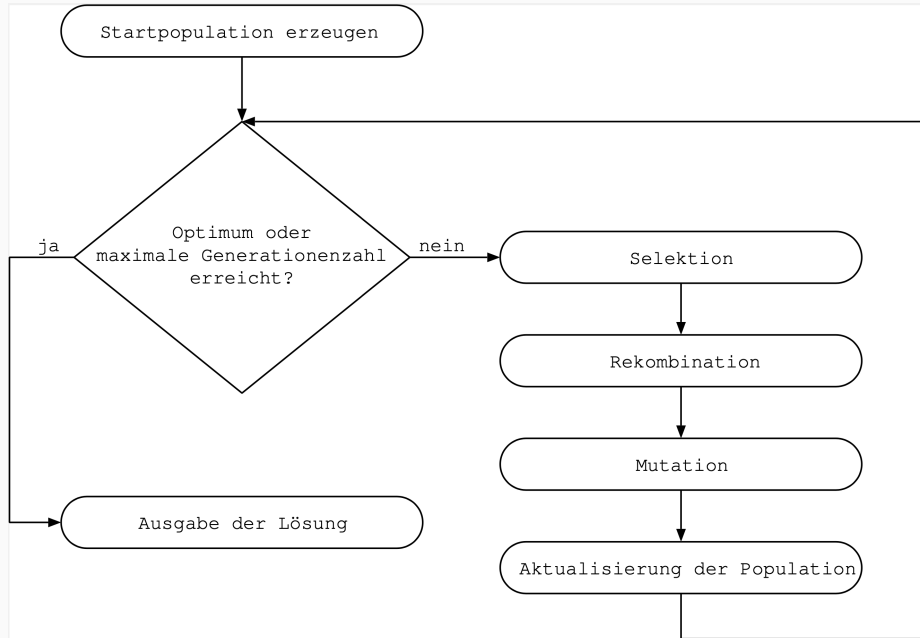


Modellierung mit Genetischen Algorithmen

Carsten Gips (FH Bielefeld)

Unless otherwise noted, this work is licensed under CC BY-SA 4.0.

EA – Allgemeiner Ablauf



- Binäre Lösungsrepräsentation (Bitstring): $\mathbf{g} = (g_1, \dots, g_m) \in \{0, 1\}^m$
 - String gliedert sich in n Elemente (mit $n \leq m$)
 \Rightarrow jedes Segment entspricht einer Problemvariablen
 - Dekodierungsfunktion $\Gamma : \{0, 1\}^m \rightarrow \mathbb{R}^n$
- Fitnessfunktion Φ ordnet jedem Individuum \mathbf{g}_i eine reelle Zahl zu:

$$\Phi(\mathbf{g}_i) = F(\Gamma(\mathbf{g}_i)) - w \cdot \sum_j (Z_j(\Gamma(\mathbf{g}_i)))^2$$

- Zielfunktion F : wie sehr genügt ein Individuum bereits dem Optimierungsproblem
- Strafterme Z_j : Anreicherung der Optimierung mit weiteren Informationen
- Gewichte w : statisch oder dynamisch (Abkühlen)

Selektion: Erstelle Matingpool mit μ Individuen

- Fitnessproportionale Selektion (*Roulette Wheel Selection*):
Auswahlwahrscheinlichkeit für Individuum \mathbf{g}_k :

$$p_{sel}(\mathbf{g}_k) = \frac{\Phi(\mathbf{g}_k)}{\sum_j \Phi(\mathbf{g}_j)}$$

=> Voraussetzung: positive Fitnesswerte

- Turnier-Selektion (*Tournament Selection*):
 - Turniergröße ξ
 - Turnier: ziehe ξ Individuen gleichverteilt (mit Zurücklegen!) und kopiere fittestes Individuum in den Matingpool
 - Führe μ Turniere durch

Crossover: Erzeuge zwei Nachkommen aus zwei Eltern

Festlegung der Crossover-Wahrscheinlichkeit p_{cross} (typisch: $p_{cross} \geq 0.6$)

1. Selektiere Eltern \mathbf{g}_a und \mathbf{g}_b **gleichverteilt** aus Matingpool
2. Zufallsexperiment:
 - mit $1 - p_{cross}$: Kinder identisch zu Eltern (kein Crossover)
 - mit p_{cross} : Crossover mit \mathbf{g}_a und \mathbf{g}_b
 - Ziehe i gleichverteilt mit $1 < i < m$
 - Kinder aus \mathbf{g}_a und \mathbf{g}_b zusammenbauen:

$$\mathbf{g}_c = (g_{a,1}, \dots, g_{a,i}, g_{b,i+1}, \dots, g_{b,m})$$

und

$$\mathbf{g}_d = (g_{b,1}, \dots, g_{b,i}, g_{a,i+1}, \dots, g_{a,m})$$

3. Gehe zu Schritt 1, bis insg. μ Nachkommen

Anmerkung: Die Eltern werden jeweils in die Ausgangsmenge zurückgelegt.

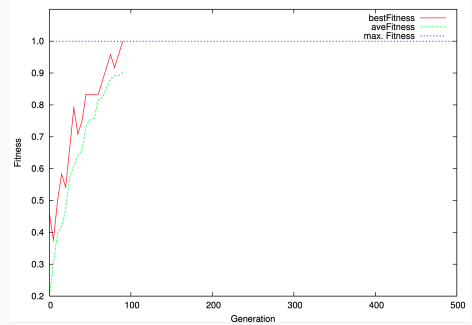
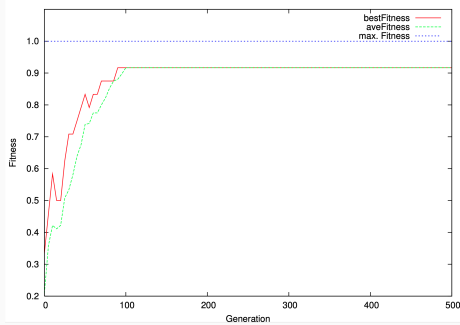
Mutation

- Mutationswahrscheinlichkeit p_{mut}
(typische Werte: $p_{mut} = 0.01$ oder $p_{mut} = 0.001$)
- Für alle Individuen:
 - Mutiere jedes Gen eines Individuums mit p_{mut} :

$$g_i^{(t+1)} = \begin{cases} \neg g_i^{(t)} & \text{falls } \chi_i \leq p_{mut} \\ g_i^{(t)} & \text{sonst} \end{cases}$$

$\Rightarrow \chi_i$ gleichverteilte Zufallsvariable (Intervall $[0, 1]$), für jedes Bit g_i neu bestimmen

Typische Läufe



- Populationsgröße $\mu = 15$
- Anzahl Nachfahren $\lambda = 100$
- Abbruch nach $maxGen = 200$ Generationen

Lokale Suchverfahren: Nur das Ergebnis zählt!

- Evolutionäre Algorithmen:
 - Begriffe: Individuum, Population, Kodierung
 - Operationen: Selektion, Rekombination, Mutation
 - Bewertung mit Fitnessfunktion



Unless otherwise noted, this work is licensed under CC BY-SA 4.0.