

Effiziente Algorithmen I

6. Übungsblatt WS 02/03

Abgabetermin: 17.12.02

Aufgabe 18

Varianten des Zuordnungsproblems. Für die folgenden Probleme sei jeweils ein bipartiter Graph $G = (N_1 \cup N_2, E)$ mit Kantengewichten c_{ij} gegeben. Außerdem sei eine Subroutine `findAssignment(E')` bekannt, die für jede Kantenmenge $E' \subseteq E$ in der Zeit $O(\sqrt{N_1 + N_2} |E'|)$ eine Zuordnung findet, die nur Kanten aus E' enthält oder ausgibt, dass keine solche Zuordnung existiert.

- **Bottleneck assignment problem.** Gesucht ist eine Zuordnung, in der das minimale in der Zuordnung vorkommende Gewicht möglichst groß ist. Finden Sie einen $O(\log k \sqrt{N_1 + N_2} |E|)$ Algorithmus, der dieses Problem optimal löst. Dabei ist k die Anzahl der verschiedenen Werte der Gewichte c_{ij} , also $k \leq |E|$.
- **Balanced assignment problem.** Gesucht ist eine Zuordnung, in der die Differenz zwischen größtem und kleinstem vorkommenden Gewicht möglichst klein ist. k sei definiert wie oben. Finden Sie einen Algorithmus, der dieses Problem in der Zeit $O(k \sqrt{N_1 + N_2} |E|)$ optimal löst.
- **Factored assignment problem.** Hier sei $E = N_1 \times N_2$ und alle Gewichte seien von der Form $c_{ij} = a_i a_j$, wobei die a_i festgelegte Knotengewichte aller Knoten aus $N_1 \cup N_2$ sind. Gesucht ist eine Zuordnung mit minimalem Gesamtgewicht. Können Sie einen $O((N_1 + N_2) \log(N_1 + N_2))$ Algorithmus angeben, der dieses Problem löst?

Aufgabe 19

Geben Sie eine Methode an, mit der man möglichst effizient die Konstruktion des neuen Super-Waldes und das Berechnen der neuen Dualvariablen im Iterationsschritt des AKP-Algorithmus bewerkstelligen kann. Dabei sollten Sie sowohl ein geeignetes Speicherformat für Super-Wälder als auch ein schnelles Update-Verfahren angeben.