Lehrstuhl für Informatik I
Algorithmen & Komplexität
Universität Würzburg

Würzburg, den 16. Juni 2020

Prof. Dr. Alexander Wolff
Diana Sieper, B. Sc.

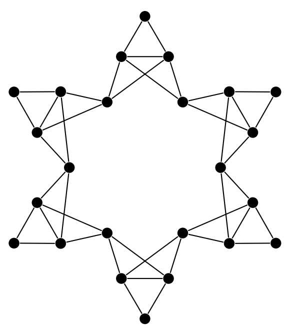
9. Übungsblatt zur Vorlesung Algorithmische Graphentheorie (Sommer 2020)

Aufgabe 1 – Dreifärbbarkeit

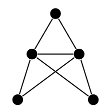
Als *Dreifärbung* eines ungerichteten Graphen $G = (V, E)$ bezeichnen wir eine Funktion $f\colon V \to \{r, g, b\}$, bei der für jede Kante $uv \in E$ die Bedingung $f(u) \neq f(v)$ erfüllt ist, d. h. zwei benachbarte Knoten erhalten niemals die selbe Farbe. Das Dreifärbbarkeitsproblem 3COL besteht darin zu entscheiden, ob ein gegebener Graph *dreifärbbar* ist, d. h. ob es eine Dreifärbung des Graphen gibt. Dieses Problem ist $\mathcal{NP}$-vollständig.

a) Finden Sie eine Dreifärbung des Sterngraphen aus Abbildung 1a und zeigen Sie, dass in jeder solchen Dreifärbung alle Knoten mit Grad 2 (die Zacken des Sterns) die selbe Farbe haben. **2 Punkte**

b) Der Sterngraph aus Teilaufgabe a) fügt sich aus sechs Exemplaren des Turmgraphen aus Abbildung 1b zusammen. Zeigen Sie, dass auch Sterne, die aus mehr oder weniger als sechs Türmen bestehen, dreifärbbar sind, wobei wiederum alle Knoten mit Grad 2 dieselbe Farbe haben müssen. **2 Punkte**



(A) Sterngraph



(B) Turmgraph

ABBILDUNG 1: Graphen für Aufgabe 1.

c) Das Problem 3COL4 besteht darin zu entscheiden, ob ein gegebener Graph mit Maximalgrad 4 dreifärbbar ist. Zeigen Sie, dass schon 3COL4 $\mathcal{NP}$-schwer ist, indem Sie eine Polynomialzeit-Reduktion von 3COL auf 3COL4 angeben.

Geben Sie also ein Verfahren an, mit der man für einen beliebigen Graphen entscheiden kann, ob er dreifärbbar ist, indem Sie einen (hypothetischen) Test `Test3COL4` auf Dreifärbbarkeit von Graphen mit Maximalgrad 4 nutzen. Zeigen Sie, dass Ihr Verfahren korrekt ist. **4 Punkte**

Hinweis: Verwenden Sie Teilaufgabe b)!

Aufgabe 2 – Chromatische Zahl

Sei $G = (V, E)$ ein ungerichteter Graph. Die kleinste Zahl k, die eine k-Färbung von G erlaubt, heißt *chromatische Zahl* $\chi(G)$.

a) Sei $\Delta(G) := \max_{v \in V} \deg v$ der maximale Knotengrad in G. Zeigen Sie, dass für die chromatische Zahl von G gilt, dass $\chi(G) \leq \Delta(G) + 1$. **2 Punkte**

b) Zeigen Sie, dass die Ungleichung aus Teilaufgabe a) scharf ist. Geben Sie dazu für jedes $n \in \mathbb{N}$ und jedes $\Delta < n$ einen Graphen $G_{n,\Delta}$ an, der n Knoten besitzt, Maximalgrad Δ aufweist und $\chi(G_{n,\Delta}) = \Delta + 1$ erfüllt. **2 Punkte**

c) Erstellen Sie ein ILP, welches $\chi(G)$ für einen gegebenen Graphen $G = (V, E)$ ermittelt. **2 Punkte**

Hinweis: Sie dürfen in den Constraints Ungleichungen der Form $\ldots \neq \ldots$ nutzen. Verwenden Sie Ganzzahlen für die Farben.

CPLEX d) Implementieren Sie in CPLEX ein ganzzahliges lineares Programm, das $\chi(G)$ für einen gegebenen Graphen $G = (V, E)$ ermittelt. Ihr Modell muss in der Lage sein, ungerichtete Graphen beliebiger Größe einzulesen. Wählen Sie dazu eine geeignete Darstellung des Graphen in ihrer Daten-Datei. Laden Sie Ihre Lösung (*Modell*- und passende *Daten*-Datei) auf WueCampus hoch. **3 Zusatzpunkte**

e) Erstellen Sie nun ein ILP, welches $\chi(G)$ für einen gegebenen Graphen $G = (V, E)$ ermittelt, und in dem alle bis auf höchstens eine der ganzzahligen Variablen $\in \{0, 1\}$ sind. **5 Zusatzpunkte**

Hinweis: Nutzen Sie als die eine ganzzahlige Variable, die nicht in $\{0, 1\}$ liegen muss, die Gesamtzahl an Farben.

CPLEX f) Implementieren Sie nun wieder ein ganzzahliges Programm in CPLEX, welches $\chi(G)$ für einen gegebenen Graphen $G = (V, E)$ ermittelt, und in dem alle bis auf höchstens eine der ganzzahligen Variablen $\in \{0, 1\}$ sind. Vergleichen Sie die Laufzeiten dieser Implementierung und der Implementierung aus Teilaufgabe (d) auf einer aussagekräftigen Anzahl an (zufällig generierten) Graphen. Laden Sie Ihre Lösung (*Modell*- und passende *Daten*-Dateien) auf WueCampus hoch.

5 Zusatzpunkte

Aufgabe 3 – Perfektes Eliminationsschema

In der Vorlesung wurde bewiesen, dass die Graphen mit perfektem Eliminationsschema gerade die chordalen Graphen sind. Für diese Aufgabe ist es essentiell, dass Sie den Beweis des Satzes von Dirac (der nicht in dem Vorlesungsvideo behandelt wurde) anschauen und verstehen. Zeigen Sie, wie man ein perfektes Eliminationsschema eines chordalen Graphen mit n Knoten in

a) $O(n^4)$ Zeit bestimmen kann. **2 Punkte**

b) $o(n^4)$ Zeit bestimmen kann. **4 Punkte**

Tipp für b): Setzen Sie den Beweis von Dirac durch einen rekursiven Algorithmus um. Der Beweis geht von einer Menge $U \subset V$ *maximaler Kardinalität* aus, für die gelten muss, dass $G[U]$ zusammenhängend ist und $U \cup N(U) \neq V$ gilt. Argumentieren Sie, dass man die Maximalität von U durch ein schwächeres Kriterium ersetzen kann, das sich leichter überprüfen lässt.

Bitte laden Sie Ihre Lösungen als pdf bis **Dienstag, 23. Juni 2020, 12:00 Uhr** im WueCampus-Kursraum beim entsprechenden Übungsblatt hoch. Geben Sie stets die Namen aller (bis zu 2) BearbeiterInnen an.

Begründen Sie stets Ihre Behauptungen und kommentieren Sie Ihren Pseudocode!

Aufgaben, die mit $\boxed{\text{CPLEX}}$ gekennzeichnet sind, fordern das Erstellen und Lösen von linearen Programmen. Laden Sie Ihren kommentierten Quellcode auf WueCampus hoch. Der Quellcode sollte von der selben Person abgegeben werden, die auch die pdf hochgeladen hat. Wenn dies nicht der Fall sein sollte, vermerken Sie auf Ihrer Abgabe, welche BearbeiterIn den Quellcode hochgeladen hat.

Plagiate werden mit 0 Punkten für das ganze Übungsblatt gewertet.