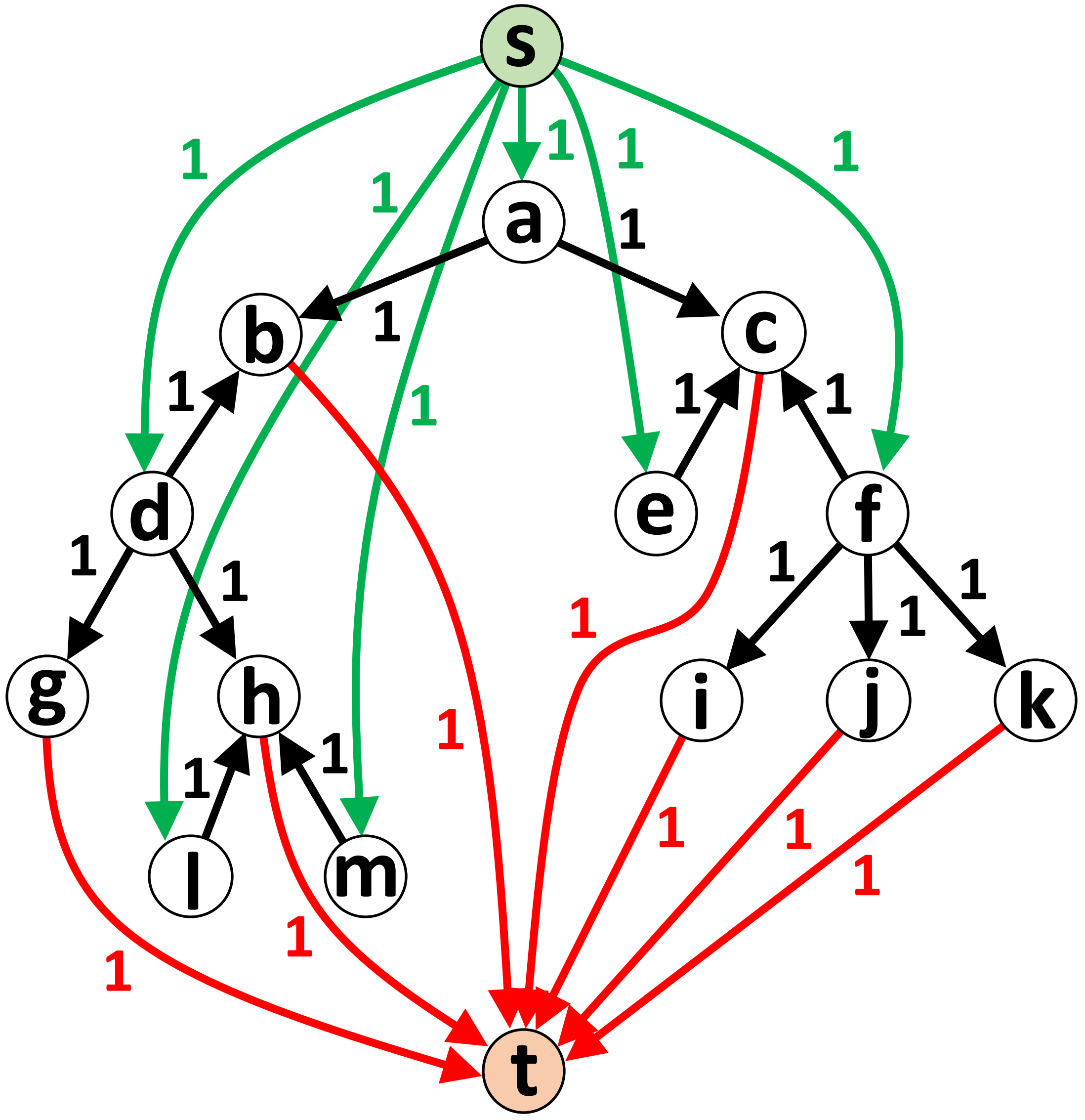
Flow Networks
CSCI 532

Maximum Matching



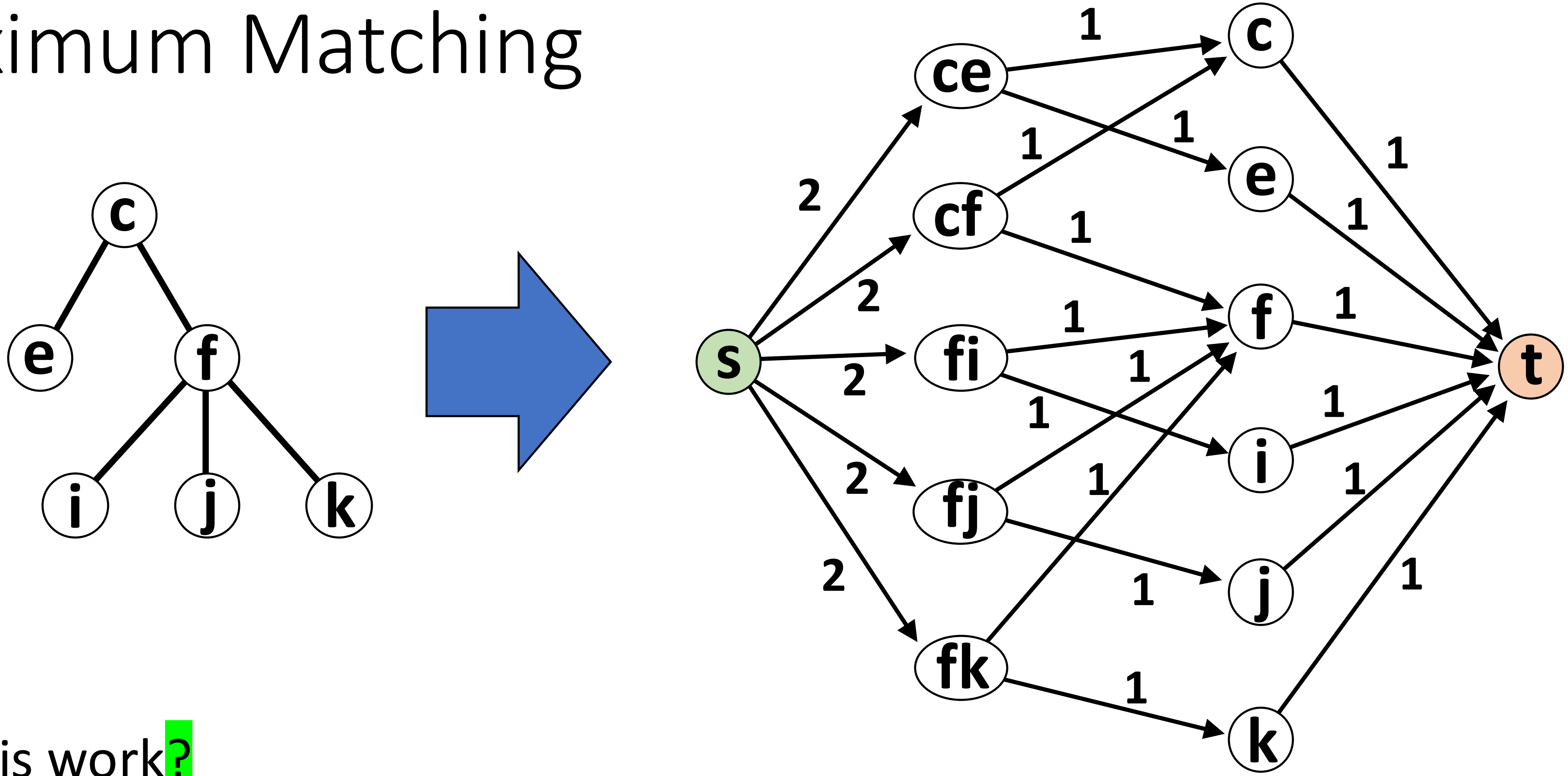
(A) Build Flow Network:

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.
3. Make edges go from **s**-connected node to **t**-connected node.
4. Make all edge capacities 1.

(B) Find Max Flow.

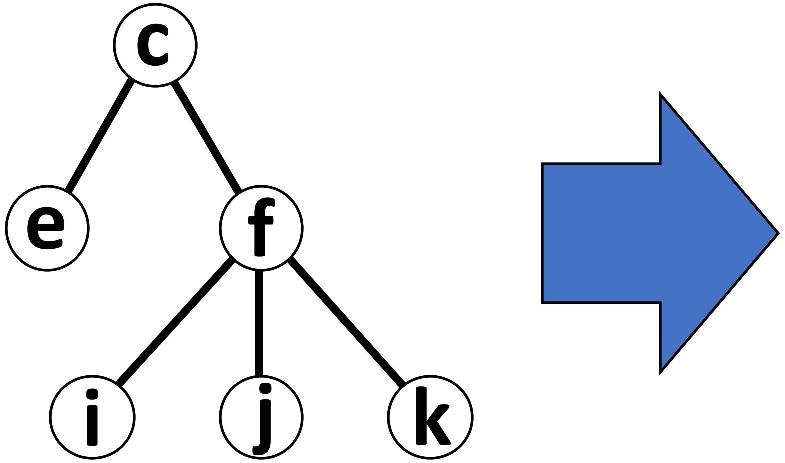
(C) If edge carries flow, select it.

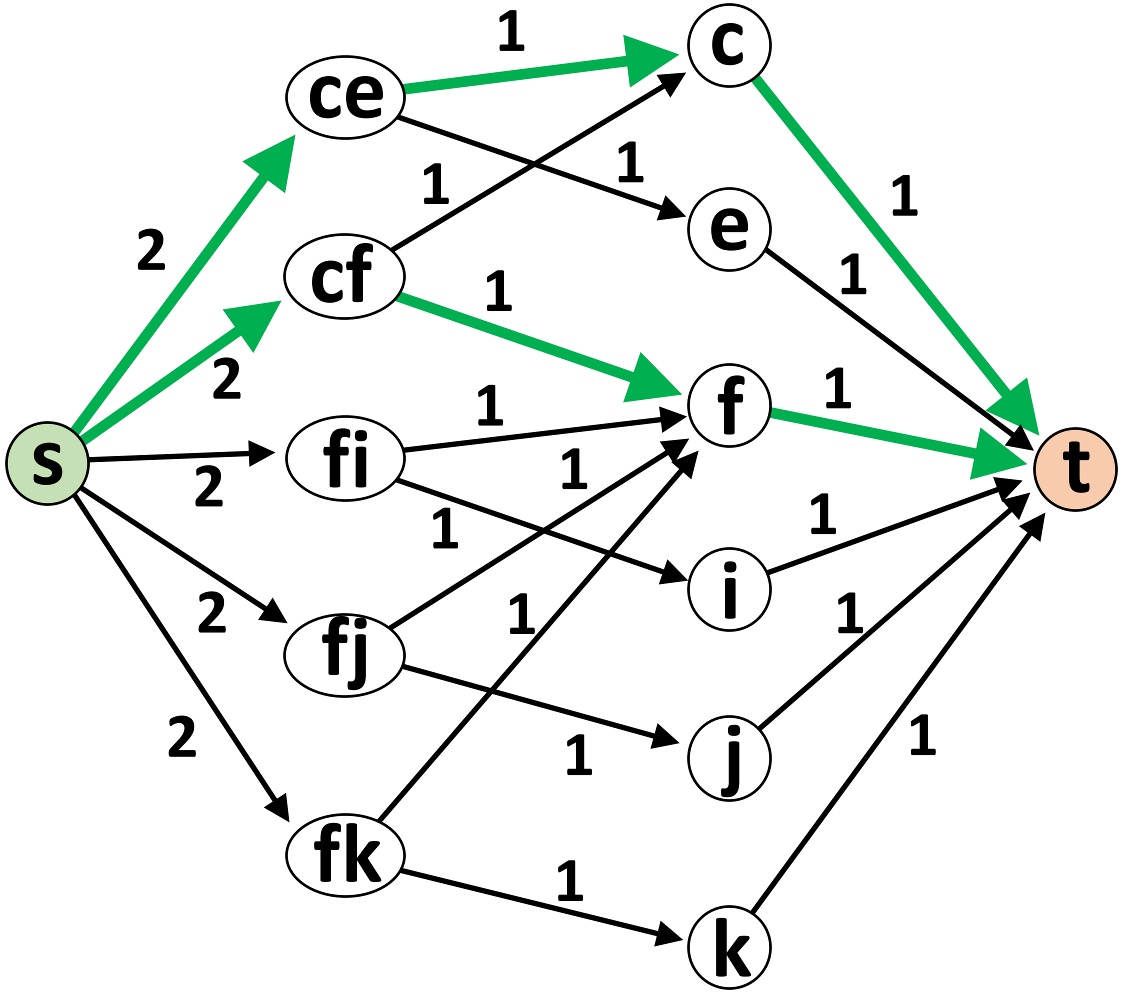
Maximum Matching



Does this work?

Maximum Matching

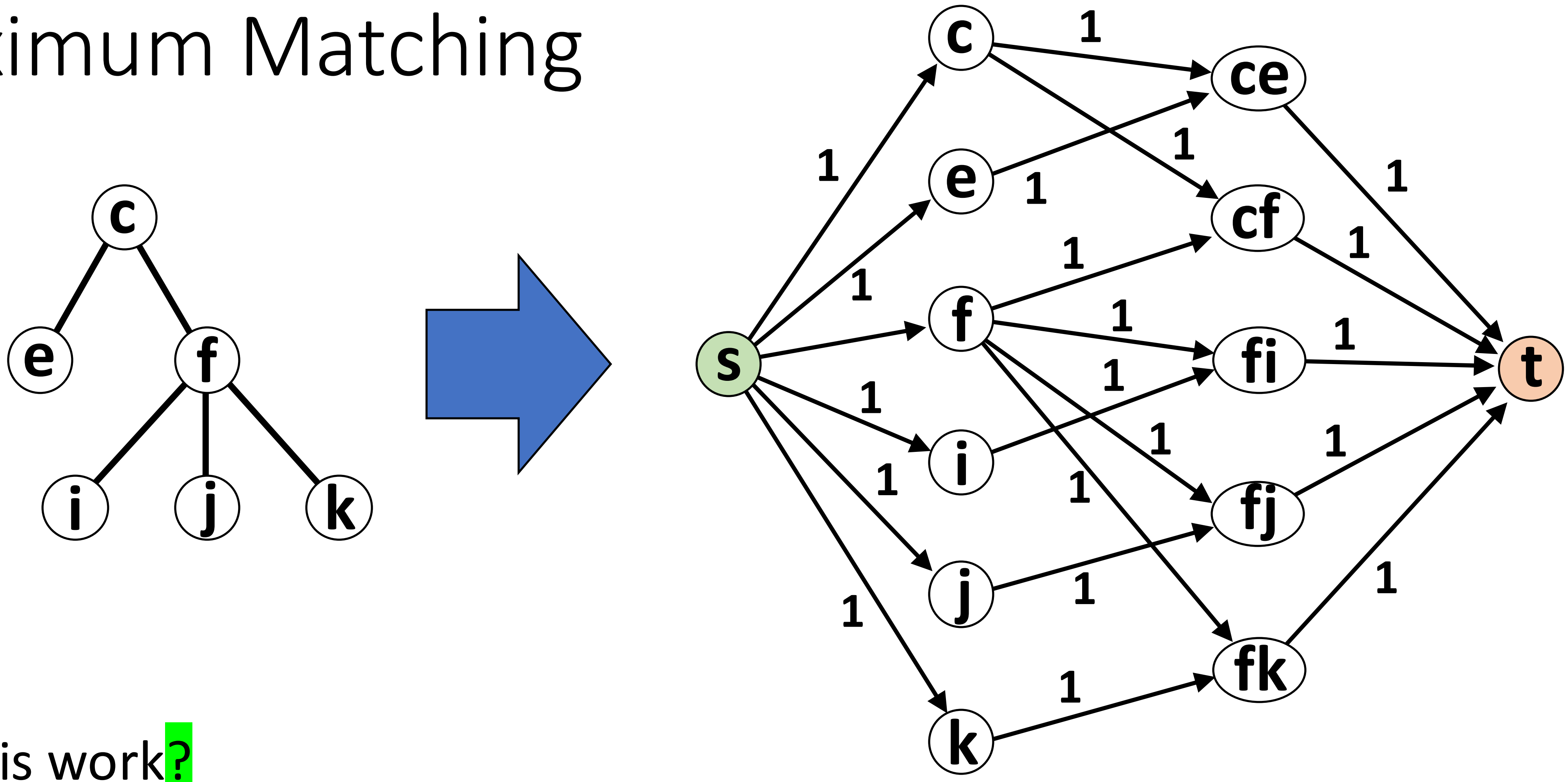




Does this work?

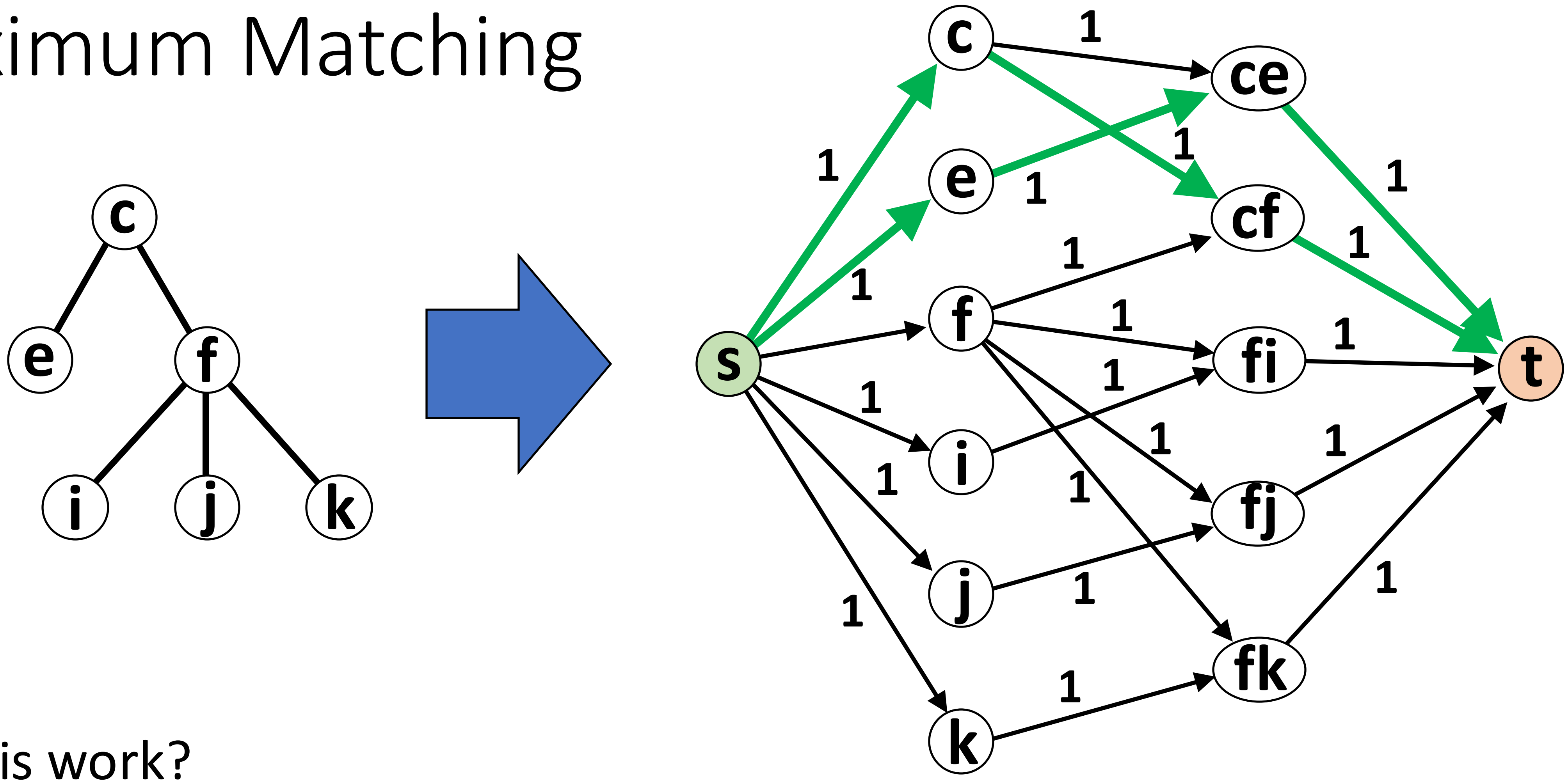
No, nothing forces edges with capacity 2 to host 2 units, so you can make some host 1 and select neighboring edges.

Maximum Matching



Does this work?

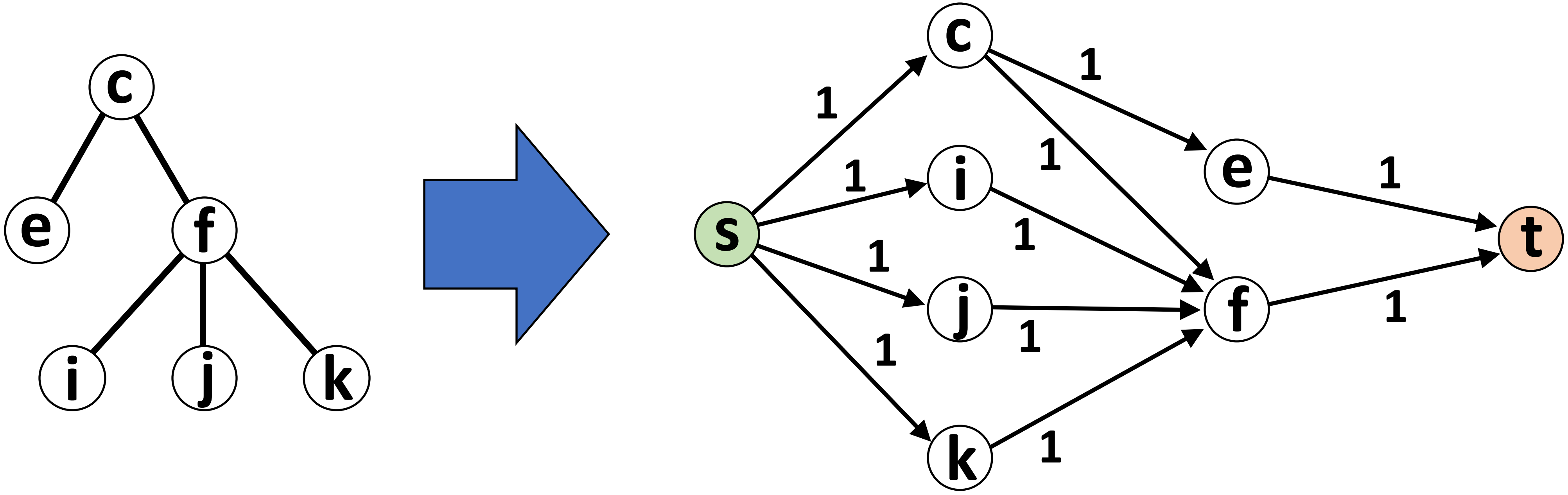
Maximum Matching



Does this work?

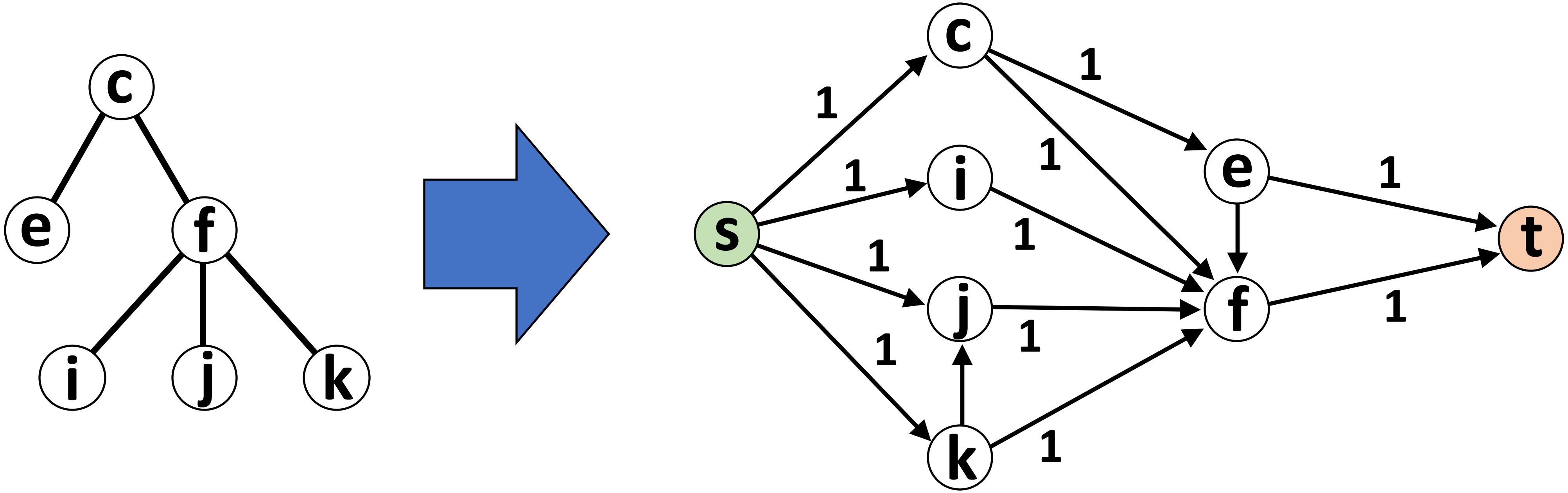
No, saturating a node prevents it from being used in other edges, but does not prevent other nodes from deploying neighboring edges.

Maximum Matching



Does this work?

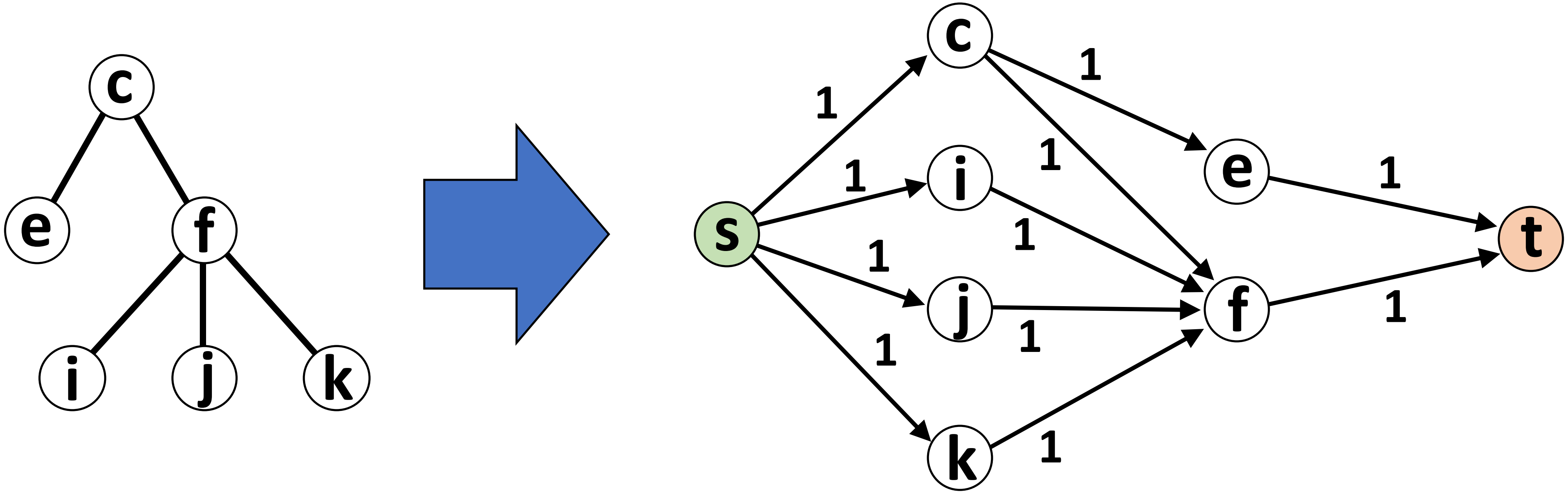
Maximum Matching



Does this work?

Yes, but it requires the graph to be bipartite (vertices can be partitioned into disjoint sets so that all edges cross between the sets).

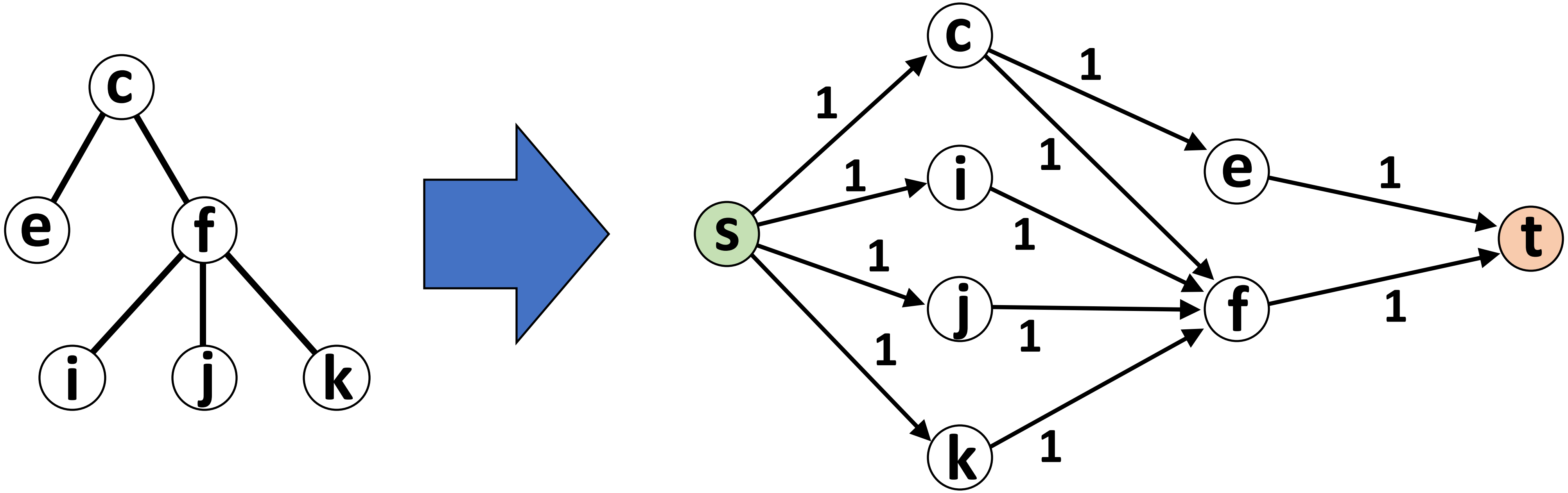
Maximum Matching



Does this work?

Yes, but it requires the graph to be bipartite (vertices can be partitioned into disjoint sets so that all edges cross between the sets). Are trees bipartite?

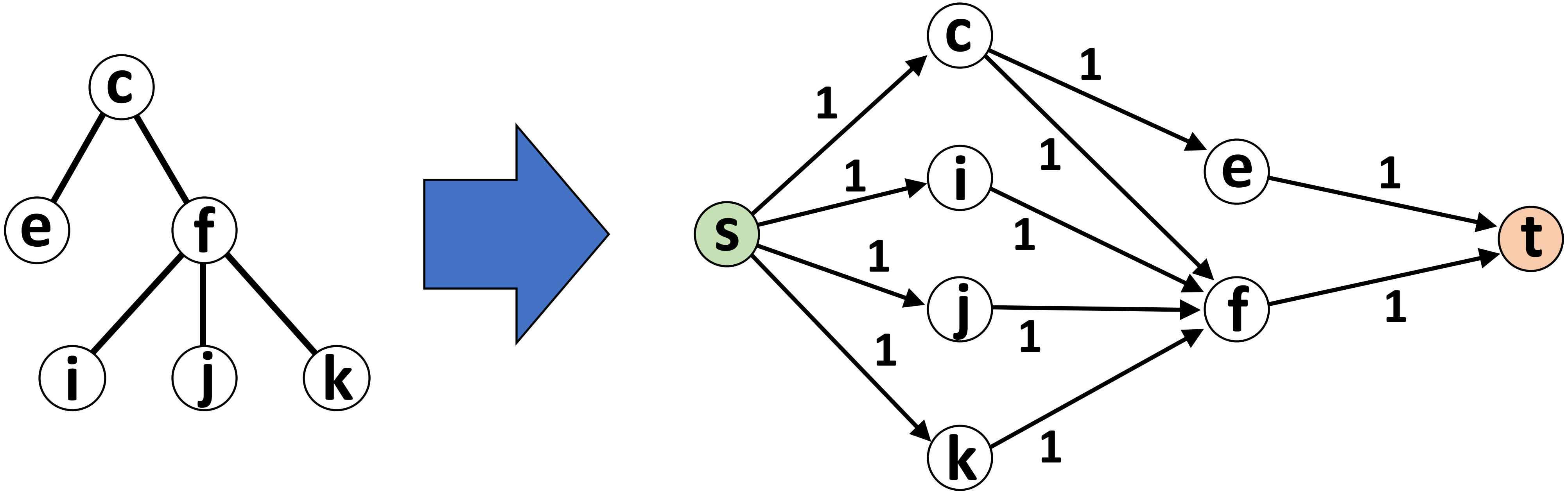
Maximum Matching



Does this work?

Yes, but it requires the graph to be bipartite (vertices can be partitioned into disjoint sets so that all edges cross between the sets). Are trees bipartite? YES (otherwise there would be a cycle).

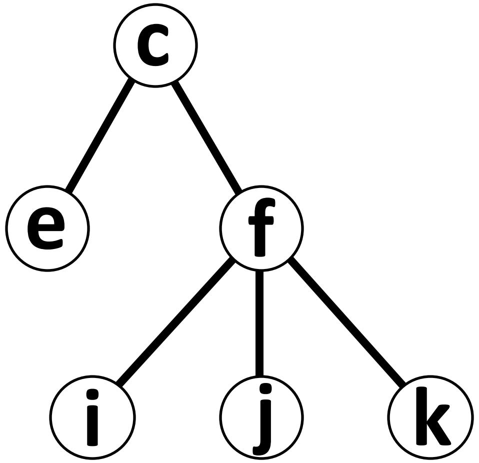
Maximum Matching

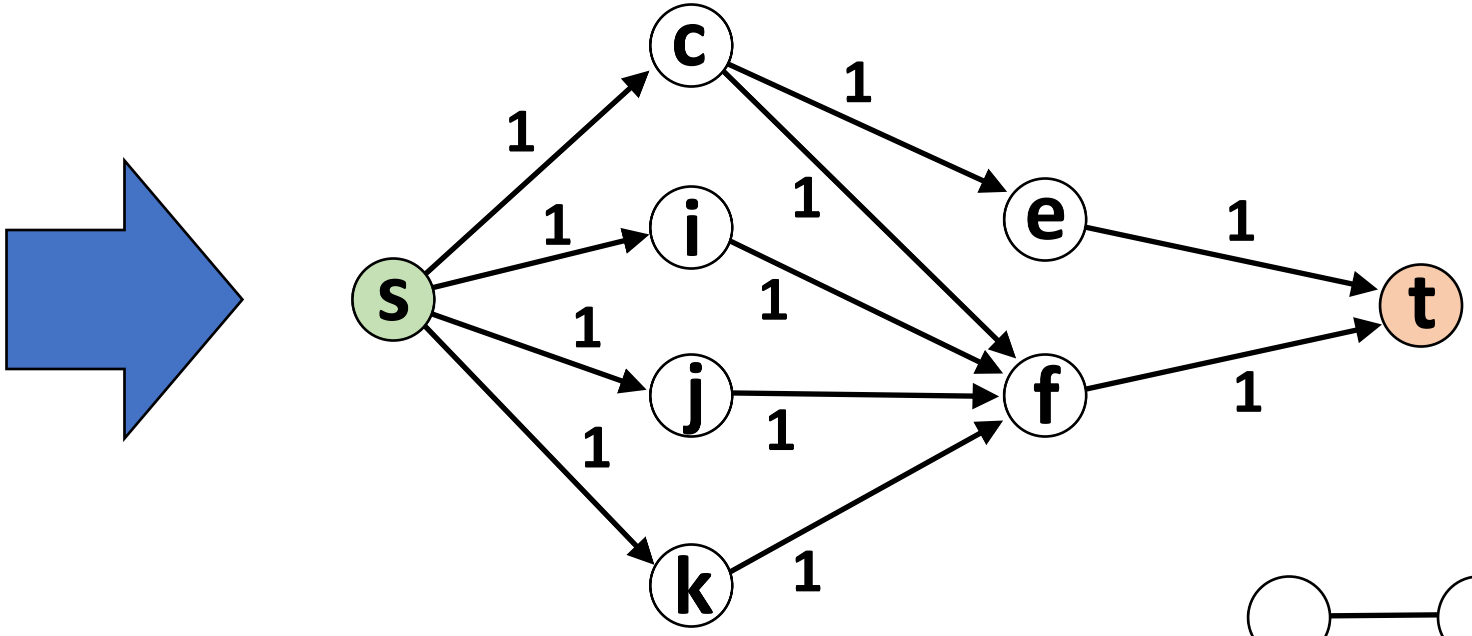


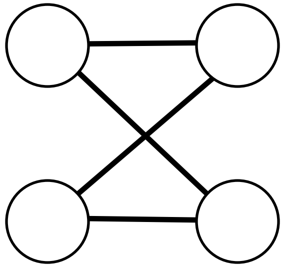
Does this work?

Yes, but it requires the graph to be bipartite (vertices can be partitioned into disjoint sets so that all edges cross between the sets). Are trees bipartite? YES (otherwise there would be a cycle). Are bipartite graphs always trees?

Maximum Matching



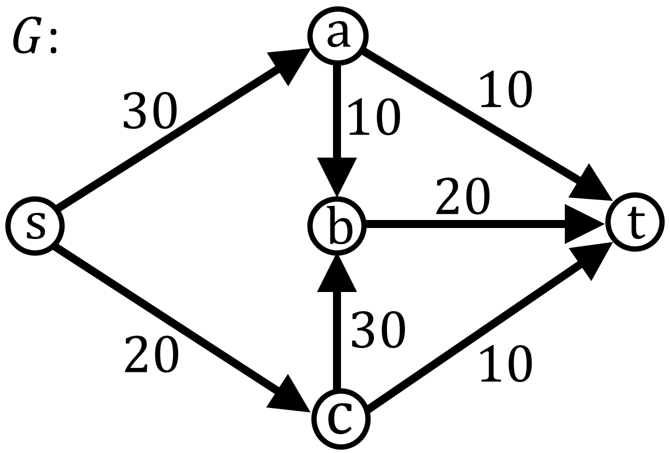


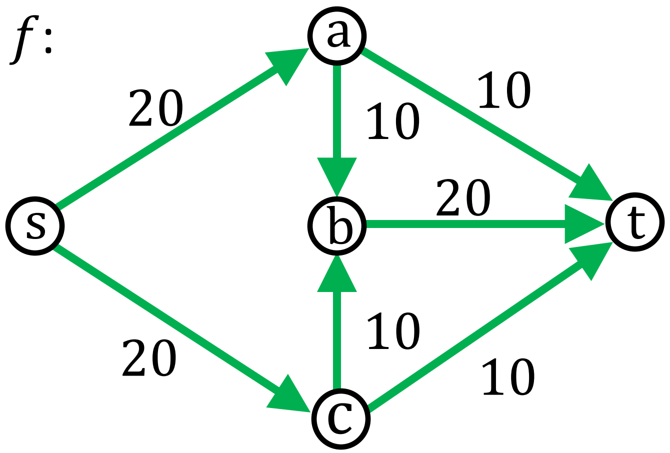


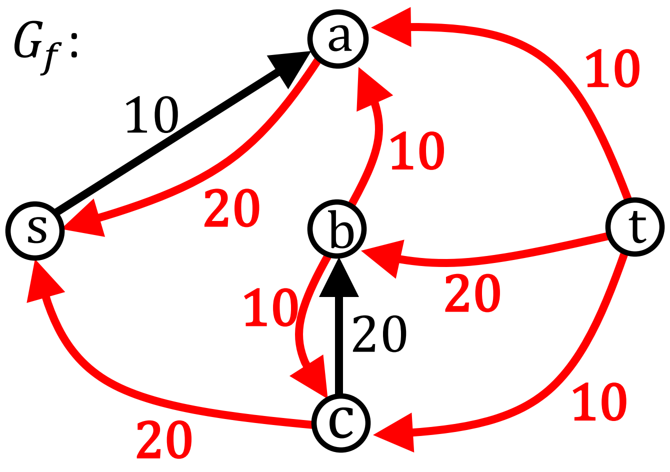
Does this work?

Yes, but it requires the graph to be bipartite (vertices can be partitioned into disjoint sets so that all edges cross between the sets). Are trees bipartite? YES (otherwise there would be a cycle). Are bipartite graphs always trees? NO.

Ford-Fulkerson








```
Max-Flow(G)
   f(e) = 0 for all
   while s-t path in
      P = simple s-t
      f'= augment(f,
      f = f'
      Gf = Gf'
   return f
```

Need to show:

1. ~~Validity.~~
2. ~~Running time.~~
3. Finds max flow.

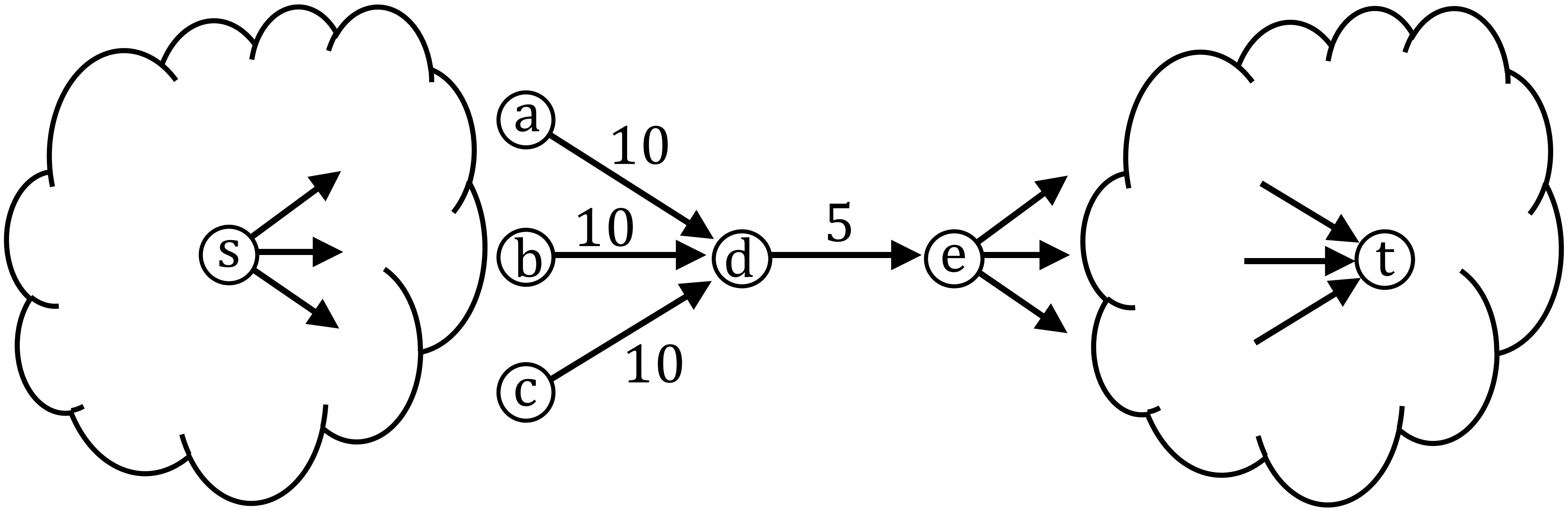
```
augment(f, P)
   b  bottleneck(P,f)
   each edge (u, v) in P
     f (u, v) is a back edge
         f((v, u)) -= b
     se
         f((u, v)) += b
   return f
```

Optimality

Theorem: The flow returned by the Ford-Fulkerson algorithm is a maximum flow.

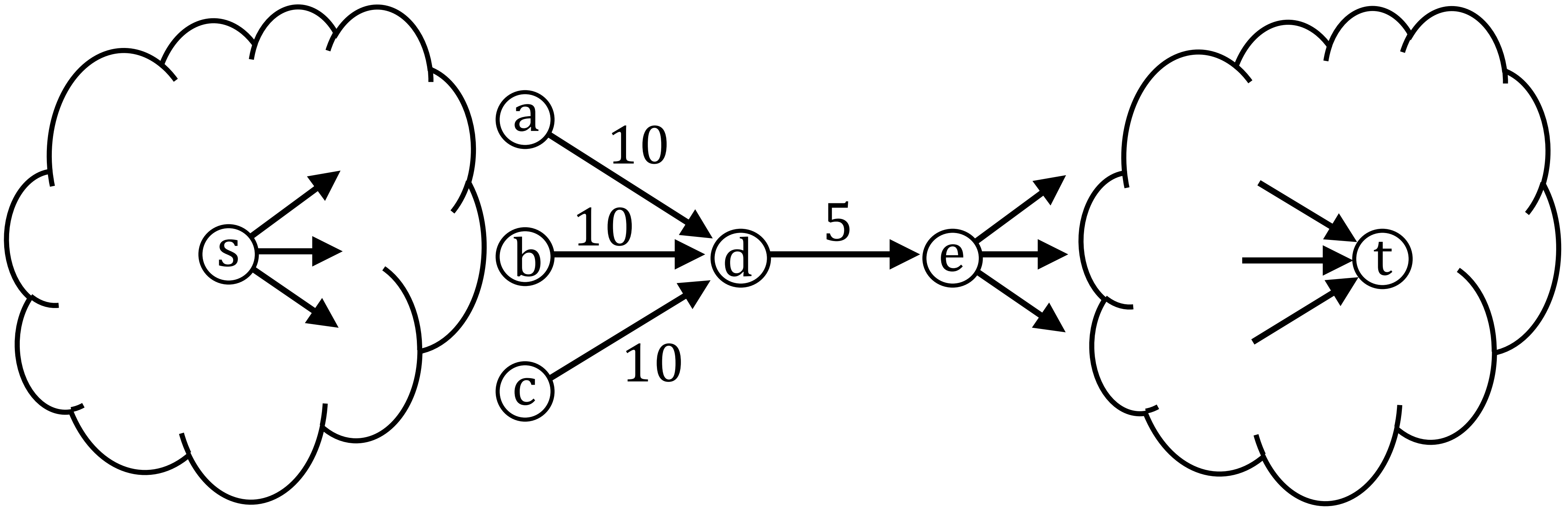
Proof: ...

Optimality



What can we say about the maximum flow on this network?

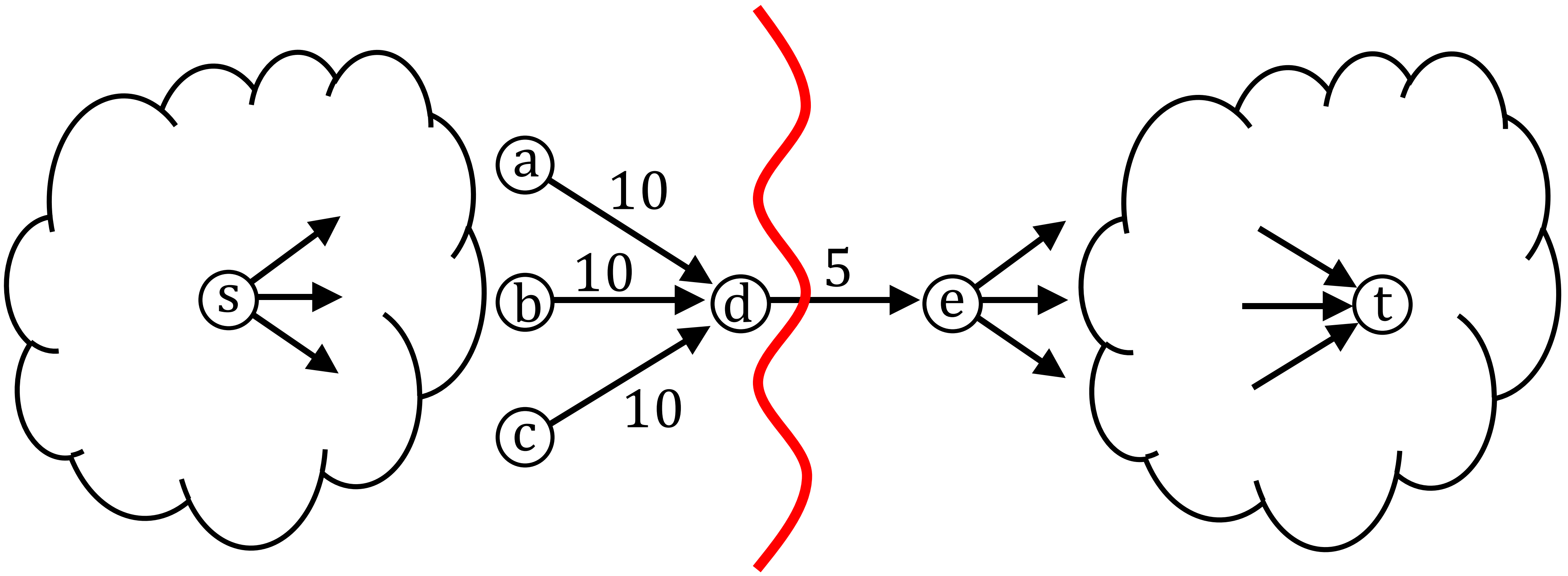
Optimality



What can we say about the maximum flow on this network?

It's not larger than 5.

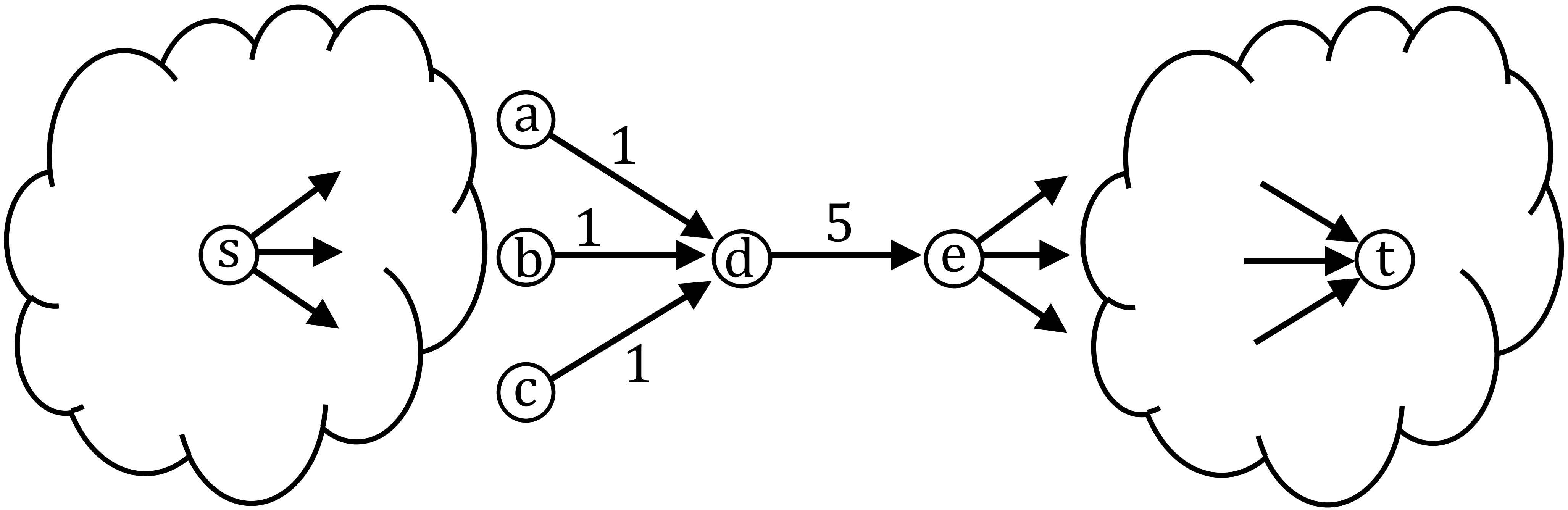
Optimality



What can we say about the maximum flow on this network?

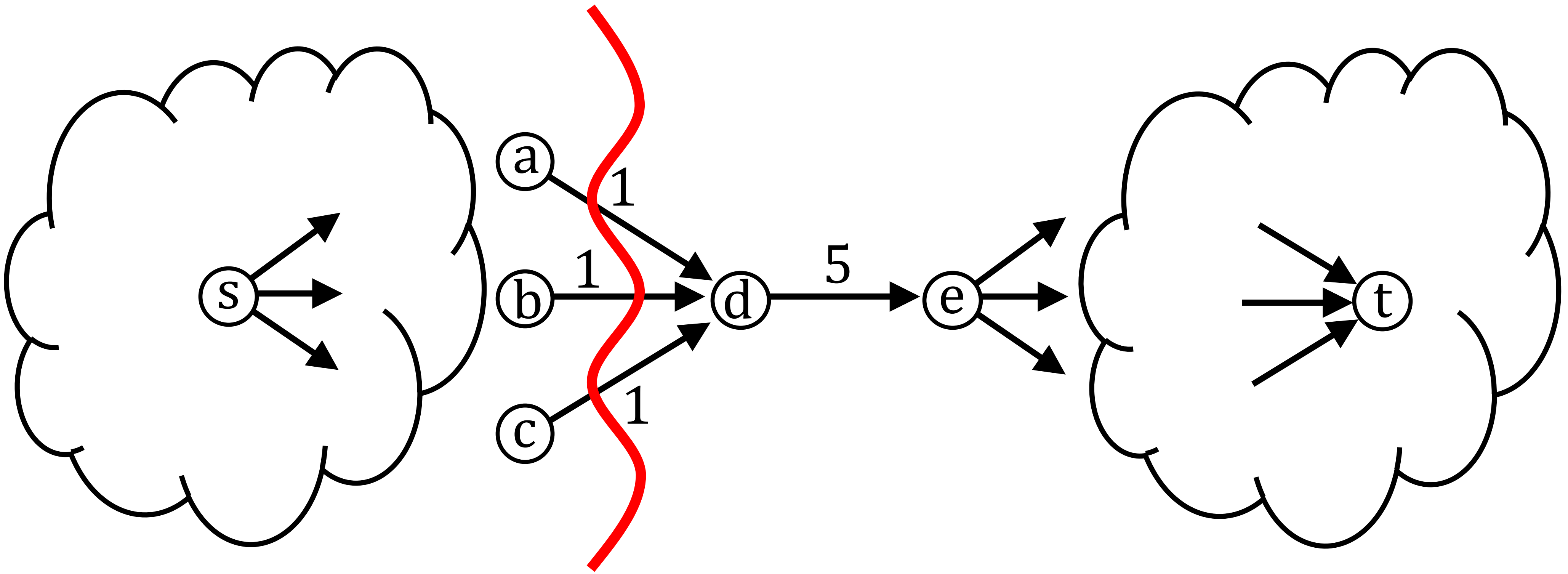
It's not larger than 5.

Optimality



What can we say about the maximum flow on this network?

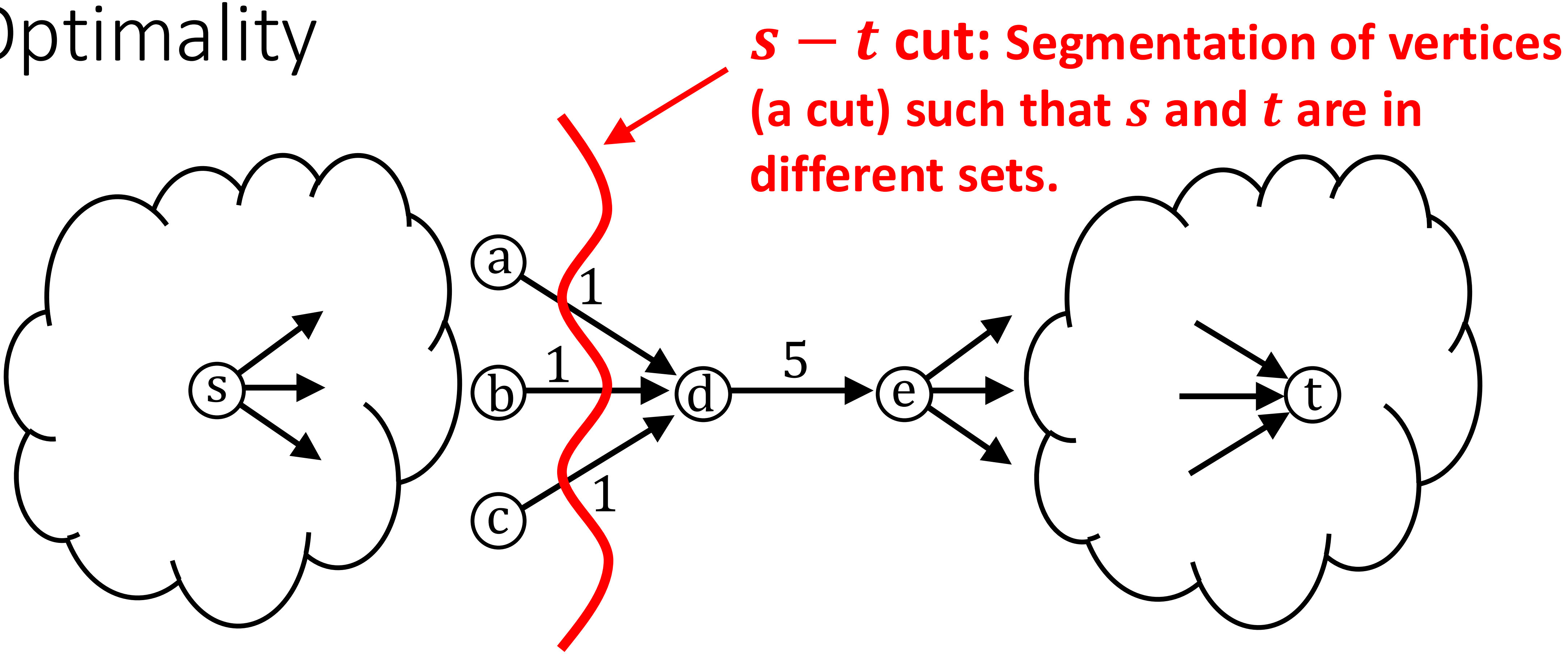
Optimality



What can we say about the maximum flow on this network?

It's not larger than 3.

Optimality

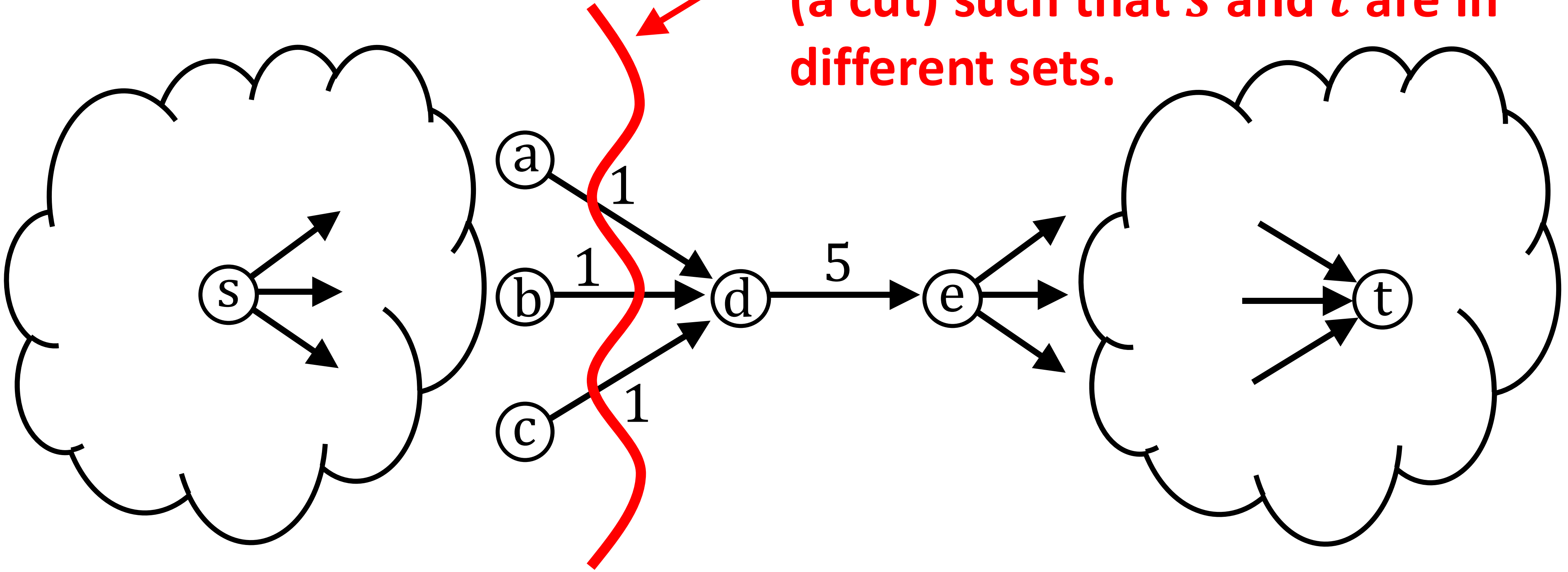


What can we say about the maximum flow on this network?

It's not larger than 3.

Optimality

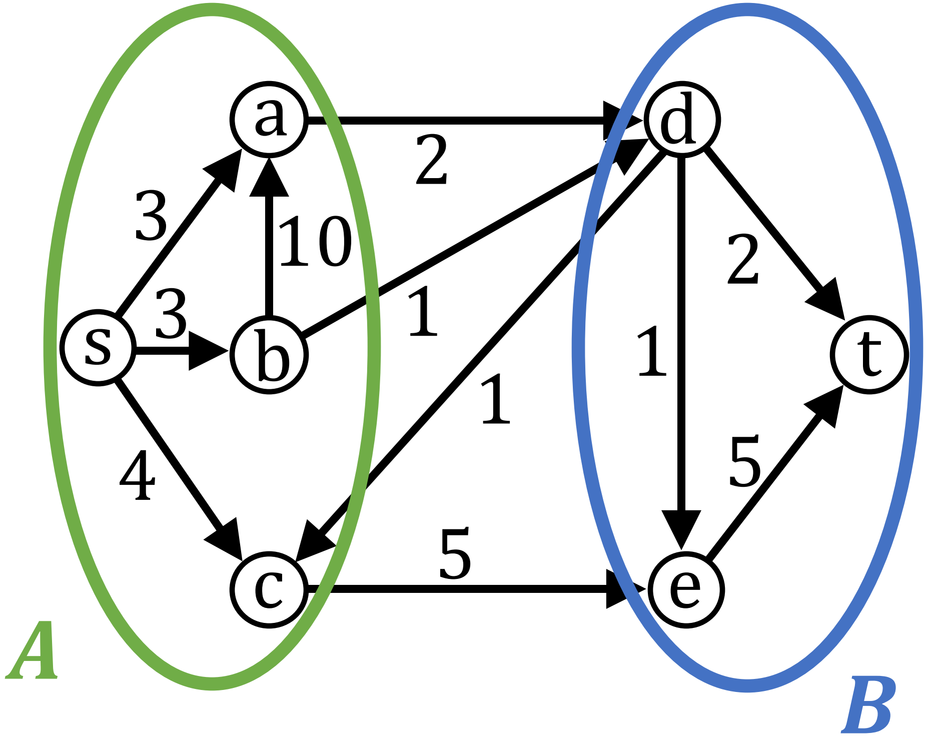
$s-t$ cut: Segmentation of vertices (a cut) such that s and t are in different sets.



The capacity of a cut is the sum of the capacities leaving s's set.

$s - t$ cuts

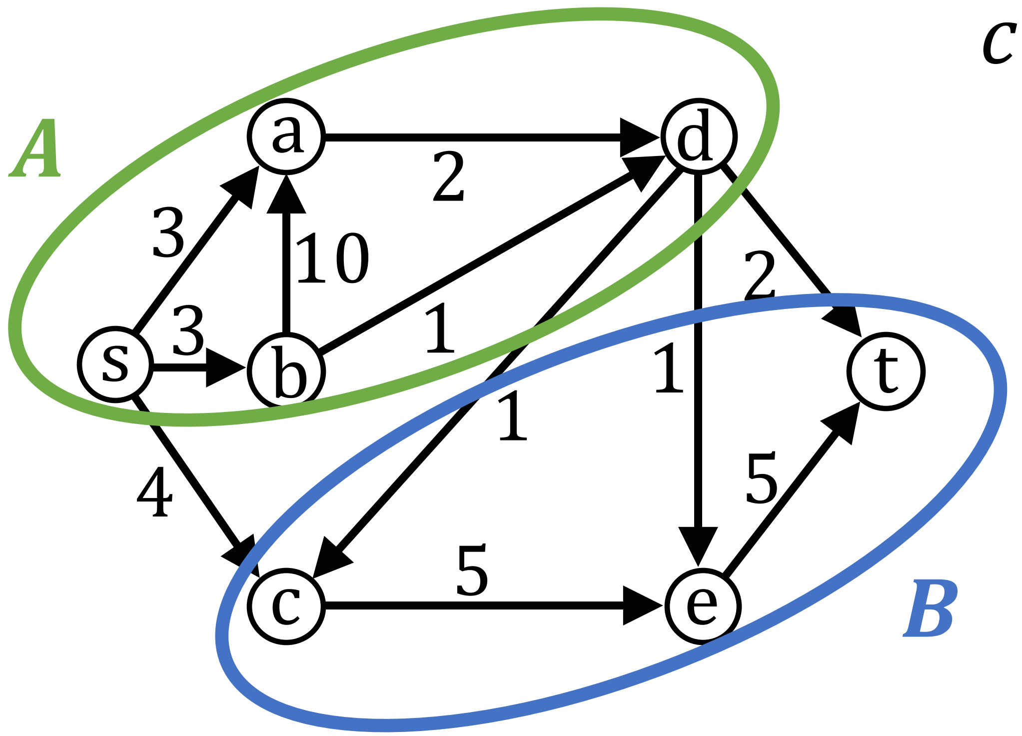
<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, $\boldsymbol{A}$ and $\boldsymbol{B}$, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.



$$c(A, B) = 8$$

$s - t$ cuts

Definitions: Suppose G is a flow network and nodes in G are divided into two sets, ***A*** and ***B***, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.

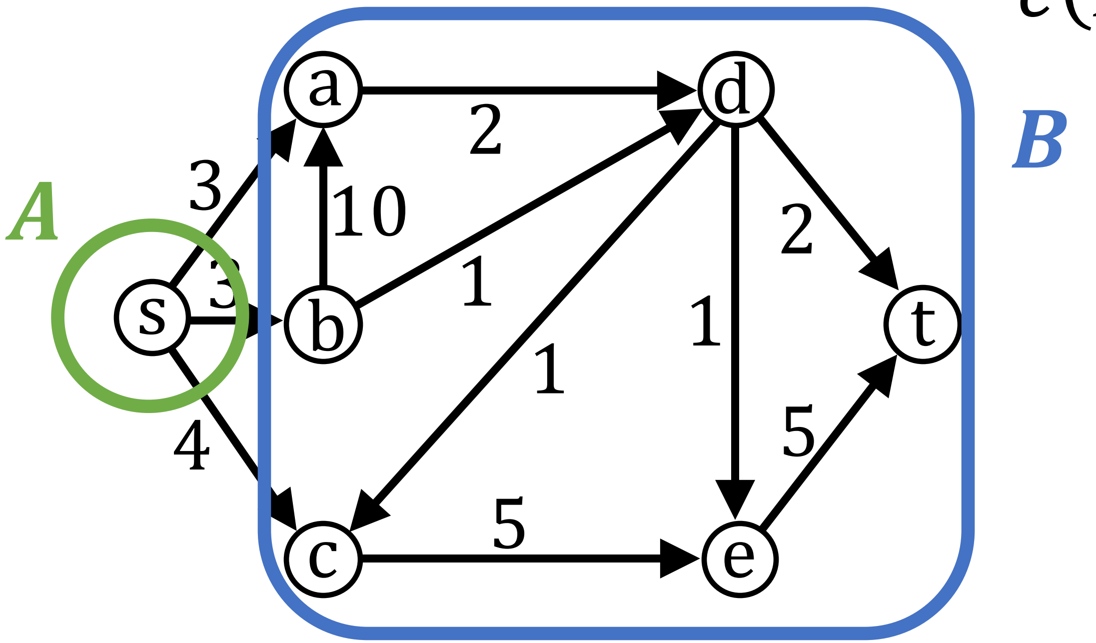


$$c(A, B) = 8$$

$s - t$ cuts

<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, ***A*** and ***B***, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.

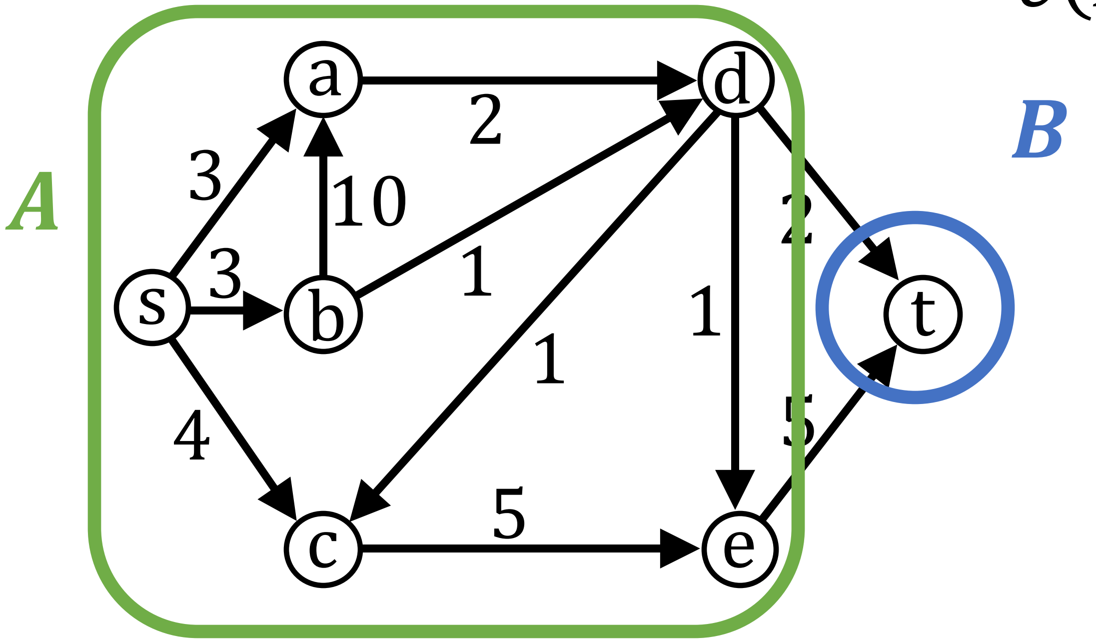
$$c(A, B) = 10$$



$s - t$ cuts

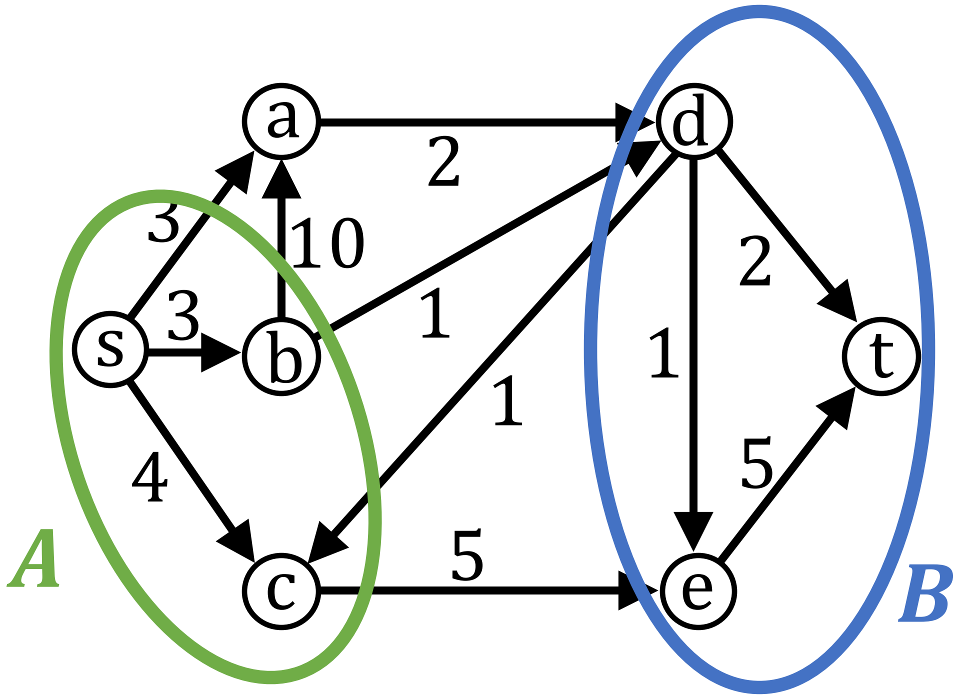
<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, ***A*** and ***B***, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.

$$c(A, B) = 7$$



$s - t$ cuts

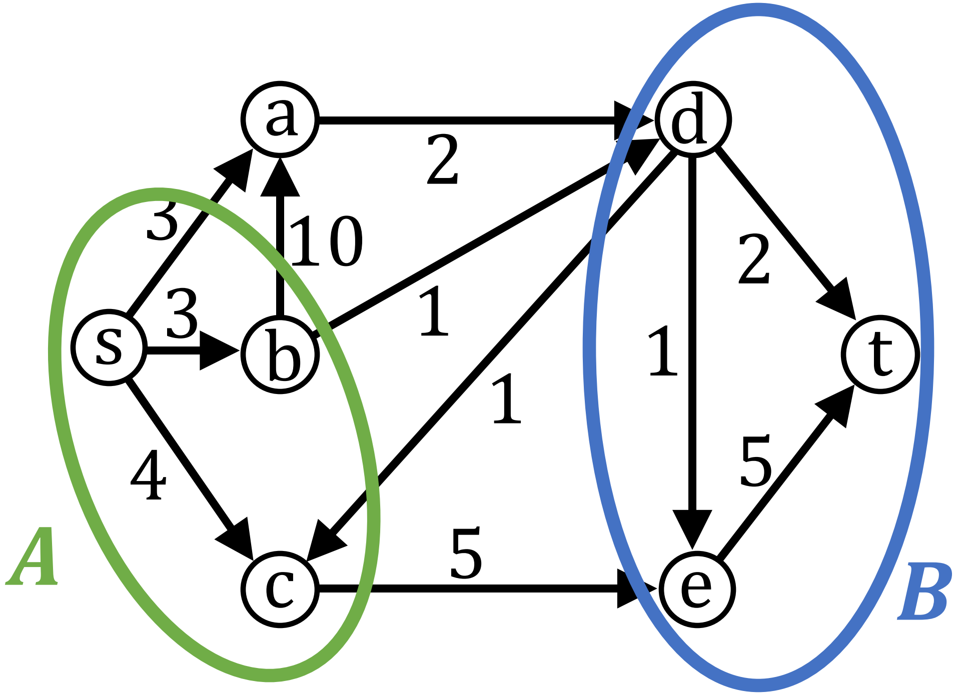
<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, $\boldsymbol{A}$ and $\boldsymbol{B}$, such that $s \in A$ and $t \in B$. We call (A, B) an *$s - t$ cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.



$$c(A, B) = ??$$

$s-t$ cuts

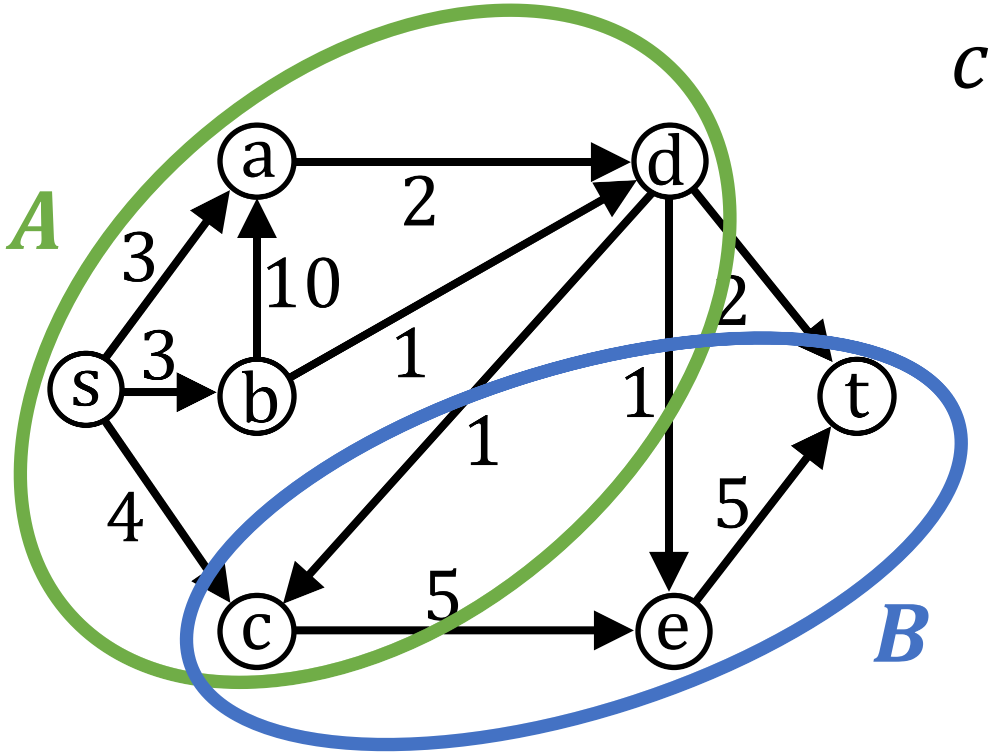
Definitions: Suppose G is a flow network and nodes in G are divided into two sets, ***A*** and ***B***, such that $s \in A$ and $t \in B$. We call (A, B) an $s-t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.



Invalid cut! Every vertex needs to be is in one of the sets!

$s - t$ cuts

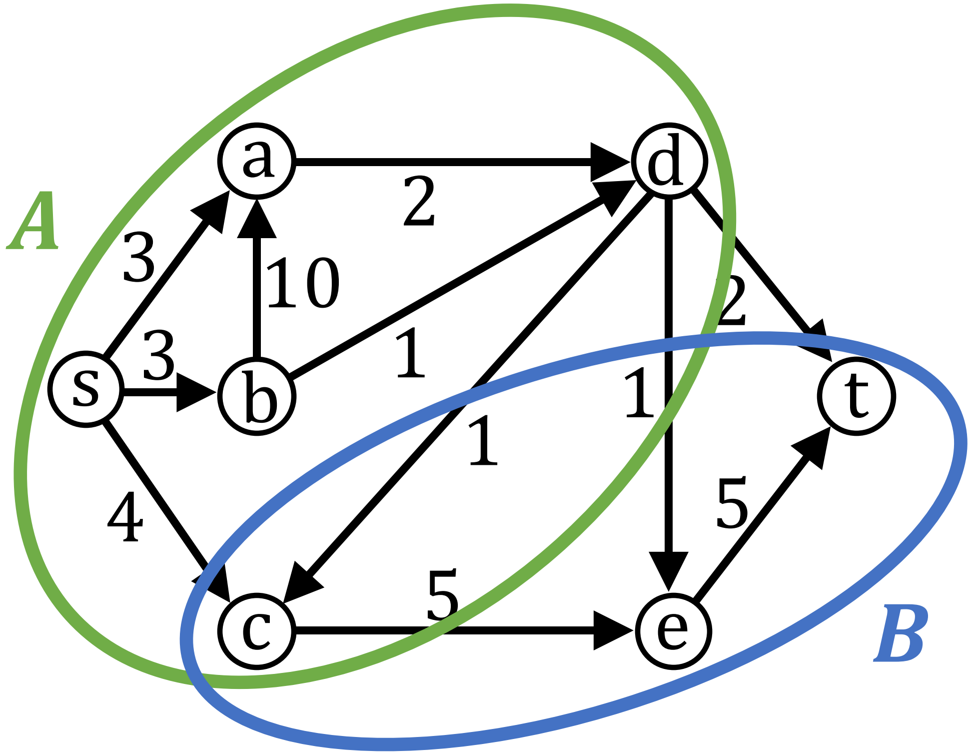
<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, ***A*** and ***B***, such that $s \in A$ and $t \in B$. We call (A, B) an *$s - t$ cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.



$$c(A, B) = ??$$

$s - t$ cuts

<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, $\boldsymbol{A}$ and $\boldsymbol{B}$, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.

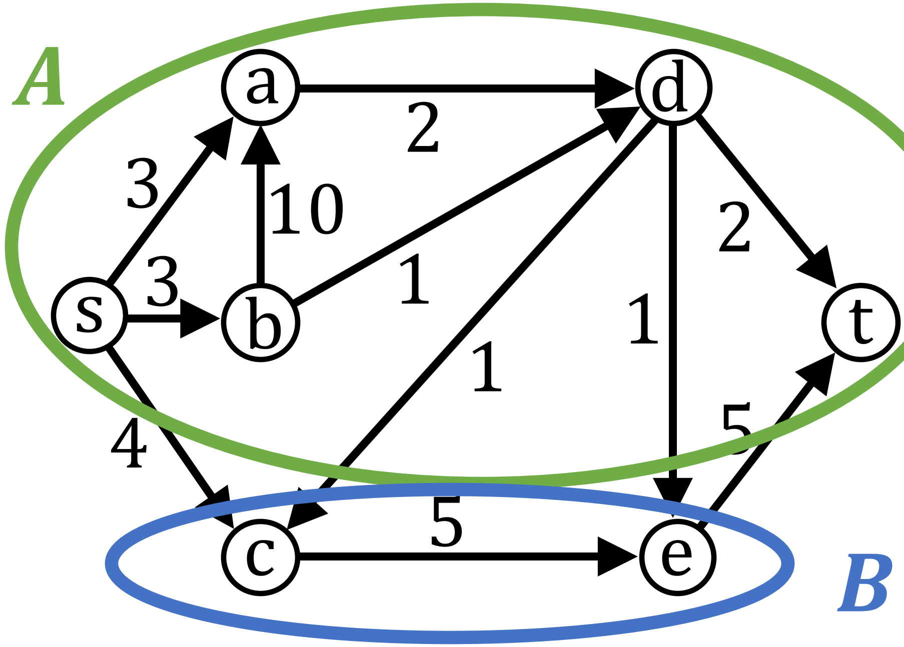


Invalid cut! Every vertex needs to be in <u>exactly</u> one of the sets!

$s - t$ cuts

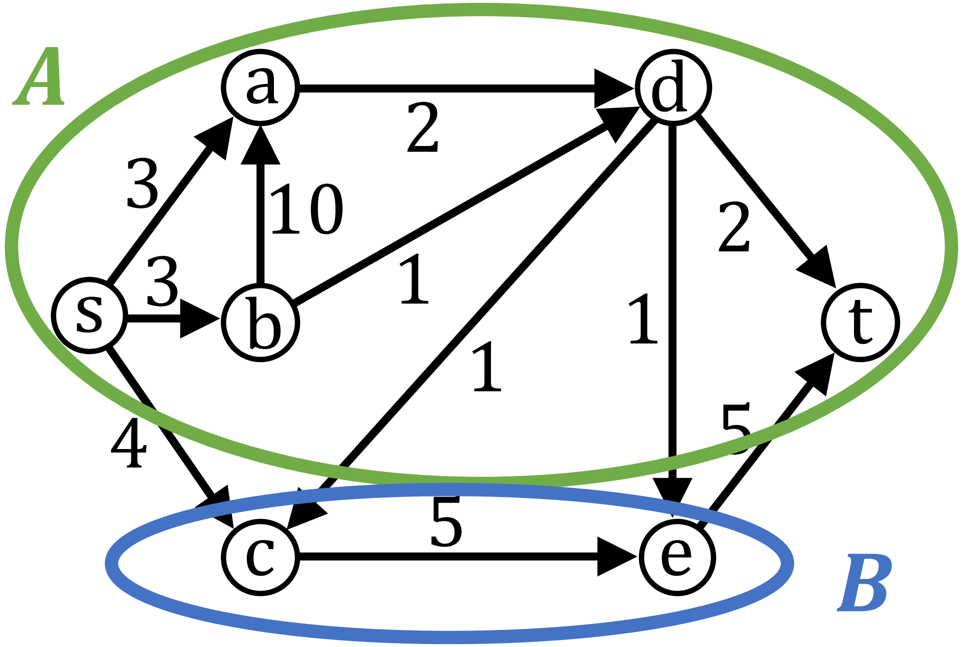
Definitions: Suppose G is a flow network and nodes in G are divided into two sets, ***A*** and ***B***, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.

$$c(A, B) = ??$$



$s - t$ cuts

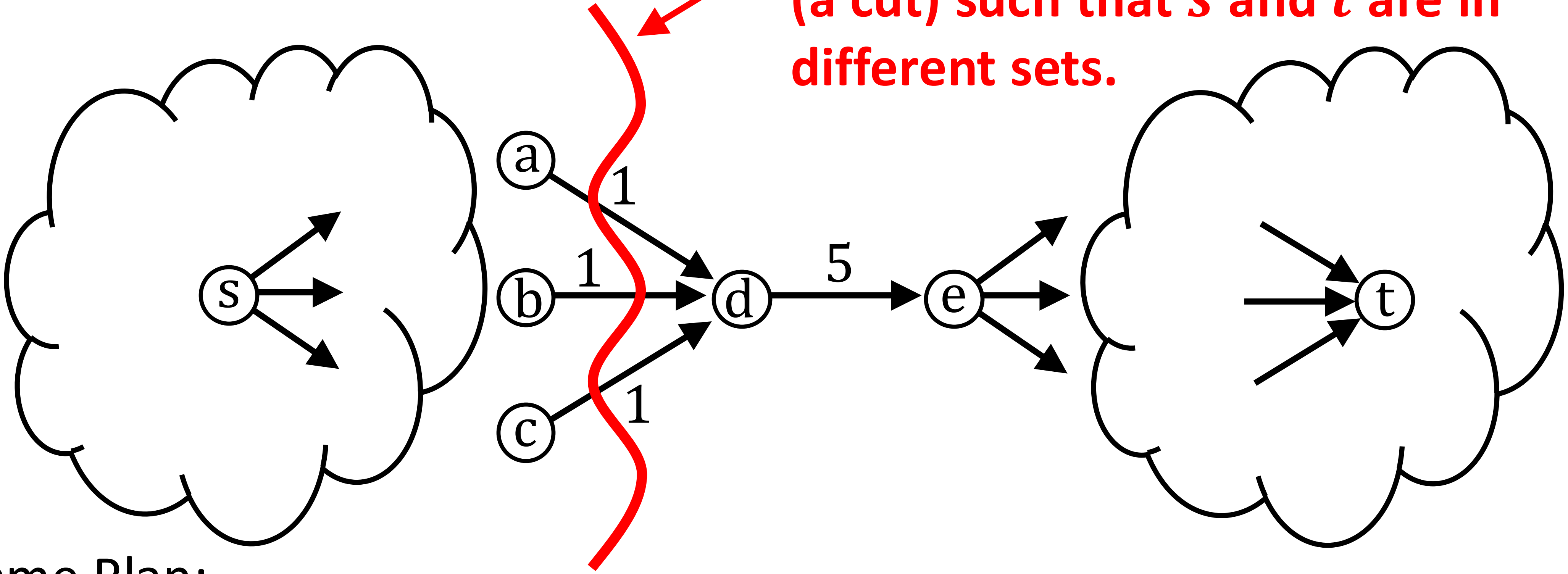
<u>Definitions:</u> Suppose G is a flow network and nodes in G are divided into two sets, $\boldsymbol{A}$ and $\boldsymbol{B}$, such that $s \in A$ and $t \in B$. We call (A, B) an $s - t$ *cut*. The *capacity* of the cut, $c(A, B)$, is the sum of capacities of all edges out of A.



Invalid $s - t$ cut! s and t need to be in different sets!

Optimality

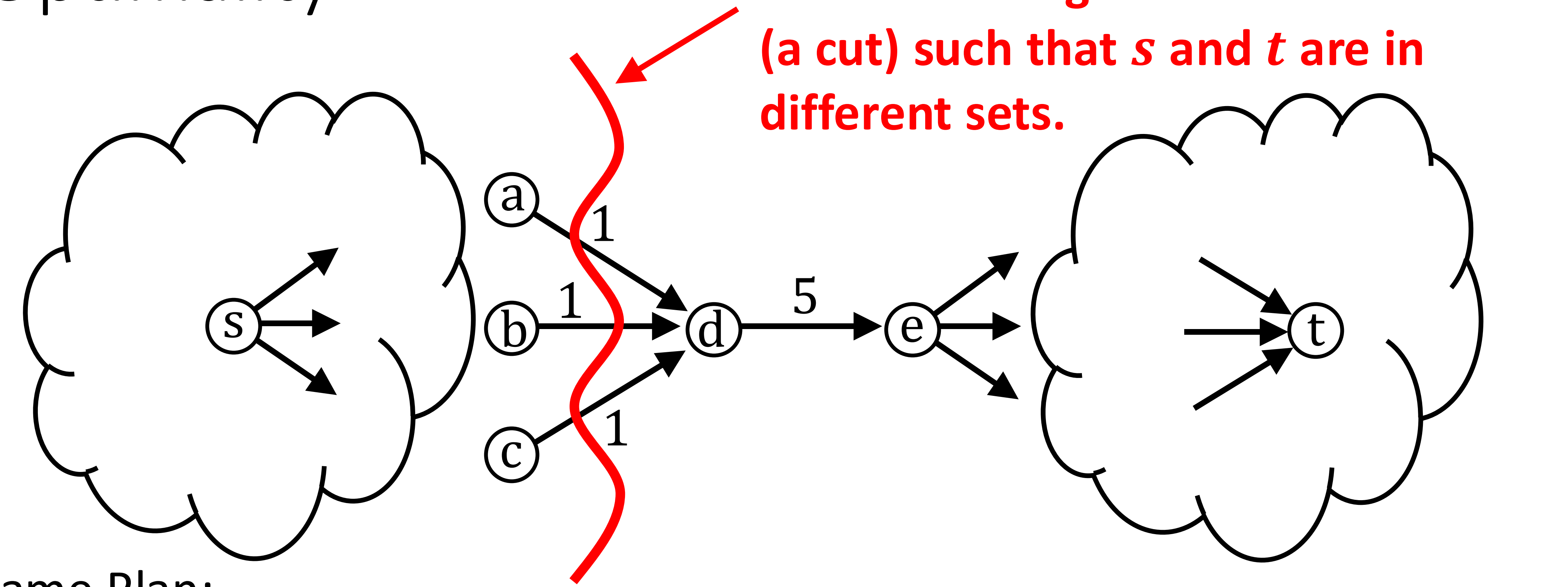
$s - t$ **cut: Segmentation of vertices (a cut) such that s and t are in different sets.**



Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.

Optimality

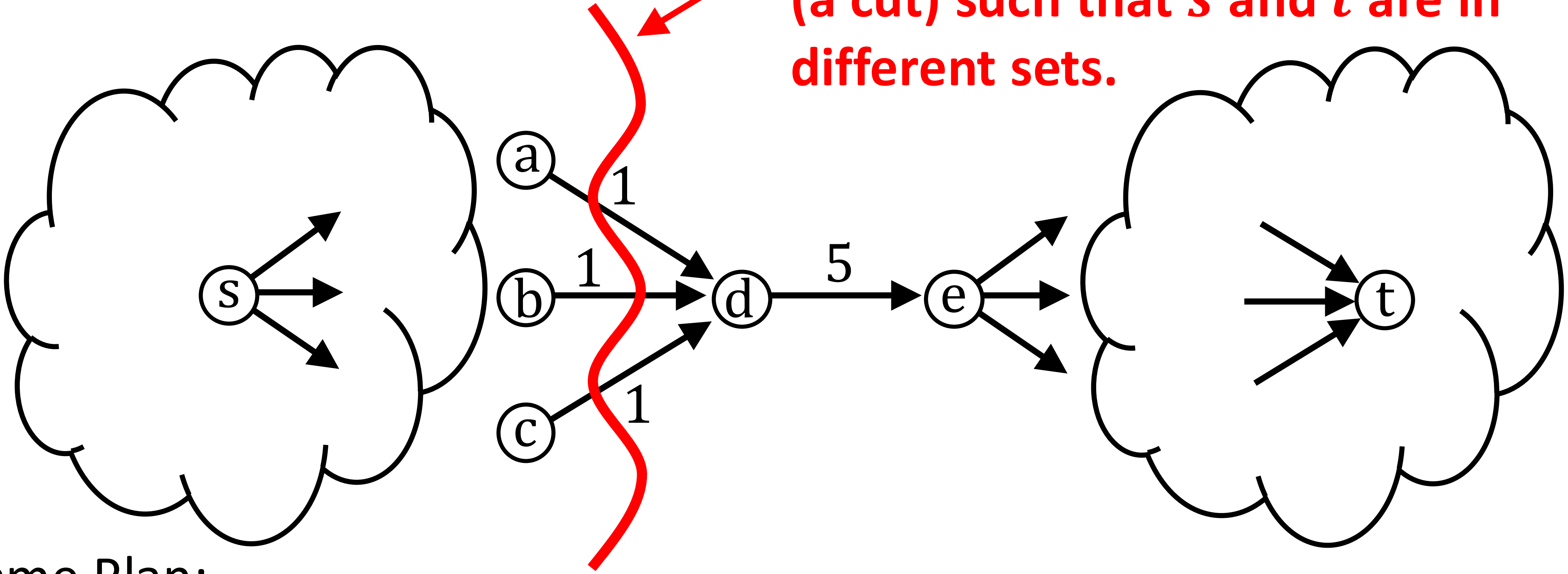


Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.

Optimality

$s - t$ cut: Segmentation of vertices (a cut) such that s and t are in different sets.

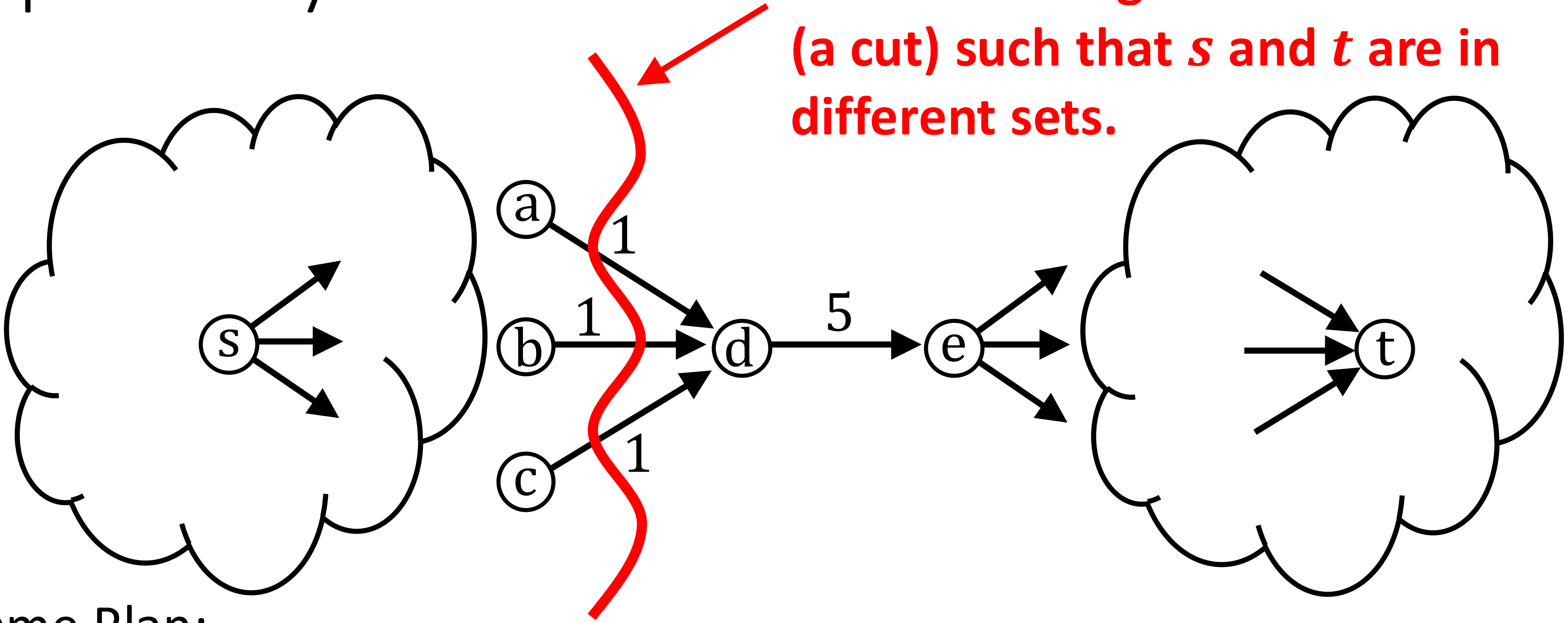


Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.

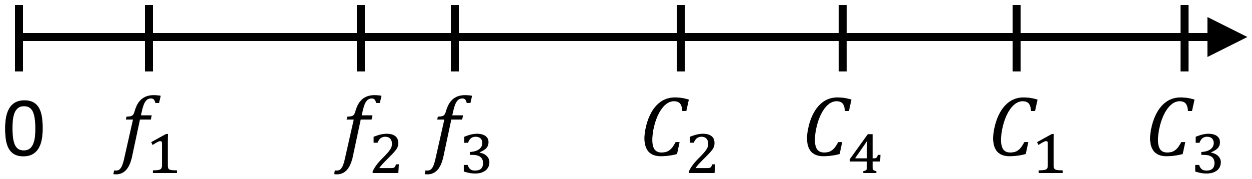
Optimality

$s-t$ cut: Segmentation of vertices (a cut) such that s and t are in different sets.



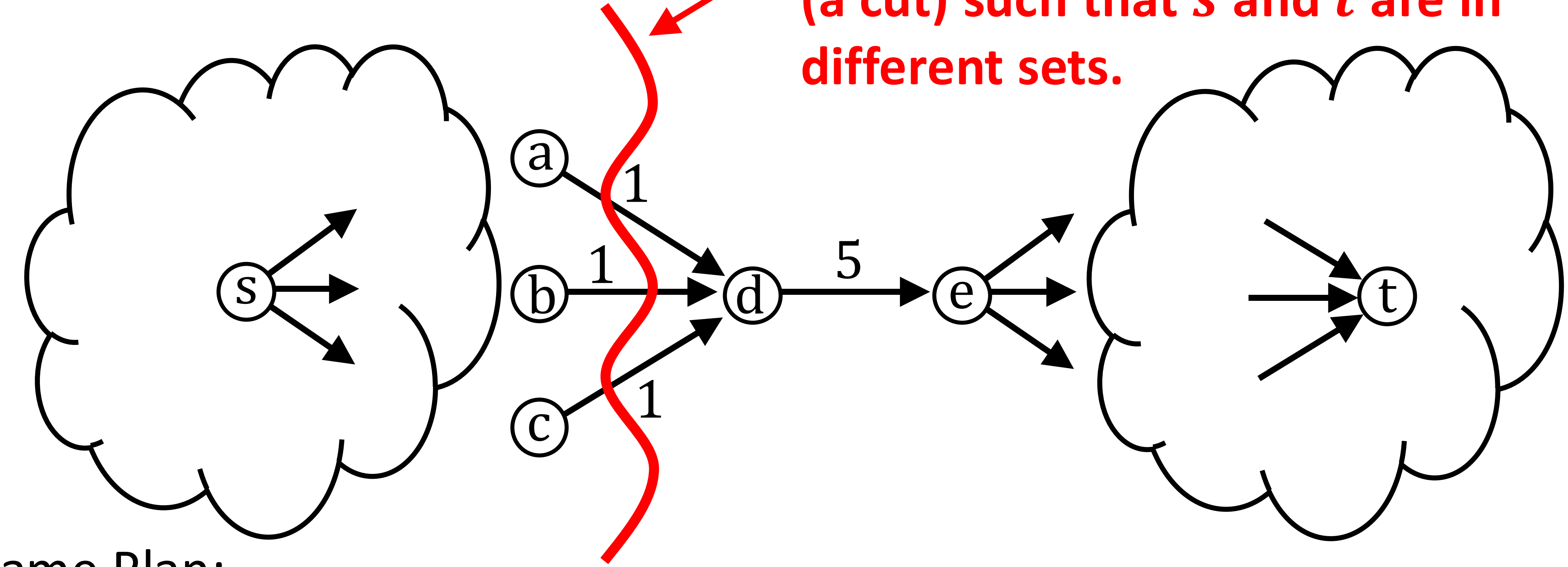
Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.



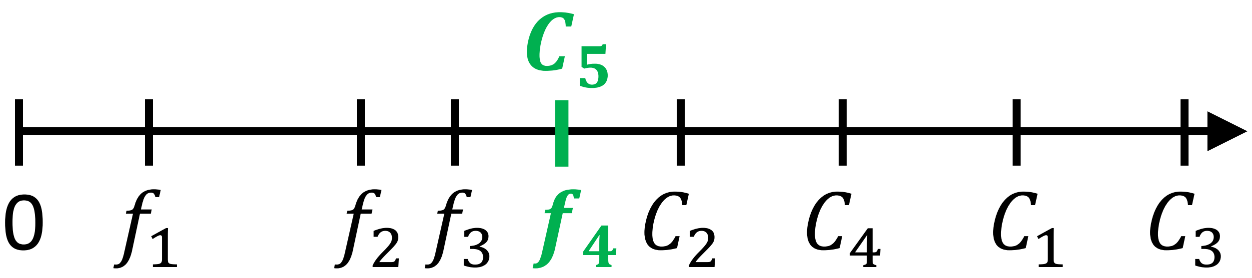
Optimality

$s - t$ cut: Segmentation of vertices (a cut) such that s and t are in different sets.

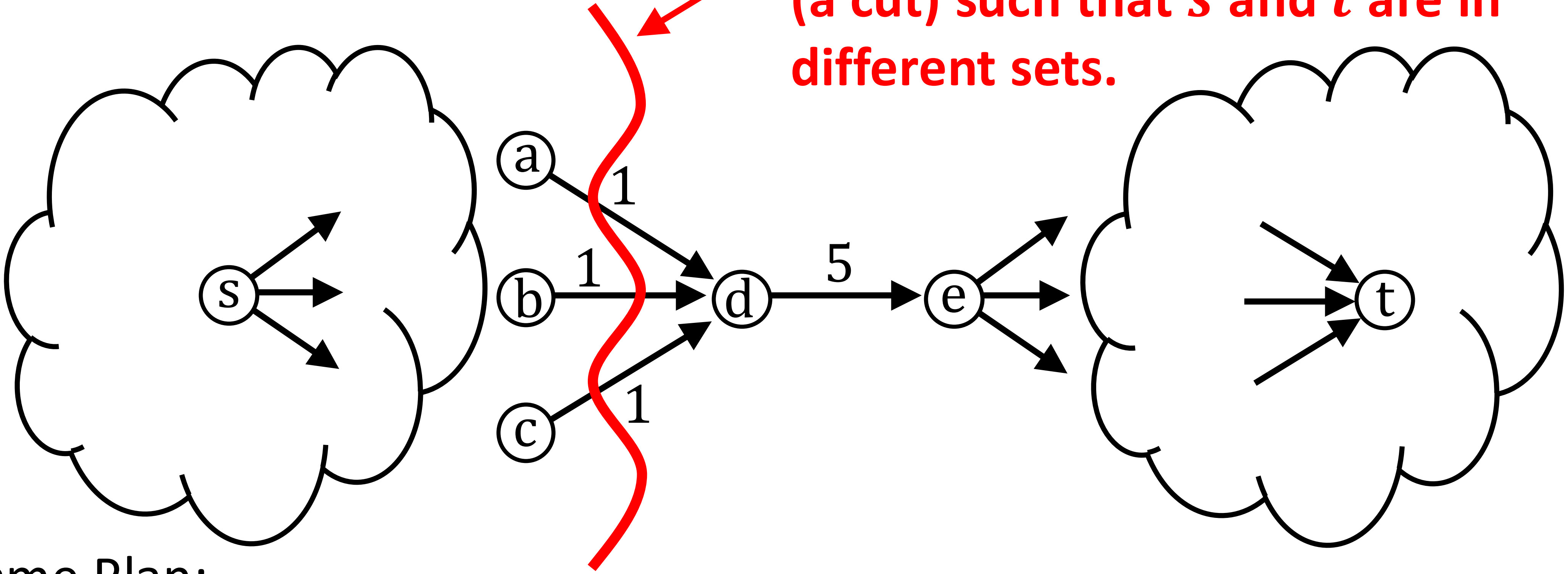


Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.



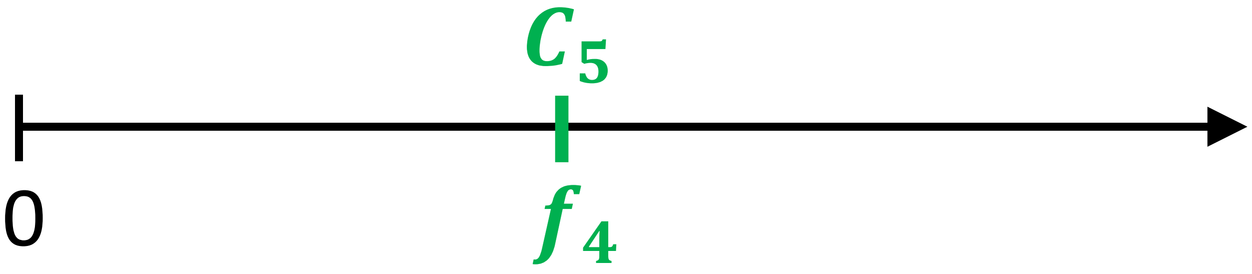
Optimality



$s-t$ **cut: Segmentation of vertices (a cut) such that** s **and** t **are in different sets.**

Game Plan:

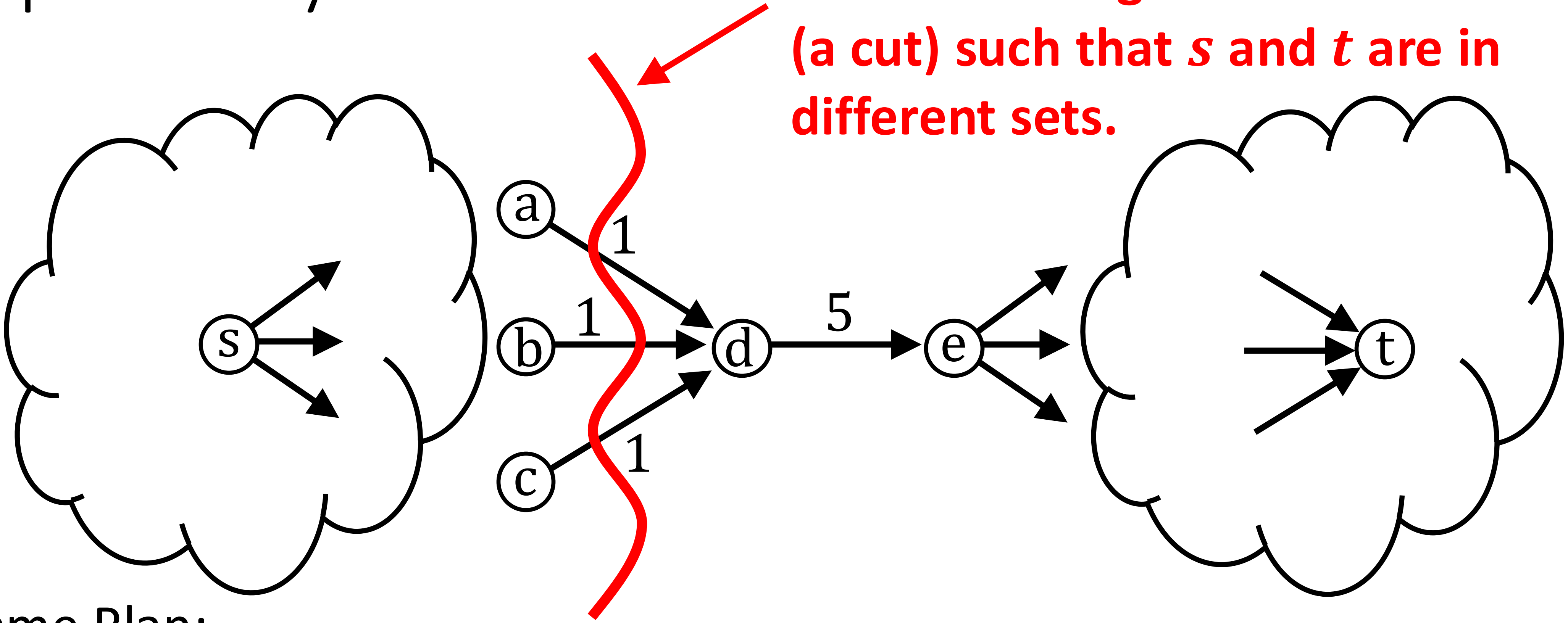
1. Show that value of every flow is $\leq$ capacity of every cut.



If we find some flow whose value equals the capacity of some cut, it must be the optimal flow.

Optimality

$s - t$ cut: Segmentation of vertices (a cut) such that s and t are in different sets.

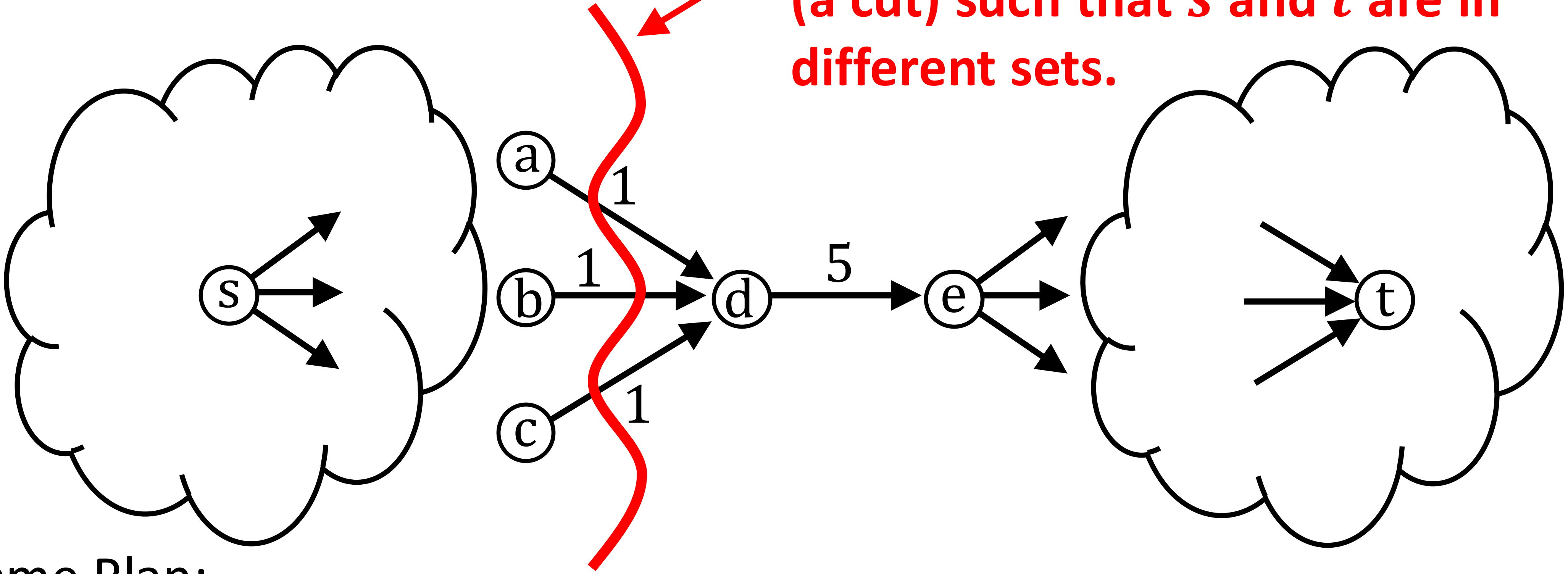


Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.
2. Given a flow where there are no $s - t$ paths left in the residual graph, there is a specific cut whose capacity = flow value.

Optimality

$s - t$ cut: Segmentation of vertices (a cut) such that s and t are in different sets.



Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.
2. Given a flow where there are no $s - t$ paths left in the residual graph, there is a specific cut whose capacity = flow value.

$\Rightarrow$ **The algorithm is optimal**

Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof:

Edges that enter the set A

Edges that leave the set A

Optimality

This relates arbitrary $s-t$ flows to arbitrary $s-t$ cuts

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \underline{\text{out}(A)}} f(e) - \sum_{e \in \underline{\text{in}(A)}} f(e)$.

<u>Proof:</u>

Edges that enter the set A

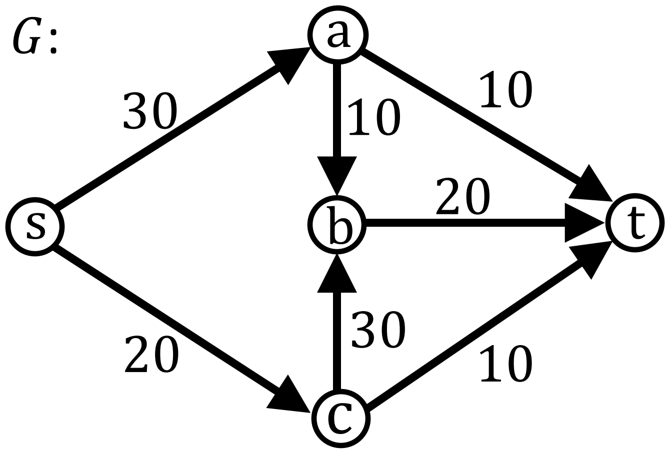
Edges that leave the set A

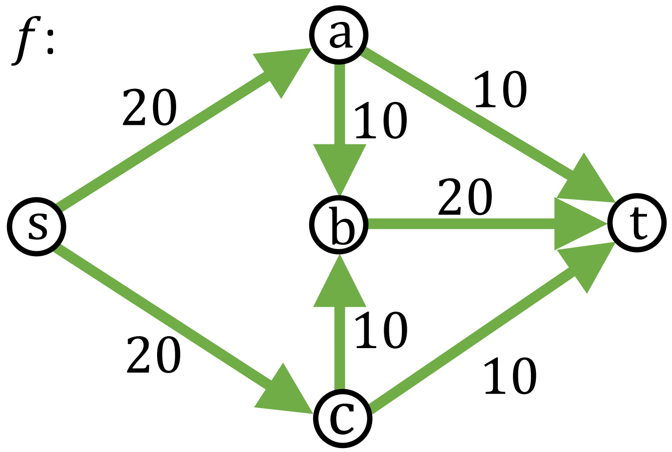
Optimality

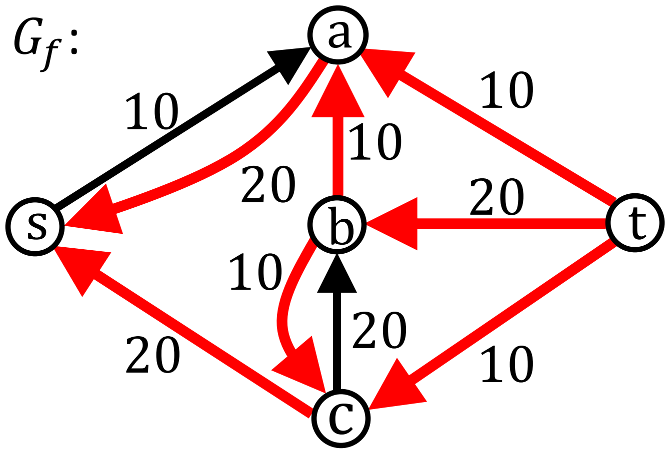
<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u>

$|f| = 40$





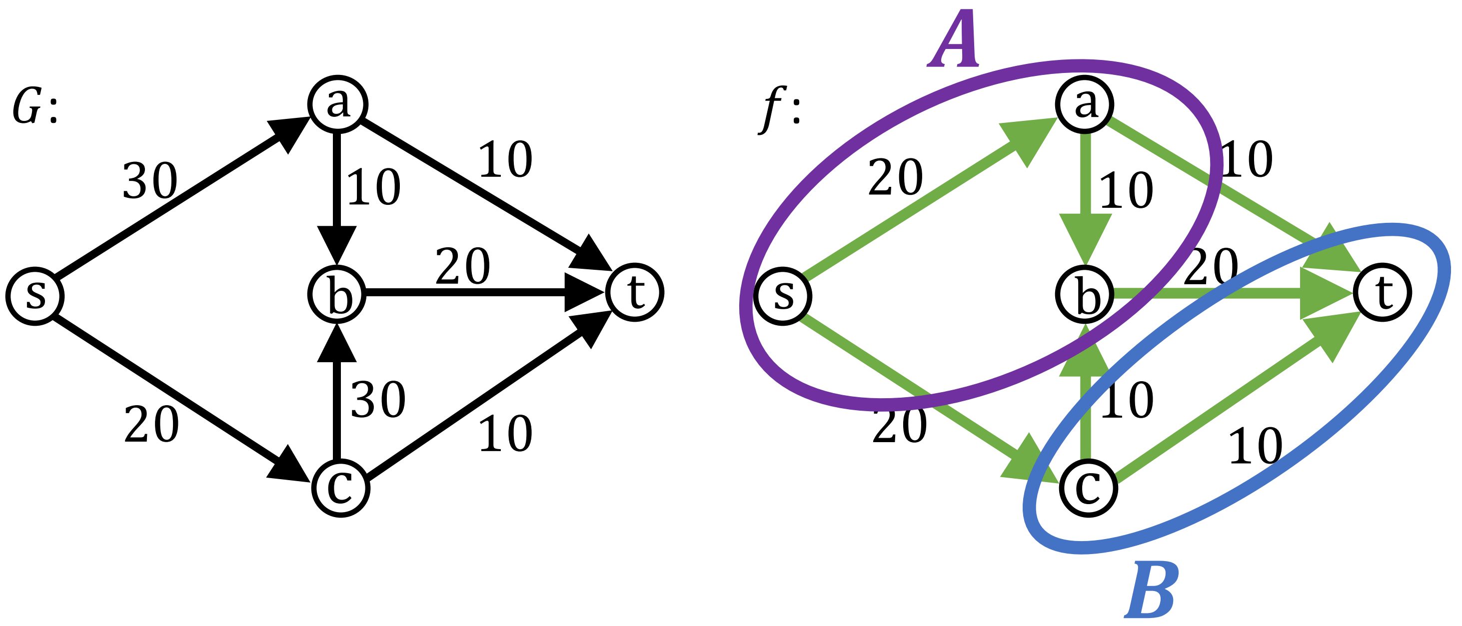


Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof:

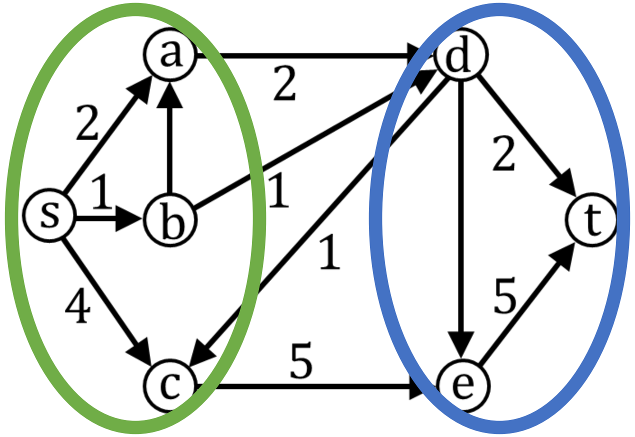
$|f| = 40$
$\sum_{e \in \text{out}(A)} f(e) = 50$
$\sum_{e \in \text{in}(A)} f(e) = 10$



Optimality

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

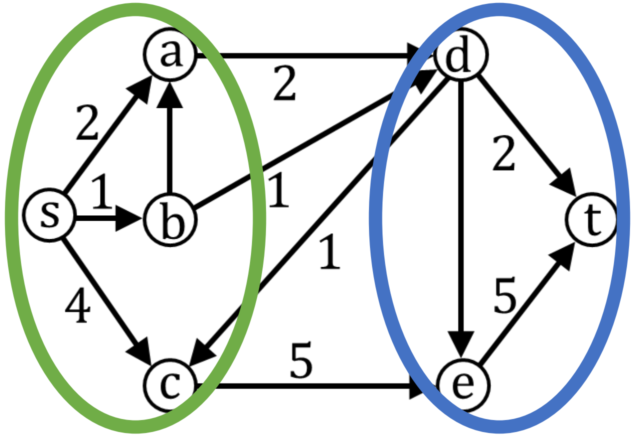


Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

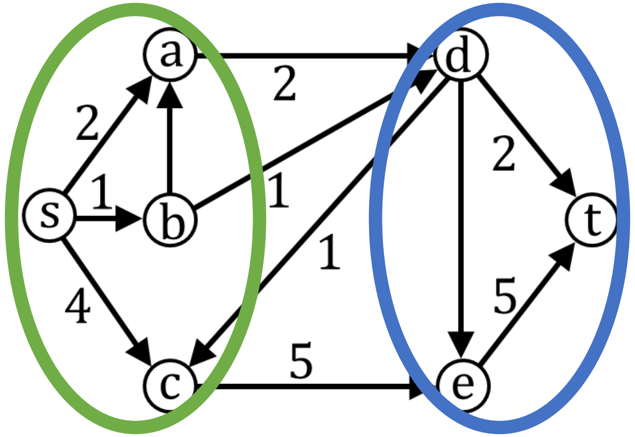
Then, $\forall v \in A, v \neq s, f^{out}(v) - f^{in}(v) = ?$



Optimality

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.
Then, $\forall v \in A, v \neq s, f^{out}(v) - f^{in}(v) = 0$ (by conservation of flow).



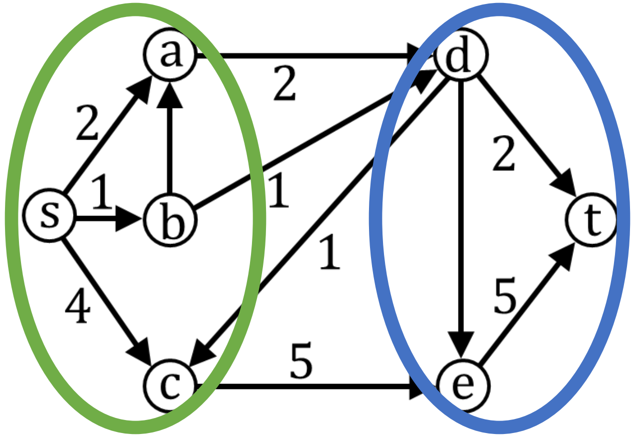
Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $\forall v \in A, v \neq s, f^{out}(v) - f^{in}(v) = 0$ (by conservation of flow).

By definition, $|f| = f^{out}(s)$.



Optimality

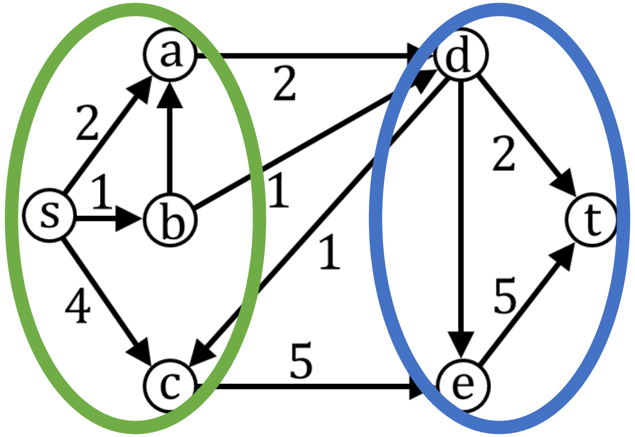
Theorem 1: Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $\forall v \in A, v \neq s, f^{out}(v) - f^{in}(v) = 0$ (by conservation of flow).

By definition, $|f| = f^{out}(s)$.

$\implies |f| = f^{out}(s) - f^{in}(s)$ (since $f^{in}(s) = 0$)



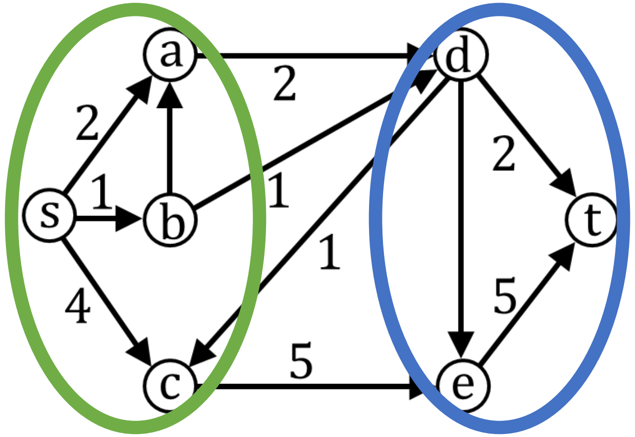
Optimality

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.
Then, $\forall v \in A, v \neq s, f^{out}(v) - f^{in}(v) = 0$ (by conservation of flow).
By definition, $|f| = f^{out}(s)$.

$\Longrightarrow |f| = f^{out}(s) - f^{in}(s)$ (since $f^{in}(s) = 0$)

$\Longrightarrow |f| = \sum_{v \in A}(f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

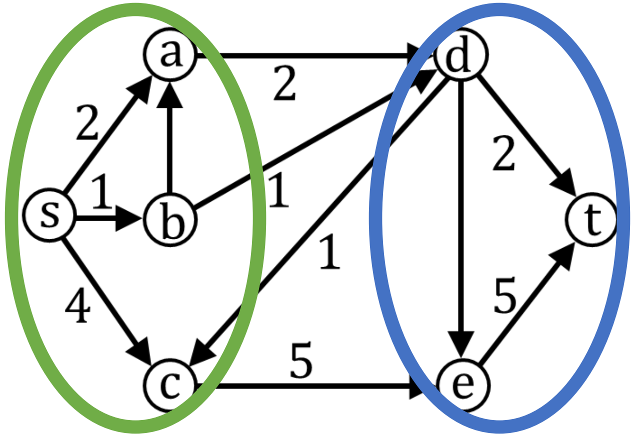


Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A}(f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).



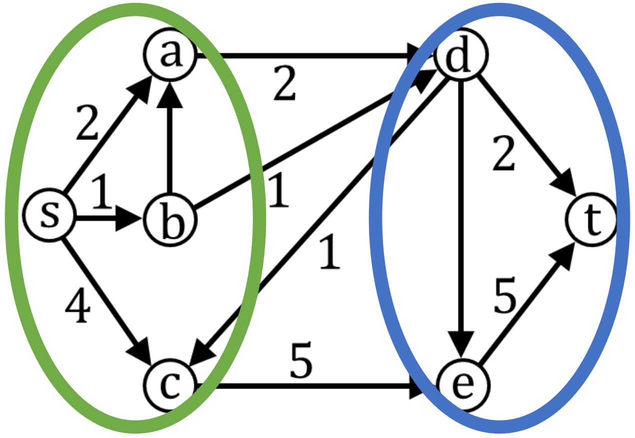
Optimality

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A}(f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

Need to translate vertices in A into edges leaving A.



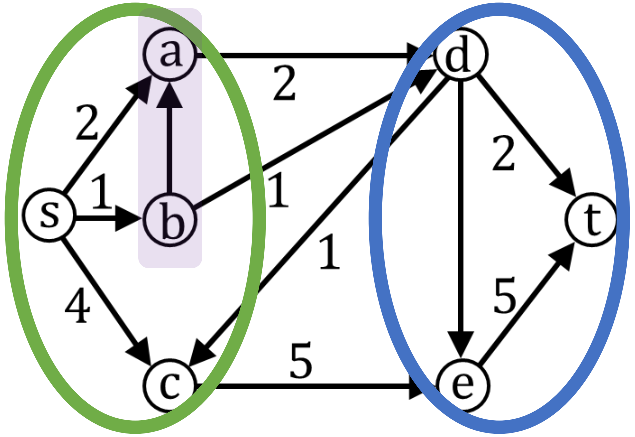
Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.
Then, $|f| = \sum_{v \in A} (f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. $u \in A, v \in A$ (edge is inside A)
2. $u \notin A, v \notin A$ (edge is outside A)
3. $u \in A, v \notin A$ (edge leaves A)
4. $u \notin A, v \in A$ (edge enters A)



Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

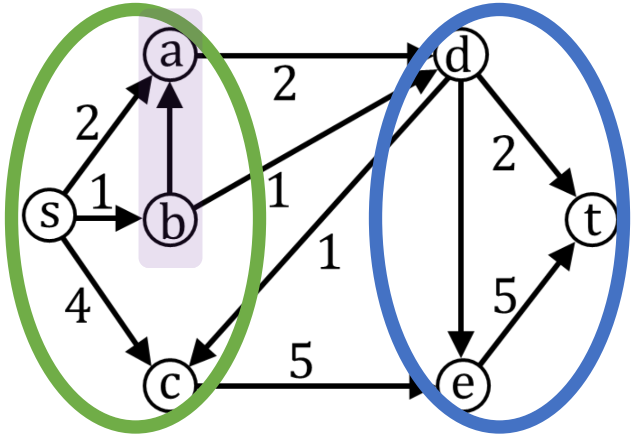
Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A} (f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. $u \in A, v \in A \rightarrow f(e)$ cancels out.
2. $u \notin A, v \notin A$ (edge is outside A)
3. $u \in A, v \notin A$ (edge leaves A)
4. $u \notin A, v \in A$ (edge enters A)

(In some $v \in A$ and out another)



Optimality

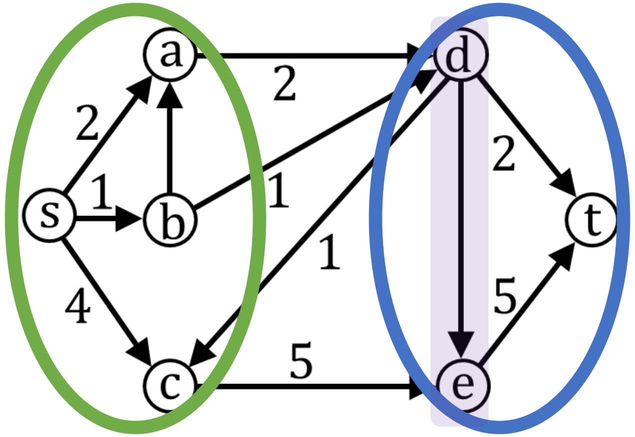
<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A} (f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. ~~$u \in A, v \in A \rightarrow f(e)$ cancels out.~~ (In some $v \in A$ and out another)
2. $u \notin A, v \notin A$ **(edge is outside A)**
3. $u \in A, v \notin A$ **(edge leaves A)**
4. $u \notin A, v \in A$ **(edge enters A)**



Optimality

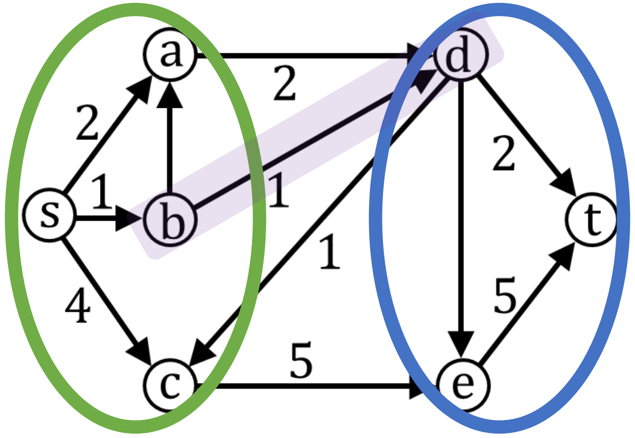
<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A}(f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. ~~$u \in A, v \in A \to f(e)$ cancels out.~~
2. ~~$u \notin A, v \notin A \to f(e)$ does not appear at all.~~
3. $u \in A, v \notin A$ **(edge leaves A)**
4. $u \notin A, v \in A$ **(edge enters A)**



Optimality

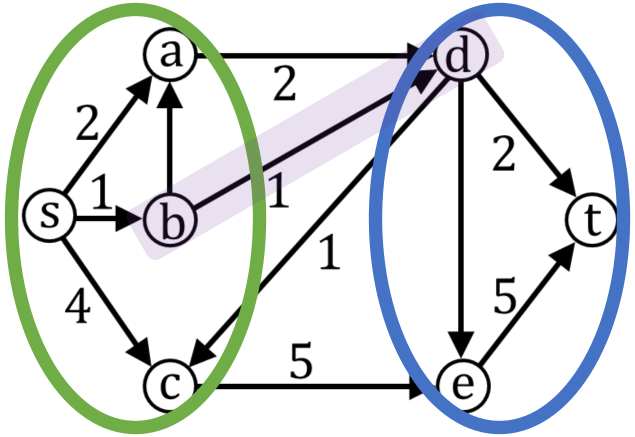
Theorem 1: Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A}(f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. ~~$u \in A, v \in A \rightarrow f(e)$ cancels out.~~
2. ~~$u \notin A, v \notin A \rightarrow f(e)$ does not appear at all.~~
3. $u \in A, v \notin A \rightarrow f(e)$ adds to the sum.
4. $u \notin A, v \in A$ **(edge enters A)**



Optimality

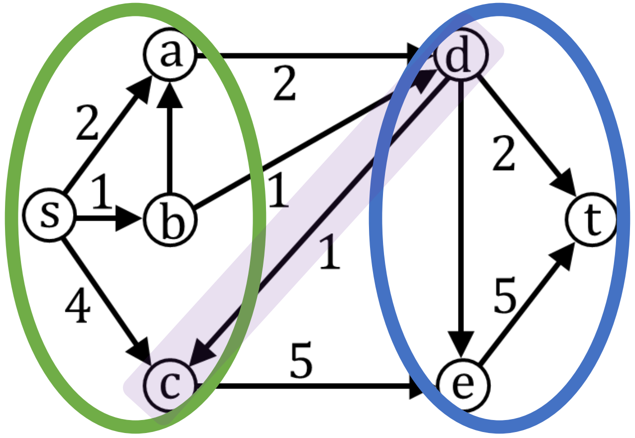
Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Proof: Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A} (f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. ~~$u \in A, v \in A \rightarrow f(e)$ cancels out.~~
2. ~~$u \notin A, v \notin A \rightarrow f(e)$ does not appear at all.~~
3. $u \in A, v \notin A \rightarrow f(e)$ adds to the sum.
4. $u \notin A, v \in A$ **(edge enters A)**



Optimality

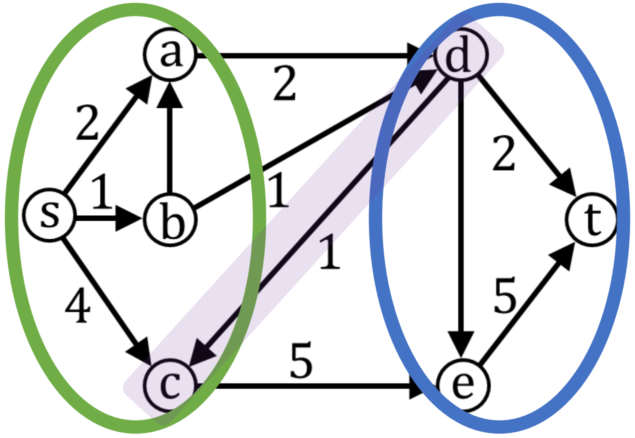
<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A} (f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

~~1. $u \in A, v \in A \rightarrow f(e)$ cancels out.~~
~~2. $u \notin A, v \notin A \rightarrow f(e)$ does not appear at all.~~
3. $u \in A, v \notin A \rightarrow f(e)$ adds to the sum.
4. $u \notin A, v \in A \rightarrow f(e)$ subtracts from the sum.



Optimality

__Theorem 1:__ Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

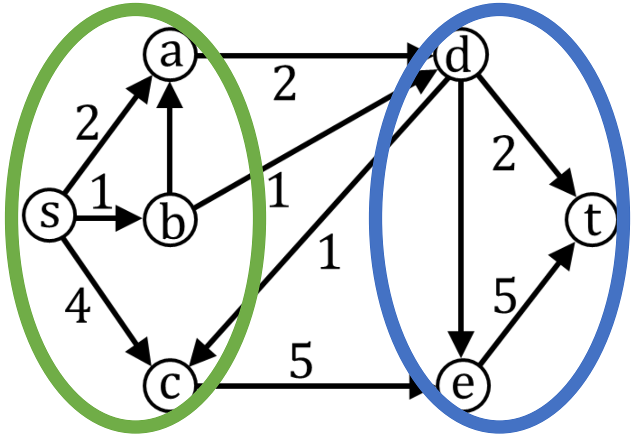
__Proof:__ Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A}(f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. ~~$u \in A, v \in A \rightarrow f(e)$ cancels out.~~
2. ~~$u \notin A, v \notin A \rightarrow f(e)$ does not appear at all.~~
3. $u \in A, v \notin A \rightarrow f(e)$ adds to the sum.
4. $u \notin A, v \in A \rightarrow f(e)$ subtracts from the sum.

$$\Longrightarrow \sum_{v \in A}(f^{out}(v) - f^{in}(v)) = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$$



Optimality

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow.
Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

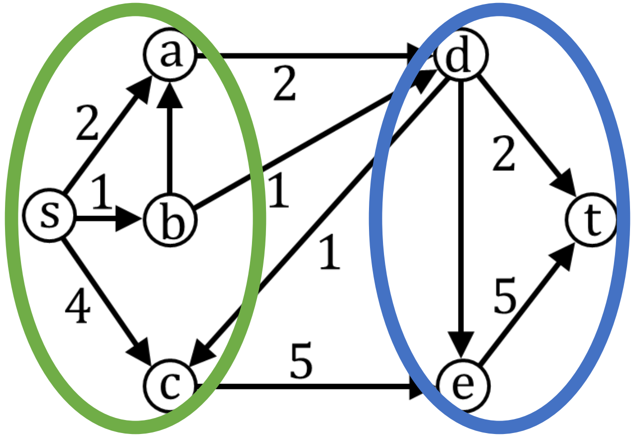
<u>Proof:</u> Let $f^{out}(v) = \sum_{e \in \text{out}(v)} f(e)$ and $f^{in}(v) = \sum_{e \in \text{in}(v)} f(e)$.

Then, $|f| = \sum_{v \in A} (f^{out}(v) - f^{in}(v))$ (Only $\neq 0$ when $v = s$).

$\forall e = (u, v) \in E$:

1. ~~$u \in A, v \in A \rightarrow f(e)$ cancels out.~~
2. ~~$u \notin A, v \notin A \rightarrow f(e)$ does not appear at all.~~
3. $u \in A, v \notin A \rightarrow f(e)$ adds to the sum.
4. $u \notin A, v \in A \rightarrow f(e)$ subtracts from the sum.

$$\Longrightarrow \sum_{v \in A} (f^{out}(v) - f^{in}(v))$$
$$= \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$$
$$\Longrightarrow |f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$$



Optimality

This relates arbitrary $s-t$ flows to arbitrary $s-t$ cuts

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Optimality

This relates arbitrary $s - t$ flows to arbitrary $s - t$ cuts

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Corollary: Suppose G is a flow network, f is an $s - t$ flow on G, and (A, B) is an $s - t$ cut. Then, $|f| \leq c(A, B)$. (i.e. *every* flow is bounded by *any* $s - t$ cut)

Proof:

?

Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Corollary: Suppose G is a flow network, f is an $s - t$ flow on G, and (A, B) is an $s - t$ cut. Then, $|f| \leq c(A, B)$. (i.e. *every* flow is bounded by *any* $s - t$ cut)

Proof:

$$|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$$

Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

Corollary: Suppose G is a flow network, f is an $s - t$ flow on G, and (A, B) is an $s - t$ cut. Then, $|f| \leq c(A, B)$. (i.e. *every* flow is bounded by *any* $s - t$ cut)

Proof:

$$
\begin{aligned}
|f| &= \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e) \\
&\leq \sum_{e \in \text{out}(A)} f(e)
\end{aligned}
$$

Optimality

<u>Theorem 1:</u> Let G be a flow network, (A, B) be an $s-t$ cut, and f be an $s-t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

<u>Corollary:</u> Suppose G is a flow network, f is an $s-t$ flow on G, and (A, B) is an $s-t$ cut. Then, $|f| \leq c(A, B)$. (i.e. *every* flow is bounded by *any* $s-t$ cut)

<u>Proof:</u>

$$\begin{aligned} |f| &= \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e) \\ &\leq \sum_{e \in \text{out}(A)} f(e) \\ &\leq \sum_{e \in \text{out}(A)} c_e = c(A, B) \end{aligned}$$

Optimality

Theorem 1: Let G be a flow network, (A, B) be an $s - t$ cut, and f be an $s - t$ flow. Then, $|f| = \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e)$.

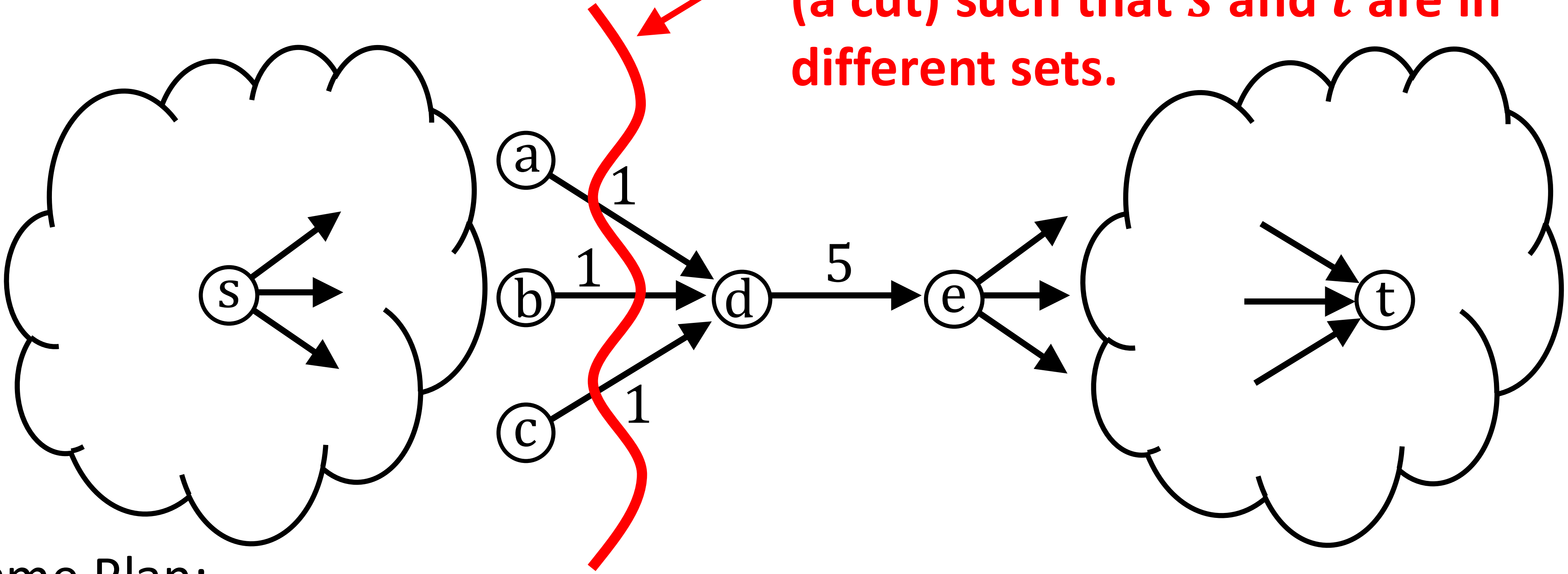
Corollary: Suppose G is a flow network, f is an $s - t$ flow on G, and (A, B) is an $s - t$ cut. Then, $|f| \leq c(A, B)$. (i.e. *every* flow is bounded by *any* $s - t$ cut)

Proof:

$$
\begin{aligned}
|f| &= \sum_{e \in \text{out}(A)} f(e) - \sum_{e \in \text{in}(A)} f(e) \\
&\leq \sum_{e \in \text{out}(A)} f(e) \\
&\leq \sum_{e \in \text{out}(A)} c_e = c(A, B)
\end{aligned}
$$

If we find some flow f and some cut (A, B) such that $|f| = c(A, B)$, then f is a maximum flow.

Optimality



Game Plan:

1. Show that value of every flow is $\leq$ capacity of every cut.
2. Given a flow where there are no $s - t$ paths left in the residual graph, there is a specific cut whose capacity = flow value.