LEHRBUCH

Ralf Hartmut Güting
Stefan Dieker

Datenstrukturen und Algorithmen

4. Auflage

Springer Vieweg

Der Aufwand für (a) und (b) beträgt insgesamt O(m_i log n).

Es gilt: $\sum m_i = e$ mit $e = |E|$. Über alle Schritte des Algorithmus summiert ist der Aufwand für (2) O(e log n). Der Aufwand für (1) ist ebenfalls O(e log n), da ein Element nur aus der Priority Queue entnommen werden kann, wenn es vorher eingefügt wurde. Also ist der Gesamtaufwand bei dieser Implementierung O(e log n), der Platzbedarf ist O($n + e$).

7.2 Bestimmung kürzester Wege zwischen allen Knoten im Graphen

Wir betrachten die Bestimmung der kürzesten Wege zwischen allen Paaren von Knoten, bekannt als das "*all pairs shortest path*"-Problem. Man kann zu seiner Lösung natürlich den Algorithmus von Dijkstra iterativ auf jeden Knoten des Graphen anwenden. Es gibt hierfür aber den wesentlich einfacheren *Algorithmus von Floyd*. Dieser Algorithmus wird gewöhnlich anhand der Kostenmatrix-Darstellung des Graphen erklärt. Wir geben hier eine allgemeinere Formulierung an, die noch unabhängig von der tatsächlichen Repräsentation ist.

Der Algorithmus berechnet eine Folge von Graphen $G_0, ..., G_n$ (unter der Annahme, dass der Ausgangsgraph G n Knoten hat, die von 1 bis n durchnummeriert sind). Graph G_i entsteht jeweils durch Modifikation des Graphen G_{i-1}. Jeder Graph G_i ist wie folgt definiert:

(i) G_i hat die gleiche Knotenmenge wie G.

(ii) Es existiert in G_i eine Kante $v \xrightarrow{\alpha} w$ $\Leftrightarrow$ es existiert in G ein Pfad von v nach w, in dem als Zwischenknoten nur Knoten aus der Menge $\{1, ..., i\}$ verwendet werden. Der kürzeste derartige Pfad hat Kosten α. – Dabei bezeichne die Notation $v \xrightarrow{\alpha} w$ eine Kante (v, w) mit $\nu(v, w) = \alpha$.

Sei $G_0 = G$. Man beachte, dass G_0 bereits die obige Spezifikation der G_i erfüllt. Der *i-te Schritt* des Algorithmus von Floyd berechnet G_i aus G_{i-1} wie folgt:

Seien $v_1, ..., v_r$ die Vorgänger von Knoten i im Graphen G_{i-1} und seien $w_1, ..., w_s$ die Nachfolger. Betrachte alle Paare (v_j, w_k) mit $j = 1, …, r$ und $k = 1, …, s$.