Augmenting Paths

Math 482, Lecture 25

Misha Lavrov

April 3, 2020

Lecture plan

We know how to find a max flow using an LP. But this is inefficient; there are many many algorithms that are faster.

Lecture plan

We know how to find a max flow using an LP. But this is inefficient; there are many many algorithms that are faster.

Plan for today:

1. Describe a simple greedy algorithm that tries to find a max flow.

Lecture plan

We know how to find a max flow using an LP. But this is inefficient; there are many many algorithms that are faster.

Plan for today:

1. Describe a simple greedy algorithm that tries to find a max flow.
2. See it get stuck.

Lecture plan

We know how to find a max flow using an LP. But this is inefficient; there are many many algorithms that are faster.

Plan for today:

1. Describe a simple greedy algorithm that tries to find a max flow.
2. See it get stuck.
3. Make the algorithm more powerful.

Directed s, t-paths

Definition

In a network, a *directed path* from s to t is a sequence

$$s, v_1, v_2, \ldots, v_k, t$$

where $v_1, v_2, \ldots, v_k \in N$ and $(s, v_1), (v_1, v_2), \ldots, (v_k, t) \in A$.

Directed s, t-paths

Definition

In a network, a *directed path* from s to t is a sequence

$$s \to v_1 \to v_2 \cdots \to v_k \to t$$

where $v_1, v_2, \ldots, v_k \in N$ and $(s, v_1), (v_1, v_2), \ldots, (v_k, t) \in A$.

Directed s, t-paths

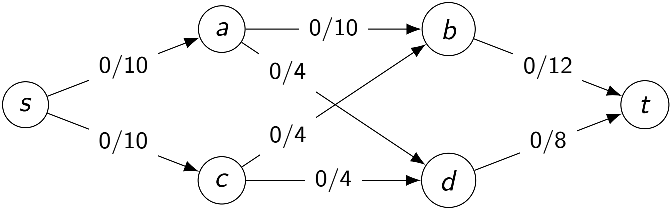
Definition

In a network, a *directed path* from s to t is a sequence

$$s \to v_1 \to v_2 \cdots \to v_k \to t$$

where $v_1, v_2, \ldots, v_k \in N$ and $(s, v_1), (v_1, v_2), \ldots, (v_k, t) \in A$.

Example:



Directed s, t-paths

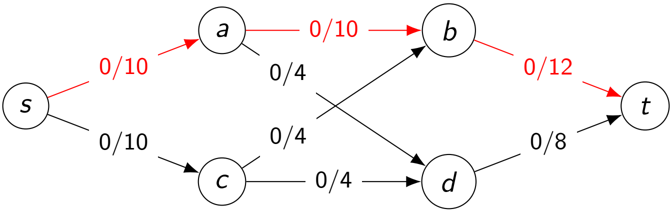
Definition

In a network, a *directed path* from s to t is a sequence

$$s \to v_1 \to v_2 \cdots \to v_k \to t$$

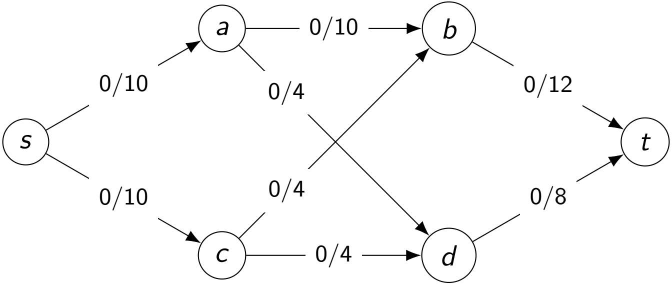
where $v_1, v_2, \ldots, v_k \in N$ and $(s, v_1), (v_1, v_2), \ldots, (v_k, t) \in A$.

Example: directed path $s \to a \to b \to t$



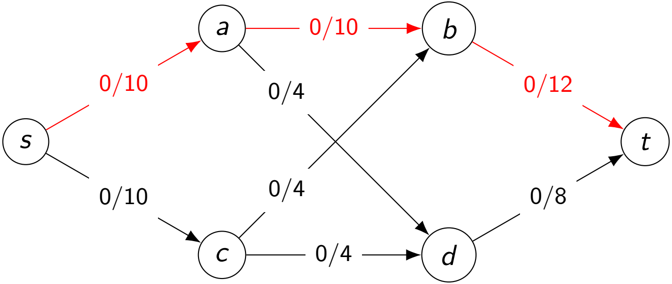
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



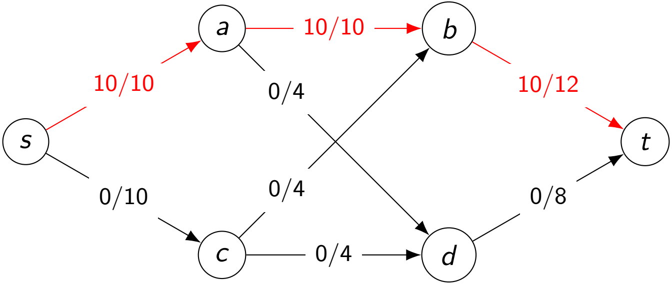
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



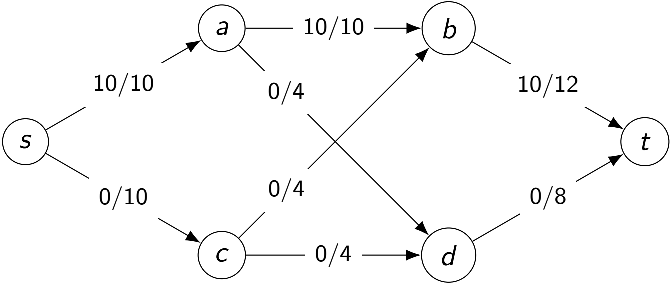
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



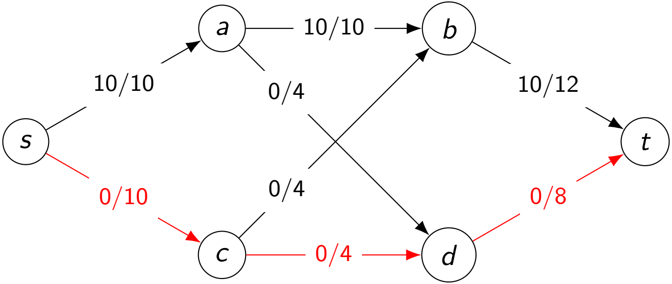
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



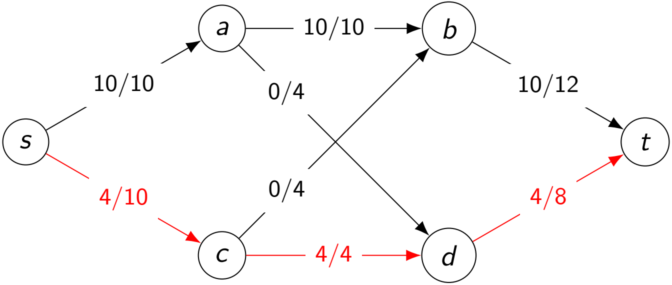
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



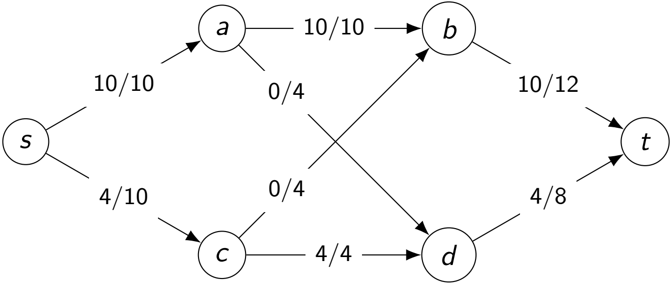
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



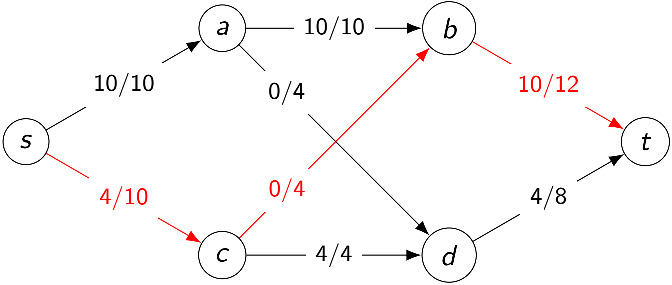
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



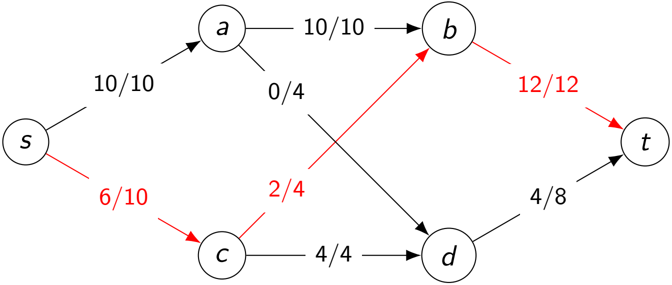
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



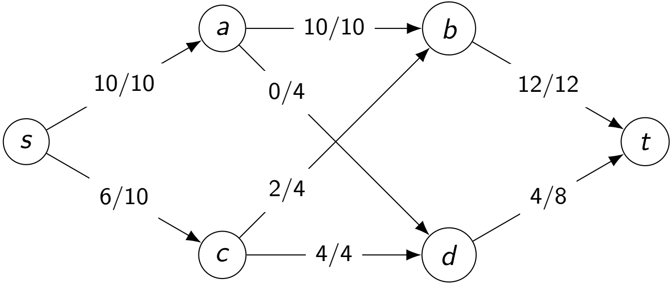
Using a directed path

Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



Using a directed path

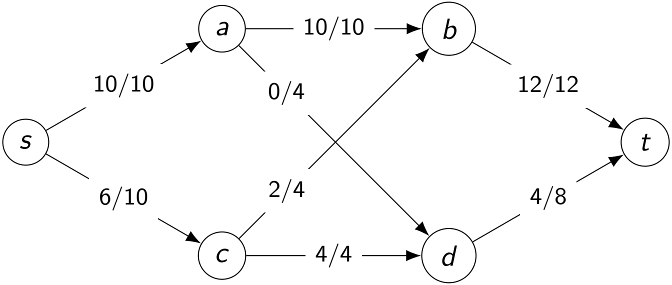
Whenever we have a directed path from s to t and all arcs along the path are below capacity, we can use it to increase the flow.



At this point, there are no more directed paths where all arcs are below capacity. But is this the maximum flow?

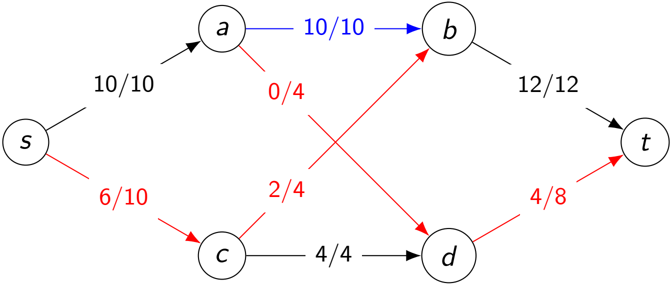
A further improvement

If we redirect some $a \to b \to t$ flow to go $a \to d \to t$, we can send more flow along the path $s \to c \to d \to t$...



A further improvement

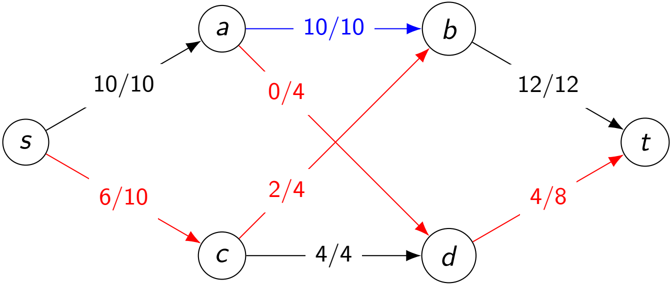
If we redirect some $a \to b \to t$ flow to go $a \to d \to t$, we can send more flow along the path $s \to c \to d \to t$...



Increase flow along red edges, decrease flow along blue edge.

A further improvement

If we redirect some $a \to b \to t$ flow to go $a \to d \to t$, we can send more flow along the path $s \to c \to d \to t$...



Increase flow along red edges, decrease flow along blue edge.

Note: $s \to c \to b \leftarrow a \to d \to t$ is *almost* a directed path.

Augmenting paths

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, an augmenting path for $\mathbf{x}$ is a sequence of nodes

$$s = v_0, v_1, v_2, \ldots, v_k, v_{k+1} = t$$

such that for each pair v_i, v_{i+1}:

- either $e = (v_i, v_{i+1})$ is an arc below capacity ($e \in A$ and $x_e < c_e$)

Augmenting paths

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, an augmenting path for $\mathbf{x}$ is a sequence of nodes

$$s = v_0, v_1, v_2, \ldots, v_k, v_{k+1} = t$$

such that for each pair v_i, v_{i+1}:

- either $e = (v_i, v_{i+1})$ is an arc below capacity ($e \in A$ and $x_e < c_e$)
- or $e = (v_{i+1}, v_i)$ is an arc with positive flow ($e \in A$ and $x_e > 0$).

Augmenting paths

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, an augmenting path for $\mathbf{x}$ is a sequence of nodes

$$s = v_0, v_1, v_2, \ldots, v_k, v_{k+1} = t$$

such that for each pair v_i, v_{i+1}:

- either $e = (v_i, v_{i+1})$ is an arc below capacity ($e \in A$ and $x_e < c_e$)
- or $e = (v_{i+1}, v_i)$ is an arc with positive flow ($e \in A$ and $x_e > 0$).

The sequence

$$s \to c \to b \leftarrow a \to d \to t$$

we found on the previous slide was an augmenting path.

Using an augmenting path to improve **x**

To augment a feasible flow **x** along an augmenting path:

1. Let $\delta > 0$ be the largest value such that
   - $x_e \leq c_e - \delta$ for all forward arcs on the path;
   - $x_e \geq \delta$ for all backward arcs on the path.

Using an augmenting path to improve **x**

To augment a feasible flow **x** along an augmenting path:

1. Let $\delta > 0$ be the largest value such that
   - $x_e \leq c_e - \delta$ for all forward arcs on the path;
   - $x_e \geq \delta$ for all backward arcs on the path.
2. For each forward arc e, increase x_e by δ.

Using an augmenting path to improve **x**

To augment a feasible flow **x** along an augmenting path:

1. Let $\delta > 0$ be the largest value such that
   - $x_e \leq c_e - \delta$ for all forward arcs on the path;
   - $x_e \geq \delta$ for all backward arcs on the path.
2. For each forward arc e, increase x_e by δ.
3. For each backward arc e, decrease x_e by δ.

Using an augmenting path to improve x

To augment a feasible flow **x** along an augmenting path:

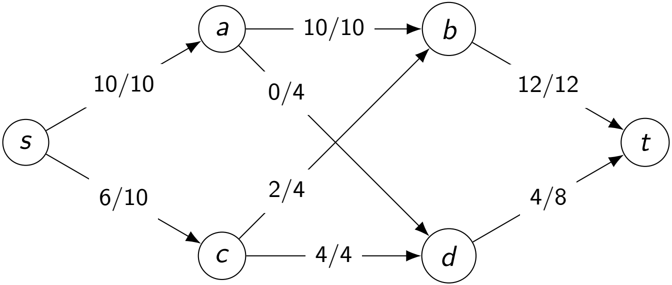
1. Let $\delta > 0$ be the largest value such that
   - $x_e \leq c_e - \delta$ for all forward arcs on the path;
   - $x_e \geq \delta$ for all backward arcs on the path.
2. For each forward arc e, increase x_e by δ.
3. For each backward arc e, decrease x_e by δ.

When we do this, flow is still conserved at internal nodes of the augmenting path. There are four possible cases:

$$\cdots \xrightarrow{+\delta} p \xrightarrow{+\delta} \cdots \qquad \cdots \xrightarrow{+\delta} q \xleftarrow{-\delta} \cdots$$

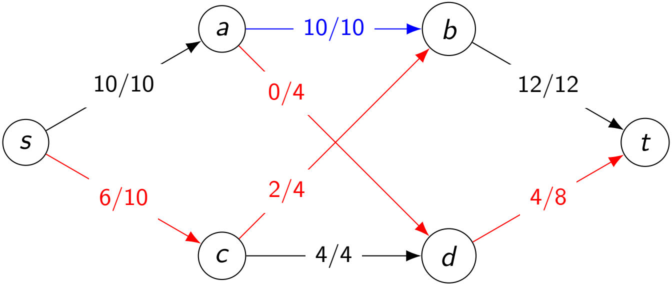
$$\cdots \xleftarrow{-\delta} r \xrightarrow{+\delta} \cdots \qquad \cdots \xleftarrow{-\delta} s \xleftarrow{-\delta} \cdots$$

Example of augmenting



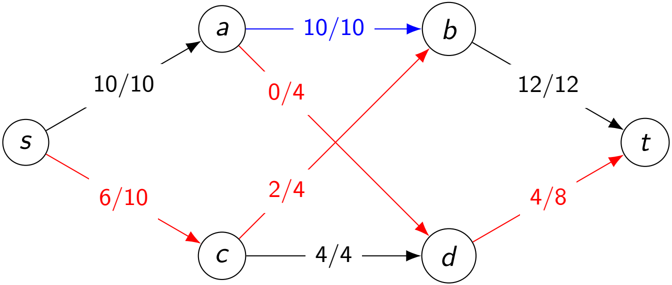
1. Find the augmenting path.

Example of augmenting



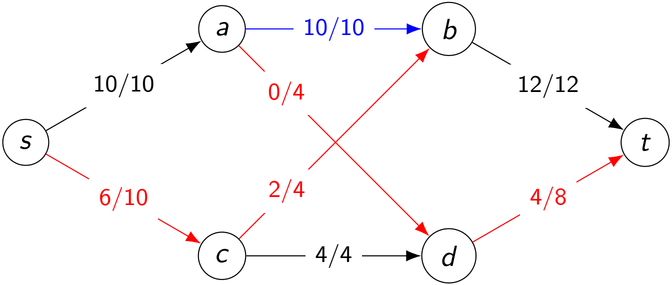
1. Find the augmenting path $s \to c \to b \leftarrow a \to d \to t$.

Example of augmenting



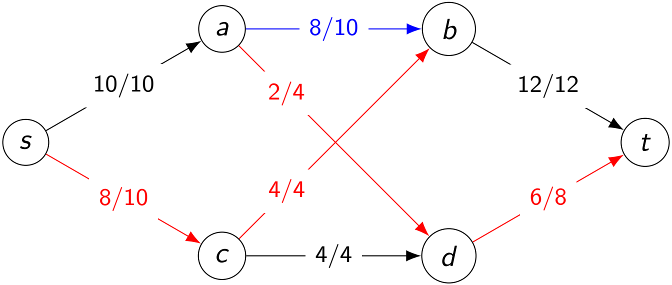
1. Find the augmenting path $s \to c \to b \leftarrow a \to d \to t$.
2. Find the value δ we can augment by.

Example of augmenting



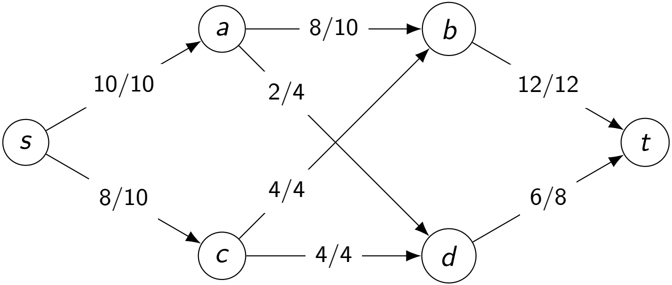
1. Find the augmenting path $s \to c \to b \leftarrow a \to d \to t$.
2. Find the value δ we can augment by: here, $\delta = 2$.

Example of augmenting



1. Find the augmenting path $s \to c \to b \leftarrow a \to d \to t$.
2. Find the value δ we can augment by: here, $\delta = 2$.
3. Increase or decrease the flow along each edge by δ.

Example of augmenting



1. Find the augmenting path $s \to c \to b \leftarrow a \to d \to t$.
2. Find the value δ we can augment by: here, $\delta = 2$.
3. Increase or decrease the flow along each edge by δ.

The residual graph

We define a *residual graph* to help us find augmenting paths.

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, the residual graph is a network with:

The residual graph

We define a *residual graph* to help us find augmenting paths.

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, the residual graph is a network with:

- The same set of nodes N as the original network.

The residual graph

We define a *residual graph* to help us find augmenting paths.

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, the residual graph is a network with:

- The same set of nodes N as the original network.
- For each arc $(i, j) \in A$ with $x_{ij} < c_{ij}$, a "forward" arc (i, j) with *residual capacity* $c_{ij} - x_{ij}$.

The residual graph

We define a *residual graph* to help us find augmenting paths.

Definition

Given a network (N, A) and a feasible flow $\mathbf{x}$, the residual graph is a network with:

- The same set of nodes N as the original network.
- For each arc $(i, j) \in A$ with $x_{ij} < c_{ij}$, a "forward" arc (i, j) with *residual capacity* $c_{ij} - x_{ij}$.
- For each arc $(i, j) \in A$ with $x_{ij} > 0$, a "backward" arc (j, i) with residual capacity x_{ij}.

The residual graph

We define a *residual graph* to help us find augmenting paths.

Definition

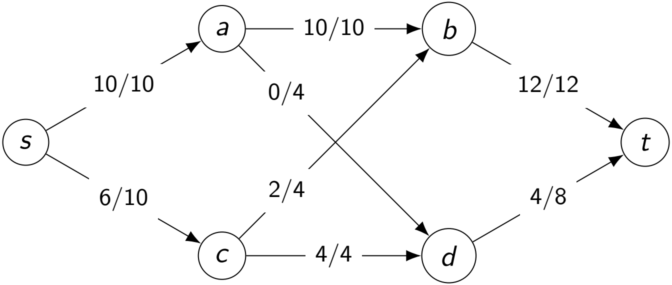
Given a network (N, A) and a feasible flow $\mathbf{x}$, the residual graph is a network with:

- The same set of nodes N as the original network.
- For each arc $(i, j) \in A$ with $x_{ij} < c_{ij}$, a "forward" arc (i, j) with *residual capacity* $c_{ij} - x_{ij}$.
- For each arc $(i, j) \in A$ with $x_{ij} > 0$, a "backward" arc (j, i) with residual capacity x_{ij}.

Idea: augmenting paths for $\mathbf{x}$ are directed paths in the residual graph.

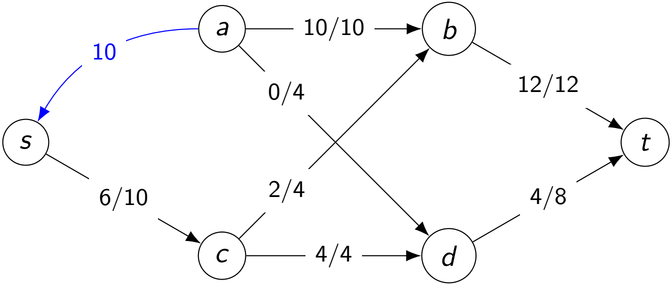
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



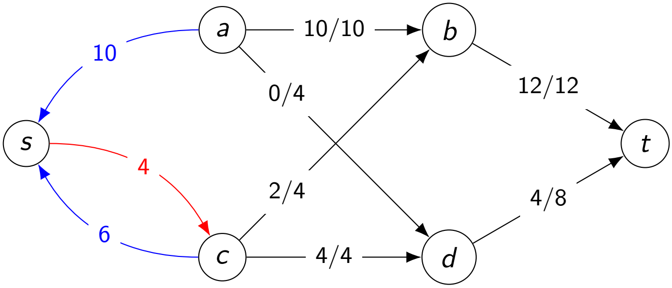
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



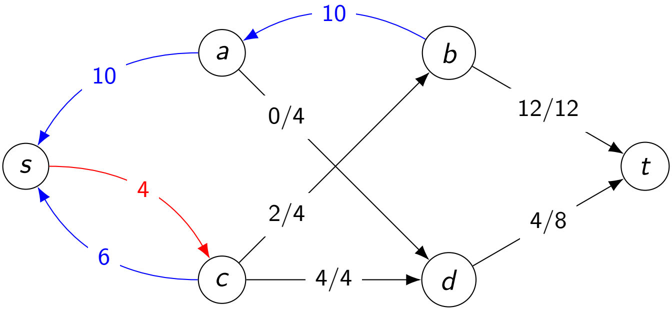
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



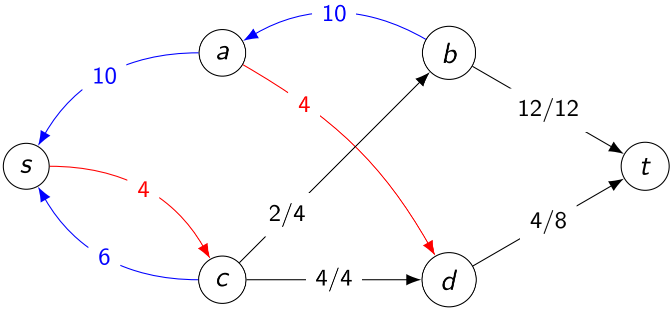
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



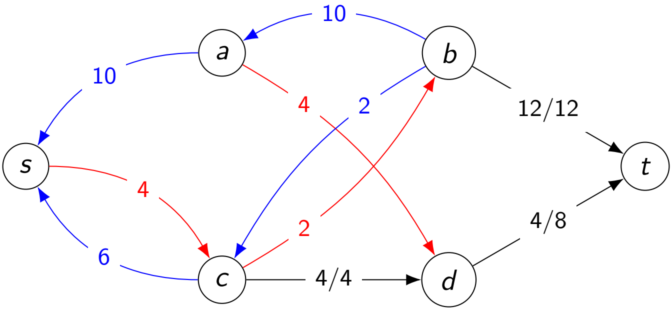
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



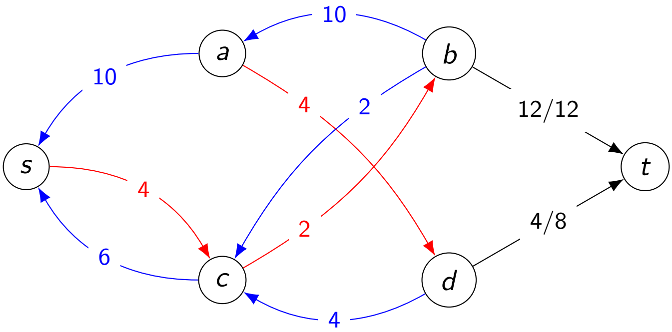
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



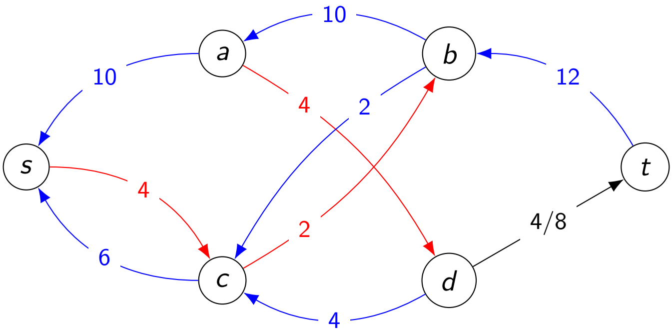
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



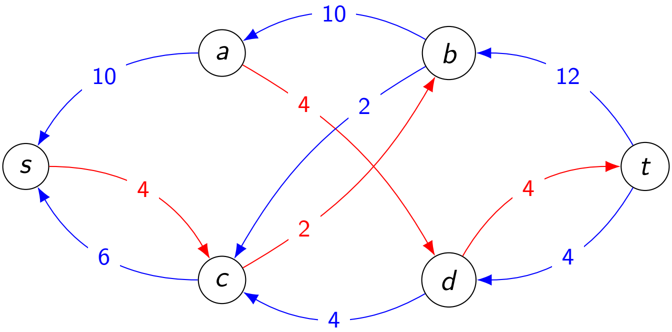
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



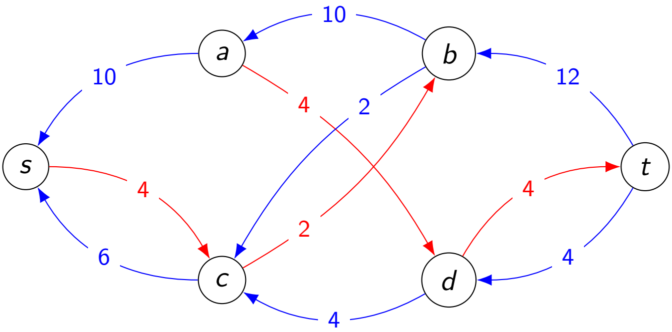
Residual graph example

We construct the residual graph for our next-to-last feasible flow:



Residual graph example

We construct the residual graph for our next-to-last feasible flow:



We can find the augmenting path $s \to b \leftarrow a \to d \to t$ by following only *forward* arrows (of either color) in the residual graph; $\delta = 2$ is the smallest number along those arrows.