technische universität
dortmund

Christian Sohler, Melanie Schmidt
Lehrstuhl 2, Fakultät für Informatik
Wintersemester 2009/2010

AuD Übung – Blatt 5

Ausgabedatum: 10.12.2009 — Abgabedatum: 16.12.2009

Aufgabe 5.1: Derandomisierung

In der Vorlesung wurden zwei randomisierte Algorithmen für Max-Sat besprochen: Algorithmus 2.5.4, der eine zufällige Belegung wählt, und Algorithmus 2.5.9 (Max-Sat-RR), der eine Lösung der LP-Relaxierung rundet (außerdem wurde eine Kombination der Algorithmen besprochen). Den ersten Algorithmus haben wir in der Vorlesung derandomisiert, um einen deterministischen Algorithmus zu erhalten, indem wir die Methode der bedingten Wahrscheinlichkeiten angewendet haben. Wende diese Methode auf den Algorithmus Max-Sat-RR an, um auch für diesen Algorithmus eine deterministische Variante zu erhalten.

Aufgabe 5.2: Randomisierte Max-Sat-Algorithmen

Zeige, dass der folgende Algorithmus eine $\frac{1}{2}$-Approximation für Max-Sat ist:

1. Wähle eine beliebige Belegung x der Variablen (z.B. zufällig).
2. Berechne x' als das Komplement von x, d.h. $\forall i = 1, \ldots, n : x_i{}' = 1 \Leftrightarrow x_i = 0$.
3. Berechne, wie viele Klauseln von x und x' erfüllt werden. Sei x^* die Belegung, die mehr Klauseln erfüllt.
4. Gib x^* aus.

Aufgabe 5.3: Randomisiertes Runden für SetCover

Betrachte den Algorithmus für SetCover, der mit randomisiertem Runden funktioniert (Abschnitt 2.3). In einer einzelnen Iteration des Algorithmus wird aus einer fraktionalen Lösung durch Runden eine ganzzahlige Lösung berechnet, die aber nicht unbedingt alle Elemente des Universums überdeckt. Gib eine untere Schranke für den Erwartungswert für die Anzahl überdeckter Elemente an. Es ist keine exakte Rechnung gewünscht (nur eine untere Schranke), und Annahmen aus der Vorlesung dürfen übernommen werden.

Aufgabe 5.4: Metrische Standortbestimmung

Betrachte das folgende Beispiel: Gegeben sind n Städte $s_1, \ldots, s_n$ und zwei Anlagen a_1 und a_2. Die Distanz zwischen s_1 und Anlage a_1 ist 1 ($c_{11} = 1$), aber die Distanz zwischen a_1 und allen anderen Städten ist 3 (also $c_{1i} = 3$ für $i = 2, \ldots, n$). Die Distanz zwischen jeder Stadt s_i und a_2 ist 1, d.h. $c_{2i} = 1$ für $i = 1, \ldots, n$. Die Öffnungskosten betragen $f_1 = \epsilon$ und $f_2 = (n+1)\epsilon$ für ein kleines $\epsilon > 0$. Was ist die optimale Lösung? Welche Lösung berechnet der Approximationsalgorithmus aus der Vorlesung (schlechtestenfalls), und welche Güte wird erreicht?