

WACHTER Johannes und ZANGERLE Michael

# Modelierung, Simulation und Optimierung komplexer und lernfähiger Systeme

Aufgabe 7-12

University of Applied Sciences: Vorarlberg

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

Dornbirn, 2014



# Abstract

In diesem Report werden die Aufgaben 7-12 von den Aufgaben in der Lehrveranstaltung S2 Einführung in Modellierung, Analyse und Optimierung komplexer Systeme behandelt.



# Inhaltsverzeichnis

<b>Abstract</b>	<b>iii</b>
<b>7</b> $(\mu/\mu_I, 10) - \sigma SA - ES$ am Kugelmodell	<b>1</b>
<b>8</b> $(3/3_I, 10) - \sigma SA - ES$ am Kugelmodell	<b>3</b>
<b>9</b> Performance Vergleich	<b>5</b>
<b>10</b> $(\mu/\mu_I, 20) - \sigma SA - ES$ am verrauschten Kugelmodell	<b>7</b>
<b>11</b> $(\mu/\mu_I, \lambda) - \sigma SA - ES$ an der Rastrigin-Funktion	<b>9</b>
<b>12</b> CMSA-ES für Linsenoptimierung	<b>11</b>
<b>13</b> Zusammenfassung	<b>13</b>



## 7 $(\mu/\mu_I, 10) - \sigma SA - ES$ am Kugelmodell

In diesem Beispiel wird der Einfluss des  $\mu$  in dem Algorithmus betrachtet. Hierfür wird die F-Dynamik in der Abbildung 7.1 dargestellt. Als Testfunktion wird das Kugelmodell mit  $N = 100$  und  $\mu = 1 \dots 10$ .

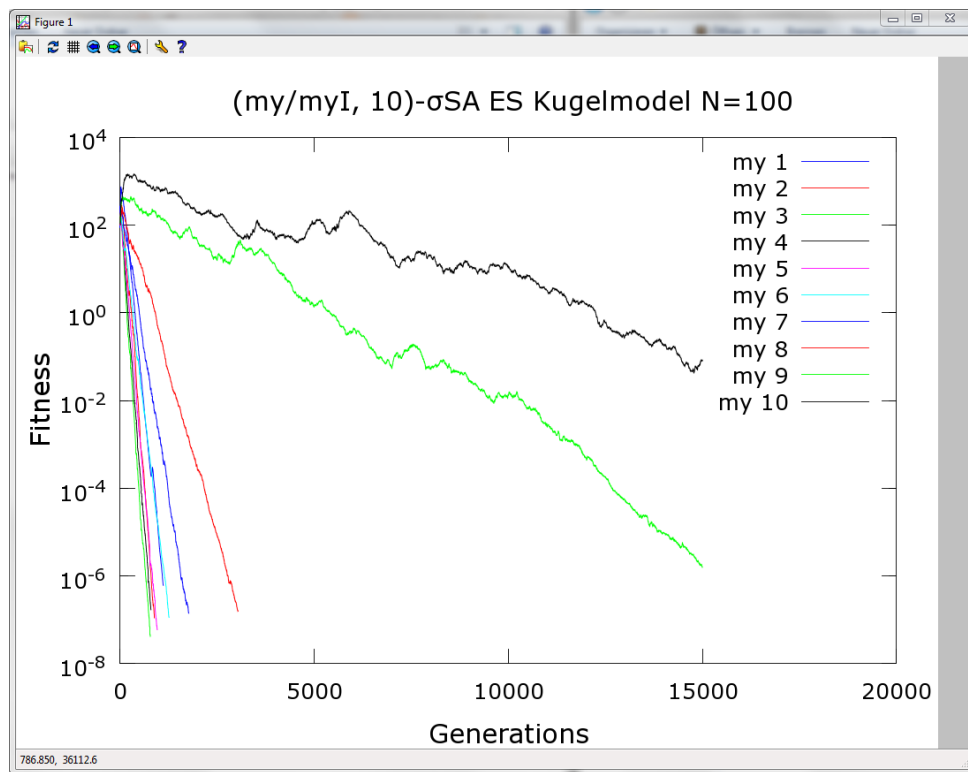


Abbildung 7.1: F-Dynamik  $(\mu/\mu_I, 10) - \sigma SA - ES$  am Kugelmodell

Die Abbildung zeigt, dass bei  $\mu = 10$  das Verfahren nicht mehr steil gegen 0 sinkt. Durch die Begrenzung nach der Iterationsgrenze von 15.000 abgebrochen.





## 8 $(3/3_I, 10) - \sigma SA - ES$ am Kugelmodell

Diese Aufgabe betrachtet den Einfluss von  $\tau$  auf den  $(3/3_I, 10) - \sigma SA - ES$  Algorithmus am Kugelmodell mit  $N = 100$ ,  $\sigma = 0.1$  und  $y_{init} = (1, \dots, 1)^T$ . Auch in dieser Aufgabe wird die F-Dynamik in der Abbildung 8.1 dargestellt.

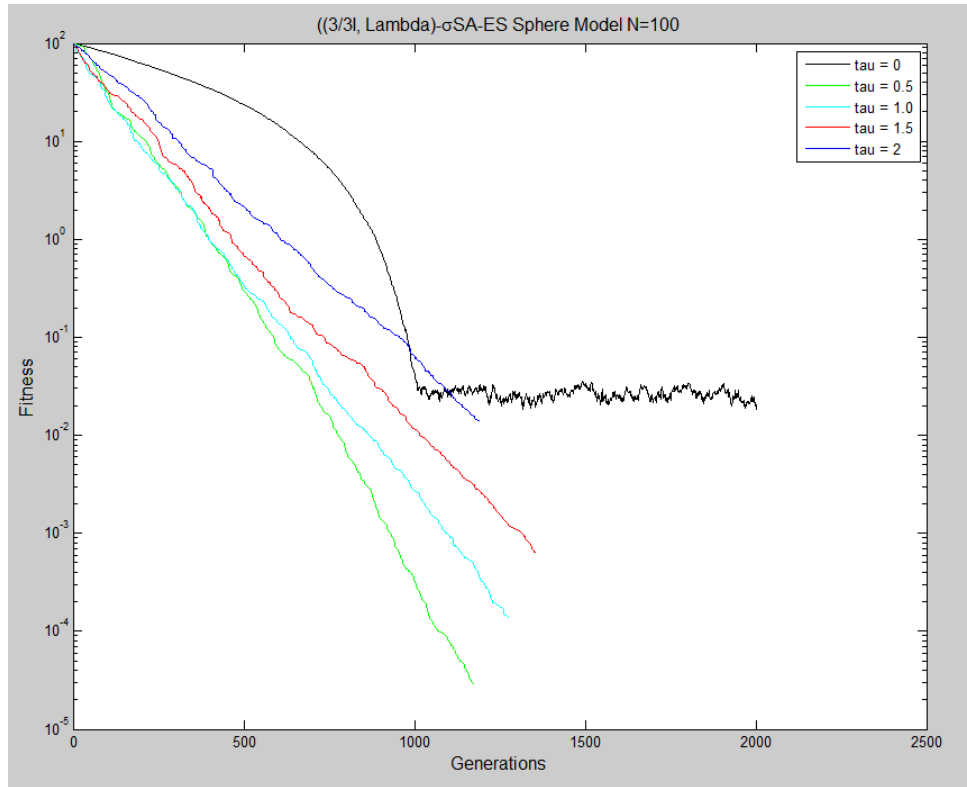


Abbildung 8.1: F-Dynamik  $(3/3_I, 10) - \sigma SA - ES$  am Kugelmodell

Bei  $\tau = 0$  kann beobachtet werden, dass aufgrund der fehlenden  $\sigma - Adaption$ , der Wert konvergiert.

Auch die Abbildung 8.2 zeigt, dass bei  $\tau = 0$  der Wert stagniert.

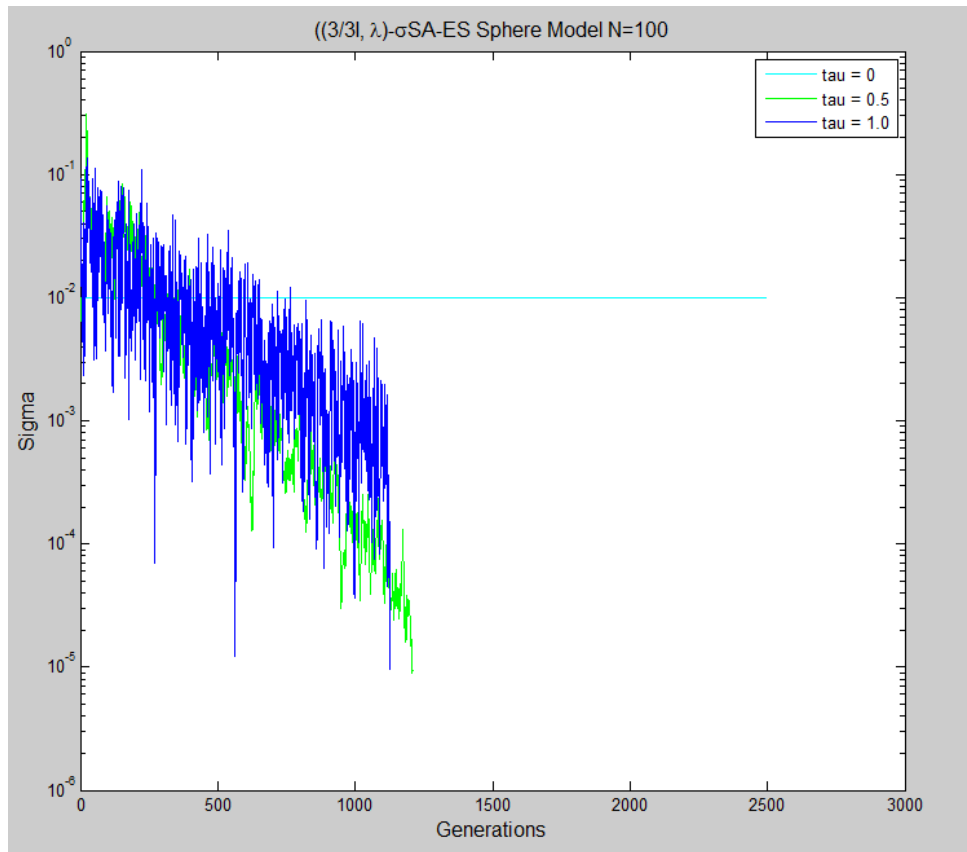


Abbildung 8.2:  $\sigma$ -Dynamik  $(3/3_I, 10) - \sigma SA - ES$  am Kugelmodell

## 9 Performance Vergleich

Verglichen wird der Algorithmus  $(1 + 1) - ES$  mit  $1/5$  Regel mit dem Algorithmus  $(3/3_I, 10) - \sigma SA - ES$ . Als Testfunktion wird das Kugelmodell mit  $N = 100$  verwendet. Der Vergleich erfolgt auf Basis der Generationen und aufgrund der Anzahl an Funktionsauswertungen.

	$(1 + 1) - ES$	$(3/3_I, 10) - \sigma SA - ES$
Generationen	7301	986
Funktionsauswertungen	7301	10846

Tabelle 9.1: Ergebnisse der Ausführung

Aus diesen Ergebnissen kann ermittelt werden, dass die Dauer der Funktionsauswertung für die Auswahl des Algorithmus ausschlaggebend ist. Ist diese Dauer klein, ist der  $(3/3_I, 10) - \sigma SA - ES$  Algorithmus insgesamt schneller, da weniger generationen benötigt werden. Ist die Dauer der Funktionsauswertung hoch, ist der  $(1 + 1) - ES$  Algorithmus aufgrund der kleineren Zahl der Funktionsauswertungen schneller.



## 10 $(\mu/\mu_I, 20) - \sigma SA - ES$ am verrauschen Kugelmodell

Diese Aufgabe beschäftigt sich mit der Performance des Algorithmus  $(\mu/\mu_I, 20) - \sigma SA - ES$  am verrauschten Kugelmodell. Hierfür wird  $F(y) = ||y||^2 + \mathcal{N}(0, 1)$ ,  $N = 100$ ,  $y_{init} = (10, \dots, 10)^T$  und  $\sigma_{init} = 1$ . Der Algorithmus wurde über 3000 Generationen durchgeführt. In der Abbildung 10.1 wird die Dynamik des Restzielabstandes  $||y^{(g)}||$  des elterlichen Restkombinanten dargestellt. In dieser Abbildung ist zu sehen, dass durch die Verrauschung das Optimum nicht erreicht werden kann.

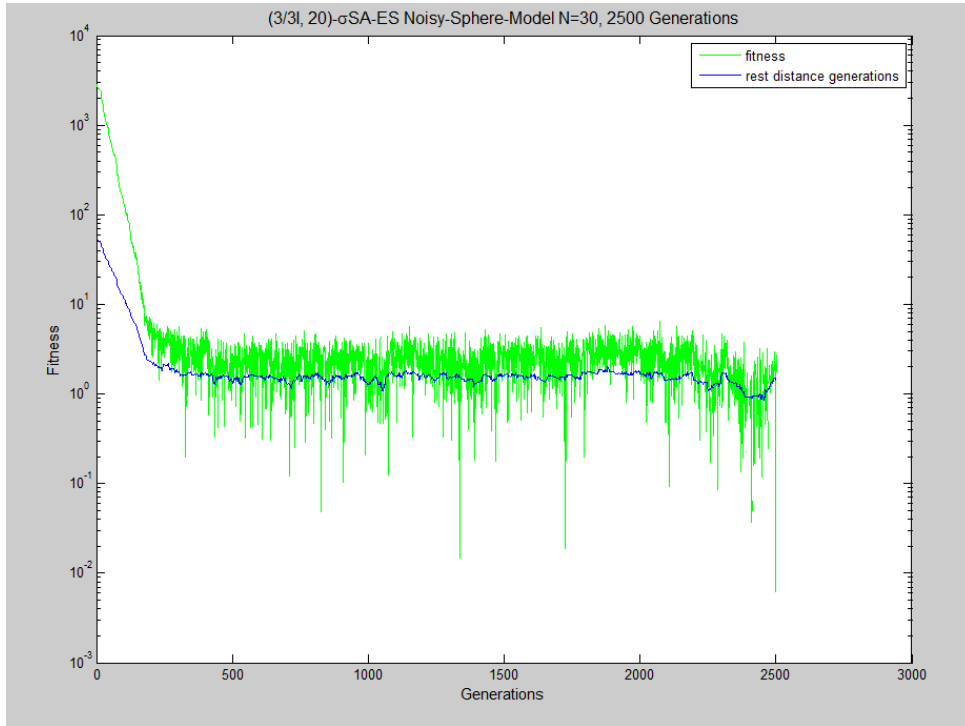


Abbildung 10.1: Dynamik des Restzielabstandes  $||y^{(g)}||$

Wird das  $\mu$  erhöht, verringert sich der Restzielabstand, daher liefert der Algorithmus bessere Ergebnisse. In der Abbildung 10.2 kann ab einem  $\mu = 5$  beobachtet werden, dass sich keine sichtliche Verbesserung mehr einstellt.

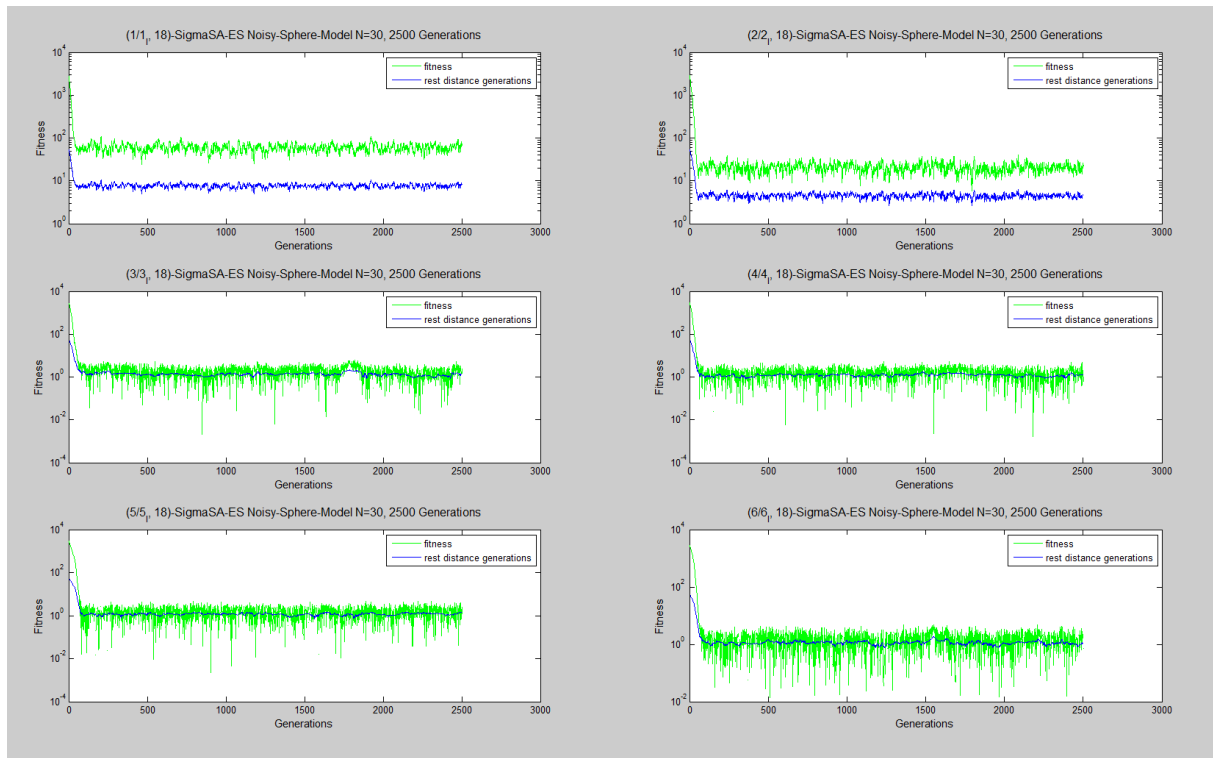


Abbildung 10.2: Optimales  $\mu$

## 11 $(\mu/\mu_I, \lambda) - \sigma SA - ES$ an der Rastrigin-Funktion

In dieser Aufgabe wird das Verhalten des  $(\mu/\mu_I, \lambda) - \sigma SA - ES$  Algorithmus an der Rastrigin-Funktion (Abbildung 11.1) untersucht.

$$F(y) = \sum_{i=1}^N (a - a * \cos(2\pi y_i) + y_i^2), a = 2$$

Abbildung 11.1: Rastrigin Funktion

Als Initialwerte wurden  $N = 30$ ,  $\sigma_{init} = 1$  und  $y_{init} = 10, \dots, 10^T$  verwendet.

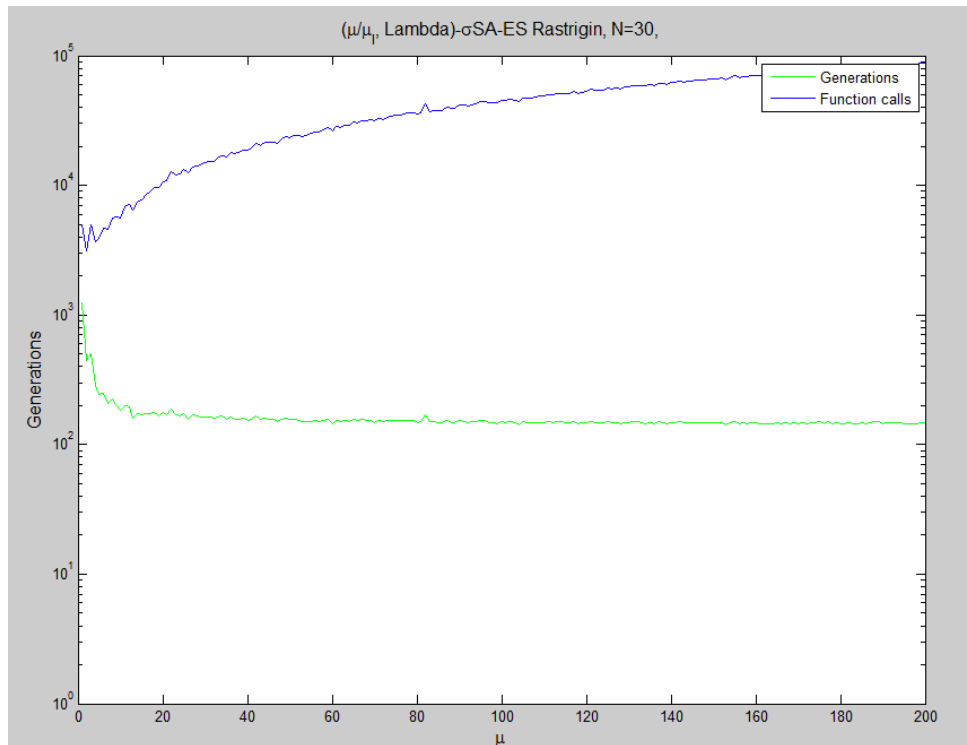


Abbildung 11.2: Testfunktion RASTRIGIN

In der Abbildung 11.2 kann beobachtet werden, dass je größer  $\mu$  wird desto weniger Generationen werden benötigt, um das Optimum zu finden. Ab etwa  $\mu = 20$  bringt eine erhöhung kaum verbesserungen mit sich. Gegenläufig kann beobachtet werden, dass die Anzahl der Funktionsaufrufe mit erhöhtem  $\mu$  steigt. Daher ist auch bei diesem Algorithmus das  $\mu$ , je nach geschwindigkeit des Funktionsaufrufes, mit vorsicht zu wählen.



## 12 CMSA-ES für Linsenoptimierung

Diese Aufgabe soll zeigen, dass die Kovarianzmatrix-Adaption wichtig für die performante Optimierung des ES ist. Diese Annahme bestätigt die Abbildung 12.1. Diese Abbildung zeigt die Dynamiken des CMSA-ES mit dem  $(\mu/\mu_I, \lambda) - \sigma SA - ES$  am Linsenproblem.

Als initialwerte wurde  $N = 30$ ,  $\sigma_{init} = 1$  und  $y_{init} = 10, \dots, 10^T$  verwendet.

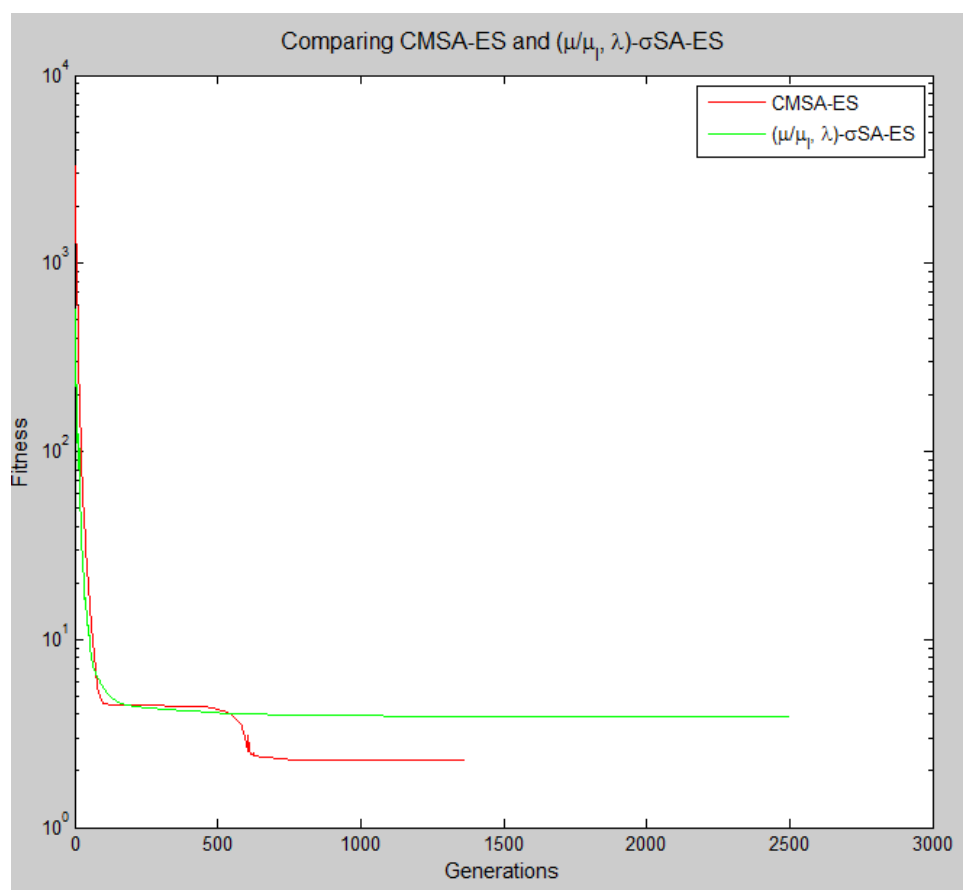


Abbildung 12.1: Dynamiken des CMSA-ES mit dem  $(\mu/\mu_I, \lambda) - \sigma SA - ES$  am Linsenproblem

In der obigen Abbildung kann beobachtet werden, dass sich beide Algorithmen ähnlich verhalten. Der Knick im CMSA-ES Algorithmus, ergibt sich durch die Anpassung der Kovarianzmatrix.

## 13 Zusammenfassung

Diese Aufgaben verdeutlichen sehr stark, dass die Wahl der Parameter für die Genauigkeit wichtig ist. Zudem kann beobachtet werden, wie diese Wahl auch die Performance beeinflusst. Diese beiden Eigenschaften verhalten sich gegenläufig zueinander.

Aufgrund der zeitweise aufwendigen Aufgaben, haben wir einen Kollegen gebeten, die Aufgaben in Matlab auszuführen. Matlab ist laut Internet aufgrund des Just-In-Time-Compilers um einiges schneller als Octave.