Grundlagen der theoretischen Informatik

Vorlesung 7:

Färbungen, Flüsse

Knotenfärbung

Definition

Sei G ein Graph und $k \geq 1$ eine ganze Zahl. Eine *Färbung* von G mit k Farben ordnet jedem Knoten von G eine von k Farben zu, so dass benachbarten Knoten nicht dieselbe Farbe zugeordnet wird. G ist mit k Farben färbbar, wenn es eine solche Färbung gibt. Die Knoten von G, denen bei einer gegebenen Färbung dieselbe Farbe zugeordnet wird, bilden eine Farbenklasse.

Behauptung. Sei G ein Graph mit n Knoten und mindestens einer Kante. Dann gelten folgende Eigenschaften:

- G ist mit 1 Farbe nicht färbbar.
- G ist mit n Farben färbbar.
- Wenn G mit k_1 Farben färbbar ist und $k_2 > k_1$, dann ist G auch mit k_2 Farben färbbar.
- Wenn G mit k_1 Farben nicht färbbar ist und $k_2 < k_1$, dann ist G mit k_2 Farben auch nicht färbbar.

Definition

Die *chromatische Zahl* (kromatikus szám) eines Graphen G ist die kleinste Zahl k, so dass G mit k Farben färbbar ist. Die chromatische Zahl von G wird mit $\chi(G)$ bezeichnet.

Definition

Im Graphen G ist der Teilgraph H eine *Clique (klikk)*, wenn H ein vollständiger Graph ist. Die *Cliquezahl* von G ist die maximale Anzahl der Knoten in einer Clique von G. Die Cliquezahl von G wird mit $\omega(G)$ bezeichnet.

Satz

In jedem Graphen G gilt $\omega(G) \leq \chi(G)$.

Beweis. Sei H eine Clique von G mit $\chi(G)$ Knoten. Da H ein vollständiger Graph ist, muss jedem Knoten von H eine andere Farbe zugeordnet werden.

Satz

Sei $\Delta(G)$ die maximale Gradzahl im Graphen G. Die gierige Färbung benutzt höchstens $\Delta(G) + 1$ Farben.

Beweis. Die nächste Methode liefert eine Färbung mit höchstens $\Delta(G) + 1$ Farbe:
Initialisierung: alle Knoten sind ungefärbt.
Wir iterieren in einer beliebigen Reihenfolge über die Knotenmenge und färben einen Knoten nach dem anderen. Der nächste Knoten erhält die Farbe minimaler Index, die unter den Farben seiner schon gefärbten Nachbarn noch nicht vorkommt.
Als ein Knoten gefärbt wird, schließen seine bereits gefärbten Nachbarn höchstens $\Delta(G)$ Farben aus, d.h. im schlimmsten Fall erhält der Knoten die $\Delta(G) + 1$ Farbe.

Bipartite Graphe

Definition

$G = (V, E)$ ist ein *bipartiter Graph* (páros gráf), wenn $V = A \cup B$, $A \cap B = \emptyset$ und es keine Kante gibt, die zwei Knoten in A oder zwei Knoten in B miteinander verbindet. G wird dann auch so bezeichnet: $G = (A, B, E)$. Der *vollständige bipartite* Graph K_{ab} ist ein bipartiter Graph mit $|A| = a$ und $|B| = b$, in dem alle Knoten in A mit allen Knoten in B benachbart sind.

Satz

Sei G ein Graph mit mindestens einer Kante. G ist genau dann bipartit wenn $\chi(G) = 2$.

Satz

Ein Graph ist genau dann bipartit, wenn er keinen Kreis ungerader Länge enthält.

Beweis. Sei K ein Kreis in einem bipartiten Graphen $G = (A, B, E)$. Dann sind die Knoten von K abwechselnd in A bzw. B. Daher besteht der Kreis aus einer geraden Anzahl von Knoten.
Umgekehrt sei $G = (A, B, E)$ ein Graph, in dem es keinen Kreis ungerader Länge gibt. Dann kann man die Mengen A und B wie folgt konstruieren. Sei S ein Spannwald des Graphen. In jedem Baum von S wähle man willkürlich eine Wurzel; diese werden in A sein. Jeder weitere Knoten ist in S über genau einen Weg aus der Wurzel seiner Komponente zu erreichen. Ist die Länge dieses Weges gerade, so wird der Knoten in A sein, sonst in B. Es muss nun bewiesen werden, dass es keine Kante innerhalb von A oder B gibt. Nehmen wir indirekt an, xy ist eine solche Kante. Es gibt aber genau ein $x - y$ Weg W in S. W hat gerade Länge deshalb $W + xy$ ist ein Kreis ungerader Länge. Das ist ein Wiederspruch.

Flüsse

Definition

Sei $G = (V, E)$ ein gerichteter Graph, $s \neq t$ zwei ausgezeichnete Knoten. Sei $c : E \to \mathbb{R}^+$, $c(e)$ wird die Kapazität von e genannt. Dann ist $N = (G, s, t, c)$ ein *Netzwerk*.

Definition

Sei $N = (G, s, t, c)$ ein Netzwerk. $f : E \to \mathbb{R}_0^+$ ist ein *Fluss*, wenn für jede Kante e die Ungleichung $f(e) \leq c(e)$ und für jeden Knoten v außer s und t die Gleichung
$m(v) = \sum_{xv \in E} f(xv) - \sum_{vy \in E} f(vy) = 0$ gilt.

$m_f = -m(s) = \sum_{sy \in E} f(sy)$ ist die *Stärke des Flusses*.

Definition

Sei f ein Fluss im Netzwerk N. f heisst ein maximaler Fluss, wenn $m_f \geq m_{f'}$ für alle Flüsse in N.

Definition

Sei $X \subseteq V$ so dass $s \in X$ und $t \notin X$. Dann ist X ein *st-Schnitt*. Die *Kapazität von* X ist

$$c(X) = \sum_{xy \in E, x \in X; y \notin X} c(xy).$$

Der *Fluss durch* X ist

$$f(X) = \sum_{xy \in E; x \in X, y \notin X} f(xy) - \sum_{xy \in E; x \notin X, y \in X} f(xy).$$

Lemma

Seien f ein Fluss und X ein *st*-Schnitt im Netzwerk N. Dann gilt:
(1) $f(X) = m_f$.
(2) $f(X) \leq c(X)$.
(3) Wenn für alle Kanten, die von X nach $V \setminus X$ laufen, $f = c$, und für alle Kanten, die von $V \setminus X$ nach X laufen, $f = 0$, dann hat f maximale Stärke.

Definition

Sei $N = (G, s, t, c)$ ein Netzwerk und f ein Fluss in N. Dazu definieren wir einen *Hilfsgraphen* G_f wie folgt. G_f hat dieselbe Knotenmenge wie G und ist auch gerichtet. In G_f gibt es eine Kante vom Knoten x zum Knoten y, wenn entweder $xy \in E(G)$ und $f(xy) < c(xy)$ (Vorwärtskante) oder $yx \in E(G)$ und $f(yx) > 0$ (Rückwärtskante). Wenn es in G_f einen gerichteten Weg von s zu t gibt, so ist das ein Verbesserungsweg.

Satz

Sei $N = (G, s, t, c)$ ein Netzwerk und f ein Fluss in N. f hat genau dann maximale Stärke, wenn es in G_f keinen Verbesserungsweg gibt.

Algorithmus (Ford-Fulkerson)

Input: Netzwerk $N = (G, s, t, c)$.
Output: Fluss f mit maximaler Stärke.
Ablauf: Wir fangen mit dem trivialen Fluss an, bei dem $f(e) = 0$ für alle Kanten. In einer allgemeinen Iteration gibt es einen Ausgangsfluss f, zu dem der Hilfsgraph G_f erstellt wird. In G_f wird nach einem Verbesserungsweg gesucht. Findet man so einen Weg, so wird f erhöht, und die nächste Iteration beginnt mit diesem geänderten Fluss. Findet man keinen Verbesserungsweg, so terminiert der Algorithmus, da f optimal ist.

Satz (Edmonds-Karp)

Wenn in jeder Iteration der kürzeste Verbesserungsweg (d.h. der Verbesserungsweg mit minimaler Anzahl von Kanten) benutzt wird, terminiert der Algorithmus von Ford und Fulkerson in Polynomialzeit.

Satz (Ford-Fulkerson, Max-Flow-Min-Cut)

In einem Netzwerk ist die maximale Flussstärke gleich der minimalen st-Schnitt-Kapazität.

Satz

Wenn alle Kapazitäten ganze Zahlen sind, ist auch die maximale Flussstärke eine ganze Zahl. Es gibt sogar einen Fluss f maximaler Stärke, für den $f(e)$ für jede Kante e eine ganze Zahl ist.

Beweis. Der durch den Algorithmus von Ford und Fulkerson gelieferte Fluss besitzt diese Eigenschaft.