Abteilung Algorithmik
Institut für Betriebssysteme und Rechnerverbund **SoSe 14**
TU Braunschweig

Dr. Christiane Schmidt
Florian Maurer
Arne Schmidt

Netzwerkalgorithmen
Übung 4 vom 16.06.2014

Abgabe der Lösungen bis Mittwoch, den 02.07.14, bis 13:00 Uhr.
Bitte die Blätter vorne deutlich mit eigenem Namen und Gruppennummer versehen!

Aufgabe 1 (Wege und Zuverlässigkeit):
Gegeben sei ein Digraph G mit $s, t \in V(G)$. Jeder Kante $e \in E(G)$ wird eine Zahl $r(e) \in [0, 1]$, ihre Zuverlässigkeit, zugewiesen. Die Zuverlässigkeit eines Weges ist das Produkt der Zuverlässigkeiten seiner Kanten. Gesucht ist der Weg von s nach t mit maximaler Zuverlässigkeit.

a) Zeige, dass man dieses Problem unter Anwendung von Logarithmus auf das Kürzeste-Wege-Problem reduzieren kann.

b) Löse dieses Problem in polynomialer Zeit ohne die Anwendung von Logarithmus.

(8 + 7 Punkte)

Aufgabe 2 (Wege und Schnitte):
Sei G ein ungerichteter Graph mit Gewichten $c : E(G) \to \mathbb{Z}_+$ und zwei Knoten $s, t \in V(G)$, wobei t von s aus erreichbar ist. Für eine Knotenmenge $X \subseteq V(G)$ nennt man die Kantenmenge $\delta(X) = \{\{x, y\} \in E(G), x \in X, y \in V(G) \setminus X\}$ einen Schnitt in G. Falls $s \in X$ und $t \notin X$ gilt, trennt $\delta(X)$ die beiden Knoten s und t.
Zeige, dass die minimale Länge eines s-t Weges gleich der maximalen Anzahl von Schnitten ist, die s und t trennen, so dass jede Kante e in höchstens $c(e)$ solcher Schnitte enthalten ist.
(Tipp: Warum reicht es, einen Graphen mit Einheitsgewichten zu betrachten? Zeige, dass die maximale Anzahl von solchen Schnitten sowohl eine obere als auch eine untere Schranke für die minimale Länge ist. Für den Beweis der unteren Schranke gib eine Menge von Schnitten an, unter der Verwendung eines BFS-Baumes.)

(15 Punkte)

Aufgabe 3 (Knapsack ohne Wiederholung):
In der großen Übung am 16.06. haben wir uns das Rucksackproblem und eine Formulierung als dynamische Programm für den Fall der unbeschränkten Anzahl der einzelnen Objekte angeguckt.

Betrachte das Rucksackproblem ohne Wiederholung, das heißt, jedes Objekt darf nur genau einmal (maximal einmal) gepackt werden.
Die Kenntnis von $K(w - w_i)$, wie sie für den Fall unbeschränkter Anzahl verwendet wurde, hilft hier nicht, da wir nicht wissen, ob Objekt i schon verwendet wurde, um diesen Wert zu erreichen.

- Welcher Wert sollte stattdessen verwendet werden?
- Wie kann man diesen aus kleineren Teilproblemen ableiten?
- Gib einen Dynamisches Programm an, für das Rucksackproblem ohne Wiederholung.

(15 Punkte)

Aufgabe 4 (Algorithmus von Ford und Fulkerson):

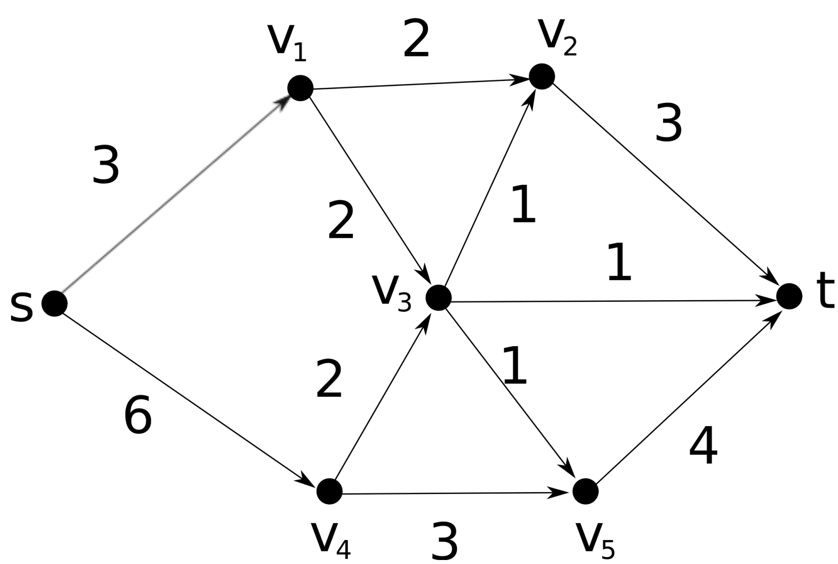

Abbildung 1: Das Netzwerk (G, u, s, t). Die Zahlen an den Kanten sind die Kapazitäten.

Bestimme mit Hilfe des Algorithmus von Ford und Fulkerson einen maximalen s-t-Fluß im Netzwerk (G, u, s, t). Gib jeweils den Residualgraphen an.
Gib außerdem einen minimalen Schnitt an. (15 Punkte)