

Kapitel 1

Flüsse in Netzwerken

Wir betrachten Verfahren zur Bestimmung von maximalen und von kostenminimalen Flüssen in Netzwerken. Anwendungsgebiet sind Verkehrs- und oder Leitungsnetze - dies inkludiert insbesondere Daten- und Kommunikationsnetze.

1.1 Flüsse und Schnitte in Netzwerken - das Maximalflußproblem

Wir gehen von einem gerichteten Graphen (Netzwerk) $N = (V, E)$ aus, in dem es keine isolierten Knoten gibt. Mit $\lambda_{ij} \in R_+$ bzw. $\kappa_{ij} \in R_+ \cup \{\infty\}$, $\lambda_{ij} \leq \kappa_{ij}$, bezeichnen wir den *Minimal-* bzw. *Maximalkapazität* der Kante $\langle i, j \rangle \in E$. Unter Kapazität verstehen wir die zu einem statischen Zeitpunkt durch eine Kante erfaßbare Menge eines Gutes (Waren, Informationen, etc.). Somit heißt $N = (V, E; \lambda, \kappa)$ ein *Netzwerk mit Kapazitäten*.

Definition 1 Für die Knoten $r \in V$ und $s \in \dot{R}(r)$ versteht man unter einem Fluß von r nach s mit Flußstärke $\omega \geq 0$ eine Abbildung $\phi : E \rightarrow R_+ \cup \{\infty\}$, $\phi(\langle i, j \rangle) = \phi_{ij}$, mit

$$M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \begin{cases} \omega & i = r \\ -\omega & i = s \\ 0 & i \in V \setminus \{r, s\} \end{cases} \quad (1.1)$$

Wir schreiben die Stärke stets in Abhängigkeit vom entsprechenden Fluß: $\omega = \omega(\phi)$. ϕ heißt zulässiger Fluß, wenn

$$\lambda_{ij} \leq \phi_{ij} \leq \kappa_{ij} \quad (1.2)$$

für alle $\langle i, j \rangle \in E$ gilt.

Die obige Definition läßt sich wie folgt erklären: das Flußelement ϕ_{ij} beschreibt die Menge eines Gutes, die vom Knoten i zum Knoten j zu einem statischen Zeitpunkt transportiert wird. Dann sind

$\sum_{j \in S(i)} \phi_{ij}$... die vom Knoten i abtransportierte Menge

$\sum_{k \in P(i)} \phi_{ki}$... die zum Knoten i antransportierte Menge

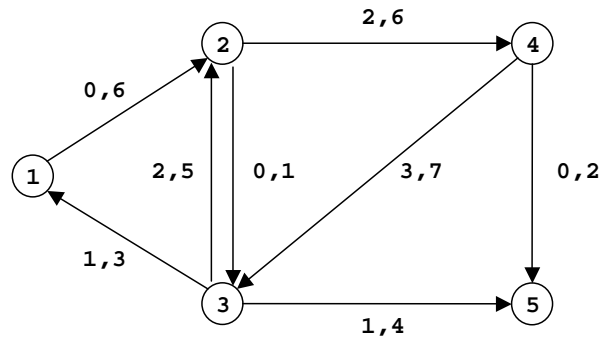


Abbildung 1.1: Netzwerk mit Minimal- und Maximalkapazität.

$$\sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} \dots \text{Differenz zwischen ab- und antransportierter Menge}$$

Damit ist ein Fluß der Stärke ω von r nach s , eine Belegung aller Kanten mit Mengen derart, daß vom Knoten r genau ω Einheiten ab- und am Knoten s antransportiert werden. An den restlichen Knoten ist Ab- gleich dem Antransport - $M(i)$ kann daher als Mengenbilanz des Knotens i interpretiert werden. Die Menge an Gütern verringert sich am Knoten r um ω und erhöht sich im selben Ausmaß am Knoten s - die Menge der Güter an den restlichen Knoten bleibt gleich. Wir bezeichnen r als *Flußquelle* und s als *Flußsenke*.

Bemerkung: da $s \in \dot{R}(r)$, befinden sich die Flußquelle und die Flußsenke immer in der selben schwachen Zusammenhangskomponente von N , i.e. $N' \subseteq N$. Beinhaltet N eine weitere schwache Zusammenhangskomponente, z.B. N'' , so ist ϕ zwar auch auf N'' definiert, besitzt jedoch keine Wirkung auf N' , da

$$S(i) \cap V'' = \emptyset \text{ für } i \in V' \quad \text{und} \quad P(i) \cap V'' = \emptyset \text{ für } i \in V'$$

Wir betrachten daher in Zukunft nur noch Flüsse auf einem schwach zusammenhängenden Netzwerk.

Beispiel 1.1(1): Wir betrachten das Netzwerk gemäß Abbildung 1.1. Für die Programmierung unter MATLAB empfiehlt es sich sowohl die Adjazenzinformation des Netzwerkes, die Kapazitäten und auch die Darstellung der Flüsse mittels **sparse**-Matrizen durchzuführen (\rightarrow `sec_flow_cut_1.m`), wobei $\lambda_{ij} = \kappa_{ij} = 0$ für $\langle i, j \rangle \notin E$. Der Fluß vom Knoten $r = 1$ zum Knoten $s = 5$ mit

$$\phi = \begin{bmatrix} 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 6 & 0 \\ 1 & 4 & 0 & 0 & 1 \\ 0 & 0 & 5 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

erfüllt die Mengenbilanz (1.1), denn es gilt

$$M = \begin{bmatrix} 2 & 0 & 0 & 0 & -2 \end{bmatrix}$$

und besitzt die Stärke $\omega(\phi) = 2$ (\rightarrow Abbildung 1.2).

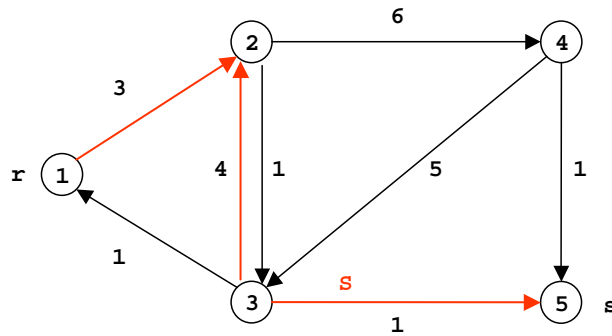


Abbildung 1.2: Netzwerk mit Kapazitäten gemäß der Abbildung 1.1. Die Zahlen auf den Kanten definieren einen zulässigen Fluß. Die roten Kanten definieren einen $(1, 5)$ -Semiweg.

Definition 2 Ein Fluß heißt Nullfluß, wenn $\phi_{ij} = 0$ für alle $\langle i, j \rangle \in E$.

Definition 3 Ein Fluß ϕ^1 heißt größer als ein Fluß ϕ^2 , wenn $\omega(\phi^1) > \omega(\phi^2)$ gilt.

Definition 4 Ein Fluß heißt maximaler Fluß, wenn er zulässig ist und maximale Stärke besitzt.

Die Bestimmung eines maximalen Flußes entspricht damit der Lösung des linearen Programmes

$$\max_{\phi} \omega(\phi) \text{ unter den Bedingungen (1.1) und (1.2)} \quad (\text{MFP})$$

(MFP) ist unter dem Namen *Maximalflußproblem* bekannt.

Verfahren zur Lösung von (MFP) beruhen auf der Konstruktion einer Folge von zulässigen Flüssen $\phi^0, \phi^1, \phi^2, \dots$ wobei $\omega(\phi^k) < \omega(\phi^{k+1})$ gilt. Wir wollen daher festhalten, wie ausgehend von einem zulässigen Fluß ϕ ein größerer Fluß $\tilde{\phi}$ konstruiert werden kann. Dazu legen wir fest:

Definition 5 Für die Flußquelle $r \in V$ und den Knoten $k \in V$ heißt der Semiweg

$$S = \langle r = i_1, i_2, \dots, i_{q-1}, i_q = k \rangle$$

ein (r, k) -Semiweg. Die Orientierung sei von r nach k definiert. Eine Kante $\langle i_k, i_{k+1} \rangle$ aus S heißt Vorwärtspfeil, wenn sie die selbe Orientierung wie S besitzt, andernfalls heißt sie Rückwärtspfeil.

Beispiel 1.1(2): Wir betrachten das Netzwerk gemäß Abbildung 1.2. Die Kanten $\langle 1, 2 \rangle$ und $\langle 3, 5 \rangle$ sind Vorwärtskanten des $(1, 5)$ -Semiweges; die Kante $\langle 3, 2 \rangle$ ist eine Rückwärtskante.

Wenn nun S ein (r, s) -Semiweg ist, mit r als Flußquelle und s als Flußsenke des zulässigen Flußes ϕ , dann definieren wir mit

$$\varepsilon_{ij} := \begin{cases} \kappa_{ij} - \phi_{ij} & \text{wenn } \langle i, j \rangle \text{ Vorwärtspfeil} \\ \phi_{ij} - \lambda_{ij} & \text{wenn } \langle i, j \rangle \text{ Rückwärtspfeil} \end{cases} \quad (1.3)$$

die maximal erlaubte Flußvergrößerung auf jedem Vorwärtspfeil bzw. maximal erlaubte Flußreduktion auf jedem Rückwärtspfeil des (r, s) -Semiwegs.

Beispiel 1.1(3): Für das Netzwerk in Abbildung 1.2 gilt:

$$\varepsilon_{1,2} = 3, \quad \varepsilon_{3,2} = 2, \quad \varepsilon_{3,5} = 3$$

Wenn wir festhalten, daß

$$\varepsilon := \min_{\text{auf } S} \varepsilon_{ij} \quad (1.4)$$

dann können wir den neuen Fluß $\tilde{\phi}$ wie folgt festlegen:

$$\tilde{\phi}_{ij} := \begin{cases} \phi_{ij} & \text{wenn } \langle i, j \rangle \text{ nicht im } (r, s)\text{-Semiweg } S \text{ enthalten} \\ \phi_{ij} + \varepsilon & \text{wenn } \langle i, j \rangle \text{ eine Vorwärtskante im } (r, s)\text{-Semiweg } S \\ \phi_{ij} - \varepsilon & \text{wenn } \langle i, j \rangle \text{ eine Rückwärtskante im } (r, s)\text{-Semiweg } S \end{cases} \quad (1.5)$$

und erhalten damit eine Flußvergrößerung, gemäß

$$\omega(\tilde{\phi}) = \omega(\phi) + \varepsilon$$

Bemerkung: man beachte in diesem Zusammenhang, daß (1.5) keine Auswirkungen auf die für Flüsse geforderte Mengenbilanz (1.1) besitzt. Liegt der Knoten i nicht am betrachteten (r, s) -Semiweg so bleibt der Gesamtan- und Abtransport unverändert. Für jeden Knoten i , der auf dem Semiweg liegt, wobei $i \notin \{r, s\}$ werden zwei Kanten verändert, wobei die folgenden Fälle eintreten können

$$r \rightarrow \begin{array}{c} \xleftarrow{-\varepsilon} i \xleftarrow{-\varepsilon} \\ \xrightarrow{+\varepsilon} i \xrightarrow{+\varepsilon} \\ \xrightarrow{+\varepsilon} i \xleftarrow{-\varepsilon} \\ \xleftarrow{-\varepsilon} i \xrightarrow{+\varepsilon} \end{array} \rightarrow s$$

In allen vier möglichen Fällen ist die Mengenveränderung am Knoten i gleich Null.

Definition 6 Existiert für einen Fluß ϕ ein (r, s) -Semiweg mit $\varepsilon > 0$, ε gemäß (1.4), dann heißt dieser ein flußvergrößernder (r, s) -Semiweg.

Aussage 1 Ein zulässiger Fluß ϕ von r nach s in einem Netzwerk mit Kapazitäten ist genau dann maximal, wenn für ϕ kein flußvergrößernder (r, s) -Semiweg gefunden werden kann.

Beispiel 1.1(4): Für den im Netzwerk der Abbildung 1.2 definierten Fluß können wir nunmehr den vergrößerten Fluß

$$\phi = \begin{bmatrix} 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 1 & 6 & 0 \\ 1 & 2 & 0 & 0 & 5 \\ 0 & 0 & 5 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

mit der Stärke $\omega(\phi) = 4$ angeben (\rightarrow Abbildung 1.3).

Für die weiteren Überlegungen zu Maximalflußproblemen definieren wir

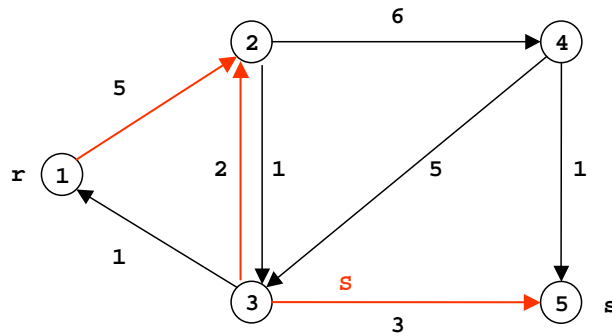


Abbildung 1.3: Vergrößerter Fluß zum Netzwerk mit Kapazitäten der Abbildung 1.1.

Definition 7 Es sei $G = (V, E)$ ein gerichteter Graph, wobei $V = A \cup B$, $A \cap B = \emptyset$, $A \neq \emptyset$ und $B \neq \emptyset$. Dann heißt die Kantenmenge

$$C \langle A, B \rangle := \{ \langle i, j \rangle \in E \mid i \in A \wedge j \in B \}$$

ein trennender Schnitt in G . $C \langle B, A \rangle$ wird als der zu $C \langle A, B \rangle$ konträre Schnitt bezeichnet. Die Kantenmenge

$$C(A, B) := C \langle A, B \rangle \cup C \langle B, A \rangle$$

wird als (A, B) -Schnitt auf G bezeichnet.

Bemerkung: In vielen Fällen wird ein (A, B) -Schnitt auch durch zwei Repräsentanten aus den Knotenmengen, o.B.d.A $i \in A$ und $j \in B$, als (i, j) -Schnitt bezeichnet.

Beispiel 1.1(5): Wenn wir das Netzwerk der Abbildung 1.1 betrachten und $A = \{1, 2\}$ und $B = \{3, 4, 5\}$ festlegen, so gilt:

$$C \langle A, B \rangle = \{ \langle 2, 3 \rangle, \langle 2, 4 \rangle \}$$

$$C \langle B, A \rangle = \{ \langle 3, 1 \rangle, \langle 3, 2 \rangle \}$$

$$C(A, B) = \{ \langle 2, 3 \rangle, \langle 2, 4 \rangle, \langle 3, 1 \rangle, \langle 3, 2 \rangle \}$$

Für Schnitte in Netzwerken mit Kapazitäten kann definiert werden:

Definition 8 Es sei $G = (V, E; \lambda, \kappa)$ ein Netzwerk mit Kapazitäten. Dann legen wir fest, daß

$$\lambda(C \langle A, B \rangle) := \sum_{\langle i, j \rangle \in C \langle A, B \rangle} \lambda_{ij} \quad \text{und} \quad \kappa(C \langle A, B \rangle) := \sum_{\langle i, j \rangle \in C \langle A, B \rangle} \kappa_{ij}$$

und

$$\mu(A, B) := \kappa(C \langle A, B \rangle) - \lambda(C \langle B, A \rangle) \tag{1.6}$$

$\mu(A, B)$ heißt die Kapazität des (A, B) -Schnittes.

Bemerkung: $\mu(A, B)$ beschreibt die Differenz zwischen der größtmöglichen Flußmenge von A nach B und der kleinstmöglichen Rückflußmenge von B nach A . Die Kapazität eines (A, B) -Schnittes entspricht also der maximal möglichen Nettoflußmenge von A nach B .

Gilt in einem Netzwerk, daß $\lambda_{i,j} = 0$, für alle $\langle i, j \rangle \in E$, so folgt

$$\lambda(C \langle B, A \rangle) = 0 \Rightarrow \mu(A, B) = \kappa(C \langle A, B \rangle)$$

In diesem Fall wird in der Literatur auch $C \langle A, B \rangle$ als (A, B) -Schnitt bezeichnet.

Beispiel 1.1(6): Wenn wir das Netzwerk der Abbildung 1.1 betrachten und wiederum $A = \{1, 2\}$ und $B = \{3, 4, 5\}$ festlegen, so gilt:

$$\kappa(C \langle A, B \rangle) = 1 + 6 = 7 \quad \text{und} \quad \lambda(C \langle B, A \rangle) = 1 + 2 = 3$$

$$\mu(A, B) = 7 - 3 = 4$$

Die maximale Nettoflußmenge von A nach B entspricht damit 4 Mengeneinheiten (\rightarrow `sec_flow_cut_6.m`, `cut.m`, `cut_capacity.m`).

Die Beschreibung eines Schnittes durch Repräsentanten ist nicht eindeutig. Wenn wir das Netzwerk der Abbildung 1.1 betrachten, so kann z.B. ein $(1, 5)$ -Schnitt mittels

$$A = \{1, 2\}, B = \{3, 4, 5\} \rightarrow \mu(A, B) = 4$$

oder

$$A = \{1, 2, 3\}, B = \{4, 5\} \rightarrow \mu(A, B) = 7$$

usw. definiert werden. Wir definieren daher:

Definition 9 *Ein Schnitt der unter allen möglichen (r, s) -Schnitten in einem Netzwerk die minimale Kapazität $\mu(r, s)$ besitzt heißt ein minimaler (r, s) -Schnitt.*

Das Auffinden eines minimalen (r, s) -Schnittes wird als *Minimalschnittproblem* bezeichnet. Man kann zeigen (\rightarrow Literatur), daß dieses Problem das duale lineare Programm zum Maximalflußproblem (MFP) darstellt. Gemäß der Aussage über die starke Dualität wissen wir, daß beide Probleme, wenn sie zulässig lösbar sind, den selben optimalen Kostenfunktionswert besitzen. Daher können wir die folgende Aussage formulieren:

Aussage 2 (*Maximalfluß-Minimalschnitt-Theorem*). *Die Stärke eines maximalen Flusses von r nach s in einem Netzwerk N mit Kapazitäten ist gleich der Kapazität eines minimalen (r, s) -Schnittes in N .*

Bemerkung: Ein minimaler (r, s) -Schnitt kann als die 'dünnste' Stelle im Netzwerk zwischen den Knoten r und s interpretiert werden. Daher besagt die Aussage 2, daß durch das Netzwerk höchstens soviel von r nach s transportiert werden kann, wie die Nettodurchflußkapazität der dünnsten Stelle erlaubt.

1.2 Berechnung maximaler Flüsse in einem Netzwerk

Wir wollen das *Verfahren nach Ford-Fulkerson* betrachten, das es erlaubt maximale Flüsse in Netzwerken mit Kapazitäten zu berechnen. Wie im Abschnitt 1.1 bereits andiskutiert, basiert dieses Verfahren auf der Konstruktion einer Folge von zulässigen Flüssen mit wachsender Flußstärke. Die in diesem Zusammenhang verwendete Strategie basiert auf der Konstruktion eines flußvergrößernden (r, s) -Semiwegs für einen aktuellen zulässigen Fluß ϕ von r nach s . Die Konstruktion eines flußvergrößernden (r, s) -Semiweg gliedert sich in zwei Schritte: einem *Markierungsprozess* und der *Flußvergrößerung*.

1.2.1 Markierungsprozess

Per Definition ist der Knoten r mit der Marke $(+, \varepsilon_r)$, $\varepsilon_r := \infty$, markiert. $\varepsilon_r = \infty$ bedeutet in diesem Zusammenhang, daß die Flußquelle r eine unendlich große Ergiebigkeit besitzt. Wir gehen davon aus, daß im Zuge des iterativen Markierungsprozesses der Knoten i bereits markiert wurde, wobei

$$i \leftarrow \begin{cases} (l^+, \varepsilon_i) & i \text{ wurde vom Knoten } l \in P(i) \text{ aus 'vorwärts' markiert} \\ (l^-, \varepsilon_i) & i \text{ wurde vom Knoten } l \in S(i) \text{ aus 'rückwärts' markiert} \end{cases}$$

Dann untersuchen wir ausgehend vom Knoten i alle noch nicht markierten Vorgänger und Nachfolger auf die folgenden beiden Bedingungen:

$j \in S(i)$, **nicht markiert** und $\phi_{ij} < \kappa_{ij}$: dann markieren wir j gemäß

$$j \leftarrow (i^+, \varepsilon_j) \quad \text{mit} \quad \varepsilon_j := \min(\varepsilon_i, \kappa_{ij} - \phi_{ij})$$

ε_j ist damit die maximale Flußvergrößerung für ϕ_{ij} unter Berücksichtigung der Zulässigkeitsbedingungen (1.2).

$k \in P(i)$, **nicht markiert** und $\phi_{ki} > \lambda_{ki}$: dann markieren wir k gemäß

$$k \leftarrow (i^-, \varepsilon_k) \quad \text{mit} \quad \varepsilon_k := \min(\varepsilon_i, \phi_{ki} - \lambda_{ki})$$

ε_k ist damit die maximale Verringerung des Rückflusses ϕ_{ki} unter Berücksichtigung der Zulässigkeitsbedingungen (1.2).

Nach Abarbeitung aller Vorgänger und Nachfolger von i wählen wir einen neuen markierten Knoten und wiederholen die Untersuchung. Dieser Markierungsprozess wird solange durchgeführt, bis eines der beiden folgenden Kriterien erfüllt ist:

1. Die Flußsenke s wurde markiert. In diesem Fall gehen wir zum Prozess der *Flußvergrößerung* über.
2. Es kann kein weiterer Knoten mehr markiert werden und die Flußsenke s bleibt unmarkiert. Im diesem Fall ist ϕ bereits der gesuchte zulässige Maximalfluß, da es zwischen r und s keine zulässige Flußvergrößerung bzw. Reduktion des Rückflusses mehr gibt und damit auch kein entsprechender flußvergrößernder (r, s) -Semiweg mehr gefunden werden kann; vergleiche dazu Aussage 1 auf Seite 4.

1.2.2 Flußvergrößerung

Die Konstruktion des flußvergrößernden (r, s) -Semiweges erfolgt von rückwärts - also von der Flußsenke s aus. Wir betrachten dazu die entsprechende Markierung und konstruieren aus ϕ einen neuen Fluß $\tilde{\phi}$ gemäß

$$\begin{aligned} s &\leftarrow \begin{cases} (q^+, \varepsilon_s) & \tilde{\phi}_{qs} = \phi_{qs} + \varepsilon \text{ mit } \varepsilon := \varepsilon_s \\ (q^-, \varepsilon_s) & \tilde{\phi}_{sq} = \phi_{sq} - \varepsilon \end{cases} \\ q &\leftarrow \begin{cases} (p^+, \varepsilon_q) & \tilde{\phi}_{pq} = \phi_{pq} + \varepsilon \\ (p^-, \varepsilon_q) & \tilde{\phi}_{qp} = \phi_{qp} - \varepsilon \end{cases} \\ &\vdots \end{aligned}$$

Wir können damit festhalten, daß

$$\omega(\tilde{\phi}) = \omega(\phi) + \varepsilon$$

gilt.

1.2.3 Endlichkeit des Verfahrens, minimaler (r, s) -Schnitt und Beispiel

Zum Nachweis der Endlichkeit des Verfahrens von Ford-Fulkerson kann wie folgt argumentiert werden: wenn wir von beschränkten Maximalkapazitäten κ_{ij} des Netzwerkes ausgehen und es eine zulässige Lösung des Maximalflußproblems gibt, dann ist auch besitzt auch der maximale Fluß eine endliche Stärke. Fordern wir zusätzlich, daß sowohl die Minimal- und Maximalkapazitäten als auch der Startfluß ganzzahlig sind, so erfolgt in jedem Durchlauf des Markierungs- und Flußvergrößerungsprozesses eine ganzzahlige Flußvergrößerung. Damit endet das Verfahren nach höchstens endlich vielen Durchläufen. Für allgemein reelwertige Kapazitäten und Flüsse kann die Endlichkeit des Verfahrens durch eine entsprechende Modifikation nachgewiesen werden (\rightarrow Literatur: Neumann, K., 1975, Operations Research Verfahren, Band III, Hanser Verlag).

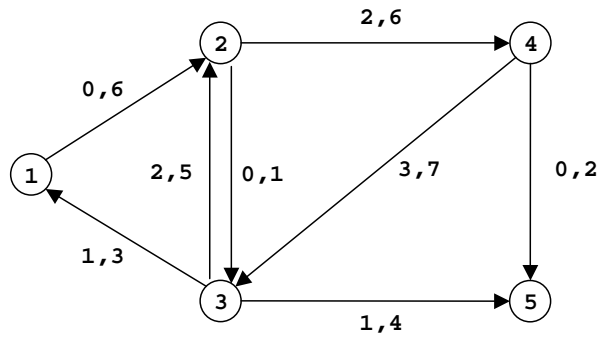
Wenn wir mit $L \subset V$ die im letzten Durchlauf markierten Knoten bezeichnen, und mit $\bar{L} := V \setminus L$ die entsprechende Komplementärmenge in V , dann ist $r \in L$ und $s \in \bar{L}$. Dann kann gezeigt werden, daß der (L, \bar{L}) -Schnitt ein minimaler (r, s) -Schnitt zum Netzwerk ist (\rightarrow Literatur: Neumann, K., 1975, Operations Research Verfahren, Band III, Hanser Verlag).

Abschließend geben wir noch ein Beispiel:

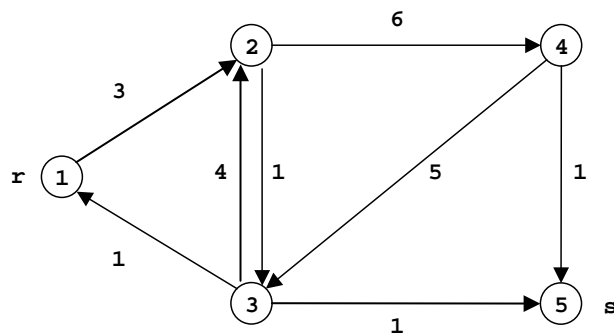
Beispiel 1.2(1): Wir gehen wiederum vom Netzwerk der Abbildung 1.1 aus und stellen sowohl das Netzwerk nochmals in der Abbildung 1.4 dar. Für die Quelle $r = 1$ und die Senke $s = 5$ gehen wir vom Fluß

$$\phi = \begin{bmatrix} 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 6 & 0 \\ 1 & 4 & 0 & 0 & 1 \\ 0 & 0 & 5 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{mit Stärke } \omega(\phi) = 2$$

Netzwerk mit Kapazitäten:



Startfluß:

Abbildung 1.4: Netzwerk mit Kapazitäten und zulässiger Startfluß von der Quelle $r = 1$ zur Senke $s = 5$.

aus. Wir erhalten im ersten Durchlauf des Markierungsprozesses:

$$r = 1 \leftarrow (+, \varepsilon_1 = \infty)$$

Untersuchung $i = 1$: $S(i) = \{2\}$, $P(i) = \{3\}$

$$j = 2, \text{ nicht markiert, } \phi_{1,2} = 3 < \kappa_{1,2} = 6$$

$$j = 2 \leftarrow (1^+, \varepsilon_2 = 3) \quad \text{wobei } \varepsilon_2 = \min(\varepsilon_1, \kappa_{1,2} - \phi_{1,2}) = \min(\infty, 6 - 3) = 3$$

$$k = 3, \text{ nicht markiert, } \phi_{3,1} = 1 \not\geq \lambda_{3,1} = 1, \text{ keine Markierung}$$

Untersuchung $i = 2$: $S(i) = \{3, 4\}$, $P(i) = \{1, 3\}$

$$j = 3, \text{ nicht markiert, } \phi_{2,3} = 1 \not\geq \kappa_{2,3} = 1, \text{ keine Markierung}$$

$$j = 4, \text{ nicht markiert, } \phi_{2,4} = 6 \not\geq \kappa_{2,4} = 6, \text{ keine Markierung}$$

$$k = 1, \text{ bereits markiert}$$

$$k = 3, \text{ nicht markiert, } \phi_{3,2} = 4 > \lambda_{3,2} = 2$$

$$k = 3 \leftarrow (2^-, \varepsilon_3 = 2) \quad \text{wobei } \varepsilon_3 = \min(\varepsilon_2, \phi_{3,2} - \lambda_{3,2}) = \min(3, 4 - 2) = 2$$

Untersuchung $i = 3$: $S(i) = \{1, 2, 5\}$, $P(i) = \{2, 4\}$

$j \in \{1, 2\}$, bereits markiert

$j = 5$, nicht markiert, $\phi_{3,5} = 1 < \kappa_{3,5} = 4$

$j = 5 \leftarrow (3^+, \varepsilon_5 = 2)$ wobei $\varepsilon_5 = \min(\varepsilon_3, \kappa_{3,5} - \phi_{3,5}) = \min(2, 4 - 1) = 2$

Die Flußsenke $s = 5$ wurde markiert und wir gehen zur Flußvergrößerung über und erhalten dem flußvergrößernden (r, s) -Semiweg gemäß

$$5 \leftarrow (3^+, 2) \Rightarrow \tilde{\phi}_{3,5} = \phi_{3,5} + \varepsilon = 3 \quad \text{mit } \varepsilon = 2$$

$$3 \leftarrow (2^-, 2) \Rightarrow \tilde{\phi}_{3,2} = \phi_{3,2} - \varepsilon = 2$$

$$2 \leftarrow (1^+, 3) \Rightarrow \tilde{\phi}_{1,2} = \phi_{1,2} + \varepsilon = 5$$

Damit erhalten wir den vergrößerten Fluß

$$\tilde{\phi} = \begin{bmatrix} 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 1 & 6 & 0 \\ 1 & 2 & 0 & 0 & 3 \\ 0 & 0 & 5 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{mit Stärke } \omega(\phi) = 4$$

und setzen für den nächsten Durchlauf $\phi = \tilde{\phi}$. Die Ergebnisse des ersten Durchlaufes sind in der Abbildung 1.5 graphisch dargestellt. Wir starten den zweiten Durchlauf mit dem Markierungsprozess:

$$r = 1 \leftarrow (+, \varepsilon_1 = \infty)$$

Untersuchung $i = 1$: $S(i) = \{2\}$, $P(i) = \{3\}$

$j = 2$, nicht markiert, $\phi_{1,2} = 5 < \kappa_{1,2} = 6$

$j = 2 \leftarrow (1^+, \varepsilon_2 = 1)$ wobei $\varepsilon_2 = \min(\varepsilon_1, \kappa_{1,2} - \phi_{1,2}) = \min(\infty, 6 - 5) = 1$

$k = 3$, nicht markiert, $\phi_{3,1} = 1 \not\geq \lambda_{3,1} = 1$, keine Markierung

Untersuchung $i = 2$: $S(i) = \{3, 4\}$, $P(i) = \{1, 3\}$

$j = 3$, nicht markiert, $\phi_{2,3} = 1 \not\geq \kappa_{2,3} = 1$, keine Markierung

$j = 4$, nicht markiert, $\phi_{2,4} = 6 \not\geq \kappa_{2,4} = 6$, keine Markierung

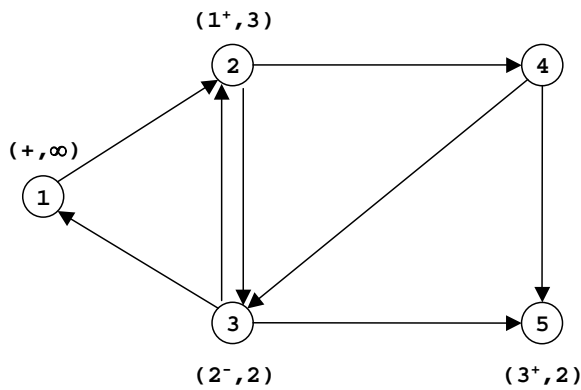
$k = 1$, bereits markiert

$k = 3$, nicht markiert, $\phi_{3,2} = 2 \not\geq \lambda_{3,2} = 2$, keine Markierung

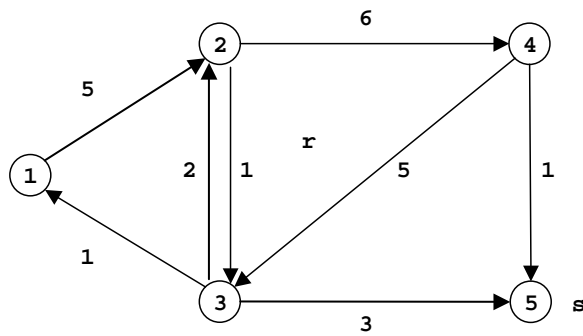
Wir brechen den Markierungsprozess ab, da kein weiterer Knoten mehr markiert werden kann. $s = 5$ wurde noch nicht markiert. Wir können daher behaupten, daß der aktuelle Fluß maximal ist.

Wenn wir nun $L = \{1, 2\}$ und $\bar{L} = \{3, 4, 5\}$ festlegen so können wir behaupten, daß der (L, \bar{L}) -Schnitt ein minimaler (r, s) -Schnitt im betrachteten Netzwerk ist. Die Kapazität $\mu(L, \bar{L}) = 4$ - damit entspricht die maximale Nettodurchflußmenge an der dünnsten Stelle des Netzwerkes von r nach s der maximalen Flußstärke von r nach s . Der minimale Schnitt ist ebenso in Abbildung 1.5 dargestellt.

1. Durchlauf - Markierungen:



1. Durchlauf - Flußvergrößerung:



Minimaler (1,5)-Schnitt:

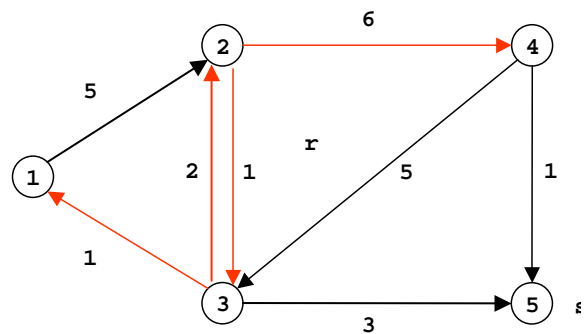


Abbildung 1.5: Erster Durchlauf des Verfahrens von Ford-Fulkerson im Beispiel 1.2(1).

1.2.4 Verfahren von Ford-Fulkerson als Pseudocode

Abschließend beschreiben wir noch das Verfahren von Ford-Fulkerson als Pseudocode, wobei wir voraussetzen, daß ein zulässiger Fluß ϕ bekannt sei. Wie wir einen solchen Fluß bzw. seine Existenz überprüfen, diskutieren wir in einem späteren Abschnitt. Weiters sei N als antisymmetrisches Netzwerk vorausgesetzt - d.h. es existieren keine antiparallelen Kanten. Dies stellt keine Einschränkung der Allgemeinheit dar, denn gilt $\langle i, j \rangle \in E$ und $\langle j, i \rangle \in E$ so führen wir einen Hilfsknoten k ein, eliminieren die Kante $\langle i, j \rangle$ und ersetzen sie durch die Kanten $\langle i, k \rangle$ mit $\lambda_{ik} = \lambda_{ij}$ und $\kappa_{ik} = \kappa_{ij}$ sowie durch die Kante $\langle k, j \rangle$ mit $\lambda_{kj} = \lambda_{ij}$ und $\kappa_{kj} = \kappa_{ij}$.

```
[ $\phi, \omega, C$ ] = FF_MaxFlow[ $V, E, \lambda, \kappa, \phi, r, s$ ]
```

% Initialisierung

```
> teste das Netzwerk auf Antisymmetrie
> bestimme  $\omega = \omega(\phi)$  und teste Initialfluß auf Mengenbilanz und Zulässigkeit
>  $b = false$  (bool'sche Variable, wird true gesetzt sobald maximaler Fluß gefunden)
>  $M(i) = 0$  für  $i = 1 \dots |V|$  (beinhaltet alle von  $i$  aus markierbaren Knoten)
>  $C = 0$ 
> für  $i = 1 \dots |V|$  führe aus
>   für  $j \in S(i)$  führe aus
```

```

> falls  $\phi_{ij} < \kappa_{ij}$ , dann setze  $M(i) = M(i) \cup \{j\}$ 
> falls  $\phi_{ij} > \lambda_{ij}$ , dann setze  $M(j) = M(j) \cup \{i\}$ 

% Iteration
>  $[b, p, \varepsilon, L] = \text{INC\_RS\_SEMIPATH}[V, E, \lambda, \kappa, \phi]$ 
> solange  $b = \text{false}$ , führe aus
>    $[\phi, M, \omega] = \text{INC\_FLOW}[p, \varepsilon_s, \phi, \kappa, \lambda, r, s, M, \omega]$ 
>    $[b, p, \varepsilon, L] = \text{INC\_RS\_SEMIPATH}[V, E, \lambda, \kappa, \phi]$ 

% Minimaler (r,s)-Schnitt
> für alle  $i \in L$  führe aus
>   für alle  $j \in S(i) \setminus L$  führe aus
>     setze  $C = C \cup \{\langle i, j \rangle\}$ 
> für alle  $i \in V \setminus L$  führe aus
>   für alle  $j \in S(i) \cap L$  führe aus
>     setze  $C = C \cup \{\langle i, j \rangle\}$ 

% SUBROUTINE  $[b, p, \varepsilon, M, L] = \text{INC\_RS\_SEMIPATH}[V, E, \lambda, \kappa, \phi, r, s, M]$ 
>  $p_r = r, \varepsilon_r = \infty, p_i = 0$  für  $i \in V \setminus \{r\}$ 
>  $Q = \{r\}, L = \{r\}$  ( $Q$  und  $L$  sind vom Typ Queue1)
> solange  $Q \neq 0$ , führe aus
>    $i = Q(1), Q = Q(2, \dots, \text{end})$ 
>   für alle  $j \in M(i) \setminus L$ , führe aus
>      $Q(\text{end}) = j, L(\text{end}) = j$ ,
>     falls  $j \in S(i)$ , dann setze  $p_j = i$  und  $\varepsilon_j = \min(\varepsilon_j, \kappa_{ij} - \phi_{ij})$  (Vorwärtsmarkierung)
>     sonst setze  $p_j = -i$  und  $\varepsilon_j = \min(\varepsilon_j, \phi_{ji} - \lambda_{ji})$  (Rückwärtsmarkierung)
>     falls  $j = s$ , dann beende die 'for'- und 'while'-Schleife (Flußsenke wurde markiert)
> falls  $p_s = 0$ , führe aus
>    $b = \text{true}$ 

% SUBROUTINE  $[\phi, M, \omega] = \text{INC\_FLOW}[p, \varepsilon_s, \phi, \kappa, \lambda, r, s, M, \omega]$ 
> setze  $\omega = \omega + \varepsilon_s, i = s$ 
> solange  $i \neq r$ , führe aus
>   setze  $j = i, i = \text{abs}(p_j), M(j) = M(j) \cup \{i\}$ 
>   falls  $p_j > 0$ , dann führe aus
>     setze  $\phi_{ij} = \phi_{ij} + \varepsilon_s$  (Flußvergrößerung)
>     falls  $\phi_{ij} = \kappa_{ij}$ , dann führe aus  $M(i) = M(i) \setminus \{j\}$ 
>   sonst führe aus
>     setze  $\phi_{ji} = \phi_{ji} - \varepsilon_s$  (Rückflußverkleinerung)
>     falls  $\phi_{ij} = \lambda_{ij}$ , dann führe aus  $M(i) = M(i) \setminus \{j\}$ 

```

Der Aufwand des Markierungsprozesses ist $O(|E|)$ und der Flußvergrößerung $O(|V|)$ - damit ist jeder Iterationsschritt von der Komplexität $O(|E|)$. Sind die Kapazitäten sowie der Initialfluß ganzzahlig und ist die maximale Flußstärke gleich ω^* , so sind höchstens $\omega^* + 1$ Iterationen

¹Queues sind FIFO-Speicher. Wenn Q eine Queue ist so gilt: schreiben des Elementes i auf Q : $Q = [Q, i]$, lesen eines Elementes von Q : $i = Q(1)$.

notwendig, da in jeder Iteration die Flußstärke um mindestens Eins erhöht wird. Es folgt damit für ganzzahlige Probleme die Komplexität $O(\omega^* |E|)$. Für nicht ganzzahlige Probleme kann gezeigt werden, daß wenn die Flußvergrößerung entlang von (r, s) -Semiwegen mit minimaler Kantenanzahl erfolgt, höchstens $(|E| \cdot |V|) / 2$ Flußvergrößerungen notwendig sind (\rightarrow Jungnickel, D., 1987, Graphen, Netzwerke und Algorithmen, Springer Verlag). Dies ist für den obigen Algorithmus erfüllt, denn der Markierungsprozess arbeitet nach dem Prinzip der Breitensuche und bestimmt daher stets Wege mit minimaler Kantenanzahl. Die Gesamtkomplexität von `FF_MaxFlow` beträgt damit $O(|E|^2 |V|)$.

Siehe dazu auch die Routine `sec_max_flow_comput_1.m` und `FF_MaxFlow.m`.

Beispiel 1.2(2): Siehe Routinen `sec_max_flow_comput_2.m`.

1.2.5 Inkrementelles Verfahren

Wir geben noch eine Modifikation des Verfahrens von Ford-Fulkerson an. Dabei wird der Markierungsprozess durch ein inkrementelles Verfahren ersetzt - wir sprechen in diesem Zusammenhang vom *inkrementellen Verfahren nach Ford-Fulkerson*. Wir gehen davon aus, daß das Netzwerk $N = (V, E; \lambda, \kappa)$ wiederum antisymmetrisch ist. r sei wiederum die Flußquelle und s die Flußsenke.

Definition 10 Für das antisymmetrische Netzwerk N mit Kapazitäten und den zulässigen Fluß ϕ heißt das Netzwerk mit Maximalkapazität $N(\phi) = (V, E(\phi); \kappa(\phi))$ das Inkrementnetzwerk, wobei

$$E(\phi) := E^+(\phi) \cup E^-(\phi) \quad (1.7)$$

$$E^+(\phi) := \{ \langle i, j \rangle \mid \langle i, j \rangle \in E, \phi_{ij} < \kappa_{ij} \}$$

$$E^-(\phi) := \{ \langle j, i \rangle \mid \langle i, j \rangle \in E, \phi_{ij} > \lambda_{ij} \}$$

und

$$\kappa_{ij}(\phi) := \begin{cases} \kappa_{ij} - \phi_{ij} & \text{für } \langle i, j \rangle \in E^+(\phi) \\ \phi_{ji} - \lambda_{ji} & \text{für } \langle i, j \rangle \in E^-(\phi) \end{cases} \quad (1.8)$$

Bemerkung: Die Kantenmenge $E^+(\phi)$ beschreibt alle möglichen Vorwärtspfeile eines flußvergrößernden (r, s) -Semiwegs in N , die Menge $E^-(\phi)$ hingegen die Umkehrpfeile zu allen möglichen Rückwärtspfeilen. Da N antisymmetrisch ist besitzt das Inkrementnetzwerk $N(\phi)$ keine parallelen Kanten.

Beobachtung: Insbesondere können wir festhalten, daß jeder für ϕ flußvergrößernde (r, s) -Semiweg in N einem Weg W von r nach s im Inkrementnetzwerk $N(\phi)$ entspricht. Diese Beobachtung stützt sich darauf, daß jeder Rückwärtspfeil im (r, s) -Semiweg von N einem Vorwärtspfeil aus $E^-(\phi)$ in $N(\phi)$ entspricht \rightarrow Abbildung 1.6.

Aus dieser Beobachtung läßt sich die folgende Aussage ableiten:

Aussage 3 Ist die Flußsenke s von der Flußquelle r aus im Inkrementnetzwerk $N(\phi)$ zum Netzwerk N nicht erreichbar, so existiert auch kein flußvergrößernder (r, s) -Semiweg in N . Der aktuelle Fluß ϕ ist maximal in N .

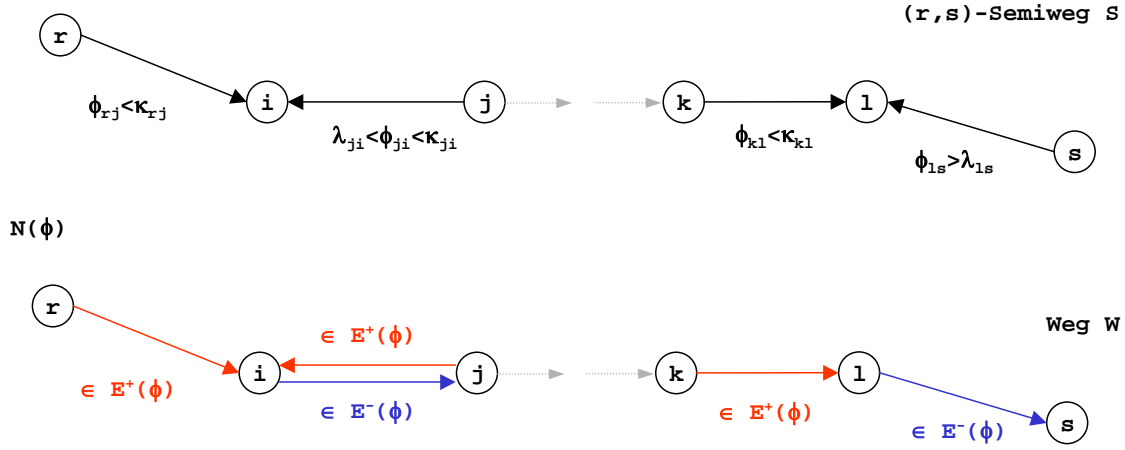


Abbildung 1.6: Überführung eines (r,s) -Semiweg in N auf einen Weg im entsprechenden Inkrementnetzwerk $N(\phi)$.

Diese Aussage bildet also ein Abbruchkriterium für das inkrementelle Verfahren. Wir können jedoch auch umgekehrt behaupten:

Aussage 4 Jeder Weg von r nach s im Inkrementnetzwerk $N(\phi)$ entspricht einem flußvergrößernden (r,s) -Semiweg in N .

Nachweis: s sei von r aus in $N(\phi)$ erreichbar - wir können dies z.B. durch Breitensuche überprüfen und bezeichnen den gefundenen Weg mit Hilfe der entsprechenden Kantenmenge W . Dann legen wir für den gesuchten (r,s) -Semiweg S fest, daß

$$\text{für } \langle i, j \rangle \in W \text{ gilt } \begin{cases} \langle i, j \rangle \text{ ist Vorwärtskante in } S \Leftrightarrow \langle i, j \rangle \in E^+(\phi) \\ \langle j, i \rangle \text{ ist Rückwärtskante in } S \Leftrightarrow \langle i, j \rangle \in E^-(\phi) \end{cases} \quad (1.9)$$

Wenn wir

$$\varepsilon := \min_{\langle i, j \rangle \in W} \kappa_{ij}(\phi) \quad (1.10)$$

definieren, dann stellt S gemäß einen flußvergrößernden (r,s) -Semiweg dar. ■

Mit dem im Nachweis konstruierten (r,s) -Semiweg können wir wiederum gemäß (1.5) einen vergrößerten Fluß konstruieren und erhalten:

$$\tilde{\phi}_{ij} := \begin{cases} \phi_{ij} & \text{für } \langle i, j \rangle \notin S \\ \phi_{ij} + \varepsilon & \langle i, j \rangle \text{ ist Vorwärtskante in } S \\ \phi_{ij} - \varepsilon & \langle i, j \rangle \text{ ist Rückwärtskante in } S \end{cases}$$

Die Konstruktion kann auch kompakter in Matrizenschreibweise ohne explizite Angabe von S erfolgen. Dazu konstruieren wir einen zulässigen Fluß im Inkrementnetzwerk $N(\phi)$ wie folgt:

$$\Psi_{ij} := \begin{cases} \varepsilon & \text{für } \langle i, j \rangle \in W \\ 0 & \text{sonst} \end{cases} \quad \text{auf allen Kanten } \langle i, j \rangle \in E(\phi) \quad (1.11)$$

und berechnen $\tilde{\phi}$ wie folgt:

$$\tilde{\phi}_{ij} := \phi_{ij} + \begin{cases} \Psi_{ij} & \text{für } \langle i, j \rangle \in E^+(\phi) \\ -\Psi_{ji} & \text{für } \langle j, i \rangle \in E^-(\phi) \end{cases} \quad \text{auf allen Kanten } \langle i, j \rangle \in E \quad (1.12)$$

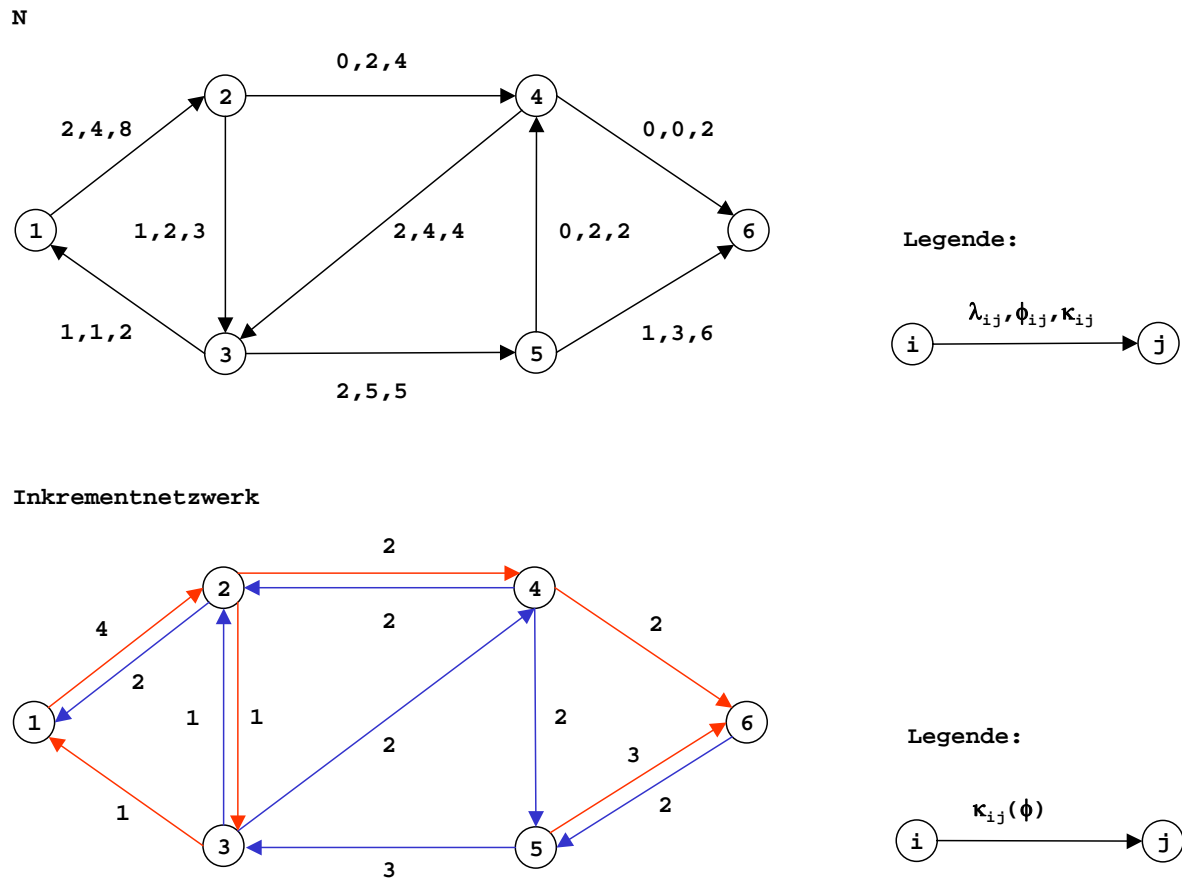


Abbildung 1.7: Netzwerk mit Kapazitäten und Fluß von $r = 1$ nach $s = 6$ sowie entsprechendem Inkrementnetzwerk. Die roten Kanten sind Elemente der Menge $E^+(\phi)$, die blauen Kanten gehören zur Menge $E^-(\phi)$.

Insbesondere gilt, daß $\omega(\tilde{\phi}) = \omega(\phi) + \varepsilon$ und sind ϕ und Ψ zulässige Flüsse in ihren jeweiligen Netzwerken, dann ist auch $\tilde{\phi}$ zulässig in N .

Beispiel 1.2(3): Wir betrachten das Netzwerk mit Kapazitäten N und einen gegebenen zulässigen Fluß ϕ von $r = 1$ nach $s = 6$, mit $\omega(\phi) = 3$, gemäß der Abbildung 1.7. Ebenso ist in der Abbildung 1.7 das entsprechende Inkrementnetzwerk $N(\phi)$ dargestellt. Die Senke s ist von der Quelle r aus in $N(\phi)$ erreichbar - wir erhalten z.B.

$$W = \langle 1, 2, 4, 6 \rangle$$

Damit folgt der zulässige Fluß am Inkrementnetzwerk $N(\phi)$:

$$\varepsilon := \min_{\langle i, j \rangle \in W} \kappa_{ij}(\phi) = 2$$

$$\Psi = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{mit } \omega(\Psi) = 2$$

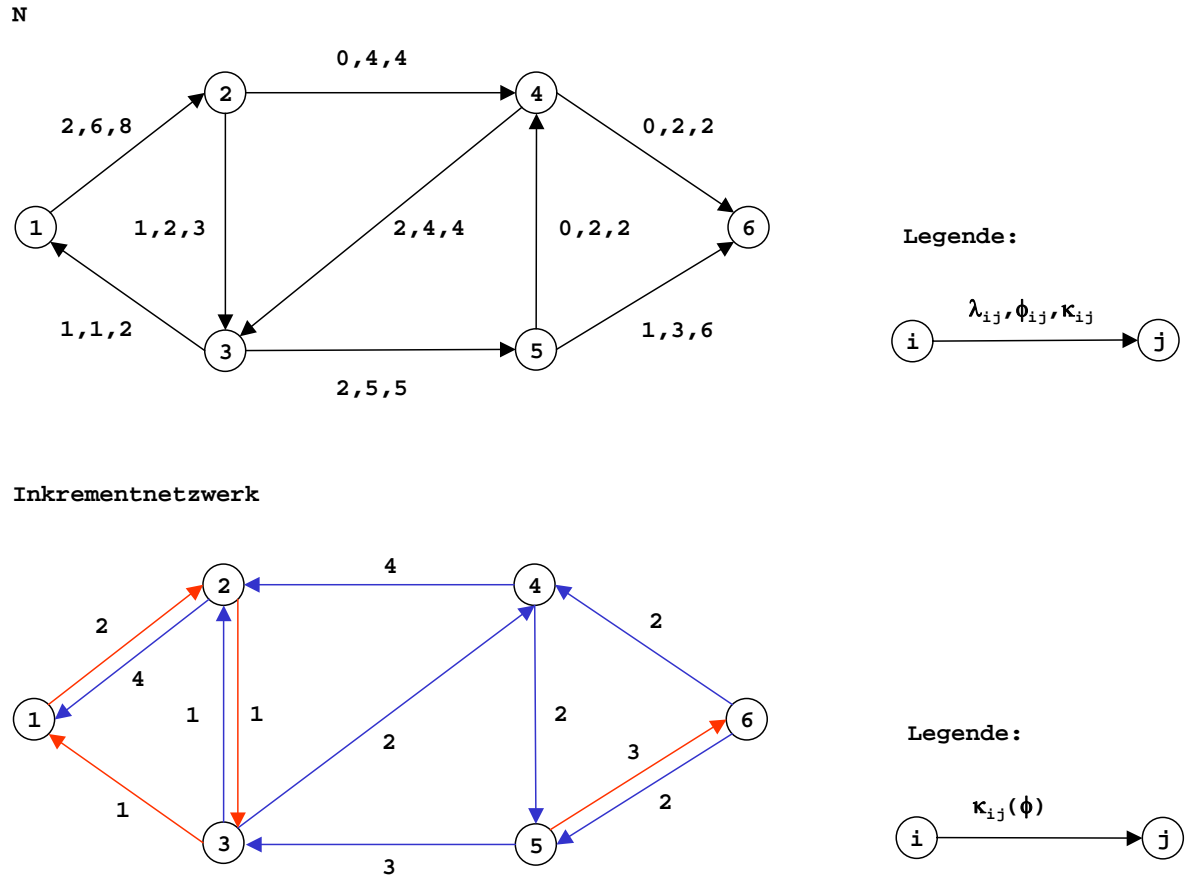


Abbildung 1.8: Netzwerk mit Kapazitäten und Fluß von $r = 1$ nach $s = 6$ sowie entsprechendem Inkrementnetzwerk. Die roten Kanten sind Elemente der Menge $E^+(\phi)$, die blauen Kanten gehören zur Menge $E^-(\phi)$.

Gemäß (1.12) konstruieren wir den für N zulässigen Fluß von $r = 1$ nach $s = 5$ und erhalten

$$\tilde{\phi} = \begin{bmatrix} 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 \\ 1 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 4 & 0 & 0 \\ 1 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 4 & 0 & 0 & 2 \\ 0 & 0 & 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

und stellen fest, daß $\omega(\tilde{\phi}) = 5$. Der vergrößerte Fluß ist in Abbildung 1.8 dargestellt. Für den nächsten Durchlauf setzen wir $\phi = \tilde{\phi}$. Wir konstruieren wiederum das Inkrementnetzwerk $N(\phi)$ - dieses ist in Abbildung 1.8 dargestellt. Die Senke s ist von der Quelle r aus in $N(\phi)$ erreichbar - wir wählen z.B.

$$W = \langle 1, 2, 3, 4, 5, 6 \rangle$$

Der zulässige Fluß am Inkrementnetzwerk $N(\phi)$ lautet:

$$\varepsilon := \min_{\langle i, j \rangle \in W} \kappa_{ij}(\phi) = 1$$

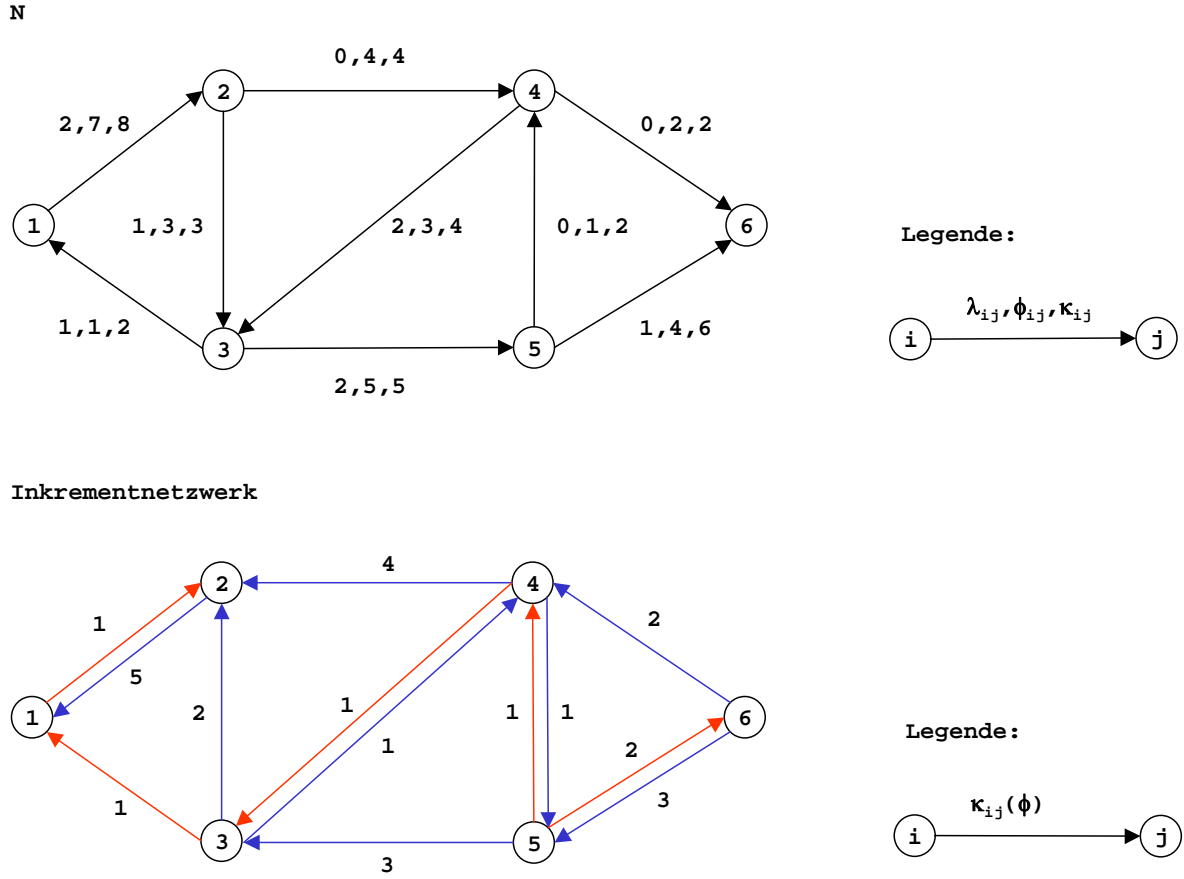


Abbildung 1.9: Netzwerk mit Kapazitäten und Fluß von $r = 1$ nach $s = 6$ sowie entsprechendem Inkrementnetzwerk. Die roten Kanten sind Elemente der Menge $E^+(\phi)$, die blauen Kanten gehören zur Menge $E^-(\phi)$.

$$\Psi = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{mit } \omega(\Psi) = 1$$

Gemäß (1.12) konstruieren wir den für N zulässigen Fluß von $r = 1$ nach $s = 5$ und erhalten

$$\tilde{\phi} = \begin{bmatrix} 0 & 6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 4 & 0 & 0 \\ 1 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 4 & 0 & 0 & 2 \\ 0 & 0 & 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 4 & 0 & 0 \\ 1 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Wir stellen fest, daß $\omega(\tilde{\phi}) = 6$. Der vergrößerte Fluß ist in Abbildung 1.9 dargestellt. Für den nächsten Durchlauf setzen wir $\phi = \tilde{\phi}$. Die Senke s ist von der Quelle r aus in $N(\phi)$ nicht mehr erreichbar - der aktuelle Fluß ϕ ist gemäß der Aussage 3 maximal (\rightarrow `sec_max_flow_comput_3.m`, `NwrtPhi.m`, `FFInc_MaxFlow.m`).

Wir wollen auch für das inkrementelle Verfahren einen Pseudocode angeben und formulieren daher:

$[\phi, \omega, C] = \text{FFInc_MaxFlow}[V, E, \lambda, \kappa, \phi, r, s]$

% Initialisierung

> teste das Netzwerk auf Antisymmetrie
 > bestimme $\omega = \omega(\phi)$ und teste Initialfluß auf Mengenbilanz und Zulässigkeit
 > $b = false$ (bool'sche Variable, wird **true** gesetzt sobald maximaler Fluß gefunden)

% Iteration

> solange $b = false$, führe aus
 > bestimme das Inkrementnetzwerk $N(\phi) = (V, E(\phi); \kappa(\phi))$ zu ϕ gemäß (1.7) und (1.8)
 > bestimme mit Hilfe der Breitensuche den Weg von r nach s in $N(\phi)$
 > wenn s in $N(\phi)$ von r aus nicht erreichbar ist, dann
 > setze $b = true$ und beende die Iteration; aktueller Fluß ist maximal
 > ansonsten,
 > bestimme ε gemäß (1.10)
 > bestimme Ψ gemäß (1.11)
 > bestimme den Update von ϕ gemäß (1.12)

Die Komplexität zur Bestimmung von $N(\phi)$ und Ψ ist pro Iteration von der Ordnung $O(|E|)$ und unter der Verwendung der Breitensuche kann wiederum gezeigt werden, daß maximal $(|E| \cdot |V|)/2$ Flußvgrößerungen notwendig sind. Die Komplexität von **FFInc_MaxFlow** beträgt daher $O(|E|^2 |V|)$.

1.2.6 Schichtnetzwerk und blockierender Fluß - das MKM Verfahren

Das Inkrement-Verfahren des vorigen Abschnittes kann erheblich beschleunigt werden, wenn wir im Inkrementnetzwerk $N(\phi)$ nicht 'irgendeinen' Fluß betrachten, sondern versuchen einen Fluß möglichst großer Stärke zu konstruieren. Dieses Verfahren wird auf die Autoren Malhotra, Kumar und Maheswari zurückgeführt und heißt daher *MKM-Verfahren*.

Im vorigen Abschnitt haben wir mit Hilfe der Breitensuche einen Weg von r nach s im Inkrementnetzwerk $N(\phi)$ bestimmt. Im Zuge der Breitensuche werden die Knoten schichtweise markiert, wobei zu einer Schicht alle Knoten gleicher Tiefe im Wegebaum mit Wurzel r gehören (\rightarrow OR, Morlock). Wir bezeichnen mit d_i die Schicht des Knoten i . Damit können nun Filterkriterien für solche Knoten und Kanten im Inkrementnetzwerk $N(\phi)$ angegeben werden, die nicht zur Bestimmung eines zulässigen Flusses in $N(\phi)$ beitragen. Es gilt, daß bei der Bestimmung eines Weges mit geringster Kantenanzahl von r nach s (z.B. Breitensuche) in $N(\phi)$ unnötig sind:

- (i) alle Knoten $j \neq s$, die $d_j \geq d_s$ erfüllen und alle Kanten $\langle i, j \rangle$ bzw. $\langle j, i \rangle \in E(\phi)$
- (ii) alle Kanten $\langle i, j \rangle$, für die $d_i \geq d_j$

Nach Elimination der Knoten und Kanten gemäß (i) und (ii) können in $N(\phi)$ noch eliminiert werden:

- (iii) alle Knoten $j \neq s$, die $S(j) = 0$ erfüllen und alle Kanten $\langle i, j \rangle \in E(\phi)$

Beispiel 1.2(4): → Abbildung 1.10

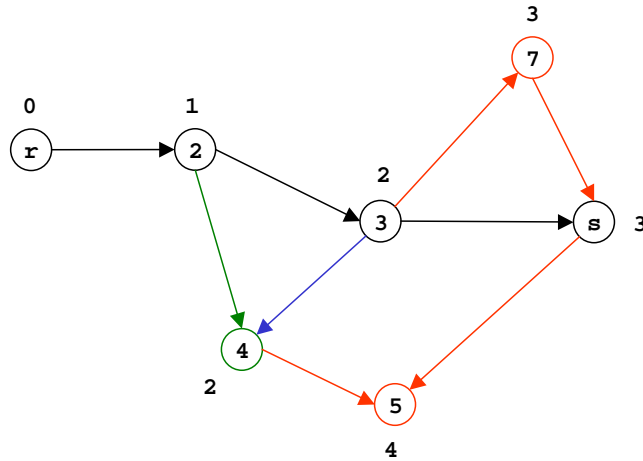


Abbildung 1.10: Inkrementnetzwerk $N(\phi)$ mit Quelle r und Senke s . Die Markierungen auf den Knoten entsprechen den Schichten der Knoten. Die rot eingefärbten Knoten und Kanten werden aufgrund des Kriteriums (i) gelöscht; entsprechend die blau eingefärbte Kante gemäß (ii) und die grün eingefärbten Informationen aufgrund des Kriteriums (iii).

Wir bezeichnen das gemäß den Kriterien (i) bis (iii) gefilterte Netzwerk als *Schichtennetzwerk* $SN(\phi) = (SV(\phi), SE(\phi); \kappa(\phi))$. Die Bezeichnung ist einleuchtend, denn wenn wir mit U_ν die ν -te Schicht im Wegebaum der Breitensuche mit Wurzel r bezeichnen

$$U_0 = \{r\}, \quad U_{d_s} = \{s\} \quad \text{und} \quad U_\nu = \{j \mid d_j = \nu\} \quad \text{für } \nu = 1, \dots, d_s - 1$$

Damit gilt

$$SV(\phi) = \bigcup_{\nu=0}^{d_s} U_\nu$$

und zudem

$$\langle j, i \rangle \in SE(\phi) \Rightarrow i \in U_{\nu-1} \wedge j \in U_\nu \quad \text{für } 1 \leq \nu \leq d_s$$

Beispiel 1.2(5): wir betrachten das Netzwerk mit Kapazitäten und zulässigen Fluß von $r = 1$ nach $s = 8$ gemäß Abbildung 1.11 (→ `sec_max_flow_comput_5.m`). und das entsprechende Inkrementnetzwerk (→ `NwrtPhi.m`) gemäß Abbildung 1.12. Gemäß dem Filterkriterium (i) kann der Knoten 5 und alle inzidenten Kanten vernachlässigt werden. Gemäß dem Kriterium (ii) können die folgenden Kanten gelöscht werden: $\langle 3, 1 \rangle$, $\langle 3, 2 \rangle$, $\langle 5, 2 \rangle$, $\langle 7, 4 \rangle$, $\langle 8, 5 \rangle$, $\langle 5, 6 \rangle$, $\langle 3, 7 \rangle$, $\langle 6, 7 \rangle$, $\langle 8, 7 \rangle$, $\langle 5, 8 \rangle$, $\langle 8, 9 \rangle$. Das Filterkriterium (iii) erlaubt das Löschen des Knoten 9 sowie der Kante $\langle 7, 9 \rangle$. Wir erhalten damit das Schichtennetzwerk gemäß der Abbildung 1.13 (→ `SNwrtPhi.m`).

Auf dem Schichtennetzwerk $SN(\phi)$ konstruieren wir einen *blockierenden Fluß* $\tilde{\Psi}$. Per Definition verstehen wir unter einem blockierenden Fluß einen Fluß von r nach s im Netzwerk $SN(\phi)$ derart, daß kein flußvergrößernder Weg, der nur Kanten aus $E^+(\phi)$ besitzt (i.e. ausschließlich Kanten

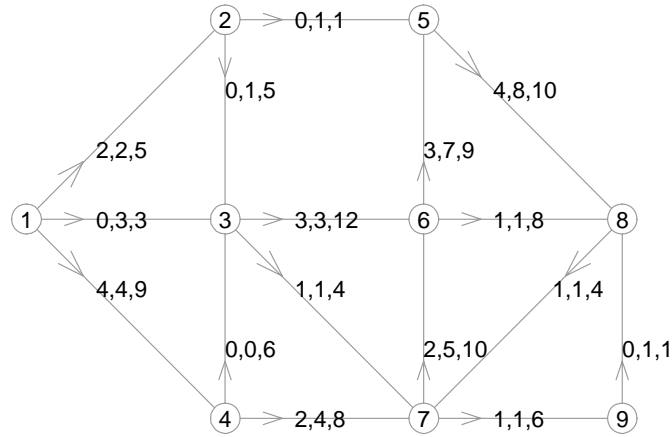


Abbildung 1.11: Netzwerk mit Kapazitäten. Die Kantenbeschriftung entspricht $\lambda_{ij}, \phi_{ij}, \kappa_{ij}$.

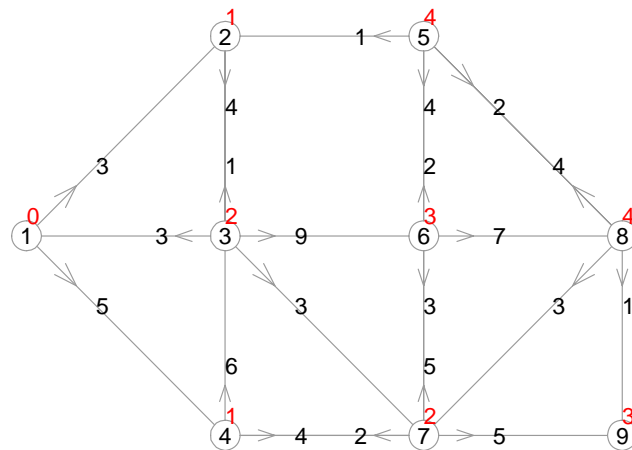


Abbildung 1.12: Inkrementnetzwerk $N(\phi)$. Die Kantenbeschriftung entspricht $\kappa_{ij}(\phi)$. Die den Knoten zugeordneten roten Zahlen entsprechen den Schichten im Breitensuch-Wegebaum mit Wurzel $r = 1$.

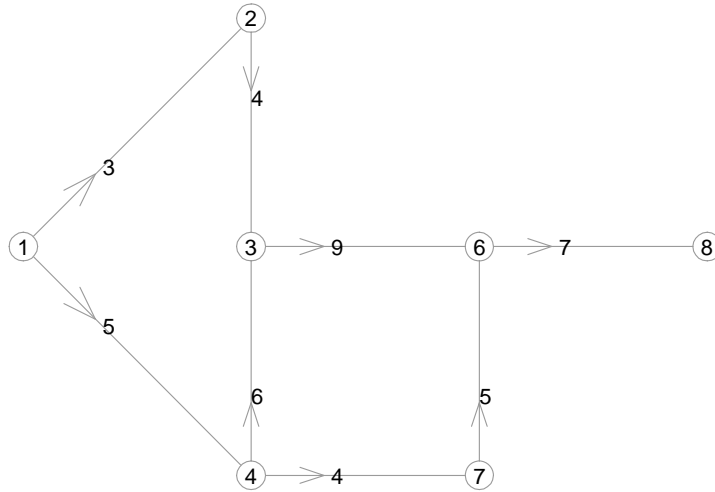


Abbildung 1.13: Schichtennetzwerk $SN(\phi)$. Die Kantenbeschriftungen entsprechen den $\kappa_{ij}(\phi)$.

mit $\phi_{ij} < \kappa_{ij}$), gefunden werden kann. Trivialerweise entspricht jeder maximale Fluß in $SN(\phi)$ einem blockierenden Fluß; die Umkehrung ist im Allgemeinen falsch. $\tilde{\Psi}$ entspricht insbesondere auch einem Fluß möglichst großer Stärke im Inkrementnetzwerk $N(\phi)$.

Zur Bestimmung von $\tilde{\Psi}$ gehen wir iterativ vor. Als Iterationsanfang setzen wir

$$SN^0(\phi) = (SV^0(\phi), SE^0(\phi); \kappa^0(\phi)) := SN(\phi) = (SV(\phi), SE(\phi); \kappa(\phi))$$

und $\Psi_{ij}^0 = 0$ für $i, j \in SV^0(\phi)$.

In der k -ten Iteration berechnen wir auf $SN^k(\phi)$ die maximale Durchflußmenge des Knoten i gemäß

$$f_i^k := \min \left\{ \sum_{j \in S(i)} \kappa_{ij}^k(\phi), \sum_{j \in P(i)} \kappa_{ji}^k(\phi) \right\} \text{ für } i \in SV^k(\phi) \setminus \{r, s\} \quad (1.13)$$

Wenn wir mit l den Knoten mit der geringsten maximalen Durchflußmenge betrachten,

$$l := \arg \min_{i \in SV^0(\phi)} f_i^k$$

so können wir einen Fluß Ψ^k wie folgt konstruieren. Es sei

$$\left. \begin{array}{l} W(l, s) \dots \text{Weg von } l \text{ nach } s \\ W(r, l) \dots \text{Weg von } r \text{ nach } l \end{array} \right\} \text{ in } SN^k(\phi)$$

und

$$\Psi_{ij}^k := \begin{cases} f_l^k & \text{für } \langle i, j \rangle \text{ in } W(l, s) \text{ oder } W(r, l) \\ 0 & \text{sonst} \end{cases} \quad \text{mit } \omega(\Psi^k) = f_l^k \quad (1.14)$$

Wir beobachten weiters:

- (i) Alle Kanten $\langle i, j \rangle \in SE^k(\phi)$, für die $\kappa_{ij}^k(\phi) - \Psi_{ij}^k = 0$ gilt, scheiden als Kanten eines flußvergrößernden Weges (bzgl. Ψ^k) von r nach s aus. Wir eliminieren diese Kanten aus $SE^k(\phi)$.

(ii) Wenn wir mit

$$f_i^k := \min \left\{ \sum_{j \in S(i)} \kappa_{ij}^k(\phi) - \Psi_{ij}^k, \sum_{j \in P(i)} \kappa_{ji}^k(\phi) - \Psi_{ji}^k \right\} \text{ für } i \in SV^k(\phi) \setminus \{r, s\}$$

die maximale restliche Durchflußmenge des Knotens i betrachten, so können für alle Knoten $j \in SV^k(\phi)$, die $f_j^k = 0$ erfüllen, die entsprechend inzidenten Kanten aus $SE^k(\phi)$ gelöscht werden.

(iii) Zuletzt löschen wir alle isolierten Knoten in $SN^k(\phi)$.

Das aus den Filterkriterien (i) bis (iii) resultierende Netzwerk sei mit $SN^{k+1}(\phi) = (SV^{k+1}(\phi), SE^{k+1}(\phi); \kappa^{k+1}(\phi))$ bezeichnet, wobei wir

$$\kappa_{ij}^{k+1}(\phi) := \kappa_{ij}^k(\phi) - \Psi_{ij}^k \quad (1.15)$$

setzen. Wir bestimmen den Update zum Iterationsindex $k+1 \rightarrow k$ und starten in Analogie die nächste Iteration. Wir beenden die Iteration, sobald im Netzwerk $SN^k(\phi)$ entweder der Knoten r oder s eliminiert wurde und erhalten damit den in $SN(\phi)$ blockierenden Fluß gemäß

$$\tilde{\Psi} = \sum_k \Psi^k \quad \text{mit} \quad \omega(\tilde{\Psi}) = \sum_k f_l^k$$

Der Update des aktuellen Flusses ϕ im Netzwerk N erfolgt wiederum gemäß (1.12), wobei wir Ψ durch $\tilde{\Psi}$ ersetzen.

Beispiel 1.2(5): (\rightarrow `sec_max_flow_comput_5.m`). Wir betrachten das Schichtennetzwerk gemäß der Abbildung 1.13 und setzen $k = 0$. Mit 3 ist im Knoten 2 der minimale Durchfluß bestimmt und damit gilt $l = 2$. Wir erhalten damit den Fluß Ψ^0 gemäß der Abbildung 1.14. Die Filterkriterien (i) bis (iii) auf $SN^0(\phi)$ angewendet und der Update der Maximalkapazitäten liefert das Netzwerk $SN^1(\phi)$ gemäß Abbildung 1.15. Nachdem weder r noch s eliminiert wurden starten wir die nächste Iteration und finden mit $l = 7$ den Knoten mit minimaler Durchflußmenge im Netzwerk $SN^1(\phi)$. Damit konstruieren wir den Fluß Ψ^1 gemäß der Abbildung 1.16. Die Filterkriterien (i) bis (iii) auf $SN^1(\phi)$ angewendet und der Update der Maximalkapazitäten liefert das Netzwerk $SN^2(\phi)$ gemäß Abbildung 1.17. Offensichtlich wurde der Knoten s eliminiert und wir beenden die Iteration. Damit ergibt sich für das Schichtennetzwerk $SN(\phi)$ und das Inkrementnetzwerk $N(\phi)$ der Fluß $\tilde{\Psi}$ gemäß Abbildung 1.18 mit der Stärke $\omega(\tilde{\Psi}) = 7$. Wir berechnen gemäß (1.12) den vergrößerten Fluß $\tilde{\phi}$ in N - dieser ist in Abbildung 1.19 dargestellt.

Wir formulieren das MKM-Verfahren als Pseudocode (\rightarrow `MKM_MaxFlow.m` und `sec_max_flow_comput_6.m`):

$$[\phi, \omega] = \text{MKM_MaxFlow}[V, E, \lambda, \kappa, \phi, r, s]$$

% Initialisierung

> teste das Netzwerk auf Antisymmetrie

> bestimme $\omega = \omega(\phi)$ und teste Initialfluß auf Mengenbilanz und Zulässigkeit

> $b = false$ (bool'sche Variable, wird `true` gesetzt sobald maximaler Fluß gefunden)

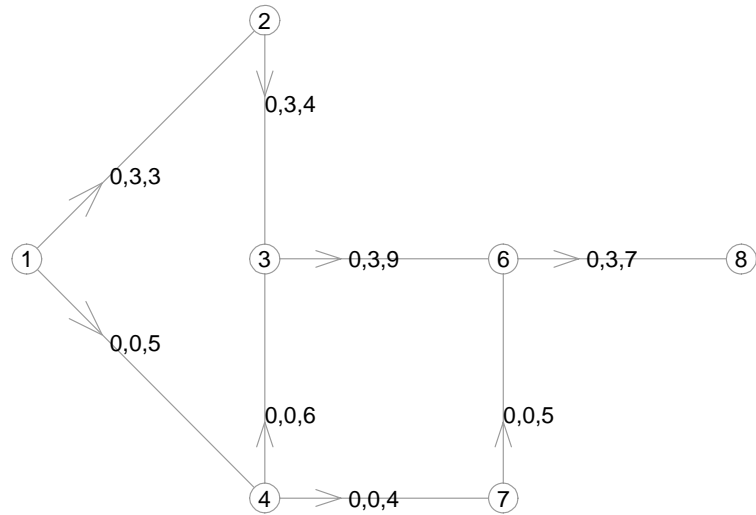


Abbildung 1.14: Schichtennetzwerk $SN^0(\phi)$ mit Fluß Ψ^0 . Die Kantenbeschriftungen zeigen die Minimalkapazität identisch Null, Ψ_{ij}^0 und $\kappa_{ij}^0(\phi)$.

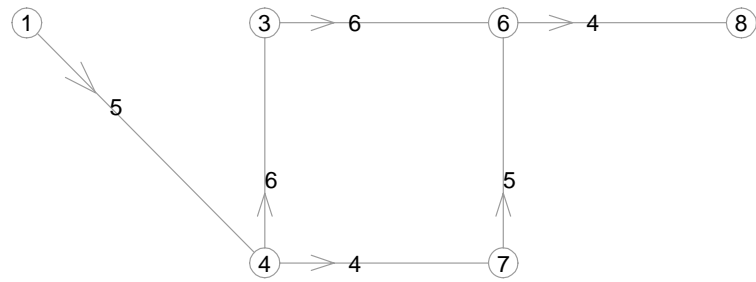


Abbildung 1.15: Schichtennetzwerk $SN^1(\phi)$. Die Kantenbeschriftungen zeigen $\kappa_{ij}^1(\phi)$.

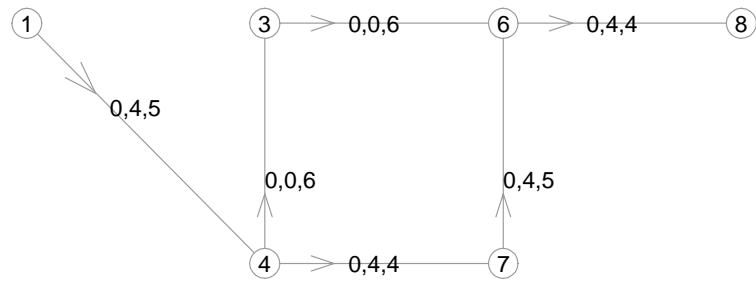


Abbildung 1.16: Schichtennetzwerk $SN^1(\phi)$ mit Fluß Ψ^1 . Die Kantenbeschriftungen zeigen die Minimalkapazität identisch Null, Ψ_{ij}^1 und $\kappa_{ij}^1(\phi)$.

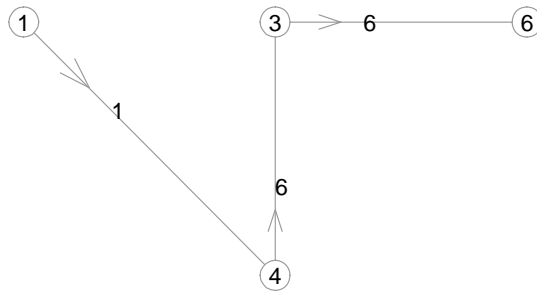


Abbildung 1.17: Schichtennetzwerk $SN^2(\phi)$. Die Kantenbeschriftungen zeigen $\kappa_{ij}^2(\phi)$.

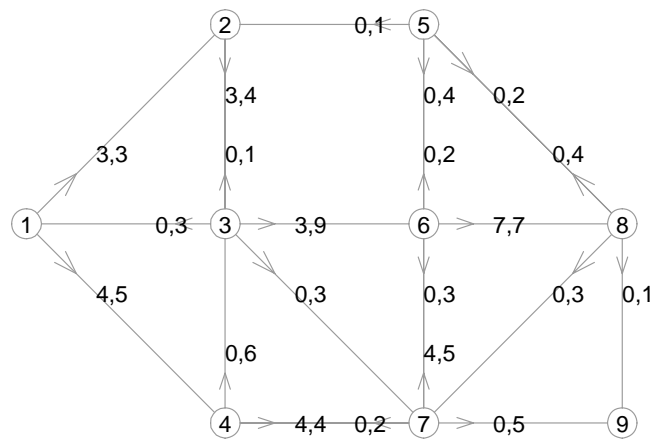


Abbildung 1.18: Inkrementnetzwerk $N(\phi)$ mit blockierendem Fluß des zugehörigen Schichtennetzwerkes $SN(\phi)$. Die Kantenbeschriftungen zeigen $\tilde{\Psi}_{i,j}$ und $\kappa_{ij}(\phi)$.

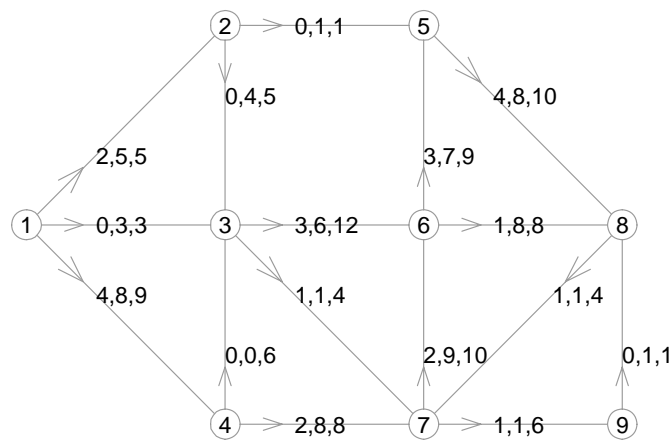


Abbildung 1.19: Vergrößerter Fluß $\tilde{\phi}$ im Netzwerk N mit Stärke $\omega(\tilde{\phi}) = 16$. Die Kantenbeschriftungen sind λ_{ij} , $\tilde{\phi}_{ij}$ und κ_{ij} .

% Iteration

```

> solange  $b = false$ , führe aus
>   bestimme das Inkrementzwerk  $N(\phi)$  bzgl. dem aktuellen Fluß  $\phi$ 
>   wenn  $s$  in  $N(\phi)$  nicht von  $r$  aus erreichbar ist, führe aus
>      $b = true$  ( $\phi$  ist maximal)
>   sonst führe aus
>     bestimme das zu  $N(\phi)$  gehörende Schichtnetzwerk  $SN(\phi)$ 
>     setze  $k = 0$ ,  $SN^0(\phi) = SN(\phi)$ ,  $\Psi_{ij}^0 = 0$  für  $i, j \in SV^0(\phi)$ ,  $blockflow = false$ 
>     solange  $blockflow = false$ , führe aus
>       bestimme  $f_i^k$  für alle  $i \in SV^k(\phi)$  gemäß (1.13)
>       bestimme  $l$ ,  $f_l^k$  und  $\Psi^k$  gemäß (1.14)
>       bestimme gemäß den Filterkriterien (i) bis (iii) und dem Update der
>       Maximalkapazitäten (1.15) das Netzwerk  $SN^{k+1}(\phi)$ 
>       wenn  $r \notin SV^{k+1}(\phi)$  oder  $r \notin SV^{k+1}(\phi)$ , führe aus
>          $blockflow = true$ 
>          $\tilde{\Psi} = \sum_k \Psi^k$  und  $\omega(\tilde{\Psi}) = \sum_k f_l^k$ 
>       sonst führe aus
>         setze  $k + 1 \rightarrow k$ 
>       setze  $\Psi = \tilde{\Psi}$  und bestimme den vergrößerten Fluß  $\tilde{\phi}$  gemäß (1.12)
>       setze  $\phi = \tilde{\phi}$ 

```

Die Komplexität von `MKM_MaxFlow` kann mit $O(|V|^3)$ abgeschätzt werden (\rightarrow Jungnickel, Seiten 171 ff).

1.2.7 Bestimmung eines zulässigen Initialflusses zum Maximalflußproblem

Alle bisher betrachteten Verfahren zur Lösung des Maximalfluß-Problems gehen davon aus, daß ein zulässiger Fluß ϕ für das Netzwerk $N = (V, E; \lambda, \kappa)$ bekannt ist. Wir wollen uns daher in diesem Abschnitt der Konstruktion eines derartigen Flusses in einem Netzwerk mit Kapazitäten widmen; oder erkennen, daß es keinen zulässigen Fluß für das entsprechende Netzwerk gibt.

Im Falle eines Netzwerkes mit verschwindenden Minimalkapazitäten, i.e. $\lambda_{ij} = 0$ für $i, j \in V$, ist mit dem Nullfluß $\phi_{ij} = 0$ für $i, j \in V$ ein zulässiger Initialfluß bestimmt.

Wir wollen daher Netzwerke betrachten, für die es zumindest eine Kante $\langle i, j \rangle \in E$ mit $\lambda_{ij} > 0$ gibt. Die Konstruktion eines zulässigen Initialflusses läßt sich durch die Aneinanderreihung von entsprechenden Problemtransformationen modular lösen. Als Ausgangsbasis formulieren wir nochmals das Initialfluß-Problem:

gesucht seien für die Quelle r und die Senke s die ϕ_{ij} für alle $\langle i, j \rangle \in E$, derart daß gilt

$$M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \begin{cases} \omega & i = r \\ -\omega & i = s \\ 0 & i \in V \setminus \{r, s\} \end{cases} \quad (\text{FP0})$$

$$\lambda_{ij} \leq \phi_{ij} \leq \kappa_{ij}$$

Modul 1: Initialflußproblem (FP0) \rightarrow Zirkulationsflußproblem (ZirkFP)

Wir konstruieren

$$N' := \begin{cases} (V, E; \kappa, \lambda) & \text{falls } \langle s, r \rangle \in E \quad (\text{Fall A}) \\ (V, E'; \lambda', \kappa') & \text{sonst} \quad (\text{Fall B}) \end{cases}$$

wobei im Fall B gilt, daß $E' = E \cup \{\langle s, r \rangle\}$, $\lambda'_{sr} = 0$ und $\kappa'_{sr} = \infty$. Wir erlauben also im Fall B einen Rückfluß von der Senke s zur Quelle r mit unendlicher Kapazität. Im Fall A ist der Rückfluß bereits vorhanden und beschränkt.

Wir formulieren nunmehr auf N' das folgende Problem

gesucht seien die ϕ'_{ij} für alle $\langle i, j \rangle \in E'$, derart daß gilt

$$M'(i) := \sum_{j \in S(i)} \phi'_{ij} - \sum_{k \in P(i)} \phi'_{ki} = 0 \quad \text{für } i \in V' \quad (\text{ZirkFP})$$

$$\lambda'_{ij} \leq \phi'_{ij} \leq \kappa'_{ij} \quad \text{für alle } \langle i, j \rangle \in E'$$

Ein ϕ , das (ZirkFP) erfüllt wird als *zulässiger Zirkulationsfluß* bezeichnet und das Problem daher als *Zirkulationsflußproblem*. Im Gegensatz zu einem Fluß ist bei einem Zirkulationsfluß die Mengenbilanz für alle Knoten in N' ausgeglichen - auch für r und s . Zirkulationsflußprobleme besitzen daher keine Quellen und Senken.

Aussage 5 *Es existiert ein zulässiger Fluß ϕ auf N genau dann, wenn ein zulässiger Zirkulationsfluß ϕ' auf N' existiert.*

\Rightarrow : ϕ sei zulässiger Fluß auf N mit Quelle r , Senke s und Stärke $\omega(\phi)$. Dann ist

$$\phi'_{ij} := \begin{cases} \phi_{ij} & \text{für } \langle i, j \rangle \in E' \setminus \{\langle s, r \rangle\} \\ \begin{cases} \phi_{sr} + \omega(\phi) & \text{im Fall A} \\ \omega(\phi) & \text{im Fall B} \end{cases} & \text{für } \langle i, j \rangle = \langle s, r \rangle \end{cases}$$

ein zulässiger Zirkulationsfluß auf $N' \rightarrow$ Abbildung 1.20.

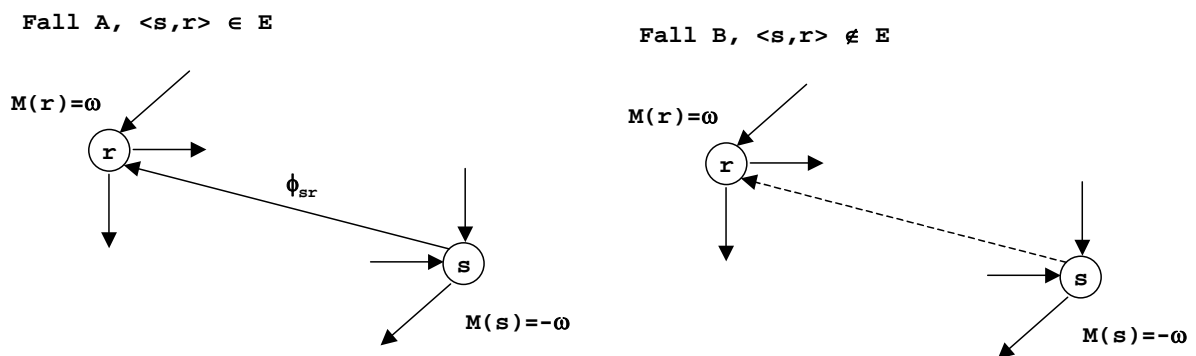


Abbildung 1.20: Konstruktion eines zulässigen Zirkulationsflusses aus einem zulässigen Fluß.

\Leftarrow : ϕ' sei zulässiger Zirkulationsfluß auf N' . Offensichtlich ist ein zulässiger Zirkulationsfluß auch ein zulässiger Fluß der Stärke Null auf N' . Damit setzen wir

$$\text{Fall A) } \phi_{ij} := \begin{cases} \phi'_{ij} & \text{für } \langle i, j \rangle \in E \setminus \{\langle s, r \rangle\} \\ 0 & \text{für } \langle i, j \rangle = \langle s, r \rangle \end{cases} \quad \begin{array}{l} \text{ist zulässig auf } N \\ \text{mit Stärke } \phi'_{sr} \end{array} \quad (1.16)$$

$$\text{Fall B) } \phi_{ij} := \phi'_{ij} \text{ für } \langle i, j \rangle \in E \text{ ist zulässig auf } N \text{ mit Stärke } \phi'_{sr} \quad (1.17)$$

Modul 2: Zirkulationsflußproblem (ZirkFP) \rightarrow zulässigen Fluß Problem (ZFP)

Das Problem (ZirkFP) läßt sich auf ein weiteres Problem transformieren. Wir wollen diese Transformation für allgemeine Netzwerke mit Kapazität beschreiben und gehen daher von einem Netzwerk $N = (V, E; \lambda, \kappa)$ aus und formulieren nochmals:

gesucht seien die ϕ_{ij} für alle $\langle i, j \rangle \in E$, derart daß gilt

$$M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = 0 \quad \text{für } i \in V \quad (\text{ZirkFP-Standard})$$

$$\lambda_{ij} \leq \phi_{ij} \leq \kappa_{ij} \quad \text{für alle } \langle i, j \rangle \in E$$

Wenn ϕ dem Problem (ZirkFP) genügt, dann gilt mit

$$\phi_{ij} := \phi'_{ij} + \lambda_{ij} \quad (1.18)$$

daß

$$M(i) = \sum_{j \in S(i)} (\phi'_{ij} + \lambda_{ij}) - \sum_{k \in P(i)} (\phi'_{ki} + \lambda_{ki}) = 0 \quad \text{für alle } i \in V$$

$$\lambda_{ij} \leq \phi'_{ij} + \lambda_{ij} \leq \kappa_{ij} \quad \text{für alle } \langle i, j \rangle \in E$$

bzw.

$$\sum_{j \in S(i)} \phi'_{ij} - \sum_{k \in P(i)} \phi'_{ki} = \underbrace{\sum_{k \in P(i)} \lambda_{ki} - \sum_{j \in S(i)} \lambda_{ij}}_{=: \nu_i} \quad \text{für alle } i \in V \quad (1.19)$$

$$0 \leq \phi'_{ij} \leq \underbrace{\kappa_{ij} - \lambda_{ij}}_{=: \kappa'_{ij}} \quad \text{für alle } \langle i, j \rangle \in E$$

Offensichtlich beschreiben die ν_i die Differenz zwischen minimalen An- und Abtransport im Knoten i . Gilt $\nu_i > 0$ so wird bei minimalen An- und Abtransport am Knoten i mehr an- als abtransportiert; für den Fall $\nu_i < 0$ mehr ab- als antransportiert. Insbesondere gilt, daß

$$\sum_{i \in V} \nu_i = 0$$

denn wenn wir für $u, w \in V$ ohne Beschränkung der Allgemeinheit die Kante $\langle u, w \rangle$ betrachten, so gilt

$$\nu_u = \dots - \lambda_{uw} \dots \quad \text{und} \quad \nu_w = \dots + \lambda_{uw} \dots$$

Wir können damit zusammenfassend behaupten: die Suche nach einem zulässigen Zirkulationsfluß auf N ist mittels der Transformation (1.18) äquivalent zur Lösung des Problems:

gesucht seien die ϕ'_{ij} für alle $\langle i, j \rangle \in E$, derart daß gilt

$$M'(i) := \sum_{j \in S(i)} \phi'_{ij} - \sum_{k \in P(i)} \phi'_{ki} = \nu_i \quad \text{für alle } i \in V \quad (\text{ZFP})$$

$$0 \leq \phi'_{ij} \leq \kappa'_{ij} \quad \text{für alle } \langle i, j \rangle \in E \quad \text{und} \quad \sum_{i \in V} \nu_i = 0$$

Wir bezeichnen (ZFP) als das *zulässigen Fluß Problem*.

Modul 3: Lösung des zulässigen Fluß Problem (ZFP)

Wir beschreiben die Lösung des Problems wiederum in einem allgemeinen Netzwerk mit verschwindenden Minimalkapazitäten $N = (V, E; \kappa)$ und formulieren daher

gesucht seien die ϕ_{ij} für alle $\langle i, j \rangle \in E$, derart daß gilt

$$M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \nu_i \quad \text{für alle } i \in V \quad (\text{ZFP-Standard})$$

$$0 \leq \phi_{ij} \leq \kappa_{ij} \quad \text{für alle } \langle i, j \rangle \in E \quad \text{und} \quad \sum_{i \in V} \nu_i = 0$$

Zur Lösung von (ZFP-Standard) konstruieren wir ein erweitertes Netzwerk $N' = (V', E'; \kappa')$ gemäß

$$V' = V \cup \{q, t\}$$

wobei q eine Quelle und t eine Senke in N' darstellen. Zur Konstruktion von E' übernehmen wir alle Kanten aus E und führen zusätzliche Kanten wie folgt ein:

$$\text{gilt } \begin{cases} \nu_i > 0 \\ \nu_i < 0 \end{cases}, \text{ so ergänze } E \text{ mit der Kante } \begin{cases} \langle q, i \rangle \text{ mit } \kappa'_{qi} := \nu_i \\ \langle i, t \rangle \text{ mit } \kappa'_{it} := -\nu_i \end{cases} \quad (1.20)$$

Anschließend lösen wir ein Maximalflußproblem auf N' , wobei der Nullfluß ein zulässiger Initialfluß ist. Wir bezeichnen ϕ' als den gefundenen Maximalfluß von q nach t in N' .

Wir bezeichnen ϕ' als *gesättigten Fluß* in N' , wenn gilt:

$$\begin{aligned} \phi'_{qi} &= \kappa'_{qi} \quad \text{für alle } \langle q, i \rangle \in E' \\ \phi'_{it} &= \kappa'_{it} \quad \text{für alle } \langle i, t \rangle \in E' \end{aligned} \quad (1.21)$$

Es ist offensichtlich, daß jeder gesättigte Fluß ein maximaler Fluß ist - die Umkehrung ist im Allgemeinen falsch.

Aussage 6 Genau dann, wenn in N' jeder maximale Fluß ϕ' gesättigt ist, existiert eine Lösung des zulässigen Fluß Problems (ZFP-Standard).

\Leftarrow : ϕ sei zulässig bzgl. (ZFP-Standard). Dann konstruieren wir gemäß

$$\phi'_{ij} := \phi_{ij} \text{ für alle } \langle i, j \rangle \in E$$

$$\phi'_{qi} := \nu_i \text{ und } \phi'_{it} := -\nu_i$$

einen sättigenden Fluß auf N' . Wir haben nachzuweisen, daß ϕ' (1.1) erfüllt ist. Sei $i \in V' \setminus \{q, t\}$. Dann gilt unter der Berücksichtigung von (1.20), daß

$$\underbrace{\sum_{j \in S(i) \setminus \{t\}} \phi'_{ij} - \sum_{k \in P(i) \setminus \{q\}} \phi'_{ki}}_{=\nu_i} \begin{cases} + \phi'_{it} & \text{für } \langle i, t \rangle \in E' \\ - \phi'_{qi} & \text{für } \langle q, i \rangle \in E' \end{cases} = 0$$

Weiters folgt aus (1.20) unmittelbar, daß

$$\sum_{i \in V} \nu_i = \sum_{i \in S(q)} \kappa'_{qi} - \sum_{i \in P(t)} \kappa'_{it} = 0$$

und somit, daß

$$\sum_{i \in S(q)} \phi'_{qi} = \sum_{i \in P(t)} \phi'_{it}$$

\Rightarrow : ϕ' sei maximaler Fluß auf N' und sättigend. Dann ist

$$\phi_{ij} := \phi'_{ij} \text{ für alle } \langle i, j \rangle \in E$$

eine Lösung von (ZFP-Standard). Wir haben also zu zeigen, daß

$$\sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \nu_i \text{ für } i \in V$$

In N' gilt, daß

$$\sum_{j \in S(i)} \phi'_{ij} - \sum_{k \in P(i)} \phi'_{ki} = 0 \text{ für } i \in V' \setminus \{q, t\}$$

Wir können wiederum unter der Berücksichtigung von (1.20) behaupten, daß

$$\underbrace{\sum_{j \in S(i) \setminus \{t\}} \phi'_{ij} - \sum_{k \in P(i) \setminus \{q\}} \phi'_{ki}}_{=0} \begin{cases} + \phi'_{it} & \text{für } \langle i, t \rangle \in E' \\ - \phi'_{qi} & \text{für } \langle q, i \rangle \in E' \end{cases} = 0 \text{ für } i \in V' \setminus \{q, t\}$$

und da ϕ' gesättigt ist, folgt mit $\phi'_{qi} = \kappa'_{qi} = \nu_i$ und $\phi'_{it} = \kappa'_{it} = -\nu_i$ die Richtigkeit der nachzuweisenden Beziehung.

Zusammenfassend können wir damit den folgenden Pseudocode zur Bestimmung einer Lösung zu (FP0) angeben:

$$[\phi, \omega] = \text{INI_Flow}[V, E, \lambda, \kappa, r, s]$$

```

% Initialisierung (der Index (k) steht für das zum Modul k gehörende Netzwerk)
>  $V' := V$ ,  $E' := E$ ,  $\lambda' := \lambda$ ,  $\kappa' := \kappa$  ( $N'$  sei zu modifizierendes Hilfsnetzwerk)
>  $caseA = true$  (bool'sche Variable, true für den Fall A)

% Modul 1 (FP0  $\rightarrow$  ZirkFP)
> wenn  $\langle s, r \rangle \notin E'$ , führe aus
>    $E' \leftarrow E' \cup \{\langle s, r \rangle\}$ ,  $\lambda'_{sr} := 0$ ,  $\kappa'_{sr} := \infty$ 
>    $caseA = false$ 

% Modul 2 (ZirkFP  $\rightarrow$  ZFP)
> für alle  $i \in V$  führe aus
>   berechne  $v_i$  gemäß (1.19)
> für alle  $\langle i, j \rangle \in E'$  führe aus
>    $\kappa'_{ij} \leftarrow \kappa'_{ij} - \lambda'_{ij}$ 

% Modul 3 (ZFP)
>  $V' \leftarrow V' \cup \{q, t\}$ 
> für alle  $i \in V$  führe aus
>   wenn  $\nu_i > 0$ , dann führe aus
>      $E' \leftarrow E' \cup \{\langle q, i \rangle\}$ ,  $\kappa'_{qi} := \nu_i$ 
>   wenn  $\nu_i < 0$ , dann führe aus
>      $E' \leftarrow E' \cup \{\langle i, t \rangle\}$ ,  $\kappa'_{it} := -\nu_i$ 

% Lösung
> bestimme den maximalen Fluß  $\phi'$  von  $q$  nach  $t$ 
> im Netzwerk  $N' = (V', E', \kappa')$ 
> wenn  $\phi'$  gesättigt, dann führe aus
>    $\omega = \phi'_{sr}$ 
>   für alle  $\langle i, j \rangle \in E$  führe aus
>      $\phi_{ij} = \phi'_{ij} + \lambda_{ij}$ 
> sonst führe aus
>   Rückmeldung: es existiert keine zulässige Lösung zu (FP0)

```

Beispiel 1.2(7): wir betrachten das Netzwerk mit Kapazitäten $N = (V, E; \lambda, \kappa)$ gemäß der Abbildung 1.21 (\rightarrow `sec_max_flow_comput_7.m`, `INI_Flow.m`). Die Abbildung 1.22 zeigt das Netzwerk zur Lösung von (ZirkFP) und die Abbildung 1.23 das Netzwerk zur Lösung des Problems (ZFP). Die Abbildung 1.24 zeigt das erweiterte Netzwerk zur Lösung von (ZFP) und den berechneten sättigenden Fluß. Der aus ϕ' gemäß den obigen Ausführungen konstruierte zulässige Fluß für N ist in Abbildung 1.25 dargestellt.

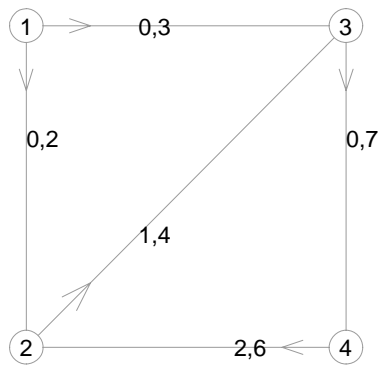


Abbildung 1.21: Netzwerk mit Kapazitäten. Die Kantenbeschriftungen bedeuten λ_{ij} und κ_{ij} .

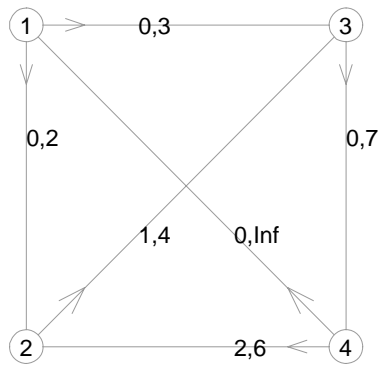


Abbildung 1.22: Netzwerk zur Betrachtung des Zirkulationsproblems. Die Kantenbeschriftungen zeigen die Minimalkapazitäten λ'_{ij} und die Maximalkapazitäten κ'_{ij} .

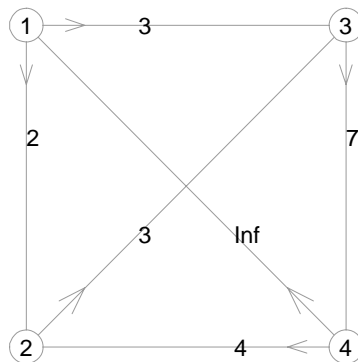


Abbildung 1.23: Netzwerk zur Betrachtung des zulässigen Fluß Problems. Die Kantenbeschriftungen zeigen die Maximalkapazitäten κ'_{ij} .

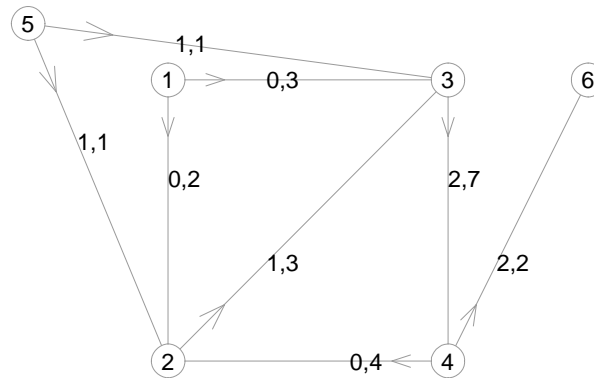


Abbildung 1.24: Erweitertes Netzwerk zum zulässigen Fluß Problem. Die Kantenbeschriftungen zeigen den sättigenden Fluß ϕ'_{ij} und die entsprechenden Maximalkapazitäten κ'_{ij} . q entspricht dem Knoten 5 und t dem Knoten 6.

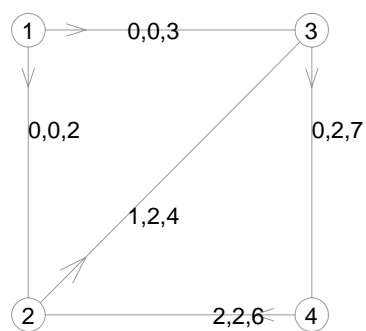


Abbildung 1.25: Netzwerk mit Kapazitäten und zulässigem Fluß. Die Kantenbeschriftungen zeigen die Minimalkapazitäten λ_{ij} , den Fluß ϕ_{ij} und die Maximalkapazitäten κ_{ij} .

1.3 Kostenoptimale Flüsse in Netzerken

1.3.1 Definition und allgemeine Zusammenhänge

1.3.1.1 Das Problem (KFS)

Ausgehend von einem Netzwerk mit Kapazitäten versteht man unter $N = (V, E; \lambda, \kappa; c)$ ein *Netzwerk mit Kapazitäten und Kosten*, wobei

$$c : E \rightarrow R$$

und die c_{ij} die Kosten pro transportierter Mengeneinheit auf der Kante $\langle i, j \rangle$ darstellen. Wenn wir mehrere zulässige Flüsse ϕ der Stärke ω von r nach s betrachten, so können wir das folgende Optimierungsproblem formulieren:

$$\begin{aligned} \min_{\phi} \sum_{\langle i, j \rangle \in E} c_{ij} \phi_{ij} \quad & \text{unter der Berücksichtigung von} \\ M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \begin{cases} \omega & i = r \\ -\omega & i = s \\ 0 & i \in V \setminus \{r, s\} \end{cases} & \quad (\text{KFS}) \\ \lambda_{ij} \leq \phi_{ij} \leq \kappa_{ij} \end{aligned}$$

Bemerkung: da $\sum c_{ij} \phi_{ij}$ die Gesamtkosten des Flusses ϕ beschreibt, bezeichnen wir (KFS) als *kostenminimales Flußproblem vorgegebener Stärke*. Man beachte in diesem Zusammenhang, daß die Flußstärke ω (fest) vorgegeben ist - wir suchen also unter allen zulässigen Flüssen der Stärke ω nach dem kostengünstigsten - dieser heißt dann auch ein ω -*optimaler Fluß*.

Wir wollen den Zusammenhang zwischen der Bestimmung (im Sinne der Kosten) kürzester Wege in Netzwerken und dem Problem (KFS) aufzeigen. Wenn wir mit $N = (V, E; c)$ ein bewertetes Netzwerk ohne Zyklen negativer Länge² betrachten und den kostengünstigsten Weg von r nach $s \in \dot{R}(r)$ suchen, so können wir jedem kürzesten Weg W von r nach s einen Fluß der Stärke 1 zuordnen

$$\phi_{ij} := \begin{cases} 1 & \text{für } \langle i, j \rangle \in W \\ 0 & \text{sonst} \end{cases} \quad (1.22)$$

der die Mengenbilanz erfüllt und insbesondere zulässig ist, wenn

$$\lambda_{ij} := 0 \quad \text{und} \quad \kappa_{ij} := \infty \quad \text{für alle } \langle i, j \rangle \in E$$

gilt. Damit läßt sich die folgende Aussage treffen:

Aussage 7 *Jede Lösung des kostenoptimalen Wegeproblems in einem bewerteten Netzwerk $N = (V, E; c)$ stellt auch eine Lösung des Problems (KFS) im Netzwerk mit Kapazitäten und Kosten $N = (V, E; \lambda, \kappa; c)$ für $\omega = 1$ dar.*

²Wir erlauben zur Bestimmung kostenoptimaler Wege auch negative Kantenbewertungen - i.e. $c_{ij} \in R \cup \{\infty\}$.

Bemerkung: wir können also (KFS) für den Fall $\omega = 1$ mittels Methoden der kürzesten Wege-Bestimmung (z.B. Label Correcting- oder Label Setting Verfahren) lösen. Umgekehrt kann bei Existenz eines optimalen Flusses für $\omega = 1$ ein optimaler Fluß der Form (1.22) gefunden werden - dies ist nicht unmittelbar einsichtig, sondern ein Ergebnis zur Untersuchung von kostenoptimalen ganzzahligen Flußproblemen (\rightarrow Bertsekas, Seite 11 und Kapitel 5).

Weiters können wir einen Zusammenhang zwischen dem Maximalflußproblem (MFP) und dem Problem (KFS) herleiten. Wir betrachten dazu ein Netzwerk mit Kapazitäten $N = (V, E; \lambda, \kappa)$, die Flußquelle r und die Flußsenke s . Weiters setzen wir voraus, daß $\langle s, r \rangle \notin E$ gilt³. Wir fügen zum Netzwerk N die Kante $\langle s, r \rangle$ hinzu und erhalten ein neues Netzwerk mit Kapazitäten und Kosten $N' = (V, E'; \lambda', \kappa'; c')$, wobei $\lambda'_{sr} := 0$, $\kappa'_{sr} := \infty$ und

$$c'_{ij} := \begin{cases} -1 & \text{für } i = s \text{ und } j = r \\ 0 & \text{sonst} \end{cases}$$

Aussage 8 Der bzgl. (KFS) für $\omega = 0$ auf dem Netzwerk N' kostenoptimale Fluß ϕ' definiert mit $\phi'_{ij} := \phi'_{ij}$ für alle $\langle i, j \rangle \in E$ den in (MFP) gesuchten Fluß mit Stärke ϕ'_{sr} auf N .

Um dies einzusehen, kann man wie folgt argumentieren. Es existiert die Kante $\langle s, r \rangle \in E'$ und mit der Forderung $\omega = 0$ wird der gesamte Fluß ϕ des Netzwerkes N im Netzwerk N' zur Quelle zurückgeleitet - man beachte: für $\omega = 0$ entspricht (KFS) einem *kostenoptimalen Zirkulationsproblem* (\rightarrow vergl. Abschnitt 1.2.7 auf Seite 25). Die Zielfunktion in (KFS) lautet $-\phi'_{sr}$, so daß im Optimum ϕ'_{sr} so groß wie möglich ist. Damit ist auch die Stärke von ϕ maximal.

1.3.1.2 Das Problem (KFS) als lineares Programm

Das Problem (KFS) ist weiters als lineares Programm darstellbar. Ausgehend vom Netzwerk $N = (V, E; \lambda, \kappa; c)$ sei $V = \{1, \dots, n\}$ und $E = \{e_1, \dots, e_m\}$. Dann betrachten wir die Inzidenzmatrix $H \in \text{Mat}(n \times m)$, mit $H = [h_{i\mu}]$, wobei

$$h_{i\mu} := \begin{cases} 1 & \text{für } e_\mu \text{ positiv inzident zu } i \in V \\ -1 & \text{für } e_\mu \text{ negativ inzident zu } i \in V \\ 0 & \text{sonst} \end{cases}$$

ϕ sei ein Fluß auf N der Stärke $\omega \geq 0$. Mit $\phi(e_\mu)$ als Flußbelegung der Kante e_μ könne wir ϕ als Vektor interpretieren - i.e. $\phi \in R^m$ und $\phi_\mu := \phi(e_\mu)$. Analog definieren wir den Vektor $\Omega \in R^n$ gemäß

$$\Omega_i := \begin{cases} \omega & \text{für } i = r \\ -\omega & \text{für } i = s \\ 0 & \text{sonst} \end{cases}$$

Damit läßt sich die Nebenbedingung der Mengenbilanz in (KFS) schreiben als

$$M(i) = \sum_{\mu=1}^m h_{i\mu} \phi_\mu = \Omega_i \quad \text{für } i \in V$$

³Dies stellt keine Einschränkung der Allgemeinheit dar; sollte $\langle s, r \rangle \in E$ gelten so können wir durch Hinzunahme eine neuen Knoten k die Kante $\langle s, r \rangle$ durch die Kanten $\langle s, k \rangle$ und $\langle k, r \rangle$ ersetzen.

oder vektorwertig

$$H\phi = \Omega$$

Auf analoge Weise definieren wir aus $\lambda(e_\mu)$, $\kappa(e_\mu)$ und $c(e_\mu)$ die Vektoren $\lambda \in R^m$, $\kappa \in R^m$ und $c \in R^m$, so daß (KFS) geschrieben werden kann als

$$\min_{\phi \in R^m} c^T \phi \quad \text{unter der Berücksichtigung von} \quad (\text{KFSLP})$$

$$H\phi = \Omega \quad \text{und} \quad \lambda \leq \phi \leq \kappa$$

1.3.1.3 Das Problem (MKF)

Im Zusammenhang mit kostenoptimalen Flüssen läßt sich ein weiteres Problem formulieren. Wir betrachten wiederum ein Netzwerk mit Kapazitäten und Kosten $N = (V, E; \lambda, \kappa; c)$ und formulieren

$$\begin{aligned} \min_{\phi} \sum_{\langle i, j \rangle \in E} c_{ij} \phi_{ij} \quad & \text{unter der Berücksichtigung von} \\ M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \begin{cases} \omega & i = r \\ -\omega & i = s \\ 0 & i \in V \setminus \{r, s\} \end{cases} & (\text{MKF}) \\ \lambda_{ij} \leq \phi_{ij} \leq \kappa_{ij} \quad \text{und} \quad \max \omega & \end{aligned}$$

(MKF) sucht unter allen maximalen Flüssen im Netzwerk N nach dem kostengünstigsten Fluß - im Gegensatz zu (KFS) ist nunmehr ω nicht vorgegeben sondern als Optimierungsnebenbedingung definiert.

1.3.1.4 Transformation des Problems (KFS)

Die Lösung von (KFS) erfolgt in einem der nächsten Abschnitte mit einem Verfahren, daß von Netzwerken mit verschwindender Minimalkapazität ausgeht. Wir wollen daher dieses Problem auf ein äquivalentes Probleme, das dieser Forderung gerecht wird, transformieren. Dazu betrachten wir ein allgemeines Netzwerk mit Kapazitäten und Kosten $N = (V, E; \lambda, \kappa; c)$ mit Quelle r und Senke s . Für eine Lösung ϕ mit Stärke ω zu (KFS) folgt mit der Transformation

$$\phi'_{ij} := \phi_{ij} - \lambda_{ij} \quad \text{für alle } \langle i, j \rangle \in E$$

das äquivalente Problem

$$\begin{aligned} \min_{\phi'} \sum_{\langle i, j \rangle \in E} c_{ij} \phi'_{ij} \quad & \text{unter der Berücksichtigung von} \\ M(i) := \sum_{j \in S(i)} \phi'_{ij} - \sum_{k \in P(i)} \phi'_{ki} = \begin{cases} \nu_i + \omega & i = r \\ \nu_i - \omega & i = s \\ \nu_i & i \in V \setminus \{r, s\} \end{cases}, \quad \nu_i \text{ gemäß (1.19)} & (\text{KFS}') \\ 0 \leq \phi'_{ij} \leq \kappa'_{ij} \quad \text{mit } \kappa'_{ij} := \kappa_{ij} - \lambda_{ij} \quad & \text{für } \langle i, j \rangle \in E \end{aligned}$$

auf dem Netzwerk $N' = (V, E; \kappa'; c)$ betrachtet.

Aussage 9 ϕ genügt (KFS) genau dann, wenn ϕ' dem Problem (KFS') genügt.

Wir konstruieren in Anlehnung zum Abschnitt 1.2.7 ein erweitertes Netzwerk $N'' = (V'', E''; \kappa''; c'')$ wie folgt:

$$V'' := V \cup \{q, t\}$$

wobei q wiederum die Flußquelle und t die Flußsenke in N'' sei. Analog ergänzen wir E zu E'' gemäß der Regel: gilt für $i \in V$

$$\begin{cases} \nu_i > 0 \\ \nu_i < 0 \end{cases}, \text{ so ergänze } E \text{ mit der Kante } \begin{cases} \langle q, i \rangle \text{ mit } \kappa''_{qi} := \nu_i \text{ und } c''_{qi} := 0 \\ \langle i, t \rangle \text{ mit } \kappa''_{it} := -\nu_i \text{ und } c''_{it} := 0 \end{cases} \quad (1.23)$$

Wir treffen die zusätzliche Annahme, daß $\langle q, r \rangle \notin E''$ und $\langle s, t \rangle \notin E''$. Sollte dies der Fall sein, müssen wir den entsprechenden Pfeil mittels eines Hilfsknoten umlenken⁴. Wir fügen dann im Netzwerk N'' die Kanten $\langle q, r \rangle$ und $\langle s, t \rangle$ ein, wobei

$$\kappa''_{qr} = \kappa''_{st} := \omega$$

$$c''_{qr} = c''_{st} := c \quad \text{mit} \quad c > \sum_{\langle i, j \rangle \in E} c_{ij}$$

Dann können wir die folgende Aussage formulieren:

Aussage 10 Es sei ϕ'' der auf N'' kostenoptimale Fluß von q nach t der Stärke $\sum_{j \in S(q)} \kappa''_{qj} = \sum_{k \in P(t)} \kappa''_{kt}$; i.e. ϕ'' ist gesättigt. Dann ist $\phi'_{ij} := \phi''_{ij}$, für $\langle i, j \rangle \in E$ Lösung zum Problem (KFS').

Nachweis. Wir zeigen zunächst die Zulässigkeit von ϕ' im Sinne von (KFS'). Es sei $i \in V'' \setminus \{q, t\}$ betrachtet. Da ϕ'' ein Fluß ist gilt nach der Mengenbilanz (1.1), daß

$$M''(i) = \sum_{j \in S(i) \setminus \{t\}} \phi''_{ij} - \sum_{k \in P(i) \setminus \{q\}} \phi''_{ki} + \begin{cases} \phi''_{it} & \text{wenn } \langle i, t \rangle \in E'' \\ -\phi''_{qi} & \text{wenn } \langle q, i \rangle \in E'' \end{cases} = 0$$

Wir untersuchen zunächst ein $i \notin \{r, s\}$. Dann gilt aber in N'' , daß entweder $\langle i, t \rangle \in E''$ oder $\langle q, i \rangle \in E''$. Mit der Sättigung von ϕ'' und (1.23) folgt für die beiden Fälle:

$$\langle i, t \rangle \in E'' \Rightarrow \phi''_{it} = \kappa''_{it} = -\nu_i \quad \text{oder} \quad \langle q, i \rangle \in E'' \Rightarrow \phi''_{qi} = \kappa''_{qi} = \nu_i$$

In beiden Fällen gilt also für $i \notin \{r, s\}$, daß

$$\sum_{j \in S(i) \setminus \{t\}} \phi''_{ij} - \sum_{k \in P(i) \setminus \{q\}} \phi''_{ki} = \nu_i$$

und unter der Berücksichtigung, daß $\phi'_{ij} = \phi''_{ij}$ auf N gilt, somit auch

$$\sum_{j \in S(i)} \phi'_{ij} - \sum_{k \in P(i)} \phi'_{ki} = \nu_i \quad \text{für } i \in V \setminus \{r, s\}$$

⁴Sollte also $\langle q, r \rangle \in E''$ gelten, so ersetzen wir diese Kante durch die Kanten $\langle q, k \rangle$ und $\langle k, r \rangle$ und übernehmen die Kapazitäten und Kosten von $\langle q, r \rangle$.

Es sei nun $i = r$. Nach Voraussetzung gilt, daß $\langle q, r \rangle \in E''$. Insbesondere ist aber auch $\langle r, t \rangle \in E''$, da $\nu_r < 0$ in N' gilt. Damit gilt in N'' für die Mengenbilanz, daß

$$M''(r) = \sum_{j \in S(r) \setminus \{t\}} \phi''_{rj} - \sum_{k \in P(r) \setminus \{q\}} \phi''_{kr} + \phi''_{rt} - \phi''_{qr} = 0$$

Aufgrund der Sättigung von ϕ'' gilt aber, daß $\phi''_{qr} = \kappa''_{qr} = \omega$ und $\phi''_{rt} = \kappa''_{rt} = -\nu_r$. Übertragen auf N folgt somit, daß

$$\sum_{j \in S(r)} \phi'_{rj} - \sum_{k \in P(r)} \phi'_{kr} = \nu_r + \omega$$

Auf analoge Weise können wir für den Fall $i = s$ zeigen, daß

$$\sum_{j \in S(s)} \phi'_{sj} - \sum_{k \in P(s)} \phi'_{ks} = \nu_s - \omega$$

Um die Optimalität von ϕ' im Sinne von (KFS') nachzuweisen, argumentieren wir wie folgt. Es gilt aufgrund der Konstruktion von N'' , daß

$$\begin{aligned} \sum_{\langle i, j \rangle \in E''} c''_{ij} \phi''_{ij} &= \sum_{\langle i, j \rangle \in E} c''_{ij} \phi''_{ij} + \sum_{j \in S(q)} c''_{qj} \phi''_{qj} + \sum_{k \in P(t)} c''_{kt} \phi''_{kt} = \\ &= \sum_{\langle i, j \rangle \in E} c_{ij} \phi'_{ij} + \underbrace{c''_{qr} \phi''_{qr} + c''_{st} \phi''_{st}}_{(*)} \end{aligned}$$

ϕ'' ist der kostengünstigste aller gesättigten Flüsse. Unter der Voraussetzung der Sättigung stellen die ϕ''_{qr} und ϕ''_{st} jedoch keine Optimierungsvariablen mehr dar, sondern sind mit der entsprechenden Maximalkapazität belegt. Ist damit $\sum_{\langle i, j \rangle \in E''} c''_{ij} \phi''_{ij}$ minimal, so muß auch $\sum_{\langle i, j \rangle \in E} c_{ij} \phi'_{ij}$ minimal sein. ■

Bemerkung: Die Bestimmung des kostenoptimalen Flusses ϕ'' mit vorgegebener Stärke entspricht damit einem Problem von Typ (KFS) im Netzwerk N'' mit verschwindenden Minimalkapazitäten.

Beispiel 1.3(1): Wir betrachten das Netzwerk $N = (V, E; \lambda, \kappa; c)$ gemäß der Abbildung 1.26. Wir suchen einen kostenminimalen Fluß von der Quelle 1 zur Senke 4. Das entsprechende Netzwerk $N'' = (V'', E''; \kappa''; c'')$ ist in Abbildung 1.27 dargestellt. Siehe auch Routine `sec_cost_opt_flow_1.m` und `N4KFS2Ndd.m`.

1.3.1.5 Lösung des Problems (MKF) mittels dem Problem (KFS)

Das im nächsten Abschnitt beschriebene Verfahren nach Busacker und Gowen löst das Problem (KFS) auf einem Netzwerk mit verschwindenden Minimalkapazität, in dem eine Folge kostenminimaler Flüsse mit wachsender Stärke konstruiert wird. Das Verfahren kann also genau dann abgebrochen werden, wenn die vorgegebene Stärke ω erreicht wurde oder die Stärke nicht mehr vergrößert werden kann.

Damit ist aber auch das Problem (MKF) gelöst. Wir wissen, daß die maximale Flußstärke im Ausgangsnetzwerk N mit

$$\omega_{\max} = \sum_{j \in S(r)} \kappa_{rj} - \sum_{k \in P(r)} \lambda_{kr} \quad (1.24)$$

beschränkt ist. Wenn wir also die vorgegebene Stärke ω mit $\omega > \omega_{\max}$ festsetzen, lösen wir mit dem Verfahren von Busaker und Gowen das Problem (MKF) in endlich vielen Schritten.

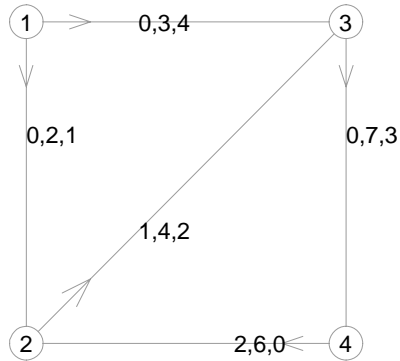


Abbildung 1.26: Netzwerk $N = (V, E; \lambda, \kappa, c)$. Die Kantenbeschriftungen zeigen λ_{ij} , κ_{ij} und c_{ij} .

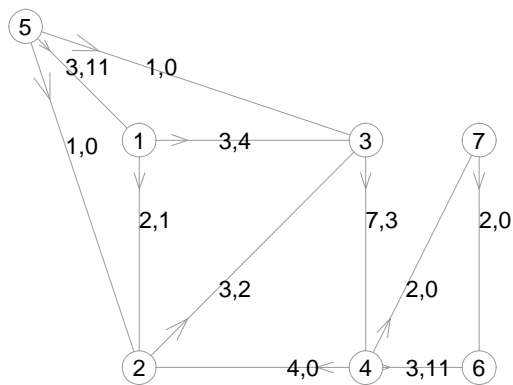


Abbildung 1.27: Netzwerk $N'' = (V'', E''; \kappa'', c'')$. Die Kantenbeschriftungen zeigen κ''_{ij} und c''_{ij} . Weiters sind $q = 5$ und $t = 6$ definiert - der Knoten 7 ist ein Hilfsknoten zur Auflösung der Kante $\langle s, t \rangle$ in N'' .

1.3.2 Berechnung kostenminimaler Flüsse nach Busacker und Gowen

Wenn wir das Netzwerk $N = (V, E; \kappa; c)$ mit Quelle r und Senke s betrachten so treffen wir die folgenden Annahmen, die keine Einschränkung der Allgemeinheit darstellen und die sequentiell abzuarbeiten sind.

- (i) N ist antisymmetrisch. Ist N nicht antisymmetrisch zu transformieren wir N auf das Netzwerk N' gemäß Abschnitt 1.2.4 auf Seite 11 und ersetzen anschließend $N \leftarrow N'$.
- (ii) Die Kosten sind nicht-negativ, i.e. $c_{ij} \geq 0$ für alle $\langle i, j \rangle \in E$. Sollte dies für eine Kante $\langle i, j \rangle \in E$ nicht erfüllt sein so ersetzen wir diese durch die entsprechende Umkehrkante; jedoch unter der Forderung dabei nicht die Zulässigkeit, Stärke und insbesondere Kosten eines Flusses zu verändern. Wir argumentieren daher wie folgt: für einen zulässigen Fluß ϕ im Netzwerk N ist die Mengenbilanz erfüllt und muß auch bei einem Tausch der Kante $\langle i, j \rangle$ gegen die Kante $\langle j, i \rangle$ erfüllt bleiben; also

$$M(j) = \sum_{l \in S(j)} \phi_{jl} - \sum_{l \in P(j) \setminus \{i\}} \phi_{kj} - \phi_{ij} \stackrel{!}{=} \sum_{l \in S(j) \setminus \{i\}} \phi_{jl} + \phi_{ji} - \sum_{l \in P(j)} \phi_{kl}$$

Damit folgt, daß $\phi_{ji} = -\phi_{ij}$ sein muß und entsprechendes auch für die Kapazitäten gilt. Damit auch die Kosten invariant sind muß $c_{ji} = -c_{ij}$ gelten. transformieren wir das Netzwerk N gemäß der Regel: für alle Kanten $\langle i, j \rangle \in E$ mit $c_{ij} < 0$ ersetze diese durch die Kanten $\langle j, i \rangle$ mit $c_{ji} := -c_{ij}$, $\lambda_{ji} := -\kappa_{ij}$ und $\kappa_{ji} := -\lambda_{ij}$. Wir erhalten wiederum ein Netzwerk N' und ersetzen anschließend $N \leftarrow N'$.

- (iii) $\lambda \equiv 0$. Sollte dies nicht der Fall sein⁵, so transformieren wir N auf N'' gemäß dem Abschnitt 1.3.1.4 auf Seite 35 und ersetzen anschließend $N \leftarrow N''$.
- (iv) $\kappa_{ij} > 0$ für alle $\langle i, j \rangle \in E$. Sollte es ein $\langle i, j \rangle \in E$ mit $\kappa_{ij} = 0$ geben, so folgt, daß für jeden Fluß $\phi_{ij} = 0$ gelten muß. Wir können daher die Kante $\langle i, j \rangle$ aus E eliminieren. Sollte danach gelten, daß $s \notin \dot{R}(r)$, so ist nur der Nullfluß in N zulässig - damit ist das Problem (KFS) nicht bzw. das Problem (MKF) nur trivial lösbar.

Wir formulieren nunmehr das Problem (KFS) für ein vorgegebenes $\omega^* \in R_+$ auf dem Netzwerk N unter Berücksichtigung der Voraussetzungen (i) bis (iv) um:

$$\begin{aligned} \min_{\phi} \sum_{\langle i, j \rangle \in E} c_{ij} \phi_{ij} \quad & \text{unter der Berücksichtigung von} \\ M(i) := \sum_{j \in S(i)} \phi_{ij} - \sum_{k \in P(i)} \phi_{ki} = \begin{cases} \omega^* & i = r \\ -\omega^* & i = s \\ 0 & i \in V \setminus \{r, s\} \end{cases} & \quad (P_{\omega^*}) \\ 0 \leq \phi_{ij} \leq \kappa_{ij} \quad & \text{für } \langle i, j \rangle \in E \end{aligned}$$

und suchen den entsprechenden ω^* -optimalen Fluß ϕ . Offensichtlich ist der Nullfluß 0-optimal. Wir wählen daher den Nullfluß als Startelement einer Iteration, die eine Folge von optimalen Flüssen wachsender Stärke konstruiert bis wir die Stärke ω^* erreichen.

⁵Dies ist insbesondere der Fall wenn auch (ii) nicht erfüllt war.

Wir beschreiben einen Iterationsschritt - also die Konstruktion eines $(\omega + \varepsilon)$ -optimalen Flusses ($\varepsilon > 0$) ausgehend von einem ω -optimalen Fluß ϕ von r nach s auf dem Netzwerk $N = (V, E; \kappa; c)$. Dazu verfolgen wir die selbe Strategie wie im Abschnitt 1.1 beginnend auf Seite 1. Wir betrachten einen flußvergrößernden (r, s) -Semiweg S , der die Stärke von ϕ gemäß (1.4) um ε vergrößert. Die Konstruktionsvorschrift (1.5) sagt aus, daß die transportierte Menge auf allen Vorwärtskanten S^+ von S um ε erhöht und auf allen Rückwärtskanten S^- von S um ε verringert wird. Wenn wir uns die Kosten des vergrößerten Flusses ϕ' betrachten, so folgt

$$\begin{aligned} \sum_{\langle i, j \rangle \in E} c_{ij} \phi'_{ij} &= \sum_{\langle i, j \rangle \in E \setminus (S^+ \cup S^-)} c_{ij} \phi_{ij} + \sum_{\langle i, j \rangle \in S^+} c_{ij} (\phi_{ij} + \varepsilon) + \sum_{\langle i, j \rangle \in S^-} c_{ij} (\phi_{ij} - \varepsilon) = \\ &= \sum_{\langle i, j \rangle \in E} c_{ij} \phi_{ij} + \varepsilon \left(\sum_{\langle i, j \rangle \in S^+} c_{ij} - \sum_{\langle i, j \rangle \in S^-} c_{ij} \right) \end{aligned}$$

Dies gibt Anlaß zur folgenden Definition.

Definition 11 *Es sei S ein flußvergrößernder (r, s) -Semiweg mit den Vorwärtskanten S^+ und den Rückwärtskanten S^- im Netzwerk $N = (V, E; \kappa; c)$. Dann verstehen wir unter den Kosten von S die Beziehung*

$$c(S) := \sum_{\langle i, j \rangle \in S^+} c_{ij} - \sum_{\langle i, j \rangle \in S^-} c_{ij} \quad (1.25)$$

$c(S)$ beschreibt damit die durch den (r, s) -Semiweges S verursachten Nettokosten für die Erhöhung der Stärke von ϕ um den Wert 1 bzw. $\varepsilon c(S)$ die Nettokosten für die Erhöhung um den Wert ε . Ein (r, s) -Semiweg mit minimalen Kosten wird folglich als *kostenoptimaler (r, s) -Semiweg* bezeichnet.

Zur Konstruktion eines (r, s) -Semiwegs verwenden wir die bereits in Abschnitt 1.2.5 auf Seite 13 beschriebene Inkrementmethode. Wir definieren wiederum ein bezgl. dem Fluß ϕ definiertes Inkrementnetzwerk $N(\phi) = (V, E(\phi); \kappa(\phi); c(\phi))$ gemäß

$$\begin{aligned} E(\phi) &:= E^+(\phi) \cup E^-(\phi) \\ E^+(\phi) &:= \{ \langle i, j \rangle \mid \langle i, j \rangle \in E, \phi_{ij} < \kappa_{ij} \} \\ E^-(\phi) &:= \{ \langle j, i \rangle \mid \langle i, j \rangle \in E, \phi_{ij} > 0 \} \end{aligned}$$

und

$$\kappa_{ij}(\phi) := \begin{cases} \kappa_{ij} - \phi_{ij} & \text{für } \langle i, j \rangle \in E^+(\phi) \\ \phi_{ji} & \text{für } \langle i, j \rangle \in E^-(\phi) \end{cases}$$

sowie

$$c_{ij}(\phi) := \begin{cases} c_{ij} & \text{für } \langle i, j \rangle \in E^+(\phi) \\ -c_{ji} & \text{für } \langle i, j \rangle \in E^-(\phi) \end{cases}$$

Die Definition der Kostenmatrix im Inkrementnetzwerk macht Sinn: eine Kante $\langle i, j \rangle \in E^-(\phi)$ repräsentiert die potentielle Rückwärtskante $\langle j, i \rangle$ eines flußvergrößernden (r, s) -Semiwegs. Eine Verringerung der transportierten Menge entlang dieser Kante um 1 führt zu einer Verringerung der Kosten um den Faktor $-c_{ji}$.

In Analogie zur Aussage 4 auf Seite 14 können wir aus einem Weg W von r nach s in $N(\phi)$ einen (r, s) -Semiweg S in N konstruieren. Wenn wir zusätzlich fordern, daß W kostenoptimal bzgl. $c(\phi)$ ist, so ist auch S ein kostenoptimaler (r, s) -Semiweg. Entsprechend der Aussage 3 auf Seite 13 ist ein Fluß ϕ maximal auf N wenn $s \notin \dot{R}(r)$ in $N(\phi)$ gilt. Die Bestimmung von W kann nach einem Baumalgorithmus (z.B. Label Setting nach Dijkstra) erfolgen - setzte jedoch voraus, daß $N(\phi)$ keine Zyklen negativer Länge besitzt. Dazu gilt:

Aussage 11 ϕ sei zulässiger Fluß der Stärke ω von r nach s im Netzwerk N . ϕ ist genau dann ω -optimal, wenn das zugehörige Inkrementnetzwerk $N(\phi)$ keine Zyklen negativer Länge besitzt.

Nachweis: siehe Neumann, Graphen und Netzwerke, in Gal, T. (ed.), Grundlagen des Operations Research, Teil 2, Springer (1987). ■

Ausgehend vom kostenoptimalen Weg W konstruieren wir in Analogie zum Abschnitt in Abschnitt 1.2.5 auf Seite 13:

$$\varepsilon := \min_{\langle i, j \rangle \in W} \kappa_{ij}(\phi)$$

$$\Psi_{ij} := \begin{cases} \varepsilon & \text{für } \langle i, j \rangle \in W \\ 0 & \text{sonst} \end{cases} \quad \text{auf allen Kanten } \langle i, j \rangle \in E(\phi)$$

$$\phi'_{ij} := \phi_{ij} + \begin{cases} \Psi_{ij} & \text{für } \langle i, j \rangle \in E^+(\phi) \\ -\Psi_{ji} & \text{für } \langle j, i \rangle \in E^-(\phi) \end{cases} \quad \text{auf allen Kanten } \langle i, j \rangle \in E$$

wobei $\omega(\phi') = \omega(\phi) + \varepsilon$. Falls aus dem Iterationsschritt folgt, daß

$$\omega(\phi) < \omega^* \quad \text{und} \quad \omega(\phi) + \varepsilon > \omega^*$$

so korrigieren wir

$$\varepsilon := \omega^* - \omega(\phi)$$

Abschließend formulieren wir den folgenden Pseudocode zur Lösung des Problems (KFS):

$$[\phi, \omega] = \text{KFS_BusGow}[N = (V, E; \lambda, \kappa; c), r, s, \omega^*]$$

% Transformation

```
> teste N auf Antisymmetrie (→ ASym.m)
> teste N auf nicht negative Kosten (→ NonNegC.m)
> teste N auf verschwindende Minimalkapazitäten (→ N4KFS2Ndd.m)
> teste N auf echt positive Maximalkapazitäten (→ KapG0.m)
> setze  $\phi_{ij} := 0$  für alle  $\langle i, j \rangle \in E$ ,  $\omega := 0$ ,  $b = false$ 
> wenn  $s \notin \dot{R}(r)$  in  $N$ , dann setze  $b = true$ 
```

% Iteration

```
> solange  $b = false$ , führe aus
> bestimme das Inkrementnetzwerk  $N(\phi)$  (→ NwrtPhi.m)
> bestimme den kostengünstigsten Weg  $W$  von  $r$  nach  $s$  in  $N(\phi)$  (→ sdp_dijkstra.m)
> wenn  $s$  in  $N(\phi)$  von  $r$  aus nicht erreichbar ist, dann
```

- > beende die Iteration
- > Hinweis: Problem (KFS) ist für ω^* nicht lösbar
- > ansonsten,
- > bestimme ε gemäß der oben stehenden Beziehung
- > wenn $\omega + \varepsilon = \omega^*$, dann setze $b = true$
- > oder wenn $\omega + \varepsilon > \omega^*$, dann führe aus
- > $\varepsilon := \omega^* - \omega$, $b = true$
- > bestimme Ψ und ϕ' gemäß den oben stehenden Beziehungen
- > $\phi \leftarrow \phi'$, $\omega \leftarrow \omega + \varepsilon$

Zur Lösung des Problems (MKF) führen wir gemäß (1.24) aus:

- > $[\phi, \omega] = \text{KFS_BusGow}[N = (V, E; \lambda, \kappa; c), r, s, \omega_{\max}]$

Wir wollen abschließend noch die Komplexität von `KFS_BusGow` diskutieren. Die Komplexität zur Bestimmung von $N(\phi)$ und Ψ ist von der Ordnung $O(|E|)$ und die Bestimmung von W nach Dijkstra von der Ordnung $O(|E| \log(|V|))$. Unter der Voraussetzung der Ganzzahligkeit der Maximalkapazitäten, wird die Flußstärke pro Iteration um mindestens Eins erhöht - es sind damit maximal ω^* Iterationen notwendig. Daraus folgt eine Gesamtkomplexität von $O(|E| \log(|V|) \omega^*)$

Beispiel 1.3(2): (\rightarrow `sec_cost_opt_flow_2.m`, `KFS_BusGow.m`) Wir betrachten das Netzwerk N mit Kapazitäten und Kosten gemäß Abbildung 1.28 - offensichtlich ist $N^{(0)}$ nicht antisymmetrisch. Wir testen $N^{(0)}$ auf Antisymmetrie und erhalten durch die obige Trans-

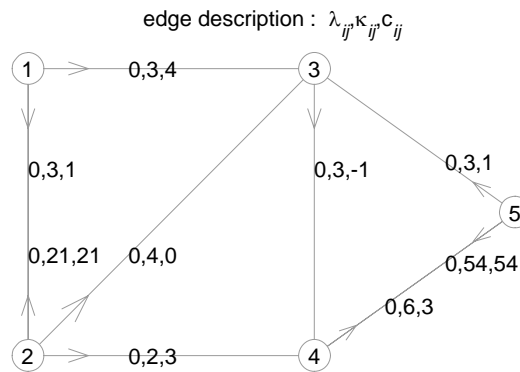


Abbildung 1.28: Netzwerk $N^{(0)}$ mit Kapazitäten und Kosten.

formation das Netzwerk $N^{(1)}$ gemäß der Abbildung 1.29. Das Netzwerk $N^{(1)}$ beinhaltet negative Kostenkoeffizienten, sodaß wir entsprechend der obigen Transformationsvorschrift das Netzwerk $N^{(2)}$ konstruieren; dieses ist in Abbildung 1.30 dargestellt. Da im Netzwerk $N^{(2)}$ nicht alle Minimalkapazitäten gleich Null sind, transformieren wir $N^{(2)}$ wiederum auf das Netzwerk $N^{(3)}$ gemäß der Abbildung 1.31. Die Lösung des Problems (KFS) in $N^{(2)}$ für $r = 1$, $s = 5$ und $\omega^* = 5$ ist gemäß der Aussage 10 äquivalent zur Lösung des Problems (KFS) ist in $N^{(3)}$ für $r = 8$, $s = 9$ und $\omega^* = 8$. Wir rechnen drei Iterationen mit den Zwischenergebnissen:

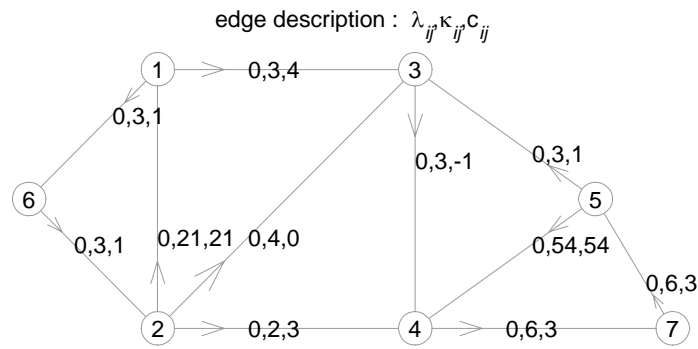


Abbildung 1.29: Transformiertes Netzwerk $N^{(1)}$. Die Knoten 6 und 7 sind Hilfsknoten zur Auflösung der antiparallelen Kanten.

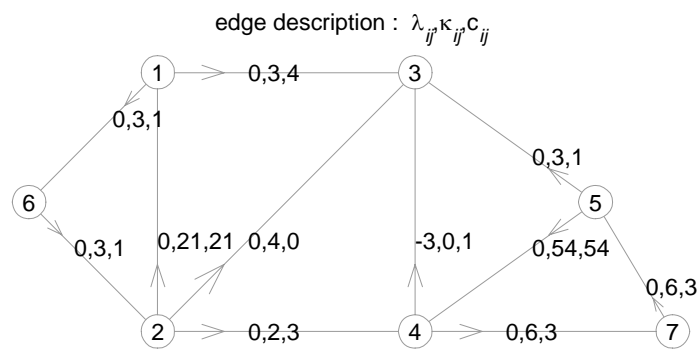


Abbildung 1.30: Netzwerk $N^{(2)}$ mit positiven Kostenkoeffizienten c_{ij} .

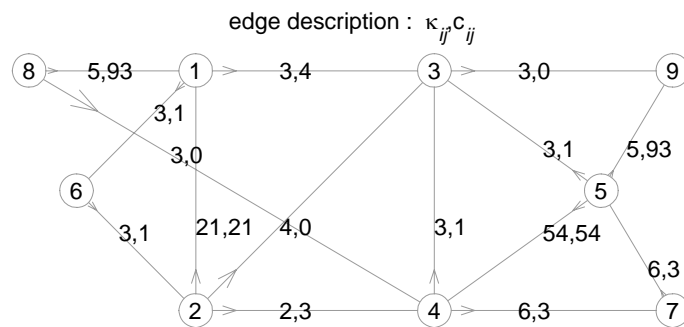


Abbildung 1.31: Netzwerk $N^{(3)}$ mit verschwindenden Minimalkapazitäten.

```

> INITIALIZATION
> iteration 0 - w(Phi) = 0, c(Phi) = 0
> ITERATION
> iteration 1 - flow augmented: w(Phi) = 3, c(phi) = 3
> iteration 2 - flow augmented: w(Phi) = 6, c(phi) = 582
> PROBLEM SOLVED
> iteration 3 - flow augmented: w(Phi) = 8, c(phi) = 980

```

Der kostenoptimale Fluß $\phi^{(3)}$ der Stärke $\omega = 8$ ist in Abbildung 1.32 dargestellt. Die Abbil-

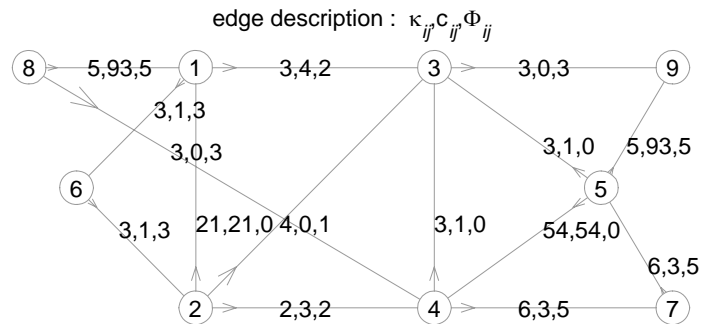


Abbildung 1.32: Lösung des Problems (KFS) im Netzwerk $N^{(3)}$.

dungen 1.33 bis 1.35 zeigen die Rücktransformation des Flusses $\phi^{(3)}$ auf das ursprüngliche Netzwerk $N^{(0)}$.

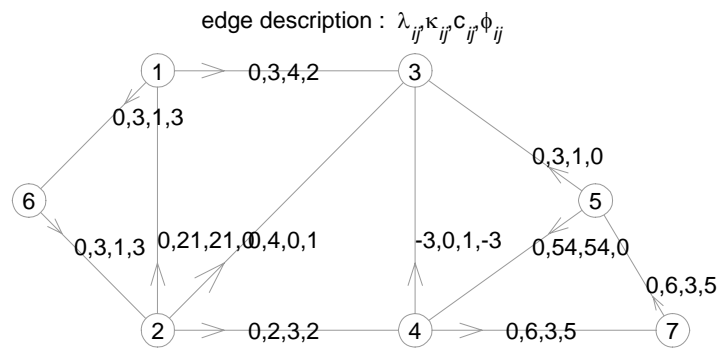
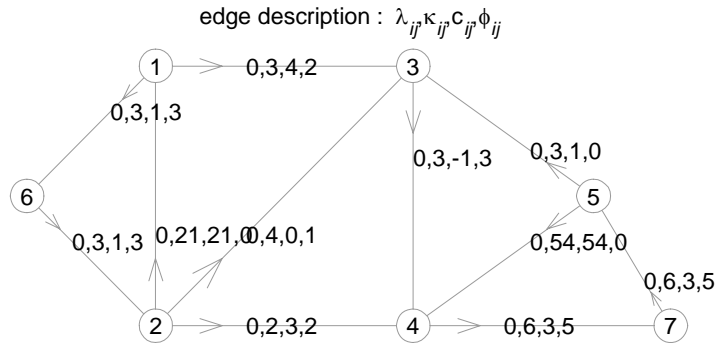
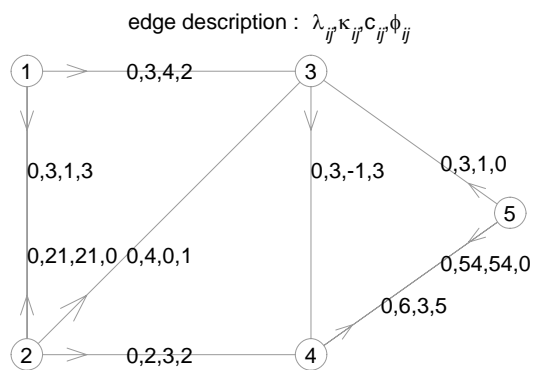


Abbildung 1.33: Rücktransformation von $\phi^{(3)}$ auf $\phi^{(2)}$ im Netzwerk $N^{(3)}$.

Abbildung 1.34: Rücktransformation von $\phi^{(2)}$ auf $\phi^{(1)}$ im Netzwerk $N^{(1)}$.Abbildung 1.35: Rücktransformation von $\phi^{(1)}$ auf $\phi^{(0)}$ im Netzwerk $N^{(0)}$.

Liste der betrachteten Probleme und ihre Abkürzungen

Abkürzung	Bestimmung ...
FP0	... eines Initialflusses (Seite 25)
KFS	... eines kostenoptimalen Flusses vorgegebener Stärke (Seite 33)
MFP	... eines maximalen Flusses (Seite 3)
MKF	... eines kostengünstigsten maximalen Flusses (Seite 35)