SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Prikaz optimiranja funkcija više varijabli

Petar Čolić

Voditelj: Prof. dr. sc. Domagoj Jakobović

Zagreb, travanj, 2012.

**Sadržaj**

[1. Uvod 1](#_Toc323721803)

[2. Opis korištenja aplikacije 2](#_Toc323721804)

[3. Analiza aplikacije 4](#_Toc323721805)

[4. Zaključak 6](#_Toc323721806)

[5. Literatura 7](#_Toc323721807)

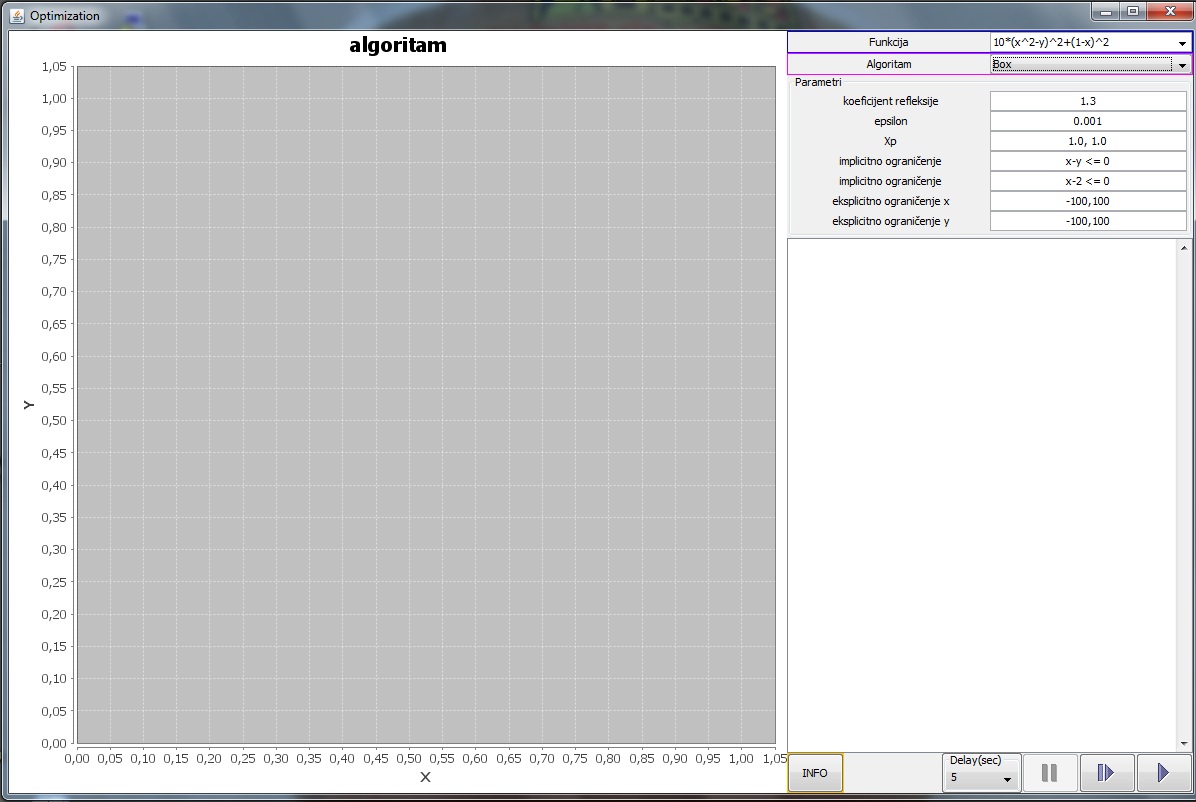
[6. Sažetak 8](#_Toc323721808)

# Uvod

Optimizacija ili matematičko programiranje je grana matematike koja proučava maksimiziranje i minimiziranje funkcija. U radu je opisana vlastita aplikacija koja obavlja optimizaciju. Konkretno, radi se o minimizaciji funkcija dvaju varijabli Hooke-Jeeves postupkom, te postupkom po Boxu. Navedeni postupci provode optimizaciju bez računanja derivacija. Postupak po Boxu optimira uz ograničenja. Dan je kratak opis korištenja, te interna analiza aplikacije.

# Opis korištenja aplikacije

Aplikacija (slika 1) se sastoji od četiri glavna dijela:



Slika 1 - aplikacija

1. Detalji optimizacije

Nudi se mogućnost odabira funkcije, algoritma, te postavki odabranog algoritma.

Može se odabrati jedna od unaprijed određenih funkcija ili upisati vlastitu. Koriste se varijable „x“ (apscisa) i „y“ (ordinata). Trenutno podržani algoritmi su Hooke-Jeevesov postupak i postupak po Boxu. Tablica postavki se dinamički mijenja. Promjenom algoritma, osvježiti će se tablica s postavkama odabranog algoritma.

1. Tekstualni ispis

Ispod postavki nalazi se prostor namijenjen tekstualnom prikazu rada algoritma. Tekst se nikada ne briše, te je moguće u bilo kojem trenutku vidjeti sve korake algoritma.

1. Grafički prikaz rada algoritma

Svaki korak algoritma se prikazuje na grafu. Graf podržava crtanje skupova točaka, te sadrži legendu koja kaže što koji skup predstavlja. Prostor koji se prikazuje će uvijek sadržavati sve točke koje se u tom trenutku prikazuju. Prikazani prostor se također može kontrolirati ručno (desnim klikom na graf).

1. Brzina algoritma

Brzina, odnosno utrošeno vrijeme na svaki korak algoritma, se može kontrolirati na dva načina. Može se odabrati „delay“ tj. pauza između svakog koraka ili koristiti „next step“ način rada u kojem korisnik odlučuje (klikom na gumb „next step“) kada će se prikazati sljedeći korak. Tijekom rada algoritma, može se prebacivati s jednog načina kontrole na drugi, te se algoritam uvijek može pauzirati.

Korištenje aplikacije je jednostavno. Korisnik samo treba odrediti detalje optimizacije, te pokrenuti algoritam u „delay“ (gumb „start“) ili „next step“ (gumb „next step“) načinu rada. Korisniku je dostupan i „INFO“ gumb pomoću kojeg dobiva sve informacije potrebne za pokretanje i kontrolu algoritma, te općenite informacije o aplikaciji. Ako odabrani detalji optimizacije nisu valjani tj. algoritam ne može raditi s njima, korisniku će se dojaviti greška te ga upozoriti koje parametre je krivo postavio. Nakon završetka rada algoritma, moguće je iznova odabrati detalje optimizacije, te pokrenuti novi algoritam.

# Analiza aplikacije

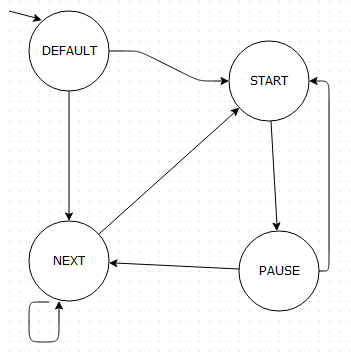
Aplikacija je pisana u programskom jeziku Java, te u programerskom okruženju

Eclipse. Osim standardnih, korištene su i dodatne biblioteke „exp4j” [2] i „JFreeChart” [3].



Slika 2 – dijagram razreda

Slika 2 prikazuje odnos korištenih razreda. Za dodavanje novog algoritma u aplikaciju, potrebno je samo napisati jedan novi razred koji implementira sučelje „Algoritam”, te ga dodati u razred „Main”. Crtanje grafa, te tekstualni ispis se kontrolira zasebno u svakoj implementaciji algoritma.



Slika 3 – stanja algoritma

Slika 3 prikazuje moguća stanja algoritma, te prijelaze između njih. Prijelaz iz „delay” načina u „next” se obavlja u dva koraka: klikom na gumb „pause” te na gumb „next step”.

Za pokretanje aplikacije, potrebno je na računalu imati instaliran programski jezik Java, verziju 7.

# Zaključak

Aplikacija detaljno prikazuje postupke optimizacije. Mogućnost pauziranja i „korak po korak“ način rada pružaju korisniku da brzinu algoritma prilagodi svojim potrebama. Grafički prikaz rada je jasan i jednostavan, te se lako kontrolira. Nadogradnja aplikacije novim algoritmima je također jednostavna. Mislim da će pomoći studentima u razumijevanju i boljem shvaćanju rada algoritma.

# Literatura

[1] skripta „Analiza i projektiranje računalom“ – L. Budin

[2] biblioteka „exp4j“ - <http://projects.congrace.de/exp4j/>

[3] biblioteka „JFreeChart“ - <http://www.jfree.org/jfreechart/>

# Sažetak

Opisana je vlastita aplikacija koja nudi minimizaciju funkcija dvaju varijabli Hooke-Jeeves i Box postupkom. Aplikacija je pisana u programskom jeziku Java, te zahtjeva instaliranu Javu na računalu. Glavni dijelovi aplikacije su: detalji optimizacije, tekstualni ispis, grafički prikaz te brzina algoritma. Korisnik ispunjava detalje optimizacije te može pokrenuti i vidjeti, korak po korak, kako algoritam radi. Za nadogradnju aplikacije potrebno je napraviti novu implementaciju sučelja „Algoritam“.