

Skript zum Seminar

Flüsse in Netzwerken

WS 2008/09

David Meier

Inhaltsverzeichnis

1	Einführende Definitionen und Beispiele	3
2	Schnitte in Flussnetzwerken	12
2.1	Maximaler s t Fluss	12
2.2	Restgraphen	13
2.3	Ford-Fulkerson-Methode	14
2.4	Eine weitere Anwendung: Matching!	17

1 Einführende Definitionen und Beispiele

Definition 1: Ein **Graph** ist ein Paar $G = (V, E)$, wobei V eine endliche Menge und $E \subseteq V^{(2)} := \{e \in P(V) : |e| (= \text{Kardinalität von } e) = 2\}$ eine Teilmenge der ungeordneten Paare $\{u, v\}$ aus Elementen $u \neq v$ von V ist. $P(V)$ ist die Potenzmenge von V .

Die Elemente von V heißen **Ecken**, **Punkte** oder **Knoten**, die Elemente von E werden **Kanten** genannt.

Beispiel 1: Graphen

(Dies sind alles endliche Graphen, da $|V| < \infty$)

Beispiel 2: Gebilde wie **Schleifen**, **Mehrfachkanten** und **Multigraphen** werden auch von manchen Autoren als Graphen zugelassen.

Definition 2: Zwei **Graphen** $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ heißen **isomorph** $G_1 \simeq G_2$, falls es eine bijektive Abbildung $\varphi : V_1 \rightarrow V_2$ derart gibt, dass für alle $u, v \in V_1$ gilt: $\{u, v\} \in E_1 \Leftrightarrow \{\varphi(u), \varphi(v)\} \in E_2$

-Ein **Isomorphismus** ist also eine bijektive Abbildung der Eckenmengen, die benachbarte Ecken auf benachbarte Ecken und nicht benachbarte Ecken auf nicht benachbarte Ecken abbildet.

Isomorphie von Graphen ist demnach Gleichheit bis auf Eckenumbenennung.

Beispiel 3:

Wenn $e = \{u, v\} \in E$ gegeben, so sagen wir, u und v seien benachbart oder **adjazent**, die Punkte u (ebenso v) und e heißen **inzident**.

Beispiel 4: u und v ist mit e inzident und u und v sind adjazent.

u und w nicht adjazent aber V und u und V und w sind adjazent

Weiterhin kann man auch zu kleineren Gebieten übergehen, zu sogenannten Teilgraphen.

Definition 3: Gegeben seien zwei Graphen $G = (V, E)$ und $H = (W, F)$. Wir nennen H einen **Teilgraphen** von G , falls gilt: $W \subseteq V$ und $F \subseteq E$, dann schreiben wir: $H \subseteq G$. Ferner heißt H ein (von W) **induzierter Teilgraph**, falls $H \subseteq G$ und $F = E \cap W^{(2)}$: $H = G[W]$

Beispiel 5:

Definition 4: Eine **Kantenfolge** in einem gegebenen Graphen G ist ein angeordnetes Tupel $(e_1 \dots e_r)$ mit $e_j \in E$, $j \in \{1 \dots, r\}$ $\{r \in \mathbb{N}\}$, wobei $|e_j \cap e_{j+1}| = 1$ (d.h. e_{j+1} schließt an e_j an, $j \in \{1, \dots, r\}$). Die Zahl r heißt **Länge** der Kantenfolge (e_1, \dots, e_r) .

Kantenfolgen werden häufig in der Form $\mathbf{v}_0, \mathbf{e}_1, \mathbf{v}_1, \mathbf{e}_2, \mathbf{v}_2, \dots, \mathbf{v}_{r-1}, \mathbf{e}_r, \mathbf{v}_r$ oder $(\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_r)$ wobei jede Kante zwischen ihren Endpunkten steht.

\mathbf{v}_0 und \mathbf{v}_r sind der **Anfangs-** bzw. **Endpunkt** der Kantenfolge, die ihrerseits \mathbf{v}_0 mit \mathbf{v}_r verbindet.

Die **kürzeste Länge** einer \mathbf{v}_0 mit \mathbf{v}_r verbindenden Kantenfolge heißt der **Abstand** von \mathbf{v}_0 und \mathbf{v}_r .

Definition 5: Wir nennen eine Kantenfolge $\mathbf{v}_0, \mathbf{e}_1, \mathbf{v}_1, \mathbf{e}_2, \mathbf{v}_2, \dots, \mathbf{v}_{r-1}, \mathbf{e}_r, \mathbf{v}_r$ **geschlossen**, falls $\mathbf{v}_0 = \mathbf{v}_r$ ist, sie heißt **Kantenzug**, falls alle \mathbf{e}_i paarweise verschieden sind, sowie **Weg**, falls alle \mathbf{v}_j (und damit alle \mathbf{e}_i) verschieden sind.

Ist die betrachtete Kantenfolge geschlossen mit $r \geq 3$ und ein Weg, so heißt sie ein **Kreis**.

Beispiel 6:

Definition 6: Wir schreiben $u \sim v$ für zwei Ecken eines Graphen G , wenn es eine u und v verbindende Kantenfolge gibt. Die **Äquivalenzklassen** der somit eingeführten **Äquivalenzrelation** " \sim " heißen die Zusammenhangskomponenten von G .

Beispiel 7: Zeigt einen aus vier Zusammenhangskomponenten bestehenden Graphen:

Definition 7: Netzwerk

Ein **gerichteter Graph** oder **Netzwerk** ist ein Paar $G := (V, B)$, wobei V eine endliche Menge und $B \subseteq V^{(2)} := V \times V$ ist.

Wir nennen V die **Knoten,- Punkt- oder Eckenmenge** von G , B die Menge der **Bögen** oder **gerichteten Kanten** von G .

Seien ein gerichteter Graph $G = (V, B)$ und ein Bogen $e = (s, t) \in B$ gegeben. Dann nennen wir s den **Startpunkt** und t den **Zielpunkt** von e ; s und t heißen die **Endpunkte** von e .

Beispiel 8: Netzwerk

w_{ij} := Kapazität, die von i nach j geschickt werden kann.

s := Startpunkt t := Zielpunkt

s wird auch Quelle und t wird auch Senke genannt.

Die Frage ist nun, wie man diese Netzwerk optimieren kann, um möglichst viel von s nach t oder über den Bogen k von t nach s zu schicken.

Nun können wir folgendes lineares Programm herleiten.

$$\begin{array}{l} \max f_{ts} \\ \text{s.t. } \sum_{j \in N_{(i)}^+} f_{ij} - \sum_{i \in N_{(i)}^-} f_{ji} = 0 \quad \forall i \in V \\ 0 \leq f_{ij} \leq w_{ij} \quad \forall ij \in E \end{array}$$

Äquivalent dazu ist folgendes lineares Programm.

$$\begin{array}{l} \max x_{ts} \\ \text{s.t. } Ax = 0 \\ 0 \leq x \leq w \end{array}$$

Zur Erläuterung:

$\max f_{ts}$ bedeutet, dass der Fluss durch das Netzwerk maximiert werden soll.

$\sum_{j \in N_{(i)}^+} f_{ij} - \sum_{i \in N_{(i)}^-} f_{ji} = 0 \quad \forall i \in N$ bedeutet, dass von i nach j genauso viel fließt wie von j nach i . Diese Bedingung nennt man ("Flusserhaltungsbedingung").

Die letzte Bedingung $0 \leq x \leq w$ ist auch klar, der Fluss von i nach j ist größer

oder gleich Null und kleiner oder gleich der maximalen Kapazität, die dort fließen kann.

Bemerkung 1: Aus der Definition eines Netzwerkes und obiger Erläuterung, ist zu sehen, dass es nur 3 verschiedene Flussmöglichkeiten geben kann, entweder der Fluss fließt von u nach i (1) oder er fließt von i nach u (-1) oder der Fluss ist (0).

Daraus kann folgende Matrix konstruiert werden.

$$a_{ij} = \begin{cases} 1 & \text{falls } \exists u \in V : j = (u, i) \in E \text{ (Pfeil } j \text{ endet in } i) \text{ (1)} \\ -1 & \text{falls } \exists u \in V : j = (i, u) \in E \text{ (Pfeil } j \text{ beginnt in } i) \text{ (2)} \\ 0 & \text{sonst.} \end{cases}$$

Diese Matrix heißt **Knoten-Kanteninzidenzmatrix** $A \in \{0, 1, -1\}^{|V| \times |E|}$

Definition 8: Die **Knoten-Kanteninzidenzmatrix** ist die Matrix $A \in \mathbf{R}^{|V| \times |E|}$ mit

$$a_{u,e} = \begin{cases} 1 & \text{falls } u \in e \\ 0 & \text{sonst.} \end{cases}$$

Die Matrix heißt **unimodular**, falls die Determinante jeder aus m linear unabhängigen Spalten bestimmten **Untermatrix** den Wert ± 1 hat.

Sie heißt **total unimodular**, wenn jede quadratische **Untermatrix** die Determinante ± 1 oder 0 hat.

Bemerkung 2: Die **Knoten-Kanteninzidenzmatrix** eines **gerichteten Graphen** ist **total unimodular**.

Zum Beweis benötigen wir noch folgenden Satz.

Satz 1: (Heller und Tompkins (1956))

Sei $A \in \{0, 1, -1\}^{m \times n}$ mit höchstens zwei nichtverschwindenden Einträgen pro

Spalte. A ist **total unimodular** \Leftrightarrow Die Zeilen von A können in zwei Klassen eingeteilt werden, so dass Zeilen mit einem $+1$ und einem -1 Eintrag in der gleichen Spalte in die gleiche Klasse und Zeilen mit zwei vorzeichengleichen Einträgen in der gleichen Spalte in unterschiedliche Klassen kommen.

Beispiel zu obigem Satz.

Ist folgende Matrix total unimodular?

Sei folgende Matrix A gegeben

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ -1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{matrix} a & a & a & a & a \\ \star & b & b & b & b \\ \star & \star & a & a & b \times \\ a & a & a & a & a \end{matrix}$$

$\Rightarrow A$ ist nicht total unimodular.

Beweis: " \Rightarrow " Spalten mit nur einem Eintrag sind vernachlässigbar. Fassen zunächst Zeilen gemäß 1. zu Zeilenmengen Z_i , $i = 1, \dots, h$ zusammen.

$$\text{also } \begin{array}{|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & -1 & -1 \\ \hline 1 & -1 & 1 & 0 & 0 & 1 \\ \hline 1 & 0 & 1 & 1 & 0 & -1 \\ \hline 0 & 0 & -1 & -1 & -1 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 & -1 \\ \hline \end{array} \begin{matrix} Z_1 \\ Z_2 \\ Z_3 \end{matrix}$$

Zwischen den Knotenmengen Z_1, Z_2, Z_3 verlaufen Kanten.

Folgende Matrix besteht aus 2 Zeilenmengen und ist die Knoten-Kanten-Inzidenzmatrix eines Kreises:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

Man kann zeigen, dass ihre Determinante 2 ist.

Dies ist ein Widerspruch zu 2. ; also gibt es eine

Untermatrix der Form

$$B = \begin{matrix} 1 \\ 2 \\ 3 \\ \vdots \\ k \end{matrix} \begin{pmatrix} 1 & 0 & 0 & \cdots & 1 \\ -1 & 1 & & & 0 \\ 0 & -1 & \ddots & & \vdots \\ \vdots & & & \ddots & 1 & 0 \\ 0 & \cdots & 0 & -1 & 1 \end{pmatrix}$$

$$\det B = B_{1,1} \det B_{2:k,2:k} + (-1)^{k-1} B_{1,k} \det B_{2:k,1:k-1} = 1 \cdot 1 + (-1)^{k-1} \cdot (-1)^{k-1} = 2$$

Zur Wiederholung Entwicklungssatz:

Für $k = 4$: $\begin{bmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$ Berechnung der Determinante!!

$$\begin{aligned} & \begin{vmatrix} 1 & 0 \\ -1 & 1 \end{vmatrix} \cdot \begin{vmatrix} 1 & 0 \\ -1 & 1 \end{vmatrix} - \begin{vmatrix} 1 & 0 \\ -1 & 0 \end{vmatrix} \cdot \begin{vmatrix} -1 & 0 \\ 0 & 1 \end{vmatrix} + \begin{vmatrix} 1 & 1 \\ -1 & 0 \end{vmatrix} \cdot \begin{vmatrix} -1 & 1 \\ 0 & -1 \end{vmatrix} + \begin{vmatrix} 0 & 0 \\ 1 & 0 \end{vmatrix} \cdot \begin{vmatrix} 0 & 0 \\ 0 & 1 \end{vmatrix} \\ & - \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix} \cdot \begin{vmatrix} 0 & 1 \\ 0 & -1 \end{vmatrix} + \begin{vmatrix} 0 & 1 \\ 0 & 0 \end{vmatrix} \cdot \begin{vmatrix} 0 & -1 \\ 0 & 0 \end{vmatrix} \\ & \Rightarrow 1 \cdot 1 + 1 \cdot 1 = 2 \end{aligned}$$

Außerdem ist dies ein Widerspruch zu "B ist Untermatrix einer total unimodularen Matrix", da Determinante = 2.

Bauen nun Graphen $G = (V, E)$ mit $V = \{Z_i : i = 1, \dots, n\}$, also Zeilenmengen sind Knoten und $\{Z_i, Z_j\} \in E \Leftrightarrow \exists$ Spalte \bar{j} mit $A_{z,\bar{j}} = A_{\bar{z},\bar{j}} \neq 0$, $\bar{z} \in Z_i$, $\bar{z} \in Z_j$, also eine Kante wird eingeführt, wenn gemäß 2. die Zeilenmengen in unterschiedliche Klassen gehören.

Behauptung: "G" ist bipartit.

Das heißt, es existieren V_1, V_2 mit $V = V_1 \cup V_2$ (disjunkt) und $E \subseteq \{\{u, v\} : u \in V_1, v \in V_2\}$

Annahme: G ist nicht bipartit, also gibt es eine Untermatrix der Form:

$$B = \begin{array}{c} \underbrace{\hspace{2cm}}_{\text{Kante } e_1} \quad \underbrace{\hspace{2cm}}_{\text{Kante } e_k} \\ \begin{pmatrix} 1 & 0 & 0 & \cdots & 1 \\ 1 & 1 & & & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & \cdots & 0 & 1 & 1 \end{pmatrix} \end{array}$$

mit $\det B = 1 + (-1)^{k-1} \cdot 1 = 2 \Rightarrow$ Widerspruch zu A ist total unimodular (Untermatrix B ist total unimodular)

Bei obiger Matrix existieren Kanten, die nicht zwischen den Knotenmengen Z_1 und Z_2 verlaufen, sondern innerhalb von z.B. Z_1 .

Beachte: Sind für $Z_i = e_i \cap e_j$ die Zeilen \bar{z} und $\bar{\bar{z}} \in Z_i$ unterschiedlich, so muss eigentlich statt

$$B = \begin{array}{c} \bar{z} \rightarrow \\ \bar{\bar{z}} \rightarrow \end{array} \begin{pmatrix} 1 & 0 & 0 & \cdots & 1 \\ 1 & 1 & & & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & \cdots & 0 & 1 & 1 \end{pmatrix} \rightarrow \begin{array}{c} \bar{z} \rightarrow \\ \bar{\bar{z}} \rightarrow \end{array} \begin{pmatrix} 1 & 0 & 0 & \cdots & 1 \\ 1 & 1 & & & 0 \\ 0 & -1 & \ddots & & \vdots \\ \vdots & & -1 & 1 & 0 \\ 0 & \cdots & 0 & -1 & 1 \end{pmatrix}$$

geschrieben werden. Aber jede -1 erhöht auch k um eins und verändert das Ergebnis daher nicht.

" \Leftarrow " Sei B eine beliebige quadratische Untermatrix O. B. d. A. hat B nur Spalten mit 2 Einträgen $\neq 0$ (die anderen können leicht mit Laplaceschem Entwicklungssatz behandelt werden (siehe oben)). Sei $\{Z_1, Z_2\}$ eine Partition der Zeilen gemäß 1. und 2., also

1	0	0	0	0	Z_1
-1	-1	1	1	1	
0	0	1	1	0	Z_2
0	0	0	0	1	
0	-1	0	0	0	

multipliziere die Zeilen aus Z_1 mit +1 und die aus Z_2 mit -1. Erhalten dann

1	0	0	0	0	Z_1
-1	-1	1	1	1	
0	0	-1	-1	0	Z_2
0	0	0	0	-1	
0	1	0	0	0	

Dann ist die Summe der Zeilen der Nullvektor, also sind sie linear abhängig. Damit haben wir Determinante Null von B erreicht, nun müssen wir jedoch noch zeigen, dass es auch Determinanten mit +1 und -1 gibt. Das können wir leicht an 2×2 Matrizen sehen.

Beispiel:

$$\begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \text{ Besitzt die Determinante 1.}$$

$$\begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \text{ Besitzt die Determinante 0.}$$

Für größere Matrizen ist dies per Induktion zeigbar. Da dies nun ein gerichteter Graph ist (genau eine 1 und eine -1 pro Spalte) haben wir erreicht, dass die Determinante jeder quadratischen Untermatrix 1 ist. Entwickeln wir nun nach Zeilen mit -1 Einträgen, wächst sich das Vorzeichen und aus 1 wird -1. #

Nun zum Beweis von Bemerkung 2:

Mit Hilfe des Satzes von **Heller und Tompkins** ergibt sich folgendes: Weil in jeder Spalte genau eine +1 und eine -1 ist liegen alle Zeilen sogar in einer Klasse. #

Denn ein gerichteter Graph sieht so aus:

Pfeil j endet in i und kommt von u danach beginnt Pfeil in i und endet in u.

Beispielmatrix dazu:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

2 Schnitte in Flussnetzwerken

2.1 Maximaler s t Fluss

Frage: Wie groß kann ein maximaler Fluss überhaupt sein?

In einem Netzwerk kann nicht mehr fließen, als aus der Quelle hinaus oder in die Senken hineinfließt.

Außerdem begrenzt jeder **Schnitt** durch den Graphen, der **s** von **t** trennt den Wert des maximalen Flusses.

Beispiel 1:

Also muss der Fluss in l und k maximiert werden.

Es gilt:

Der Fluss ist genauso stark wie die Kapazität des schwächsten Gliedes es zulässt.

Definition 1: Schnitte:

Die Kantenmenge $\delta^+(S) = \{i, j \in E : i \in S, j \in V \setminus S\}$ heißt Schnitt also

- Zerlegung der Knotenmenge V in S und T - mit $s \in S$ und $t \in T$
- Der Nettofluss in einem Schnitt ist die Summe des positiven Flusses von S nach T minus dem von T nach S .
- Die Kapazität $c(s, t)$ eines Schnittes S, T ist die Summe der Kapazitäten aller Kanten die von S nach T führen.
- Ein Schnitt mit kleinster Kapazität unter allen möglichen Schnitten heißt minimaler Schnitt.
- Der Nettofluss ist über jeden Schnitt gleich.

Bemerkung 1: Der Wert eines jeden Flusses f in einem Flussnetzwerk G wird durch die Kapazität eines beliebigen Schnittes (S, T) begrenzt.

2.2 Restgraphen

Definition 1:

- Restgraph $G_f(V, E_f)$ ist ein Graph mit der Kantenmenge $E_f = \{(u, v) \in V \times V : c_f(u, v) > 0\}$
- Der Restgraph $G_f(V, E_f)$ zu einem Fluss f beschreibt gerade alle Flussvergrößerungsmöglichkeiten, das heißt der Restgraph gibt an, wie meine Netzwerkausnutzung gerade ist und wo noch freie Kapazitäten sind.
- Restkapazität : $c_f(u, v) = c(u, v) - f(u, v)$
- **Der Restgraph besteht aus Kanten, deren Kapazitäten nicht voll ausgeschöpft sind.**
- Jeder Weg von S nach T im Restgraphen ist ein zunehmender Weg
- Den gesamten Fluss kann man höchstens um das Minimum der gefundenen Restkapazitäten erhöhen.
- **Ein Fluss ist genau dann maximal, wenn es für f keinen zunehmenden Weg gibt.**

-Denn wenn es einen zunehmenden Weg für einen Fluss f gibt, dann könnte man den Fluss entlang dieses Weges vergrößern.

Satz 1: Max Fluss - Min Schnitt Theorem

Sei f ein Fluss in einem Flussnetzwerk $G = (V, E)$ dann sind äquivalent:

1. f ist ein maximaler Fluss
2. Der Restgraph enthält keine zunehmenden Wege
3. Der Wert des Flusses f entspricht der Kapazität eines Schnittes: $|f| = c(s, t)$ für einen Schnitt (S, T) in G

Beweis:

(1) \rightarrow (2)

Sofern es für einen maximalen Fluss einen zunehmenden Weg gibt, kann dieser Fluss nicht maximal sein.

(2) \rightarrow (3)

- $S = \{v \in V, \text{ es existiert ein Weg von } s \text{ nach } v \text{ in } G_f\}$
- $T = V - S$
- (S, T) ist Schnitt
- $|f| = c(S, T)$

(3) \rightarrow (1)

- der Schnitt ist minimal
- daher muss f maximal sein

2.3 Ford-Fulkerson-Methode

Methode zur Lösung des Problems des maximalen Flusses!
Benutzt werden Restgraph, zunehmende Wege

Algorithmus:

for jede Kante $(u, v) \in E$

do $f[u, v] \leftarrow 0$

$f[v, u] \leftarrow 0$

while es gibt einen Weg p von s nach t im Restgraphen

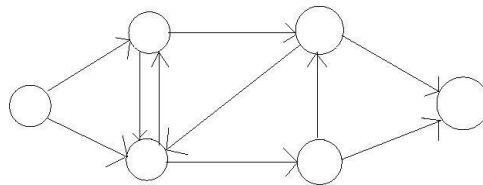
do $c_f(p) \leftarrow \min \{c_f(u, v) : (u, v) \text{ ist in } p\}$

for jede Kante (u, v) in p

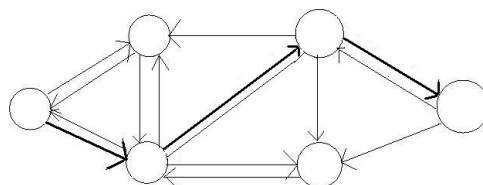
do $f[u, v] \leftarrow f[u, v] + c_f(p)$

$f[v, u] \leftarrow -f[v, u]$

Beispiel zur Flussmaximierung mit obigem Algorithmus:

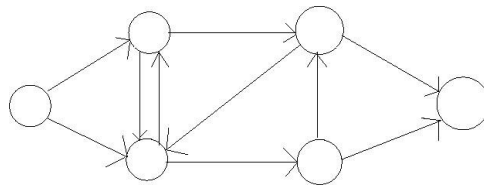


Flussnetzwerk Fluss/Kapazität (1)

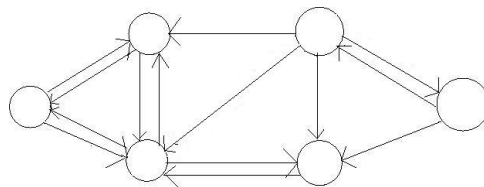


Restgraph mit zunehmendem Weg (2)

$$c_f(p) = c(v_3, v_2) = 4$$



Flussnetzwerk mit zunehmendem Weg aus (2) (3)



Restgraph des maximalen Flusses. (4)

2.4 Eine weitere Anwendung: Matching!

Problem: In einem Arbeitsamt sind s Stellen zu vermitteln und r Bewerber gegeben. Jedem Bewerber ordne man die Menge der Stellen zu, für die er qualifiziert ist.

Frage: Wieviele Arbeitsplätze kann das Arbeitsamt maximal vermitteln so, dass jeder vermittelte Bewerber eine Stelle bekommt, für die er qualifiziert ist.

Formal: Wir führen einen Graphen mit $r + s$ Ecken ein:

$$G := (V, E), V = V_1 \cup V_2, |V_1| = r, |V_2| = s,$$

wobei jedem Bewerber ein Eckpunkt aus V_1 und jede Stelle ein Eckpunkt aus V_2 entsprechen mag. Wir ziehen eine Kante zwischen $u \in V_1$ und $w \in V_2$, wenn der Bewerber u für die Stelle w qualifiziert ist. Wir suchen dann eine möglichst große Anzahl von Kanten derart, dass keine zwei dieser Kanten einen Endpunkt gemeinsam haben.

Definition 1: Ein Graph $G = (V, E)$ heißt **bipartit**, falls $\exists V_1, V_2$ mit $V = V_1 \cup V_2$ (disjunkt vereinigt) und $E \subseteq \{\{u, v\} : u \in V_1, v \in V_2\}$. Der vollständige bipartite Graph $|V_1| = n$ und $|V_2| = m$ Knoten wird mit $K_{n,m}$ bezeichnet.

Definition 2: Seien ein bipartiter Graph $G = (V, E)$ und eine Teilmenge $M \subseteq E$ vorgegeben.

- a) Die Menge M heißt ein **Matching (auch eine Heirat)**: in G , falls je zwei verschiedene Kanten $e_1, e_2 \in M$ keine gemeinsame Ecke $v \in V$ haben.
- b) Als ein Matching heißt **maximal**, falls M unter allen Matchings die größte Kardinalität hat. Dann nennen wir $m(G) := |M|$ die **Matching-Zahl** von G .

Ziel: Konstruktion eines maximalen Matchings für einen Graphen.

Dies wollen wir mit der **ungarischen Methode** erreichen. Doch zuvor wieder einige Definitionen.

Definition 3: gesättigtes Matching

Ein Matching M heißt **gesättigt**, falls kein Matching M' mit $M \subset M'$ existiert.

Definition 4: Ein Matching M heißt **perfekt**, falls alle Ecken von G durch M überdeckt werden.

Ungarische Methode (Algorithmus)

Das folgende Verfahren konstruiert in jedem gegebenen bipartiten Graphen $G = (V, E)$, $V = V_1 \cup V_2$, o.B.d.A. mit $|V_1| \leq |V_2|$ ein maximales Matching

1. Konstruieren wir ein gesättigtes Matching (M) (einfach durch Auswahl beliebiger Kanten, welche ein Matching bilden, und anschließender Hinzunahme weiterer Kanten bis das Matching gesättigt ist)
2. Falls M bereits V_1 überdeckt, so sind wir fertig. Andernfalls wählen wir ein $a \in V_1 \setminus \{V|V : \text{Ecke in } M\}$ und setzen $S := \{a\}, I := \emptyset, B := \{a\}$
3. Ist $I = N(S)$ (d.h. I besteht aus allen Nachbarn von S in G), so setzen wir $V_1 := V_1 \setminus \{a\}$ und gehen zu Schritt 2.
Andernfalls: Ist $I \neq N(S)$, so wählen wir ein $y \in N(S) \setminus I$ und eine Kante $\{x, y\}$ mit $x \in V_1(B)$, nun gehen wir zu Schritt
4. 1 Falls y von M überdeckt wird, so gibt es eine Kante $\{y, z\} \in M$, für die z nicht in $V_1(B)$ liegt. Wir setzen $S := S \cup \{z\}, I = I \cup \{y\}$ und erweitern B um die Ecken y, z sowie die Kanten $\{x, y\}$ und $\{y, z\}$. Gehen wir nun mit dem neuen B zu Schritt 3. **Andernfalls:**
4. 2 Falls y nicht von M überdeckt wird, so ist der Weg von x nach a in B , zusammen mit der Kante $\{x, y\}$, M -zunehmend in $G \Rightarrow$ (**erweitere** M)
Wir gehen zurück zu Schritt 2.

Erweitere M wie folgt:

$$M = \{\{v_1, v_2\}, \{v_3, v_4\}, \dots, \{v_{2k-1}, v_{2k}\}\}$$

Also: Nimm weitere Kante hinzu

Beispiel maximales Matching:

Zusammenhang Flüsse in Netzwerken und Matchingtheorie:

Umwandlung:

- Hinzufügen von s und t zum bipartiten Graphen aus der ungarischen Methode
- Ausrichten der vorhandenen Kanten von V_1 nach V_2
- Neue Kante von s nach V_1 und V_2 nach t hinzufügen
- Jeder Kante gleichbleibende Kapazität 1 zuweisen.

Finden des maximalen Matchings im Flussnetzwerk G'

Definition 5: Das dem bipartiten Graphen G entsprechende Flussnetzwerk $G' = (V', E')$ ist wie folgt definiert:

- Quelle s und Senke t sind neue Knoten nicht in V aber in $V' = V \cup \{s, t\}$
- Wenn die Aufteilung der Knoten von G gleich $V = V_1 \cup V_2$ ist, dann sind die gerichteten Kanten G' die Kanten von E , gerichtet von V_1 nach V_2 mit V neuen Kanten.
- Letztendlich ordnet man wie oben jeder Kante von E' eine Kapazität von eins zu.

Lemma (O.B.): Sei $G = (V, E)$ ein bipartiter Graph mit Knotenaufteilung $V = V_1 \cup V_2$ und sei $G' = (V', E')$ das G entsprechende Flussnetzwerk. Wenn M ein Matching in G ist, dann gibt es einen Fluss f in G' mit dem Flusswert $|f| = |M|$. Umgekehrt, wenn f ein Fluss in G' ist, dann gibt es ein Matching M in G mit der Kardinalität $|M| = |f|$.

Endresultat: Die Kardinalität eines maximalen Matchings M in einem bipartiten Graphen G ist der Wert eines maximalen Flusses f in seinem entsprechenden Flussnetzwerk G' .

Literaturverzeichnis

- [1] Helmberg, C.: Optimierung 1: 2004
- [2] Prof. Weber : Diskrete Mathematik : 2001
- [3] Fersch Thomas : Flüsse, Schnitte und bipartite Abbildungen
- [4] Itet D.: Diskrete Mathematik : 2001
- [5] (zum nicht erwähnten Netzwerksimplex) Kalcsics Jörg : Das Netzwerksimplex-Verfahren