

Flussprobleme in Graphen

**und ihre Anwendung auf
0/1-Netzwerke**

Informatikseminar WS04/05

Claudia Padberg WI4409

Flüsse in Netzwerken

- Transport von Datenmengen durch ein Netzwerk
- Netzwerke besitzen obere Kapazitätsbeschränkungen
- Suche nach Verfahren zur Bestimmung maximaler Flüsse

Übersicht

- Theoretische Grundlagen
- Satz von Ford und Fulkerson
- Bestimmung von Erweiterungswegen
- Algorithmus von Edmonds und Karp
- Algorithmus von Dinic
- 0-1-Netzwerke

Übersicht

- **Theoretische Grundlagen**
 - q-s-Netzwerk
 - q-s-Fluss
 - Wert eines Flusses
 - q-s-Schnitt
 - Erweiterungswege
- Satz von Ford und Fulkerson
- Bestimmung von Erweiterungswegen
- Algorithmus von Edmonds und Karp
- Algorithmus von Dinic
- 0-1-Netzwerke

Theoretische Grundlagen

■ q-s-Netzwerk

kantenbewerteter, gerichteter Graph mit ausgezeichneten Ecken q und s

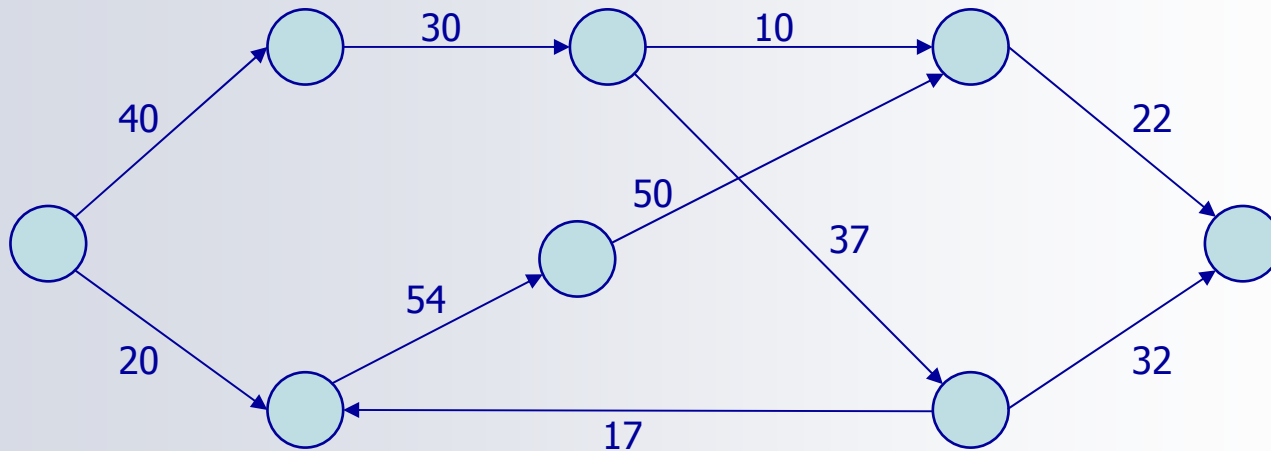
$$N = (G, q, s, c)$$

mit

- $G = (E, K)$
- $E = \{e_1 \dots e_n\}$ Eckenmenge
- $K = \{k_1 \dots k_m\}$ Kantenmenge
- $c(e) =$ Kapazität c der Kante e
- $q =$ Quelle
- $s =$ Senke

Theoretische Grundlagen

- q-s-Netzwerk



Theoretische Grundlagen

■ q-s-Fluss

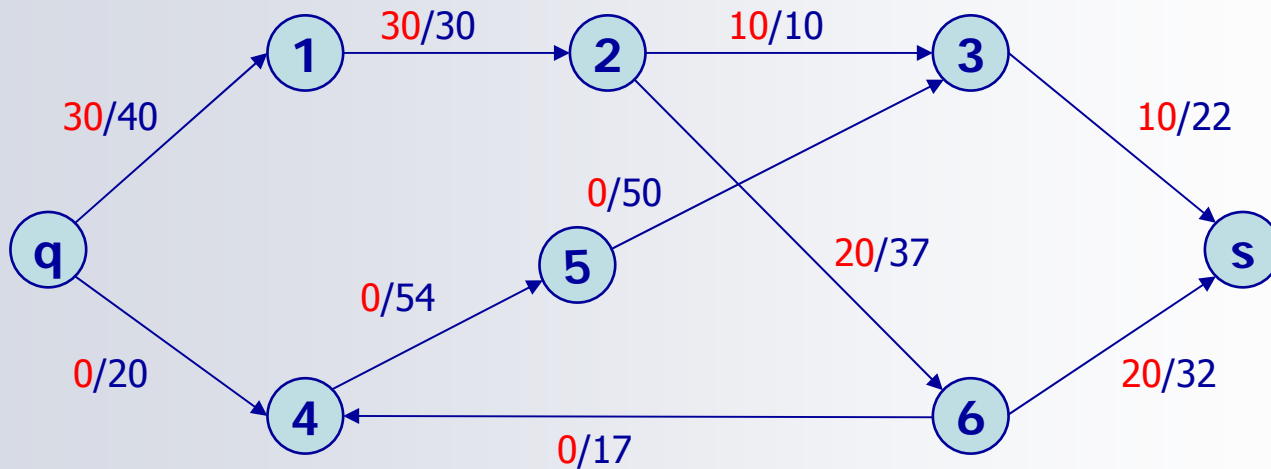
Funktion f , die jeder Kante k des Netzwerkes eine nichtnegative, reelle Zahl zuordnet

Bedingungen

- Der Fluss entspricht maximal der Kapazität der Kante
- Der Fluss in eine Ecke e ist gleich dem Fluss aus dieser Ecke

Theoretische Grundlagen

■ q-s-Fluss



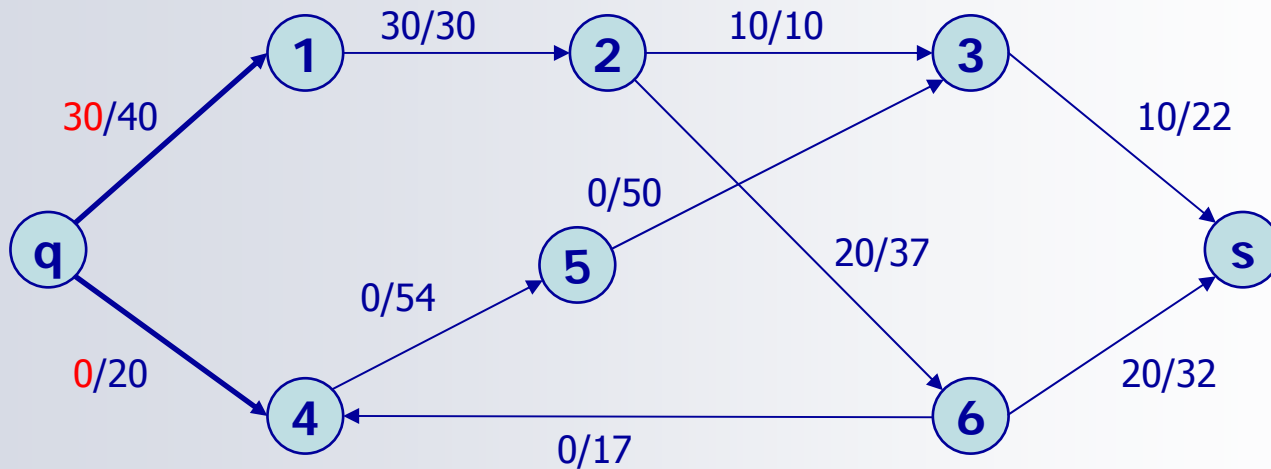
Theoretische Grundlagen

- Wert eines Flusses $|f|$
 - Summe aller Flüsse aus der Quelle
 - Summe aller Flüsse in die Senke
- Was aus der Quelle hinaus fließt, fließt in die Senke hinein

$$|f| = \sum_{k=(q,i) \in E} f(k) - \sum_{k=(i,q) \in E} f(k)$$

Theoretische Grundlagen

- Wert eines Flusses $|f|$



$$|f| = 30$$

Theoretische Grundlagen

- **q-s-Schnitt** von G

Eine Partition von E in die Mengen X und \bar{X} ,
so dass $q \in X$ und $s \in \bar{X}$

$$C(X, \bar{X}) = \sum_{\substack{k=(i,j) \in E \\ i \in X, j \in \bar{X}}} c(k)$$

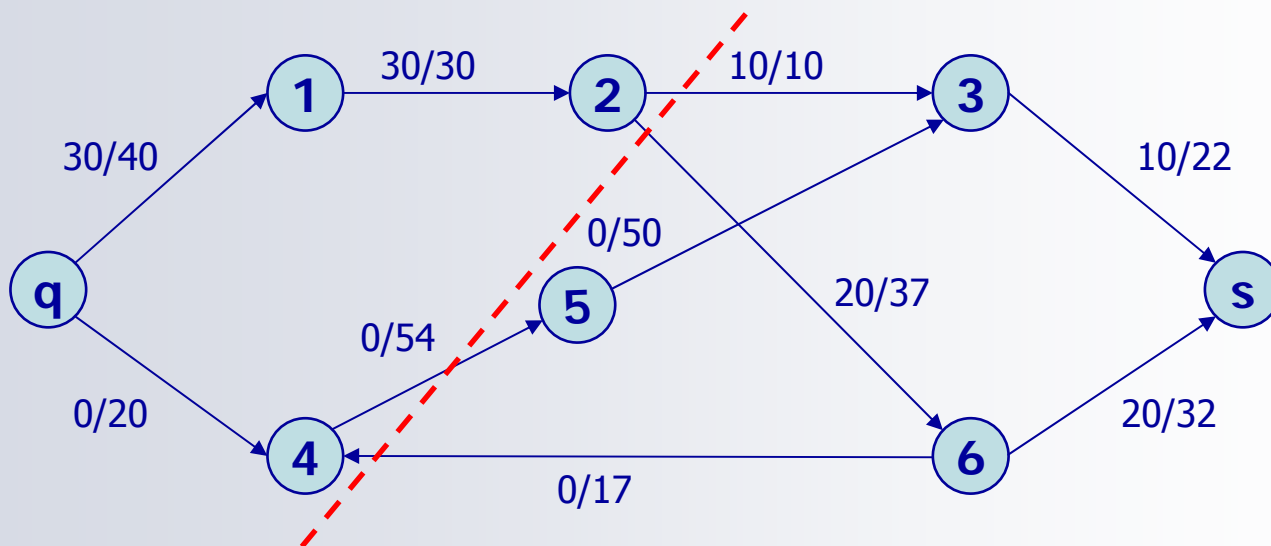
$$f(X, \bar{X}) = \sum_{\substack{k=(i,j) \in E \\ i \in X, j \in \bar{X}}} f(k)$$

Theoretische Grundlagen

- q-s-Schnitt von G

$$X = \{q, 1, 2, 4\}$$

$$\bar{X} = \{3, 5, 6, s\}$$



$$C(X, \bar{X}) = 101$$

$$f(X, \bar{X}) = 30$$

Theoretische Grundlagen

- Erweiterungsweg EW

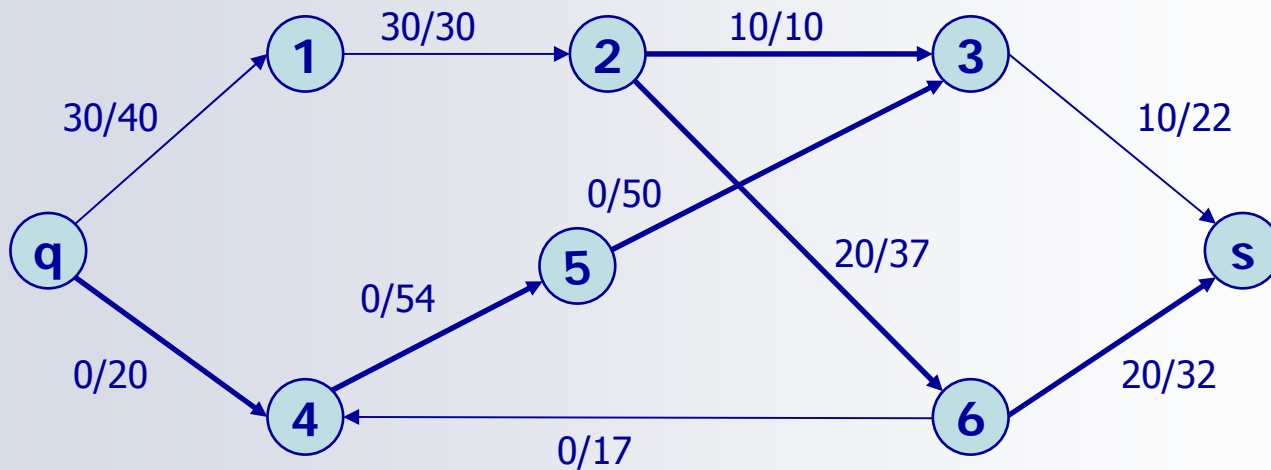
Weg auf dem zu G entsprechenden ungerichteten Graphen G^u , über den ein neuer Fluss mit höherem Wert erzeugt werden kann

Bedingungen

- bei Vorwärtskanten ist $f(k) < c(k)$
- bei Rückwärtskanten ist $f(k) > 0$

Theoretische Grundlagen

- Erweiterungsweg *EW*

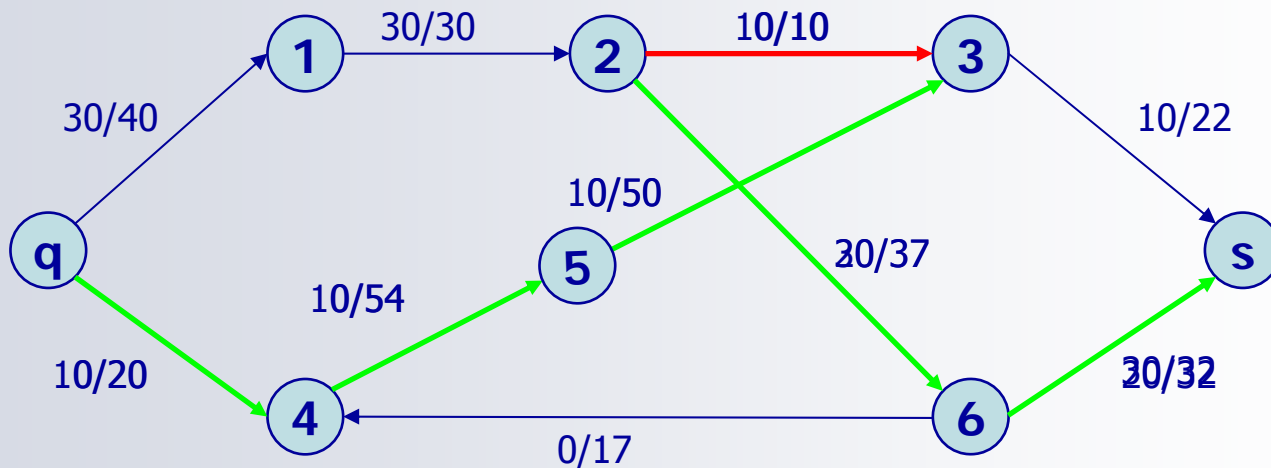


Theoretische Grundlagen

- „engste Stelle“ f_{Δ} des Erweiterungsweges finden durch
 - $f_v = \min \{c(k) - f(k) \mid k \text{ ist VWK des EW}\}$
 - $f_r = \min \{f(k) \mid k \text{ ist RWK des EW}\}$
gibt es keine RWK im EW, setzt man
 $f_r = \infty$
- $f_{\Delta} = \min \{f_v, f_r\}$
 - Fluss auf VWK um f_{Δ} erhöhen
 - Fluss auf RWK um f_{Δ} senken

Theoretische Grundlagen

■ Erweiterungsweg EW



$$f_v = 12$$

$$f_r = 10$$

$$f_{\Delta} = 10$$

Übersicht

- Theoretische Grundlagen
- **Satz von Ford und Fulkerson**
- Bestimmung von Erweiterungswegen
- Algorithmus von Edmonds und Karp
- Algorithmus von Dinic
- 0-1-Netzwerke

Der Satz von Ford und Fulkerson

- Satz

- a) Es sei f ein Fluss eines Netzwerkes. Genau dann ist f ein maximaler Fluss, wenn es keinen Erweiterungsweg bezüglich f gibt.
- b) Der Wert des maximalen Flusses in einem Netzwerk ist gleich der minimalen Kapazität eines Schnittes

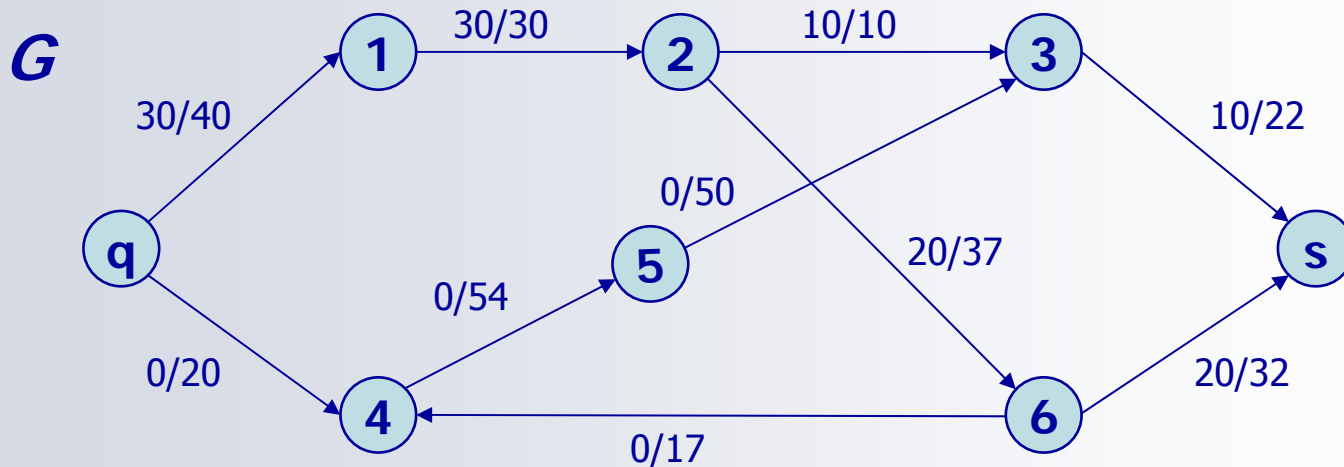
Übersicht

- Theoretische Grundlagen
- Satz von Ford und Fulkerson
- **Bestimmung von Erweiterungswegen**
 - **Konstruktion eines Erweiterungsweges**
 - **Beispiel**
- Algorithmus von Edmonds und Karp
- Algorithmus von Dinic
- 0-1-Netzwerke

Bestimmung von Erweiterungswegen

- Aus dem Netzwerk G wird ein Restnetzwerk G_f konstruiert.
- G_f besitzt die gleiche Eckenmenge wie G und folgende Kanten:
 - k , falls $f(k) < c(k)$,
mit der Bewertung $c(k) - f(k)$
 - $\neg k$, falls $f(k) > 0$,
mit der Bewertung $f(k)$

Bestimmung von Erweiterungswegen



- Bestimmung von EW mit minimaler Anzahl von Kanten mit Hilfe der Breitensuche auf G_f mit Startknoten q
 - EW gefunden, sobald s erreicht wurde
 - Endet die Breitensuche, bevor s erreicht ist, so ist der Fluss bereits maximal
-

Übersicht

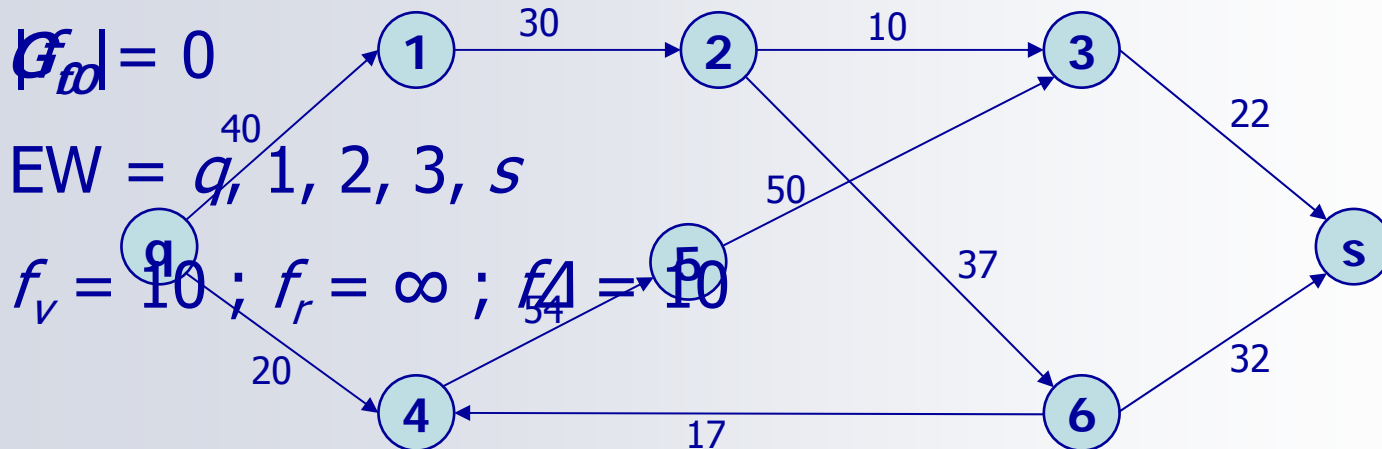
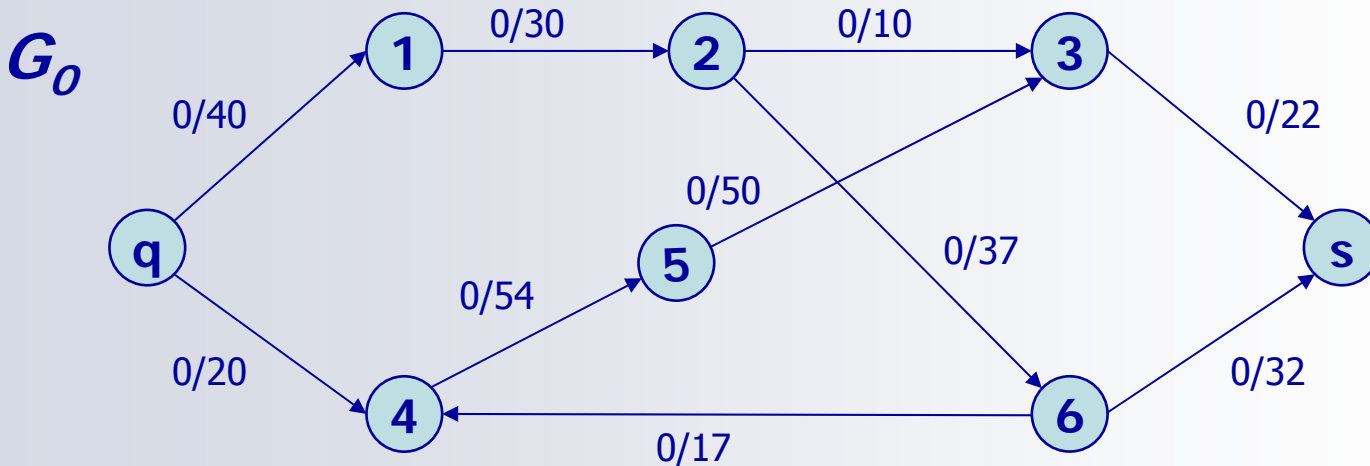
- Theoretische Grundlagen
- Satz von Ford und Fulkerson
- Bestimmung von Erweiterungswegen
- **Algorithmus von Edmonds und Karp**
 - **Vorgehensweise**
 - **Beispiel**
- Algorithmus von Dinic
- 0-1-Netzwerke

Der Algorithmus von Edmonds und Karp

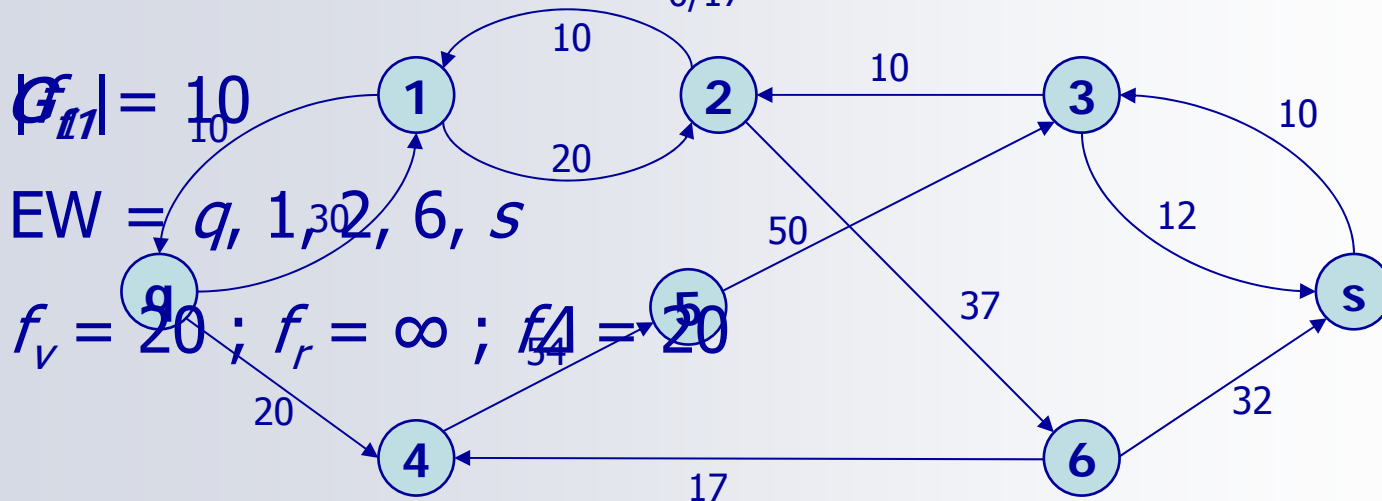
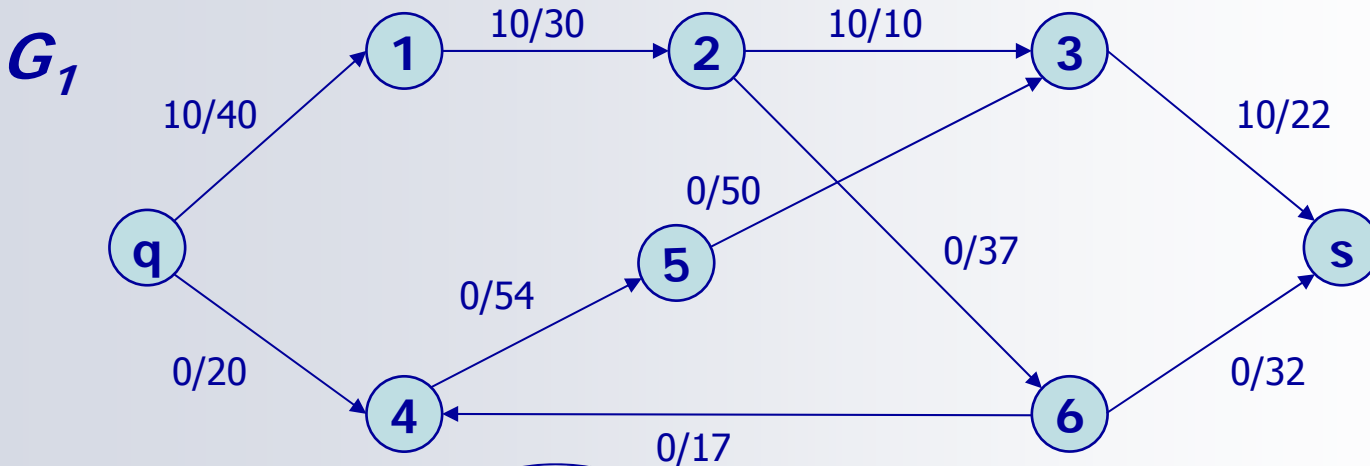
■ Vorgehensweise

- Konstruiere für den Graphen G mit Fluss f den entsprechenden Restgraph G_f
- Finde auf G_f mit Hilfe der Breitensuche einen Erweiterungsweg
- Erhöhe den Fluss um $f\Delta$ auf dem EW
- Wiederhole solange, bis es keinen Erweiterungsweg mehr von q nach s gibt, und damit der maximale Fluss gefunden ist

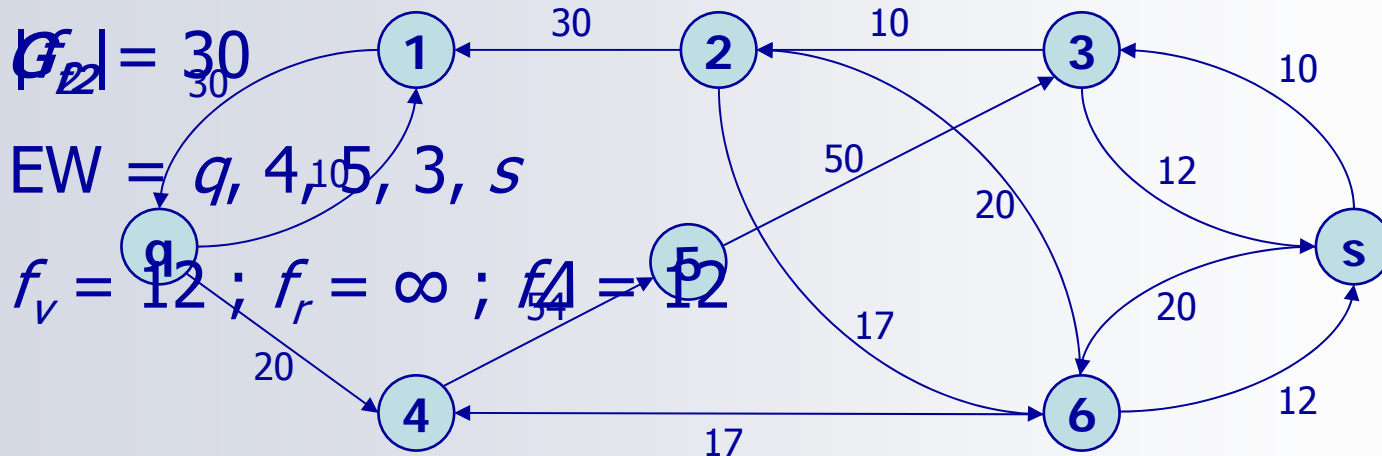
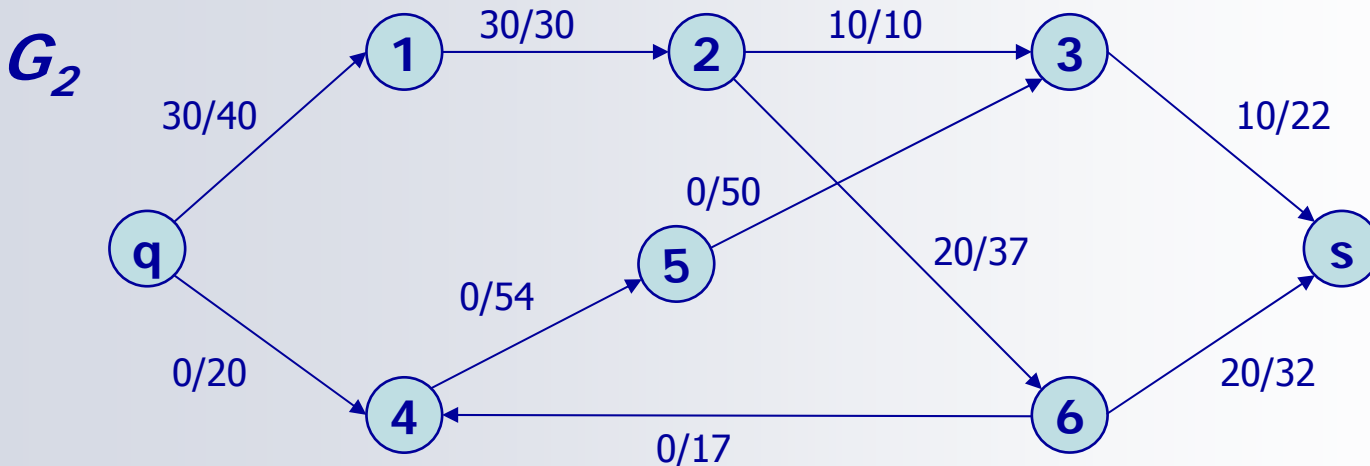
Der Algorithmus von Edmonds und Karp



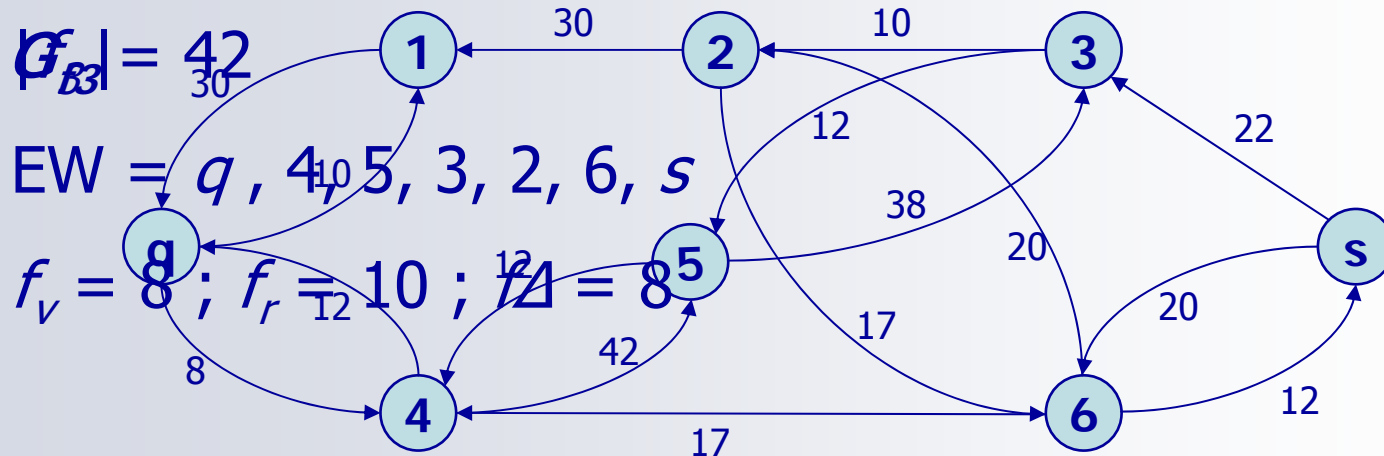
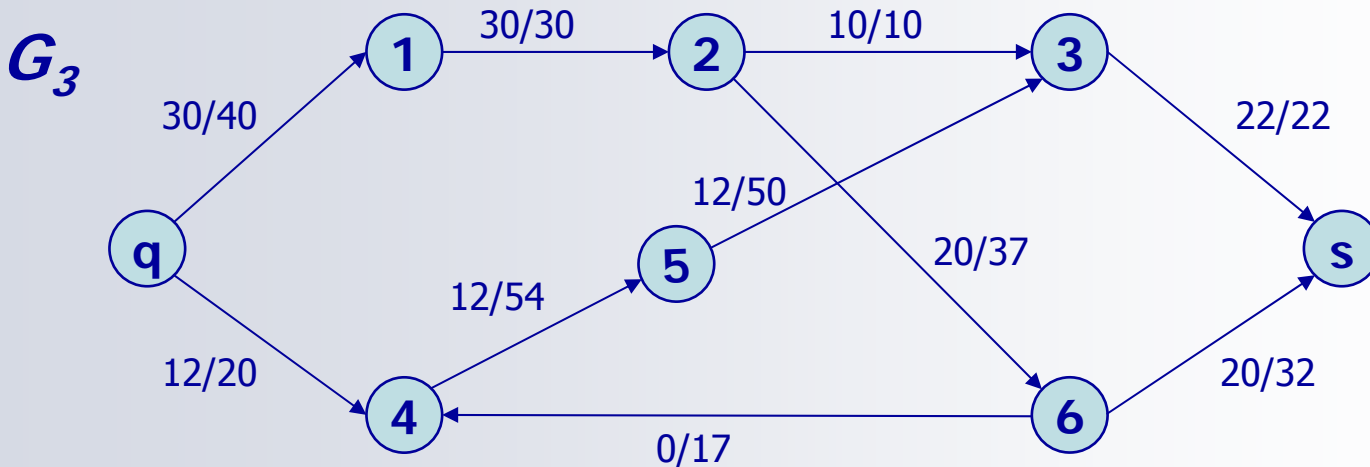
Der Algorithmus von Edmonds und Karp



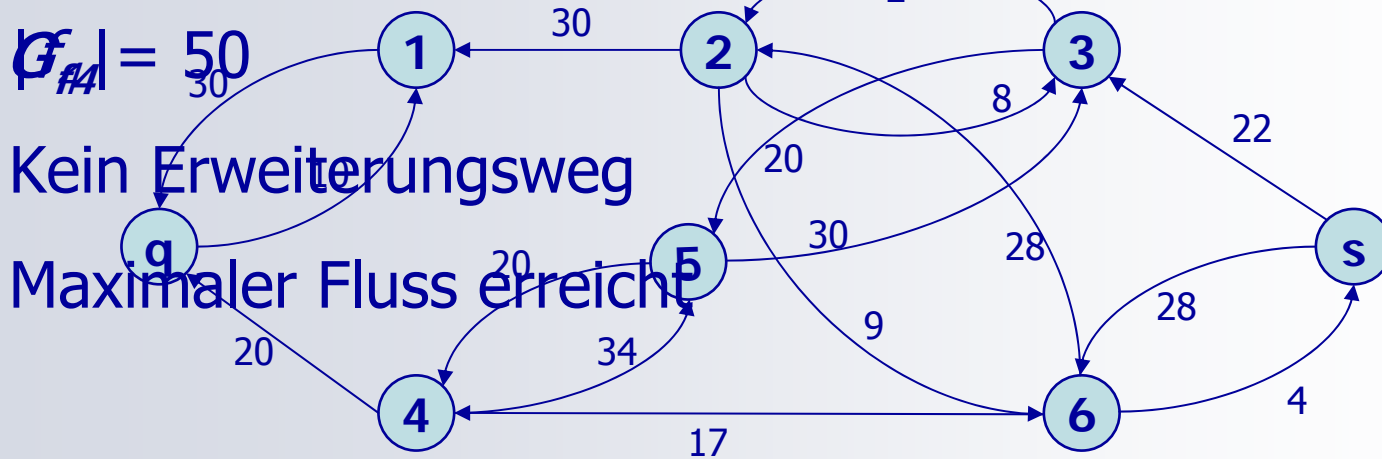
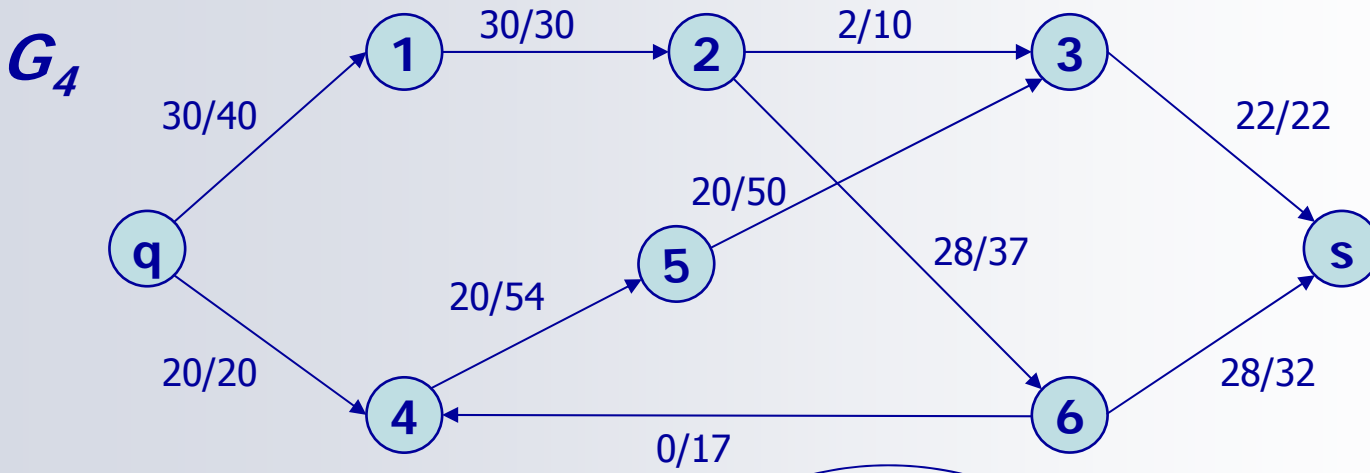
Der Algorithmus von Edmonds und Karp



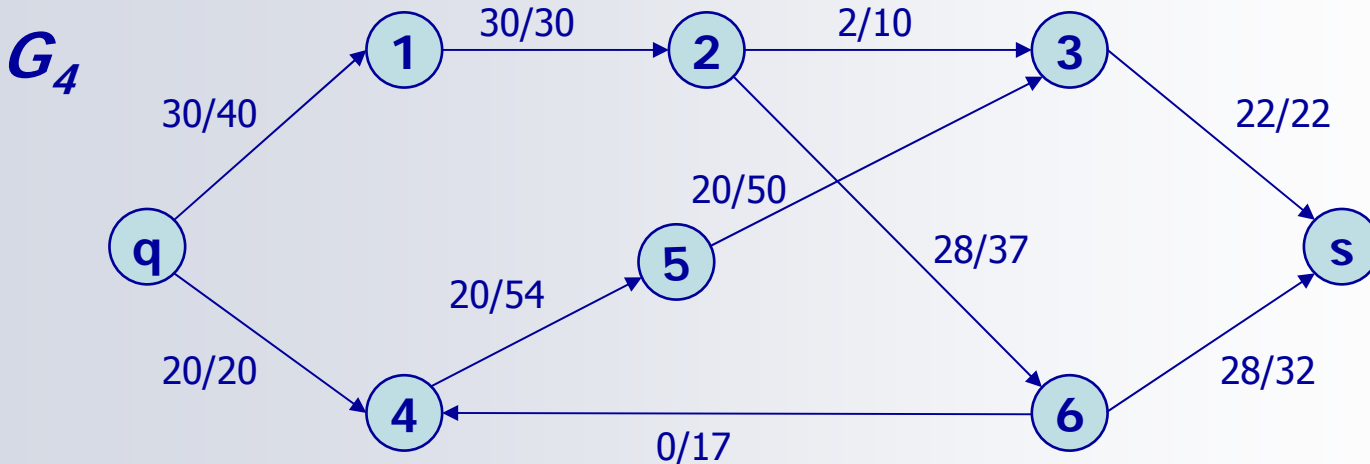
Der Algorithmus von Edmonds und Karp



Der Algorithmus von Edmonds und Karp



Der Algorithmus von Edmonds und Karp



$$|f_4| = 50$$

Übersicht

- Theoretische Grundlagen
- Satz von Ford und Fulkerson
- Bestimmung von Erweiterungswegen
- Algorithmus von Edmonds und Karp
- **Algorithmus von Dinic**
 - **Vorgehensweise**
 - **Beispiel**
- 0-1-Netzwerke

Der Algorithmus von Dinic

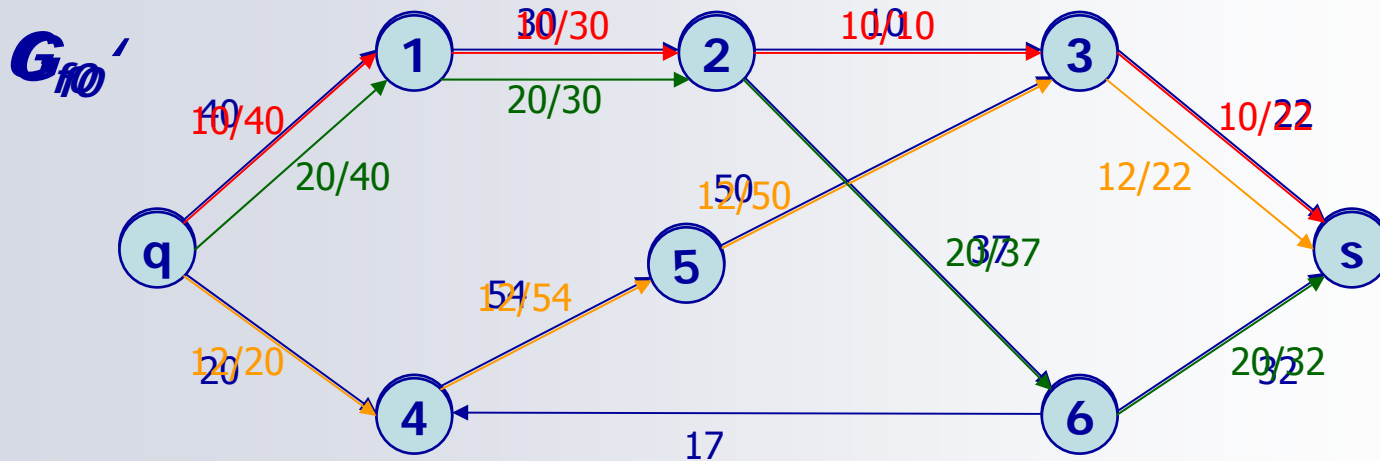
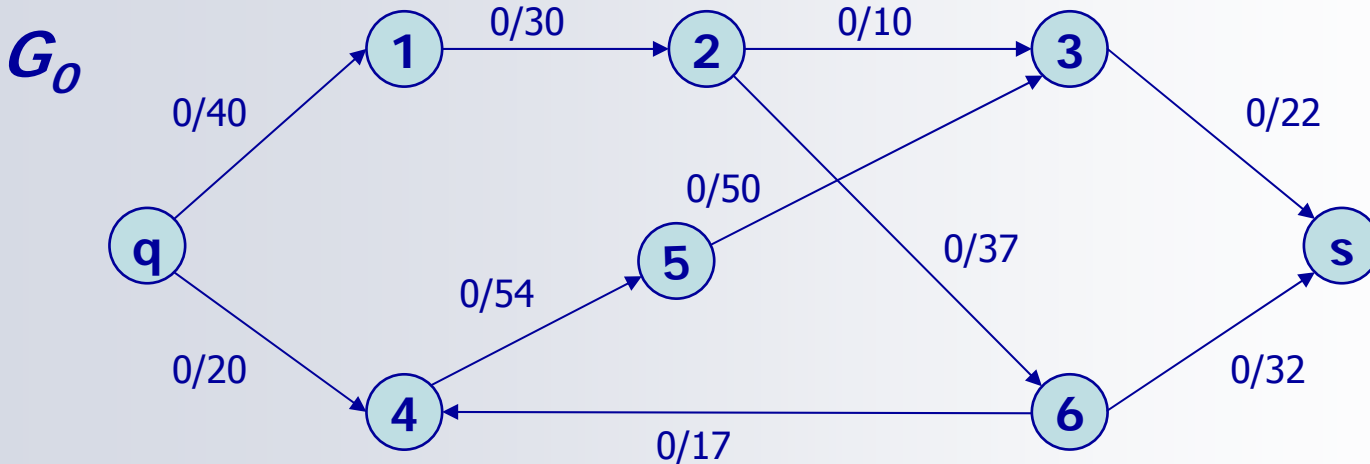
- Betrachtung geeigneter Hilfsnetzwerke
→ *Niveaunetzwerke*
- Konstruktion eines Flusses, der nicht maximal sein muss, für den es aber keinen EW gibt, der nur aus VWK besteht. → *blockierender Fluss*
- Erhöhung des Flusses auf dem gesamten Netzwerk

Der Algorithmus von Dinic

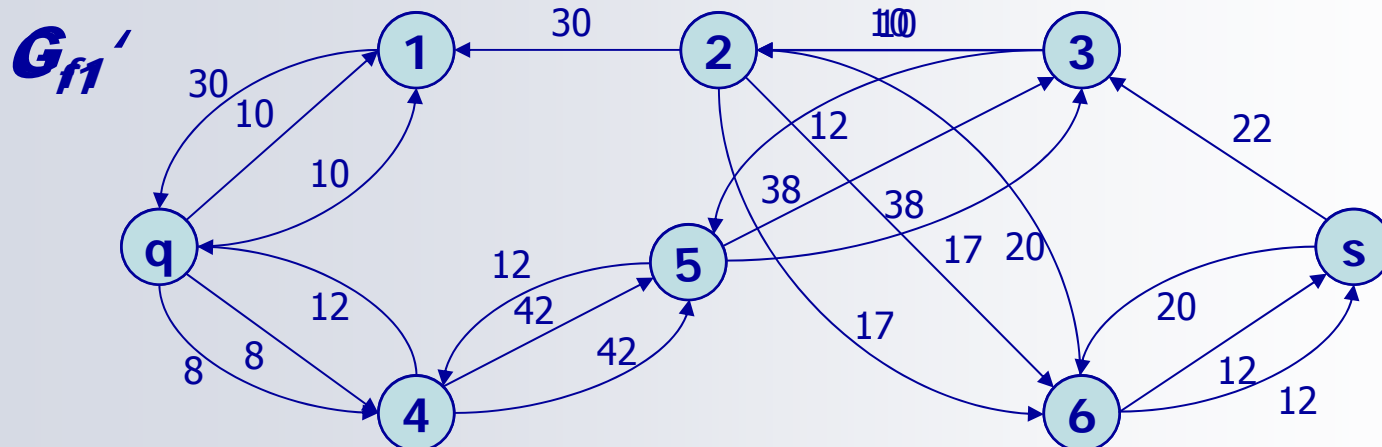
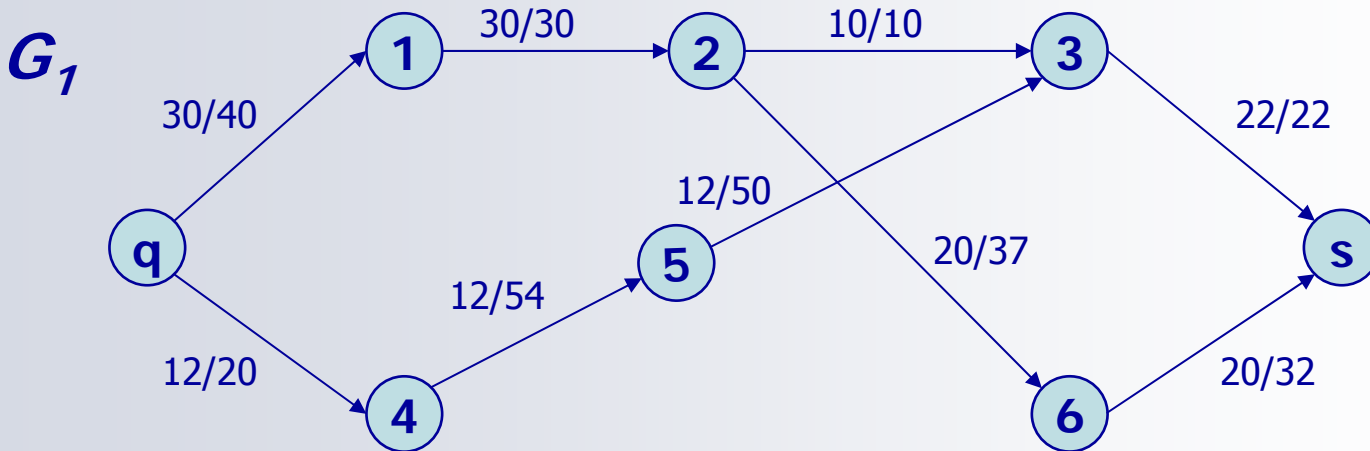
■ Vorgehen:

- Aus dem Netzwerk G wird das Restnetzwerk G_f gebildet
- Aus dem Restnetzwerk G_f wird das Niveaunetzwerk G_f' konstruiert, wobei „überflüssige Kanten“ entfernt werden
- G_f' enthält die Kanten $k = (e, f)$ von G_f , für die gilt
$$Niv(e) + 1 = Niv(f)$$
- In G_f' gilt $g(q) = g^+(s) = 0$
- Alle möglichen Erweiterungswege werden mit der Breitensuche auf G_f' bestimmt und geschichtet

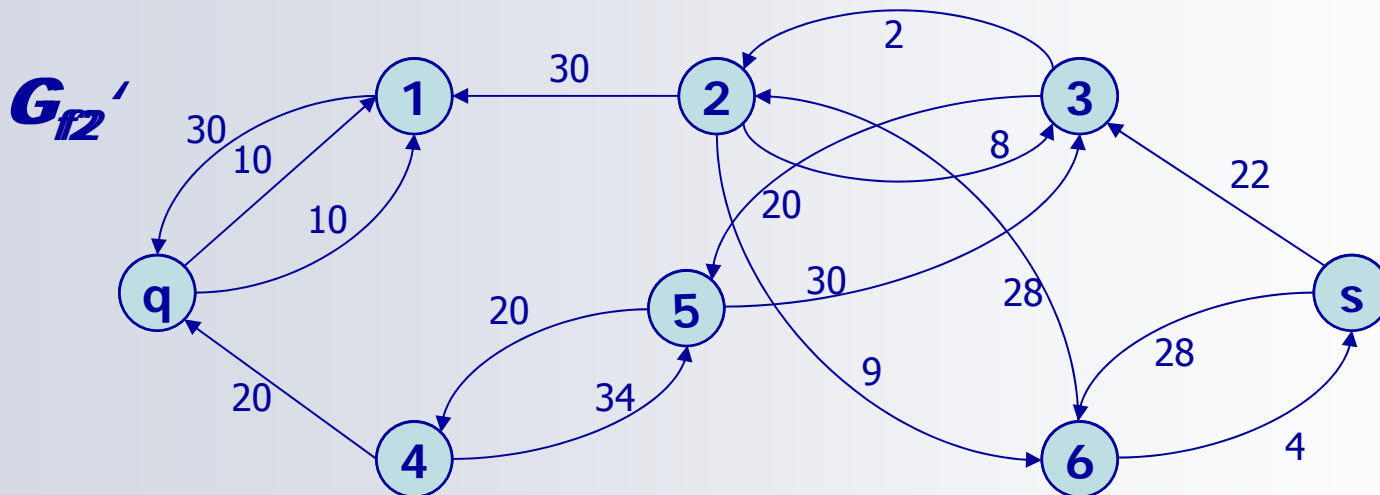
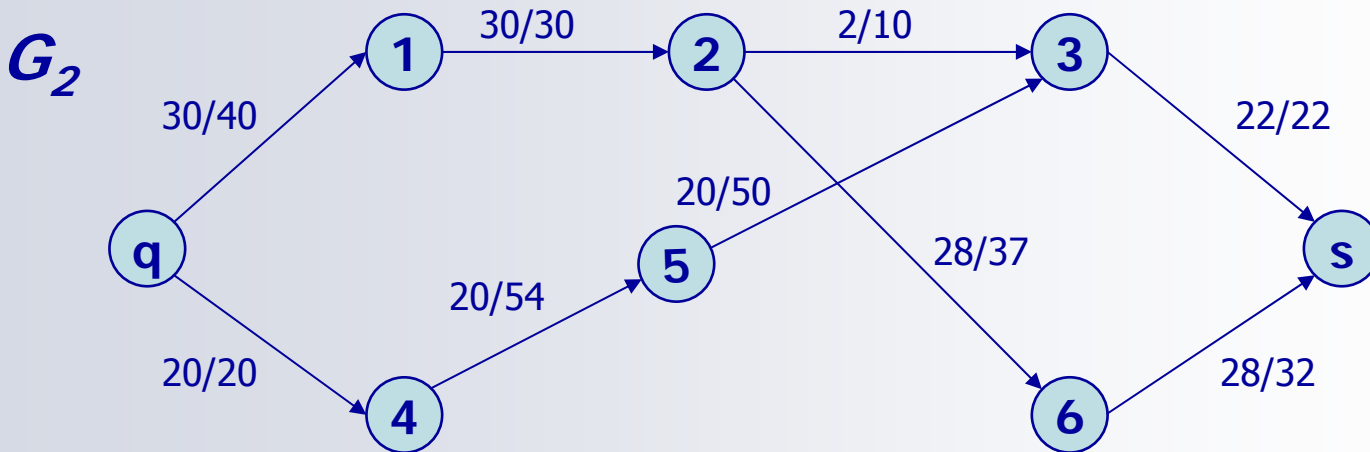
Der Algorithmus von Dinic



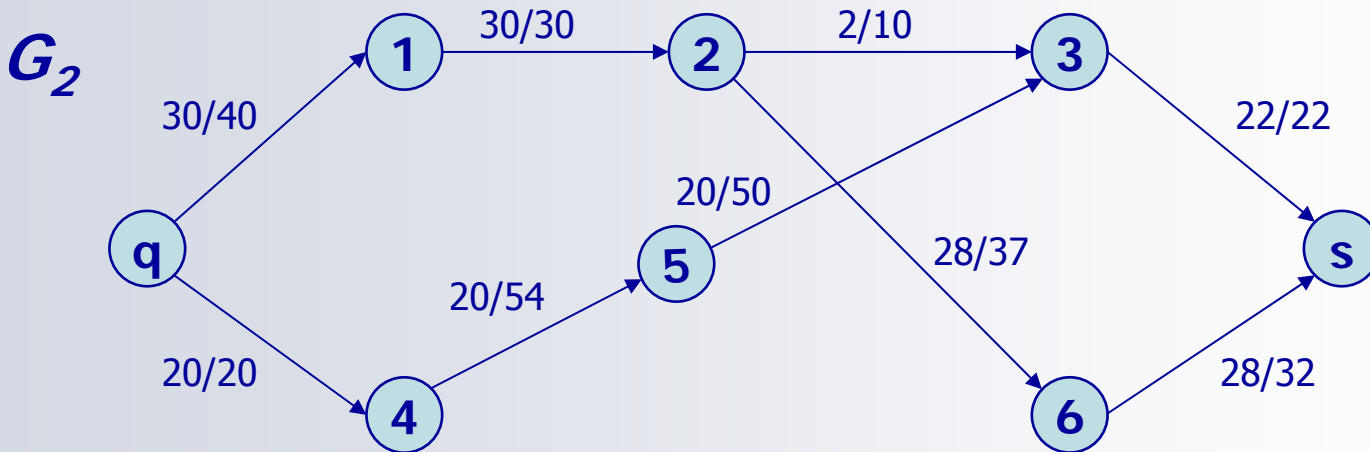
Der Algorithmus von Dinic



Der Algorithmus von Dinic



Der Algorithmus von Dinic



- Im Niveaunetzwerk gibt es keinen weiteren Erweiterungsweg
- Maximaler Fluss erreicht mit $|f_2| = 50$

Übersicht

- Theoretische Grundlagen
- Satz von Ford und Fulkerson
- Bestimmung von Erweiterungswegen
- Algorithmus von Edmonds und Karp
- Algorithmus von Dinic
- **0-1-Netzwerke**

0-1-Netzwerke

- Spezialfall von allgemeinen Netzwerken
- Jede Kante besitzt die Kapazität 0 oder 1
- Es existiert ein maximaler Fluss f , der auf jeder Kante den Wert 1 oder 0 hat
- Es gibt $|f|$ Wege von q nach s , welche paarweise keine Kante gemeinsam haben
→ $|f|$ kantendisjunkte Wege

**Vielen Dank für Ihre
Aufmerksamkeit!**