Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Fachbereich Informatik
Sommersemester 2024
Prof. Dr. Peter Becker

Kombinatorische Optimierung

Lösungen zu Aufgabenblatt 3

Aufgabe 1 (Klee-Minty-Polytop)

Ermitteln Sie alle Ecken des 3-dimensionalen Klee-Minty-Polytops.

Lösung: Wir nutzen Fakt 2.7:

Für die Zielfunktion

$$\max 2^{d-1}x_1 + 2^{d-2}x_2 + \cdots + 2x_{d-1} + x_d$$

durchläuft der Simplexalgorithus mit der Pivotregel von Danzig alle 2^d Ecken des Klee-Minty-Polytops.

	x_1	x_2	x_3	x_4	x_5	x_6	
x_4	1	0	0	1	0	0	5
x_5	4	1	0	0	1	0	25
x_6	8	4	1	0	0	1	125
	-4	-2	-1	0	0	0	0

1. Ecke : $\begin{pmatrix}0\\0\\0\end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	1	0	0	1	0	0	5
x_5	0	1	0	-4	1	0	5
x_6	0	4	1	-8	0	1	85
	0	-2	-1	4	0	0	20

2. Ecke : $\begin{pmatrix}5\\0\\0\end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	1	0	0	1	0	0	5
x_2	0	1	0	-4	1	0	5
x_6	0	0	1	8	-4	1	65
	0	0	-1	-4	2	0	30

3. Ecke : $\begin{pmatrix}5\\5\\0\end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_4	1	0	0	1	0	0	5
x_2	4	1	0	0	1	0	25
x_6	-8	0	1	0	-4	1	25
	4	0	-1	0	2	0	50

4. Ecke : $\begin{pmatrix}0\\25\\0\end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_4	1	0	0	1	0	0	5
x_2	4	1	0	0	1	0	25
x_3	-8	0	1	0	-4	1	25
	-4	0	0	0	-2	0	75

5. Ecke : $\begin{pmatrix}0\\25\\25\end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	1	0	0	1	0	0	5
x_2	0	1	0	-4	1	0	5
x_3	0	0	1	8	-4	1	65
	0	0	0	4	-2	1	95

6. Ecke : $\begin{pmatrix}5\\5\\65\end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	1	0	0	1	0	0	5
x_5	0	1	0	−4	1	0	5
x_3	0	4	1	−8	0	1	85
	0	2	0	−4	0	1	105

, 7. Ecke : $\begin{pmatrix} 5 \\ 0 \\ 85 \end{pmatrix}$

	x_1	x_2	x_3	x_4	x_5	x_6	
x_4	1	0	0	1	0	0	5
x_5	4	1	0	0	1	0	25
x_3	8	4	1	0	0	1	125
	4	2	0	0	0	1	125

, 8. Ecke : $\begin{pmatrix} 0 \\ 0 \\ 125 \end{pmatrix}$

Aufgabe 2 (Erkennung eines bipartiten Graphen)

Schreiben Sie eine Java-Methode, die erkennt, ob ein Graph $G = (V, E)$ bipartit ist.

Lösung: siehe Homepage

Aufgabe 3 (Weighted/Bipartite Vertex Cover)

Gegeben sei ein Graph $G = (V, E)$ und eine Kostenfunktion $c : V \to \mathbb{N}$. Eine *Knotenüberdeckung* von G ist eine Knotenmenge $U \subseteq V$, so dass für jede Kante $e = \{v, w\} \in E$ mindestens einer der Knoten v, w in U enthalten ist. Weiterhin ist

$$c(U) = \sum_{v \in U} c(v)$$

das Gewicht von U.

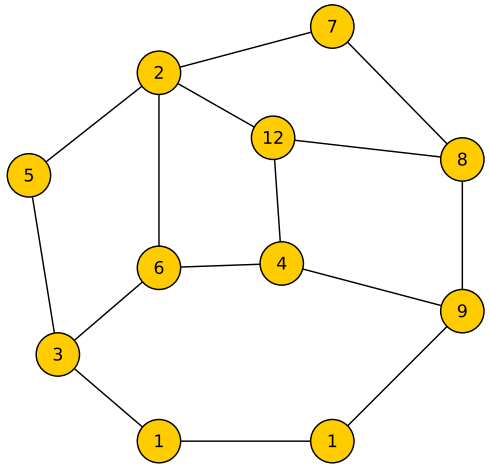
(a) Formulieren Sie das Problem, eine Knotenüberdeckung mit minimalem Gewicht zu bestimmen, als ganzzahliges LP.

(b) Gegeben sei zusätzlich eine Konstante $C \in \mathbb{N}$. Wir betrachten das folgende Entscheidungsproblem:

> Existiert für einen bipartiten Graphen $G = (V, E)$ eine Knotenüberdeckung U mit einem Gewicht $c(U) \leq C$?

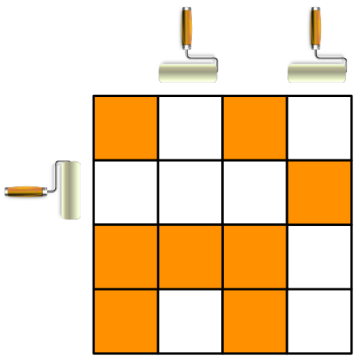
Ist dieses Entscheidungsproblem in $\mathcal{P}$? Begründen Sie Ihre Antwort.

(c) Berechnen Sie mit dem GLPK eine Knotenüberdeckung mit minimalem Gewicht für den folgenden Graphen (die Nummern sind die Knotengewichte):



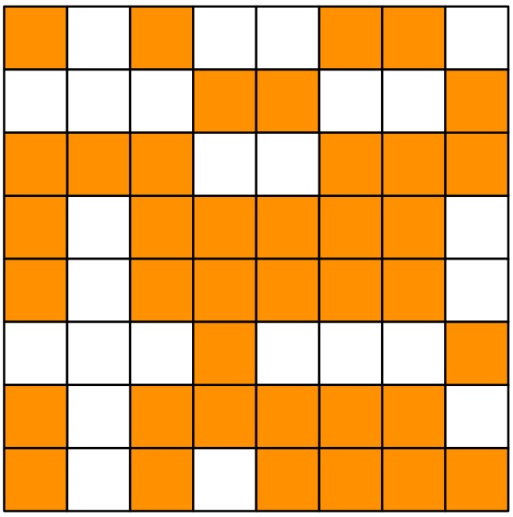
(d) Gegeben sei ein quadratisches Gitter mit weißen und orangen Feldern. Wir möchten eine möglichst kleine Menge U (also $\min |U|$) von Zeilen und Spalten finden, so dass das Gitter vollständig orange wird, wenn wir alle Zeilen und Spalten aus U orange färben.

Die folgende Graphik zeigt eine Probleminstanz mit einer optimalen Lösung ($|U| = 3$). Wenn wir die zweite und vierte Spalte sowie die zweite Zeile färben, ist das Gitter komplett orange.



Zeigen Sie, dass sich dieses Problem als Knotenüberdeckungsproblem in einem bipartiten Graphen formulieren lässt.

(e) Berechnen Sie mit dem GLPK eine optimale Lösung für die folgende Probleminstanz:



(f) Auf der Homepage der Vorlesung finden Sie die LP-Modelle `vc1.lp` und `vc2.lp` für zwei Instanzen des Knotenüberdeckungsproblems. Beide Graphen haben jeweils 200 Knoten und 1000 Kanten und beide Graphen sind nicht bipartit.

Sowohl das GLPK als auch Gurobi können in weniger als einer Zehntelsekunde für `vc1.lp` eine optimale Lösung ermitteln (Gurobi benötigt dafür nur 236 Simplex-Iterationen). Demgegenüber braucht Gurobi für `vc2.lp` fast eine Minute zur Bestimmung der optimalen Lösung und führt dabei 9.883.905 Simplex-Iterationen aus. Das GLPK ist sogar nicht in der Lage, für dieses Problem in akzeptabler Zeit eine optimale Lösung zu bestimmen.

Wie erklären Sie sich diese gravierenden Unterschiede trotz gleicher Problemgröße?

Lösung:

(a) Es sei $V = \{v_1, \ldots, v_n\}$ die Knotenmenge von G. Wir definieren für jeden Knoten v_i eine Variable x_i mit

$$x_i = \begin{cases} 1 & v_i \in U \\ 0 & v_i \notin U \end{cases}$$

Weiterhin sei $c_i = c(v_i)$. Damit lautet die Zielfunktion des LP:

$$\min \sum_{i=1}^{n} c_i x_i.$$

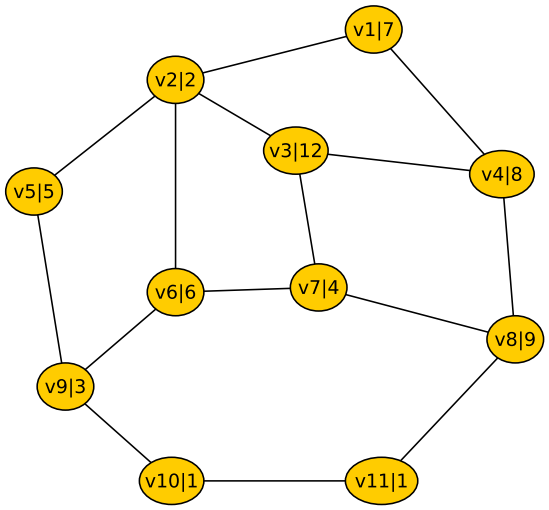
Für die Nebenbedingungen erhalten wir

$$x_i + x_j \geq 1 \quad \text{für alle } e = \{v_i, v_j\} \in E$$

sowie $x_i \in \{0, 1\}$ für $i = 1, \ldots, n$.

(b) Die Koeffizientenmatrix des LP aus (a) ist die transponierte Inzidenzmatrix von G. Wenn G bipartit ist, ist diese Koeffizientenmatrix total unimodular und wir können das Problem als LP in polynomieller Zeit lösen.

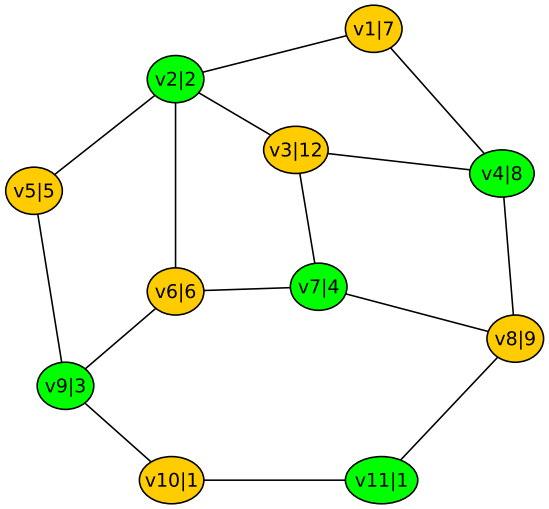
(c) Wir sollten uns noch Bezeichnungen für die Knoten ausdenken.



Da der Graph bipartit ist, benötigen wir keine Einschränkung der Variablen auf $\{0, 1\}$.

LP siehe Homepage.

Optimale Lösung:



Die Knoten der optimalen Knotenüberdeckung sind grün. Optimaler Zielfunktionswert ist 18.

(d) Bezeichne r_i die i-te Zeile und c_j die j-te Spalte. Es sei $G = (V, E)$ mit

- $V = \{r_1, \ldots, r_n\} + \{c_1, \ldots, c_n\}$ und
- $E = \{\{r_i, c_j\} |$ Position (i, j) ist weiß$\}$

ein bipartiter Graph.

Wir zeigen, dass eine Knotenüberdeckung äquivalent zu einer Färbung aller weißen Felder ist.

Es sei U eine Knotenüberdeckung von G. Da es für jedes weiße Feld (i, j) eine Kante $\{r_i, c_j\}$ gibt, muss U den Knoten r_i oder den Knoten c_j enthalten. Wenn wir dementsprechend die i-te Zeile oder j-te Spalte färben, wird somit auch das weiße Feld (i, j) gefärbt.

Es sei (i, j) ein weißes Feld. Dann muss eine Färbung die i-te Zeile oder die j-te Spalte enthalten. Da jedes weiße Feld (i, j) einer Kante $\{r_i, c_j\}$ in G entspricht, bildet somit die Menge der gefärbten Zeilen und Spalten eine Knotenüberdeckung U.

(e) LP siehe Homepage.

Eine optimale Lösung hat den Zielfunktionswert 6 wobei die Zeilen 2 und 6 sowie die Spalten 2, 4, 5 und 8 gefärbt werden.

(f) Es sind zwar beide Graphen nicht bipartit, aber der Graph des LP-Modells `vc1.lp` liegt erheblich näher an einem bipartiten Graphen als bei `vc2.lp`.

Der Graph von `vc1.lp` wurde aus einem bipartiten Graphen erzeugt, indem eine einzige zusätzliche Kante hinzugefügt wurde, die die Bipartitheit zerstört. Somit ist dieser Graph “fast” bipartit.

Demgegenüber wurde der Graph von `vc2.lp` zufällig erzeugt.