

5. Übungsblatt

(zum 18.11., 04.12. bzw. 05.12.2013)

Aufgabe 16 Evolutionstheorie

Die biologische Evolutionstheorie wird z.B. von den sogenannten Kreationisten (die glauben, dass der Schöpfungsbericht der Bibel wortwörtlich richtig ist) heftig kritisiert, und zwar unter anderem mit dem folgenden Argument:

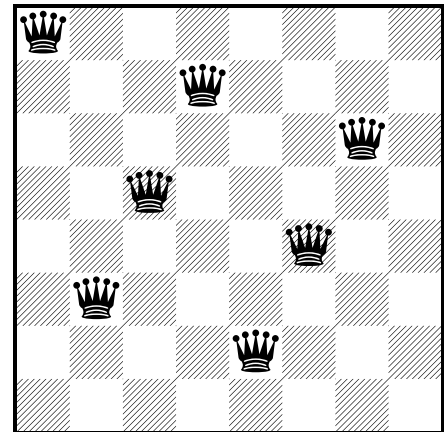
„Die Evolutionstheorie kann die Komplexität der Lebewesen nicht erklären, denn sie beruht ja auf blindem, zufälligem Probieren. Sie behauptet im Grunde, dass man einen VW-Käfer erhalten kann, indem man einen Haufen Schrott in einen Kasten schüttet, und dann diesen Kasten lange genug schüttelt. Aber es ist doch offensichtlich, dass man so keine Autos bauen kann. Das Vorhandensein derart komplexer Dinge wie Autos, geschweige denn Lebewesen, kann man nur durch Annahme eines Schöpfers erklären.“

- Was ist an dem Argument richtig, was falsch?
- Was würden Sie einem Kreationisten, der dieses Argument vorträgt, antworten?
- Warum funktionieren evolutionäre Algorithmen (obwohl gegen sie ein analoges Argument vorgebracht werden kann)?

Aufgabe 17 n -Damen-Problem

Beim n -Damen-Problem geht es darum, n gleichfarbige Damen (Figur beim Schachspiel, die horizontal, vertikal und diagonal beliebig weit ziehen kann) so auf einem $n \times n$ -Schachbrett aufzustellen, dass keine auf einem Feld steht, auf das eine andere ziehen könnte, keine Dame also einer anderen „im Weg steht“. Dürften Figuren gleicher Farbe einander schlagen, könnte man auch sagen: so, dass keine eine andere schlagen kann.

Das nebenstehende Diagramm zeigt *keine* Lösung des 8-Damen-Problems, da nur sieben Damen aufgestellt sind und keine weitere mehr platziert werden kann.



Geben Sie an, wie man das n -Damen-Problem mithilfe eines evolutionären Algorithmus lösen könnte. Gehen Sie dabei auf die folgenden Punkte ein:

- Erklären Sie die Elemente eines evolutionären Algorithmus mithilfe des generischen Grundalgorithmus aus der Vorlesung.
- Versuchen Sie anschaulich zu machen, wie der evolutionäre Algorithmus eine Lösung findet.

- c) Entwerfen Sie eine Kodierung für die Lösungskandidaten, sodass jeder Kandidat anhand einer Zeichenkette eindeutig beschrieben werden kann.

Hinweis: In jeder Zeile kann nur eine Dame stehen. Lösungen mit mehr als einer Dame in einer Zeile können so schon durch die Kodierung ausgeschlossen werden.

- d) Welche Fitnessfunktion käme in Frage? Welche Variations- und Rekombinationsoperatoren wären geeignet?
- e) Welche Funktion haben Variation und Rekombination anschaulich?

Aufgabe 18 Vierfarbenproblem

Das Vierfarbenproblem ist eines der berühmtesten Probleme der Mathematik. Es besteht aus der Frage, ob jede Landkarte mit höchstens vier Farben so eingefärbt werden kann, dass keine zwei Staaten mit gemeinsamer Grenze die gleiche Farbe haben. Dieses Problem, das im Jahre 1852 von Francis Guthrie aufgeworfen wurde (publiziert 1878), war lange ungelöst. Erst im Jahre 1976 konnten Wolfgang Haken und Kenneth Appel den Vierfarbensatz mithilfe eines umfangreichen Computerprogramms beweisen.

- a) Betrachten Sie eine (politische) Karte von Mitteleuropa mit den Staaten Belgien, Deutschland, Frankreich, Luxemburg, Niederlande, Österreich, Polen, Schweiz, Slowakei, Tschechische Republik und Ungarn. Wie kann man für diese Karte eine dem Vierfarbensatz genügende Färbung mithilfe eines evolutionären Algorithmus finden?
- b) Verallgemeinern Sie den Ansatz aus Teilaufgabe a) auf beliebige Graphenfärbungsprobleme: Bestimmen von Farben für die Knoten eines Graphen, sodass keine zwei Knoten, die durch eine Kante verbunden sind, gleiche Farbe haben.

Aufgabe 19 Genetische Programmierung: SR-Agent

Geben Sie einen Algorithmus an, der einen zufälligen Lisp/Scheme-Ausdruck erzeugt, wie er in der Vorlesung zur Darstellung der Programme für den Stimulus-Response-Agenten der Gitterwelt verwendet wurde!

Verwenden Sie als Operatoren `if`, `and`, `or` und `not` and als Konstanten/Variablen `east`, `north`, `west`, `south` (Aktionen), `s1`, `s2`, `...`, `s8` (Sensoreingaben). Sehen Sie als Parameter die maximale Verschachtelungstiefe des Ausdrucks vor.