Karlsruher Institut für Technologie

Algorithmen II
Vorlesung am 23.10.2012

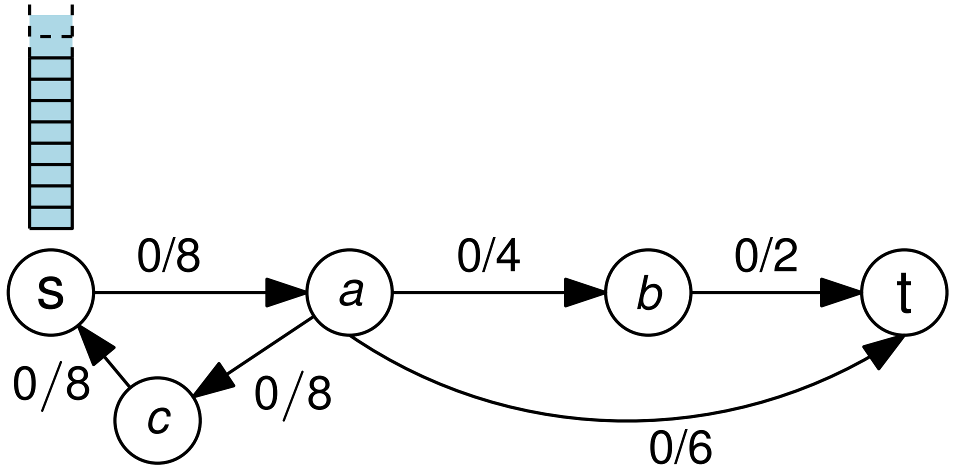
INSTITUT FÜR THEORETISCHE INFORMATIK · PROF. DR. DOROTHEA WAGNER

Flussalgorithmus von Goldberg und Tarjan (1988)

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Grundprinizip

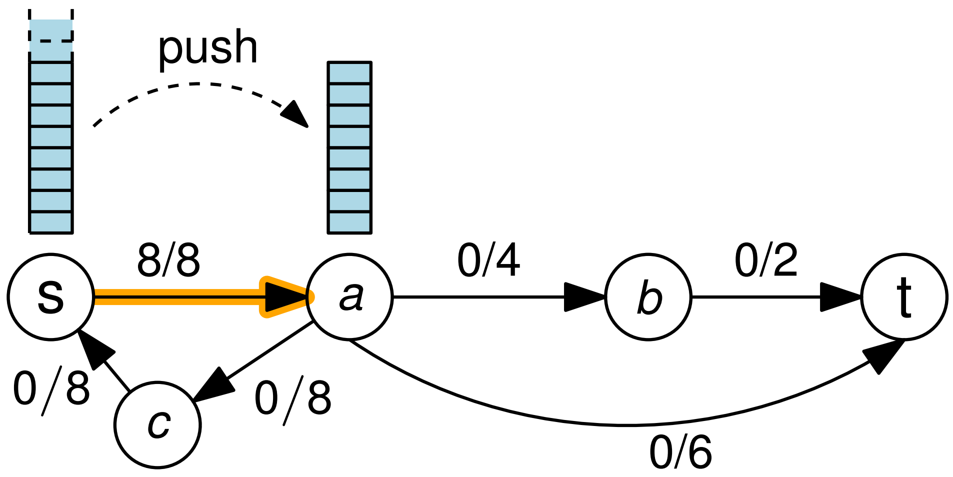
- Basiert nicht auf erhöhenden Wegen sondern auf zwei Operationen: PUSH und RELABEL.
- Fluss in Zwischenschritten nicht unbedingt gültig (Flusserhaltung nicht garantiert).



- Im Beispiel nur PUSH-Operation veranschaulicht: Drückt den Fluss zum nächsten Knoten und erzeugt dort gegebenfalls Überschuss, der zurück geführt werden muss.
- RELABEL-Operation garantiert, dass Fluss in die richtige Richtung gedrückt wird.
  - im Beispiel darf der Fluss nicht von *a* über *c* nach *s* zurück gedrückt werden.

Grundprinizip

- Basiert nicht auf erhöhenden Wegen sondern auf zwei Operationen: PUSH und RELABEL.
- Fluss in Zwischenschritten nicht unbedingt gültig (Flusserhaltung nicht garantiert).

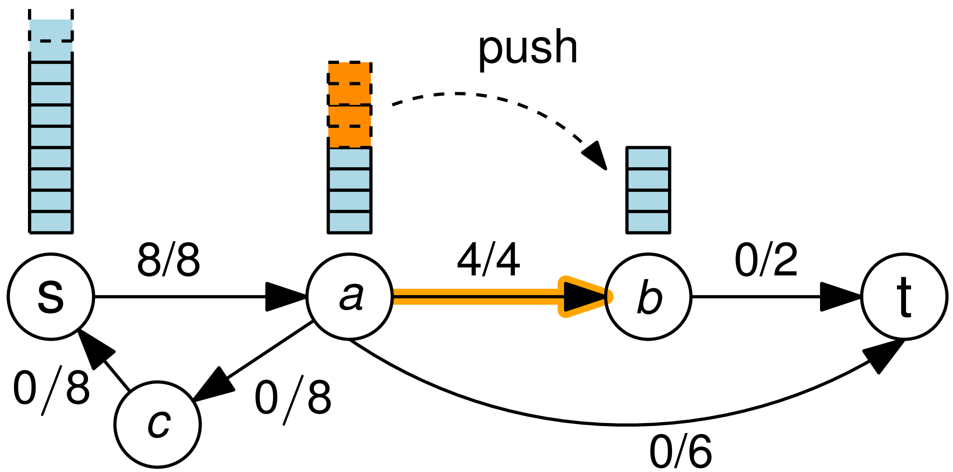


- Im Beispiel nur PUSH-Operation veranschaulicht: Drückt den Fluss zum nächsten Knoten und erzeugt dort gegebenfalls Überschuss, der zurück geführt werden muss.
- RELABEL-Operation garantiert, dass Fluss in die richtige Richtung gedrückt wird.
  - im Beispiel darf der Fluss nicht von *a* über *c* nach *s* zurück gedrückt werden.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Grundprinizip

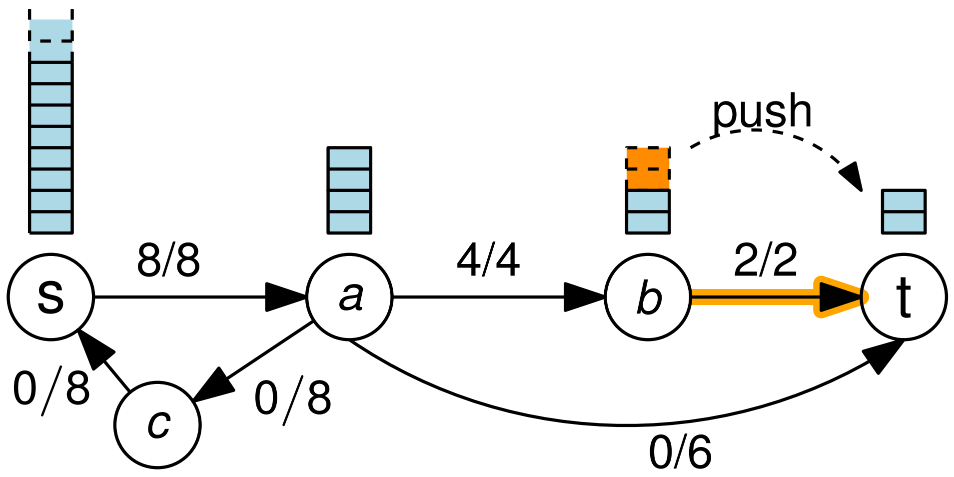
- Basiert nicht auf erhöhenden Wegen sondern auf zwei Operationen: PUSH und RELABEL.
- Fluss in Zwischenschritten nicht unbedingt gültig (Flusserhaltung nicht garantiert).



- Im Beispiel nur PUSH-Operation veranschaulicht: Drückt den Fluss zum nächsten Knoten und erzeugt dort gegebenfalls Überschuss, der zurück geführt werden muss.
- RELABEL-Operation garantiert, dass Fluss in die richtige Richtung gedrückt wird.
  - im Beispiel darf der Fluss nicht von *a* über *c* nach *s* zurück gedrückt werden.

Grundprinizip

- Basiert nicht auf erhöhenden Wegen sondern auf zwei Operationen: PUSH und RELABEL.
- Fluss in Zwischenschritten nicht unbedingt gültig (Flusserhaltung nicht garantiert).

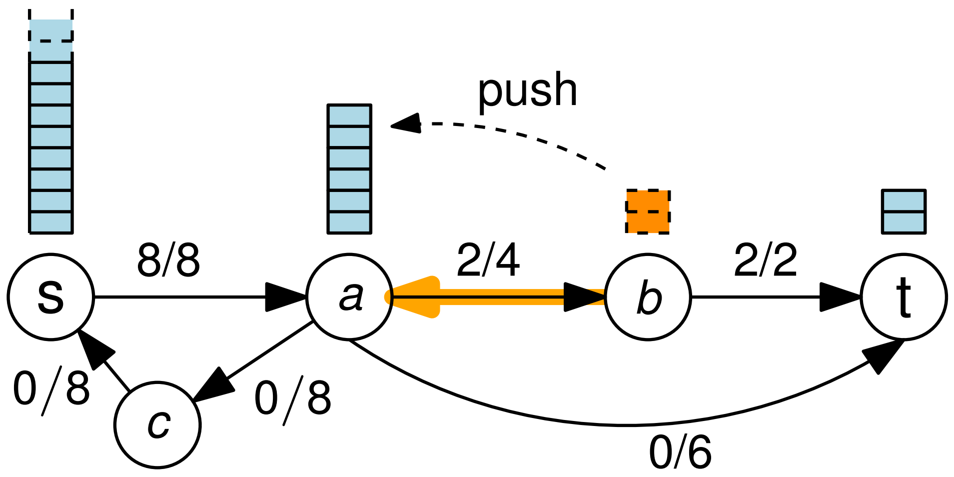


- Im Beispiel nur PUSH-Operation veranschaulicht: Drückt den Fluss zum nächsten Knoten und erzeugt dort gegebenfalls Überschuss, der zurück geführt werden muss.
- RELABEL-Operation garantiert, dass Fluss in die richtige Richtung gedrückt wird.
  - im Beispiel darf der Fluss nicht von *a* über *c* nach *s* zurück gedrückt werden.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Grundprinizip

- Basiert nicht auf erhöhenden Wegen sondern auf zwei Operationen: Push und Relabel.
- Fluss in Zwischenschritten nicht unbedingt gültig (Flusserhaltung nicht garantiert).

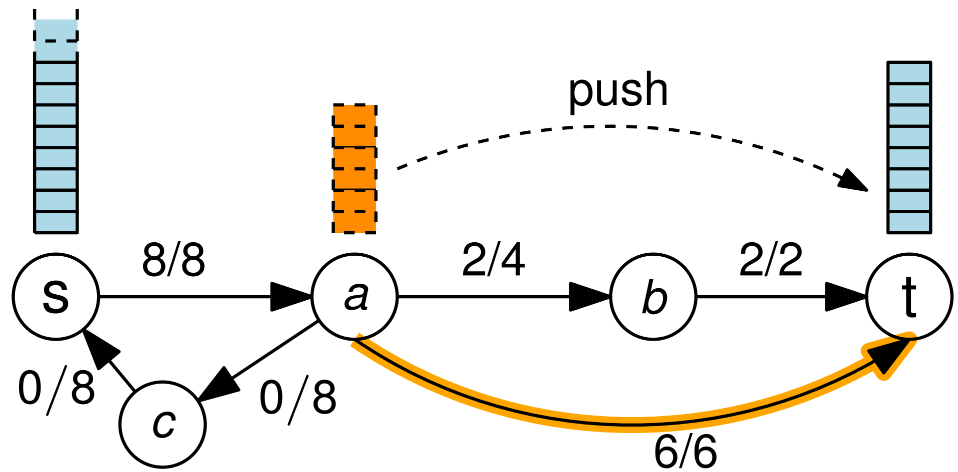


- Im Beispiel nur Push-Operation veranschaulicht: Drückt den Fluss zum nächsten Knoten und erzeugt dort gegebenfalls Überschuss, der zurück geführt werden muss.
- Relabel-Operation garantiert, dass Fluss in die richtige Richtung gedrückt wird.
  - im Beispiel darf der Fluss nicht von *a* über *c* nach *s* zurück gedrückt werden.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Grundprinizip

- Basiert nicht auf erhöhenden Wegen sondern auf zwei Operationen: PUSH und RELABEL.
- Fluss in Zwischenschritten nicht unbedingt gültig (Flusserhaltung nicht garantiert).



- Im Beispiel nur PUSH-Operation veranschaulicht: Drückt den Fluss zum nächsten Knoten und erzeugt dort gegebenfalls Überschuss, der zurück geführt werden muss.
- RELABEL-Operation garantiert, dass Fluss in die richtige Richtung gedrückt wird.
  - im Beispiel darf der Fluss nicht von *a* über *c* nach *s* zurück gedrückt werden.

Anpassung

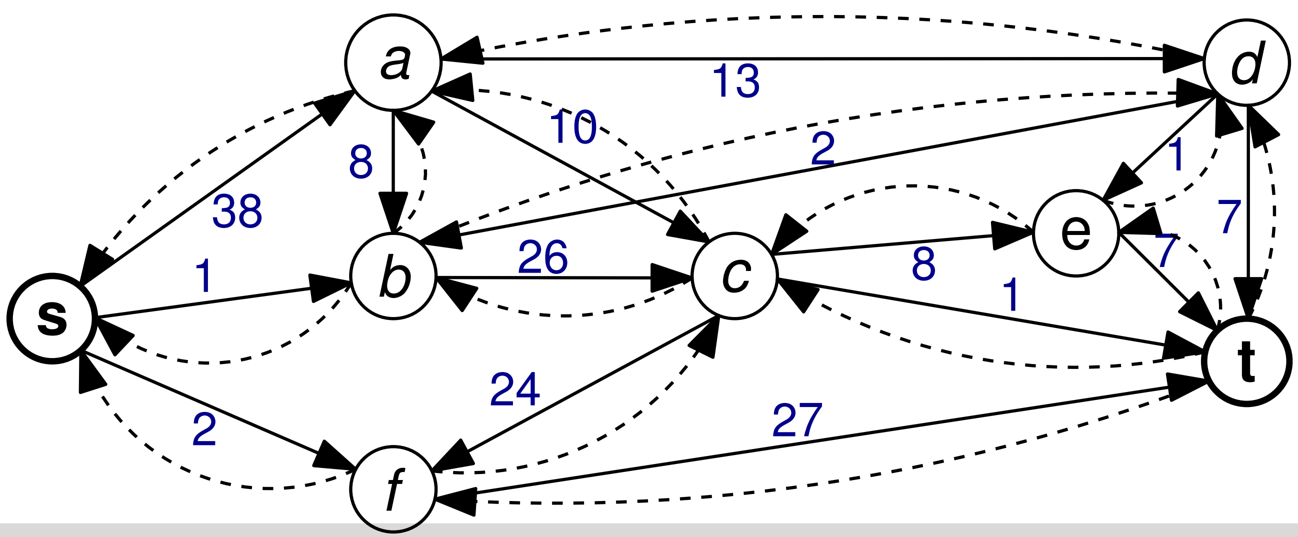
Gegeben: Netzwerk (D, s, t, c)

Erweitere c von $c\colon E \to \mathbb{R}_0^+$ auf $c\colon V \times V \to \mathbb{R}_0^+$, indem

$$c(v, w) := \begin{cases} \text{bisheriger Wert} & (v, w) \in E \\ 0 & (v, w) \notin E \end{cases}$$

Konstruiere aus $D = (V, E)$ neuen Graphen $D' = (V, E')$:

$$E' := E \cup \{(v, w) \in V \times V \mid (w, v) \in E \text{ und } (v, w) \notin E\}$$



Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Erweiterte Flussdefinition

Definition: Gegeben ein Netzwerk (D,s,t,c) mit angepasster Gewichtsfunktion, dann ist ein *Fluss* eine Abbildung $f\colon V \times V \to \mathbb{R}$ mit

1. *Kapazitätsbedingung:* für alle $(v, w) \in V \times V$ gilt $f(v, w) \leq c(v, w)$
2. *Antisymmetrie:* für alle $(v, w) \in V \times V$ gilt $f(v, w) = -f(w, v)$
3. *Flusserhaltung:* für alle $v \in V \setminus \{s, t\}$ gilt $\sum_{u \in V} f(u, v) = 0$

Wert eines Flusses: $w(f) = \sum_{v \in V} f(s, v) = \sum_{v \in V} f(v, t)$

Definition: Ein *Präfluss* ist eine Abbildung $f\colon V \times V \to \mathbb{R}$, welche die Kapazitätbedingung und die Antisymmetriebedingung erfüllt sowie

$$\text{für alle } v \in V \setminus \{s\} \sum_{u \in V} f(u, v) \geq 0$$

Flussüberschuss und Restkapazität

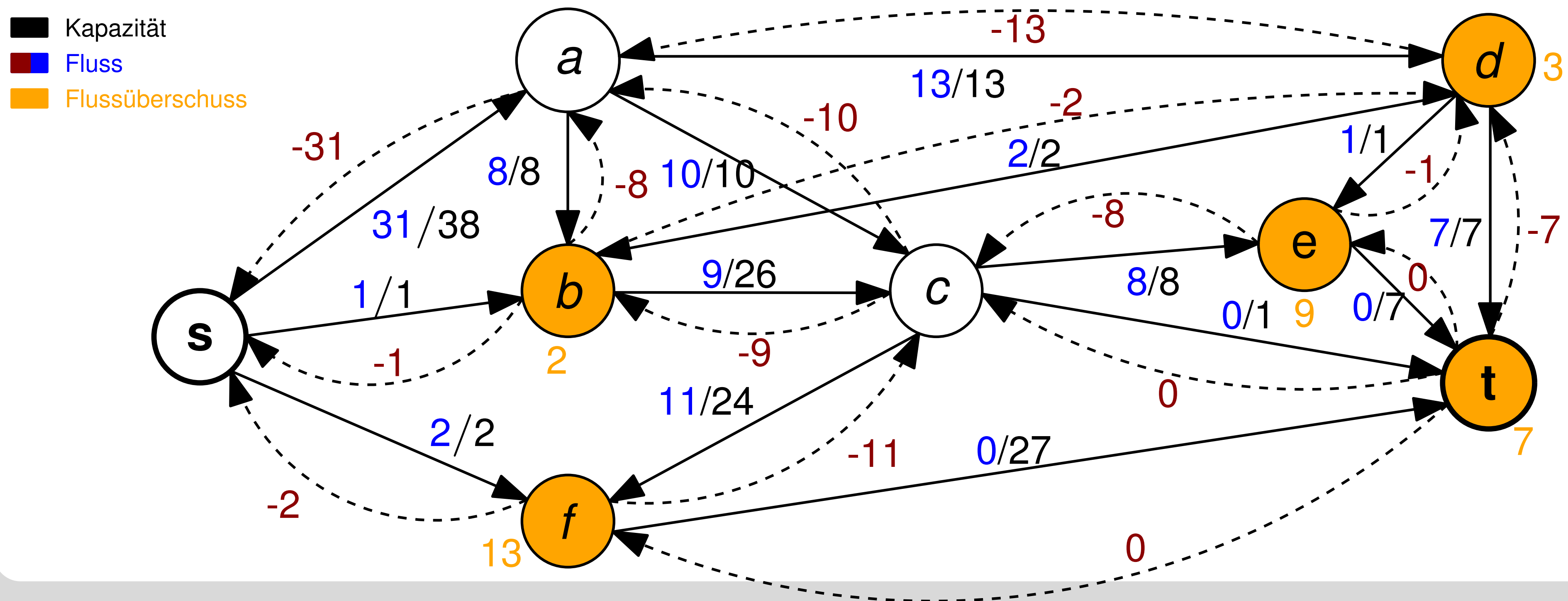
Definition: Sei f ein Präfluss. Für $v \in V \setminus \{t\}$ heißt der Wert

$$e(v) := \sum_{u \in V} f(u, v)$$

Flussüberschuss und die Abbildung $r_f \colon E' \to \mathbb{R}$ mit

$$\forall (u, v) \in E' \; r_f(u, v) := c(u, v) - f(u, v)$$

heißt *Restkapazität*.



Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

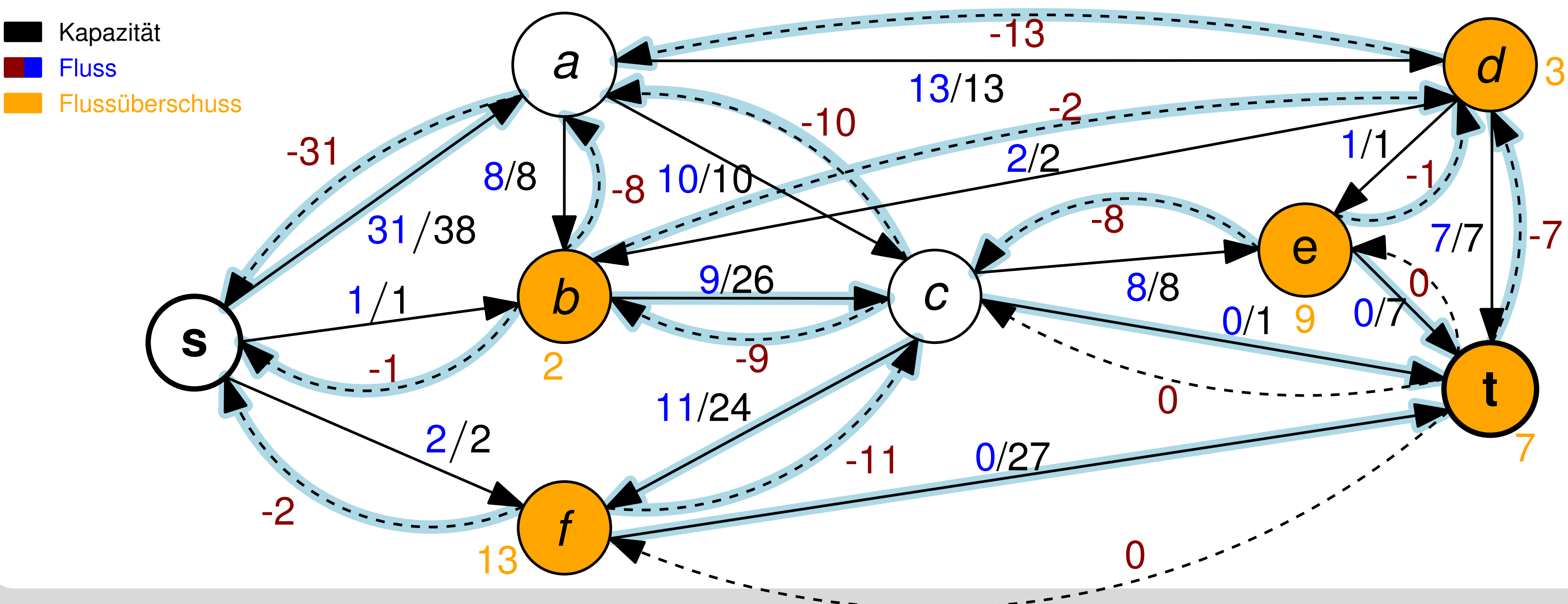
Residualgraph

Karlsruher Institut für Technologie

Definition: Eine Kante $(u, v) \in E'$ heißt *Residualkante* bezüglich eines Präflusses f, falls $r_f(u, v) > 0$.

Der *Residualgraph* zu f ist gegeben durch $D_f(V, E_f)$ mit $E_f := \{(u, v) \in E' \mid r_f(u, v) > 0\}$

Erinnerung: $r_f(u, v) := c(u, v) - f(u, v)$



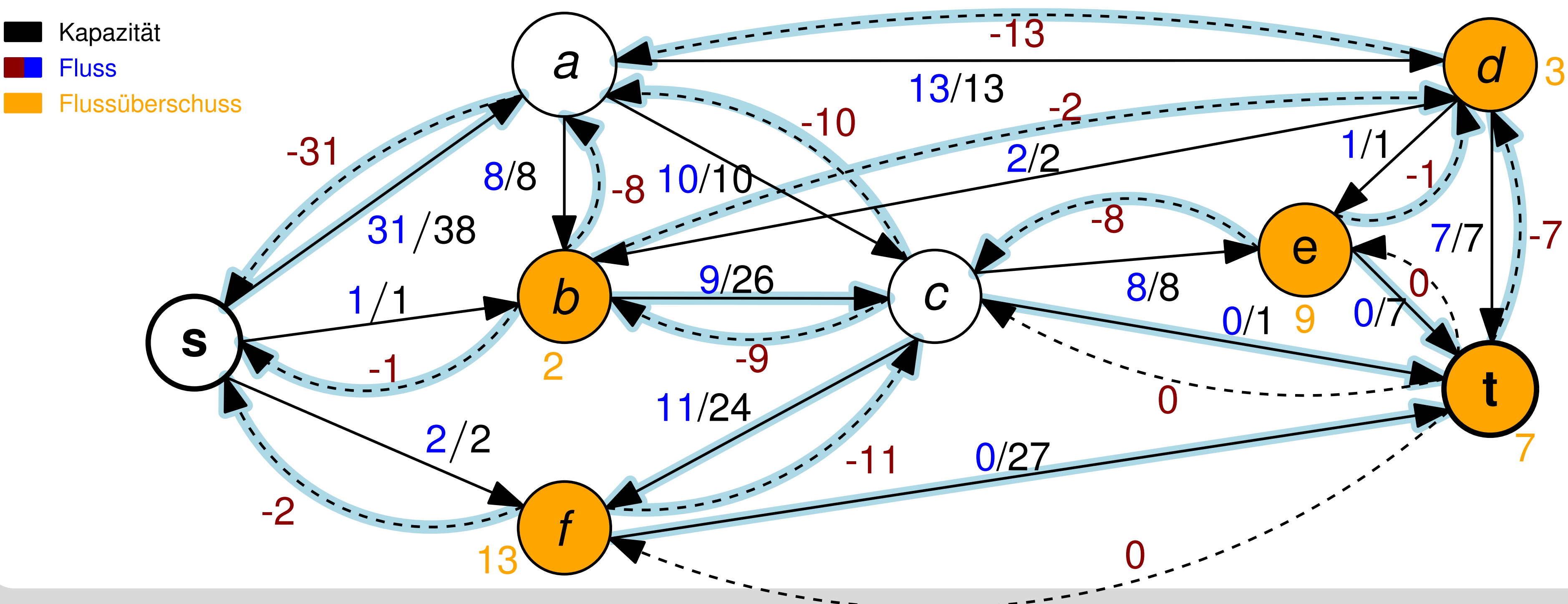
Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Residualgraph

Karlsruher Institut für Technologie

Definition: Eine Kante $(u, v) \in E$ heißt

- *nicht saturiert*, falls $0 \leq f(u,v) < c(u,v)$, und
- *nicht leer*, falls $0 < f(u,v) \leq c(u,v)$



Zulässige Markierung

Definition: Eine Abbildung $dist\colon V \to \mathbb{N}_0 \cup \{\infty\}$ heißt *zulässige Markierung* bzgl. eines Präflusses f, falls:

- $dist(s) = |V|$ und $dist(t) = 0$ und
- für alle $v \in V \backslash \{t\}$ und alle $(v, w) \in E_f$ gilt $dist(v) \leq dist(w) + 1$

Ein Knoten $v \in V \backslash \{t\}$ heißt *aktiv* im Laufe des Algorithmus, wenn $e(v) > 0$ und $dist(v) < \infty$.

Erinnerung: e(v) ist der Flussüberschuss von v.

- Zu Beginn wird $dist(s) := |V|$ und $dist(v) := 0$ für alle $v \in V \backslash \{s\}$ gesetzt.
- $dist(v)$ wird geändert, aber stets zulässig gehalten.
- Es gilt stets:
  - $dist(s) = |V|$
  - Falls $dist(v) < |V|$ für $v \in V$, so ist $dist(v)$ eine untere Schranke für den Abstand von v zu t im Residualgraph D_f.
  - Falls $dist(v) > |V|$, so ist t von v in D_f nicht erreichbar und $dist(v) - |V|$ ist untere Schranke für Abstand von v zu s in D_f.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Algorithmus von Goldberg und Tarjan

Karlsruher Institut für Technologie

Für alle $(v, w) \in V \times V$ mit $(v, w) \notin E$ setze:

- $c(v, w) \leftarrow 0$

Für alle $(v, w) \in V \times V$ setze:

- $f(v, w) \leftarrow 0$
- $r_f(v, w) \leftarrow c(v, w)$

Setze $dist(s) \leftarrow |V|$

Für alle $v \in V \backslash \{s\}$ setze:

- $f(s, v) \leftarrow c(s, v)$, $r_f(s, v) \leftarrow 0$
- $f(v, s) \leftarrow -c(s, v)$, $r_f(v, s) \leftarrow c(v, s) - f(v, s)$
- $dist(v) \leftarrow 0$
- $e(v) \leftarrow c(s, v)$

Eingabe: Netzwerk (D, s, t, c) mit $D = (V, E)$ und $c : E \rightarrow \mathbb{R}_0^+$
Ausgabe: Maximaler Fluss f.

Solange es aktiven Knoten gibt:

- Wähle beliebigen aktiven Knoten v.
- Führe für v eine zulässige Operation PUSH oder RELABEL aus.

PUSH und RELABEL

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

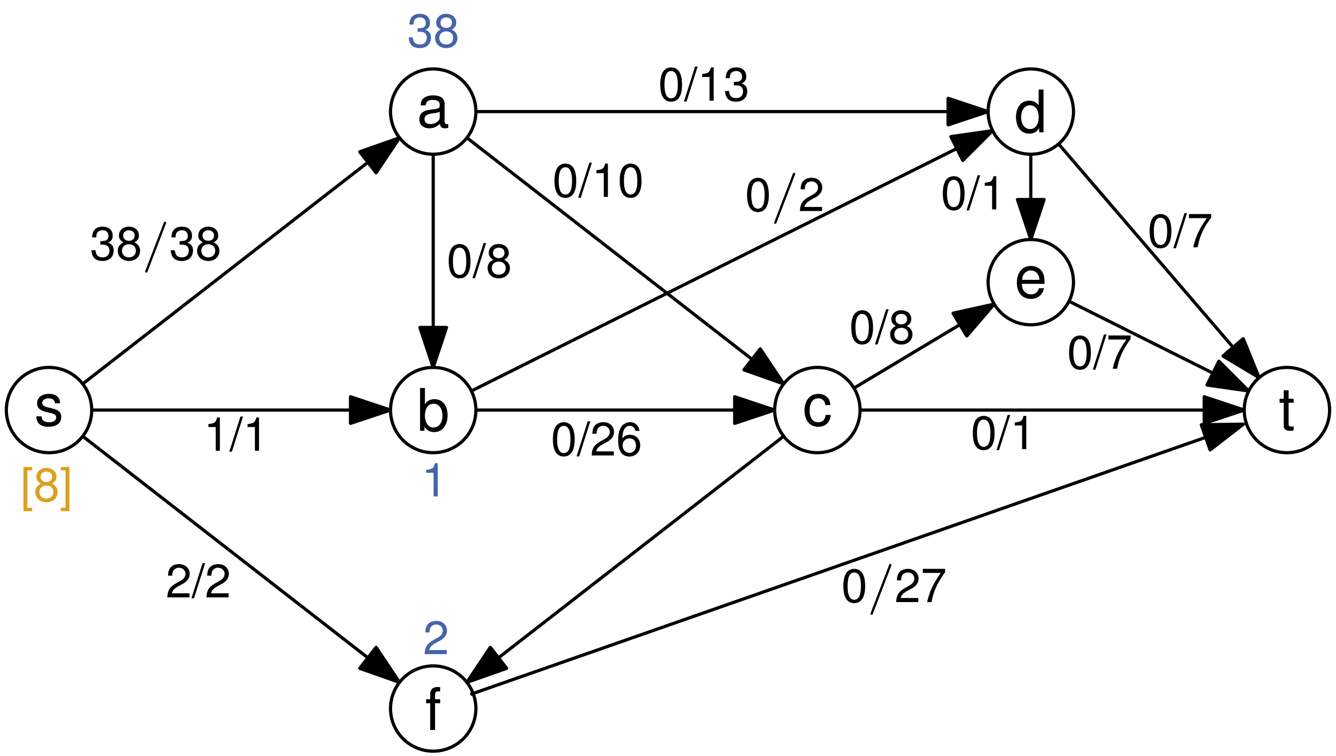
- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel



Ausgeführte Operation:
Initialisierung

Anstehende Operation:
RELABEL(a)

$dist(v)$ Überschuss $e(v)$

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

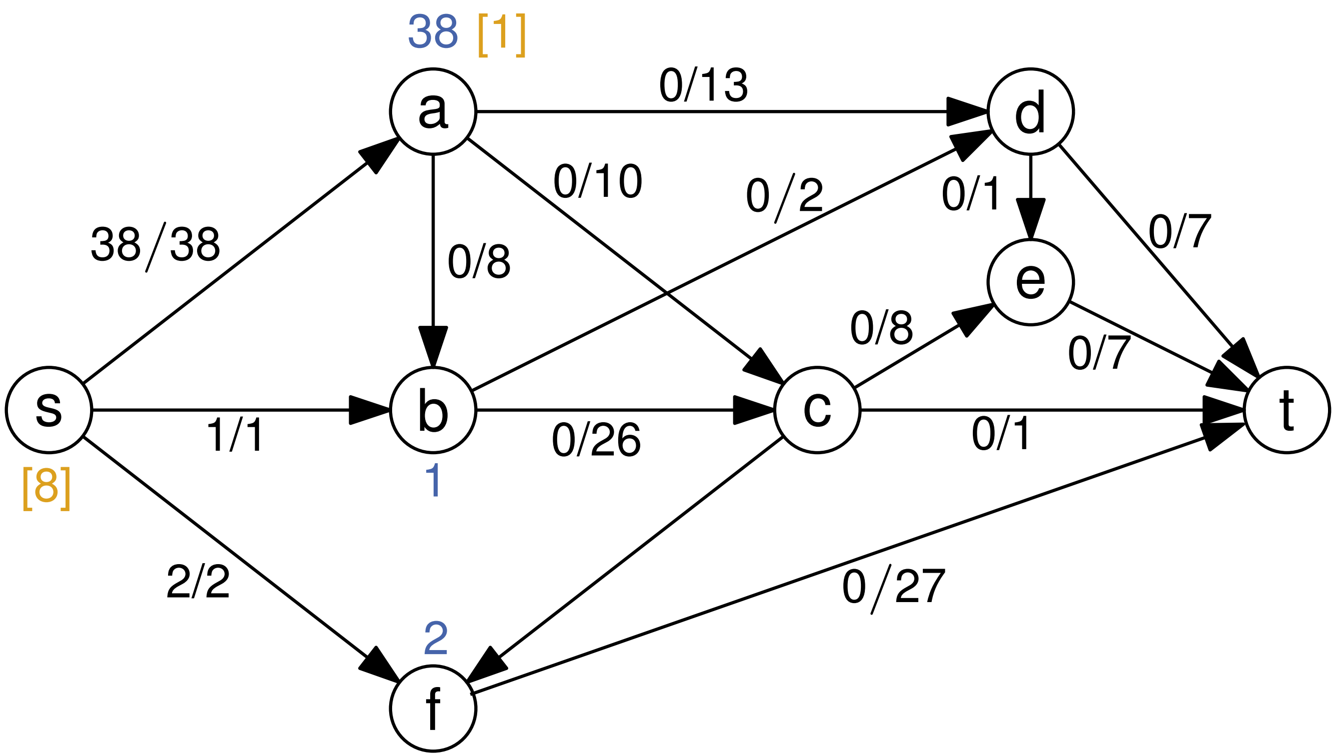
- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel



Ausgeführte Operationen:

Initialisierung

RELABEL(a)

Anstehende Operationen:

PUSH(a, b) mit $\Delta = 8$

PUSH(a, c) mit $\Delta = 10$

PUSH(a, d) mit $\Delta = 13$

dist(v) Überschuss $e(v)$

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

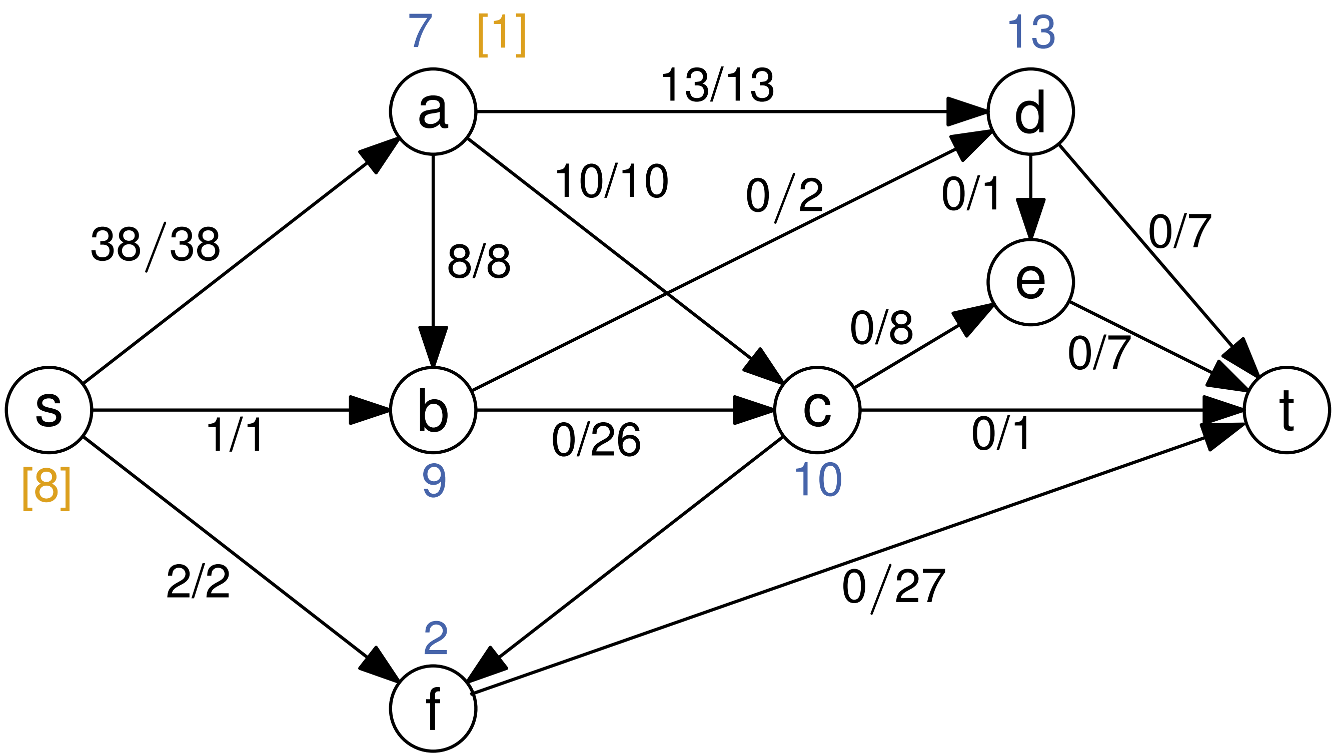
- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel



Ausgeführte Operationen:

Initialisierung

RELABEL(a)

PUSH(a, b) mit Δ = 8

PUSH(a, c) mit Δ = 10

PUSH(a, d) mit Δ = 13

Anstehende Operation:

RELABEL(a)

dist(v) Überschuss $e(v)$

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

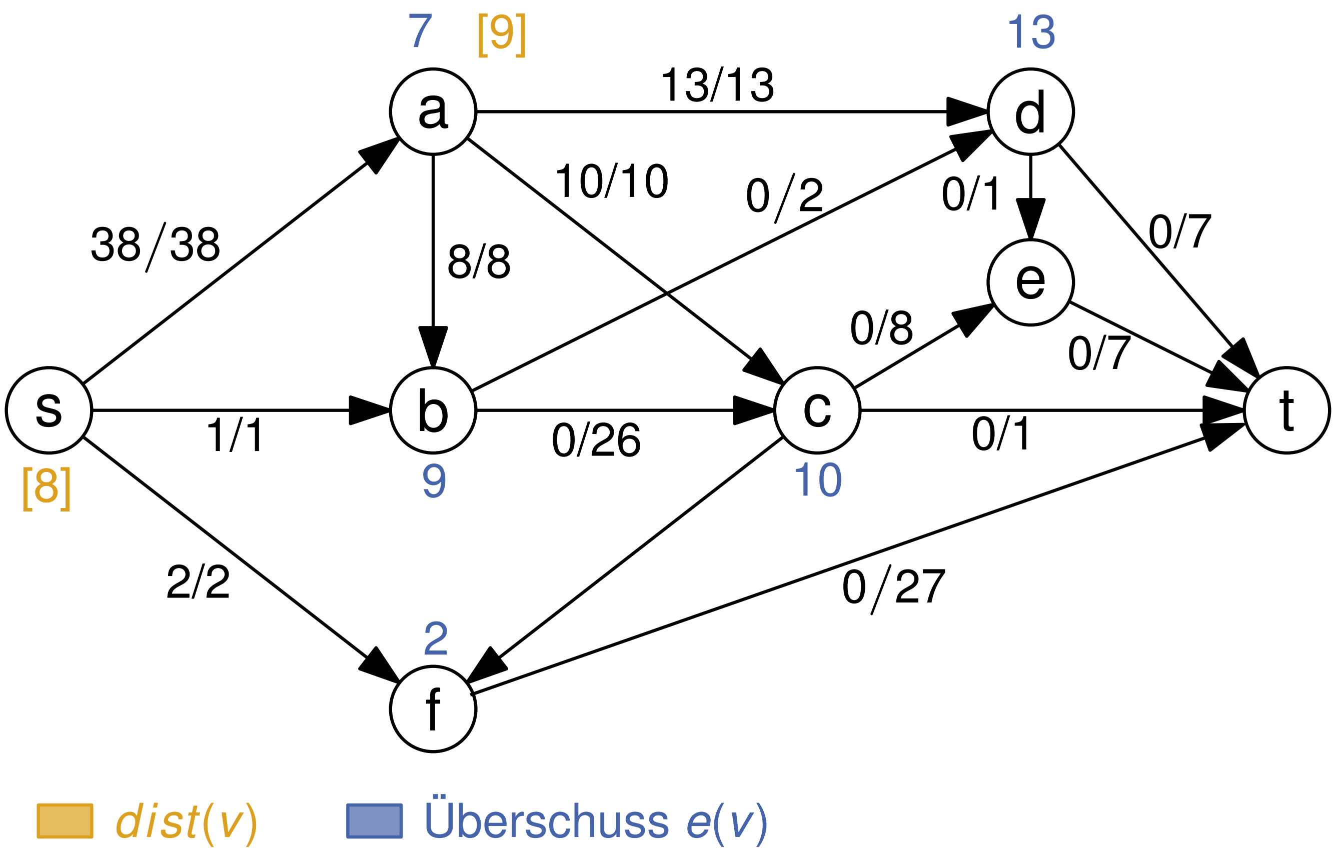
- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel



Ausgeführte Operationen:
Initialisierung
RELABEL(a)
PUSH(a, b) mit $\Delta = 8$
PUSH(a, c) mit $\Delta = 10$
PUSH(a, d) mit $\Delta = 13$
RELABEL(a)

Anstehende Operation:
PUSH(a, s) mit $\Delta = 7$

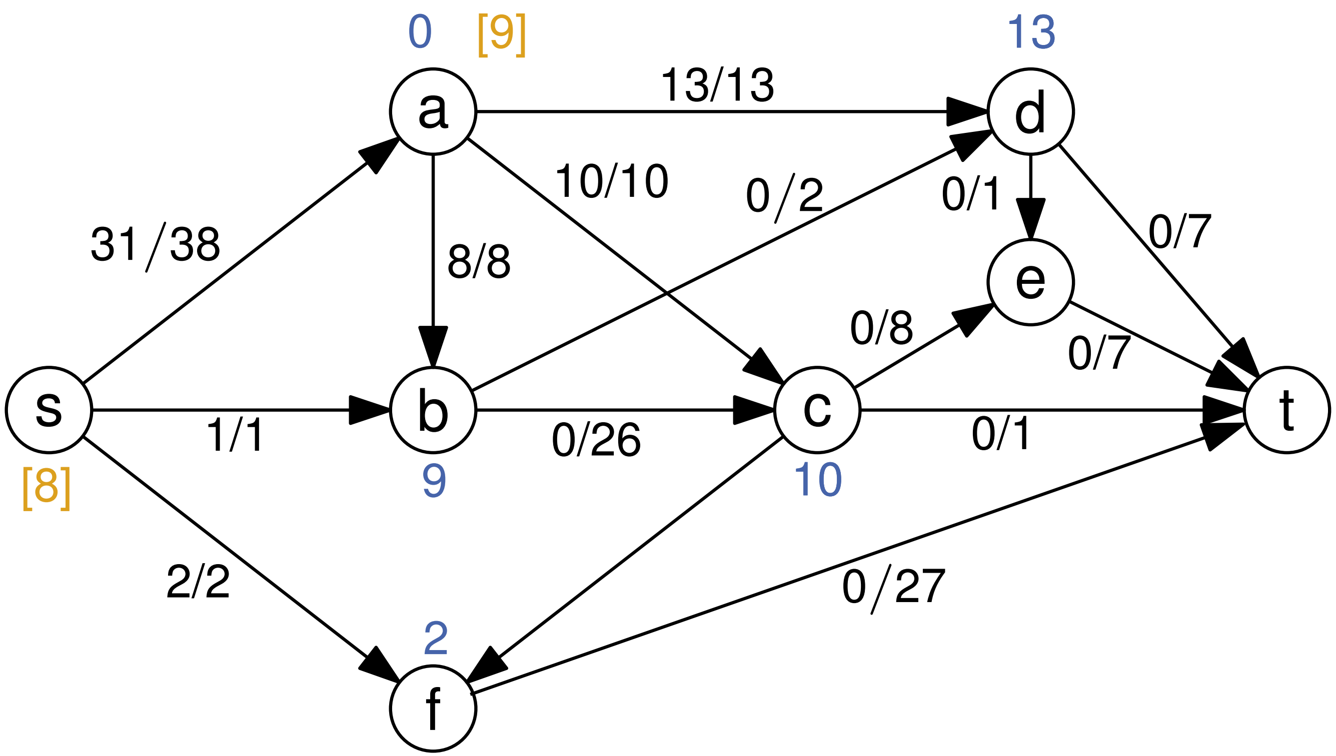
Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Beispiel



Ausgeführte Operationen:
Initialisierung
RELABEL(a)
PUSH(a, b) mit $\Delta = 8$
PUSH(a, c) mit $\Delta = 10$
PUSH(a, d) mit $\Delta = 13$
RELABEL(a)
PUSH(a, s) mit $\Delta = 7$

$dist(v)$ Überschuss $e(v)$

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Korrektheitsbeweis des Algorithmus von Goldberg und Tarjan

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beweisstruktur

1. Schritt: Wenn Algorithmus terminiert und die Markierungen endlich bleiben, dann ist das Ergebnis ein Maximalfluss f.

a) Solange aktiver Knoten vorhanden, kann Operation PUSH oder Operation RELABEL angewendet werden.

b) Invariante: f ist stets Präfluss und $dist$ ist stets bezüglich f zulässige Markierung.

c) t ist im Residualgraph D_f des Präflusses f von s aus nicht erreichbar, wenn $dist$ zulässig.

2. Schritt: Algorithmus terminiert und Markierungen bleiben endlich:

a) Finde obere Schranke für $dist$.

b) Finde obere Schranke für Anzahl Aufrufe von PUSH und RELABEL.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Zulässigkeit der Operationen

Lemma 4.20: Sei f ein Präfluss auf D, die Funktion $dist$ eine bezüglich f zulässige Markierung auf V und $v \in V$ ein aktiver Knoten. Dann ist entweder eine PUSH-Operation von v oder eine RELABEL-Operation von v zulässig.

Erinnerung:

Definition: Eine Abbildung $dist\colon V \to \mathbb{N}_0 \cup \{\infty\}$ heißt *zulässige Markierung* bzgl. eines Präflusses f, falls:

- $dist(s) = |V|$ und $dist(t) = 0$ und
- für alle $v \in V \backslash \{t\}$ und alle $(v, w) \in E_f$ gilt $dist(v) \leq dist(w) + 1$

Ein Knoten $v \in V \backslash \{t\}$ heißt *aktiv* im Laufe des Algorithmus, wenn $e(v) > 0$ und $dist(v) < \infty$.

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Präfluss und zulässige Markierung

Karlsruher Institut für Technologie

Lemma 4.21: Während des Ablaufs des Algorithmus von Goldberg und Tarjan ist f stets ein Präfluss und $dist$ stets eine bezüglich f zulässige Markierung.

Operation $\textsc{Push}(D, f, v, w)$
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Operation $\textsc{Relabel}(D, f, v, dist)$
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty\,, & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset\,, \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Erreichbarkeit der Senke im Residualgraph

Lemma 4.22: Sei f ein Präfluss und *dist* bezüglich f zulässig. Dann ist t im Residualgraph D_f von s aus nicht erreichbar (es gibt also keinen gerichteten s-t-Weg in D_f).

Erinnerung:

Definition: Eine Abbildung $dist\colon V \to \mathbb{N}_0 \cup \{\infty\}$ heißt *zulässige Markierung* bzgl. eines Präflusses, falls:

- $dist(s) = |V|$ und $dist(t) = 0$ und
- für alle $v \in V \setminus \{t\}$ und alle $(v, w) \in E_f$ gilt $dist(v) \leq dist(w) + 1$

Ein Knoten $v \in V \setminus \{t\}$ heißt *aktiv* im Laufe des Algorithmus, wenn $e(v) > 0$ und $dist(v) < \infty$.

Partielle Korrektheit des Algorithmus

Satz 4.23: Falls der Algorithmus von Goldberg und Tarjan terminiert und am Ende alle Markierungen endlich sind, dann ist der konstruierte Präfluss ein Maximalfluss im Netzwerk $(D; s, t; c)$.

Erinnerung:

Lemma 4.20: Sei f ein Präfluss auf D, die Funktion $dist$ eine bezüglich f zulässige Markierung auf V und $v \in V$ ein aktiver Knoten. Dann ist entweder eine PUSH-Operation von v oder eine RELABEL-Operation von v zulässig.

Lemma 4.22: Sei f ein Präfluss und $dist$ bezüglich f zulässig. Dann ist t im Residualgraph D_f von s aus nicht erreichbar (es gibt also keinen gerichteten s-t-Weg in D_f).

Erreichbarkeit der Quelle im Residualgraph

Lemma 4.24: Sei f ein Präfluss auf D. Wenn für v gilt, dass $e(v) > 0$, so ist s in D_f von v aus erreichbar.

Erinnerung:

Definition: Gegeben ein Netzwerk (D,s,t,c) mit angepasster Gewichtsfunktion, dann ist ein *Fluss* eine Abbildung $f\colon V \times V \to \mathbb{R}$ mit

1. *Kapazitätsbedingung:* für alle $(v, w) \in V \times V$ gilt $f(v, w) \leq c(v, w)$
2. *Antisymmetrie:* für alle $(v, w) \in V \times V$ gilt $f(v, w) = -f(w, v)$
3. *Flusserhaltung:* für alle $v \in V \setminus \{s, t\}$ gilt $\sum_{u \in V} f(u, v) = 0$

Wert eines Flusses: $w(f) = \sum_{v \in V} f(s, v) = \sum_{v \in V} f(v, t)$

Obere Schranke für Distanz

Lemma 4.25: Während des gesamten Algorithmus gilt

$$\forall v \in V \ \ dist(v) \leq 2|V| - 1 \ .$$

Erinnerung:

Definition: Ein Knoten $v \in V \setminus \{t\}$ heißt *aktiv* im Laufe des Algorithmus, wenn $e(v) > 0$ und $dist(v) < \infty$.

Operation $\textbf{R}\textsc{elabel}(D, f, v, dist)$
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty \ , & \text{falls } \{w \mid r_f(v, w) > 0\} = \emptyset \ , \\ \min\{dist(w) + 1 \mid r_f(v, w) > 0\} & \text{sonst.} \end{cases}$$

Obere Schranke für RELABEL-Operationen

Lemma 4.26: Im Laufe der Ausführung werden höchstens $2|V| - 1$ Operationen RELABEL pro Knoten ausgeführt. Die Gesamtzahl der RELABEL-Operationen ist also höchstens $2|V|^2$.

Erinnerung:

Lemma 4.25: Während des gesamten Algorithmus gilt

$$\forall v \in V \ \ dist(v) \leq 2|V| - 1 \, .$$

Operation RELABEL(D, f, v, $dist$)
Vorbedingung: v ist aktiv und für alle w mit $r_f(v, w) > 0$ gilt $dist(v) \leq dist(w)$
Effekt: $dist(v)$ wird erhöht.

$$dist(v) = \begin{cases} \infty \, , & \text{falls } \{w \mid r_f(v,w) > 0\} = \emptyset \, , \\ \min\{dist(w) + 1 \mid r_f(v,w) > 0\} & \text{sonst.} \end{cases}$$

Saturierende PUSH-Operationen

Definition 4.27: Eine Operation PUSH(v, w) heißt *saturierend*, wenn danach $r_f(v, w) = 0$ gilt. Ansonsten heißt PUSH(v, w) *nicht saturierend*.

Lemma 4.28: Während des Algorithmus werden höchstens $2|V||E|$ saturiernde PUSH ausgeführt.

Erinnerung:

Operation PUSH(D, f, v, w)
Vorbedingung: v ist aktiv, $r_f(v, w) > 0$ und $dist(v) = dist(w) + 1$.
Effekt: Flussüberschuss wird von v nach w über Kante (v, w) geschoben.

- $\Delta \leftarrow \min\{e(v), r_f(v, w)\}$
- $f(v, w) \leftarrow f(v, w) + \Delta$, $f(w, v) \leftarrow f(w, v) - \Delta$
- $r_f(v, w) \leftarrow r_f(v, w) - \Delta$, $r_f(w, v) \leftarrow r_f(w, v) + \Delta$
- $e(v) \leftarrow e(v) - \Delta$, $e(w) \leftarrow e(w) + \Delta$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Nicht saturierende PUSH-Operationen

Lemma 4.29: Während des Algorithmus werden höchstens $4|V|^2|E|$ nicht saturierende PUSH ausgeführt.

Erinnerung:

Definition 4.27: Eine Operation PUSH(v, w) heißt *saturierend*, wenn danach $r_f(v, w) = 0$ gilt. Ansonsten heißt PUSH(v, w) *nicht saturierend*.

Lemma 4.25: Während des gesamten Algorithmus gilt

$$\forall v \in V \;\; dist(v) \leq 2|V| - 1 \; .$$

Lemma 4.26: Während des Algorithmus werden höchstens $2|V| - 1$ Operationen RELABEL pro Knoten ausgeführt. Die Gesamtzahl der RELABEL-Operationen ist also höchstens $2|V|^2$.

Lemma 4.28: Während des Algorithmus werden höchstens $2|V||E|$ saturiernde PUSH ausgeführt.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Terminierung des Algorithmus

Satz 4.30: Der Algorithmus von Goldberg und Tarjan terminiert nach spätestens $\mathcal{O}(|V|^2|E|)$ Ausführungen zulässiger PUSH- oder RELABEL-Operationen.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Implementation

Laufzeit hängt stark von Wahl der aktiven Knoten ab:

- **FIFO-Implementierung:** Wähle aktive Knoten entsprechend *first-in-first-out*-Prinzip: $\mathcal{O}(|V|^3)$ Laufzeit.

  Mit dynamischen Bäumen: $\mathcal{O}(|V||E| \log \frac{|V|^2}{|E|})$
- **Highest-Label:** Für PUSH wähle aktiven Knoten mit höchstem $dist$-Wert: $\mathcal{O}(|V|^2|E|^{\frac{1}{2}})$
- **Excess-Scaling:** Für PUSH(v,w) wird die Kante (v, w) so gewählt, dass v aktiv, $e(v)$ *geeignet groß* und $e(w)$ *geeignet klein* ist: $\mathcal{O}(|E| + |V|^2 \log C)$, mit $C = \max\limits_{(u,v) \in E} c(u, v)$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner