

5. Aufgabe zum Softwarepraktikum “Muster- und Bilderkennung” im Grundstudium: Traveling Salesman Problem

Stefan Hahn, Georg Heigold

Human Language Technology and Pattern Recognition
Lehrstuhl für Informatik VI
Computer Science Department
RWTH Aachen University
52056 Aachen, Germany

Aufgabenstellung

Bekanntes Optimierungsproblem: Traveling Salesman Problem

- Handlungsreisender muss n Städte genau einmal besuchen,
- Kosten für jede Reise von einer Stadt zur nächsten,
- Reise beginnt und endet in derselben Stadt,
- Gesamtkosten der Reise sollen minimal sein.

Formalisierung des Problems

- Gegeben
 - Menge von Städten $\{s_1, \dots, s_n\}$,
 - Distanzen zwischen Städten $d(s_i, s_j) > 0$ (keine Verbindung $\rightarrow d(s_i, s_j) = \infty$).
- Gesucht: Permutation π so dass $\sum_{i=1}^{n-1} d(s_{\pi(i)}, s_{\pi(i+1)}) + d(s_{\pi(n)}, s_{\pi(1)})$ minimal ist.

Aufgabe 1

Implementieren Sie eine naive Lösung des Traveling Salesman Problem, indem Sie alle möglichen Rundreisen durchprobieren und die optimale Tour bestimmen (Komplexität $n!$).

Aufgabe 2

Implementieren Sie eine Lösung des Traveling Salesman Problems mit dynamischer Programmierung.

Algorithmus von Held/Karp

- Komplexität = $2^n \cdot n^2$;
- rekursive Größe: $D(S, k)$ = minimale Kosten für einen Weg von Stadt s_1 zu s_k , der alle Städte aus der Menge S besucht.

Eingabe: n Städte $i = 1, \dots, n$ mit Abstandsmatrix d_{ij}	
Initialisierung: $D(\{j\}, j) = d_{1j}$ für $j = 2, \dots, n$	
für jede Kardinalität $l = 2, \dots, n - 1$	
für jede Menge S mit $ S = l$	
	für jede Stadt $j \in S$
	$D(S, j) = \min_{j' \in S \setminus \{j\}} \{d_{jj'} + D(S \setminus \{j\}, j')\}$

Ein/Ausgabe

Eingabe

- Spalte 1: Stadt A
- Spalte 2: Stadt B
- Spalte 3: $d(A, B) > 0$
- keine Verbindung: $d(A, B) = \infty$ (nicht explizit angegeben)

Ausgabe

- Kosten der optimalen Rundreise; für Aufgabe 2:

$$D^* = \min_{j \in \{s_2, \dots, s_n\}} \{D(\{s_2, \dots, s_n\}, j) + d_{j, s_1}\}$$

- Permutation π (Traceback der optimalen Rundreise)