4.3.8. GA sa realnim kodiranjem

U dosadašnjim razmatranjima je korišteno kodiranje potencijalnih rješenja u binarne stringove. Iako je ovaj način kodiranja korišten u izvornom GA i vrlo jednostavan, performansa GA sa binarnim kodiranjem nije najbolja u svakoj primjeni. To je razlog korištenja drugih načina kodiranja, od kojih je najrašireniji u primjeni kodiranje potencijalnog rješenja u vektor realnih brojeva. Michalewicz, Bäck, Davis i drugi [4] su pokazali da za većinu realnih primjena performansa GA sa realnim kodiranjem bitno nadmašuje performansu jednostavno GA. Kod GA sa realnim kodiranjem je $\mathcal{A} = \mathcal{R}$, odnosno kao vrijednost lokusa se može pojaviti bilo koji realni broj.

Najjednostavniji pristup je da se za preslikavanje $\mathrm{C} : \boldsymbol{x} \leftrightarrow \boldsymbol{a}$ koristi identitet, odnosno da se direktno kodiraju komponente vektora $\boldsymbol{x}$. U takvom slučaju je $\boldsymbol{x} = \boldsymbol{a}$.

Korištenje kodiranja realnim brojevima zahtijeva i definiranje adekvatnih operatora ukrštanja i mutacije, te se u praksi koristi čitav niz ovih operatora. U nastavku će biti predstavljeni najčešće korišteni.

4.3.8.1 Operatori ukrštanja korišteni u GA sa realnim kodiranjem

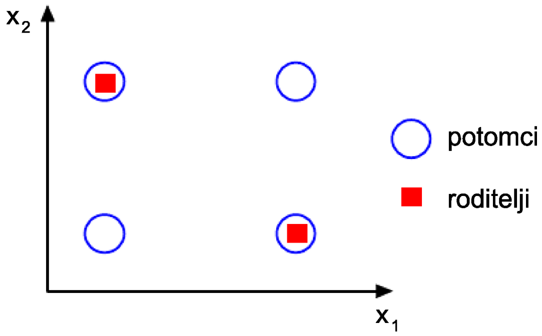
Operatori ukrštanja hromozoma roditelja predstavljeni binarnim stringovima se mogu koristiti i u slučaju predstavljanja hromozoma vektorima realnih brojeva. Međutim, obzirom da je dužina hromozoma u slučaju realnog kodiranja dosta manja nego u slučaju binarnog kodiranja, efikasnost opisanih operatora ukrštanja u pogledu sposobnosti kreiranja novih hromozoma potomaka je manja. Radi toga će biti predstavljeni operatori ukrštanja koji se češće koriste u GA sa realnim kodiranjem.

Diskretna rekombinacija

Ovaj operator ukrštanja se može koristiti neovisno o šemi kodiranja (binarni string, cijeli brojevi, realni brojevi, simboli), kada je hromozom predstavljen vektorom. Svaki od elemenata vektora hromozoma potomka se sa istom vjerovatnoćom uzima od jednog od hromozoma roditelja, prema izrazu [11]:

$$
\begin{aligned}
& x_i^r = x_i^p \cdot z_i + x_i^q \cdot (1 - z_i) \\
& z_i \in \{0,1\} \\
& i \in \{1,2,\ldots,l\}
\end{aligned}
$$

Potomci koji se mogu dobiti primjenom diskretne rekombinacije za slučaj hromozoma predstavljenog vektorom sa dva elementa je prikazan na slici 4.11. Vidi se da se potomci mogu naći u bilo kojem vrhu hiperkocke definirane koordinatama roditelja.



Slika 4.11. Potomci koji se mogu dobiti primjenom diskretne rekombinacije

Međurekombinacija

Operator međurekombinacije (Intermediate Recombination) bira vrijednosti elemenata hromozoma potomka oko ili između vrijednosti elemenata na istoj poziciji hromozoma roditelja, prema izrazu [11]:

$$x_i^r = x_i^p \cdot z_i + x_i^q \cdot (1 - z_i)$$
$$z_i \in [-d, 1+d]$$
$$i \in \{1,2,\ldots,l\}$$

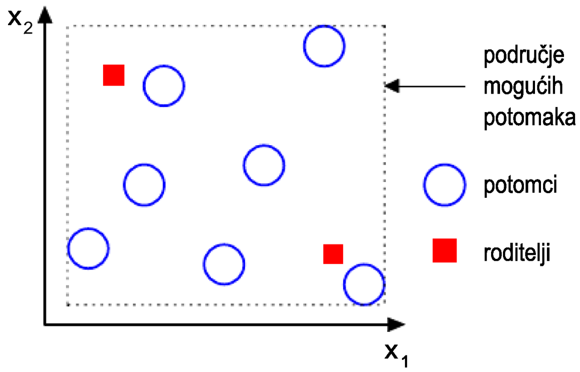
Kod ovog operatora rekombinacije vrijednost faktora z_i se nalazi između vrijednosti $-d$ i $1+d$. Parametar d određuje veličinu područja u kojem se može pojaviti potomak. Tako, ako je $d=0$ potomci se mogu naći bilo gdje unutar hiperkocke definirane vrijednostima elemenata roditelja. Ako se uzme nenulta vrijednost parametra d, hromozomi potomci se mogu naći unutar hiperkocke koja je veća od one definirane elementima vektora hromozoma roditelja. U praksi je poželjno usvojiti nenultu vrijednost parametra, pošto nulta vrijednost parametra d dovodi kroz do postepenog smanjivanja hiperkocke definirane roditeljima. Naime, vrlo je malo vjerovatno da će slučajna vrijednost z_i biti takva da za svaki element hromozoma potomka bude izabrana tačna (granična) vrijednost elementa jednog od hromozoma roditelja. Vrijednost $d=0,25$ uz uniformnu raspodjelu vjerovatnoće pri izboru z_i će obezbjediti održavanje veličine hiperkocke definirane vrijednostima elemenata hromozoma roditelja. Dvodimenzionalni slučaj primjene ovog operatora ukrštanja je predstavljen na slici 4.12.

Linearna rekombinacija

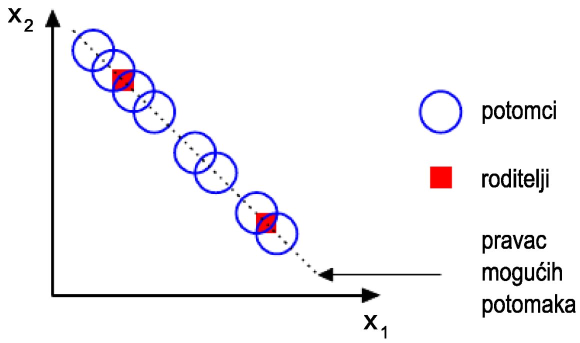
Linearna rekombinacija [11] je vrlo slična prethodno opisanoj, s razlikom da se za odabir svakog elementa hromozoma potomka koristi ista vrijednost faktora z:

$$x_i^r = x_i^p \cdot z + x_i^q \cdot (1 - z)$$
$$z \in [-d, 1+d]$$
$$i \in \{1,2,\ldots,l\}$$

Posljedica ovakvog odabira faktora z je da će se svi mogući potomci nastali primjenom linearne rekombinacije hromozoma roditelja nalaziti na pravcu koji prolazi kroz tačke određene hromozomima roditeljima (slika 4.13). Uloga parametra d kao i njegova preporučena vrijednost su identične kao i kod operatora međurekombinacije.



Slika 4.12. Potomci koji se mogu dobiti primjenom međurekombinacije



Slika 4.13. Potomci koji se mogu dobiti primjenom linearne rekombinacije

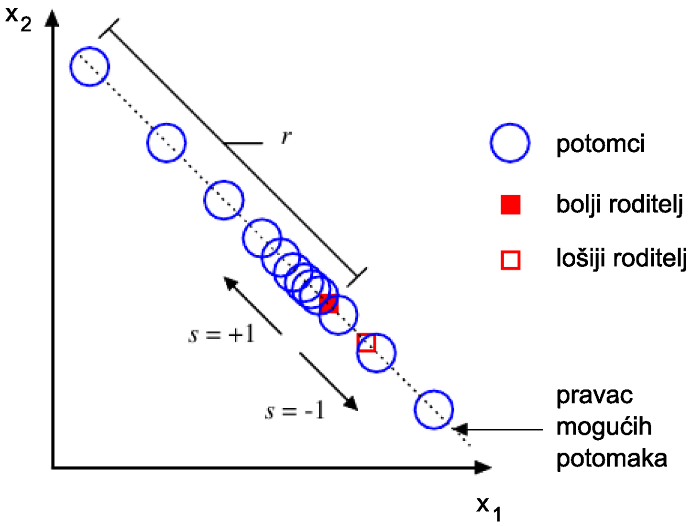
Produžena linearna rekombinacija

Pokazano je da hromozomi potomci formirani linearnom rekombinacijom leže na pravcu između hromozoma roditelja, ili malo izvan tog područja. Produžena linearna rekombinacija [11] predstavlja generalizaciju prethodno opisane. Sada hromozomi roditelji samo definiraju pravac na kome leže potomci, ali je područje na kome se oni mogu naći sada puno šire. Međutim, potomci sada neće biti uniformno raspoređeni po pomenutom području, nego je vjerovatnoća generiranja potomaka bliže roditeljima veća i opada sa udaljenošću od roditelja. Pri tome, veća je i vjerovatnoća generiranja potomaka u smjeru prema hromozomu roditelju sa većim fitnessom. Produžena linearna rekombinacija je predstavljena izrazom:

$$x_i^r = x_i^p + b_i \cdot c_i \cdot e \cdot \frac{x_i^q - x_i^p}{\left\| \boldsymbol{x}^p - \boldsymbol{x}^q \right\|}, \qquad i = 1, 2, \ldots, n$$
$$e = 2^{-k \cdot u}, \qquad u \in [0, 1]$$
$$c_i = c \cdot (x_{i\,max} - x_{i\,min})$$
$$b_i \in \{-1, 1\}$$

Parametar b_i određuje smjer u kome će biti generiran homozom potomak, c_i određuje maksimalnu veličinu pomaka, a e relativnu vrijednost pomaka (obično oko 0,1). Parametar u predstavlja slučajni broj sa uniformnom raspodjelom između 0 i 1, a k koeficijent, čija vrijednost je obično u rasponu od 4 do 20. Ukoliko se za vrijednost parametra b_i sa istom vjerovatnoćom biraju vrijednosti -1 i +1, rekombinacija će biti neusmjerena. Međutim, ukoliko je vjerovatnoća izbora vrijednosti +1 veća od 0,5 rekombinacija će biti usmjerena prema hromozomu roditelju sa boljim fitnessom.

Slika 4.14 predstavlja potomke koji se mogu dobiti primjenom ovog operatora ukrštanja.



Slika 4.14. Potomci koji se mogu dobiti primjenom produžene linearne rekombinacije

4.3.8.2 Operatori mutacije korišteni u GA sa realnim kodiranjem

Operator mutacije i kod GA sa realnim kodiranjem slučajno modificira hromozom. Postoji nekoliko varijanti ovog operatora, a najčešće korištene su date u nastavku.

Slučajna vrijednost elementa vektora

Najjednostavnija varijanta operatora mutacije koja se može primjeniti na hromozom predstavljen vektorom realnih brojeva je zamjena nekog od njegovih elemenata slučajnim brojem sa uniformnom raspodjelom:

$$x_{i\ mut} \in [x_{i\ min}, x_{i\ max}]$$

Međutim, pokazano je da izbor ovakvog operatora mutacije u većini primjena ne daje najbolje rezultate [4]. Poželjnije djelovanje operatora mutacije bi se ogledalo u generiranju hromozoma u okolini hromozoma koji se mutira, pri čemu su vjerovatnije mutacije koje generiraju hromozom bliže polaznom.

Slučajno modificiranje elementa vektora

Bolji rezultati se obično postižu dodavanjem slučajne vrijednosti elementu hromozoma:

$$x_{i\,mut}=x_i+\delta$$
$$\delta\in[-d,d]$$

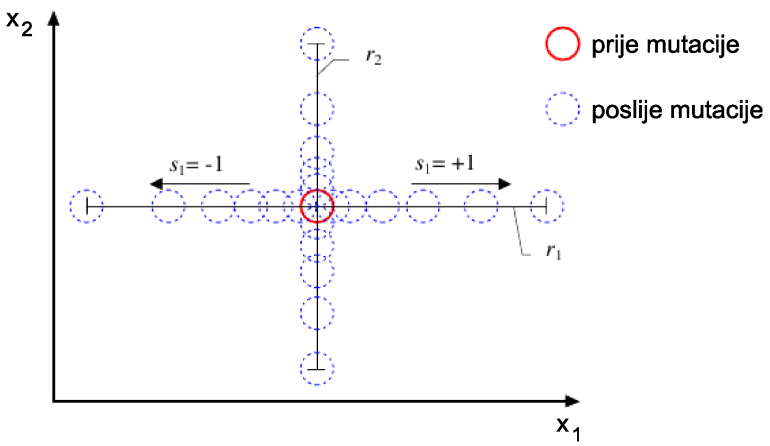
Ovdje δ predstavlja slučajan broj, sa većom vjerovatnoćom pojave malih vrijednosti.

Nedostatak ovog operatora mutacije se sastoji u potrebi da se ispravno odabere parametar d koji određuje opseg vrijednosti za koju se mutira element hromozoma.

Ono što bi bilo poželjno je da se izbjegne potreba za podešavanjem ovog operatora mutacije. Takav operator mutacije je opisan izrazom:

$$x_i^q=x_i^p+b_i\cdot c_i\cdot e$$
$$e=2^{-k\cdot u},\qquad u\in[0,1]$$
$$c_i=c\cdot(x_{i\,max}-x_{i\,min})$$
$$b_i\in\{-1,1\}$$

Tumačenje parametara ovog operatora mutacije je identično kao kod operatora produžene linearne rekombinacije. Efekat operatora je predstavljen na slici 4.15.



Slika 4.15. Slučajna mutacija elemenata hromozoma

4.3.9. *Osvrt na još neke načine kodiranja*

GA sa binarnim i GA sa realnim kodiranjem se najčešće koriste kao opći algoritmi za rješavanje različitih problema.

Međutim, osim ova dva načina kodiranja u primjeni se može naći još niz drugih načina kodiranja, koji su bolje usklađeni sa problemom koji se rješava. Ovdje ćemo se kratko osvrnuti na neke od njih, kao što su:

- permutacije,
- konačni automati,
- parsiranje,
- diploidna reprezentacija,
- hijerarhijska reprezentacija,
- lista simbola.

Osnovni uslov za primjenu ovih reprezentacija je da je jasno definirano preslikavanje između potencijalnih rješenja i njihove reprezentacije u vidu hromozoma, da su definirani specifični operatori i mehanizmi koji će procesirati populaciju, te da se svakom hromozomu jednoznačno može dodijeliti fitness.

Permutacije

Postoji niz praktičnih problema, kao što su TSP, raspoređivanje ograničenih resursa i sl., kod kojih potencijalno rješenje predstavlja niz vrijednosti elemenata poredanih u određenom redoslijedu. Svaki mogući hromozom sadrži iste elemente, a hromozomi se međusobno razlikuju po redoslijedu kojim se ti elementi pojavljuju.

Prirodan način kodiranja za takve probleme je korištenje permutacija. Prema Knuthu [4], permutaciju konačnog skupa predstavlja poredak njegovih elemenata u vektor. Ako konačni skup ima n elemenata, postoji $n!$ mogućih poredaka tih elemenata. Radi toga je moguće, uz specificiranje sortiranog redoslijeda permutacija, definirati obostrano jednoznačno preslikavanje između skupa permutacija i skupa cijelih brojeva. Svaki od cijelih brojeva predstavlja indeks permutacije u sortiranoj listi permutacija. GA sada može pretraživati indekse permutacija umjesto permutacija direktno.

Konačni automati

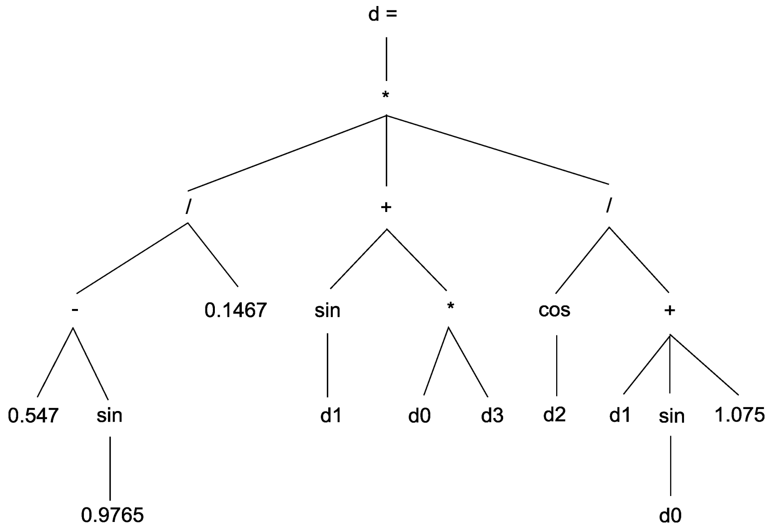
Predstavljanje hromozoma konačnim automatom se koristi za rješavanje problema kod kojih je potrebno odrediti sekvencu simbola u ovisnosti o vremenu ili nekoj drugoj slobodnoj promjenljivoj. Za kodiranje konačnih automata u hromozom se koriste različiti načini.

Parsiranje

Ovaj način prezentacije hromozoma se bazira na predstavljanju potencijalnog rješenja drvetom (parse tree). Pogodan je kada je potencijalno rješenje algoritam, odnosno računarski program. Drvetom se mogu predstaviti vrlo kompleksne matematičke funkcije ili algoritmi (slika 4.16), koji se onda modificiraju od strane GA. Za definiranje ovog načina kodiranja je potrebno uspostaviti obostrano jednoznačno preslikavanje između drveta kodiranog u hromozom i algoritma kojeg ono predstavlja.

Diploidna reprezentacija

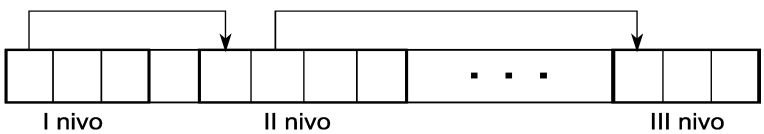
Diploidna reprezentacija za svaki lokus hromozoma definira dvije vrijednosti. Uvodi se poseban mehanizam koji za svako potencijalno rješenje određuje koja od dvije vrijednosti će se koristiti. Najčešći pristup je da se zajedno sa populacijom evoluira i mapa dominantnosti, koja određuje dominantnu vrijednost za svaki od lokusa. Ovaj način reprezentacije je inspiriran postojanjem parova gena (dominantni i recesivni) kod viših živih organizama.



Slika 4.16. Predstavljanje izraza drvetom

Hijerarhijska reprezentacija

I hijerarhijska reprezentacija je inspirirana mehanizmom funkcioniranja genoma kod živih organizama. Naime, umjesto jednostavnog mapiranja između spoljašnjih osobina i gena, kod živih organizama su spoljašnje osobine posljedica vrlo kompleksnog dekodiranja genoma. Naime, osim gena koji kodiraju spoljašnje osobine postoje kontrolni geni koji uključuju i isključuju druge gene (slika 4.17). Na ovaj način je uvedena hijearhija i upravo ova ideja se koristi za realiziranje hijerarhijske reprezentacije u GA.



Slika 4.17. Hijerarhijska reprezentacija

Lista simbola

GA je dovoljno općenit algoritam da ne zahtijeva definiranje bilo kakve funkcije egzaktne funkcije kodiranja, niti zahtijeva da elementi hromozoma imaju matematičku interpretaciju. U tom smislu, hromozom se može predstaviti bilo kakvim nizom simbola, sve dok se mogu definirati operatori i mehanizmi koji će procesirati takve hromozome, te dok se može utvrditi njihov fitness.

4.3.9. Nestandardni operatori i elementi GA

Osim gore navedenih elemenata i operatora, koji se susreću praktično u svakoj implementaciji GA, egzistira još nebrojeno mnogo nestandardnih operatora i elemenata [5], kao što su npr. operatori inverzije i promjene poretka, mehanizmi diploidnih i poliploidnih hromozoma i dominantnosti (već spomenuti u okviru diploidne reprezentacije), uvođenje segregacije, translokacije, različitih mehanizama koji impelentiraju podvrste i sl. [12] O njima ovdje neće biti riječi, a više informacija se može pronaći u literaturi [3], [4], [5], [6], [12].

2.2.1. Vrijednosti parametara GA

Genetički algoritam predstavlja u stvari adaptivni mehanizam koji u okviru populacije čuva informaciju i statistiku o prostoru koji se pretražuje i njegovoj vezi sa prostorom vrijednosti. Kroz generacije populacija postepeno evoluira i cilj realizacije efikasnog GA je da se postigne što je moguće brža konvergencija ka globalno optimalnoj tački prostora koji se pretražuje.

GA predstavlja algoritam sa velikim brojem stepeni slobode. Zadatak podešavanja GA za dostizanje optimalne performanse podrazumijeva određivanje vrijednosti njegovih parametara koji će osigurati takvu performansu. Neki od pomenutih stepeni slobode GA su[3]:

- veličina populacije,
- dužina stringa koji predstavlja hromozom,
- vrsta i parametri mehanizma selekcije,
- vrsta i parametri operatora ukrštanja,
- vrsta i parametri operatora mutacije,
- broj najboljih jedinki koje će se prenijeti u narednu generaciju (elitizam).

Veličina populacije određuje kvalitet uzorkovanja prostora koji se pretražuje, i neophodno je da broj hromozoma koji čine populaciju bude dovoljan da osigura da početna populacija sadrži što je moguće više informacija o odnosu prostor stanja-prostor vrijednosti. Jedini način da GA nadoknadi nedostatak ovakve informacije, ako to uopšte uspije, je u istraživanju prostora stanja. S druge strane, veličina populacije utiče na vrijeme neophodno da se svi hromozomi populacije procesiraju unutar jedne generacije, tako da je neophodno odabrati kompromisnu veličinu populacije.

Dužina binarnog stringa hromozoma određuje preciznost sa kojom se predstavljaju tačke iz prostora stanja i jasno je da je poželjno da ta preciznost bude što je moguće veća. Međutim, obzirom na binarnu

3 Ovdje se razmatra samo standardni GA sa binarnim kodiranjem hromozoma.

reprezentaciju u hromozomu, veća preciznost znači i veću dužinu binarnog stringa, a samim tim i povećanje diskretnog binarnog prostora koji se pretražuje. S druge strane, povećanje prostora u kome djeluju operatori GA bitno utiče na efekat njihovog djelovanja, tako da je i ovdje neophodno odabrati onu preciznost koja je zadovoljavajuća sa stanovišta problema za čije se rješavanje primjenjuje GA.

Već je rečeno da je elitizam neophodan da bi GA bio dovoljno efikasan za primjenu u optimizaciji, pošto je postojanje elitizma uslov za konvergenciju. Radi toga je neophodno da se u narednu generaciju bez izmjena prenese barem najbolji hromozom tekuće populacije, a povećanje broja hromozoma koji su preneseni povećava i broj generacija neophodnih za konvergenciju, jer se time usporava djelovanje operatora GA.

Vrsta i parametri mehanizma selekcije i operatora koji osiguravaju efikasnost GA su dosta problemski specifični i nemoguće je donijeti generalan sud o njihovom izboru.

Određivanje optimalnih postavki parametara GA je bilo predmet intenzivnih istraživanja još od njegove pojave i vrlo rano su se pojavili empirijski određene vrijednosti parametara.. De Jong je tako empirijski odredio vrijednosti parametara SGA koje obezbjeđuju njegovu efikasnost na skupu određenih testnih funkcija [7], [8] (tabela 4.1). Ove vrijednosti i danas predstavljaju vodilju za određivanje statičkih postavki vrijednosti parametara GA. Grefenstette je za određivanje vrijednosti parametara GA primjenio meta-GA pristup. Prema ovom pristupu, određivanje optimalnih vrijednosti parametara GA, koje osiguravaju njegovu efikasnost za dati skup testnih funkcija, povjereno je drugom nivou GA koji je koristio postavke na osnovu De Jongovih rezultata. Treći skup vrijednosti parametara koji se danas primjenjuje su formirali Schaffer, Caruana, Eshelman i Das.

Postavka	De Jong	Grefenstette	Schaffer et al.
Veličina populacije	50 – 100	30	20 – 30
Vjerovatnoća ukrštanja	~ 0,6	0,95	0,75 – 0,95
Vjerovatnoća mutacije	0,001	0,01	0,005 – 0,01

Tabela 2.1. Postavke parametara GA

Sva tri slučaja su razmatrala GA sa ukrštanjem u jednoj tački, binarnom mutacijom i elitizmom. Često se vrijednosti parametara daju i u formi empirijskih izraza. Tako je Schaffer je predložio za vjerovatnoću mutacije:

$$p_m = \frac{1,75}{m \cdot \sqrt{l}}$$

gdje je m broj hromozoma u populaciji, a l dužina binarnog stringa hromozoma. Na sličan način, prema Mühlenbeinu je optimalna vrijednost vjerovatnoće mutacije:

$$p_m = \frac{1}{l}$$

Treba napomenuti da su sve navedene vrijednosti parametara određene sa ciljem da se što je moguće više poveća efikasnost primjene GA za određivanje optimuma funkcija iz testnog skupa, i za očekivati je da će GA sa istim vrijednostima parametara pokazati lošije rezultate kada se primjeni na funkcije koje imaju karakteristike različite od onih iz skupa testnih funkcija. Isto tako, za bilo koji skup funkcija je moguće na sličan način odrediti optimalne vrijednosti parametara. Međutim, takav pristup je prilično neefikasan kada se GA želi primjeniti u uslovima koji nisu strogo određeni. Osim toga, bolju performansu GA može postići kada se vrijednosti parametara ne drže konstantnim kroz generacije nego se mijenjaju, pri čemu je nemoguće dati univerzalnu ovisnost ovih vrijednosti kroz generacije.

4.4. Specifičnosti GA kao algoritma optimizacije

Konvergencija GA kao tipičnog metaheurističkog algoritma podrazumijeva, kako je već rečeno da:

$$\begin{aligned} & p\left(\left|y\left(\boldsymbol{x}^{*\mathrm{k}}\right)-y^*\right|\le\varepsilon\right)\to 1 \\ & \text{za } \mathrm{k}\to\infty \\ & \forall\, \varepsilon>0 \end{aligned} \tag{4.15}$$

GA bez elitizma nije globalno konvergentan, što su pokazali De Jong i Rudolph. Sljedeća bitna osobina GA kao algoritma optimizacije je brzina njegove konvergencije ka globalnom optimumu, odnosno dinamika konvergencije. Rappl je prvi dao razmatranje reda konvergencije i pokazao da ukoliko fitness ispunjava određene uslove i za slučaj primjene operatora mutacije sa konstantnom vjerovatnoćom mutacije za očekivanu vrijednost kriterija u odnosu na optimalnu vrijedi [12]:

$$E\left(f\left(\boldsymbol{a}^{*\mathrm{k}}\right)-f^*\right)=k^{-\Theta\left(\frac{1}{m}\right)} \tag{4.16}$$

odakle se vidi da GA pod datim uvjetima ispoljava eksponencijalnu konvergenciju ka globalnom optimumu.

Sljedeći bitan element za teorijsko razmatranje GA predstavljaju mjere performanse, koje mogu biti lokalne (za jednu generaciju) ili globalne (za veći broj generacija). Lokalne mjere performanse mogu biti u domenu prostora vrijednosti fitnessa ili u domenu prostora stanja E^n. Prve se nazivaju dobitkom na kvalitetu [13]: