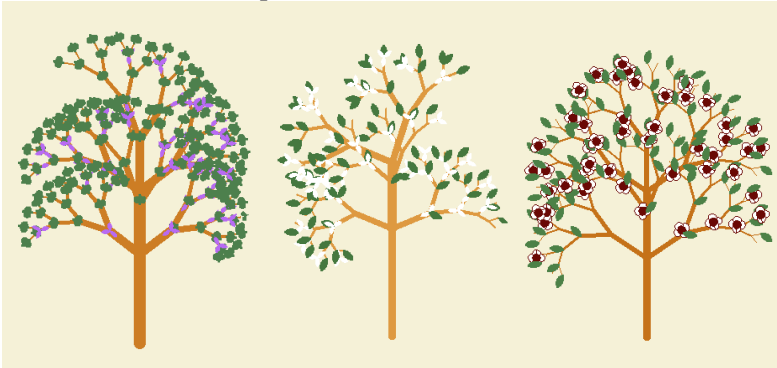
Rezultati ukazuju na to da je takmičarska selekcija nepogodna za ovaj slučaj, jer se zbog njene prirode više neprilagođenijih jedinki može ugurati u ukrštanje, nego što je to slučaj sa selekcijom preko ruletskog točka ponderisanog ocenom prilagođenosti. Takođe, ona daje veću varijabilnost jedinki, odnosno manju stabilnost dominantnog tipa jedinki u populaciji.

Zbog prirode zadatka, čini se da veći iznos mutacije bolje deluje na diverzitet populacije. Međutim, pri većim iznosima mutacije, kod izvršenja genetskog algoritma bez elitizma, javlja se veći broj jako neprilagođenih jedinki, pa je time veće variranje prosečne ocene. Veći iznos mutacije nema toliko uticaja na prosečnu ocenu kroz generacije kada je elitizam prisutan.

Na konačne rezultate testova značajno može uticati način ocenjivanja. Ocenjivanje se izvršava serijski, odnosno ocenjivaču se prikazuje jedna po jedna slika jedinke iz trenutne generacije, na osnovu čega on od početka nema opštu sliku o izgledu cele generacije. Tako se ocene formiraju na osnovu iskustva iz prethodnih generacija i očekivanja za trenutnu generaciju. Ova očekivanja i iskustva ne moraju se nužno slagati sa realnim stanjem, te zbog toga vrlo lako može doći do „nazadovanja“ rezultata. Međutim, iskustvo ocenjivača

govori da je kroz generacije uglavnom dolazilo do napretka, što se vidi i po konačnom rastu krive prosečne ocene.

Neki od rezultata prikazani su na Sl. 9.



Sl. 9 Drveće generisano pomoću L-sistema

#### IV. DALJI RAD

Za dalji rad mogu se podržati određena uopštavanja L-sistema sa kojim algoritam radi, kao i proširenja samog genetskog algoritma. Što se tiče uopštavanja L-sistema, pogodno je proširiti ga na trodimenzionalnu kornjaču, uz dodavanje odgovarajuće grafičke biblioteke koja bi to mogla da podrži, a da softver ostane interaktivan. Većina radova u ovoj oblasti odnosi se na trodimenzionalnu kornjaču [3], [5], [7]. Još jedno proširenje moglo bi biti podrška za kontekstno-zavisne L-sisteme [5].

U domenu interaktivnosti, interesantno i poželjno bi bilo podržati pregled čitave generacije na jednom ekranu, kako bi se stekao realan utisak o međusobnom odnosu prilagođenosti postojećih jedinki, i kako bi ocene bile što realnije. Tako nešto bi smanjilo uticaj prvih nekoliko prikazanih jedinki u generaciji na prosek čitave generacije, tj. rezultati bi bili stabilniji.

Još jedan novi način ocenjivanja može biti primenjen, a to je sortiranje cele generacije, sa ocenama od 1 do 20. U interaktivnom prozoru, ocenjivač bi sortirao sličice stabala po prilagođenosti, umesto da unosi numeričke ocene. Ovakav proces bio bi ocenjivaču manje zamoran od serijskog dodeljivanja numeričkih ocena, spasao bi ga

eventualnog razmišljanja o statistici generacije i o dodeljivanju prave ocene, a možda bi pomogao i u bržem dostizanju najboljih jedinki.

Sa strane izvodljivosti eksperimenta, bilo bi dobro primeniti *crowdsourcing*, odnosno napraviti interaktivni softver na Internetu, koji bi grupa ljudi koristila da oceni trenutne generacije. Konačna ocena svake jedinke bi mogla da se prikuplja u toku ograničenog vremena, a onda da se izračuna proseka, i na taj način da se dobije nešto realnija slika. Ovaj proces bi mogao trajati i značajno veći broj generacija, a tehnikama A-B testiranja, mogli bi biti postavljeni različiti parametri genetskog algoritma, kako bi se videle razlike u rezultatima.

Zbog prirode zadatka, bilo bi dobro omogućiti ocenjivaču da iz svake generacije izvuče i pregleda sva stabla koja su trenutno najprilagođenija. Kako ne postoji jedno konkretno optimalno rešenje, a budući da više jedinki može imati istu ocenu, bilo bi lepo videti koje sve postavke L-sistema mogu da dovedu do poželjnih rezultata. Ove jedinke mogu biti prikazane na razne načine, a može se zadati i veći ili manji broj ekspanzija, kako bi se videlo kako se te jedinke dalje razvijaju, i koja od njih možda ima lepši razvoj za veći broj ekspanzija.

## LITERATURA

- [1] Gemma B. Danks, Susan Stepney, Leo SD Caves, "Protein folding with stochastic L-systems," *Artificial Life XI*, 2008.
- [2] Hanan, Jim, "Modelling Cotton (*Gossypium hirsutum* L. with L-Systems: A Template Model for Incorporating Physiology," *FSMP04*, 2004.
- [3] McCormack, Jon, "Interactive evolution of L-system grammars for computer graphics modelling," *Complex Systems: from biology to computation*, 1993, pp. 118-130.
- [4] Humera Farooq, M. Nordin Zakaria, Mohd. Fadzil Hassan, Suziah Sulaiman, "An approach to derive parametric l-system using genetic algorithm," *Visual Informatics: Bridging Research and Practice*, Kuala Lumpur, Malaysia: Springer, 2009. pp. 455-466.
- [5] Prusinkiewicz, Przemyslaw and Lindenmayer, Aristid, *The algorithmic beauty of plants*, Regina, Canada: Springer Science & Business Media, 2012.
- [6] Randy L. Haupt, Sue Ellen Haupt, *Practical Genetic Algorithms*, Second Edition. Hoboken, New Jersey: John Wiley Sons, Inc., 2004.
- [7] Joanna L Power, AJ Brush, Przemyslaw Prusinkiewicz, David H Salesin, "Interactive Arrangement of Botanical L-System Models," *Proceedings of the 1999 symposium on Interactive 3D graphics*, 1999, pp. 175-182.

ABSTRACT

Lindenmayer Systems (L-Systems) are a method for procedural content generation, which has been used successfully for years. Their drawback is that the rules and parameters of the system must be manually constructed. Genetic Algorithms (GA) are a method for optimization of the existing systems. By combining L-Systems for content generation and GA for the system optimization, it is possible to reach new and high quality content easily, without need for a large amount of time to adjust the rules and parameters.

*Keywords* – Genetic Algorithm, L-System, procedural content generation

**OPTIMIZATION OF PROCEDURALLY GENERATED TREES**

Darko Brdareski, Milan Tomić