Flow-Augmenting Paths and Max-Flow Min-Cut Theorem

Algorithm Design & Analysis

Spring 2018
Outline
Our goal for today

**Theorem**

Let $f$ be an $(s, t)$-flow on a network $G = (V, E, c)$ and let $(A, B)$ be an $(s, t)$-cut of $G$. Then $\nu(f) = \text{cap}(A, B)$ if and only if $f$ is a flow of maximum value and $(A, B)$ is a cut of minimum value.
Residual Graph of a Flow on $G$

Build associated graph from $G$ that includes \textit{backward} edges.
Residual Graph of a Flow on $G$

Build associated graph from $G$ that includes \textit{backward} edges.

- $G_f = (V_f, E_f)$, the \textit{residual flow graph} of $G = (V, E)$ and $f$, has $V_f = V$ and $E_f$ defined as follows:

  - For each $e \in E$ with $f(e) < c(e)$, $e$ is in $E_f$, and $e$ has capacity $c_f(e) = c(e) - f(e)$. $e$ is a \textit{forward} edge of $G_f$.

  - For each $e = (u, v) \in E$ with $f(e) > 0$, $e_R = (v, u)$ is in $E_f$ and $e_R$ has capacity $c_f(e) = f(e)$. $e$ is a \textit{backward} edge of $G_f$.

- $G_f$ is a flow network built from $G$ and an $(s, t)$-flow on $G$.

- $G_f$ can be used to identify improvements or augmentations to $f$; that is, changes that will increase $v(f)$. Let’s do an example!
Residual Graph of a Flow on $G$

Build associated graph from $G$ that includes \textit{backward} edges.

- $G_f = (V_f, E_f)$, the \textit{residual flow graph} of $G = (V, E)$ and $f$, has $V_f = V$ and $E_f$ defined as follows:
  - For each $e \in E$ with $f(e) < c(e)$, $e$ is in $E_f$, and $e$ has capacity $c_f(e) = c(e) - f(e)$. $e$ is a \textit{forward} edge of $G_f$.
Residual Graph of a Flow on $G$

Build associated graph from $G$ that includes *backward* edges.

- $G_f = (V_f, E_f)$, the *residual flow graph* of $G = (V, E)$ and $f$, has $V_f = V$ and $E_f$ defined as follows
  - For each $e \in E$ with $f(e) < c(e)$, $e$ is in $E_f$, and $e$ has capacity $c_f(e) = c(e) - f(e)$. $e$ is a *forward* edge of $G_f$.
  - For each $e = (u, v) \in E$ with $f(e) > 0$, $e^R = (v, u)$ is in $E_f$ and $e^R$ has capacity $c_f(e^R) = f(e)$. $e$ is a *backward* edge of $G_f$. 

Let's do an example!
Residual Graph of a Flow on $G$

Build associated graph from $G$ that includes backward edges.

- $G_f = (V_f, E_f)$, the residual flow graph of $G = (V, E)$ and $f$, has $V_f = V$ and $E_f$ defined as follows
  - For each $e \in E$ with $f(e) < c(e)$, $e$ is in $E_f$, and $e$ has capacity $c_f(e) = c(e) - f(e)$. $e$ is a forward edge of $G_f$.
  - For each $e = (u, v) \in E$ with $f(e) > 0$, $e^R = (v, u)$ is in $E_f$ and $e^R$ has capacity $c_f(e^R) = f(e)$. $e$ is a backward edge of $G_f$.
- $G_f$ is a flow network built from $G$ and an $(s, t)$-flow on $G$. 

Let's do an example!
Build associated graph from $G$ that includes *backward* edges.

- $G_f = (V_f, E_f)$, the *residual flow graph* of $G = (V, E)$ and $f$, has $V_f = V$ and $E_f$ defined as follows:
  - For each $e \in E$ with $f(e) < c(e)$, $e$ is in $E_f$, and $e$ has capacity $c_f(e) = c(e) - f(e)$. $e$ is a *forward* edge of $G_f$.
  - For each $e = (u, v) \in E$ with $f(e) > 0$, $e^R = (v, u)$ is in $E_f$ and $e^R$ has capacity $c_f(e^R) = f(e)$. $e$ is a *backward* edge of $G_f$.
- $G_f$ is a flow network built from $G$ and an $(s, t)$-flow on $G$.
- $G_f$ can be used to identify improvements or *augmentations* to $f$; that is, changes that will increase $\nu(f)$.
Build associated graph from $G$ that includes \textit{backward} edges.

- $G_f = (V_f, E_f)$, the \textit{residual flow graph} of $G = (V, E)$ and $f$, has $V_f = V$ and $E_f$ defined as follows
  - For each $e \in E$ with $f(e) < c(e)$, $e$ is in $E_f$, and $e$ has capacity $c_f(e) = c(e) - f(e)$. $e$ is a \textit{forward} edge of $G_f$.
  - For each $e = (u, v) \in E$ with $f(e) > 0$, $e^R = (v, u)$ is in $E_f$ and $e^R$ has capacity $c_f(e^R) = f(e)$. $e$ is a \textit{backward} edge of $G_f$.

- $G_f$ is a flow network built from $G$ and an $(s, t)$-flow on $G$.
- $G_f$ can be used to identify improvements or \textit{augmentations} to $f$; that is, changes that will increase $\nu(f)$.

Let's do an example!
Take-aways from example

• The "middle edge" becomes two edges.
• The ad-hoc pushing back of flow in $G$ preserves flow conservation at all nodes and corresponds to a directed $(s, t)$-path of positive flow in $G_f$.
• Such a path can be found by BFS from $s$ in $G_f$.
• The smallest capacity edge (bottleneck) on the path determines increase in flow in $G_f$. 
Take-aways from example

- The "middle edge" becomes two edges.
Take-aways from example

- The "middle edge" becomes two edges.
- The ad-hoc pushing back of flow in $G$ preserves flow conservation at all nodes and corresponds to a directed $(s, t)$-path of positive flow in $G_f$. 
Take-aways from example

- The "middle edge" becomes two edges.
- The ad-hoc pushing back of flow in $G$ preserves flow conservation at all nodes and corresponds to a \textit{directed} $(s, t)$-path of positive flow in $G_f$.
- Such a path can be found by BFS from $s$ in $G_f$. 
Take-aways from example

- The "middle edge" becomes two edges.
- The ad-hoc pushing back of flow in $G$ preserves flow conservation at all nodes and corresponds to a directed $(s, t)$-path of positive flow in $G_f$.
- Such a path can be found by BFS from $s$ in $G_f$.
- The smallest capacity edge (bottleneck) on the path determines increase in flow in $G$. 
Flow-Augmenting Paths

Inspired by the demo....
Inspired by the demo....

- Let $f$ be an $(s, t)$-flow in $G$,
Flow-Augmenting Paths

Inspired by the demo....

- Let $f$ be an $(s, t)$-flow in $G$,
- let $G_f$ be the residual graph of $f$ and $G$, 

- Let $b$ be the smallest capacity among the edges of $P$, called the bottleneck of $P$.
- So $b = c_f(\bar{e})$ for some $\bar{e} \in P$. 
Flow-Augmenting Paths

Inspired by the demo....

- Let $f$ be an $(s, t)$-flow in $G$,
- let $G_f$ be the residual graph of $f$ and $G$,
- let $P$ be an $(s, t)$-path in $G_f$ of positive capacity; that is, every edge of $e \in P$ has $c_f(e) > 0$. 
Inspired by the demo....

- Let $f$ be an $(s, t)$-flow in $G$,
- let $G_f$ be the residual graph of $f$ and $G$,
- let $P$ be an $(s, t)$-path in $G_f$ of positive capacity; that is, every edge of $e \in P$ has $c_f(e) > 0$.
- Let $b$ be the smallest capacity among the edges of $P$, called the *bottleneck* of $P$
Inspired by the demo....

- Let $f$ be an $(s, t)$-flow in $G$,
- let $G_f$ be the residual graph of $f$ and $G$,
- let $P$ be an $(s, t)$-path in $G_f$ of positive capacity; that is, every edge of $e \in P$ has $c_f(e) > 0$.
- Let $b$ be the smallest capacity among the edges of $P$, called the bottleneck of $P$
- So $b = c_f(\bar{e})$ for some $\bar{e} \in P$. 
Using $P$ to Improve $f$
Using $P$ to Improve $f$

- If $e \in E$ is a forward edge on $P$, then add $b$ to $f(e)$ in $G$.
Using $P$ to Improve $f$

- If $e \in E$ is a forward edge on $P$, then add $b$ to $f(e)$ in $G$.
- Note that
  \[ 0 \leq f(e) + b \leq f(e) + c_f(e) = f(e) + (c(e) - f(e)) = c(e) \] (feasible)
Using $P$ to Improve $f$

- If $e \in E$ is a forward edge on $P$, then add $b$ to $f(e)$ in $G$.
- Note that
  \[ 0 \leq f(e) + b \leq f(e) + c_f(e) = f(e) + (c(e) - f(e)) = c(e) \]  
  (feasible)
- If $e \in E$ is a backward edge on $P$, $(e^R = (v, u) \in P)$, subtract $c_f(e)$ from $f(e)$ in $G$.  

Using $P$ to Improve $f$

- If $e \in E$ is a forward edge on $P$, then add $b$ to $f(e)$ in $G$.
- Note that
  
  \[ 0 \leq f(e) + b \leq f(e) + c_f(e) = f(e) + (c(e) - f(e)) = c(e) \]  
  (feasible)

- If $e \in E$ is a backward edge on $P$, $(e^R = (v, u) \in P)$, subtract $c_f(e)$ from $f(e)$ in $G$.
- Note that
  
  \[ c(e) \geq f(e) > f(e) - b \geq f(e) - c_f(e) = f(e) - f(e) = 0 \]  
  (feasible)
Using $P$ to Improve $f$
Using $P$ to Improve $f$

- At every vertex $v \neq \{s, t\}$ of $P$, considering $v$ in $G$, the flows of exactly 2 edges incident with $v$ have been changed. No other internal vertices of $G$ have had flow changed.
Using $P$ to Improve $f$

- At every vertex $v \neq \{s, t\}$ of $P$, considering $v$ in $G$, the flows of exactly 2 edges incident with $v$ have been changed. No other internal vertices of $G$ have had flow changed.
- Consider the portion $v_{i-1}, e_i, v_i, e_{i+1}, v_{i+1}$ of $P$ in $G_f$. There are 4 cases
Using $P$ to Improve $f$

- At every vertex $v \neq \{s, t\}$ of $P$, considering $v$ in $G$, the flows of exactly 2 edges incident with $v$ have been changed. No other internal vertices of $G$ have had flow changed.
- Consider the portion $v_{i-1}, e_i, v_i, e_{i+1}, v_{i+1}$ of $P$ in $G_f$. There are 4 cases
  - Both $e_i, e_{i+1}$ are forward edges: So, $b$ additional units of flow enter $v_i$ from $e_i$ and then leave $v_i$ via $e_{i+1}$. 
  - Both $e_i, e_{i+1}$ are backward edges: Similar to previous case, but flow is subtracted.
  - $e_i$ is forward, $e_{i+1}$ is backward: So, $b$ additional units of flow enter $v_i$ from $e_i$ and $b$ fewer units of flow enter $v_i$ via $e_{i+1}$.
  - $e_i$ is backward, $e_{i+1}$ is forward: Similar to previous case.
- Thus flow is conserved at every vertex of $G$. 

Flows in Networks
Using $P$ to Improve $f$

- At every vertex $\nu \neq \{s, t\}$ of $P$, considering $\nu$ in $G$, the flows of exactly 2 edges incident with $\nu$ have been changed. No other internal vertices of $G$ have had flow changed.
- Consider the portion $\nu_{i-1}, e_i, \nu_i, e_{i+1}, \nu_{i+1}$ of $P$ in $G_f$. There are 4 cases
  - Both $e_i, e_{i+1}$ are forward edges: So, $b$ additional units of flow enter $\nu_i$ from $e_i$ and then leave $\nu_i$ via $e_{i+1}$.
  - Both $e_i, e_{i+1}$ are backward edges: Similar to previous case, but flow is subtracted.
Using $P$ to Improve $f$

- At every vertex $v \neq \{s, t\}$ of $P$, considering $v$ in $G$, the flows of exactly 2 edges incident with $v$ have been changed. No other internal vertices of $G$ have had flow changed.
- Consider the portion $v_{i-1}, e_i, v_i, e_{i+1}, v_{i+1}$ of $P$ in $G_f$. There are 4 cases
  - Both $e_i, e_{i+1}$ are forward edges: So, $b$ additional units of flow enter $v_i$ from $e_i$ and then leave $v_i$ via $e_{i+1}$.
  - Both $e_i, e_{i+1}$ are backward edges: Similar to previous case, but flow is subtracted.
  - $e_i$ is forward, $e_{i+1}$ is backward: So, $b$ additional units of flow enter $v_i$ from $e_i$ and $b$ fewer units of flow enter $v_i$ via $e_{i+1}^R$.

Thus flow is conserved at every vertex of $G$. 
Using $P$ to Improve $f$

- At every vertex $v \neq \{s, t\}$ of $P$, considering $v$ in $G$, the flows of exactly 2 edges incident with $v$ have been changed. No other internal vertices of $G$ have had flow changed.
- Consider the portion $v_{i-1}, e_i, v_i, e_{i+1}, v_{i+1}$ of $P$ in $G_f$. There are 4 cases
  - Both $e_i, e_{i+1}$ are forward edges: So, $b$ additional units of flow enter $v_i$ from $e_i$ and then leave $v_i$ via $e_{i+1}$.
  - Both $e_i, e_{i+1}$ are backward edges: Similar to previous case, but flow is subtracted.
  - $e_i$ is forward, $e_{i+1}$ is backward: So, $b$ additional units of flow enter $v_i$ from $e_i$ and $b$ fewer units of flow enter $v_i$ via $e_{i+1}^R$.
  - $e_i$ is backward, $e_{i+1}$ is forward: Similar to previous case.
Using $P$ to Improve $f$

- At every vertex $v \neq \{s, t\}$ of $P$, considering $v$ in $G$, the flows of exactly 2 edges incident with $v$ have been changed. No other internal vertices of $G$ have had flow changed.
- Consider the portion $v_{i-1}, e_i, v_i, e_{i+1}, v_{i+1}$ of $P$ in $G_f$. There are 4 cases
  - Both $e_i, e_{i+1}$ are forward edges: So, $b$ additional units of flow enter $v_i$ from $e_i$ and then leave $v_i$ via $e_{i+1}$.
  - Both $e_i, e_{i+1}$ are backward edges: Similar to previous case, but flow is subtracted.
  - $e_i$ is forward, $e_{i+1}$ is backward: So, $b$ additional units of flow enter $v_i$ from $e_i$ and $b$ fewer units of flow enter $v_i$ via $e_{i+1}^R$.
  - $e_i$ is backward, $e_{i+1}$ is forward: Similar to previous case.
- Thus flow is conserved at every vertex of $G$
Cuts Defined by Flows

Summarizing
Cuts Defined by Flows

Summarizing

Theorem

If $G_f$ contains an $(s, t)$-path $P$ then $f$ is not a maximum flow.
Cuts Defined by Flows

Summarizing

*Theorem*

If $G_f$ contains an $(s, t)$-path $P$ then $f$ is not a maximum flow.

$P$ is called a *flow-augmenting path*. Put another way
Cuts Defined by Flows

Summarizing

**Theorem**
If $G_f$ contains an $(s, t)$-path $P$ then $f$ is not a maximum flow.

$P$ is called a *flow-augmenting path*. Put another way

**Theorem**
If $f$ is a maximum flow then $G_f$ contains no flow-augmenting path.
Cuts Defined by Flows

Summarizing

Theorem
If $G_f$ contains an $(s, t)$-path $P$ then $f$ is not a maximum flow.

$P$ is called a flow-augmenting path. Put another way

Theorem
If $f$ is a maximum flow then $G_f$ contains no flow-augmenting path.

A flow $f$ creates a natural partition $[A, B]$ in $G$ where $A$ consists of all vertices reachable from $s$ by a directed path in $G_f$. If $t \notin A$ then $[A, B]$ is a cut, called the cut induced by $f$. 
Cuts Defined by Flows

Summarizing

*Theorem*

If $G_f$ contains an $(s, t)$-path $P$ then $f$ is not a maximum flow.

$P$ is called a *flow-augmenting path*. Put another way

*Theorem*

If $f$ is a maximum flow then $G_f$ contains no flow-augmenting path.

A flow $f$ creates a natural partition $[A, B]$ in $G$ where $A$ consists of all vertices reachable from $s$ by a directed path in $G_f$. If $t \notin A$ then $[A, B]$ is a cut, called the *cut induced by $f$*.

No flow-augmenting path in $G_f$ implies $t \notin A$, so $[A, B]$ is a cut.
**Induced Cuts are Minimum Capacity**

**Theorem**

Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$,
- for every $e \in [B, A]$, $f(e) = 0$.

So by the Flow Value Lemma, $\nu(f) = \operatorname{cap}(A, B)$: $f$ is a maximum flow & $[A, B]$ is a minimum capacity cut.
Induced Cuts are Minimum Capacity

**Theorem**

Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$

Proof.

- for $e = (u, v) \in [A, B]$, if $f(e) < c(e)$, $u$ is reachable from $s$ in $G$ since $c(e) = c(e) - f(e) > 0$

- for $e \in [B, A]$, if $f(e) > 0$, $u$ is reachable from $s$ in $G$ since $e$ is in $G$ and $c(e) = f(e) > 0$
Induced Cuts are Minimum Capacity

Theorem
Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$
- for every $e \in [B, A]$, $f(e) = 0$
**Induced Cuts are Minimum Capacity**

**Theorem**

Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$
- for every $e \in [B, A]$, $f(e) = 0$

So by the Flow Value Lemma, $v(f) = \text{cap}(A, B)$: $f$ is a maximum flow & $[A, B]$ is a minimum capacity cut.
Induced Cuts are Minimum Capacity

Theorem
Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$
- for every $e \in [B, A]$, $f(e) = 0$

So by the Flow Value Lemma, $v(f) = \text{cap}(A, B)$: $f$ is a maximum flow & $[A, B]$ is a minimum capacity cut.

Proof.
Induced Cuts are Minimum Capacity

Theorem
Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$
- for every $e \in [B, A]$, $f(e) = 0$

So by the Flow Value Lemma, $\nu(f) = \text{cap}(A, B)$: $f$ is a maximum flow & $[A, B]$ is a minimum capacity cut.

Proof.

- for $e = (u, v) \in [A, B]$, if $f(e) < c(e)$, $u$ is reachable from $s$ in $G_f$ since $c_f(e) = c(e) - f(e) > 0$
**Induced Cuts are Minimum Capacity**

**Theorem**
Suppose flow $f$ on $G$ has no flow-augmenting path and let $[A, B]$ be the cut induced by $f$. Then

- for every $e = (u, v) \in [A, B]$, $f(e) = c(e)$
- for every $e \in [B, A]$, $f(e) = 0$

So by the Flow Value Lemma, $v(f) = \text{cap}(A, B)$: $f$ is a maximum flow & $[A, B]$ is a minimum capacity cut.

**Proof.**

- for $e = (u, v) \in [A, B]$, if $f(e) < c(e)$, $u$ is reachable from $s$ in $G_f$ since $c_f(e) = c(e) - f(e) > 0$
- for $e \in [B, A]$, if $f(e) > 0$, $u$ is reachable from $s$ in $G_f$ since $e^R$ is in $G_f$ and $c_f(e^R) = f(e) > 0$
Pulling It All Together

To summarize

- If $G_f$ contains no flow-augmenting path, then there is a cut $[A, B]$ with $\nu(f) = \text{cap}(A, B)$.
- If there is a cut $[A, B]$ with $\nu(f) = \text{cap}(A, B)$, then $f$ is a maximum flow (and $[A, B]$ is a minimum capacity cut).
- If $f$ is a maximum flow, then $G_f$ has no flow-augmenting path.

Thus $f$ is a maximum flow if and only if $G_f$ has no flow-augmenting path if and only if $\nu(f) = \text{cap}(A, B)$ for some cut $[A, B]$. 
Flows in Networks

Pulling It All Together

To summarize

• If $G_f$ contains no flow-augmenting path, then there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$.
To summarize

• If $G_f$ contains no flow-augmenting path, then there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$.

• If there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$ then $f$ is a maximum flow (and $[A, B]$ is a minimum capacity cut).
Pulling It All Together

To summarize

• If $G_f$ contains no flow-augmenting path, then there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$.

• If there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$ then $f$ is a maximum flow (and $[A, B]$ is a minimum capacity cut).

• If $f$ is a maximum flow, then $G_f$ has no flow-augmenting path.
To summarize

- If $G_f$ contains no flow-augmenting path, then there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$.
- If there is a cut $[A, B]$ with $v(f) = \text{cap}(A, B)$ then $f$ is a maximum flow (and $[A, B]$ is a minimum capacity cut).
- If $f$ is a maximum flow, then $G_f$ has no flow-augmenting path.

Thus $f$ is a maximum flow if and only if $G_f$ has no flow-augmenting path if and only if $v(f) = \text{cap}[A, B]$ for some cut $[A, B]$. 