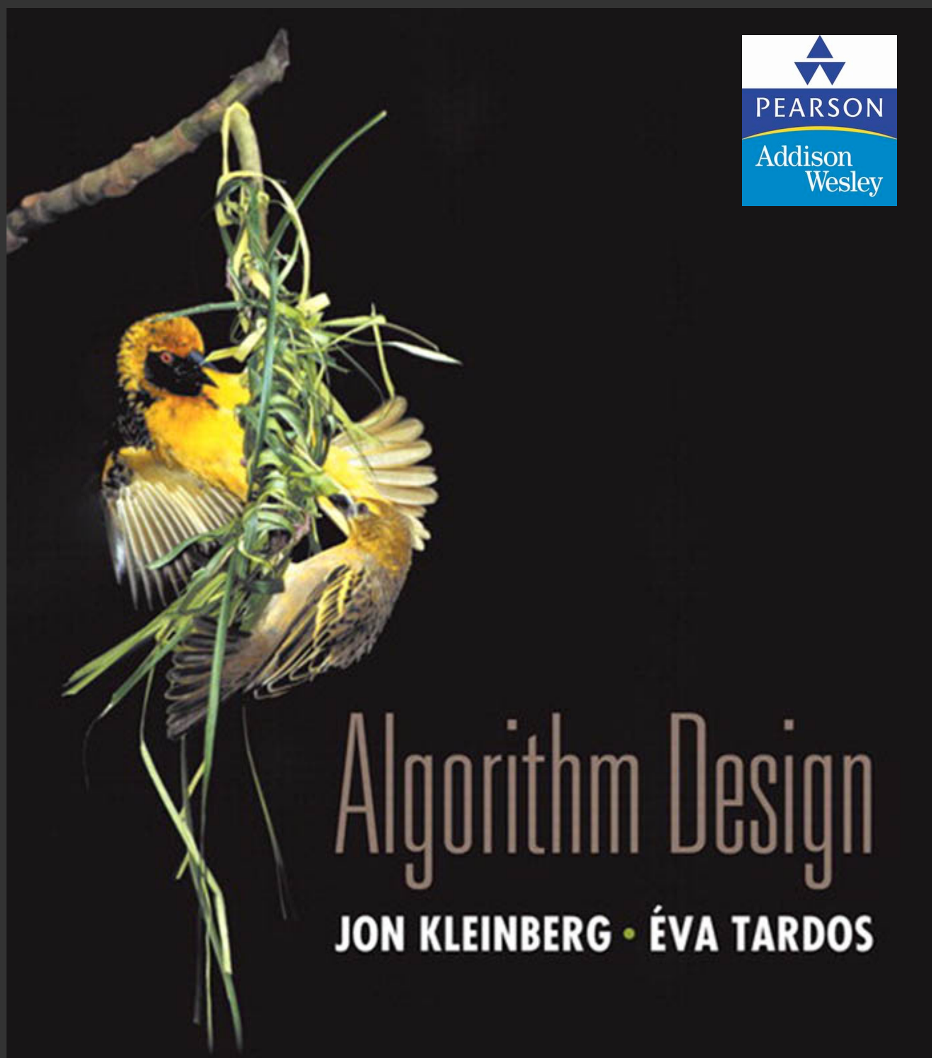


Traduzione e adattamento di Vincenzo Bonifaci
Original lecture slides by Kevin Wayne

http://www.cs.princeton.edu/~wayne/kleinberg-tardos

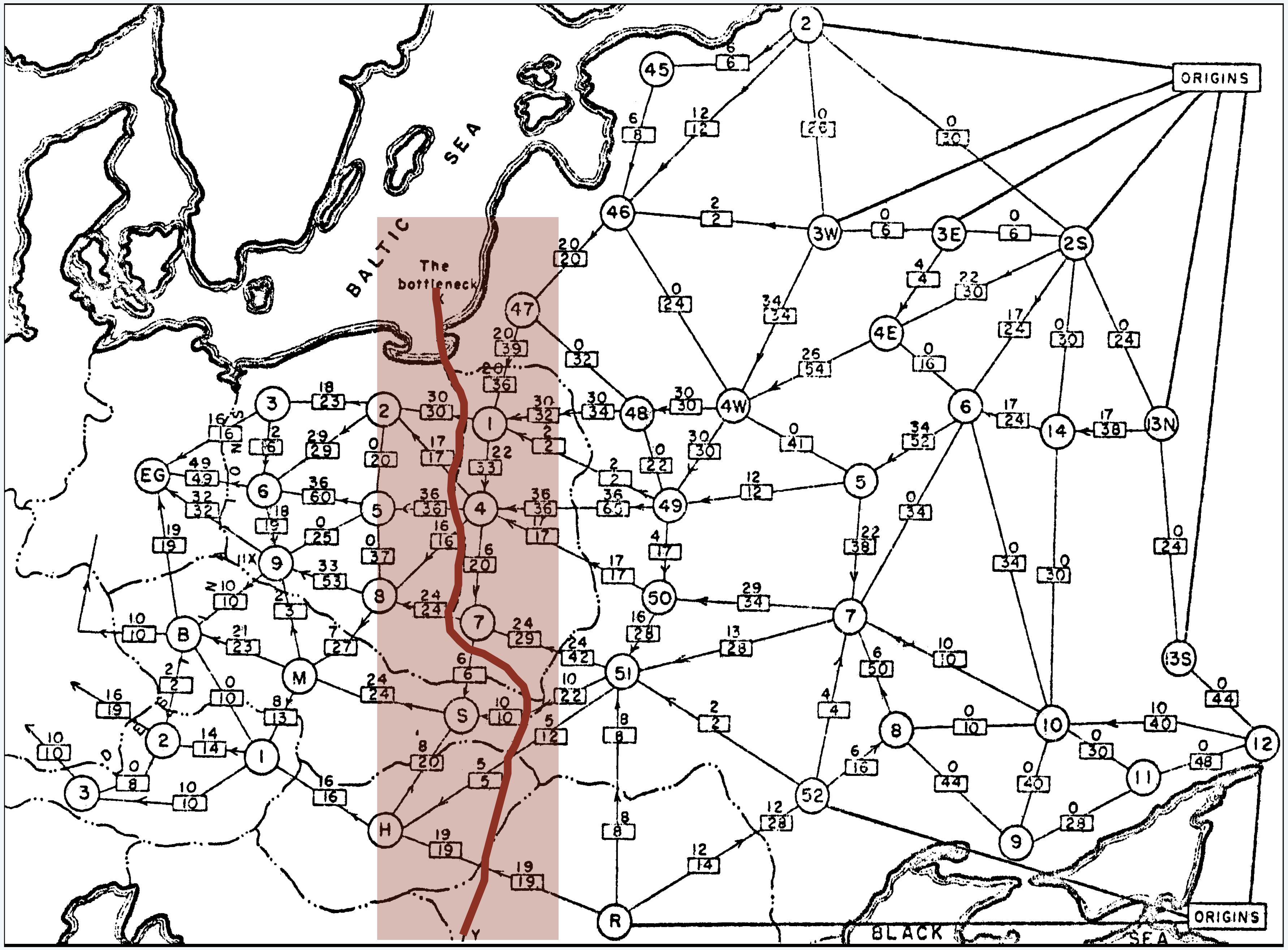
7. Flussi di Rete II

- *abbinamento bipartito*
- *cammini disgiunti*
- *estensioni al massimo flusso*
- *progetto di sondaggi*
- *schedulazione di linee aeree*
- *segmentazione di immagini*
- *selezione di progetti*

Last updated on 27/04/2020 09:01

Applicazione del minimo taglio (RAND anni 1950)

Scopo del "mondo libero". Tagliare i rifornimenti (se la Guerra Fredda divenisse guerra).



rete ferroviaria che collegava l'Unione Sovietica con i paesi Est Europei

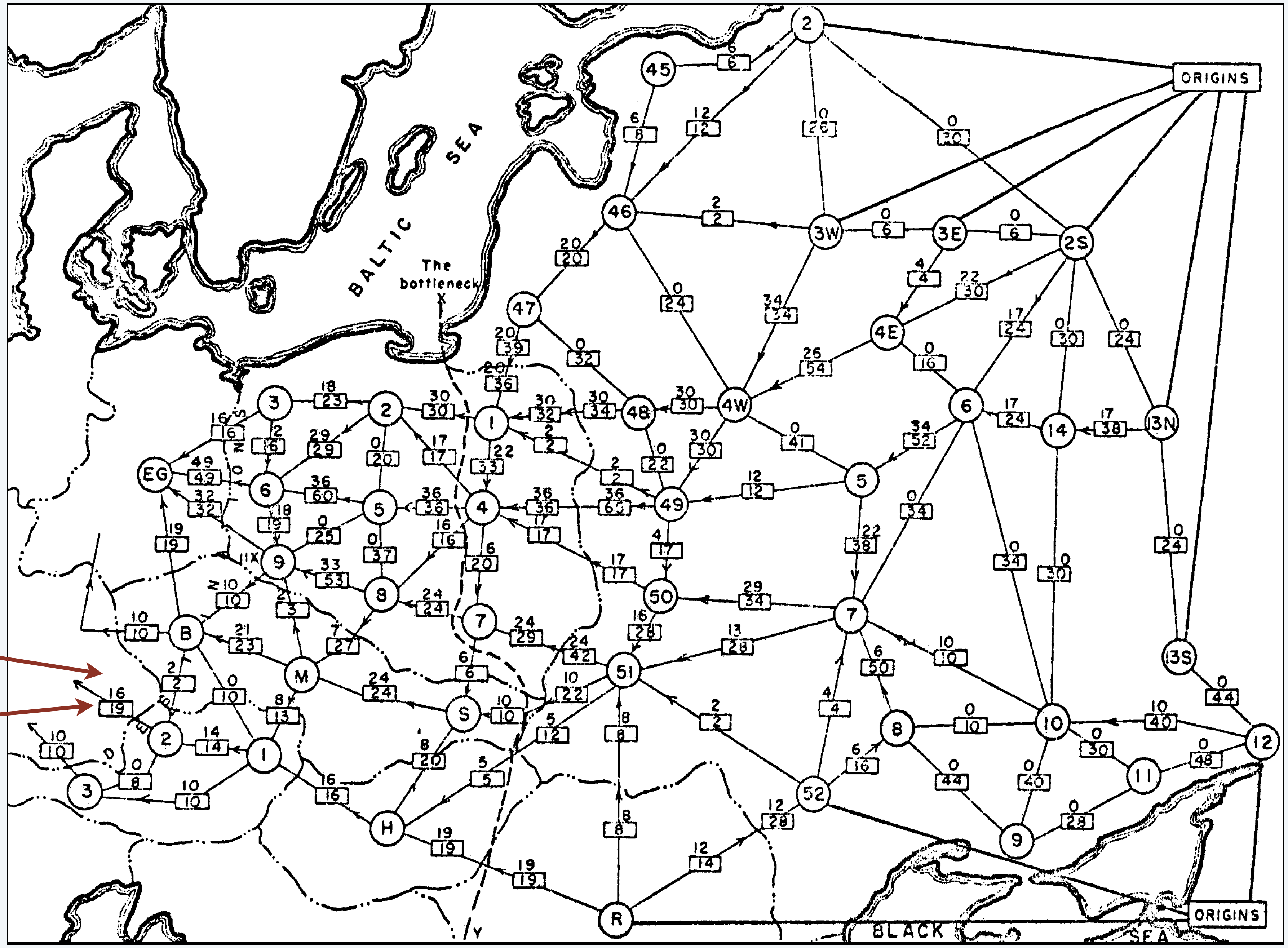
(mappa desecretata dal Pentagono nel 1999)

Applicazione del massimo flusso (Tolstoĭ anni 1930)

Scopo dell'USSR. Massimizzare il flusso di rifornimenti all'Europa dell'Est.

flusso

capacità



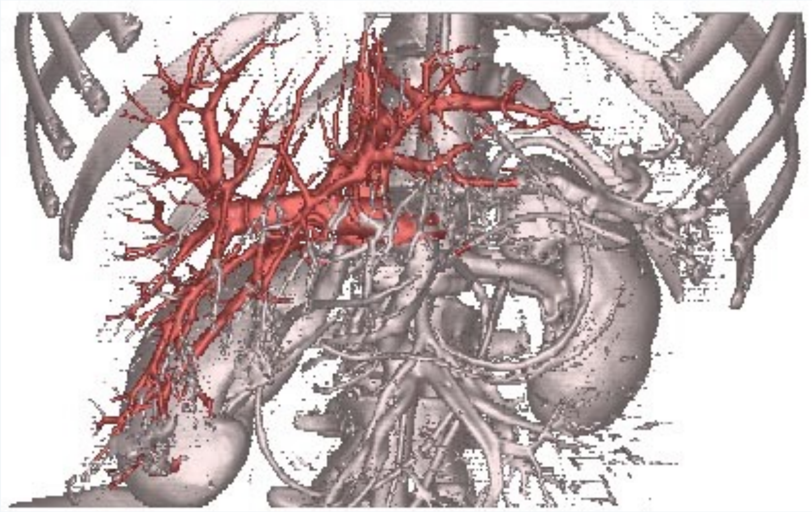
rete ferroviaria che collegava l'Unione Sovietica con i paesi Est Europei

(mappa desecretata dal Pentagono nel 1999)

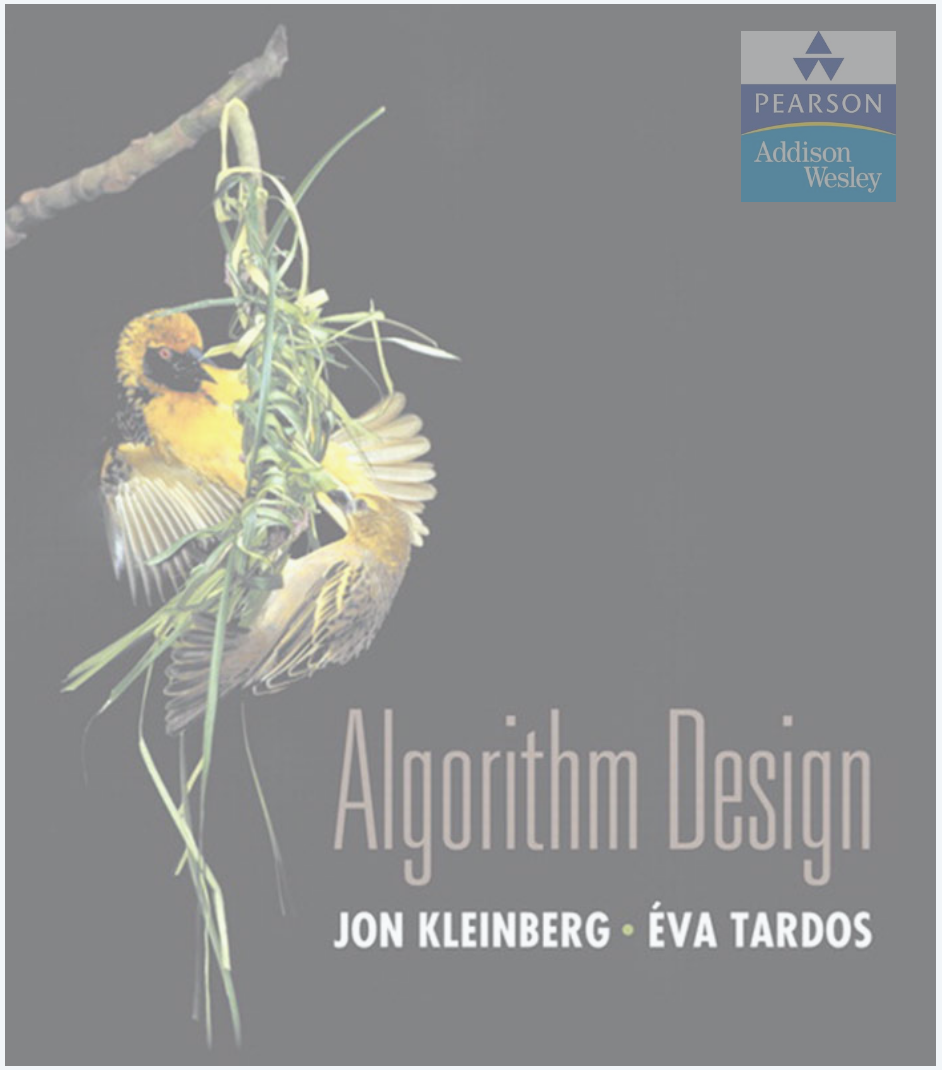
Applicazioni del massimo flusso e del minimo taglio

I modelli di massimo flusso e di minimo taglio sono largamente applicabili.

- Data mining.
- Open-pit mining.
- Abbinamenti bipartiti.
- Affidabilità di rete.
- Eliminazione nel baseball.
- Segmentazione di immagini.
- Connettività di rete.
- Campi aleatori Markoviani.
- Calcolo distribuito.
- Sicurezza di dati statistici.
- Abbinamenti stabili egalitari.
- Riconoscimento di intrusioni di rete.
- Ricostruzione di scene da più telecamere.
- Piazzamento di sensori per la sicurezza militare.
- Molti altri esempi.



segmentazione del fegato e della vascolarizzazione epatica



SECTION 7.5

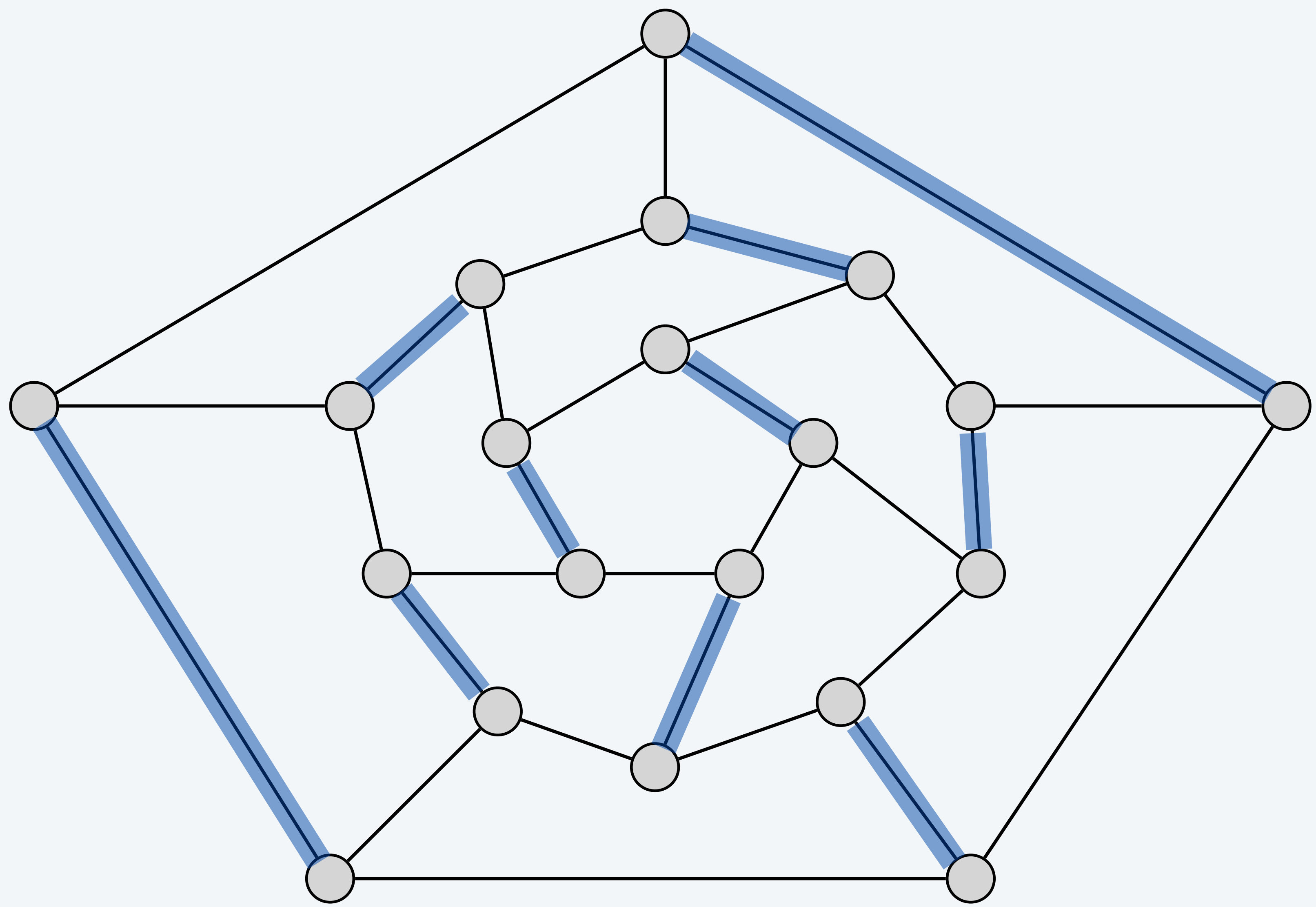
7. FLUSSI DI RETE II

- *abbinamento bipartito*
- *disjoint paths*
- *extensions to max flow*
- *survey design*
- *airline scheduling*
- *image segmentation*
- *project selection*
- *baseball elimination*

Abbinamento [Matching]

Def. Dato un grafo non orientato $G = (V, E)$, un sottoinsieme di archi $M \subseteq E$ è un abbinamento se ogni nodo appare in al più un arco di M.

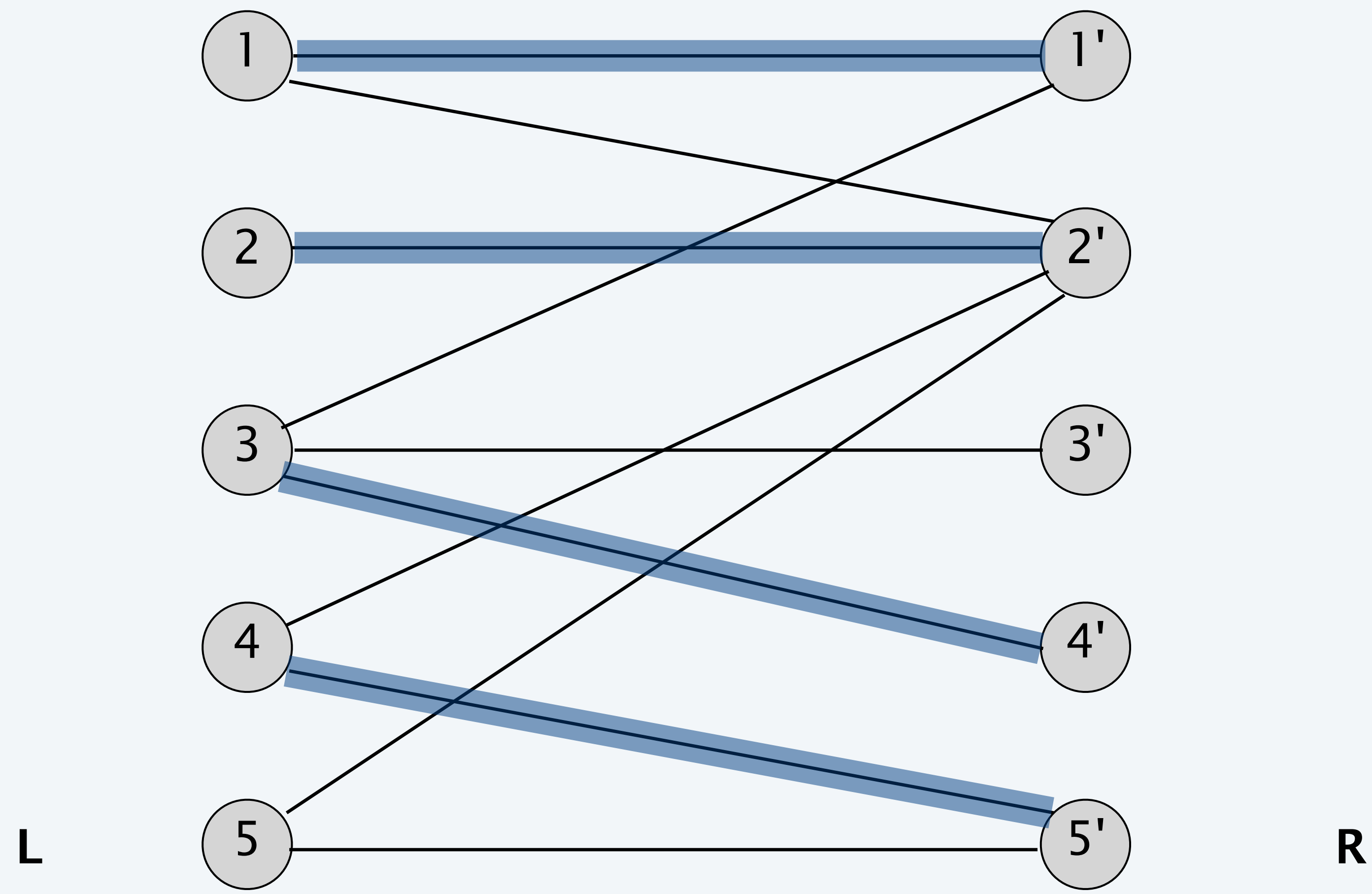
Abbinamento massimo. Dato un grafo G, trovare un abbinamento a cardinalità massima.



Abbinamento bipartito

Def. Un grafo G è bipartito se i nodi possono essere partizionati in due sottoinsiemi L ed R in modo che ogni arco colleghi un nodo in L con un nodo in R.

Abbinamento bipartito. Dato un grafo bipartito $G = (L \cup R, E)$, trovare un abbinamento a cardinalità massima.

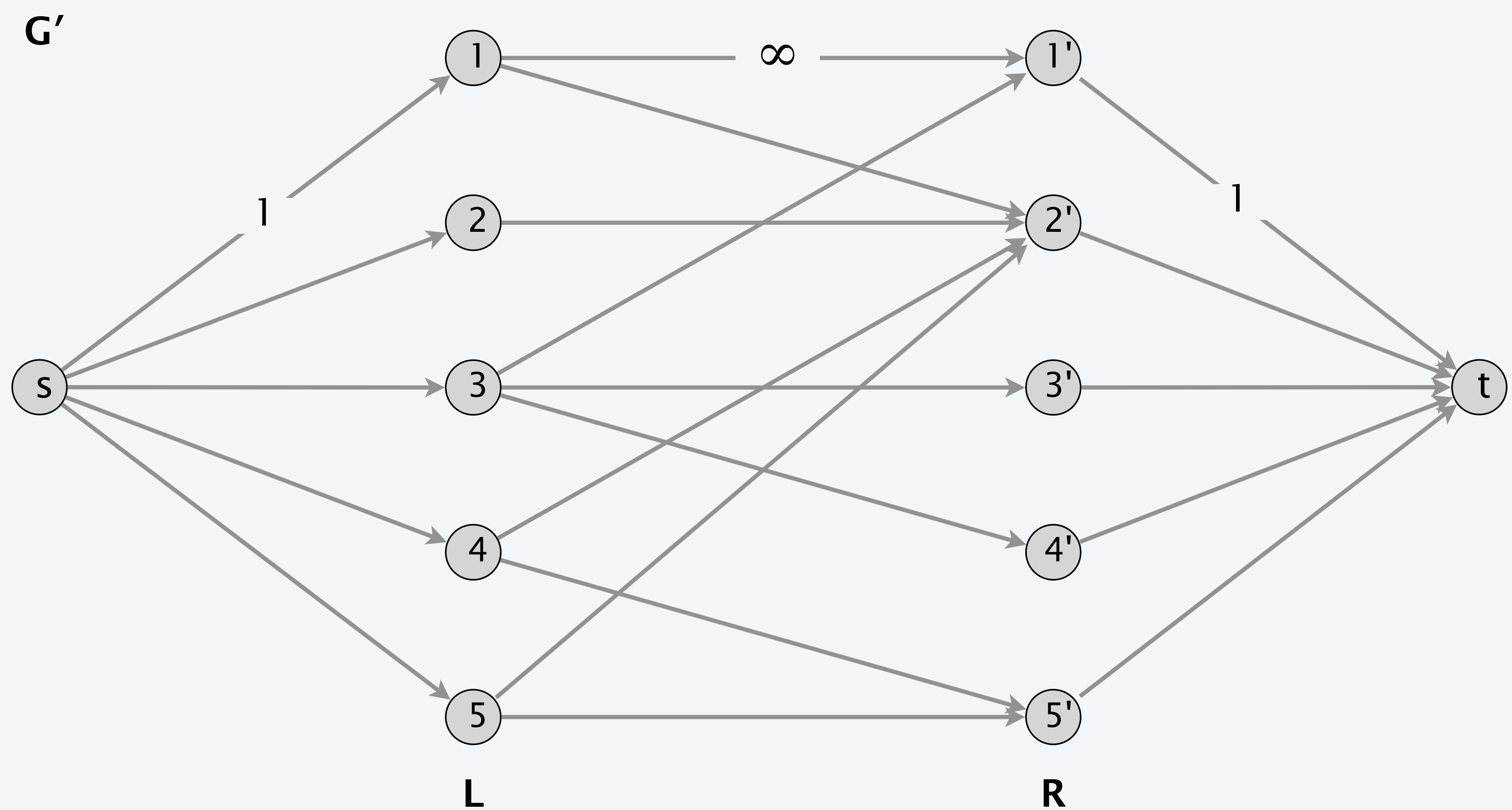


abbinamento: 1–1', 2–2', 3–4', 4–5'

Abbinamento bipartito: formulazione come massimo flusso

Formulazione.

- Crea un digrafo $G' = (L \cup R \cup \{s, t\},\ E')$.
- Orienta tutti gli archi da L ad R, dandogli capacità infinita (o unitaria).
- Aggiungi archi a capacità 1 da s verso ogni nodo in L.
- Aggiungi archi a capacità 1 da ogni nodo in R verso t.



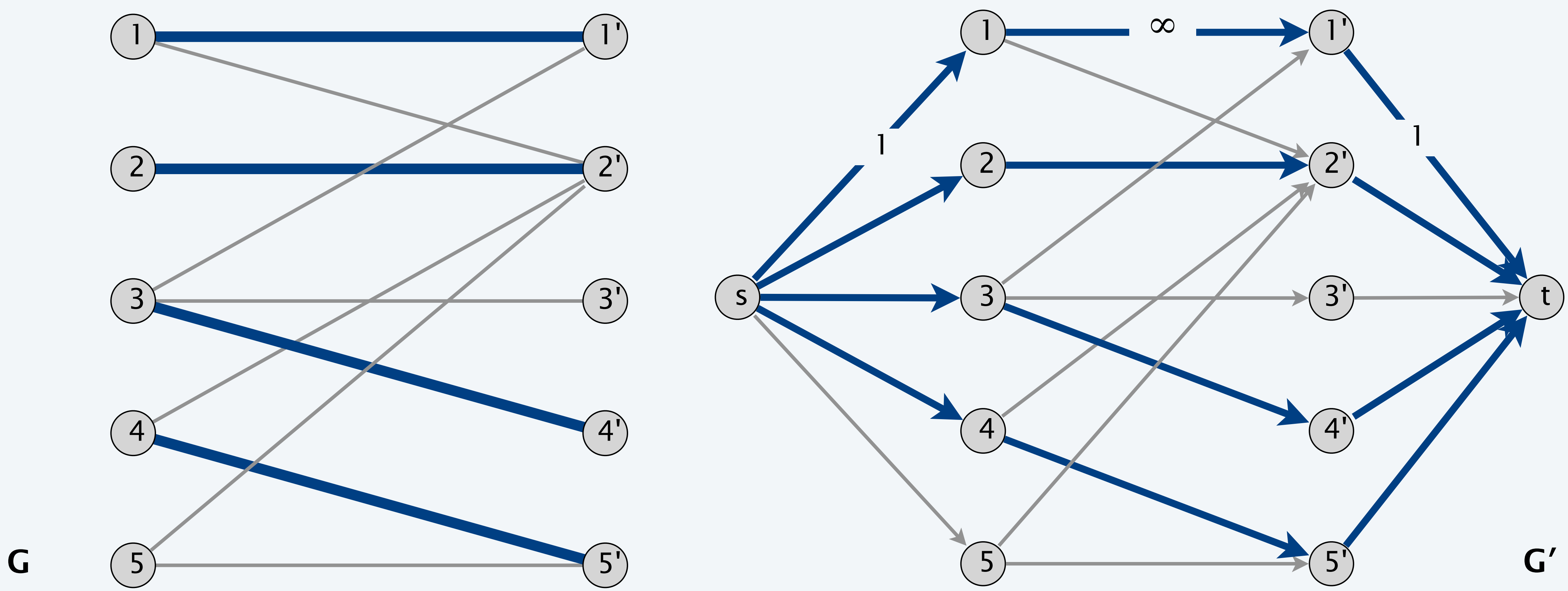
Formulazione come massimo flusso: dimostrazione di correttezza

Teorema. Corrispondenza 1–1 tra abbinamenti di cardinalità k in G e flussi interi di valore k in G'.

per ogni arco e: $f(e) \in \{ 0, 1 \}$

Dim. $\Rightarrow$

- Sia M un abbinamento in G di cardinalità k.
- Sia f il flusso che manda 1 unità su ognuno dei k cammini corrispondenti.
- f è un flusso di valore k. ▪



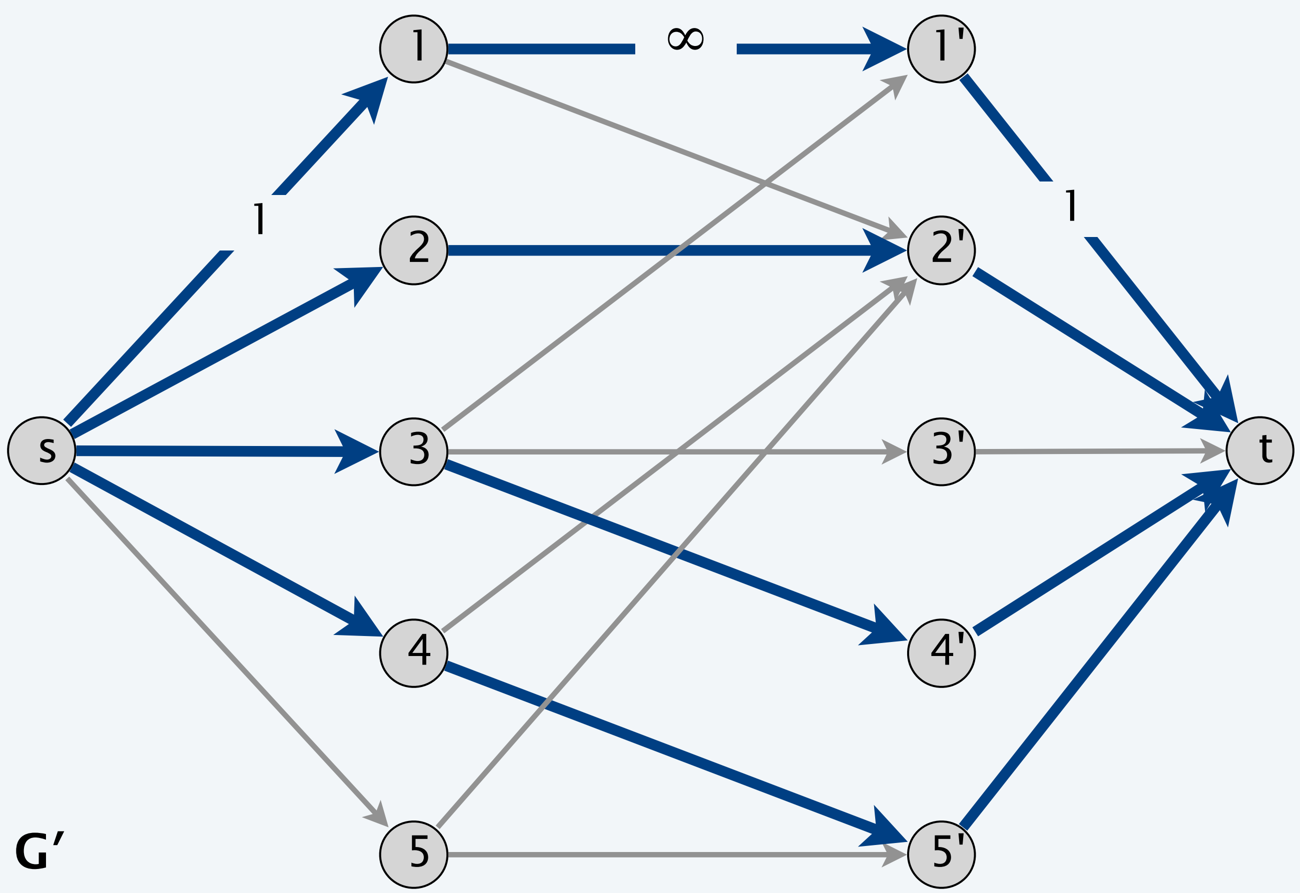
Formulazione come massimo flusso: dimostrazione di correttezza

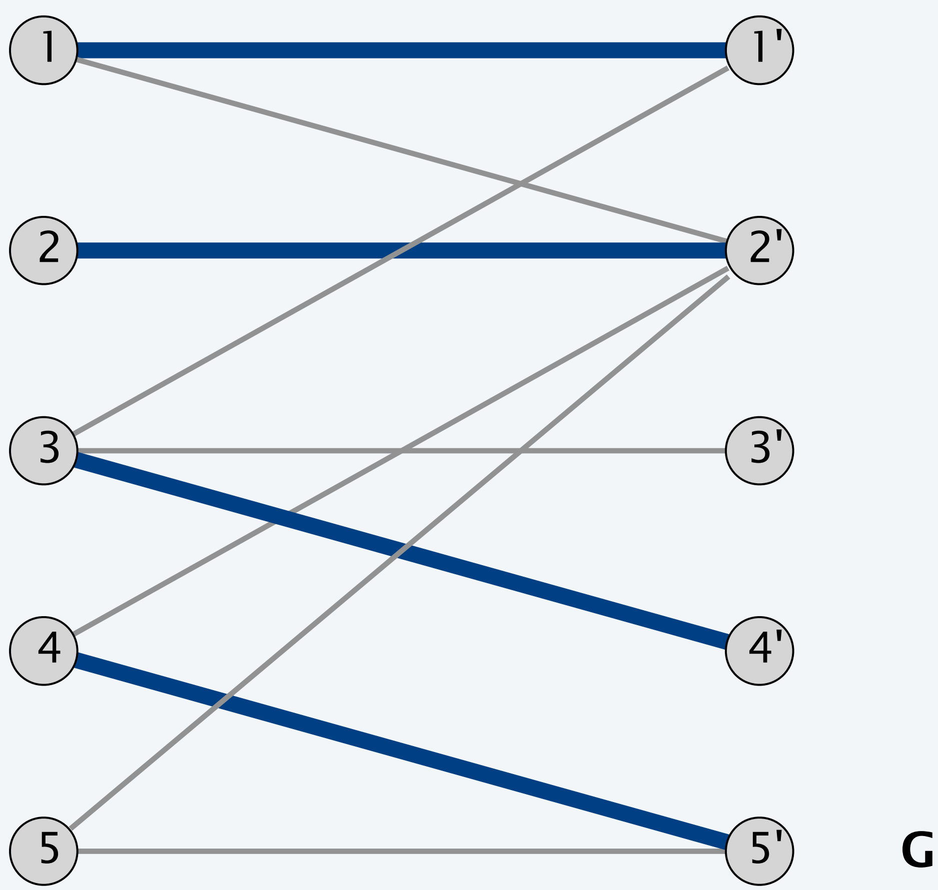
Teorema. Corrispondenza 1–1 tra abbinamenti di cardinalità k in G e flussi interi di valore k in G'.

per ogni arco e: $f(e) \in \{ 0, 1 \}$

Dim. $\Leftarrow$

- Sia f un flusso intero in G' di valore k.
- Sia M = insieme di archi da L ad R con $f(e) = 1$.
  - ogni nodo in L ed R partecipa in al più un arco in M
  - $|M| = k$: applica il lemma valore del flusso al taglio $(L \cup \{s\}, R \cup \{t\})$ ▪





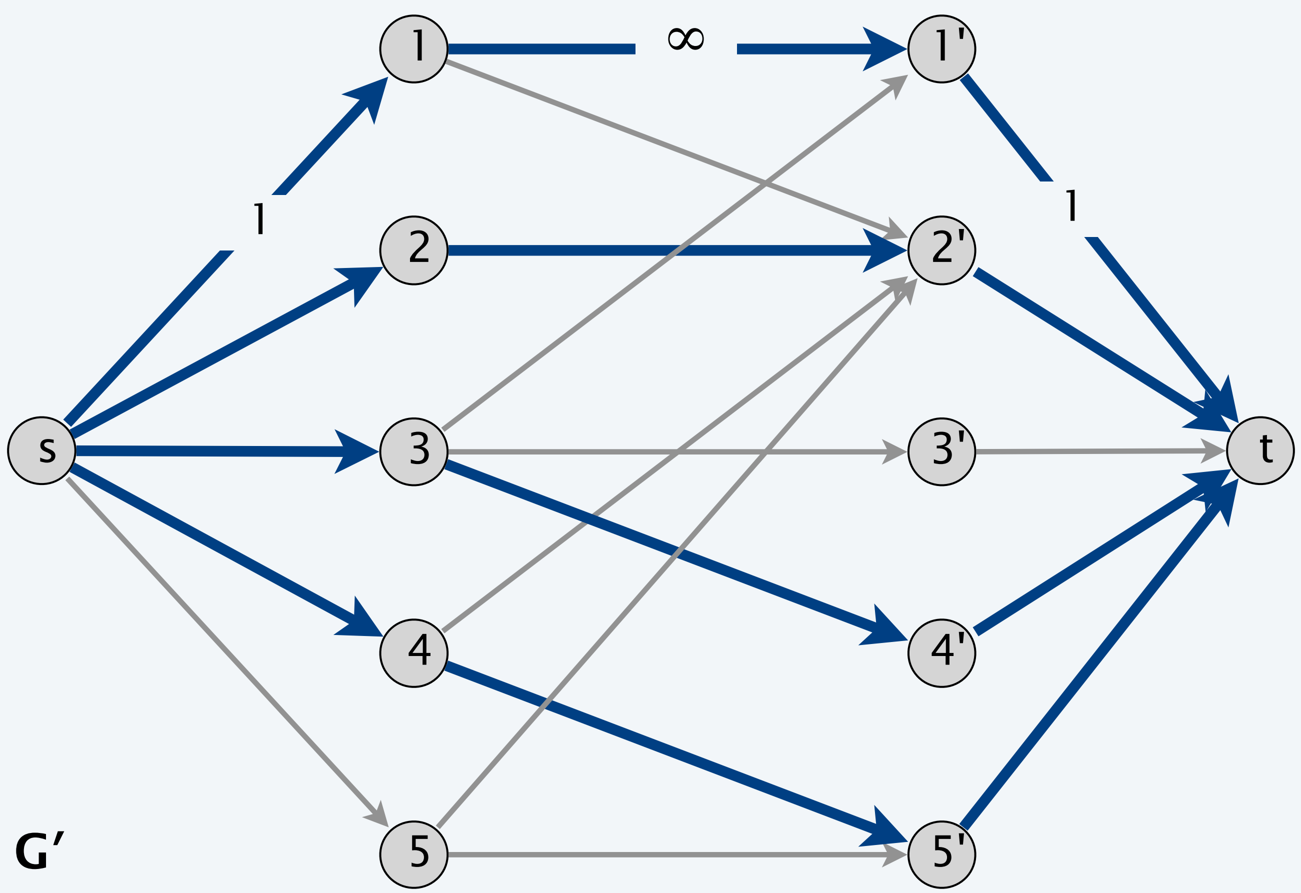
Formulazione come massimo flusso: dimostrazione di correttezza

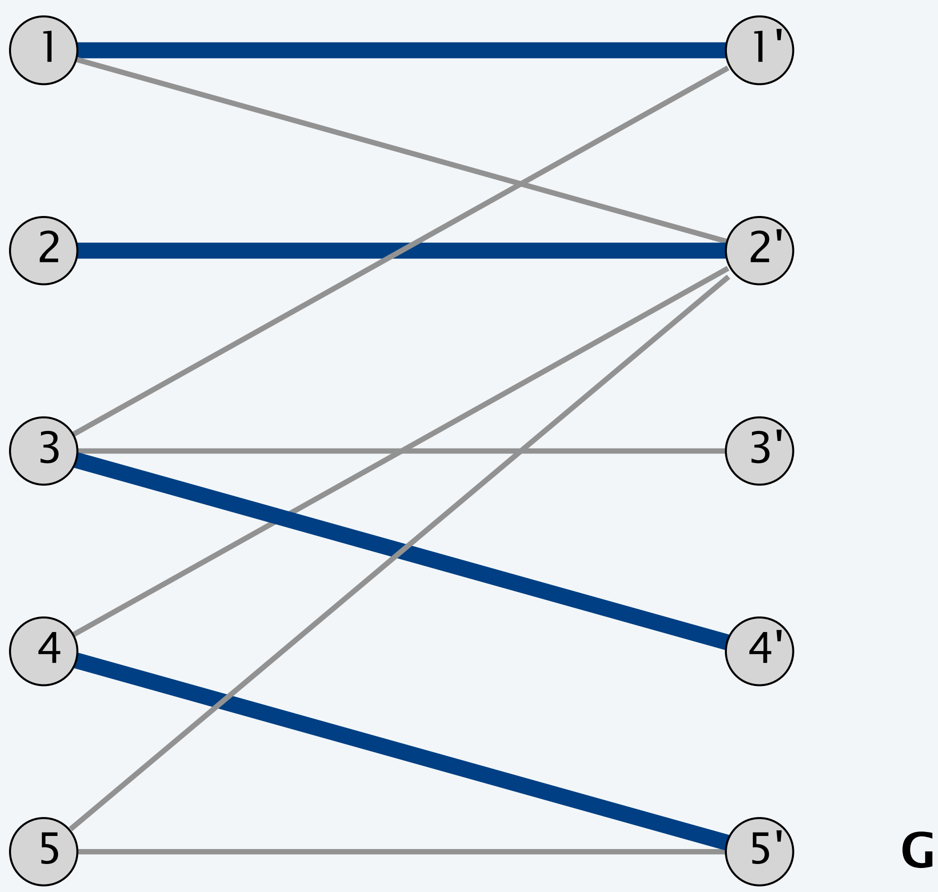
Teorema. Corrispondenza 1–1 tra abbinamenti di cardinalità k in G e flussi interi di valore k in G'.

Corollario. Il problema dell'abbinamento bipartito può essere risolto tramite una formulazione di massimo flusso.

Dim.

- Teorema di interezza $\Rightarrow$ esiste un massimo flusso f^* in G' che è intero.
- Corrispondenza 1–1 $\Rightarrow$ f^* corrisponde ad un abbinamento massimo. ▪





Qual è il tempo di esecuzione di Ford–Fulkerson per determinare l'abbinamento a massima cardinalità in un grafo bipartito con $|L| = |R| = n$?

A. $O(m + n)$

B. $O(mn)$

C. $O(mn^2)$

D. $O(m^2 n)$

Abbinamenti perfetti in grafi bipartiti

Def. Dato un grafo $G = (V, E)$, un sottoinsieme di archi $M \subseteq E$ è un abbinamento perfetto se ogni nodo appare in esattamente un arco di M.

D. Quand'è che un grafo bipartito ammette un abbinamento perfetto?

Struttura dei grafi bipartiti con abbinamenti perfetti.

- Chiaramente, occorre avere $|L| = |R|$.
- Quali altre condizioni sono necessarie?
- Quali altre condizioni sono sufficienti?

Abbinamenti perfetti in grafi bipartiti

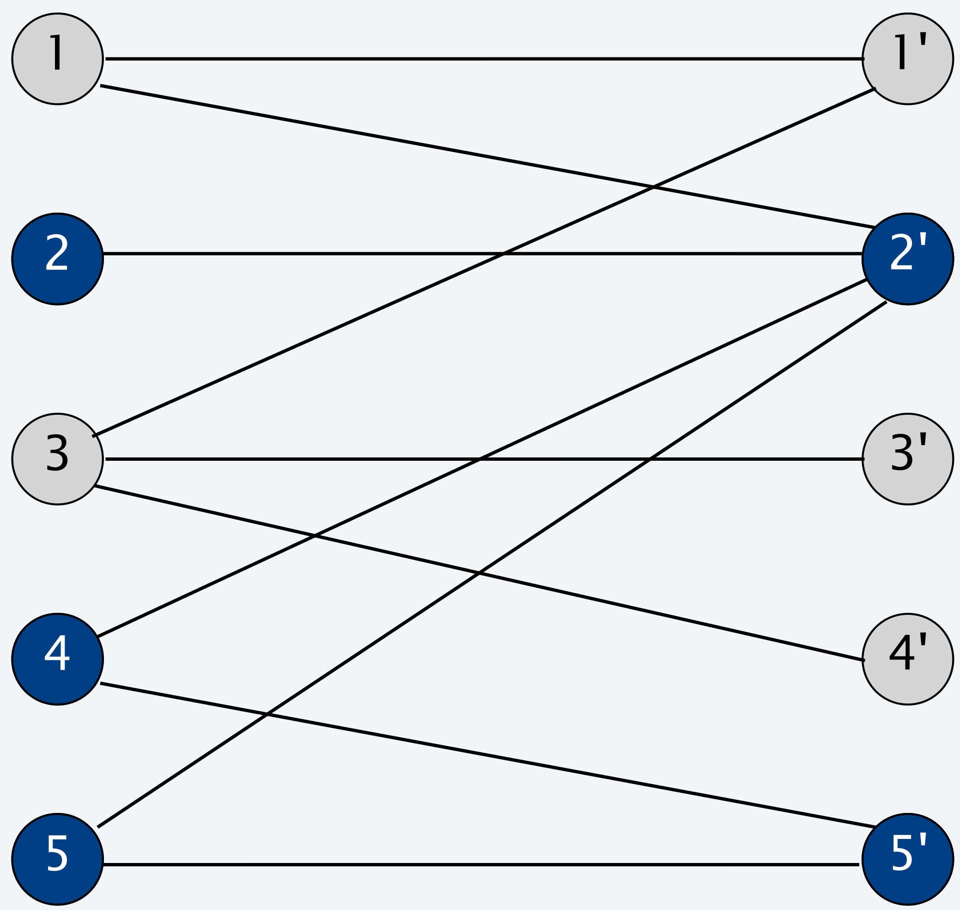
Notazione. Sia S un sottoinsieme di nodi, e sia $N(S)$ l'insieme dei nodi adiacenti ai nodi in S.

Osservazione. Se un grafo bipartito $G = (L \cup R, E)$ ha un abbinamento perfetto, allora $|N(S)| \geq |S|$ per ogni sottoinsieme $S \subseteq L$.

Dim. Ogni nodo in S deve essere abbinato ad un diverso nodo di $N(S)$. ▪

S = { 2, 4, 5 }

N(S) = { 2', 5' }



nessun abbinamento perfetto

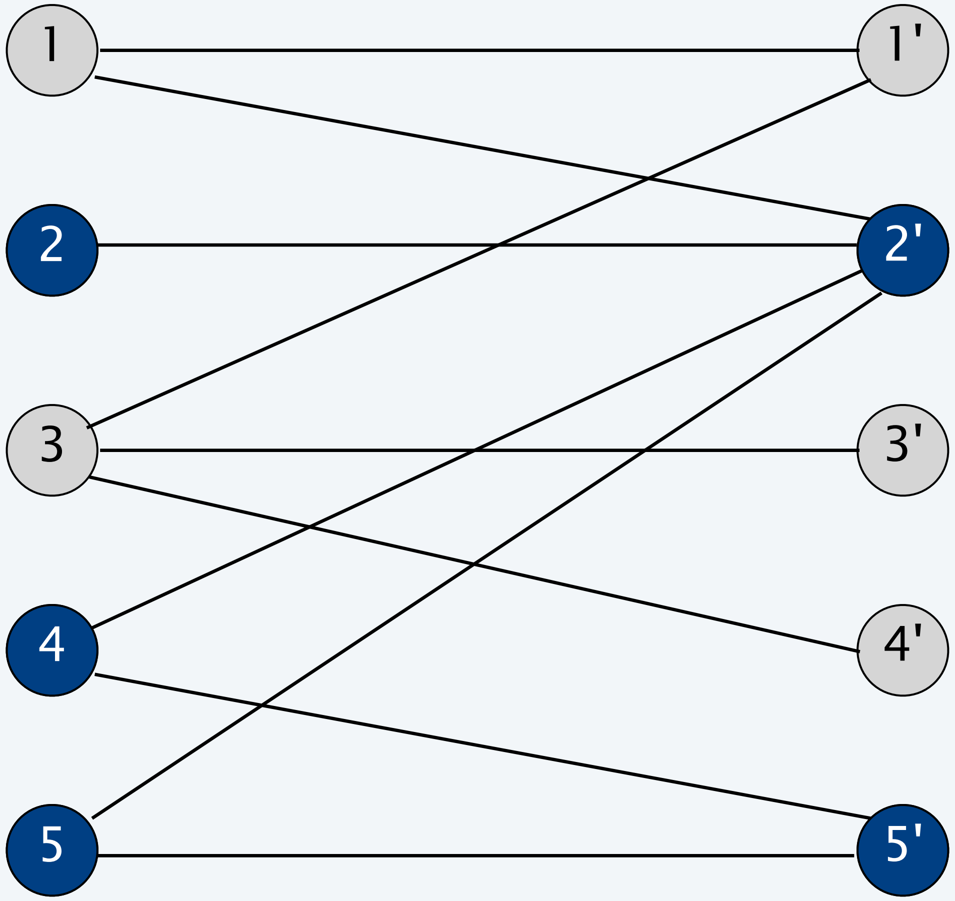
Teorema dei matrimoni di Hall

Teorema. [Frobenius 1917, Hall 1935] Sia $G = (L \cup R, E)$ un grafo bipartito con $|L| = |R|$. Allora, il grafo G ha un abbinamento perfetto sse $|N(S)| \geq |S|$ per ogni sottoinsieme $S \subseteq L$.

Dim. $\Rightarrow$ È l'osservazione precedente.

S = { 2, 4, 5 }

N(S) = { 2', 5' }

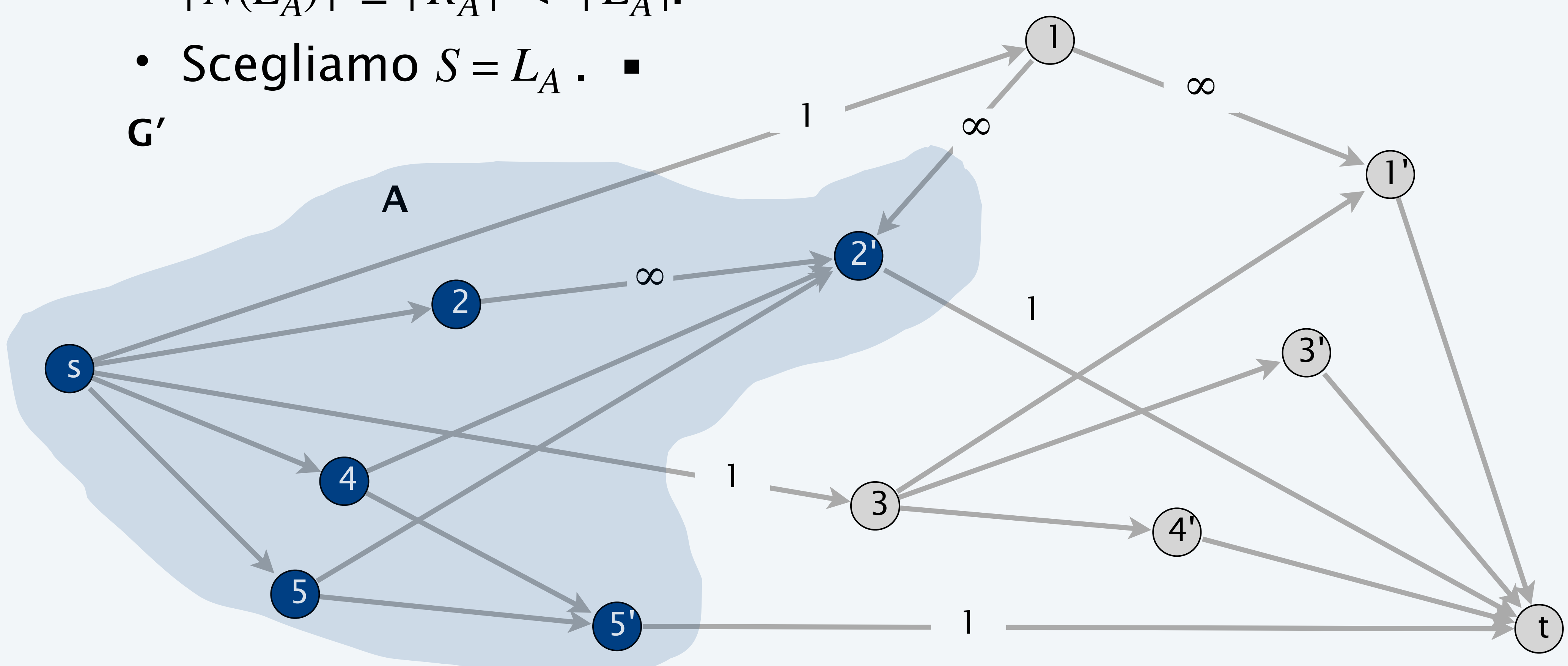


nessun abbinamento perfetto

Teorema dei matrimoni di Hall

Dim. $\Leftarrow$ Supponiamo che G non abbia un abbinamento perfetto.

- Formuliamo il problema come massimo flusso e sia (A, B) un taglio minimo in G'.
- Per il teorema massimo flusso – minimo taglio, $cap(A, B) < |L|$.
- Definiamo $L_A = L \cap A$, $L_B = L \cap B$, $R_A = R \cap A$.
- $cap(A, B) = |L_B| + |R_A| \Rightarrow |R_A| < |L_A|$.
- Il taglio minimo non può usare archi $\infty \Rightarrow N(L_A) \subseteq R_A$.
- $|N(L_A)| \leq |R_A| < |L_A|$.
- Scegliamo $S = L_A$. ▪



$L_A = \{2, 4, 5\}$
$L_B = \{1, 3\}$
$R_A = \{2', 5'\}$
$N(L_A) = \{2', 5'\}$

Abbinamento bipartito

Problema. Dato un grafo bipartito, trovare un abbinamento a massima cardinalità.

anno	caso peggiore	tecnica	scoperta da
1955	$O(m\ n)$	**cammino aumentante**	Ford–Fulkerson
1973	$O(m\ n^{1/2})$	**flusso bloccante**	Hopcroft–Karp, Karzanov
2004	$O(n^{2.378})$	**moltiplicazione di matrici**	Mucha–Sankowsi
2013	$\tilde{O}(m^{10/7})$	**flussi elettrici**	Mądry
20xx	???		

tempi di esecuzione di algoritmi per l'abbinamento bipartito in grafi di n nodi ed m archi

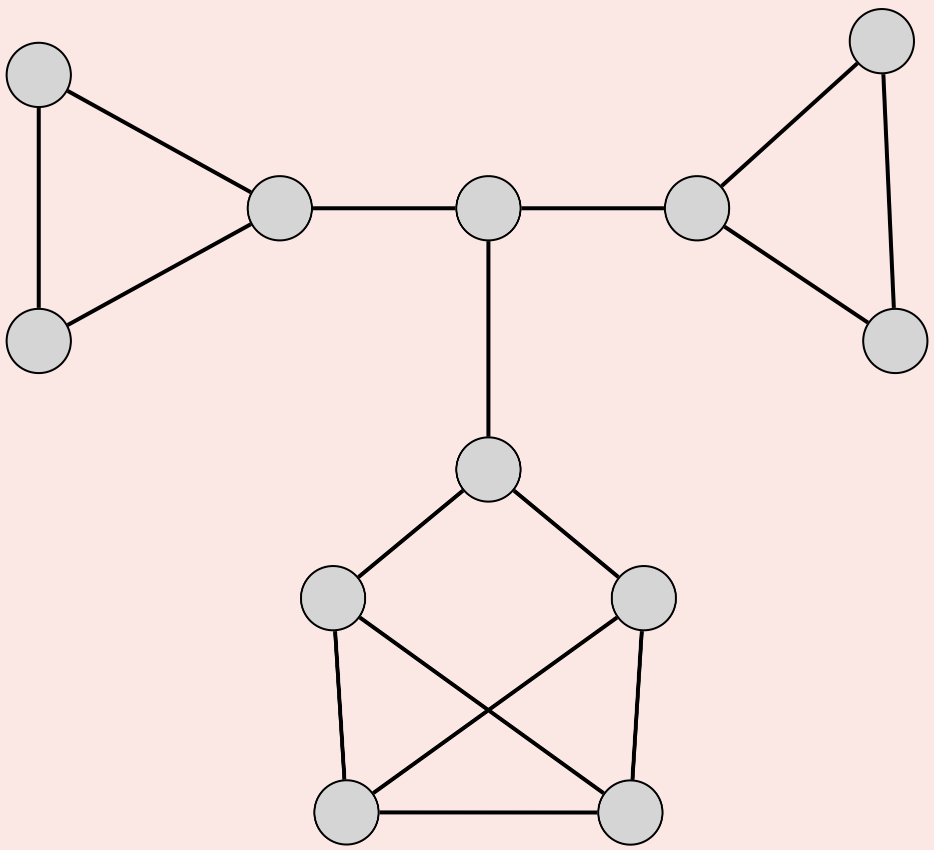
Quali proprietà ha il grafo in figura G = (V, E)?

A. G ha un abbinamento perfetto.

B. La condizione di Hall è soddisfatta: $|N(S)| \geq |S|$ per ogni $S \subseteq V$.

C. Sia A che B.

D. Né A né B.



Abbinamento nonbipartito

Problema. Dato un grafo, trovare un abbinamento a massima cardinalità.

- La struttura dei grafi nonbipartiti è più complicata.
- Ma ben nota. [Tutte–Berge formula, Edmonds–Gallai]
- Algoritmo blossom: $O(n^4)$. [Edmonds 1965]
- Migliore limitazione nota: $O(m\, n^{1/2})$. [Micali–Vazirani 1980, Vazirani 1994]

PATHS, TREES, AND FLOWERS

JACK EDMONDS

1. Introduction. A *graph G* for purposes here is a finite set of elements called *vertices* and a finite set of elements called *edges* such that each edge *meets* exactly two vertices, called the *end-points* of the edge. An edge is said to *join* its end-points.

A *matching* in G is a subset of its edges such that no two meet the same vertex. We describe an efficient algorithm for finding in a given graph a matching of maximum cardinality. This problem was posed and partly solved by C. Berge; see Sections 3.7 and 3.8.

COMBINATORICA
Akadémiai Kiadó – Springer-Verlag

COMBINATORICA **14** (1) (1994) 71–109

A THEORY OF ALTERNATING PATHS AND BLOSSOMS FOR PROVING CORRECTNESS OF THE $O(\sqrt{V}E)$ GENERAL GRAPH MAXIMUM MATCHING ALGORITHM

VIJAY V. VAZIRANI[1]

Received December 30, 1989
Revised June 15, 1993

Importanza storica (Jack Edmonds 1965)

2. Digression. An explanation is due on the use of the words "efficient algorithm." First, what I present is a conceptual description of an algorithm and not a particular formalized algorithm or "code."

For practical purposes computational details are vital. However, my purpose is only to show as attractively as I can that there is an efficient algorithm. According to the dictionary, "efficient" means "adequate in operation or performance." This is roughly the meaning I want—in the sense that it is conceivable for maximum matching to have no efficient algorithm. Perhaps a better word is "good."

I am claiming, as a mathematical result, the existence of a *good* algorithm for finding a maximum cardinality matching in a graph.

There is an obvious finite algorithm, but that algorithm increases in difficulty exponentially with the size of the graph. It is by no means obvious whether *or not* there exists an algorithm whose difficulty increases only algebraically with the size of the graph.

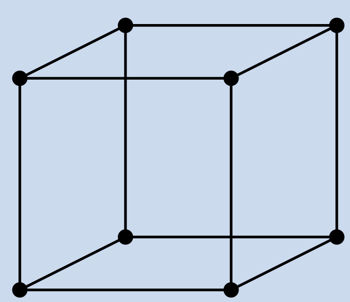
Problema dell'Hackathon

Problema dell'Hackathon.

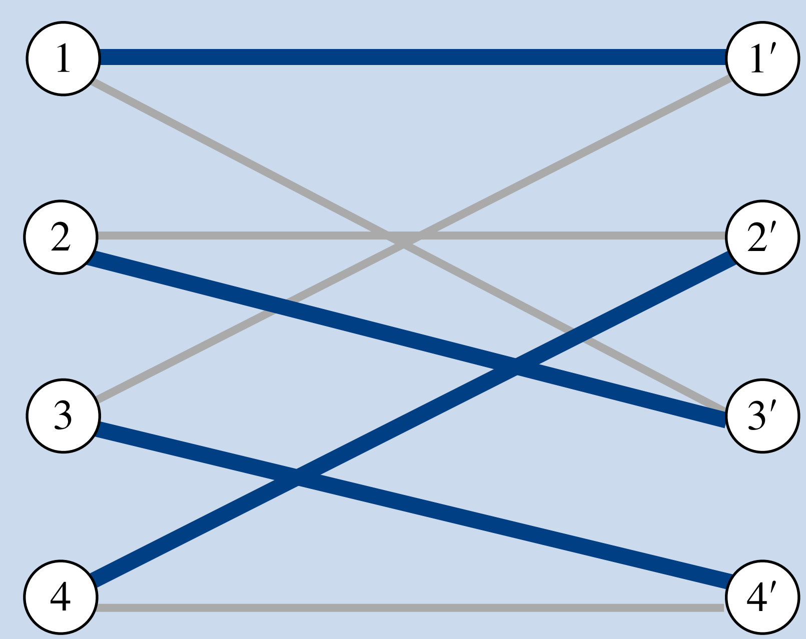
- All'Hackathon partecipano n studenti di Harvard ed n di Princeton
- Ogni studente di Harvard è amico di esattamente $k > 0$ studenti di Princeton; ogni studente di Princeton è amico di esattamente k studenti di Harvard.
- È possibile organizzare l'hackathon in modo che ogni studente di Princeton programmi in coppia con un diverso amico di Harvard?

Riformulazione matematica. Ogni grafo bipartito k-regolare ammette un abbinamento perfetto?

Es. Ipercubo booleano.



grafo bipartito 2-regolare



PROBLEMA DELL'HACKATHON

Teorema. Ogni grafo bipartito k-regolare G ha un abbinamento perfetto.

Pf 1.


Kőnig


Frobenius

- Deve essere $|L| = |R|$, dove $G = (L \cup R, E)$
- Consideriamo qualunque $S \subseteq L$.
- Numero di archi tra S ed $N(S) = k\ |S|$.
- Numero di archi tra L ed $N(S) = k\ |N(S)|$.
- Poiché $S \subseteq L$, abbiamo $k\ |S| \leq k\ |N(S)|$.
- O equivalentemente, $|S| \leq |N(S)|$.
- Il risultato segue dal teorema dei matrimoni di Hall. ▪

Problema dell'Hackathon

Teorema. Ogni grafo bipartito k-regolare G ha un abbinamento perfetto.

Pf 2.

König

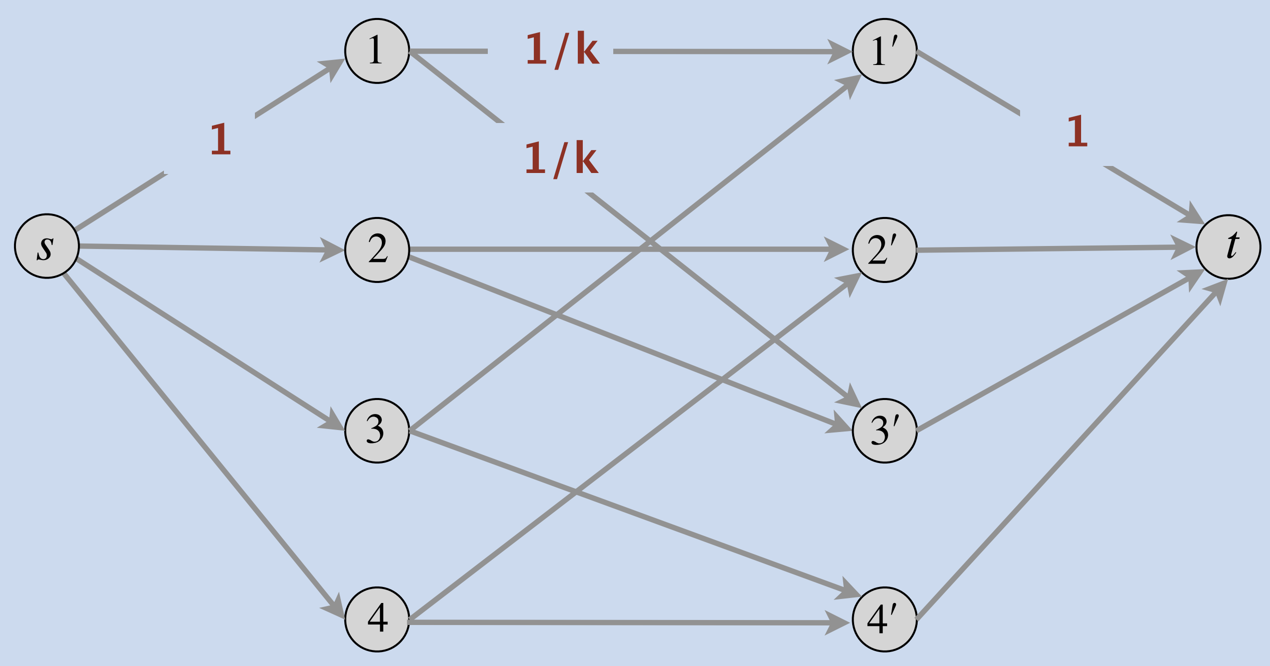
Frobenius

- Taglia di un abbinamento massimo = valore del massimo flusso in G'.
- Consideriamo il flusso

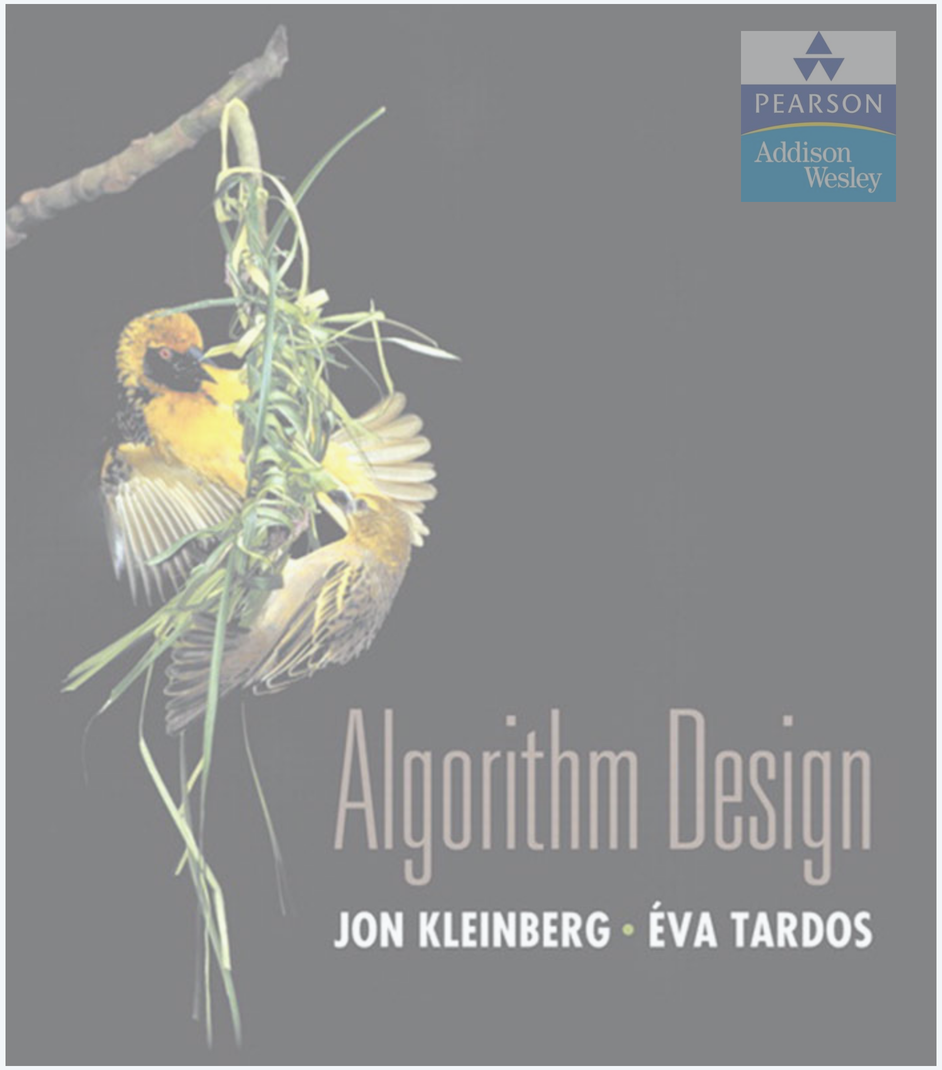
$$f(u,v) = \begin{cases} 1 & \text{if } u = s \text{ or } v = t \\ 1/k & \text{otherwise} \end{cases}$$

- Il valore del flusso f è $n \Rightarrow G'$ ha un abbinamento perfetto. ▪

G′



un flusso f di valore n



SECTION 7.6

7. FLUSSI DI RETE II

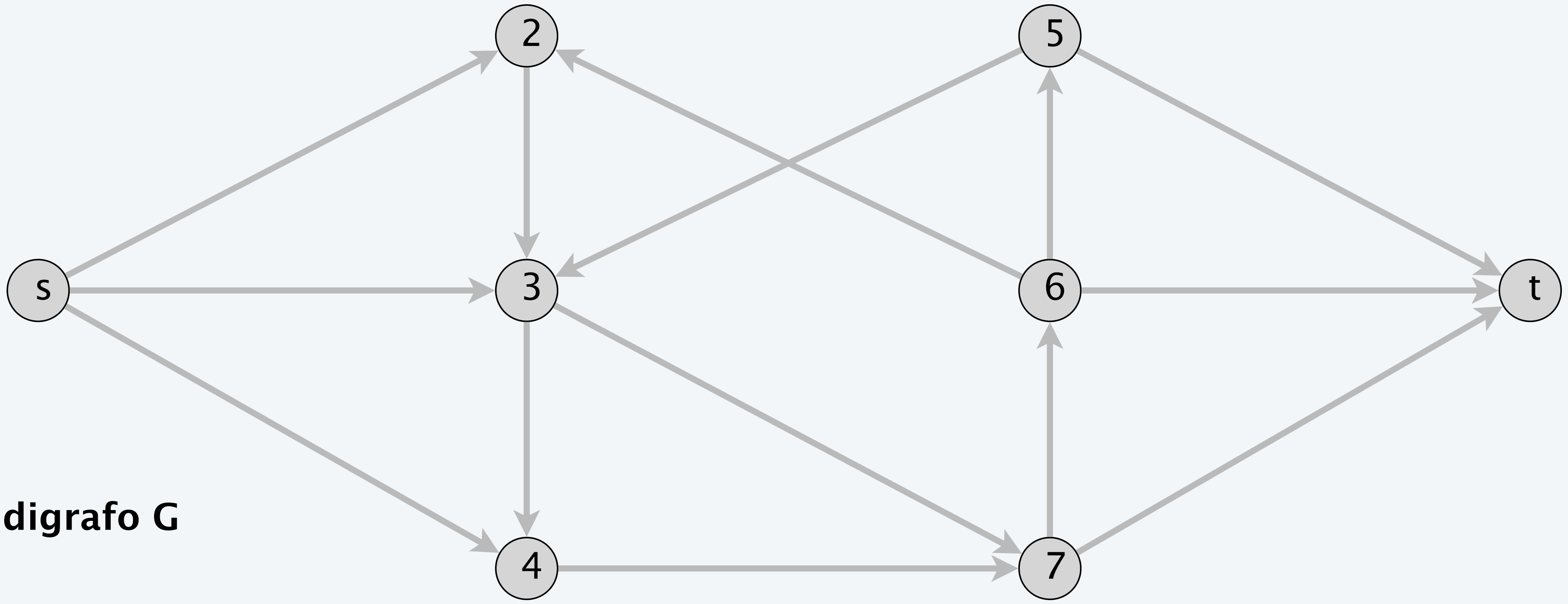
- *bipartite matching*
- *cammini disgiunti*
- *extensions to max flow*
- *survey design*
- *airline scheduling*
- *image segmentation*
- *project selection*
- *baseball elimination*

Cammini arco-disgiunti

Def. Due cammini sono arco-disgiunti se non hanno archi in comune.

Problema dei cammini arco-disgiunti. Dato un digrafo $G = (V, E)$ e due nodi s e t, trovare il massimo numero di cammini $s \rightsquigarrow t$ arco-disgiunti.

Es. Reti di comunicazione.



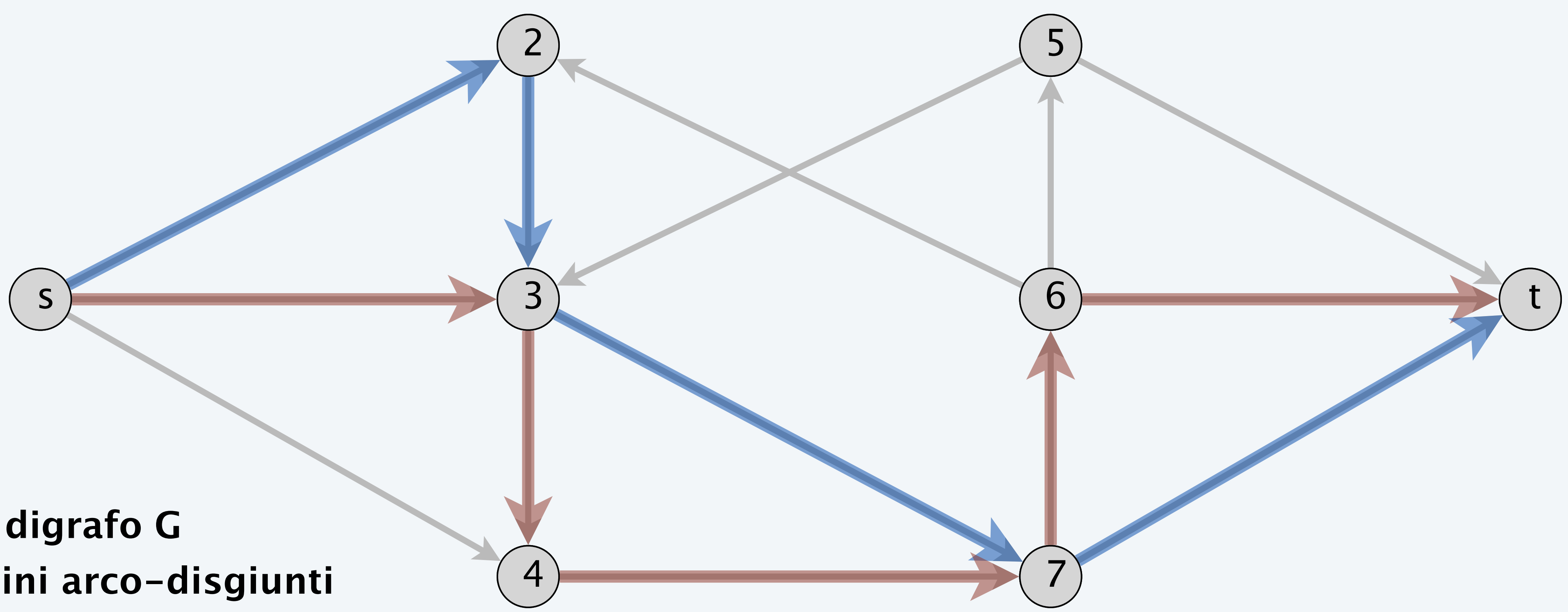
digrafo G

Cammini arco-disgiunti

Def. Due cammini sono arco-disgiunti se non hanno archi in comune.

Problema dei cammini arco-disgiunti. Dato un digrafo $G = (V, E)$ e due nodi s e t, trovare il massimo numero di cammini $s \rightsquigarrow t$ arco-disgiunti.

Es. Reti di comunicazione.



digrafo G
2 cammini arco-disgiunti

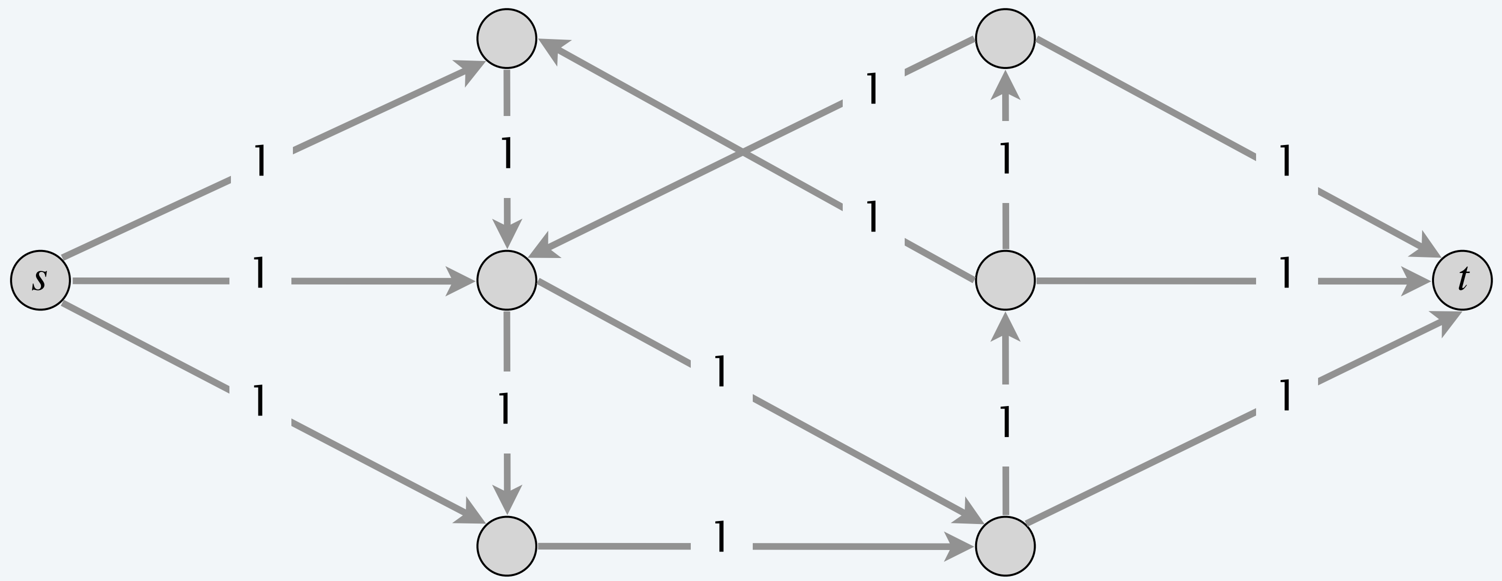
Cammini arco-disgiunti

Formulazione massimo flusso. Assegna capacità unitaria ad ogni arco.

Teorema. C'è corrispondenza 1–1 tra k cammini $s \rightsquigarrow t$ arco-disgiunti in G e flussi interi di valore k in G'.

Dim. $\Rightarrow$

- Siano $P_1, \ldots, P_k$ i k cammini $s \rightsquigarrow t$ arco-disgiunti in G.
- Sia $f(e) = \begin{cases} 1 & \text{edge } e \text{ participates in some path } P_j \\ 0 & \text{otherwise} \end{cases}$
- Poiché i cammini sono arco-disgiunti, f è un flusso di valore k. ▪



Cammini arco-disgiunti

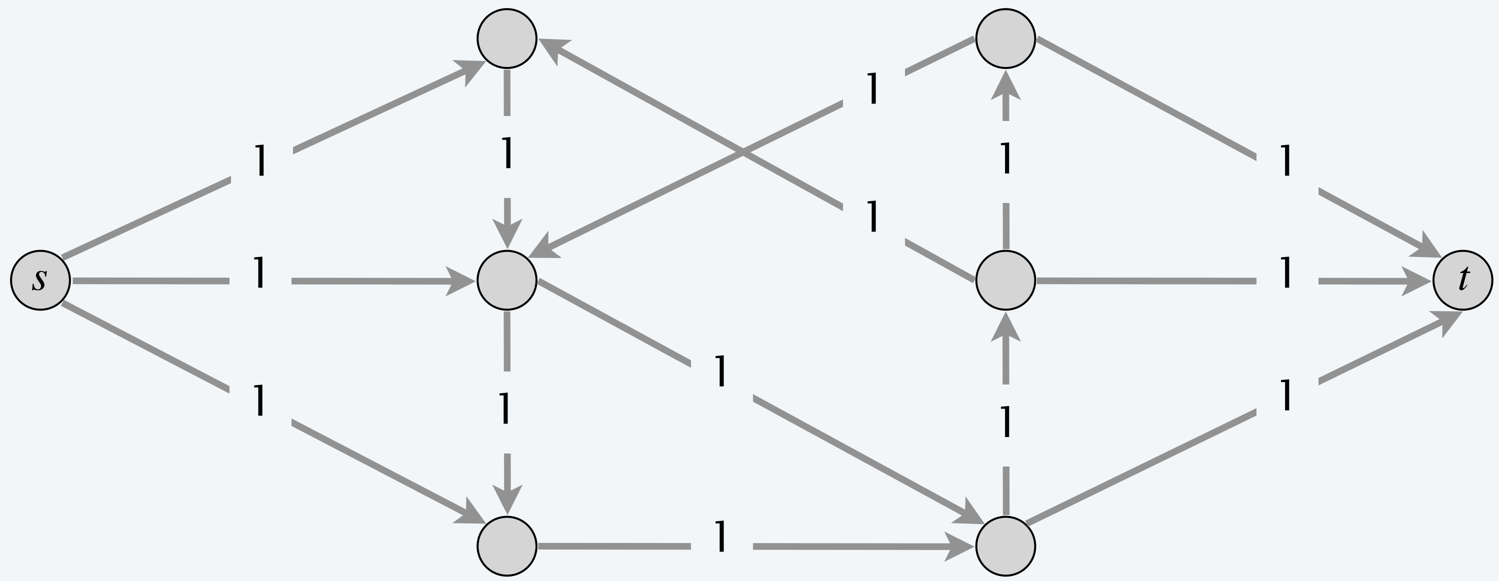
Formulazione massimo flusso. Assegna capacità unitaria ad ogni arco.

Teorema. C'è corrispondenza 1–1 tra k cammini $s \rightsquigarrow t$ arco-disgiunti in G e flussi interi di valore k in G'.

Dim. $\Leftarrow$

- Sia f un flusso intero in G' di valore k.
- Consideriamo un arco (s, u) con $f(s, u) = 1$.
  - per conservazione del flusso, esiste un arco (u, v) con $f(u, v) = 1$
  - continuiamo fino a raggiungere t, scegliendo sempre nuovi archi
- Ciò produce k cammini arco-disgiunti (non necessariamente semplici). ▪

si possono eliminare i cicli in tempo $O(mn)$ per ottenere cammini semplici se lo si desidera (decomposizione di flusso)



Cammini arco-disgiunti

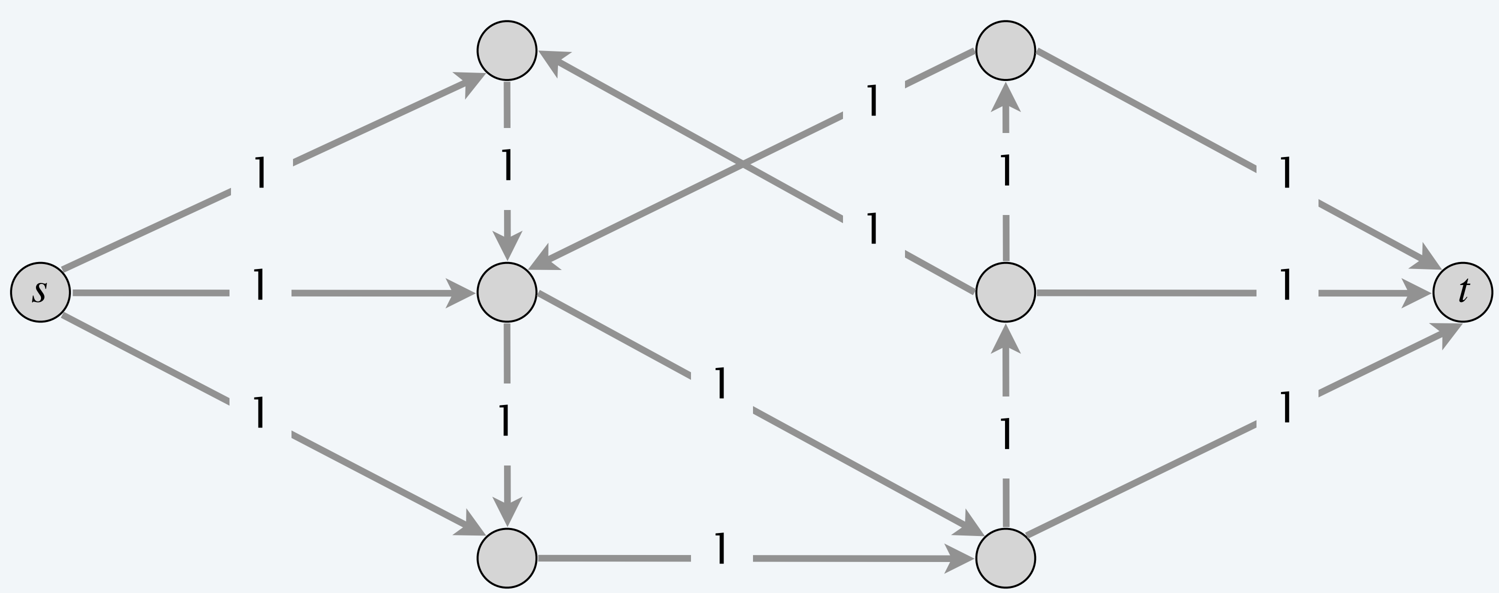
Formulazione massimo flusso. Assegna capacità unitaria ad ogni arco.

Teorema. C'è corrispondenza 1–1 tra k cammini $s \rightsquigarrow t$ arco-disgiunti in G e flussi interi di valore k in G'.

Corollario. Il problema dei cammini arco-disgiunti può essere risolto con una formulazione di massimo flusso.

Dim.

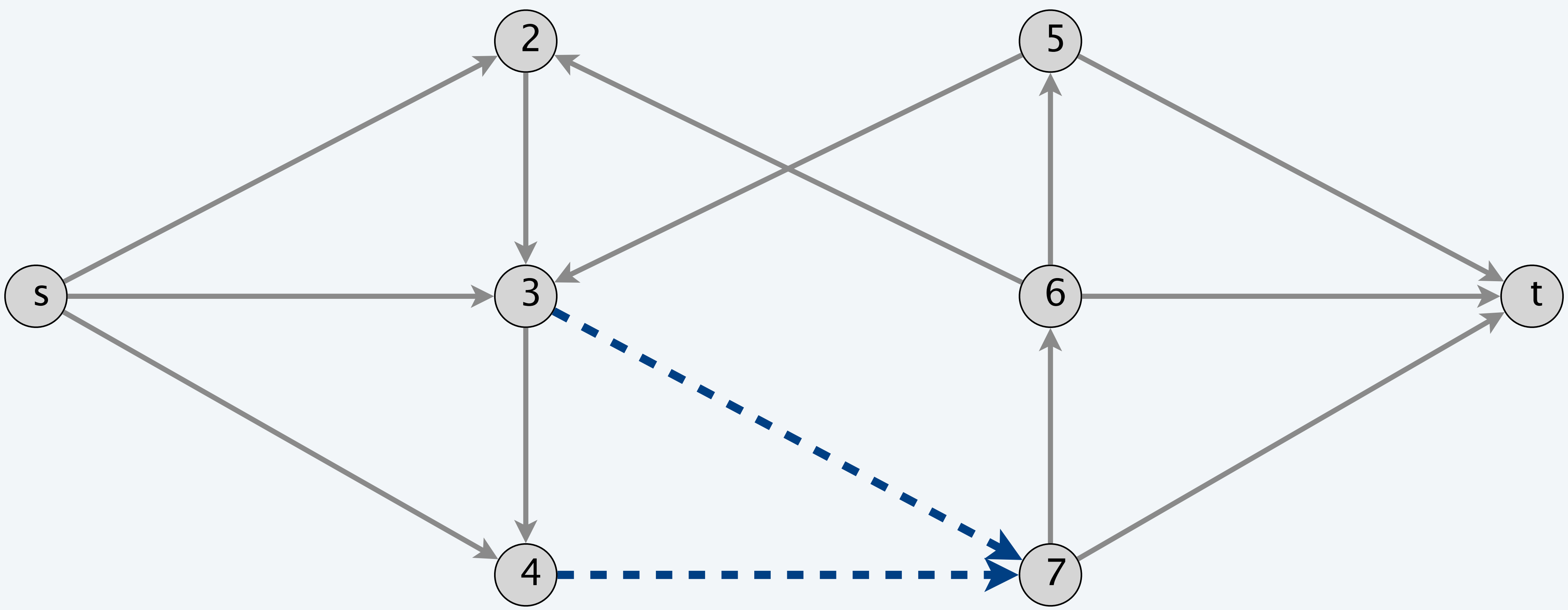
- Teorema di interezza $\Rightarrow$ esiste un flusso massimo f^* in G' che è intero.
- Corrispondenza 1–1 $\Rightarrow$ f^* corrisponde al massimo numero di cammini $s \rightsquigarrow t$ arco-disgiunti in G. ▪



Connettività di rete

Def. Un insieme di archi $F \subseteq E$ sconnette t da s se ogni cammino $s \rightsquigarrow t$ usa almeno un arco di F.

Connettività di rete. Dato un digrafo $G = (V, E)$ e due nodi s e t, trovare il minimo numero di archi la cui rimozione sconnette t da s.

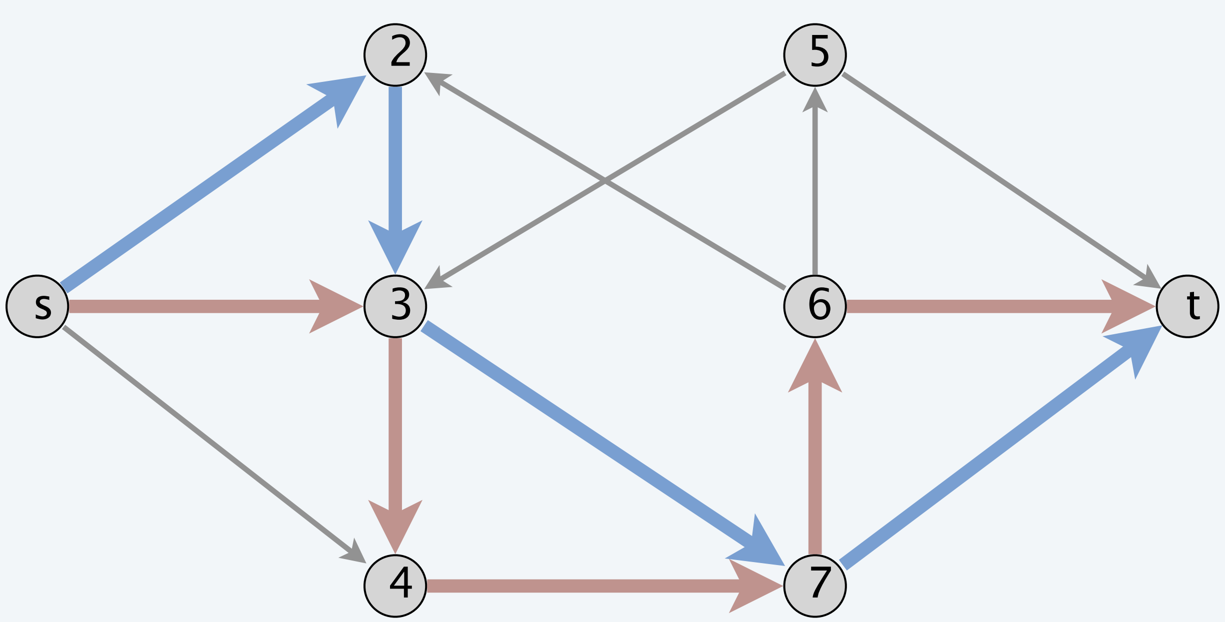


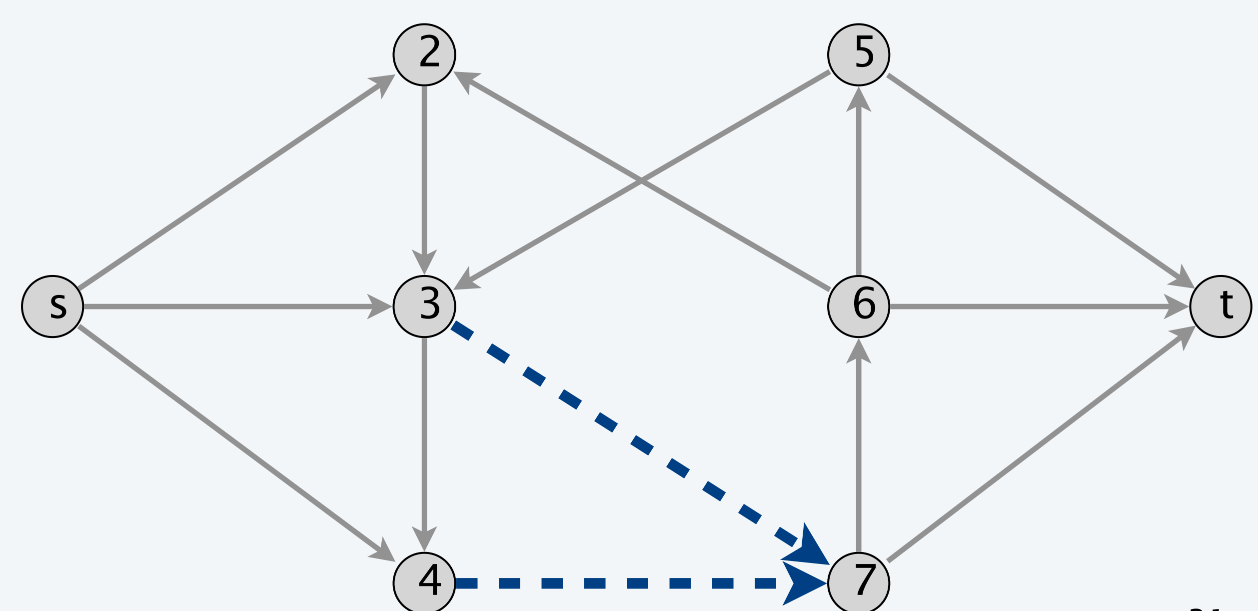
Teorema di Menger

Teorema. [Menger 1927] Il massimo numero di cammini $s \rightsquigarrow t$ arco-disgiunti uguaglia il minimo numero di archi la cui rimozione sconnette t da s.

Dim. $\leq$

- Supponiamo la rimozione di $F \subseteq E$ sconnetta t da s, e $|F| = k$.
- Ogni cammino $s \rightsquigarrow t$ usa almeno un arco di F.
- Quindi, il numero di cammini arco-disgiunti è $\leq k$. ▪



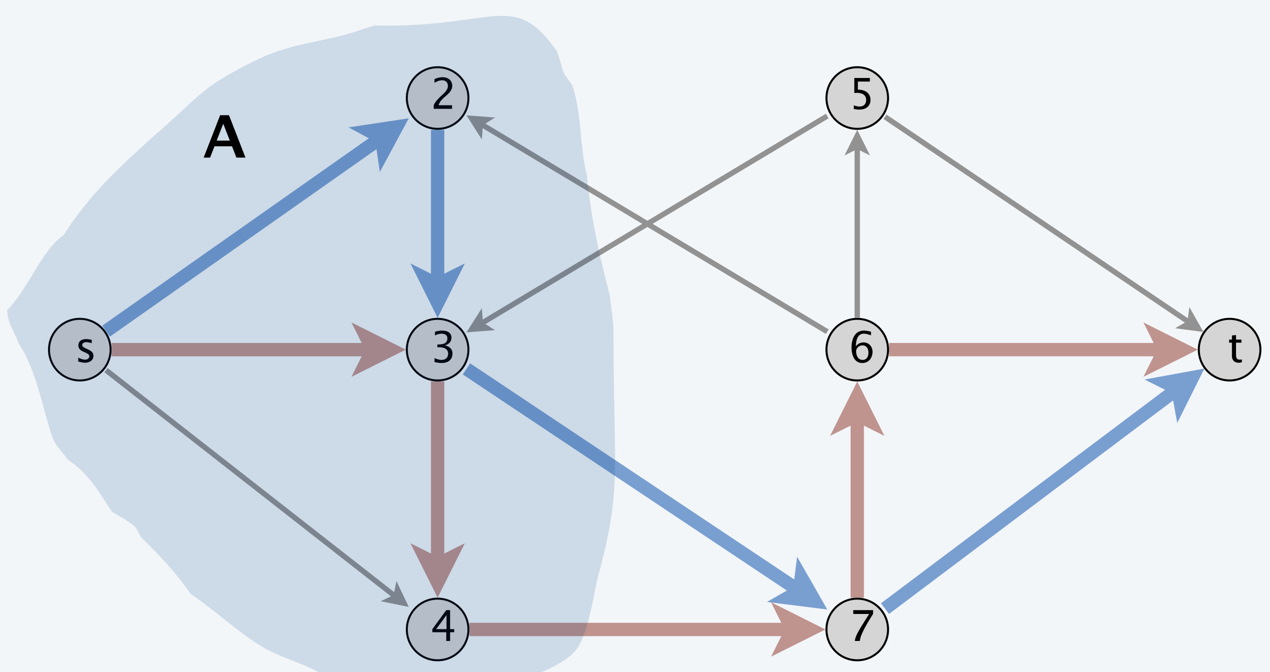


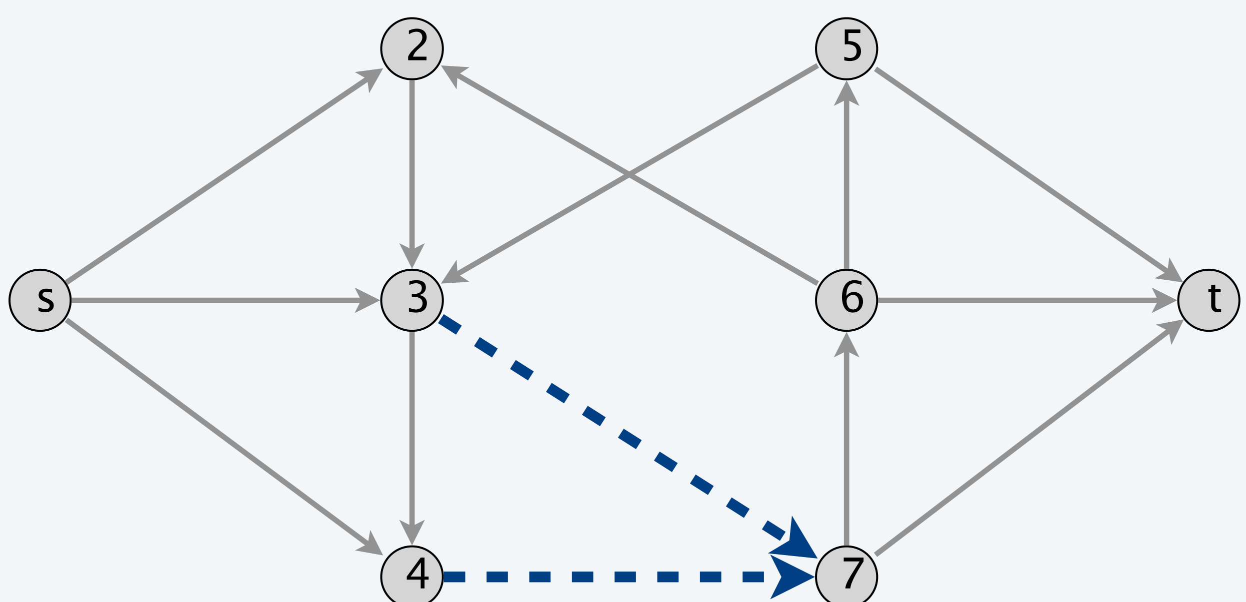
Teorema di Menger

Teorema. [Menger 1927] Il massimo numero di cammini $s \leadsto t$ arco-disgiunti uguaglia il minimo numero di archi la cui rimozione sconnette t da s.

Dim. $\geq$

- Supponiamo il massimo numero di cammini $s \leadsto t$ arco-disgiunti sia k.
- Allora valore del massimo flusso = k.
- Teorema massimo flusso minimo taglio $\Rightarrow$ esiste un taglio (A, B) di capacità k.
- Sia F l'insieme degli archi da A a B.
- $|F| = k$ e l'insieme sconnette t da s. ▪





Come si trova il massimo numero di cammini arco-disgiunti in un grafo non orientato?

A. Risolvi il problema dei cammini arco-disgiunti in un digrafo (rimpiazzando ogni arco non orientato con due archi contrapposti).

B. Risolvi un problema di massimo flusso in un grafo non orientato.

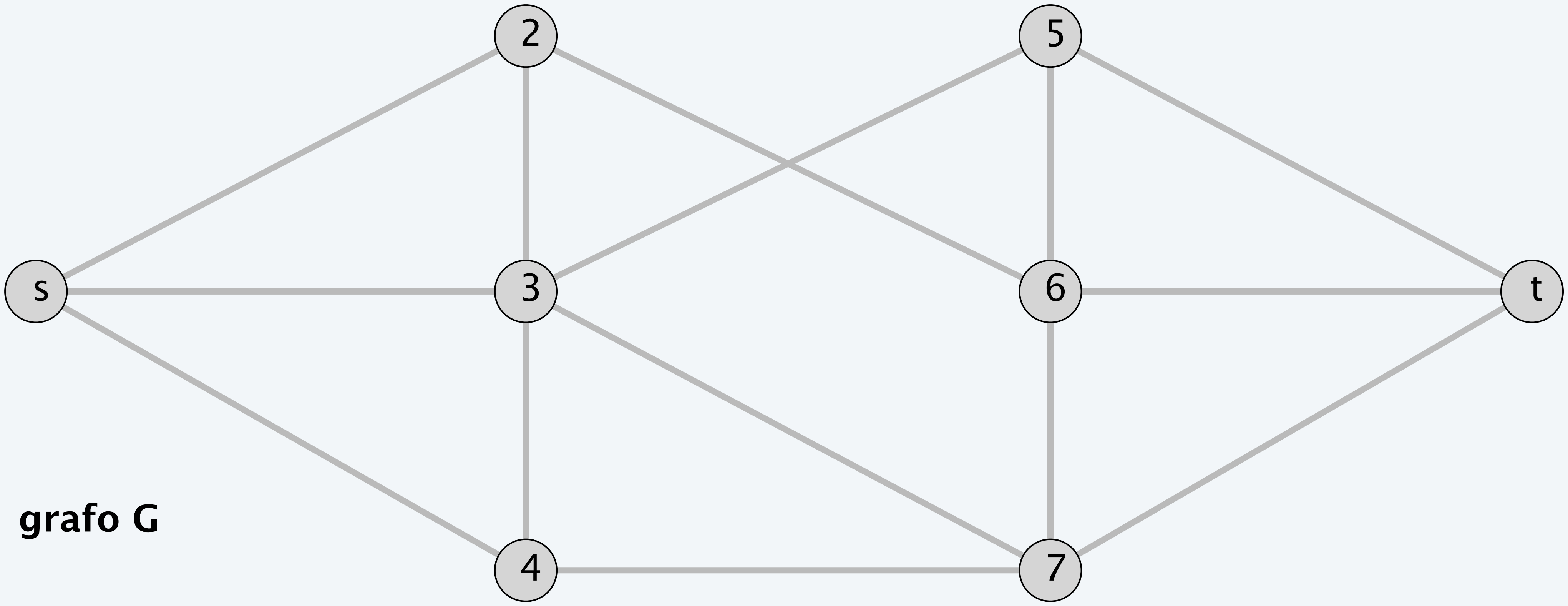
C. Sia A che B.

D. Né A né B.

Cammini arco-disgiunti in grafi non orientati

Def. Due cammini sono arco-disgiunti se non hanno archi in comune.

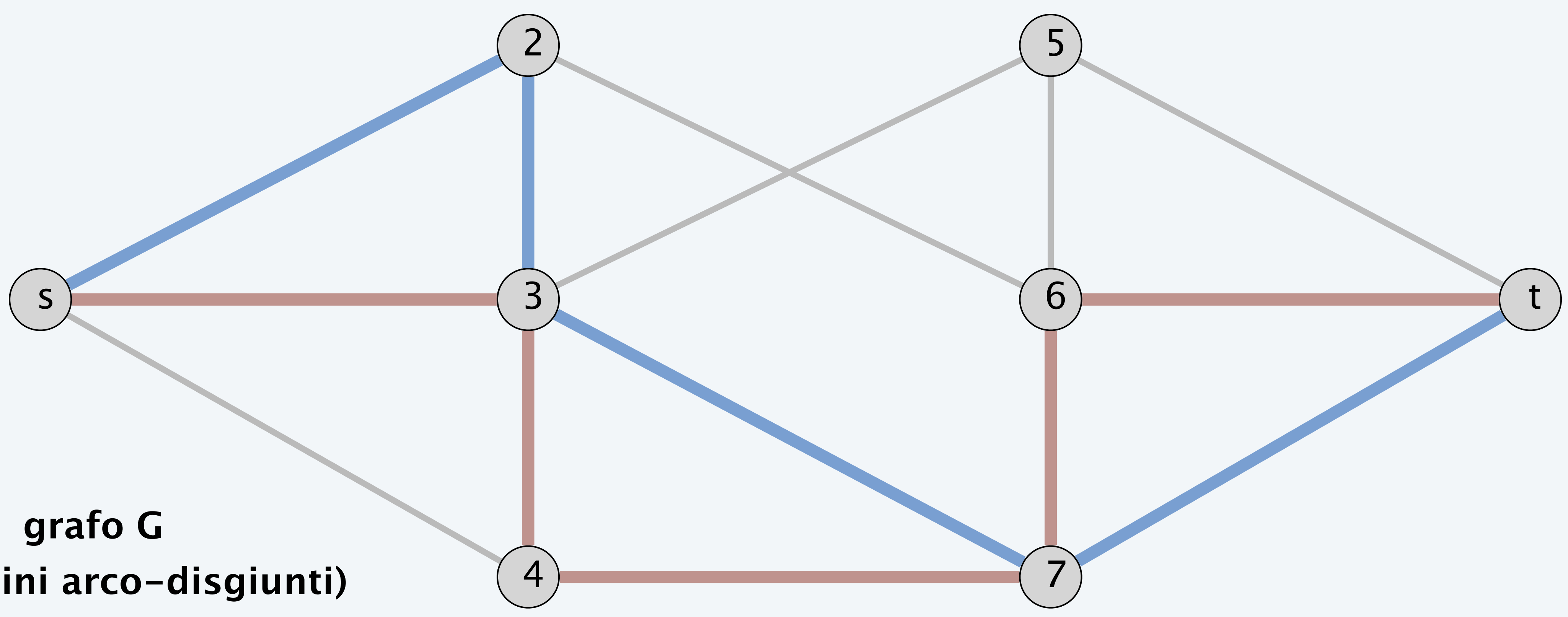
Problema dei cammini arco-disgiunti in grafi non orientati. Dato un grafo $G = (V, E)$ e due nodi s e t, trovare il massimo numero di cammini s–t disgiunti.



Cammini arco-disgiunti in grafi non orientati

Def. Due cammini sono arco-disgiunti se non hanno archi in comune.

Problema dei cammini arco-disgiunti in grafi non orientati. Dato un grafo $G = (V, E)$ e due nodi s e t, trovare il massimo numero di cammini s–t disgiunti.

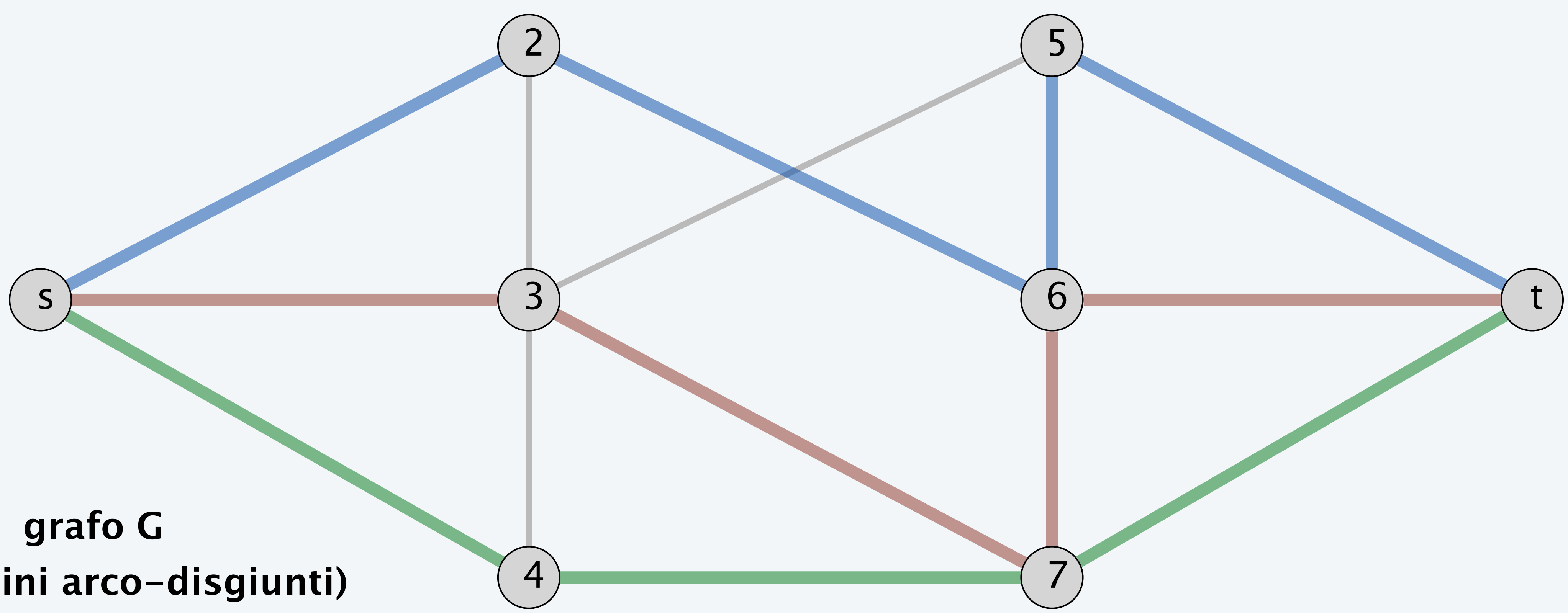


grafo G
(2 cammini arco-disgiunti)

Cammini arco-disgiunti in grafi non orientati

Def. Due cammini sono arco-disgiunti se non hanno archi in comune.

Problema dei cammini arco-disgiunti in grafi non orientati. Dato un grafo $G = (V, E)$ e due nodi s e t, trovare il massimo numero di cammini s–t disgiunti.



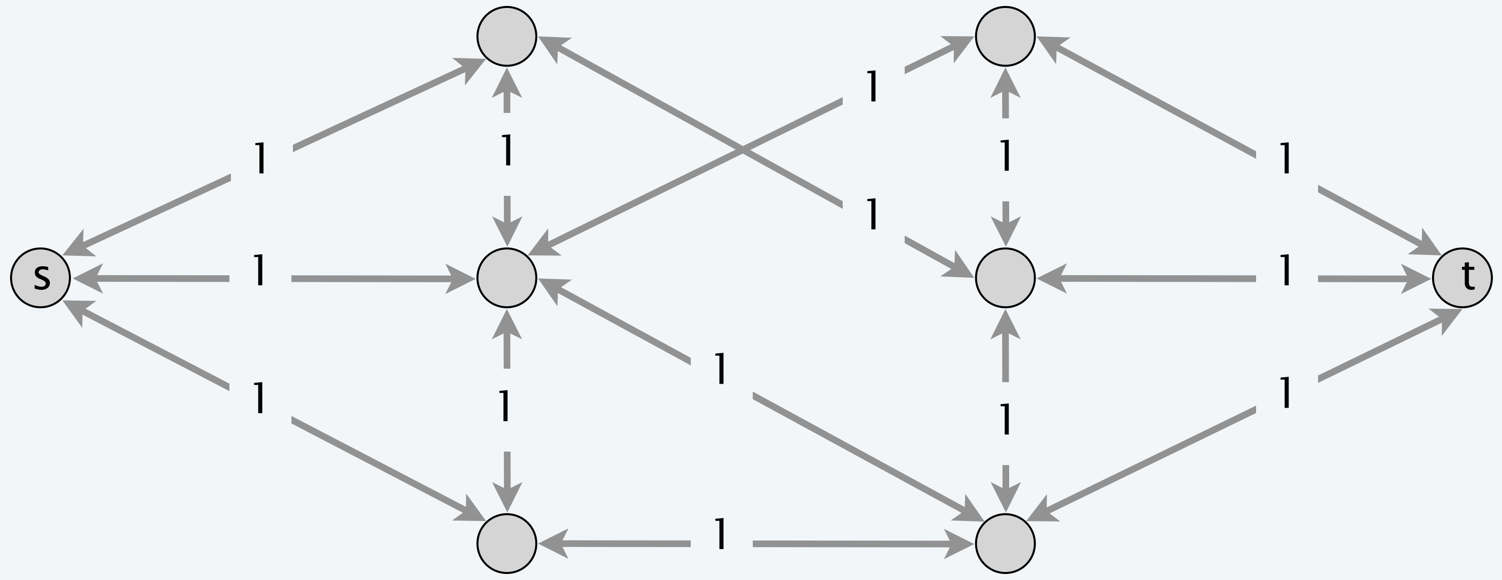
grafo G
(3 cammini arco-disgiunti)

Cammini arco-disgiunti in grafi non orientati

Formulazione di massimo flusso. Rimpiazza ogni arco con due archi contrapposti e assegna capacità unitaria ad ogni arco.

Osservazione. Due cammini P_1 e P_2 possono essere arco-disgiunti nel digrafo ma non arco-disgiunti nel grafo non orientato.

se P_1 usa un arco (u, v)
e P_2 usa il suo arco contrapposto (v, u)



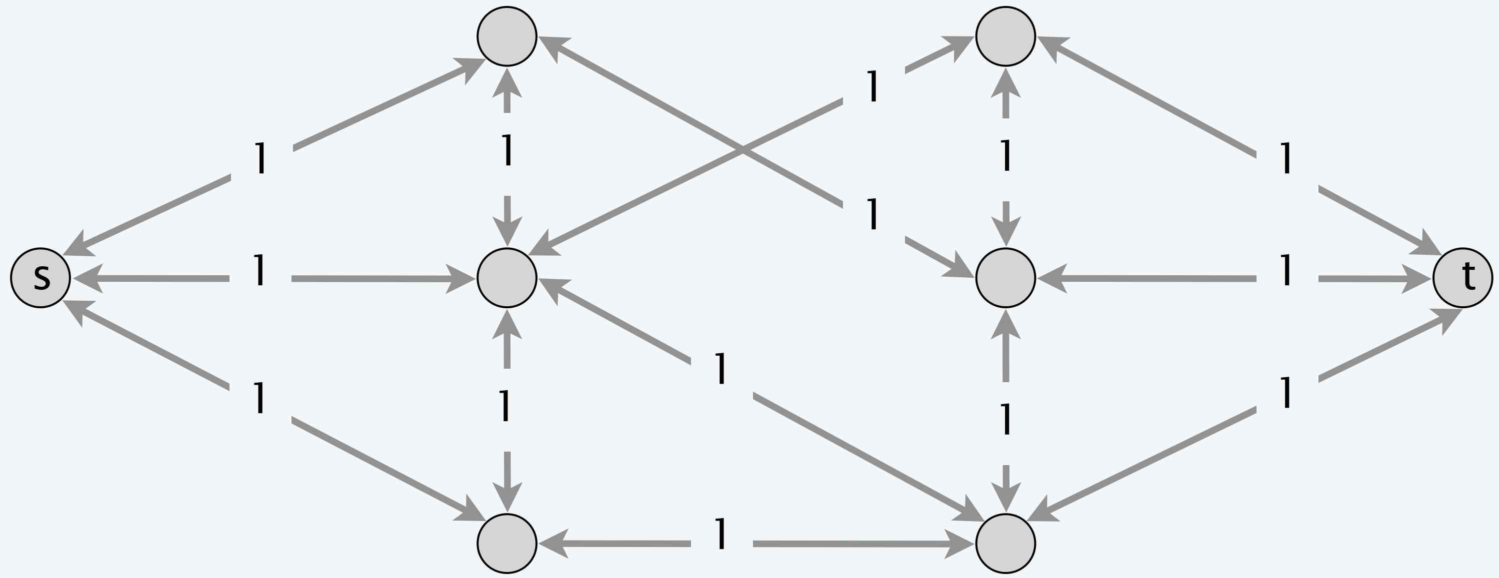
Cammini arco-disgiunti in grafi non orientati

Formulazione massimo flusso. Rimpiazza ogni arco con due archi contrapposti ed assegna capacità unitaria ad ogni arco.

Lemma. In ogni rete di flusso, esiste un flusso massimo f in cui per ogni coppia di archi contrapposti e e e' : o $f(e) = 0$ o $f(e') = 0$ o entrambe. Inoltre, il teorema di interezza è valido anche con tale restrizione.

Dim. [per induzione sul numero di coppie di archi contrapposti]

- Supponiamo $f(e) > 0$ e $f(e') > 0$ per una coppia contrapposta e ed e'.
- Aggiorniamo $f(e) = f(e) - \delta$ e $f(e') = f(e') - \delta$, dove $\delta = \min \{ f(e), f(e') \}$.
- f è ancora un flusso dello stesso valore ma ha una tale coppia contrapposta in meno. ▪

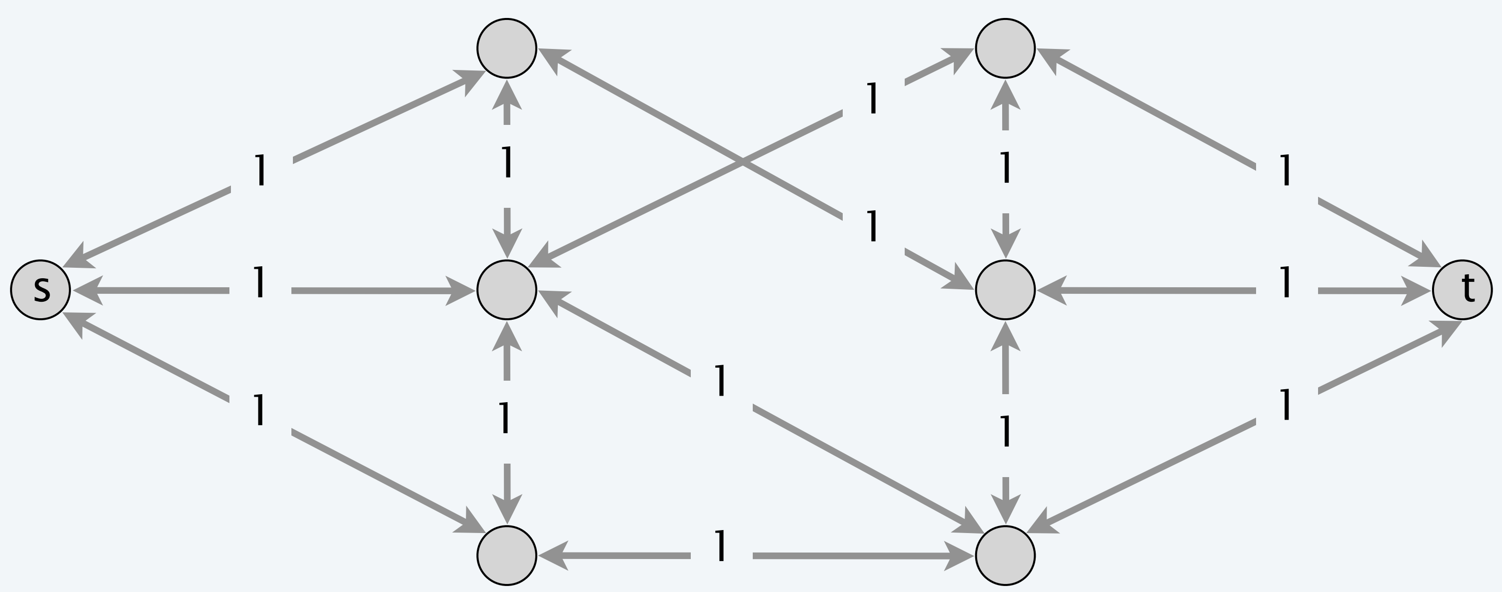


Cammini arco-disgiunti in grafi non orientati

Formulazione massimo flusso. Rimpiazza ogni arco con due archi contrapposti ed assegna capacità unitaria ad ogni arco.

Lemma. In ogni rete di flusso, esiste un flusso massimo f in cui per ogni coppia di archi contrapposti e e e' : o $f(e) = 0$ o $f(e') = 0$ o entrambe. Inoltre, il teorema di interezza è valido anche con tale restrizione.

Teorema. Numero massimo di cammini $s \rightsquigarrow t$ arco-disgiunti = valore del massimo flusso.
Dim. Simile alla dimostrazione per i digrafi; usare il lemma.

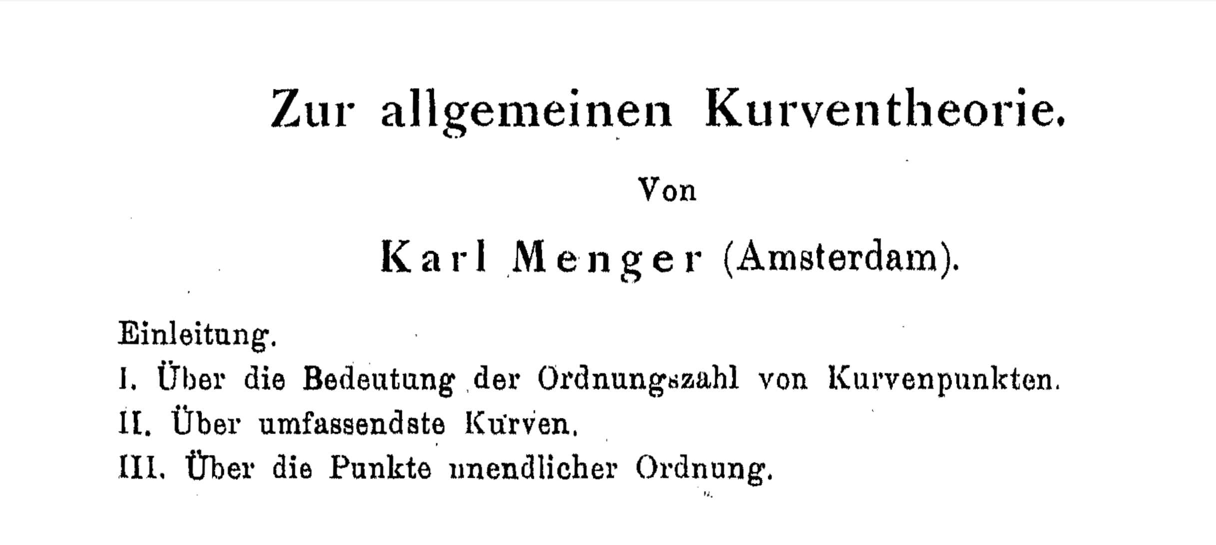


Altri teoremi di Menger

Teorema. Dato un grafo non orientato e due nodi s e t, il numero massimo di cammini s–t arco-disgiunti uguaglia il numero minimo di archi la cui rimozione sconnette s e t.

Teorema. Dato un grafo non orientato e due nodi non adiacenti s e t, il numero massimo di cammini s–t internamente nodo–disgiunti uguaglia il numero minimo di nodi interni la cui rimozione sconnette s e t.

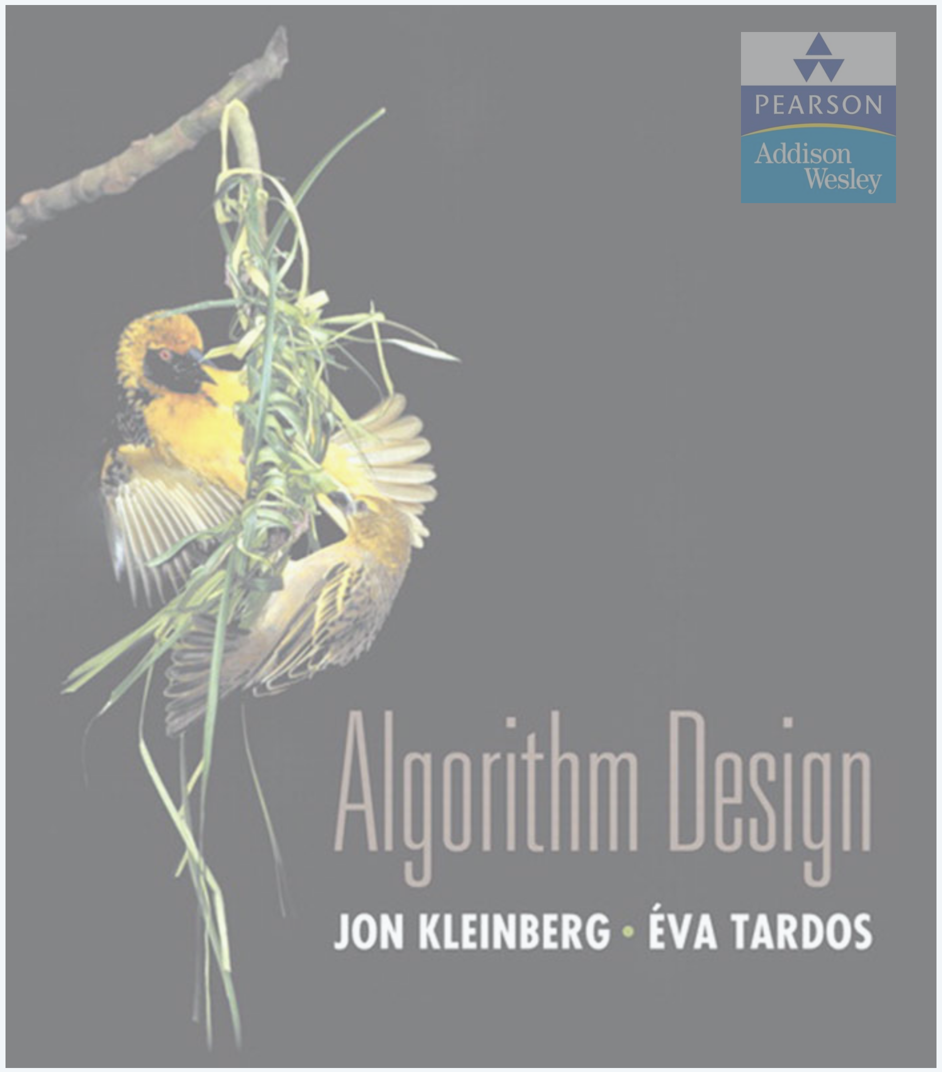
Teorema. Dato un digrafo con due nodi non adiacenti s e t, il numero massimo di cammini $s \to t$ internamente nodo–disgiunti uguaglia il numero minimo di nodi interni la cui rimozione sconnette t da s.


Zur allgemeinen Kurventheorie.

Von

Karl Menger (Amsterdam).

Einleitung.
I. Über die Bedeutung der Ordnungszahl von Kurvenpunkten.
II. Über umfassendste Kurven.
III. Über die Punkte unendlicher Ordnung.



SECTION 7.7

7. FLUSSI DI RETE II

- *bipartite matching*
- *disjoint paths*
- *estensioni al massimo flusso*
- *survey design*
- *airline scheduling*
- *image segmentation*
- *project selection*
- *baseball elimination*

Quali estensioni al massimo flusso si possono modellare facilmente?

A. Più sorgenti e più pozzi.

B. Grafi non orientati.

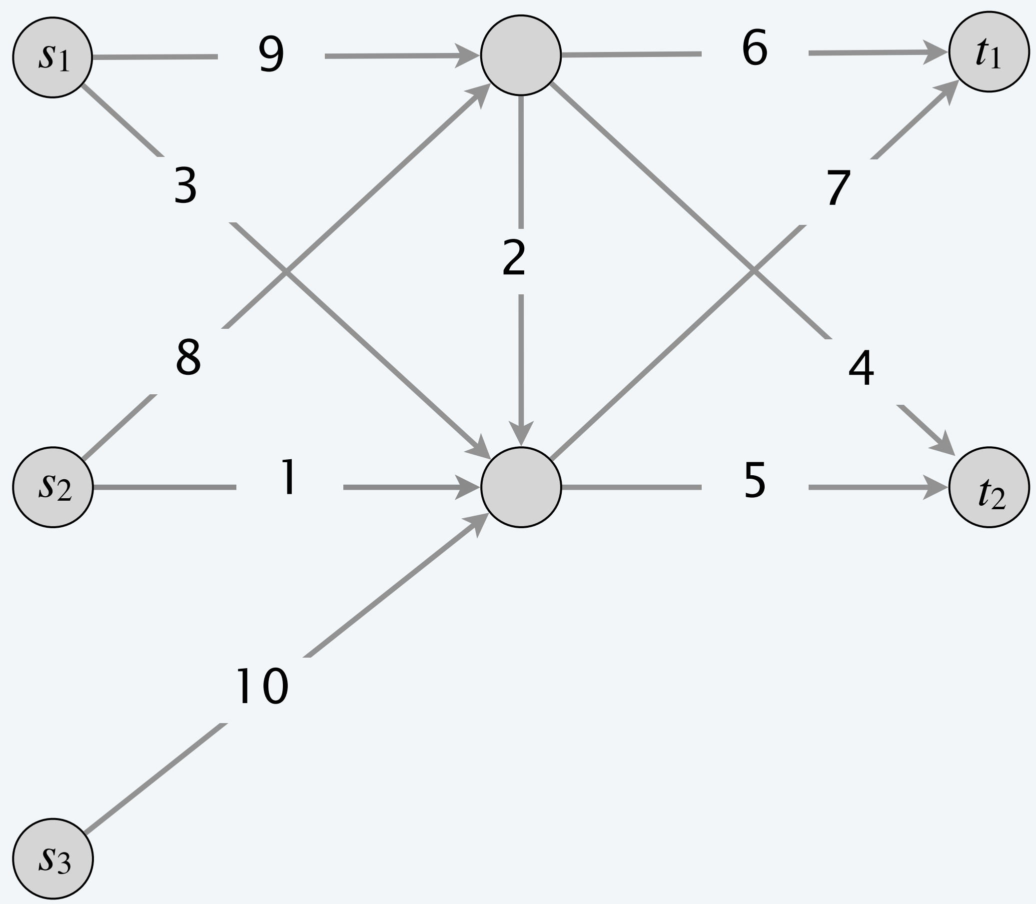
C. Limiti inferiori sui flussi di arco.

D. Tutte le precedenti.

Sorgenti e pozzi multipli

Def. Dato un digrafo $G = (V, E)$ con capacità agli archi $c(e) \geq 0$ e più nodi sorgenti e più nodi pozzo, trovare un massimo flusso che può essere mandato dai nodi sorgenti ai nodi pozzo.

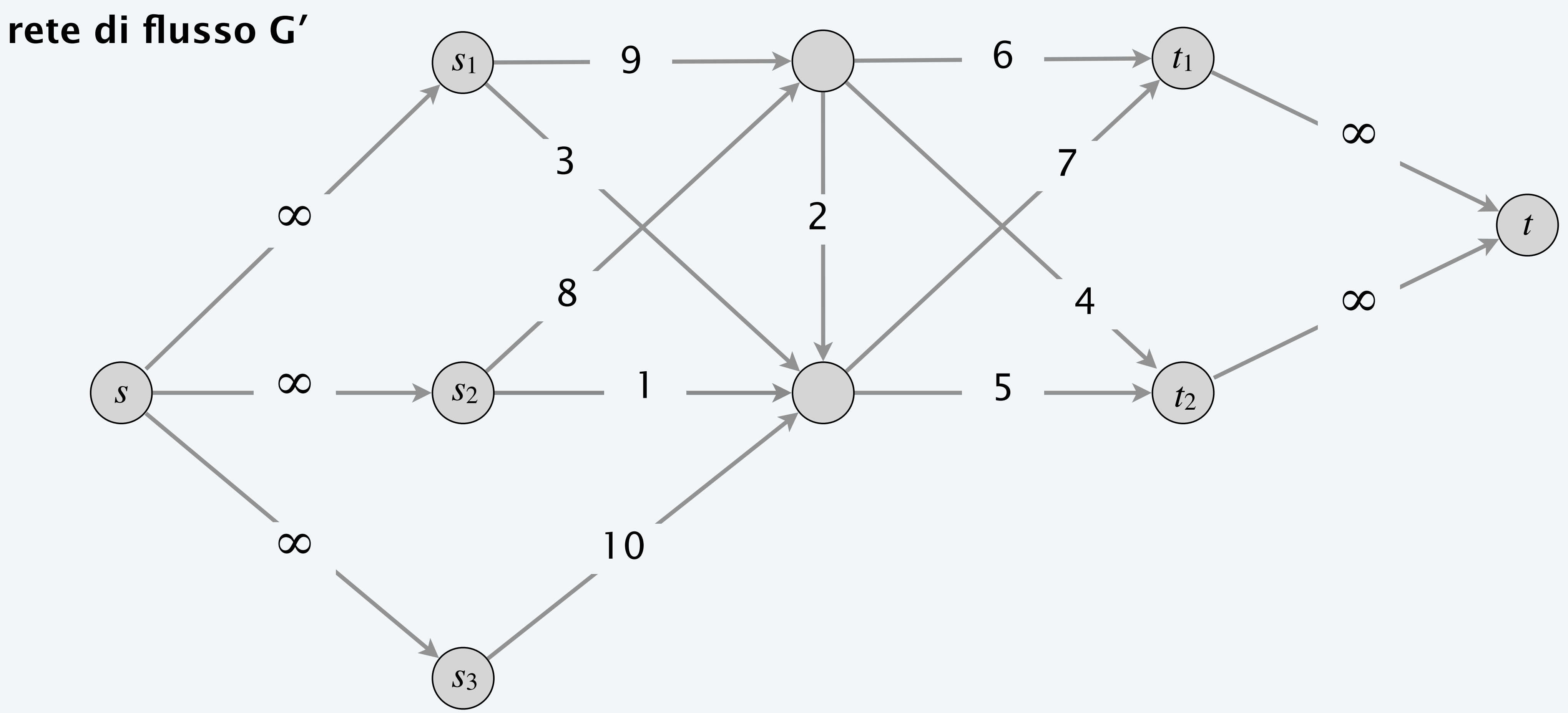
rete di flusso G



Sorgenti e pozzi multipli: formulazione massimo flusso

- Aggiungi un nodo sorgente s ed un nodo pozzo t.
- Per ogni nodo sorgente originale s_i aggiungi arco (s, s_i) con capacità ∞.
- Per ogni nodo pozzo originale t_j, aggiungi arco (t_j, t) con capacità ∞.

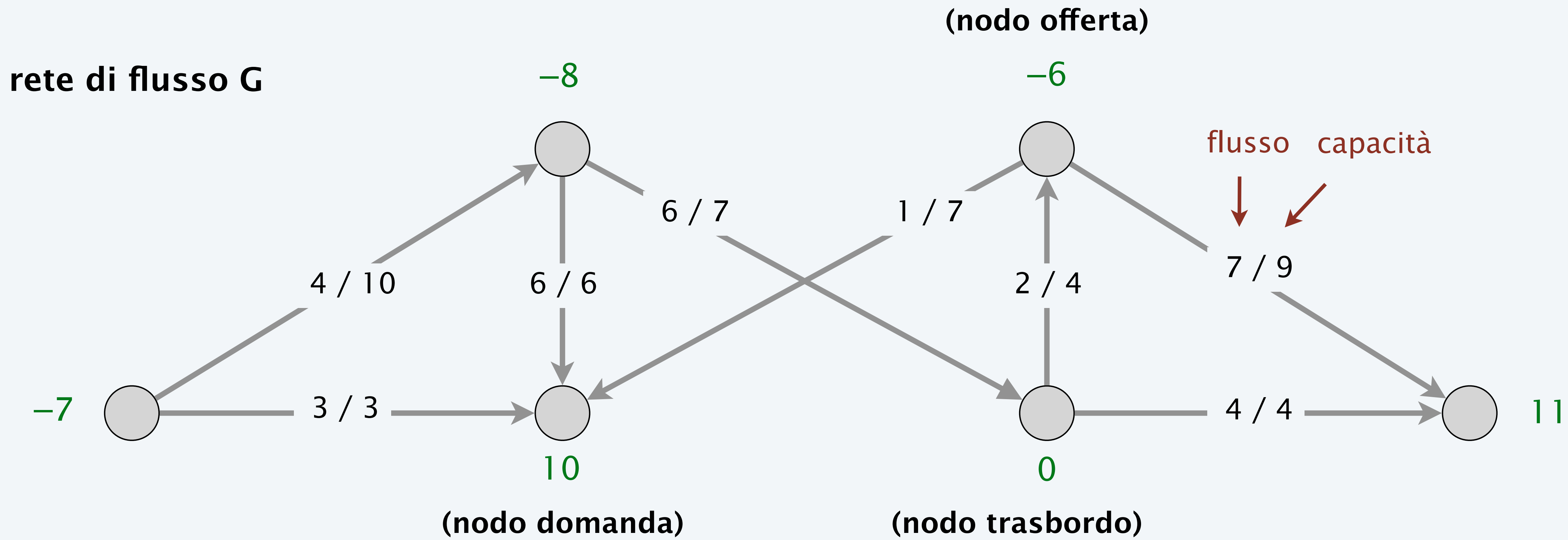
Prop. Corrispondenza 1–1 tra flussi in G e G'.



Circolazione con domande e offerte

Def. Dato un digrafo $G = (V, E)$ con capacità agli archi $c(e) \geq 0$ e domande ai nodi $d(v)$, una circolazione è una funzione $f(e)$ che soddisfa:

- Per ogni $e \in E$: $0 \leq f(e) \leq c(e)$ (capacità)
- Per ogni $v \in V$: $\sum_{e \text{ in to } v} f(e) - \sum_{e \text{ out of } v} f(e) = d(v)$ (conservazione del flusso)



Circolazione con domande e offerte: formulazione massimo flusso

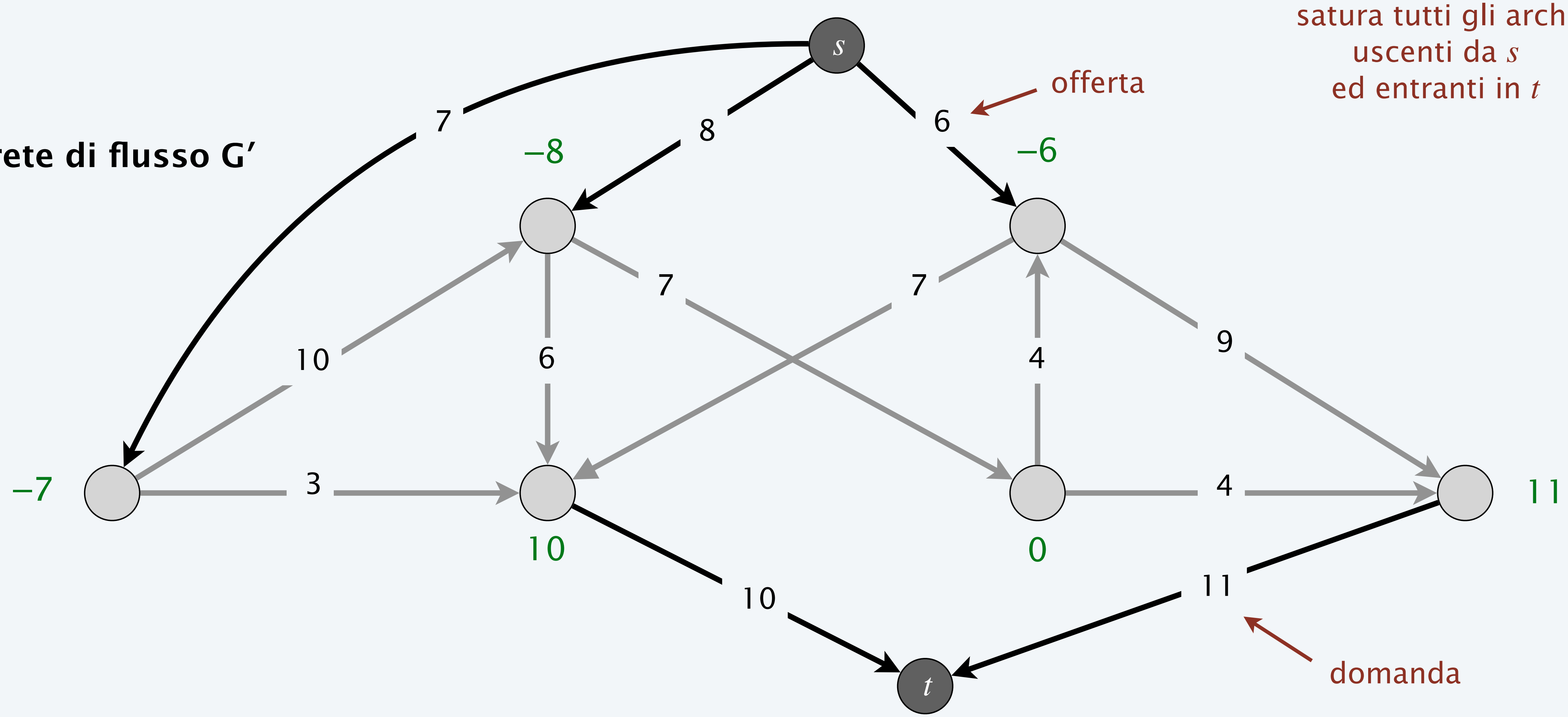
- Aggiungi una sorgente s e un pozzo t.
- Per ogni v con $d(v) < 0$, aggiungi arco (s, v) con capacità $-d(v)$.
- Per ogni v con $d(v) > 0$, aggiungi arco (v, t) con capacità $d(v)$.

Prop. G ha una circolazione sse G' ha un flusso di valore D =

$$\sum_{v\,:\,d(v)>0} d(v) = \sum_{v\,:\,d(v)<0} -d(v)$$

satura tutti gli archi uscenti da s ed entranti in t

rete di flusso G′



Circolazione con domande e offerte

Teorema di interezza. Se tutte le capacità e le domande sono intere, ed una circolazione esiste, allora ne esiste una a componenti intere.

Dim. Segue dalla formulazione massimo flusso + teorema di interezza per il massimo flusso.

Teorema. Dato (V, E, c, d), non esiste una circolazione sse esiste una partizione dei nodi (A, B) tale che $\Sigma_{v \in B} d(v) > cap(A, B)$.

la domanda dei nodi in B eccede
l'offerta dei nodi in B più
la capacità max degli archi da A a B

Pf sketch. Si consideri il taglio minimo in G'.

Circolazione con domande, offerte, e limiti inferiori

Def. Dato un digrafo $G = (V, E)$ con capacità agli archi $c(e) \geq 0$, limiti inferiori $\ell(e) \geq 0$, e domande ai nodi $d(v)$, una circolazione $f(e)$ è una funzione tale che:

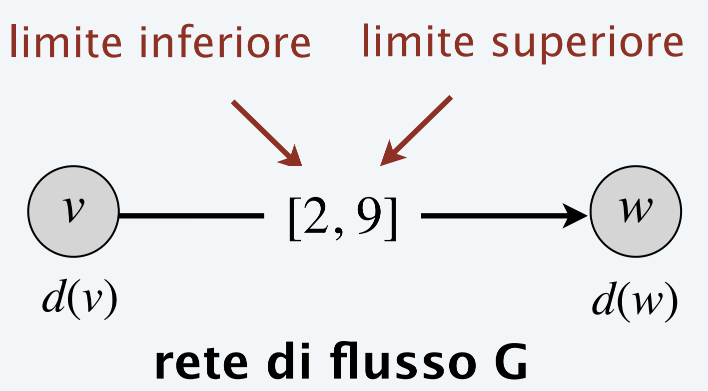
- Per ogni $e \in E$: $\boxed{\ell(e) \leq f(e)} \leq c(e)$ (capacità)
- Per ogni $v \in V$: $\sum_{e \text{ in to } v} f(e) - \sum_{e \text{ out of } v} f(e) = d(v)$ (conservazione del flusso)

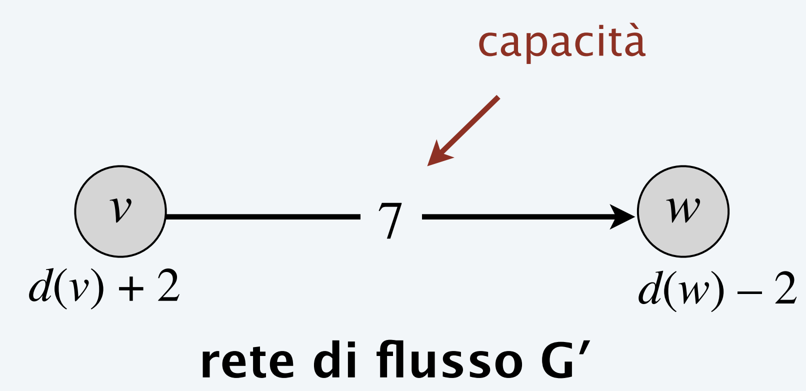
Problema di circolazione con limiti inferiori. Dato (V, E, ℓ, c, d), esiste una circolazione ammissibile?

Circolazione con domande, offerte, e limiti inferiori

Formulazione massimo flusso. Modelliamo i limiti inferiori come domande.

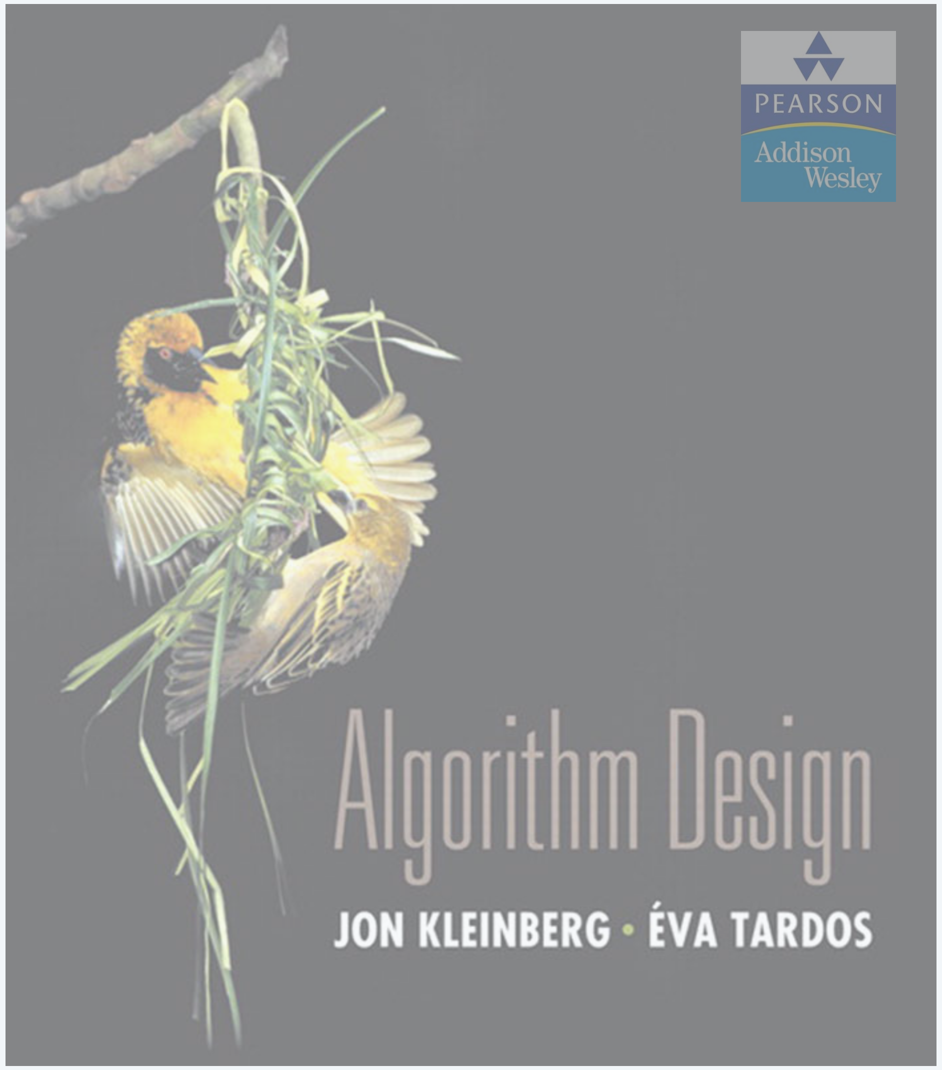
- Manda $\ell(e)$ unità di flusso lungo l'arco e.
- Aggiorna le domande di entrambi gli estremi.





Teorema. Esiste una circolazione in G sse esiste una circolazione in G'. Inoltre, se le domande, le capacità e i limiti in G sono tutti interi, allora esiste una circolazione in G che ha componenti intere.

Pf sketch. $f(e)$ è una circolazione in G sse $f'(e) = f(e) - \ell(e)$ è una circolazione in G'.



SECTION 7.8

7. FLUSSI DI RETE II

- *bipartite matching*
- *disjoint paths*
- *extensions to max flow*
- *progetto di sondaggi*
- *airline scheduling*
- *image segmentation*
- *project selection*
- *baseball elimination*

Progetto di sondaggi

- Sondaggio a n_1 consumatori riguardo n_2 prodotti.
- Il consumatore i può rispondere sul prodotto j solo se lo possiede.
- Al consumatore i vanno poste un numero di domande tra c_i e c_i'.
- Sul prodotto j vanno poste complessivamente tra p_j e p_j' domande.

una domanda nel sondaggio per ogni prodotto

Scopo. Progettare un sondaggio che soddisfi queste specifiche, se possibile.

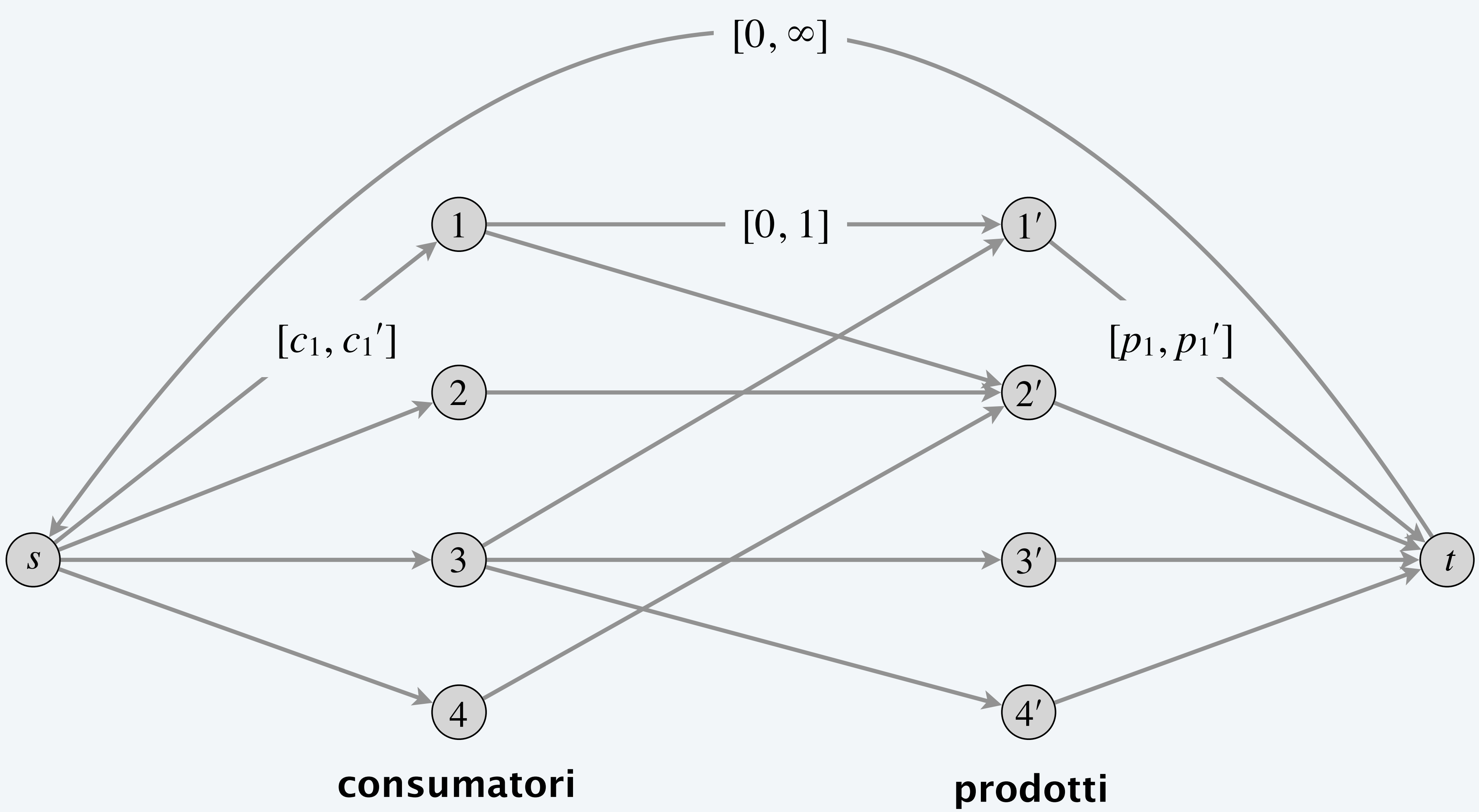
Abbinamento bipartito perfetto. Caso particolare con $c_i = c_i' = p_j = p_j' = 1$.

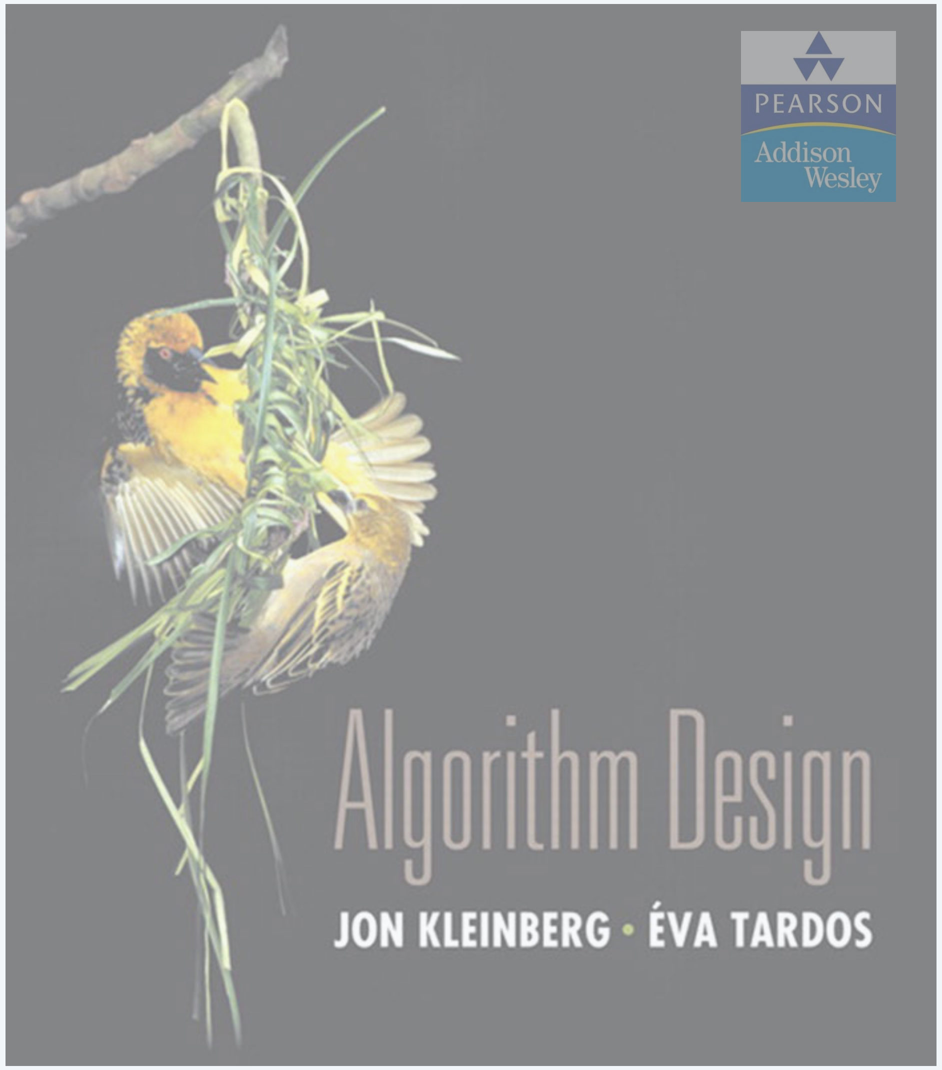
Progetto di sondaggi

Formulazione. Problema di circolazione con limiti inferiori.

- Aggiungi arco (i, j) se consumatore j ha il prodotto i.
- Aggiungi arco da s al consumatore j.
- Aggiungi arco dal prodotto i a t.
- Aggiungi un arco da t ad s.
- Tutte le domande = 0.
- Circolazione intera $\Leftrightarrow$ sondaggio ammissibile.

tutte le offerte e le domande sono 0





SECTION 7.9

7. NETWORK FLOW II

- *bipartite matching*
- *disjoint paths*
- *extensions to max flow*
- *survey design*
- *schedulazione di linee aeree*
- *image segmentation*
- *project selection*
- *baseball elimination*

Schedulazione di linee aeree

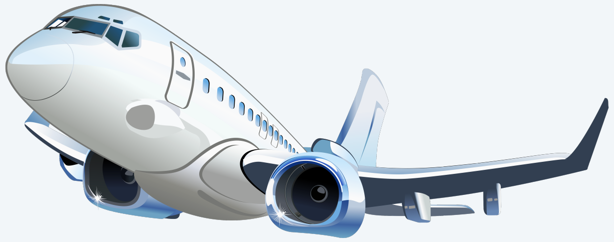
Schedulazione di linee aeree.

- Problema computazionale complesso affrontato dalle compagnie aeree.
- Deve produrre schedule efficienti in termini di uso di equipaggio, allocazione equipaggi, e soddisfazione utenti.
- Uno dei più grossi consumatori di tecniche algoritmiche.

anche in presenza di eventi imprevisti, quali eventi atmosferici e guasti

"Problema giocattolo."

- Gestire gli equipaggi di volo riutilizzandoli su più di un volo.
- Input: insieme di k voli per un dato giorno.
- Volo i parte dall'origine o_i al tempo s_i ed arriva alla destinazione d_i al tempo f_i.
- Minimizzare il numero di equipaggi di volo.

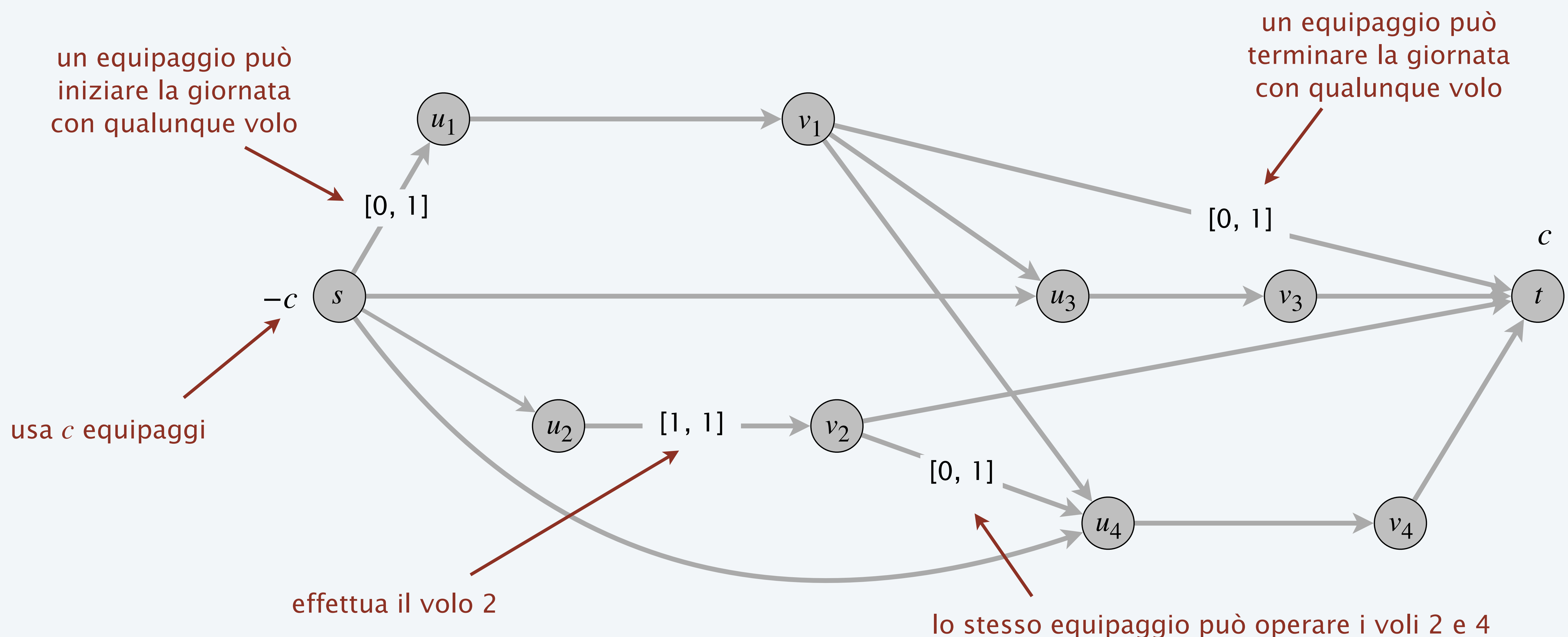


Schedulazione di linee aeree

Formulazione come circolazione. [per testare se c equipaggi bastano]

- Per ogni volo i, includi due nodi u_i e v_i.
- Aggiungi sorgente s con domanda $-c$, e archi (s, u_i) con capacità 1.
- Aggiungi pozzo t con domanda c, e archi (v_i, t) con capacità 1.
- Per ogni i, aggiungi arco (u_i, v_i) con limite inferiore e capacità 1.
- Se volo j raggiungibile da i, aggiungi arco (v_i, u_j) con capacità 1.

u_i = inizio del volo i
v_i = fine del volo i



Schedulazione di linee aeree: tempo di esecuzione

Teorema. Il problema della schedulazione di linee aeree è risolvibile in tempo $O(k^3 \log k)$.

Dim.

- k = numero di voli.
- c = numero di equipaggi (sconosciuto).
- $O(k)$ nodi, $O(k^2)$ archi.
- Al più k equipaggi sono necessari.
  $\Rightarrow$ risolvi $\log_2 k$ problemi di circolazione. ⟵ ricerca binaria per il valore minimo c^*
- Il valore di qualunque flusso è tra 0 e k.
  $\Rightarrow$ al più k aumenti di flusso per problema di circolazione.
- Tempo complessivo = $O(k^3 \log k)$.

Nota. Si può risolvere in tempo $O(k^3)$ riformulandolo come problema di flusso minimo.

Schedulazione di linee aeree: commenti

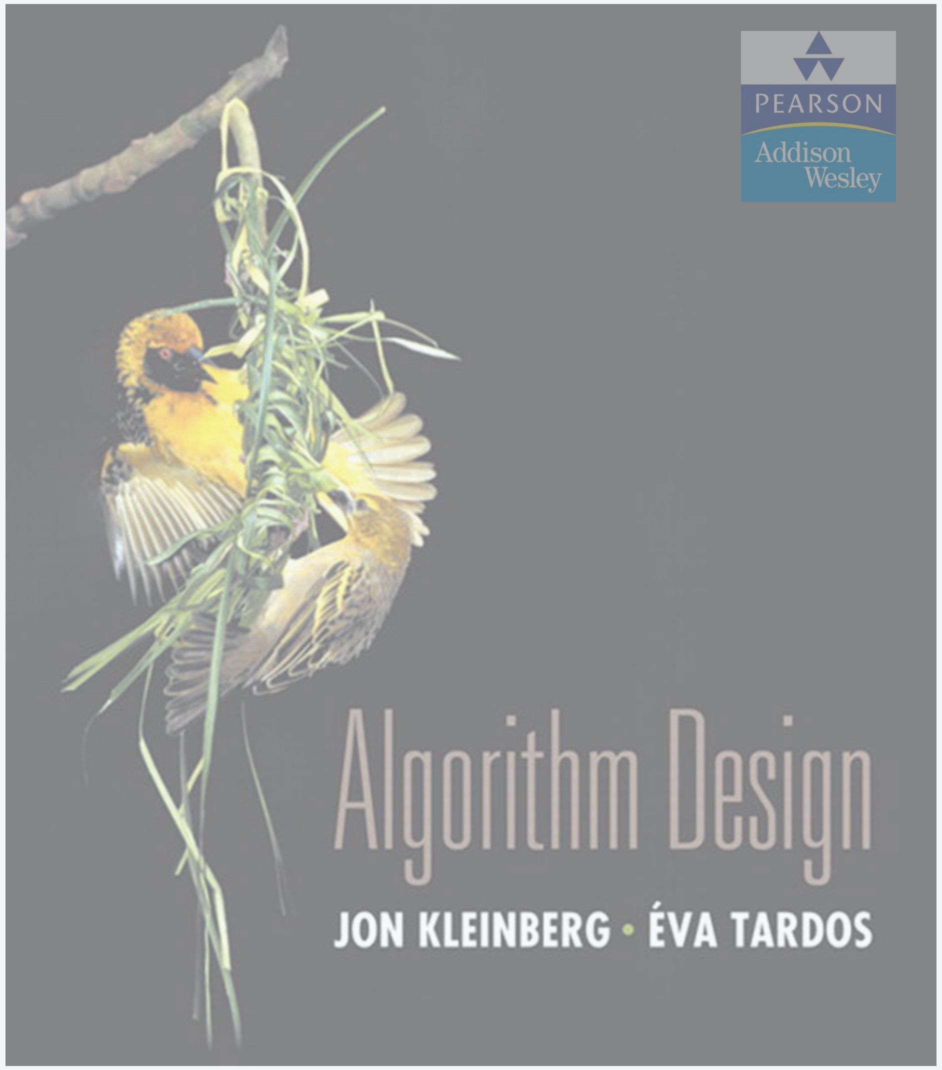
Nota. Abbiamo risolto una versione giocattolo del problema reale.

Modelli del problema reale includono innumerevoli altri fattori:

- Regolamenti sindacali: p.e., equipaggi di volo possono volare solo un certo numero di ore consecutive in una data finestra temporale.
- Serve uno schedule ottimo sull'intero orizzonte temporale,non 1 giorno.
- I posti a sedere rimasti liberi hanno un costo.
- Non sempre i voli partono o arrivano in orario.
- Si vuole ottimizzare sia lo schedule dei voli che la struttura delle tariffe.

Messaggio.

- La nostra soluzione è una tecnica di larga applicabilità per il riuso efficiente di risorse limitate, ma semplifica drasticamente il vero problema affrontato dalle compagnie aeree.
- Tecniche di flusso sono utili per risolvere problemi di schedulazione di linee aeree (e sono largamente utilizzate in pratica).
- Operare una società aerea in modo efficiente è un problema difficile.



SECTION 7.10

7. FLUSSI DI RETE II

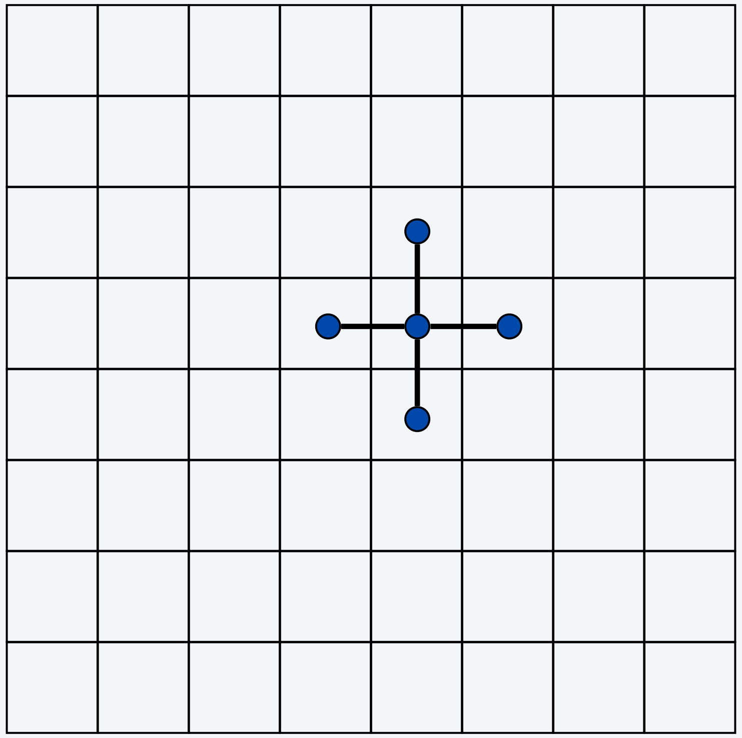
- *bipartite matching*
- *disjoint paths*
- *extensions to max flow*
- *survey design*
- *airline scheduling*
- *segmentazione di immagini*
- *project selection*
- *baseball elimination*

Segmentazione di immagini.

- Dividere un'immagine in regioni coerenti.
- Problema centrale nell'elaborazione di immagini.

Es. Separare umano e robot dallo sfondo.

Segmentazione di immagini



Segmentazione primo piano / sfondo.

- Etichetta ogni pixel dell'immagine come appartenente al primo piano o allo sfondo.
- V = insieme dei pixel, E = coppie di pixel adiacenti.
- $a_i \geq 0$ è probabilità che pixel i sia di primo piano.
- $b_i \geq 0$ è probabilità che pixel i sia dello sfondo.
- $p_{ij} \geq 0$ è una penalità dell'etichettare uno tra i e j come primo piano, e l'altro come sfondo.

Scopi.

- Accuratezza: se $a_i > b_i$ in isolamento, etichetta i come primo piano.
- Levigatezza: se molti vicini di i sono classificati come primo piano, dovremmo essere inclinati a classificare i come primo piano.
- Trova una partizione (A, B) che massimizzi:

primo piano sfono

$$\sum_{i \in A} a_i + \sum_{j \in B} b_j - \sum_{\substack{(i,j) \in E \\ |A \cap \{i,j\}| = 1}} p_{ij}$$

Segmentazione di immagini

Formulazione come problema di minimo taglio.

- Massimizzazione.
- Nessuna sorgente o pozzo.
- Grafo non orientato.

Trasformalo in un problema di minimizzazione.

- Massimizzare $\displaystyle \sum_{i \in A} a_i \; + \; \sum_{j \in B} b_j \; - \sum_{\substack{(i,j) \in E \\ |A \cap \{i,j\}| = 1}} p_{ij}$

- è equivalente a minimizzare

$$\underbrace{\left(\sum_{i \in V} a_i \; + \; \sum_{j \in V} b_j \right)}_{\text{una costante}} \; - \; \sum_{i \in A} a_i \; - \; \sum_{j \in B} b_j \; + \sum_{\substack{(i,j) \in E \\ |A \cap \{i,j\}| = 1}} p_{ij}$$

- o in alternativa $\displaystyle \sum_{j \in B} a_j \; + \; \sum_{i \in A} b_i \; + \sum_{\substack{(i,j) \in E \\ |A \cap \{i,j\}| = 1}} p_{ij}$

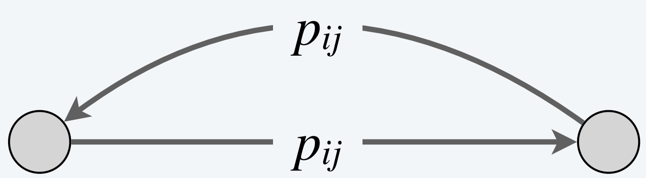
Segmentazione di immagini

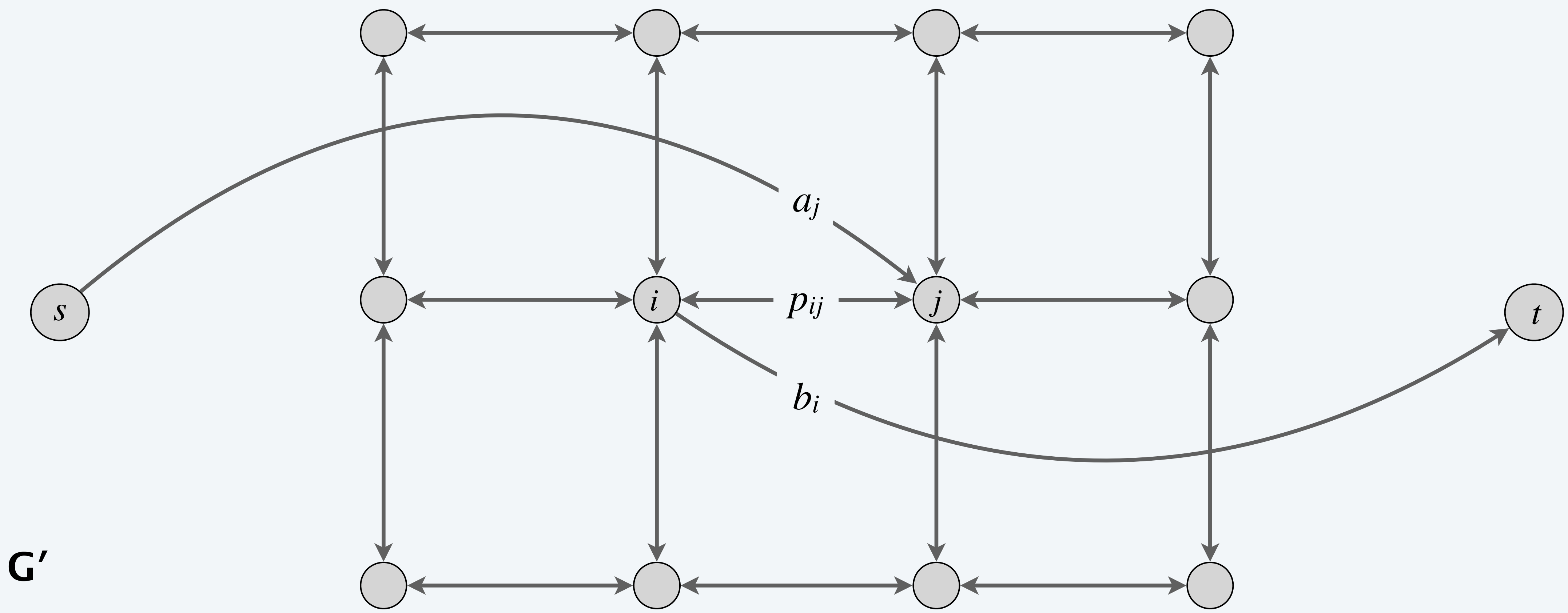
Formulazione di minimo taglio su $G' = (V', E')$.

- Includi un nodo per ogni pixel
- Usa due archi contrapposti per ogni arco non orientato.
- Aggiungi sorgente s (corrisponde al primo piano).
- Aggiungi pozzo t (corrisponde allo sfondo).

arco in G

due archi contrapposti in G′





Segmentazione di immagini

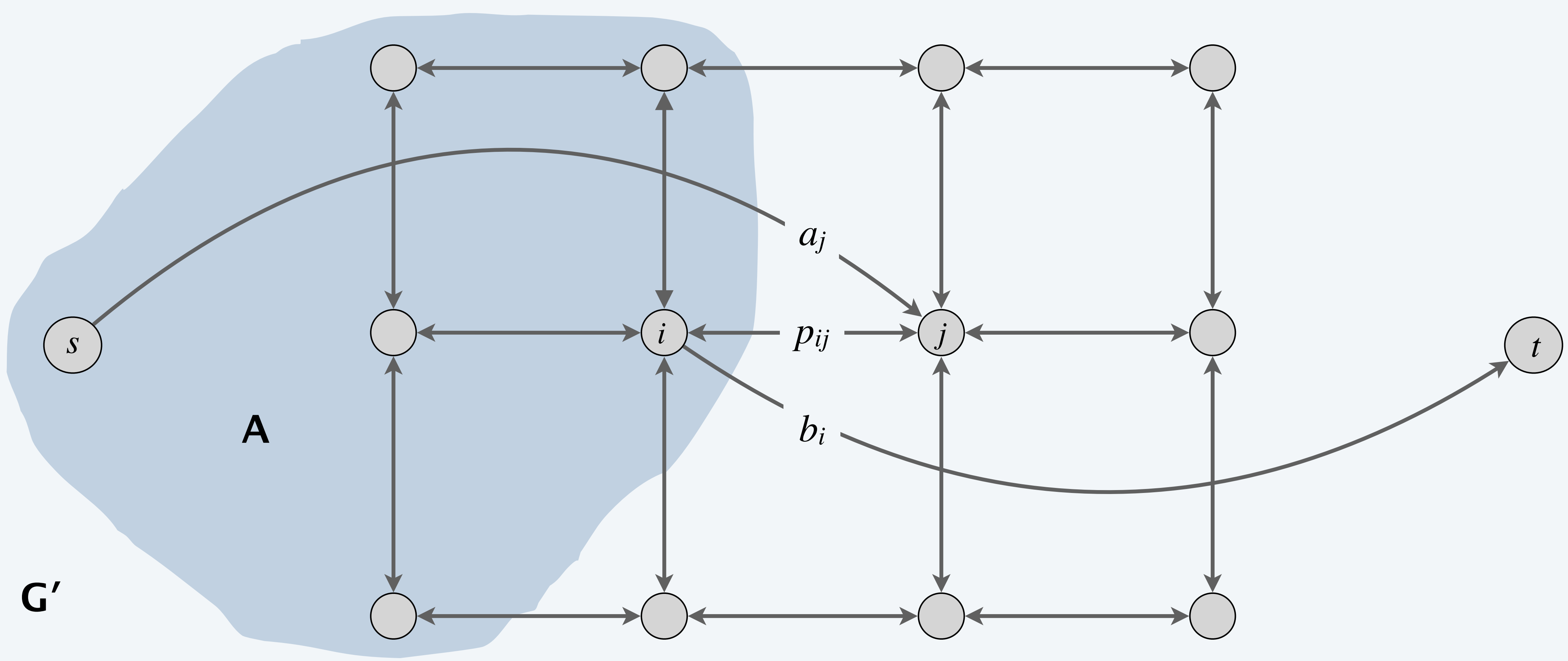
Considera un minimo taglio (A, B) in G'.

- A = primo piano.

$$cap(A,B) \;=\; \sum_{j \in B} a_j \;+\; \sum_{i \in A} b_i \;+\; \sum_{\substack{(i,j) \in E \\ i \in A,\; j \in B}} p_{ij}$$

se i e j sono su piani diversi, p_{ij} è contato esattamente 1 volta

- Precisamente la quantità che vogliamo minimizzare.



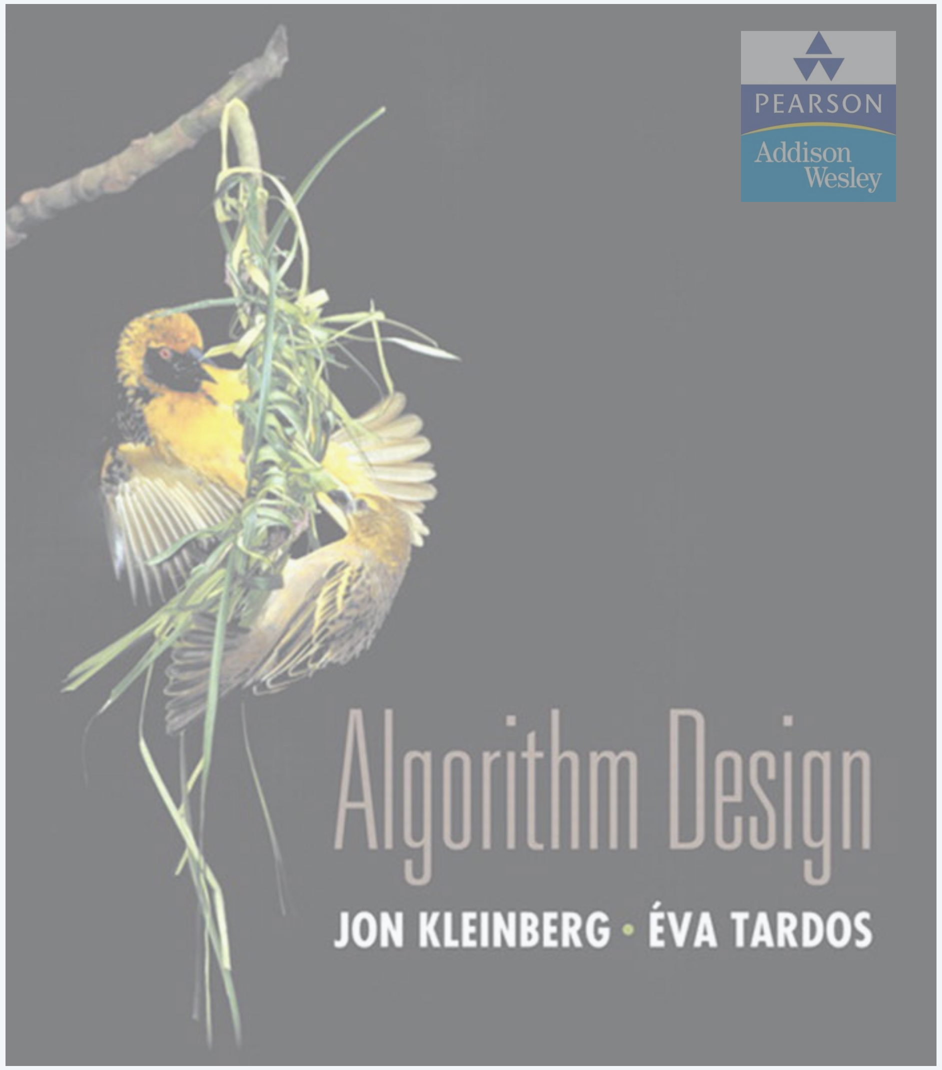
Segmentazione di immagini con Grabcut

Grabcut. [Rother–Kolmogorov–Blake 2004]

“GrabCut” — Interactive Foreground Extraction using Iterated Graph Cuts

Carsten Rother* Vladimir Kolmogorov† Andrew Blake‡
Microsoft Research Cambridge, UK

Figure 1: **Three examples of GrabCut .** The user drags a rectangle loosely around an object. The object is then extracted automatically.



SECTION 7.11

7. FLUSSI DI RETE II

- bipartite matching
- disjoint paths
- extensions to max flow
- survey design
- airline scheduling
- image segmentation
- *selezione di progetti*
- baseball elimination

Selezione di progetti (problema della chiusura a peso massimo)

Progetti con prerequisiti.

può essere positivo o negativo

- Insieme P di possibili progetti: ogni progetto v ha un ricavo p_v.
- Insieme di prerequisiti E: $(v, w) \in E$ significa che w è un prerequisito di v.
- Un sottoinsieme di progetti $A \subseteq P$ è ammissibile se il prerequisito di ogni progetto di A appartiene ad A.

Problema della selezione dei progetti. Dato un insieme di progetti P e dei prerequisiti E, scegli un sottoinsieme ammissibile di progetti che massimizzi il ricavo totale.

MANAGEMENT SCIENCE
Vol. 22, No. 11, July, 1976
Printed in U.S.A.

MAXIMAL CLOSURE OF A GRAPH AND APPLICATIONS TO COMBINATORIAL PROBLEMS*†

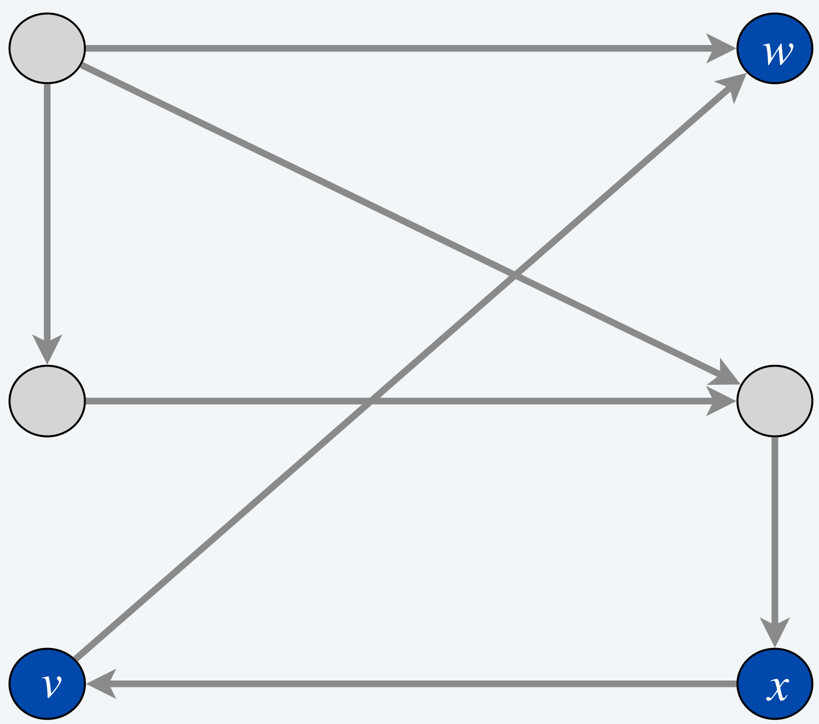
JEAN-CLAUDE PICARD

Ecole Polytechnique, Montreal

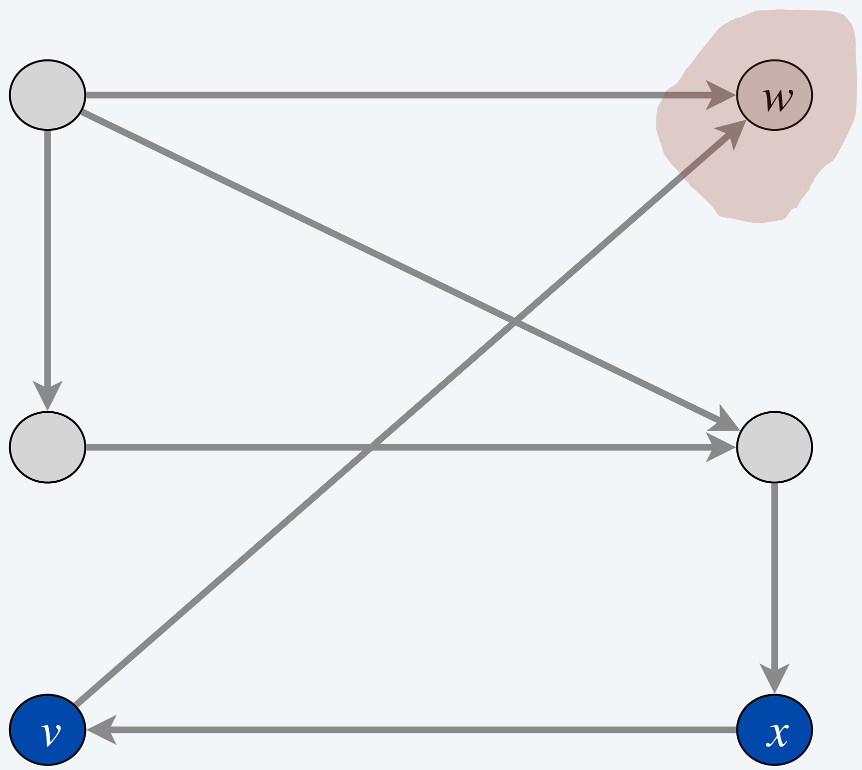
This paper generalizes the selection problem discussed by J. M. Rhys [12], J. D. Murchland [9], M. L. Balinski [1] and P. Hansen [4]. Given a directed graph G, a closure of G is defined as a subset of nodes such that if a node belongs to the closure all its successors also belong to the set. If a real number is associated to each node of G a maximal closure is defined as a closure of maximal value.

Selezione di progetti: grafo dei prerequisiti

(Di)grafo dei prerequisiti. Aggiungi l'arco (v, w) se w è un prerequisito di v.



{ v, w, x } è ammissibile

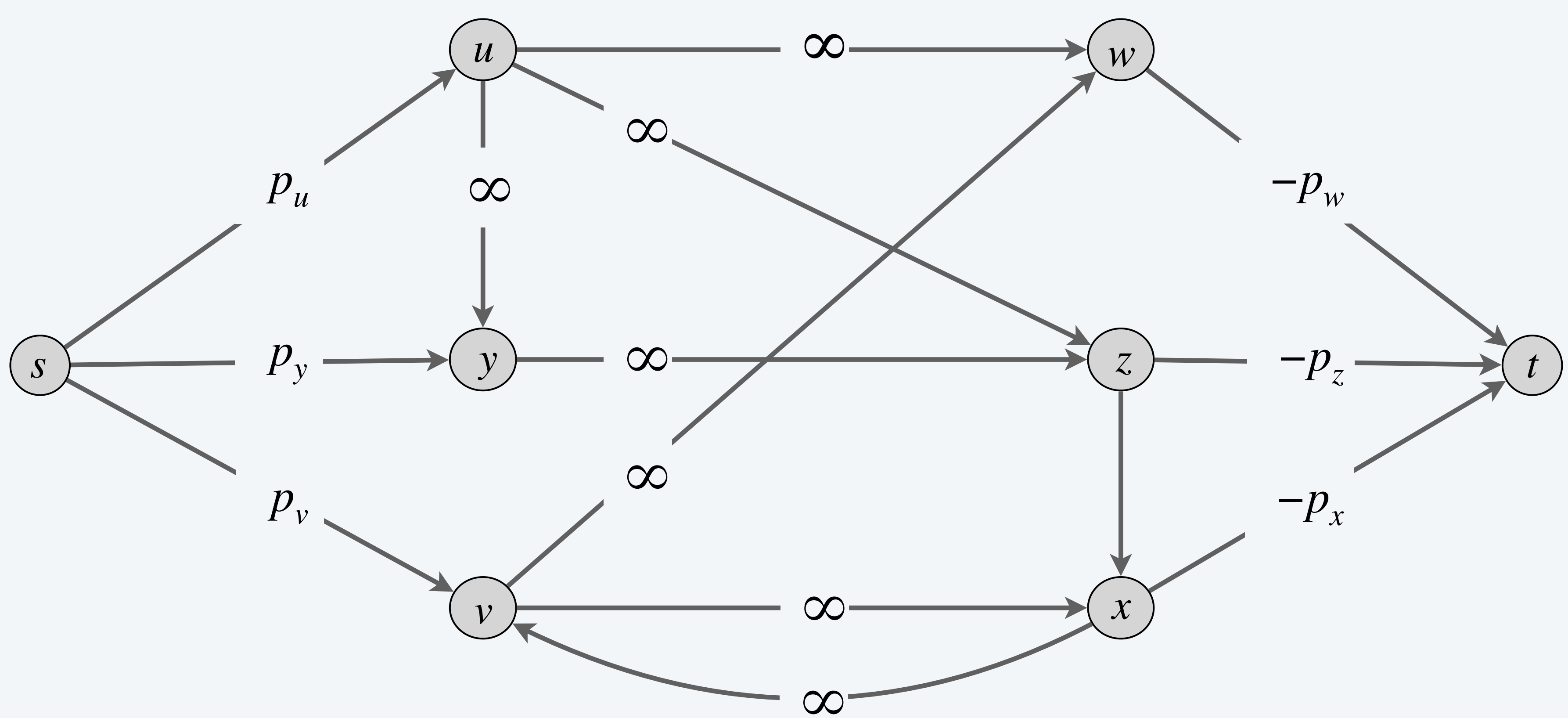


{ v, x } è inammissibile

Selezione di progetti: formulazione minimo taglio

Formulazione minimo taglio.

- Assegna una capacità ∞ ad ogni arco prerequisito.
- Aggiungi arco (s, v) con capacità p_v se $p_v > 0$.
- Aggiungi arco (v, t) con capacità $-p_v$ se $p_v < 0$.
- Per comodità di notazione, definiamo $p_s = p_t = 0$.



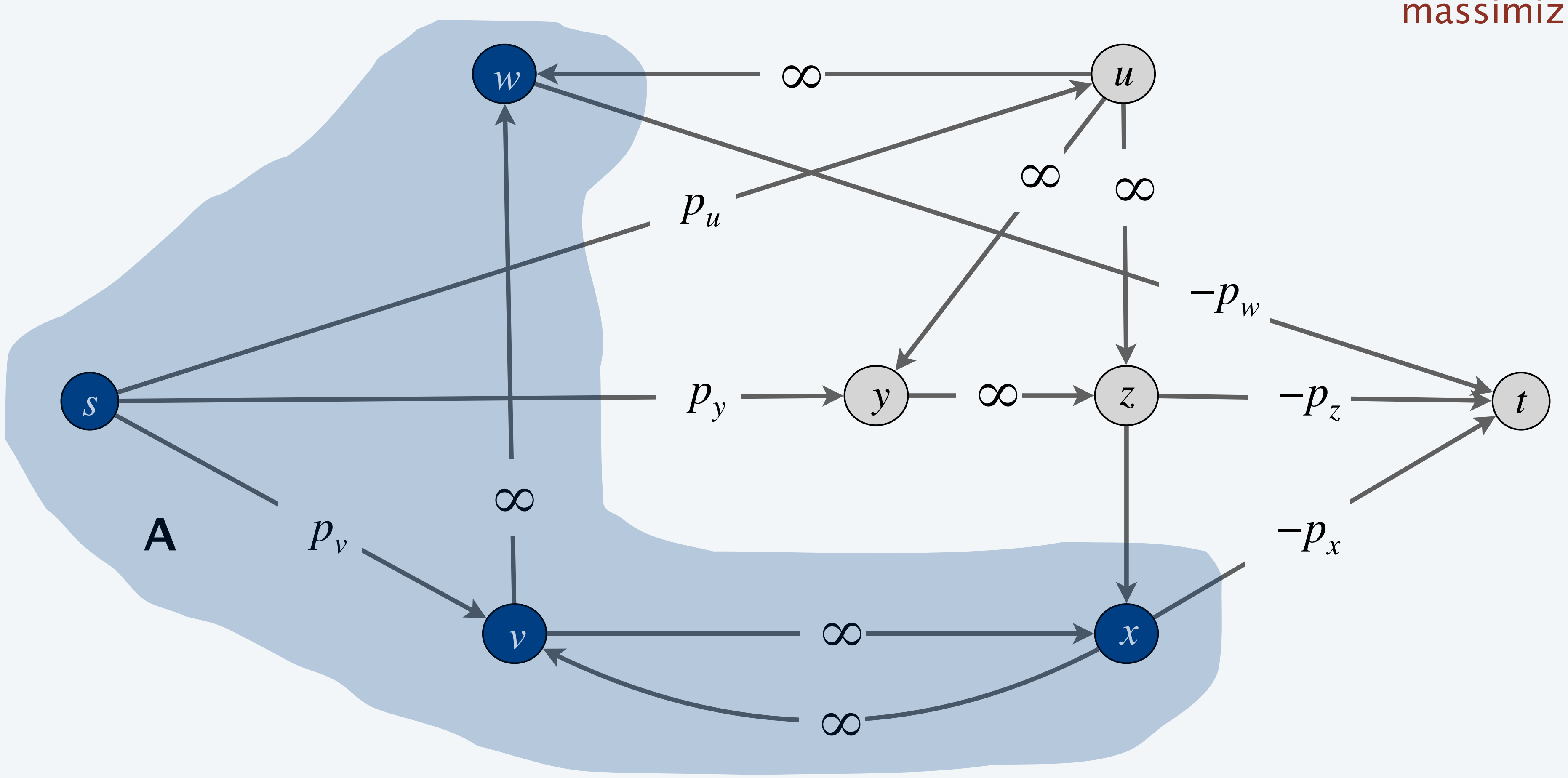
Selezione di progetti: formulazione minimo taglio

Prop. (A, B) è un taglio minimo sse $A - \{ s \}$ è un insieme ottimo di progetti.

- Gli archi a capacità infinita garantiscono che $A - \{ s \}$ sia ammissibile.
- Ricavo è max poiché: $$cap(A, B) = \sum_{v \in B:\ p_v > 0} p_v + \sum_{v \in A:\ p_v < 0} (-p_v)$$ $$= \sum_{v:\ p_v > 0} p_v - \sum_{v \in A} p_v$$

una costante

minimizzare questo equivale a massimizzare i ricavi



Estrazione mineraria a cielo aperto

Estrazione mineraria a cielo aperto. [studiato dall'inizio degli anni 1960]

- Blocchi di terra estratti dalla superficie per ricavarne minerale.
- Ogni blocco v ha un valore netto p_v = valore minerario − costo di lavorazione.
- Non si può rimuovere il blocco v finché non si rimuovono sia w che x.