Institut für Theoretische Informatik
Lehrstuhl Prof. Dr. D. Wagner

Übungsblatt 2

Vorlesung Algorithmentechnik im WS 09/10

Ausgabe 5. November 2009
Abgabe 19. November, 15:30 Uhr in der Übung (oder im Kasten vor Zimmer 319, Geb. 50.34)

Bitte schreiben Sie Ihren Namen und Ihre E-Mailadresse aber keine Matrikelnummer auf Ihr Übungsblatt. Die Bearbeitung in Zweiergruppen ist ausdrücklich erwünscht.

Problem 1: Minimal spannende Bäume – ein bisschen Theorie [vgl. Kapitel 2.1–2.5 im Skript] *

(a) Zeigen Sie: Wenn in einem Graphen G mit reellen Kantengewichten für jeden Schnitt die den Schnitt kreuzende Kante minimalen Gewichts eindeutig ist, dann hat G einen eindeutigen spannenden Baum minimalen Gewichts.

(b) Gilt die Umkehrung von (a)?

(c) Zeigen oder widerlegen Sie: Wenn in einem Graphen G mit reellen Kantengewichten alle Kanten paarweise verschiedene Gewichte haben, dann hat G einen eindeutigen spannenden Baum minimalen Gewichts.

(d) Zeigen oder widerlegen Sie die Umkehrung der Aussage in (c).

Problem 2: Zweitbeste minimal spannende Bäume [vgl. Kapitel 2.1–2.5 im Skript] **

Sei $G = (V, E)$ ein ungerichteter, zusammenhängender Graph mit Gewichtsfunktion $w : E \to \mathbb{R}$. Weiterhin seien alle Kantengewichte paarweise verschieden.

Wir definieren einen *zweitbesten minimal spannenden Baum* B wie folgt: Sei $\mathcal{B}$ die Menge aller spannenden Bäume von G, und $B_{\min} \in \mathcal{B}$ der minimal spannende Baum mit Gewicht $w(B_{\min})$. Dann ist der zweitbeste minimal spannende Baum B definiert durch $B := \operatorname{argmin}_{B \in \mathcal{B} \setminus \{B_{\min}\}} \{w(B)\}$.

(a) Zeigen oder widerlegen Sie: Ist der minimal spannende Baum eindeutig, so ist auch der zweitbeste minimal spannende Baum eindeutig.

(b) Sei $B_{\min}$ der minimal spannende Baum in G. Zeigen Sie: Es gibt Kanten $e \in B_{\min}$ und $e' \in E \setminus B_{\min}$ so, dass $B_{\min} \setminus \{e\} \cup \{e'\}$ ein zweitbester minimal spannender Baum von G ist.

(c) Sei B ein spannender Baum in G. Für zwei Knoten $u, v \in V$ werde mit $\operatorname{maxedge}(u, v)$ die Kante *maximalen* Gewichts auf dem eindeutigen u-v-Pfad in B bezeichnet. Geben Sie einen Algorithmus mit Laufzeit $\mathcal{O}(|V|^2)$ an, der $\operatorname{maxedge}(u, v)$ für alle Knotenpaare berechnet.

Hinweis: Starten Sie von jedem Knoten eine modifizierte Breitensuche eingeschränkt auf B.

(d) Basierend auf Aufgabe (b) und (c), geben Sie einen Algorithmus mit Laufzeit $\mathcal{O}(|V|^2)$ an, der einen zweitbesten minimal spannenden Baum berechnet.

Problem 3: Matroide und kürzeste Wege [vgl. Kapitel 2.6 im Skript] ***

Gegeben sei ein gerichteter, gewichteter und stark zusammenhängender Graph $G = (V, E)$ mit Gewichtsfunktion $w : E \to \mathbb{R}^+$ und eindeutigen kürzesten Wegen, das heißt für je zwei Knoten $s, t \in V$ existiert genau ein kürzester s-t-Weg.

Der Algorithmus von Dijkstra berechnet zu einem gegebenem Knoten $s \in V$ die kürzesten Wege von s zu jedem Knoten $t \in V$ durch ein Greedy-Verfahren (siehe Algorithmus 1).

Algorithmus 1 : Dijkstra's Algorithmus

Eingabe : Graph $G = (V, E)$ und Startknoten $s \in V$.
Seiteneffekte : Kürzeste Wege zu allen Knoten $u \in V$. Distanzen zu allen Knoten $u \in V$.

```
S ← V;                          // Menge abzuarbeitender Knoten
P ← ∅;                          // Menge der Kanten auf den kürzesten Wegen
pre(u) ← ⊥ für alle u ∈ V;      // Knoten über den u erreicht wird
dist(u) ← ∞ für alle u ∈ V;     // Länge des kürzesten s-u-Weges
dist(s) ← 0
solange S ≠ ∅ tue
    u ← argmin_{u∈S}{dist(u)}
    S ← S \ {u};                // Kürzester Weg zu u ist berechnet
    P ← P ∪ {(pre(u), u)};      // Kante die zu u auf dem kürzesten Weg führt
    // Expandiere zu adjazenten Knoten von u
    für alle e = (u, v) ∈ E tue
        Wenn dist(v) > dist(u) + w(e)
            dist(v) ← dist(u) + w(e)
            pre(v) ← u
```

Wir wollen in dieser Aufgabe die Korrektheit von Dijkstra's Verfahren beweisen, indem wir es auf ein Optimierungsproblem über einer Matroid-Struktur zurückführen.

(a) Begründen Sie, warum Dijkstra's Algorithmus ein Greedy-Verfahren ist.

(b) Definieren Sie ein geeignetes unabhängiges Mengensystem $(M, \mathcal{U})$. Dabei sei M die Menge aller azyklischen Pfade in G, die bei s anfangen. Überlegen Sie sich ausgehend davon, wie die Lösung von Dijkstra's Algorithmus zusammengesetzt ist und definieren Sie darauf basierend das Mengensystem $\mathcal{U}$.

(c) Zeigen Sie, dass $(M, \mathcal{U})$ ein Matroid ist.

(d) Geben Sie eine Kostenfunktion c für die Elemente $I \in M$ an.

(e) Zeigen Sie, dass die Greedy-Methode über $(M, \mathcal{U})$ mit Dijkstra's Algorithmus übereinstimmt.