

# Evolucija programa upotrebnom genetičkog programiranja

Kristina Petrović, Nikola Milovanović

Jun 2020

## Uvod

Genetičko programiranje je tehnika koja omogućava rešavanje nekog problema bez potrebe za stvaranjem programa koji rešava zadati problem.

- Način rada genetskog programiranja je upotreba evolutivnog procesa u stvaranju računarskih programa
- Jedinke u populaciji genetičkog programa predstavljaju računarske programe
- Programi su u većini primera genetskog programiranja napisani u obliku stabla
- Novi programi se dobijaju jednostavnim manipulacijama nad već postojećim stablima

# Elementi genetičkog programiranja

- Podacijski i funkcijski elementi rešenja
- Funkcija prilagođenosti
- Parametri genetskog programiranja
- Uslov zaustavljanja
- Građa rešenja

## Odabir podatkovnih i funkcijskih elemenata

Zahtevi koje treba zadovoljiti prilikom odabira podatkovnih i funkcijskih elemenata:

- **potpunost** - označava da je moguće izraziti rešenje problema uz pomoc odabranih podatkovnih i funkcijskih elemenata
- **zatvorenost** - označava sposobnost funkcijskog elementa da prihvata rezultat od bilo kojeg drugog funkcijskog ili podatkovnog elementa

## Funkcija prilagođenosti

- Funkcija prilagođenosti ili fitnes funkcija je najzaslužniji element u rukovođenju evolucijom i upravljanjem kretanja celokupne populacije
- Treba da nagrađuje ne samo bolja rešenja, već i sva poboljšanja pronađena tokom evolucije
- U većini primera određivanje fitnes funkcije oduzima najviše procesorskog vremena

## Parametri genetskog programa

- Izbor odgovarajućih vrednosti zavisi od konkretne primene
- Postavljanje parametara postaje složenije sa povećanjem zavisnosti između njih
- Jedan od najvažnijih parametara u genetskom programiranju je veličina populacije, za netrivialne probleme preporučuje se da ona bude veća od 1000 jedinki

## Uslov zaustavljanja

- Pronalazak tačnog rešenja
- Dostizanje predefinisane broja generacija
- Ako u zadatom broju uzastopnih generacija nije postignuto poboljšanje fitnes funkcije najbolje jedinice

# Prednosti i nedostaci genetičkog programiranja

## Prednosti upotrebe:

- Može rešavati proizvoljni optimizacioni problem
- Rešenjem se dobija algoritam rešavanja koji se može primeniti nad proizvoljnim podacima
- Jednostavnost izvođenja

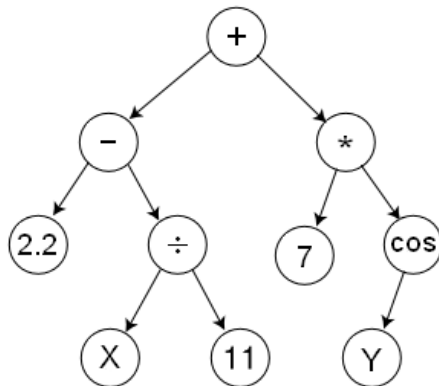
## Nedostaci upotrebe:

- Preuranjena konvergencija
- Spora konvergencija



- Jedinske se predstavljaju najčešće upotrebom strukture stabla
- U listovima se nalaze terminali
- Terminale čine ulazni parametri, numeričke konstante...
- U ostalim čvorovima se nalaze funkcije
- Funkcije mogu biti logičke, aritmetičke, različite programske konstrukcije...

Reprezentacija jedinice upotrebom stabla:



- Genetičko programiranje opisuje proces evolucije u prirodi
- Varijacija gena je neophodna
- Bolje i prilagodljivije jedinke opstaju
- Genetički operatori:
  - Selekcija
  - Ukrštanje
  - Mutacija

## Selekcija

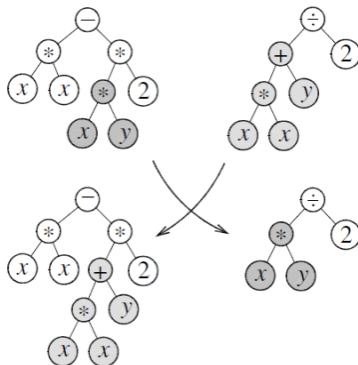
Proces kojim se određene jedinke biraju iz trenutne generacije za roditelje naredne generacije se naziva selekcija.

- Najčešće metode selekcije su turnirska i ruletska
- Turnirska selekcija:
  - Jedinke se izvlače slučajno
  - Poređenje se vrši preko funkcije prilagođenosti
  - Najbolje jedinke učestvuju u reprodukciji
- Ruletska selekcija:
  - Jedinke se izvlače sa verovatnoćom:  $p_i = \frac{f(i)}{\sum_j^N f(i)}$   
f(i) - funkcija prilagođenosti za i-tu jedinku

## Ukrštanje

Ukrštanje predstavlja razmenu genetskog materijala između dve jedinke populacije.

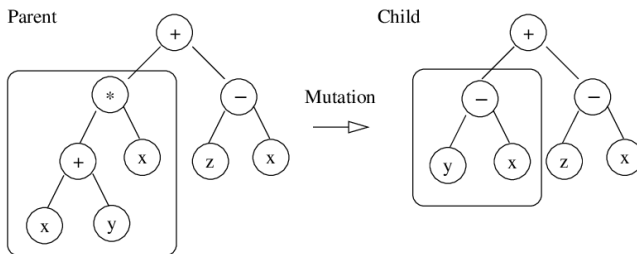
- Na slučajan način se odaberu tačke ukrštanja roditelja
- Zamene se podstabla sa korenom u tački ukrštanja



## Mutacija

Mutacija predstavlja izmenu slučajno izabranog gena jedinke.

- Primjenjuje se da jedinke vremenom ne bi bile slične
- Preporučljivo da verovatnoća pojave mutacije bude što manja
- Podstablo se menja slučajno generisanim podstablom



## Simbolička regresija

Simbolička regresija predstavlja postupak pronalaženja matematičkog izraza iz empirijskih podataka.

- Pronalazi se izraz koji se sastoji od članova primitivnog skupa
- Evaluacija izraza što približnija evaluaciji funkcije

# Praktični problem

- Problem je rešen u programskog jeziku Python, pomoću metoda DEAP biblioteke
- Ciljne funkcije:  $x^2 - 5x$ ,  $x^4 + x^3 + x^2 + x$ ,  $x^6 - 2 * x^4 + x^2$ ,  $\sin(\pi/4 + 2 * x)$ ,  $\sin(3 * x^3 - x^2/7)$ ,  $N(0, 1)$
- Terminali:  $\{x, -1, 1\}$
- Funkcije:  $\{+, -, *, /, \sin, \cos\}$
- Veličina populacije, broj generacija, verovatnoća mutacija i ukrštanja i broj članova turnirske selekcije u različitim eksperimentima imaju različite vrednosti

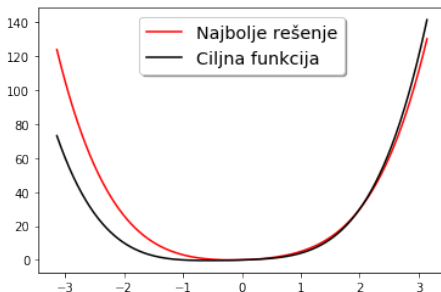


# Eksperimenti

Veličina populacije	Broj generacija	p mutacije	Turnirska selekcija	Metrika greške
50	30	0,3	2 jedinke	Kvadratna

Ciljna funkcija:  $x^4 + x^3 + x^2 + x$

Najbolja jedinka:  $x^2 + (x + (x^2 * (1 + (1 + x^2))))$

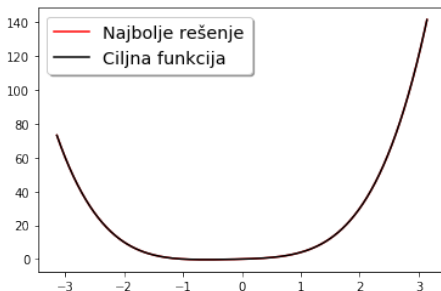


# Eksperimenti

Veličina populacije	Broj generacija	p mutacije	Turnirska selekcija	Metrika greške
300	30	0,1	2 jedinke	Kvadratna

Ciljna funkcija:  $x^4 + x^3 + x^2 + x$

Nabolja jedinka:  $((1 - (1 - x^2)) - (\cos(-x) + 1)) + ((x * (x + 1)) * (x^2 + \sin(x)))$

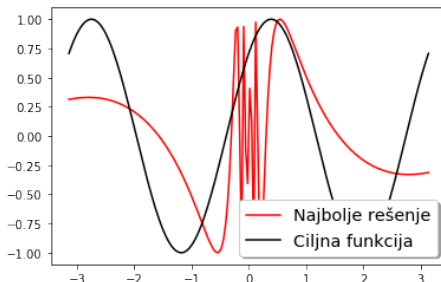


# Eksperimenti

Veličina populacije	Broj generacija	p mutacije	Turnirska selekcija	Metrika greške
300	5	0	5 jedinki	Apsolutna

Ciljna funkcija:  $\sin(\pi/4 + 2 * x)$

Najbolja jedinka:  $\sin(\cos(x)/x)$

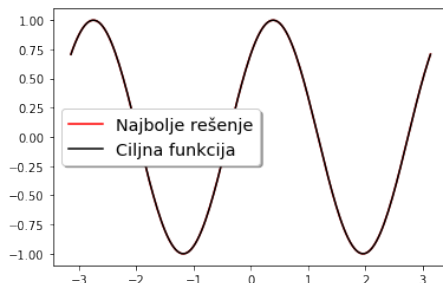


# Eksperimenti

Veličina populacije	Broj generacija	p mutacije	Turnirska selekcija	Metrika greške
300	30	0,3	5 jedinki	Apsolutna

Ciljna funkcija:  $\sin(\pi/4 + 2 * x)$

Najbolja jedinka:  $\cos(\sin((( -x + x) + 1)) + (-x - x))$



Nastanak živoga svijeta možemo poistovetiti sa generiranjem nulte generacije. Nastanak prvih jednoćelijskih organizama i njihova evolucija, preko koje se ukrštanjem i mutiranjem gena došlo do složenijih organizama savršeno odgovara ukrštanju i mutaciji unutar genetičkog programa.

Oponašanjem prirodne evolucije omogućava se rešavanje najrazličitijih problema što inspiriše nastavak istraživanja i evoluciju algoritama u tom pravcu.

Hvala na pažnji!

John Koza, Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems, 1990

A Aho, M.S. Lam, R. Sethi, and J. Ullman. Compilers: principles, techniques and tools. Addison-Wesley Longman Publishing Co., 2006.

[https://en.wikipedia.org/wiki/Genetic\\_programming](https://en.wikipedia.org/wiki/Genetic_programming)

<http://poincare.matf.bg.ac.rs/kartelj/nastava/RI2019/07.Genetsko.programiranje.pdf>

<https://deap.readthedocs.io/en/master/>

<https://www.mi.sanu.ac.rs/jkratica/>