WACHTER Johannes und ZANGERLE Michael

Modelierung, Simulation und Optimierung komplexer und lernfähiger Systeme

Aufgabe 2-6

University of Applied Sciences

Department of Computer Science
Prof. Dr. rer. nat. habil. Hans-Georg Beyer

Dornbirn, 2014

Abstract

In diesem Report werden die Aufgaben 2-6 von den Aufgaben in der Lehrveranstaltung S2 Einführung in Modellierung, Analyse und Optimierung komplexer Systeme behandelt.

Inhaltsverzeichnis

2 (1+1) - ES

2.1 Implementierung: Minimierung

Der (1+1) Algorithmus wurde laut der Folie 61 implementiert. Im Anhang kann der Code begutachtet werden. Der Vergleich zwischen der Abbildung 2.1 und der Abbildung auf Folie 71 ergab, das wir die Aufgabe gelöst haben. Der Eingabevektor wird minimiert. Wenn nur eine kleine Grenze der Generationen übergeben wird, kann der Algorithmus nicht den Optimalen Wert finden.

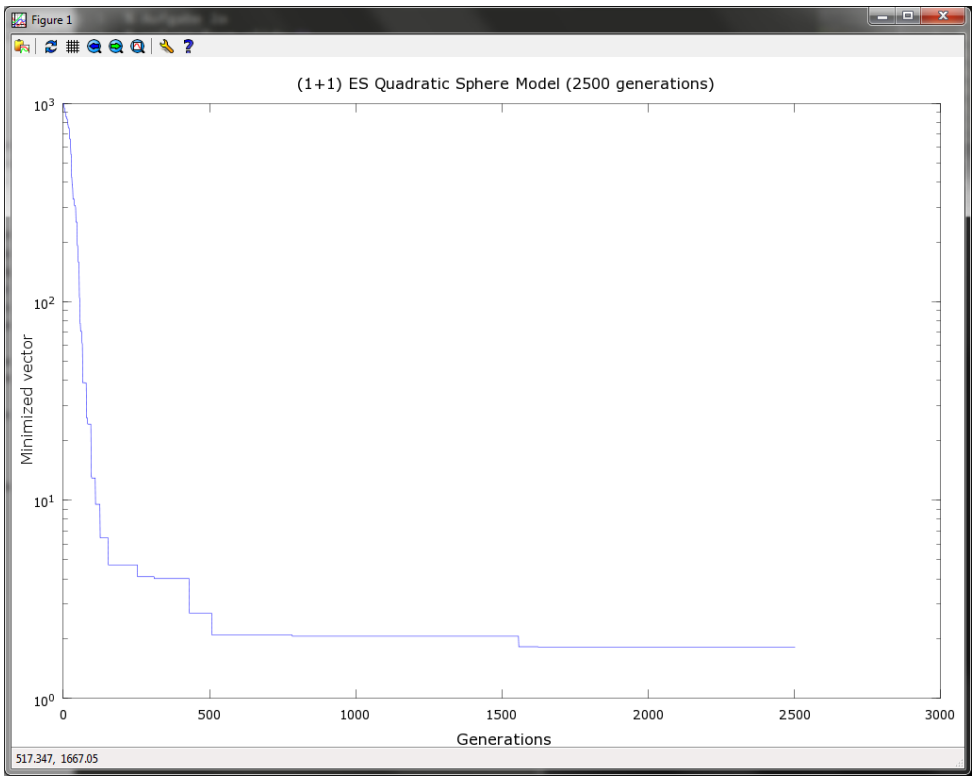

Abbildung 2.1: (1+1)-ES Minimierung

2.2 Implementierung: Maximierung

Im Gegensatz zur vorherigen Implementierung, wurde die Bedienung zur Übernahme des besseren Wertes geändert, dadurch wird nicht der kleinere weiter verwendet sondern der größere. Die Abbruchbedienung wurde so abgeändert, das der Fitnesswert dem übergebenen N entspricht. Zusätzlich wird laut Aufgabe eine andere Fitness-Funktion verwendet. Für die Fitness-Funktion wurde eine Modulo 2 Operation verwendet. In Abbildung **??** erkennt man, das im Gegensatz zu Teilaufgabe 2.1 die Kurve eine Maximierung zeigt.

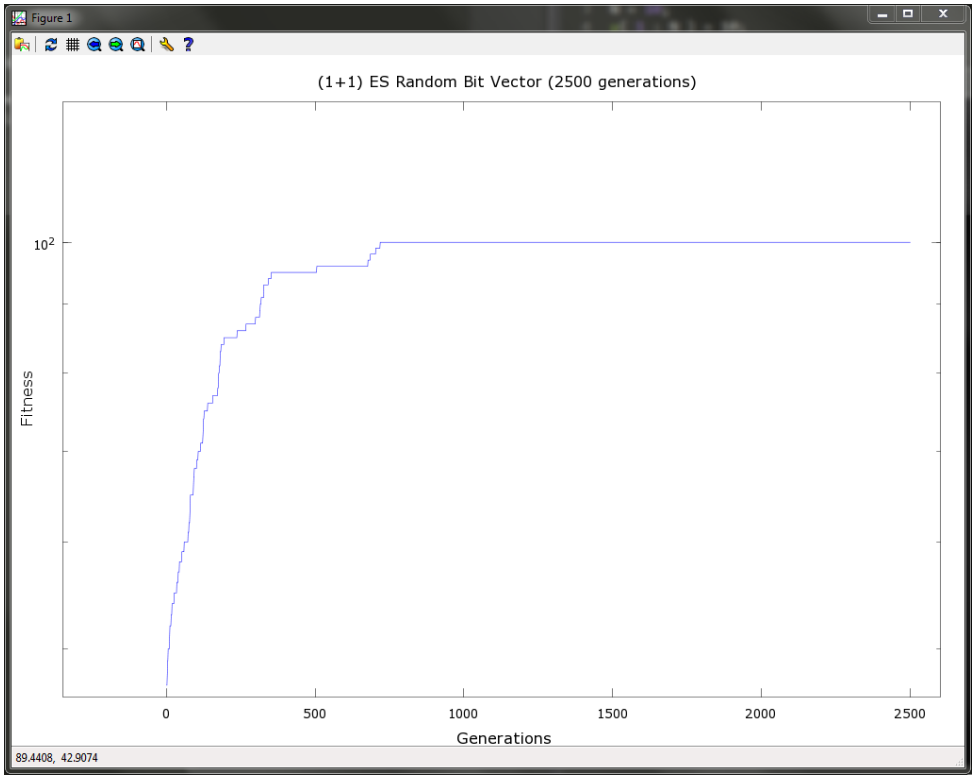

Abbildung 2.2: (1+1)-ES Maximierung

Die Laufzeitkomplexität des Algorithmus im Intervall von $N = 10...300$ überprüft. Dazu wurden die Ergebnisse der Laufzeit über 10 Durchläufe gemittelt. Aufgrund der langen Laufzeit haben wir uns dafür entschieden den Bereich in fünfziger Schritten abzustasten, dadurch ist in der Abbildung 2.3 die Achsen Beschriftung mit $1-6$ angegeben. Zusätzlich kann aus der Abbildung herausgelesen werden, dass die Laufzeit nahezu linear ist.

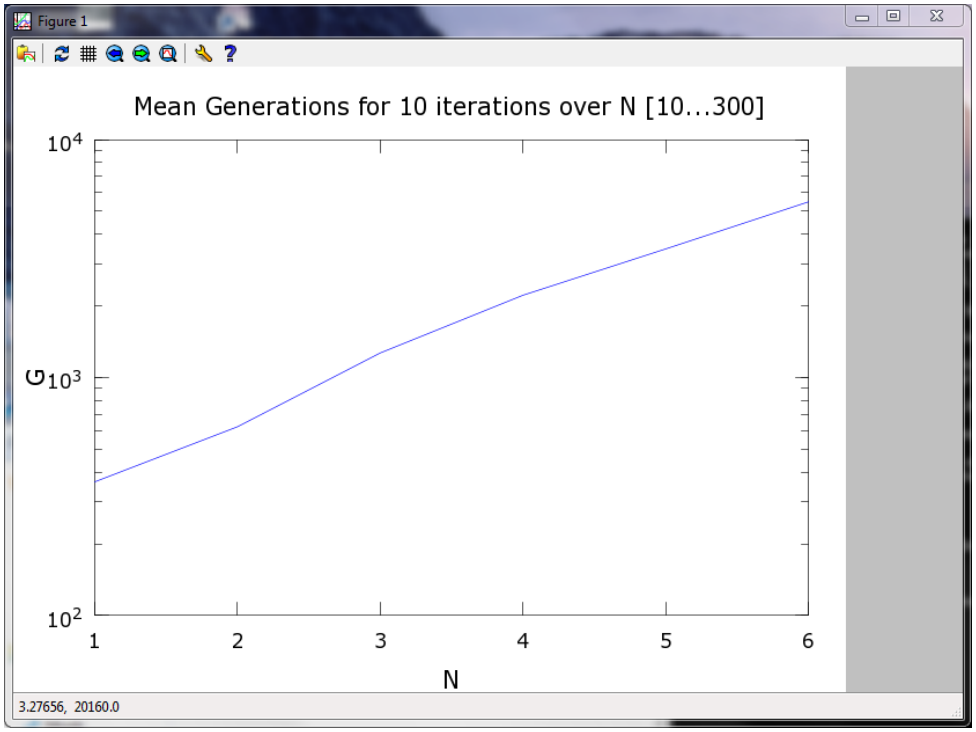

Abbildung 2.3: (1+1)-ES Laufzeit 10 Durchgänge Mittelung

Deterministischer Algorithmus

Der einfachste Algorithmus wäre es all einzelnen Bits der Reihe nach kippen. Nach jedem dieser Vorgänge wird der Fitnesswert berechnet, falls dieser besser ist, wird dieser als neuer Wert verwendet. Dieser Algorithmus besitzt eine Wort case Laufzeit von $O(N)$.

3 (1+1) - ES: 1/5 Regel

Laut Aufgabe wurde der Algorithmus mit der 1/5 Regel erweitert. Dazu wurden die Änderungen aus Folie 76. Auch hier kann in erkannt werden, das die Abbildung 3.1 ähnlich wie der in Folie 77 aussieht. Wobei Kurve einen flacheren Verlauf aufweise. Die Ergebnisse sind $sigma = 1.1808e^{-}7$ und $Generationen = 7501$.

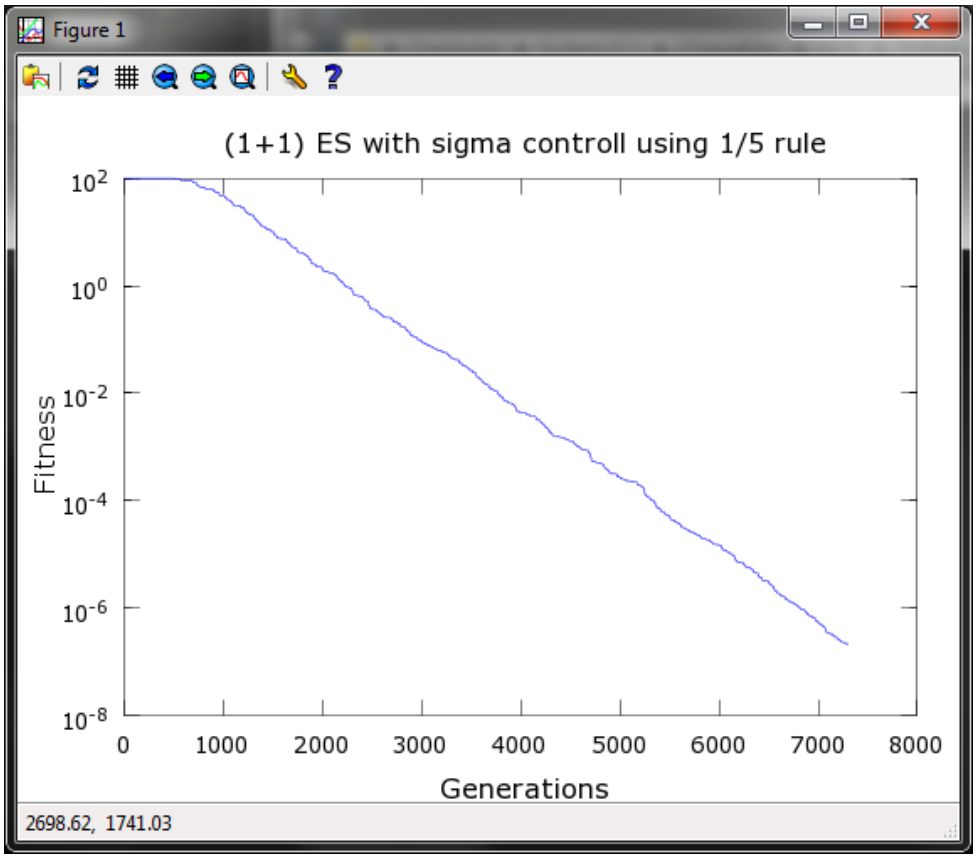

Abbildung 3.1: (1+1)-ES

4 Abwandlung Aufgabe 3

Die Implementierung aus der Aufgabe 3 wurde so abgeändert, dass die Fitness-Funktion als Übergabeparameter übergeben werden kann. In Abbildung 4.1 ist die Auswertung mit der Parabolic-Ridge und in Abbildung 4.3 mit einem Faktor $d = 1$ zusehen. Der Faktor wurde in den Abbildungen4.2 und 4.4 auf $d = 10$ erhöht.
Aus den Abbildungen von Parabolic-Ridge ist ersichtlich, das nach 1000*Generationen* bei $d = 10$ der Fitnesswert nicht weiter sinkt, jedoch bei $d = 1$ weiter sinkt.
Bei Sharp-Ridge zeigen die beiden Abbildungen einen ziemlich unterschiedlichen Verlauf.

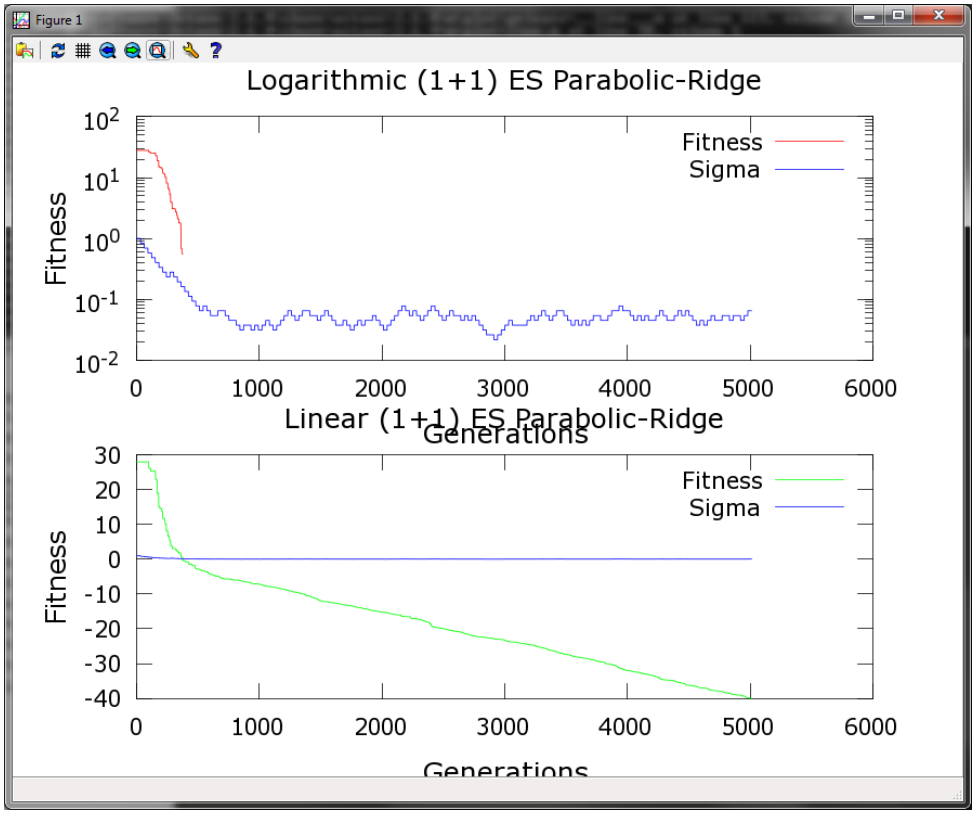

Abbildung 4.1: Parabolic-Ridge d=1

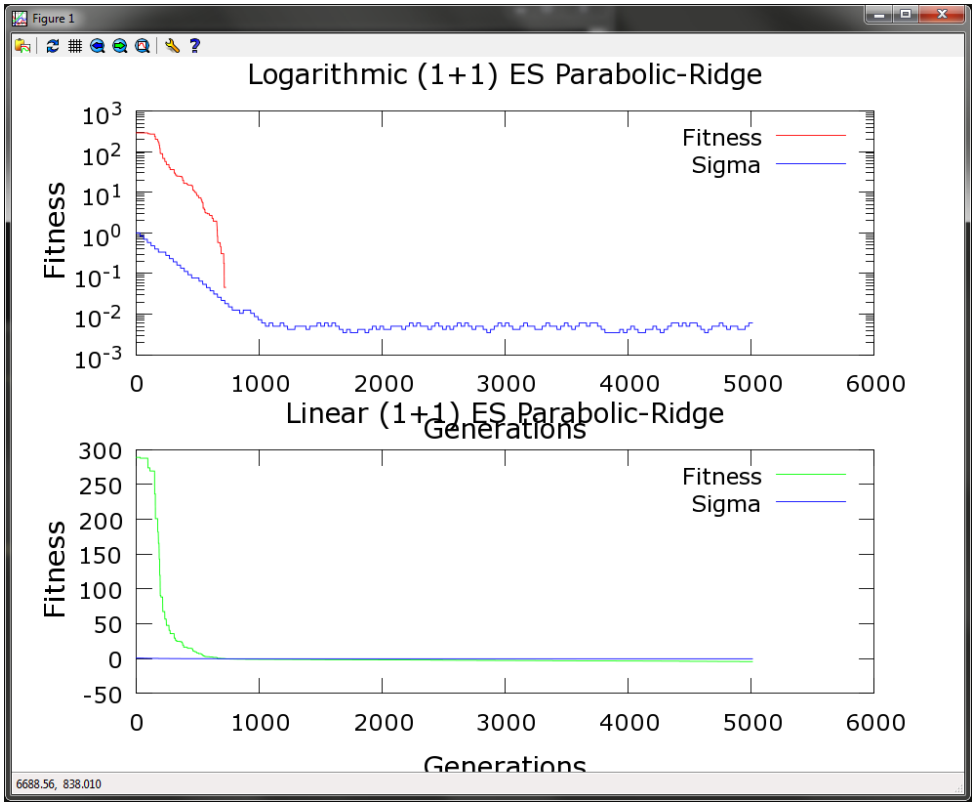

Abbildung 4.2: Parabolic-Ridge d=10

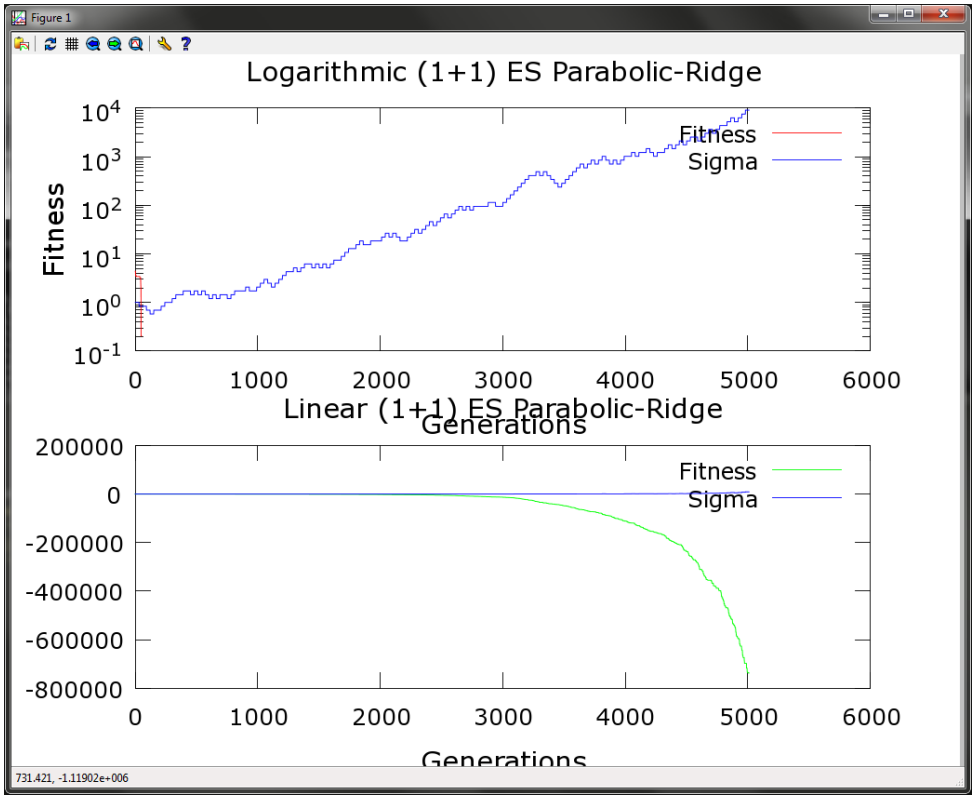

Abbildung 4.3: Sharp-Ridge d=1

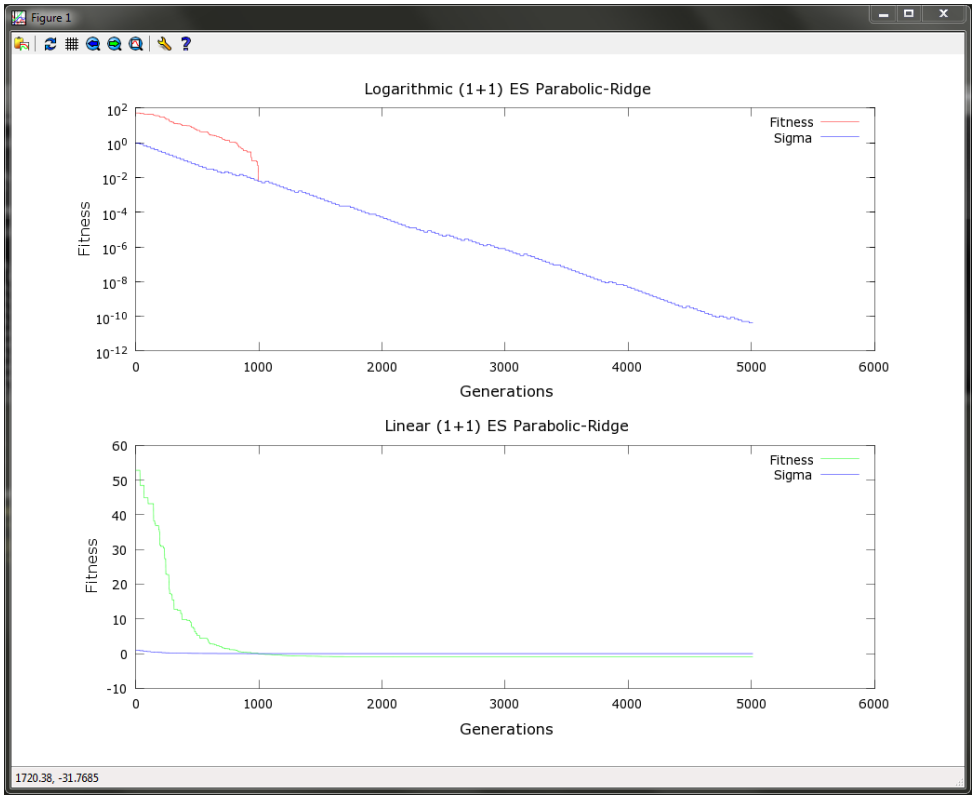

Abbildung 4.4: Sharp-Ridge d=10

5 (1, λ) - σSA - ES

Der (1, λ) - σ SA - ES Algorithmus wurde laut Folie 92 implementiert. Die werte wurden aus der Folie 95 entnommen. Im Vergleich zwischen der Abbildung 5.1 und der Abbildung auf Folie 95 kann erkannt werden, dass der Algorithmus richtig implementiert wurde.

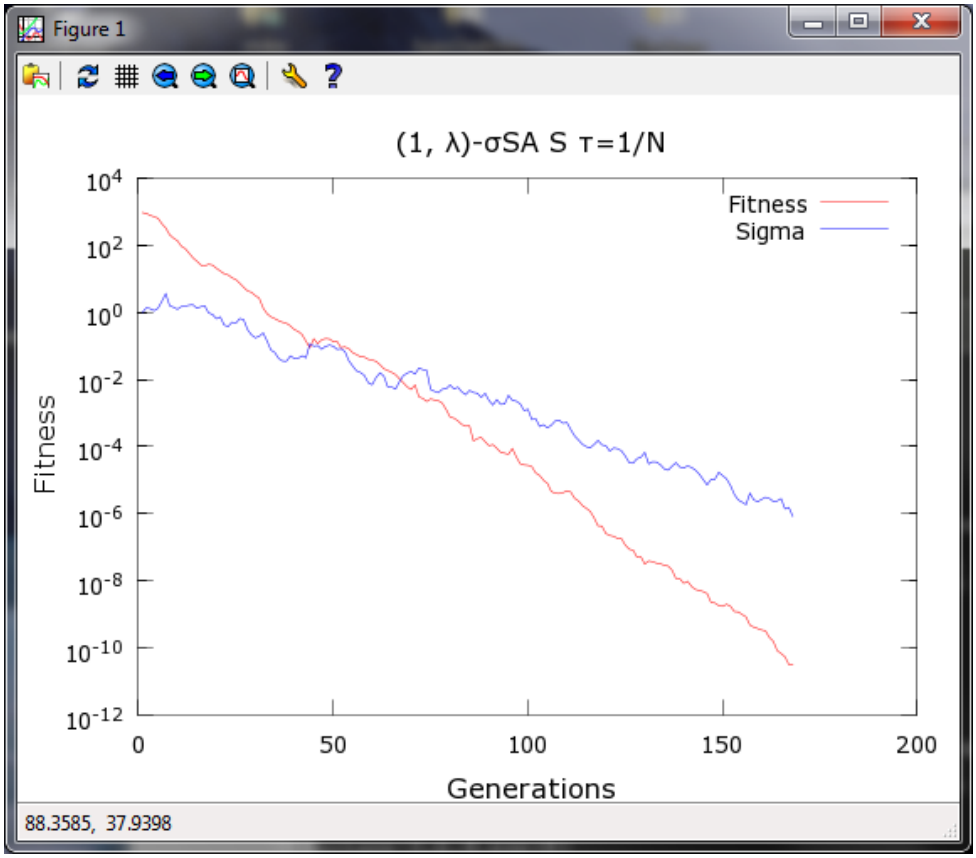

Abbildung 5.1: (1, λ) - σSA - ES: Kugelmodel

6 $(\mu$ / μ_1, $\lambda)$ - σSA - ES

Die Implementierung aus der Aufgabe 5 wurde so abgewandelt, dass der σSA - ES Algorithmus als Funktion aufgerufen werden kann und der μ / μ_1 aus Parameter übergeben werden kann. Der $3/3r$ - ES wurde am Kugelmodell mit $N = 30$ getestet. Die Ergebnisse können in der Abbildung **??** begutachtet werden.

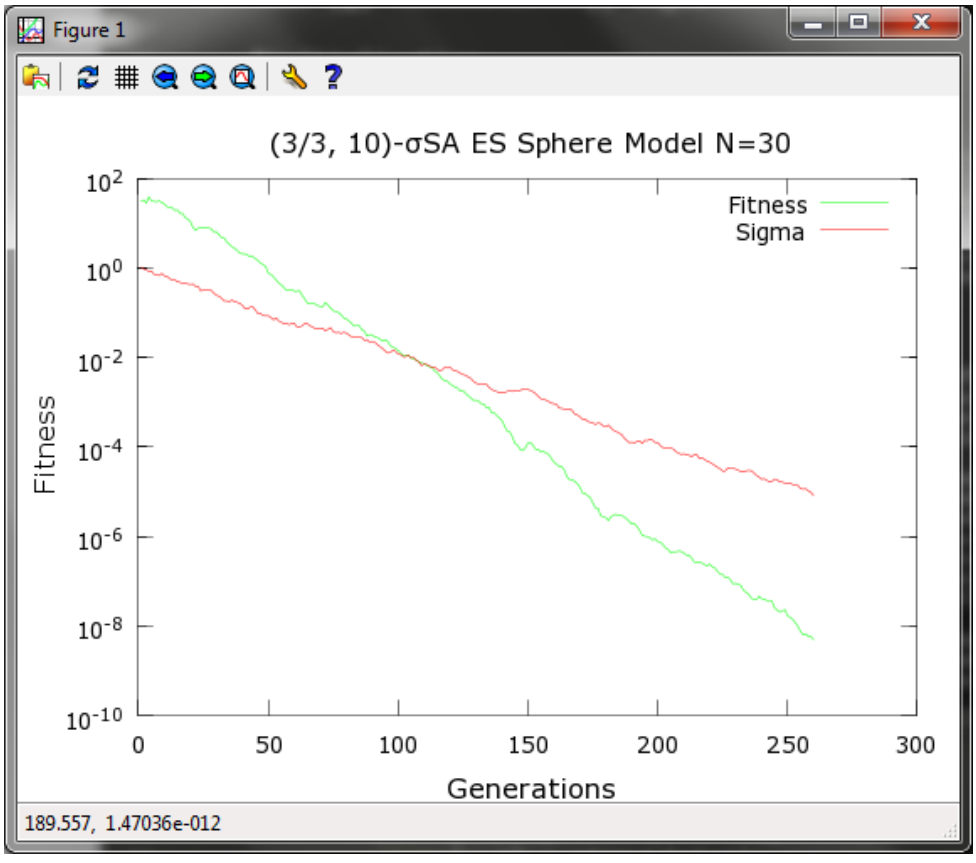

Abbildung 6.1: $\mu = 3$ und N=30

Beim Test mit $N = 100$, befinden sich die Werte im Zulässigen Bereich ($\mu = 3$ $y = 3.7111e-8$ mit $785 Generationen$ und $\mu = 1$ $y = 2.2245e-7$ mit $1314 Generationen$).

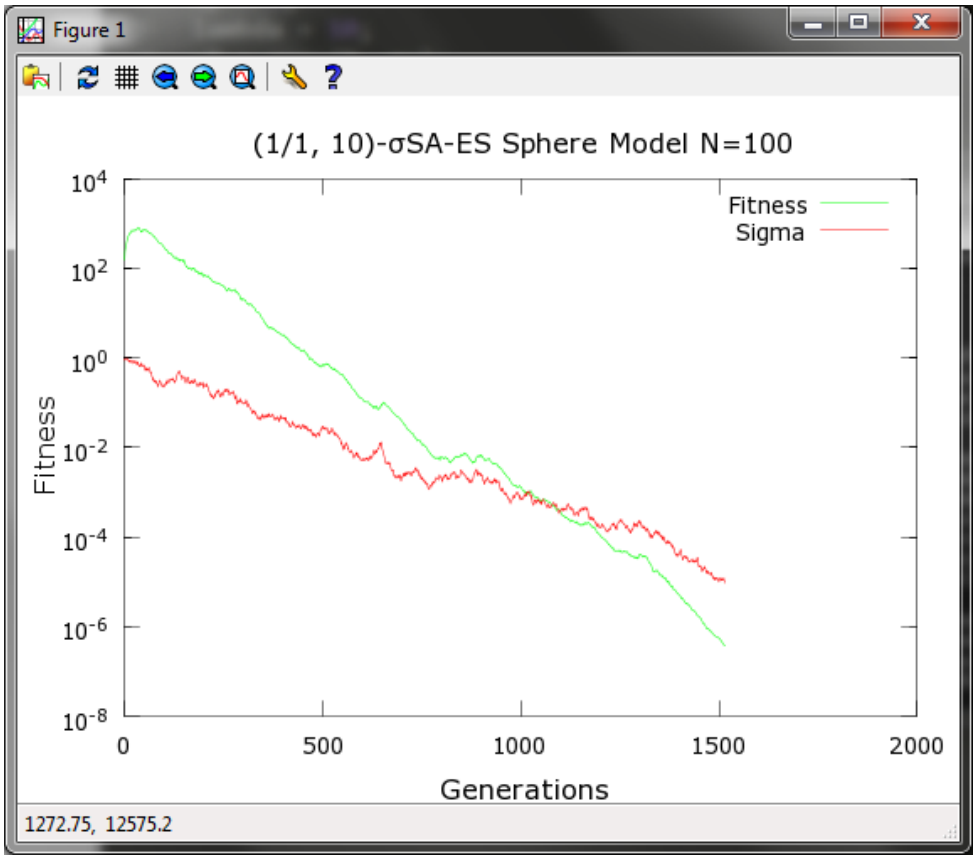

Abbildung 6.2: $\mu = 1$ und N=100

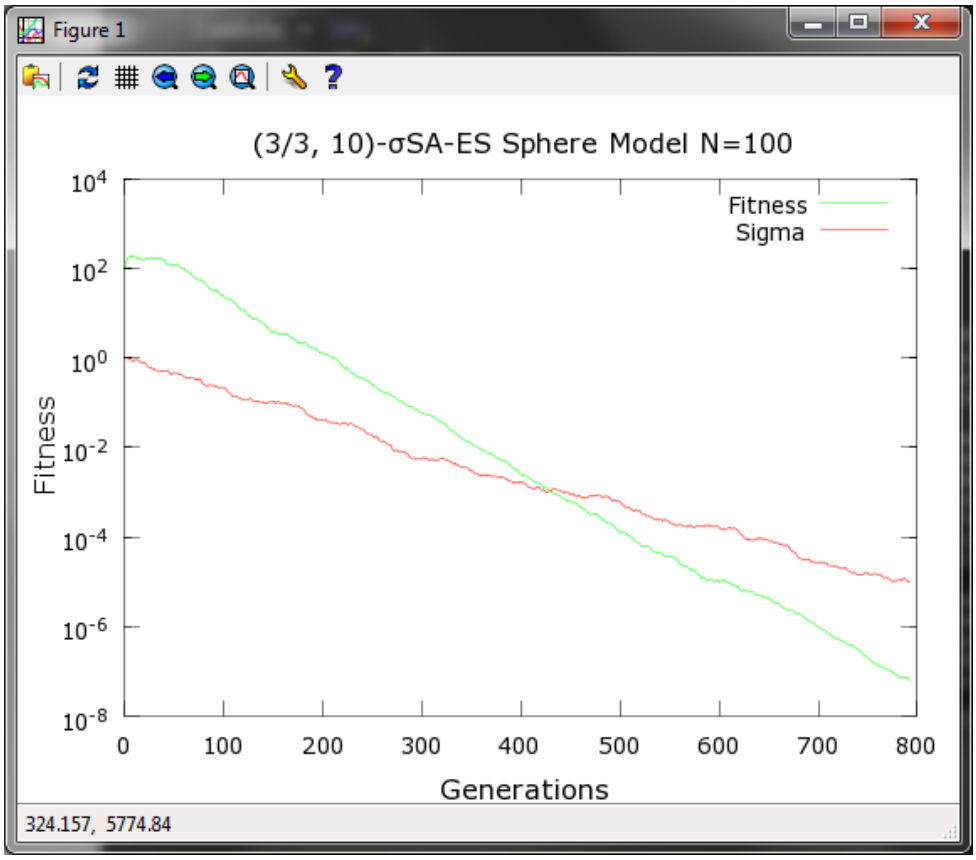

Abbildung 6.3: $\mu = 3$ und N=100

7 Zusammenfassung

Im Allgemeinen konnten wir die Ergebnisse aus den Skripten bestätigen. In allen Aufgaben haben wir ähnliche Ergebnisse erreicht.

Bei der Aufgabe 4 konnten die Ergebnisse mit dem Faktor d stark beeinflusst werden.

Aufgrund der Zeitintensiven Berechnungen musste man bei einigen der Aufgaben erheblich mehr Zeit einplanen als gedacht.