Ganzzahlige Flüsse

Theorem

Wenn alle Kapazitäten ganzzahlig sind, dann findet die Ford–Fulkerson–Methode einen maximalen Fluß f, so daß alle $f(u, v)$ ganzzahlig sind.

Beweis

Induktion zeigt, daß die Kapazität eines augmentierenden Pfads ganzzahlig ist und $f(u, v)$ stets ganzzahlig bleiben.

Bipartites Matching

Gegeben:

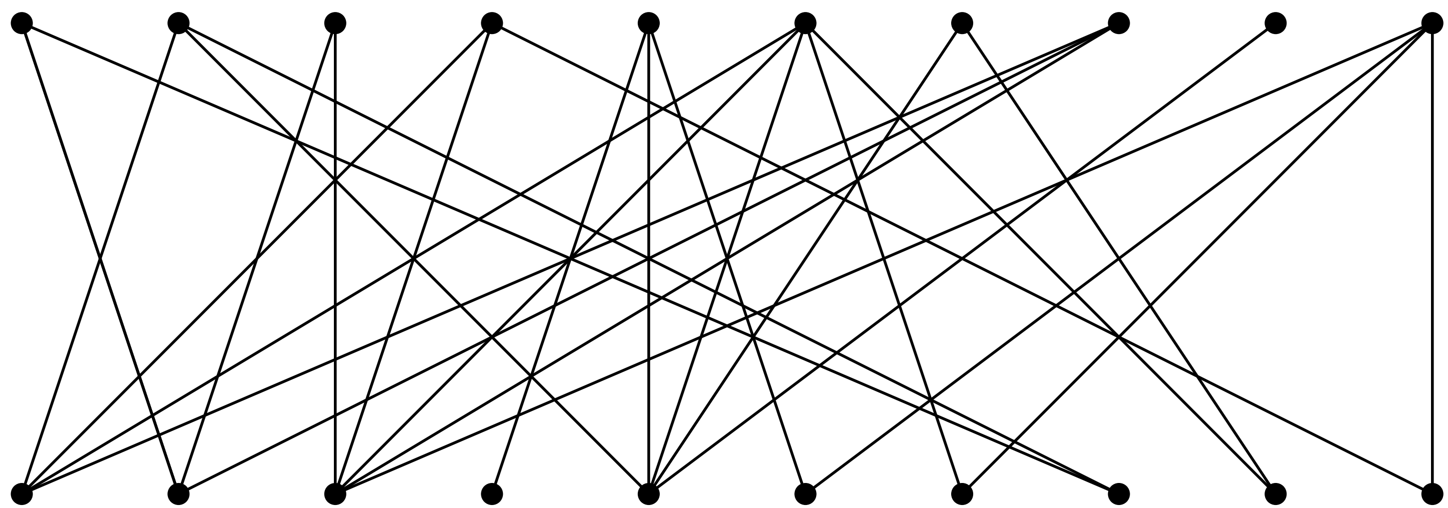
Ein bipartiter, ungerichteter Graph (V_1, V_2, E).

Gesucht:

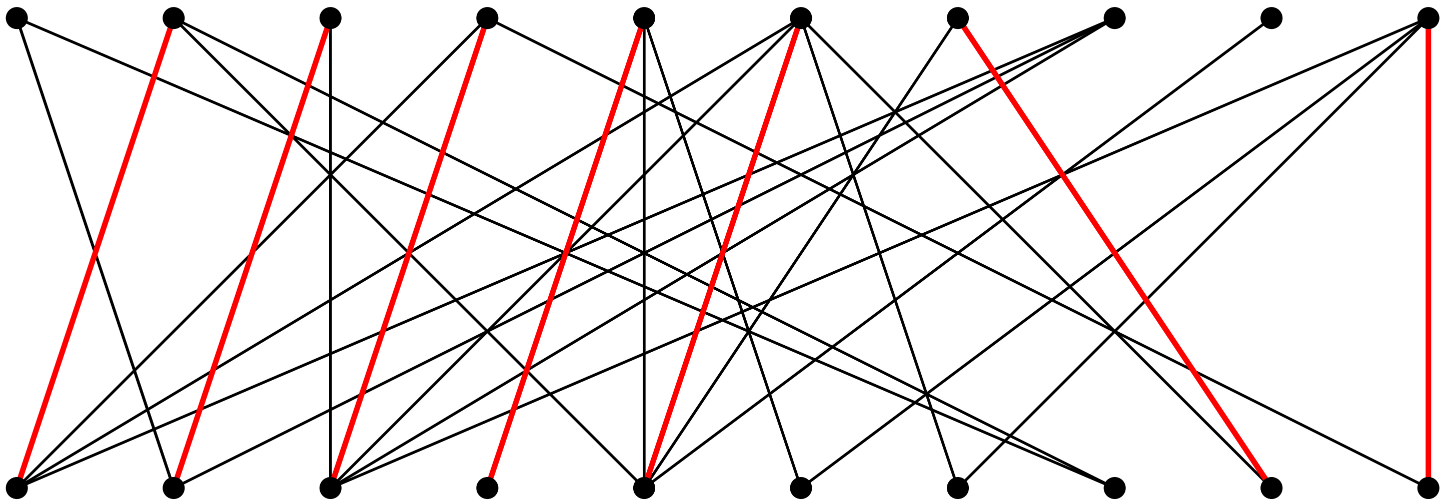
Ein Matching (Paarung) maximaler Kardinalität.

Ein *Matching* ist eine Menge paarweise nicht inzidenter Kanten, also $M \subseteq E$ mit $m_1, m_2 \in M,\ m_1 \neq m_2 \Rightarrow m_1 \cap m_2 = \emptyset$.

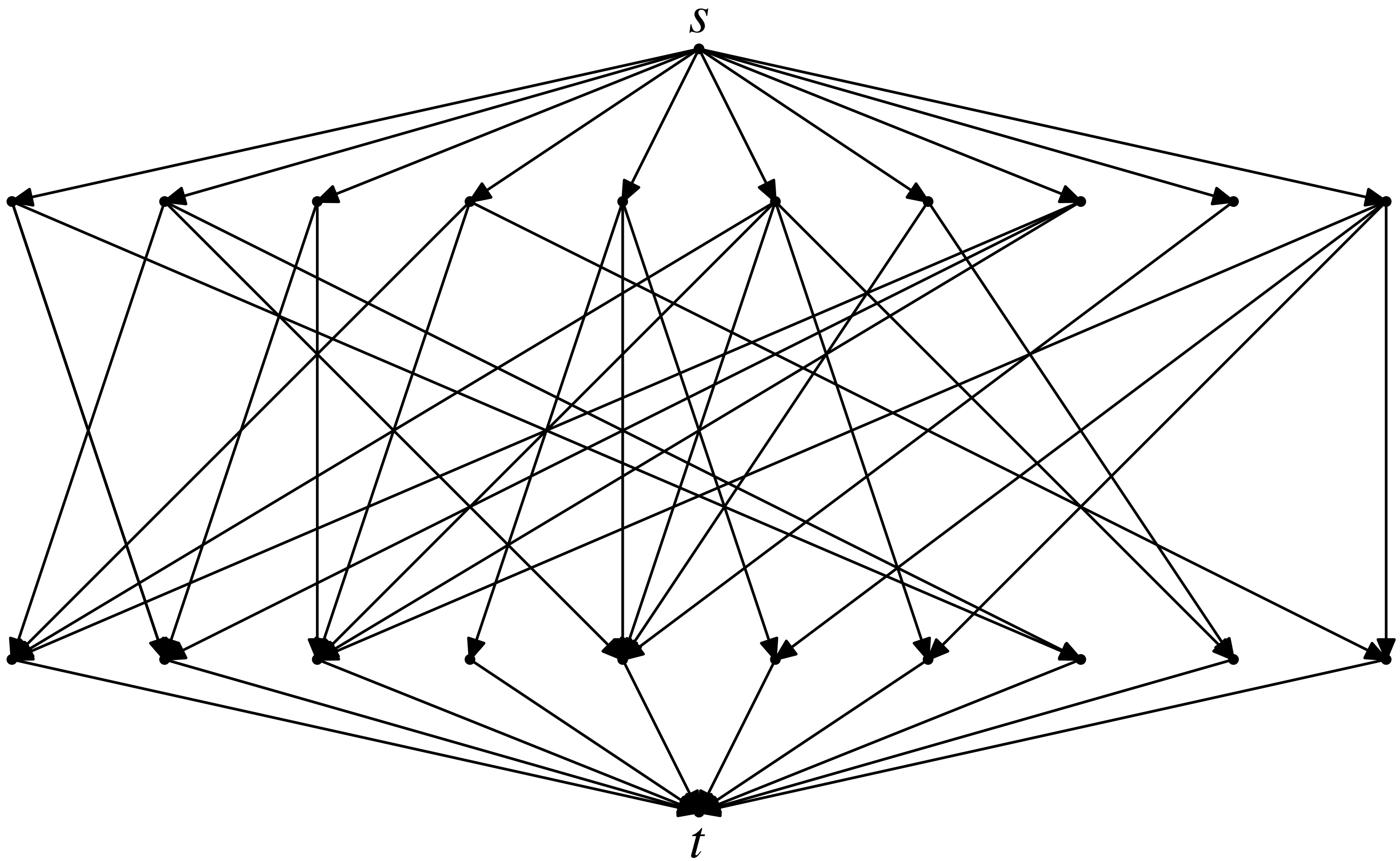
Beispiel



Beispiel

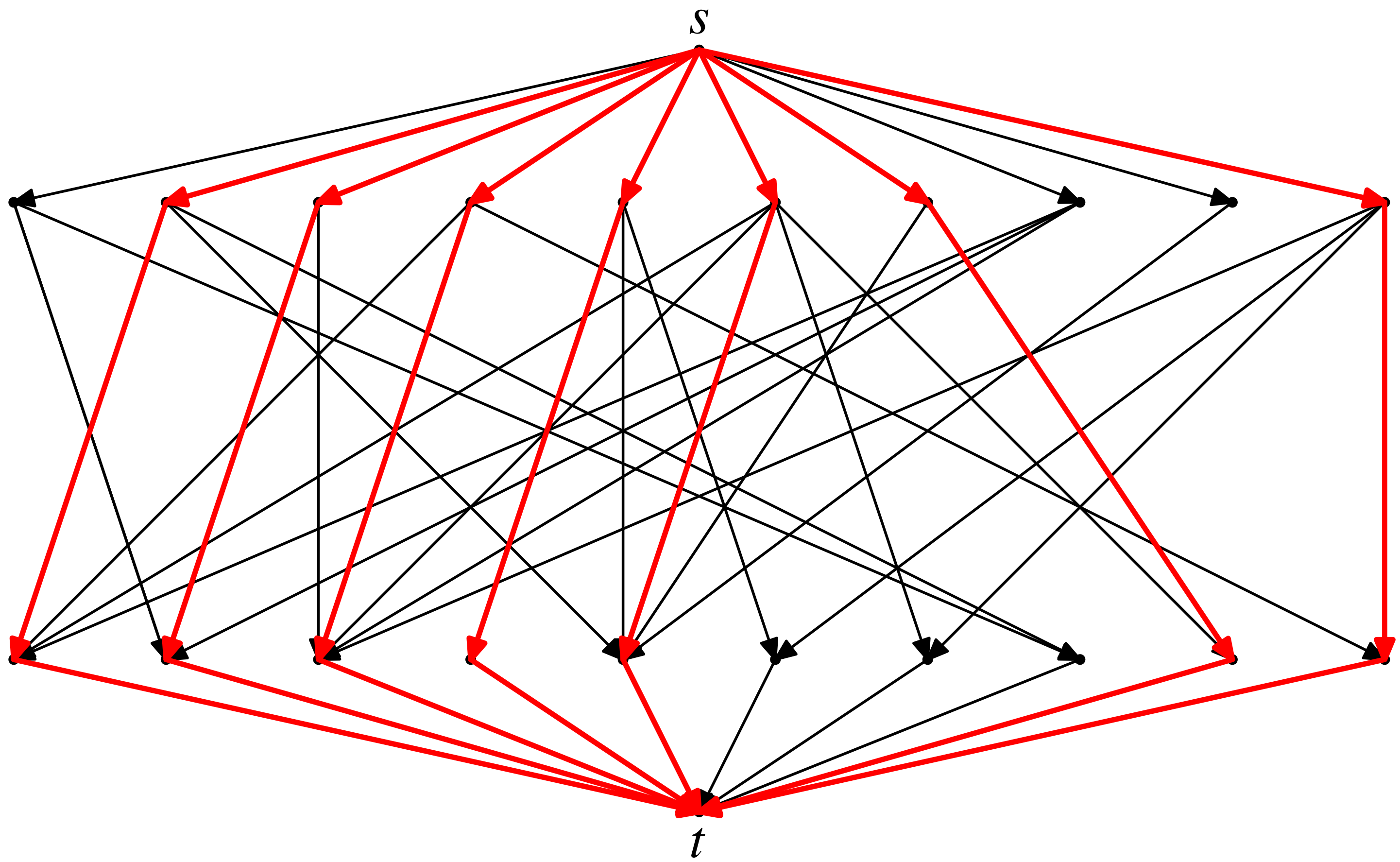


Lösung als Flußproblem

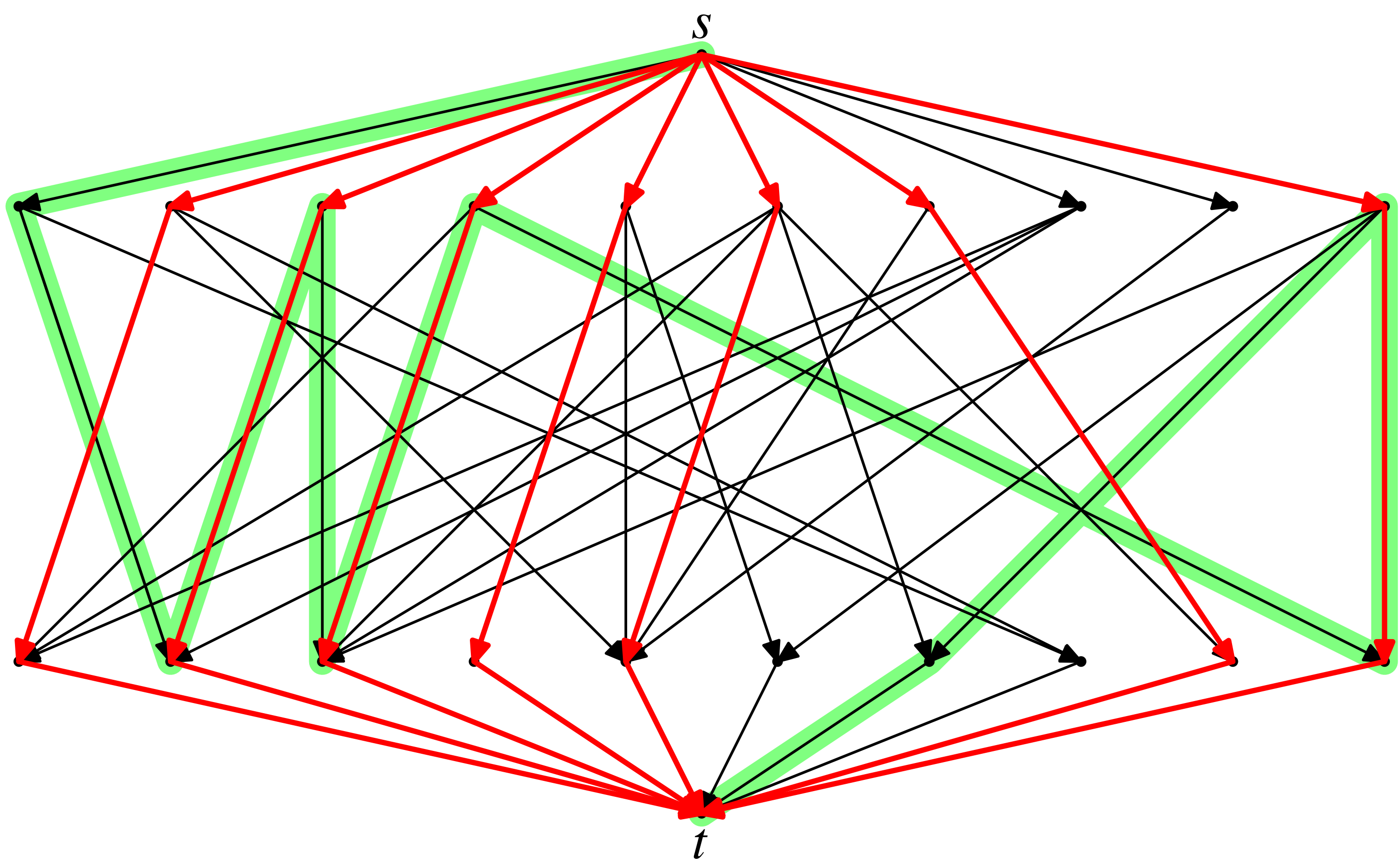


Alle Kapazitäten sind 1. Maximaler ganzzahliger Fluß entspricht einem Matching maximaler Kardinalität.

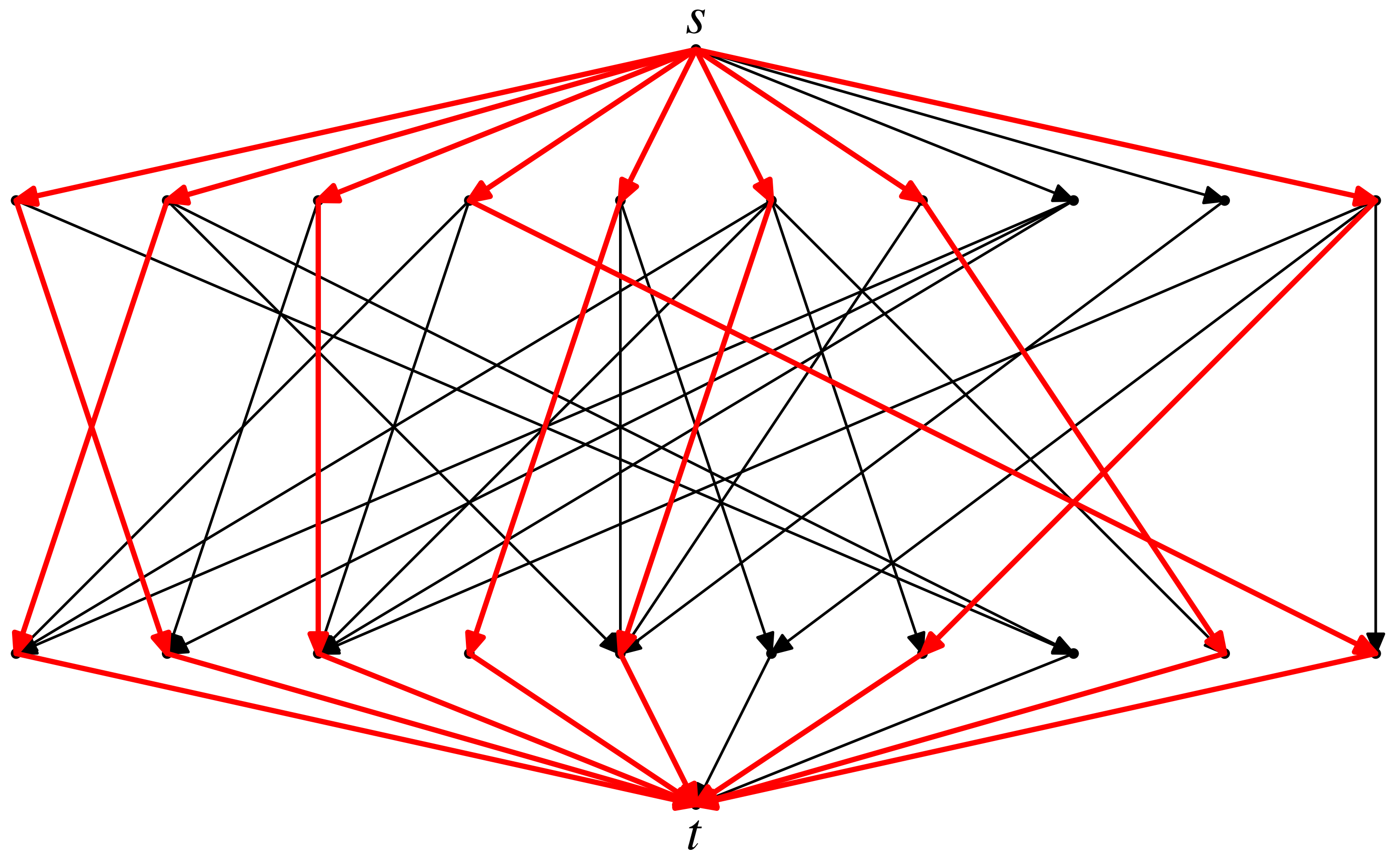
Gibt es einen augmentierenden Pfad?



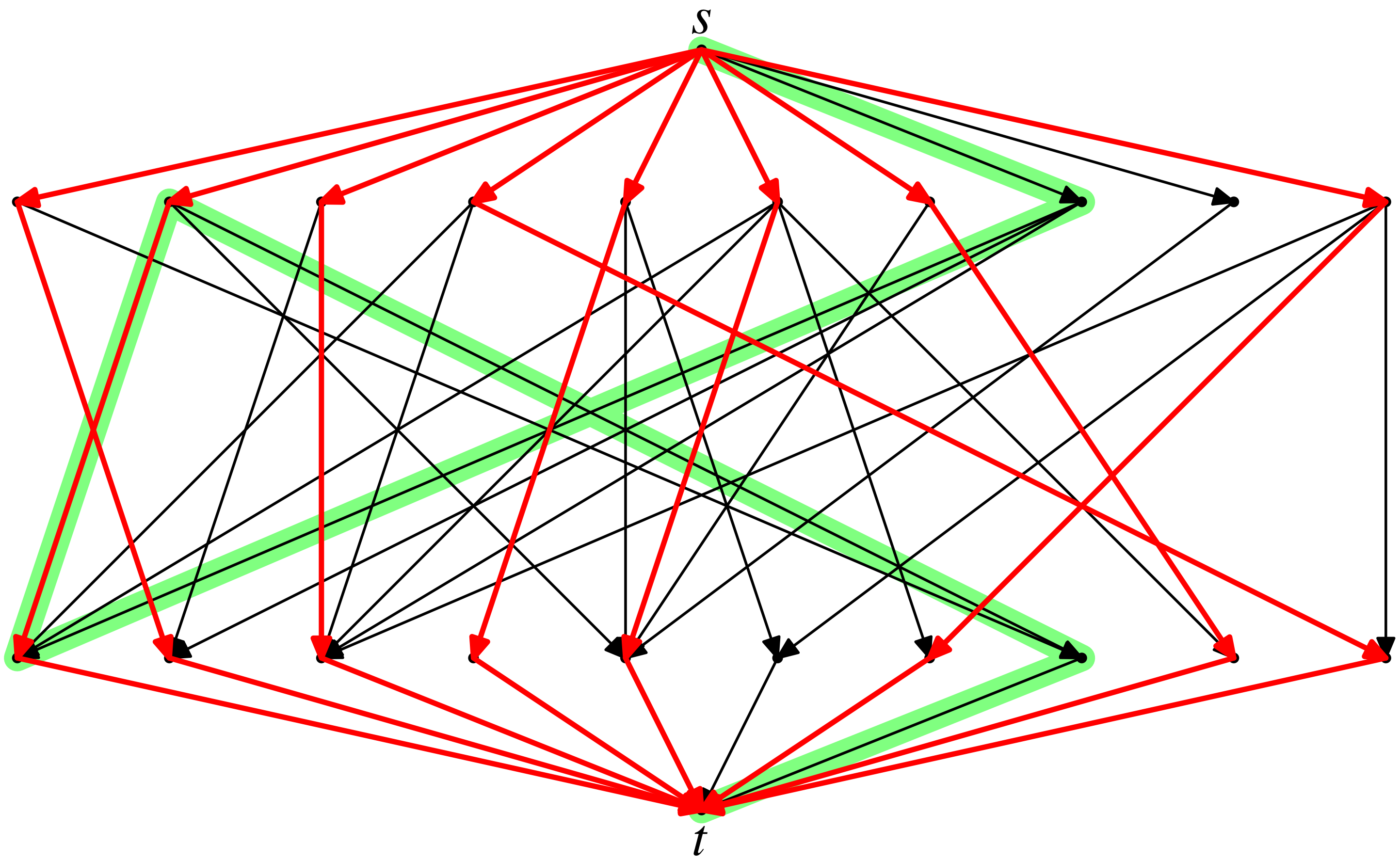
Gibt es einen augmentierenden Pfad? – Ja



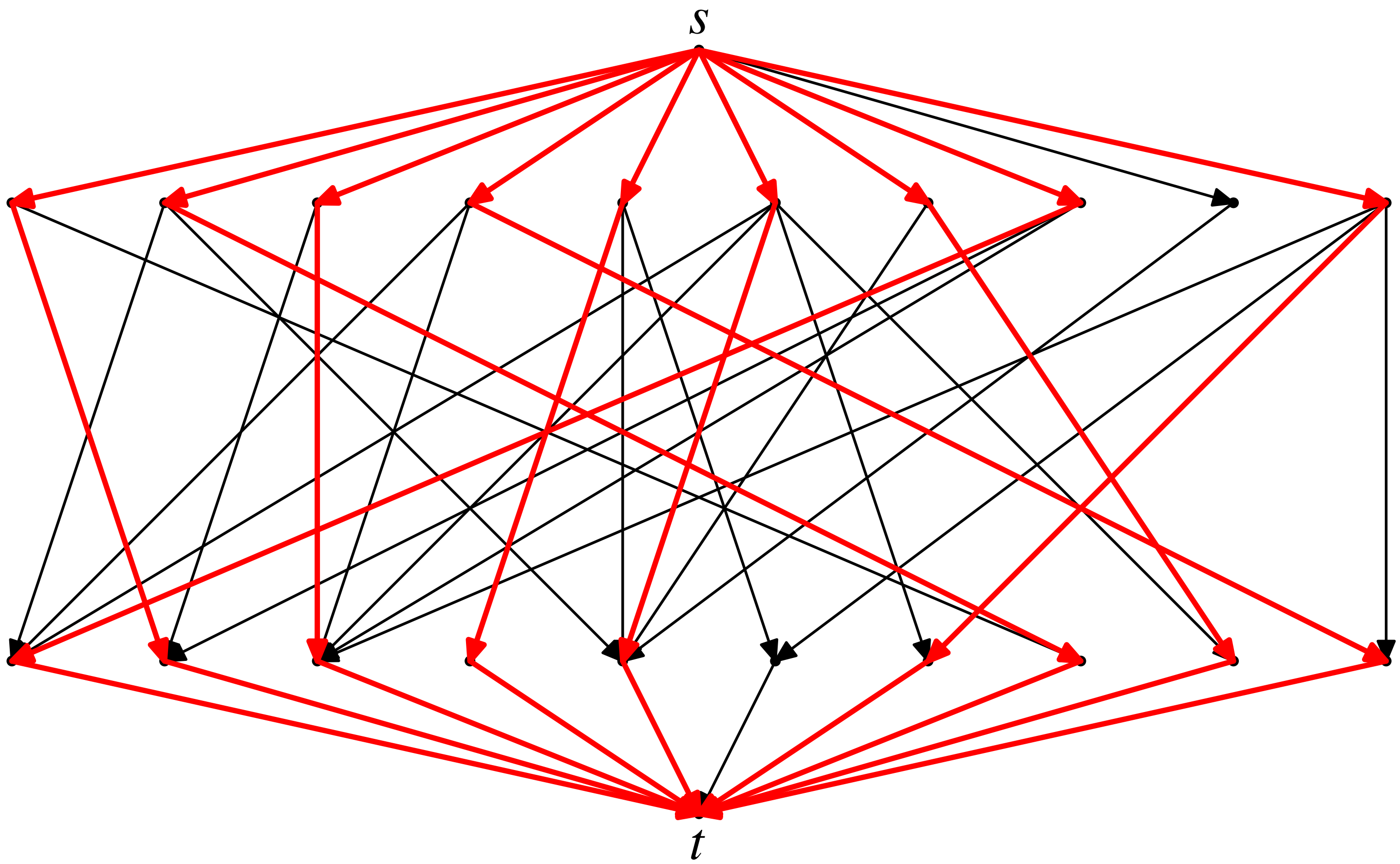
⇒ Neues Matching



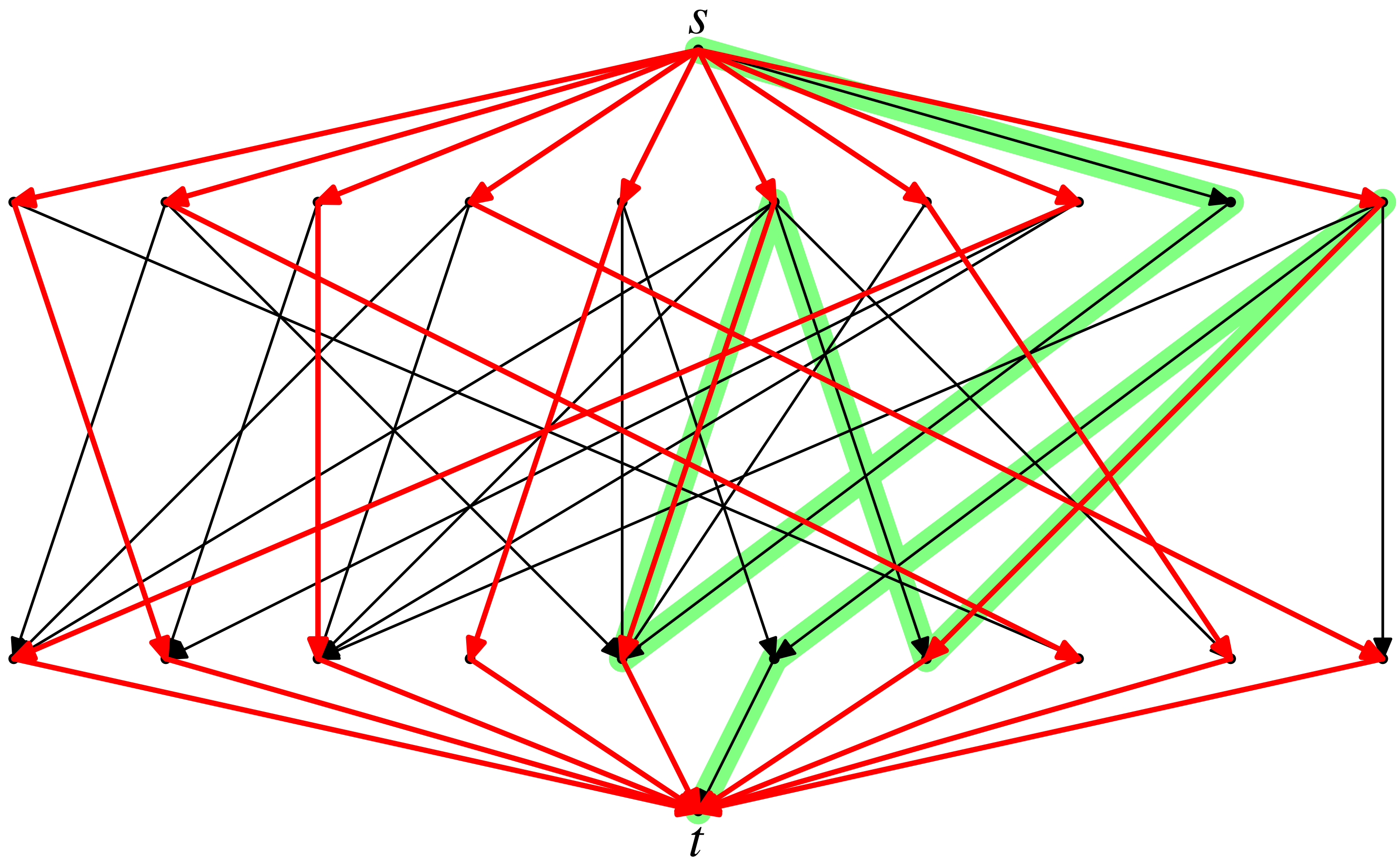
Es gibt wieder einen augmentierenden Pfad



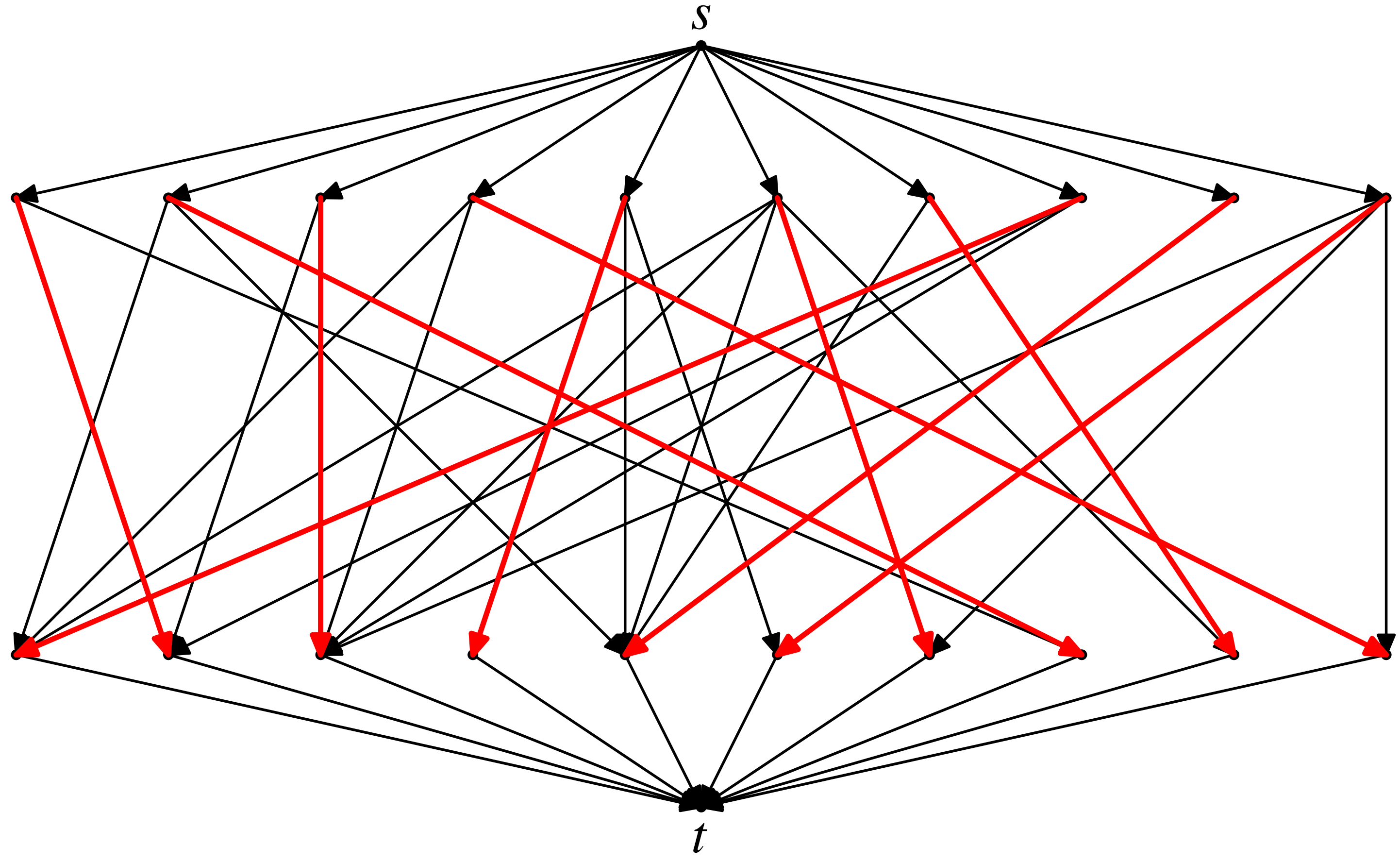
⇒ Neues Matching



Es gibt wieder einen augmentierenden Pfad



Ergebnis: Perfektes Matching



Laufzeit

Finden eines Matchings maximaler Kardinalität dauert nur $O(|E| \cdot \min\{|V_1|, |V_2|\})$ mit der Ford–Fulkerson–Methode.

- Der Fluß ist höchstens $f^* = \min\{|V_1|, |V_2|\}$.
- Finden eines Pfads dauert $O(|E|)$.

Variante der Ford–Fulkerson–Methode

Dieser Algorithmus funktioniert bei ganzzahligen Kapazitäten:

$K := 2^{\lfloor \log_2(\max\{\, c(u,v) \mid (u,v) \in E \,\}) \rfloor}$

$f := 0$

while $K \geq 1$ **do**

  while es gibt einen augmentierenden

    Pfad p mit $c_f(p) \geq K$ **do**

    augmentiere f entlang p

  $K := K/2$

return f

Die Laufzeit ist $O(|E|^2 \log K)$.

$\Rightarrow$ vergleiche mit $O(|E| f^*)$.

Variante der Ford–Fulkerson–Methode

Theorem

Die Laufzeit dieser Variante beträgt $O(|E|^2 \log C)$, wobei $C = \max\{\, c(u,v) \mid (u,v) \in E \,\}$.

Beweis

- Die Restkapazität eines minimalen Schnitts ist stets höchstens $2K|E|$.
- Für jedes K gibt es nur $|E|$ Augmentierungen
- Es gibt $O(\log C)$ verschiedene K

Der Edmonds–Karp–Algorithmus

Die Ford–Fulkerson–Methode kann sehr langsam sein, auch wenn das Netzwerk klein ist.

Der Edmonds–Karp–Algorithmus ist polynomiell in der Größe des Netzwerks.

```
Initialisiere Fluß f zu 0
while es gibt einen augmentierenden Pfad do
    finde einen kürzesten augmentierenden Pfad p
    augmentiere f entlang p
return f
```

Unterschied: Es wird ein kürzester Pfad gewählt

Der Edmonds–Karp–Algorithmus

for each edge $(u, v) \in E$ **do**
  $f(u, v) := 0$
  $f(v, u) := 0$
while there exists a path from s to t in G_f **do**
  $p :=$ a shortest path from s to t in G_f
  $c_f(p) := \min\{\, c_f(u, v) \mid (u, v) \text{ is in } p \,\}$
  for each edge (u, v) in p **do**
    $f(u, v) := f(u, v) + c_f(p)$
    $f(v, u) := -f(u, v)$
return f

Analyse des Edmonds–Karp–Algorithmus

Lemma D

Gegeben ist ein s-t-Netzwerk $G = (V, E)$. Sei f ein Fluß, der während der Ausführung des Edmonds–Karp–Algorithmus vorkommt und f' der Fluß nach der anschließenden Augmentierung.

Dann gilt: $\delta'(s, v) \geq \delta(s, v)$ für alle $v \in V$.

Hierbei ist $\delta(u, v)$ (bzw. $\delta'(u, v)$) die Länge des kürzesten Pfads von u nach v in G_f (bzw. in $G_{f'}$).

Analyse des Edmonds–Karp–Algorithmus

Beweis

Durch Widerspruch. Nehmen wir an, es gibt ein v mit

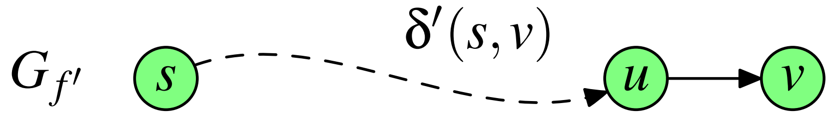
$$\delta(s,v) > \delta'(s,v). \tag{W}$$

Ohne Einschränkung können wir annehmen, daß

$$\delta(s,u) > \delta'(s,u) \Longrightarrow \delta'(s,u) \geq \delta'(s,v). \tag{M}$$

Wir müssen nur unter den möglichen v ein solches mit minimalem $\delta'(s,v)$ wählen.

Analyse des Edmonds–Karp–Algorithmus



u ist vorletzter Knoten auf kürzestem s-v-Pfad.

1. Möglichkeit: $(u, v) \in E[G_f]$

Dann gilt:

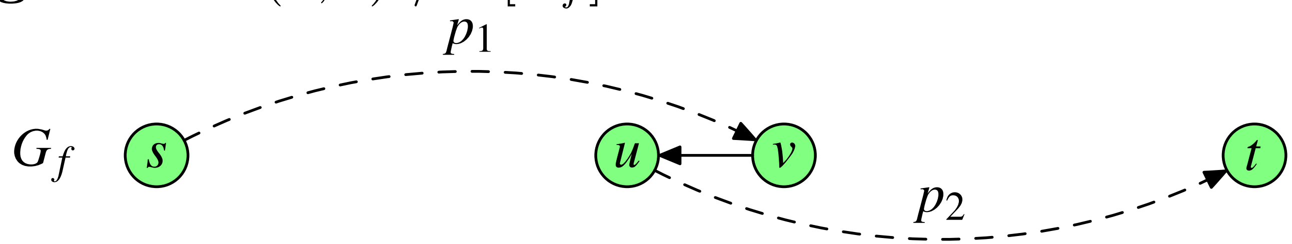
$$\delta'(s, v) = \delta'(s, u) + 1 \geq \delta(s, u) + 1 \text{ wegen (M)}$$

und

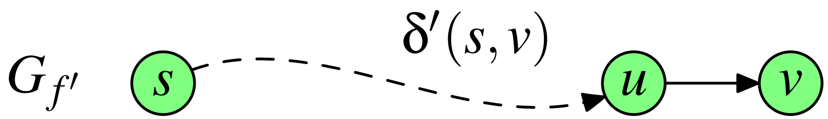
$$\delta(s, v) \leq \delta(s, u) + 1 \text{ wegen } (u, v) \in E[G_f]$$

Daraus folgt $\delta'(s, v) \geq \delta(s, v)$. Widerspruch zu (W).

2. Möglichkeit: $(u, v) \notin E[G_f]$



$p_1, (v, u), p_2$ wird als augmentierender Pfad benutzt.



Es gilt:

$$\delta(s, u) = \delta(s, v) + 1 \text{ und } \delta'(s, u) = \delta'(s, v) - 1$$

Wegen (M) gilt ausserdem:

$$\delta'(s, u) \geq \delta(s, u)$$

Zusammen folgt $\delta'(s, v) \geq \delta(s, v) + 2$, ein Widerspruch zu (W). $\square$

Analyse des Edmonds–Karp–Algorithmus

Theorem

Der Edmonds–Karp–Algorithmus findet einen maximalen Fluß in $O(|E|^2|V|)$ Schritten.

Beweis

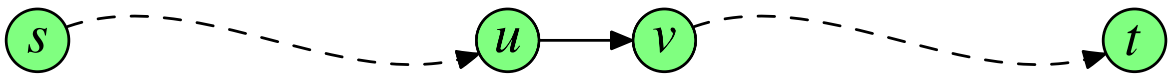
- Als Spezialfall der Ford–Fulkerson–Methode ist der Algorithmus korrekt.
- Jede Augmentierung benötigt $O(|E|)$ Schritte. Ein kürzester augmentierender Pfad kann mit BFS gefunden werden.
- Wir werden zeigen, daß nur $O(|E| \cdot |V|)$ Augmentierungen vorgenommen werden.

Sei f der Fluß vor und f' der Fluß nach einer Augmentierung.

Eine Kante (u, v) heißt *kritisch*, wenn $f(u, v) < c(u, v)$ und $f'(u, v) = c(u, v)$.

Jeder augmentierende Pfad enthält eine kritische Kante:

G_f

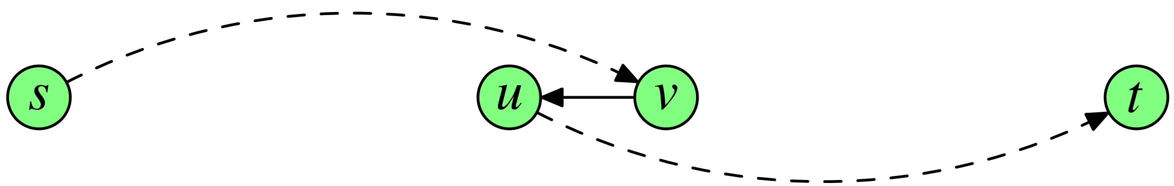


Sei (u, v) kritisch. Kann (u, v) später nocheinmal kritisch werden?

(u, v) verschwindet aus dem Residualnetzwerk.

Bevor (u, v) wieder kritisch wird, muß (u, v) wieder im Residualnetzwerk erscheinen:

$G_{f''}$



$\Rightarrow$ Der augmentierende Pfad geht durch (v, u).

- $\delta''(s, u) = \delta''(s, v) + 1$
- $\delta(s, v) = \delta(s, u) + 1$
- $\delta''(s, v) \geq \delta(s, v)$ wegen Lemma D
- $\Rightarrow \delta''(s, u) > \delta(s, u)$.

Bevor (u, v) wieder kritisch werden kann, hat sich der Abstand von s zu u erhöht.

Also kann eine Kante nur $|V|$ mal kritisch werden und es gibt höchstens $|E| \cdot |V|$ Augmentierungen. $\square$