Ungarische Methode

Zuweisungsproblem

Netzwerksimplex

Transportproblem

Cycle Cancelling

Minimum mean cost cycles

Netzwerksimplex

MCNFP

Minimaler Kostenfluss — Simplex: Lineares Programm

Dijkstra / BF

Kürzeste Wege Problem

Ford-Fulkerson

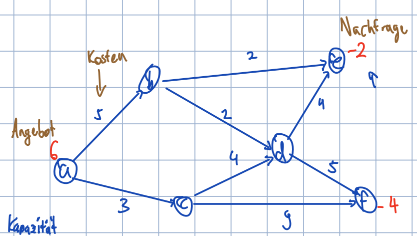
Maximales Fluss Problem

Edmonds-Karp

$\longrightarrow$ Allgemeinheit

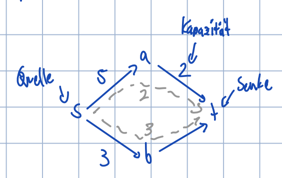
Gegeben: Minimum Cost Network Flow Problem: $N = (\overset{\text{Knoten}}{V}, \overset{\text{Kanten}}{E}, \underset{\text{Transportkosten}}{c}, \overset{\text{Kapazitäten}}{u}, \underset{\text{Bedarf}}{b})$

↳ "Minimiere die Kosten, während Bedarfe gedeckt werden."



Max Flow Problem: $N = (V, E, u, \underset{\text{Quelle}}{s}, \overset{\text{Senke}}{t})$

↳ "Maximiere den Fluss durch das Netzwerk"



Matroide

$M = (E, I)$ — E: Grundmenge; $I \subseteq P(E)$ "Menge aller Teilmengen"

Axiome:

1. $\emptyset \in I$
2. Hereditärbedingung: Wenn $I \in I$ und $I' \subseteq I$ dann ist $I' \in I$

} Unabhängigkeitssystem

3. $A, B \in I$ $\quad |A| < |B|$ $\quad \exists x \in B \setminus A$, sodass $A \cup \{x\} \in I$ — "Austauscheigenschaft"

Vektoren

Graphen

$$A := \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1/2 & 1 & 0 \\ 0 & 0 & 1 & 1/2 & 1 & 0 \end{pmatrix} \end{array}$$

$\mathbb{R}^3$

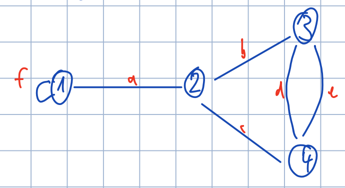
$E = \{a, b, c, d, e, f\}$

$I = \{\emptyset, \{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{a, b\}, \{a, c\}, \dots, \{a, c, e\}\} \subseteq P(E)$

↳ Menge von zueinander linear unabhängigen Vektoren

$G = (V, E)$ $\quad V = \{1, 2, 3, 4\}$

$E = \{a, b, c, d, e, f\}$



1. "I ist nicht leer"
2. "Wenn B eine Menge linear unabhängiger Spalten ist, dann ist jede Teilmenge A von B linear unabhängig"

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

3. Wenn $A, B \subseteq I'$, und $|A| < |B|$, dann $\dim(\mathrm{span}(A)) < \dim(\mathrm{span}(B))$
   → Wähle eine Spalte $x \in B$ und nicht in $\mathrm{span}(A)$
   → $A \cup \{x\} \subseteq M$ und linear unabhängig

B: $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ A

I = Alle kreisfreien Teilgraphen

$I = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \dots, \{a, c, e\}\} \subseteq P(E)$

Rangfunktion von Matroiden:

$B \subseteq E$; $\quad r(B) = \max\{|A| \mid A \subseteq B,\ A \in I\}$

"Kardinalität der größten unabhängigen Teilmenge von B"

! Wenn die Lösung der IP-Relaxation = Lösung des IPs ist, dann entspricht das LP einem ganzzahligen Polyeder und kann einfacher und in schnellerer Laufzeit gelöst werden. !

Tritt auf, wenn

- A vollständig unimodular und b ganzzahlig (Beweis s. Vorlesung)
- Nebenbedingungen durch Rangfunktion eines Matroiden bestimmt

Matroid $M = (E, I)$

1. $\emptyset \in I$
2. Hereditärbedingung: Wenn $I \in I$ und $I' \subseteq I$ dann $I' \in I$ } Unabhängigkeitssystem
3. $A, B \in I$ $|A| < |B|$ $\exists x \in B \setminus A$, sodass $A \cup \{x\} \in I$ "Austauscheigenschaften

Aufgabe Ü9.1

$V_1 = (E_1, I_1)$ $V_2 = (E_1, I_1)$ sind Matroide

a) $V_1 \cap V_2 := (E_1 \cap E_2, I_1 \cap I_2)$

1. $\emptyset \in I \Leftrightarrow \emptyset \in I_1 \cap I_2$

Wir wissen, dass $\emptyset \in I_1$ und $\emptyset \in I_2$ ist, da V_1 und V_2 Matroide. $\Rightarrow \emptyset \in I_1 \cap I_2$

2. Wenn $I \in I_1 \cap I_2$ und $I' \subseteq I$ dann $I' \in I$

$\hookrightarrow$ $A \in I_1 \cap I_2$ und $B \subseteq A$

$\Downarrow$ $A \in I_1$ und $A \in I_2$ $\Rightarrow$ $B \in I_1$, $B \in I_2$ gilt $\Rightarrow B \in I_1 \cap I_2$

b) Widerlegen, dass $V_1 \cap V_2$ ein Matroid

In unserem Fall: $\Leftrightarrow$ Zeige, dass 3tes Axiom nicht gilt

Beweis durch Widerspruch: (Finde ein beispiel für dass es nicht gilt)

$V_1 = (\{a,b,c\}, \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}\})$
$V_2 = (\{a,b,c\}, \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,c\}, \{b,c\}\})$

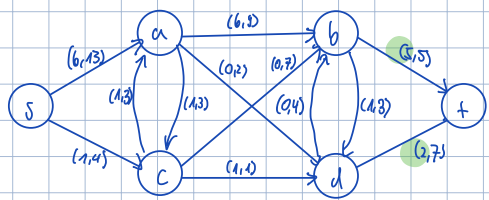
$V_1 \cap V_2 = (\{a,b,c\}, \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,c\}\})$

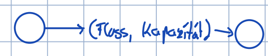
3. $A, B \in I$ $|A| < |B|$ $\exists x \in B \setminus A$, sodass $A \cup \{x\} \in I$ "Austauscheigenschaften

$A = \{b\}$
$B = \{a, c\}$
$\overset{1}{|A|} \le \overset{2}{|B|}$ $B \setminus A = \{a, c\}$

Entweder $A \cup \{a\}$ oder $A \cup \{b\}$ muss $\in I$: $\{b, a\}$ $\{b, c\}$ ist beides nicht drinnen $\Rightarrow$ Widerlegt

Aufgabe Üg 2

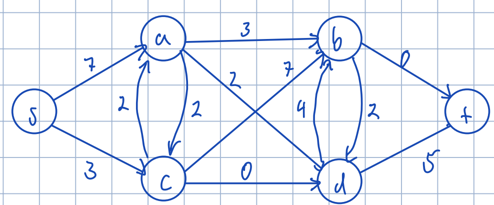




a) Was ist der aktuelle Fluss (⇔ Wie viel Einheiten kommen an der Senke an)

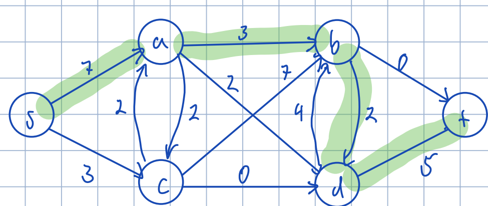
$5+2=7$

b) Residualnetzwerk (Vorwärtskante für Restkapazität aka wie viel noch übrig ist)



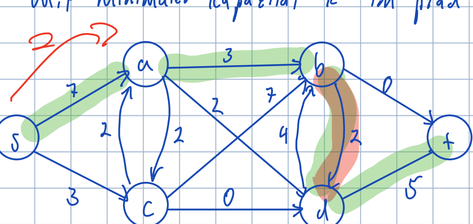
Ford-Fulkerson:

1. Finde einen beliebigen Flusserhöhenden Pfad von s nach t (augmentierter Pfad)



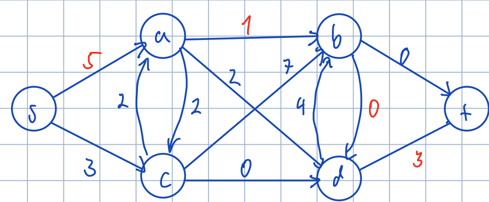
(s, a, b, d, t)

2. Suche Kante mit minimaler Kapazität K im Pfad und schicke Fluss größe K entlang

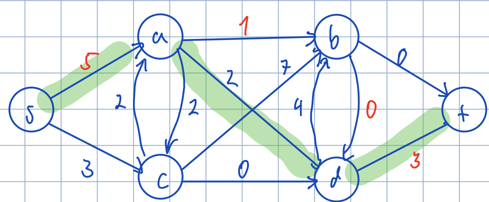


$s \xrightarrow{7} a \xrightarrow{3} b \xrightarrow{2} d \xrightarrow{5} t$

3. Neues Residualnetzwerk:

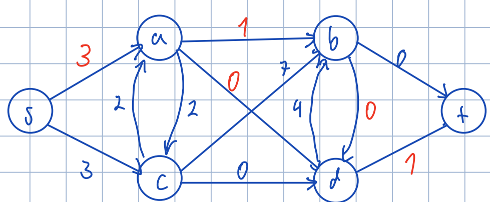


2ter Pfad:

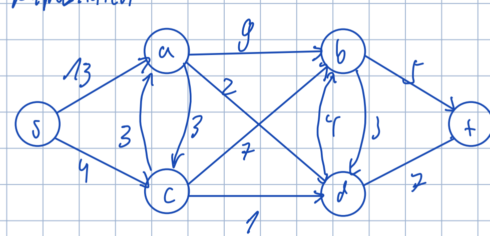


Bekommen Kapazität von 2 durch

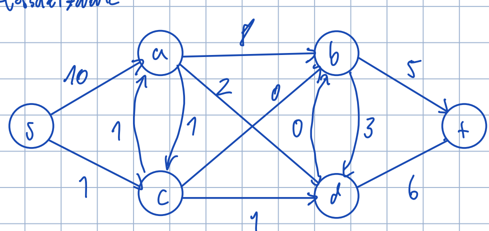
Residualnetzwerk nach 2 Schritten



Kapazitäten:



Flussnetzwerk



Residualnetzwerk ≙ Was noch übrig ist (Residuals)

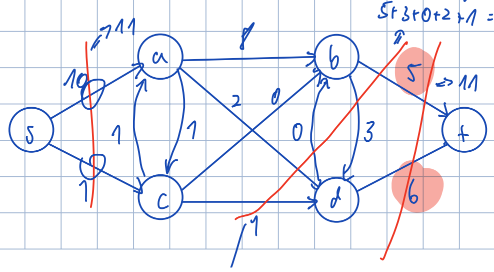
⟺ Kapazitäten − Flussnetzwerk

c) Ist der Fluss maximal? ⇒ Ja, weil es im Residualnetzwerk keinen augmentierenden Pfad >0 gibt

Separiere Senke und Quelle bipartit

d) Min cut = Max flow (Theorem)

5+3+0+2+1 = 11



⇒ Min cut = 11

⟺ Aktuellen Fluss = 11

5+6

rote Linien = 3 mögliche cuts

$X = \{s, a, b, c\}$ $V \setminus X = \{f, d\}$

Aufgabe Ü9.3 Ungarische Methode (Lösen von Zuweisungsproblemen)

$$\begin{matrix} 90 & 75 & 75 & 80 \\ 35 & 85 & 55 & 65 \\ 125 & 95 & 90 & 105 \\ 45 & 110 & 85 & 115 \\ 55 & 85 & 60 & 85 \end{matrix}$$

Schritt 0: Brauchen Quadratische Matrix: Input: Quadratische $m \times m$ Kostenmatrix

$$\begin{matrix} 90 & 75 & 75 & 80 & 0 \\ 35 & 85 & 55 & 65 & 0 \\ 125 & 95 & 90 & 105 & 0 \\ 45 & 110 & 85 & 115 & 0 \\ 55 & 85 & 60 & 85 & 0 \end{matrix}$$

Schritt 1: Subtrahiere Zeilenminimum

/ (wegen 0)

Schritt 2: subtrahiere Spaltenminimum

$$\begin{matrix} 90 & 75 & 75 & 80 & 0 \\ 35 & 85 & 55 & 65 & 0 \\ 125 & 95 & 90 & 105 & 0 \\ 45 & 110 & 85 & 115 & 0 \\ 55 & 85 & 60 & 85 & 0 \end{matrix} \Longrightarrow \begin{matrix} 55 & 0 & 20 & 15 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 90 & 20 & 35 & 40 & 0 \\ 10 & 35 & 40 & 50 & 0 \\ 20 & 10 & 5 & 20 & 0 \end{matrix}$$

Schritt 3 Geringste Anzahl an Linien um alle Nullen zu überdecken

$$\begin{matrix} 55 & 0 & 20 & 15 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 90 & 20 & 35 & 40 & 0 \\ 10 & 35 & 40 & 50 & 0 \\ 20 & 10 & 5 & 20 & 0 \end{matrix}$$

→ sind erst fertig wenn Anzahl Linien = m = #Zeilen
= #Spalten

$3 < 5$

Schritt 4: Kleinsten, positiven Eintrag der Matrix von anderen abziehen und zu doppelt abgedeckten addieren

$$\begin{matrix} 55 & 0 & 20 & 15 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 90 & 20 & 35 & 40 & 0 \\ 10 & 35 & 40 & 50 & 0 \\ 20 & 10 & 5 & 20 & 0 \end{matrix} \Longrightarrow \begin{matrix} 55 & 0 & 20 & 15 & 5 \\ 0 & 10 & 0 & 0 & 5 \\ 85 & 15 & 30 & 35 & 0 \\ 5 & 30 & 35 & 45 & 0 \\ 15 & 5 & 0 & 15 & 0 \end{matrix}$$

Minimale Abdeckung:

55	0	20	15	5
0	10	0	0	5
85	15	30	35	0
5	30	35	45	0
15	5	0	15	0

=> 4 < 5

55	0	20	15	10
0	10	0	0	10
80	10	35	30	0
0	25	30	40	0
15	5	0	15	5

55	0	20	15	10
0	10	0	0	10
80	10	35	30	0
0	25	30	40	0
15	5	0	15	5

=> Terminiert

Optimale Zuordnung:

55	$\boxed{0}$	20	15	10
0	10	0	$\boxed{0}$	10
80	10	35	30	$\boxed{0}$
$\boxed{0}$	25	30	40	0
15	5	$\boxed{0}$	15	5

(Nullen pro Spalte)

Bagger \ Bau	A	B	C	D
1	90	$\boxed{75}$	75	80
2	35	85	55	$\boxed{65}$
3	125	95	90	105
4	$\boxed{45}$	110	95	115
5	55	85	$\boxed{60}$	85

Bagger 1 ↔ Baustelle B

Bagger 2 → Baustelle D

⋮