

Lecture 10: Graph Algorithms (IV)

Dr. Tsung-Wei Huang

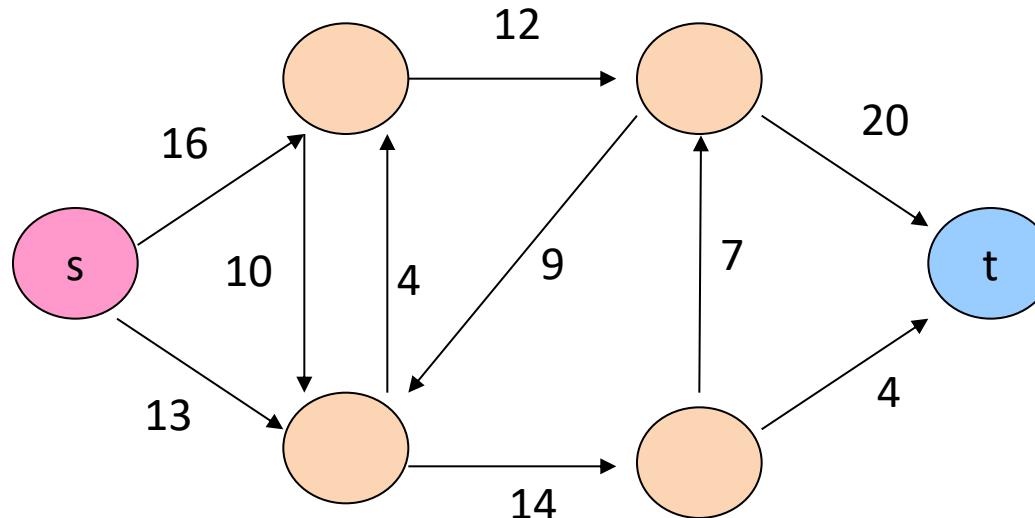
Department of Electrical and Computer Engineering
University of Utah, Salt Lake City, UT



Recap: Maximum Flow Problem

□ Network flow problem

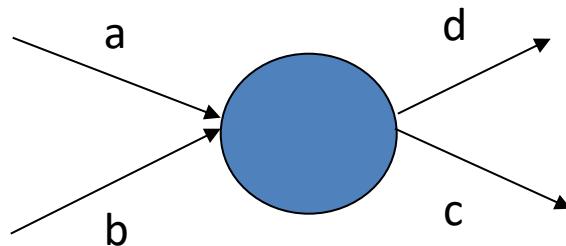
- A **flow network** $G=(V,E)$: a directed graph, where each edge $(u,v) \in E$ has a nonnegative **capacity** $c(u,v) \geq 0$.
- If $(u,v) \notin E$, we can assume that $c(u,v)=0$.
- two distinct vertices : a **source** s and a **sink** t .



Flow Constraint

- **$G=(V,E)$: a flow network with capacity function c .**
- **s -- the source and t -- the sink.**
- **A flow $f(u, v)$ in G must satisfy**
 1. **Capacity constraint**
 - For all $u, v \in V$, we require $f(u, v) \leq c(u, v)$.
 2. **Flow conservation**
 - For all $u \in V - \{s, t\}$, we require

$$\sum_{e.in.v} f(e) = \sum_{e.out.v} f(e)$$



$$a+b = d+c$$

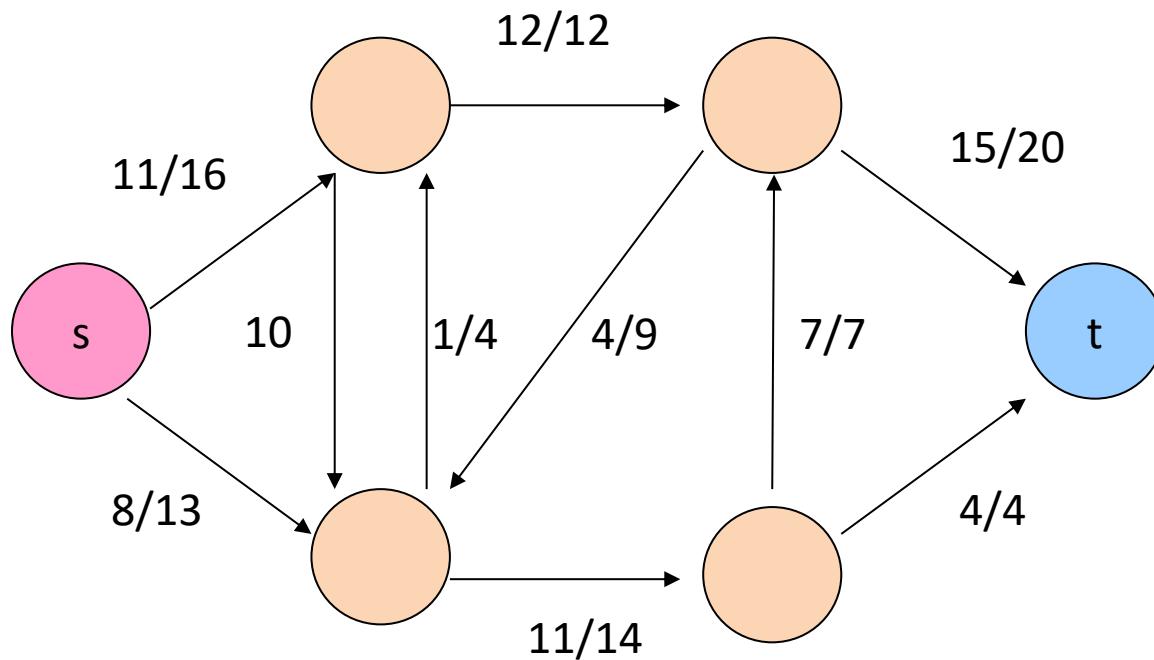
Objective

- The quantity $f(u, v)$ is called the **net flow** from vertex u to vertex v .
- The **value** of a flow is defined as

$$|f| = \sum_{v \in V} f(s, v)$$

- The total flow from source to any other vertices.
- The same as the total flow from any vertices to **the sink**.

Example



A flow f in G with value $|f| = 19$

Method to Compute Maximum Flow

FORD-FULKERSON-FRAMEWORK(G, s, t)

initialize flow f to 0

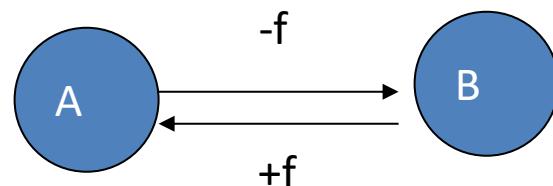
while there exists an *augmenting* path p

do *augment* flow f along p

return f

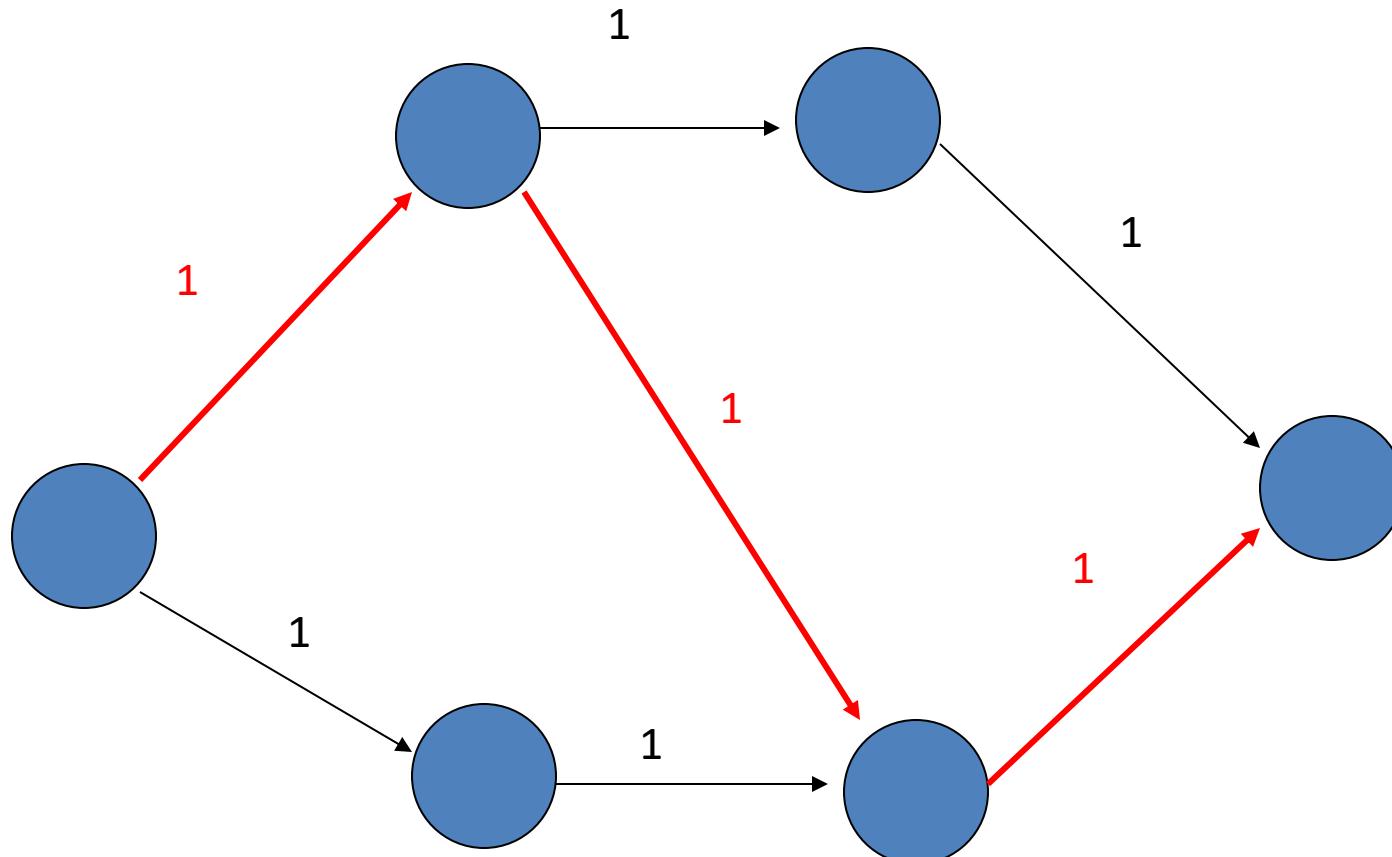
Residual Network

- Residual network defines edges to admit net flow
 - The amount of additional net flow from u to v before exceeding the capacity $c(u,v)$ is the **residual capacity** of (u,v) , given by:
 - In the regular direction: $c_f(u,v) = c(u,v) - f(u,v)$
 - In the opposite direction: $c_f(v, u) = c(v, u) + f(u, v)$.
- If you flow f from A to B
 - Subtract the regular direction capacity from f
 - Add f to the opposite direction capacity



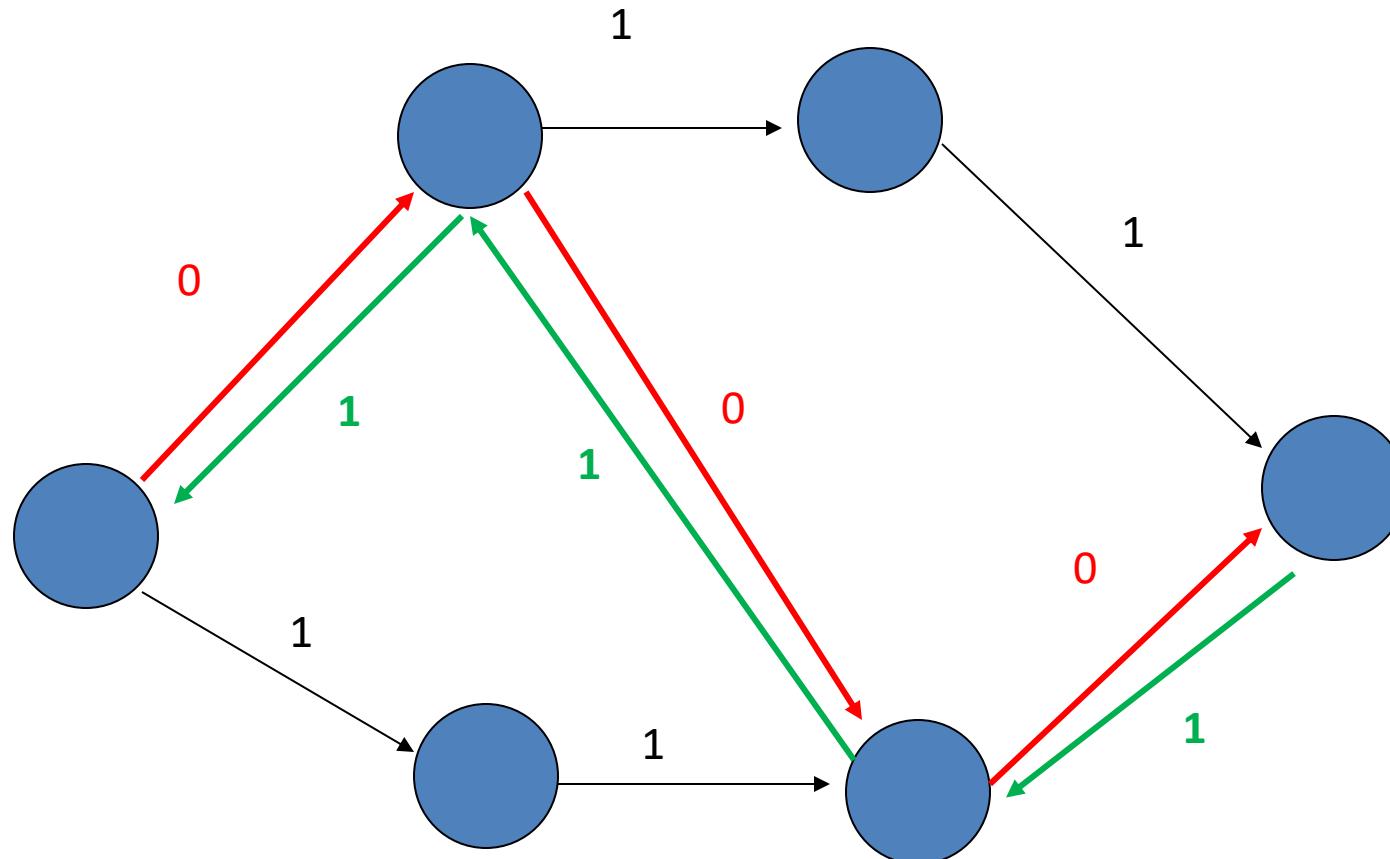
Example

- DFS augments a unit flow in the first iteration



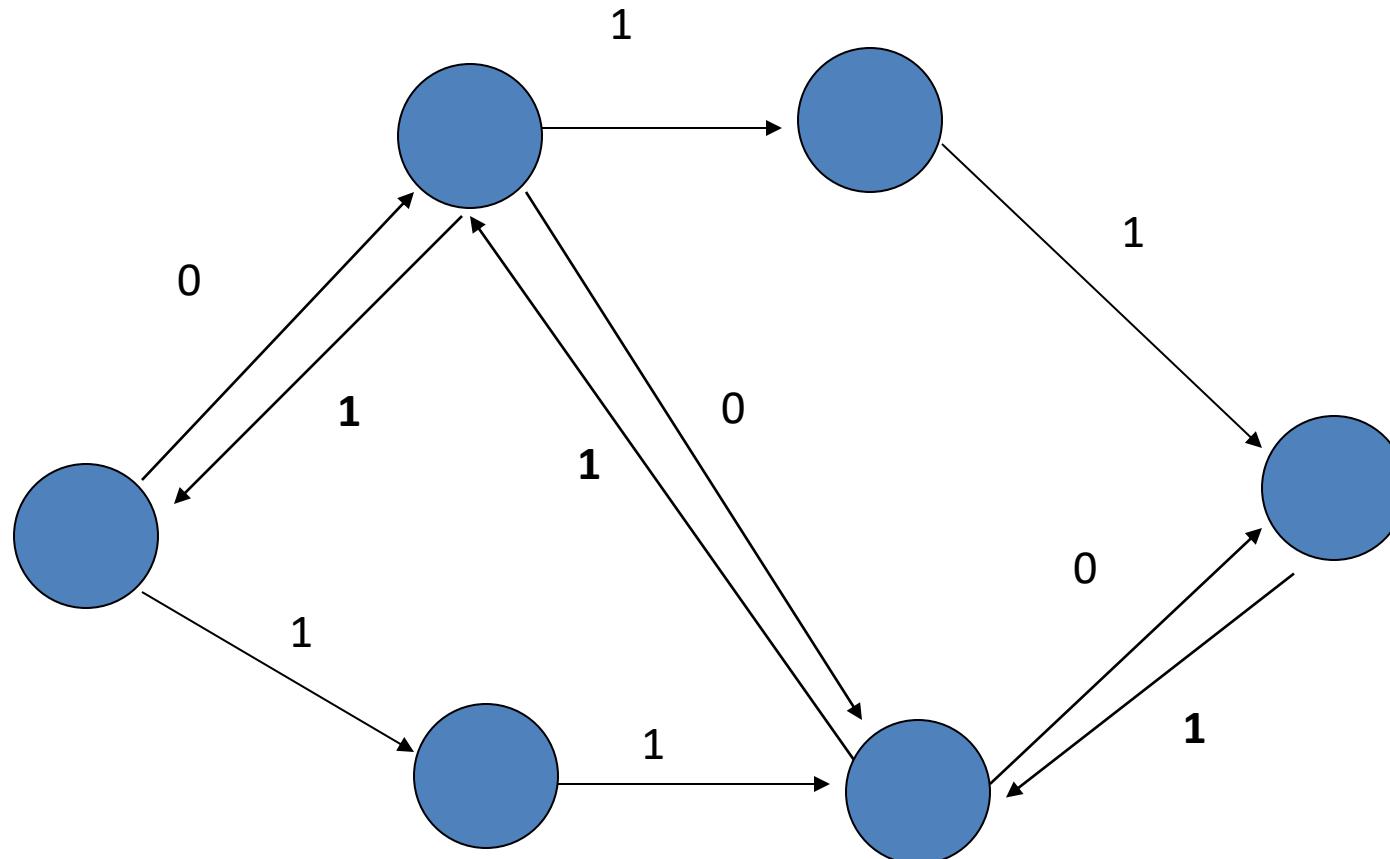
Example

□ Update the residual network



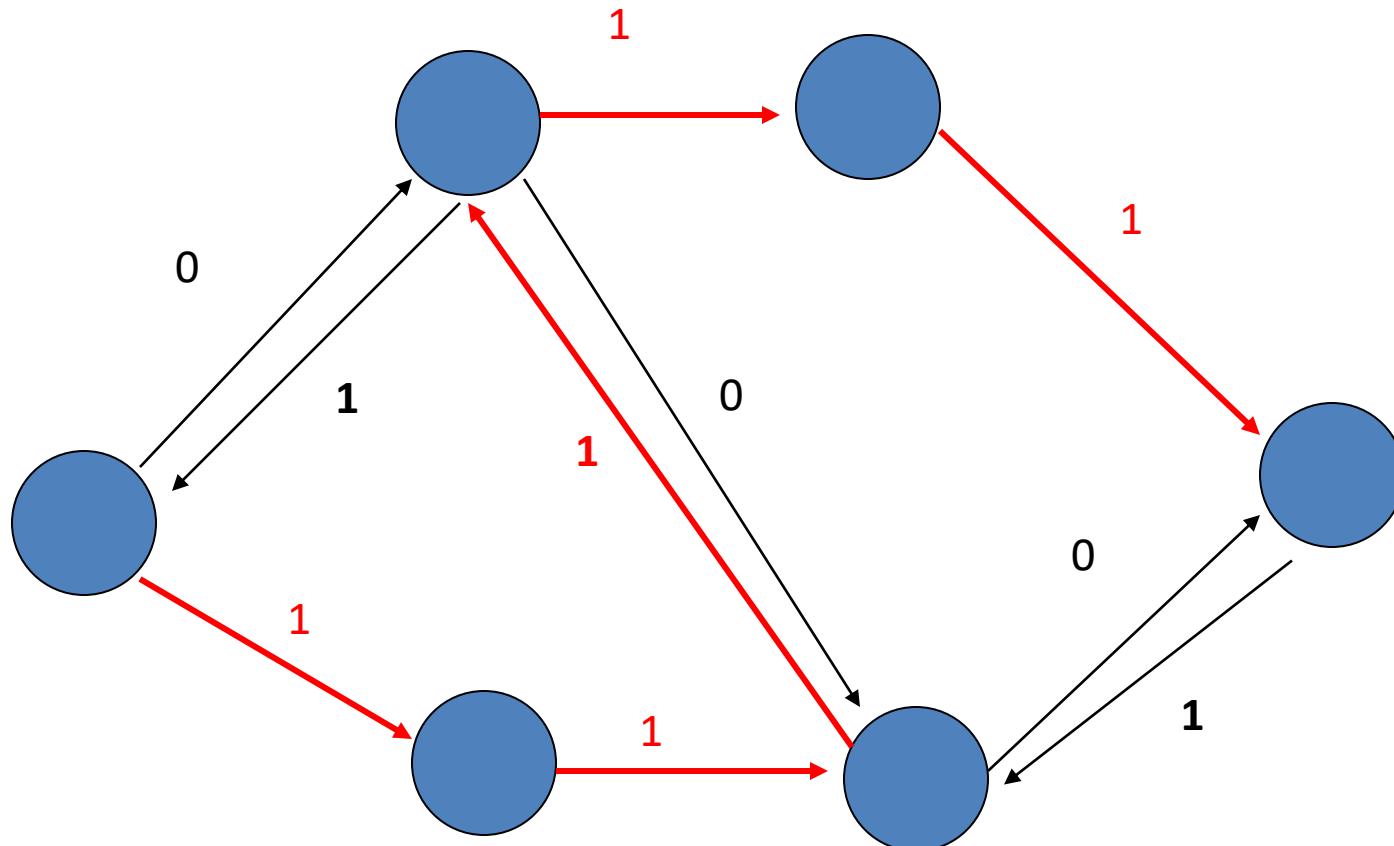
Example

- Residual network gives a chance to “circle back”



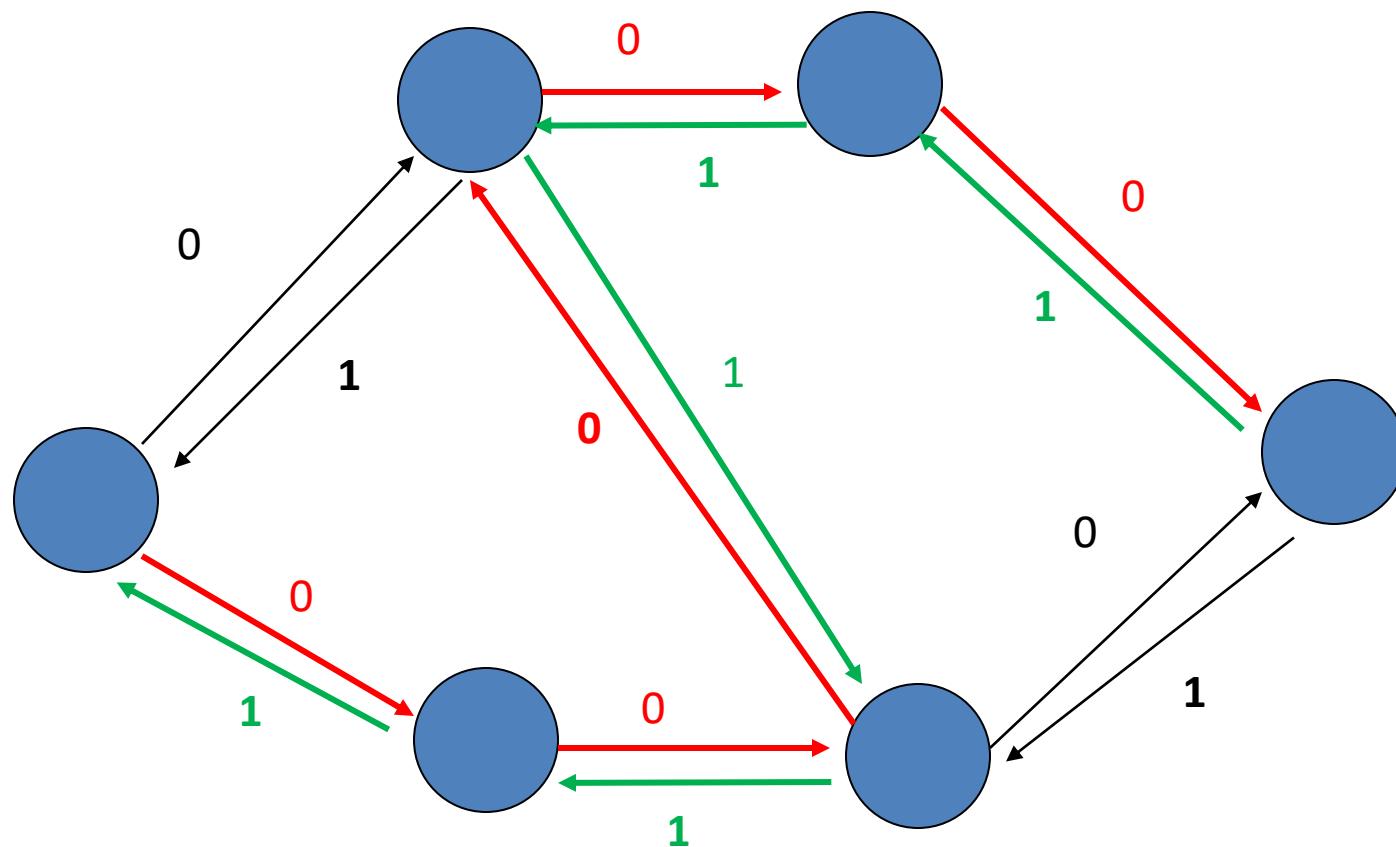
Example

- DFS augments another unit flow in the second iter



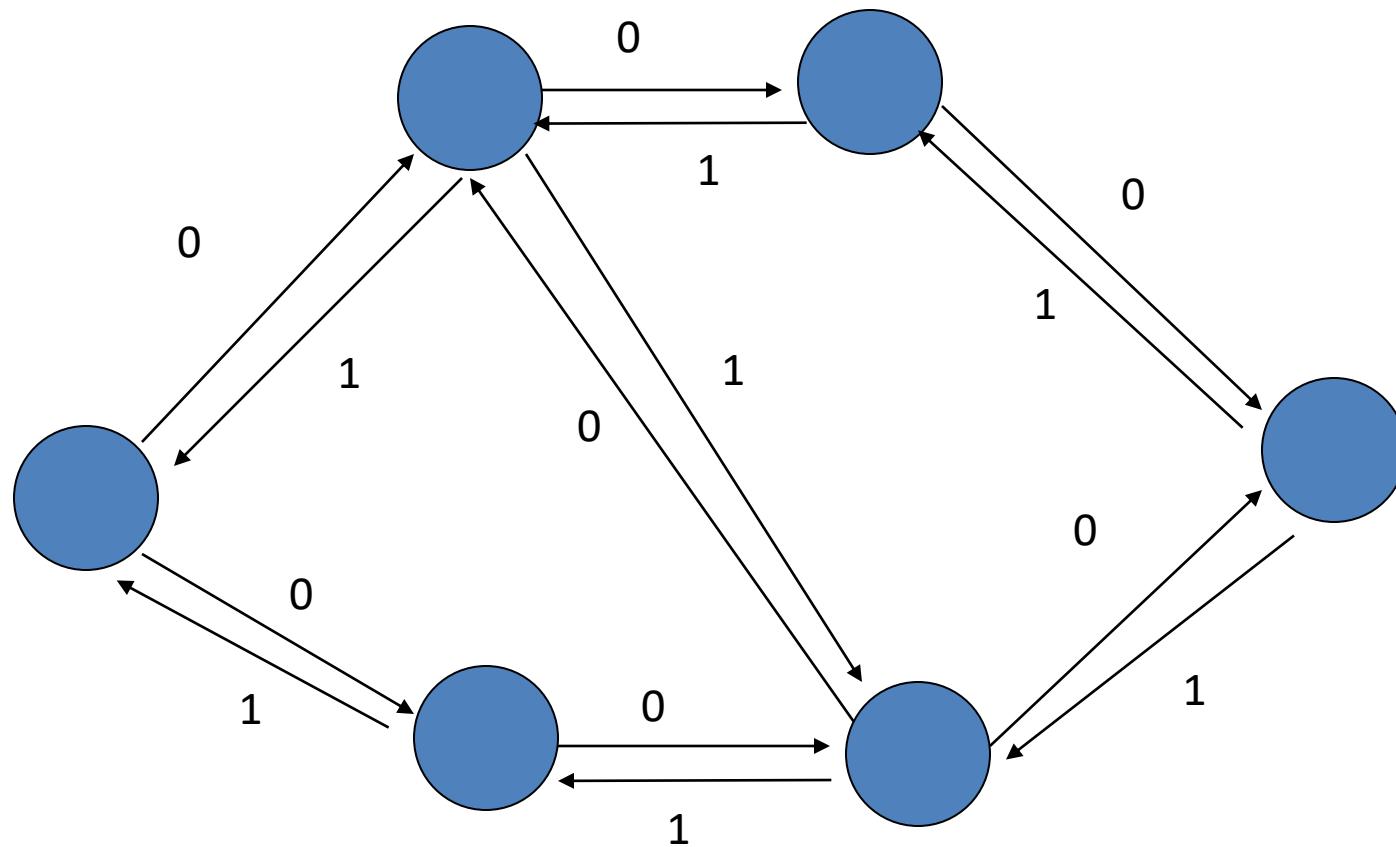
Example

□ Update residual network



Example

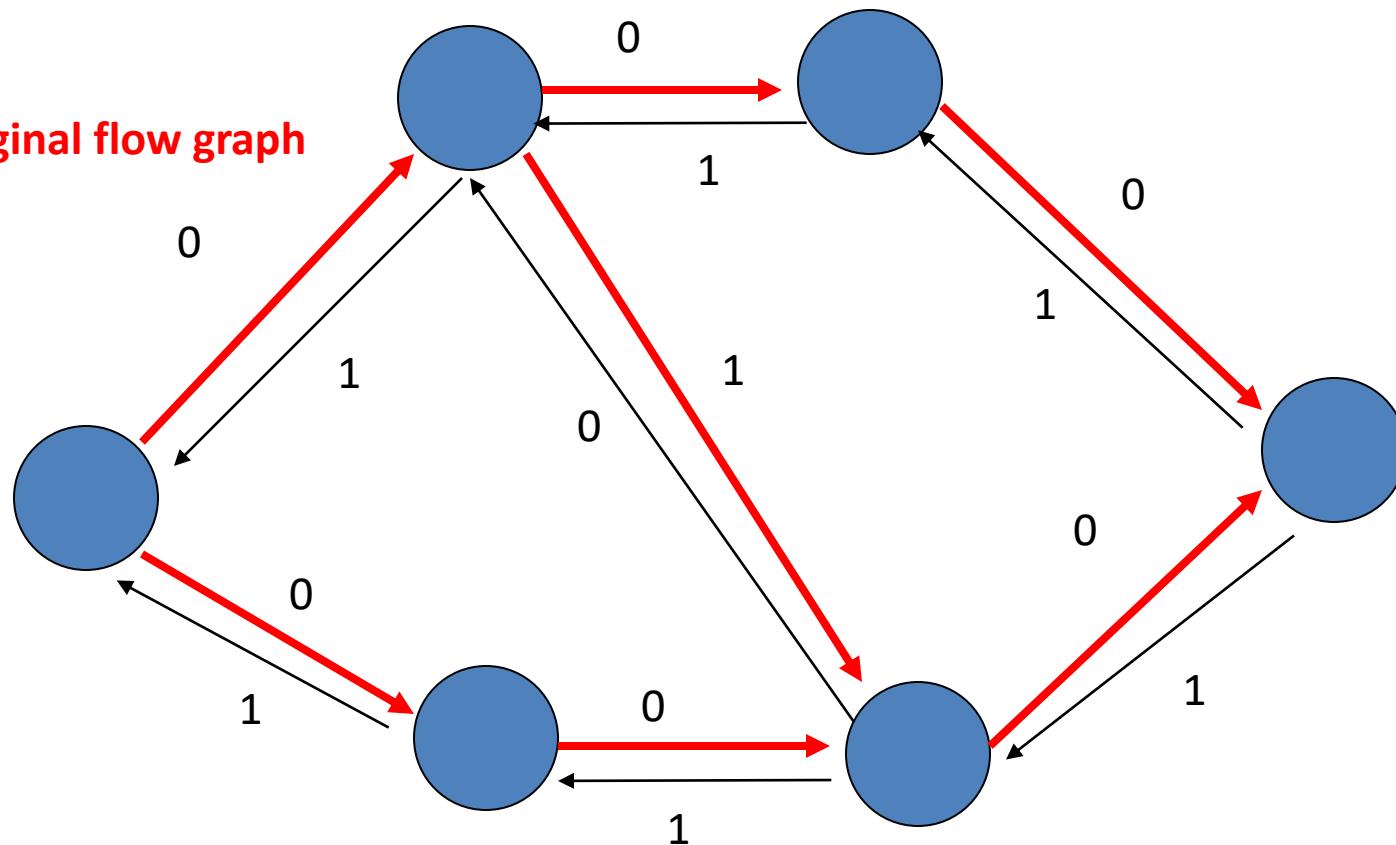
□ Maximum flow: 2



Example

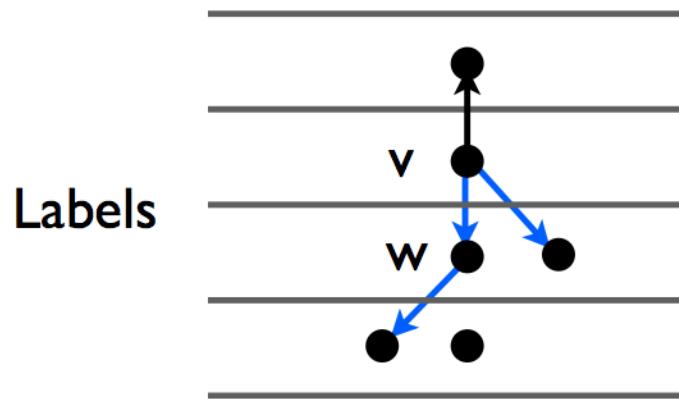
- Residual network gives us a way to circle flow back

The original flow graph



Push-Relabel Algorithm

- Main idea
 - Each node has a label which is a potential
 - Route flow from high to low height
- We augment a flow "downhill" (blue edge)



Push-Relabel Algorithm Terminology

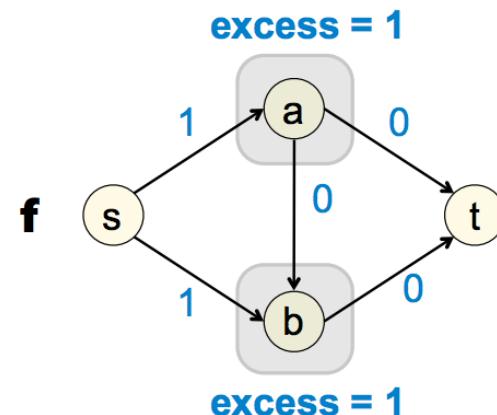
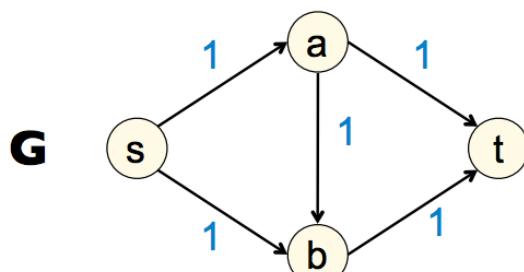
Preflow: A function $f: E \rightarrow R$ is a preflow if:

1. **Capacity Constraints:** $0 \leq f(e) \leq c(e)$
2. Instead of conservation constraints:

$$\sum_{e \text{ into } v} f(e) - \sum_{e \text{ out of } v} f(e) \geq 0$$

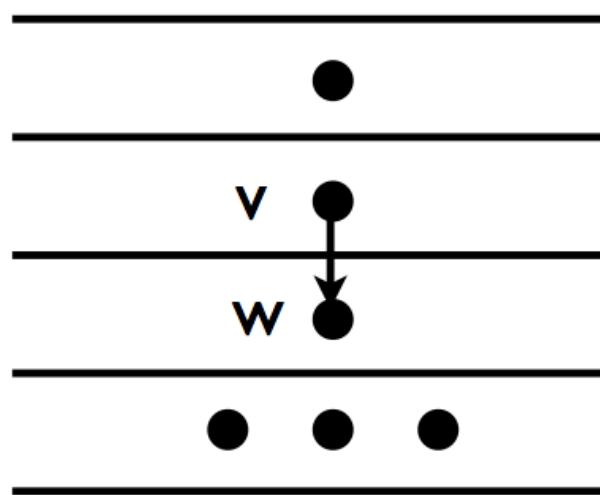
$$\text{Excess}(v) = \sum_{e \text{ into } v} f(e) - \sum_{e \text{ out of } v} f(e)$$

Example



Two Operations

- **Push(v, w): applies if $\text{excess}(v) > 0$, $h(w) < h(v)$**
 - $f = \min(\text{excess}(v), c_f(v, w))$
 - Add f to $f(v, w)$ and update residual network
- **Relabel(v): applies if $\text{excess}(v) > 0$ and no $h(w) < h(v)$**
 - Increase $h(v)$ by 1



Algorithm

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for all other v

Start with preflow f : $f(e) = c(e)$ for $e = (s, v), f(e) = 0$, for all other edges e

While there is a node (other than t) with positive excess

Pick a node v with $\text{excess}(v) > 0$

If there is an edge (v, w) in E_f such that $\text{push}(v, w)$ can be applied

$\text{Push}(v, w)$

Else

$\text{Relabel}(v)$

Push(v, w): Applies if $\text{excess}(v) > 0, h(w) < h(v), (v, w)$ in E_f

$q = \min(\text{excess}(v), c_f(v, w))$

Add q to $f(v, w)$

Relabel(v): Applies if $\text{excess}(v) > 0$, for all w s.t (v, w) in $E_f, h(w) \geq h(v)$

Increase $h(v)$ by 1

Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

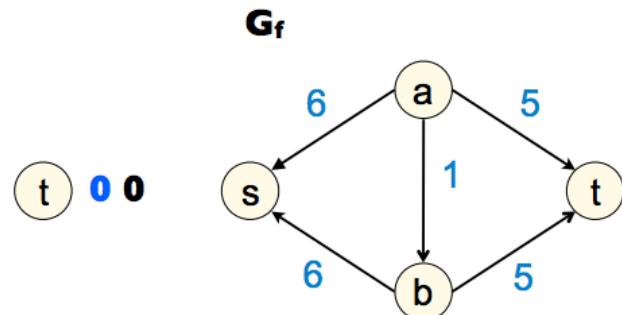
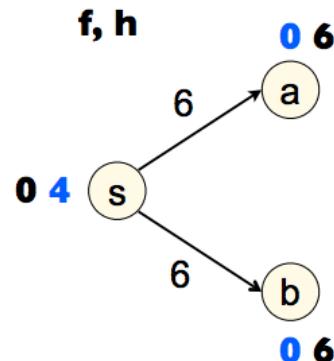
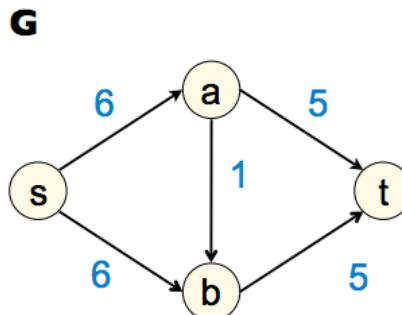
While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all
 w s.t (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1



Labels

Excesses

Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

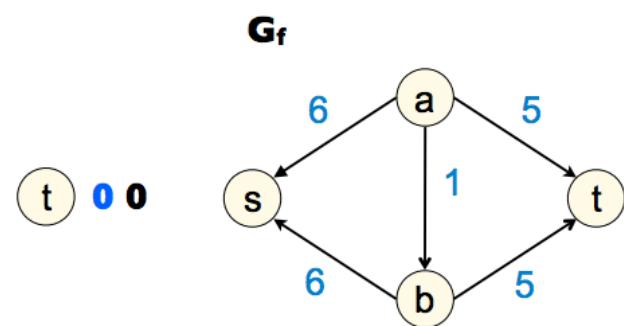
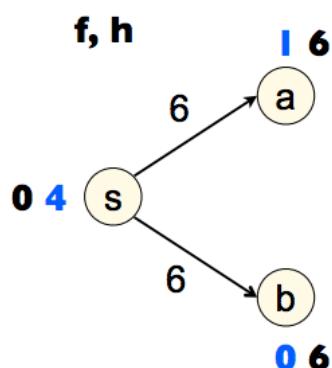
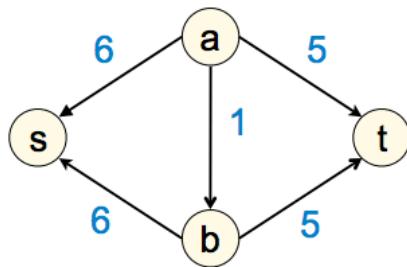
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all
 w s.t. (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)



Labels

Excesses

Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow f : $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

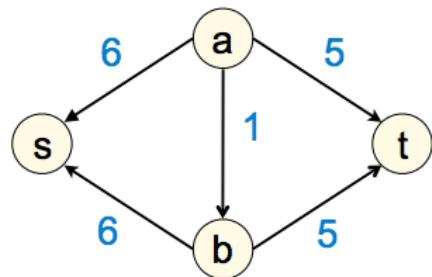
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

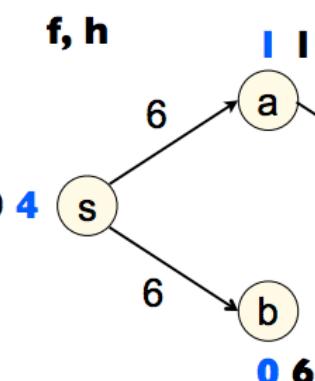
Applies if $\text{excess}(v) > 0$ and for all w s.t (v, w) in E_f , $h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)

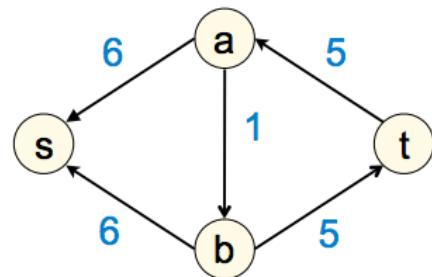


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
Pick a node v with $\text{excess}(v) > 0$
If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
Else
 Relabel(v)

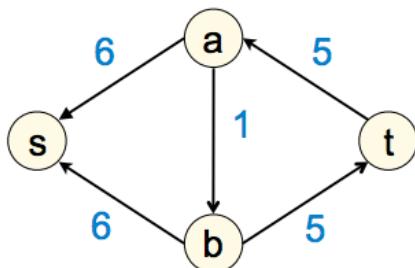
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
Add q to $f(v, w)$

Relabel(v):

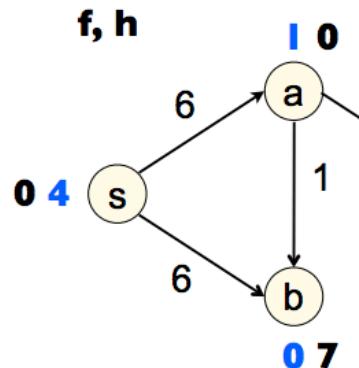
Applies if $\text{excess}(v) > 0$ and for all w s.t (v, w) in E_f , $h(w) \geq h(v)$
Increase $h(v)$ by 1

G_f (before)

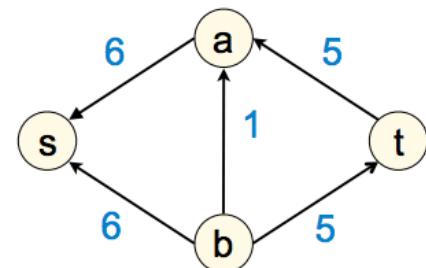


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

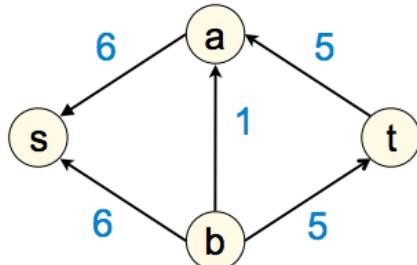
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all
 w s.t. (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

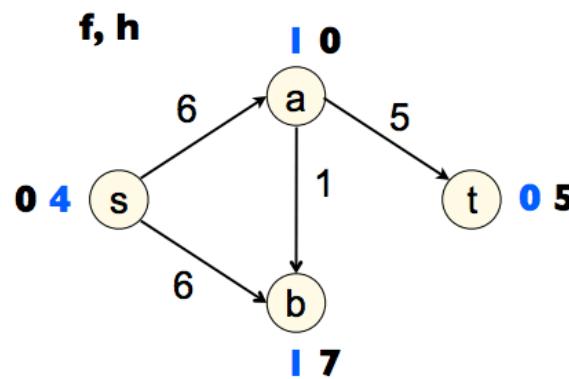
G_f (before)



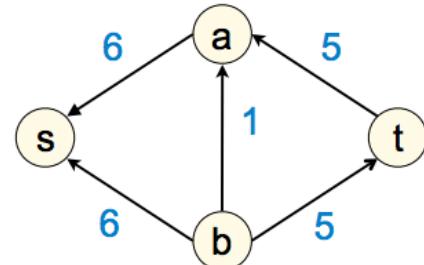
Labels

Excesses

f, h



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

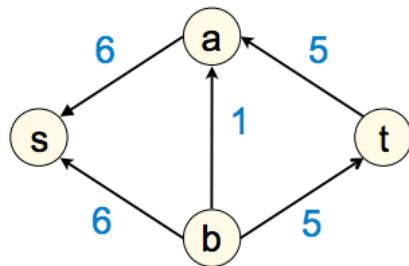
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
Add q to $f(v, w)$

Relabel(v):

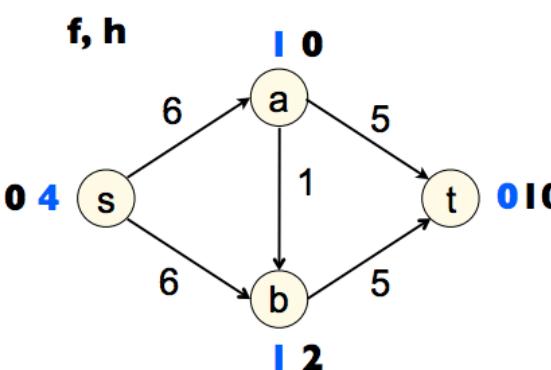
Applies if $\text{excess}(v) > 0$ and for all w s.t. (v, w) in $E_f, h(w) \geq h(v)$
Increase $h(v)$ by 1

G_f (before)

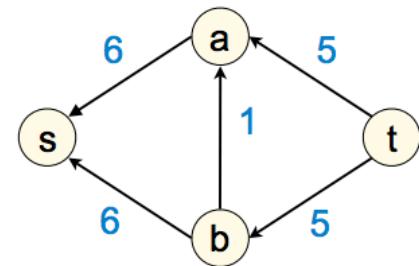


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s. t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

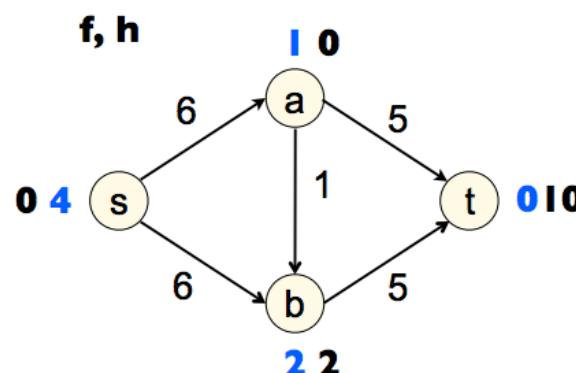
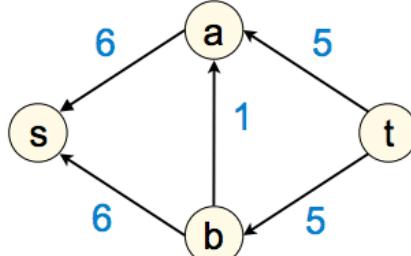
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

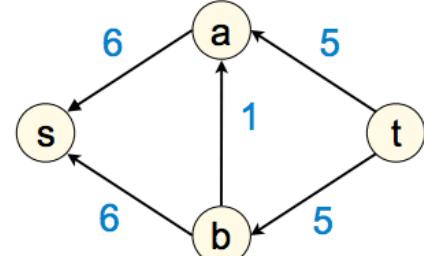
Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all
 w s.t (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)



G_f



Labels

Excesses

Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

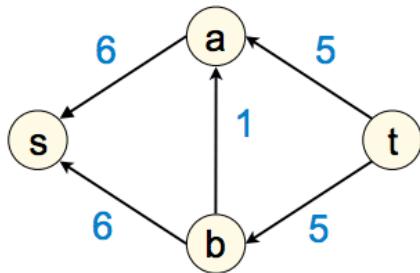
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all w s.t. (v, w) in E_f , $h(w) \geq h(v)$
 Increase $h(v)$ by 1

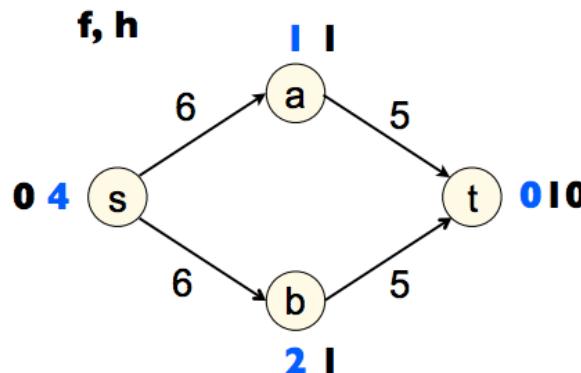
G_f (before)



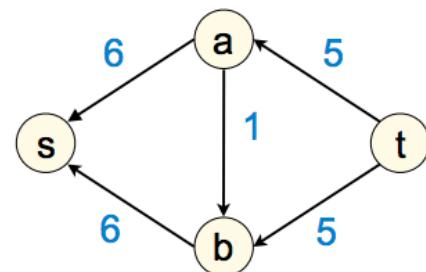
Labels

Excesses

f, h



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

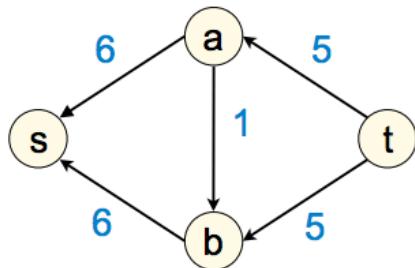
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

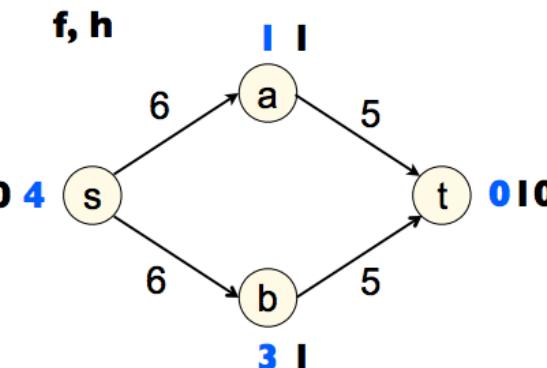
Applies if $\text{excess}(v) > 0$ and for all
 w s.t. (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)

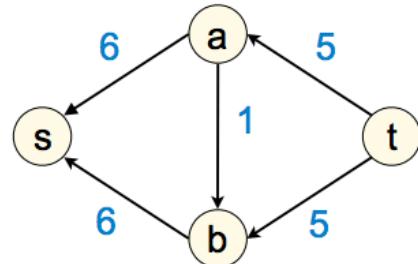


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

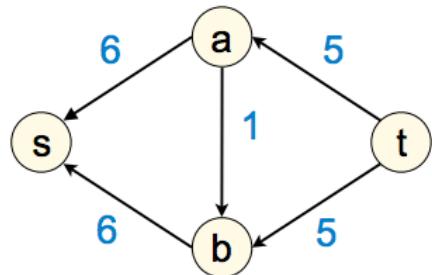
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

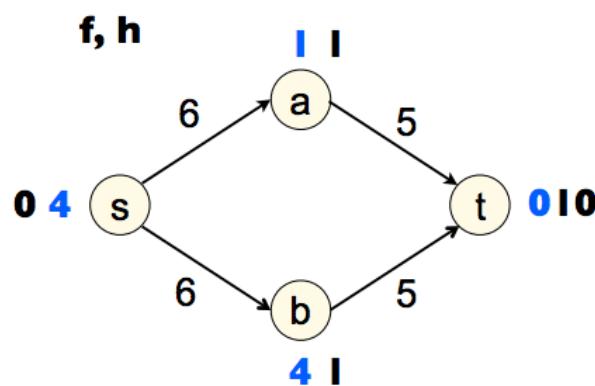
Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all
 w s.t (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

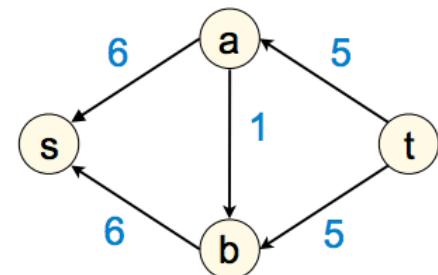
G_f (before)



f, h



G_f



Labels

Excesses

Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f : $f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

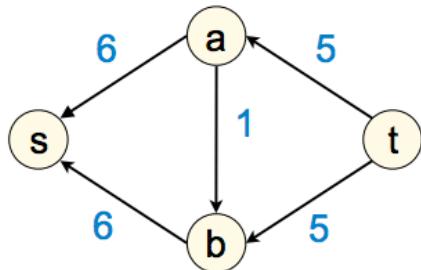
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
Add q to $f(v, w)$

Relabel(v):

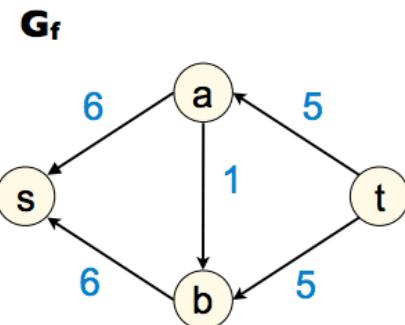
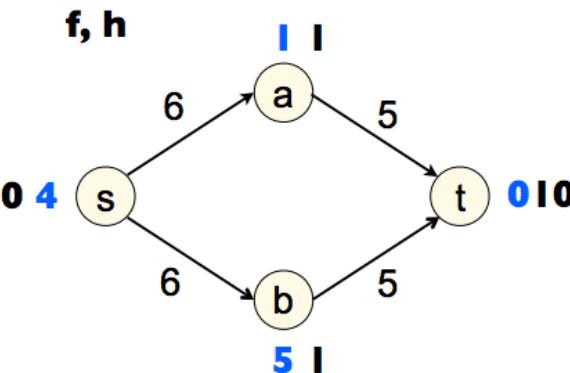
Applies if $\text{excess}(v) > 0$ and for all w s.t. (v, w) in E_f , $h(w) \geq h(v)$
Increase $h(v)$ by 1

G_f (before)



Labels

Excesses



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

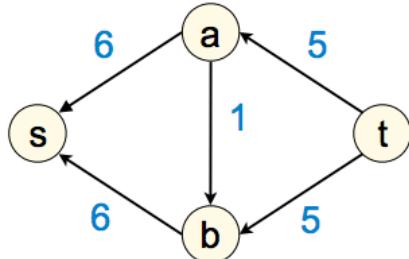
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all w s.t (v, w) in E_f , $h(w) \geq h(v)$
 Increase $h(v)$ by 1

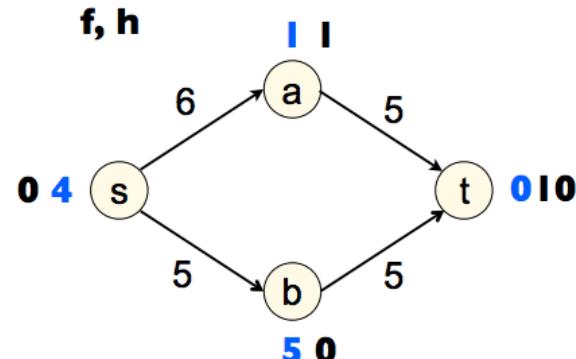
G_f (before)



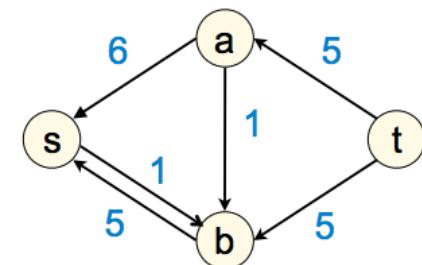
Labels

Excesses

f, h



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

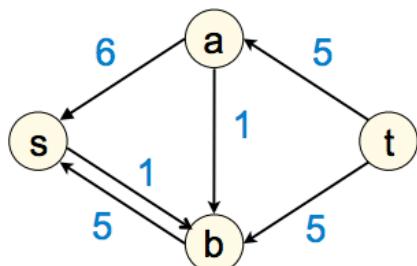
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

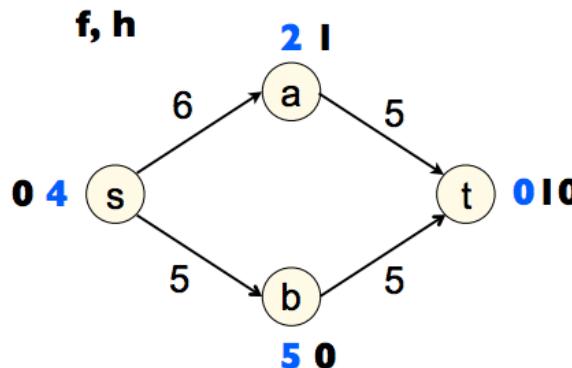
Applies if $\text{excess}(v) > 0$ and for all
 w s.t (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)

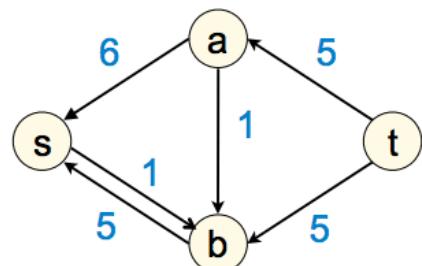


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s. t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

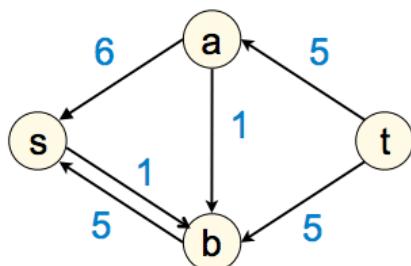
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
Add q to $f(v, w)$

Relabel(v):

Applies if $\text{excess}(v) > 0$ and for all
 w s.t (v, w) in $E_f, h(w) \geq h(v)$
Increase $h(v)$ by 1

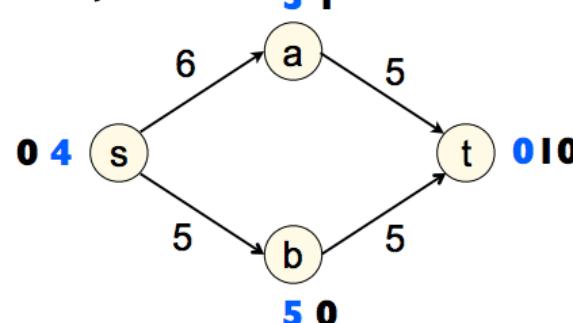
G_f (before)



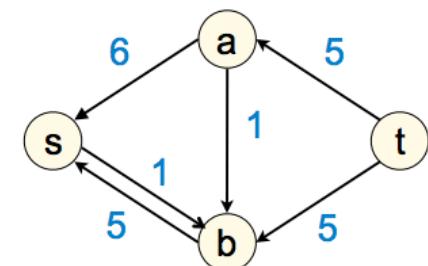
Labels

Excesses

f, h



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

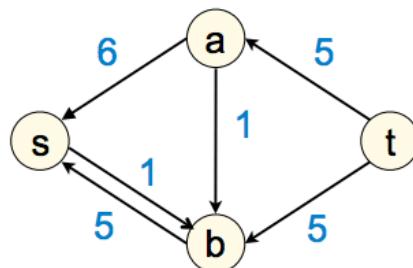
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

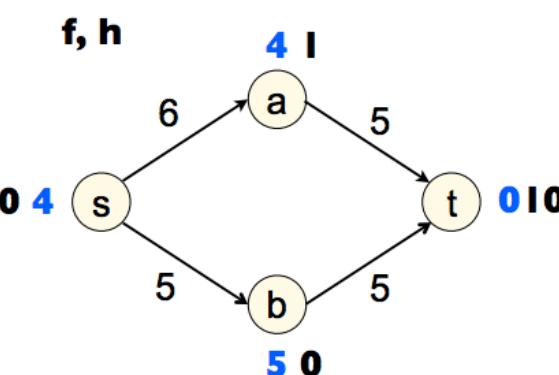
Applies if $\text{excess}(v) > 0$ and for all
 w s.t. (v, w) in $E_f, h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)

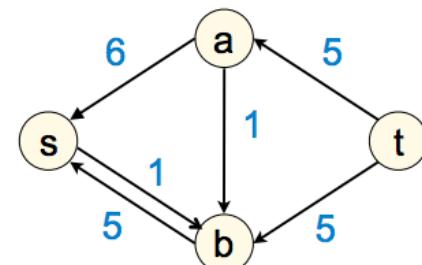


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess
 Pick a node v with $\text{excess}(v) > 0$
 If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies
 Push(v, w)
 Else
 Relabel(v)

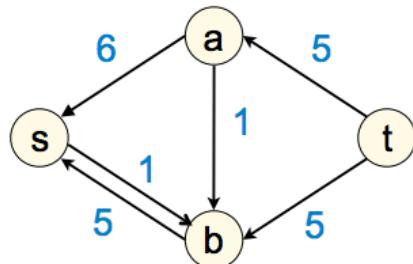
Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$
 $q = \min(\text{excess}(v), c_f(v, w))$
 Add q to $f(v, w)$

Relabel(v):

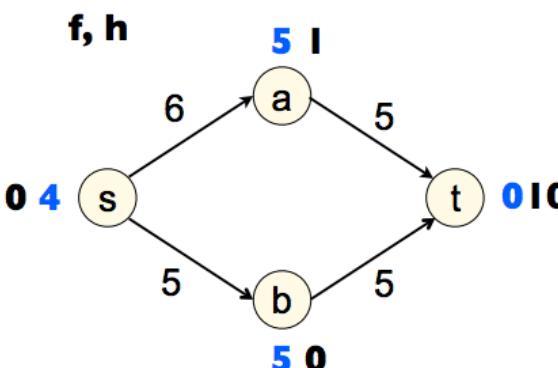
Applies if $\text{excess}(v) > 0$ and for all w s.t. (v, w) in E_f , $h(w) \geq h(v)$
 Increase $h(v)$ by 1

G_f (before)

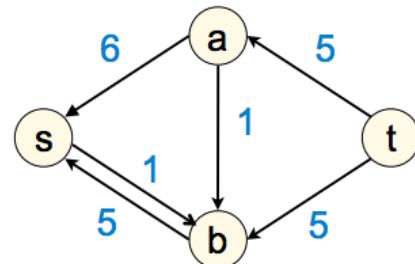


Labels

Excesses



G_f



Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
 Start with preflow $f: f(e) = c(e)$ for $e = (s, v), f(e) = 0$, otherwise

While there is a node (other than t) with positive excess

Pick a node v with $\text{excess}(v) > 0$

If there is an edge (v, w) in E_f s.t. $\text{push}(v, w)$ applies

Push(v, w)

Else

Relabel(v)

Push(v, w):

Applies if $\text{excess}(v) > 0, h(w) < h(v)$

$$q = \min(\text{excess}(v), c_f(v, w))$$

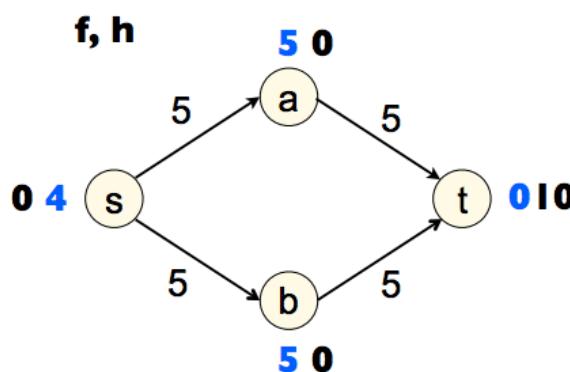
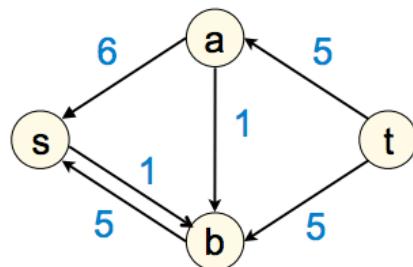
Add q to $f(v, w)$

Relabel(v):

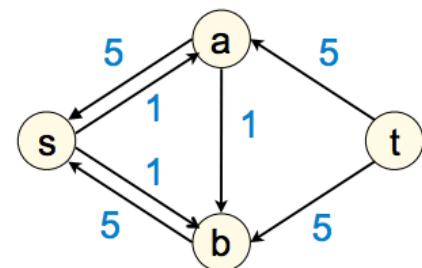
Applies if $\text{excess}(v) > 0$ and for all w s.t (v, w) in E_f , $h(w) \geq h(v)$

Increase $h(v)$ by 1

G_f (before)



G_f



Labels

Excesses

Time Complexity

- How many push/relabel operations?
 - Bound the maximum value of $h(v)$ for any node v
 - Bound the number of relabel operations

Min-cost Flow

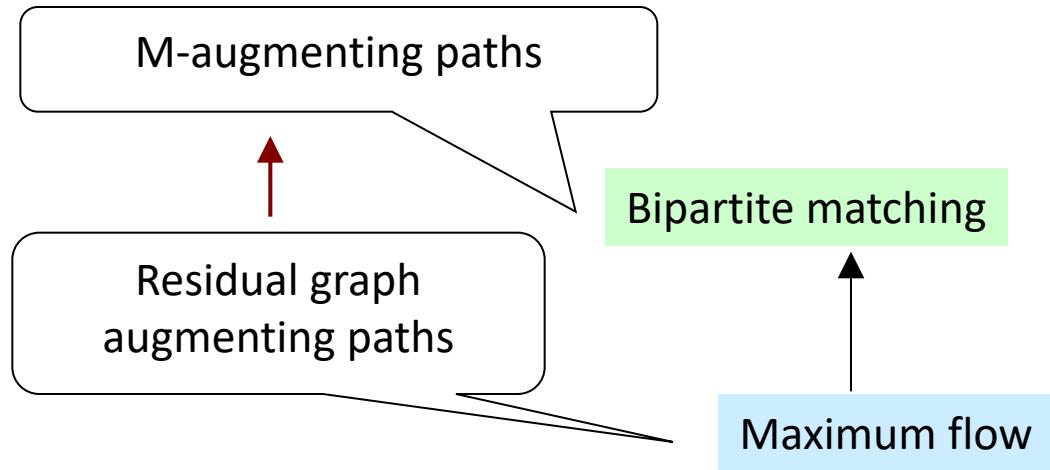
Goal: Build a cheapest network that satisfies the flow constraint

Input:

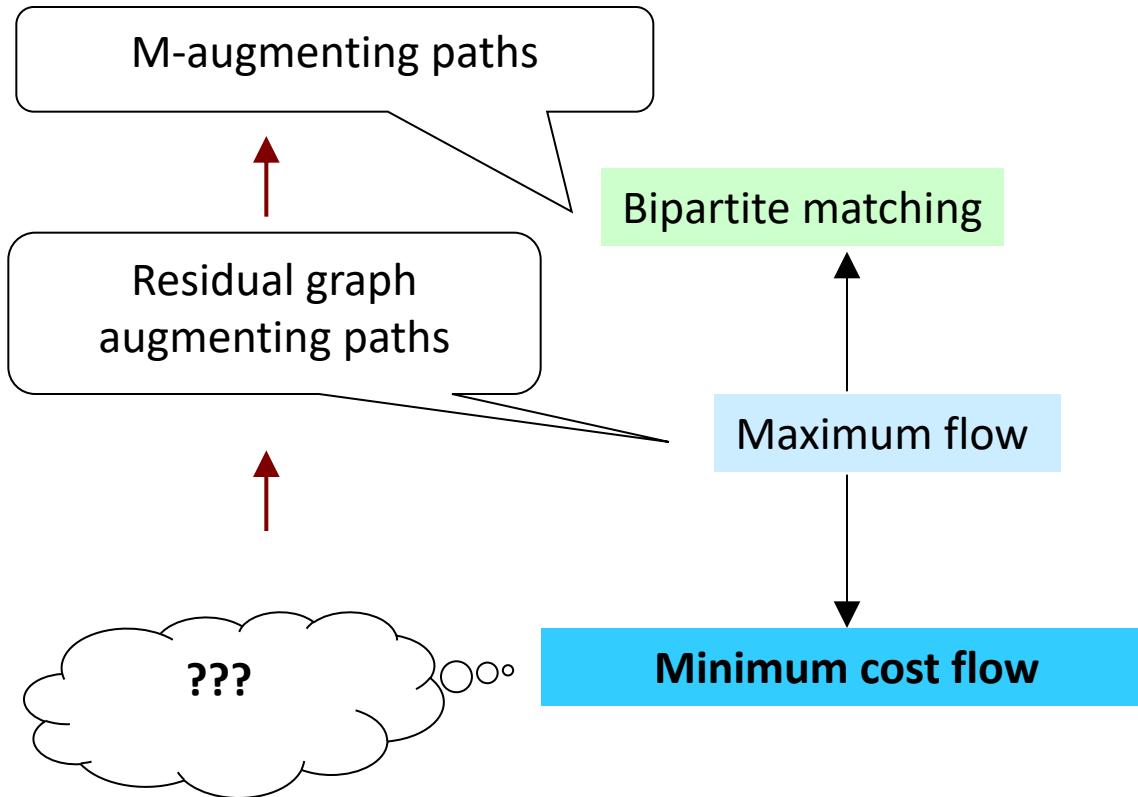
- ➊ A directed graph G
- ➋ A source vertex s
- ➌ A sink vertex t
- ➍ A capacity function c on the edges, i.e. $c: E \rightarrow \mathbb{R}$
- ➎ A cost function w on the edges, i.e. $w: E \rightarrow \mathbb{R}$

Output: a **maximum** s - t flow f which **minimizes** $\sum f(e) w(e)$

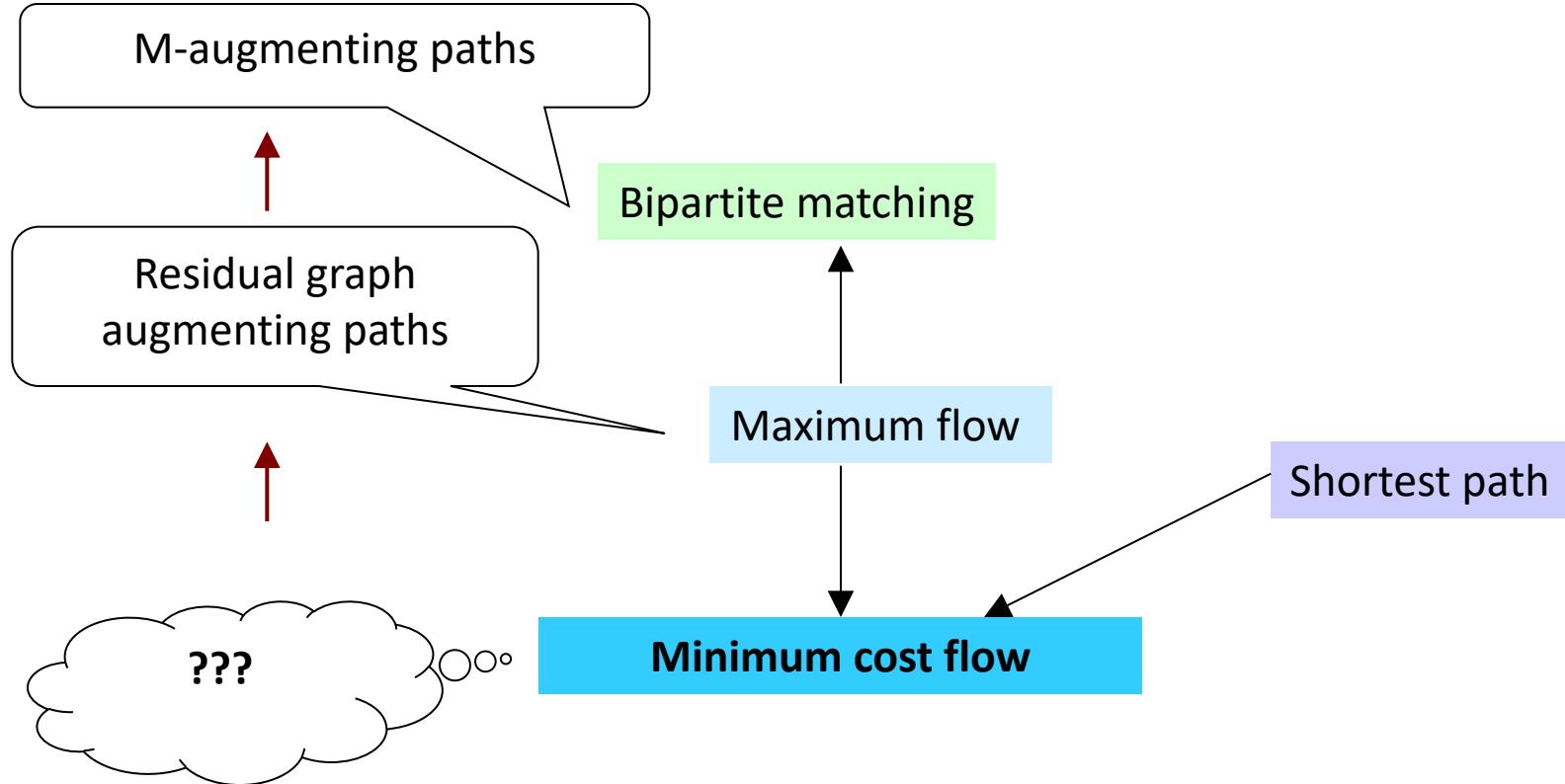
Problem Structure



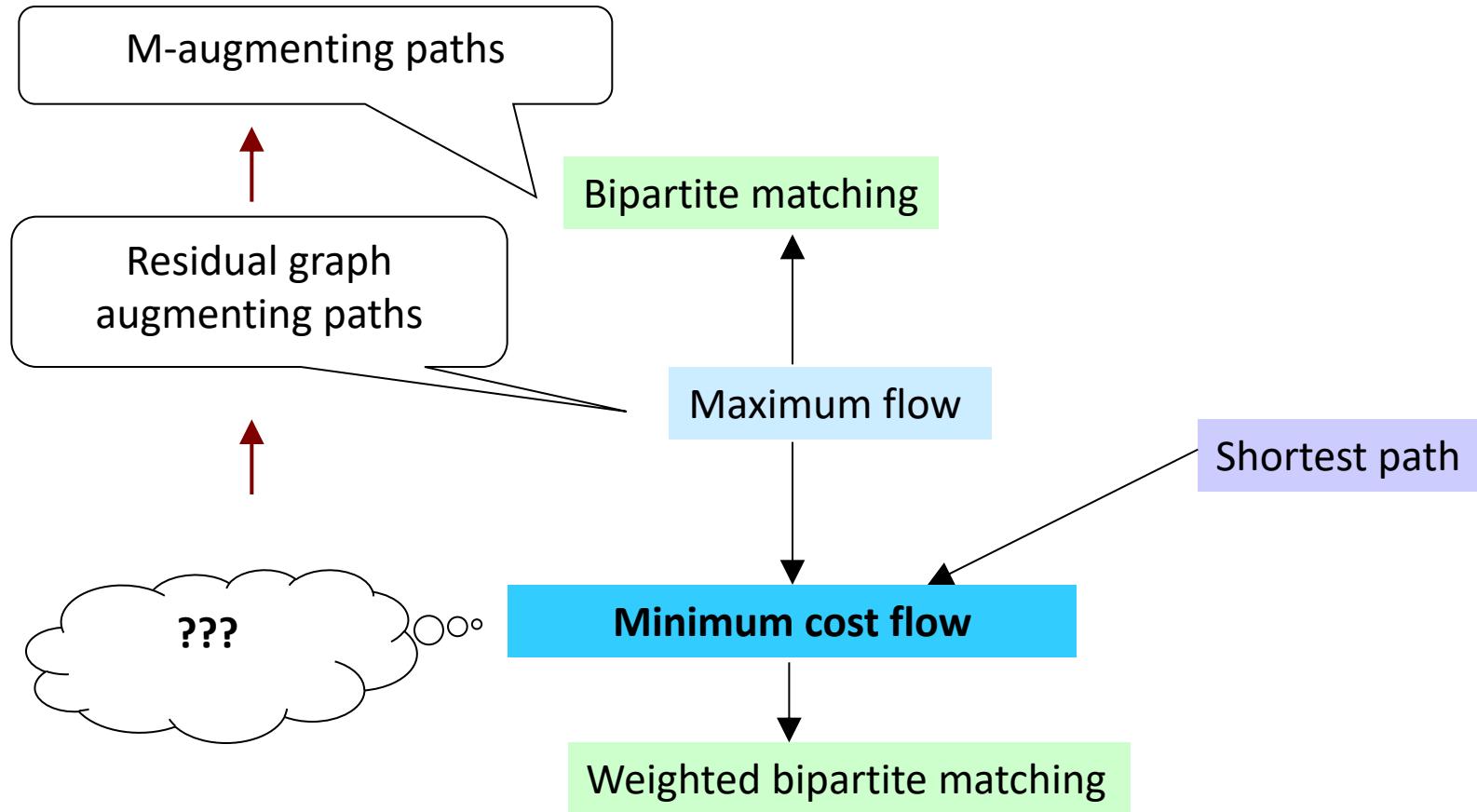
Problem Structure



Problem Structure



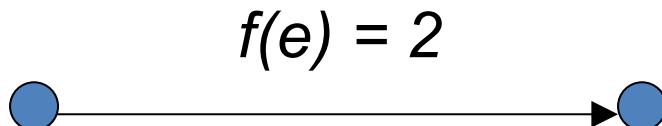
Problem Structure



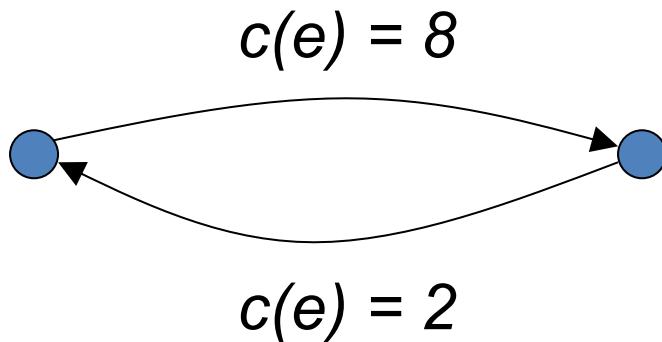
Algorithm?

- Two parameters: **value** and **cost** of the flow
 - Each edge is associated with two values c and w
 - c is the capacity and w is the cost per unit flow
- What is the residual network structure

Residual Network

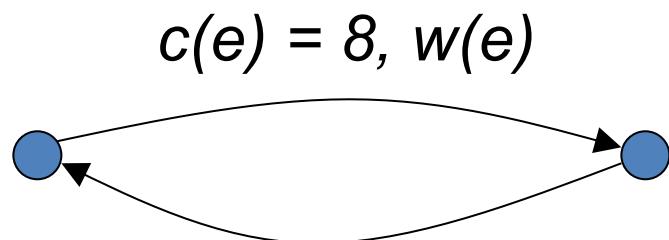
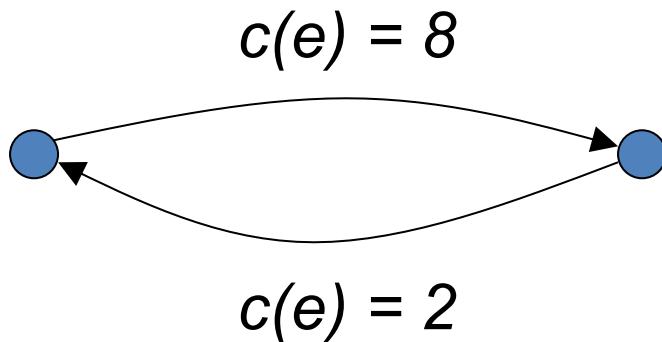
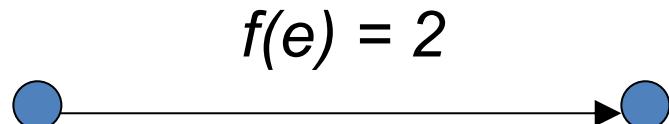
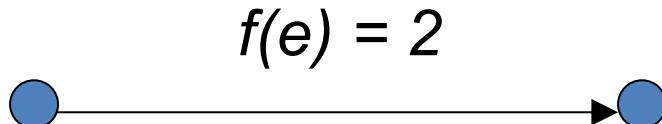


$$c(e) = 10$$



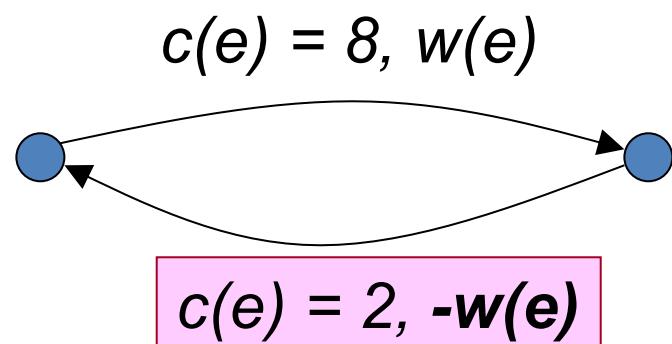
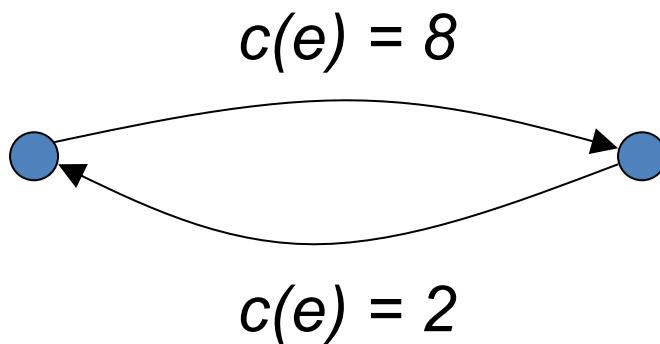
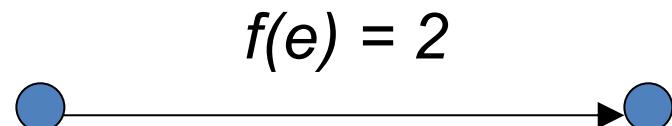
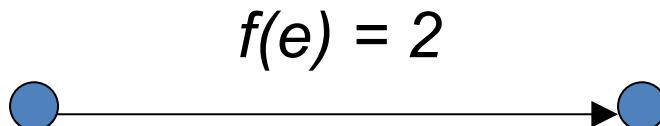
Max Flow

Residual Network



Max Flow

Residual Network

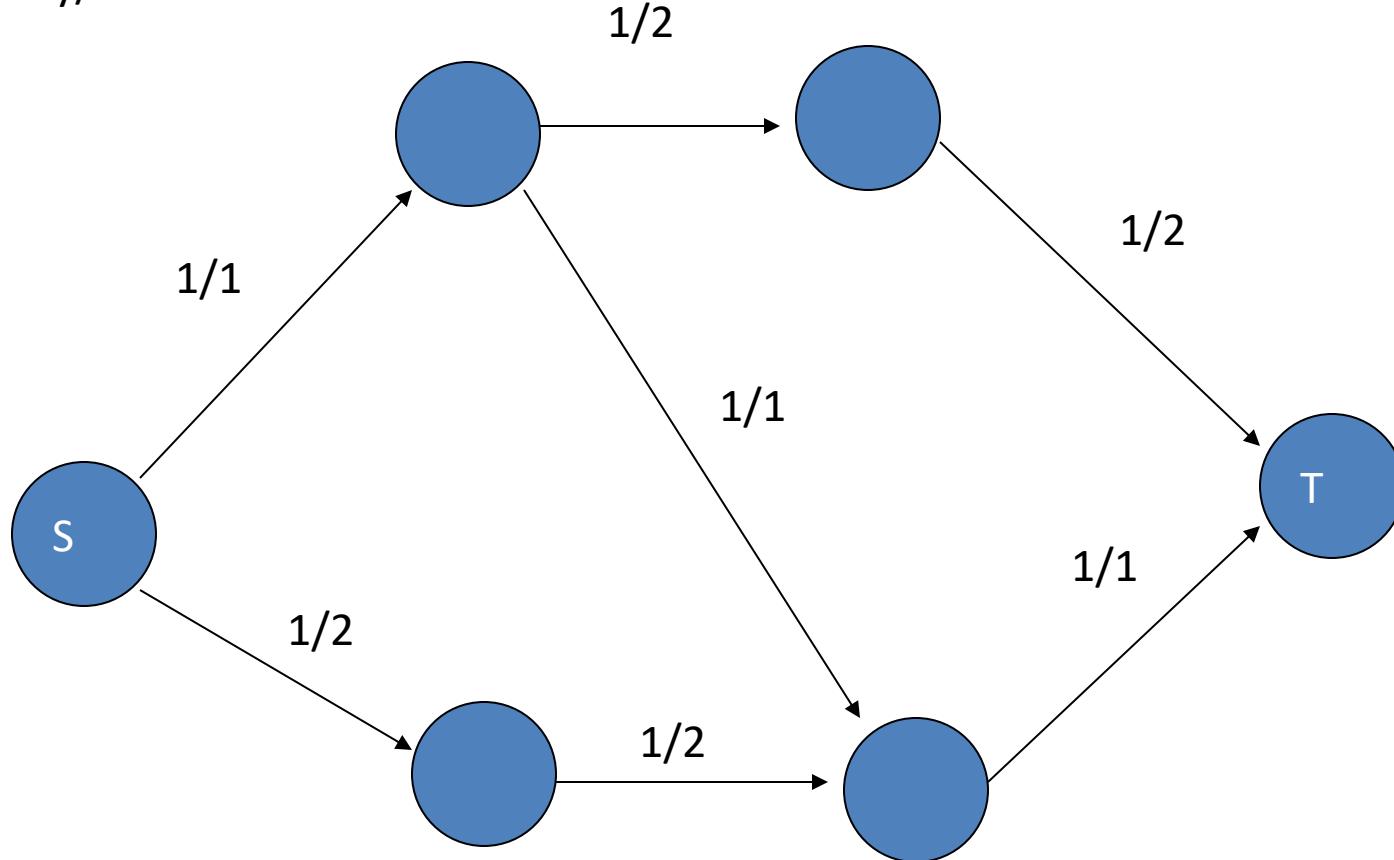


Max Flow

Min-cost Flow

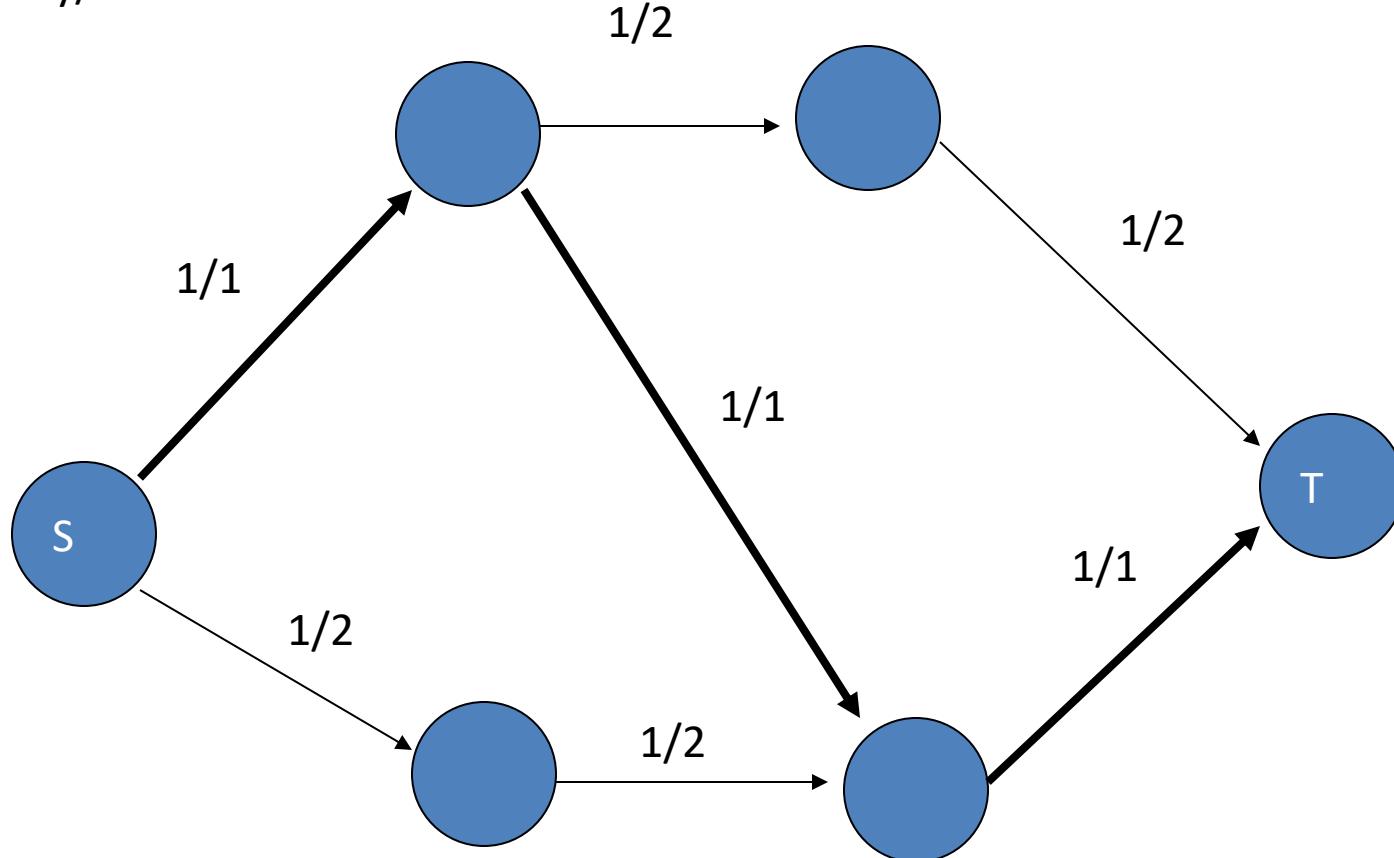
Example

Capacity/Cost



Example

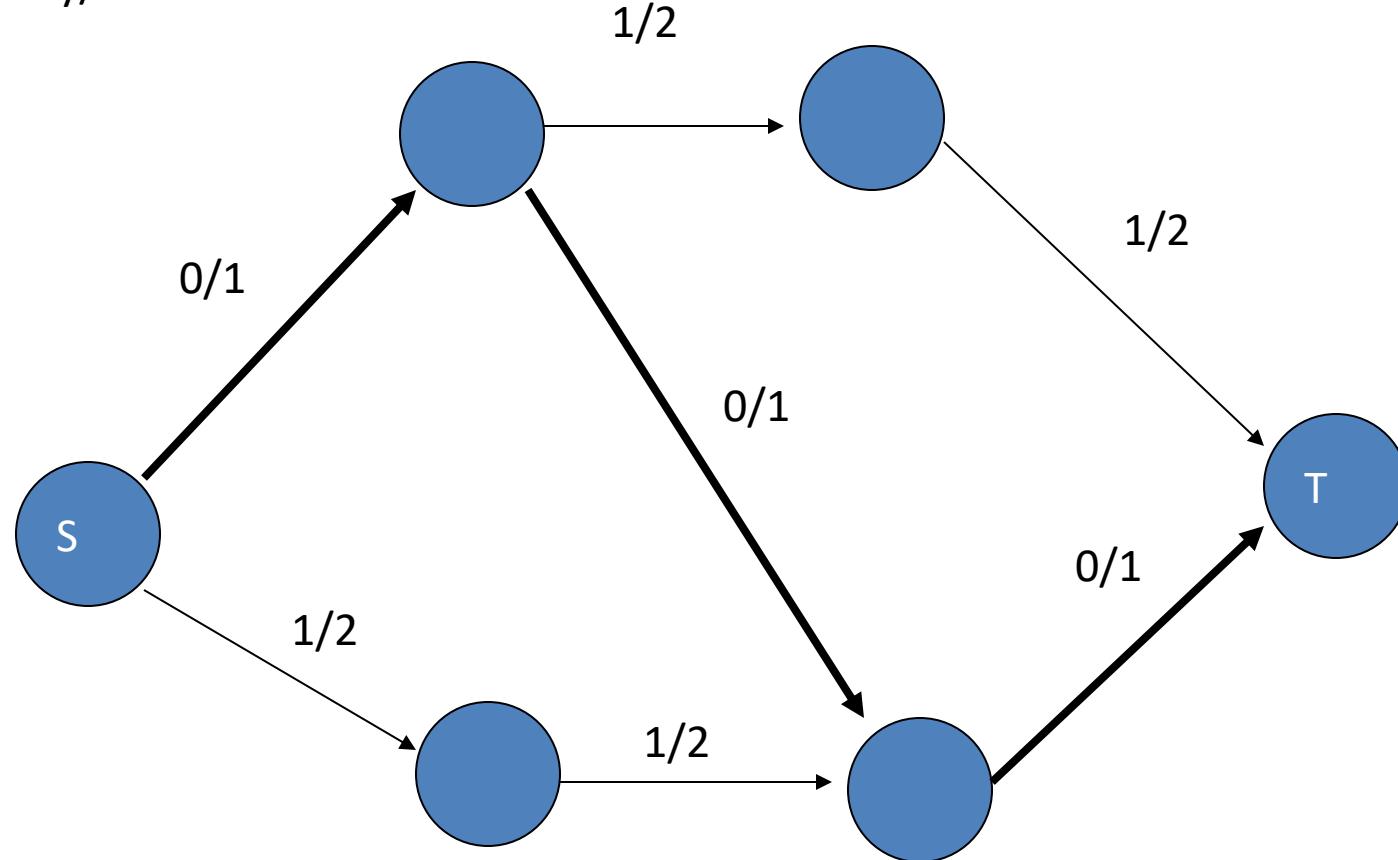
Capacity/Cost



Find the shortest path

Example

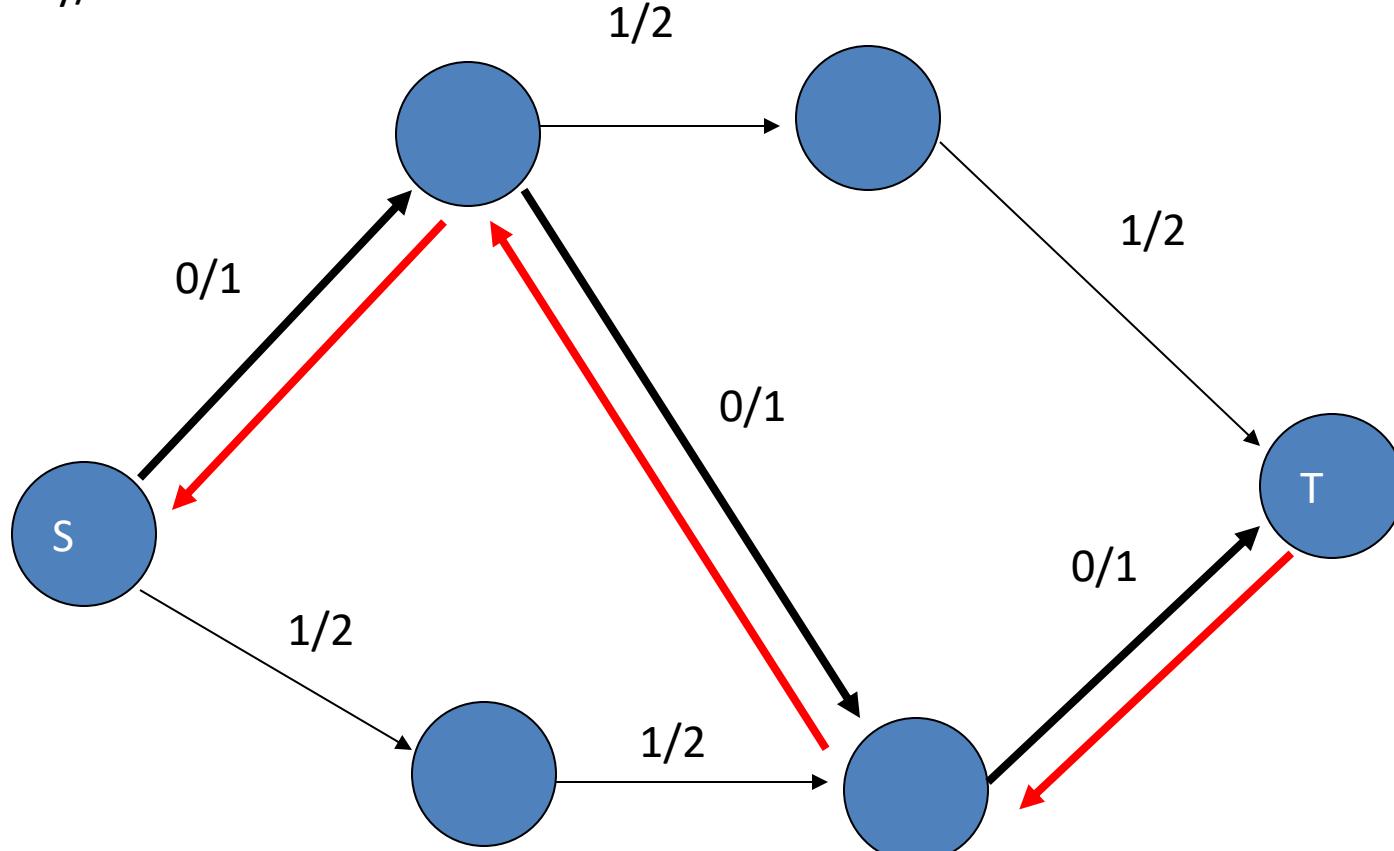
Capacity/Cost



Flow=1, cost=3

Example

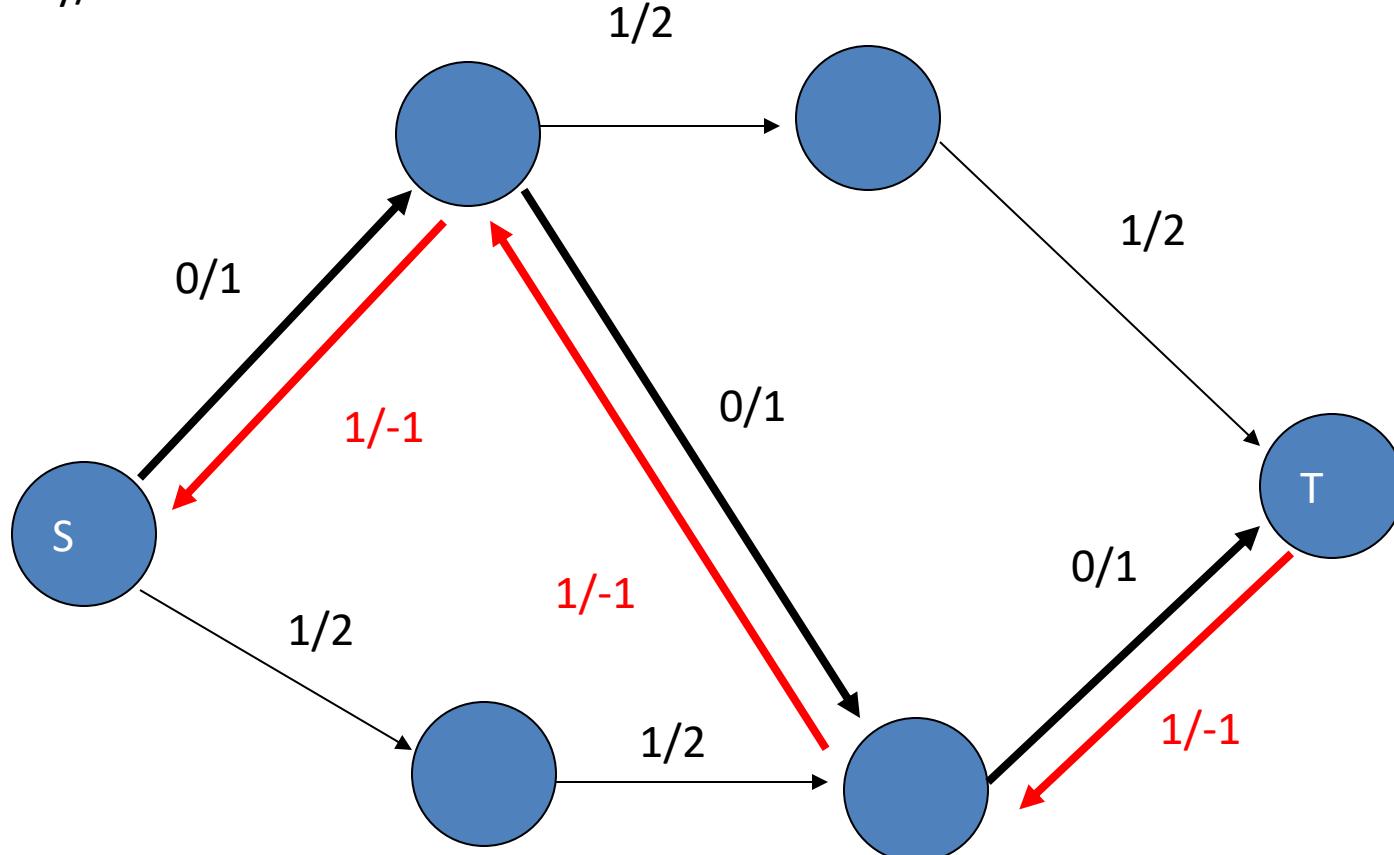
Capacity/Cost



Flow=1, cost=3

Example

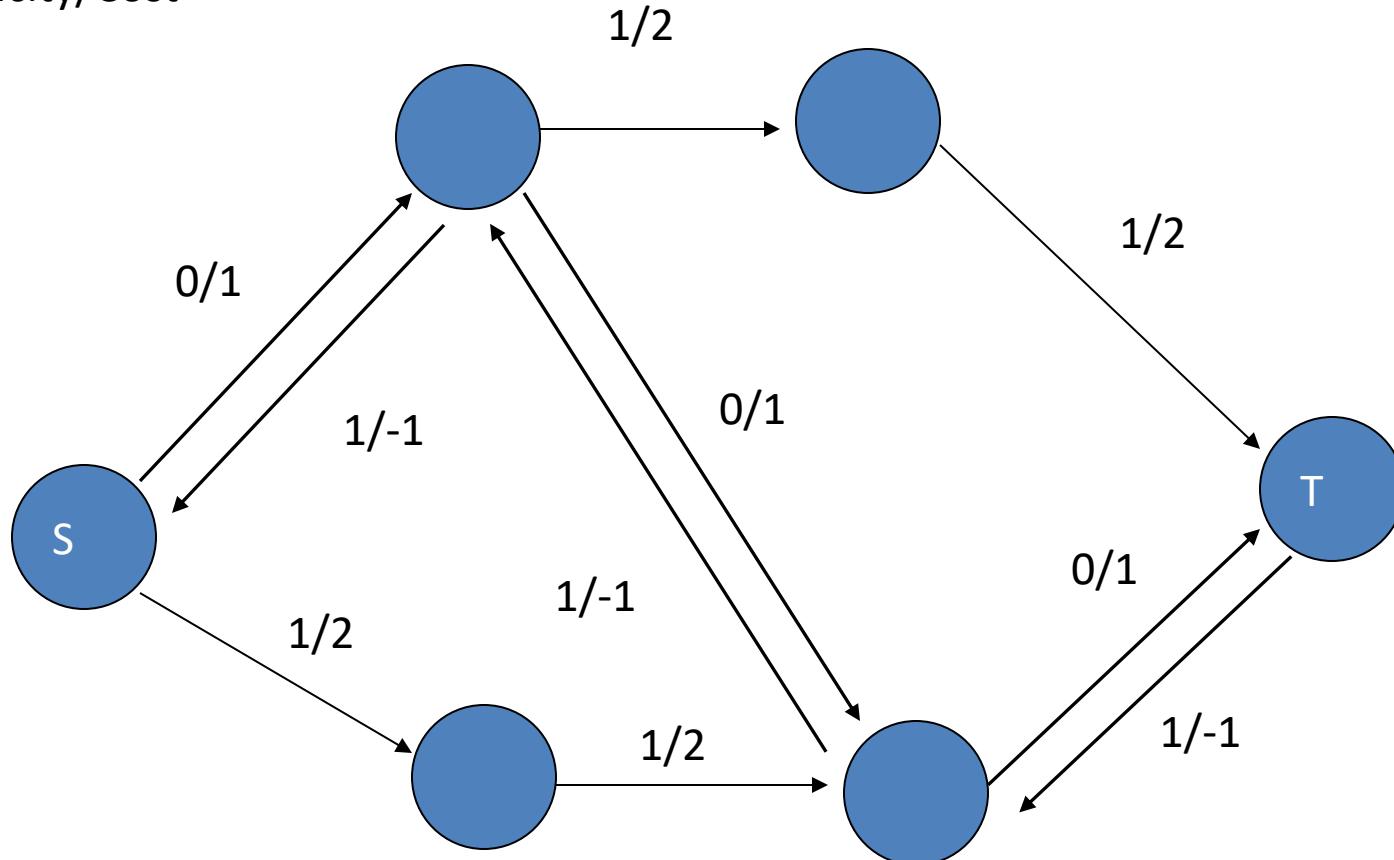
Capacity/Cost



Flow=1, cost=3

Example

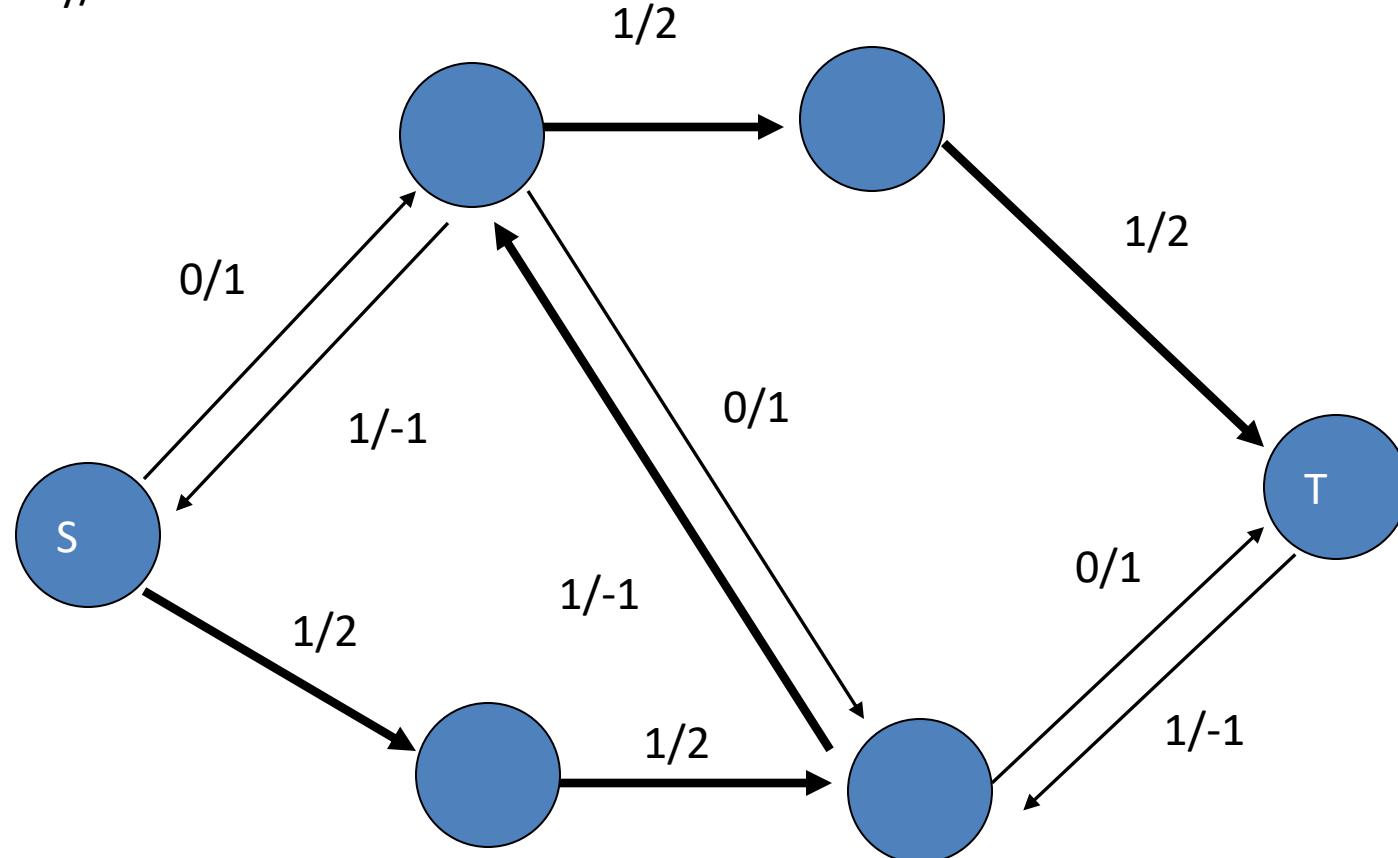
Capacity/Cost



Find another shortest path

Example

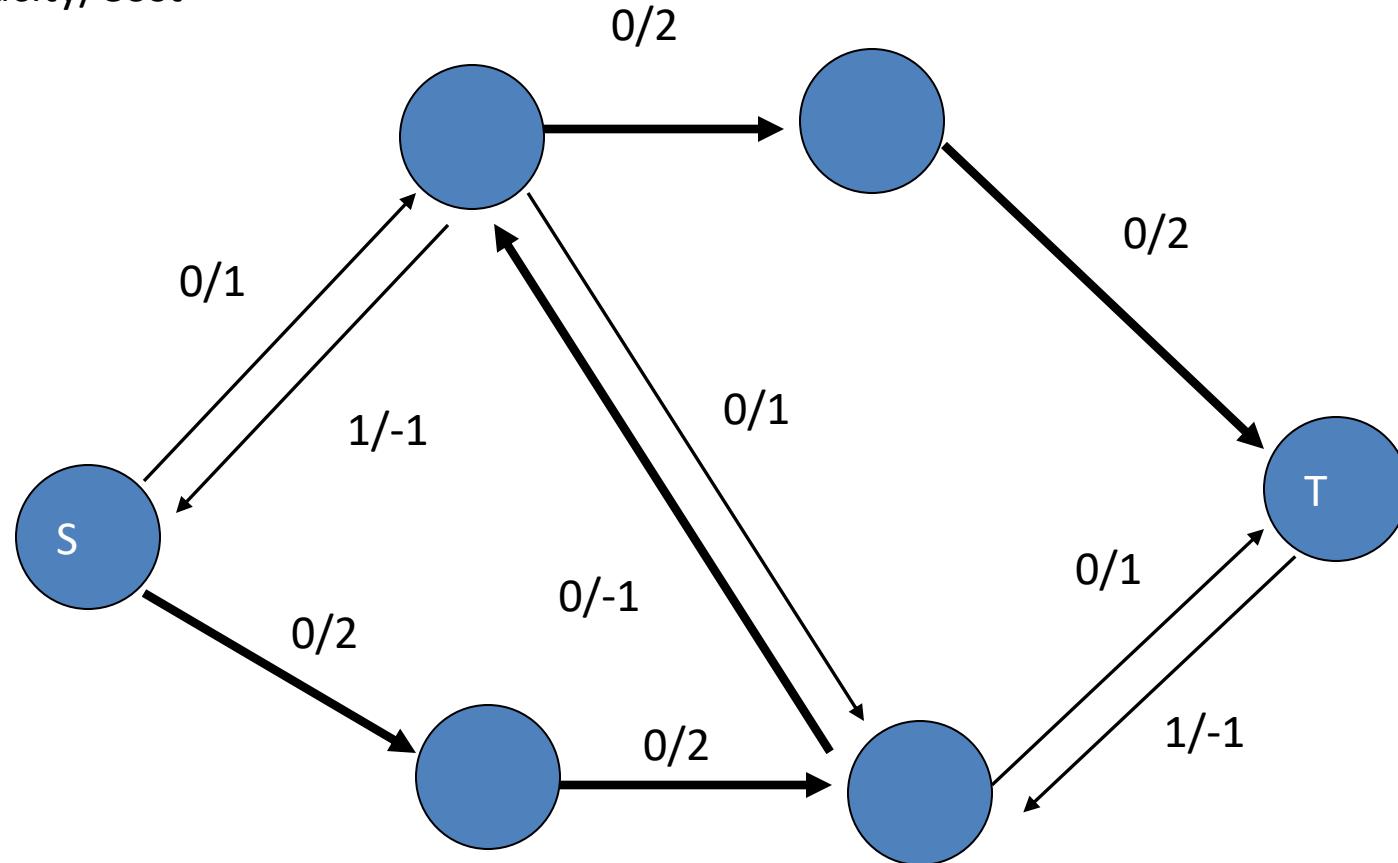
Capacity/Cost



Flow=1, cost = $2+2-1+2+2=7$

Example

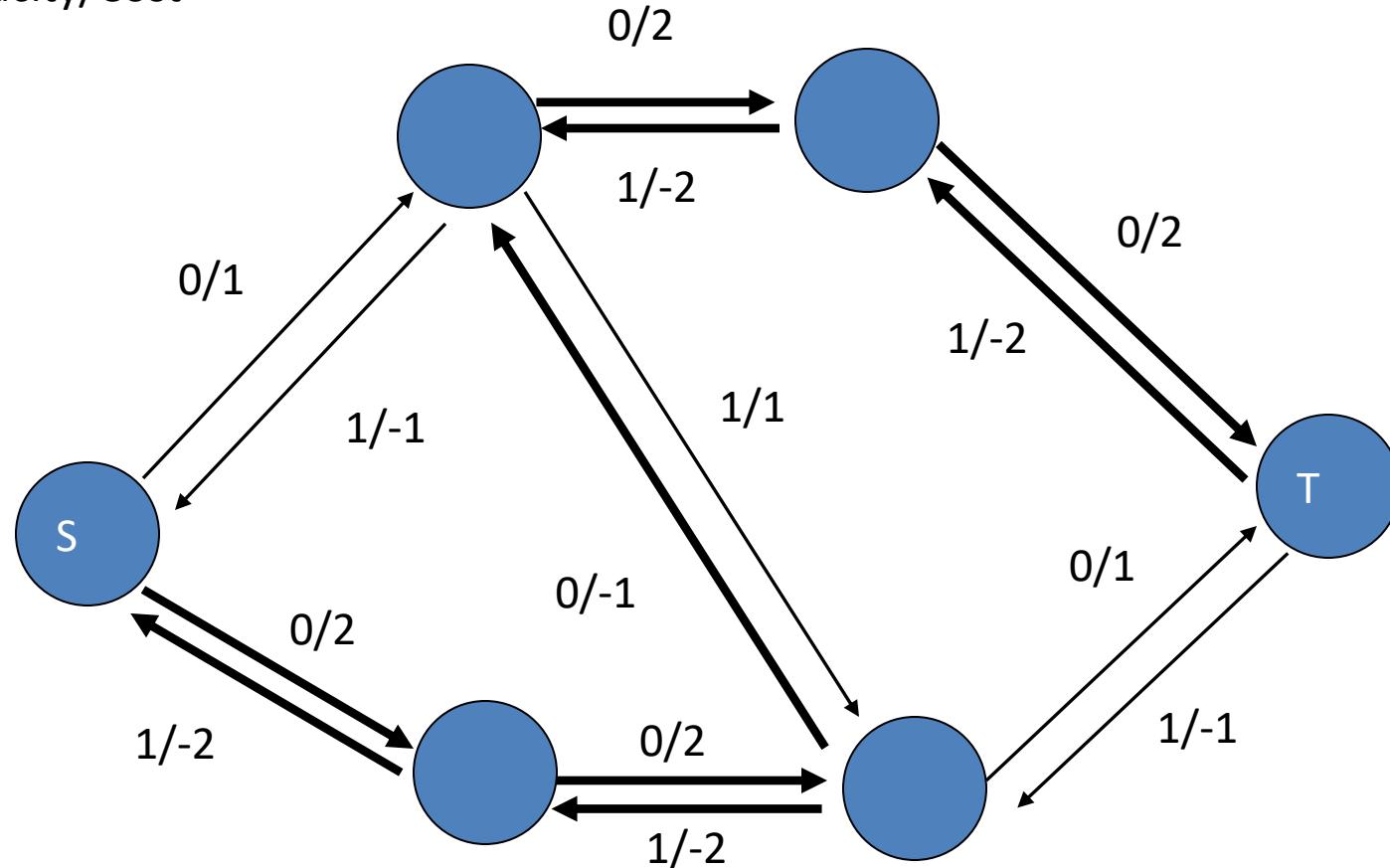
Capacity/Cost



Flow=1, cost = $2+2-1+2+2=7$

Example

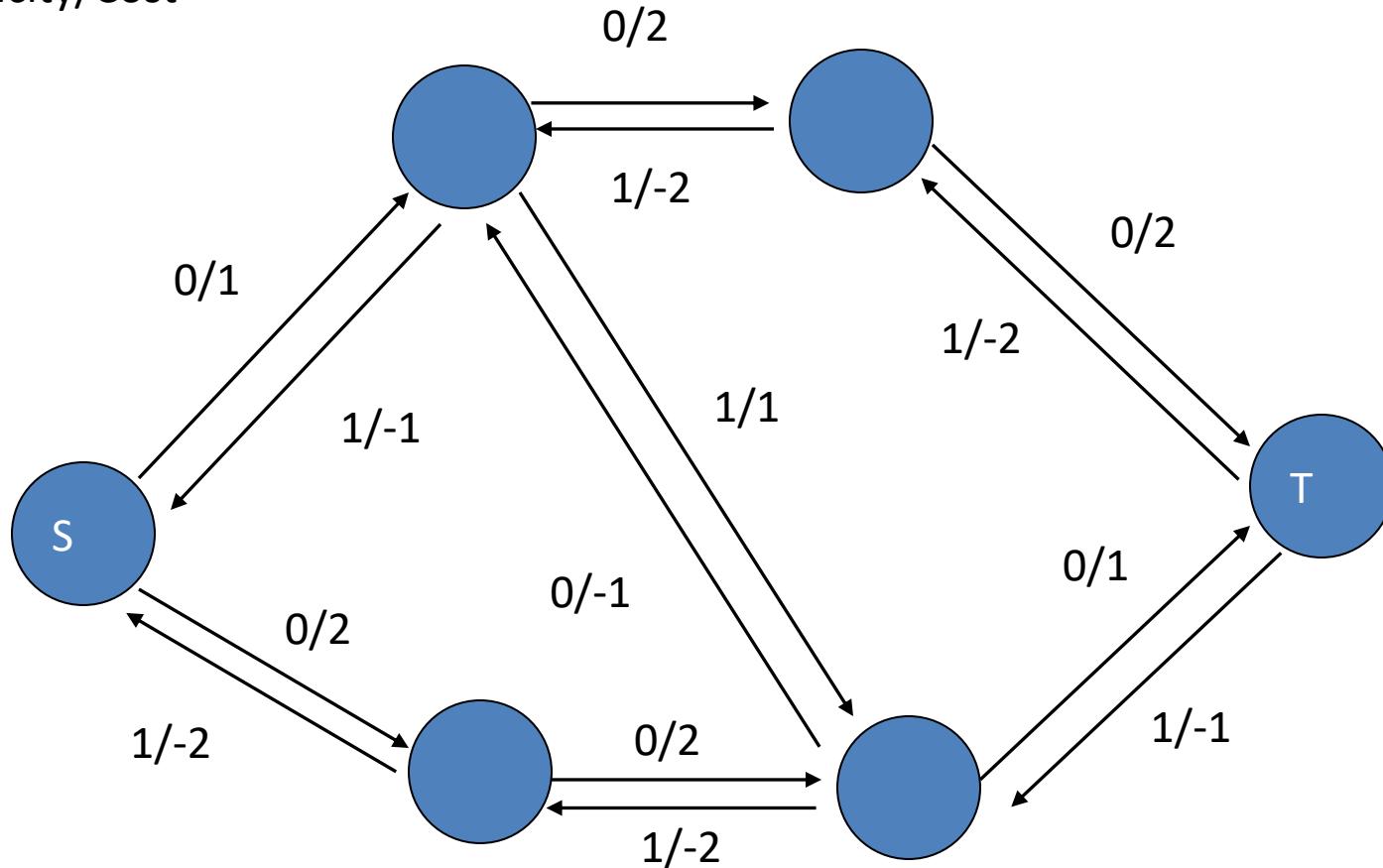
Capacity/Cost



Flow=1, cost = $2+2-1+2+2=7$

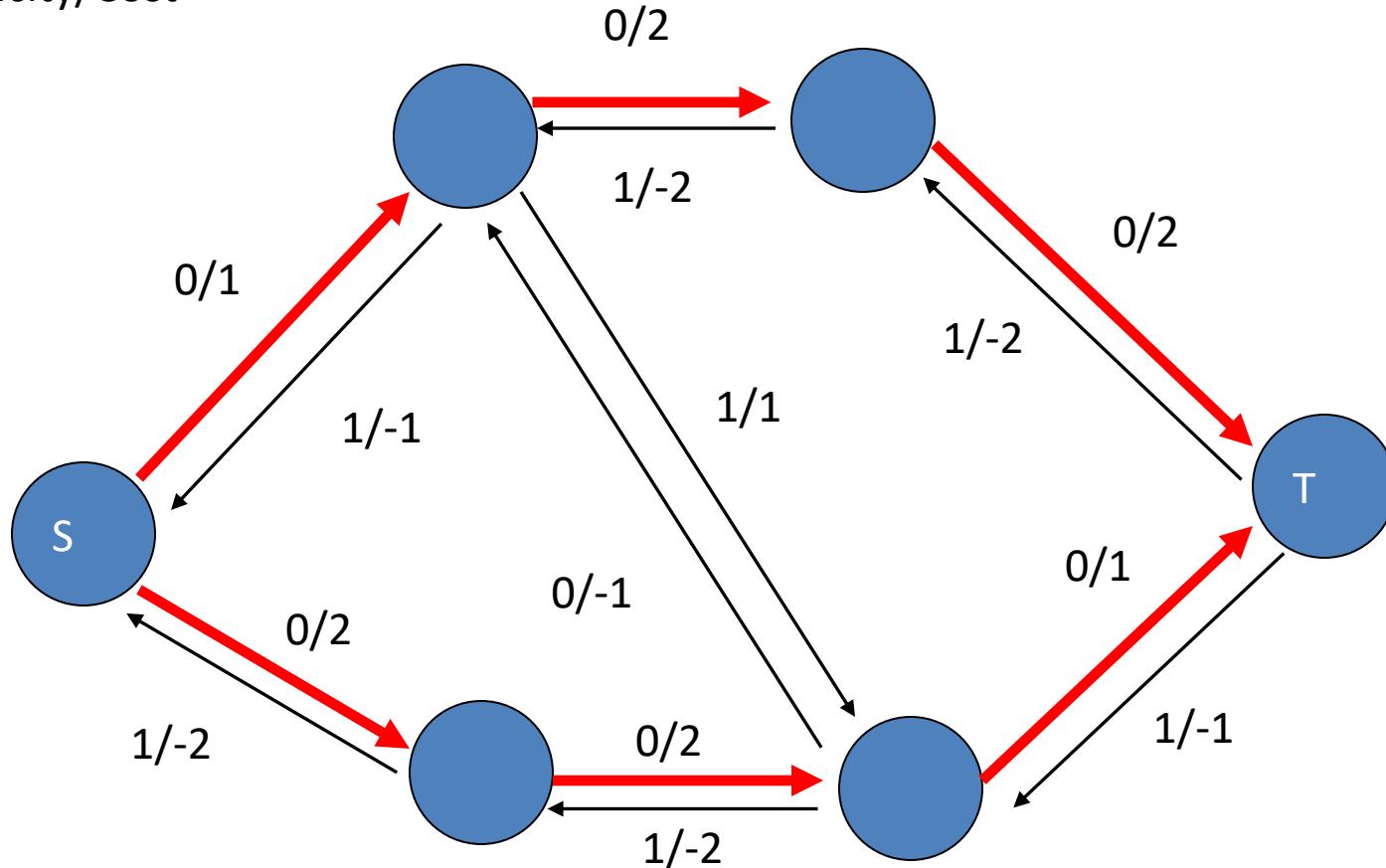
Example

Capacity/Cost



Example

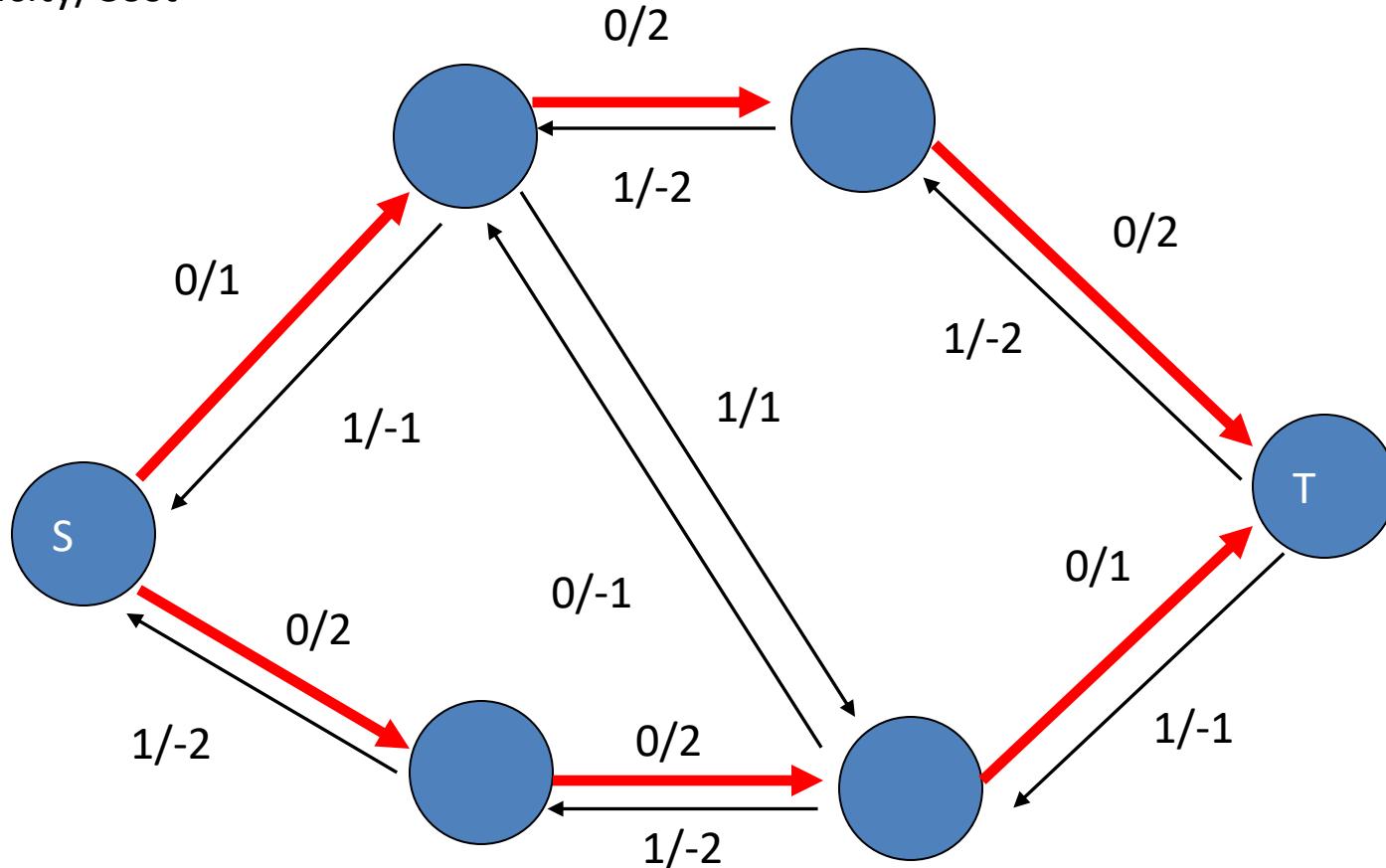
Capacity/Cost



Flow=1, cost=3 and Flow=1 cost=7

Example

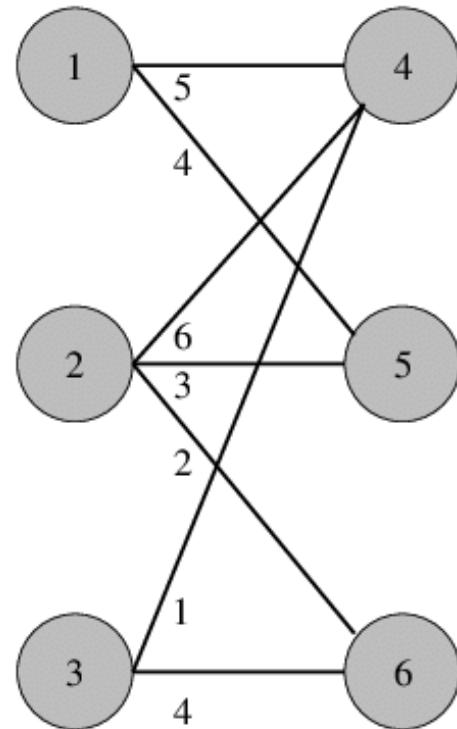
Capacity/Cost



Flow = 2, cost=10

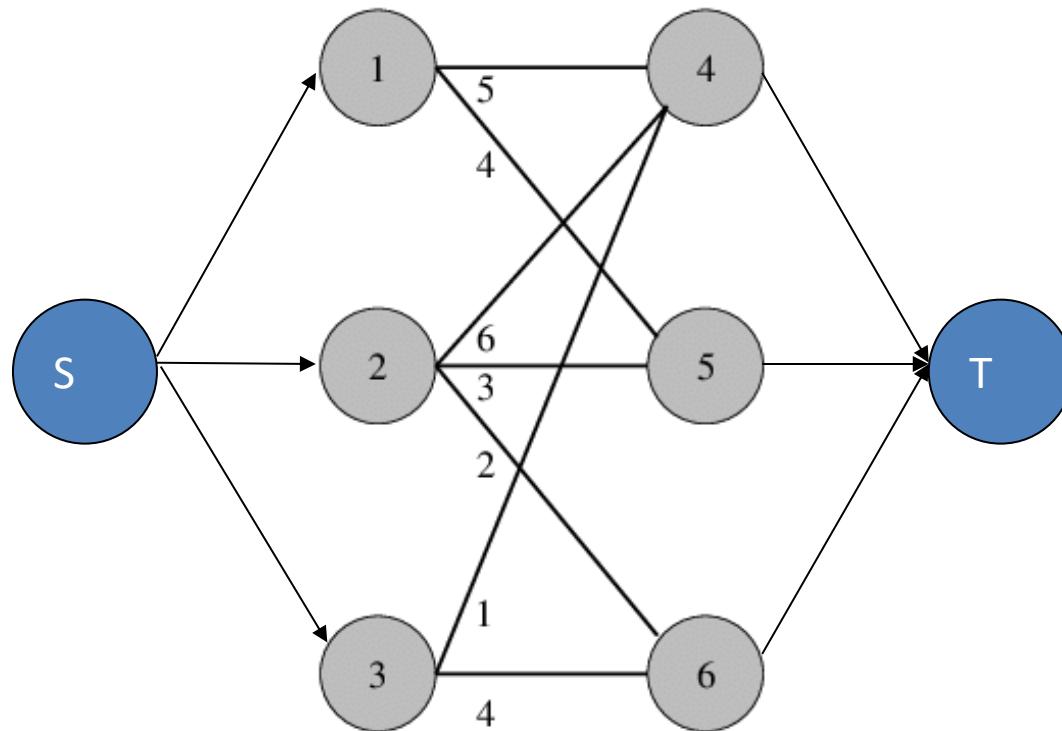
Minimum Weight Matching (MWM)

- Similar to bipartite matching but with weight



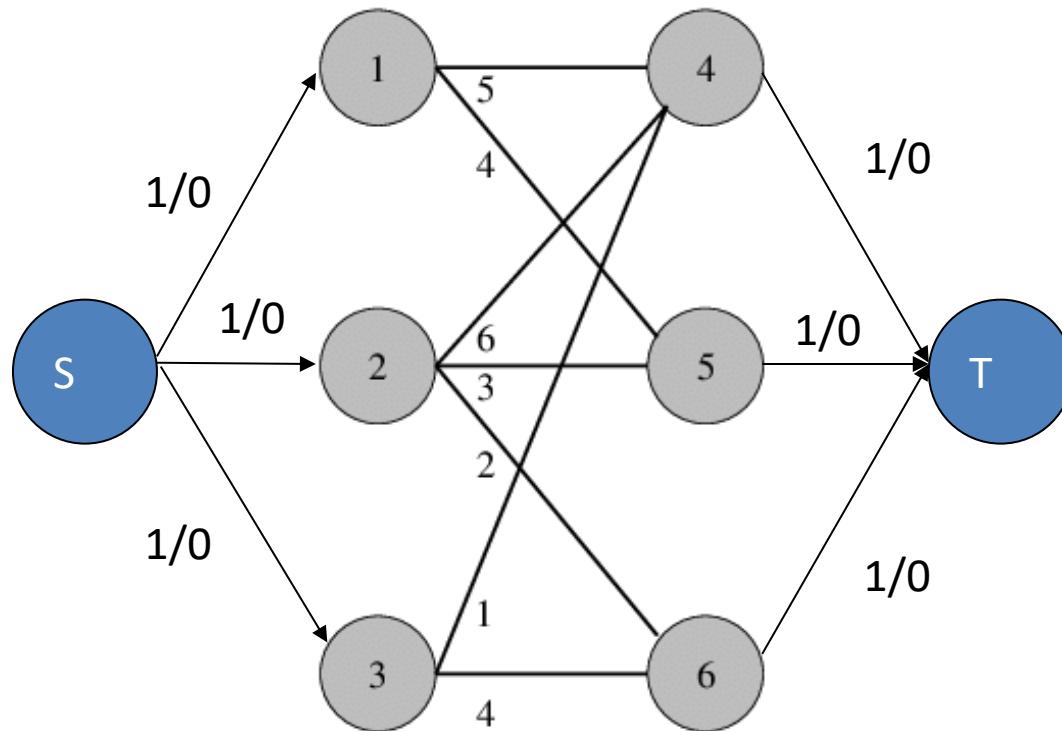
Minimum Weight Matching (MWM)

- Transform to a flow network with cost



Minimum Weight Matching (MWM)

- Attach capacity and weight



Minimum Weight Matching (MWM)

- Min-cost flow equals MWM

