Softwarepraktikum

4. Frontalveranstaltung
01.12.2017

ES Real-Time Systems Lab

Prof. Dr. rer. nat. Andy Schürr

Dept. of Electrical Engineering and Information Technology

Dept. of Computer Science (adjunct Professor)

www.es.tu-darmstadt.de

Dr. Malte Lochau

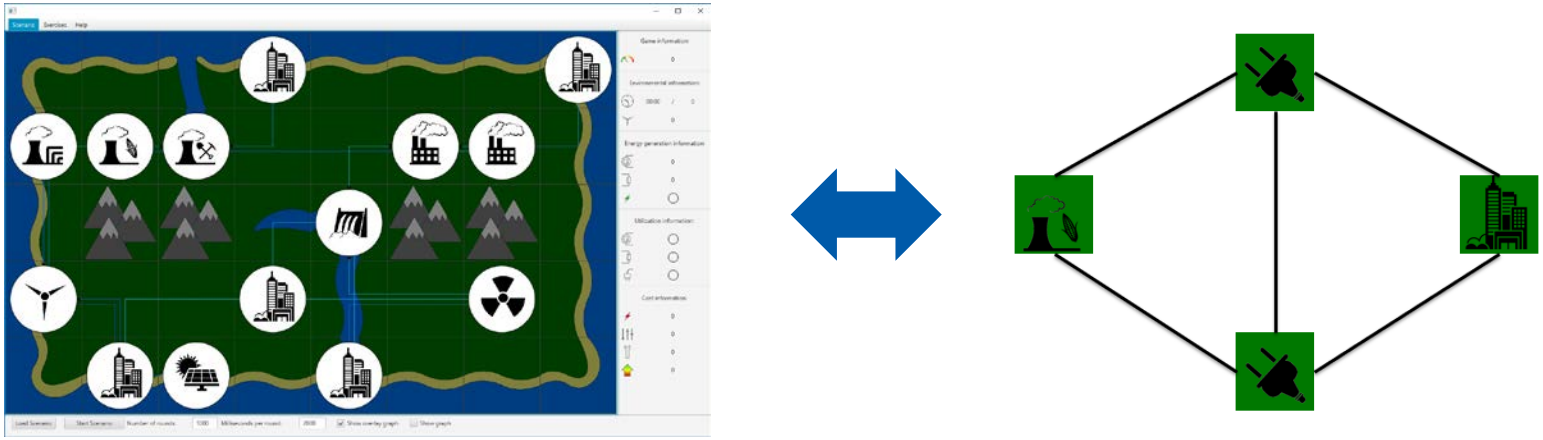
Malte.Lochau@es.tu-darmstadt.de

Gliederung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Pfadsuche in Graphen
- Algorithmus von Ford-Fulkerson
- Aufgabenblock 4

Planung und Steuerung im EVS



- Die Bestimmung des maximalen Flusses im Flussgraphen eines EVS-Szenarios bildet die Grundlage für die Planung und Regelung
- Planungsphase: Bau von Produzenten und Ausbau von Leitungen, sodass der **maximale Bedarf** aller Konsumenten gedeckt werden kann
- Ausführungsphase: Regelung regelbarer Konsumenten und Produzenten, sodass der **aktuelle Bedarf** aller Konsumenten in der jeweiligen Runde gedeckt ist

Rückblick: Umwandlung von gerichteten symmetrischen Flussgraphen und vereinfachten symmetrischen Residualgraphen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Wir bezeichnen mit e_{ij} die Kante im Flussgraphen von Knoten i nach j und e_{ij}^r die zugehörige Kante im Residualgraphen

- Berechnung der Kantenkapazitäten des Residualgraphen aus dem Flussgraphen:
  - $c_f(e_{ij}^r) = if\ f(e_{ij}) > 0\ then\ c(e_{ij}) - f(e_{ij})\ else\ c(e_{ij}) + f(e_{ji})$

- Berechnung der Flüsse und Kapazitäten des Flussgraphen aus dem Residualgraphen:
  - $f(e_{ij}) = if\ c_f(e_{ij}^r) < c_f(e_{ji}^r)\ then\ c_f(e_{ji}^r) - c(e_{ij})\ else\ 0$
  - $c(e_{ij}) = [c_f(e_{ij}^r) + c_f(e_{ji}^r)]/2$

Rückblick: Maximaler Fluss in gerichteten symmetrischen Flussgraphen und vereinfachten symmetrischen Residualgraphen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Ausgehend vom Startknoten s wird der Fluss entlang der Kanten mit verbliebener Restkapazität immer weiter erhöht, bis keine weitere Flusserhöhung möglich ist

- Der maximale Fluss ist erreicht, wenn kein weiterer Pfad vom Startknoten s zum Zielknoten t mit Restkapazitäten größer Null vorhanden ist

- Die Berechnung der Kapazitäten und Flüsse der Kanten erfolgt nach den zuvor genannten Formeln

- Legende für die Beschriftung der Graphen:

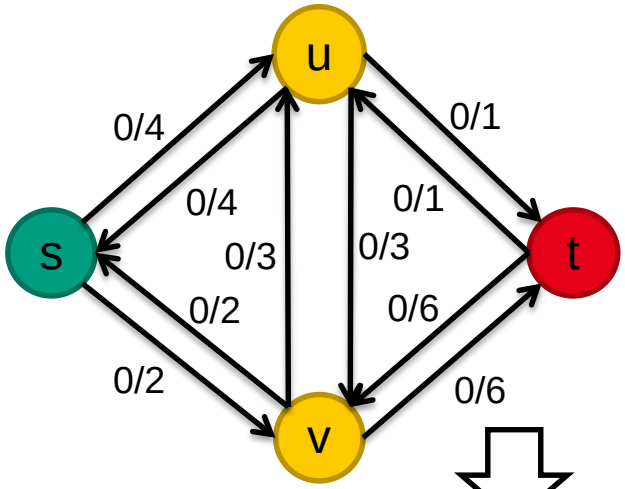
$f(e_{ij})$ / $c(e_{ij})$ ⟶ Kante e_{ij}

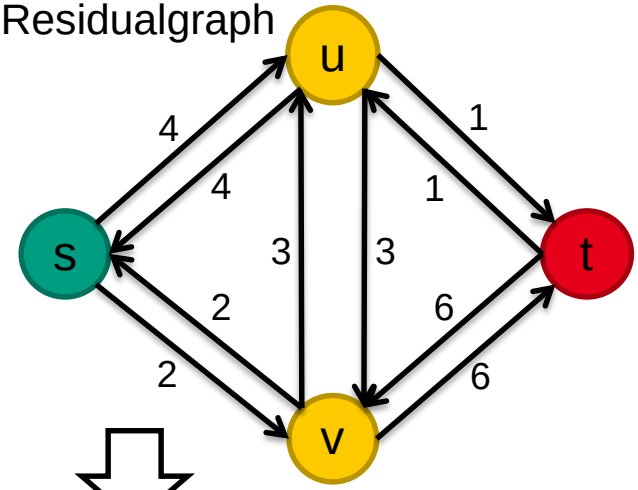
$c_f(e_{ij}^r)$ ⟶

Rückblick: Maximaler Fluss in gerichteten symmetrischen Flussgraphen und vereinfachten symmetrischen Residualgraphen

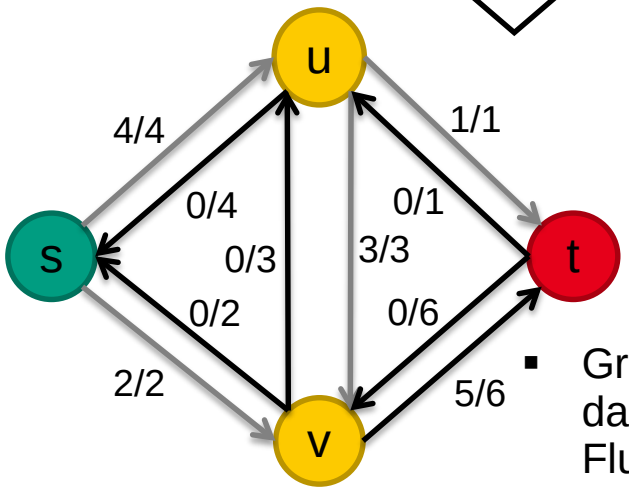
TECHNISCHE UNIVERSITÄT DARMSTADT

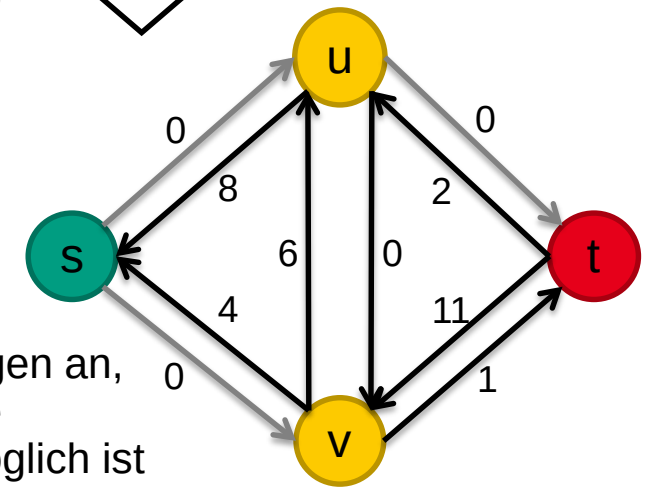
- gerichteter symmetrischer Flussgraph
- vereinfachter symmetrischer Residualgraph





Flusserhöhung





- Graue Kanten zeigen an, dass keine weitere Flusserhöhung möglich ist

Algorithmus von Ford-Fulkerson - Grundidee

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Eingabe:** Flussgraph G
- **Ausgabe:** Maximaler Fluss f von s nach t in G
- **Initialisierung:**
  Erstelle Residualgraph G_f für den initialen Fluss $f(e_{ij}) = 0$ für alle $e_{ij} \in E$ mit $e_{ij} :=$ Kante von Knoten i nach Knoten j
- **Iteration:**
  Solange es im Residualgraph G_f einen kürzesten Pfad mit Kapazitäten größer Null von s nach t gibt, bestimme einen solchen Pfad $W = (s^r, j_1^r), (i_1^r, j_2^r), \ldots, (i_{k-1}^r, t^r)$ und
  - Bestimme Flusswert $\gamma = \min\{ c_f(e_{ij}^r) \mid e_{ij}^r = (i^r, j^r)$ liegt auf dem Pfad $W \}$
  - Setze $f(e_{ij}) := f(e_{ij}) + \gamma$ und $c_f(e_{ij}^r) := c_f(e_{ij}^r) - \gamma$ für alle Kanten $e_{ij}^r = (i^r, j^r)$, die auf dem Pfad W liegen
  - Setze $f(e_{ji}) := f(e_{ji}) - \gamma$ und $c_f(e_{ji}^r) := c_f(e_{ji}^r) + \gamma$ für alle Rückkanten $e_{ji}^r = (j^r, i^r)$ der Kanten $e_{ji}^r = (j^r, i^r)$, die auf dem Pfad W liegen
  - Anpassen der Kantengewichte im Residualgraphen gemäß der Änderungen des Flusses f über Pfad W im Flussgraphen

Gliederung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Pfadsuche in Graphen
- Algorithmus von Ford-Fulkerson
- Aufgabenblock 4

Pfadsuche in Graphen durch Breitensuche

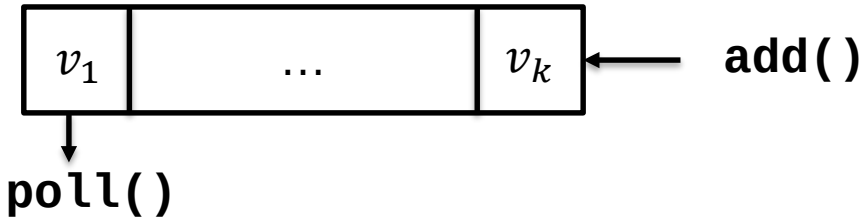
TECHNISCHE UNIVERSITÄT DARMSTADT

- Bestimmung derjenigen Pfade $(s^r, j_1^r), (i_1^r, j_2^r), \ldots, (i_{k-1}^r, t^r)$ mit minimalen Abständen von einem Startknoten s zu allen von s aus erreichbaren Knoten t im Graphen
- Der minimale Abstand k von s nach t ergibt sich aus der **Kantenanzahl** auf einem ermittelten **kürzesten Pfad**
- Die Breitensuche durchläuft ausgehend vom Startknoten s alle von s aus erreichbaren Knoten eines Graphen G
- Jeder neu entdeckte Knoten v wird erst komplett abgearbeitet (d.h. Entdeckung aller seiner Nachbarknoten v', die bisher noch nicht über einen anderen Knoten entdeckt worden sind), bevor mit einem der neu entdeckten Knoten fortgesetzt wird

Breitensuche

- Während jedes Schrittes der Breitensuche ist jeder Knoten v des Graphen mit genau einer „Farbe“ markiert
  - unbearbeitet (weiß)
  - entdeckt (grau)
  - fertig bearbeitet (schwarz)

- Die Breitensuche verwendet eine **Warteschlange** (**Deque**)

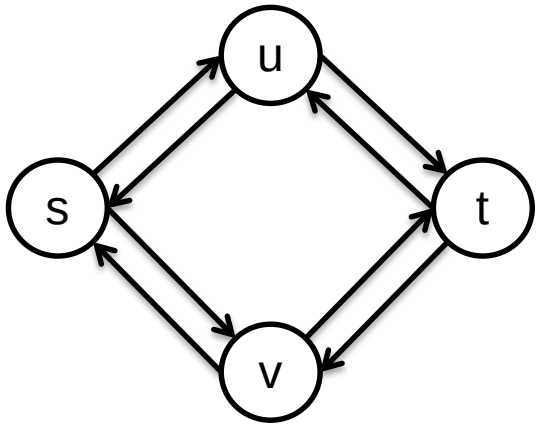


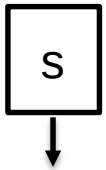
- Die Breitensuche konstruiert einen **Erreichbarkeitsbaum**

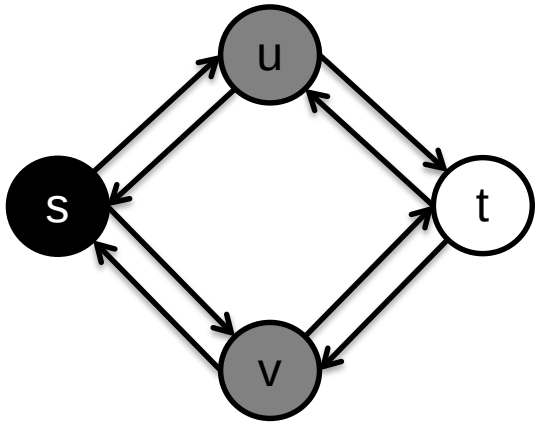
Beispiel: Breitensuche

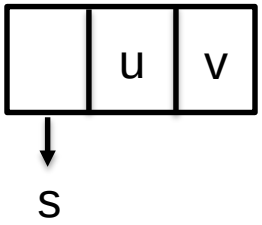
TECHNISCHE UNIVERSITÄT DARMSTADT

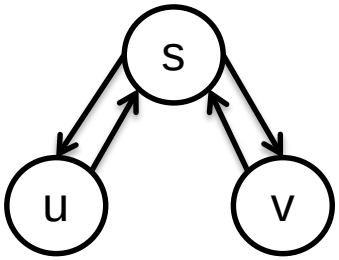
Graph G | Warteschlange | Erreichbarkeitsbaum







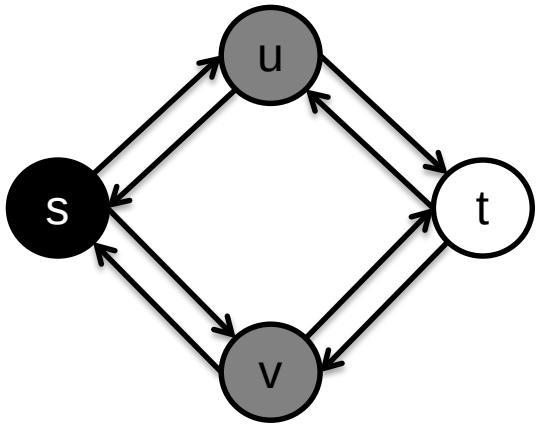


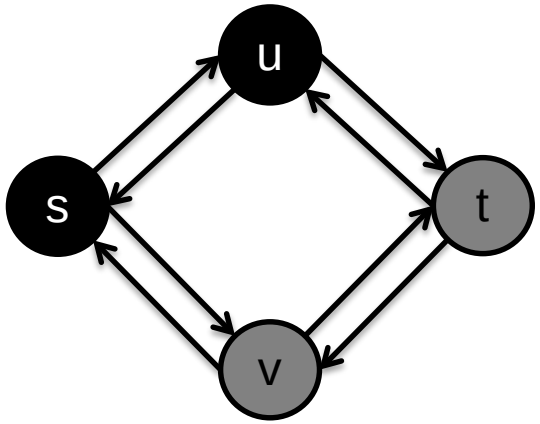


Beispiel: Breitensuche

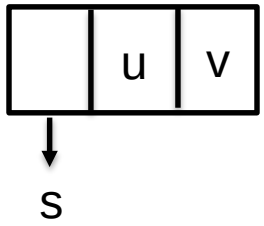
TECHNISCHE UNIVERSITÄT DARMSTADT

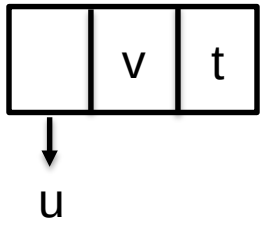
Graph G



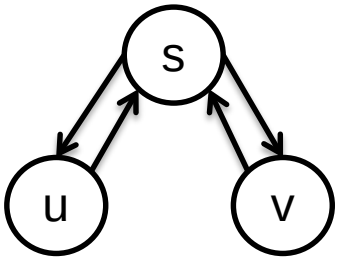


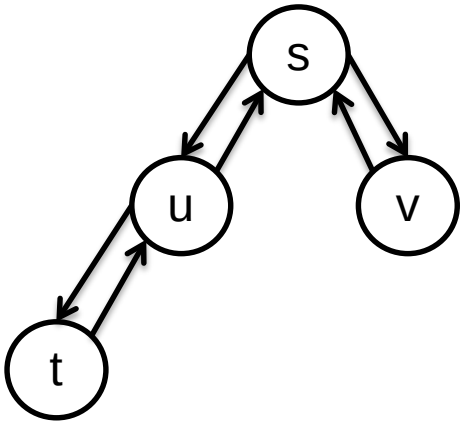
Warteschlange





Erreichbarkeitsbaum

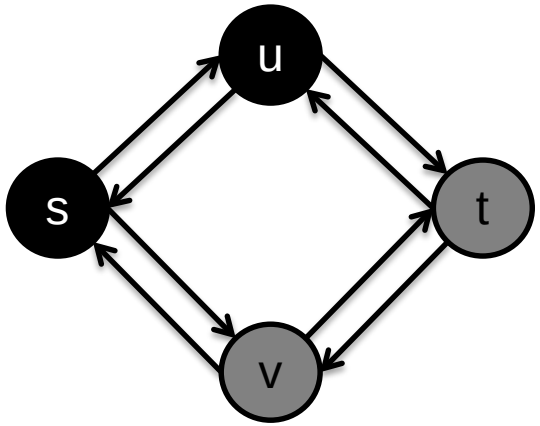




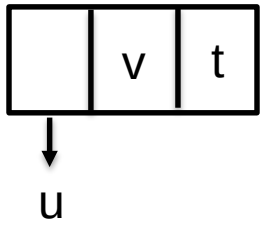
Beispiel: Breitensuche

TECHNISCHE UNIVERSITÄT DARMSTADT

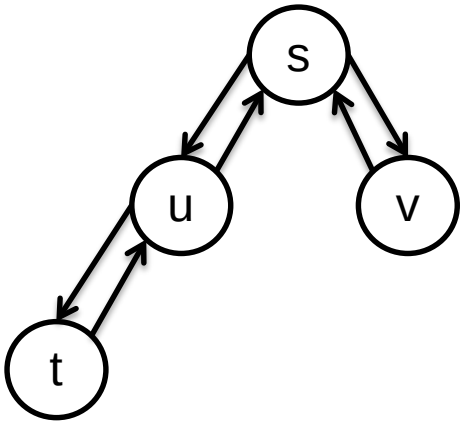
Graph G

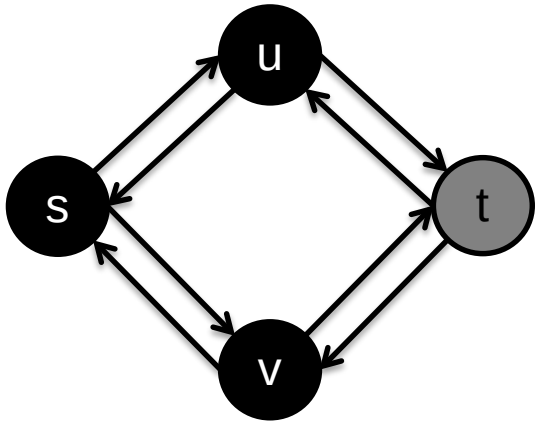


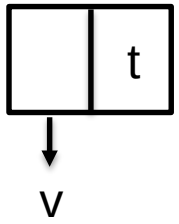
Warteschlange

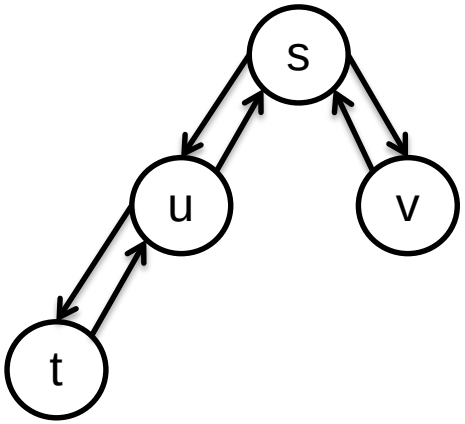


Erreichbarkeitsbaum



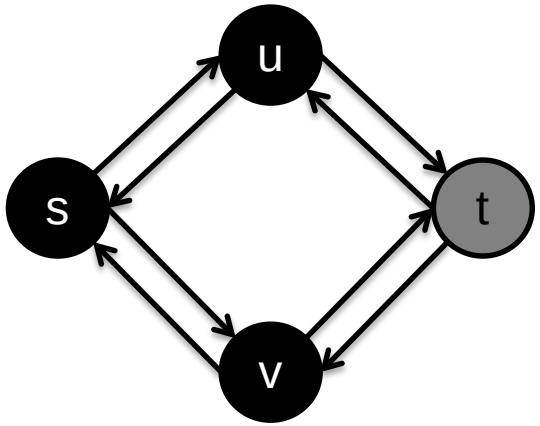


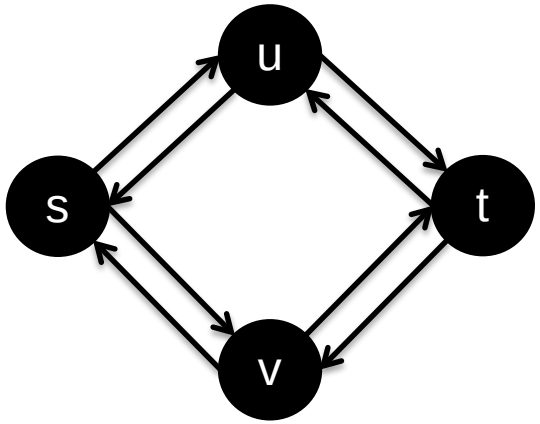




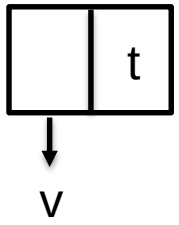
Beispiel: Breitensuche

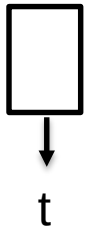
Graph G



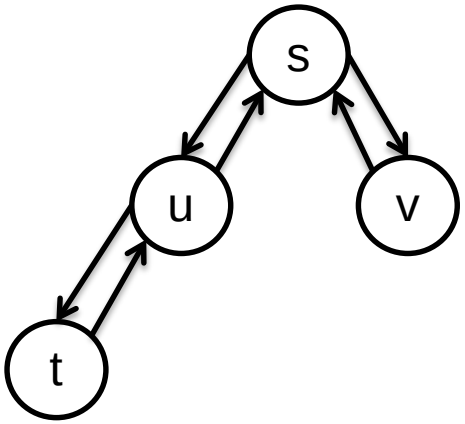


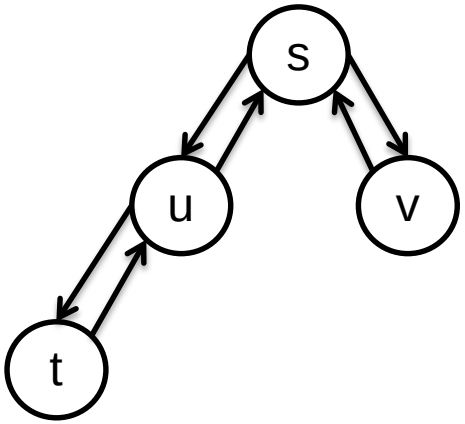
Warteschlange



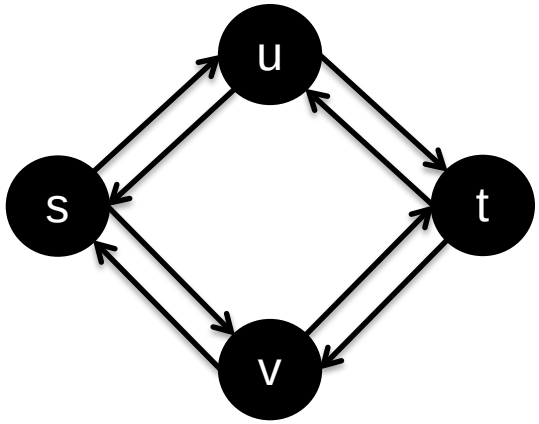


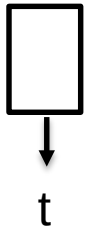
Erreichbarkeitsbaum

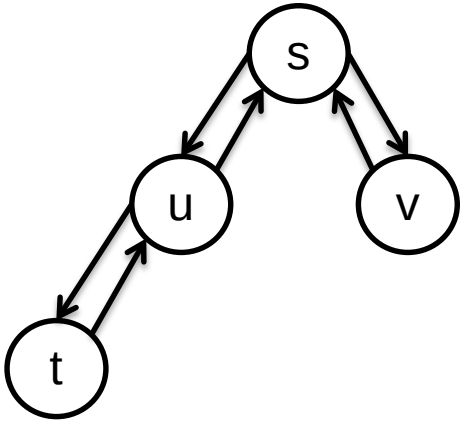




Beispiel: Breitensuche







- Die Pfadlänge von s nach t beträgt in diesem Fall 2

Modifikation der Breitensuche

TECHNISCHE UNIVERSITÄT DARMSTADT

- Im Ford-Fulkerson Algorithmus für das EVS soll die Breitensuche genutzt werden, um im aktuellen Residualgraphen einen kürzesten Weg mit verbliebener Restkapazität zwischen einem Startknoten s und einem Zielknoten t zu finden

- Soll mithilfe der Breitensuche lediglich ein kürzester Pfad von einem Startknoten s zu einem bestimmten Zielknoten t ermittelt werden, kann der Breitensuche Algorithmus abgebrochen werden, sobald der Knoten t entdeckt (d.h. grau eingefärbt) wurde

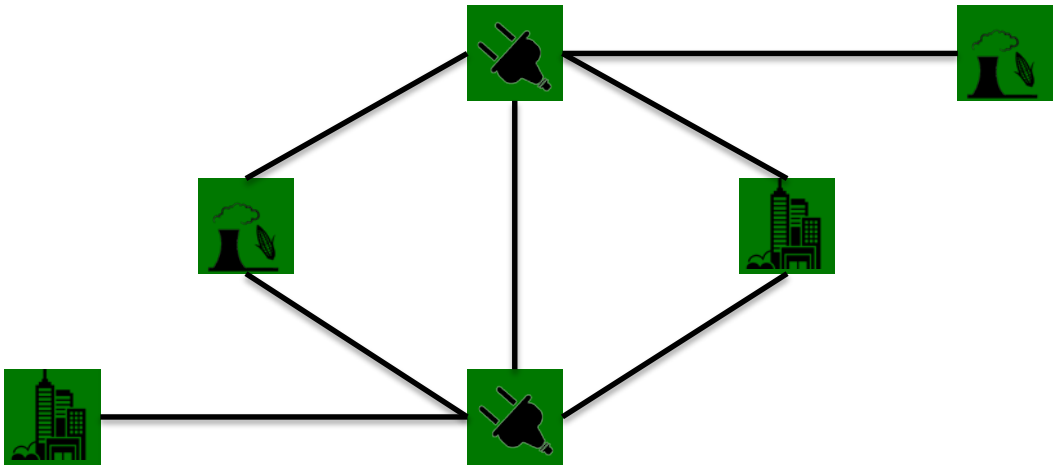
Gliederung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Pfadsuche in Graphen
- Algorithmus von Ford-Fulkerson
- Aufgabenblock 4

Behandlung mehrerer Quellen und Senken

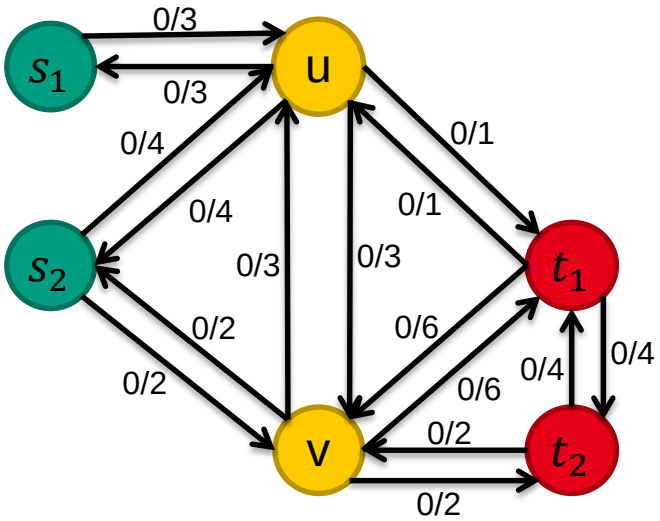
- Das Fluss-Netzwerk des EVS enthält **beliebig viele Konsumenten und Produzenten**, die beliebig miteinander verbunden sein können
- Wir führen deshalb eine (virtuelle) Super-Quelle und Super-Senke ein, um eine maximale Flussberechnung über das gesamte Netzwerk zu ermöglichen
  - Die Super-Quelle verbindet alle Produzenten mit einem Knoten, wobei die maximale Leistung der Produzenten als Kantenkapazität dient
  - Analog verbindet die Super-Senke alle Konsumenten mit einem Knoten



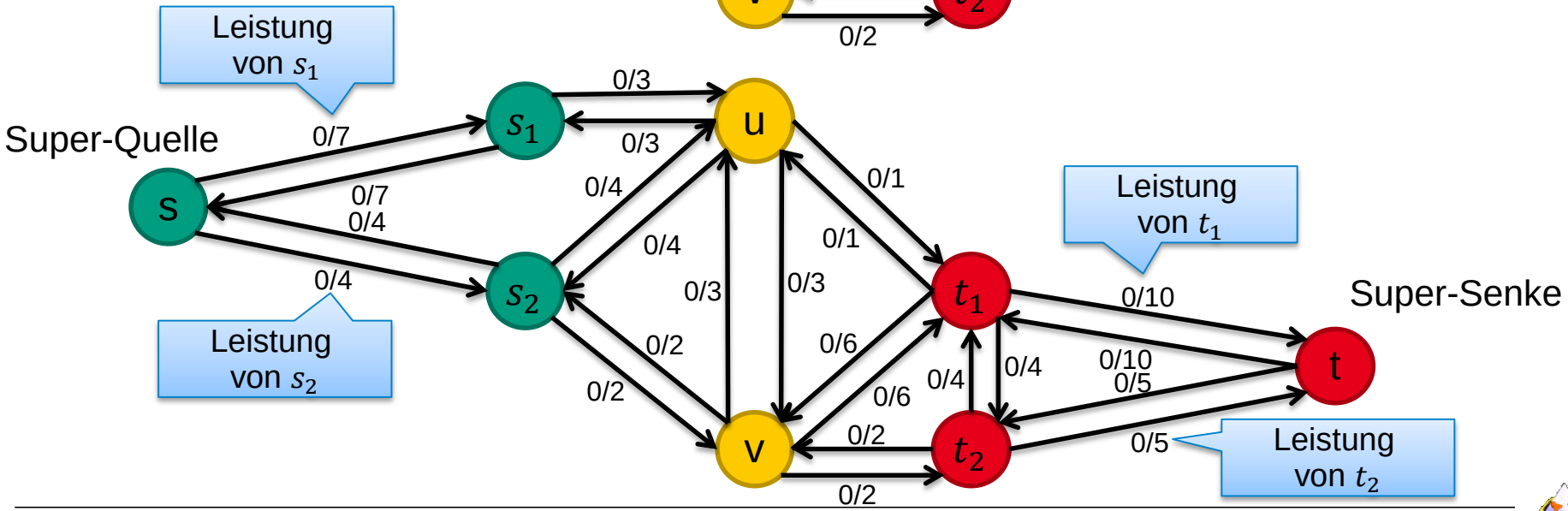
Behandlung mehrerer Quellen und Senken

- Grüne Knoten kennzeichnen Produzenten

- Rote Knoten kennzeichnen Konsumenten



$f(e_{ij})$ / $c(e_{ij})$ → Kante e_{ij}

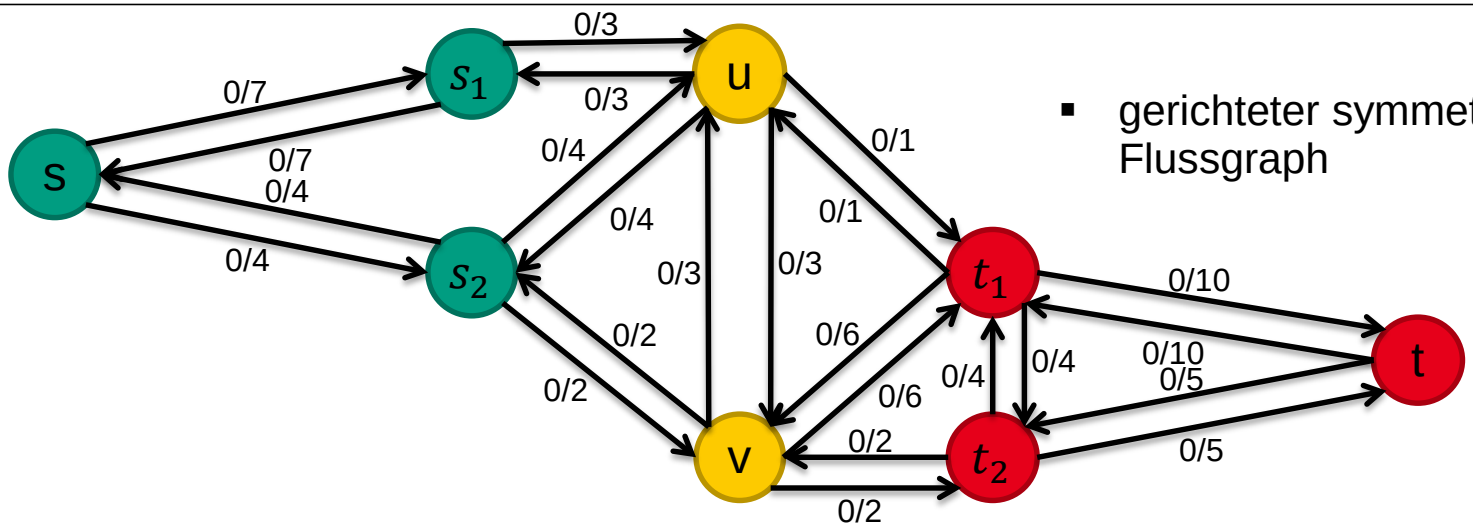


Maximaler Fluss mehrerer Quellen und Senken

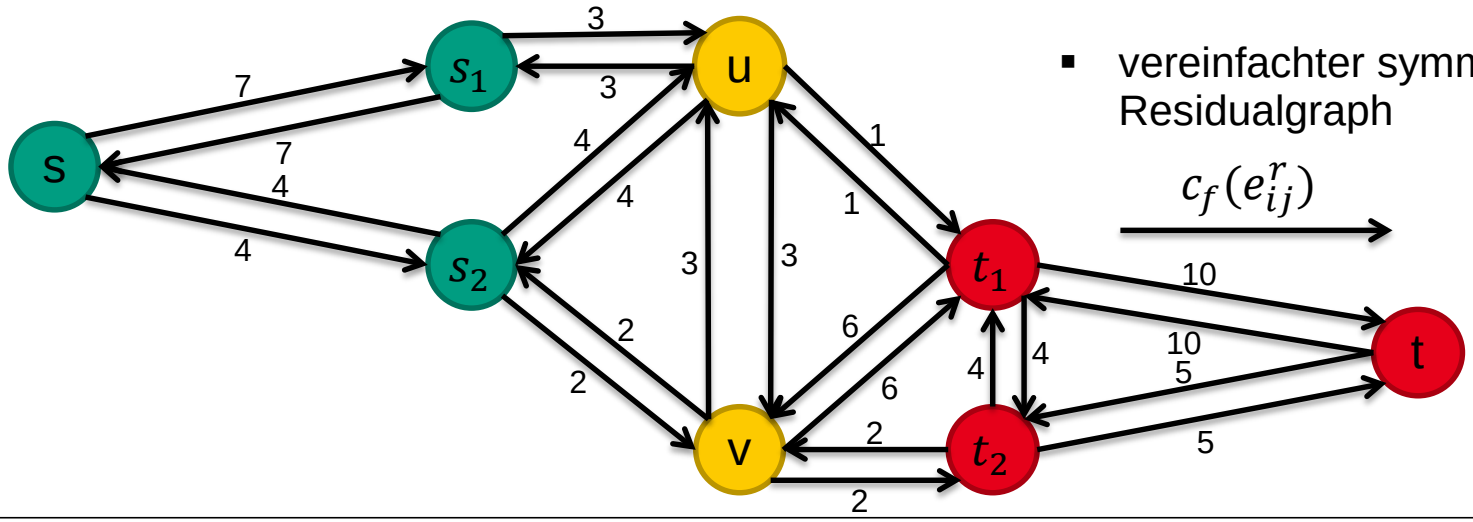
TECHNISCHE UNIVERSITÄT DARMSTADT

- Die maximale Flussberechnung erfolgt analog zu den vorherigen Beispielen
- Der maximale Fluss ist erreicht, wenn kein Pfad vom Startknoten s zum Zielknoten t mit Restkapazitäten größer Null vorhanden ist
- Ausgehend vom Startknoten s wird dabei der Fluss entlang der Kanten mit verbliebener Restkapazität immer weiter erhöht, bis keine weitere Flusserhöhung mehr möglich ist
- Die Berechnung der Kapazitäten und Flüsse der Kanten erfolgt nach den zuvor genannten Formeln

Maximaler Fluss mehrerer Quellen und Senken



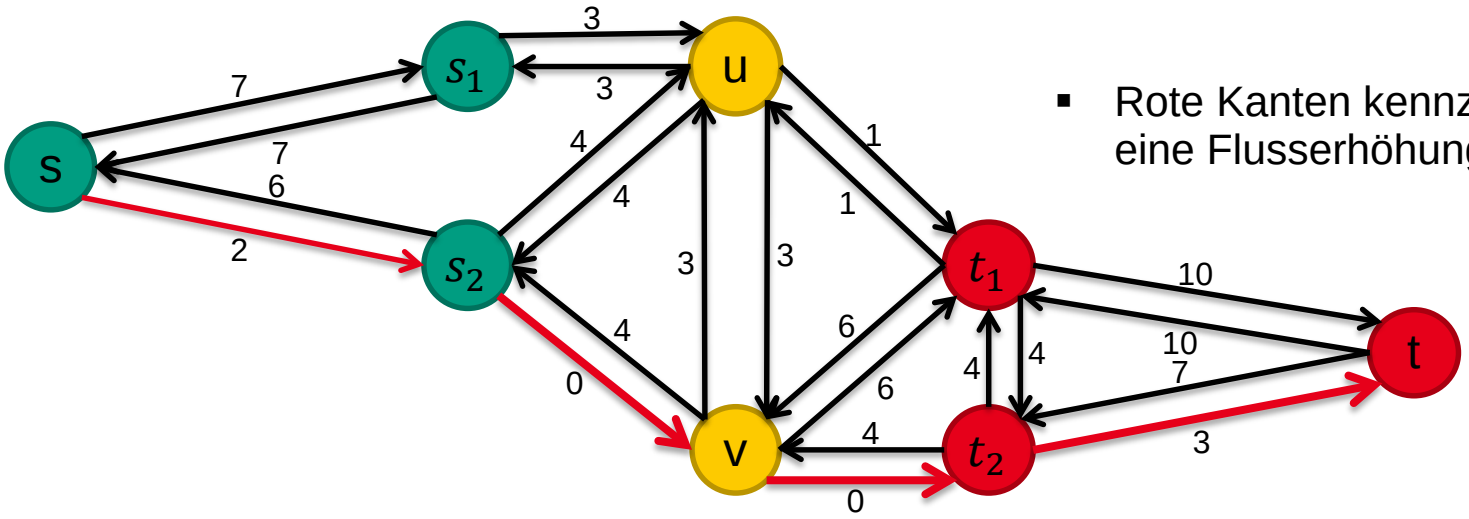
- gerichteter symmetrischer Flussgraph



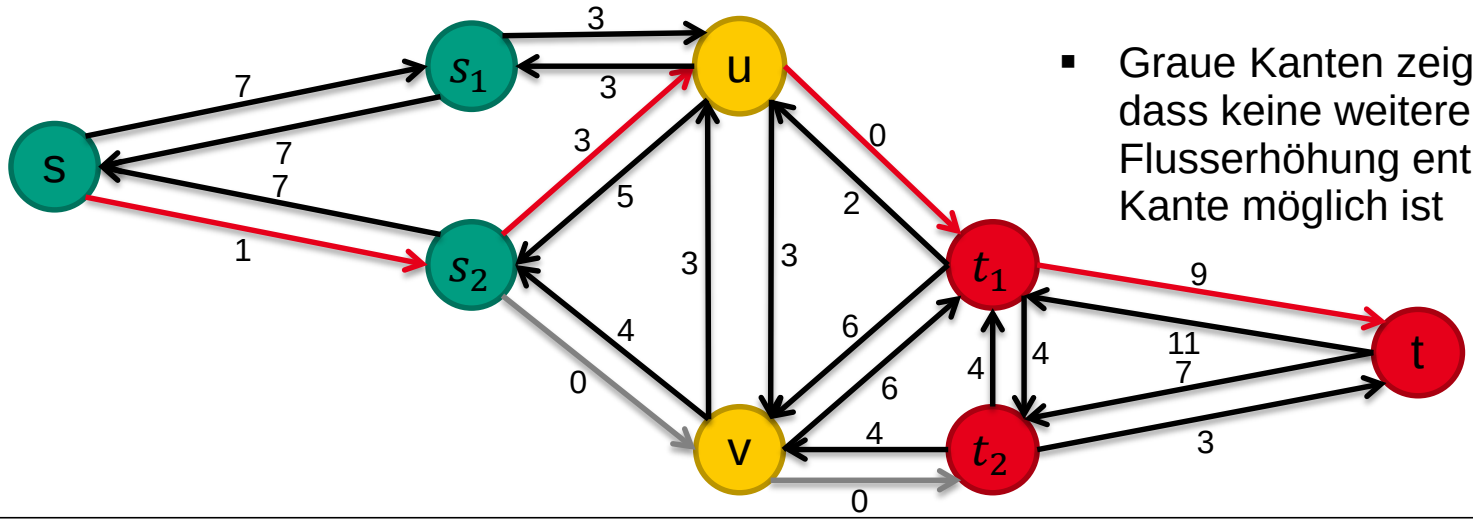
- vereinfachter symmetrischer Residualgraph

$c_f(e_{ij}^r)$

Maximaler Fluss mehrerer Quellen und Senken

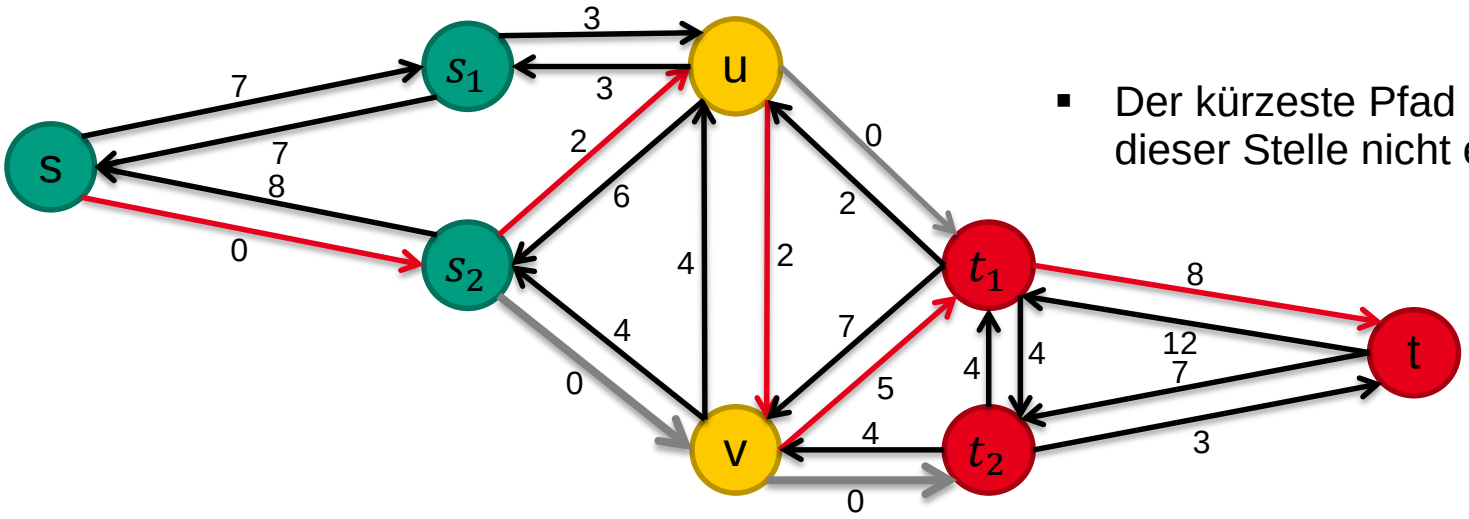


- Rote Kanten kennzeichnen eine Flusserhöhung

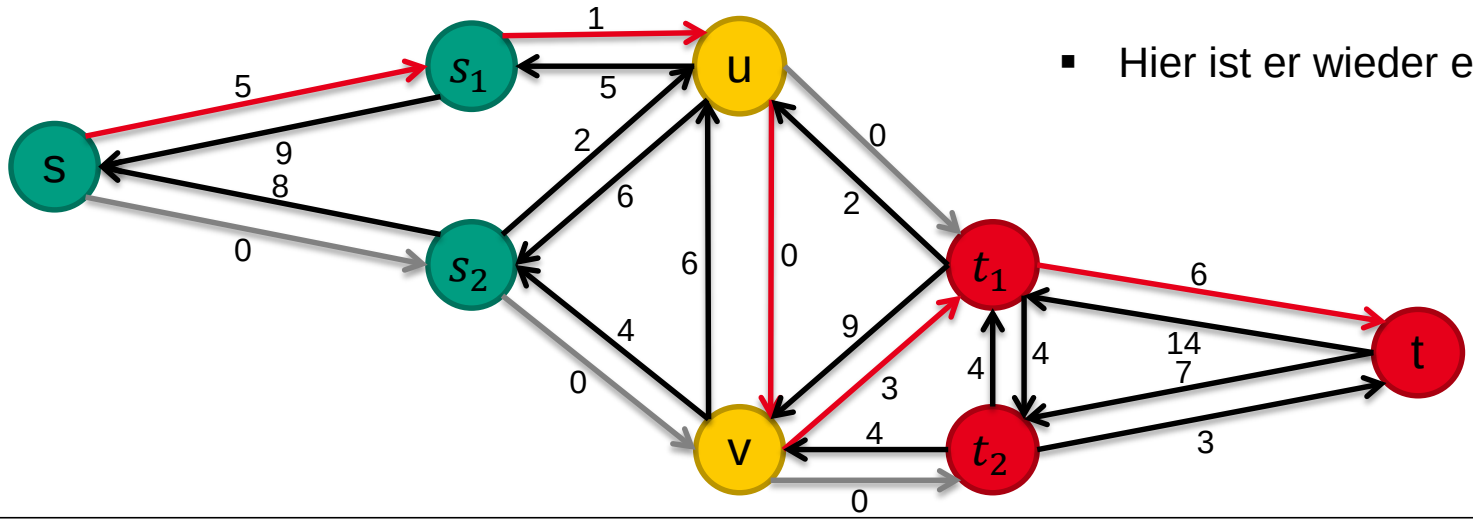


- Graue Kanten zeigen an, dass keine weitere Flusserhöhung entlang der Kante möglich ist

Maximaler Fluss mehrerer Quellen und Senken

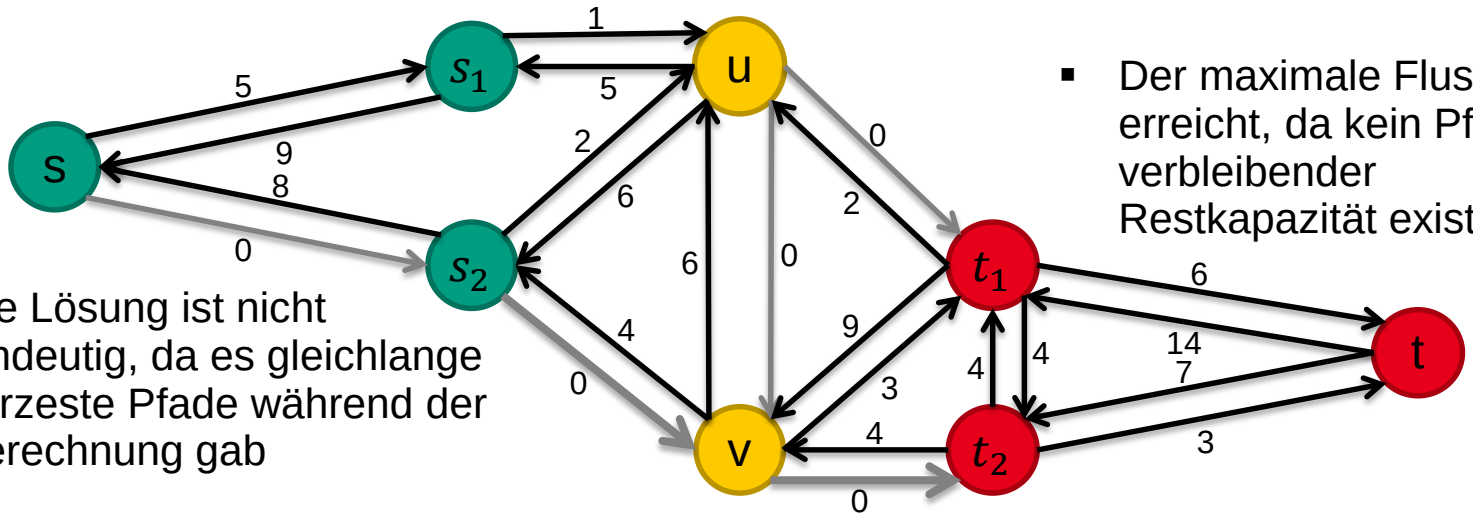


- Der kürzeste Pfad ist an dieser Stelle nicht eindeutig

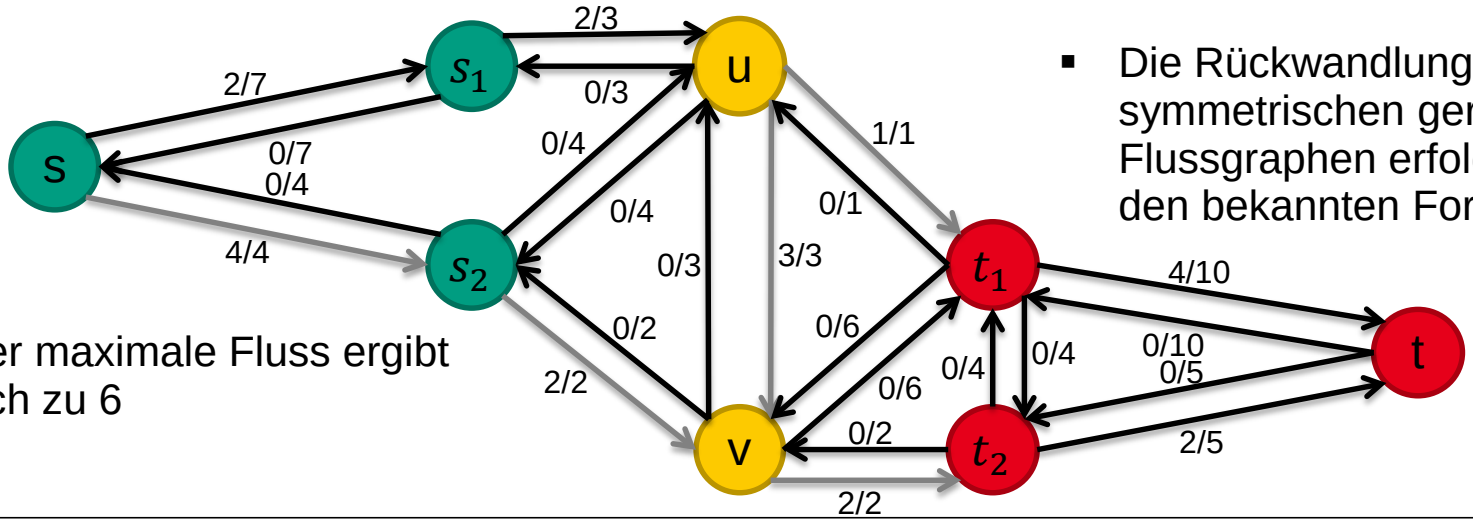


- Hier ist er wieder eindeutig

Maximaler Fluss mehrerer Quellen und Senken



- Der maximale Fluss ist erreicht, da kein Pfad mit verbleibender Restkapazität existiert
- Die Lösung ist nicht eindeutig, da es gleichlange kürzeste Pfade während der Berechnung gab



- Die Rückwandlung in den symmetrischen gerichteten Flussgraphen erfolgt nach den bekannten Formeln
- Der maximale Fluss ergibt sich zu 6

Eigenschaften des Algorithmus von Ford-Fulkerson

TECHNISCHE UNIVERSITÄT DARMSTADT

- Für Flussgraphen mit ganzzahligen oder rationalen Kapazitäten terminiert der Algorithmus stets mit dem maximalen Fluss, der ebenfalls ganzzahlig bzw. rational ist
- Die Lösung ist nicht immer eindeutig (mehrere gleichlange kürzeste Pfade sind möglich), der resultierende maximale Fluss aber schon
- Wählt man in jedem Schritt stets einen kürzesten Pfad, so erhält man den *Algorithmus von Edmonds und Karp* mit der Laufzeitkomplexität:

$$O(|V| * |E|^2)$$

  - mit $|V| :=$ Anzahl der Knoten und $|E| :=$ Anzahl der Kanten

Gliederung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Pfadsuche in Graphen
- Algorithmus von Ford-Fulkerson
- Aufgabenblock 4

Aufgabenblock 4

TECHNISCHE UNIVERSITÄT DARMSTADT

4.1. Ford-Fulkerson Algorithmus implementieren

4.2. Ford-Fulkerson Algorithmus testen