Univerzitet u Novom Sadu

Fakultet tehničkih nauka

Dokumentacija za projektni zadatak

Studenti: Vasilije Marković, SV15/2021

Predmet: Nelinearno programiranje i evolutivni algoritmi

Broj projektnog zadatka: 11

Tema projektnog zadatka: Genetski algoritam, „black-box“ optimizacija

# Opis problema

Zadatak je implementirati genetski algoritam i iskoristiti ga za pronalazak minimuma date funkcije *f*. Funkcija kao argument prima jedan vektor od 60 realnih vrednosti, vraća realan broj i nije poznata u analitičkom smislu, međutim za bilo koji ulaz moguće je dobiti njenu vrednost.

# Struktura programa

Program je najvećim delom sadržan u Pajton funkciji koja predstavlja algoritam. U main delu se ona poziva sa željenim parametrima. Unutar funkcije se na početku izvršava faza Kodiranja, to jest nasumično generiše inicijalna populacija pomoću zadate veličine populacije. Algoritam zatim ulazi u for petlju koja iterira broj puta koji je takođe prosleđen kao parametar funkcije. U petlji se u svakom koraku odvijaju preostali koraci genetskog algoritma: Selekcija, Ukrštanje, Mutacija i Selekcija nove populacije.

# Način implementacije operatora mutacije i ukrštanja

U ukrštanje ulazi rangirana lista koja sadrži populaciju. Vrši se onoliko iteracija koliko je polovina broja populacije i u svakoj se odredi po jedan par roditelja na sledeći način: rangovi svih jedinki se pomnože nasumično generisanim brojem od 0 do 1, te one 2 koje imaju najveće proizvode se biraju kao roditelji – postupak ruletskog ukrštanja sa rangiranjem. Dvoje dece koju oni naprave se po hromozomima nalaze između svojih roditelja, simetrično ukrštena.

Kad se izgenerišu sva deca, ona ulaze u fazu mutacije: svakom detetu se za svaki hromozom odredi da li će biti mutiran, sa verovatnoćom zadatom među parametrima algoritma. Ukoliko nasumice generisani broj od 0 do 1 bude manji ili jednak verovatnoći, detetu se vrednost hromozoma sabere ili oduzme za novi nasumice generisani broj od 0 do 0.5.

# Strategija odabira jedinki za ukrštanje

Kao što je rečeno biranje jedinki unutar populacije koje će praviti decu korišćeno je ruletsko rangiranje sa: nakon što su sve jedinke sortirane po fitnesu, ulaze u fazu Ukrštanja koja je objašnjena u prethodnoj tački.

Za odabir članova populacije koji će ući u sledeću generaciju iz liste prvo se sortira lista *newpop* u kojoj su zamapćene sve jedinke koje su ili učestvovale u ukrštanju u toj generaciji ili su deca, tj. proizvod faze ukrštanja. Lista se sortira po vrednosti, i onda se jedan po jedan član iz nje ubacije u listu *popul*, poštujući maksimalni broj roditelja koji sme da se pojavi u novoj populaciji kao i veličinu populacije. Taj broj je zadat kao jedan od parametara algoritma.

U listi *newpop* je za svaku jedinku zapamćeno da li je u toj generaciji učestvovala kao roditelj ili kao dete. *popul*, kada se napuni, ulazi u narednu iteraciju kao trenutna celokupna populacija.

# Odabir parametara algoritma

Odabrani parametri algoritma su sledeći:

*lowerLimit* i *upperLimit* – minimalni i maksimalni realni broj koji svaki od hromozoma u inicijalno generisanoj populaciji može imati. U fazi kodiranja se svaka jedinka generiše sledećom linijom:

w = list(np.random.uniform(lowerLimit, upperLimit, 60))

*popSize* – konstantna veličina populacije, koristi se pri popunjavanju inicijalne populacije i pravljenju nove prilikom biranja koja deca i roditelji će opstati na kraju jedne iteracije.

*mutatProb* – verovatnoća mutacije, broj između 0 i 1, u fazi Mutacije određuje koji procenat hromozoma u jedinki će približno podleći mutaciji.

*maxParentCount* – maksimalni broj roditelja koji će preživeti u narednu generaciju, tokom formiranja nove populacije postupak elitizma diktira da se najviše ovaj broj roditelja može ubaciti u novu listu.

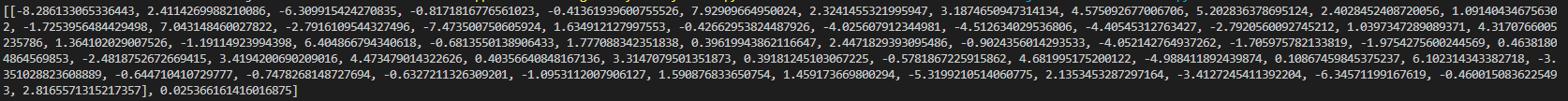
*iterations* – broj iteracija kroz koje algoritam prolazi.

# Rezultati algoritma

Najbolji rezultat pronađen je za sledeći odabir parametara algoritma:

lowerLimit = -5; upperLimit = 5; popSize = 600; mutatProb = 0.3;  
maxParentCount = 10; iterations = 20

i rešenje koje je dobijeno je sledeće:



odnosno .

Sa obzirom na to da izbor parametara, kao i nasumice generisana početna populacija snažno utiču na rezultat algoritma, on često na izlazu dobija neku potpuno različitu tačku sa drugačijom vrednošću funkcije u njoj. Međutim, dobijeni minimum nikada ne prevagne 0.09, niti ode ispod 0.02, te je red veličine istinskog rešenja određen relativno precizno.