

Übungen zur Vorlesung
Komplexitätstheorie
SS 2009
Blatt 12

Betrachte folgendes Optimierungsproblem:

MAX-CUT

Eingabe: Graph $G = (V, E)$ und Gewichte $w(e) \in \mathbb{N}^+$ für jede Kante $e \in E$

Lösung: Eine Partition von V in V_1 und V_2

Ziel: Maximiere $\sum_{e \in E \cap (V_1 \times V_2)} w(e)$

mit der *move*-Nachbarschaftsfunktion: (V'_1, V'_2) ist *move*-Nachbar von (V_1, V_2) , wenn (V'_1, V'_2) aus (V_1, V_2) genau durch Entfernen eines Knoten aus einer der Teilmengen und Hinzufügen dieses Knotens zu der anderen Teilmenge entsteht.

und

STABLECONFIGURATION

Eingabe: Ein Hopfield Netzwerk gegeben durch Graph $G = (V, E)$, Gewichte $w(e) \in \mathbb{Z}$ für jede Kante $e \in E$ und Schwellwerte $T(v) \in \mathbb{Z}$ für jeden Knoten $v \in V$.

Lösung: Konfiguration $f : V \rightarrow \{-1, 1\}$

Ziel: Maximiere $\sum_{\{u,v\} \in E} w(\{u,v\}) \cdot f(u) \cdot f(v) - \sum_{v \in V} T(v) \cdot f(v)$

mit der *flip*-Nachbarschaftsfunktion: f' ist *flip*-Nachbar von f , wenn f' aus f genau Umkehren des Wertes von f für ein v (d.h. $f(v) \rightsquigarrow -f(v)$) entsteht.

Aufgabe 12.1

Zeige: MAX-CUT/move \propto_{tight} MAX-2SAT/flip

Aufgabe 12.2

Zeige: MAX-CUT/move \propto_{tight} UGP/swap

Aufgabe 12.3

Zeige: MAX-CUT/move \propto_{tight} STABLECONFIGURATION/flip

Aufgabe 12.4

Betrachte MULTIPROCESSORSCHEDULING mit der *move*-Nachbarschaftsfunktion und folgender *Pivoting-Rule*: Wähle einen beliebigen Job einer kritischen Maschine und verschiebe (*move*) diesen auf eine Maschine mit minimalem Makespan.

- a) Zeige, dass zwischen zwei *move*-Operationen die den selben Job i betreffen, ein Job j zum letzten Mal verschoben (*move*) wird. (Es kann auch $i = j$ sein)
- b) Benutze a) um eine obere Schranke für die Anzahl der Iterationen herzuleiten, die *iterative improvement* benötigt, um in ein lokales Optimum zu gelangen.