Übung zur Vorlesung Effiziente Algorithmen

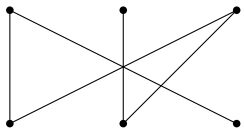
Tutoraufgabe 4

Gegeben sind ein s-t-Netzwerk mit ganzzahligen Kapazitäten und ein ganzzahliger, maximaler Fluß. Wenn die Kapazität *einer* Kante um eins erhöht wird, wie schnell kann dann ein neuer maximaler Fluß berechnet werden?

Lösung: Dies geht in einem Schritt, da die optimale Lösung durch die Änderung höchstens um den Wert eins erhöht wurde und der Ford–Fulkerson-Algorithmus den alten Fluß in jedem Schritt um einen ganzzahligen Wert verbessert.

Tutoraufgabe 5

Verwenden Sie Preflow-Push, um ein maximales Matching für den folgenden Graphen zu ermitteln.



Lösung: Man wende den Algorithmus aus der Vorlesung an, welcher in wenigen Schritten ein perfektes Matching errechnet.

Tutoraufgabe 6

Beweisen Sie den Satz von Hall: Sei $G = (A, B, E)$ ein bipartiter Graph. Ein Matching, das jeden Knoten aus A überdeckt, existiert genau dann, wenn für die Nachbarschaft $N(S)$ einer jeden Teilmenge $S \subseteq A$ die Ungleichung $|N(S)| \geq |S|$ erfüllt ist.

Lösung:
Wenn es ein $S \subseteq A$ mit $|N(S)| < |S|$ gibt, dann kann offenkundig kein Matching alle Knoten aus S überdecken.
Es gelte nun $\forall S \subseteq A.\ |N(S)| \geq |S|$, und M sei ein Matching maximaler Kardinalität in G. Setze

- $X_0 = \{u \in A \mid u \text{ nicht überdeckt}\}$,
- $X = \{u \in A \mid \exists s \in X_0 \exists M\text{-alternierender Pfad von } s \text{ nach } u\}$ und
- $Y = \{u \in B \mid \exists s \in X_0 \exists M\text{-alternierender Pfad von } s \text{ nach } u\}$.

Dann gilt $N(X) \subseteq Y$. Außerdem wird jeder Knoten in Y von M überdeckt (sonst gäbe es einen augmentierenden Pfad, im Widerspruch zur Maximalität von M), daher gilt auch $|X \setminus X_0| = |Y|$. Also $|X_0| + |Y| = |X_0| + |X \setminus X_0| = |X| \leq |N(X)| \leq |Y|$ und damit $X_0 = \emptyset$.

– bitte wenden –

Hausaufgabe 3 (10 Punkte)

In dieser Aufgabe betrachten wir hypothetische Internetgraphen auf Städten: Für jede Stadt, die mit dem Internet verbunden ist, gibt es einen Knoten. Wenn zwei Städte über eine direkte Internetleitung miteinander verbunden sind, werden die entsprechenden Knoten durch eine ungerichtete Kante verknüpft. Geben Sie einen effizienten Algorithmus für das folgende Problem an:

Gegeben ist ein Internetgraph mit zwei markierten Städten. Diese beiden Städte sollen nun voneinander getrennt werden, wozu entweder Leitungen durchschnitten oder Städte ausradiert werden müssen. Das Durchschneiden einer Leitung kostet genau einen Euro (symbolischer Kaufpreis für einen Bagger aus der Konkursmasse eines Bauunternehmens), während das Ausradieren einer Stadt bekanntermaßen Kosten von $\sqrt{2}$ Euro aufwirft. Die beiden markierten Sädte hingegen dürfen nicht ausradiert werden (das wäre zu auffällig).

Ansatz: Hier bietet sich die Suche nach einem minimalen Schnitt an. Da aber auch Knoten (Städte) entfernt werden dürfen, müssen wir zunächst jeden Knoten v_i in einen Eingangs- und einen Ausgangsknoten e_i und a_i aufteilen, wobei eine Kante mit Kapazität $\sqrt{2}$ von e_i nach a_i läuft.

Eine Möglichkeit Kanten einzufügen, wäre jede Kante $\{v_i, w_i\}$ aus dem Internetgraphen in die gerichteten Kanten (a_i, e_j) und (a_j, e_i) mit Kapazität eins zu überführen. Dann müßte allerdings bewiesen werden, daß immer nur höchstens eine dieser beiden Kanten geschnitten werden, da sonst die Kosten nicht mit der Ursprungsinstanz übereinstimmen.

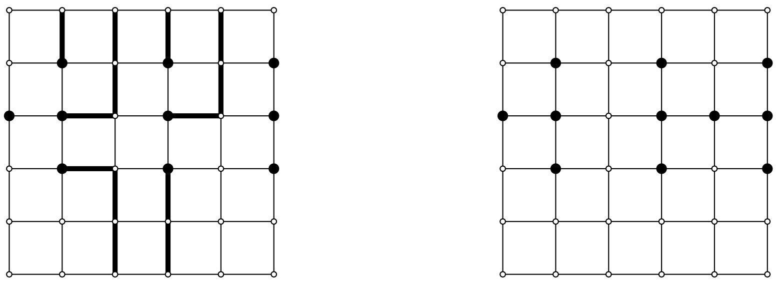
Anstelle dessen führen wir für jede Kante $e = \{u, v\}$ folgendes Gadget ein: Wir fügen die Knoten e_1 und e_2 hinzu sowie die Kanten $(u, e_1), (v, e_1), (e_1, e_2), (e_2, u), (e_2, v)$. Es ist leicht einzusehen, daß falls eine dieser Kanten geschnitten wird, wir einen genauso guten Schnitt erhalten indem wir (e_1, e_2) schneiden. Dies unterbindet sowohl die Verbindung von u nach v als auch von v nach u.

Abschließend wählen wir die beiden markierten Städte als Quelle bzw. Senke und berechnen den minimalen Schnitt, der dann genau einer Lösung entspricht.

Hausaufgabe 4 (10 Punkte)

Gegeben ist eine Leiterplatte mit einem rechteckigen Gitternetz. Auf gewissen Gitterpunkten gibt es Anschlüsse, die mit einem Lötpunkt über eine Leiterbahn verbunden werden sollen. Der Lötpunkt kann an beliebiger Stelle am Rand des Gitters liegen. Die Leiterbahn muß entlang der Gitterlinien verlaufen. Natürlich dürfen sich Leiterbahnen nicht überkreuzen.

Beispiele (links gibt es eine Lösung, rechts aber nicht):



Entwerfen Sie einen Algorithmus, der eine Lösung findet oder meldet, daß es keine Lösung gibt.

Ansatz:

Wir modellieren dieses Entwurfsproblem als Flußproblem:

Zunächst überführen wir jeden Knoten v_i der Leiterplatte in einen Eingangs- und einen Ausgangsknoten e_i und a_i, wobei eine Kante mit Kapazität eins von e_i nach a_i führt. Die ungerichteten Kanten $\{v_i, v_j\}$ der Leiterplatte übersetzen wir jeweils in zwei gerichtete Kanten (a_i, e_j) und (a_j, e_i). Abschließend führen wir Kanten mit Kapazität eins von der Quelle s zu jedem Anschluß und von jedem Randpunkt zur Senke t.

Zu zeigen ist nun: Es gibt genau dann einen Fluß, dessen Wert der Anzahl der Anschlußpunkte entspricht, wenn es eine gültige Positionierung der Leiterbahnen gibt, bei der alle Anschlußpunkte nach außen geführt werden. Dies läßt sich recht einfach konstruktiv beweisen.