Berufsakademie Sachsen

Staatliche Studienakademie Leipzig

Untersuchen des Einflusses von Mutation und Populationsgröße auf die Fitness der entstehenden Individuen

Evolutionäre Algorithmen

Autoren: Björn Schapitz

Matrikelnummer: 210812

Paul Jähne

Matrikelnummer:

Seminargruppe: IT2011

Datum: Leipzig, 03.03.2014

Inhalt

[1 Einführung 3](#_Toc381784960)

[1.1 Zielstellung 3](#_Toc381784961)

[1.2 Umsetzung 3](#_Toc381784962)

[1.3 Griewank-Funktion 3](#_Toc381784963)

[1.4 Nullstellenberechnung 3](#_Toc381784964)

[2. Umsetzung 3](#_Toc381784965)

[2.1 Einstellungen und Ablauf des Algorithmus 3](#_Toc381784966)

[3. Ergebnisse 4](#_Toc381784967)

# 1 Einführung

## Zielstellung

Die Aufgabe umfasste das Entwickeln eines evolutionären Algorithmus zum Berechnen einer näherungsweise optimalen Lösung der Berechnung des globalen Minimums der Griewank-Funktion auf der einen und der Nullstellenberechnung einer gegebenen Funktion auf der anderen Seite. Die Güte der Lösung sollte dabei durch Variation der Parameter verbessert werden.

Insbesondere der Einfluss der Populationsgröße und der Einfluss der Mutation bei konstanten übrigen Parametern wurden in Rahmen dieser Aufgabe untersucht.

## Umsetzung

Zur Entwicklung des Programms wurde die Programmiersprache Java in der Version 1.7 verwendet.

## Griewank-Funktion

Die Griewank-Funktion wird häufig zum Testen der Konvergenz von Optimierungsfunktionen verwendet. Sie besitzt ein globales Minimum von 0 bei x = 0. Das zu betrachtende Intervall beträgt

## Nullstellenberechnung

Die Funktion besitzt die Lösung, wobei eine Lösung der Gleichung ist. Das zu betrachtende Intervall beträgt.

# 2. Umsetzung

## 2.1 Einstellungen und Ablauf des Algorithmus

Um die Aufgabenstellung zu lösen, wurden Konzepte der evolutionären Programmierung verwendet. Als Selektionstyp wurde die Kommaselektion gewählt, da ….. Als Rekombination kommt die arithmetische Rekombination zum Einsatz. Die Auswahl der zu rekombinierenden Individuen erfolgt dabei zufällig, eine Mehrfachauswahl desselben Individuums ist möglich. Da Umweltselektion nicht unbedingt notwendig ist, wurde auf sie verzichtet.

Der Algorithmus durchläuft folgende Schritte:

1. Erzeuge Startpopulation *p*
2. Berechne Fitnesswerte der Individuen der Startpopulation
3. Erzeuge neue Population *pnew* durch komma-Selektion der vorhandenen Population
4. Rekombiniere neue Population *pnew*
5. Mutiere *pnew*
6. Berechne Fitnesswerte der Individuen von *pnew*
7. Ersetze *p* durch *pnew*
8. Sortiere *p* nach Fitnesswerten
9. Wiederhole Schritte 3 bis 8 für die Anzahl der Generationen

# 3. Untersuchen des Einflusses der Populationsgröße

Bei der Populationsgröße wurden drei Fälle unterschieden:

1. Konstante Populationsgröße: in jeder Generation bleibt die Populationsgröße erhalten
2. wachsende Population: mit jeder neuen Generation wächst die Populationsgröße an
3. sinkende Populationsgröße: durch Selektion sinkt die Populationsgröße mit jeder Generation

Zu beobachten war, dass die Ergebnisse generell besser wurden, wenn die Startpopulation größer war.

# 4. Untersuchen des Einflusses der Mutation

* Mutation: können auch mehrmals mutiert werden
* Ergebnisse über 100 Läufe gemittelt, pro Lauf hundert Generationen
* Fitness vom Besten verbessert sich nicht mehr nach max. 40 Generationen 🡪
* Algorithmus: arithm. Reko., keine Umwelt, Eltern = komma, 🡪 sehr deterministisch 🡪 mittelt sich aus…
* Größere Population = besseres Ergebnis
* Mutation: verschiedene Modi:
  + Linear degression: schlecht
  + Konstant: schlecht
  + Exponential: besser als linear
  + Exp.dec. insgesamt schlechter
* Allele: mehr = schlechter, warum?
* Griewank besser Null