Lehrstuhl für Informatik I
Algorithmen, Komplexität und
wissensbasierte Systeme
Universität Würzburg

Würzburg, den 21. Mai 2019

Prof. Dr. Alexander Wolff
Johannes Zink, M. Sc.

4. Übungsblatt zur Vorlesung Algorithmische Graphentheorie (Sommer 2019)

Aufgabe 1 – Matchings in Bäumen

a) Beweisen oder widerlegen Sie: Besitzt ein Baum ein perfektes Matching, so ist dieses eindeutig. **2 Punkte**

Hinweis: Betrachten Sie die Blätter des Baumes, d. h. die Knoten $v \in V$ mit $\deg v = 1$, und ihre jeweilige (einzige) inzidente Kante.

b) Wie viele andere Probleme ist auch das Problem MAXIMUM MATCHING in Bäumen einfacher als in allgemeinen Graphen.

Geben Sie einen effizienten Algorithmus in Pseudocode an, der einen Baum $T = (V, E)$ entgegen nimmt und für diesen ein größtes Matching berechnet. Zeigen Sie die Korrektheit Ihres Algorithmus und schätzen Sie die asymptotische Laufzeit möglichst genau ab. **6 Punkte**

Hinweis: Betrachten Sie wie in Teilaufgabe a) die Blätter des Baumes, d. h. die Knoten $v \in V$ mit $\deg v = 1$, und ihre jeweilige (einzige) inzidente Kante.

c) Entwickeln Sie einen effizienten Algorithmus der auf einem Baum mit Kantengewichten ein *gewichtsmaximales Matching* findet. Zeigen Sie die Korrektheit Ihres Algorithmus und schätzen Sie die asymptotische Laufzeit möglichst genau ab. Beachten Sie, dass ein gewichtsmaximales Matching im allgemeinen nicht perfekt ist. **4 Zusatzpunkte**

Aufgabe 2 – LP-Runden

Sei $G = (V, E)$ ein ungerichteter Graph. Eine *Knotenüberdeckung* ist eine Menge $V' \subseteq V$ von Knoten, so dass für jede Kante $e \in E$ mindestens einer der Endknoten in V' enthalten ist. Das Problem eine minimale Knotenüberdeckung (d. h. mit möglichst wenig Knoten) zu finden ist $\mathcal{NP}$-schwer. In der zweiten Vorlesung haben wir diese Modellierung des Problems als ganzzahliges lineares Programm kennengelernt:

$$\begin{array}{rll} \text{Minimiere} & \sum_{v \in V} x_v & \\ \text{Beschränkungen:} & x_u + x_v \geq 1 & \text{für jede Kante } uv \in E \\ & x_v \in \{0, 1\} & \text{für jeden Knoten } v \in V \end{array}$$

CPLEX

a) Implementieren Sie das ganzzahlige lineare Programm aus der Vorlesung in CPLEX. Ihr Modell muss in der Lage sein, beliebige einfache, ungerichtete Graphen aus einer .dat-Datei einzulesen und eine minimale Knotenüberdeckung zu finden.

Entwerfen Sie zunächst ein geeignetes Format, um Graphen in `.dat`-Form zu speichern. Ihr Modell muss dieses Format einlesen und mittels Quantifizierung alle nötigen Constraints erzeugen und so eine optimale Lösung finden können.

Laden Sie bis zur Abgabefrist dieses Blattes eine Modell- (`.mod`) und eine passende Daten-Datei (`.dat`) in WueCampus hoch. **3 Punkte**

Hinweis: Es steht ein weiterer Teil der CPLEX-Einführung im Kursraum zur Verfügung. Wenn Sie Ihren Code testen, versuchen Sie sich an kleinen Graphen ($|V| \leq 10$) und verifizieren Sie die Lösung von Hand.

Wir betrachten nun die zugehörige LP-Relaxierung, d. h. folgendes lineares Programm:

$$\begin{array}{rll} \text{Minimiere} & \sum_{v \in V} x_v & \\ \text{Beschränkungen:} & x_u + x_v \geq 1 & \text{für jede Kante } uv \in E \\ & 0 \leq x_v \leq 1 & \text{für jeden Knoten } v \in V \end{array}$$

b) Wir betrachten eine optimale Lösung für die LP-Relaxierung. Seien dazu die Mengen $U = \left\{v \in V \mid 0 < x_v < \frac{1}{2}\right\}$ und $W = \left\{v \in V \mid \frac{1}{2} < x_v < 1\right\}$.

   Wir nehmen an, dass $|U| + |W| \geq 1$ und $|W| \geq |U|$ gilt. Zeigen Sie, wie sich in $O(n)$ Zeit eine neue optimale Lösung finden lässt, so dass $|U| + |W|$ kleiner wird. **4 Punkte**

   Hinweis: Suchen sie ein geeignetes $\varepsilon > 0$ und ändern sie manche Werte um ε.

   Für den Fall, dass $|U| + |W| \geq 1$ und $|W| < |U|$ ist, lässt sich ebenfalls eine solche Transformation finden; das müssen Sie hier aber nicht tun.

c) Entwickeln Sie einen effizienten Algorithmus, der eine optimale Lösung für die LP-Relaxierung findet, in der als Variablenwerte nur 0, $\frac{1}{2}$ und 1 vorkommen. Zeigen Sie, dass Ihr Algorithmus korrekt ist und schätzen Sie seine Laufzeit scharf ab. **3 Punkte**

d) Entwickeln Sie einen effizienten Algorithmus, der eine 2-Approximation für eine minimale Knotenüberdeckung liefert. **2 Punkte**

   Hinweis: Runden Sie geschickt.

Bitte werfen Sie Ihre Lösungen bis **Dienstag, 28. Mai 2019, 8:30 Uhr** in den Vorlesungs-Briefkasten im Informatik-Gebäude. Geben Sie stets die Namen und Übungsgruppen aller BearbeiterInnen sowie die Übungsgruppe, in der das Blatt zurückgegeben werden soll, an.

Begründen Sie stets Ihre Behauptungen und kommentieren Sie Ihren Pseudocode!

Aufgaben, die mit $\boxed{\text{CPLEX}}$ gekennzeichnet sind, fordern das Erstellen und Lösen von linearen Programmen. Laden Sie Ihren kommentierten Quellcode auf WueCampus hoch und vermerken Sie auf Ihrer Abgabe, welche BearbeiterIn hochgeladen hat.