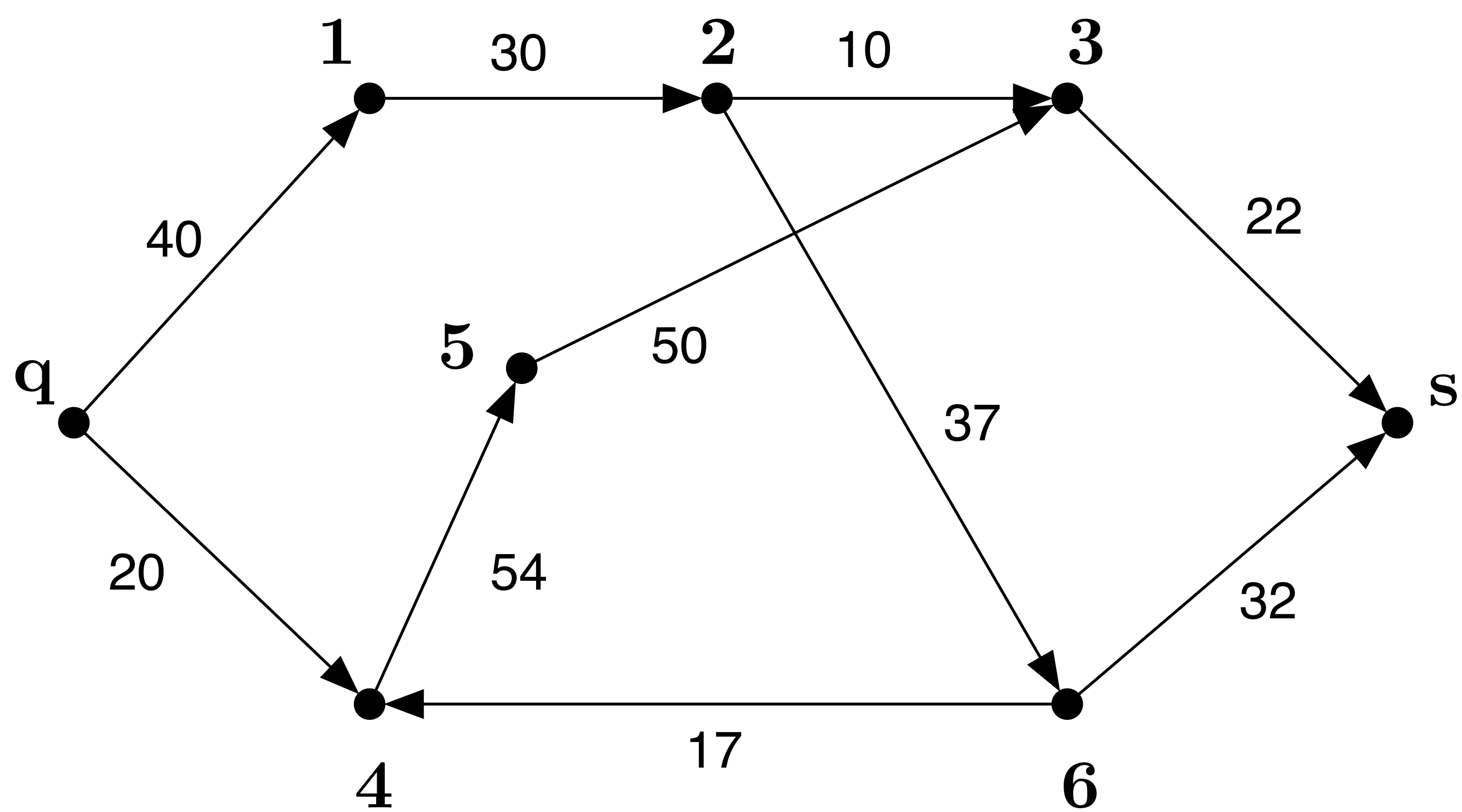
Algorithmische Graphentheorie

Teil 5

Mehr zu algorithmischer Graphentheorie: *Netzwerke, Flüsse, etc*.

Flüsse in Netzwerken



Von Quelle q (ohne eingehende Kanten) zur Senke s (ohne ausgehende Kanten) [manchmal ohne diese Einschränkungen!] soll im kantengewichteten ($g(e)$), zyklenfreien Digraphen oder Multigraphen [mehr als eine Kante zwischen gleichen Knoten mit verschiedenen Gewichten erlaubt] ($0 < g(e) \in \mathbb{R}$) die maximale Menge der je Zeiteinheit von q nach s fließenden Menge von Waren (z.B. Rohöl in Pipelines mit maximalen Transport-kapazitäten) berechnet, bzw. die Auslastung eines Teilstückes bestimmt werden. (In den Knoten wird nichts nichts "aufbewahrt" oder "entfernt").

q□ s - Fluss

Definition:

Eine Funktion $f{:}E \longrightarrow \mathbb{R}$ heißt $q-s$-Fluss, wenn:

- $0 \leq f(e) \leq g(e)$
- für jeden Knoten $v \in V \setminus \{q,s\}$ gilt Flussgleichgewicht:

$$\sum_{e=(i,v)\in E} f(e) = \sum_{e=(v,j)\in E} f(e)$$

- der Gesamtfluss des Netzwerks ist:

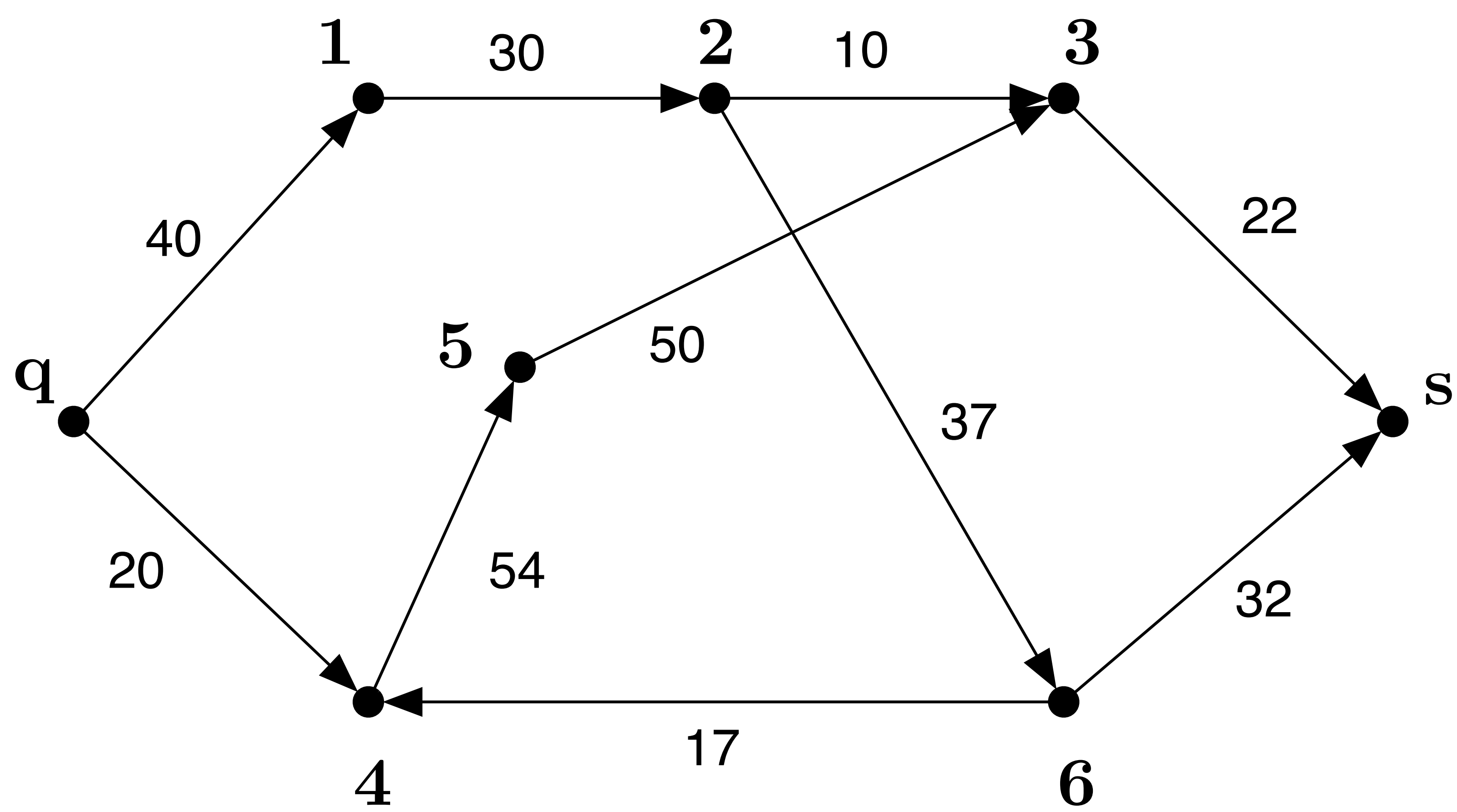
$$|f| := \sum_{e=(q,j)\in E} f(e)$$

Frage:

Was wäre angemessene Definition für “Gesamtfluss des Netzwerks”, wenn Quelle und Senke beliebige Knoten sein dürften?

$$\boxed{|f| := \sum_{e=(q,i)\in V} f(e) - \sum_{e=(j,q)\in V} f(e)}$$

Welche Flüsse in diesem Netzwerk?



$f(q,1) = 30$, weil Fluss durch Knoten 1 maximal 30.
$f(1,2) = 30$
$f(2,6) = 20,\ f(2,3) = 10$
$f(6,s) = 20,\ f(3,s) = 10$
$f(q,4) = 0,\ f(4,5) = 0,\ f(6,4) = 0,$ und $f(5,3) = 0$ warum?

Dies ist ein möglicher Fluss, aber sicher nicht der Maximale!

Schnitt und Fluss im Netzwerk G

Definition:

Zu einem $q-s$-Netzwerk G ist ein ***Schnitt*** $S := (V_1, V_2)$ eine Partition von $V := V_1 \cup V_2$ mit $q \in V_1$ und $s \in V_2$.

Oft wird auch die Menge $E(V_1,V_2) := \{(u,v) \mid u \in V_1 \text{ und } v \in V_2\} \subset E$ als Schnitt bezeichnet. Jedoch auch $E(V_2,V_1) := \{(v,u) \mid v \in V_2 \text{ und } u \in V_1\}$ sind Kanten zwischen den Teilmengen der Partition!

Die Kapazität eines Schnittes S ist $k(V_1,V_2) := \sum g(e)$ mit $e \in E(V_1,V_2)$.

Ist f ein Fluss im Netzwerk, so ist der ***Fluss eines Schnittes*** $S = (V_1,V_2)$ definiert durch: $f(V_1,V_2) := \sum g(e)$ mit $e \in E(V_1,V_2)$.

Satz:

Wenn f Fluss eines Netzwerkes ist, so gilt für jeden Schnitt $S=(V_1,V_2)$,

$$|f| = f(V_1,V_2) - f(V_2,V_1)$$

und

$$|f| \leq \min\Big(k(V_1,V_2) \mid (V_1,V_2) \text{ ist Schnitt}\Big)$$

Beweis:

nach den Flussgleichgewichten folgt:

$$\begin{aligned}
|f| &= \sum_{u\in V} \left\{ \sum_{e=(u,v)\in E} f(e) - \sum_{e=(v,u)\in E} f(e) \right\} \\
&= \sum_{\substack{(i,j)\in E \\ i\in V_1, j\in V_2}} + \sum_{\substack{j,j\in V \\ e=(i,j)\in E}} - \sum_{\substack{j,j\in V \\ e=(i,j)\in E}} - \sum_{\substack{(j,i)\in E \\ j\in V_2, i\in V_1}} \\
&= f(V_1,V_2) - f(V_2,V_1)
\end{aligned}$$

Gesamt-Fluss und Schnitt-Fluss 2

Da für alle Kanten e immer $0 \leq f(e) \leq g(e)$ gilt, folgt

$$|f| = f(V_1,V_2) - f(V_2,V_1) \leq f(V_1,V_2) \leq g(V_1,V_2)$$

D.h., durch ein Netzwerk kann maximal so viel fließen, wie seine engste Stelle durchlässt!

Der Satz von Ford und Fulkerson sagt:

Der maximale Fluss ist identisch zu dem minimalen Schnitt!

Der Satz von Ford und Fulkerson

Satz:
Der Wert eines maximalen Flusses in einem Netzwerk ist gleich der minimalen Kapazität eines Schnittes.

Das Verfahren von Ford/Fulkerson zum Finden eines maximalen Flusses ist leider im schlechtesten Fall exponentiell.
Das Max-flow Min-cut Problem kann vom Dualitätstheorem der linearen Programmierung hergeleitet werden.
Auf Edmonds und Karp (1972) geht ein polynomielles Verfahren zurück, dass einen maximalen Fluss in $O(nm^2)$ Schritten.
Das Verfahren von Dinic (1970) arbeitet in $O(n^2m)$ Schritten.

Der Beweis zum Satz von Ford und Fulkerson

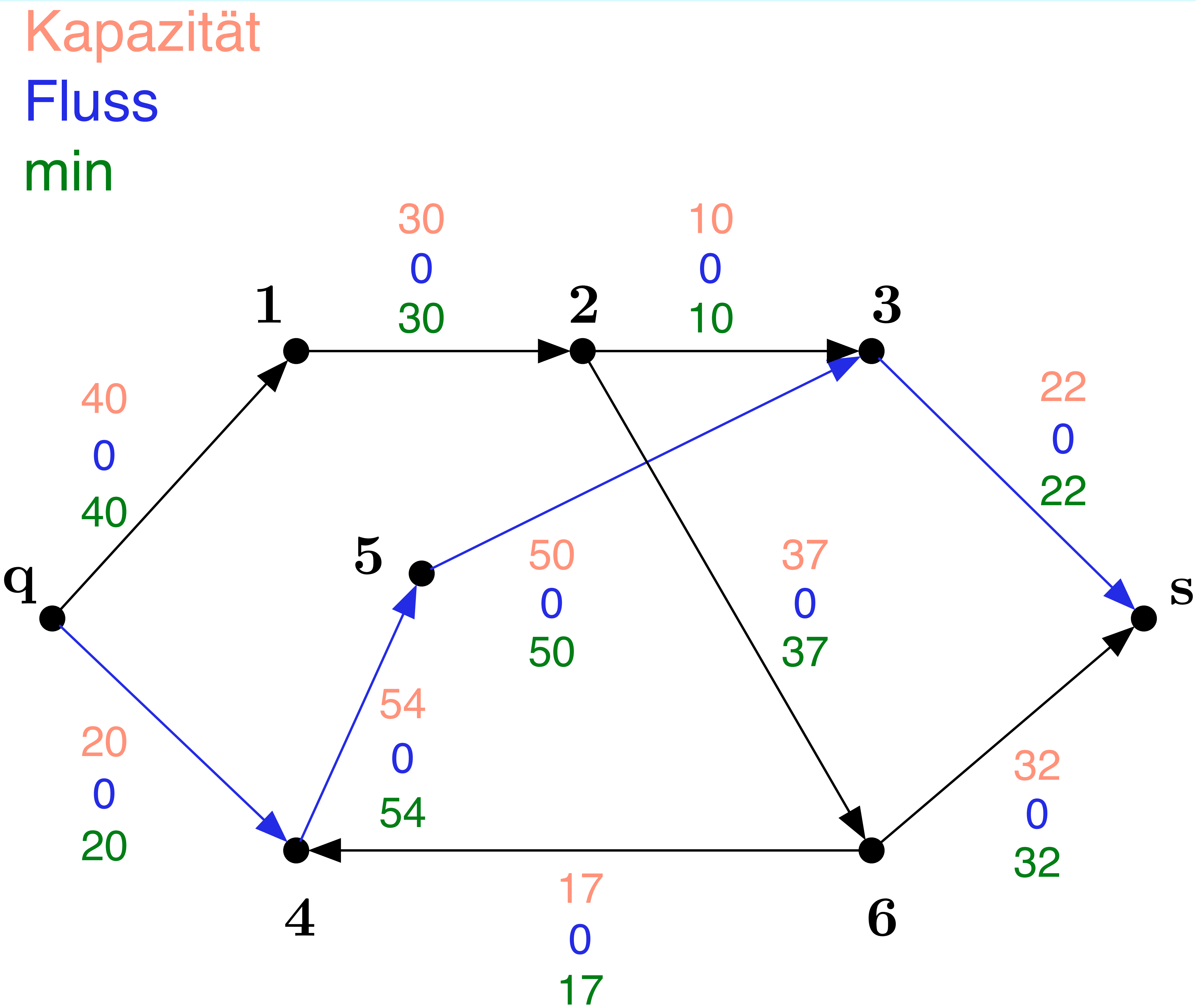
Sei $G := (V, E, g)$ ein Digraph mit Kapazitätsfunktion $g : E \to \mathbb{N}$.
zu Fluss f
Konstruktion von Folge ganzzahliger Flüsse f_i mit $|f_0| < |f_1| < |f_2| < \cdots,$ die bei f_m abbrechen muss! (Warum?) Also ist der letzte Fluss f_m maximal.

Beginne: $f_0 := 0$

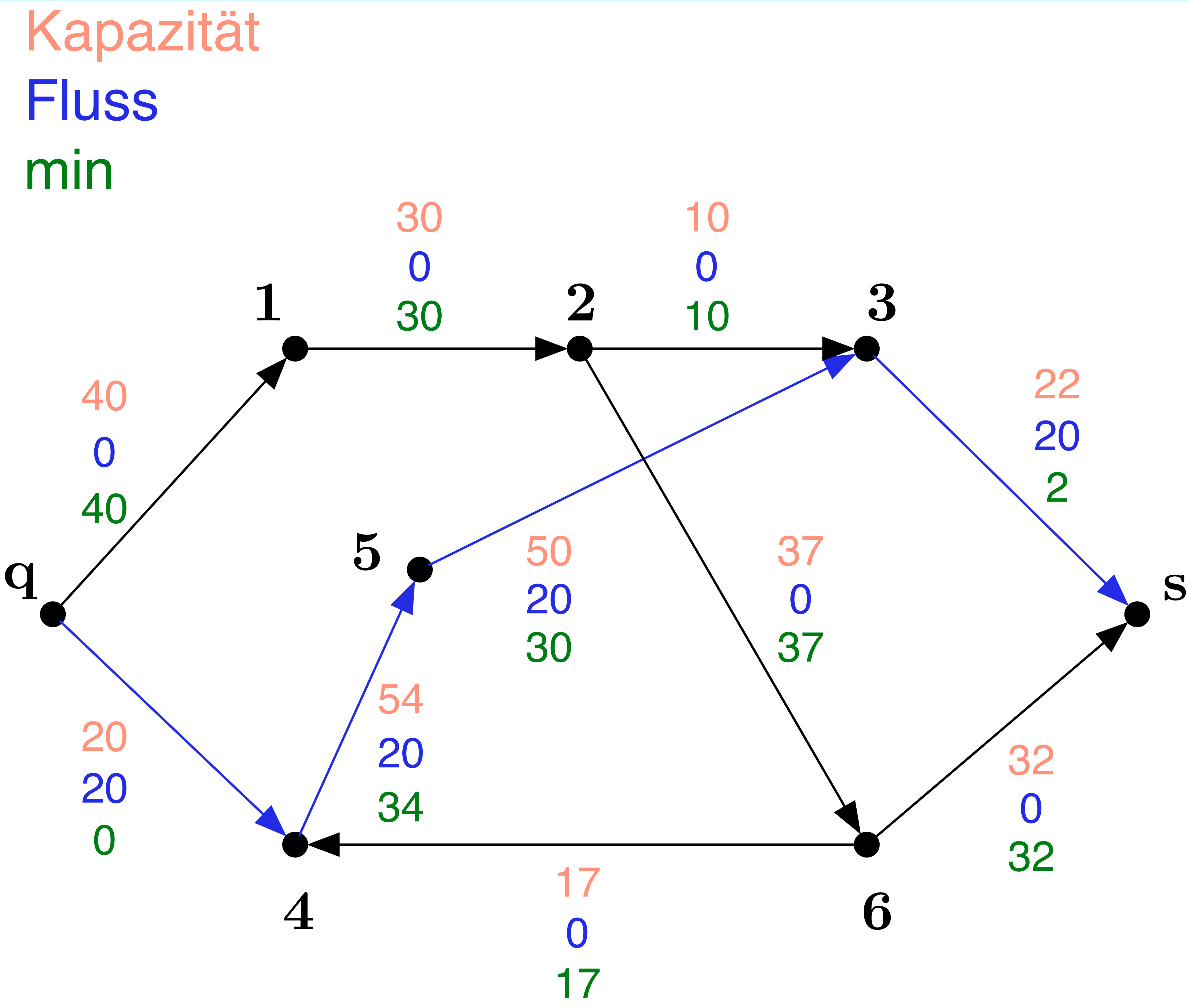
Zu einem Fluss f_n und einem $q-s$ Pfad sei

$$\delta := \min(g(e) - f_n(e) |\ e \text{ ist Kante im } q-s \text{ Pfad})$$

Der Fluss f_n wird auf dem $q-s$ Pfad zu f_{n+1} erweitert, indem an jeder Kante des Pfades der Fluss um δ erhöht wird. Damit wird $|f_n| < |f_{n+1}|$. Dies wird so lange gemacht, bis keine dieser Erweiterungen mehr möglich sind!


Kapazität
Fluss
min
1
2
3
q
5
s
4
6
30
0
30
10
0
10
40
0
40
22
0
22
50
0
50
37
0
37
54
0
54
20
0
20
32
0
32
17
0
17

Flusserweiterung am Beispiel

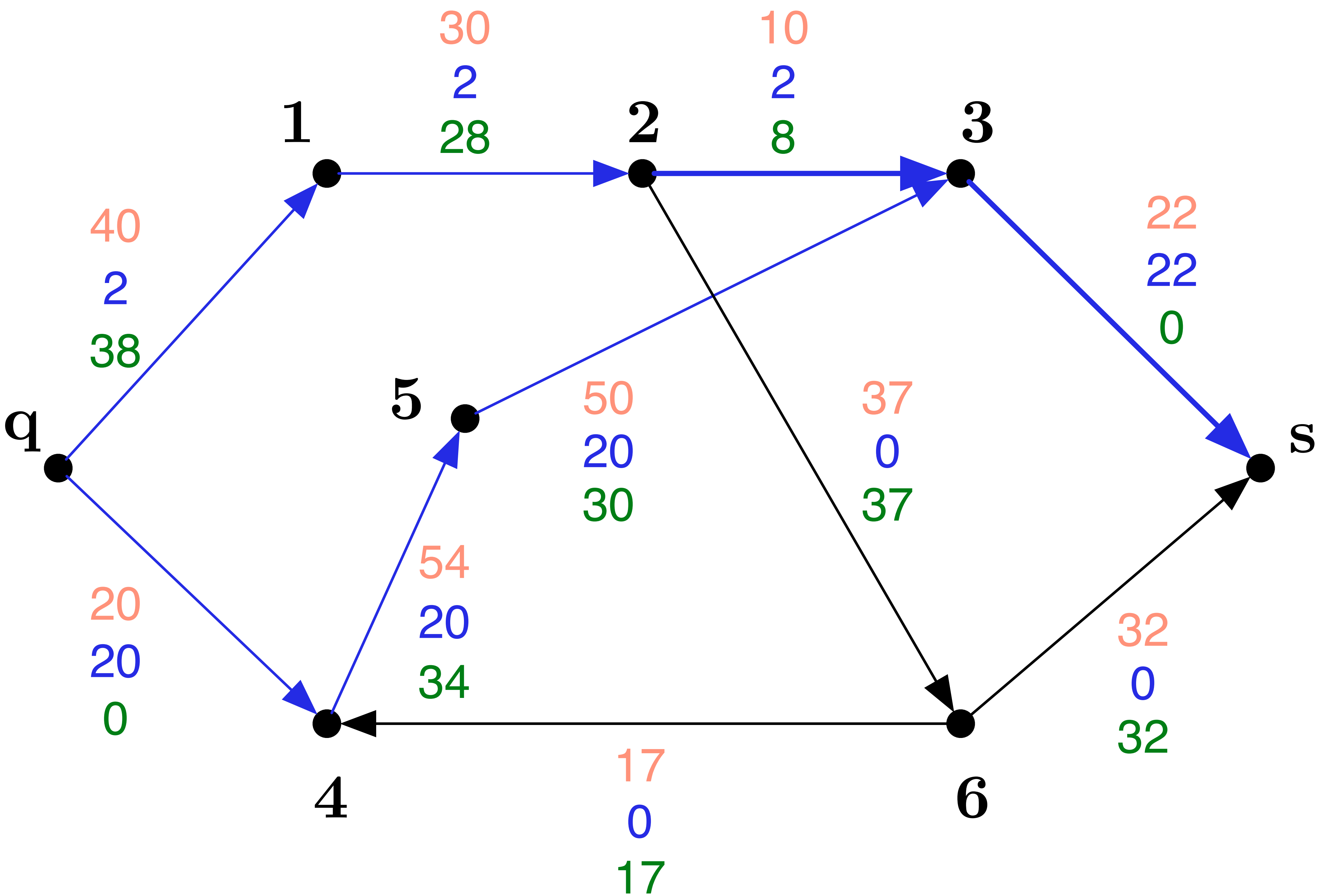


Flusserweiterung am Beispiel

Kapazität

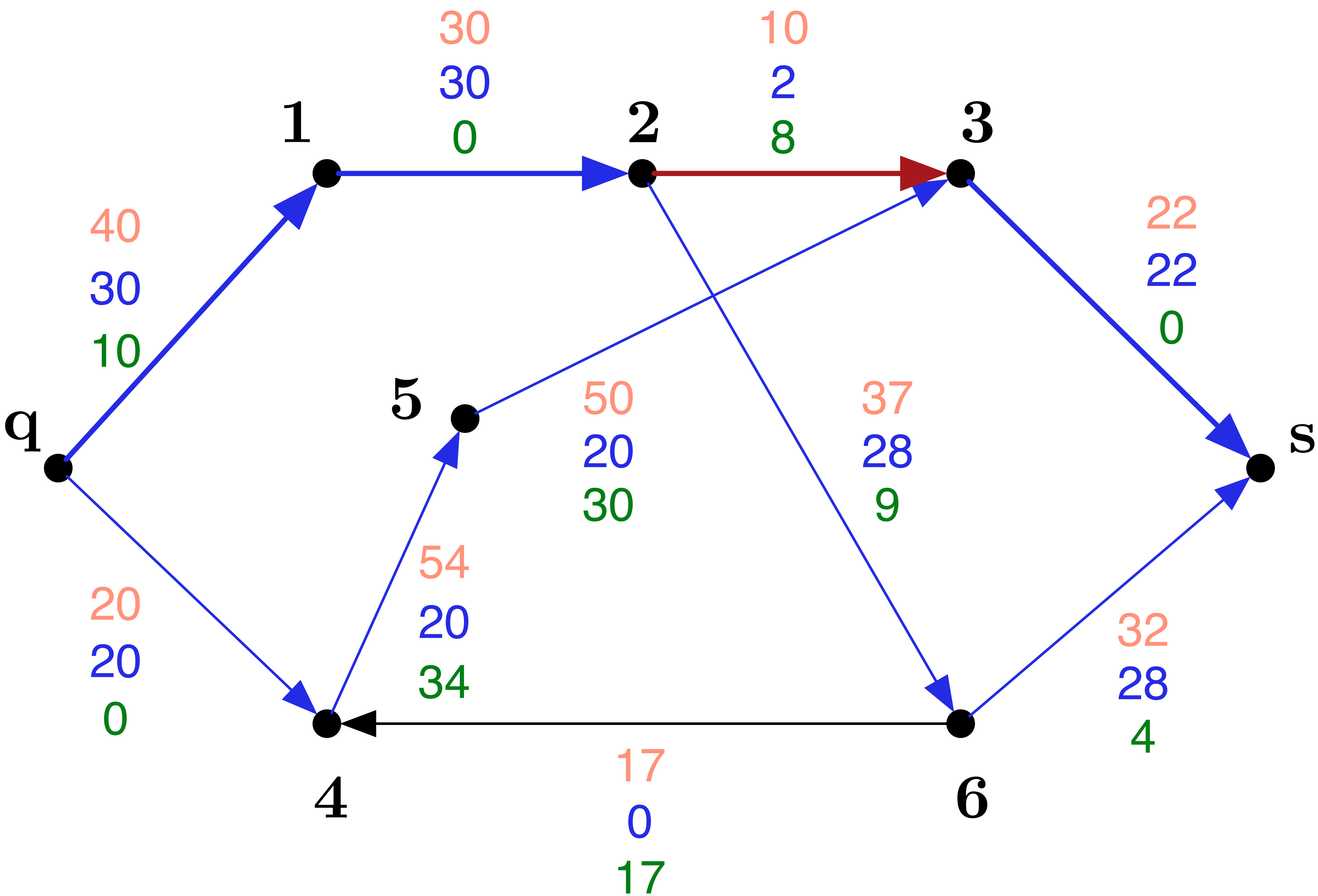
Fluss

min



Flusserweiterung am Beispiel

Kapazität
Fluss
min



Beweis *informal beenden*

Jeder Fluss f_n ist kleiner als jeder Schnitt und der maximale Fluss also kleiner als der minimale Schnitt.
Da für einen Fluss, alle Schnitte gleiche Kapazität haben (Kirchhoff'sches Gesetz zu Flussgleichgewichten) und der minimale Schnitt dort erreicht wird, wo die Flusserweiterung durch Ausschöpfen der Kapazität einer Kante ein Ende erreicht wurde, gilt nicht nur Fluss ≤ Schnitt, sondern auch

maximaler Fluss = minimaler Schnitt.

Der diesem Gedanken folgende formale Beweis ist zu finden in:
R. Diestel: Graphentheorie [Springer, 1996/2000]

Der Satz von Menger [1927] ist wohl die berühmteste “Folgerung”, geht aber nicht so einfach wie der direkte Beweis (vergl. [Diestel]):

Satz:
Sei $G := (V, E)$ ein ungerichteter Graph und $A, B \subseteq V$, dann hat die kleinste A und B trennende Eckenmenge die selbe Mächtigkeit, wie die größte Menge von disjunkten $A - B$ Wegen.

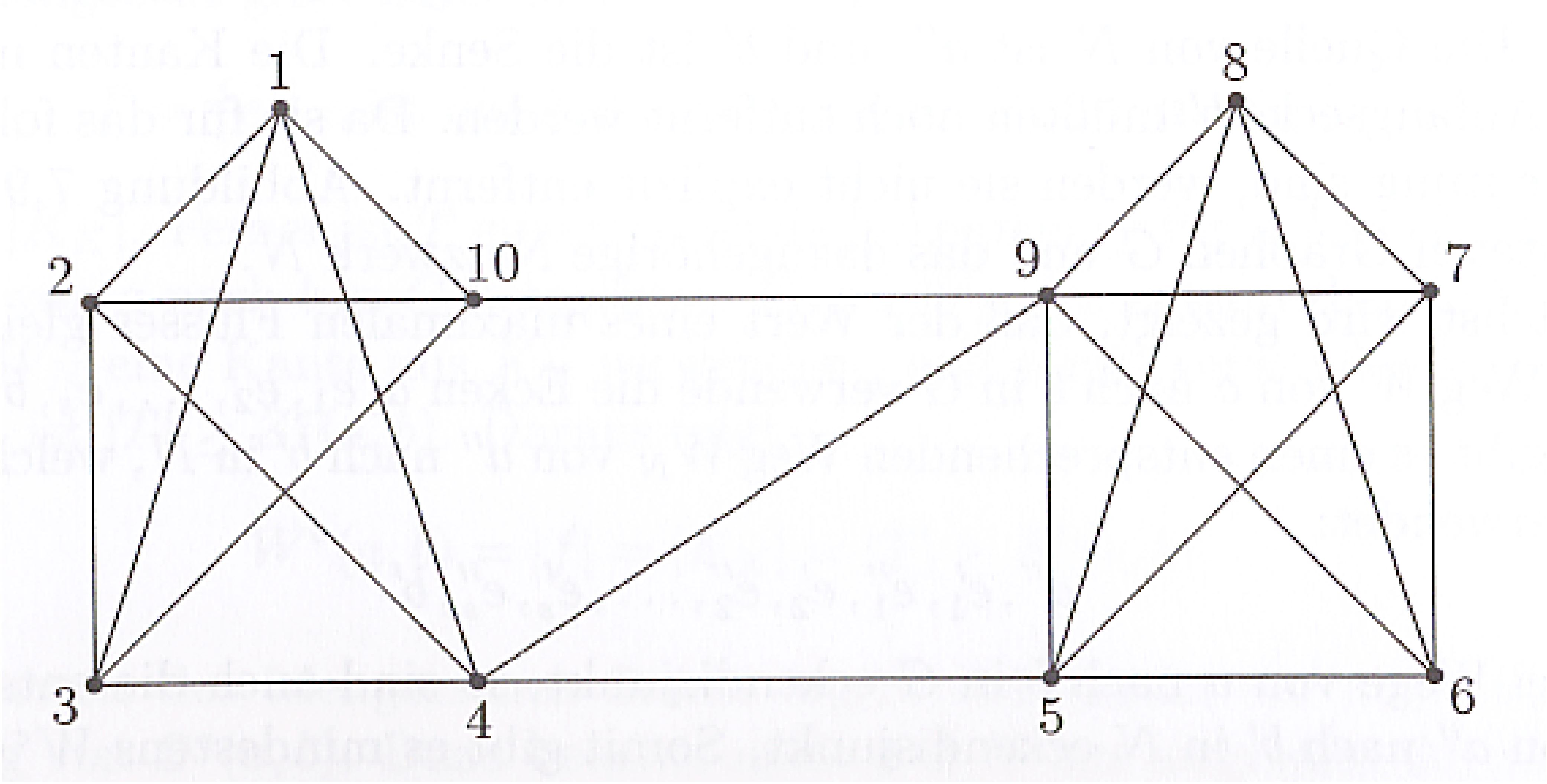
T ist eine, die Knoten $a, b \in V$ trennende Eckenmenge, wenn jeder Weg von a nach b mindestens einen Knoten aus T durchläuft.

T ist eine, die Knoten $a, b \in V$ trennende Kantenmenge, wenn jeder Weg von a nach b mindestens eine Kante aus benutzt durchläuft.

Trennende Eckenmengen

Wenn die Knoten die Relaisstationen eines Netzwerks symbolisieren, so ist die Mächtigkeit einer minimnale trennenden Eckenmenge ein Maß für die Ausfallsicherheit.

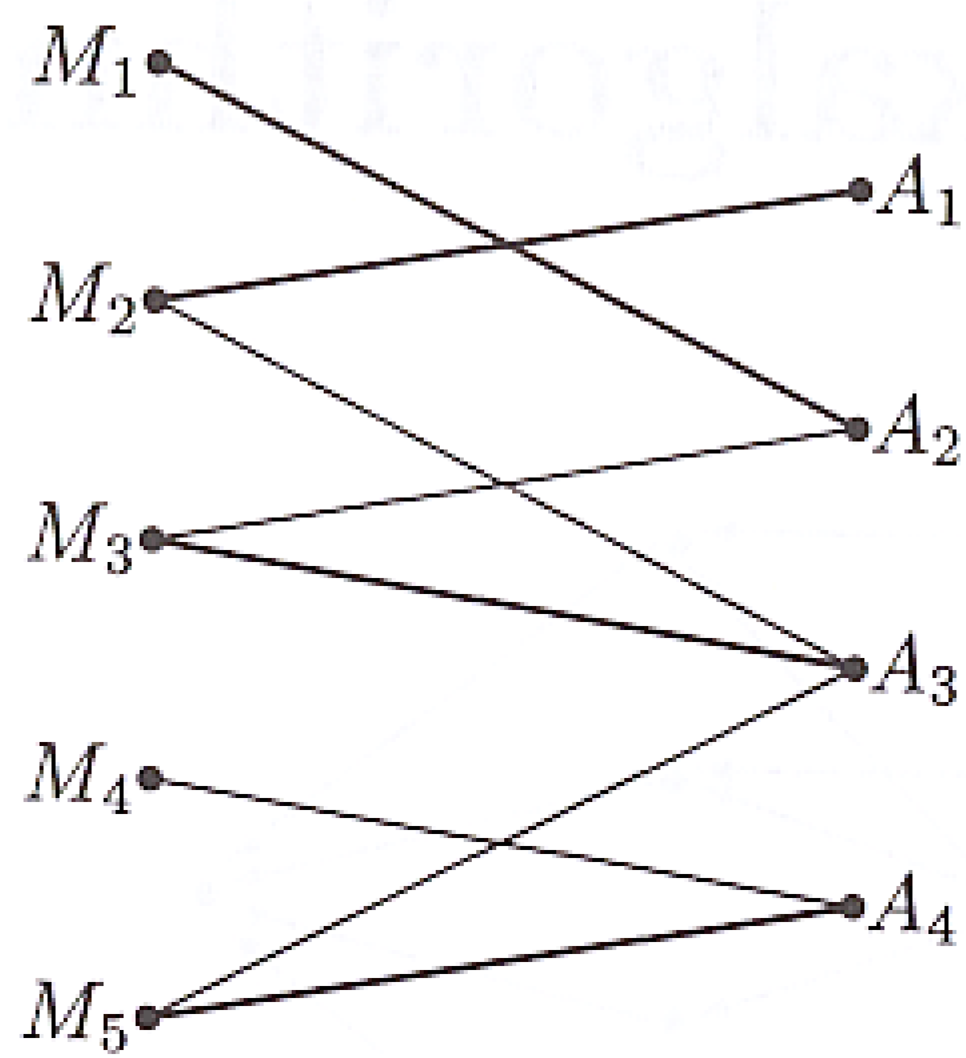
Der Beweis ist gut im Buch von V.Turau zu finden.



Maximalen Zuordnungen (Paarungen)

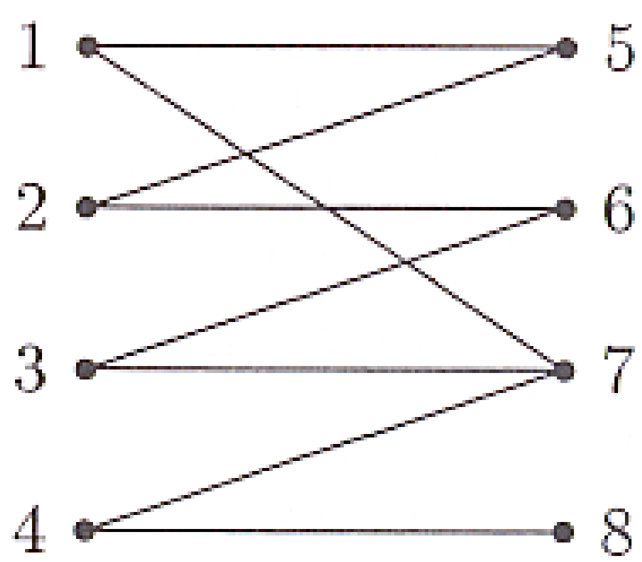
Ein anderes Problem ist das der Maximalen Zuordnungen, ein Problem aus dem Operations Research:

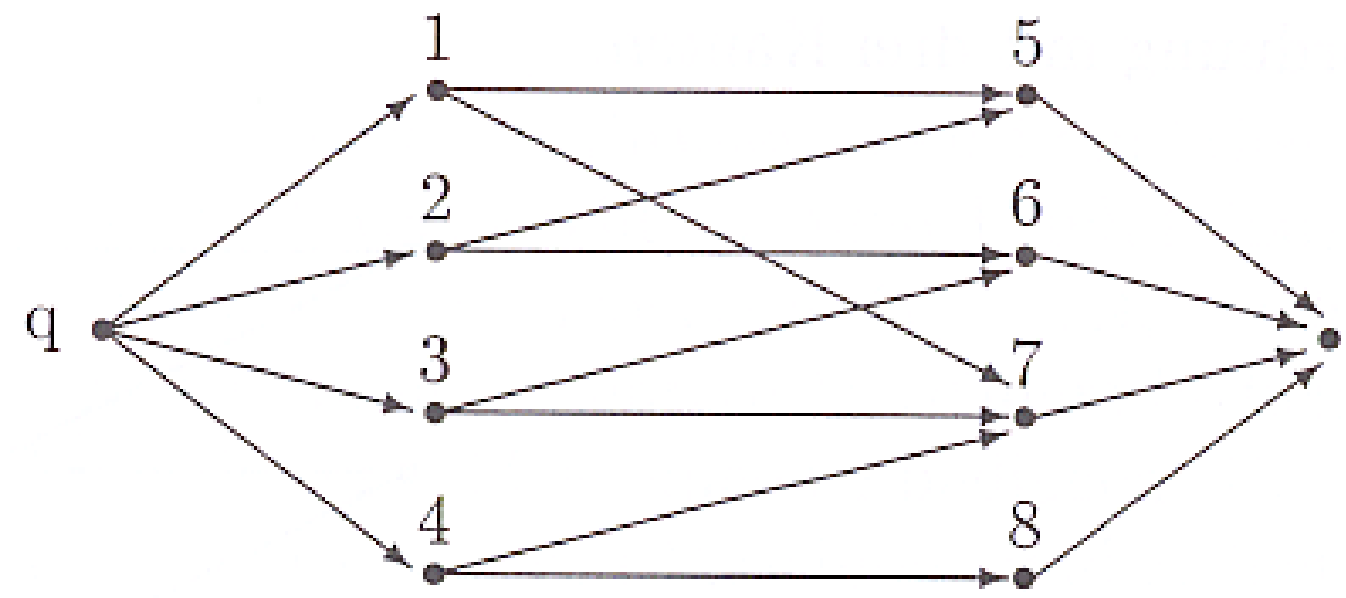
Ein Arbeiter A_i kann stets nur eine Maschine M_i bedienen, aber kennt sich nicht mit allen aus! Wie können möglichst viele Maschinen bedient werden?
Ein Verfahren von Hopcroft und Karp arbeitet in $O(\sqrt{n}m)$. Ein für dichte Graphen besseres Verfahren von Alt et. al. arbeitet in $O(n^{1,5}\sqrt{m}/\ell ogn)$.



Maximale Zuordnung (*matching*)

Sei $G := (V, E)$ ein bipartiter Graph. Eine Zuordnung ist eine Auswahl von Kanten, die paarweise keine gemeinsamen Knoten haben. Eine Zuordnung heißt maximal, wenn es keine mit mehr Kanten gibt.





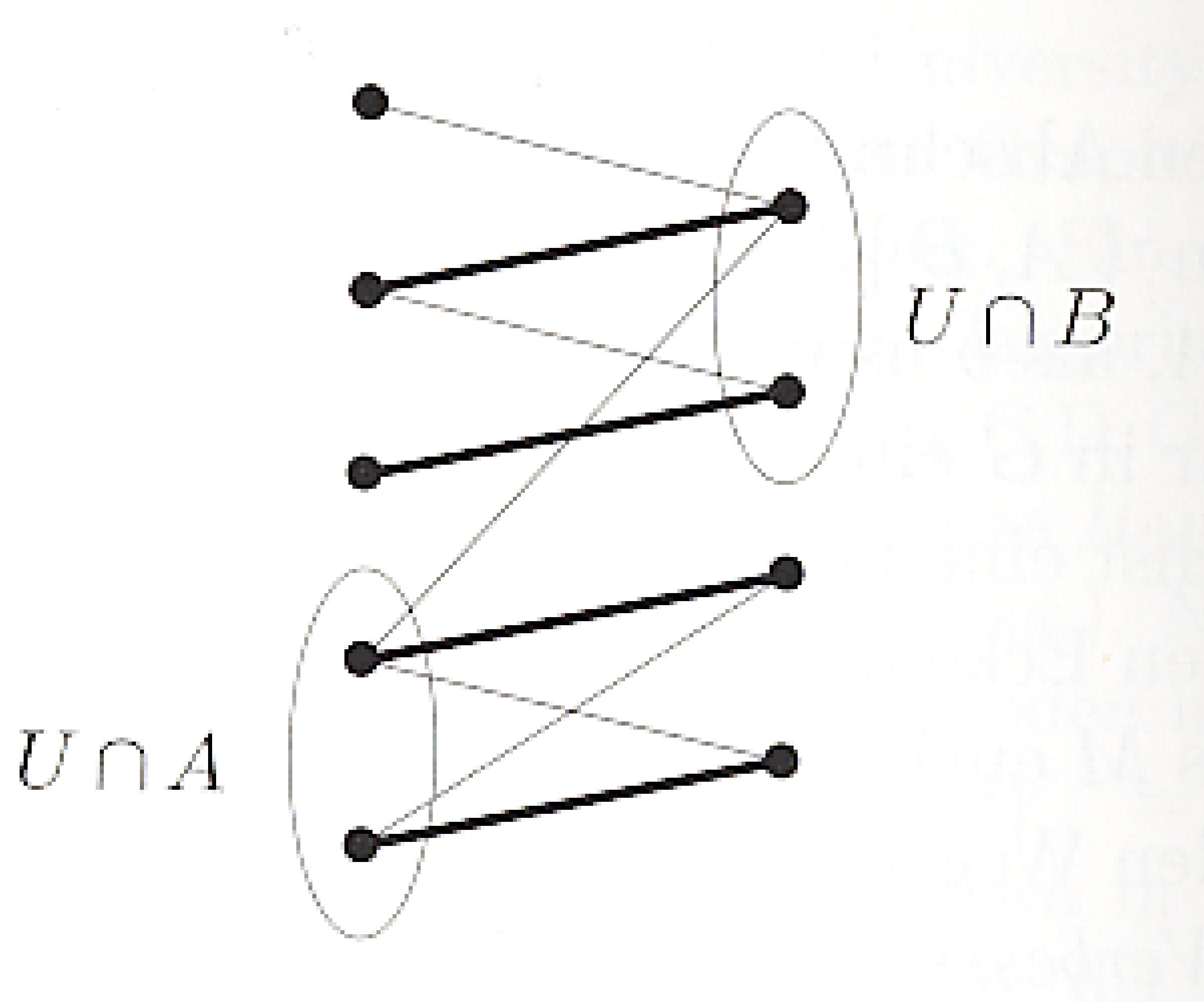
Zu $G := (V, E)$ wird einfaches Netzwerk konstruiert. Dann folgt:

Satz:
Die maximale Zuordnung entspricht dem maximalen Fluss im zugeordneten Netzwerk.

Eckenüberdeckung und Satz von König (1931)

Eine Eckenüberdeckung in $G:=(V,E)$ ist eine Teilmenge $U \subseteq V$ derart, dass jede Kante mit einer Ecke aus U inzident ist.

Satz [König 1931]:
Die größte Mächtigkeit einer Paarung entspricht der geringsten Mächtigkeit einer Eckenüberdeckung



Man folgert daraus, dass für eine Paarung der Teilmenge A (jede Ecke ist an einer Kante beteiligt) notwendig ist, dass für alle $S \subseteq A$: $|N(S)| \geq |S|$.

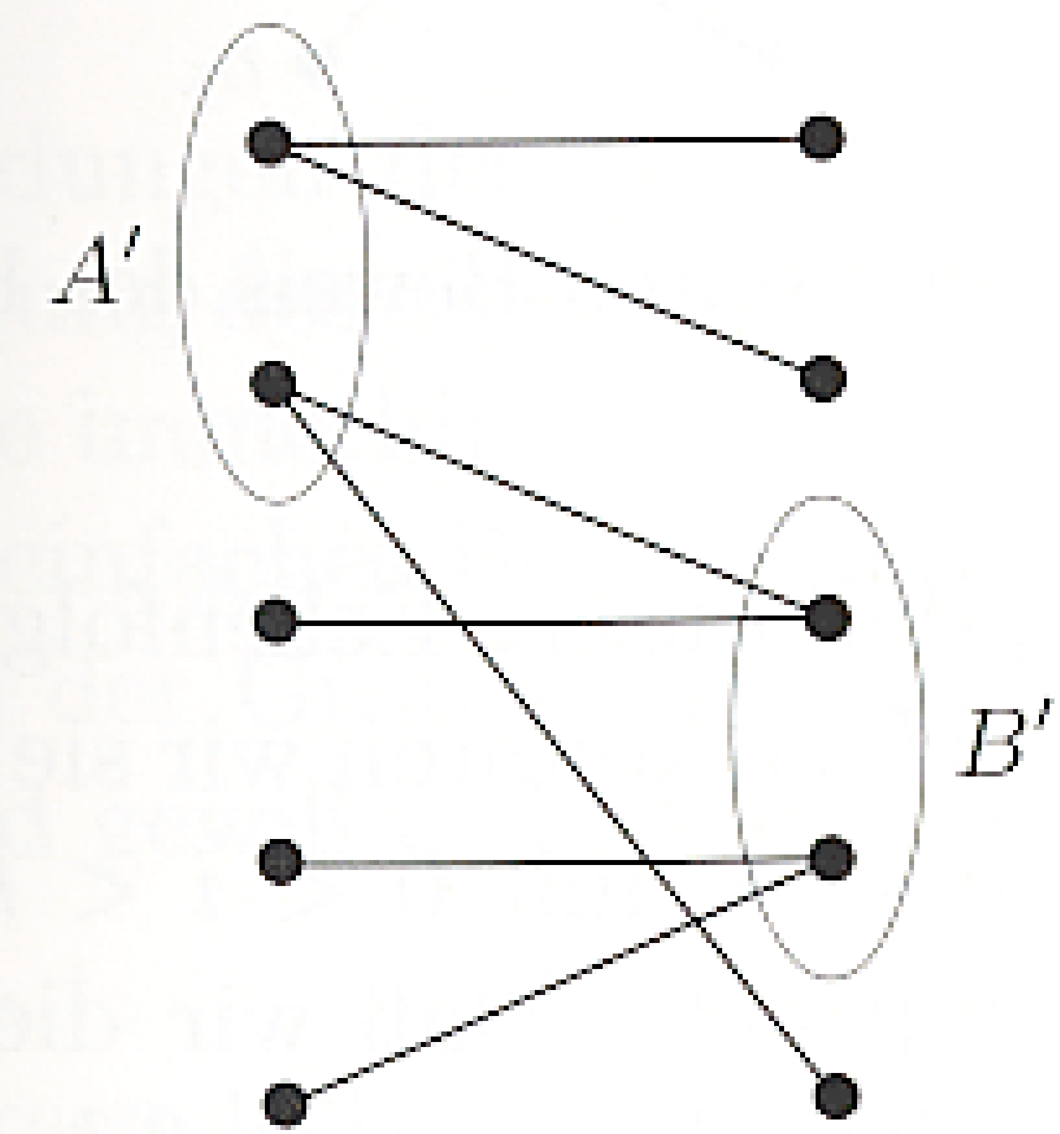
Der Satz von Hall (Heiratssatz)

Satz [Hall 1935]:

Sei $G := (V, E)$ ein bipartiter Graph mit Eckenpartition $E = A \cup B$. G enthält genau dann eine Paarung von A, wenn für alle $S \subseteq A$ die Heiratsbedingung $|N(S)| \geq |S|$ gilt!

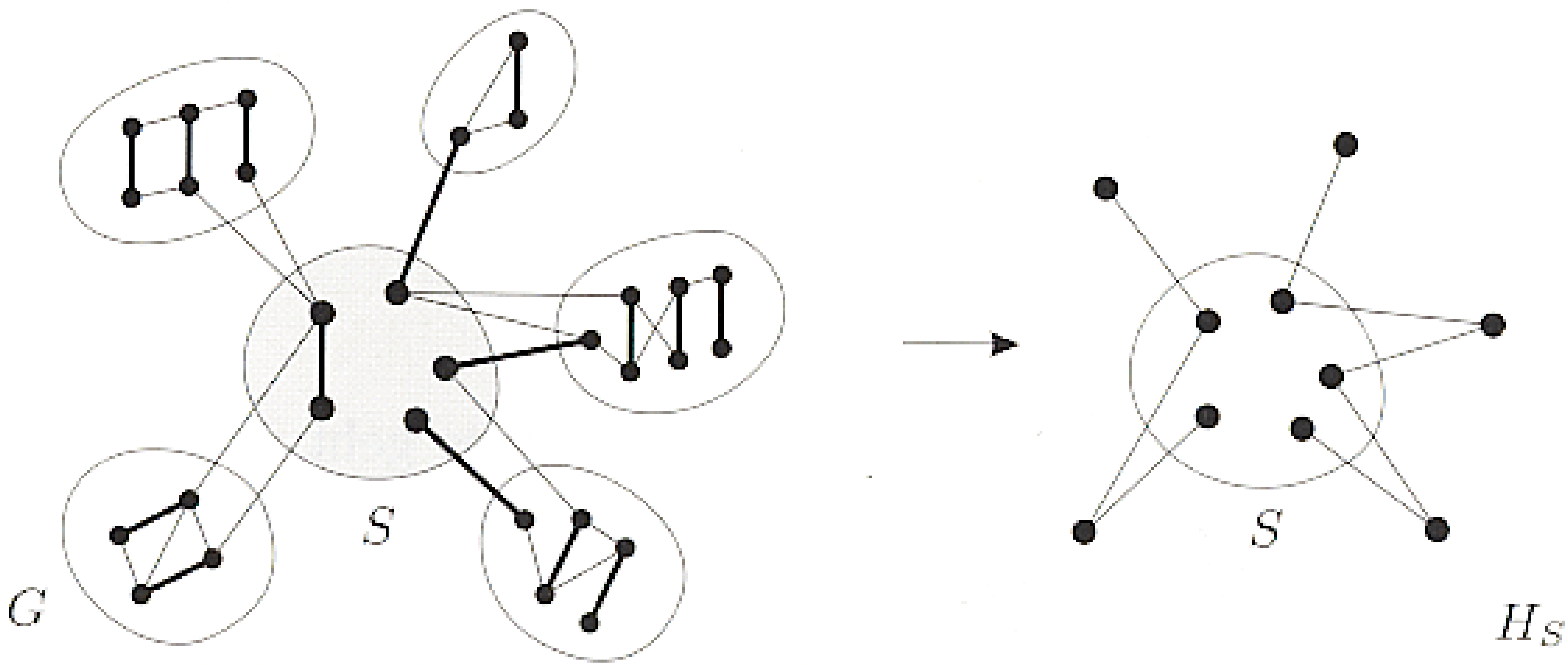
Beweis:

Enthält G keine Paarung von A, so hat G nach dem Satz von König eine Ecken-überdeckung U mit weniger als $|A|$ Ecken, z.B. $U := A^* \cup B^*$ mit $A^* \subseteq A$, $B^* \subseteq B$ und jetzt: $|A^*| + |B^*| = |U| < |A|$. Also folgt: $|B^*| < |A| - |A^*| = |A \backslash A^*|$. Nach Def. von U hat G *keine* Kante zwischen $A \backslash A^*$ und $B \backslash B^*$. Mithin: $N(A \backslash A^*) \leq |B^*| < |A| - |A^*|$ und für $S := A \backslash A^*$ ist die Heiratsbedingung nicht erfüllt.



Zuordnungsproblem auf beliebigen Graphen?

Kann man das Zuordnungsproblem auf Graphen erweitern, die nicht bipartit sind? Ja, und die Algorithmen sind komplizerter, aber die Besten im *worst case* genauso wie im bipartiten Fall!



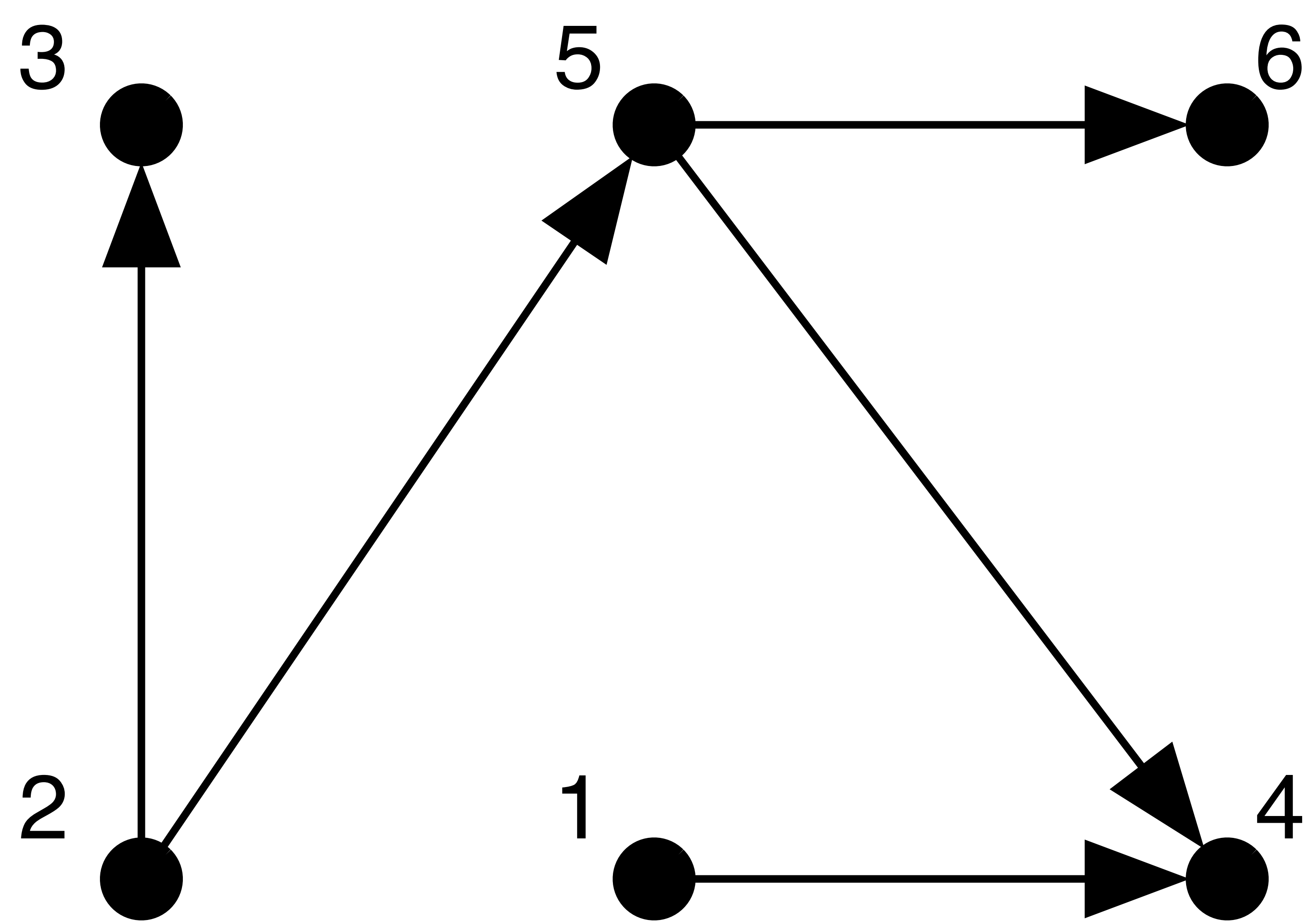
Satz [Dilworth 1950]:

Sei $G\!:=\!(M,\leq)$ eine partiell geordnete endliche Menge (*finite poset*), [d.h. $\leq$ ist reflexiv, transitiv und antisymmetrisch].

Dann ist die kleinste Zahl $k(M)$ von Ketten einer Zerlegung von M gleich der größten Anzahl $d(M)$ von paarweise unvergleichbaren Elementen!

$k(M)\!=\!3$

$d(M)\!=\!3$



Beziehungen der Sätze

Es lassen sich Äquivalenzen und Beweisketten zwischen den wichtigen graphentheoretischen Sätzen verfolgen [Selecta Mathematik I, Konrad Jakobs, Springer (1969)].

Dieses Gebiet nennt man nunmehr *matching theory*, und wichtige Sätze und deren Zusammenhänge werden hier genannt:

Schnitt-Fluss Theorem [Ford/Fulkerson (1956)]

$\Rightarrow$

Satz von König [König (1936)]

$\Leftrightarrow$

Der Heiratssatz [Miller (1910), König (1916),Weyl (1949)]

$\Leftrightarrow$

Satz von Dilworth [Dilworth (1950)]

$\Leftrightarrow$

Dualitäts Theorem der linearen Programmierung
[Fourier (1826), Dantzig (1940)]

Die wichtigsten Sätze und deren Zusammenhänge werden sehr gut dargestellt in:

A. Schrijver, Theory of Linear and Integer Programming, Wiley (1994)
H. Möller, Algorithmische Lineare Algebra, Vieweg (1997)

Fundamentasatz der linearen Ungleichungen:
[Farkas (1894), Caratheodory (1911), Weyl (1935)]
Seien $a_1, a_2, \ \dots \ , a_m, b \in \mathbb{R}^n$, dann gilt:
entweder ist b Linearkombination der $a_1, a_2, \dots, a_m$,
oder es gibt Hyperebene $\{x \,|\, c \cdot x = 0\}$ mit $t-1$ linear unabhängigen Voktoren aus $\{a_1, a_2, \dots, a_m\}$, so dass $c \cdot b < 0$ und $c \cdot a_1 \geq 0, \dots, c \cdot a_m \geq 0$,
wobei $t := \ rang\{a_1, a_2, \dots, a_m, b\}$
$(rang(M)$ ist maximale Zahl linear unabhängiger Elemente in M)

Polyeder (*polyhedron*), Polytop, Kegel (*cone*)

Ein (konvexer) Kegel (*convex cone*) ist eine Menge C mit $\{\alpha x + \beta y \mid \alpha, \beta \geq 0\} \subseteq C$ für alle $x, y \in C$.
$C\{x_1, x_2, \ldots, x_m\} := \{\alpha_1 \cdot x_1 + \alpha_2 \cdot x_2 + \ldots + \alpha_m \cdot x_m\}$ ist der von $x_1, \ldots, x_m$ erzeugte Kegel.

Ein Kegel C ist ein Polyeder (*polyhedral cone*), wenn $C = \{x \mid A \cdot x \leq 0\}$.

P ist ein Polyeder, falls $P = \{x \mid A \cdot x \leq b\}$.

Ein Polytop ist ein beschränkter Kegel.

Aus dem Fundamentalsatz folgt

Satz:[Farkas (1898, 1902), Minkowski (1896), Weyl (1935)]

Ein konvexer Kegel ist genau dann ein Polyeder, wenn er endlich erzeugt ist!

Farkas Lemma und spezielle Variante

Lemma (Farkas Lemma)**:**
Es seien $A \in \mathbb{R}^{m \times n}$ und $b \in \mathbb{R}^m$.
Dann existiert ein Vektor $x \geq 0$ mit $A \cdot x = b$ genau dann, wenn
$y' \cdot b \geq 0$ für jeden Zeilenvektor y' mit $y' \cdot A \geq 0$.

Lemma (Variante)**:**
Es seien $A \in \mathbb{R}^{m \times n}$ und $b \in \mathbb{R}^m$.
Dann existiert ein Vektor x mit $A \cdot x \leq b$ genau dann, wenn
$y' \cdot b \geq 0$ für jeden Zeilenvektor y' mit $y' \cdot A = 0$.

Farkas Lemma (viele Varianten, hier für rationale Zahlen formuliert)

Seien A Matrix und b ein Vektor über den rationalen Zahlen, dann ist in jeder Zeile der Tabelle genau eine der beiden Ungleichungen in $\mathbb{Q}$ lösbar!

1.	$A\cdot x<0$	$y\cdot A=0,\ y\geq 0,\ y\neq 0$
2.	$A\cdot x<0,\ x\geq 0$	$y\cdot A\geq 0,\ y\geq 0,\ y\neq 0$
3.	$A\cdot x\leq 0,\ A\cdot x\neq 0$	$y\cdot A=0,\ y>0$
4.	$A\cdot x\leq 0,\ A\cdot x\neq 0,\ x\geq 0$	$y\cdot A\geq 0,\ y>0$
5.	$A\cdot x=b,\ x\geq 0$	$y\cdot A\geq 0,\ y\cdot b<0$
6.	$A\cdot x<b$	$y\cdot A=0,\ y\cdot b\leq 0,\ y\neq 0$
7.	$A\cdot x<b,\ x\geq 0$	$y\cdot A\geq 0,\ y\cdot b\leq 0,\ y\geq 0,\ y\neq 0$
8.	$A\cdot x\leq b$	$y\cdot A=0,\ y\cdot b<0,\ y\geq 0$
9.	$A\cdot x\leq b,\ x\geq 0$	$y\cdot A\geq 0,\ y\cdot b<0,\ y\geq 0$

Farkas Lemma folgt aus dem Fundamentalsatz der linearen Ungleichungen!

Dualität von linearen Optimierungsaufgaben

Definition:

Es seien $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$. Die Problemstellungen

(1) finde x mit $A \cdot x \leq b$, $x \geq 0$, so dass $c' \cdot x$ maximal ist

und

(2) finde y mit $A \cdot y \geq c$, $y \geq 0$, so dass $b' \cdot x$ minimal ist

heißen ***dual*** zu einander. Die zuerst gegebene Fragestellung nennt man ***primal.***

Das Konzept der Dualität geht zurück auf
[von Neumann (1947), Gale, Kuhn und Tucker (1951)].

Wieder als Konsequenz des Fundamentalsatzes der linearen Ungleichungen ergibt sich das Dualitäts-Theorem der linearen Programmierung.

Dualitätssatz der linearen Programmierung

Satz:

Es seien $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$. Dann gilt:

$$max\{c' \cdot x \,|\, A \cdot x \leq b\} = min\{y' \cdot b \,|\; y' \cdot A = c', y \geq 0\},$$

sofern diese Mengen nicht leer sind.

Beweisskizze:

Falls $A \cdot x \leq b$ und $y' \cdot A = c, y \geq 0$ so folgt $c' \cdot x = y' \cdot A \cdot x \leq y' \cdot b$ und $max \leq min$, falls es endliche Optima gibt. Es muss also nur noch gezeigt werden, dass es x, y gibt mit: $A \cdot x \leq b$ und $y' \cdot A = c, y \geq 0$ und $c' \cdot x \geq y' \cdot b$. Letzteres läßt sich mit Hilfe der Variante von Farkas Lemma zeigen (siehe Schrijver)

Auch der Dualitätssatz besitzt Varianten

Satz:

Es seien $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$. Dann gilt:

$$max\{c' \cdot x \,|\, A \cdot x \leq b, x \geq 0\} = min\{y' \cdot b \,|\, y' \cdot A \geq c', y \geq 0\},$$

$$max\{c' \cdot x \,|\, A \cdot x = b, x \geq 0\} = min\{y' \cdot b \,|\, y' \cdot A \geq c'\}$$

sofern diese Mengen nicht leer sind.

Um zu dem Schnitt/Fluss Theorem von Ford/Fulkerson zu kommen müssen die vorhandenen Daten zu Ungleichungen umgeschrieben werden:
Für einen Knoten $v \in V$ eines gerichteten Graphen $G := (V, E)$ mit Quelle q und Senke s ist $\delta^+(v)$ (bzw. $\delta^-(v)$) die Menge der eingehenden (bzw. ausgehenden) Kanten. Man erhält:

Dualitätsprinzip anwenden

$max\{\ x\cdot(\ \delta^+(q)\ -\ \delta^-(q)\ )\}$ unter den Bedingungen:

$x\cdot\delta^+(v)\ =\ x\cdot\delta^-(v)$ Flussgleichgewicht für $v \neq q, s$

$\forall e \in E: 0 \leq x(e) \leq g(e)$

$=$

$min(\sum c(e)\cdot y(e))$unter den Bedingungen:

$\forall e \in E: y(e) \geq 0$

$\forall v \in V: z(v) \in \mathbb{R}$ □ $z(r) - z(t)\ + y(e) \geq 0$ für $e = (r,t)$

□ $z(q) = 1$

□ $z(s) = 0$

Das Maximum entsprichtoffensichtlich dem maximalen Fluss. Man kann ebenfalls zeigen, dass obiges Minimum dem minimalen Schnitt von G entspricht. Dazu benötigt man jedoch noch den Satz über den Komplementären Schlupf (*complementary slackness*):

Satz:

Es seien $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$. Und die Dualitätsgleichung $max\{c' \cdot x \,|\, A \cdot x \leq b\} = min\{y' \cdot b \,|\, y' \cdot A = c', y \geq 0\}$ mit endlichen Optima. Seien weiter x_0 und y_0 Lösungen der Ungleichungen dann sind folgende Aussagen äquivalent:

1. x_0 und y_0 sind die optimalen Lösungen
2. $c' \cdot x_0 = b' \cdot y_0$
3. Gibt es eine positive Komponente in y_0, dann wird $A \cdot x \leq b$ durch x_0 mit Gleichheit $y_0'(b - A \cdot x_0) = 0$ erfüllt.