
Genetski algoritam

— Križanje i Mutacija —

Izradio: Ivan Laković

Ponavljjanje

- imitacija proces prirodne selekcije
- pripada klasi evolucijiskih algoritama
- koristi se pri rješavanju optimizacijskih problema
- iterativna metoda
- problem kodiranja (reprezentacije)
- problem inicijalizacije
- selekcija (problem odabira)
- fitness function (funkcija dobrote)

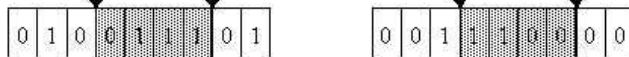
Križanje

- iz dvoje ili više “roditelja” (rješenja) nastaje “dijete” (novo rješenje)
- kombiniramo gene roditelja
- novonastala djeca imaju mnogo sličnosti s roditeljima
- nadamo se da će djeca biti “bolja” rješenja

Parents:



Children:

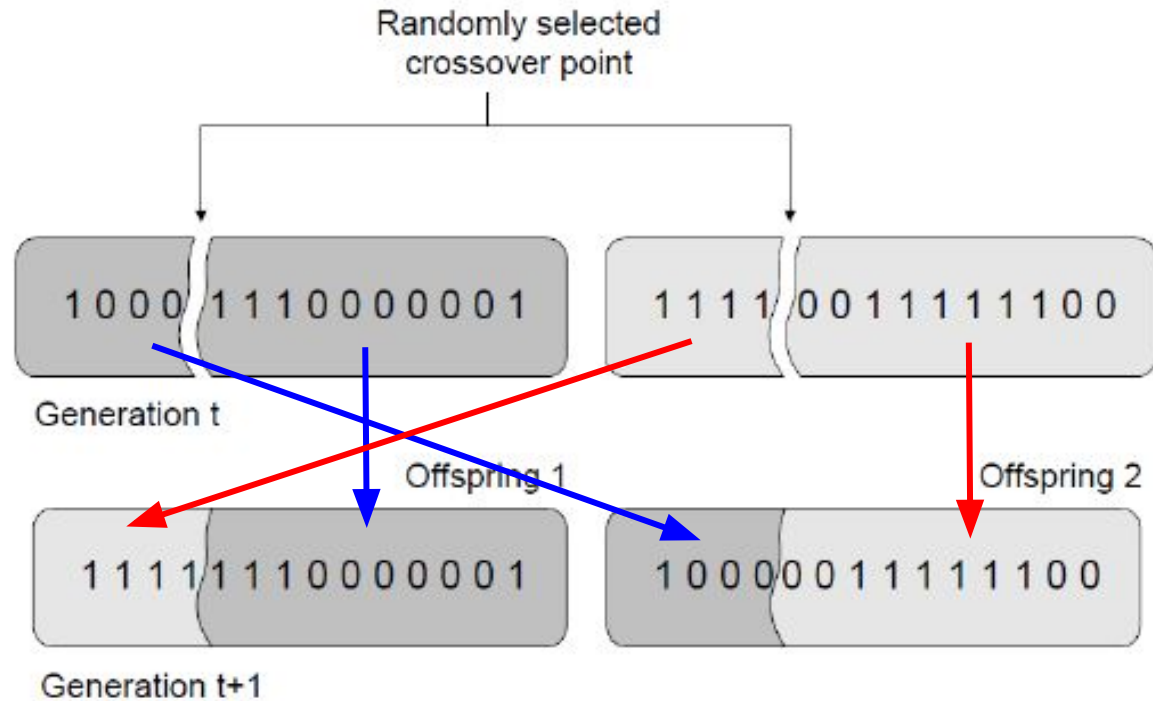


Vrste križanja

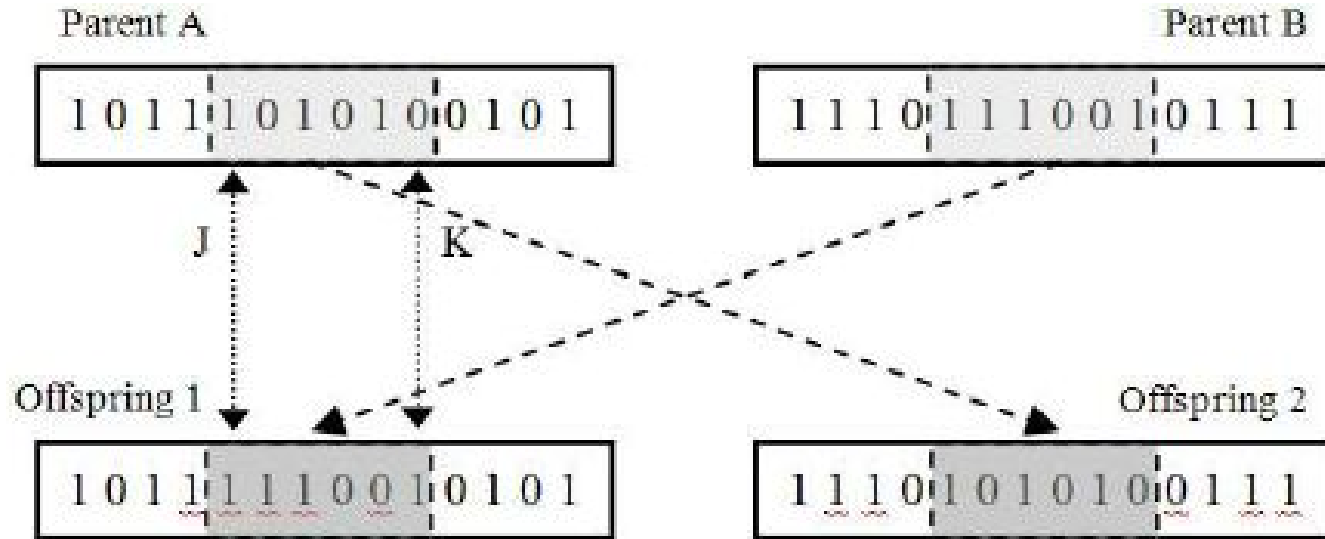
- križanje s jednom točkom prekida (2 roditelja)
- križanje s t-točaka prekida (2 ili više roditelja)
- uniformno križanje
- cikličko križanje
- parcijalno usklađeno križanje

Postoje i mnoge druge vrste križanja koje se po potrebi problema implementiraju.

Križanje s jednom točkom prekida



Križanje s t-točka prekida



Uniformno križanje

- dodijelimo jednom roditelju “glavu”, a drugom “pismo” novčića
- za svaki gen bacamo novčić
- prvo dijete dobiva gen dobitnika, a drugo komplement

parents

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

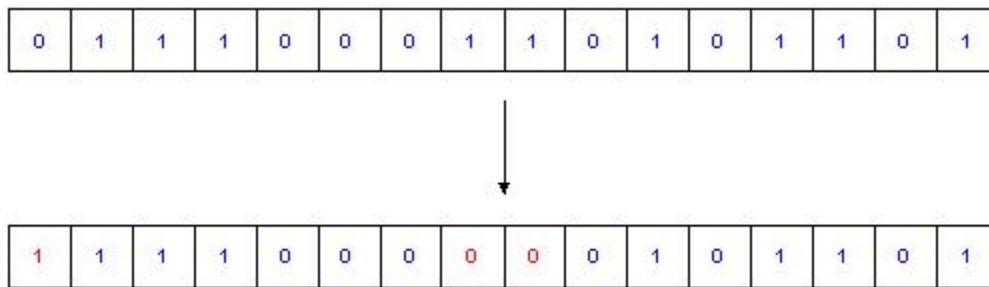
children

0	1	0	0	1	0	1	1	0	0	0	1	0	1	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1	0	1	1	0	0	0	0	1	1	1	0	1	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Mutacija

- operator mutacije djeluje kao unarni operator
- unutar kromosoma (koda rješenja) nad svakim genom birmo hoće li doći do mutacije
- uobičajeno je da operator mijenja gen s vrijednosti između 1% i 5%
- moguće su i drugačije implementacije operatora npr. negacija, ...



Problemi

- zapinjanje u lokalnom ekstremu
- implementacija križanja/mutacije kod ne binarne reprezentacije rješenja
- određivanje završetka traženja
- često računanje fitness funkcije koja ne mora biti jednostavna
- povećanjem kompleksnosti problema računanje često raste eksponencijalno
- nemogućnost korištenja kod problema točno/netočno

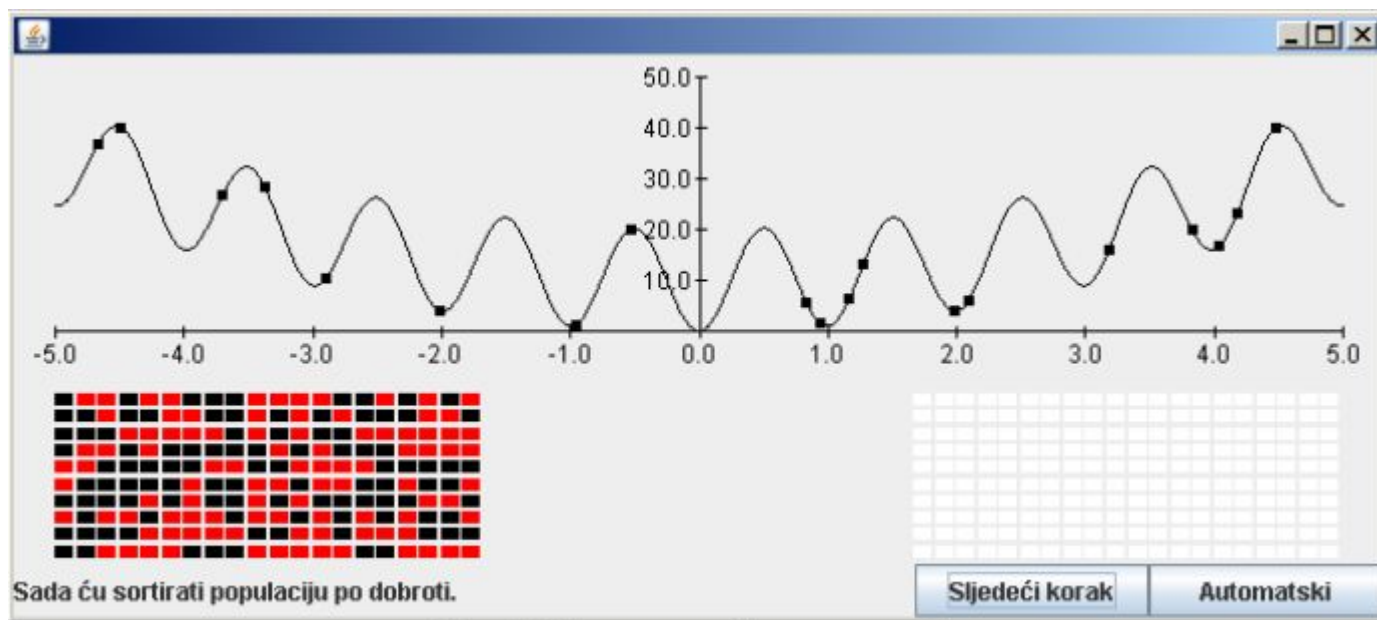
Prednosti

- GA je odličan za pronalaženje “skoro” globalnog ekstrema u kompleksnim prostorima
- u odnosu na primjerice hill-climbing algoritam teže zapne u lokalnom ekstremu (mutacija)
- GA ne nema nikakve zahtjeve na glatkost prostora na kojem pretražuje (Newtonova metoda)
- GA je relativno brz pri nalaženju globalnog ekstrema posebice ako je podešen za traženi problem
- GA je moguće paralelizirati

Rezime

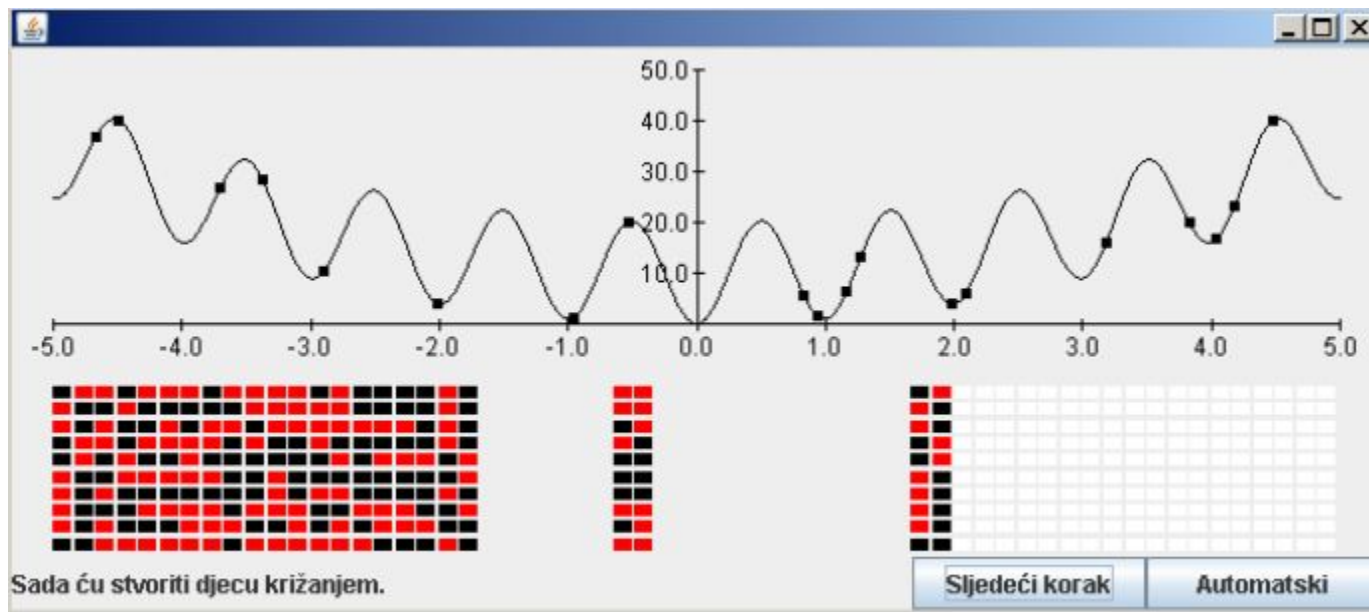


Primjer



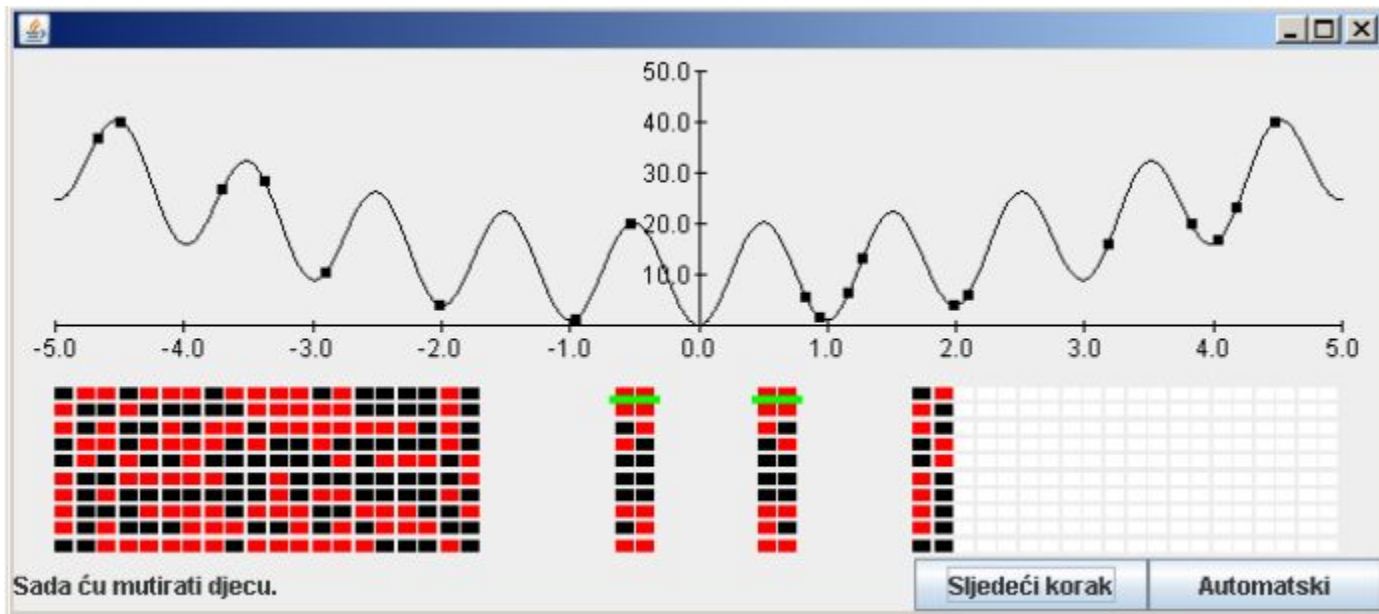
Slika 2-6 Stvorena početna populacija

Primjer



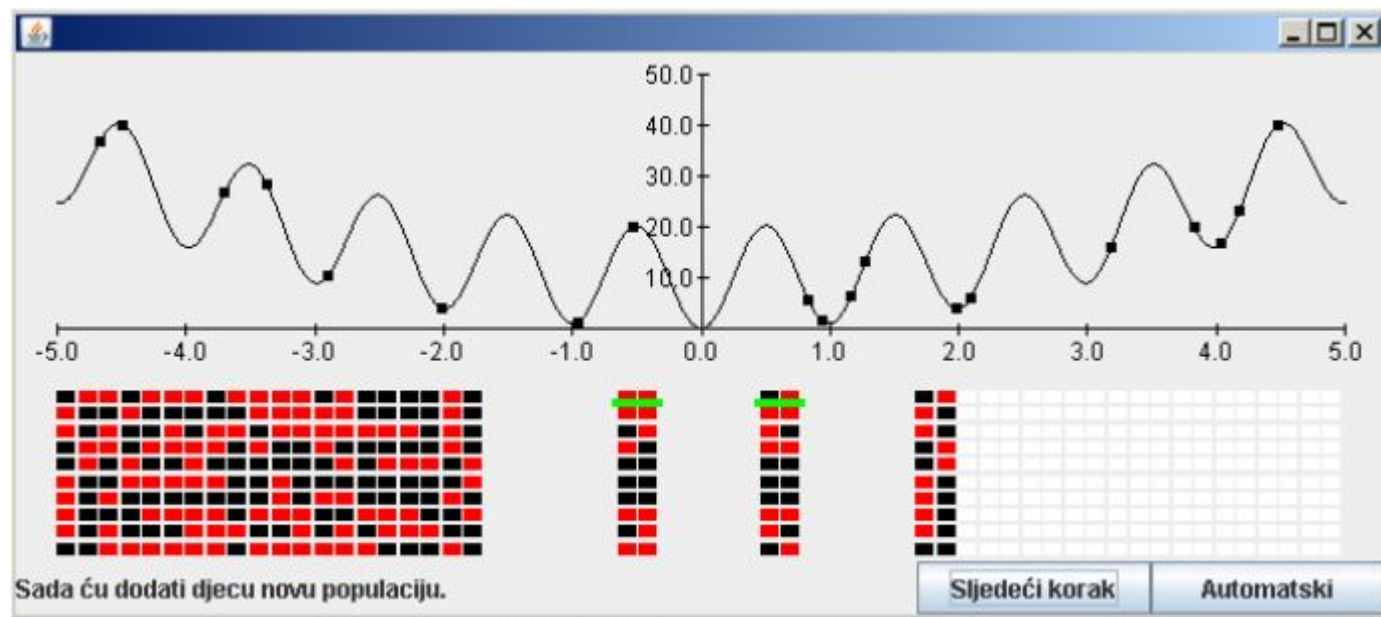
Slika 2-7 Najbolji roditelji dodani u novu populaciju, odabrani roditelji za križanje

Primjer



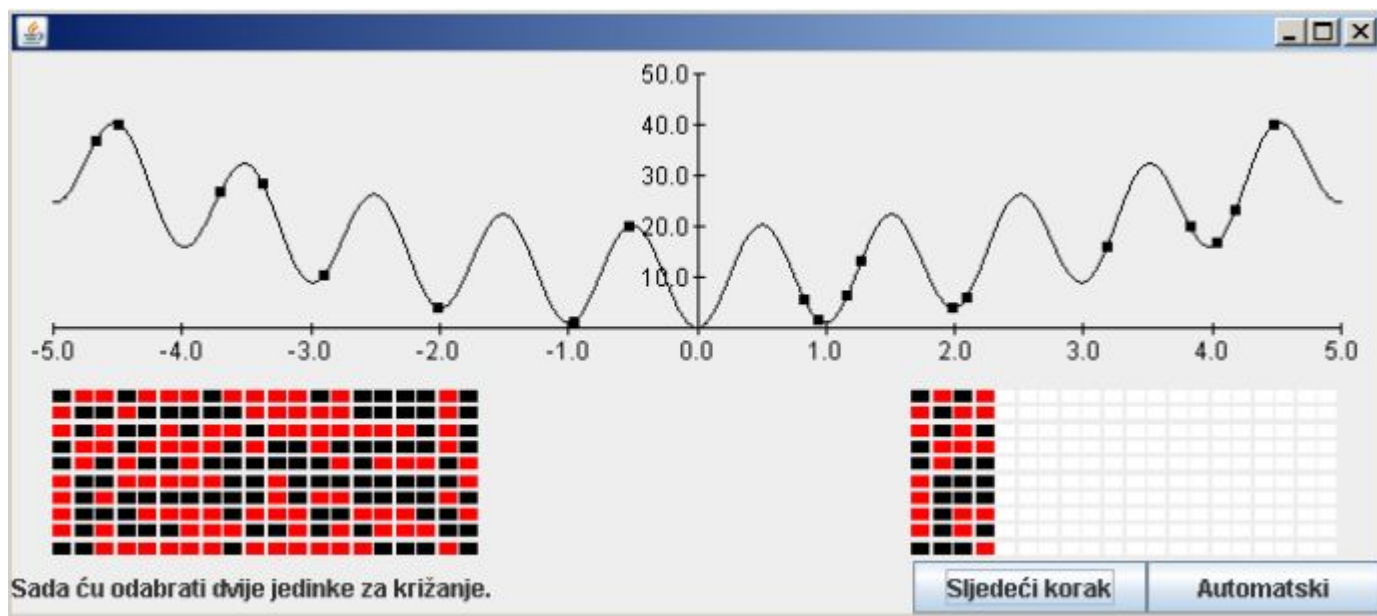
Slika 2-8 Definirana točka prijeloma i obavljeno križanje

Primjer



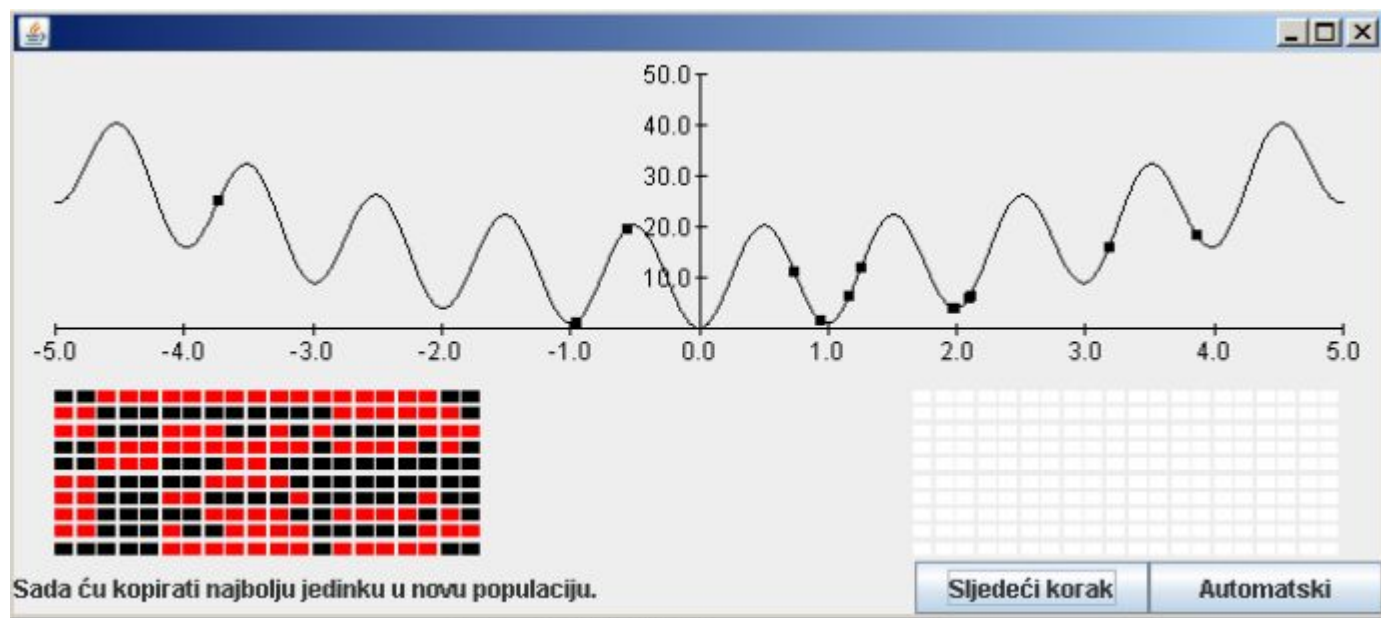
Slika 2-9 Djeca su mutirana

Primjer



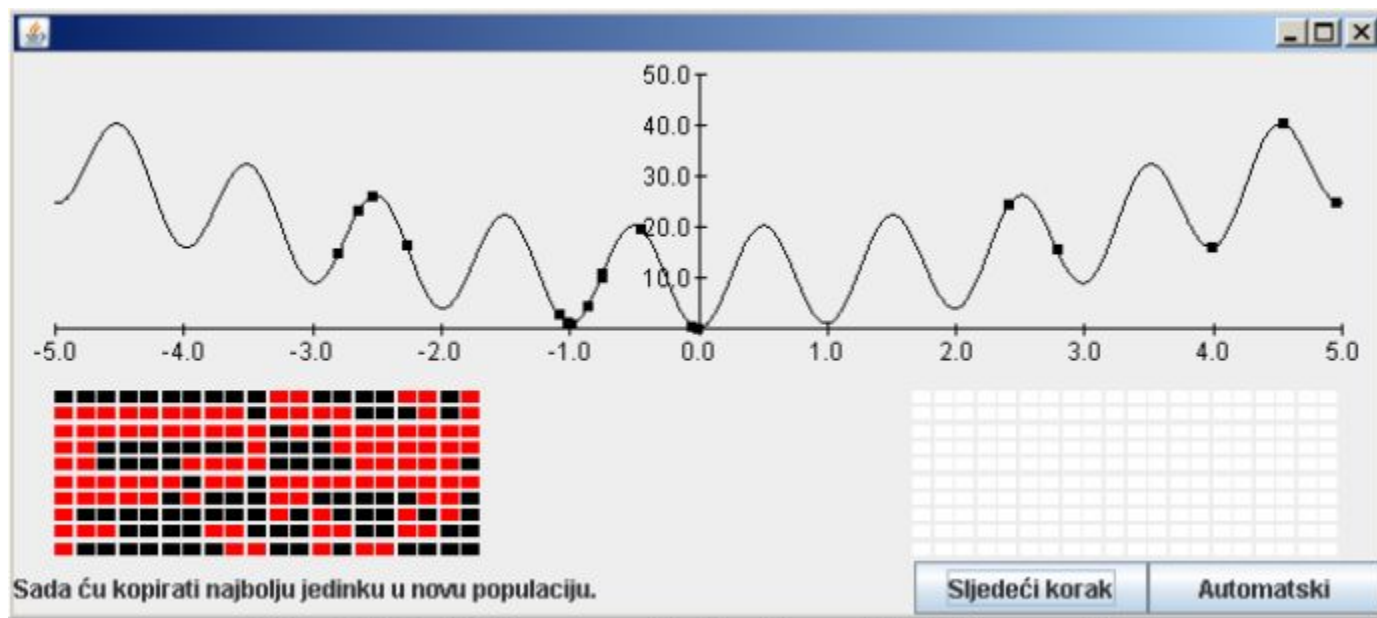
Slika 2-10 Djeca su dodana u novu populaciju

Primjer



Slika 2-11 Početak nove generacije

Primjer



Slika 2-12 I nekoliko generacija kasnije – rješenje je pronađeno

Literatura

- A.E. Eiben, P-E. Raué, Zs. Ruttkay, *Genetic algorithms with multi-parent recombination*,
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.56.5147&rep=rep1&type=pdf>
- Marko Čupić, *Prirodom inspirirani optimizacijski algoritmi*,
https://www.fer.unizg.hr/_download/repository/Cupic2009-PrirodomInspiriraniOptimizacijskiAlgoritmi.pdf
- https://en.wikipedia.org/wiki/Genetic_algorithm#Parallel_implementations
- [https://en.wikipedia.org/wiki/Crossover_\(genetic_algorithm\)](https://en.wikipedia.org/wiki/Crossover_(genetic_algorithm))

Pitanja???