Vorlesung
Kombinatorische Optimierung
(Wintersemester 2016/17)

Kapitel 2: Flüsse und Zirkulationen

Volker Kaibel

Otto-von-Guericke Universität Magdeburg

(Version vom 24. Oktober 2016)

Definition 2.1

Ein **Netzwerk** ist ein Paar (D, u) bestehend aus einem Digraphen $D = (V, A)$ und nicht-negativen **Bogenkapazitäten** $u \in \mathbb{R}_+^A$. Ein **Fluss** in (D, u) ist eine Abbildung $f : A \to \mathbb{R}_+$ ($f \in \mathbb{R}_+^A$) mit $f(a) = f_a \leq u_a$ für alle $a \in A$. Für $v \in V$ definieren wir den **Überschuss**

$$\mathrm{ex}_f(v) := f(\delta^{\mathrm{ein}}(v)) - f(\delta^{\mathrm{aus}}(v))$$

von f in v. Ein Fluss $f \in \mathbb{R}_+^A$ mit

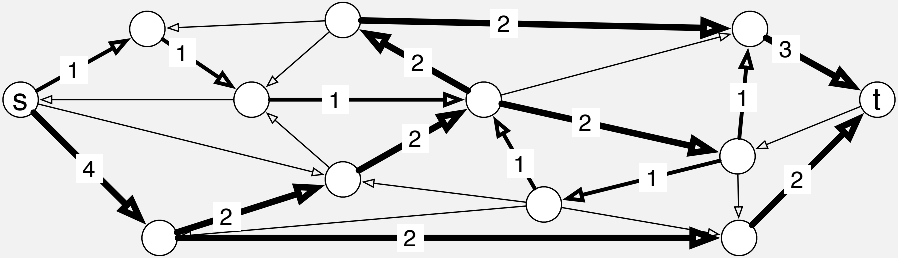
$$\mathrm{ex}_f(v) = 0 \quad \text{für alle } v \in V$$

(**Flusserhaltungsbedingungen**) heißt eine **Zirkulation**.
Für $s, t \in V$ ist ein s-t-**Fluss** ein Fluss $f \in \mathbb{R}_+^A$ in (D, u) mit

$$\mathrm{ex}_f(v) = 0 \quad \text{für alle } v \in V \setminus \{s, t\}$$

und $\mathrm{ex}_f(s) \leq 0$.

s-t-Flüsse



Das Max-Flow Problem

Problem 2.3 (Max-Flow Problem)

Instanz: *Netzwerk* $N = (D = (V, A), u)$*, zwei Knoten* $s, t \in V$

Aufgabe: *Ein s-t-Fluss in* N *mit maximalem Flusswert*

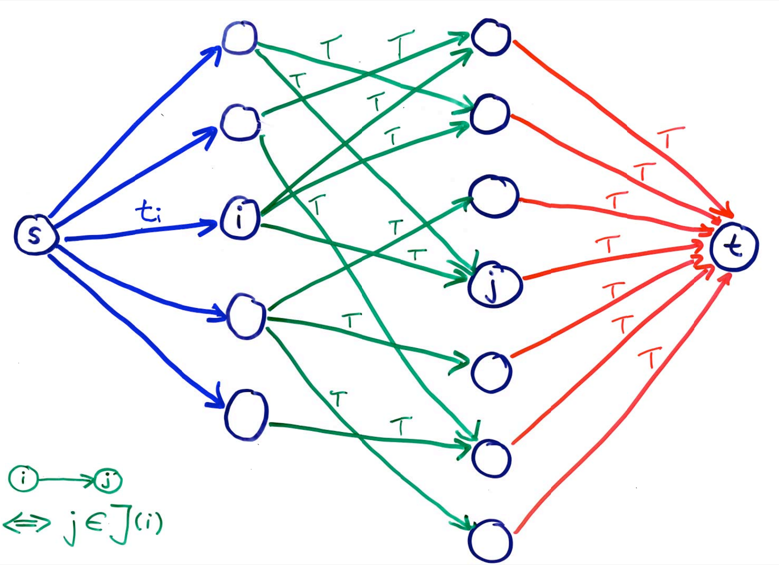
Anwendung:

Problem 2.4 (Job-Assignment Problem)

Instanz: *n Jobs, m Arbeiter, Arbeitszeit* $t_i \in \mathbb{Q}_+$*, die Job* i *benötigt und Teilmenge* $J(i) \subseteq [m]$ *von Arbeitern, die Job* i *können (für alle* i*),* $T \in \mathbb{Q}_+$

Aufgabe: *Finde, wenn möglich, einen Plan, mit dem alle Jobs nach* T *Zeiteinheiten fertig sind*

Job-Assignment als Flussproblem



Inzidenzmatrizen

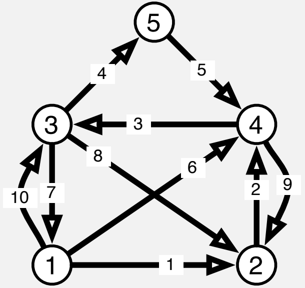
Definition 2.5

Die **Inzidenzmatrix** eines Digraphen $D = (V, A)$ ist die Matrix $\mathrm{Inz}(D) \in \{-1, 0, 1\}^{V \times A}$ mit

$$\mathrm{Inz}(D)_{v,a} = \begin{cases} -1 & , \quad \text{falls } a \in \delta^{\text{aus}}(v) \\ +1 & , \quad \text{falls } a \in \delta^{\text{ein}}(v) \\ 0 & , \quad \text{sonst} \end{cases}$$

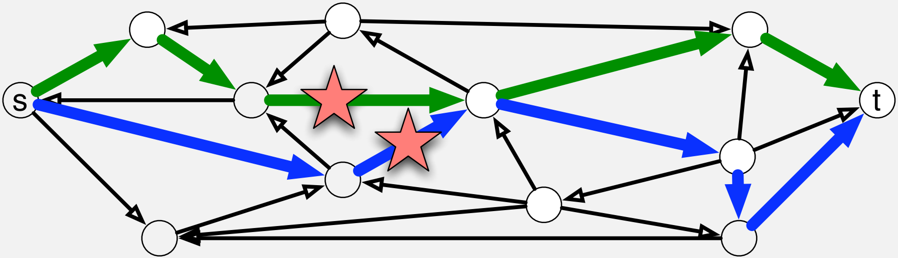
für alle $v \in V$ und $a \in A$.

Inzidenzmatrizen



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

Satz von Menger (bogendisjunkt, gerichtet)



Spezielle Untergraphen

Definition 2.20

Für einen Graphen $G = (V, E)$ und $F \subseteq E$, $W \subseteq V$ definieren wir die folgenden Graphen:

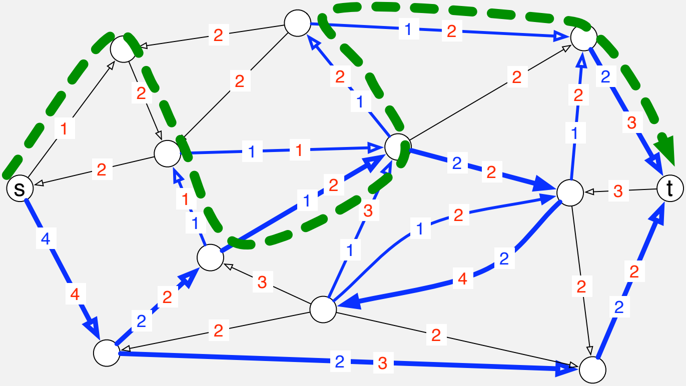
- $G[F] := (V, F)$
- $G \setminus F := G[E \setminus F]$
- $G[W] := (W, E \cap \binom{W}{2})$
- $G \setminus W := G[V \setminus W]$

Zusammenhang von Graphen

Definition 2.23

Für $k \in \mathbb{N}_{>0}$ heißt ein Graph $G = (V, E)$ k-**fach kantenzusammenhängend**, wenn $|V| \geq 2$ ist und für alle $F \subseteq E$ mit $|F| < k$ der Graph $G \setminus F$ zusammenhängend ist. Der Graph G ist k-**fach knotenzusammenhängend** (oder: k-**zusammenhängend**), wenn $|V| > k$ ist und für alle $W \subseteq V$ mit $|W| < k$ der Graph $G \setminus W$ zusammenhängend ist.

Idee: Augmentierung



Multi-Digraphen...

Definition 2.26

Ein **Multi-Digraph** ist ein Tripel $D = (V, A, \Psi)$ mit einer endlichen Knotenmenge V und einer endlichen Bogenmenge A und einer Abbildung

$$\Psi : A \to V \times V \setminus \{(v, v) \,|\, v \in V\}.$$

Zwei Bögen $a, a' \in A$ mit $a \neq a'$ und $\Psi(a) = \Psi(a')$ heißen **parallel**. Sie heißen **anti-parallel**, wenn $\Psi(a) = (v, w)$ und $\Psi(a') = (w, v)$ ist.

. . . Multi-Digraphen

Definition 2.27

Ein Multi-Digraph, der keine parallelen Bögen hat, heißt **einfach**. Ein einfacher Multi-Digraph (V, A, Ψ) ist ein Digraph, wenn man jeden Bogen $a \in A$ (eineindeutig) mit $\Psi(a) \in V \times V$ identifiziert.

Definition 2.28

Ein Multi-Digraph $D' = (V', A', \Psi')$ ist ein **Unter-Multi-Digraph** von $D = (V, A, \Psi)$, wenn $V' \subseteq V$, $A' \subseteq A \cap (V' \times V')$ und $\Psi' = \Psi|_{A'}$ sind.

Weitere Definitionen

Seien $D = (V, A, \Psi)$ ein Multi-Digraph mit Bogenlängen $c \in \mathbb{R}^A$.

- Sind $v_0, v_1, \ldots, v_l \in V$ und $a_1, \ldots, a_l \in A$ mit $\Psi(a_i) = (v_{i-1}, v_i)$ für alle $i \in [l]$, so heißt $Q = \{a_1, \ldots, a_k\}$ ein v_0-v_l-**Weg** in D, falls $v_i \neq v_j$ für alle $i, j \in \{0, 1, \ldots, l\}$ mit $i \neq j$. Falls $l \geq 2, v_0 = v_l$ und $v_i \neq v_j$ für alle $i, j \in [l]$ mit $i \neq j$, so ist Q ein **Kreis** in D.
- Die **(kombinatorische) Länge** von Q ist $|Q| = l$. Die c-**Länge** von Q ist $c(Q)$.
- Für $s \in V$ bezeichnet $R_D(s) \subseteq V$ die Menge aller Knoten $w \in V$, für die ein s-w-Weg in D existiert.
- Haben alle Kreise in D nicht-negative c-Länge, so heißt c **konservativ**.
- Ein Vektor $\pi \in \mathbb{R}^V$ ist ein c-**Potenzial** für D, wenn für alle $a \in A$ mit $\Psi(a) = (v, w)$

  $$\pi_w \leq \pi_v + c_a$$

  gilt.

Der Multi-Digraph $\overleftrightarrow{D}$

Definition 2.30

Für einen Digraphen $D = (V, A)$ definieren wir den Multi-Digraphen $\overleftrightarrow{D} = (V, \overleftrightarrow{A}, \Psi)$ mit $\overleftrightarrow{A} = A \uplus \overleftarrow{A}$ und einer Bijektion $A \to \overleftarrow{A}$ mit $a \mapsto \overleftarrow{a}$ für alle $a \in A$, sowie für $a \in A$ mit $a = (v, w)$

$$\begin{aligned} \Psi(a) &= (v, w) \\ \Psi(\overleftarrow{a}) &= (w, v). \end{aligned}$$

Die Bögen in A heißen **Vorwärtsbögen**, die in $\overleftarrow{A}$ heißen **Rückwärtsbögen**. Bögen a und $\overleftarrow{a}$ heißen **revers** zueinander.

Das Residualnetzwerk

Definition 2.31

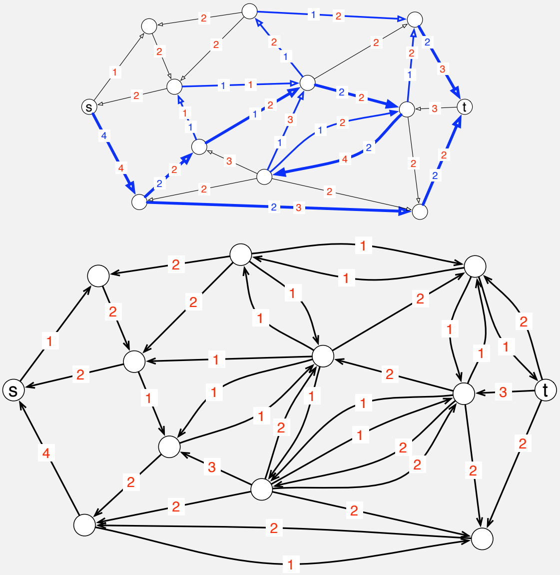
Ist $f \in \mathbb{R}_+^V$ ein Fluss in einem Netzwerk $(D = (V, A), u)$, so ist das **Residual-Netzwerk** bzgl. f der Multi-Digraph $D_f = (V, A_f, \Psi|_{A_f})$ wobei A_f die Teilmenge von $\overset{\leftrightarrow}{A}$ ist, für die für alle $a = (v, w) \in A$ gilt:

$$\begin{aligned} a \in A_f &\quad\Leftrightarrow\quad f_a < u_a \\ \overleftarrow{a} \in A_f &\quad\Leftrightarrow\quad f_a > 0 \end{aligned}$$

Die **Residualkapazitäten** $\bar{u} \in \mathbb{R}_{>0}^{A_f}$ sind definiert durch

$$\begin{cases} \bar{u}_a := u_a - f_a & \text{für alle } a \in A \text{ mit } a \in A_f \\ \bar{u}_{\overleftarrow{a}} := f_a & \text{für alle } a \in A \text{ mit } \overleftarrow{a} \in A_f \end{cases}$$

Beispiel: Residualnetzwerk



Augmentierende Wege

Ein f-**augmentierender Weg/Kreis** ist ein Weg/Kreis $R \subseteq A_f$ in D_f. Seine **Kapazität** ist

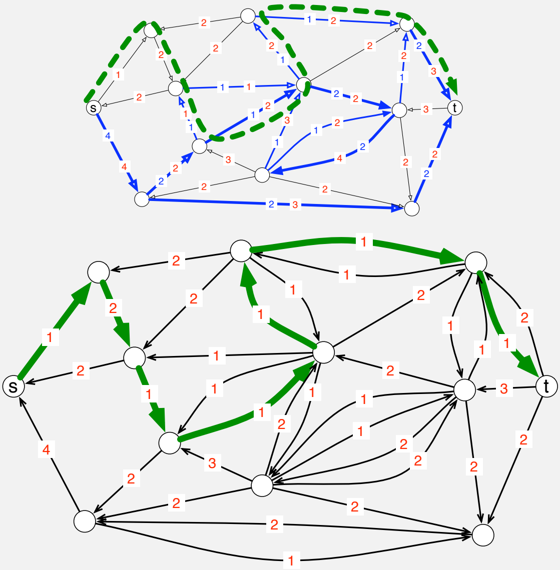
$$\text{cap}_f(R) := \min\{\bar{u}_r \mid r \in R\}\,.$$

Ist $R \subseteq A_f$ ein f-augmentierender Weg/Kreis und $\gamma \in \mathbb{R}$, so definieren wir $\text{aug}(f, R, \gamma)$ als $\tilde{f} \in \mathbb{R}_+^A$ durch

$$\tilde{f}_a := \begin{cases} f_a + \gamma & , \quad \text{falls} \quad a \in R \\ f_a - \gamma & , \quad \text{falls} \quad \overleftarrow{a} \in R \\ f_a & , \quad \text{sonst} \end{cases}$$

für alle $a \in A$ ("Augmentierung von f um γ entlang R").

Augmentierender Weg im Residualnetzwerk



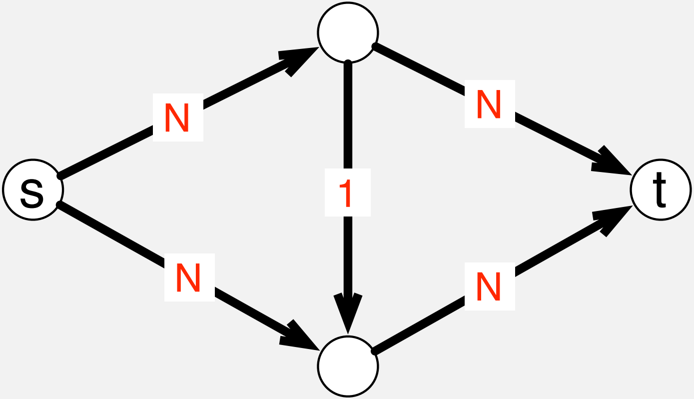
Algorithmus 2.35 (Ford-Fulkerson Algorithmus)

Eingabe: *Netzwerk* $N = (D = (V, A), u)$, $s, t \in V$ $(s \neq t)$

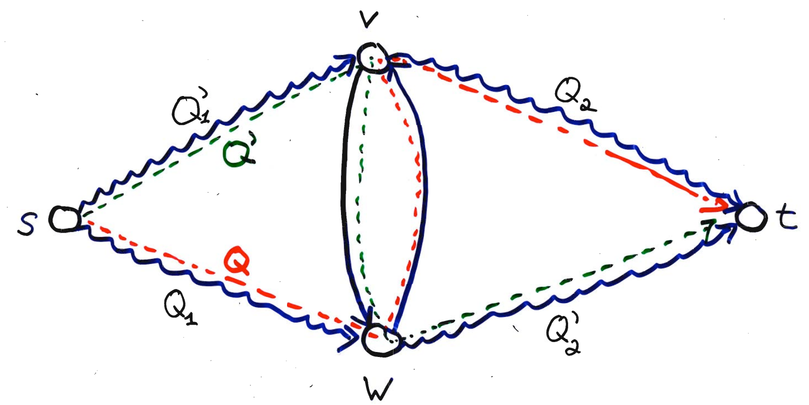
Ausgabe: *Ein s-t-Fluss* $f \in \mathbb{Q}_+^A$ *in* N *mit maximalem Flusswert*

```
1: for all a ∈ A do
2:     f_a ← 0
3: if t ∉ R_{D_f}(s) then
4:     Stop
5: Bestimme einen f-augmentierenden s-t-Weg R in D_f.
6: f ← aug(f, R, cap_f(R))
7: Gehe zu Schritt 3
```

Exponentielle Laufzeit des FF-Algorithmus



Beweis von Lemma 2.40



Der Harris-Ross Report

Ford & Fulkerson (1954):
(Beschreiben das Problem, das T. E. Harris formuliert hatte.)

> Consider a rail network connecting two cities by way of a number of intermediate cities, where each link of the network has a number assigned to it representing its capacity. Assuming a steady state condition, find a maximal flow from one given city to the other.

Ford & Fulkerson (1962):
(Über das Max-Flow Problem.)

> It was posed to the authors in the spring of 1955 by T.E. Harris, who, in conjunction with General F.S. Ross (Ret.), had formulated a simplified model of railway traffic flow, and pinpointed this particular problem as the central one suggested by the model [11].

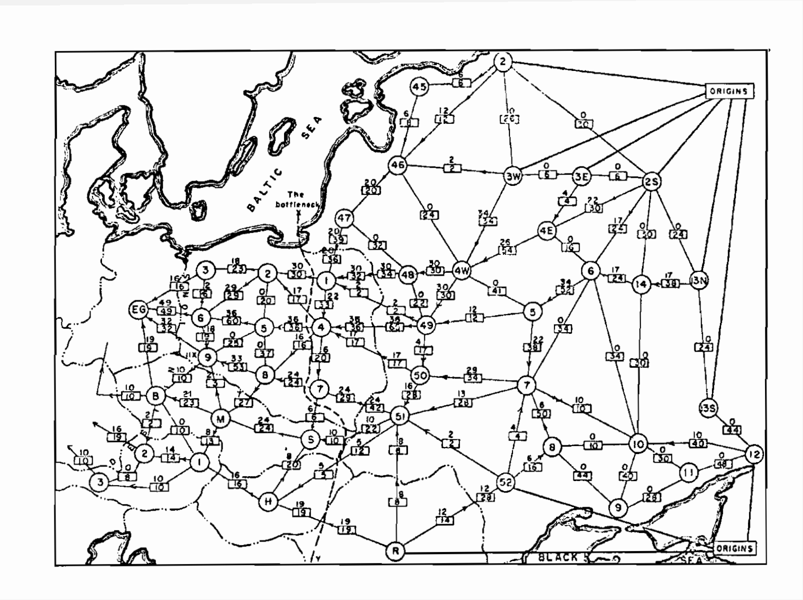
Schrijver (*Combinatorial Optimization*, S. 166)

(Zitiert aus dem Report von Harris und Ross aus dem Jahre 1955 an die Air Force, "downgraded to *unclassified*" im Jahr 1999)

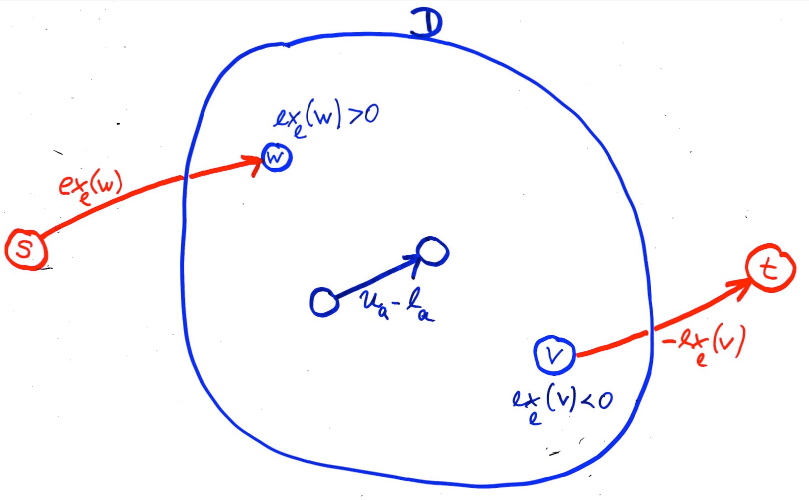
Unlike what Ford and Fulkerson write, the interest of Harris and Ross was not to find a maximum flow, but rather a minimum cut ('interdiction') of the Soviet railway system. We quote:

> Air power is an effective means of interdicting an enemy's rail system, and such usage is a logical and important mission for this Arm.
> As in many military operations, however, the success of interdiction depends largely on how complete, accurate, and timely is the commander's information, particularly concerning the effect of his interdiction-program efforts on the enemy's capability to move men and supplies. This information should be available at the time the results are being achieved.
> The present paper describes the fundamentals of a method intended to help the specialist who is engaged in estimating railway capabilities, so that he might more readily accomplish this purpose and thus assist the commander and his staff with greater efficiency than is possible at present.

Abbildung aus dem Harris-Ross Report



Finden von Zirkulationen



Min-Cost Zirkulationen

Problem 2.45 (Min-Cost Circulation Problem)

Instanz: $D = (V, A)$, $l, u \in \mathbb{Q}^A$ $(l \leq u)$, $c \in \mathbb{Q}^A$

Aufgabe: *Optimallösung von*

$$\min\{\langle c, f\rangle \mid f \in \mathrm{Circ}(D, l, u)\}$$

oder Feststellung, dass $\mathrm{Circ}(D, l, u) = \emptyset$

Cycle Cancelling Algorithmus

Algorithmus 2.49 (Cycle Cancelling)

Eingabe: $D = (V, A)$, *$l, u \in \mathbb{Q}^A$ ($l \leq u$), $c \in \mathbb{Q}^A$*

Ausgabe: *Optimallösung von*

$$\min\{\langle c, f\rangle \mid f \in \mathrm{Circ}(D, l, u)\}$$

oder Feststellung, dass $\mathrm{Circ}(D, l, u) = \emptyset$

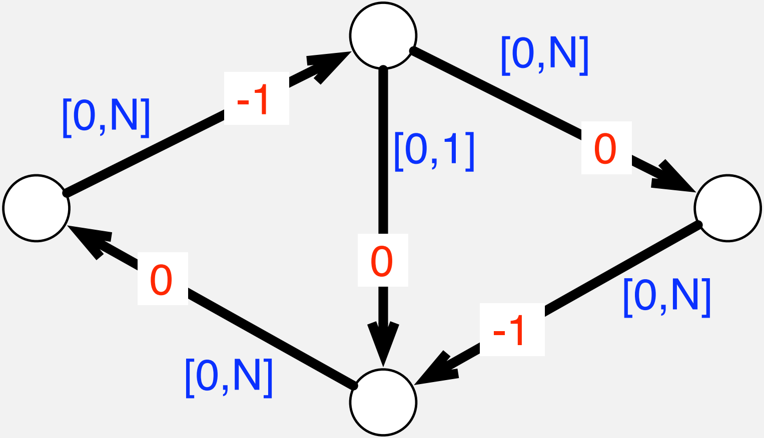
1: *Bestimme irgendein $f \in \mathrm{Circ}(D, l, u)$ (oder stelle fest, dass $\mathrm{Circ}(D, l, u) = \emptyset$ ist (Lem. 2.44 und stoppe).*

2: **while** *D_f hat $\overset{\leftrightarrow}{c}$-negativen Kreis* **do**

3: *Bestimme Kreis $C \subseteq A_f$ mit $\overset{\leftrightarrow}{c}(C) < 0$.*

4: $f \leftarrow \mathrm{aug}(f, C, \mathrm{cap}_f(C))$

Exponentiell viele Augmentationen



(Augmentatiere stets entlang von Dreiecken.)

b-Fluss als Zirkulation