Universität Karlsruhe
Fakultät für Informatik
ITI Wagner

Algorithmentechnik
WS 06/07

3. Übungsblatt

Ausgabe: 28. November 2006
Abgabe: 6. Dezember, 15:30 Uhr im ITI Wagner (Informatik-Hauptgebäude, 3. Stock)

Die Bearbeitung in Zweiergruppen ist ausdrücklich erwünscht.

Problem 1: Boruvka-MST 3pt

Sei $G = (V, E)$ ein einfacher, ungerichteter, zusammenhängender Graph mit injektiver Kostenfunktion $w : E \to \mathbb{N}^+$. Wir bezeichnen eine Kante als *lokal minimal* wenn es einen Knoten gibt, mit dem sie inzidiert und wenn sie unter allen Kanten, die mit diesem Knoten inzidieren, das kleinste Gewicht hat.

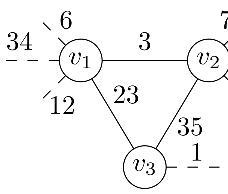
(a) Zeigen sie: Jede lokal minimale Kante gehört zu allen MST von G.

(b) Die Menge aller lokal minimalen Kanten eines Graphen ist ein Wald.

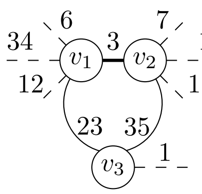
Eine *Boruvka-Phase* ist ein Algorithmus, der Folgendes leistet:

Schritt (1) Er bestimmt die Menge $L(G)$ aller lokal minimalen Kanten von G.

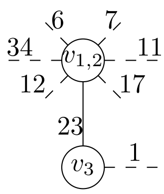
Schritt (2) Er berechnet den kontrahierten Graphen $G/L(G)$, wie in der Abbildung unten illustriert. Falls Mehrfachkanten entstünden, wird nur die Kante mit geringsten Gewicht behalten, und die anderen entfernt.



Der Graph vor der Boruvka Phase. Kante mit Gewicht 3 ist lokal minimal.



Knoten v_1 und v_2 werden entlang Kante mit Gewicht 3 kontrahiert.



Knoten v_1 und v_2 kontrahiert; teurere, doppelte Kante zu v_3 gelöscht.

Der so aus G entstehende Graph werde mit $B(G)$ bezeichnet.

(c) Implementieren sie in Pseudocode die Boruvka-Phase mit einer Worst-case-Laufzeit von $O(m)$ Vergleichsoperationen. Begründen Sie die Laufzeit. Ihnen steht eine Funktion LöscheTeureMehrfachkanten(G) zur Verfügung, die (im Worst-case) mit $O(m)$ Vergleichsoperationen auskommt.

(d) Zeigen sie, dass durch eine Boruvka-Phase die Anzahl der Knoten um mindestens die Hälfte reduziert wird.

Man kann zeigen, dass für jeden Graphen G gilt: Die Kanten in $L(G)$ und die Kanten in einem MST von $B(G)$ bilden zusammen einen MST von G.

(e) Nutzen sie Teilaufgabe (c) um einen MST mit $O(m \log n)$ Vergleichsoperationen im Worst-case zu bestimmen (Pseudocode). Begründen sie die Worst-case-Laufzeit von $O(m \log n)$.

Problem 2: Maybe-MST 2pt

Die Algorithmen 1 bis 3 seien gegeben. Alle verwenden als Eingabe einen ungerichteten, einfachen gewichteten Graphen G (mit Gewichtsfunktion $w : E \to \mathbb{R}$) und geben eine Kantenmenge T aus. Beweisen oder widerlegen sie für jeden Algorithmus, dass T ein MST von G ist. Geben Sie die asymptotische Laufzeit an (Worst-case, möglichst scharfe obere Schranke, Vergleichsoperationen).

Algorithmus 1 : MAYBE-MST-A(G, w)

sortiere die Kanten in nichtsteigender Reihenfolge der Gewichte w
$T \leftarrow E$
Für *jede Kante e in nichtsteigender Reihenfolge der Gewichte w*
    Wenn $T - \{e\}$ *ist ein zusammenhängender Graph*
        $T \leftarrow T - \{e\}$
return T

Algorithmus 2 : MAYBE-MST-B(G, w)

$T \leftarrow \emptyset$
Für *jede Kante e in beliebiger Reihenfolge*
    Wenn $T \cup \{e\}$ *hat keine Zyklen*
        $T \leftarrow T \cup \{e\}$
return T

Algorithmus 3 : MAYBE-MST-C(G, w)

$T \leftarrow \emptyset$
Für *jede Kante e in beliebiger Reihenfolge*
    $T \leftarrow T \cup \{e\}$
    Wenn T *hat einen Zyklus c*
        $e' \leftarrow$ Kante mit maximalem Gewicht auf c
        $T \leftarrow T - \{e'\}$
return T

Problem 3: MST 2pt

Zeigen Sie, dass ein Graph einen eindeutigen minimalen Spannbaum besitzt, falls es für jeden Schnitt des Graphen eine eindeutige leichte Kante gibt, die diesen Schnitt kreuzt. Zeigen Sie durch ein Gegenbeispiel, dass die Umkehrung dieser Aussage nicht gilt.

Bemerkung: Von allen Kanten, die einen Schnitt kreuzen, heißt eine Kante mit geringstem Gewicht *leicht*.

Problem 4: Stoer-Wagner 1pt

Wenden Sie auf den rechts abgebildeten Graphen (Kantengewichte in Klammern) den Algorithmus von Stoer & Wagner an. Geben Sie nach jeder Phase die Knoten s und t, den Schnitt der Phase und dessen Gewicht an und zeichnen Sie den nach dem Verschmelzen resultierenden Graphen (mit Kantengewichten). **Verwenden Sie in Phase i den Knoten als Startknoten, der Knoten i des Originalgraphen enthält.** Geben Sie zum Schluss den minimalen Schnitt S_{min} an.