Max flow and flow network

FORD FULKERSON

- Once the path is chosen, what ever will be left over we call it as **“Residual Graph”.**
- **Augmenting path** is the path we are selecting
- Capacity minimum is the → **Augmenting Flow.**
- $\mathbf{Cf(u,v) \quad = \quad c(u,v) \quad - \quad f(u,v)}$

Capacity of edge in residual graph

Capacity of edge in flow network

Flow going through edge

- **REVERSE EDGE**

- $\mathbf{Cf(v,u) \quad = \quad c(v,u) \quad + \quad f(u,v)}$
- **Capacity of reverse edge is increased every time when a new flow is flowing through it.**

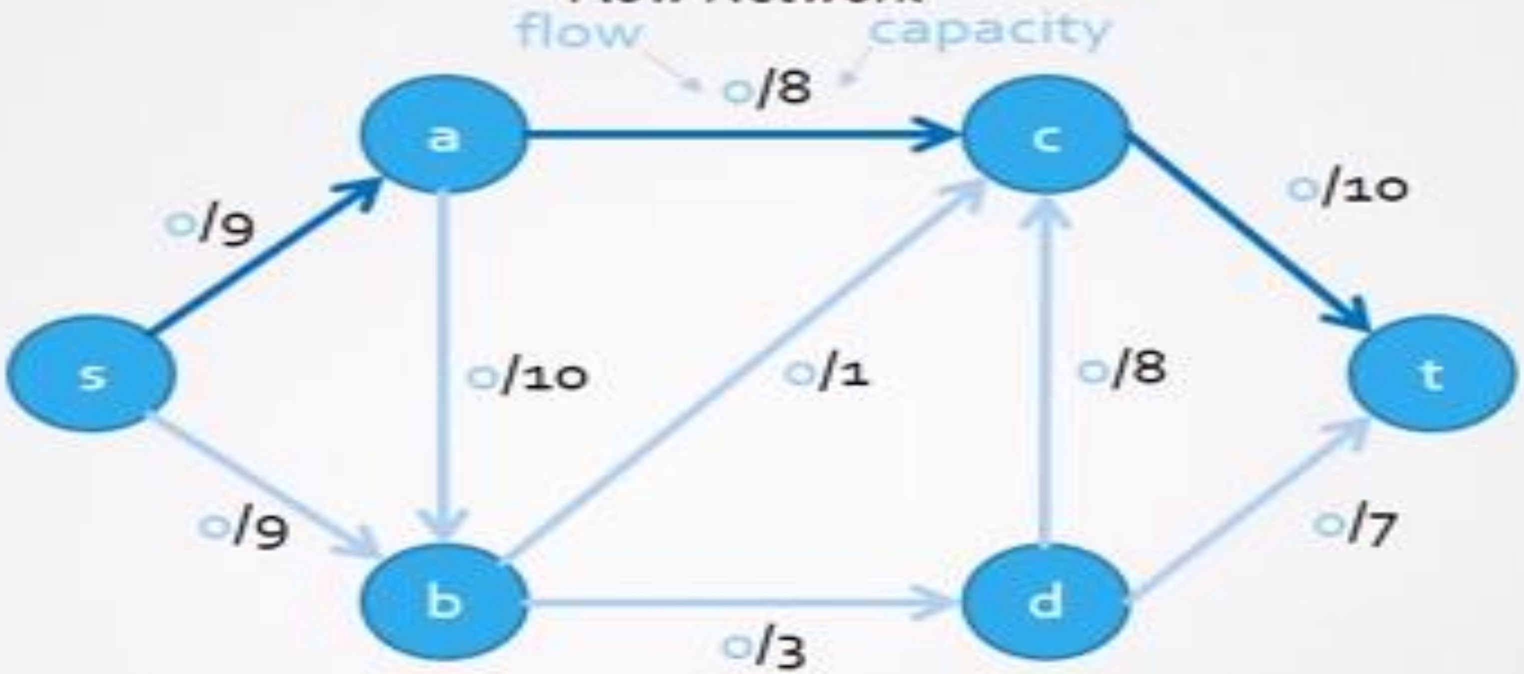
- Flow network (G)
  - Simple Directed graph
  - 1 source s, 1 sink t
  - Non-negative capacity for each edge
- Residual Graph (Gf)
  - Only edges from G that can still have more flow
  - $Cf(u,v) = c(u,v) - f(u,v)$
  - Added edges: reverse edges to decrease flow *'cancellation'*

    $cf(v,u)=cf(v,u)+f(u,v)$
- Augmenting Paths (p)
  - Path from s to t in Gf
  - Residual capacity = smallest capacity of edges of p

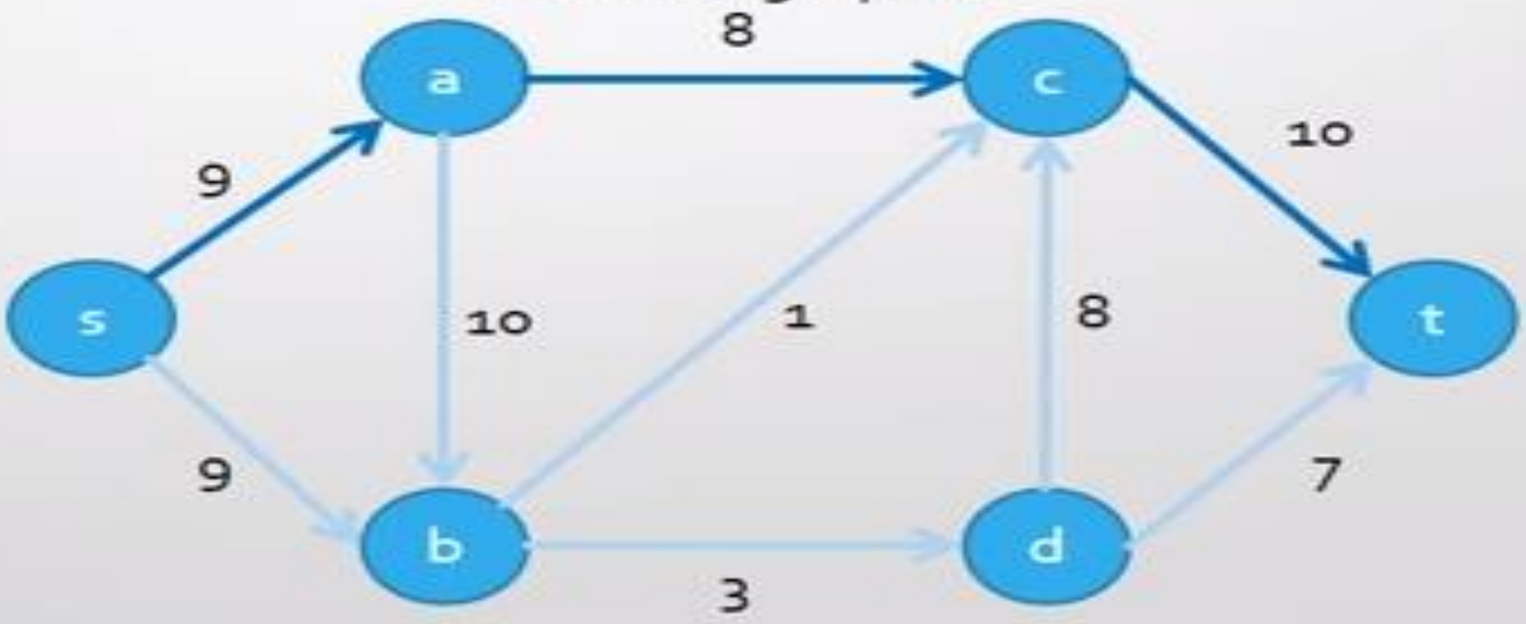
Start:
F(u,v) = 0 for all edges

Fm = o

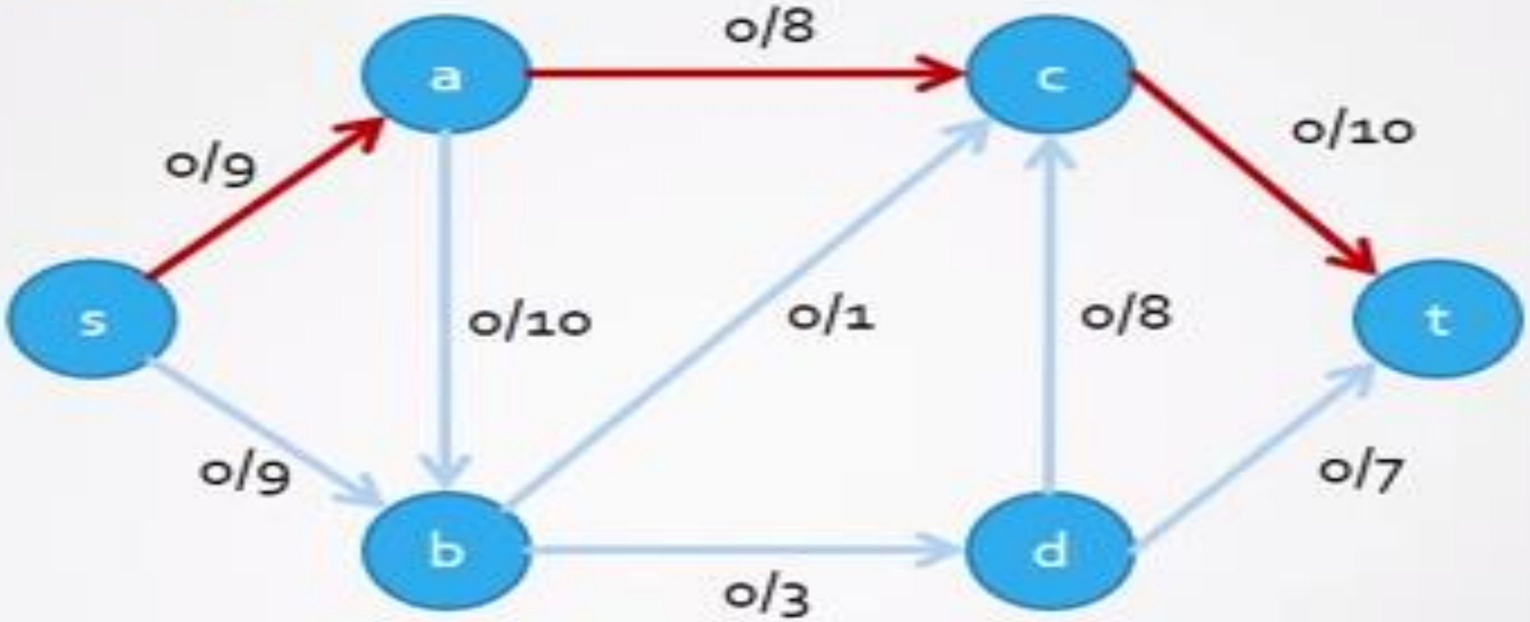
Flow Network



Residual graph Gf



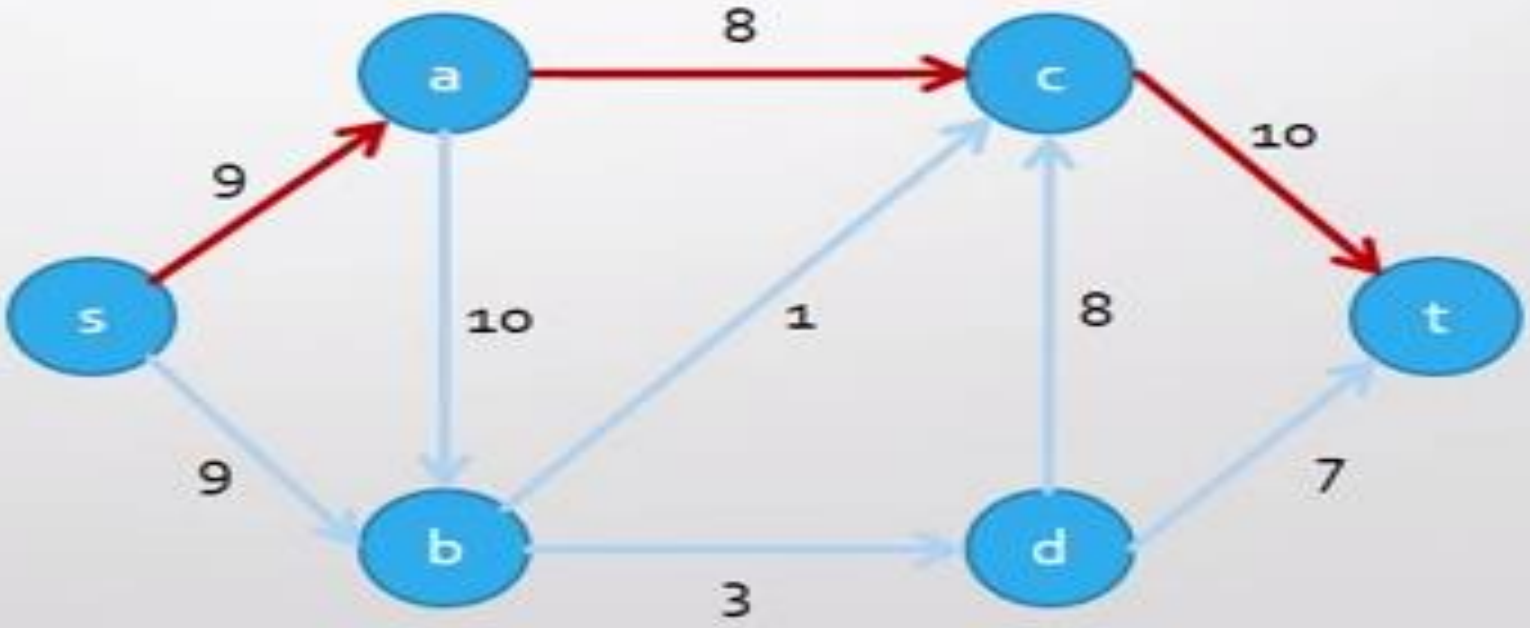
Find Augmenting path: s->a->c->t
Find Residual capacity:



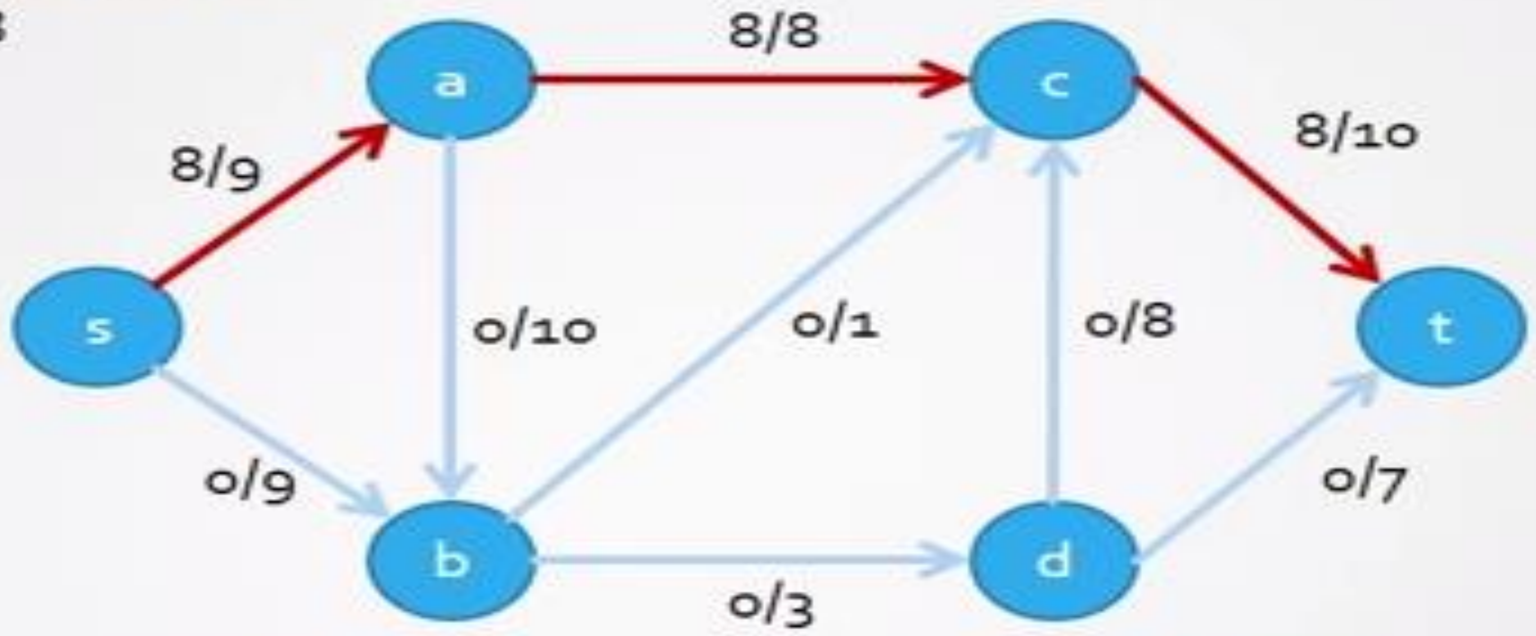
Fm = 0

Cf(u,v)=cf(u,v)-f(u,v)
Cf(v,u)=cf(v,u)+f(u,v)

Residual graph Gf



Find Augmenting path: s->a->c->t
Find Residual capacity: 8



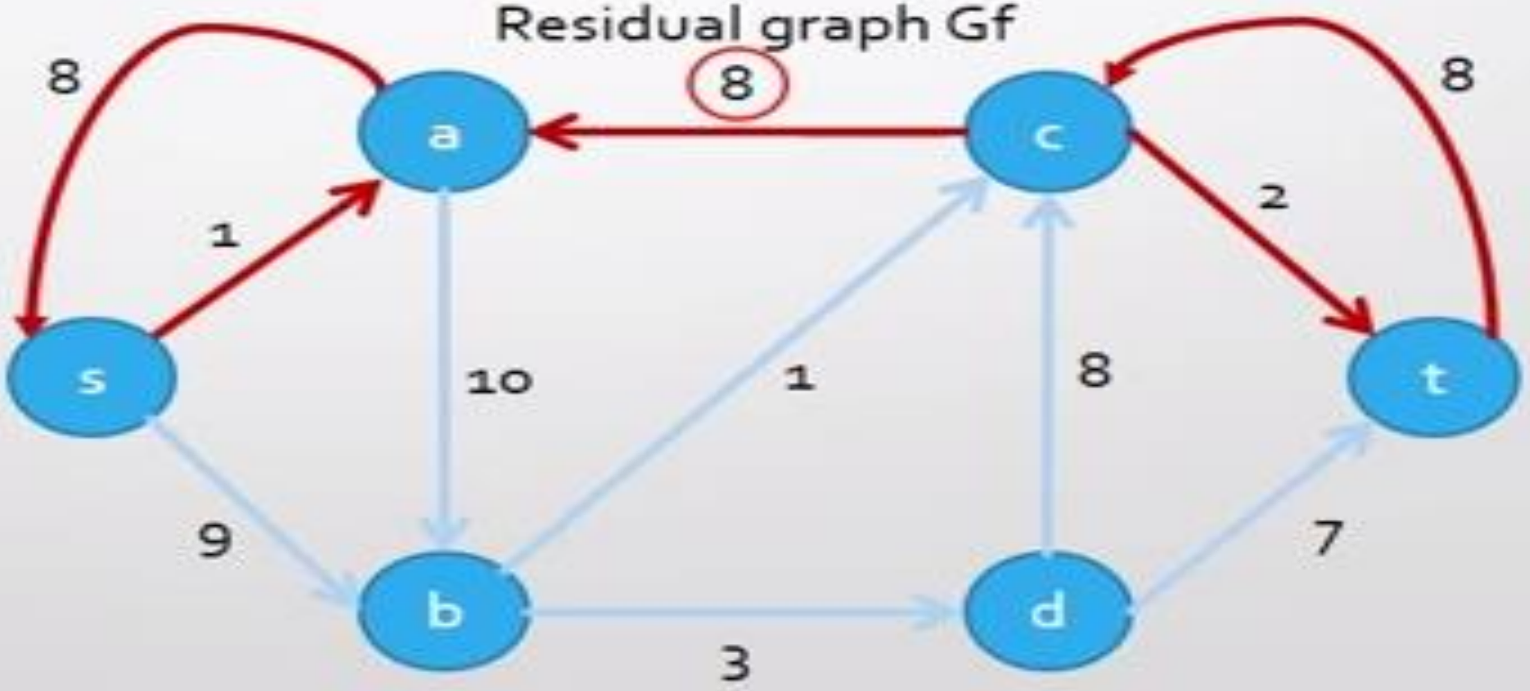
Fm = 0+ 8

Cf(u,v)=cf(u,v)-f(u,v)
Cf(v,u)=cf(v,u)+f(u,v)

Cf(s,a)=9-8=1
Cf(a,c)=8-8=0
Cf(c,t)=10-8=2

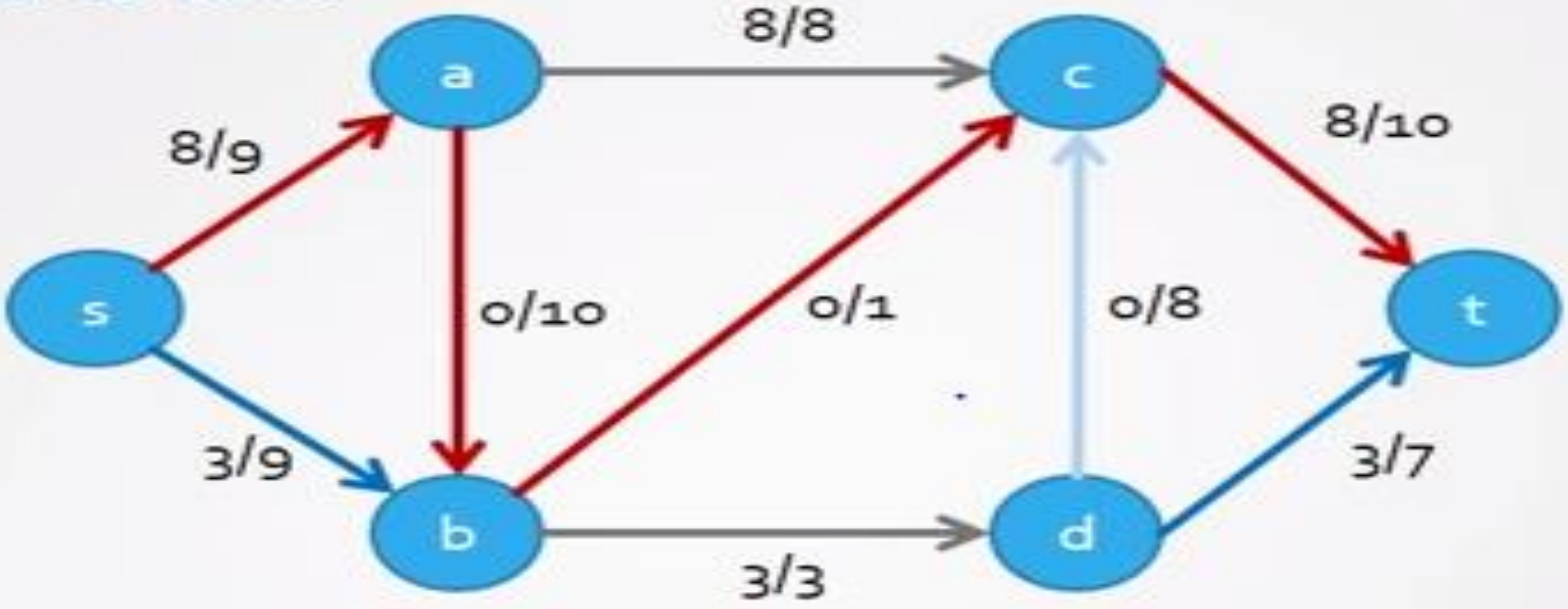
Cf(a,s)=8
Cf(c,a)=8
Cf(t,c)=8

Residual graph Gf



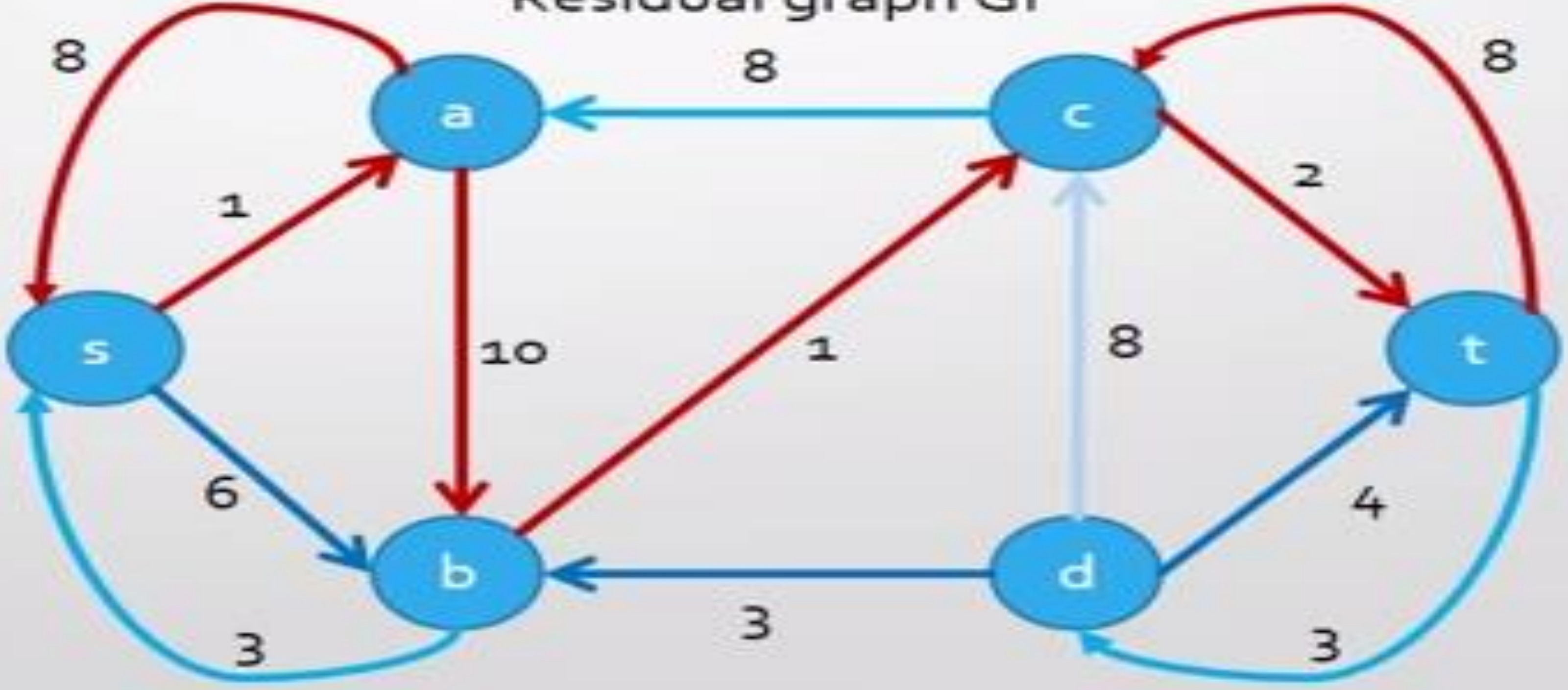
Find augmenting path: s->a->b->c->t

Find residual capacity:



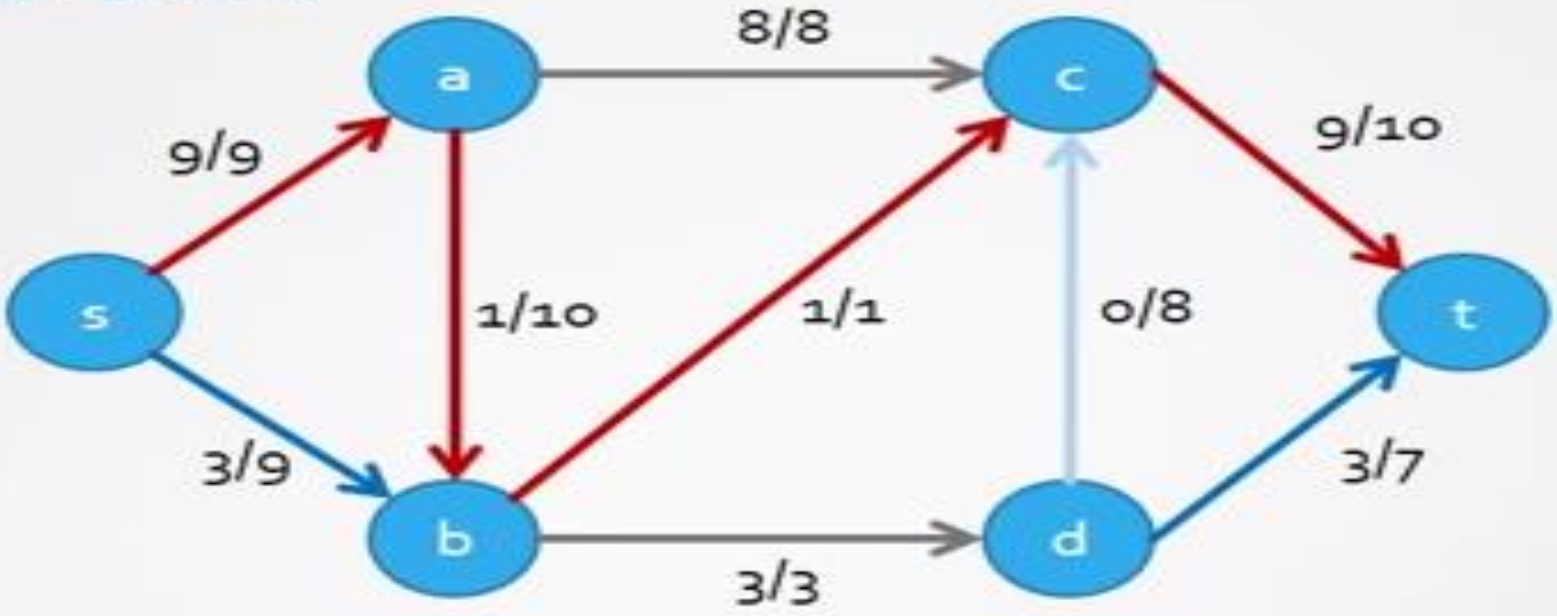
Fm = 0+8 + 3

Residual graph Gf

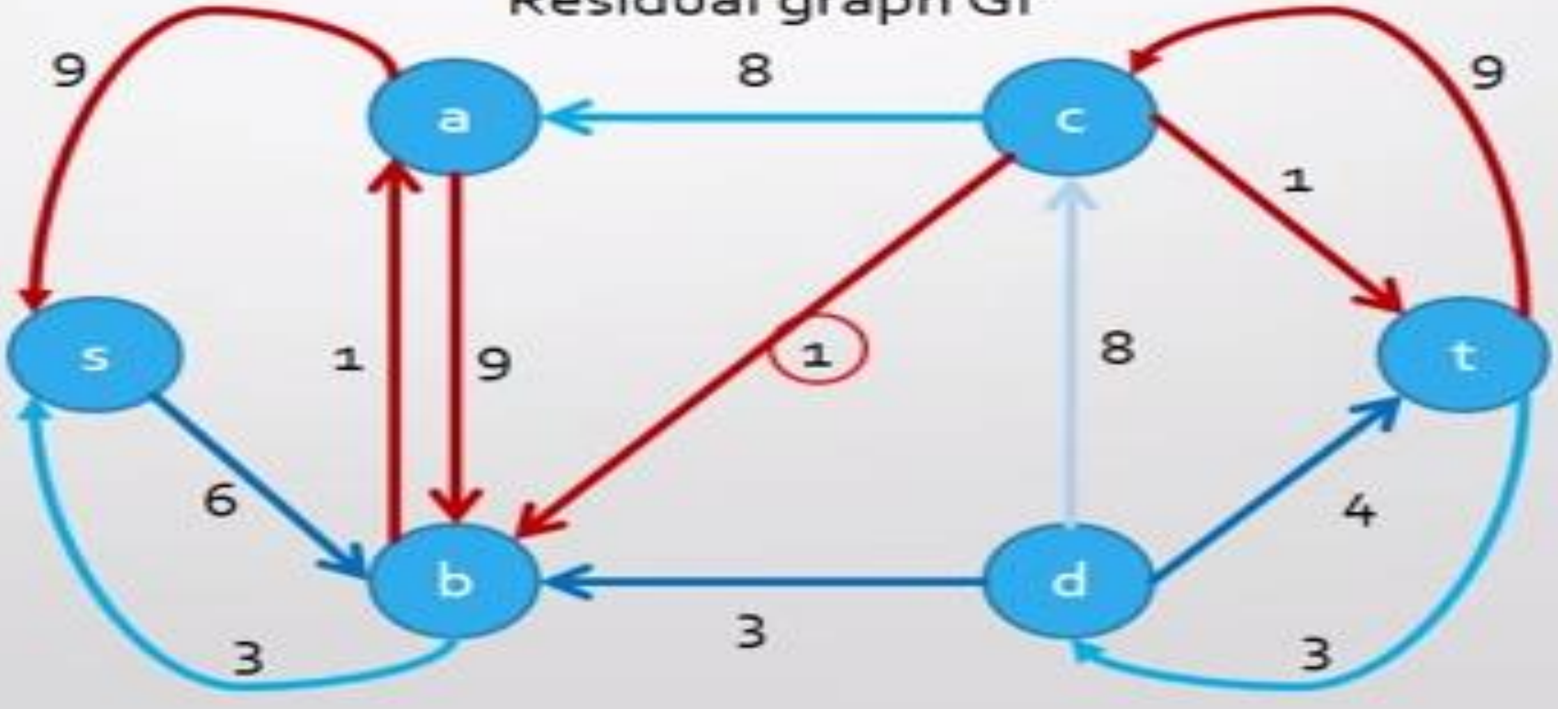


Find augmenting path: s->a->b->c->t
Find residual capacity: 1

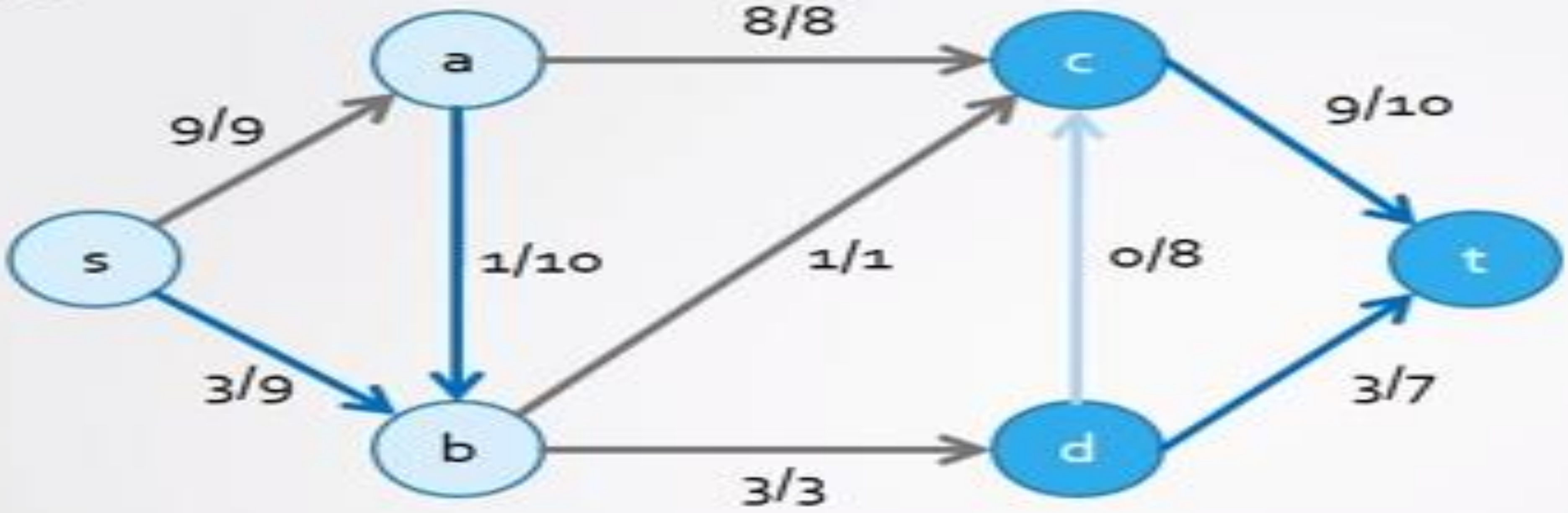
Fm = 0+8+3+1



Residual graph Gf

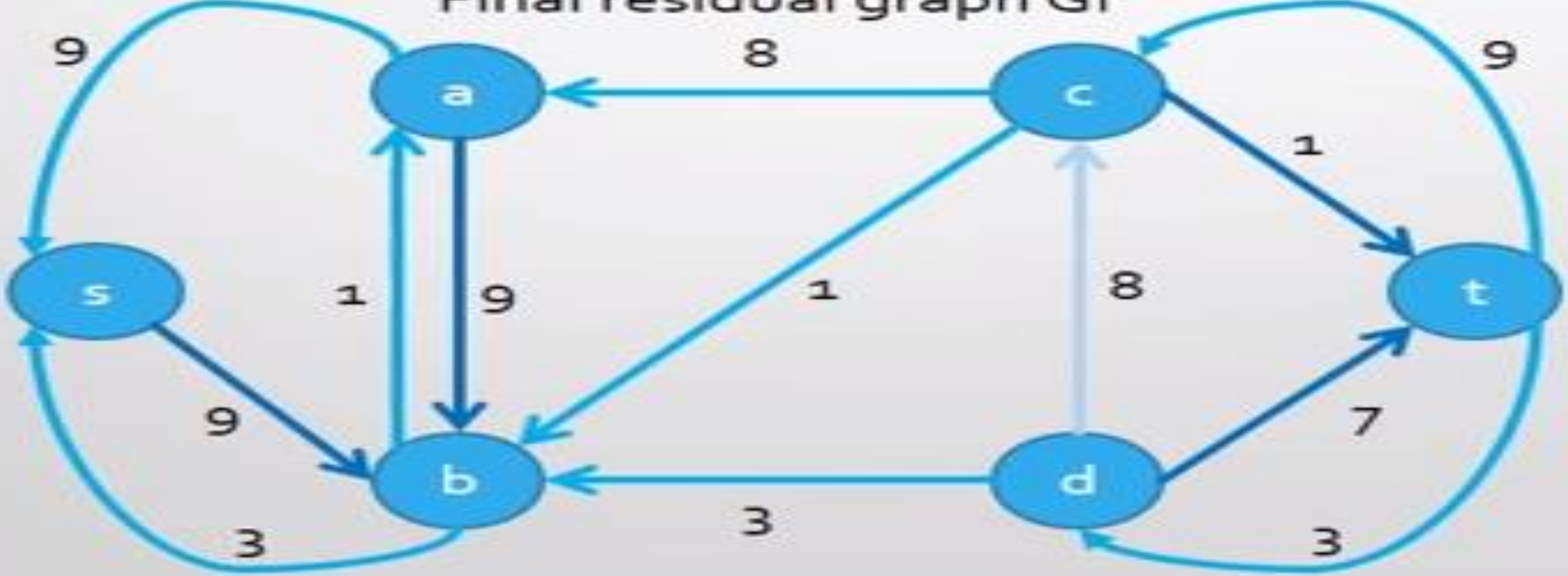


Augmenting path: none left



Fm = 12

Final residual graph Gf



The algorithm

Fm = o

While there exist an augmenting path p in Gf

Find augmenting path p

Cf (p)= smallest capacity on p

Fm = Fm + Cf(p)

For each edge in p

cf(u,v) = cf(u,v)-Cf(p)

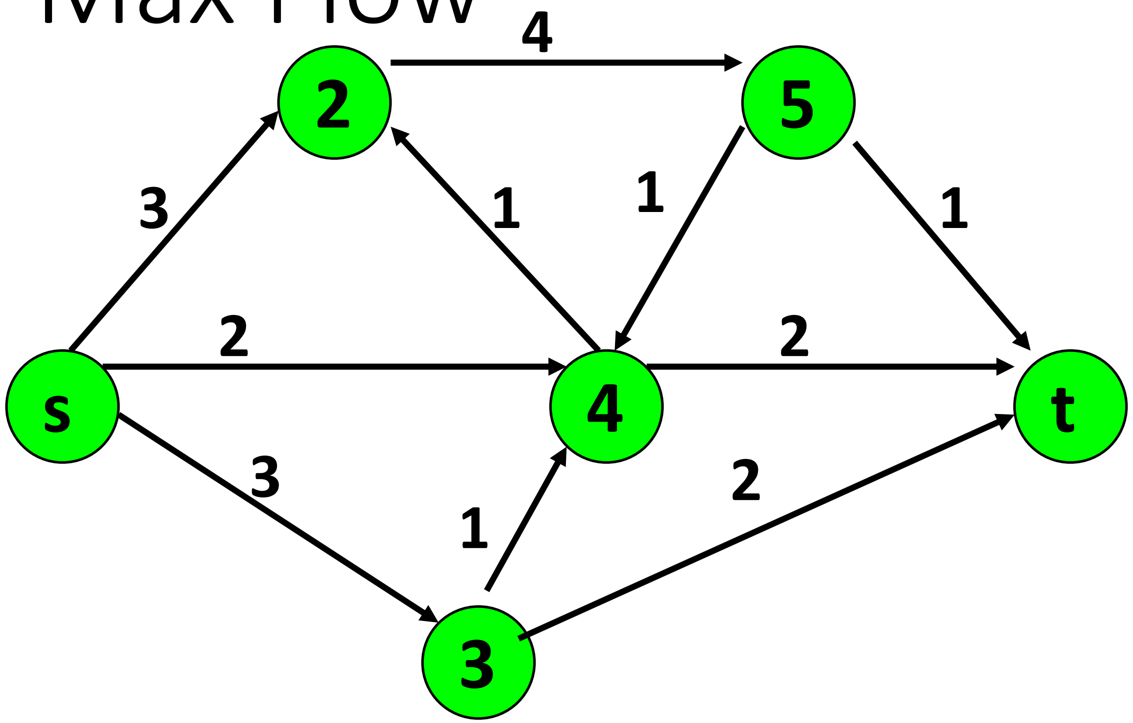
cf(v,u) = cf(v,u)+cf(p)

Analysis
Complexity of the Algorithm

- Flow increased by at least 1 at every iteration, so the while loop is repeated Fm times at most, where Fm is the maximum flow
  - Finding an augmenting path : O(E) where E is the number of edges
  - Operations on value: O(1) Update the maximum flow, and capacities on the edges
  - Each iteration: O(E)

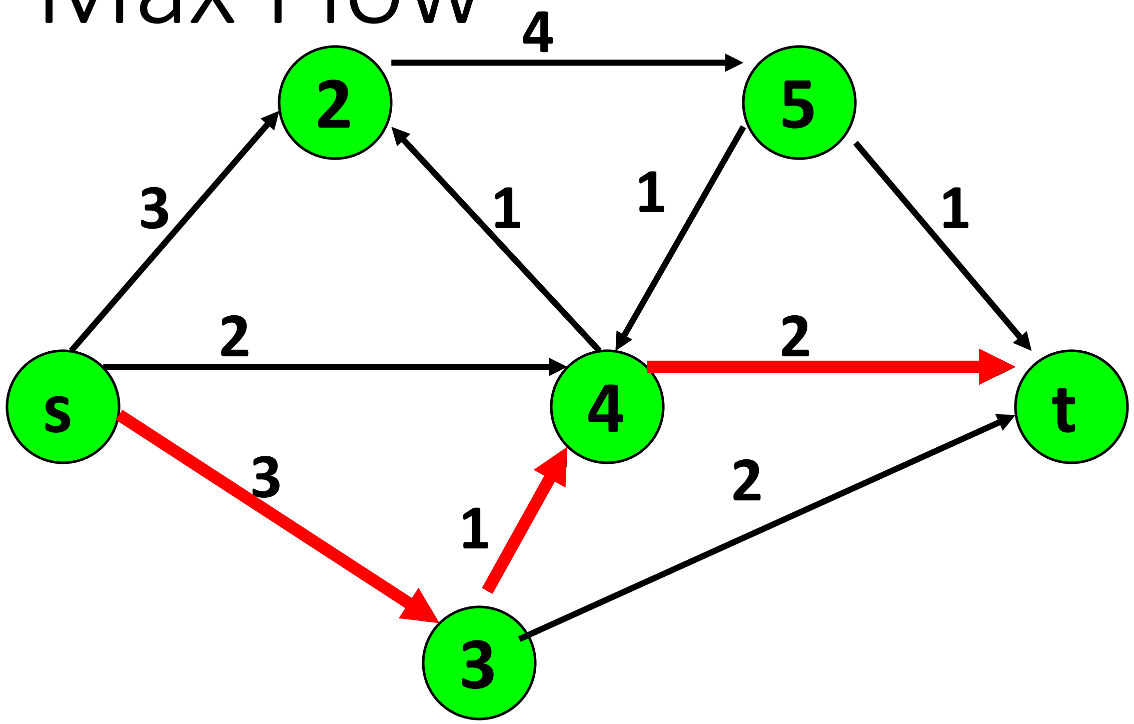
This algorithm runs in time O(E*Fm)

Ford-Fulkerson Max Flow



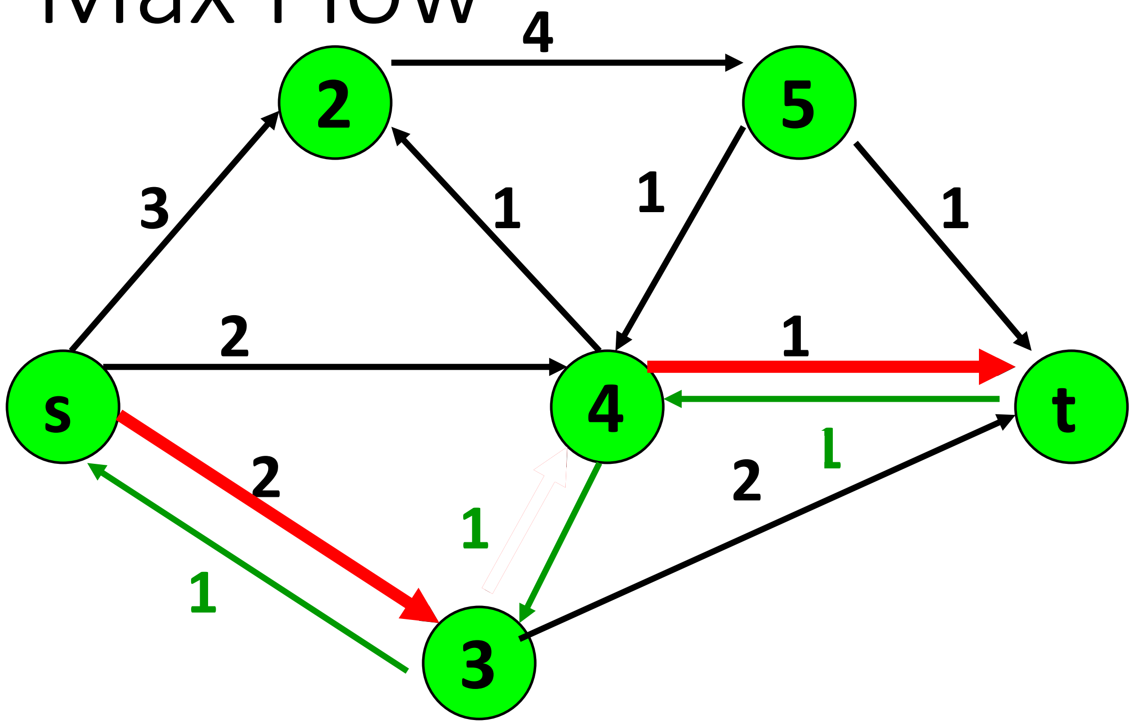
This is the original network, and the original residual network.

Ford-Fulkerson Max Flow



Find any s-t path in G(x)

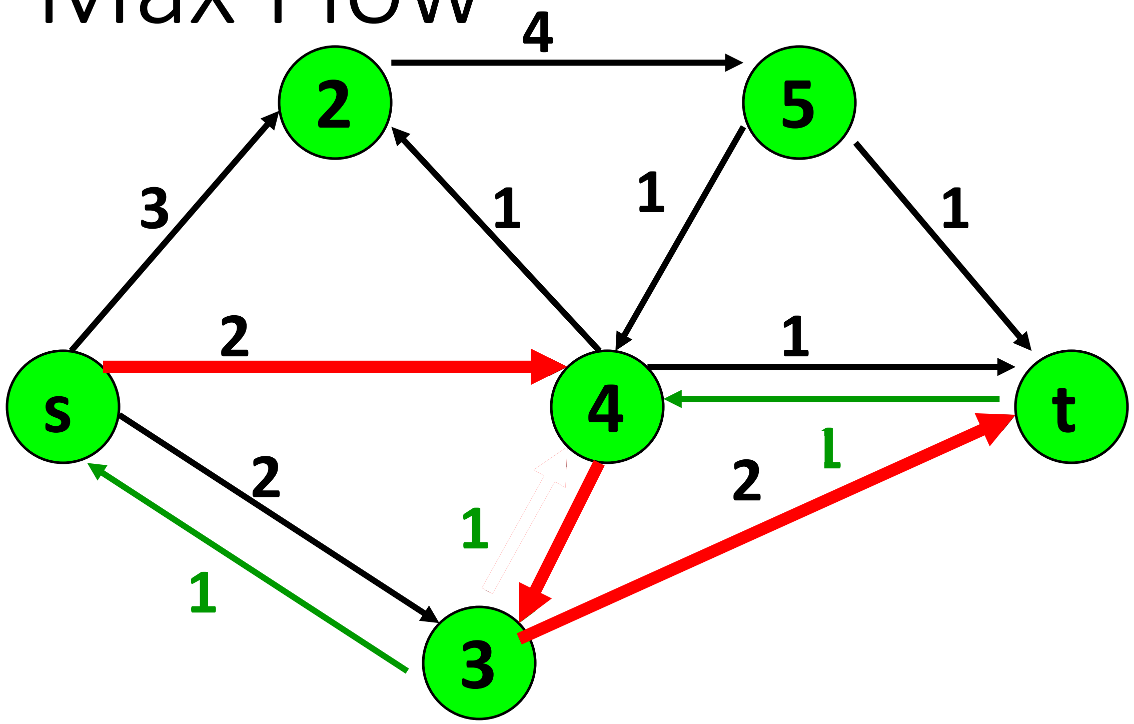
Ford-Fulkerson Max Flow



Determine the capacity Δ of the path.

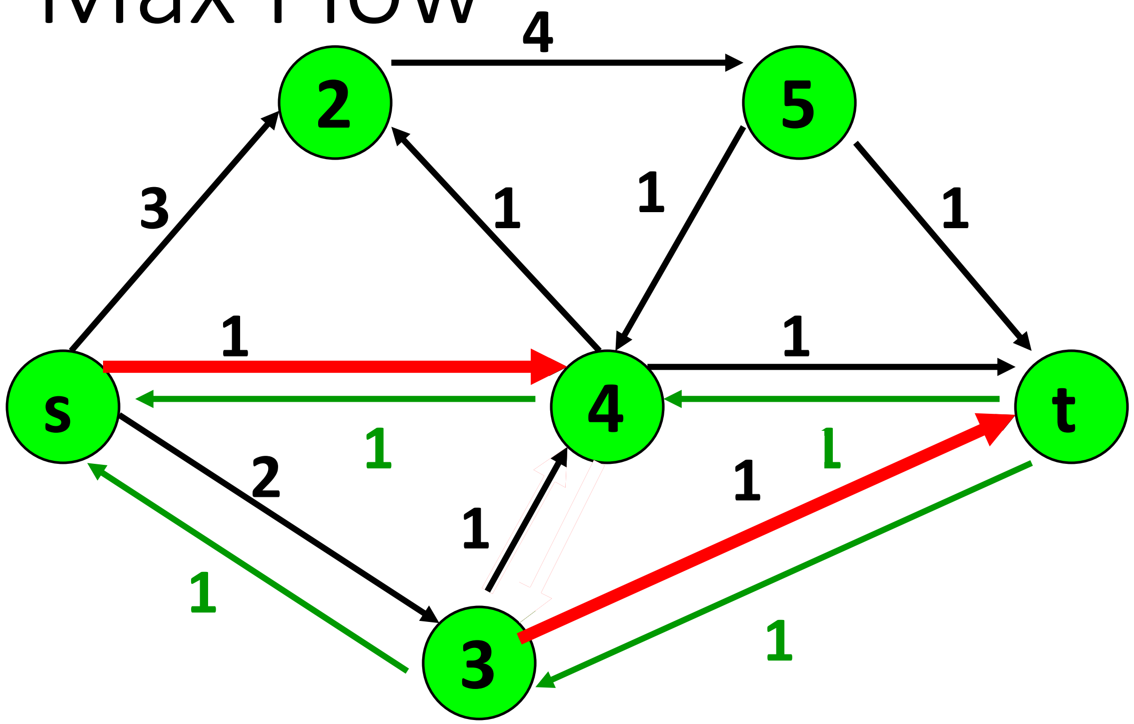
Send Δ units of flow in the path.
Update residual capacities.

Ford-Fulkerson Max Flow



Find any s-t path

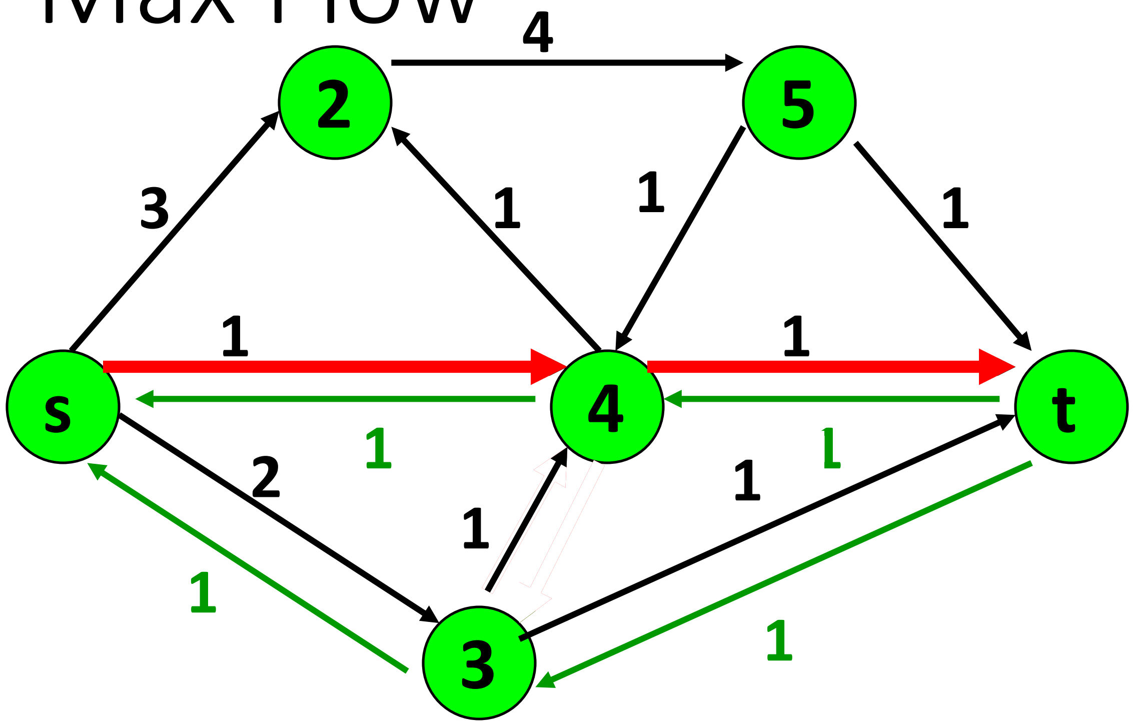
Ford-Fulkerson Max Flow



Determine the capacity Δ of the path.

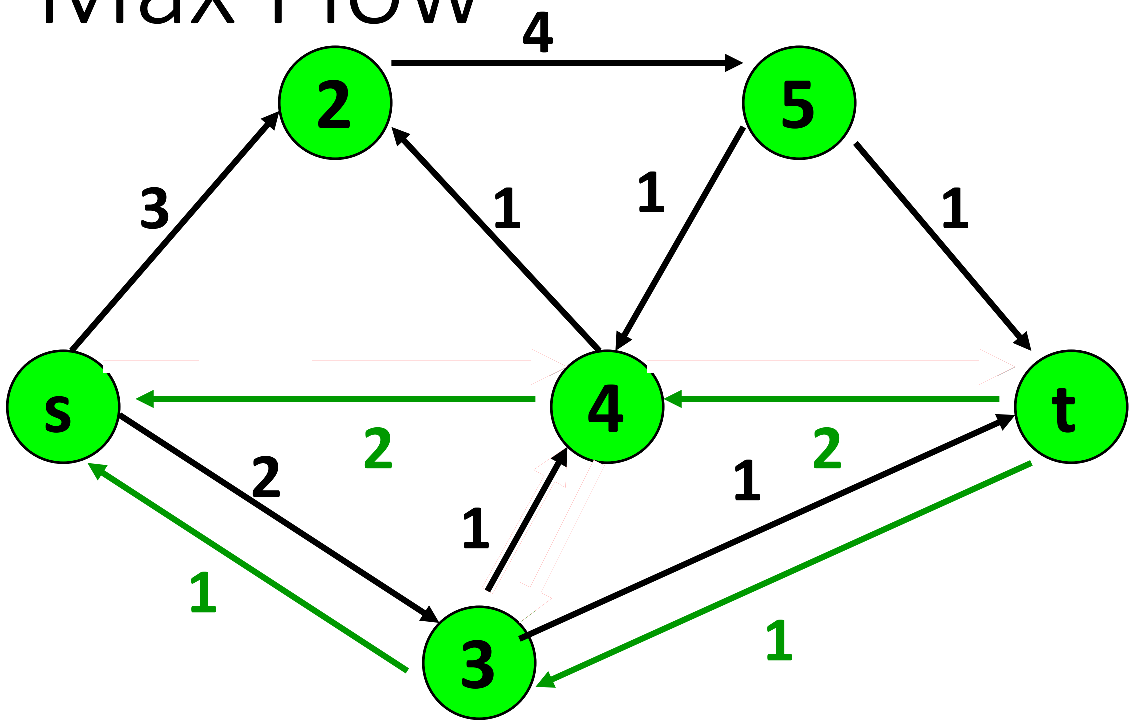
Send Δ units of flow in the path.
Update residual capacities.

Ford-Fulkerson Max Flow



Find any s-t path

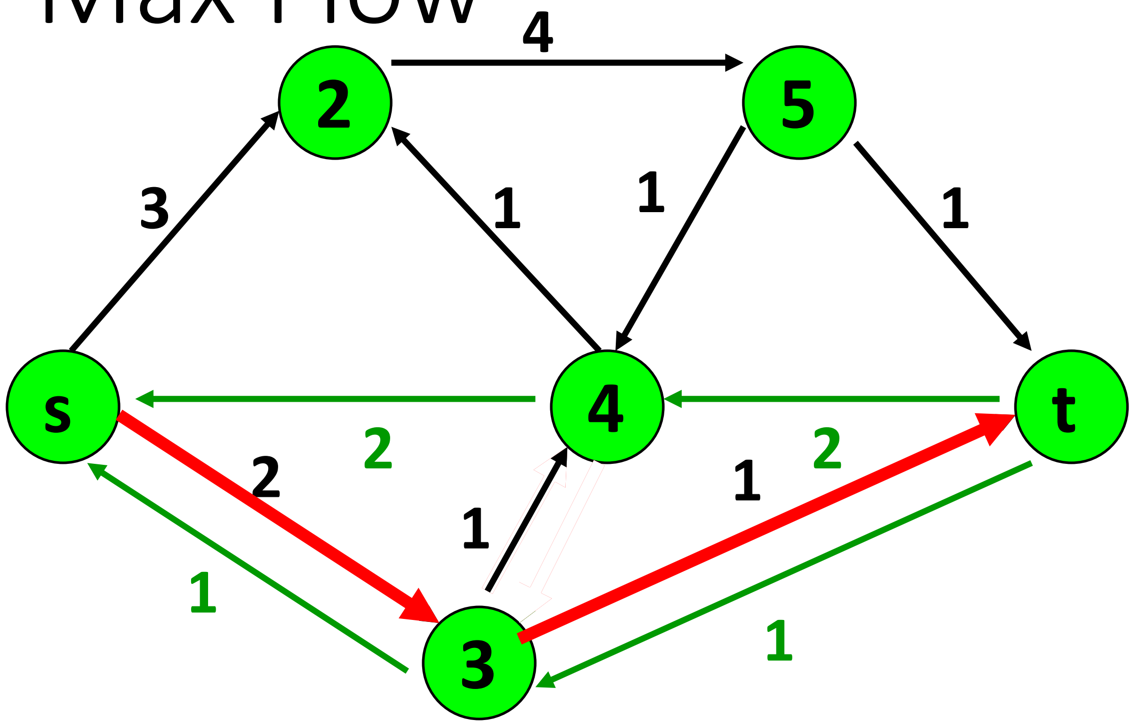
Ford-Fulkerson Max Flow



Determine the capacity Δ of the path.

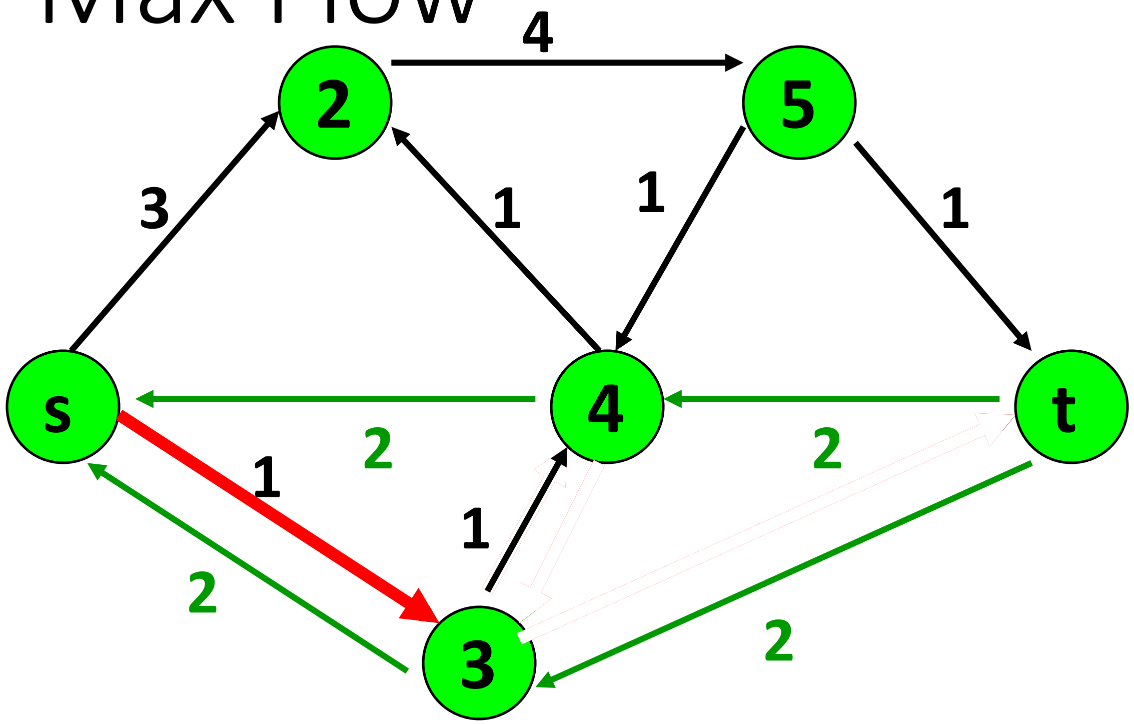
Send Δ units of flow in the path.
Update residual capacities.

Ford-Fulkerson Max Flow



Find any s-t path

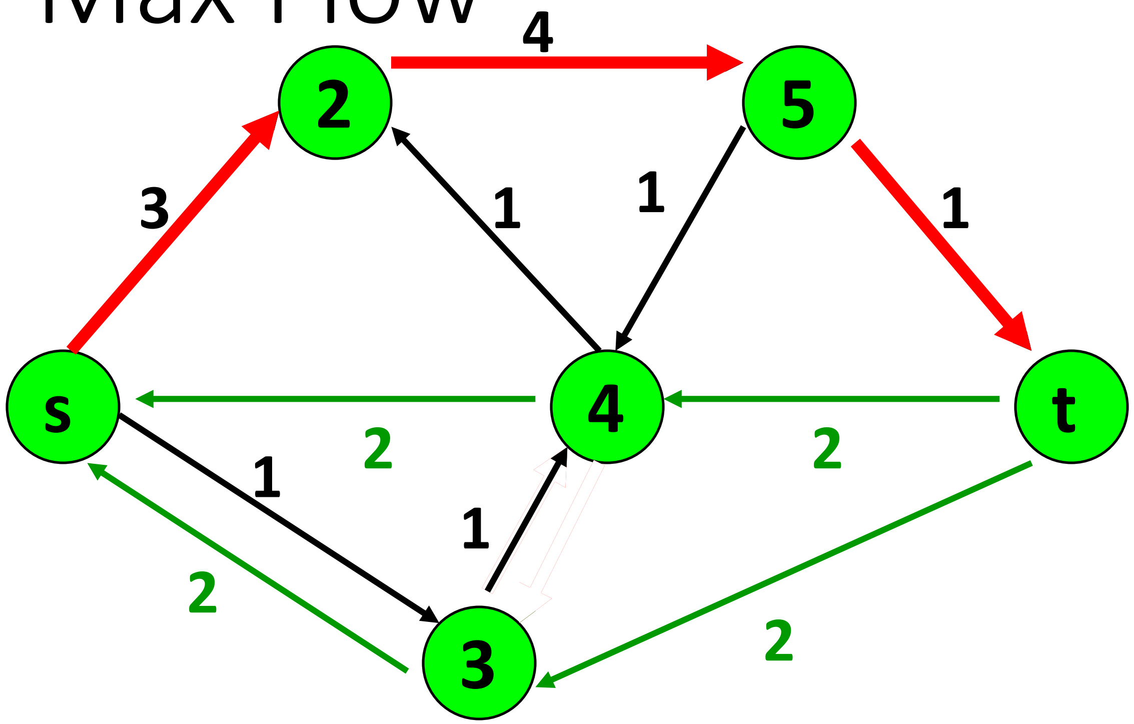
Ford-Fulkerson Max Flow



Determine the capacity Δ of the path.

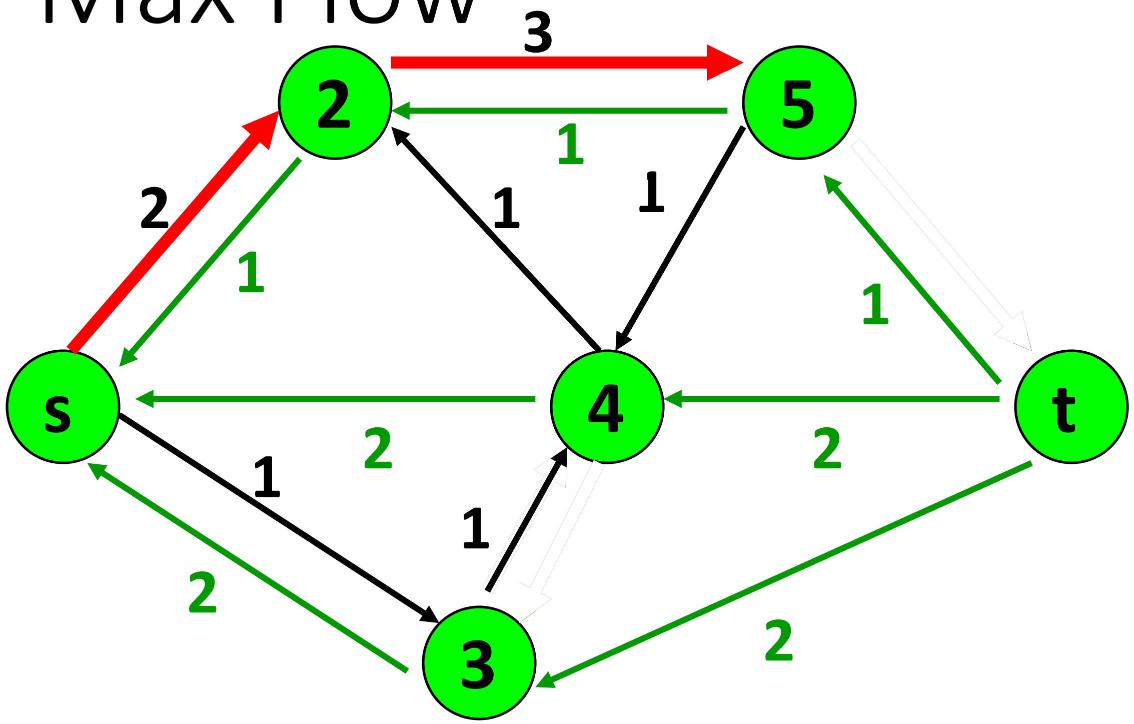
Send Δ units of flow in the path.
Update residual capacities.

Ford-Fulkerson Max Flow



Find any s-t path

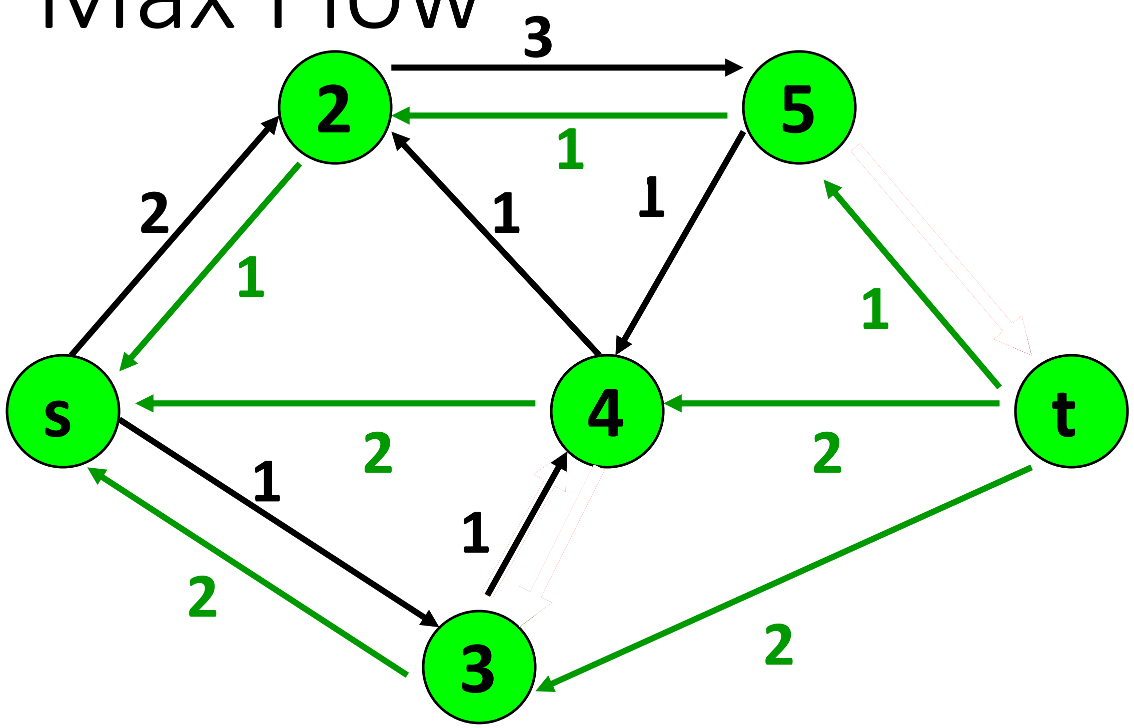
Ford-Fulkerson Max Flow



Determine the capacity Δ of the path.

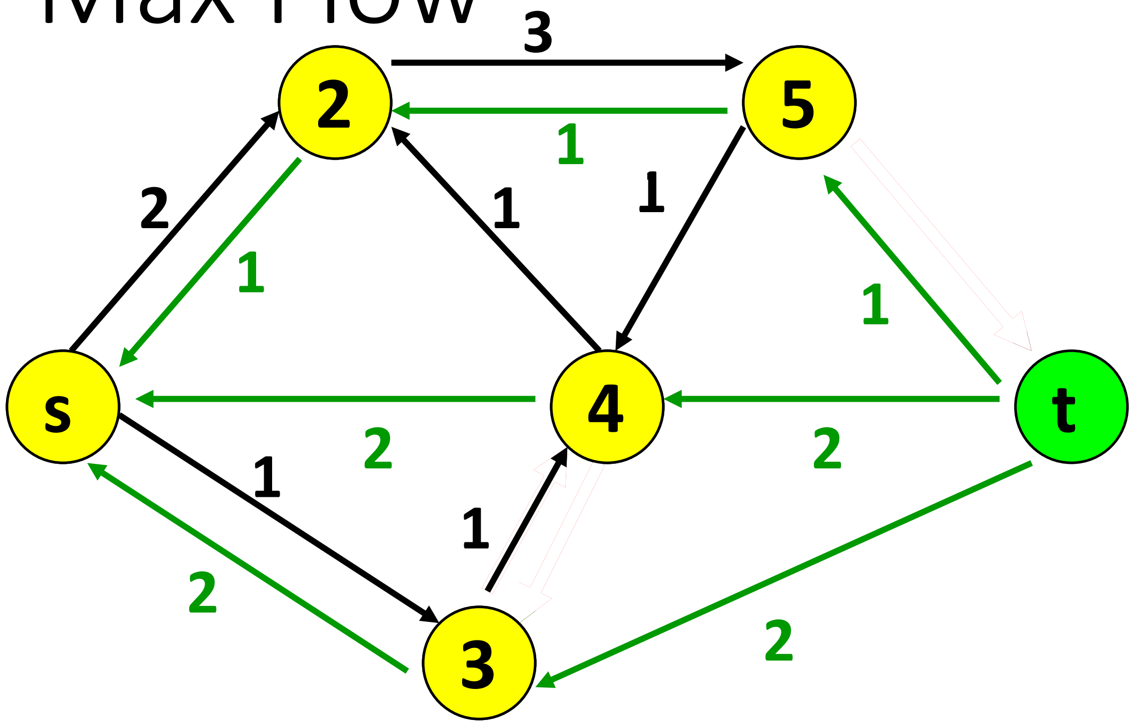
Send Δ units of flow in the path.
Update residual capacities.

Ford-Fulkerson Max Flow



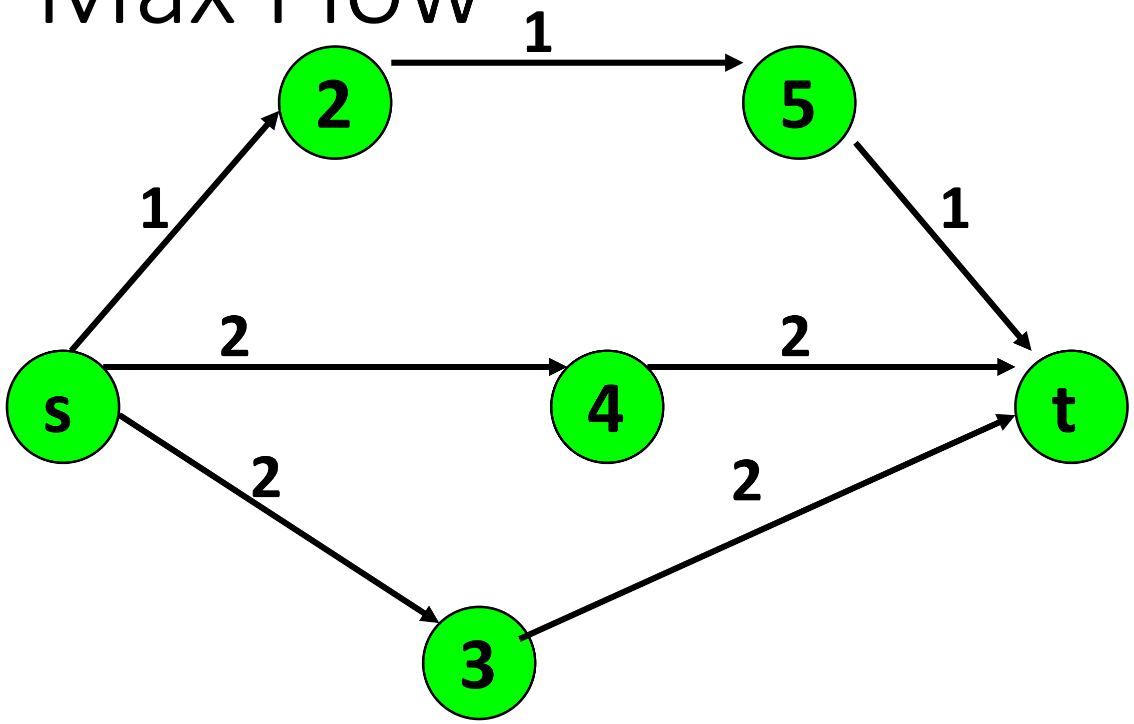
There is no s-t path in the residual network.
This flow is optimal

Ford-Fulkerson Max Flow



These are the nodes that are reachable from node s.

Ford-Fulkerson Max Flow



Here is the optimal flow