

Lineare und ganzzahlige Optimierung

Übungszettel 9 — Programmieraufgabe

Implementieren Sie den Branch-&-Bound-Algorithmus 10 für das MINIMUM-WEIGHT-VERTEX-COVER-PROBLEM (gewichtete Knotenüberdeckungsproblem). Als Ergänzung zum MINIMUM-VERTEX-COVER-PROBLEM aus Beispiel 1.6 sind jetzt zusätzlich Knotengewichte $w : V(G) \rightarrow \mathbb{N}$ gegeben.

Das Problem ist dann durch folgendes IP gegeben:

$$\min \sum_{v \in V(G)} w(v)x_v \quad (1)$$

$$s.d. \quad x_v + x_w \geq 1 \quad \forall \{v, w\} \in E(G) \quad (2)$$

$$x_v \in \{0, 1\} \quad \forall v \in V(G) \quad (3)$$

Als LP-Solver sollten Sie einen der für akademische Zwecke frei verfügbaren Programme **QSopt** (und hiermit wird Ihr Code getestet), CPLEX oder Gurobi verwenden. Die externen LP-Löser müssen durch das auf den Webseiten vorhandene Interface in `lp.h` angesteuert werden. Dazu stehen jeweils Implementierungen zur Verfügung. Um die softwaretechnische Hürde zu reduzieren, finden Sie auf den Webseiten zur Übung ein vorgefertigtes Programm, das

1. eine Instanz einliest,
2. die LP-Relaxierung des obigen IPs über das Interface in `lp.h` aufbaut,
3. löst und den Lösungsvektor x ausgibt

(siehe <http://www.or.uni-bonn.de/held/lpip/1112/mvc.zip>). Die ZIP-Datei enthält auch Testinstanzen. **Bitte lesen Sie auch die beigefügte README Datei für weitere Informationen!** Das Programm ist unter Linux oder Windows/Cygwin kompilierbar. Dazu muss im entpackten Verzeichnis 'mss' in einem Terminal 'make' ausgeführt werden.

Es ist zu beachten, dass die Matrix A in modernen LP-Lösern immer dünn gespeichert wird, da i.d.R. die meisten Koeffizienten gleich Null sind. Daher werden, wie in `lp.h` kommentiert, neue Nebenbedingungen/Zeilen immer über die Nicht-Null-Koeffizienten definiert. Entsprechende Funktionen sind in `lp.h` kommentiert und die Anwendung in `mss.c` ersichtlich.

Um die Ganzzahligkeitslücke zu reduzieren, sollten Sie zunächst das Root-LP (LP_0) durch das **Hinzufügen von Klickenrestriktionen** verschärfen. Überlegen Sie sich hierzu, wie viele Knoten aus einer Clique in einer Knotenüberdeckung mindestens enthalten sein müssen.

Um Cliques zu bestimmen, können Sie einen einfachen Greedy-Algorithmus implementieren, der, startend mit $C = \{v\}$ für ein $v \in V(G)$, solange wie möglich Knoten $w \in V(G) \setminus C, C \subseteq \delta(w)$, mit einem großen Wert x_w zu C hinzufügt.

Diesen sollten Sie zu Beginn iterativ mit unterschiedlichen $v \in V$ starten lassen, bis alle Knoten Teil einer solchen (inklusions-) maximalen Clique sind. Anschließend brauchen Sie keine Cliquerestriktionen mehr hinzufügen, auch wenn das inkrementelle Hinzufügen weiterer verletzter Restriktionen ggf. von Vorteil wäre.

Die initiale obere Schranke im Branch-&-Bound sollte durch Rundung der fraktionalen Lösung bestimmt werden, wie in der 1. Anwesenheitsübung durchgenommen. Der Algorithmus sollte die Werte des Root-LPs ohne Cliquerestriktionen, des Root-LPs mit Cliquerestriktionen, den Wert der initialen Rundungsheuristik und das minimale Gewicht einer Knotenüberdeckung S sowie die Indizes der Knoten in S in der Konsole ausgeben. (15 Punkte)

Abgabetermin ist Dienstag, der 17.01.2012, vor der Vorlesung (12:15).