Graphen und diskrete Optimierung

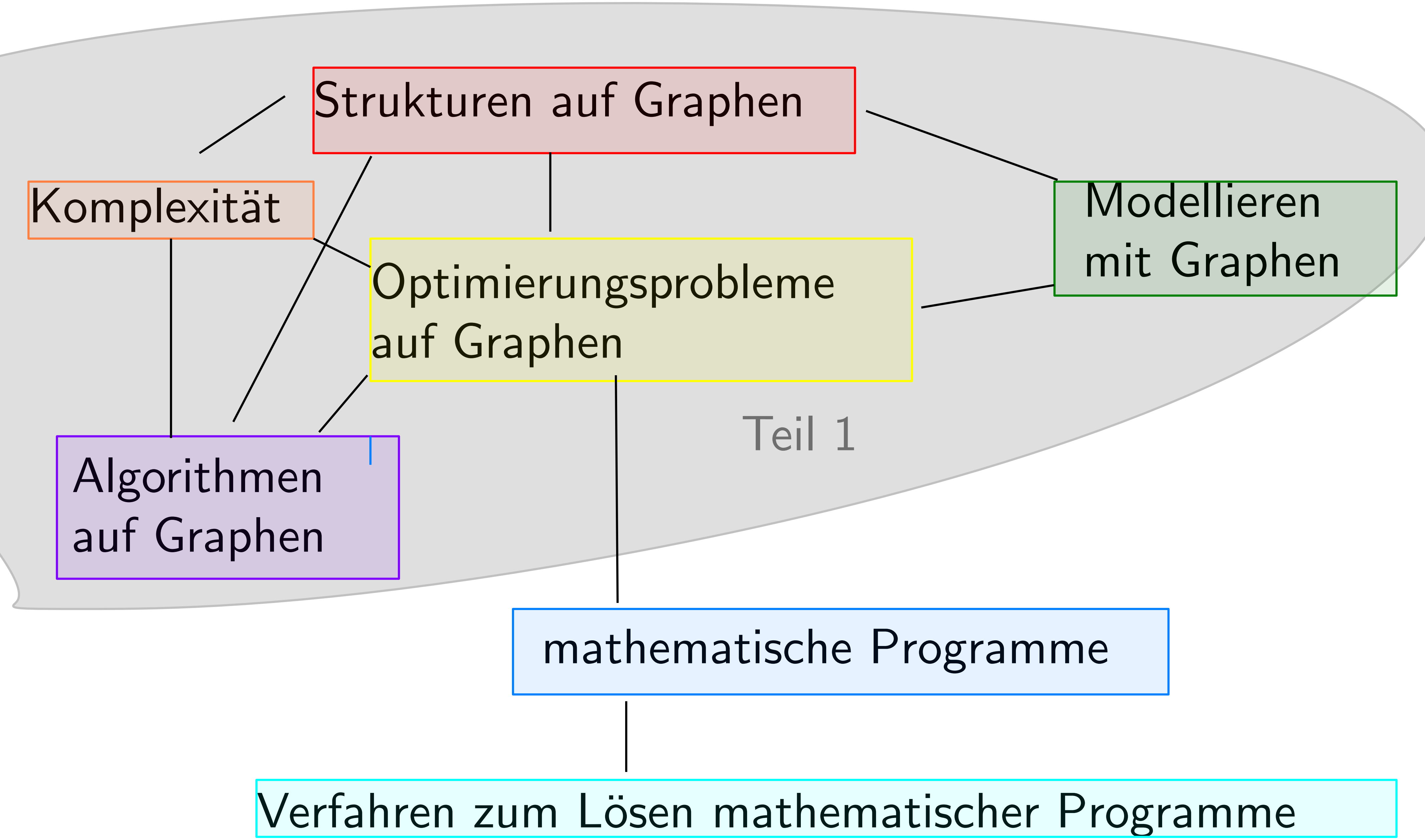
im Bachelorstudiengang 'Informatik und Nachhaltigkeit'

Flussprobleme

Marie Schmidt

03.05.2023

Worum soll er hier gehen?

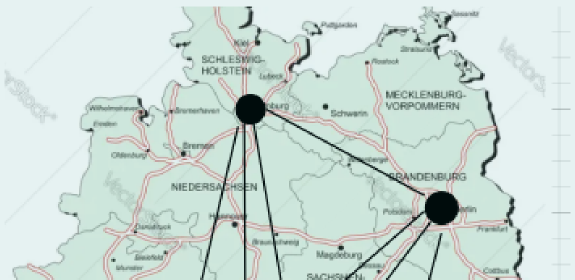


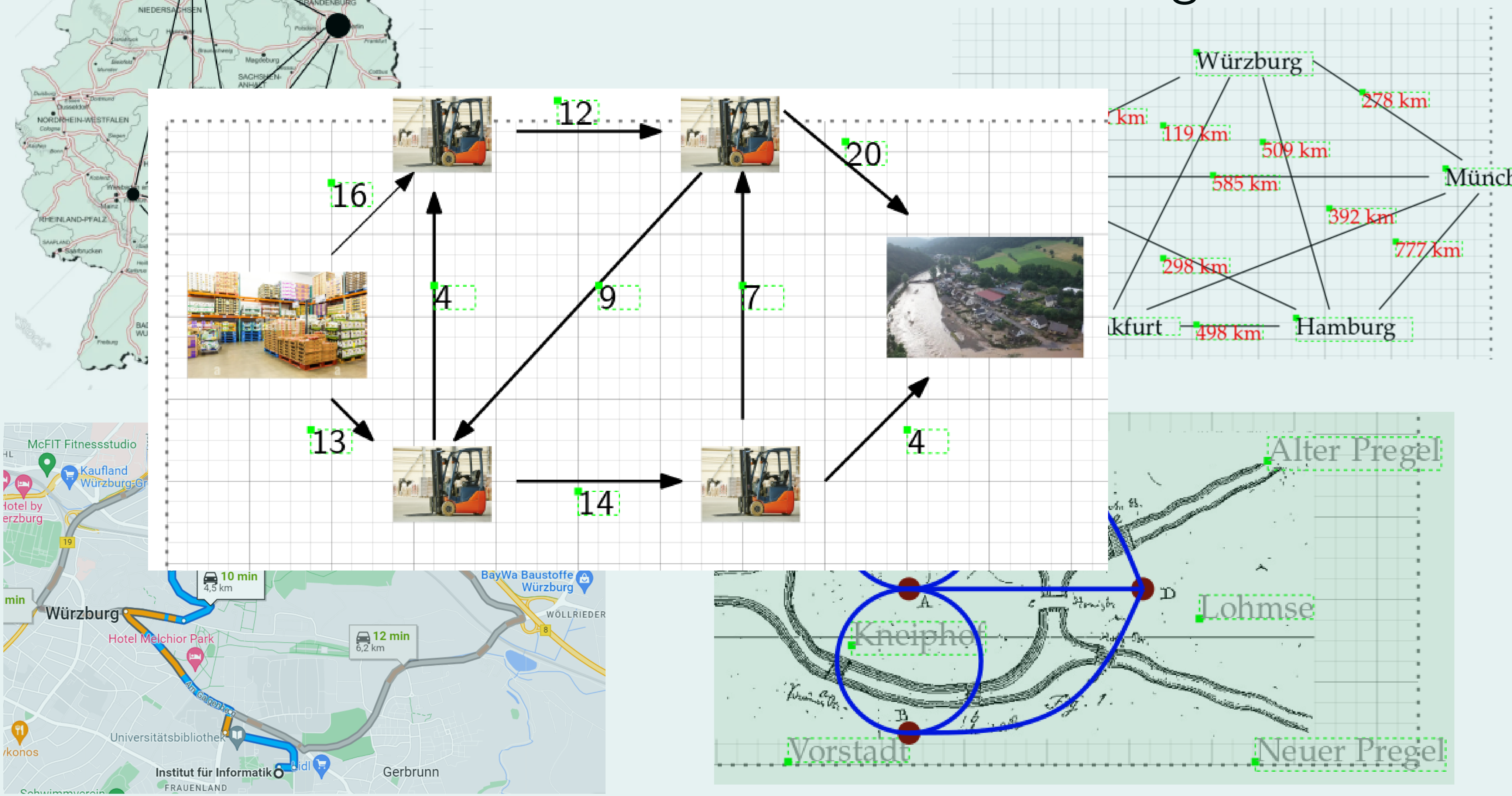
Vorlesungsübersicht

- Einführung: Modellieren auf Graphen ✓
- Teil 1: Graphen und Algorithmen auf Graphen
  - Graphen modellieren räumliche Zusammenhänge
    - Wege: Dijkstra ✓, Bellmann-Ford ✓, multikriterielle Wege ✓
    - Touren ✓
    - Flüsse: maximale Flüsse, (Strömungen, b-Flüsse, kostenminimale Flüsse)
  - Graphen modellieren Beziehungen
    - Matchings
    - Eckenüberdeckungen
    - Färbungsprobleme
    - ...
  - Algorithmen und (Problem-)Komplexität
- Teil 2: Lineare und ganzzahlige Optimierung auf Graphen
  - ...

Worum geht es heute: Flüsse

heute: Knoten stehen für Orte, Kanten für Verbindungen zwischen Orten, Kantenlabels für Entfernungen





Agenda

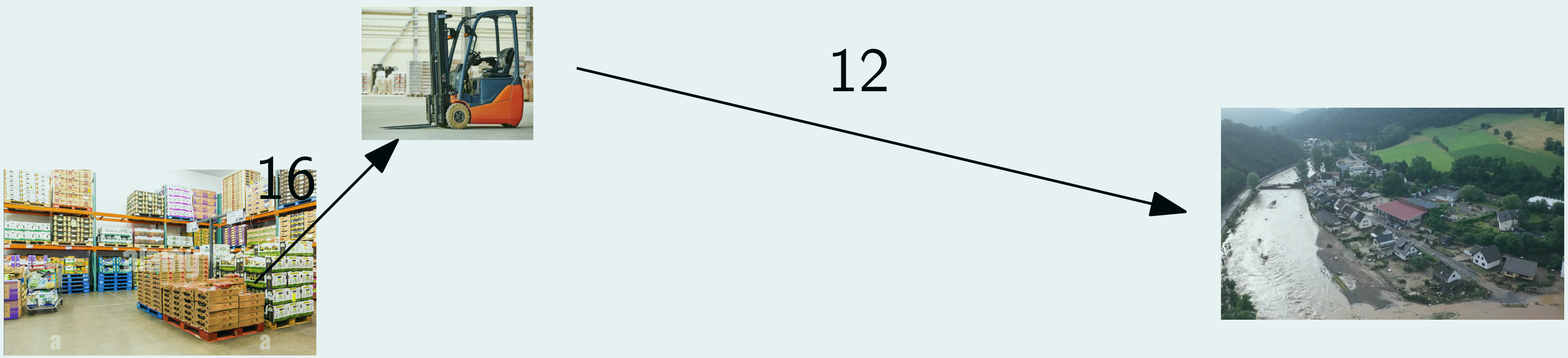
1. Flussprobleme
    - ausführlich und mit Beweisen: maximale Flüsse - und minimale Schnitte auf Netzwerken mit oberen Kantenkapazitätsschranken
    - kurz angerissen und ohne Beweise: Verallgemeinerungen: untere Kantenkapazitätsschranken, Bedarfe, Kosten

Hilfslieferungen im Katastrophenfall

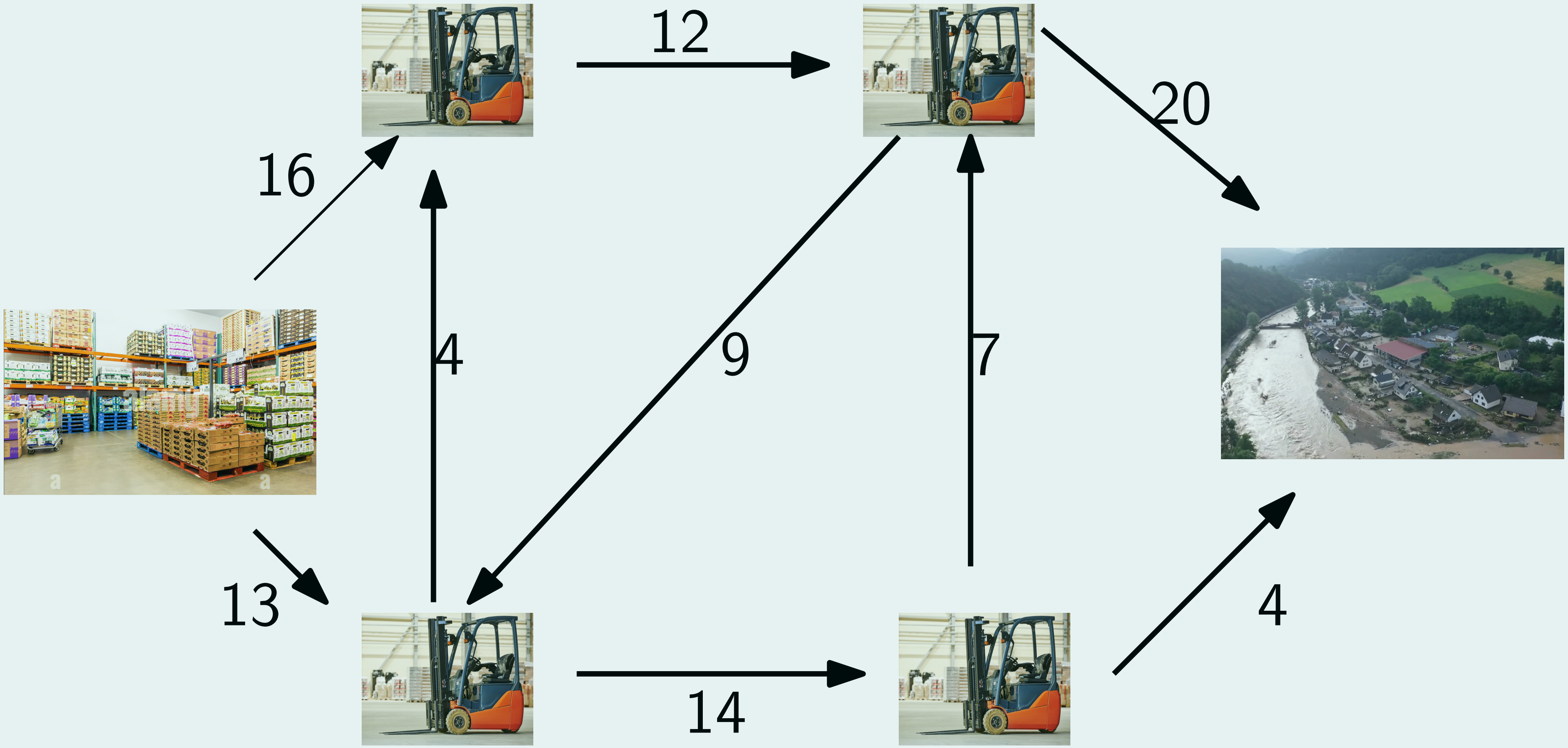
Wie bekommen wir möglichst viele Hilfslieferungen dorthin, wo sie benötigt werden?

Koordination von Hilfsgüterlieferungen als Flussproblem

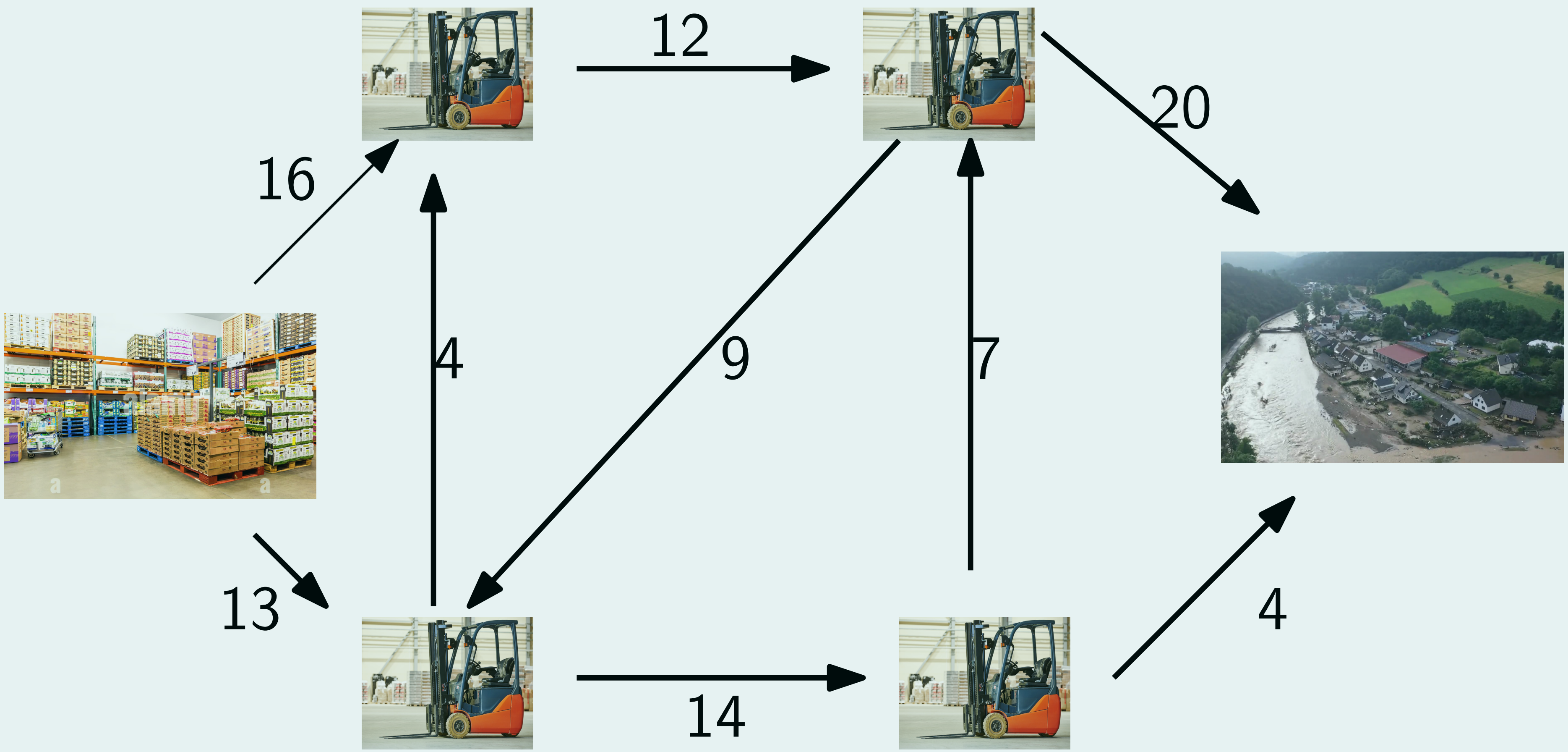
Koordination von Hilfsgüterlieferungen als Flussproblem



Koordination von Hilfsgüterlieferungen als Flussproblem

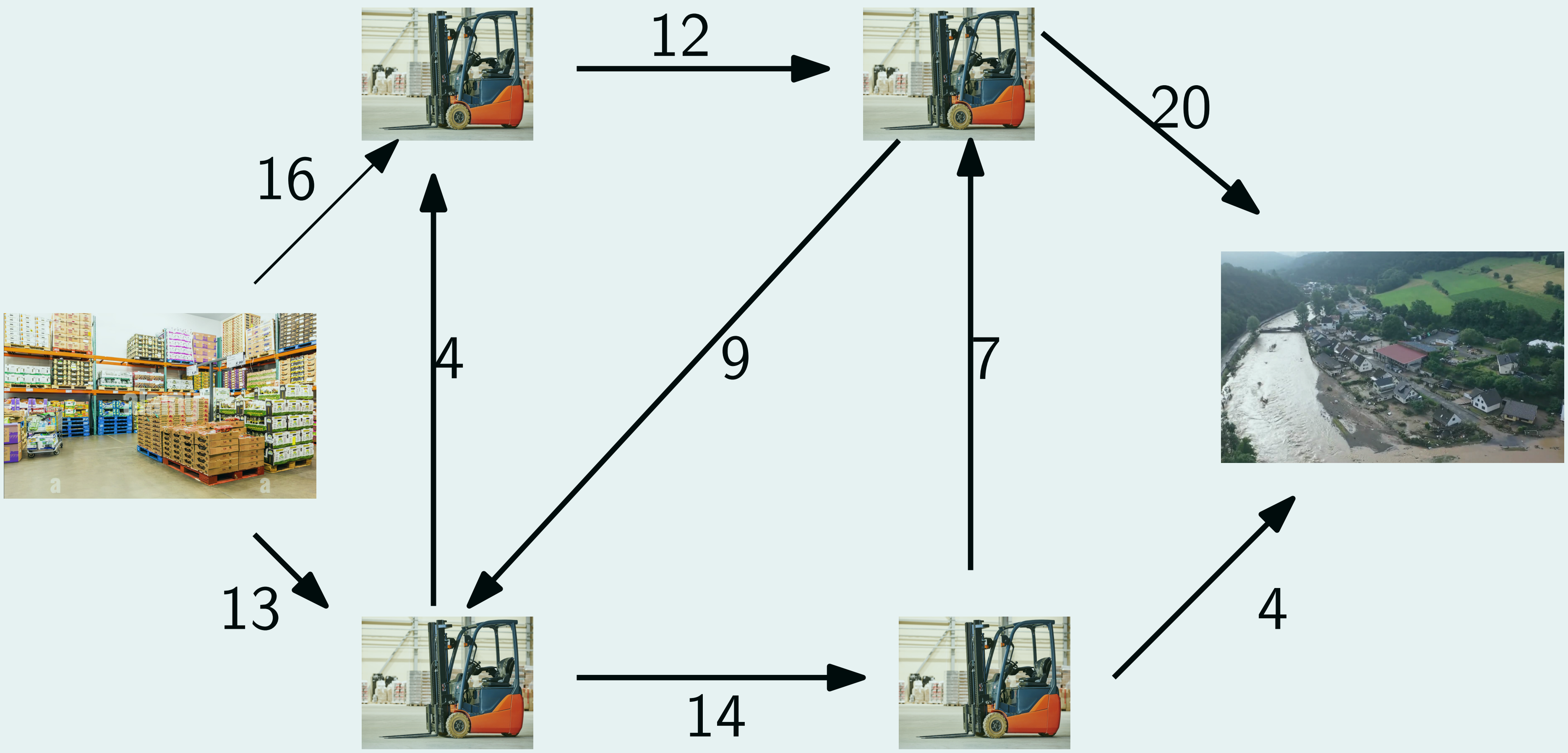


Koordination von Hilfsgüterlieferungen als Flussproblem



Knoten: Zielort mit Bedarf, Startort mit Vorräten, Transshipment-Punkte

Koordination von Hilfsgüterlieferungen als Flussproblem

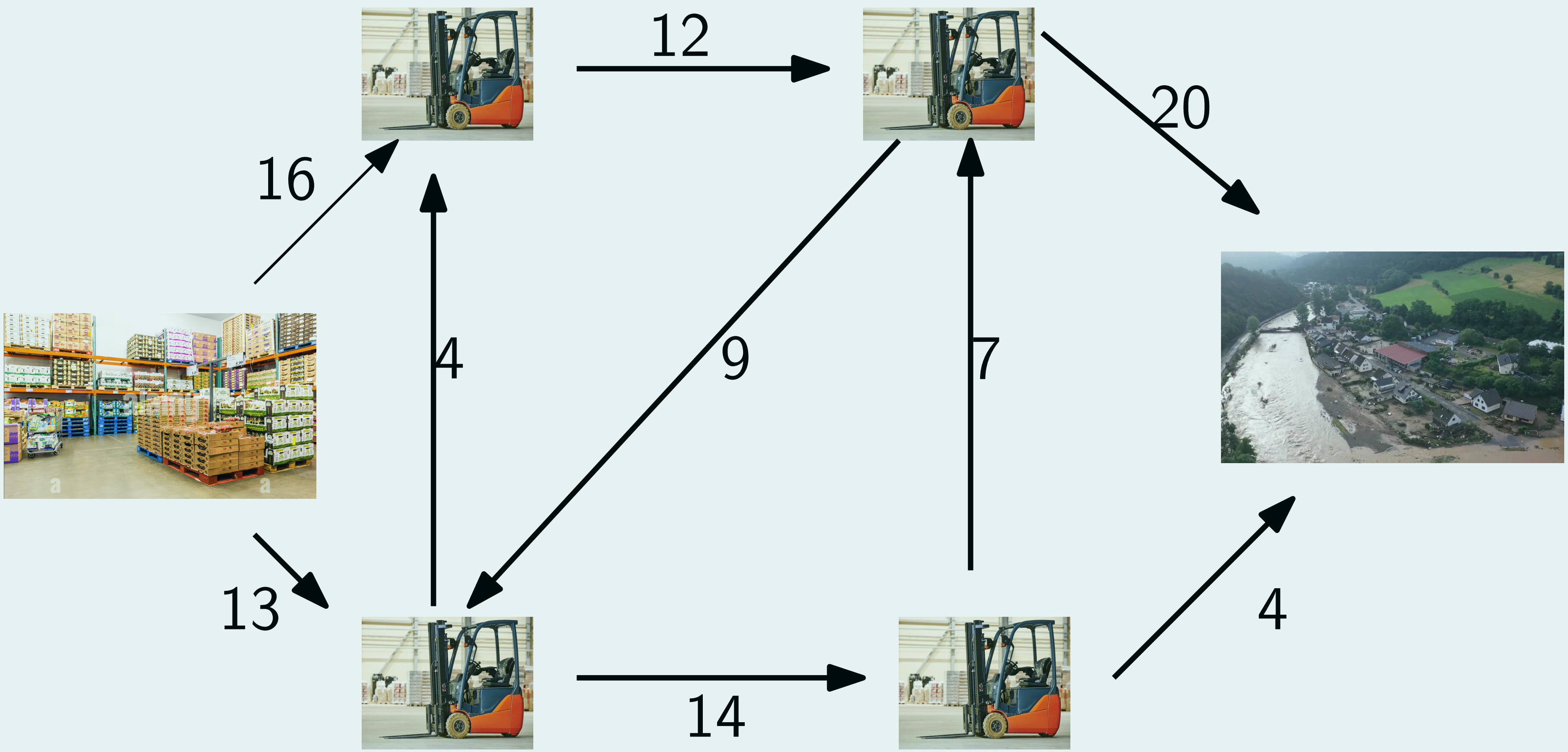


Knoten: Zielort mit Bedarf, Startort mit Vorräten, Transshipment-Punkte

Kanten: Transportwege, die am Stück von einem Fahrzeug zurückgelegt werden

Kantenlabel: Transportkapazitäten der Kanten

Koordination von Hilfsgüterlieferungen als Flussproblem



Knoten: Zielort mit Bedarf, Startort mit Vorräten, Transshipment-Punkte

Kanten: Transportwege, die am Stück von einem Fahrzeug zurückgelegt werden

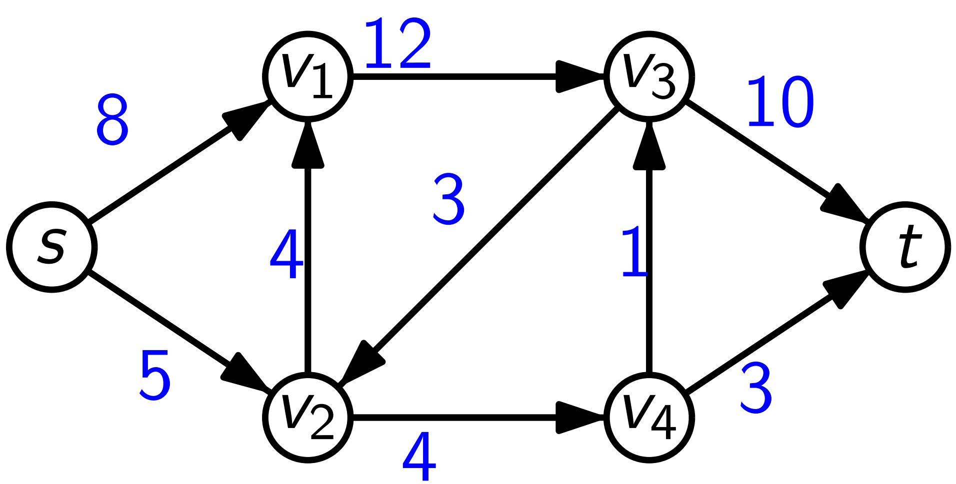
Kantenlabel: Transportkapazitäten der Kanten

Suche: maximalen Hilfsgüterfluss vom Lager zum Zielort

Flüsse

Notation

Sei $G = (V, E)$ ein gerichteter Graph.
Fluss: Funktion $f : E \to \mathbb{R}$
$f(e)$ gibt an, wieviel über Kante e fließt.



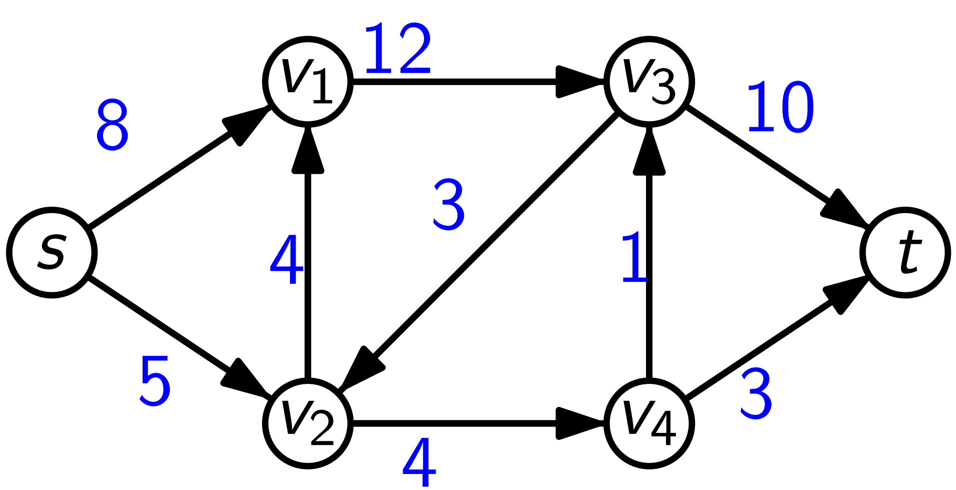
Flüsse

Notation

Sei $G = (V, E)$ ein gerichteter Graph.
Fluss: Funktion $f : E \to \mathbb{R}$
$f(e)$ gibt an, wieviel über Kante e fließt.

- $\delta^+(v)$ ausgehende Kanten von v
-$\delta^-(v)$ eingehende Kanten in v

Nettozufluss$_f(v) := f(\delta^-(v)) - f(\delta^+(v)) = \sum_{(u,v)\in E} f((u,v)) - \sum_{(v,w)\in E} f((v,w))$

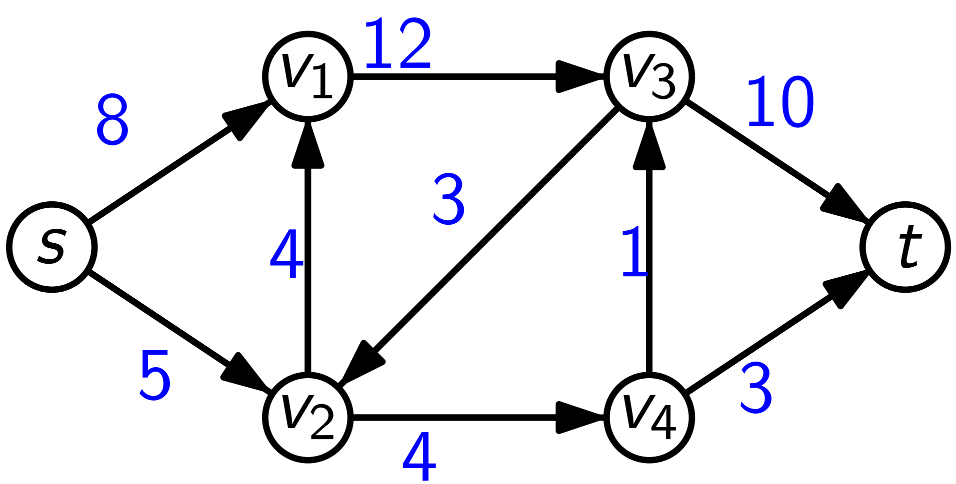


Flüsse

Notation
Sei $G = (V, E)$ ein gerichteter Graph.
Fluss: Funktion $f : E \to \mathbb{R}$
$f(e)$ gibt an, wieviel über Kante e fließt.

- $\delta^+(v)$ ausgehende Kanten von v
-$\delta^-(v)$ eingehende Kanten in v
Nettozufluss$_f(v) := f(\delta^-(v)) - f(\delta^+(v)) = \sum_{(u,v)\in E} f((u,v)) - \sum_{(v,w)\in E} f((v,w))$

Seien $s, t \in V$. Ein zulässiger s-t-**Fluss** ist eine Funktion $f : E \to \mathbb{R}$ mit Nettozufluss$_f(v) = 0\ \forall v \in V \setminus \{s, t,\}$.
Wir nennen s **Quelle** und t **Senke** des Flusses.



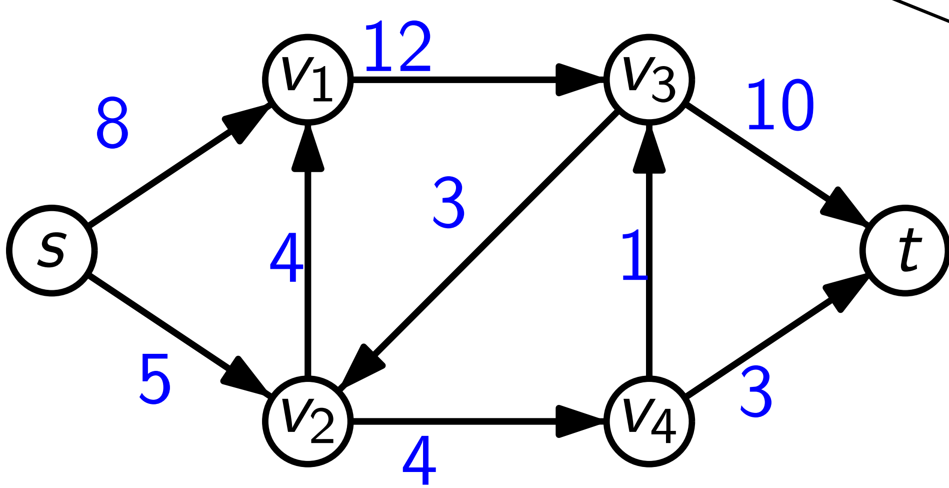
Flüsse

Notation
Sei $G = (V, E)$ ein gerichteter Graph.
Fluss: Funktion $f : E \to \mathbb{R}$
$f(e)$ gibt an, wieviel über Kante e fließt.

- $\delta^+(v)$ ausgehende Kanten von v
- $\delta^-(v)$ eingehende Kanten in v

$\text{Nettozufluss}_f(v) := f(\delta^-(v)) - f(\delta^+(v)) = \sum_{(u,v)\in E} f((u,v)) - \sum_{(v,w)\in E} f((v,w))$

Seien $s, t \in V$. Ein zulässiger s-t-**Fluss** ist eine Funktion $f : E \to \mathbb{R}$ mit $\text{Nettozufluss}_f(v) = 0\ \forall v \in V \setminus \{s, t, \}$.
Wir nennen s **Quelle** und t **Senke** des Flusses.



kurz: **'Fluss'**, wenn aus dem Kontext klar ist, was Quelle und Senke sind.

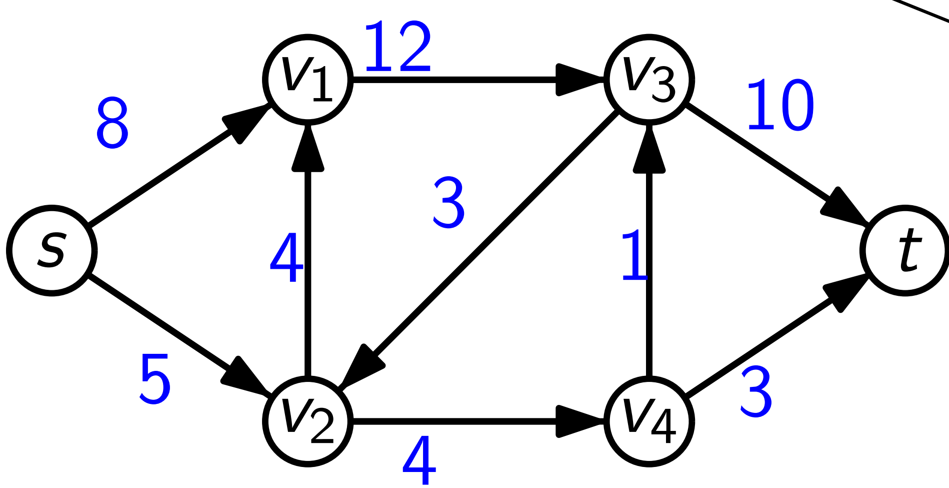
Flüsse

Notation
Sei $G = (V, E)$ ein gerichteter Graph.
Fluss: Funktion $f : E \to \mathbb{R}$
$f(e)$ gibt an, wieviel über Kante e fließt.

- $\delta^+(v)$ ausgehende Kanten von v
-$\delta^-(v)$ eingehende Kanten in v
$\text{Nettozufluss}_f(v) := f(\delta^-(v)) - f(\delta^+(v)) = \sum_{(u,v)\in E} f((u,v)) - \sum_{(v,w)\in E} f((v,w))$

Seien $s, t \in V$. Ein zulässiger s-t-**Fluss** ist eine Funktion $f : E \to \mathbb{R}$ mit $\text{Nettozufluss}_f(v) = 0\ \forall v \in V \setminus \{s, t, \}$.
Wir nennen s **Quelle** und t **Senke** des Flusses.

$|f| := \text{Nettozufluss}_f(t)$ heißt **Wert des Flusses**



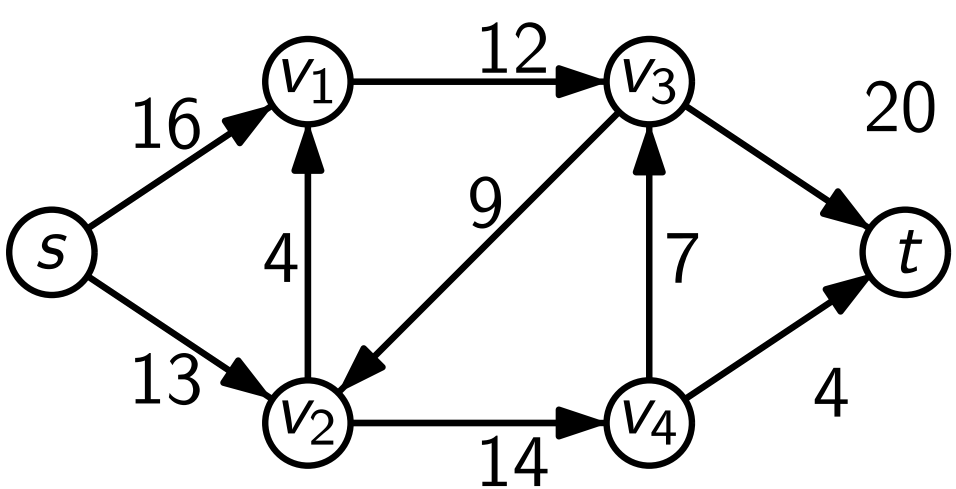
kurz: **'Fluss'**, wenn aus dem Kontext klar ist, was Quelle und Senke sind.

Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss

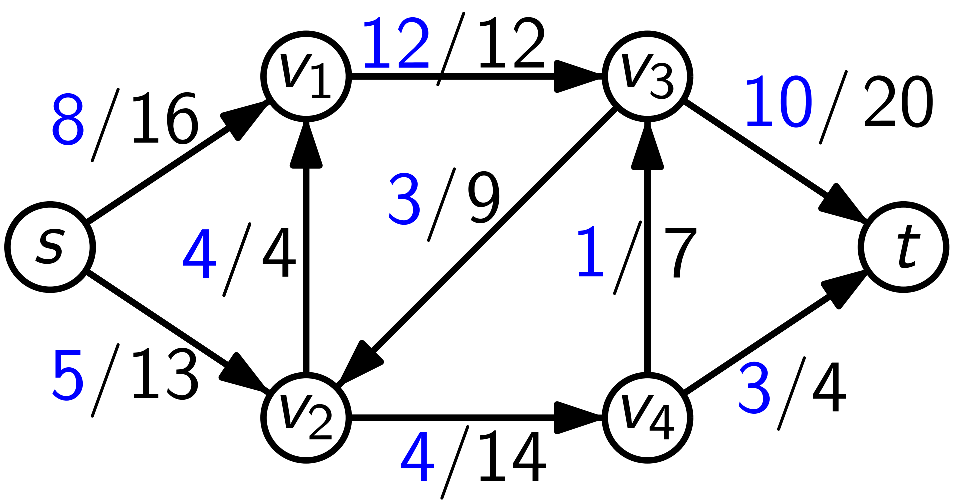
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.

Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$



Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$



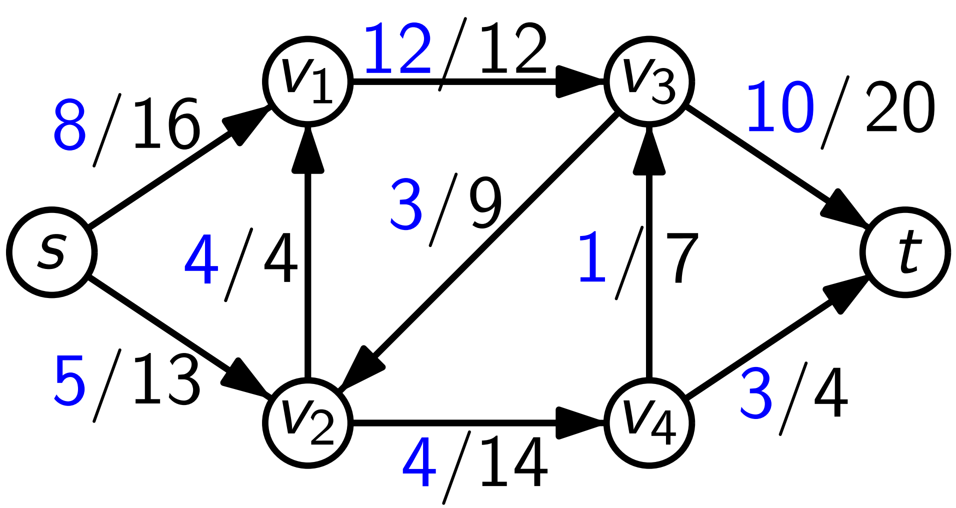
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss

Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.

Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

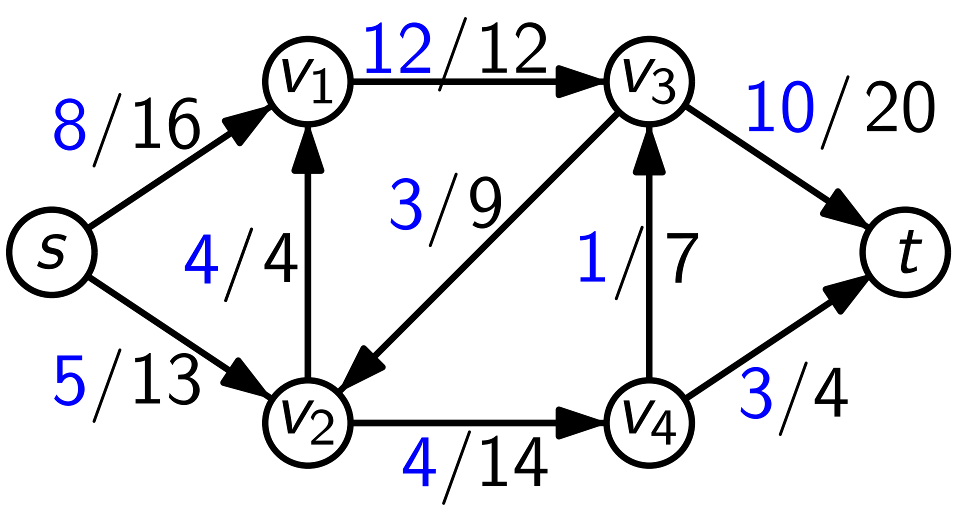


Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?



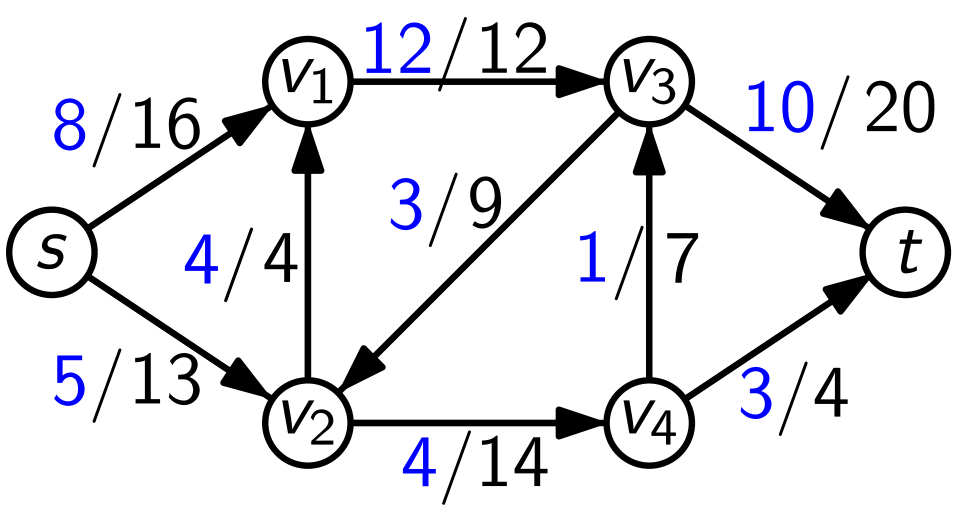
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



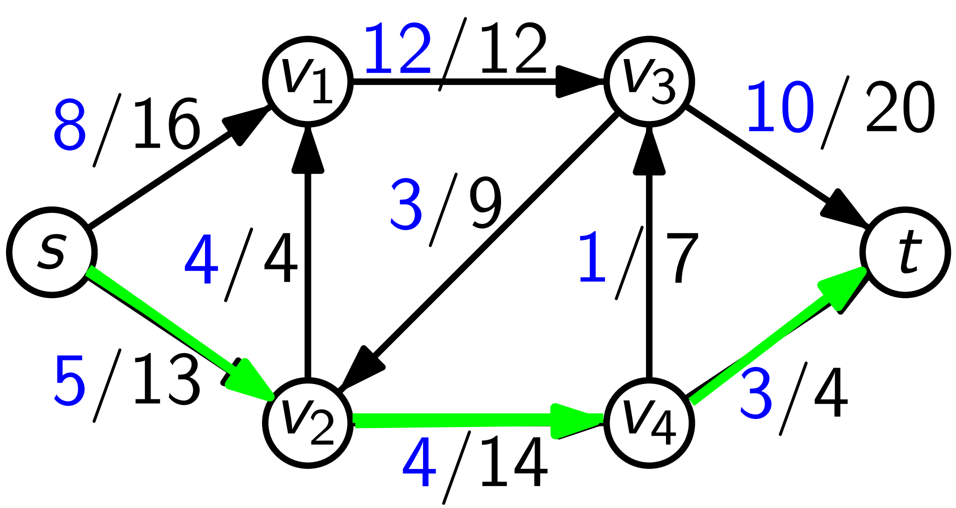
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



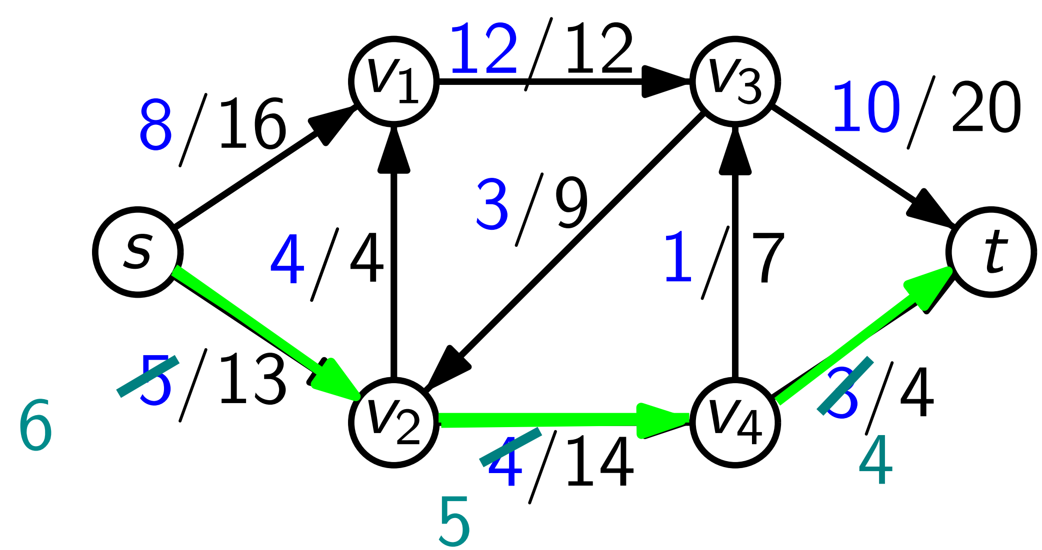
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



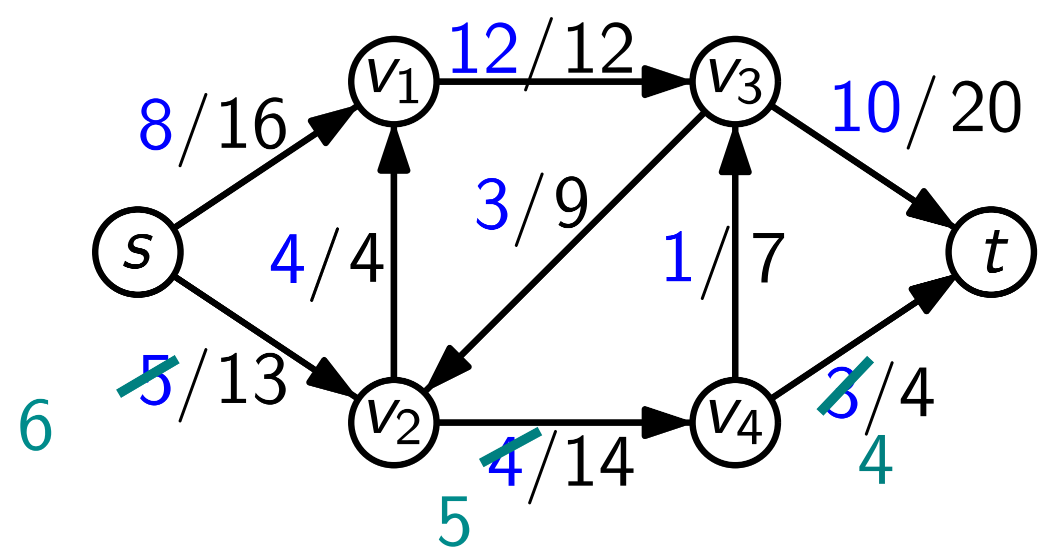
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



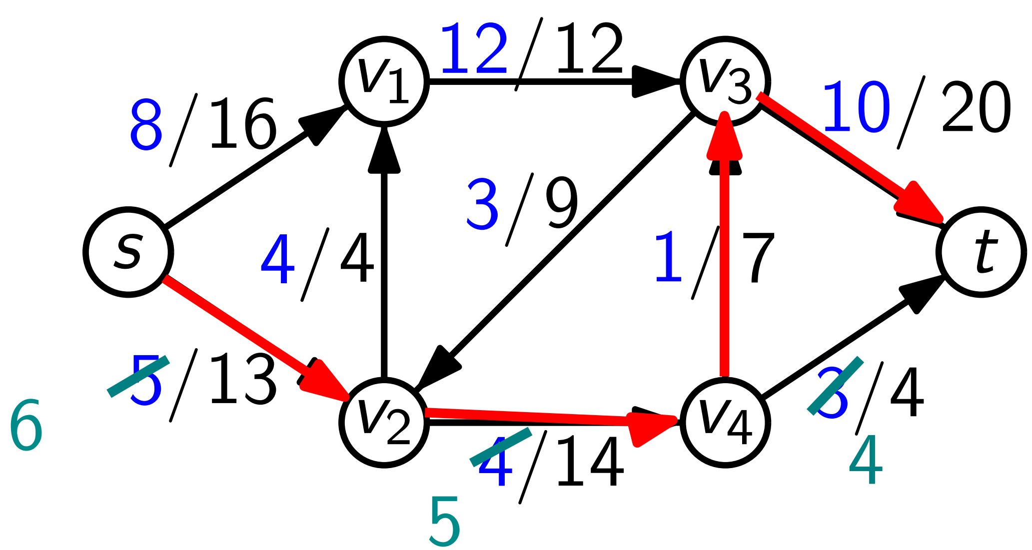
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



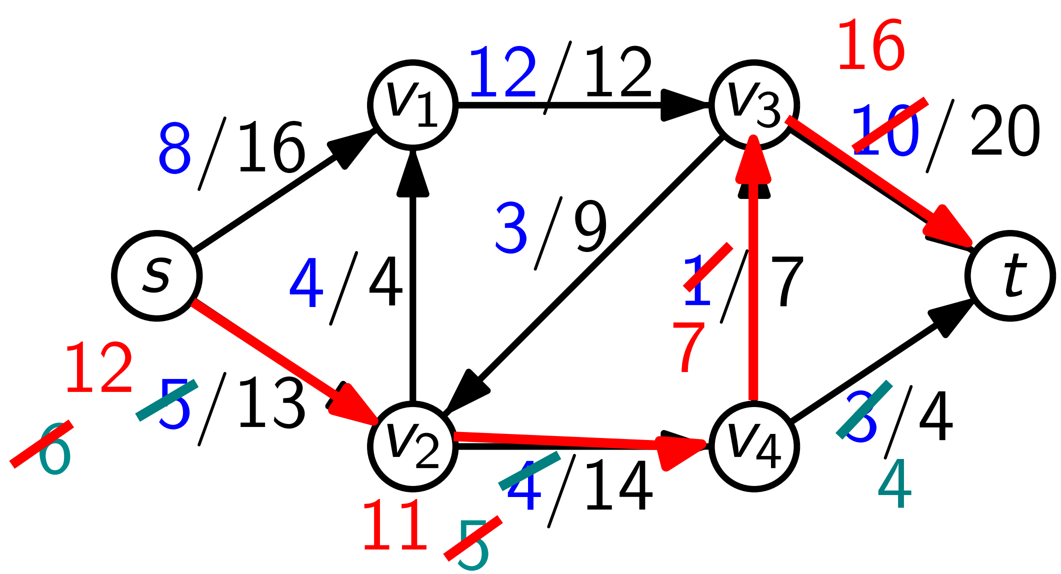
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



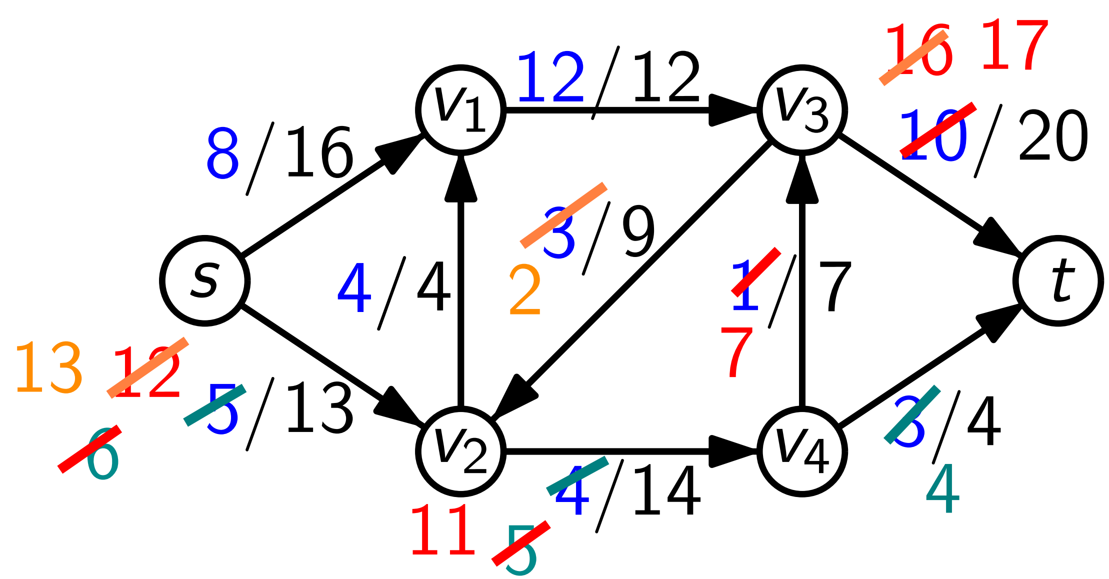
Optimierungsproblem 'Maximaler Fluss'

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.



Optimierungsproblem 'Maximaler Fluss'

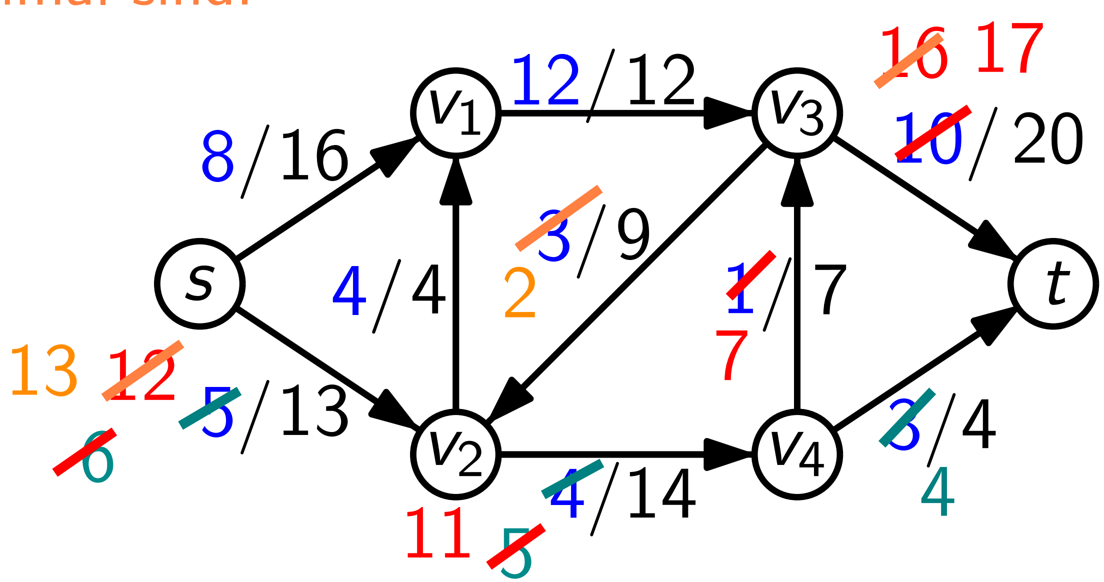
Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Ist das ein maximaler Fluss?

Wie kann ich einen Fluss mit größerem Wert finden?

Beobachtung: Falls es einen s-t-Weg gibt, bei dem auf keiner Kante die Kapazität ausgeschöpft ist, können wir f vergrößern.

Aber wenn es **keinen** solchen Weg gibt, heißt das **nicht** automatisch, dass wir schon optimal sind.

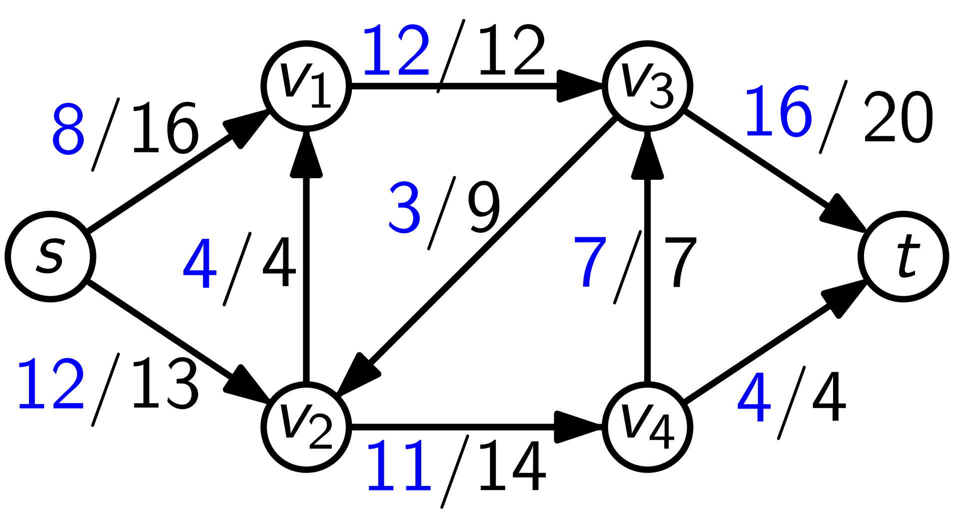


Residualgraph

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$



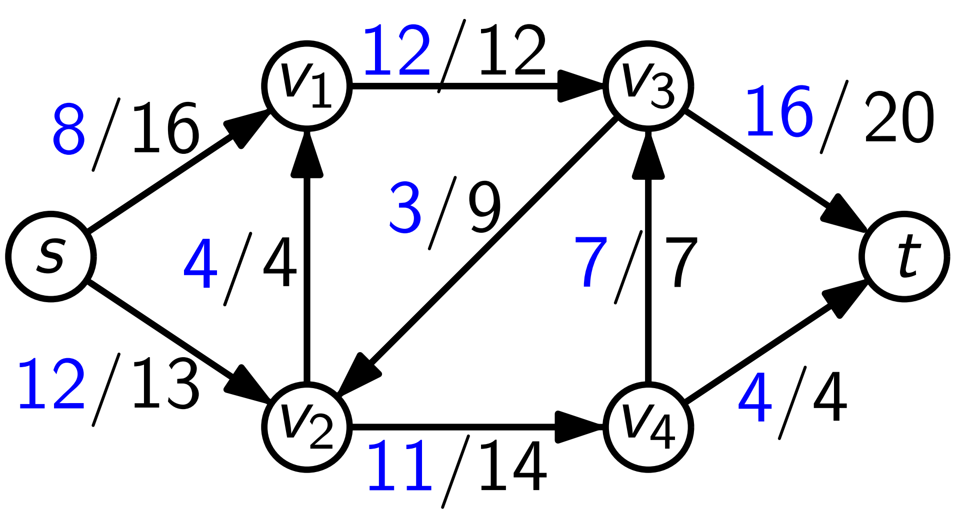
Residualgraph

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten



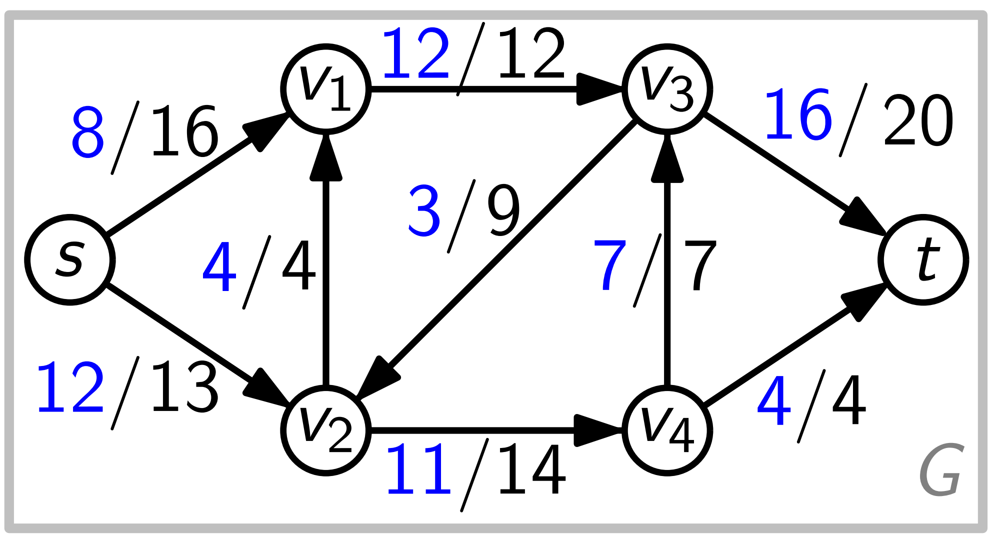
Residualgraph

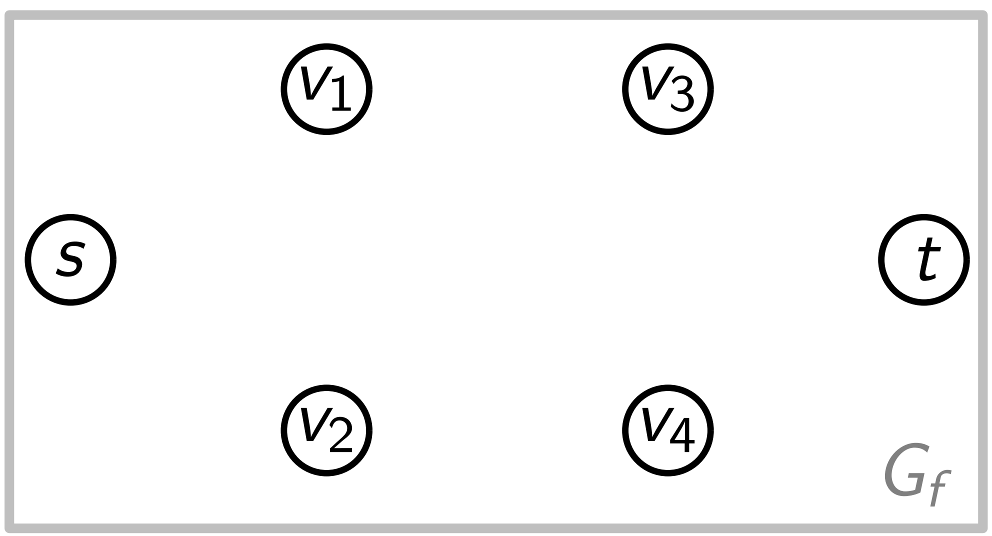
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten





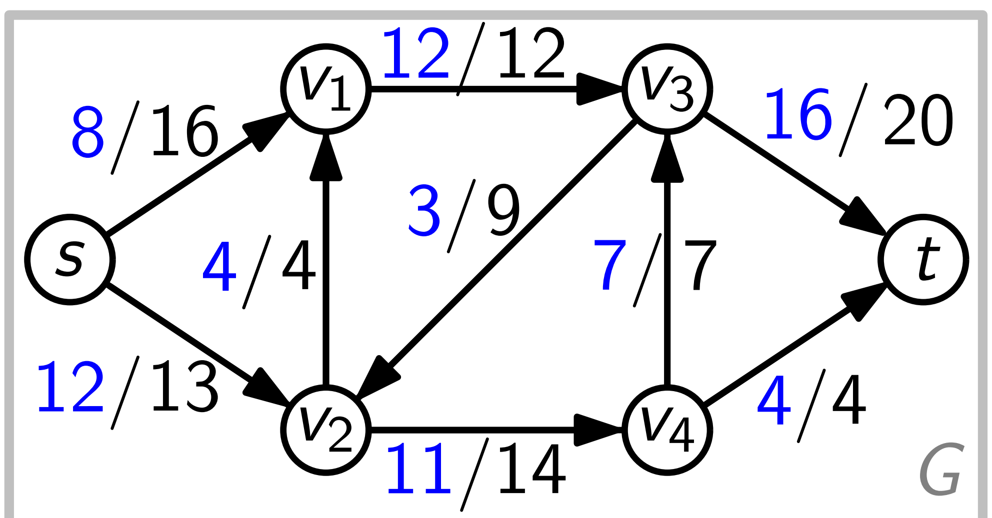
Residualgraph

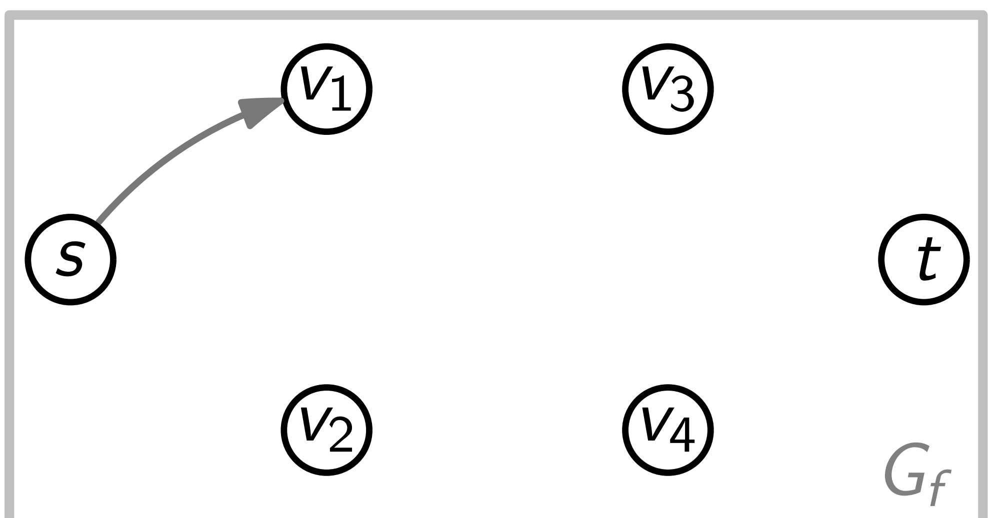
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten



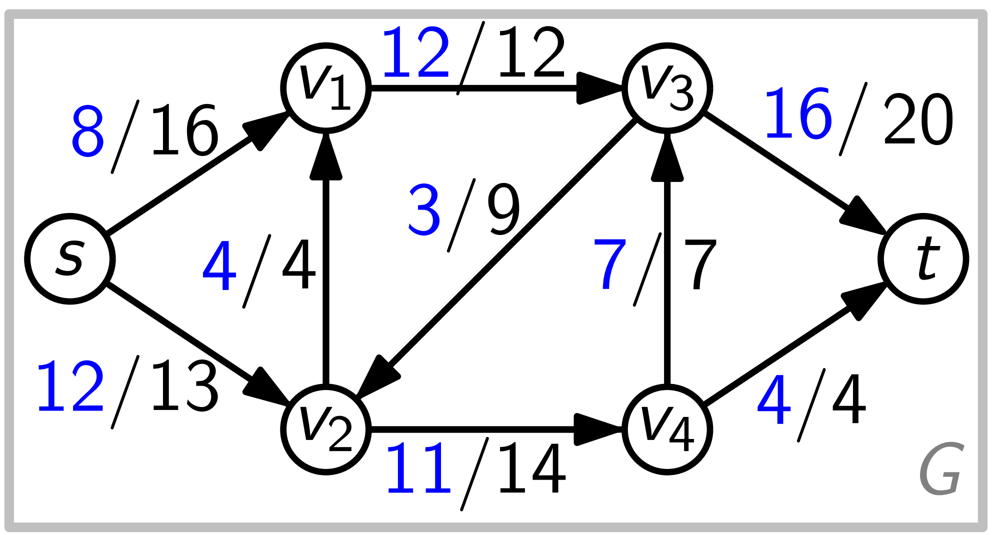


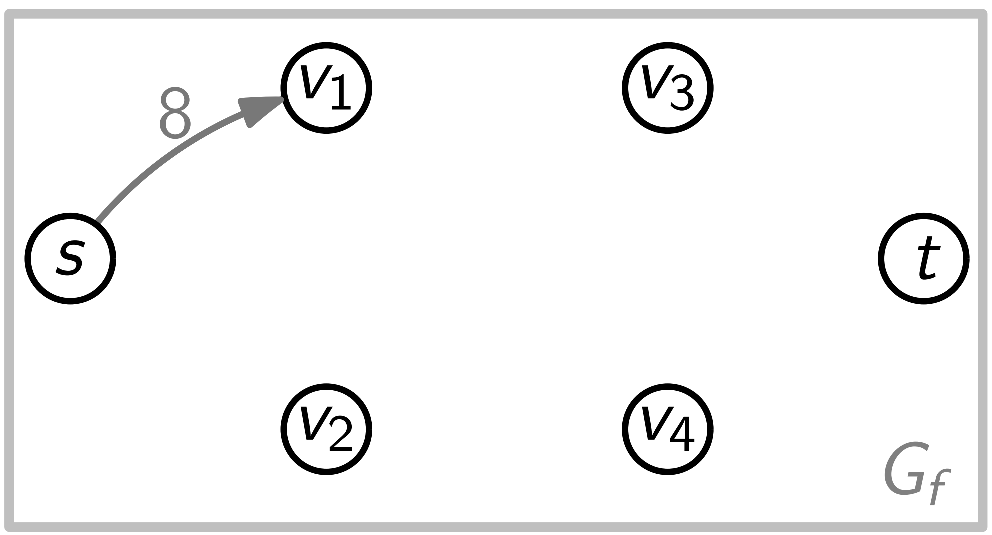
Residualgraph

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.
Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten





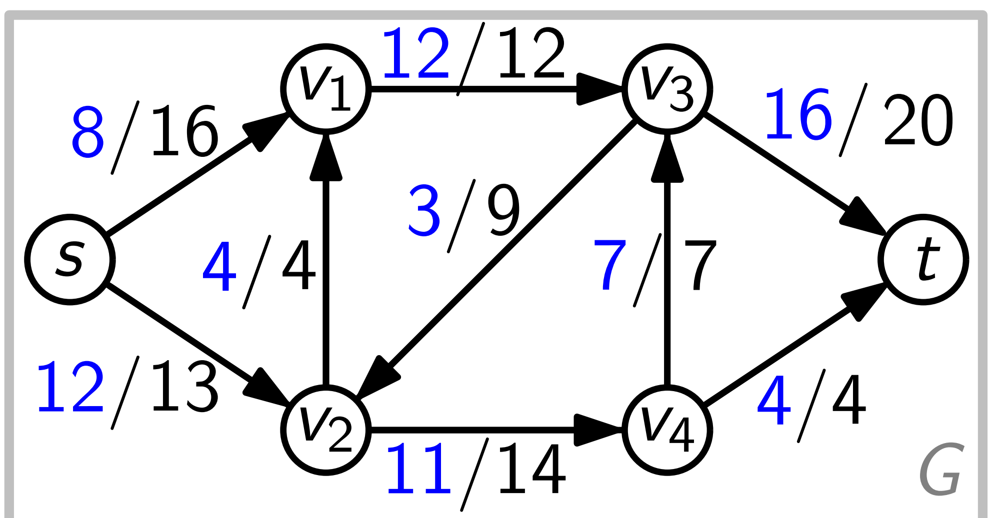
Residualgraph

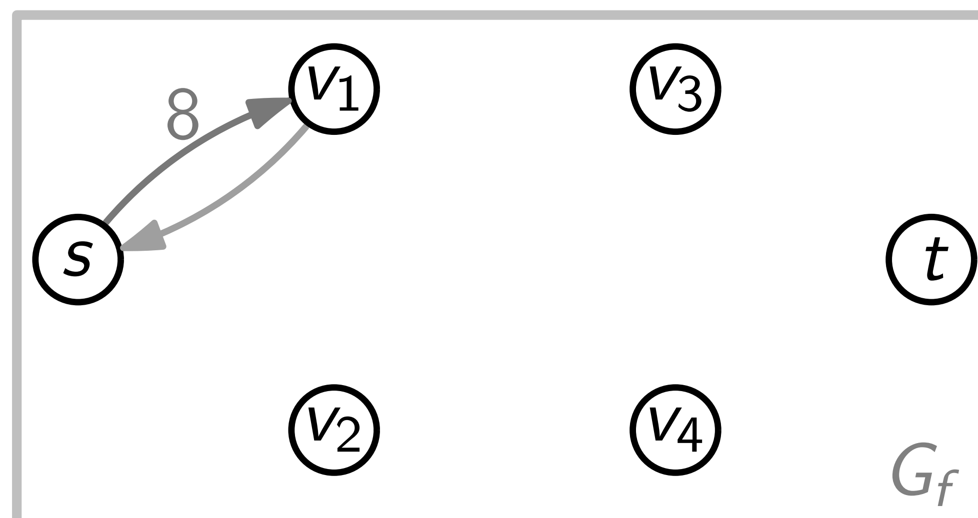
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten





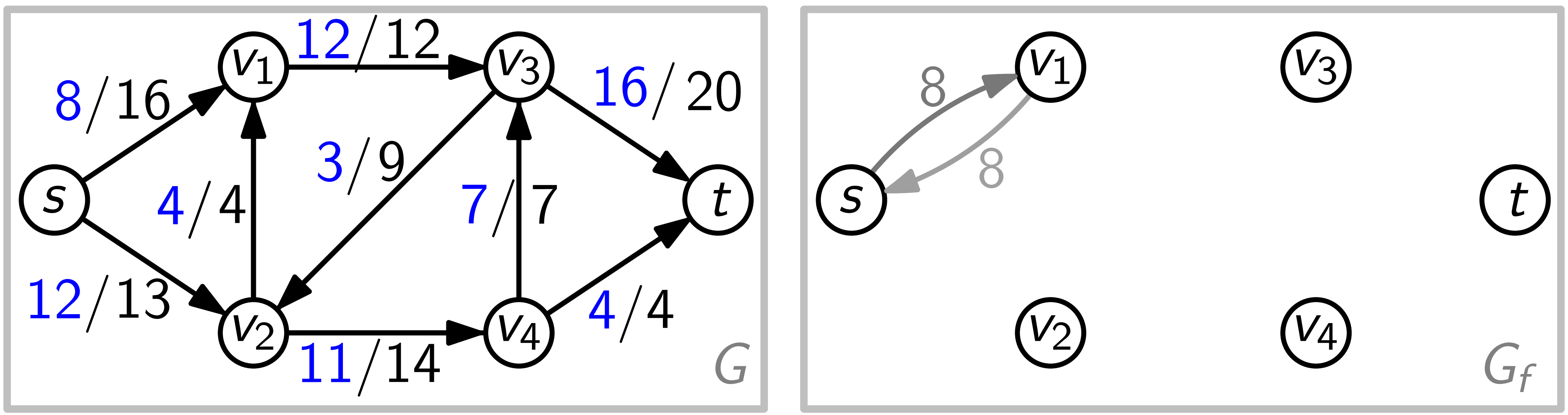
Residualgraph

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten



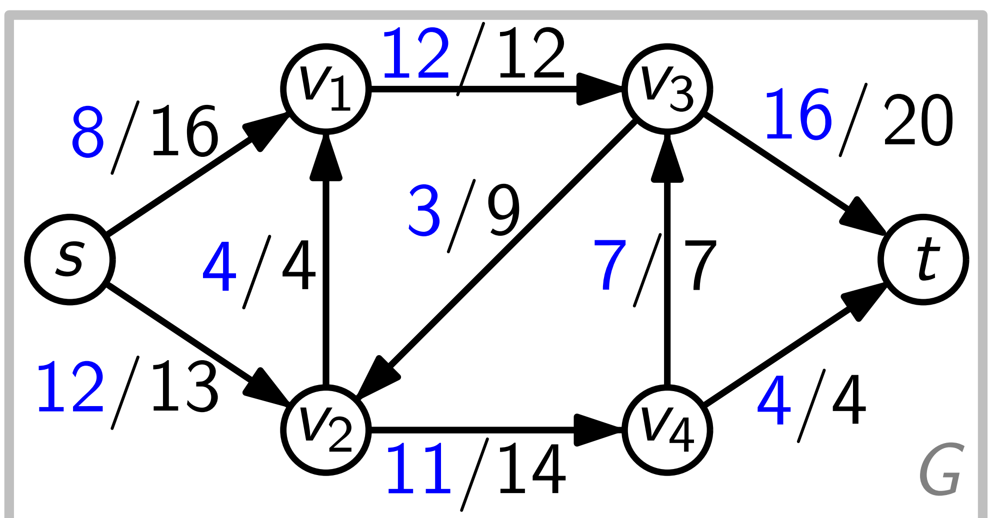
Residualgraph

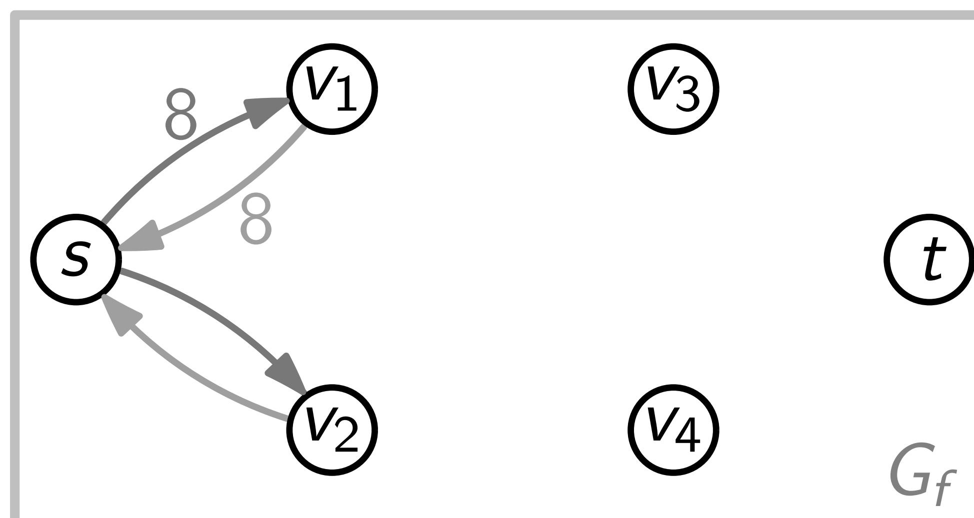
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten



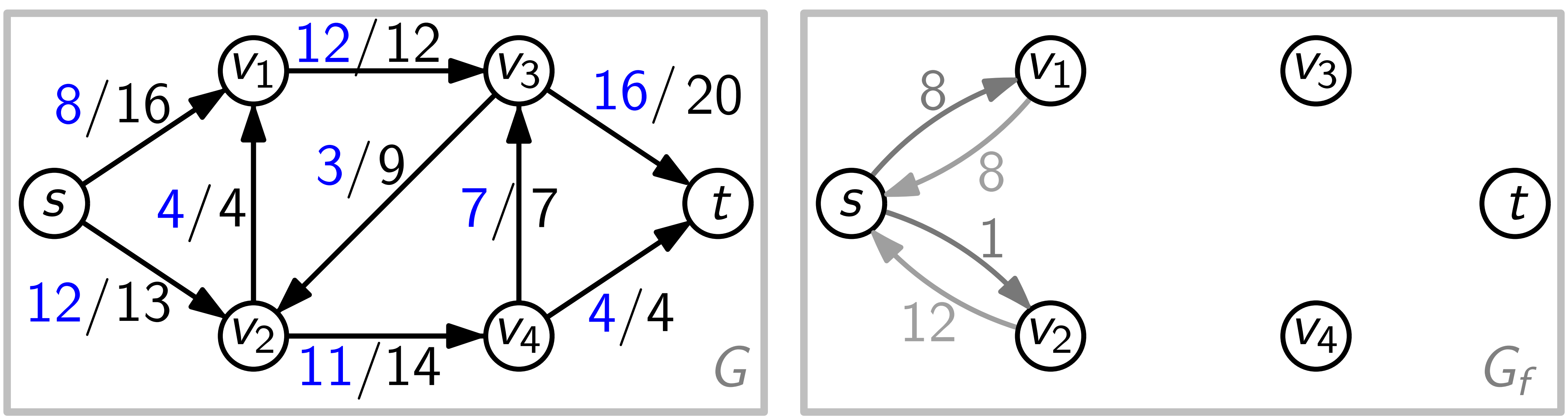


Residualgraph

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.
Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten



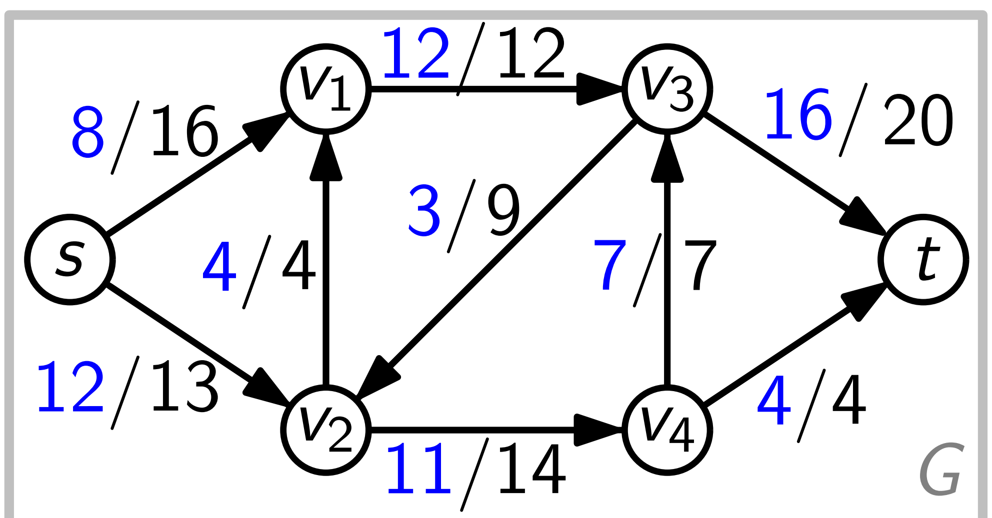
Residualgraph

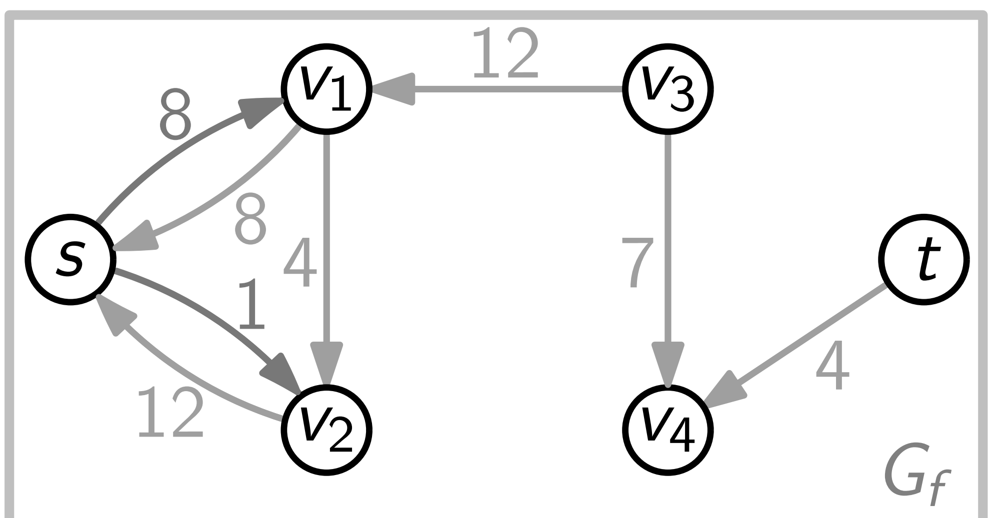
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

Residual-kapazitäten





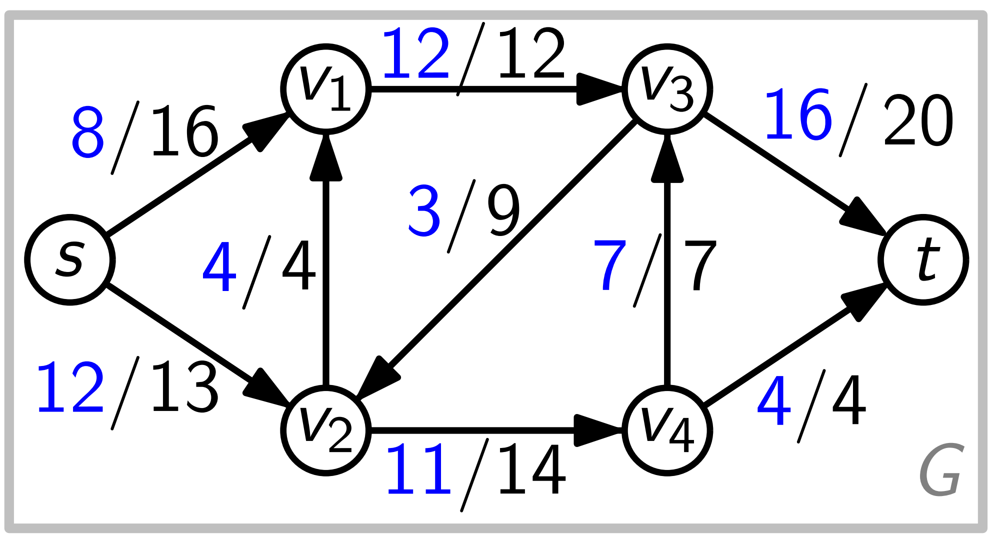
Residualgraph

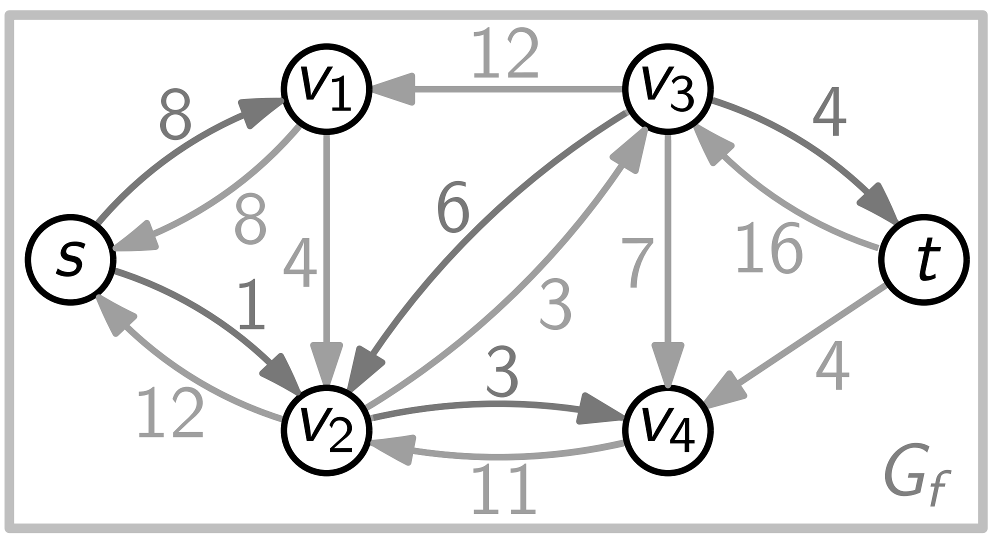
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G.

Der **Residualgraph** $G_f = (V, E_f)$ enthält
für jede Kante $e = uv$ von $G = (V, E)$ die Kante(n)

- $+e := (u, v)$ falls $f(e) < c(e)$ mit $c_f(+e) := c(e) - f(e)$
- $-e := (v, u)$ falls $f(e) > 0$ mit $c_f(-e) := f(e)$

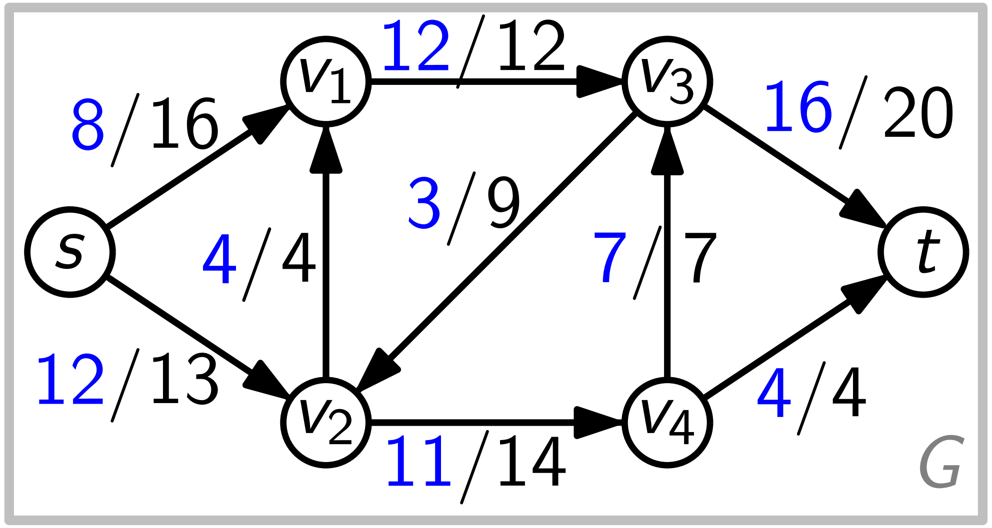
Residual-kapazitäten

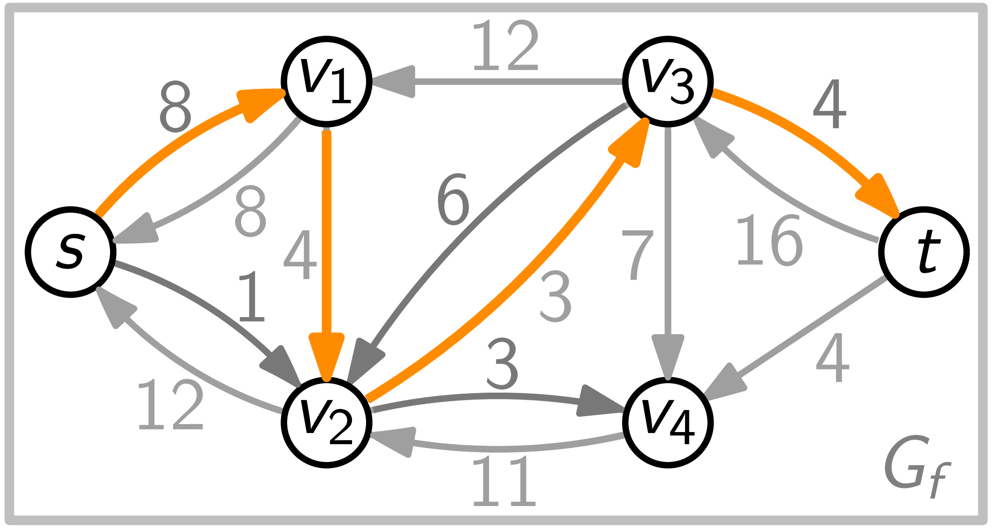




Residualgraph

Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

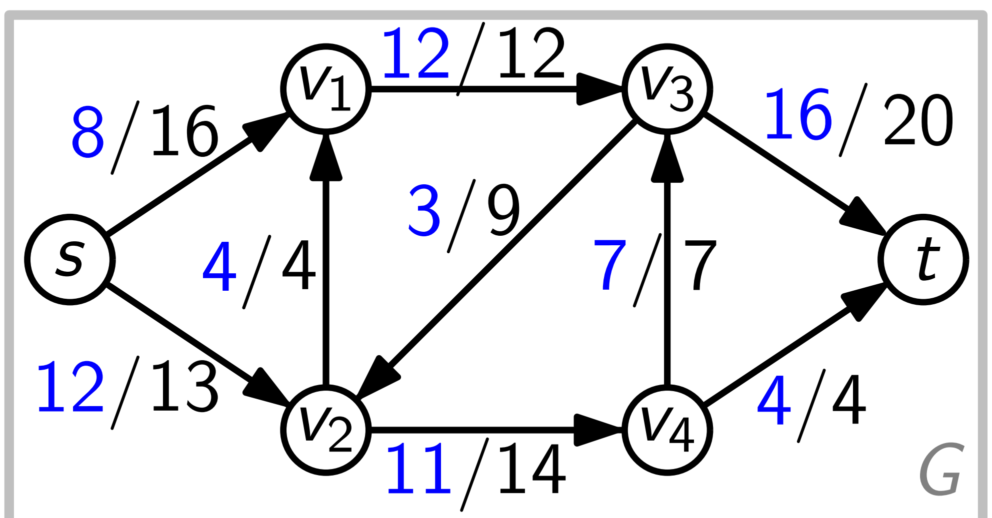


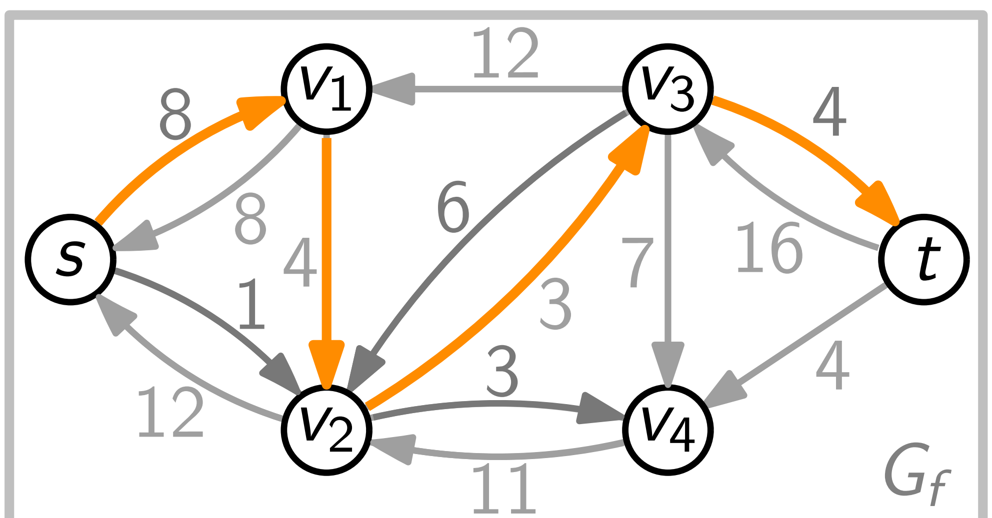


Residualgraph

Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,



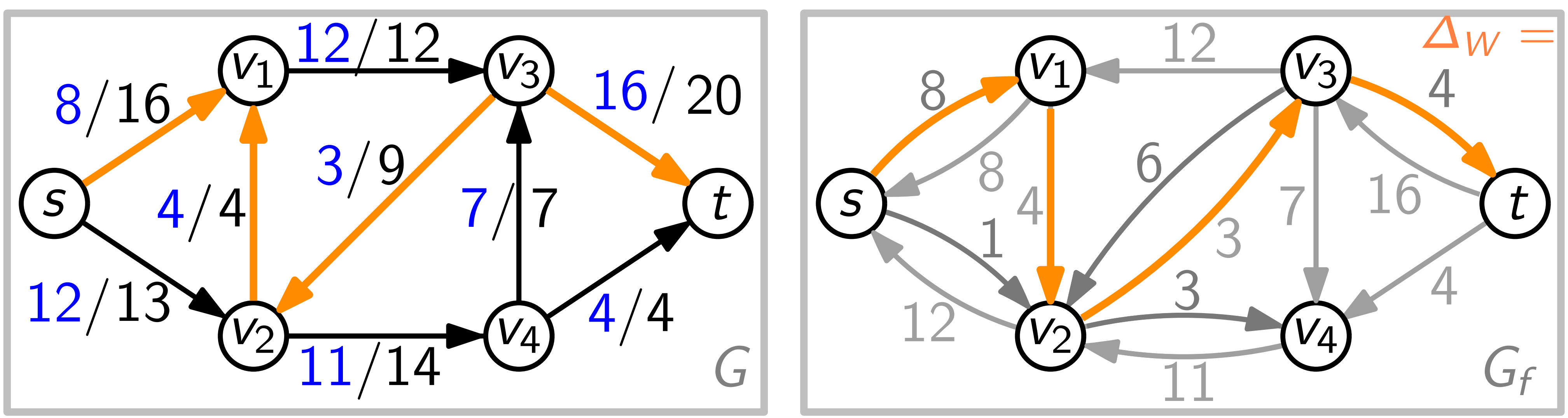


Residualgraph

Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:



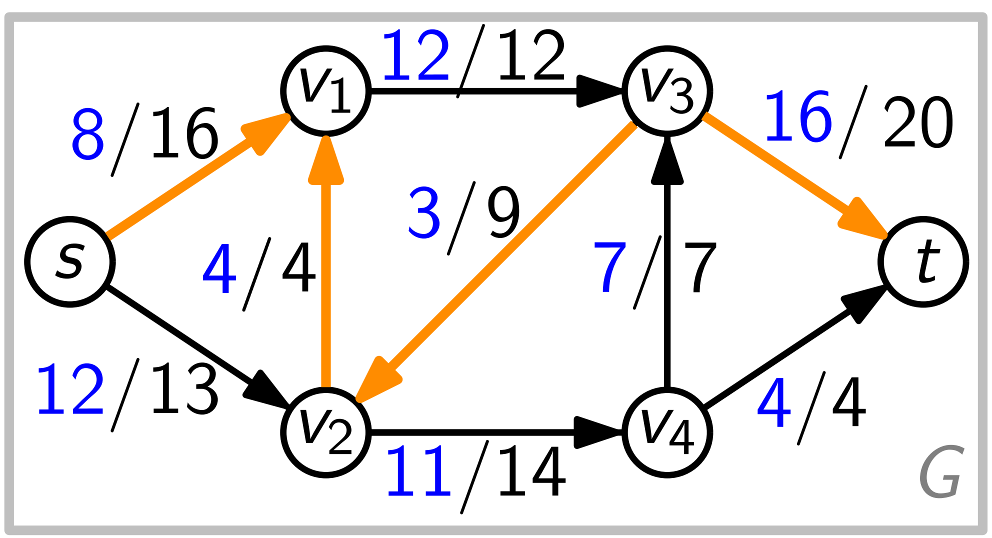
Residualgraph

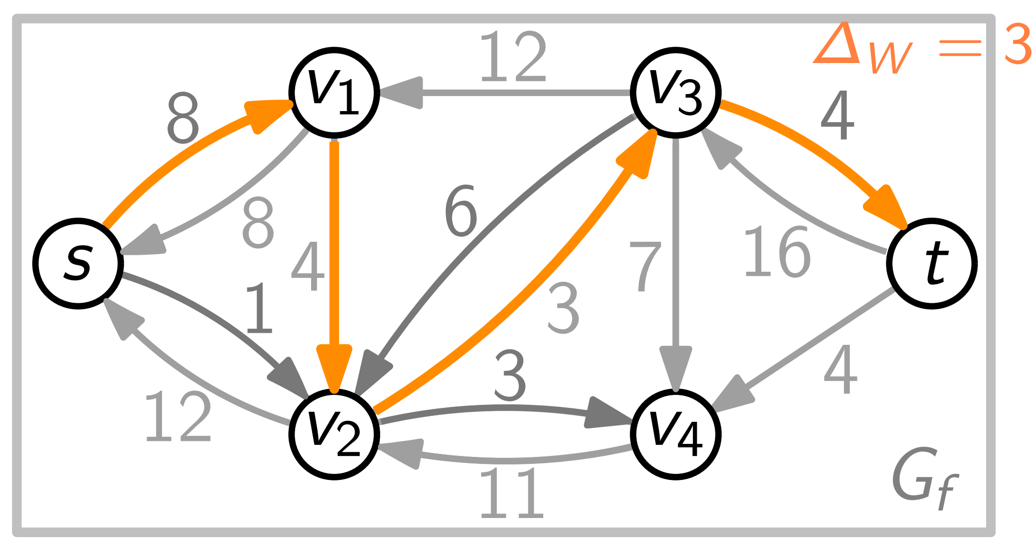
Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:

- falls $(u, v) \in E_W$: $\hat{f}(e) = f(e) + \Delta_W$





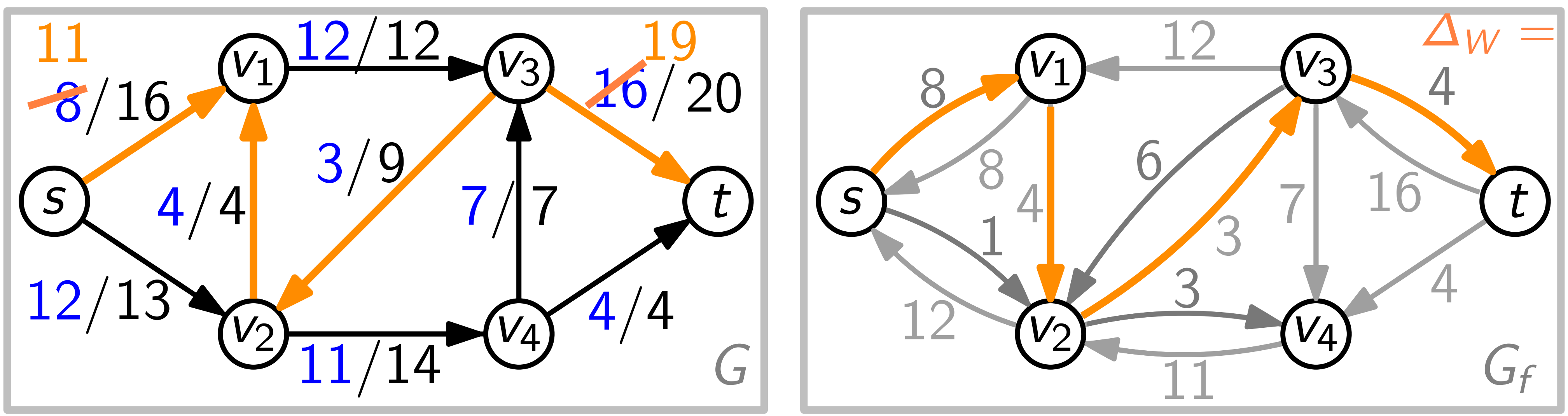
Residualgraph

Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:

- falls $(u, v) \in E_W$: $\hat{f}(e) = f(e) + \Delta_W$



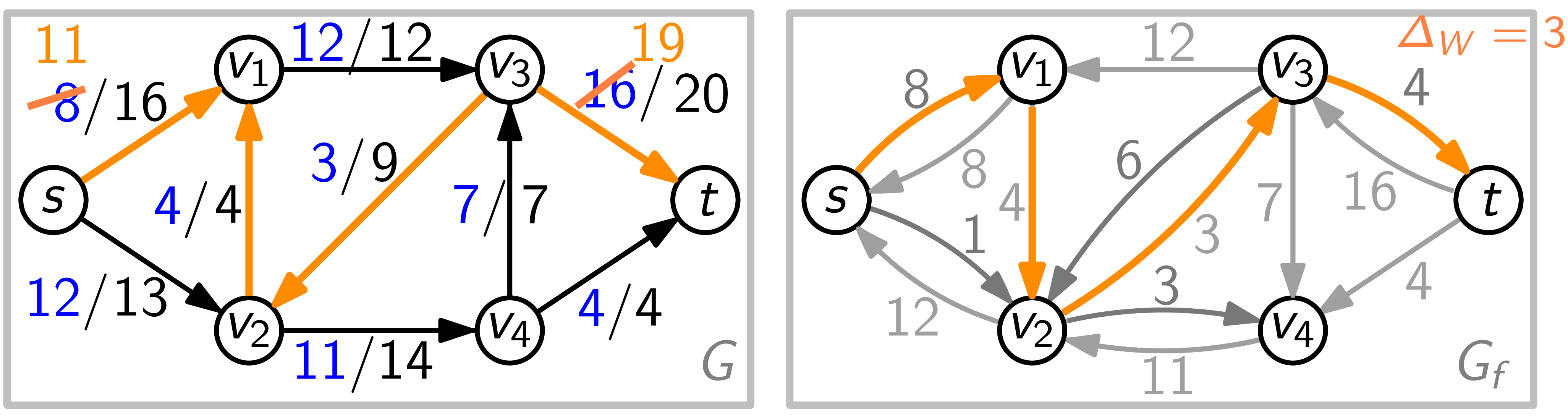
Residualgraph

Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:

- falls $(u, v) \in E_W$: $\hat{f}(e) = f(e) + \Delta_W$
- falls $(v, u) \in E_W$: $\hat{f}(e) = f(e) - \Delta_W$



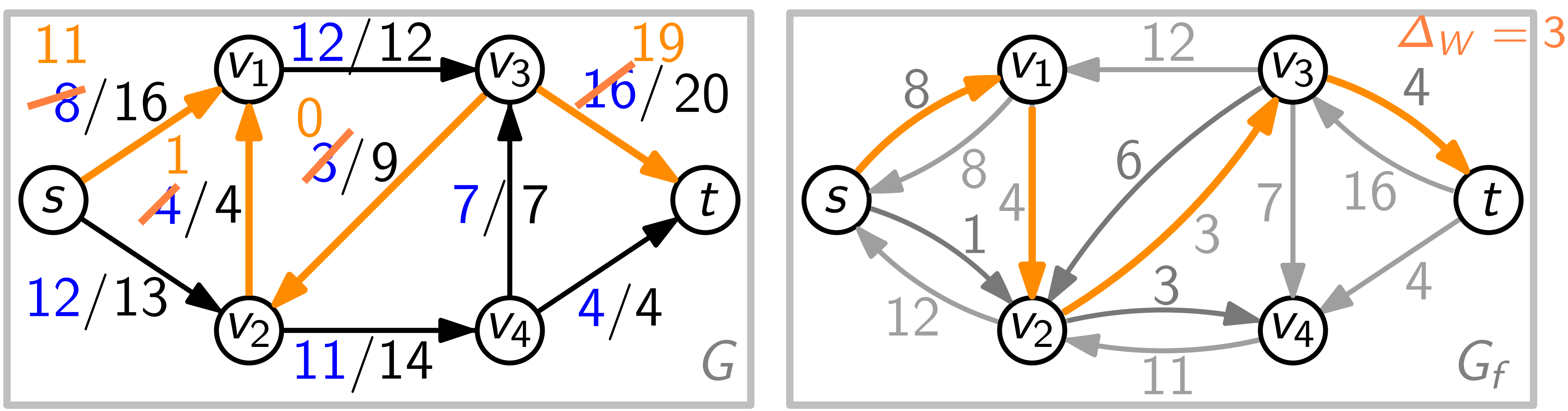
Residualgraph

Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:

- falls $(u, v) \in E_W$: $\hat{f}(e) = f(e) + \Delta_W$
- falls $(v, u) \in E_W$: $\hat{f}(e) = f(e) - \Delta_W$



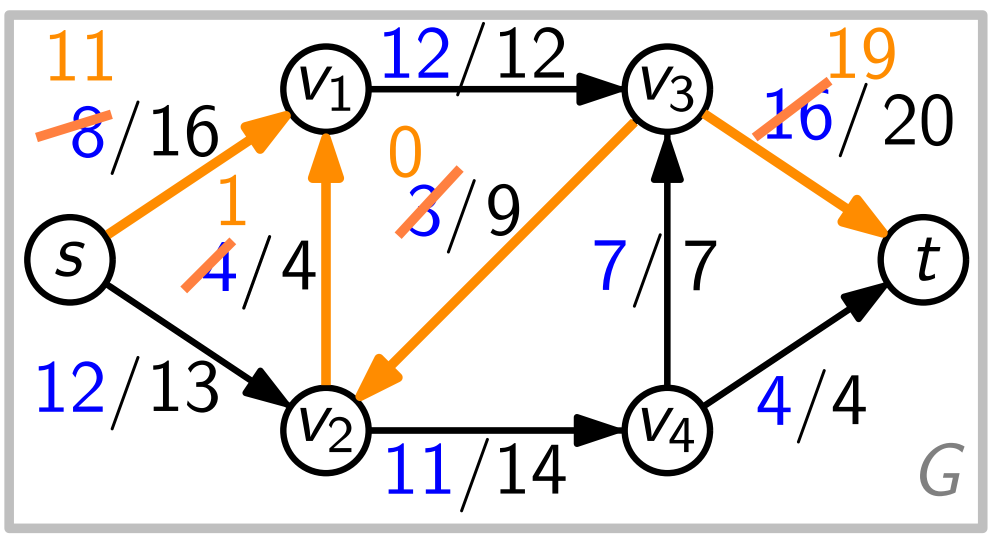
Residualgraph

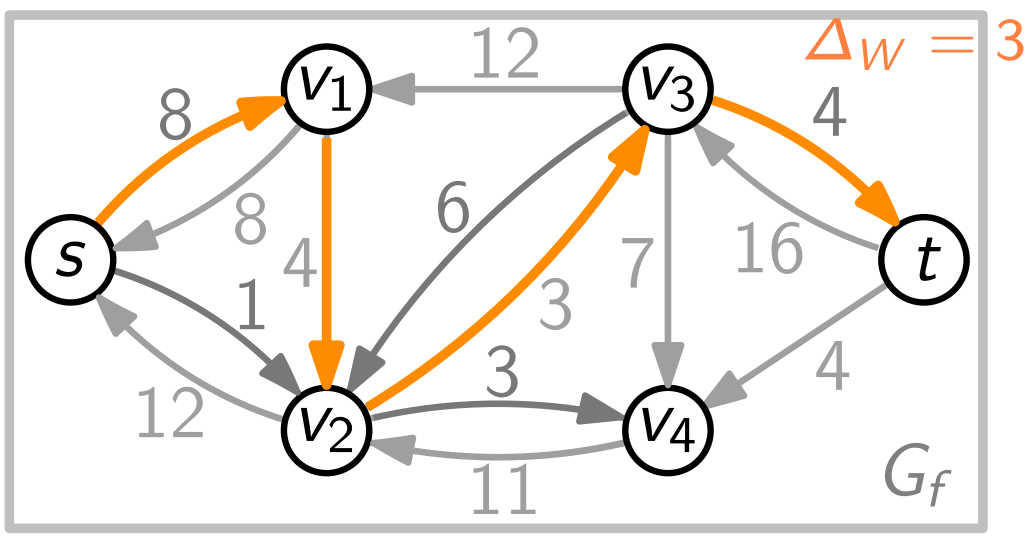
Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:

- falls $(u, v) \in E_W$: $\hat{f}(e) = f(e) + \Delta_W$
- falls $(v, u) \in E_W$: $\hat{f}(e) = f(e) - \Delta_W$
- sonst: $\hat{f}(e) = f(e)$





Residualgraph

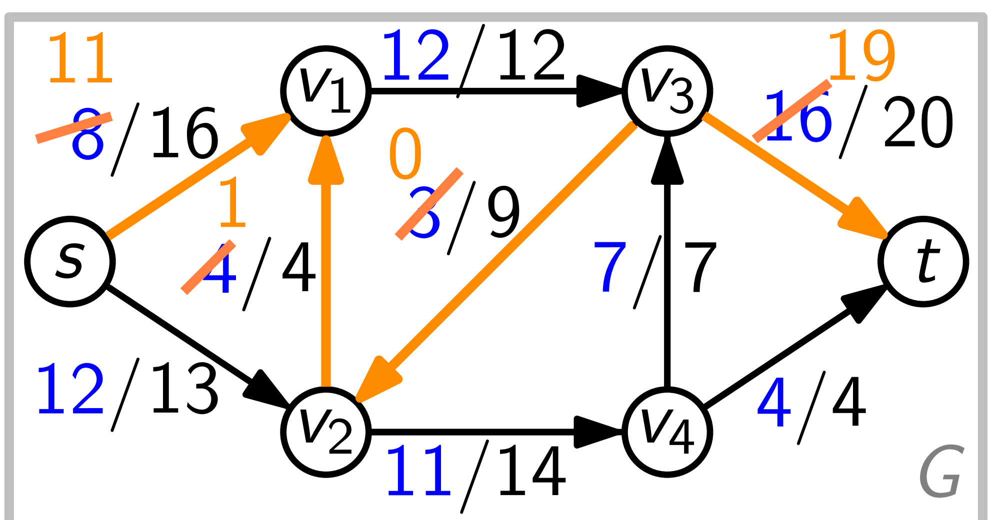
Ein s-t-Weg $W = (V_W, E_W)$ in G_f heißt **flussvergrößernder Weg** für f.

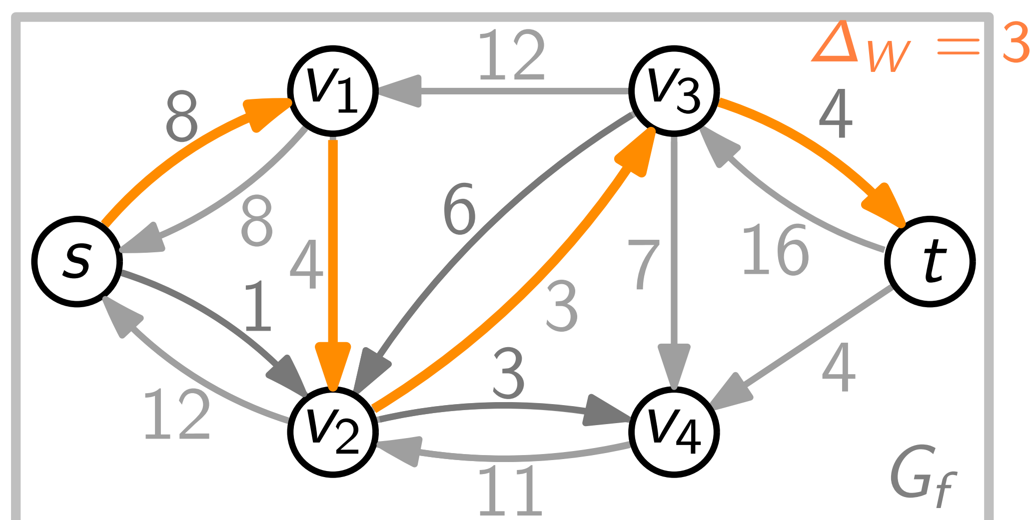
Die **Residualkapazität** von W ist $\Delta_W := \min_{e \in W} c_f(e)$,

Definiere neuen s-t-Fluss $\hat{f}$ auf G kantenweise: für alle $e = (u, v) \in E$:

- falls $(u, v) \in E_W$: $\hat{f}(e) = f(e) + \Delta_W$
- falls $(v, u) \in E_W$: $\hat{f}(e) = f(e) - \Delta_W$
- sonst: $\hat{f}(e) = f(e)$

Begründen Sie: 1. $\hat{f}$ ist ein s-t-Fluss
2. $\hat{f}$ ist ein *zulässiger* s-t-Fluss
3. $|\hat{f}| = |f| + \Delta_W$
4. Wenn f^* ein maximaler Fluss ist, gibt es in G_{f^*} keinen Weg von s nach t.

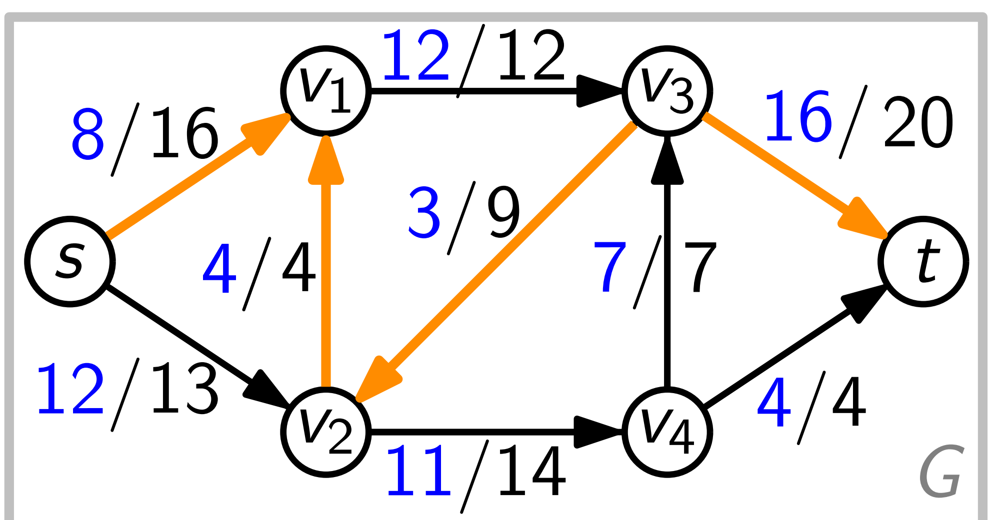


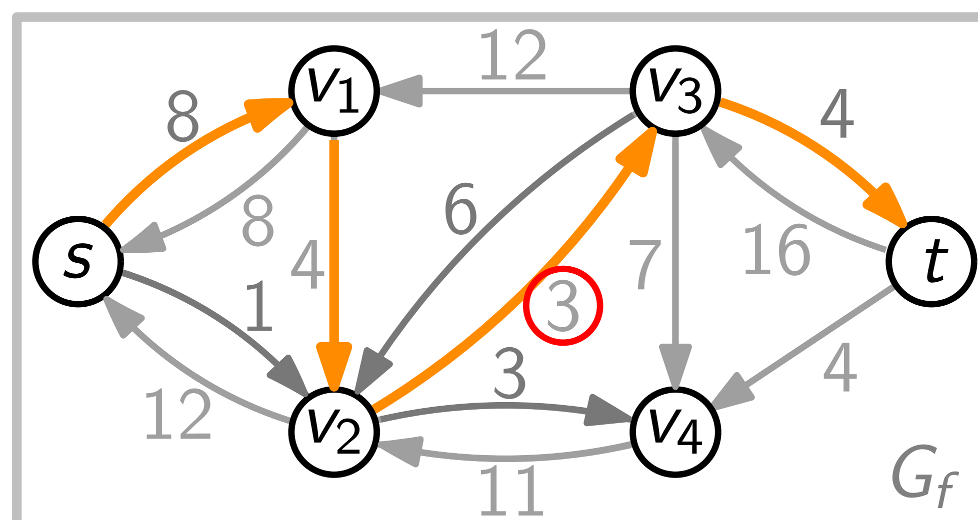


Der Satz vom flussvergrößernden Weg

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.



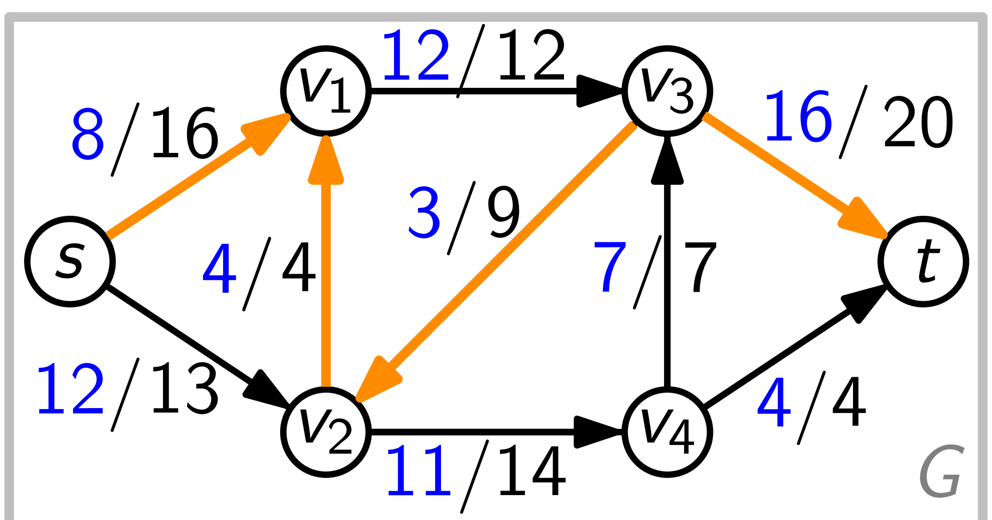


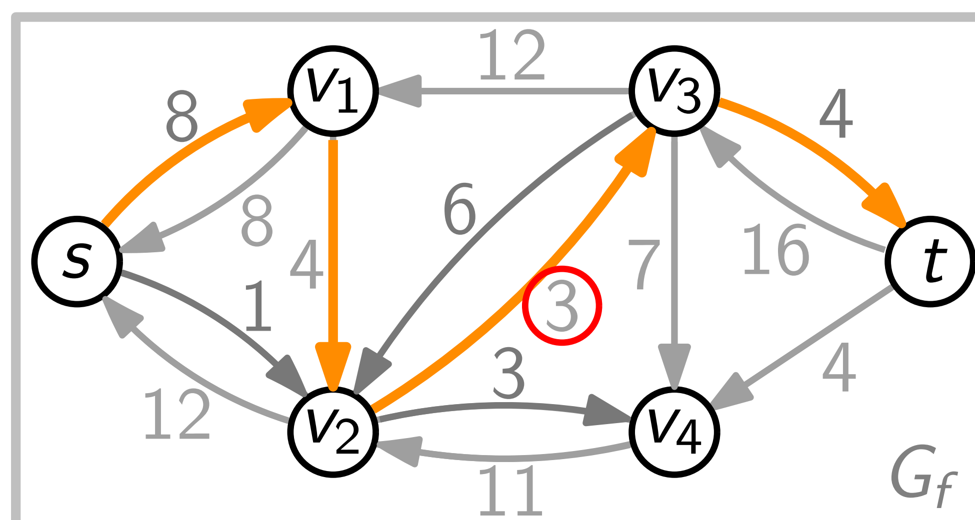
Der Satz vom flussvergrößernden Weg

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis?





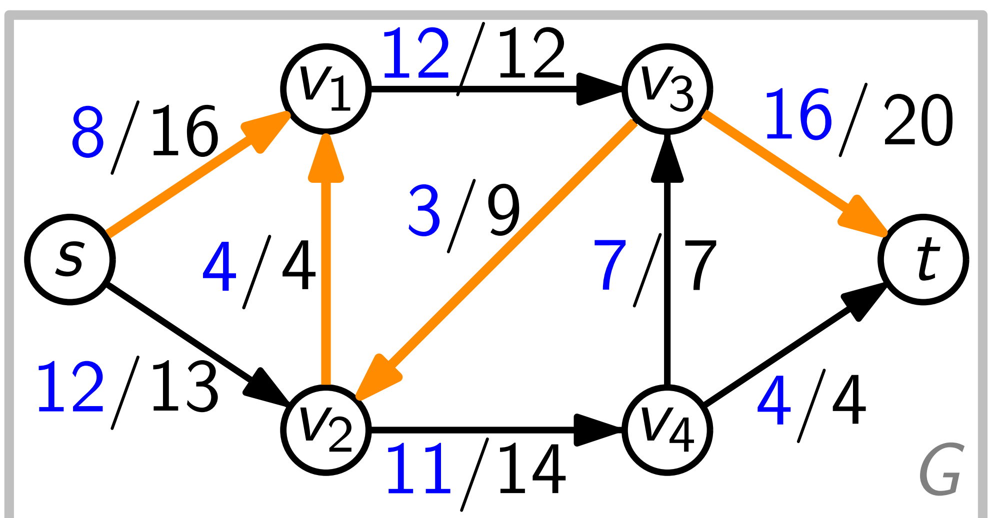
Der Satz vom flussvergrößernden Weg

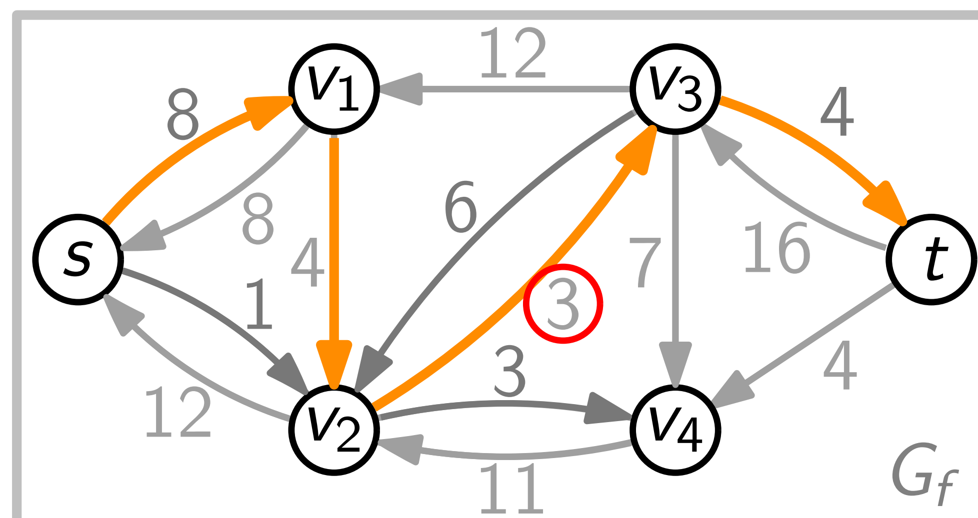
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis?

"$\Rightarrow$ siehe Überlegungen vorherige Seite





Der Satz vom flussvergrößernden Weg

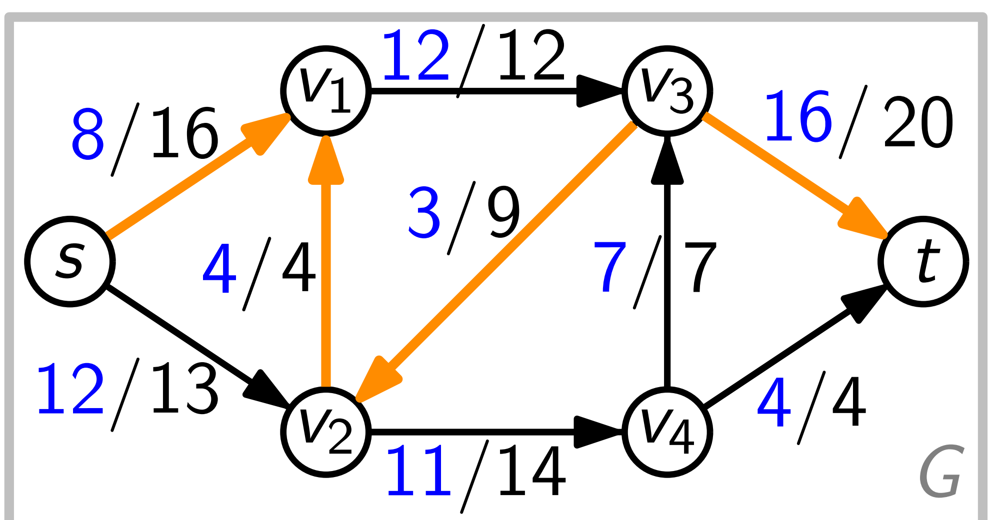
$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

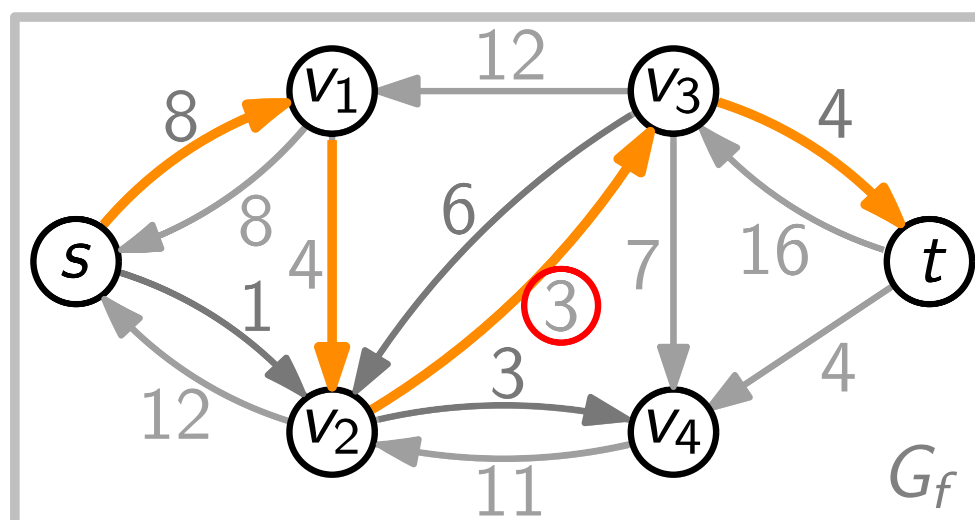
Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis?

"$\Rightarrow$ siehe Überlegungen vorherige Seite

"$\Leftarrow$" Beweis Rückrichtung folgt später!





Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W **do**
  $\Delta_W = \min_{(u,v)\in W} c_f((u,v))$
  for $(u, v) \in W$ **do**
    if $(u, v) \in E$ **then**
      $f_{(u,v)} = f_{(u,v)} + \Delta_W$
    else
      $f_{(v,u)} = f_{(v,u)} - \Delta_W$
    end if
  end for
  Erstelle Residualgraph G_f
end while
return f

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v)\in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Korrektheit?

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W **do**
  $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
  for $(u, v) \in W$ **do**
    if $(u, v) \in E$ **then**
      $f_{(u,v)} = f_{(u,v)} + \Delta_W$
    else
      $f_{(v,u)} = f_{(v,u)} - \Delta_W$
    end if
  end for
  Erstelle Residualgraph G_f
end while
return f

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t

Ensure: s-t-Fluss f

```
Initialisiere: f(e) = 0 für alle e ∈ E
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W do
    Δ_W = min_{(u,v)∈W} c_f((u, v))
    for (u, v) ∈ W do
        if (u, v) ∈ E then
            f_(u,v) = f_(u,v) + Δ_W
        else
            f_(v,u) = f_(v,u) − Δ_W
        end if
    end for
    Erstelle Residualgraph G_f
end while
return f
```

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v)\in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit? 1. $c\colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W **do**
  $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
  for $(u, v) \in W$ **do**
    if $(u, v) \in E$ **then**
      $f_{(u,v)} = f_{(u,v)} + \Delta_W$
    else
      $f_{(v,u)} = f_{(v,u)} - \Delta_W$
    end if
  end for
  Erstelle Residualgraph G_f
end while
return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c \colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c\colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v)\in W} c_f((u,v))$
    for $(u,v) \in W$ **do**
      if $(u,v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
– Breitensuche
– Tiefensuche

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c\colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u,v))$
    for $(u, v) \in W$ **do**
        if $(u, v) \in E$ **then**
            $f_{(u,v)} = f_{(u,v)} + \Delta_W$
        else
            $f_{(v,u)} = f_{(v,u)} - \Delta_W$
        end if
    end for
    Erstelle Residualgraph G_f
end while
return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c \colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

Anz. Schleifendurchläufe

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit? 1. $c \colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:

$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen

– Breitensuche
– Tiefensuche
$\}$ $O(|E|)$ Zeit

Anz. Schleifendurchläufe

– in jedem Durchlauf wird f um ≥ 1 vergrößert

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit? 1. $c \colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
– Breitensuche
– Tiefensuche
$\Big\}$ $O(|E|)$ Zeit

Anz. Schleifendurchläufe
– in jedem Durchlauf wird f um ≥ 1 vergrößert
– max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit? 1. $c \colon E \to \mathbb{N}$:
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u,v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

Anz. Schleifendurchläufe
- in jedem Durchlauf wird f um ≥ 1 vergrößert
- max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c\colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen

$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

Anz. Schleifendurchläufe
- in jedem Durchlauf wird f um ≥ 1 vergrößert
- max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c \colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$:
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W **do**
$\Delta_W = \min_{(u,v) \in W} c_f((u,v))$
for $(u,v) \in W$ **do**
if $(u,v) \in E$ **then**
$f_{(u,v)} = f_{(u,v)} + \Delta_W$
else
$f_{(v,u)} = f_{(v,u)} - \Delta_W$
end if
end for
Erstelle Residualgraph G_f
end while
return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

Anz. Schleifendurchläufe
– in jedem Durchlauf wird f um ≥ 1 vergrößert
– max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit? 1. $c\colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$: erweitern...
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

```
Initialisiere: f(e) = 0 für alle e ∈ E
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W do
    Δ_W = min_{(u,v)∈W} c_f((u,v))
    for (u,v) ∈ W do
        if (u,v) ∈ E then
            f_(u,v) = f_(u,v) + Δ_W
        else
            f_(v,u) = f_(v,u) − Δ_W
        end if
    end for
    Erstelle Residualgraph G_f
end while
return f
```

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

Anz. Schleifendurchläufe
- in jedem Durchlauf wird f um ≥ 1 vergrößert
- max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c\colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$: erweitern...
3. $\mathbb{R}_{>0}$:

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v)\in W} c_f((u, v))$
    for $(u, v) \in W$ **do**
      if $(u, v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen
$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\}$ $O(|E|)$ Zeit

Anz. Schleifendurchläufe
– in jedem Durchlauf wird f um ≥ 1 vergrößert
– max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit? 1. $c\colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$: erweitern...
3. $\mathbb{R}_{>0}$: problematisch!

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t

Ensure: s-t-Fluss f

  Initialisiere: $f(e) = 0$ für alle $e \in E$
  Erstelle Residualgraph G_f
  while G_f enthält s-t-Weg W **do**
    $\Delta_W = \min_{(u,v) \in W} c_f((u,v))$
    for $(u,v) \in W$ **do**
      if $(u,v) \in E$ **then**
        $f_{(u,v)} = f_{(u,v)} + \Delta_W$
      else
        $f_{(v,u)} = f_{(v,u)} - \Delta_W$
      end if
    end for
    Erstelle Residualgraph G_f
  end while
  return f

Erstelle G_f:
$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen

- Breitensuche
- Tiefensuche

$\left.\begin{array}{l}\text{– Breitensuche}\\ \text{– Tiefensuche}\end{array}\right\} O(|E|)$ Zeit

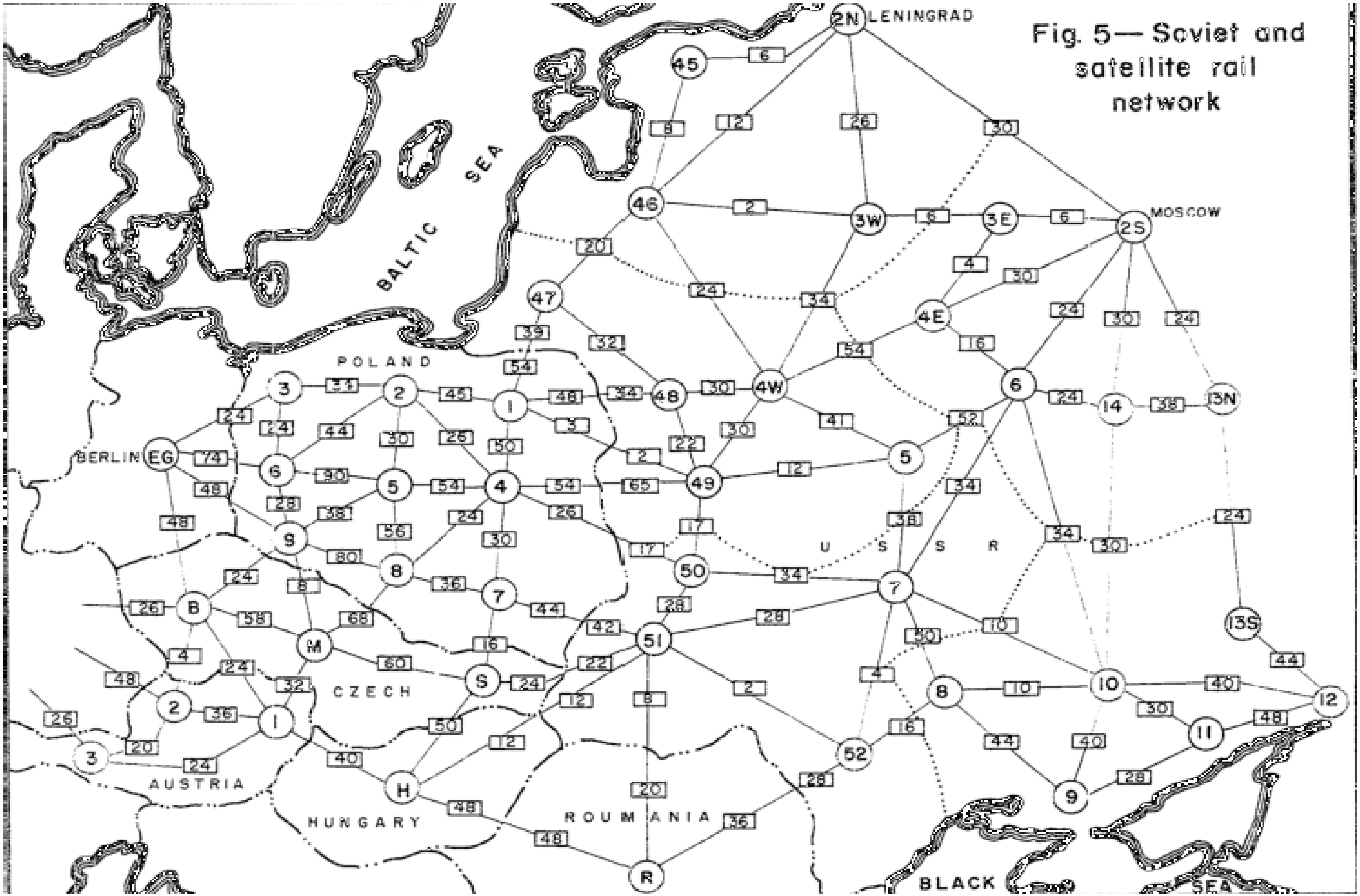
Anz. Schleifendurchläufe

- in jedem Durchlauf wird f um ≥ 1 vergrößert
- max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c \colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$: erweitern...
3. $\mathbb{R}_{>0}$: problematisch!

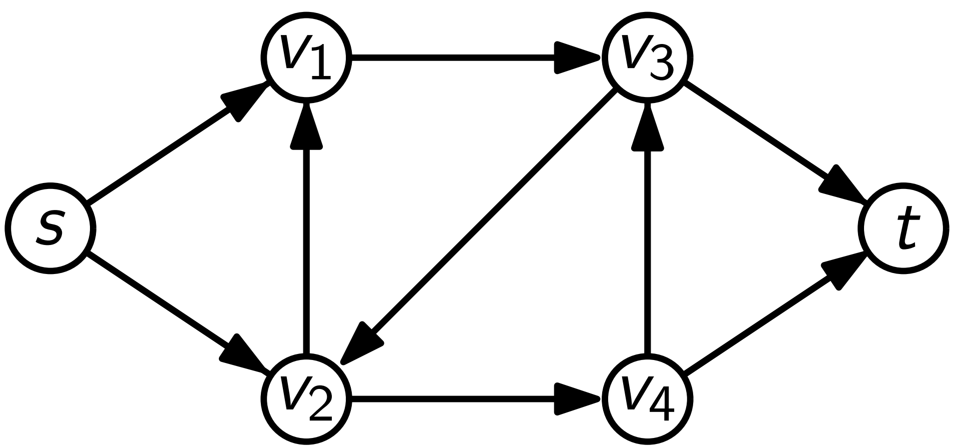
Flüsse und Schnitte



Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

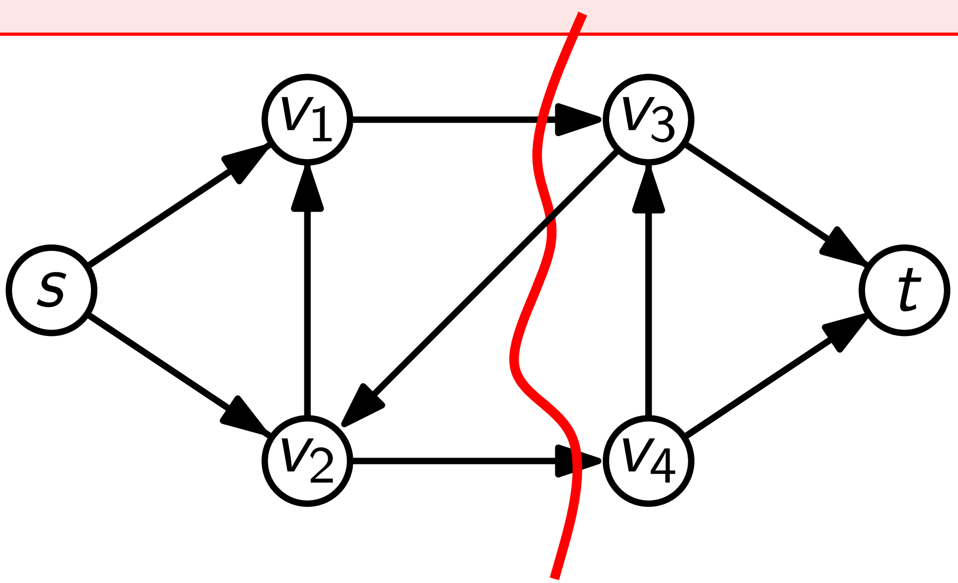
Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.



Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

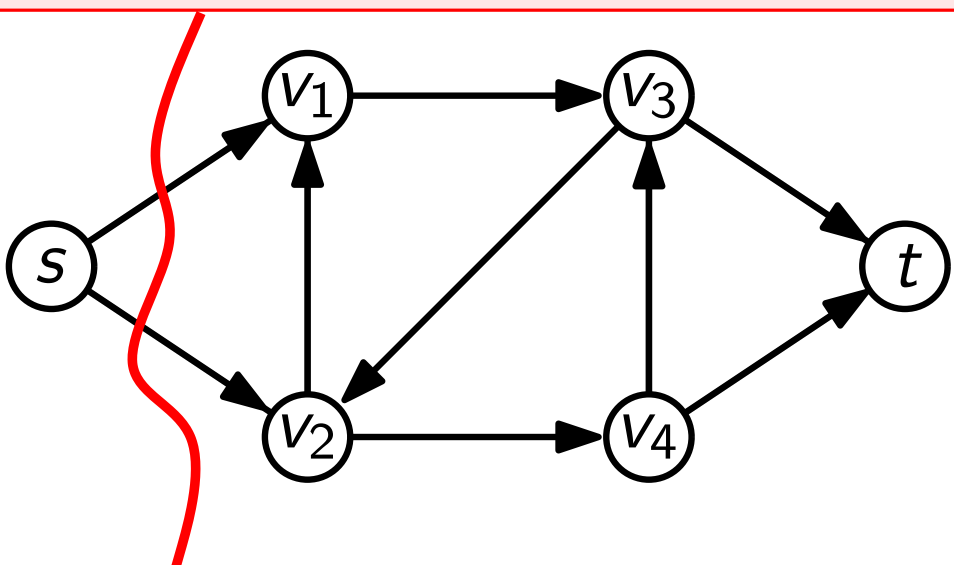
Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot{\cup} T$ mit $s \in S$, $t \in T$.



Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

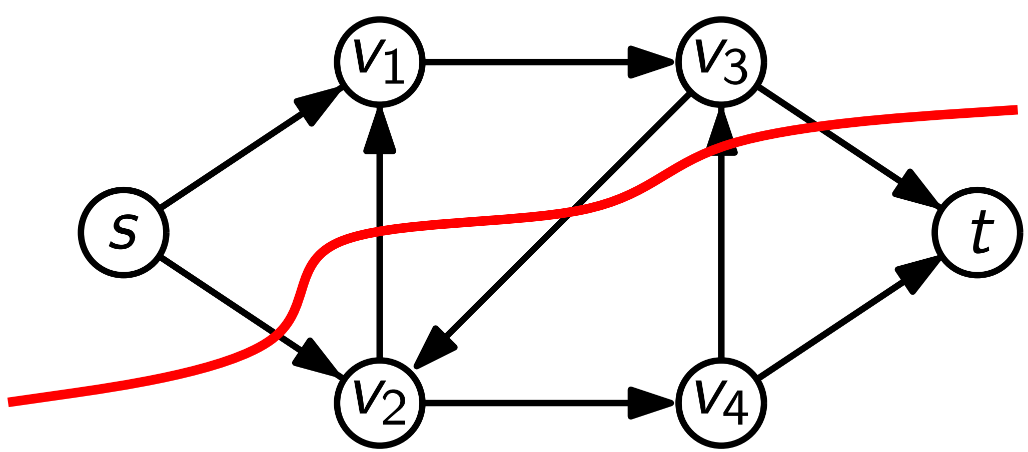
Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot{\cup} T$ mit $s \in S$, $t \in T$.



Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot{\cup} T$ mit $s \in S$, $t \in T$.

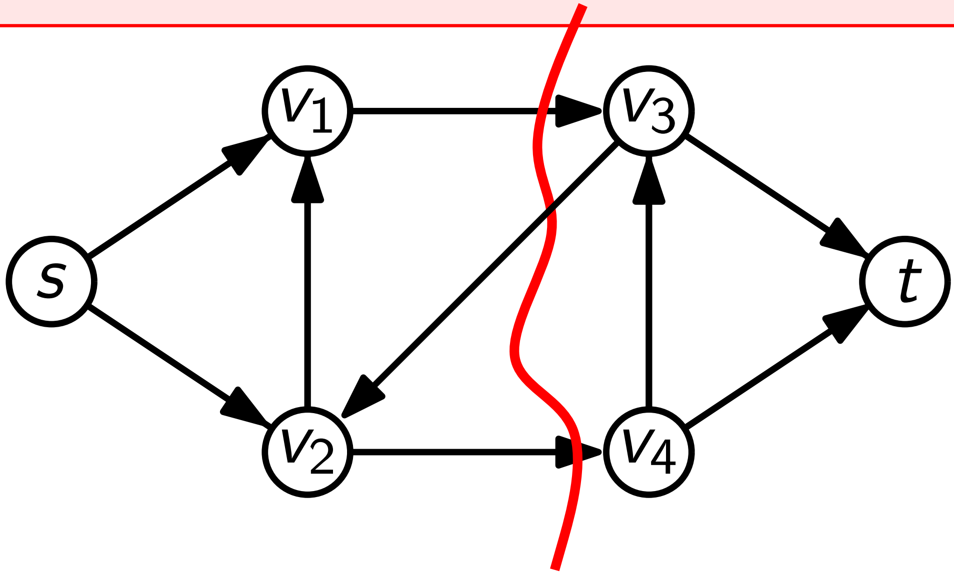


Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$

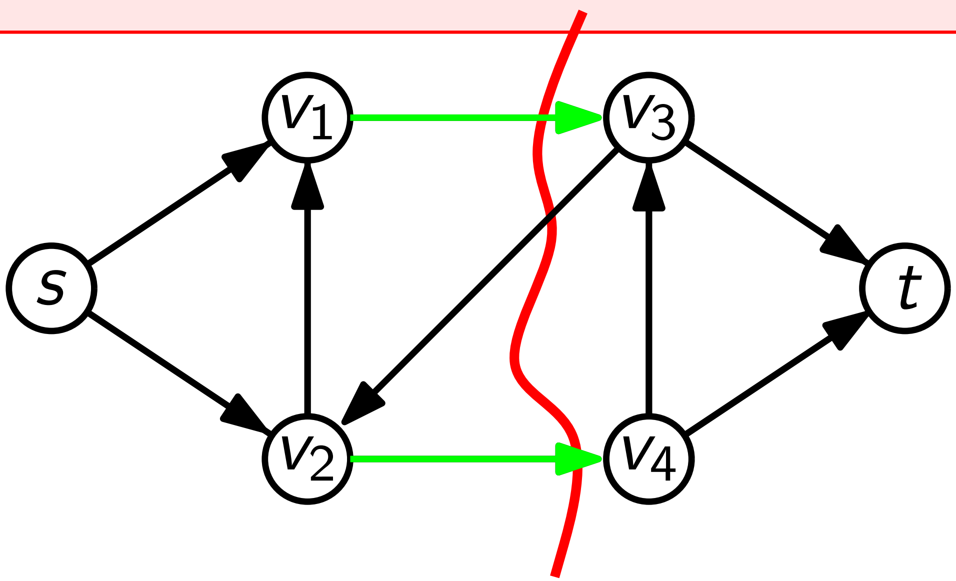


Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$

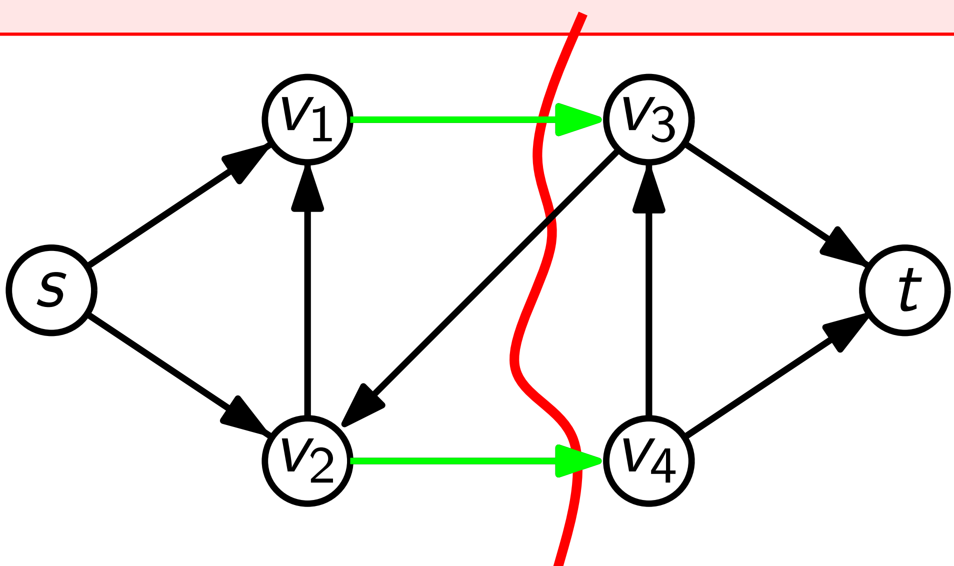


Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$



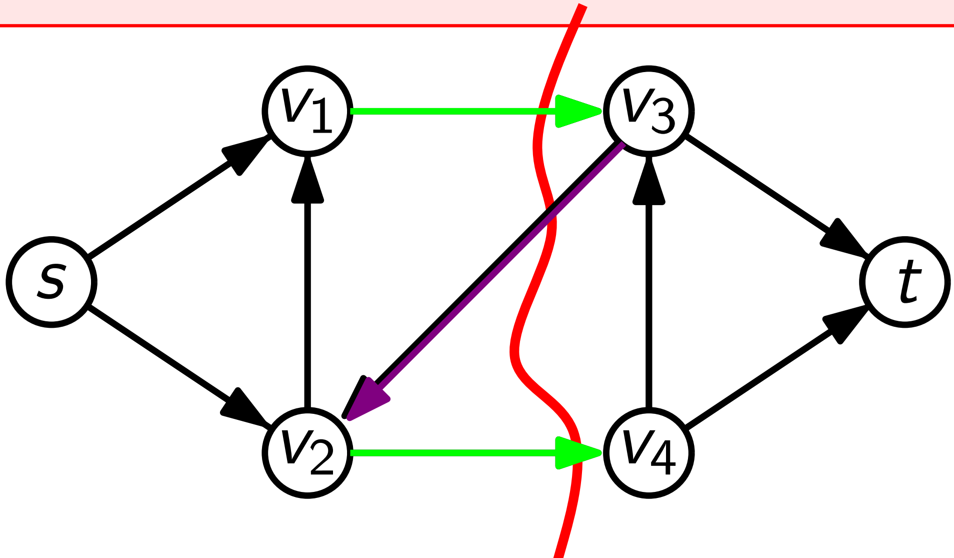
Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$

$\delta^-(S) = \{(u, v) \in E \mid u \in T, v \in S\}$



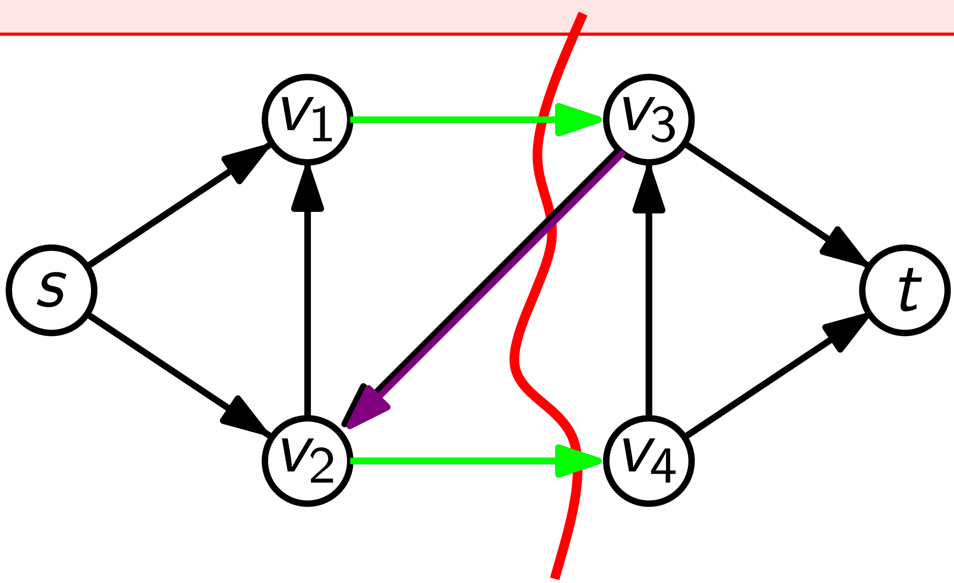
Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$

$\delta^-(S) = \{(u, v) \in E \mid u \in T, v \in S\}$



Schnitte

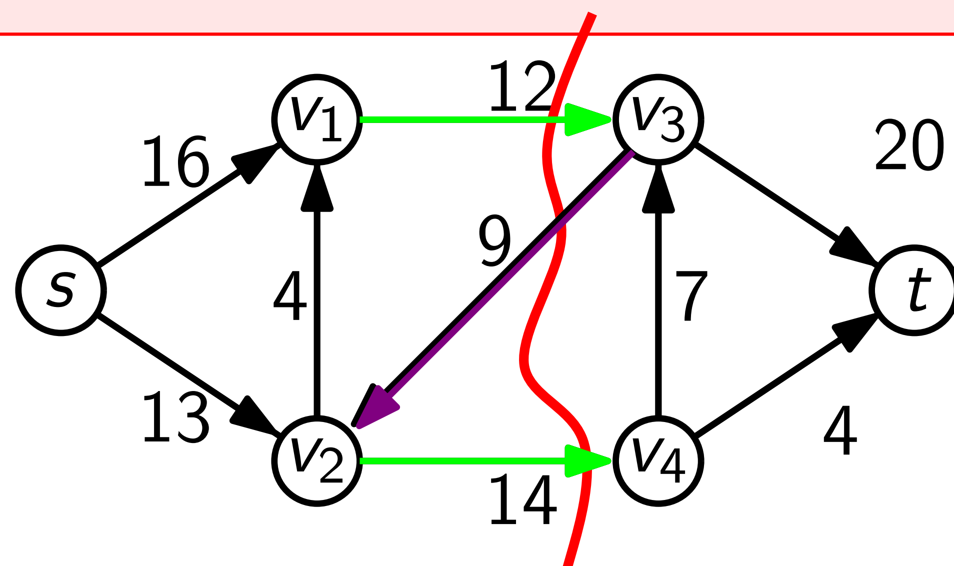
Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot\cup T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$

$\delta^-(S) = \{(u, v) \in E \mid u \in T, v \in S\}$

Seien $c(e)$ für $e \in E$ Kantenkapazitäten.
Die **Kapazität** von Schnitt (S, T) ist $c(\delta^+(S)) := \sum_{e \in \delta^+(S)} c(e)$



Schnitte

Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$.

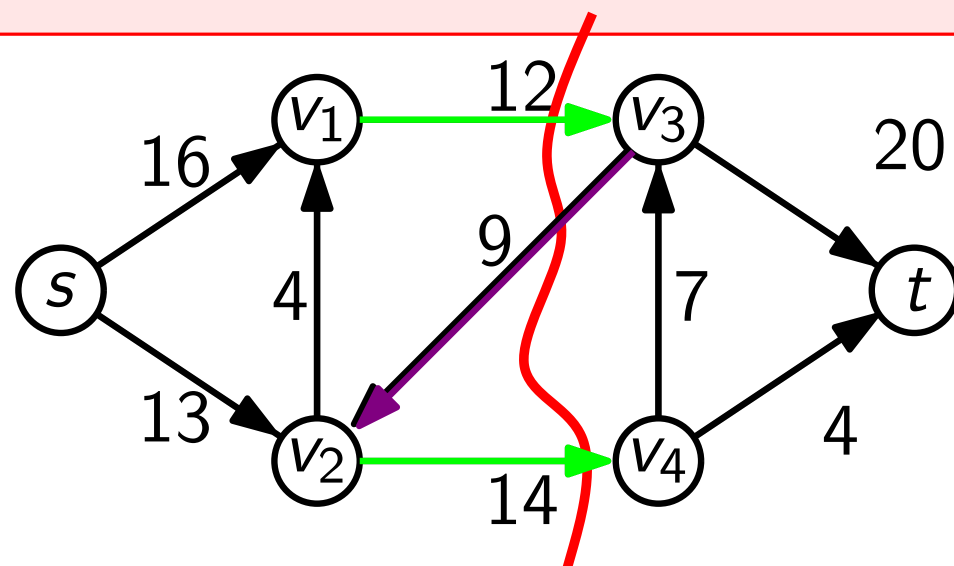
Ein s-t-**Schnitt** ist eine Zerlegung (S, T) der Knotenmenge $V = S \dot{\cup} T$ mit $s \in S$, $t \in T$.

$\delta^+(S) := \{(u, v) \in E \mid u \in S, v \in T\}$

$\delta^-(S) = \{(u, v) \in E \mid u \in T, v \in S\}$

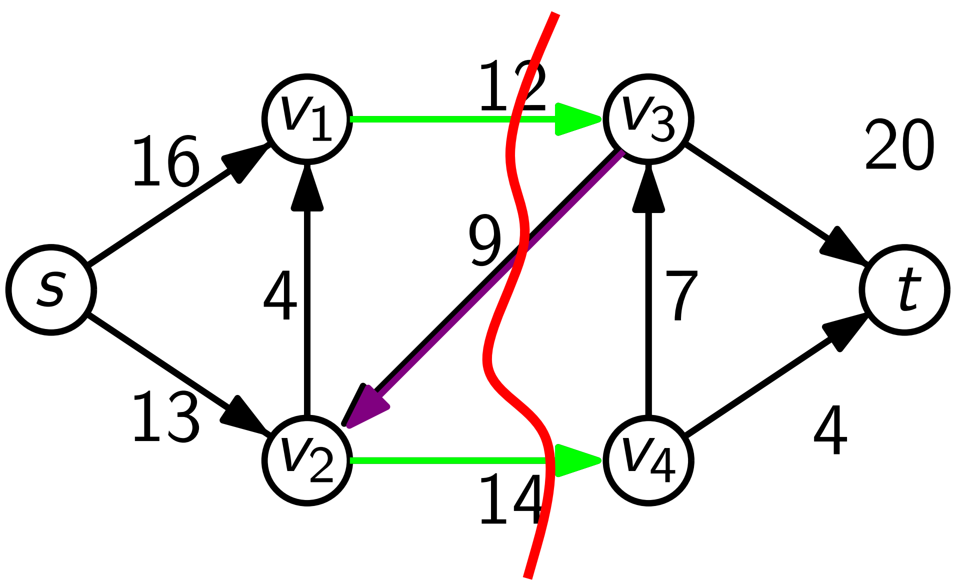
Seien $c(e)$ für $e \in E$ Kantenkapazitäten.
Die **Kapazität** von Schnitt (S, T) ist $c(\delta^+(S)) := \sum_{e \in \delta^+(S)} c(e)$

$c(\delta^+(S)) = 26$



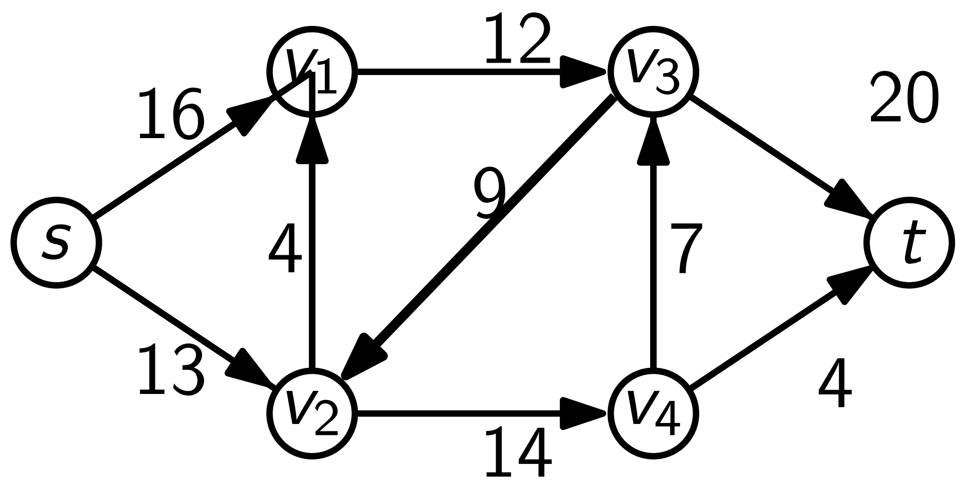
Optimierungsproblem 'Minimaler Schnitt'

Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.



Zusammenhang Flüsse und Schnitte

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

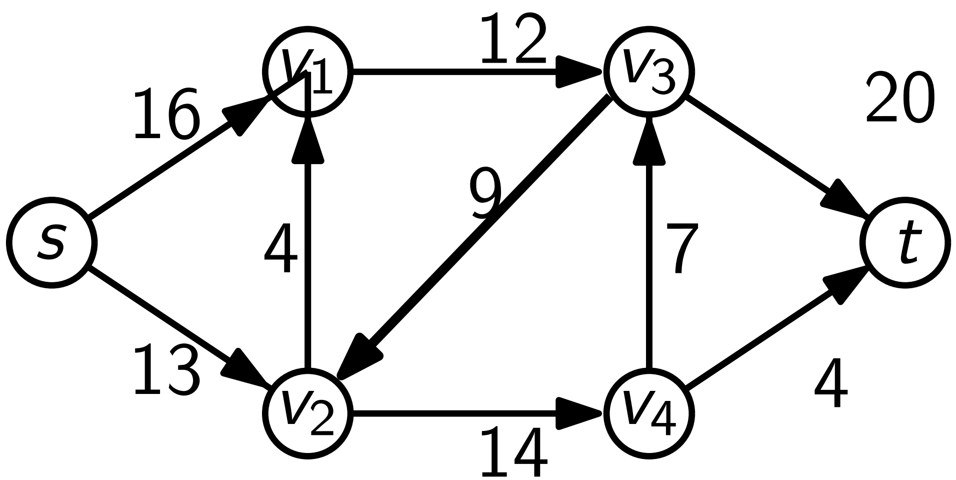


Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Zusammenhang Flüsse und Schnitte

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Was haben diese beiden Optimierungsproblem miteinander zu tun?

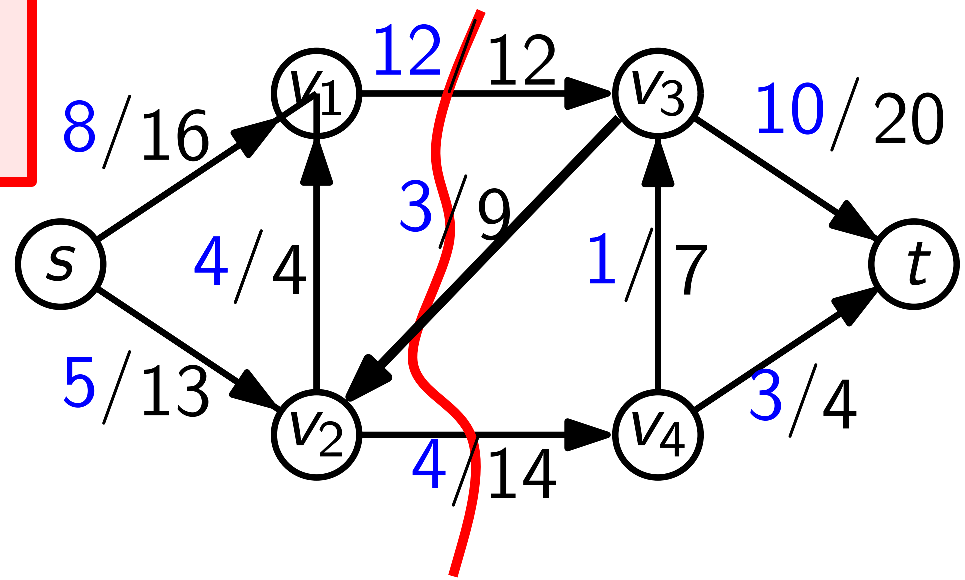


Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Zusammenhang Flüsse und Schnitte

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Lemma: Ist f ein s-t-Fluss und (S, T) ein s-t-Schnitt, so gilt:
$|f| = f(\delta^+(S)) - f(\delta^-(S))$



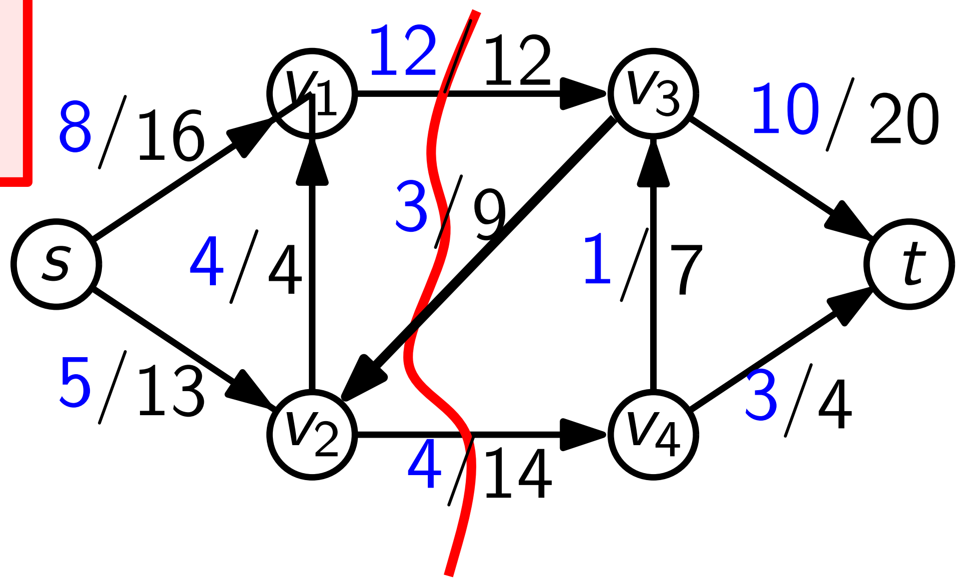
Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Zusammenhang Flüsse und Schnitte

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Lemma: Ist f ein s-t-Fluss und (S, T) ein s-t-Schnitt, so gilt:
$|f| = f(\delta^+(S)) - f(\delta^-(S))$

Intuitiv klar?



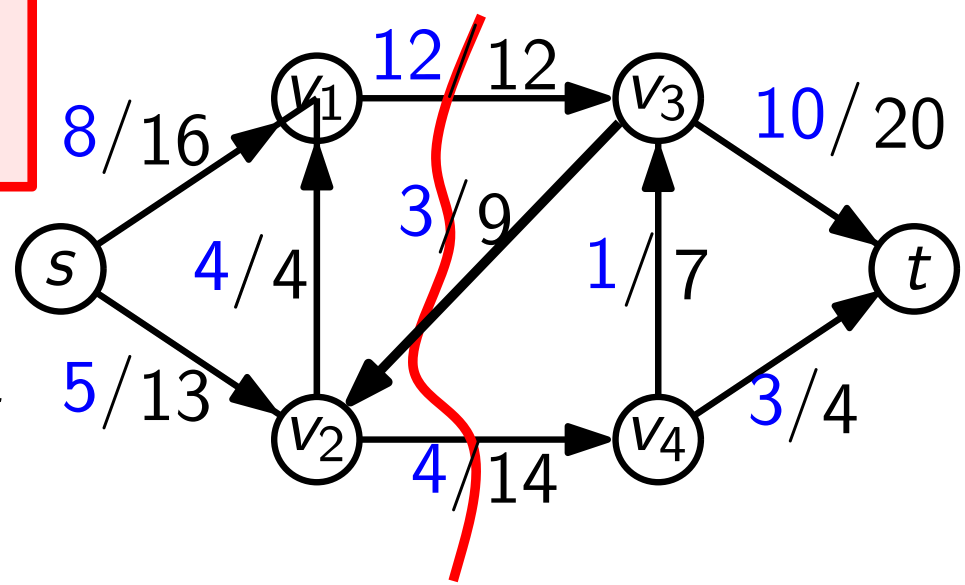
Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c \colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Zusammenhang Flüsse und Schnitte

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Lemma: Ist f ein s-t-Fluss und (S, T) ein s-t-Schnitt, so gilt:
$|f| = f(\delta^+(S)) - f(\delta^-(S))$

Beweis: $|f| = \mathit{Nettozufluss}_f(t)$
$= \sum_{v \in T} \mathit{Nettozufluss}_f(v)$
$=$
$\sum_{v \in T} \left(\sum_{e \in \delta^-(v)} f(e) - \sum_{e \in \delta^+(v)} f \right.$
$= \sum_{e \in \delta^-(T)} f(e) - \sum_{e \in \delta^+(T)} f(e)$
$= \sum_{e \in \delta^+(S)} f(e) - \sum_{e \in \delta^-(S)} f(e)$

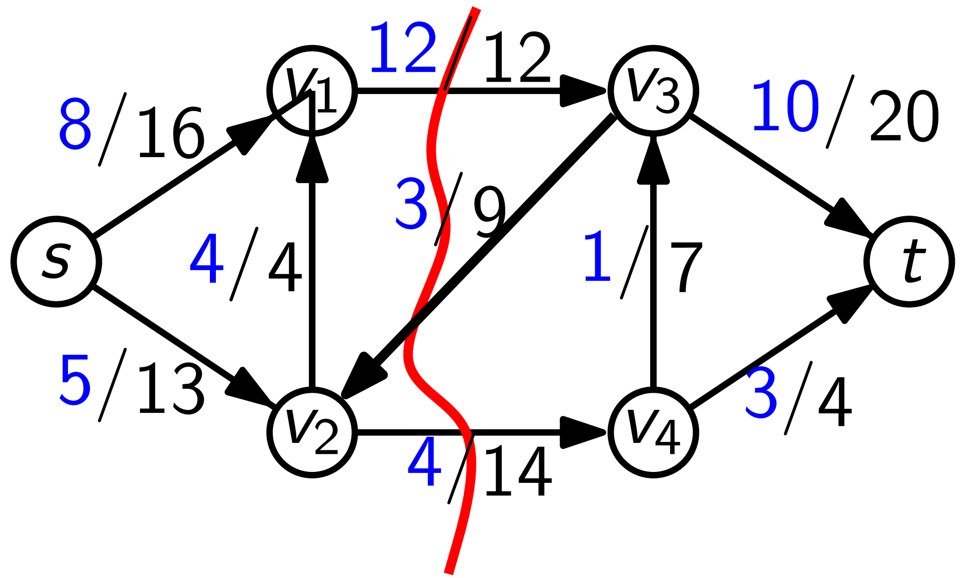


Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Zusammenhang Flüsse und Schnitte

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Korollar: Sei f ein zulässiger s-t-Fluss und (S, T) ein s-t-Schnitt.
Dann ist $|f| \leq c(\delta^+(S))$.



Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$,

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$,

Dann gilt: $T := V \setminus S = \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ nicht erreichbar}\} \ni t$, denn sonst gäbe es s-t-Weg in G_f und also flussvergrößernden Weg in G.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$,

Dann gilt: $T := V \setminus S = \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ nicht erreichbar}\} \ni t$, denn sonst gäbe es s-t-Weg in G_f und also flussvergrößernden Weg in G.

Also ist (S, T) ein s-t-Schnitt.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v$ ist in G_f von s erreichbar$\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$:

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$

(sonst gäbe es Kante (u, v) in G_f, Endknoten v wäre also über u in G_f erreichbar: Widerspruch zu $v \in T = V \setminus S$)

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$:

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$

(sonst gäbe es Kante (v, u) in G_f, u wäre also über v in G_f erreichbar: Widerspruch zu $u \in T = V \setminus S$)

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v$ ist in G_f von s erreichbar$\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$
$c(\delta^+(S)) = \sum_{e \in \delta^+(S)} c(e)$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$
$c(\delta^+(S)) = \sum_{e \in \delta^+(S)} c(e) = \sum_{e \in \delta^+(S)} f(e)$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$

$$c(\delta^+(S)) = \sum_{e \in \delta^+(S)} c(e) = \sum_{e \in \delta^+(S)} f(e)$$
$$= \sum_{e \in \delta^+(S)} f(e) - \underbrace{\sum_{e \in \delta^-(S)} f(e)}_{0}$$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$': Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Betrachte $(u, v) \in \delta^+(S)$: es gilt $f((u, v)) = c((u, v))$
Betrachte $(u, v) \in \delta^-(S)$: es gilt $f((u, v)) = 0$

$$\begin{aligned} c(\delta^+(S)) = \textstyle\sum_{e \in \delta^+(S)} c(e) &= \textstyle\sum_{e \in \delta^+(S)} f(e) \\ &= \textstyle\sum_{e \in \delta^+(S)} f(e) - \underbrace{\sum_{e \in \delta^-(S)} f(e)}_{0} \\ &= |f^*| \end{aligned}$$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Dann gilt $c(\delta^+(S)) = |f|$

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v \text{ ist in } G_f \text{ von } s \text{ erreichbar}\}$, (S, T) ist s-t-Schnitt.
Dann gilt $c(\delta^+(S)) = |f|$
Wegen $|f^*| \leq c(\delta^+) \leq |f| \leq |f^*|$ für einen maximalen Fluss f^* folgt:

Der Satz vom flussvergrößernden Weg - Beweis

$G = (V, E)$ Graph mit Kap. $c(e)$ für $e \in E$, $s, t \in V$, und f s-t-Fluss auf G, G_f Residualgraph

Satz vom flussvergrößernden Weg
Ein zulässiger s-t-Fluss f in G ist maximal
$\Leftrightarrow$ es gibt keinen flussvergrößernden Weg in G_f.

Beweis.

"$\Rightarrow$ ✓

'$\Leftarrow$':Sei f ein Fluss, für den es keinen flussvergrößernden Weg in G_f gibt.
Sei $S := \{v \in V : v$ ist in G_f von s erreichbar$\}$, (S, T) ist s-t-Schnitt.
Dann gilt $c(\delta^+(S)) = |f|$
Wegen $|f^*| \leq c(\delta^+) \leq |f| \leq |f^*|$ für einen maximalen Fluss f^* folgt:
$|f| = |f^*|$, also ist f maximal.

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f
Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg W **do**
  $\Delta_W = \min_{(u,v) \in W} c_f((u,v))$
  for $(u, v) \in W$ **do**
    if $(u, v) \in E$ **then**
      $f_{(u,v)} = f_{(u,v)} + \Delta_W$
    else
      $f_{(v,u)} = f_{(v,u)} - \Delta_W$
    end if
  end for
  Erstelle Residualgraph G_f
end while
return f

Erstelle G_f:

$O(|V| + |E|) = O(|E|)$

Berechnung von s-t-Wegen

– Breitensuche
– Tiefensuche
$\left.\right\}$ $O(|E|)$ Zeit

Anz. Schleifendurchläufe

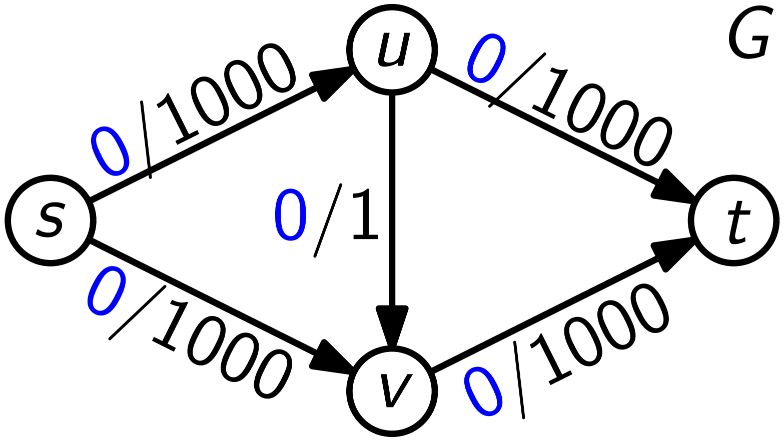
– in jedem Durchlauf wird f um ≥ 1 vergrößert

– max. $|f^\star|$ Durchläufe, wobei $f^\star$ ein max. Fluss

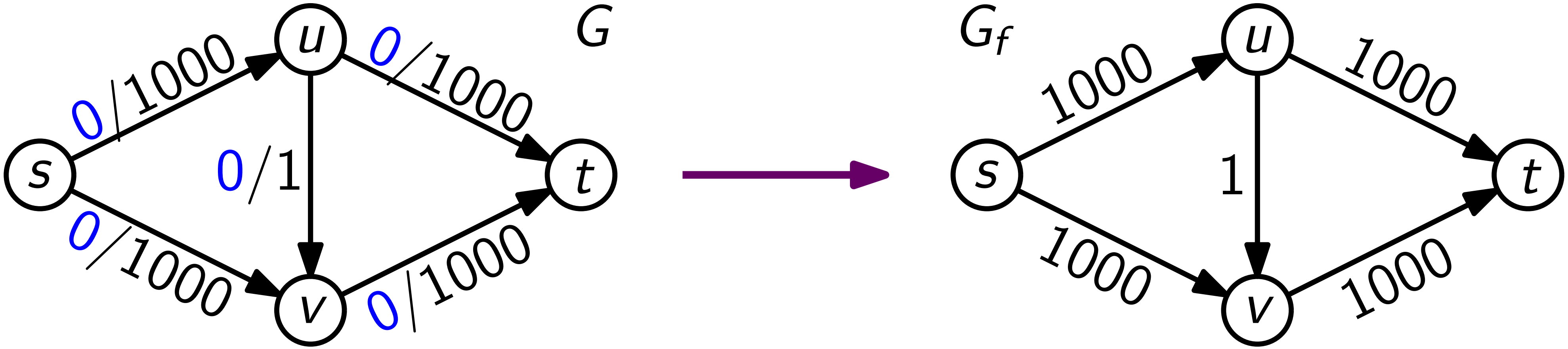
Korrektheit: Folgt, falls Algorithmus terminiert, direkt aus Satz vom flussvergrößernden Pfad

Laufzeit?
1. $c\colon E \to \mathbb{N}$: $O(|f^\star| \cdot |E|)$
2. $\mathbb{Q}_{>0}$: erweitern...
3. $\mathbb{R}_{>0}$: problematisch!

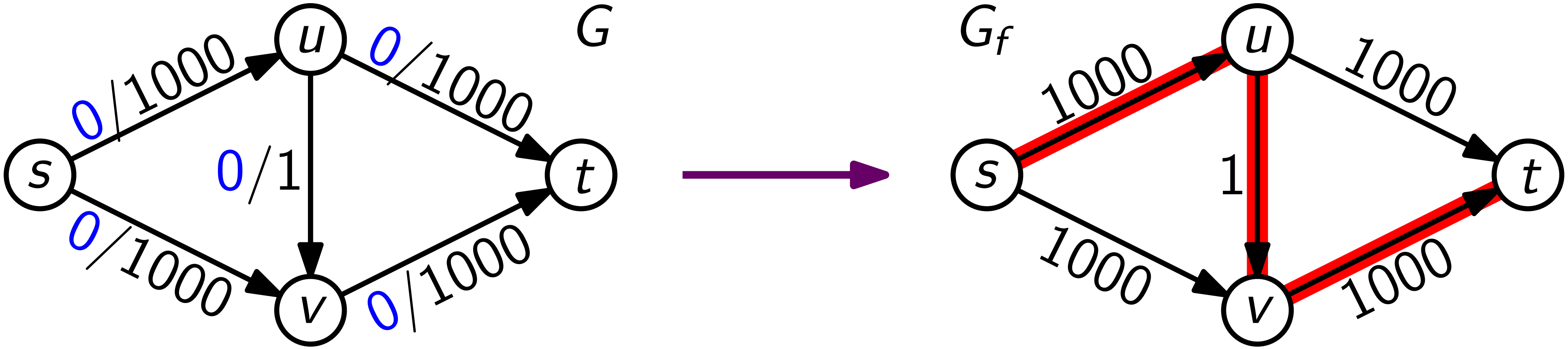
Ford-Fulkerson im Beispiel



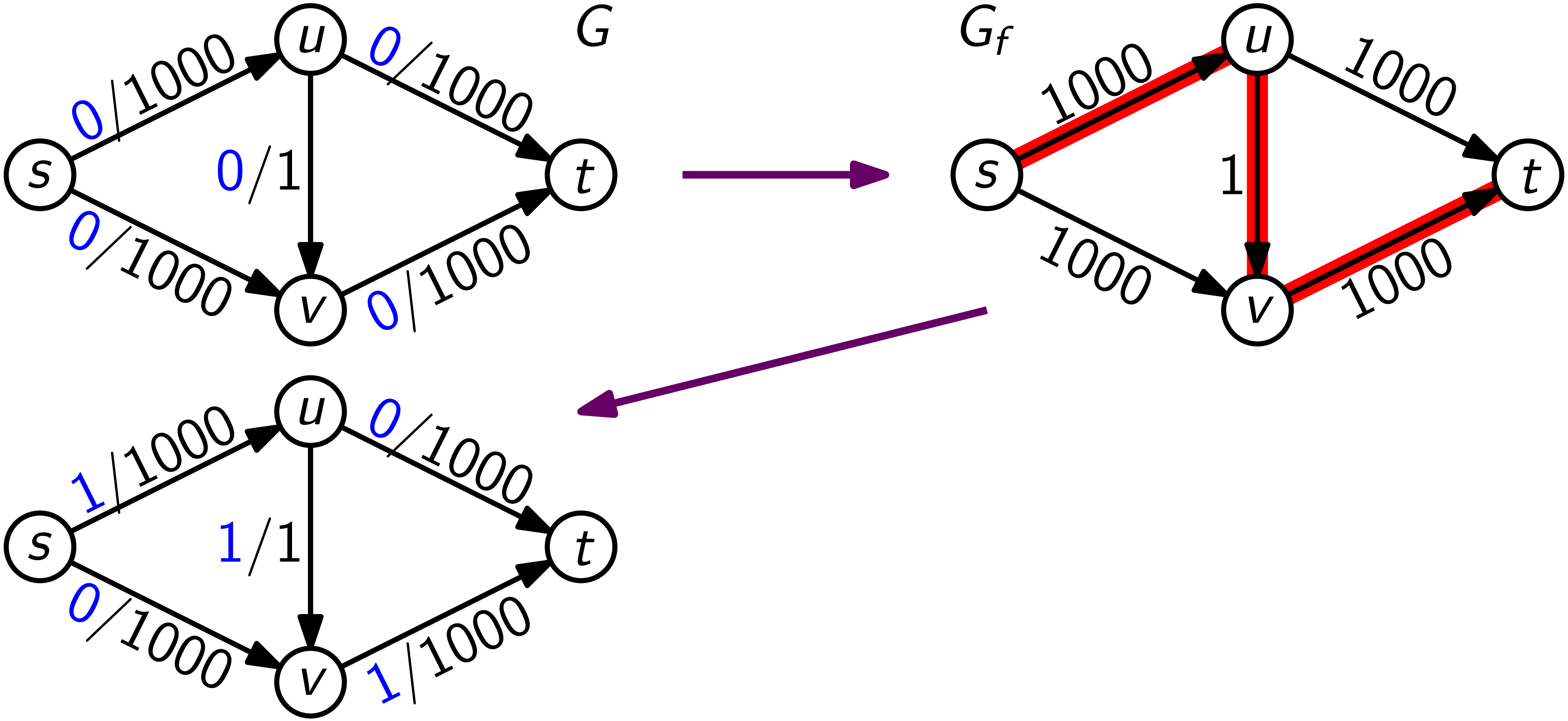
Ford-Fulkerson im Beispiel



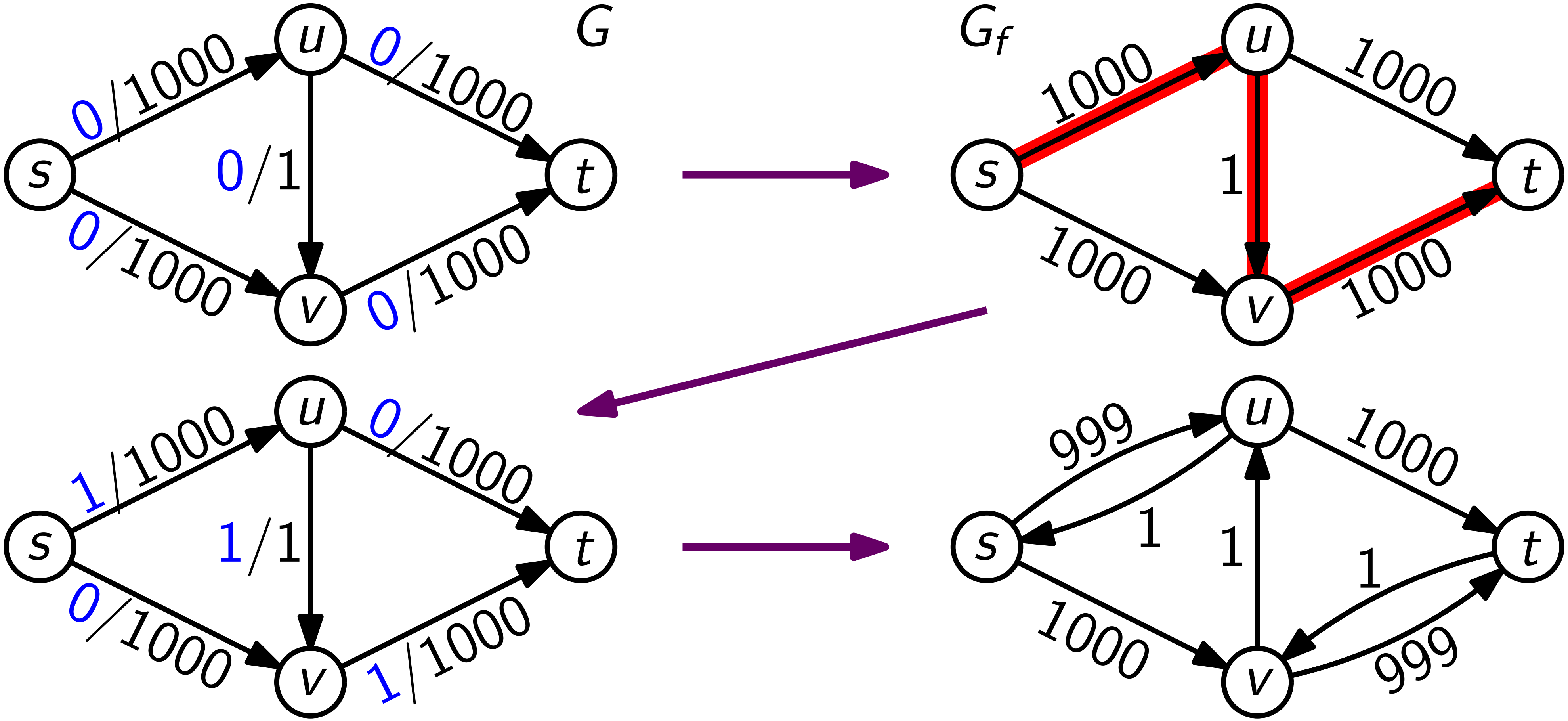
Ford-Fulkerson im Beispiel



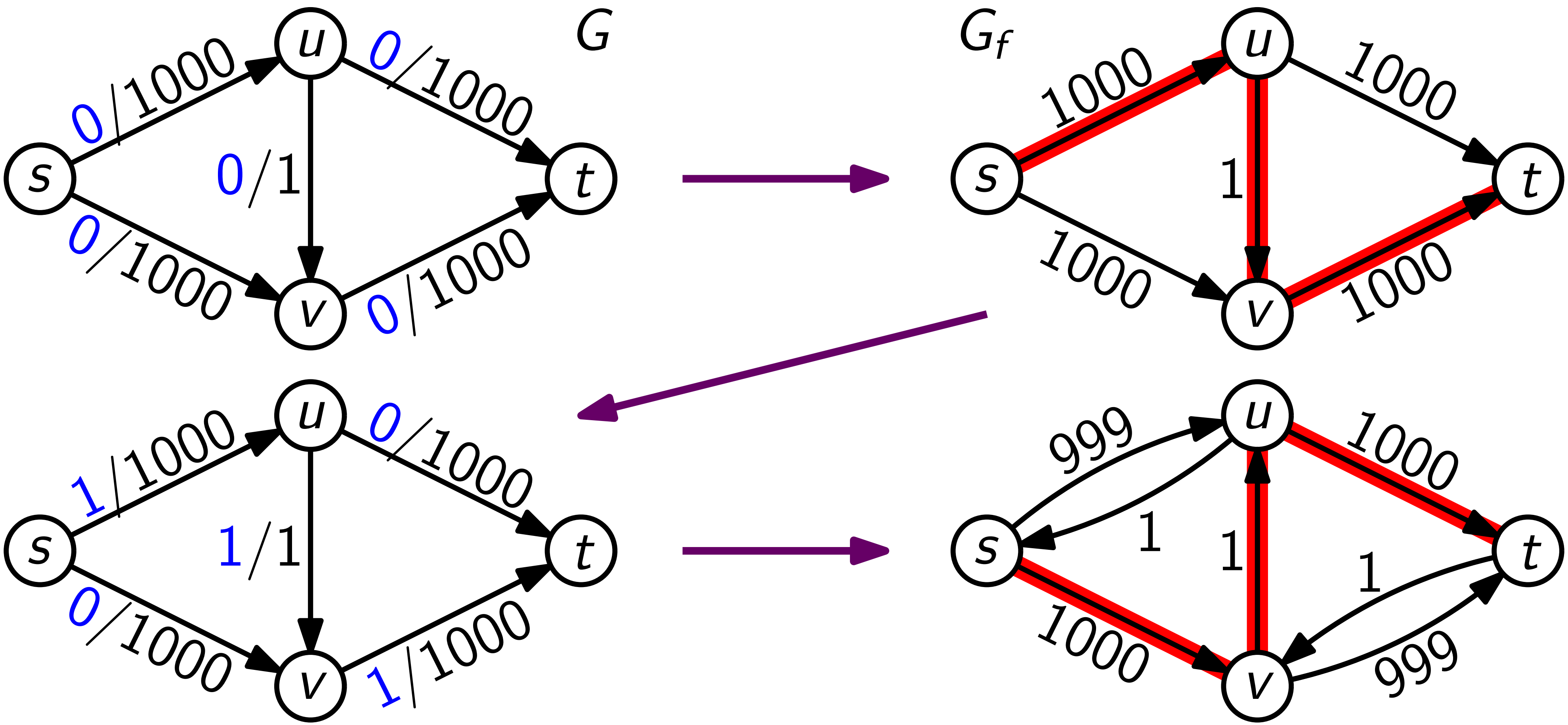
Ford-Fulkerson im Beispiel



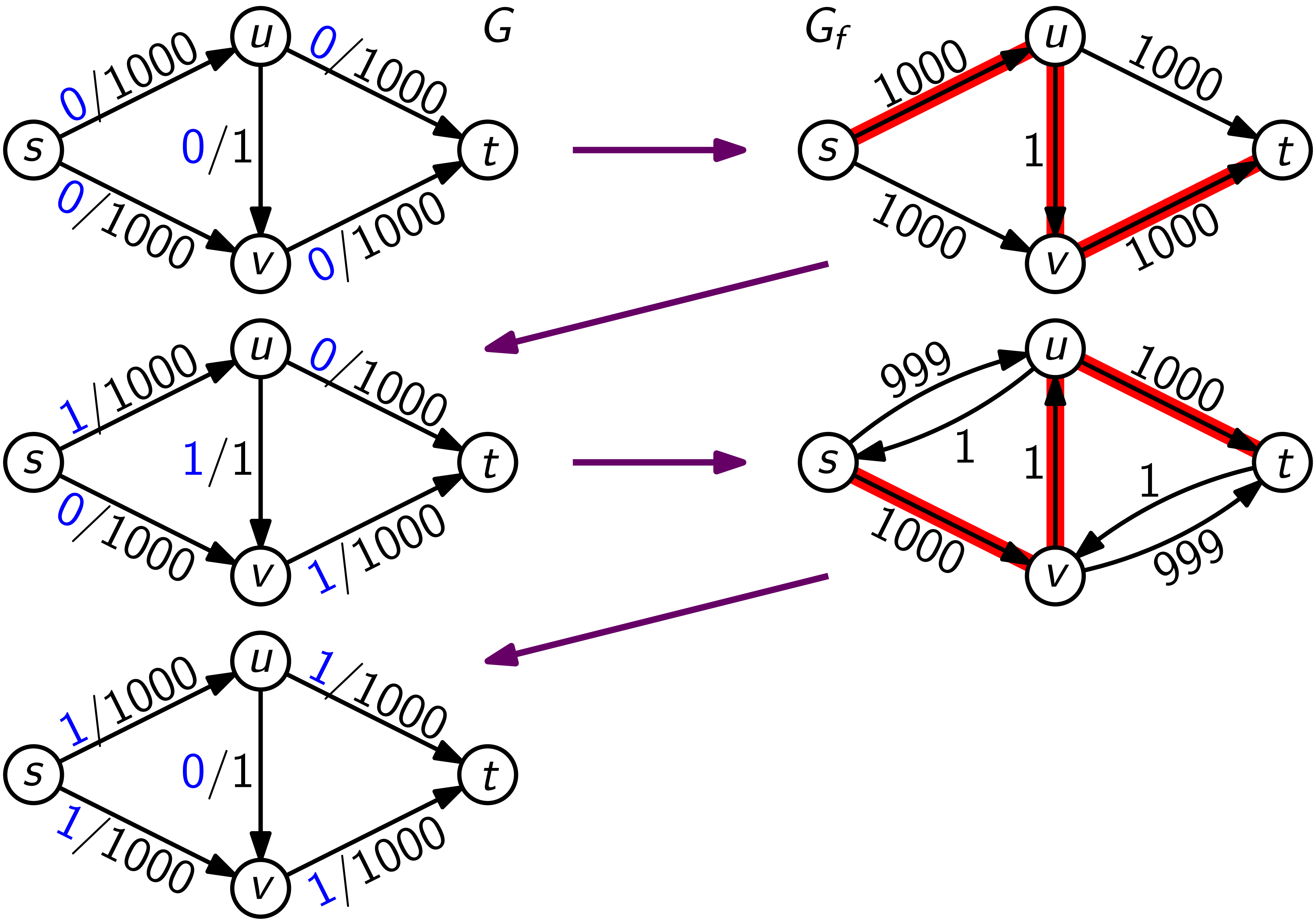
Ford-Fulkerson im Beispiel



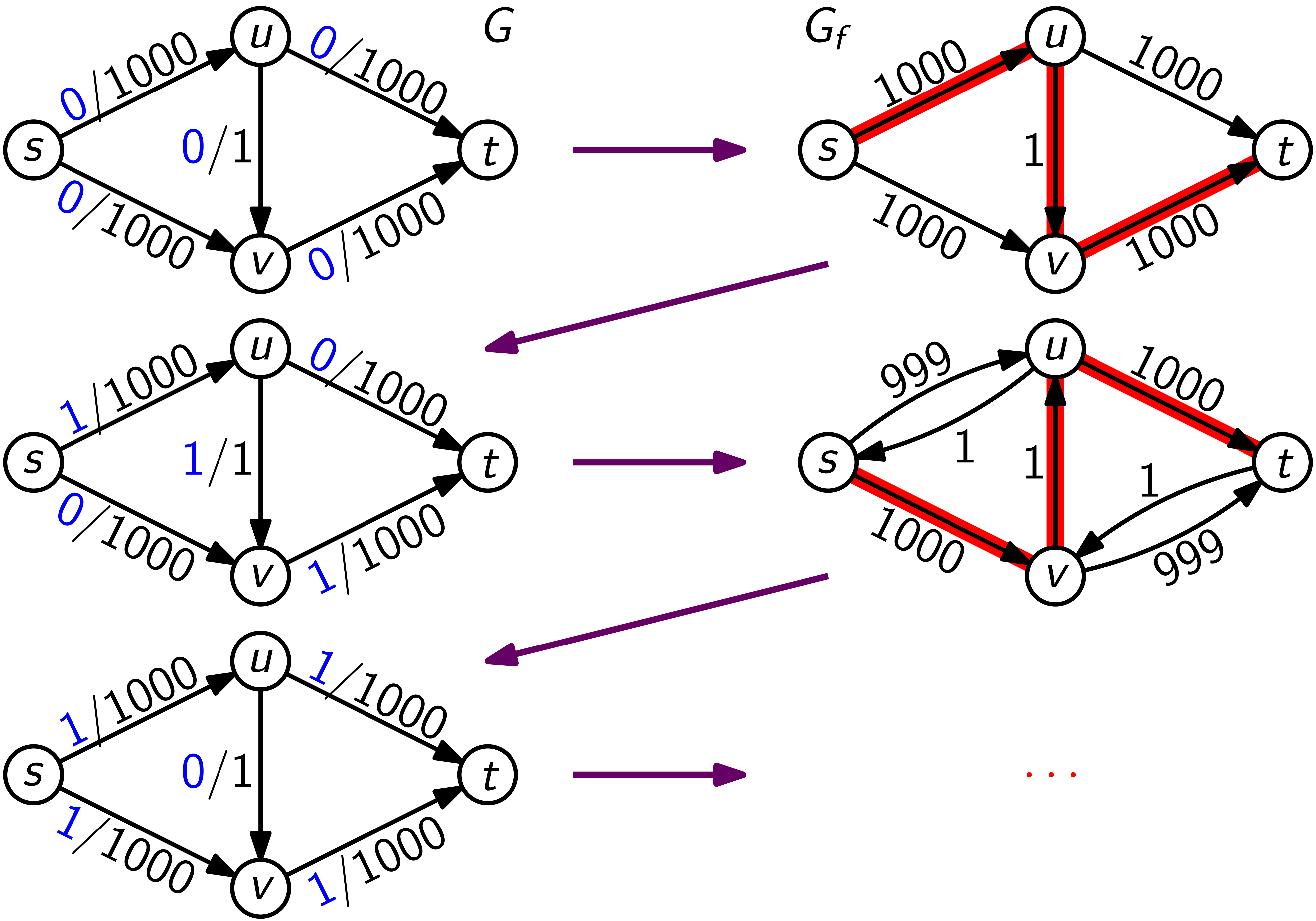
Ford-Fulkerson im Beispiel



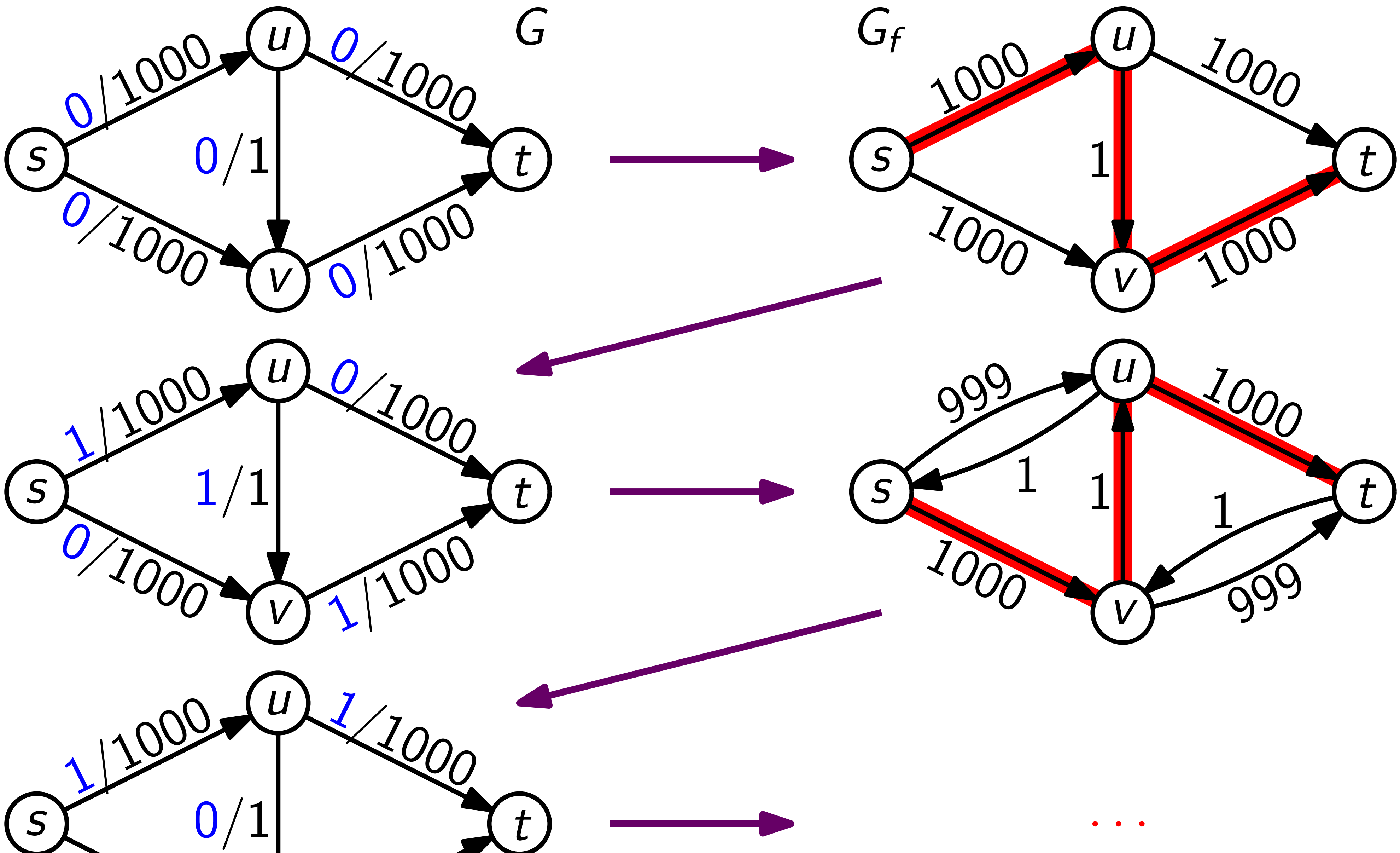
Ford-Fulkerson im Beispiel



Ford-Fulkerson im Beispiel



Ford-Fulkerson im Beispiel



Laufzeit $\in \Omega(|f^\star| \cdot E)$

Der Algorithmus von Ford & Fulkerson

Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

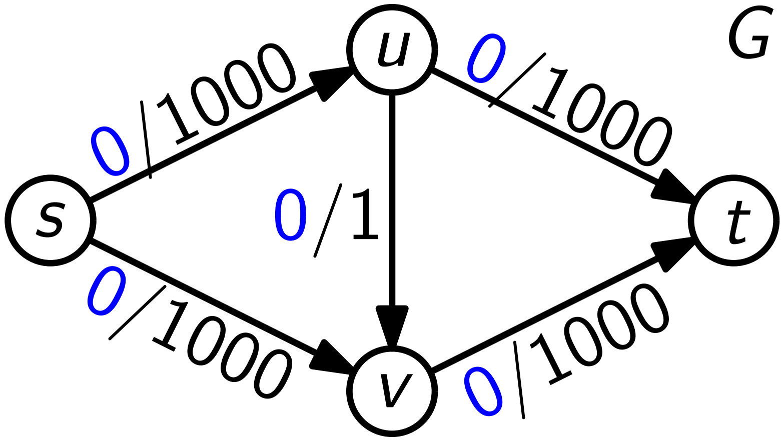
```
Initialisiere: f(e) = 0 für alle e ∈ E
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg do
    Bestimme s-t-Weg W
    Δ_W = min_{(u,v)∈W} c_f((u,v))
    for (u,v) ∈ W do
        if (u,v) ∈ E then
            f_(u,v) = f_(u,v) + Δ_W
        else
            f_(v,u) = f_(v,u) − Δ_W
        end if
    end for
    Erstelle Residualgraph G_f
end while
return f
```

Der Algorithmus von Edmonds & Karp

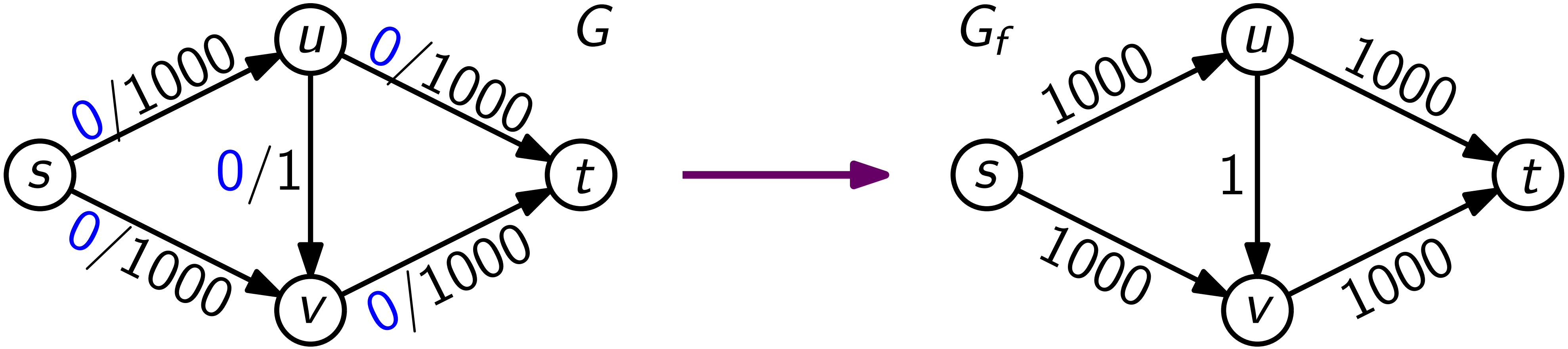
Require: Gerichteter, zusammenhängender Graph $G = (V, E)$ mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t
Ensure: s-t-Fluss f

Initialisiere: $f(e) = 0$ für alle $e \in E$
Erstelle Residualgraph G_f
while G_f enthält s-t-Weg **do**
  Bestimme s-t-Weg W dessen Länge minimal bezüglich Kantenanzahl ist
  $\Delta_W = \min_{(u,v)\in W} c_f((u,v))$
  for $(u,v) \in W$ **do**
    if $(u,v) \in E$ **then**
      $f_{(u,v)} = f_{(u,v)} + \Delta_W$
    else
      $f_{(v,u)} = f_{(v,u)} - \Delta_W$
    end if
  end for
  Erstelle Residualgraph G_f
end while
return f

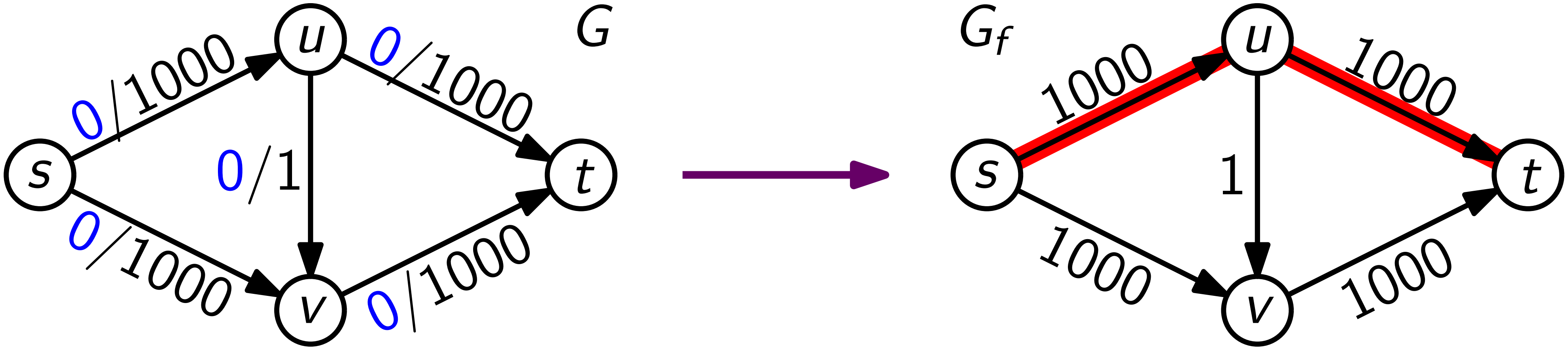
Edmonds-Karp im Beispiel



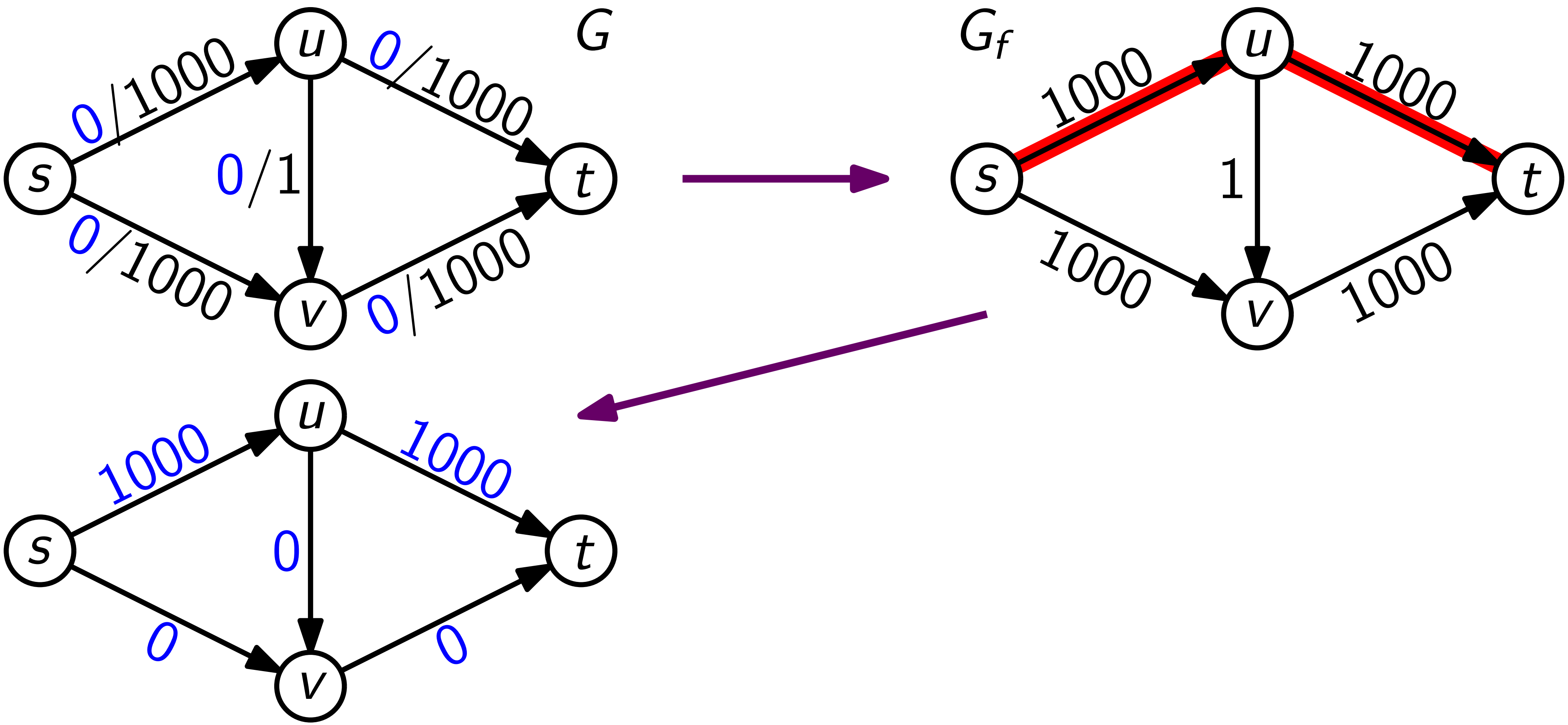
Edmonds-Karp im Beispiel



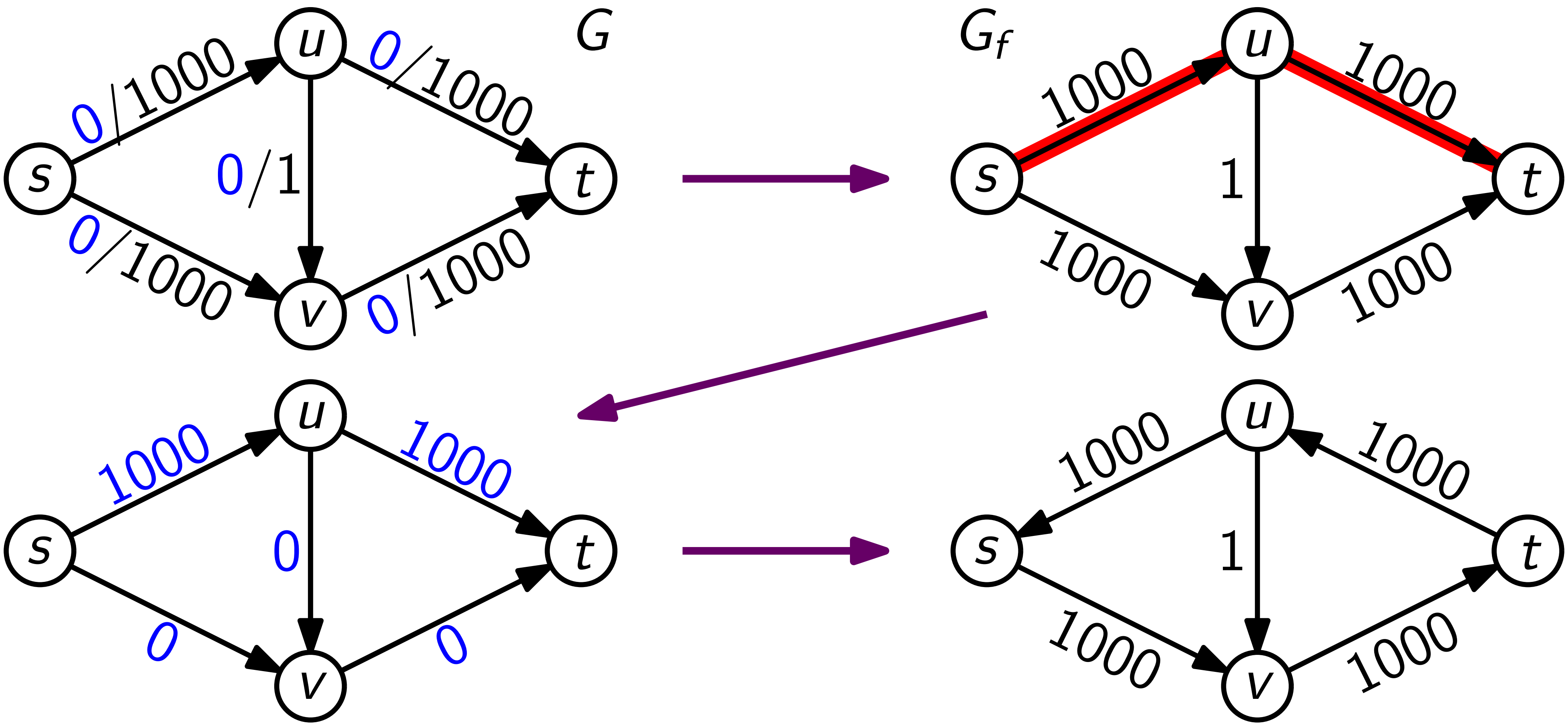
Edmonds-Karp im Beispiel



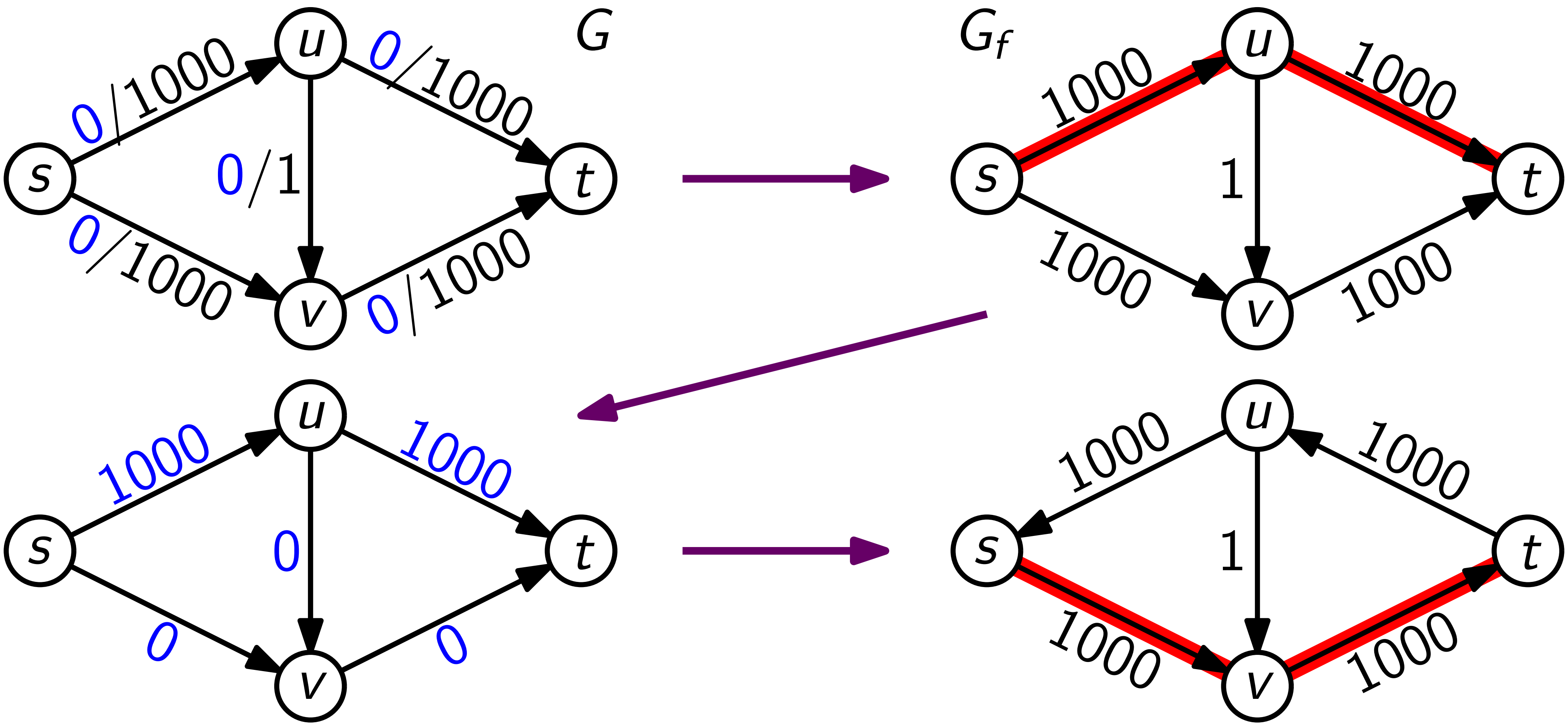
Edmonds-Karp im Beispiel



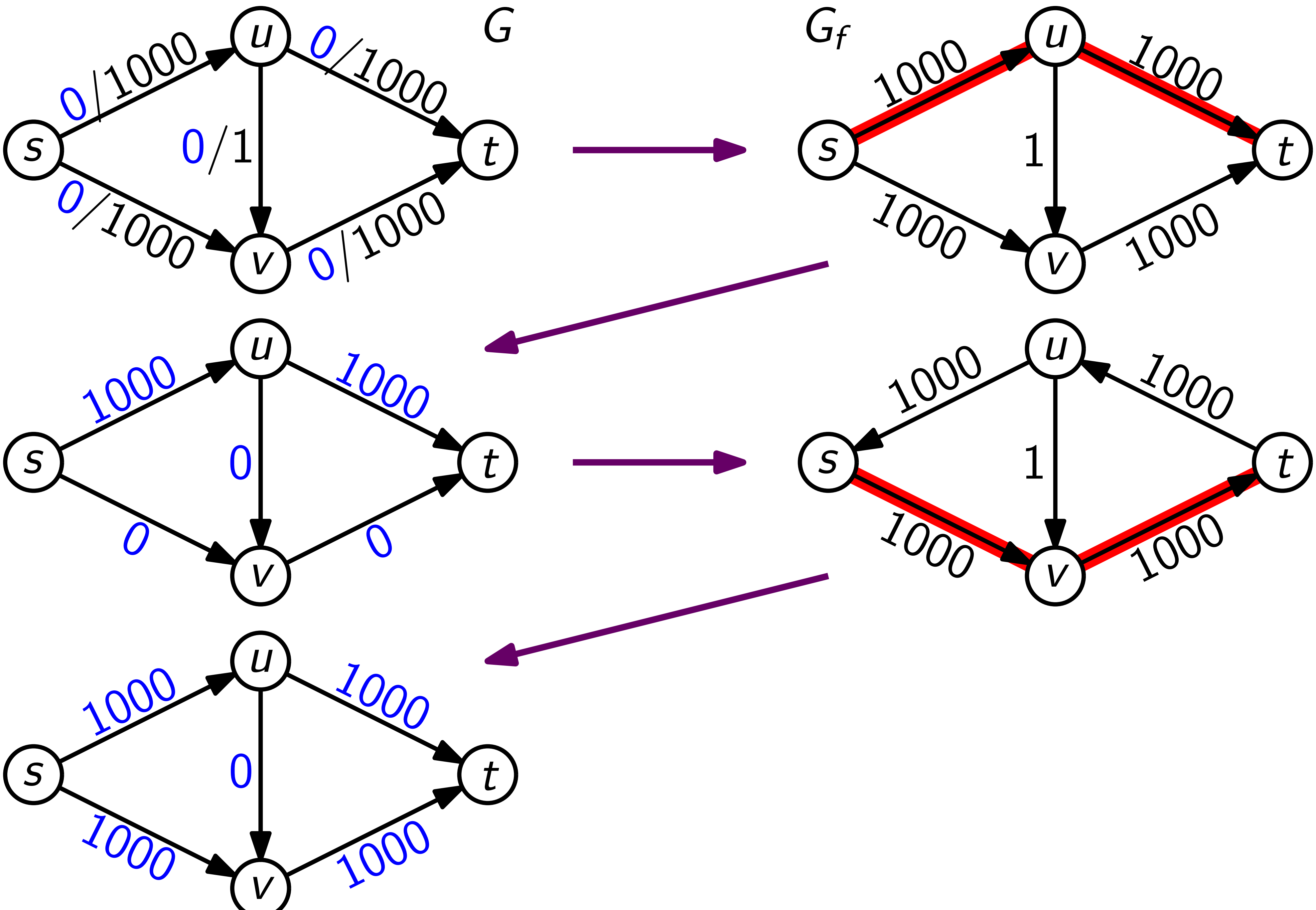
Edmonds-Karp im Beispiel



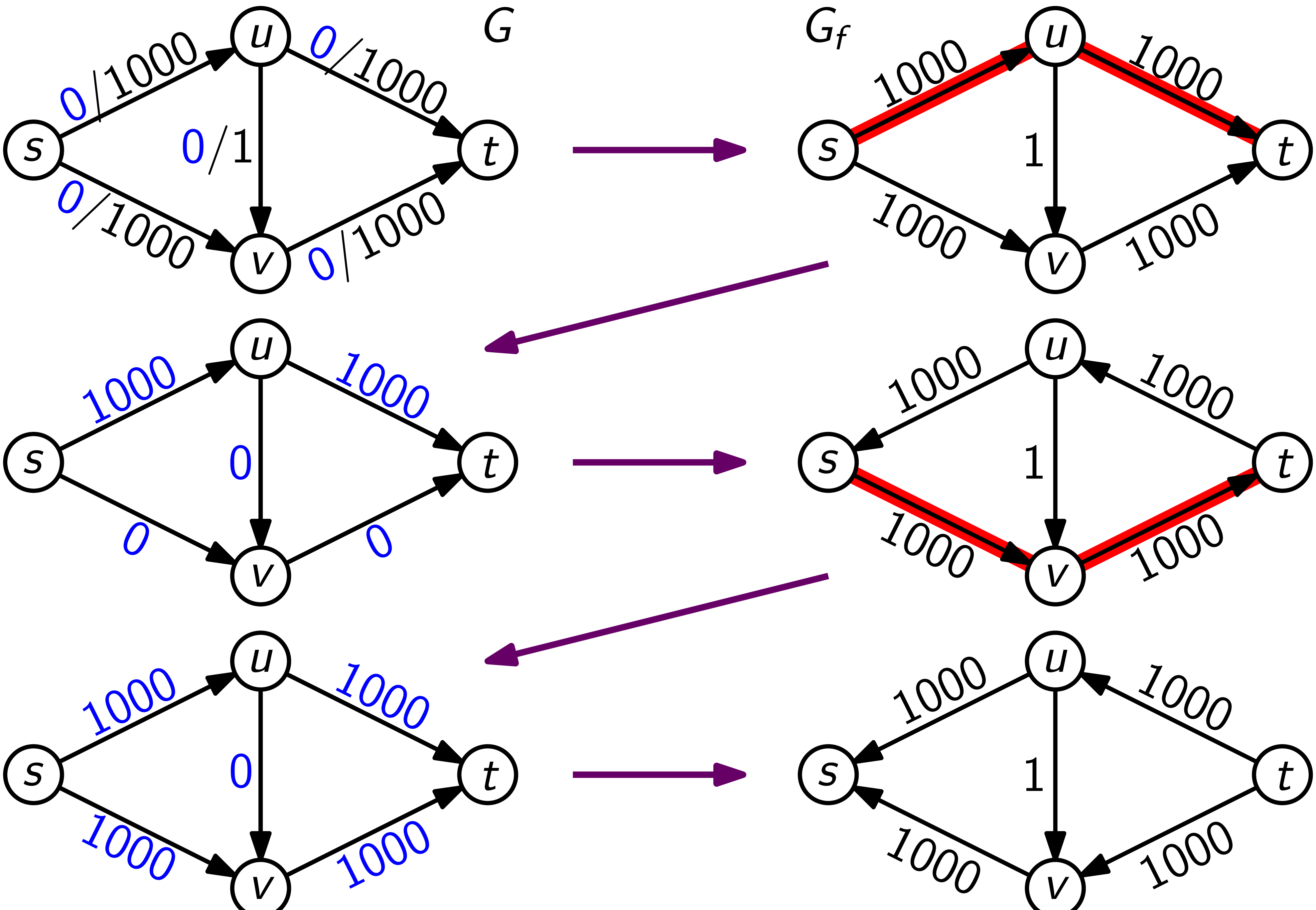
Edmonds-Karp im Beispiel



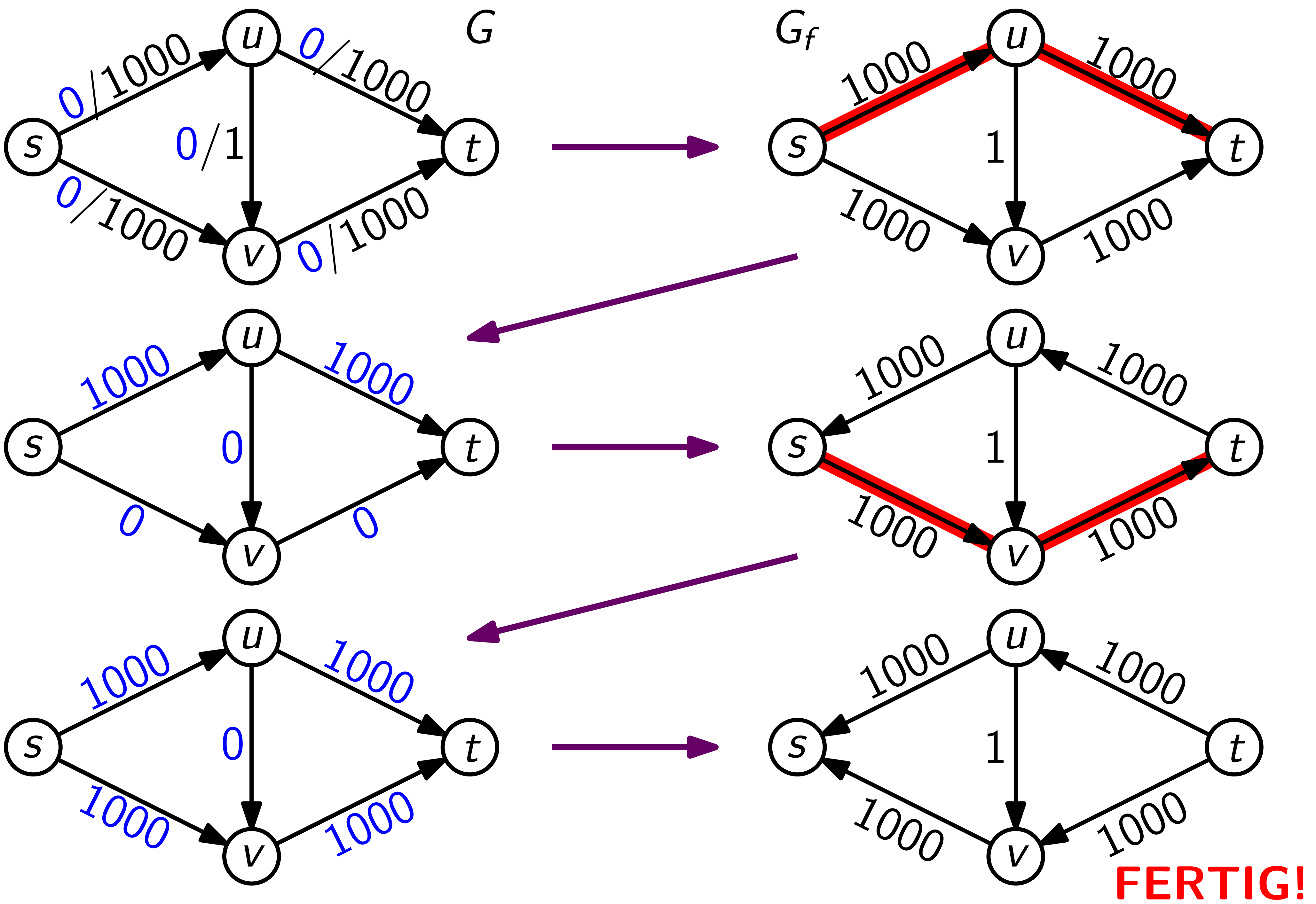
Edmonds-Karp im Beispiel



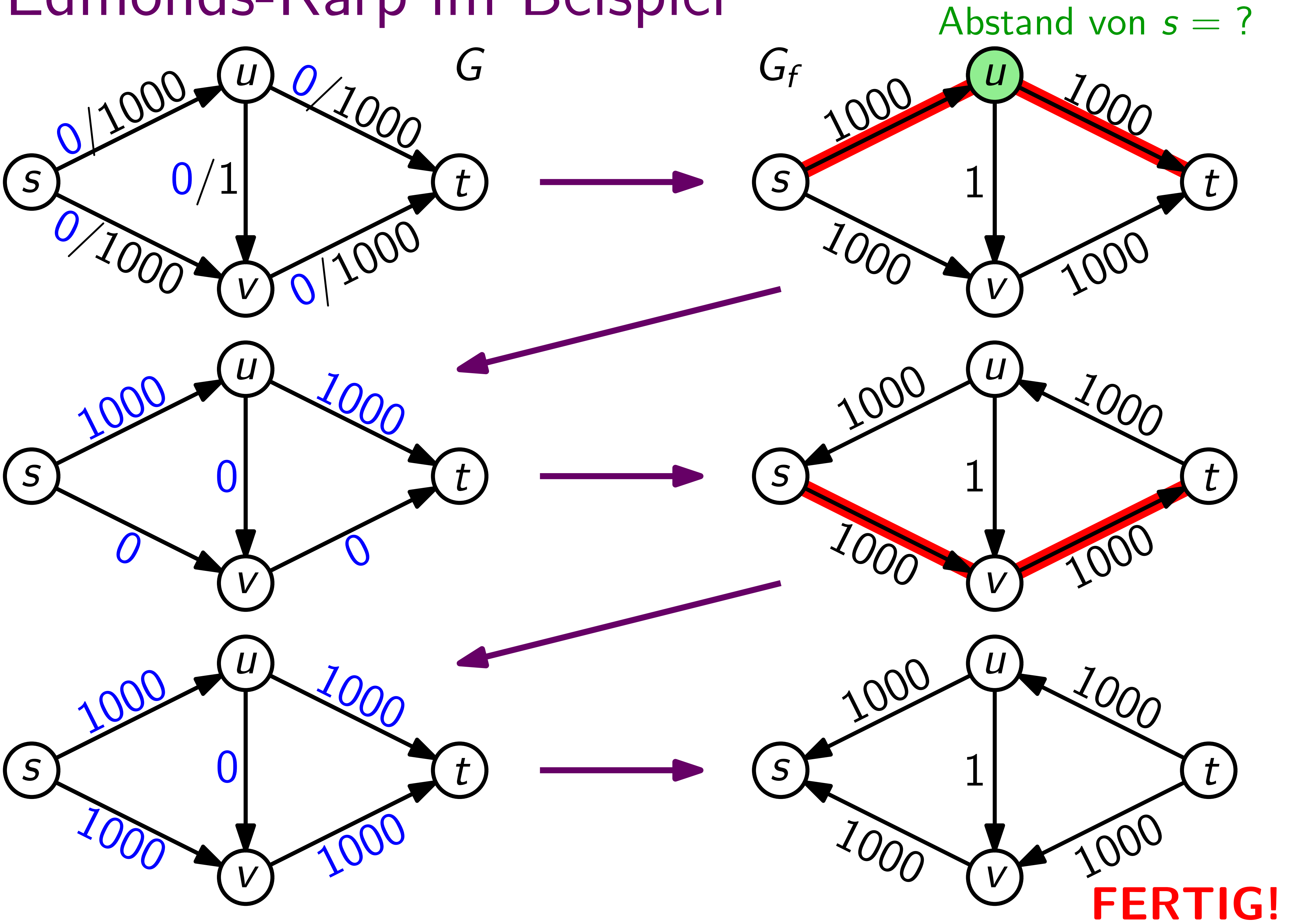
Edmonds-Karp im Beispiel



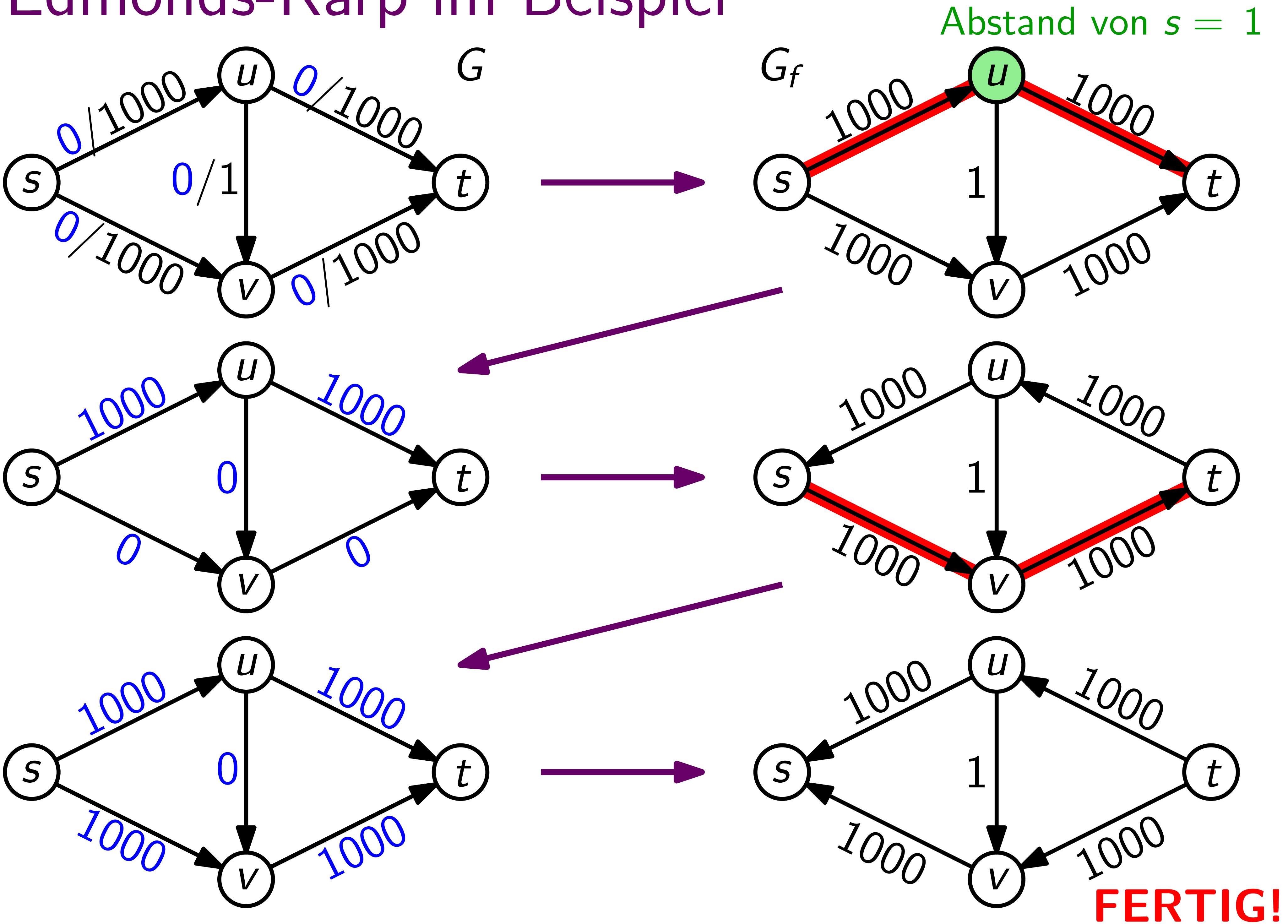
Edmonds-Karp im Beispiel



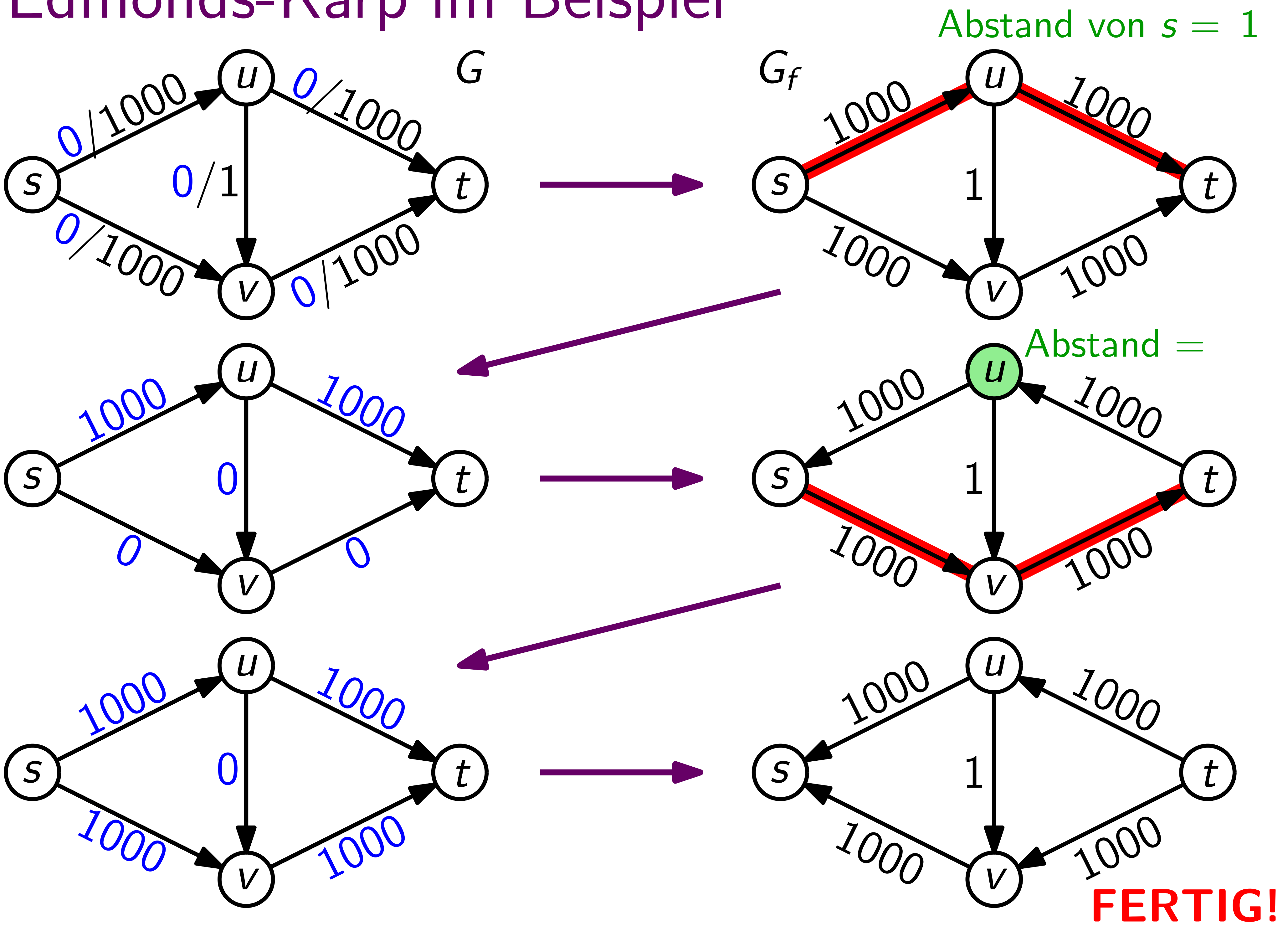
Edmonds-Karp im Beispiel



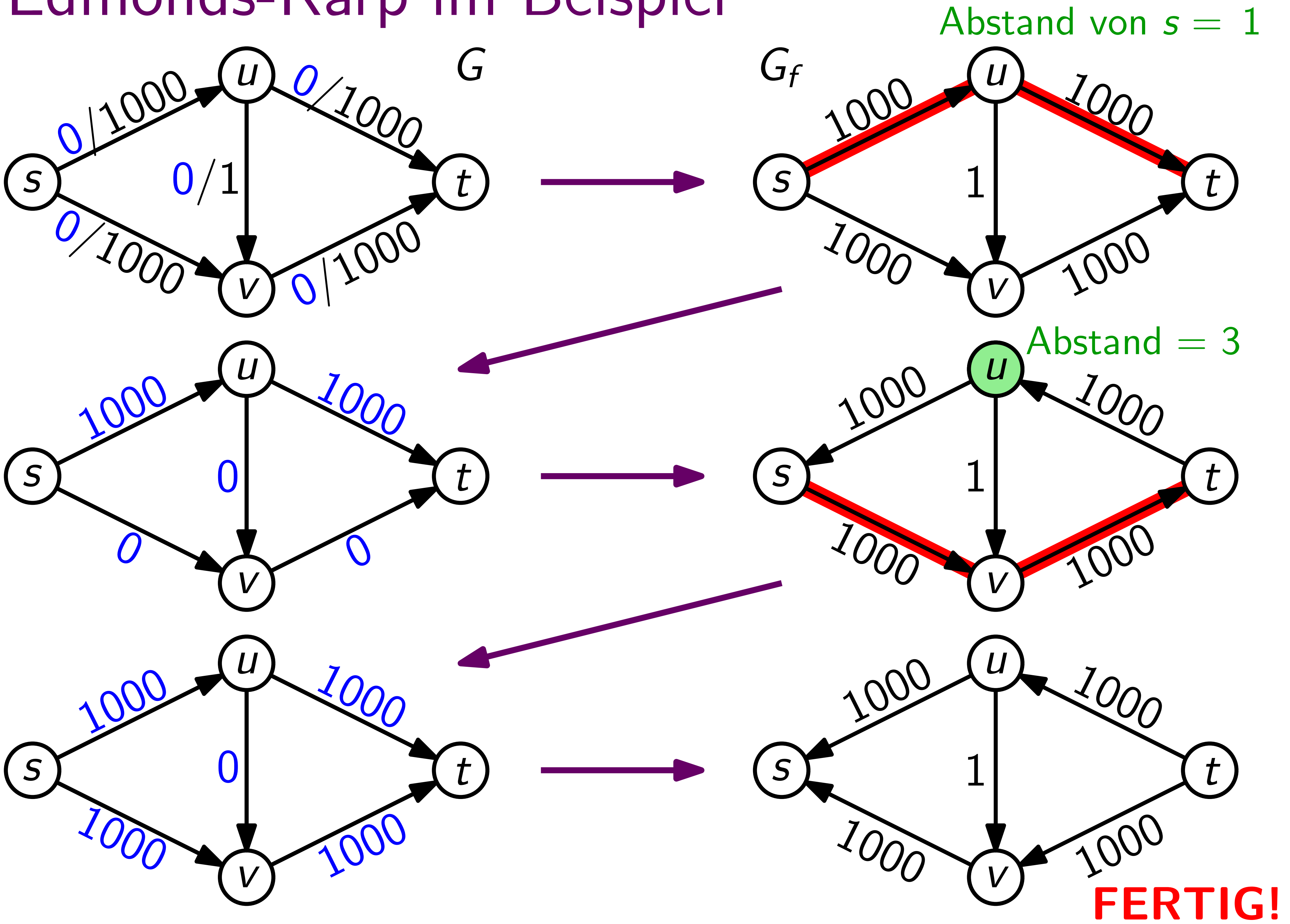
Edmonds-Karp im Beispiel



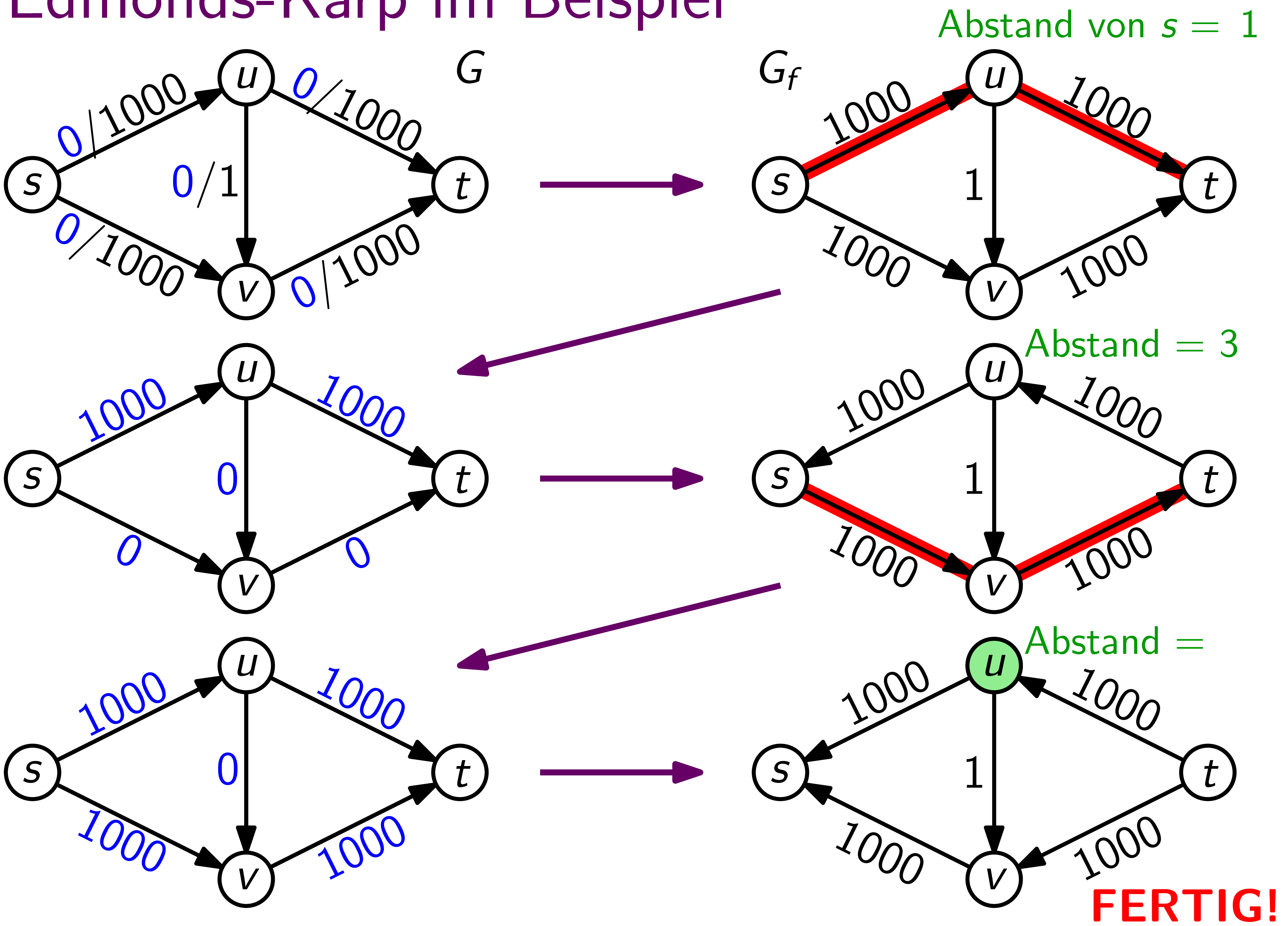
Edmonds-Karp im Beispiel



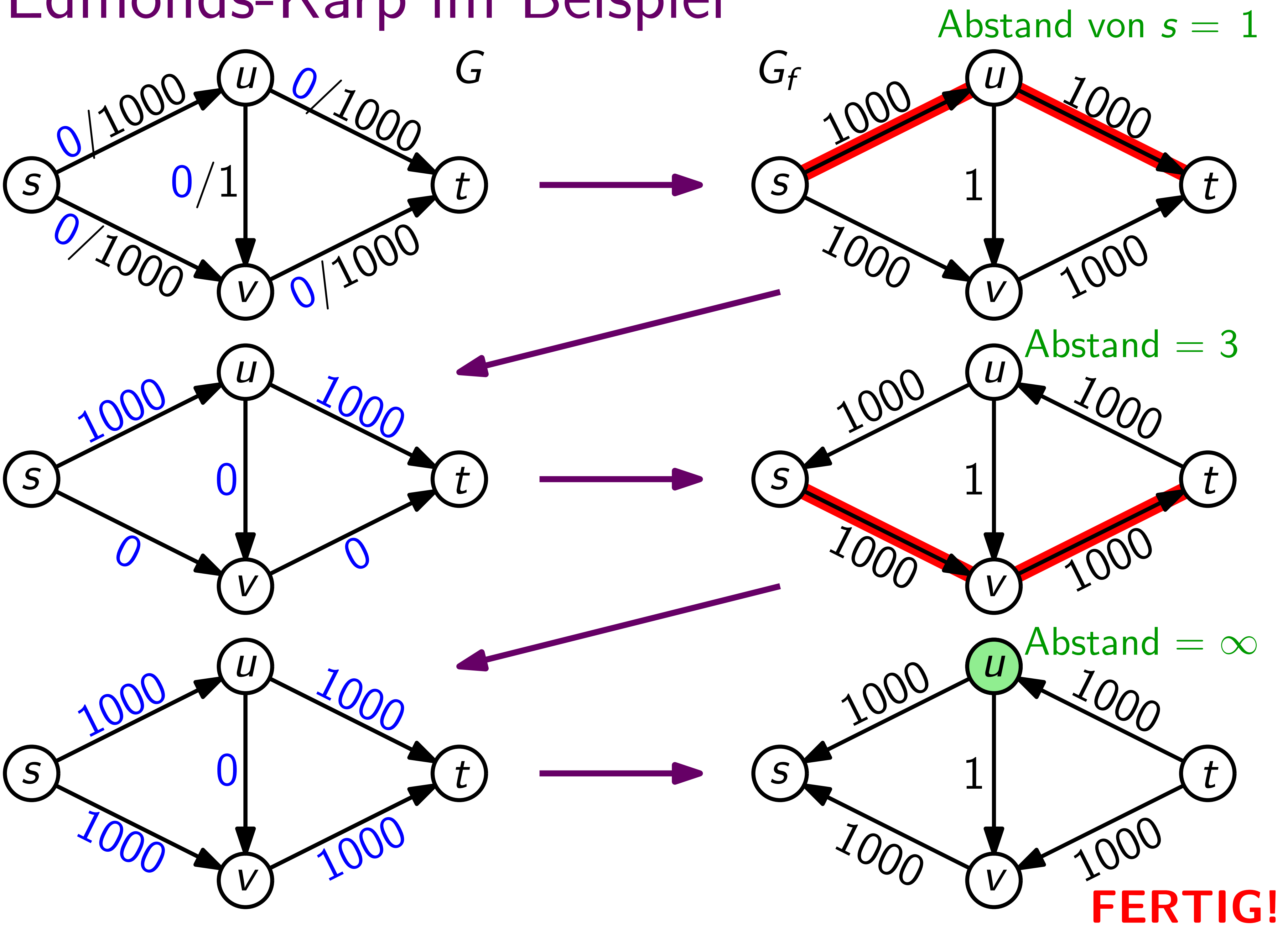
Edmonds-Karp im Beispiel



Edmonds-Karp im Beispiel



Edmonds-Karp im Beispiel



Anzahl Flusserhöhungen & Laufzeit

Satz
Sei $G = (V, E)$ ein gerichteter Graph mit Kantenkapazitäten $c(e)$ für $e \in E$, Quelle s, Senke t.
Der Edmonds-Karp-Algorithmus findet einen maximalen s-t-Fluss in G in $O(|V||E|^2)$ Zeit.

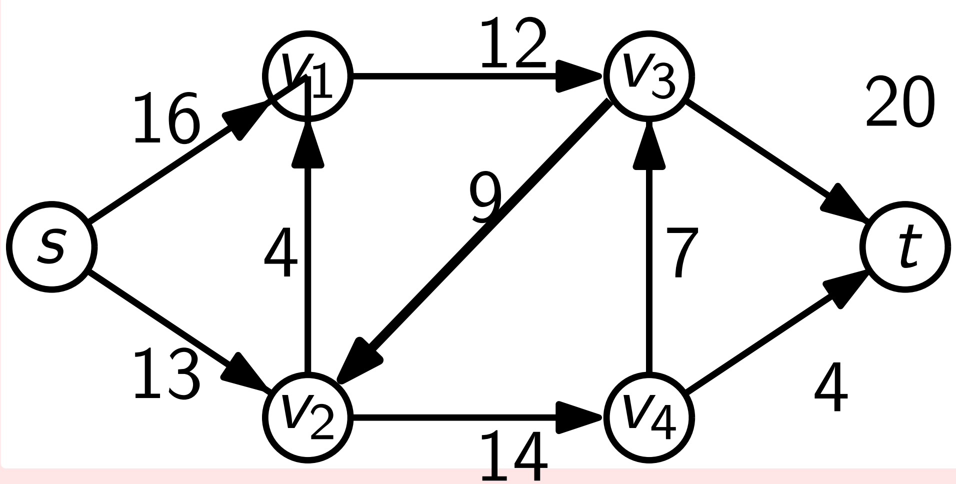
Beweis:

zu zeigen: Der Edmonds-Karp-Algorithmus führt $O(|V||E|)$ Flussvergrößerungen durch.
$\rightarrow$ siehe Extrafolien im WueCampus

Zusammenhang Flüsse und Schnitte 2

Maximaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: zulässiger Fluss f, mit maximalem Wert $|f| = \text{Nettozufluss}_f(t)$

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:



$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Minimaler Schnitt
Gegeben: gerichteter Graph $G = (V, E)$ mit Quelle $s \in V$, Senke $t \in V$ und Kantenkapazitäten $c\colon E \to \mathbb{R}_{>0}$.
Gesucht: s-t Schnitt mit minimaler Kapazität $c(\delta^+(S))$.

Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$

Beweis.

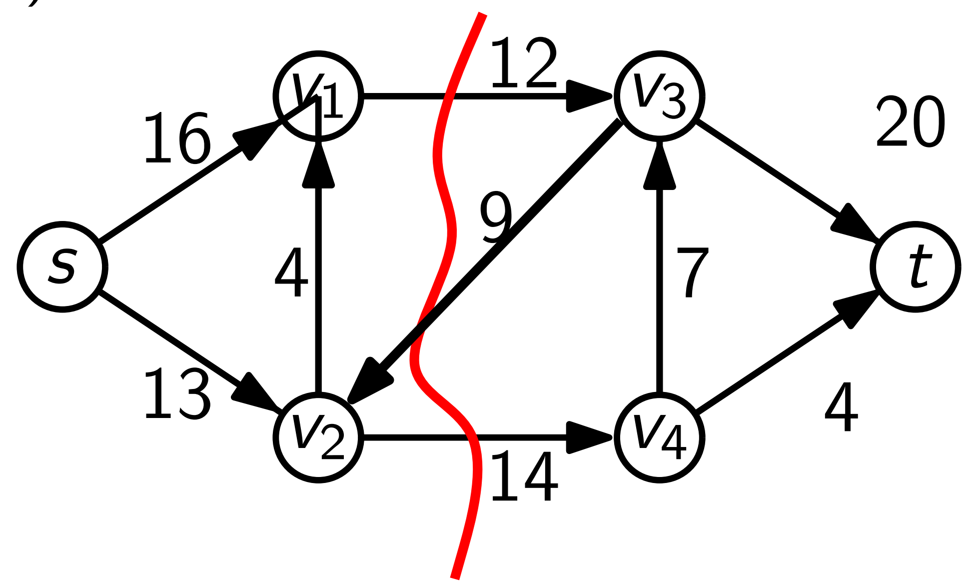
Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.

Korollar (slide 15):
Sei f ein zulässiger s-t-Fluss und (S, T) ein s-t-Schnitt.
Dann ist $|f| \leq c(\delta^+(S))$.



Das Max-Flow-Min-Cut-Theorem - Beweis

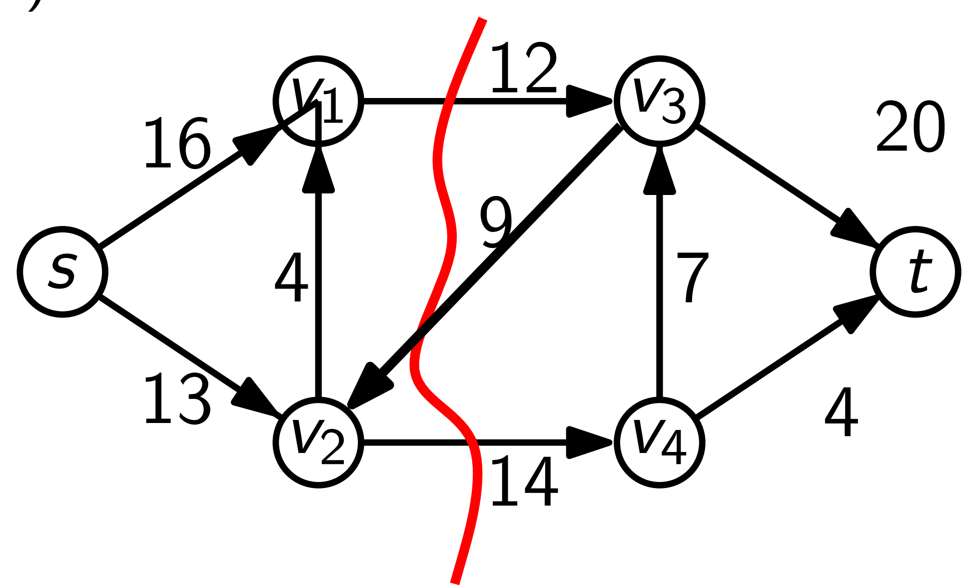
Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.

Korollar (slide 15):
Sei f ein zulässiger s-t-Fluss und (S, T) ein s-t-Schnitt.
Dann ist $|f| \leq c(\delta^+(S))$.

zu zeigen: es gibt zulässigen Fluss f^* und Schnitt (S^*, T^*) mit $|f^*| = \delta^+(S^*)$



Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.
zu zeigen: es gibt zulässigen Fluss f^* und Schnitt (S^*, T^*) mit $|f^*| = c(\delta^+(S^*))$

Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.
zu zeigen: es gibt zulässigen Fluss f^* und Schnitt (S^*, T^*) mit $|f^*| = c(\delta^+(S^*))$
Sei f^* ein maximaler Fluss.

Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.
zu zeigen: es gibt zulässigen Fluss f^* und Schnitt (S^*, T^*) mit $|f^*| = c(\delta^+(S^*))$

Sei f^* ein maximaler Fluss.

Sei $S^* := \{v \in V : v \text{ ist in } G_{f^*} \text{ von } s \text{ erreichbar}\}$,

Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.
zu zeigen: es gibt zulässigen Fluss f^* und Schnitt (S^*, T^*) mit $|f^*| = c(\delta^+(S^*))$

Sei f^* ein maximaler Fluss.

Sei $S^* := \{v \in V : v \text{ ist in } G_{f^*} \text{ von } s \text{ erreichbar}\}$, (S^*, T^*) ist s-t-Schnitt.

Das Max-Flow-Min-Cut-Theorem - Beweis

Max-Flow-Min-Cut Theorem
Sei f ein zulässiger s-t-Fluss in einem gerichteten Graphen $G = (V, E)$ mit Kapazitäten $c(e) > 0$ für alle $e \in E$.
Dann gilt:

$$\max_{f \text{ zulässiger } s-t- \text{ Fluss in } G} |f| = \min_{(S,T) \text{ ist } (s,t)\text{-Schnitt in } G} c(\delta^+(S))$$

Beweis.
zu zeigen: es gibt zulässigen Fluss f^* und Schnitt (S^*, T^*) mit $|f^*| = c(\delta^+(S^*))$

Sei f^* ein maximaler Fluss.

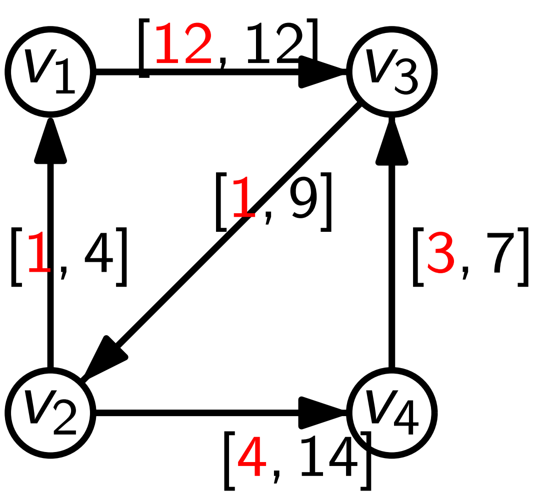
Sei $S^* := \{v \in V : v \text{ ist in } G_{f^*} \text{ von } s \text{ erreichbar}\}$, (S^*, T^*) ist s-t-Schnitt.
Analog zu Beweis des Satzes über den flusserhöhenden Weg:
$c(\delta^+(S^*)) = \ldots = |f^*|$

Schnellere Algorithmen für maximale Flüsse

Methode	*Laufzeit* $O(\cdot)$	*Autoren*
	Allgemeine gerichtete Graphen	
shortest resid. s-t path	$\lvert V\rvert\lvert E\rvert^2$	Dinic '70, Ed. & Karp
push relabel	$\lvert V\rvert^2\lvert E\rvert$	Goldberg '87
relabel to front	$\lvert V\rvert^3$	Goldberg & Tarjan '8
	$\lvert V\rvert\lvert E\rvert \log(\lvert V\rvert^2/\lvert E\rvert+2)$	— " —
blocking flow	$\min(\lvert V\rvert^{2/3}, \lvert E\rvert^{1/2}) \cdot \lvert E\rvert \cdot$ $\cdot \log(\lvert V\rvert^2/\lvert E\rvert+2) \cdot \log C,$	Goldberg & Rao '98 wobei $C = \sum_{e\in E} c(\epsilon$
new	$\lvert V\rvert\lvert E\rvert$	Orlin '13
	s-t-planare Graphen	
shortest path in dual	$\lvert V\rvert$	Hassin '81 + Henzinger et al. '9
	Planare Graphen	
leftmost resid. s-t path	$\lvert V\rvert \log\lvert V\rvert$	Borradaile & Klein '0
+ vertex capacities	$\lvert V\rvert \log\lvert V\rvert$	Kaplan & Nussbaum '

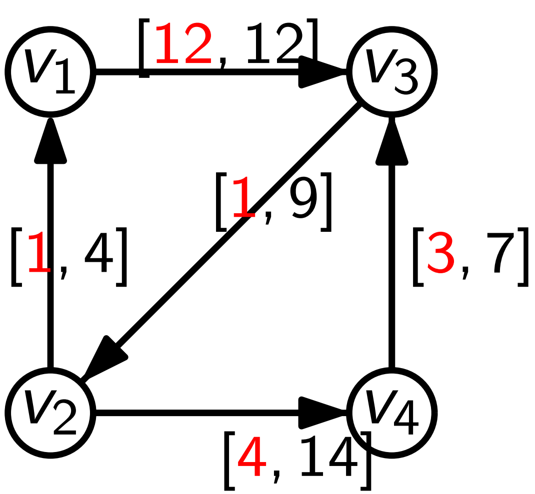
Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V :$ Nettozufl.$_f(v) = 0, \forall e \in E : l(e) \le f(e) \le c(e)$



Verallgemeinerung: untere Schranken auf den Kanten

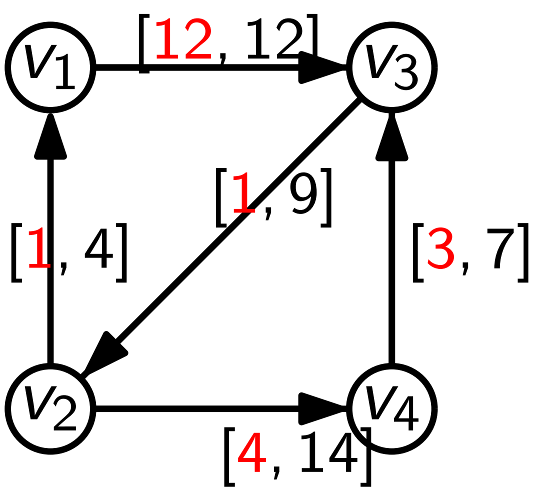
Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$



Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

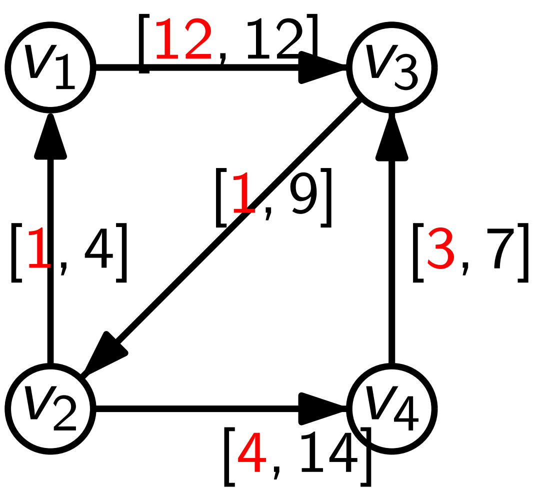
Konstruiere **Obergraph** $G' = (V', E')$:

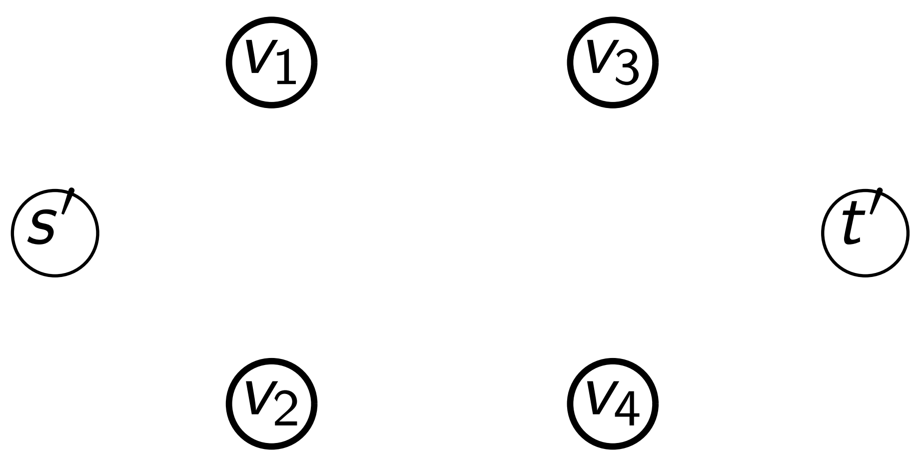


Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V$: Nettozufl.$_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,

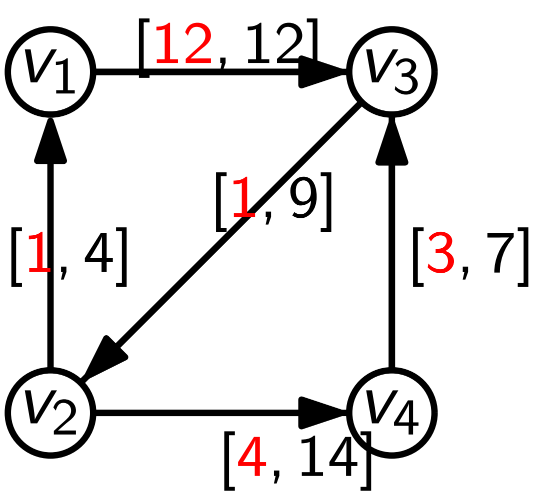


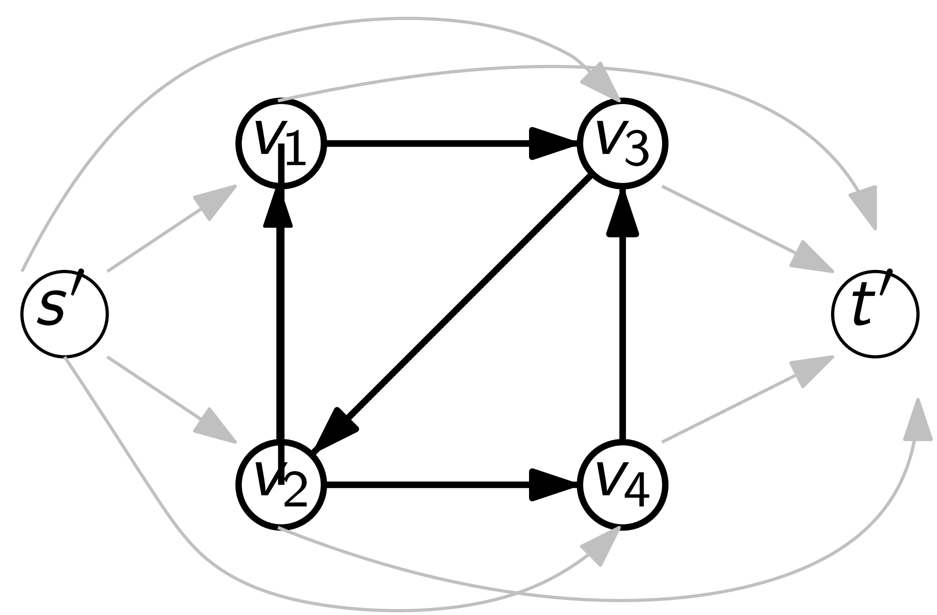


Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,
$E' = E \cup \{(s', v) : v \in V\} \cup \{(v, t') : v \in V\}$

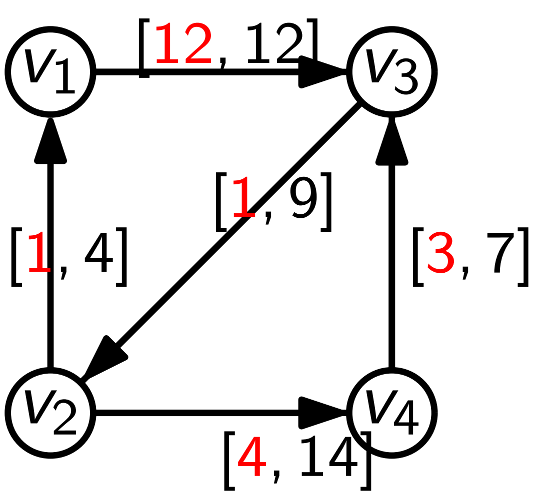


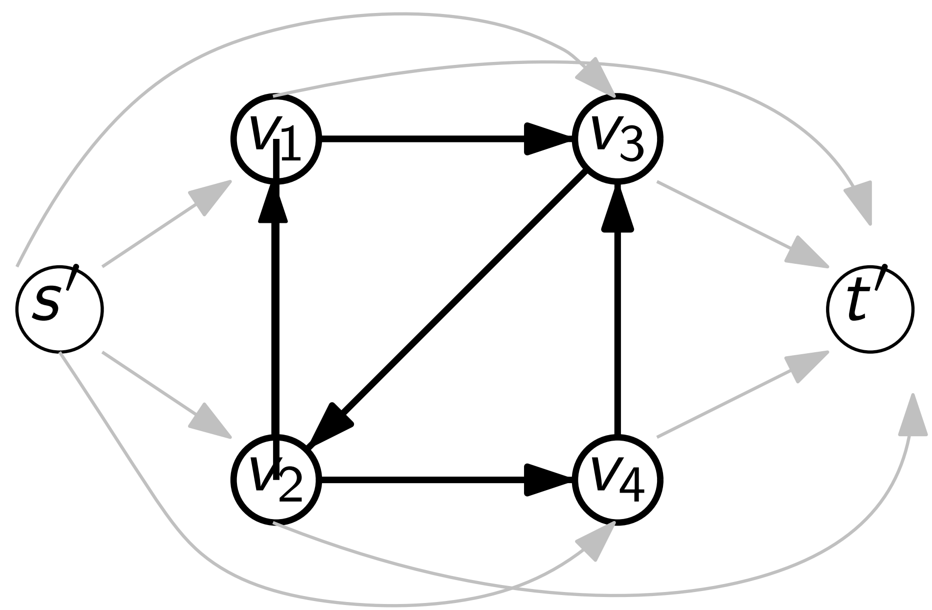


Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,
$E' = E \cup \{(s', v) : v \in V\} \cup \{(v, t') : v \in V\}$
$l'(e) := 0 \ \forall e \in E'$

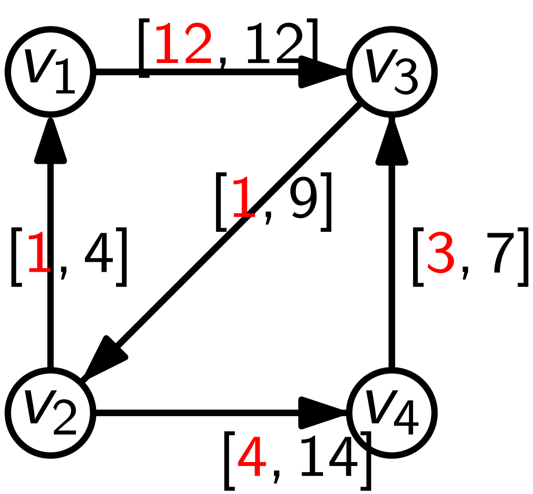


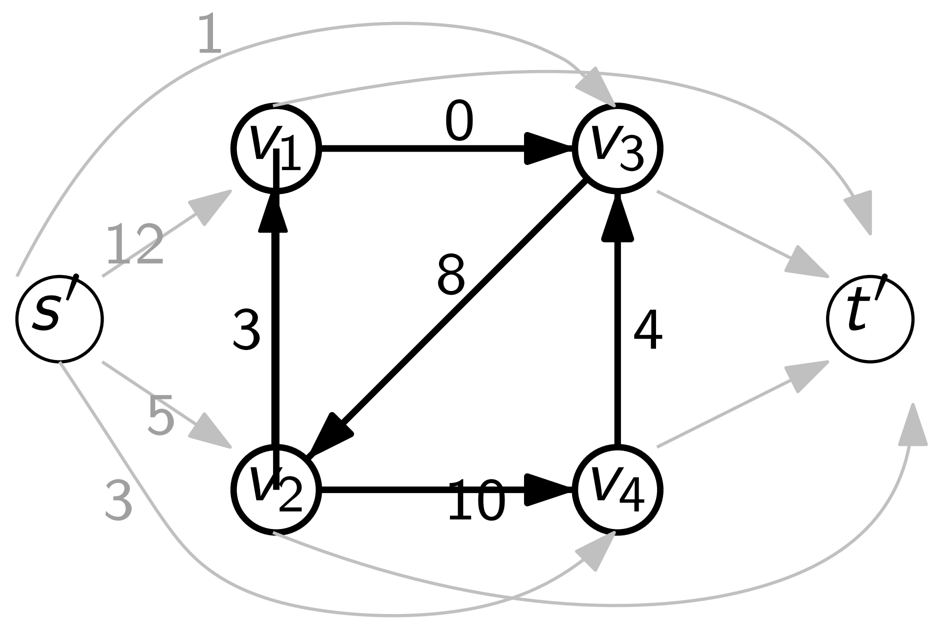


Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,
$E' = E \cup \{(s', v) : v \in V\} \cup \{(v, t') : v \in V\}$
$l'(e) := 0 \ \forall e \in E'$ $c'(e) := c(e) - l(e) \ \forall e \in E$





Verallgemeinerung: untere Schranken auf den Kanten

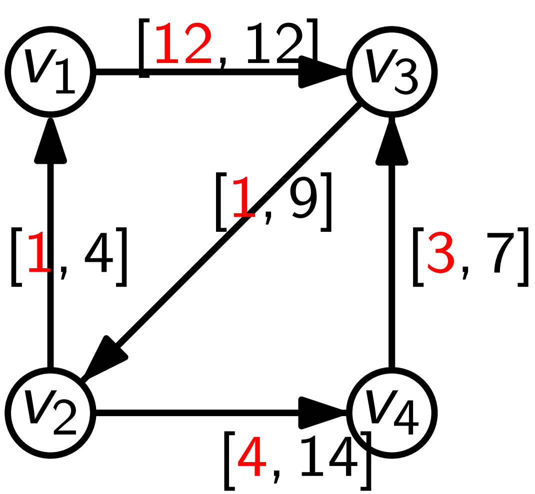
Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

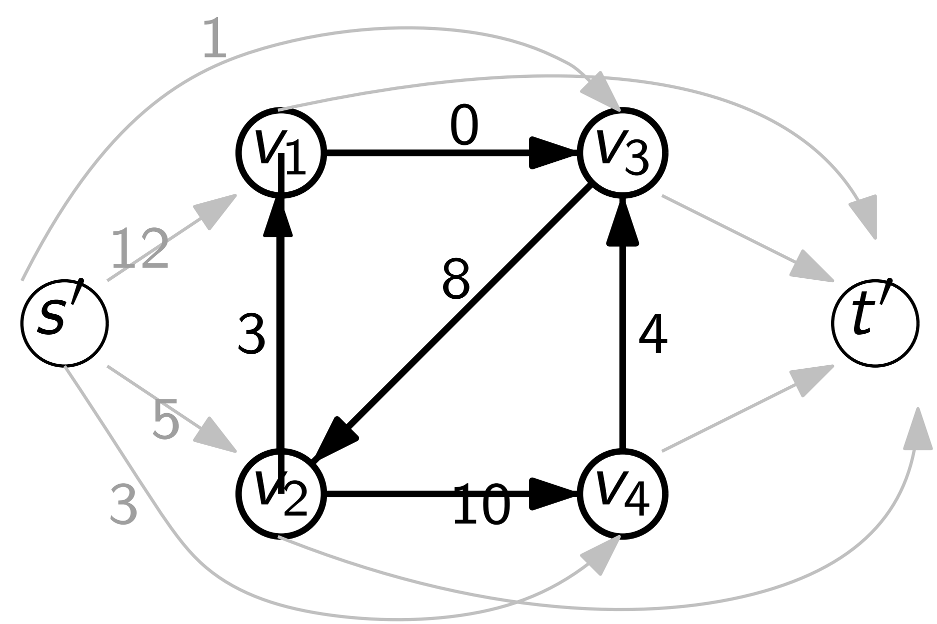
Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,

$E' = E \cup \{(s', v) : v \in V\} \cup \{(v, t') : v \in V\}$

$l'(e) := 0 \; \forall e \in E' \; c'(e) := c(e) - l(e) \; \forall e \in E$

für alle $v \in V$: $c'((s', v)) = \sum_{e \in \delta^-(v)} l(e)$





Verallgemeinerung: untere Schranken auf den Kanten

Strömungen

Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$

Gesucht: zulässige **Strömung**

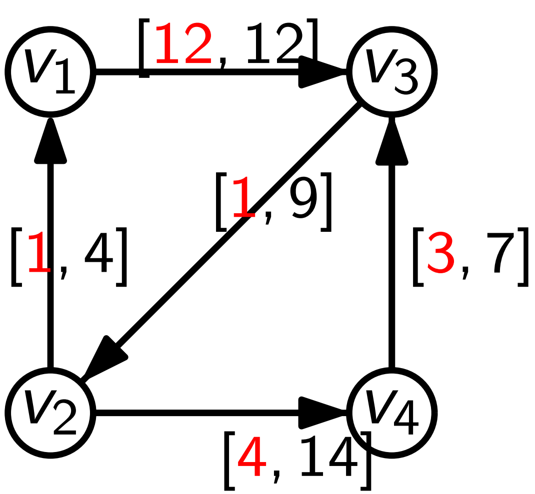
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

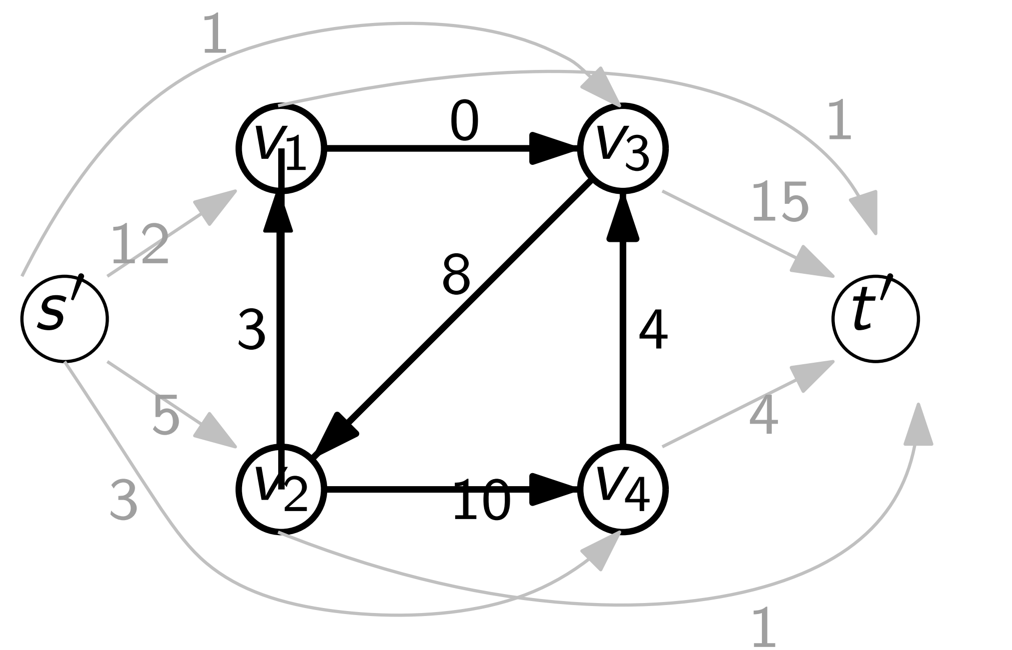
Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,

$E' = E \cup \{(s', v) : v \in V\} \cup \{(v, t') : v \in V\}$

$l'(e) := 0 \ \forall e \in E' \ c'(e) := c(e) - l(e) \ \forall e \in E$

für alle $v \in V$: $c'((s', v)) = \sum_{e \in \delta^-(v)} l(e)$ und $c'((v, t') = \sum_{e \in \delta^+(v)} l(e)$

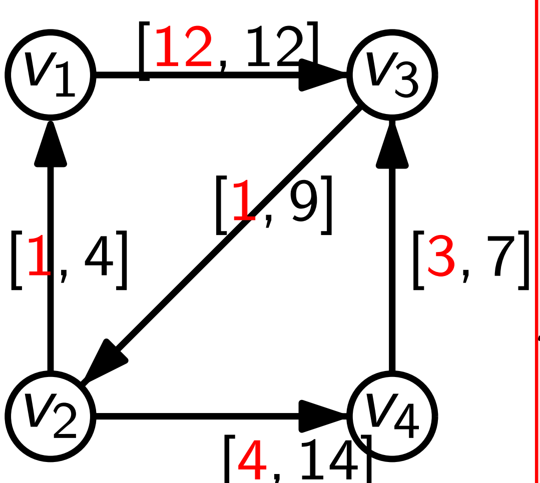




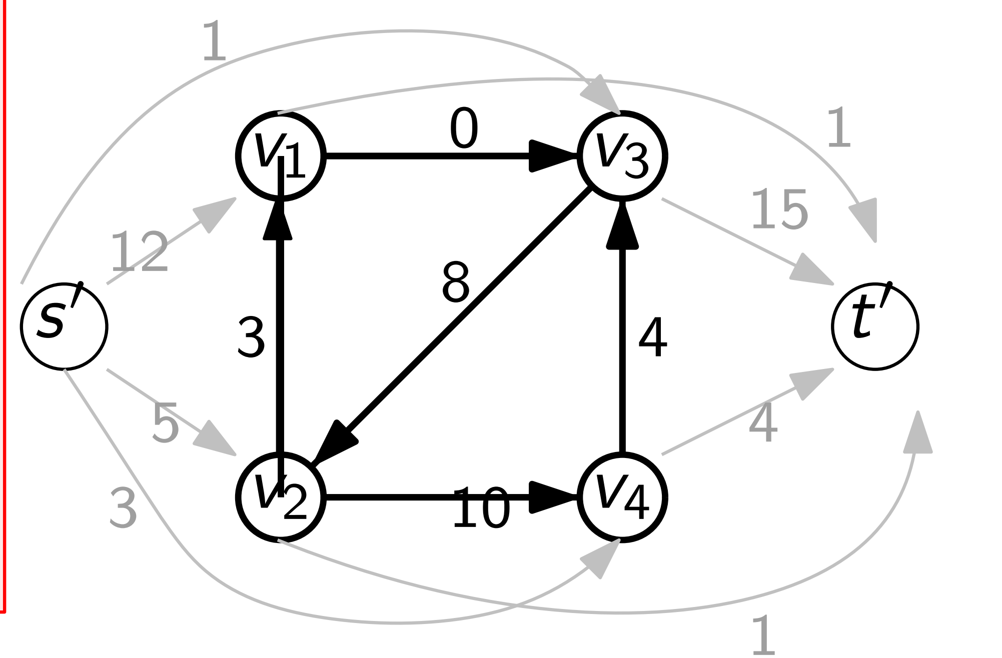
Verallgemeinerung: untere Schranken auf den Kanten

Strömungen
Gegeben: gerichteter Graph $G = (V, E)$ mit unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässige **Strömung**
$f : E \to \mathbb{R} : \forall v \in V : \text{Nettozufl.}_f(v) = 0, \forall e \in E : l(e) \leq f(e) \leq c(e)$

Konstruiere **Obergraph** $G' = (V', E')$: $V' = V \cup \{s', t'\}$,
$E' = E \cup \{(s', v) : v \in V\} \cup \{(v, t') : v \in V\}$
$l'(e) := 0 \ \forall e \in E'$ $c'(e) := c(e) - l(e) \ \forall e \in E$
für alle $v \in V$: $c'((s', v)) = \sum_{e \in \delta^-(v)} l(e)$ und $c'((v, t')) = \sum_{e \in \delta^+(v)} l(e)$



Lemma
Es gibt eine zulässige Strömung in G
$\Leftrightarrow$
Der maximale s'-t'-Fluss in G' hat Wert $\sum_{e \in E} l(e)$.



Verallgemeinerung: Bedarf und Überschuss

b=10

b=5

b=20

Verallgemeinerung: Bedarf und Überschuss

b=10

b=-13

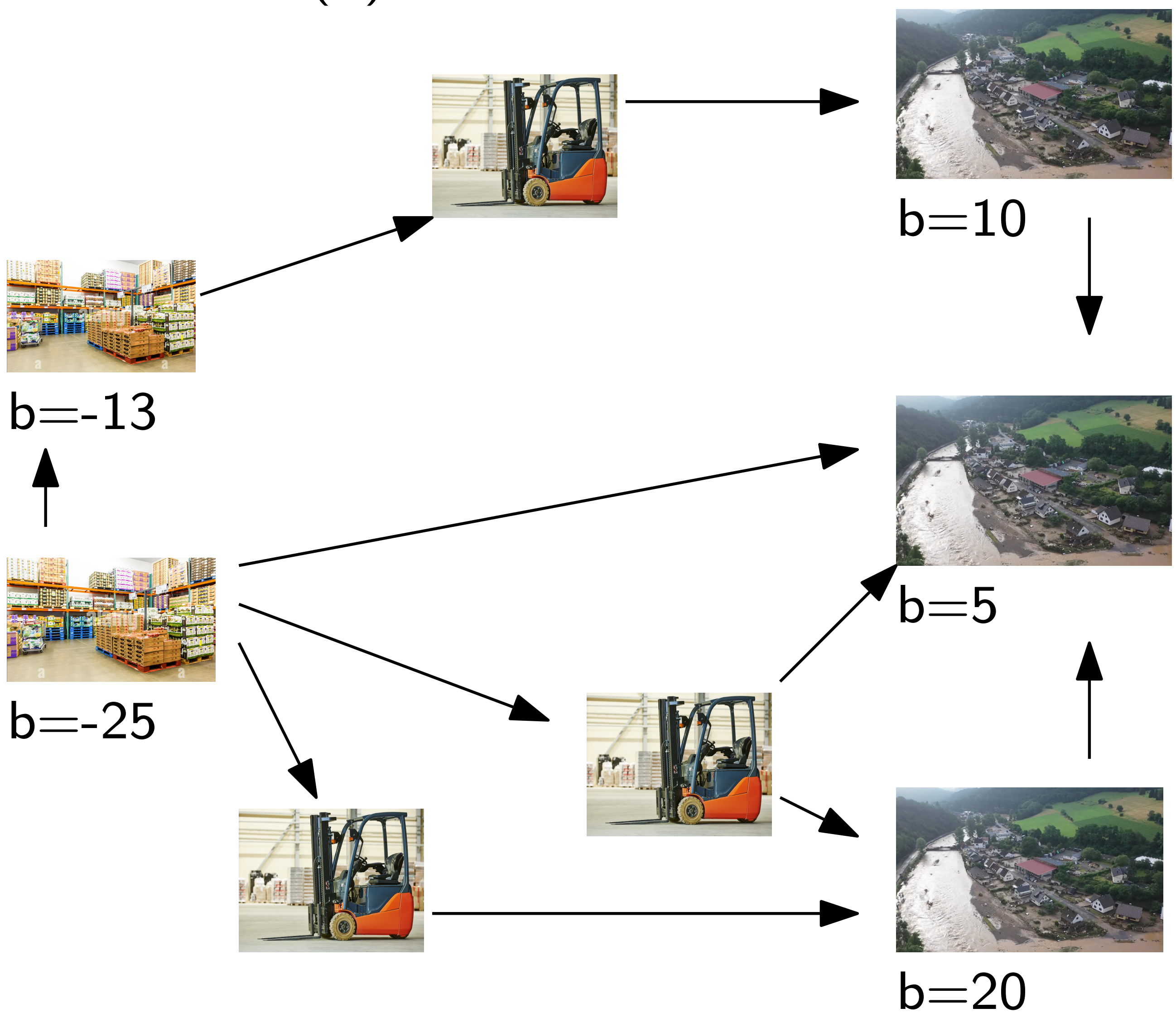
b=5

b=-25

b=20

Verallgemeinerung: Bedarf und Überschuss

Knoten haben **Bedarf** $b(v)$
oder **Überschuss** $-b(v)$



Verallgemeinerung: Bedarf und Überschuss

b=-13

b=-25

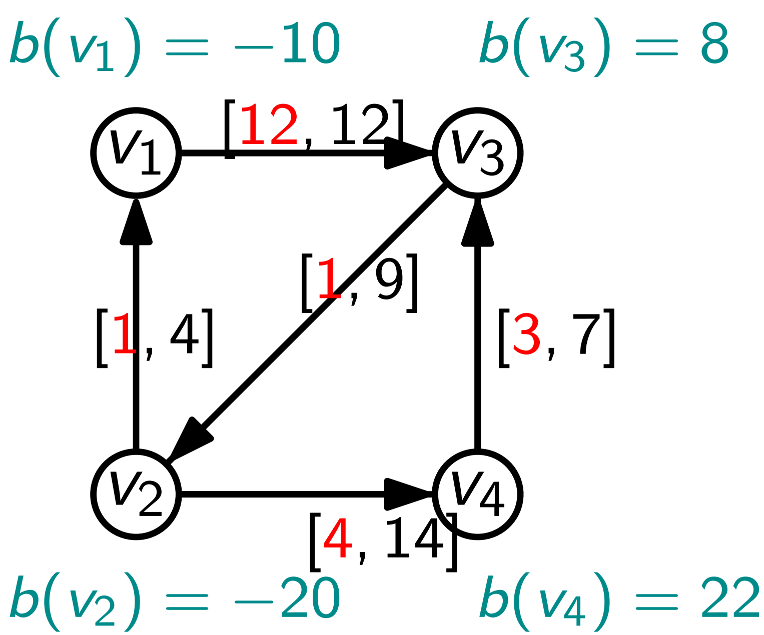
b=10

b=5

b=20

Verallgemeinerung: Bedarf und Überschuss

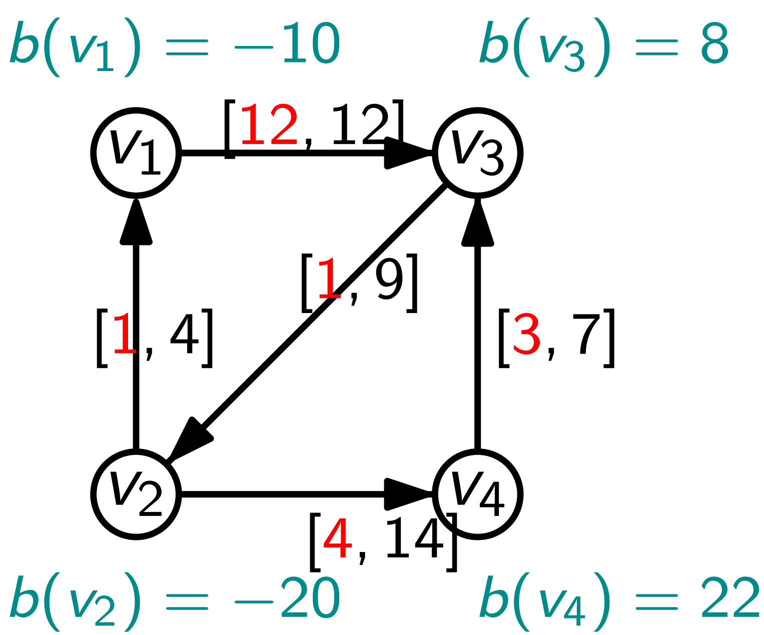
b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$,
$\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$



Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$,
$\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

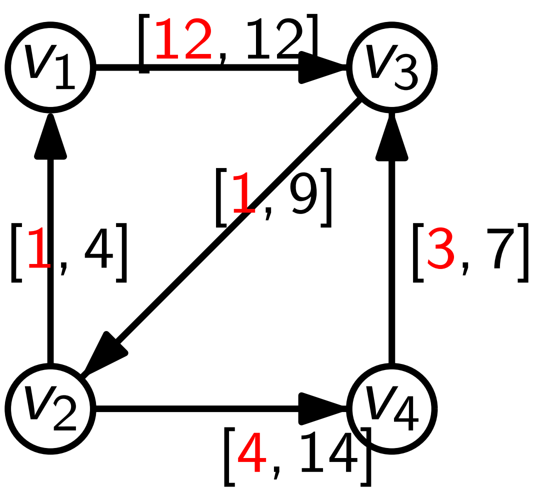
Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$:

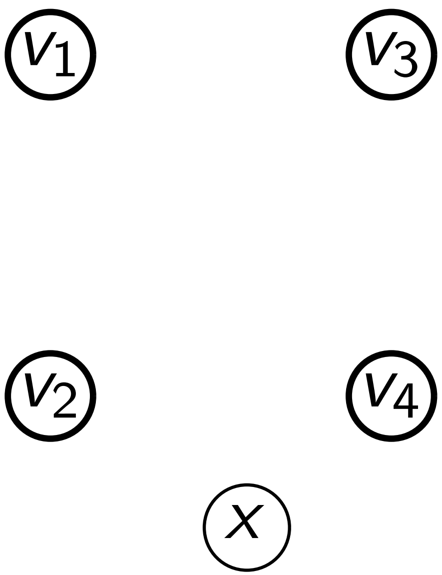


Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \le f(e) \le c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\hat{V} := V \cup \{x\}$,

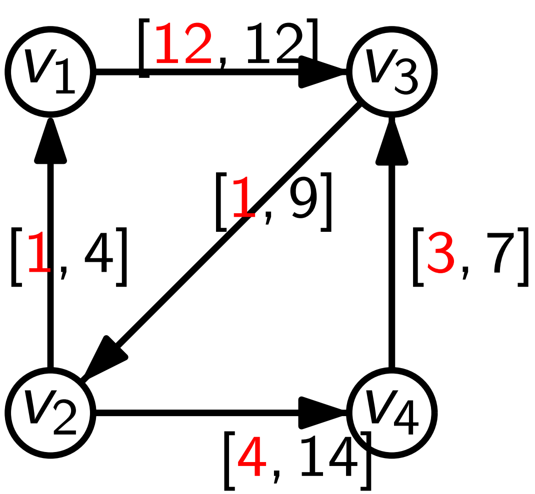


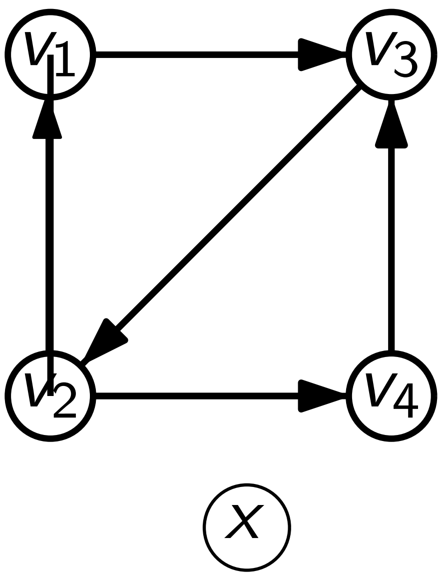


Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\hat{V} := V \cup \{x\}$,
$\hat{E} := E$

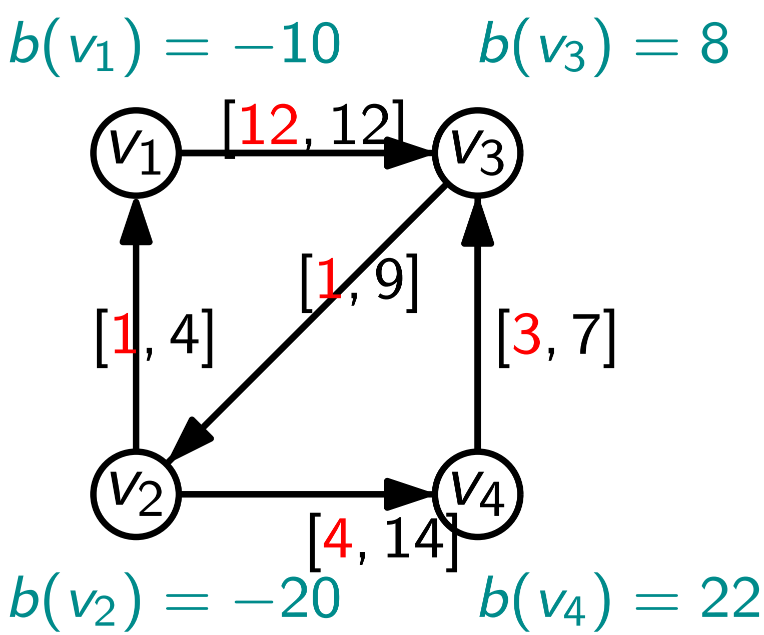


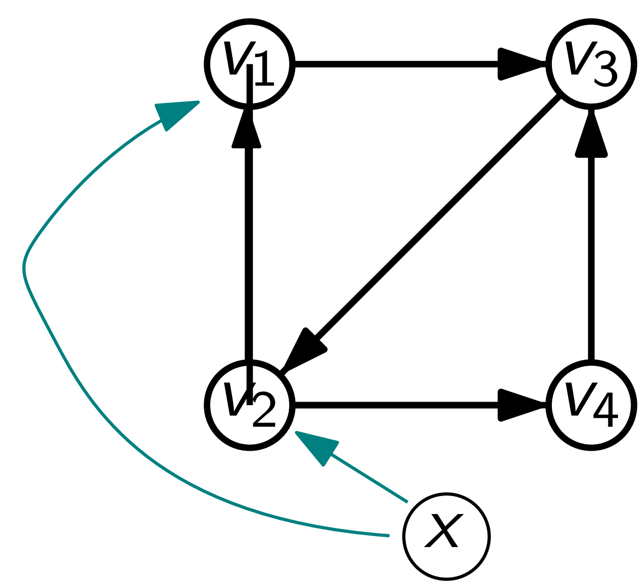


Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\hat{V} := V \cup \{x\}$,
$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\}$

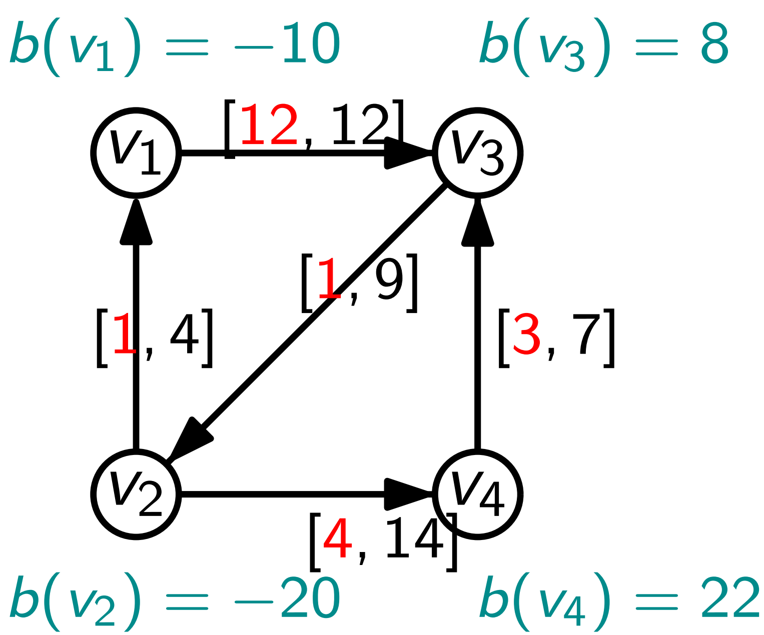


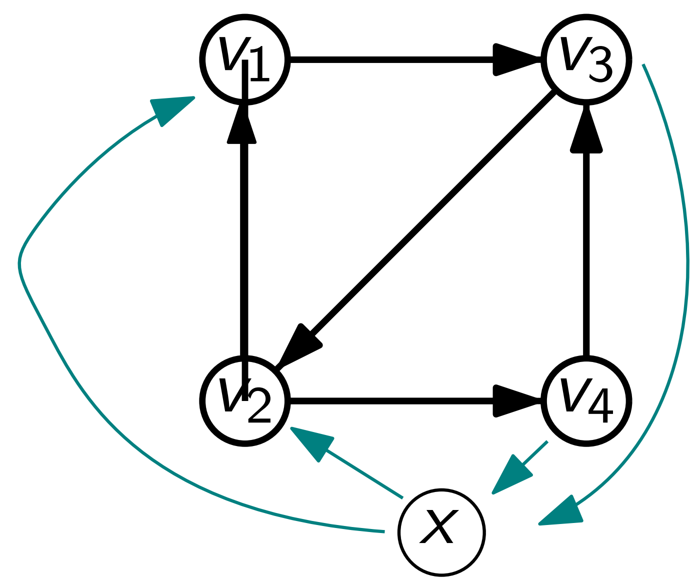


Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\hat{V} := V \cup \{x\}$,
$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$





Verallgemeinerung: Bedarf und Überschuss

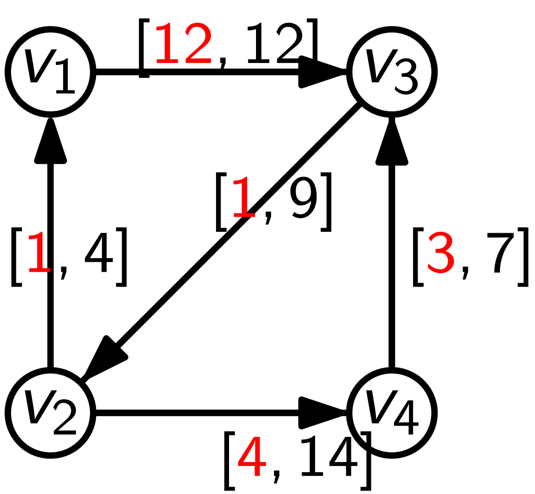
b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$,
$\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

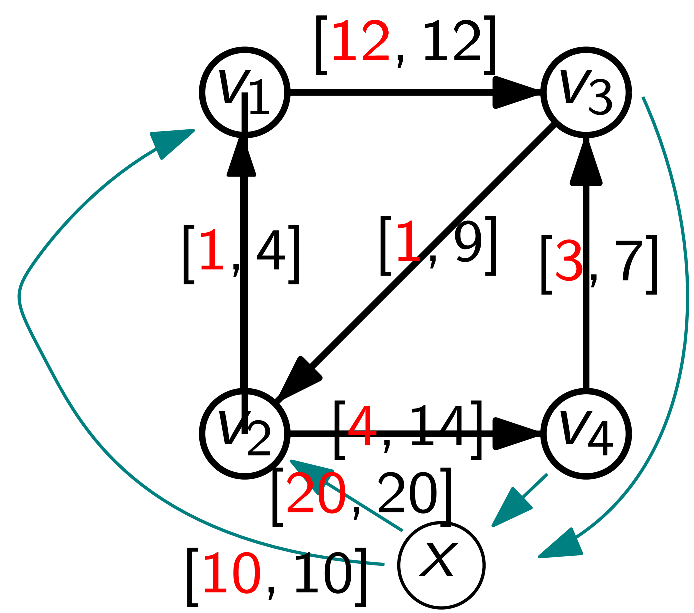
Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\quad \hat{V} := V \cup \{x\}$,

$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$

$\hat{l}(e) := l(e), \hat{c} = c(e) \; \forall e \in E'$

$l(x, v) := c(x, v) := -b(v)$

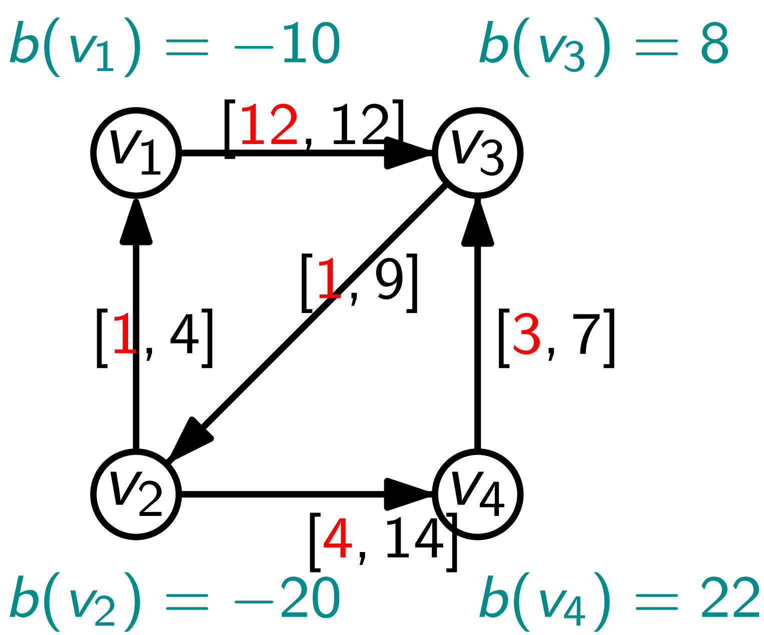


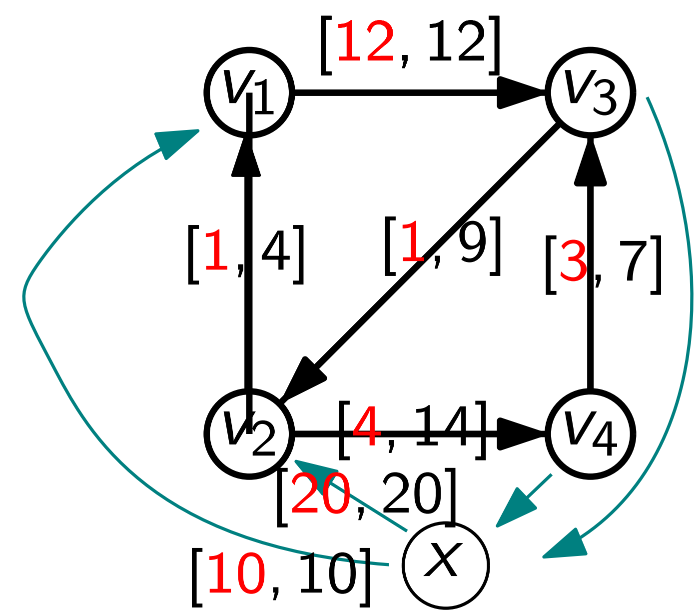


Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$,
$\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\hat{V} := V \cup \{x\}$,
$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$
$\hat{l}(e) := l(e), \hat{c} = c(e) \; \forall e \in E'$
$l(x, v) := c(x, v) := -b(v)$





Verallgemeinerung: Bedarf und Überschuss

b-Flüsse

Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$

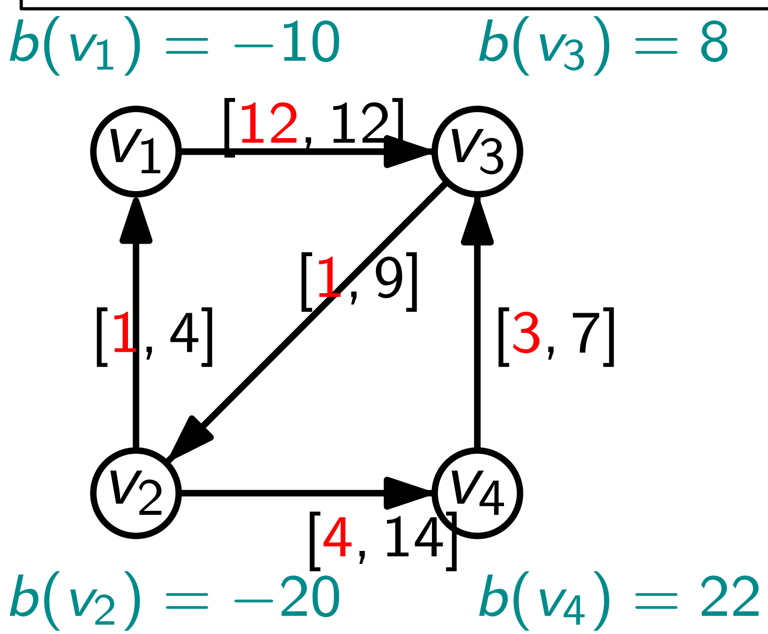
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

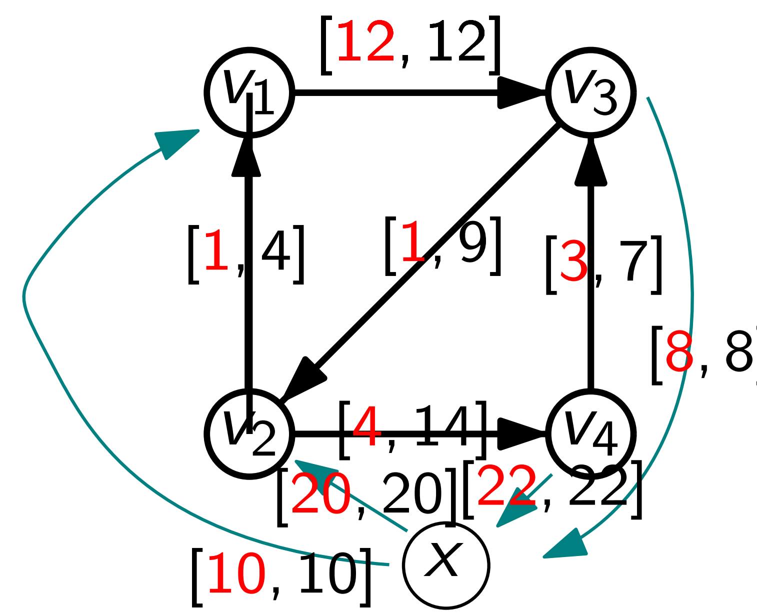
Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\quad \hat{V} := V \cup \{x\}$,

$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$

$\hat{l}(e) := l(e), \hat{c} = c(e) \; \forall e \in E'$

$l(x, v) := c(x, v) := -b(v) \quad l((u, v)) := c((u, v)) = b(u)$





Verallgemeinerung: Bedarf und Überschuss

b-Flüsse

Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$

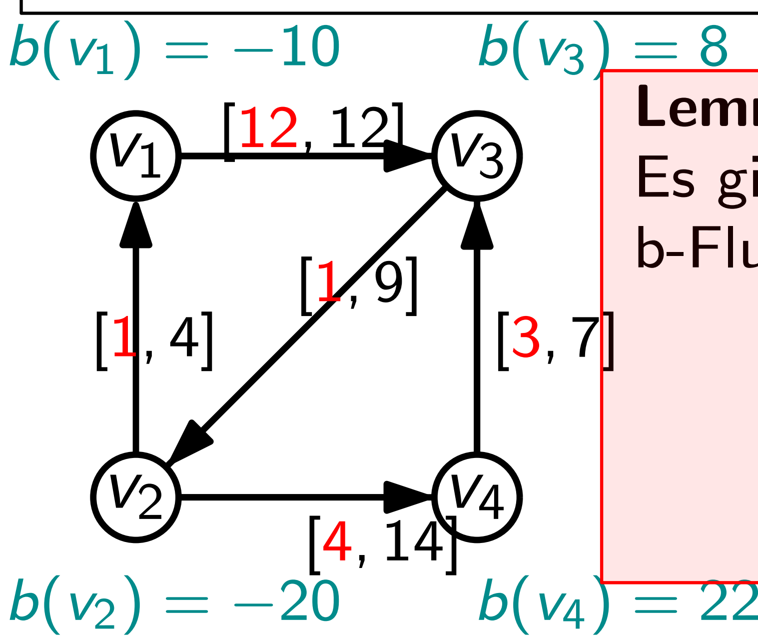
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\quad \hat{V} := V \cup \{x\}$,

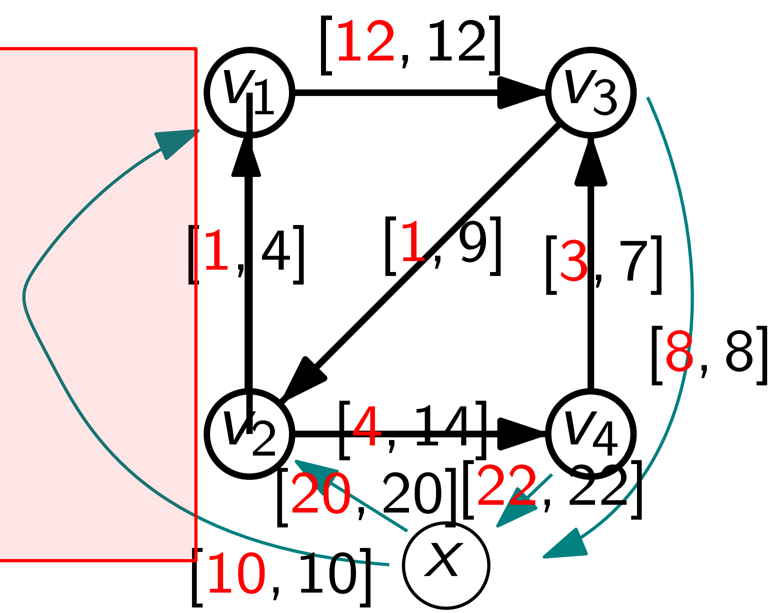
$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$

$\hat{l}(e) := l(e), \hat{c} = c(e)\ \forall e \in E'$

$l(x, v) := c(x, v) := -b(v) \quad l((u, v)) := c((u, v)) = b(u)$



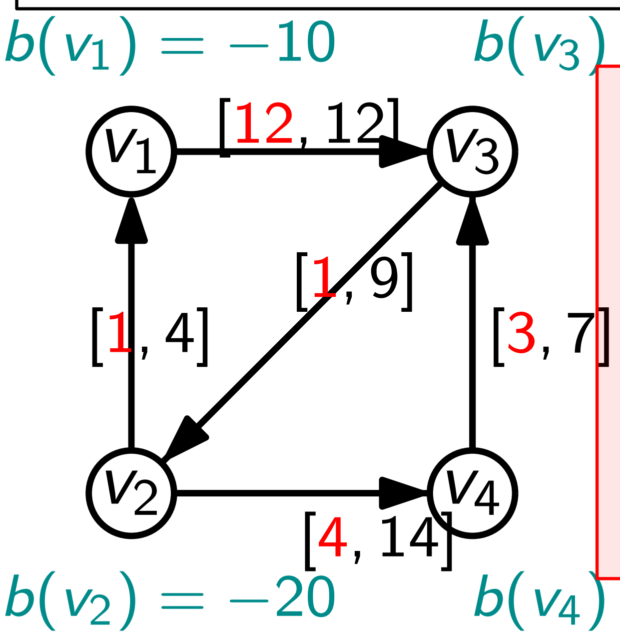
Lemma:
Es gibt einen zulässigen b-Fluss in G $\Leftrightarrow$



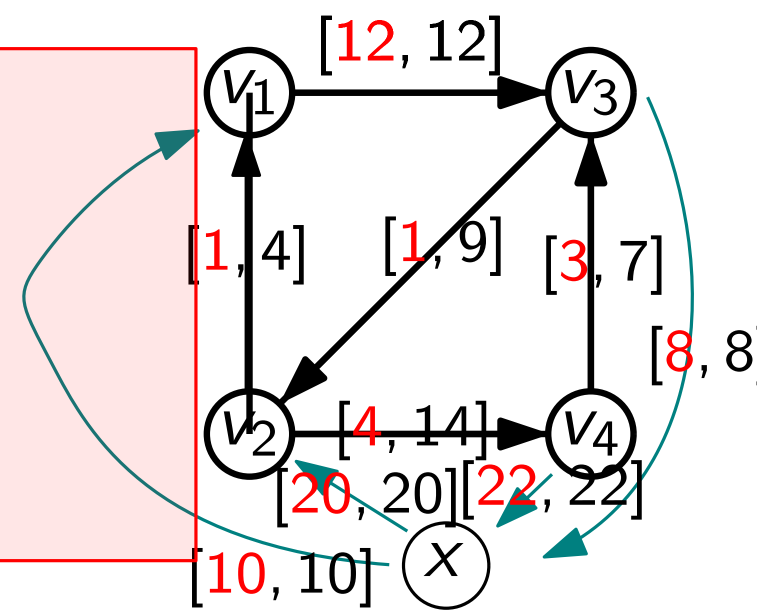
Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\quad \hat{V} := V \cup \{x\}$,
$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$
$\hat{l}(e) := l(e), \hat{c} = c(e) \; \forall e \in E'$
$l(x, v) := c(x, v) := -b(v) \;\; l((u, v)) := c((u, v)) = b(u)$



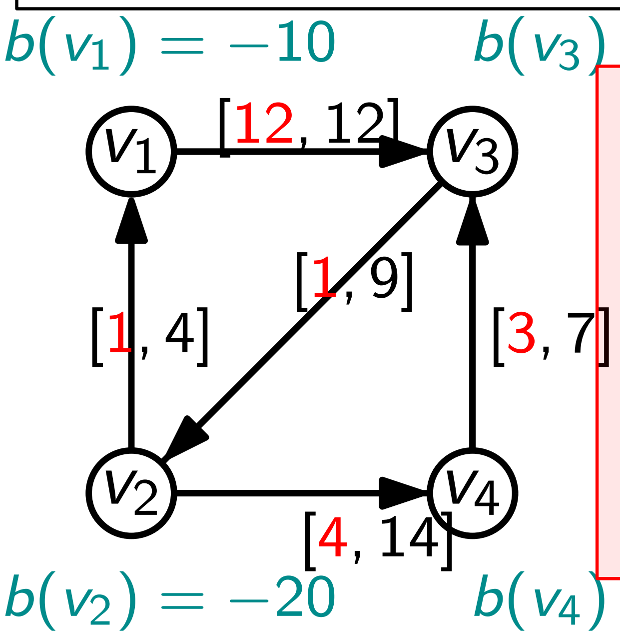
Lemma:
Es gibt einen zulässigen b-Fluss in $G \Leftrightarrow$
Es gibt eine zulässige Strömung in $\hat{G} \Leftrightarrow$



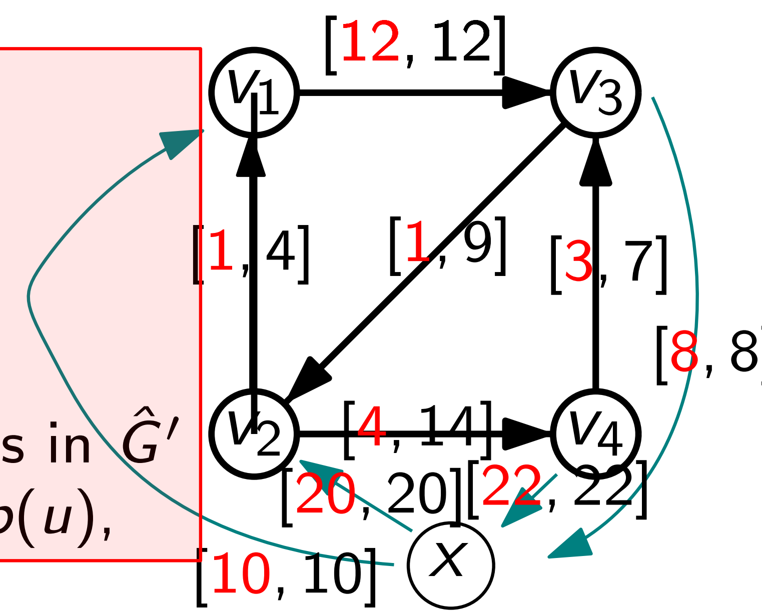
Verallgemeinerung: Bedarf und Überschuss

b-Flüsse
Gegeben: gerichteter Graph $G = (V, E)$ mit Bedarfen $b(v)$ für $v \in V$, unteren & obere Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$
Gesucht: zulässiger **b-Fluss** $f : E \to \mathbb{R} : \forall e \in E : l(e) \leq f(e) \leq c(e)$, $\forall v \in V : b(v) = \text{Nettozufl.}_f(v)$

Konstruiere **Obergraph** $\hat{G} = (\hat{V}, \hat{E})$: $\hat{V} := V \cup \{x\}$,
$\hat{E} := E \cup \{(x, v) : v \in V : b(v) < 0\} \cup \{(u, x) : u \in V : b(u) > 0\}$
$\hat{l}(e) := l(e), \hat{c} = c(e) \; \forall e \in E'$
$l(x, v) := c(x, v) := -b(v)$ $\; l((u, v)) := c((u, v)) = b(u)$

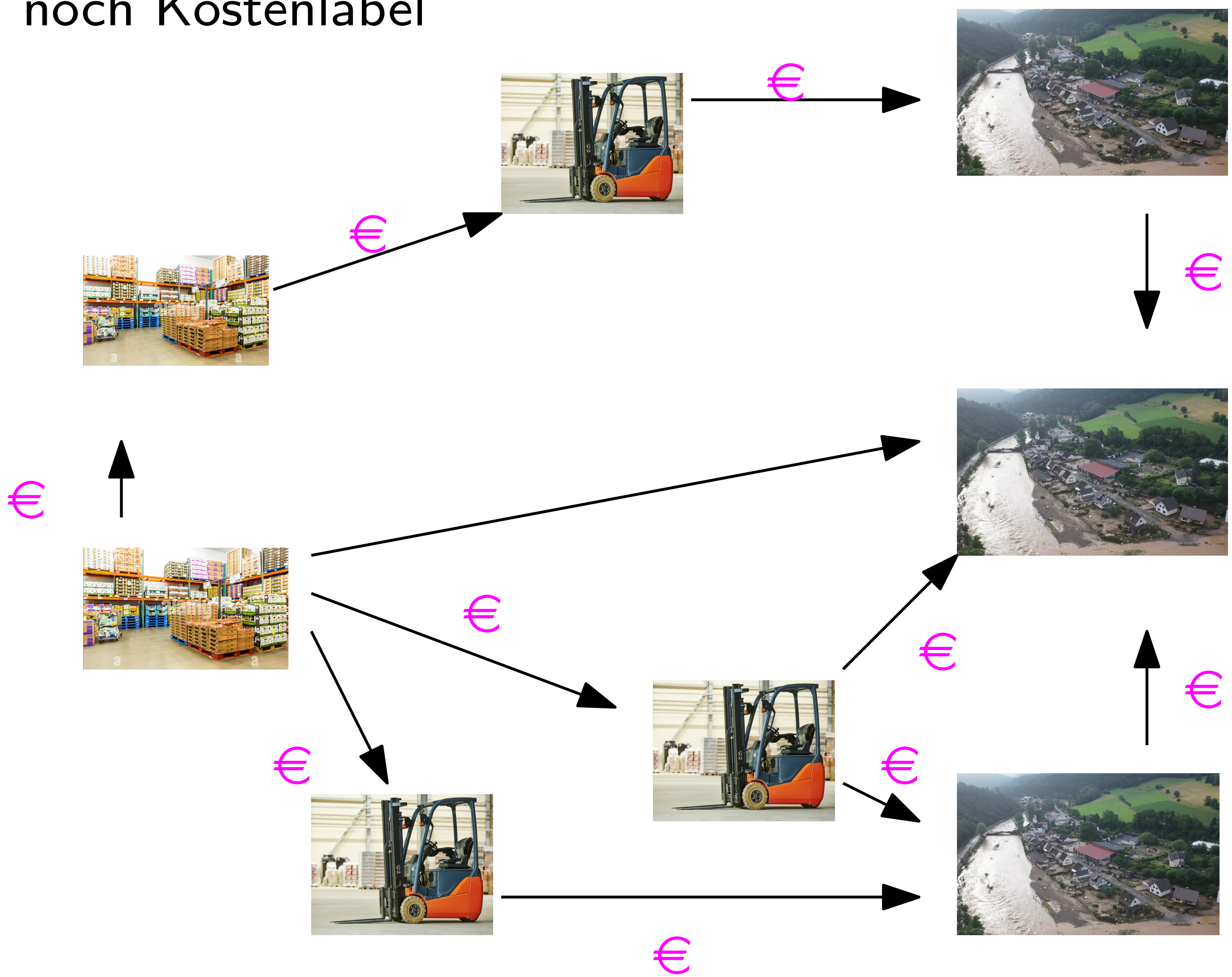


Lemma:
Es gibt einen zulässigen b-Fluss in G $\Leftrightarrow$
Es gibt eine zulässige Strömung in $\hat{G}$ $\Leftrightarrow$
Der maximale s'-t'-Fluss in $\hat{G}'$ hat Wert $\sum_{u \in V : b(u) > 0} b(u)$,



Verallgemeinerung: Kostenminimale Flüsse

Knoten haben **Bedarf** $b(v)$ oder **Überschuss** $-b(v)$
Neben (unteren und) oberen Kapazitätsschranken haben Kanten jetzt auch noch Kostenlabel



Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss

Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.

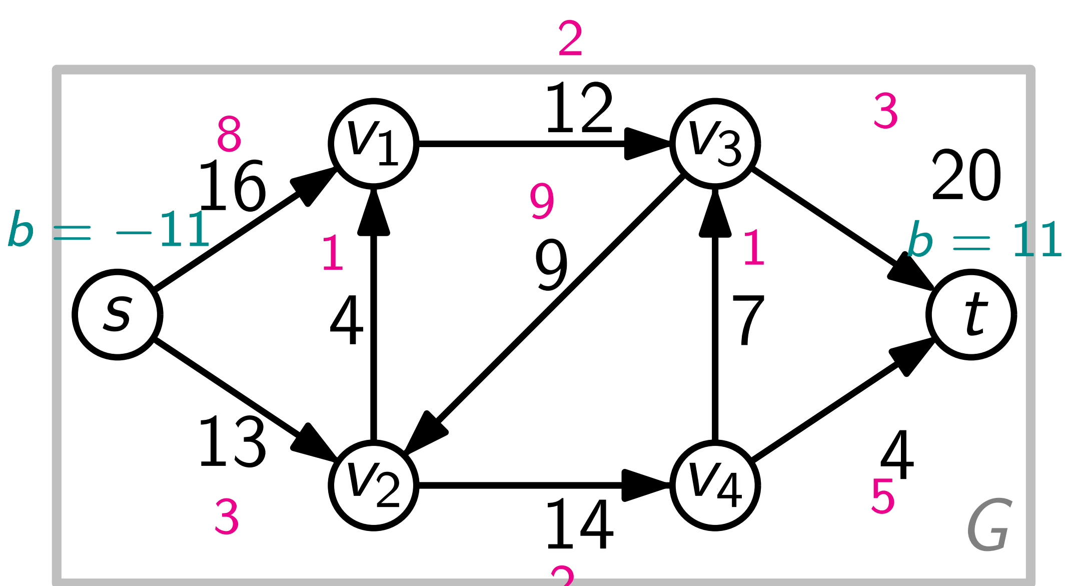
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss

Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.

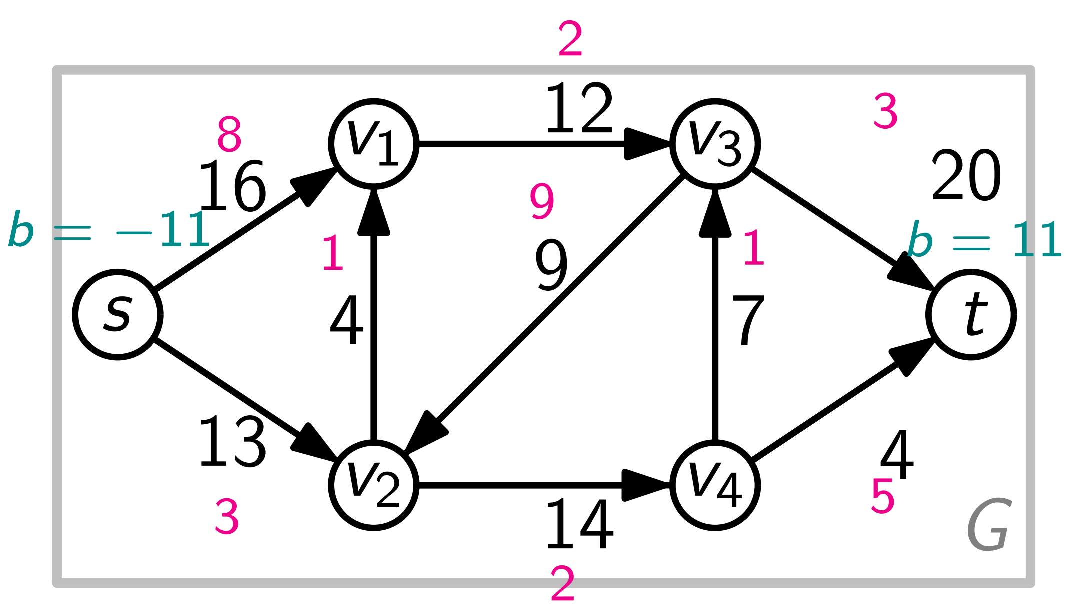
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.



Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

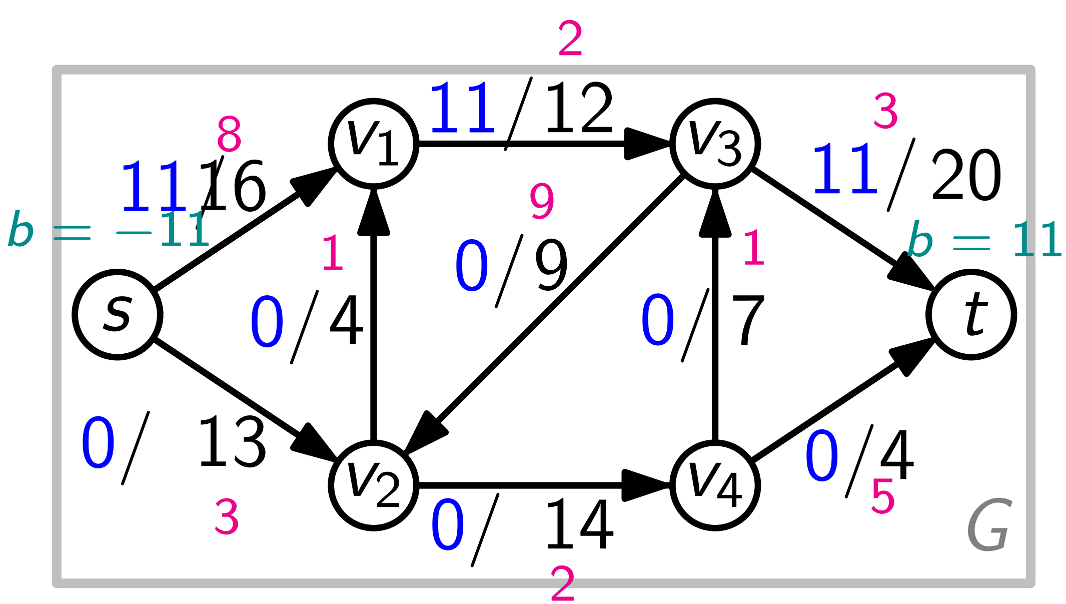
Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u, v)) = k((u, v))$ für $(u, v) \in E$
-$k_f((u, v)) = -k((v, u))$ für $(v, u) \in E$



Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

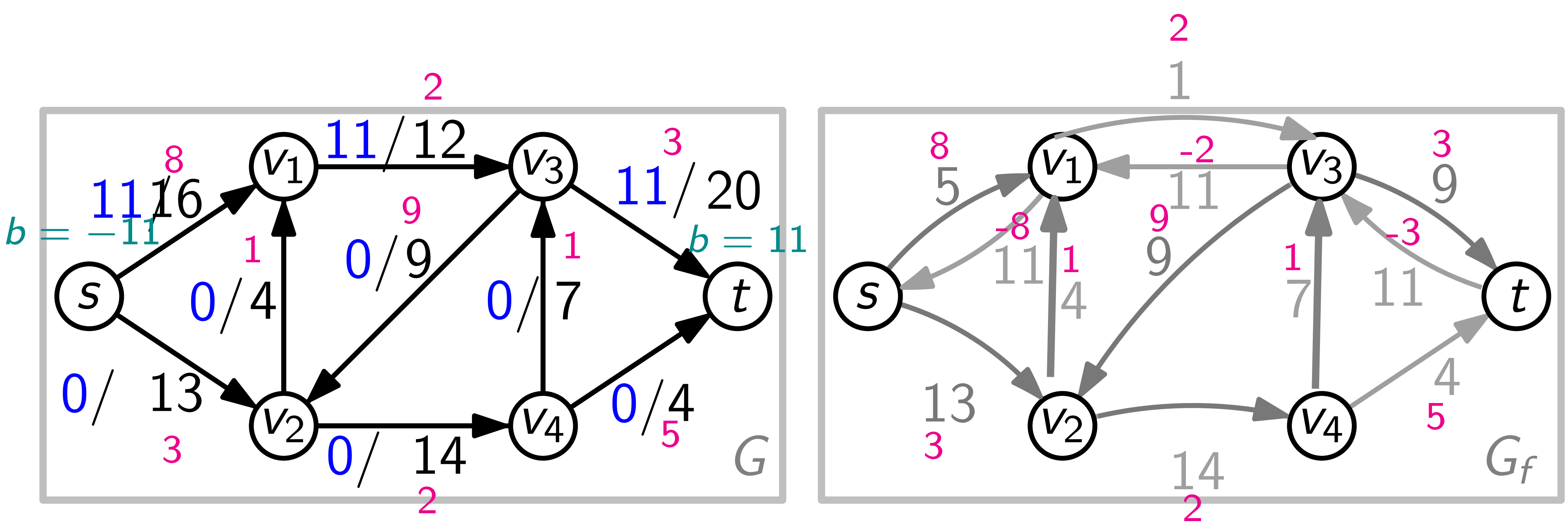
Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$



Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u, v)) = k((u, v))$ für $(u, v) \in E$
-$k_f((u, v)) = -k((v, u))$ für $(v, u) \in E$

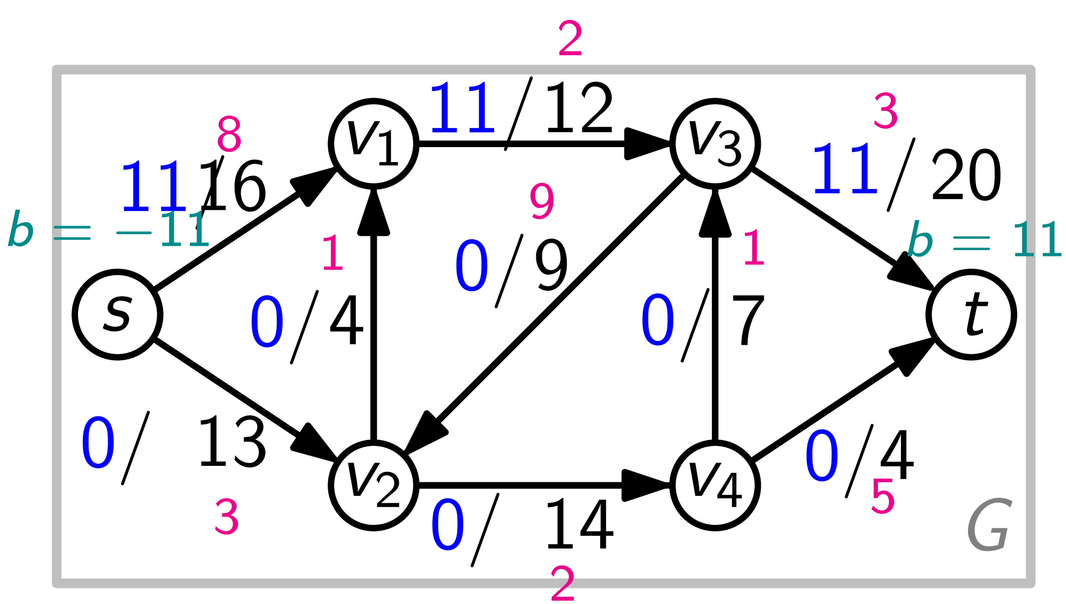


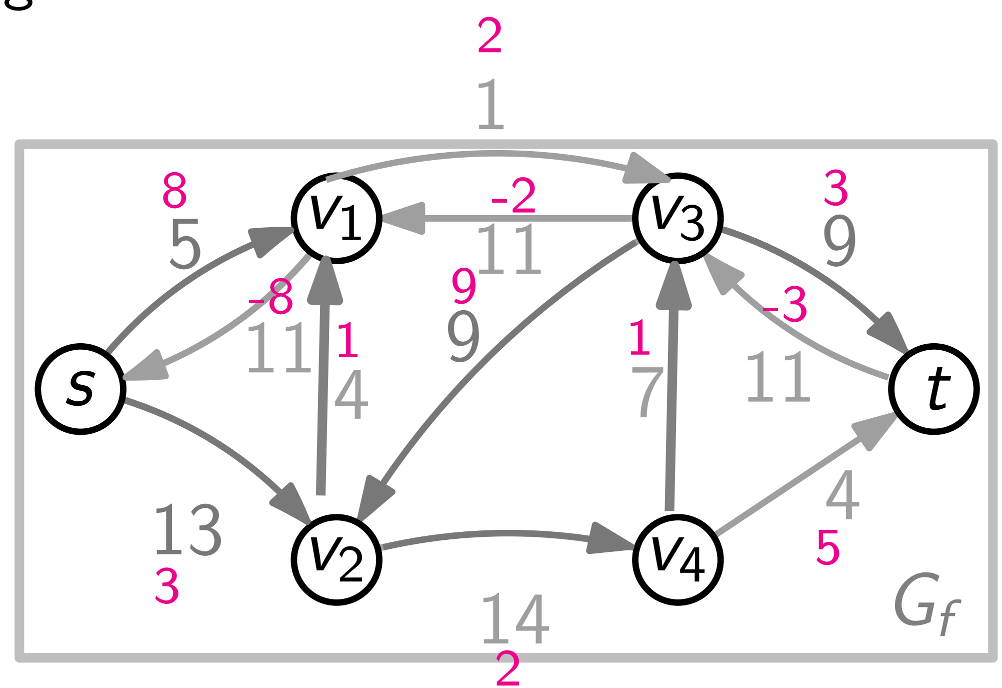
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss



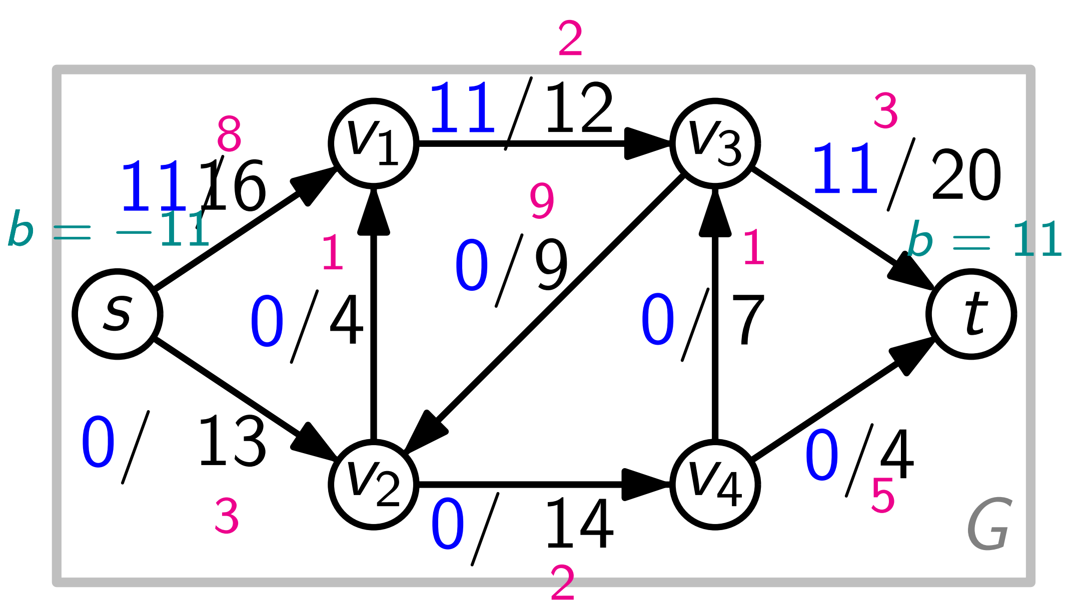


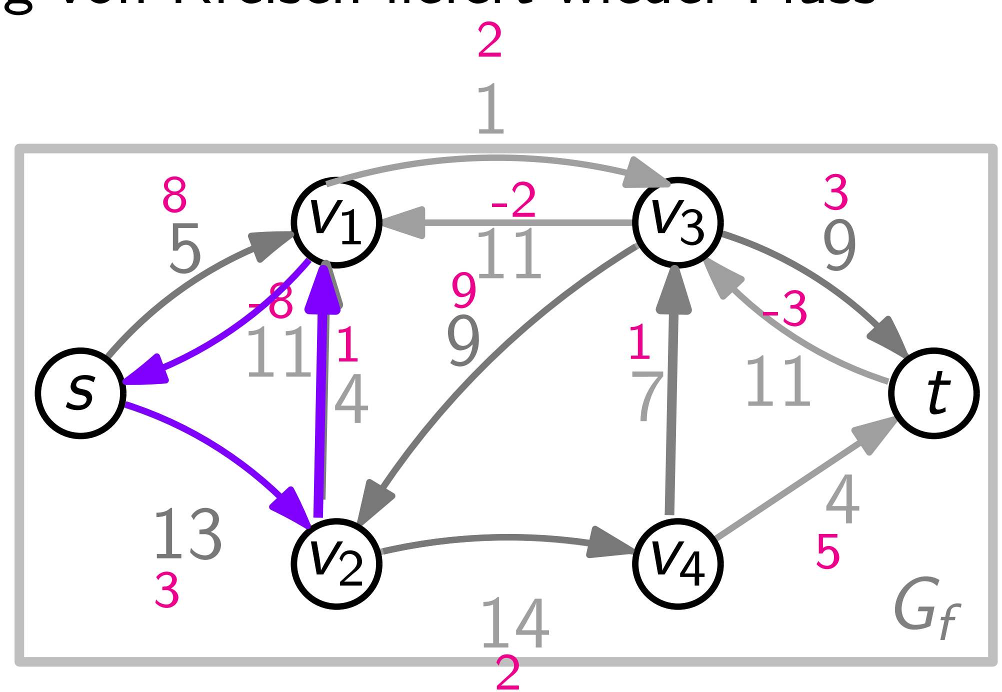
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss



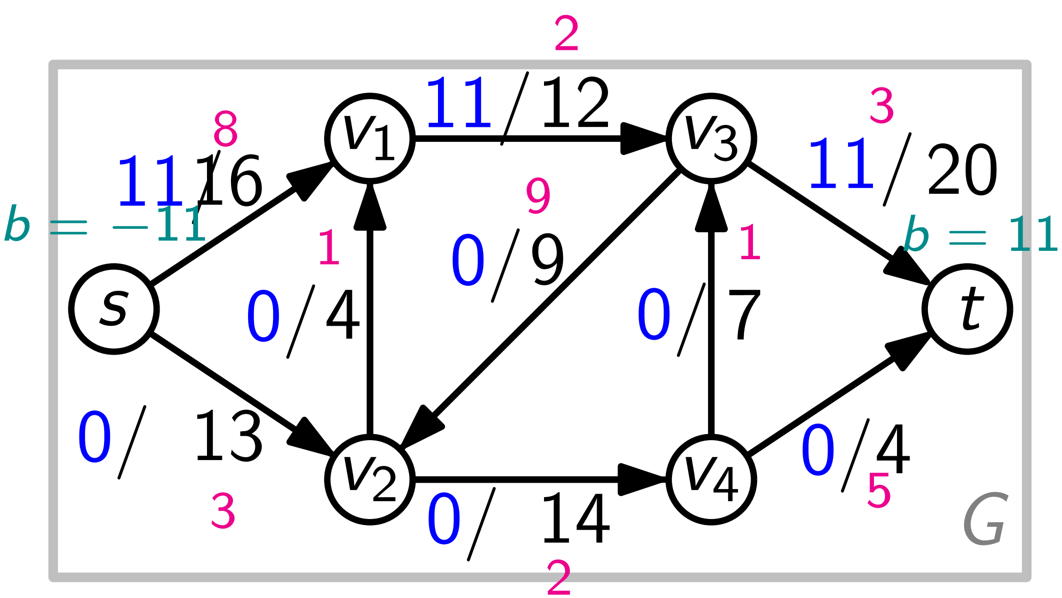


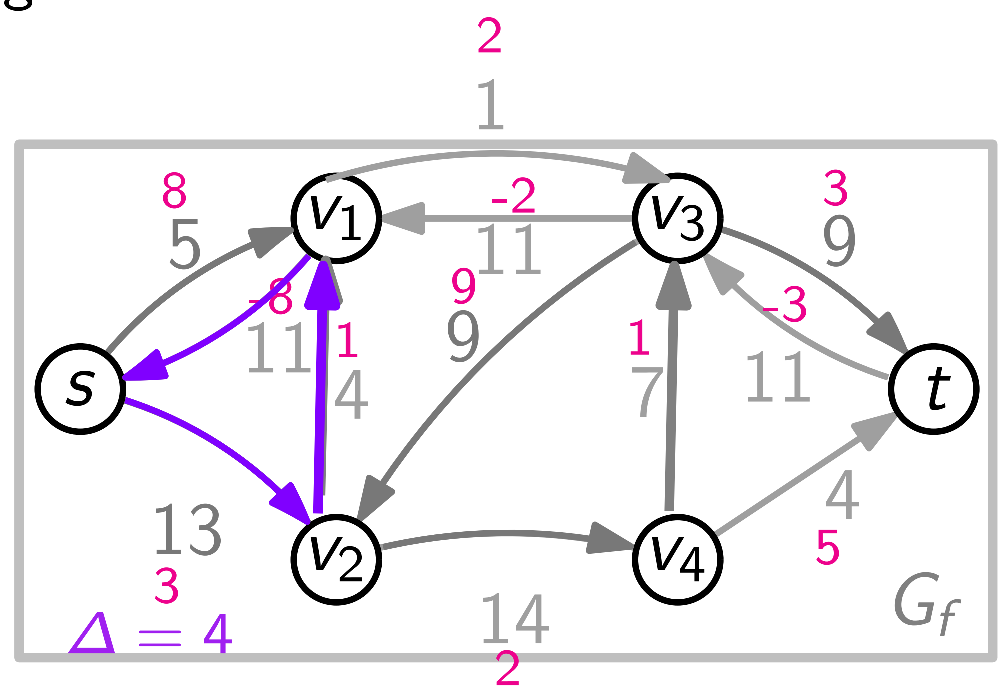
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss



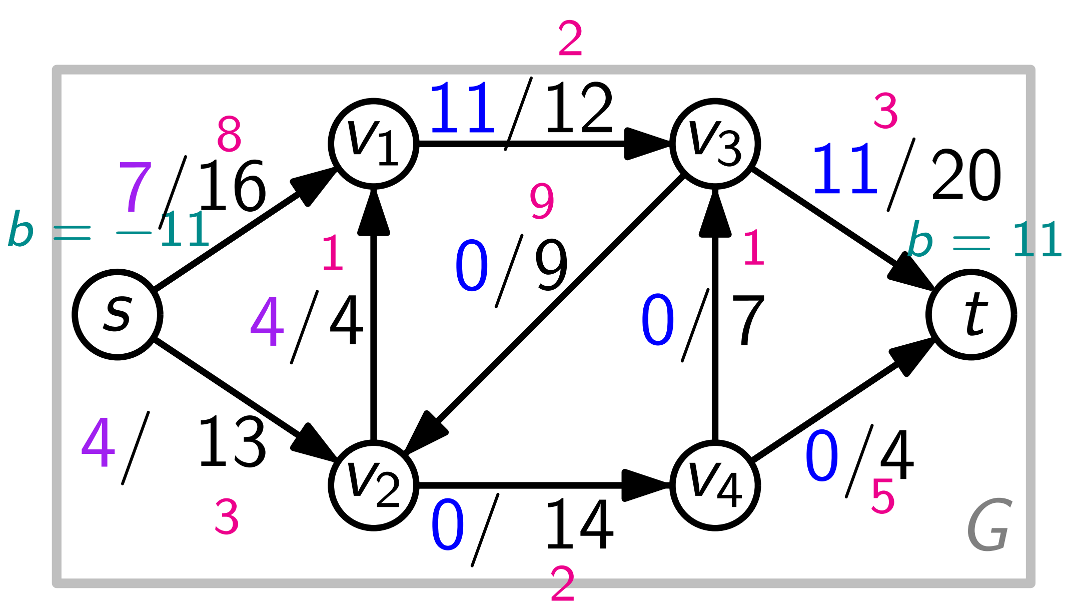


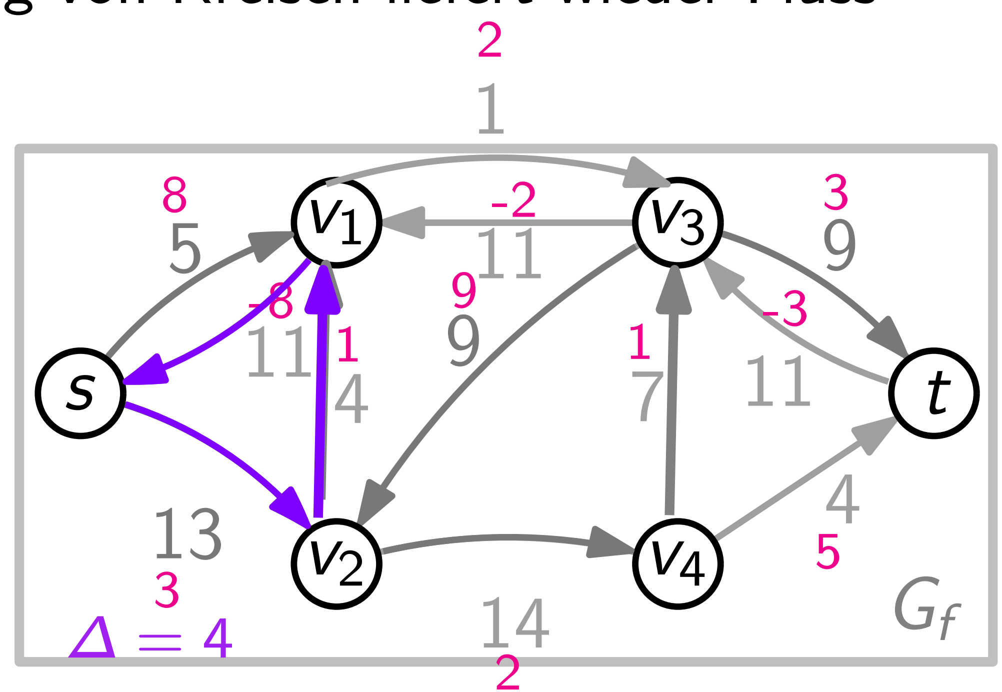
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u, v)) = k((u, v))$ für $(u, v) \in E$
-$k_f((u, v)) = -k((v, u))$ für $(v, u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss



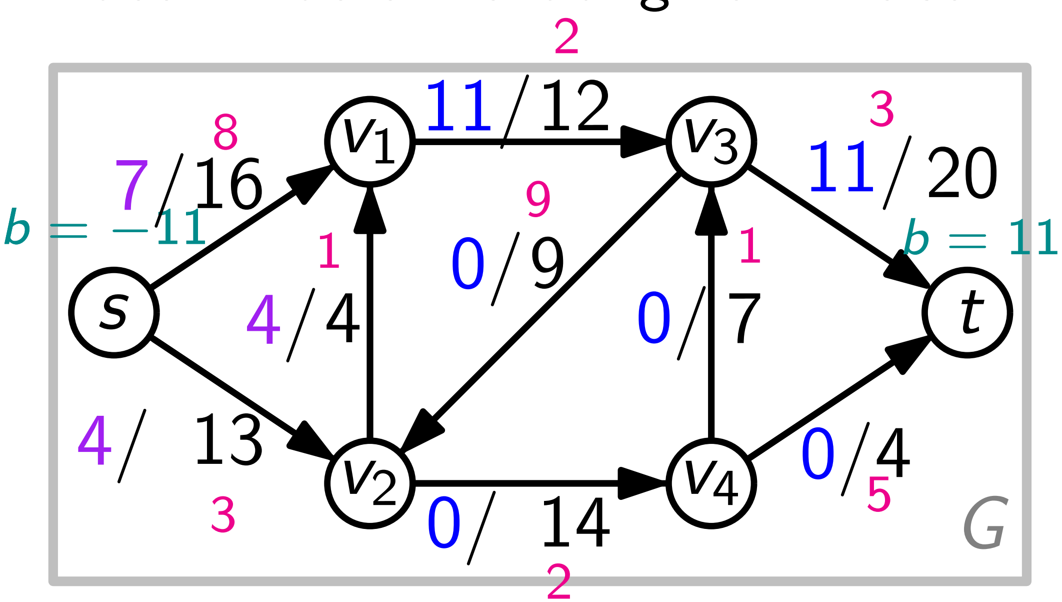


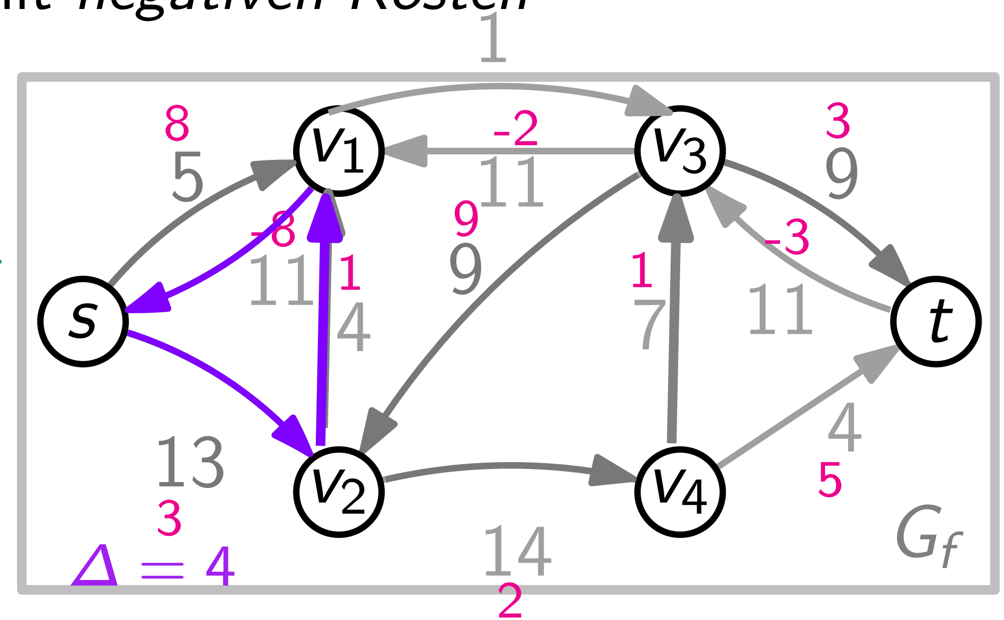
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten*



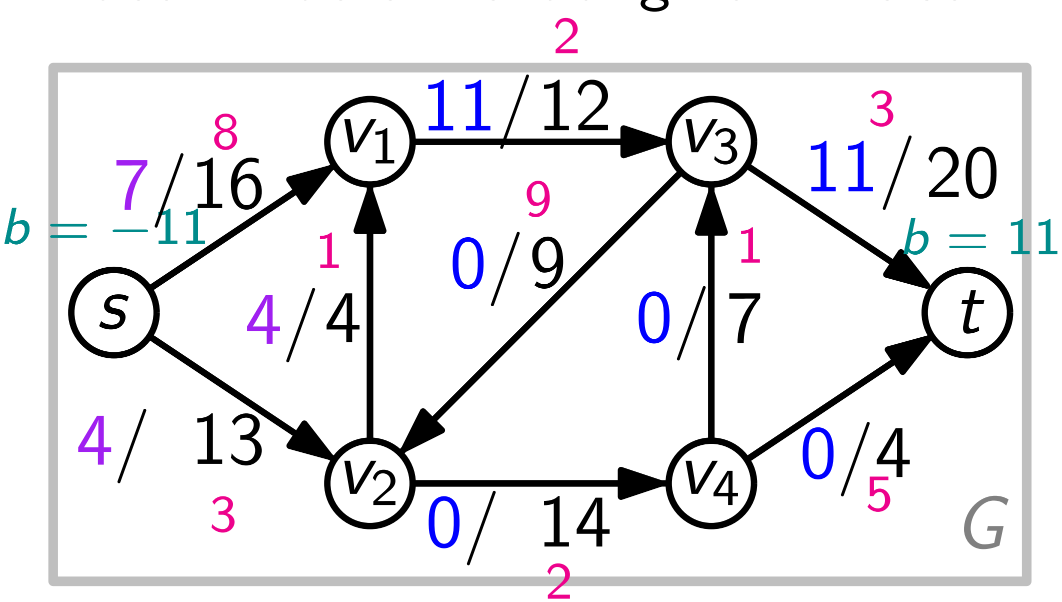


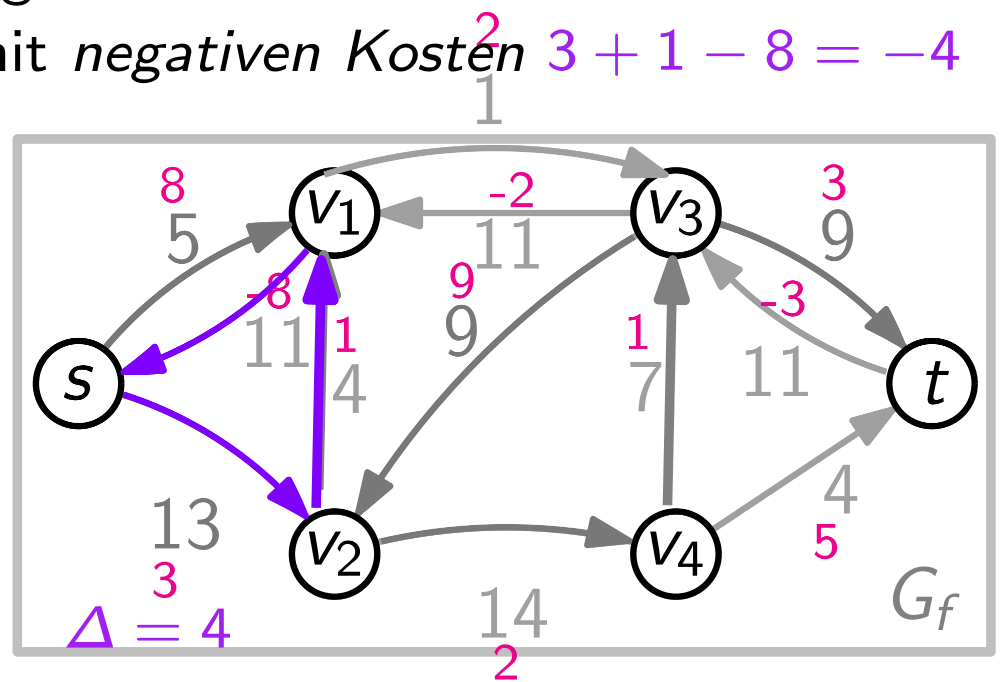
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten* $3 + 1 - 8 = -4$



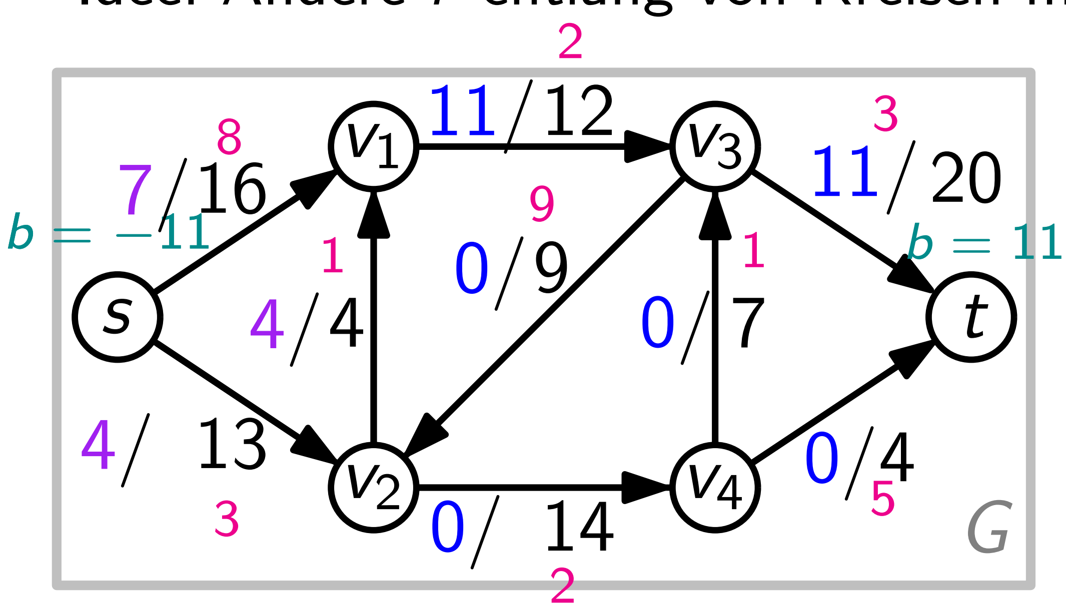


Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u, v)) = k((u, v))$ für $(u, v) \in E$
-$k_f((u, v)) = -k((v, u))$ für $(v, u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten*

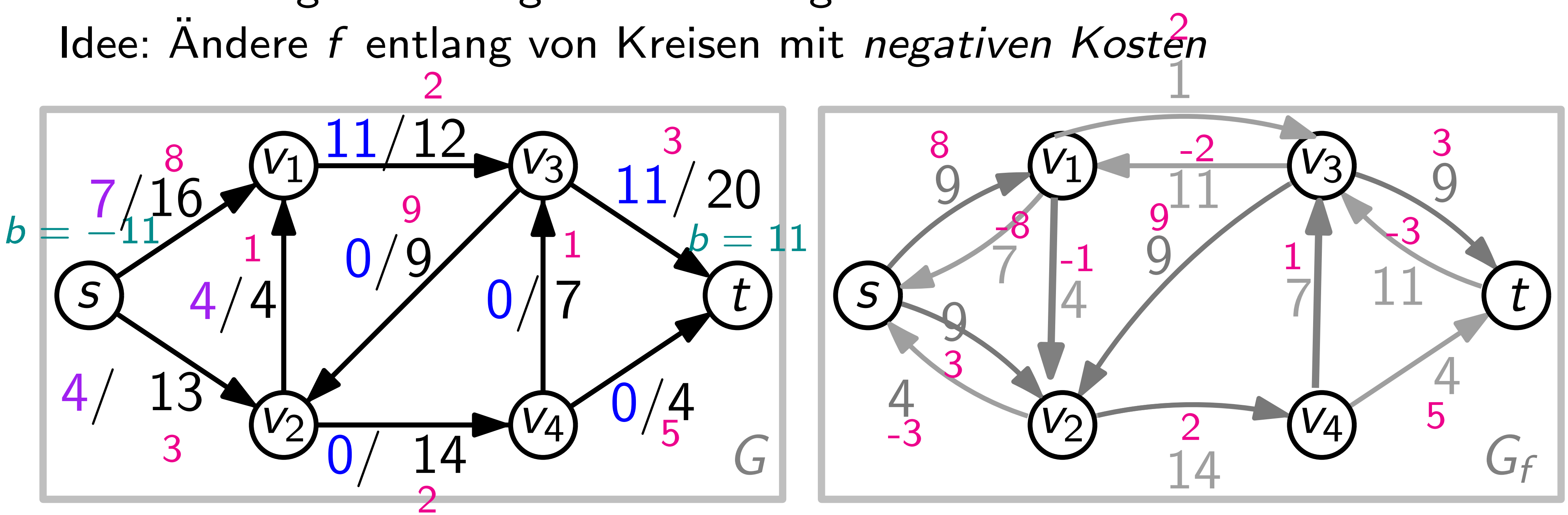


Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
-$k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
-$k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten*



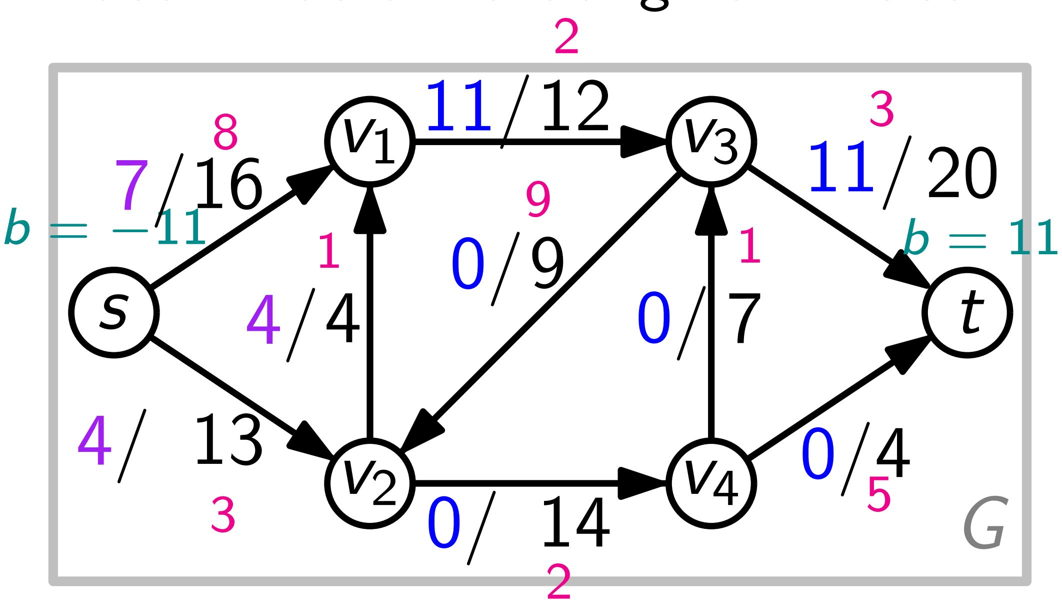
Verallgemeinerung: kostenminimale Flüsse

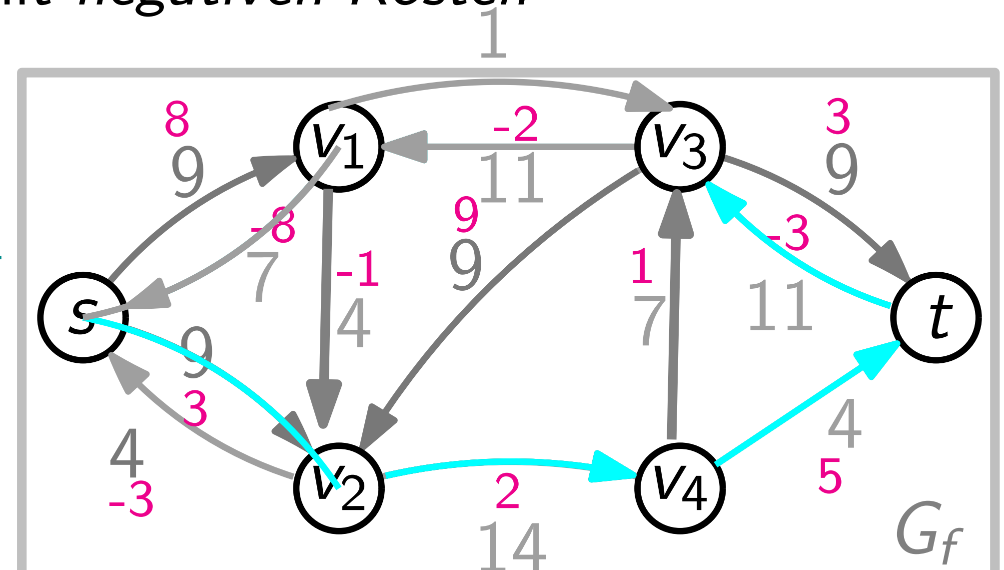
Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
- $k_f((u, v)) = k((u, v))$ für $(u, v) \in E$
- $k_f((u, v)) = -k((v, u))$ für $(v, u) \in E$

$3 + 2 + 5 - 3 - 2 - 8 = -3$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten*





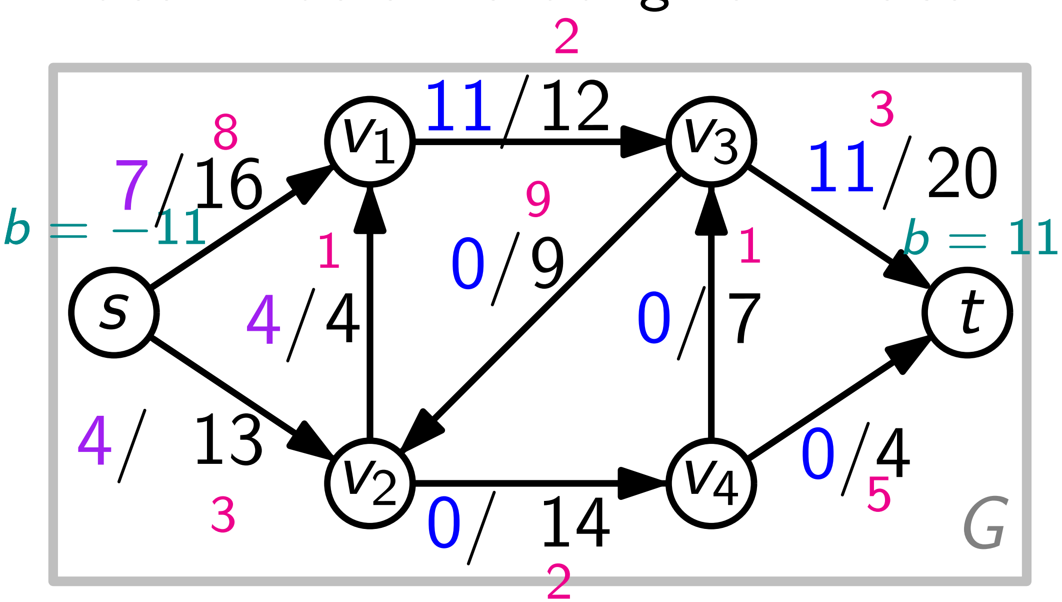
Verallgemeinerung: kostenminimale Flüsse

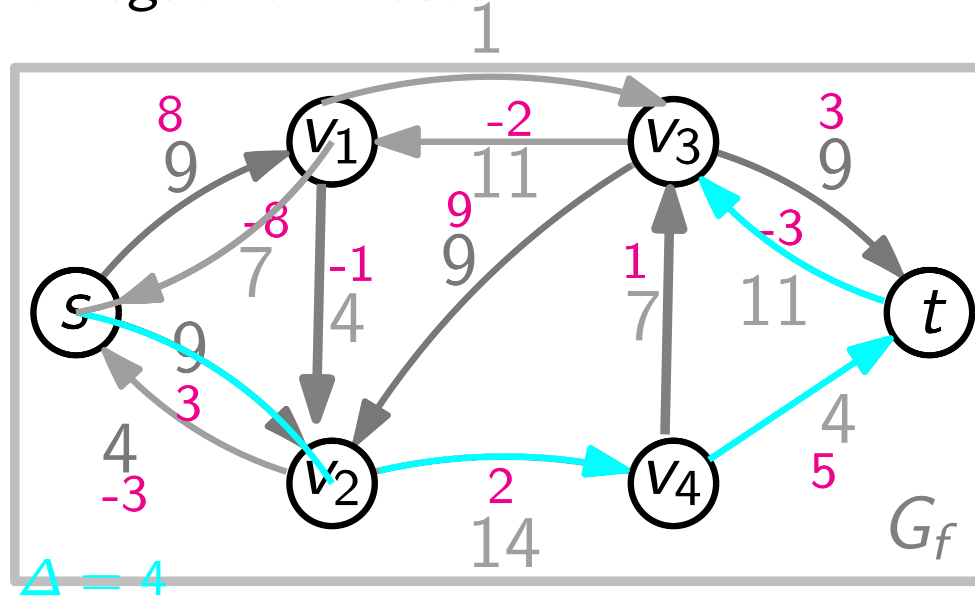
Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e)$, $c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
- $k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
- $k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

$3 + 2 + 5 - 3 - 2 - 8 = -3$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten*





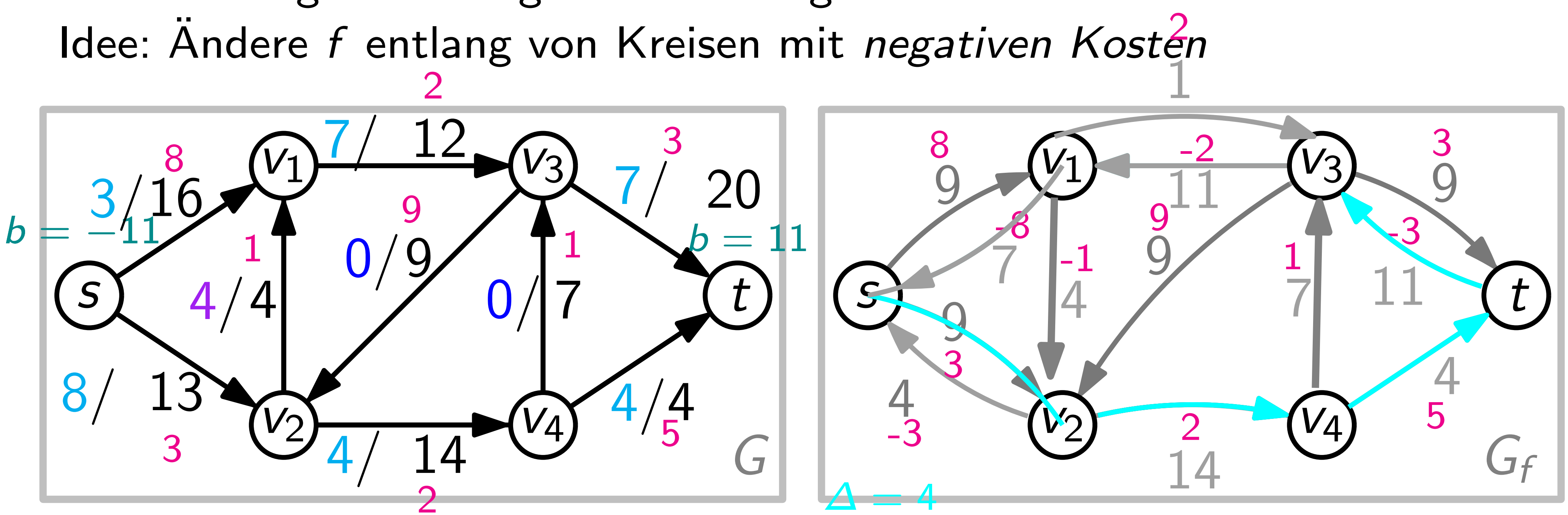
Verallgemeinerung: kostenminimale Flüsse

Kostenmininmaler Fluss
Gegeben: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$.
Gesucht: zulässiger b-Fluss f, mit minimalen Kosten $\sum_{e \in E} k(e) \cdot f(e)$.

Für gegebenen Fluss f: Erstelle Residualgraph G_f mit Kosten
- $k_f((u,v)) = k((u,v))$ für $(u,v) \in E$
- $k_f((u,v)) = -k((v,u))$ für $(v,u) \in E$

$3 + 2 + 5 - 3 - 2 - 8 = -3$

Beobachtung: Änderung von f entlang von Kreisen liefert wieder Fluss
Idee: Ändere f entlang von Kreisen mit *negativen Kosten*



Algorithmus von Klein

Require: gerichteter Graph $G = (V, E)$ mit oberen und unteren Kantenkapazitäten $l(e), c(e)$ für $e \in E$. Kantenkosten $k(e)$ für $e \in E$, Bedarfe $b(v)$ für $v \in V$

Ensure: Zulässiger b-Fluss mit minimalen Kosten

  Berechne einen zulässigen b-Fluss f

  while Residualnetzwerk G_f enthält einen negativen Kreis $C = (V_C, E_C)$

  do

    Setze $\Delta := \min_{e \in E_C} c_f(e)$

    for $(u, v) \in C$ **do**

      if $(u, v) \in E$ **then**

        $f(u, v) = f(u, v) + \Delta$

      else

        $f(u, v) = f(u, v) - \Delta$

      end if

    end for

  end while

  return f

Terminiert bei ganzzahligen Kapazitäten, Kosten und Bedarfen mit kostenminimalem b-Fluss

Stichworte heute

Konzepte: Flüsse, b-Flüsse, Schnitte, *Max-Flow-Min-Cut Theorem*

Optimierungsprobleme: maximaler Fluss, kostenminimaler Schnitt

Modellierung: Hilfsgüterlieferung

Algorithmen: Ford-Fulkerson, Edmmonds-Karp, Algorithmus von Klein