Universität Bremen

Nicole Drechsler
Email: nd@informatik.uni-breme.de
Tel: 0421 218 63948

Bremen, 20. Oktober 2009

Auswahlliste der Optimierungsprobleme zur Vorlesung

Heuristische Optimierung

Im Folgenden stehen einige Problemstellungen zur Auswahl, die mit einem hier angegebenen und aus der Vorlesung bekannten Optimierungsverfahren bearbeitet werden sollen.

Zur Wahl der Programmiersprache gibt es keine Einschränkungen, geeignet sind Java oder C++. Zum Testen der Algorithmen werden für die jeweiligen Problemstellungen Testinstanzen (Benchmarks) benötigt, für die exemplarisch Referenzen angegeben sind. Oftmals hängt die Performanz einer Heuristik von Parametern ab, so dass eine Untersuchung von unterschiedlichen Parameterwerten sinnvoll ist, weil sie euch Einblicke in die Funktionsweise der Algorithmen gibt.

Das heißt nun, eine Gruppe von 1-2 Personen sucht sich ein Optimierungsproblem aus und bearbeitet dieses mit einem Verfahren nach Wahl aus untenstehender Liste. Natürlich dürft ihr auch wählen, mit wem ihr zusammenarbeiten möchtet und welche Programmiersprache ihr benutzen wollt … ☺

Zur Auswahl stehende Verfahren und Untersuchungen zur Optimierung:

- Untersuchung von Konstruktionsheuristiken in Kombination mit lokalen Suchen
- Simulated Annealing
- Vergleichsstudie von *Threshhold Accepting* und *Sintflut-Methode*
- Evolutionärer Algorithmus
- Tabu Suche
- Ameisenkolonie
- Partikel-Schwärme
- Branch&Bound (für *kleine* Probleminstanzen)

Optimierungsprobleme

Aufgabe 1 – N-Damenproblem
Auf einem Schachbrett der Größe NxN sollen N Damen so aufgestellt werden, dass keine eine andere schlagen kann.

Aufgabe 2 – Handlungsreisendenproblem (TSP)
Die Definition ist bereits bekannt aus der Vorlesung (Folien). Instanzen zum Testen der Optimierungsverfahren gibt es beispielsweise unter: http://www.tsp.gatech.edu/ oder http://www.sintef.no/static/am/opti/projects/top/vrp/benchmarks.html

Aufgabe 3 – Graph-Bipartitionierung (GBP)
Die Definition des Problems ist ebenfalls bekannt aus der Vorlesung. Instanzen zum Testen der Optimierungsverfahren gibt es beispielsweise unter:
http://netmob.unitn.it/graph-benchmark.html

Aufgabe 4 – Maximum Clique (MAXCL)
Die Definition des Problems ist ebenfalls bekannt aus der Vorlesung. Instanzen zum Testen der Optimierungsverfahren gibt es beispielsweise unter:
http://www.nlsde.buaa.edu.cn/~kexu/benchmarks/graph-benchmarks.htm

Aufgabe 5 – Maximum Independent Set (MIS)
Die Definition des Problems ist ebenfalls bekannt aus der Vorlesung. Instanzen zum Testen der Optimierungsverfahren gibt es beispielsweise unter:
http://www.nlsde.buaa.edu.cn/~kexu/benchmarks/graph-benchmarks.htm

Aufgabe 6 – Vehicle Routing Problem (VRP)
Die Definition des Problems ist ebenfalls bekannt aus der Vorlesung. Instanzen zum Testen der Optimierungsverfahren gibt es beispielsweise unter:
http://www.sintef.no/static/am/opti/projects/top/vrp/benchmarks.html

Aufgabe 7 – Quadratic Assignment Problem (QAP)
Die Definition des Problems ist ebenfalls bekannt aus der Vorlesung. Instanzen zum Testen der Optimierungsverfahren gibt es beispielsweise unter:
http://dbkweb.ch.umist.ac.uk/knowles/mQAP/#suites

Aufgabe 8 – Freies Thema
Vielleicht habt ihr selbst ein Optimierungsproblem, das ihr angehen möchtet?
Bitte mit Nicole vorher absprechen.