

7. Übungsblatt zur Vorlesung

Heuristische Optimierung

Aufgabe 1 (Evolutionäre Programmierung)

Untersuchen Sie an beispielhaften Zeitreihen und den unterschiedlichen Mutationen vom *ursprünglichen* Evolutionären Programmieren, inwieweit eine kleine Veränderung am Genotyp einer kleinen Veränderung am Phänotyp entspricht.

Aufgabe 2 (Genetische Programmierung)

Die ursprüngliche Repräsentation von Individuen als Bäume bei GPs stellt nur eine von mehreren Möglichkeiten dar, wie Programme repräsentiert werden können. Stattdessen kann hier auch mit einer linearen Struktur gearbeitet werden, bei der es beispielsweise um Maschinencode handelt. Aus einer kontextfreien Grammatik wird das Individuum so interpretiert, dass ausgehend von einem Startsymbol ein Syntaxbaum durch iterative Expansion der Nichtterminalsymbole aufgebaut wird. (*Grammatikevolution*)

Als Beispiel für die *Grammatikevolution* betrachten wir die folgende Grammatik:

$$S \rightarrow T_0 \mid x_1$$
$$T \rightarrow S+S_0 \mid S-S_1 \mid S*S_2 \mid S\%S_3$$

Die kleinen Zahlen geben dabei die Nummer der jeweiligen Ableitung an, S ist das Startsymbol.

Führen Sie die Grammatikevolution für die folgenden Individuen durch:

$$A = (0, 2, 1, 0, 0, 3, 1, 0, 2)$$
$$B = (2, 1, 1, 0, 0, 1, 0, 3)$$

Aufgabe 3 (Genetische Programmierung)

Geben Sie einen Algorithmus zur zufälligen Initialisierung der Individuen bei GPs an. Verwendung Sie die in der Vorlesung vorgestellten Konzepte.

Aufgabe 4

Betrachten Sie das Symbolic-Regression-Problem: Gegeben ist eine Menge von Datenpunkten, die durch eine mathematische Funktion angenähert werden soll. Welches evolutionäre Verfahren ist hierfür am besten geeignet? Wie sollten Kodierung, Bewertungsfunktion und evolutionäre Operatoren gewählt werden?