Effiziente Algorithmen

Übungsblatt 7

Prof. Dr. Woeginger, PD Dr. Unger, Prof. Dr. Rossmanith
Dennis Fischer
Lehrstuhl für Informatik 1
RWTH Aachen

WS 18/19
29. November
Abgabe: 6. Dezember 18:00

- Die Übungsblätter sollen in Gruppen von 3-5 Studierenden abgegeben werden.
- Die abgegebenen Lösungen mit Namen und Matrikelnummern aller Teammitglieder und der Übungsgruppe beschriften.
- Um zur Klausur zugelassen zu werden müssen 50% aller möglichen Übungspunkte erreicht werden.

Aufgabe 1 (3 Punkte)

Ein Student möchte für eine Prüfung lernen. Aus den Prüfungsprotokollen anderer Studenten meint er herausgelesen zu haben, dass im Wesentlichen folgende vier Faktoren ausschlaggebend für den Erfolg in einer Prüfung sind: Faktenwissen, Verständnis, Beispielkenntnisse und Transferfähigkeit.

In aufwendigen Selbstversuchen untersucht er, welchen Nutzen die Nachbereitung der Vorlesung anhand der Folien und durch Beschäftigung mit den Übungsaufgaben hat. Er stellt fest, dass sowohl Folien als auch Übung zu allen der vier Faktoren beitragen, die Übung jedoch zum Faktenwissen am wenigsten. Kurioserweise eignen sich die Folien zur Erlangung von Transferfähigkeit genauso wenig wie die Übung zum Anhäufen von Faktenwissen. Er normiert diesen Wert auf 1.

Jeder Arbeitstag Auseinandersetzung mit den Folien bringen ihm auf seiner Skala genau 1 Einheit Transferfähigkeit, 5 Einheiten Beispielkenntnisse, 3 Einheiten Faktenwissen, und 3 Einheiten Verständnis. Die Beschäftigung mit den Übungsaufgaben dagegen ergibt pro Arbeitstag 2 Einheiten Transferfähigkeit, 6 Einheiten Beispielkenntnisse, 1 Einheit Faktenwissen und 2 Einheiten Verständnis.

Der Student veranschlagt, dass er für eine 4.0 in der Prüfung mindestens 10 Einheiten Transferfähigkeit, 30 Einheiten Beispielkenntnisse, 12 Einheiten Faktenwissen und 18 Einheiten Verständnis erlangt haben muss. Das Durcharbeiten der Folien bereitet ihm aufgrund des trockenen Stoffes jedoch pro Stunde doppelt so viele Qualen wie das Bearbeiten von Übungen. Ziel für den Studenten ist es nun, eine 4.0 unter minimalen Qualen zu erlangen.

Formuliere das Problem als LP und finde durch geometrische Lösung des linearen Programms einen optimalen Arbeitsplan zum Erreichen seines Ziels.

Aufgabe 2 (5 Punkte)

Im folgenden betrachten wir das Vertex Cover Problem:

Vertex Cover

Eingabe: Ein ungerichteter Graph $G = (V, E)$, eine Zahl $k \in \mathbb{N}$.

Frage: Gibt es eine Menge $V' \subseteq V$ mit $|V'| \leq k$ so dass für alle $\{v, w\} \in E$ gilt, das $v \in V'$ oder $w \in V'$.

Die Entscheidungsvariante des Vertex Cover Problems ist NP-vollständig. Um das Problem dennoch zu lösen bietet es sich an nicht nach der optimalen Lösung zu fragen, sondern nach einer Lösung, die beweisbar höchstens einen konstanten Faktor c von der optimal Lösung entfernt ist. Algorithmen die solche Lösungen berechnen heißen *c-Approximationsalgorithmus*. Wir wollen einen 2-Approximationsalgorithmus für das Vertex Cover Problem mit Hilfe von linearer Programmierung finden.

(a) Formuliere das Vertex Cover Problem als Integer Linear Program (ILP) und gib auch die relaxierte Version des ILPs an.

(b) Gib einen Algorithmus an, der mit Hilfe einer Lösung des relaxierten LPs aus (a) eine Lösung für das Vertex Cover Problem berechnet, welche höchstens doppelt so groß ist wie eine optimale (kleinste) Lösung und zeige dessen Korrektheit.

(c) Formuliere das Independent Set Problem als Integer Linear Program (ILP) und gib auch die relaxierte Version des ILPs an.

Independent Set

Eingabe: Ein ungerichteter Graph $G = (V, E)$, eine Zahl $k \in \mathbb{N}$.

Frage: Gibt es eine Menge $V' \subseteq V$ mit $|V'| \geq k$ so dass für alle $\{v, w\} \in E$ gilt, das $v \in V' \implies w \notin V'$.

(d) Zeige, dass sich die Relaxierung des Independent Set ILPs nicht eignet, um einen Approximationsalgorithmus zu entwerfen. Zeige dazu, dass

$$\max_{I \in \mathcal{I}} \frac{\mathrm{opt}(\mathrm{LP}_{IS}(I))}{\mathrm{opt}(\mathrm{ILP}_{IS}(I))} = \infty$$

die *integrality gap*, beliebig groß sein kann.

Aufgabe 3 (4 Punkte)

Betrachte das Graphisomorphieproblem:

Graphisomorphie

Eingabe: Zwei einfache, ungerichtete Graphen G_A, G_B mit $|V(G_A)| = |V(G_B)|$.

Frage: Ist $G_A \cong G_B$? D.h. gibt es eine bijektive Abbildung $f\colon V(G_A) \to V(G_B)$ mit

$$\{v, w\} \in E(G_A) \iff \{f(v), f(w)\} \in E(G_B).$$

(a) Formuliere das Graphisomorphieproblem als ILP. Die Graphen sind als Adjazenzmatrizen der Größe n^2 gegeben.
Hinweis: Das ILP enthält n^2 Variablen und keine Zielfunktion.

(b) Betrachte die Relaxierung des ILPs aus (a). Wir schreiben $G_1 \simeq G_2$, falls die Relaxierung eine Lösung zulässt. Finde Graphen G_1, G_2 mit $G_1 \simeq G_2$ aber $G_1 \ncong G_2$.

Aufgabe 4 (4 Punkte)

LPs und ILPs haben viele praktische Anwendungen. Oftmals scheinen rein lineare Bedingungen auf den ersten Blick nicht auszureichen. In dieser Aufgabe wollen wir einige Tricks zum Aufstellen von LPs und ILPs entwickeln.

(a) Seien $f_1(x_1, \ldots, x_n), f_2(x_1, \ldots, x_n), \ldots, f_k(x_1, \ldots, x_n)$ lineare Terme. Angenommen als Zielfunktion soll $\max_{i=1}^{k} \{f_i(x_1, \ldots, x_n)\}$ minimiert werden. Wie lässt sich das als lineares Programm formulieren?

(b) Wie lässt sich der Betrag einer Variablen in der Zielfunktion minimieren?

(c) Angenommen wir wollen allgemeinere nichtlineare Zielfunktionen in einem ILP minimieren:

$$\begin{array}{ll} \text{minimize} & f(A) \\ \text{subject to} & \ldots \\ & \ldots \end{array}$$

Wobei f eine nichtlineare Funktion ist, A ein beliebiger Term. Das ist im Allgemeinen nicht mittels eines ILP machbar. Was ist, wenn A nur eine kleine (konstante) Zahl von verschiedenen Werten $a \in \mathcal{A}$ annehmen kann? Gib ein Verfahren an solche nichtlineare Zielfunktionen umzusetzen.

Abgabefrist: Die Lösungen müssen bis zum **6. Dezember 18:00** in der Vorlesung oder im Abgabekasten vor dem Lehrstuhl i1 abgegeben werden.