Lehrstuhl für Informatik I
Algorithmen & Komplexität
Universität Würzburg

Würzburg, den 09. Juni 2020

Prof. Dr. Alexander Wolff
Diana Sieper, B. Sc.

8. Übungsblatt zur Vorlesung
Algorithmische Graphentheorie (Sommer 2020)

Aufgabe 1 – Kleinste Schnitte

Gegeben sei ein ungerichteter zusammenhängender (einfacher) Graph $G = (V, E)$.

a) Einer Ihrer Kommilitonen ist der Meinung, dass die Algorithmen zur Bestimmung eines minimalen Schnitts aus der Vorlesung viel zu kompliziert seien. Er behauptet, dass schon $(\{v\},\ V \setminus \{v\})$ ein minimaler Schnitt ist, falls $v \in V$ ein Knoten kleinsten Grades ist.

   Widerlegen Sie die Behauptung Ihres Kommilitonen. **2 Punkte**

b) Zeigen Sie, dass es höchstens $O(V^2)$ viele verschiedene kleinste Schnitte geben kann. **3 Punkte**

   Hinweis: Berücksichtigen Sie die Erfolgswahrscheinlichkeit des Algorithmus Contract aus der Vorlesung.

Aufgabe 2 – Implementierung von Contract

In der Vorlesung wurde der randomisierte Algorithmus Contract zur Bestimmung eines minimalen Schnitts in einem ungerichteten Graphen $G = (V, E)$ vorgestellt. Bei diesem Algorihmus wurden schrittweise in je $O(E)$ Zeit zwei Knoten einer Kante verschmolzen. Die Kanten wurden gleichverteilt gewählt.

Beschreiben Sie eine Implementierung von Contract, die nur $O(V)$ Zeit pro Iteration und insgesamt $O(V^2)$ Zeit benötigt. Zeigen Sie, dass Ihr Algorithmus die zu kontrahierenden Kanten auch weiterhin gleichverteilt auswählt.

Hinweis: Gehen Sie davon aus, dass das Bestimmen der Anzahl der Kanten zwischen zwei Knoten, das Ziehen einer Zufallszahl beliebiger Größe sowie eine arithmetische Operation über zwei Zahlen stets in konstanter Zeit möglich ist. Beachten Sie weiter, dass das Ziehen einer Zufallszahl nicht mit der zufälligen Wahl einer Kante gleichzusetzen ist. **6 Punkte**

Aufgabe 3 – Randomisierte größte Schnitte

Das Problem MaxCut ist das Problem in einem gegebenen ungerichteten Graphen $G = (V, E)$ die Menge V so in zwei Mengen S und T zu partitionieren, dass die Anzahl der Kanten st, mit $s \in S$ und $t \in T$, maximal ist. Im Gegensatz zu MinCut ist MaxCut $\mathcal{NP}$-vollständig.

Betrachten Sie daher Algorithmus 1.

randMaxCut(Ungerichteter Graph $G = (V, E)$)
  Verwende S und T für die Mengen des Schnitts
  while $S = \emptyset$ **or** $T = \emptyset$ **do**
    $S \leftarrow \emptyset; T \leftarrow \emptyset$
    foreach $v \in V$ **do**
      Mit Wahrscheinlichkeit $1/2$ setze $S \leftarrow S \cup v$
      Sonst setze $T \leftarrow T \cup v$
  return (S, T)

Algorithmus 1 liefert offensichtlich einen zulässigen Schnitt und besitzt eine erwartete Laufzeit von $O(V)$.

a) Ihr Kommilitone aus Aufgabe 1 überlegt, ob seine Idee zumindest einen größten Schnitt in einem Graphen $G = (V, E)$ findet.

   Zeigen oder widerlegen Sie, dass schon $(\{v\},\ V \setminus \{v\})$, ein größter Schnitt ist, falls $v \in V$ ein Knoten höchsten Grades ist. **2 Punkte**

b) Was ist die Wahrscheinlichkeit dafür, dass eine fest gewählte Kante $e \in E$ auf einem mit Algorithmus 1 berechneten Schnitt liegt? **1 Punkt**

c) Was ist die Wahrscheinlichkeit dafür, dass ein fester maximaler Schnitt (S, T) von Algorithmus 1 geliefert wird? **2 Punkte**

   Hinweis: Beachten Sie, dass das Vertauschen von S und T den gleichen Schnitt liefert.

d) Zeigen Sie, dass Algorithmus 1 eine erwartete Güte von $1/2$ besitzt. Zeigen Sie dazu zunächst, dass die erwartete Anzahl an Kanten, die auf dem Schnitt liegen, $|E|/2$ ist. **4 Punkte**

Bitte laden Sie Ihre Lösungen als pdf bis **Dienstag, 16. Juni 2020, 12:00 Uhr** im WueCampus-Kursraum beim entsprechenden Übungsblatt hoch. Geben Sie stets die Namen aller (bis zu 2) BearbeiterInnen 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. Der Quellcode sollte von der selben Person abgegeben werden, die auch die pdf hochgeladen hat. Wenn dies nicht der Fall sein sollte, vermerken Sie auf Ihrer Abgabe, welche BearbeiterIn den Quellcode hochgeladen hat.

Plagiate werden mit 0 Punkten für das ganze Übungsblatt gewertet.