

## 1 Modellierung von GA

(6 Punkte)

Folgende Probleme sollen mit einem GA gelöst werden:

- a) das *Travelling Salesman Problem* für 10 Städte, d.h. das Finden der kürzesten Route zwischen 10 Städten,
- b) ein  $9 \times 9$ -*Sudoku*-Rätsel,
- c) das  $n$ -Queens-Problem (für ein beliebiges, aber festes  $n$ ).

Geben Sie für diese Probleme jeweils eine geeignete **Kodierung** der Individuen, passende Operatoren (**Crossover**, **Mutation**) und eine geeignete **Fitnessfunktion** an, damit das Problem mit einem GA gelöst werden kann. Begründen Sie Ihre Wahl!

Was würden Sie noch benötigen, um die Probleme jeweils mit Simulated Annealing lösen zu können?

**Thema:** Modellierung für GA und Gradientensuche

## 2 Implementierung

(8 Punkte)

Suchen Sie sich ein geeignetes Problem und implementieren Sie den in der Vorlesung besprochenen GA.

Wenn Sie den Algorithmus nicht komplett selbst schreiben wollen, können Sie beispielsweise die Java-Klassen im Paket `aima.core.search.local` als Ausgangspunkt nutzen.<sup>1</sup>

Untersuchen Sie **systematisch** unterschiedliche Varianten/Einstellungen der in der VL vorgestellten Operatoren. Führen Sie pro Einstellung jeweils mind. 100 Läufe durch und messen Sie die besprochenen Kennzahlen.

Erstellen Sie eine geeignete (systematische!) Auswertung Ihrer Experimente.

**Thema:** Implementierung von GA und Auswertung der Ergebnisse, wissenschaftl. Untersuchung

## 3 Heuristische Suche

(4 Punkte)

Konstruieren Sie in Form eines Graphen jeweils ein Beispiel, in welchem Best-First-Search

- a) genau dieselbe, bzw.
- b) eine bessere

Lösung findet als die Gradientensuche.<sup>2</sup>

Führen Sie für den letzteren Fall eine Handsimulation der beiden Algorithmen durch.

**Thema:** Verständnis Best-First-Search und Gradienten-Suche

## 4 Anwendungen

(2 Punkte)

Recherchieren Sie, in welchen Anwendungen Evolutionäre Algorithmen eingesetzt werden. Erklären Sie kurz, wie und wofür die EA/GA jeweils genutzt werden.

**Thema:** Anwendungen von EA/GA

<sup>1</sup>Sie finden das Repository unter [github.com/aimacode/aima-java](https://github.com/aimacode/aima-java).

<sup>2</sup>Die Bewertung der Aktionen und die Heuristik muss dabei für beide Suchverfahren einheitlich sein.