Approximationsalgorithmen

Prof. Dr. Klaus Meer, Ameen Naif

Aufgabenblatt 5
Version 23.01.2018

Aufgabe 1.

(Knapsack) Betrachten wir das *Fractional Knapsack* Problem: Gegeben sei eine Menge S mit n Elementen und Zahlen $c, a \in \mathbb{N}^S$ und die Bingröße $B \in \mathbb{N}$. Finde *reelle* Zahlen $x \in [0,1]^S$ mit

$$\sum_{i \in S} a_i \cdot x_i \leq B \text{ and } \sum_{i \in S} c_i \cdot x_i \text{ ist maximal.}$$

Beweisen Sie, dass das Fractional Knapsack Problem in polynomieller Zeit exakt lösbar ist.

Tip: Ordne die Elemente von S nach c_i/a_i absteigend. Eine Belegung der x_i entsteht, wenn vom ersten Element so viel wie möglich genommen wird, dann vom zweiten in den Restplatz und so weiter. Wie sieht diese Belegung aus? Zeigen Sie, dass sie zulässig und optimal ist.

Aufgabe 2.

(MAX-CUT) Konstruieren Sie Worst-Case Instanzen für den vorgestellten lokalen Such-Algorithmus für MAX-CUT. Zeigen Sie damit, dass die untere Schranke für seine Performance ebenfalls 2 ist.

Aufgabe 3.

(MAX-CUT) In der Lokalen Suche für MAX-CUT soll nun anstelle eines beliebigen Knotens in jedem Schritt der Knoten mit der global besten Verbesserung getauscht werden. Welche untere Schranke für die Performance können Sie für diese Variante beweisen? (Es ginge mit Faktor 2, allerdings mit deutlich komplizierteren Graphen.)

Aufgabe 4.

(Lokale Suche, polynomiell beschränkt) Sei Π ein kombinatorisches Maximierungsproblem mit ganzzahliger Zielfunktion, zudem sei das Optimum durch eine Zahl aus der Eingabe beschränkt.

(a) Zeigen Sie, dass die auf lokaler Suche basierenden Verfahren für dieses Problem pseudo-polynomialzeit Algorithmen sind.

(b) Wie verhält sich dies bei polynomiell-beschränkten Optimierungsproblemen?