CMPSCI 611: Advanced Algorithms

Lecture 11: Network Flow

Andrew McGregor

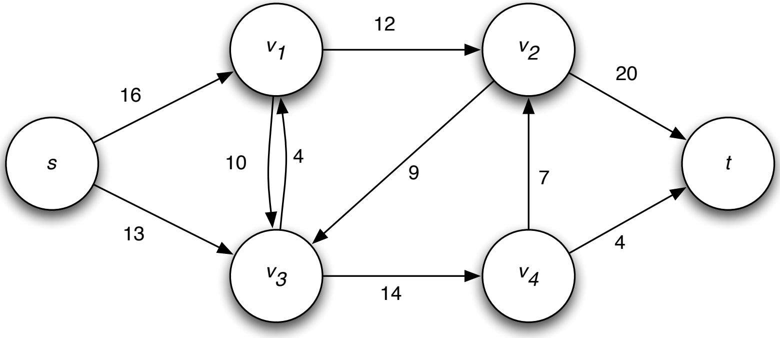
Last Compiled: April 4, 2024

Definitions

Input:

- Directed Graph $G = (V, E)$
- Capacities $C(u, v) > 0$ for $(u, v) \in E$ and $C(u, v) = 0$ for $(u, v) \notin E$
- A source node s, and sink node t

Capacity



Definitions

Input:

- Directed Graph $G = (V, E)$
- Capacities $C(u, v) > 0$ for $(u, v) \in E$ and $C(u, v) = 0$ for $(u, v) \notin E$
- A source node s, and sink node t

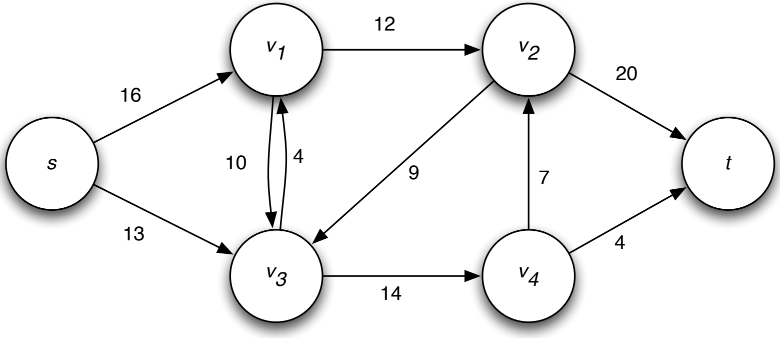
Output: A flow f from s to t where $f : V \times V \to \mathbb{R}$ satisfies

- Skew-symmetry: $\forall u, v \in V, f(u, v) = -f(v, u)$
- Conservation of Flow: $\forall v \in V - \{s, t\}, \sum_{u \in V} f(u, v) = 0$
- Capacity Constraints: $\forall u, v \in V,\ f(u, v) \leq C(u, v)$

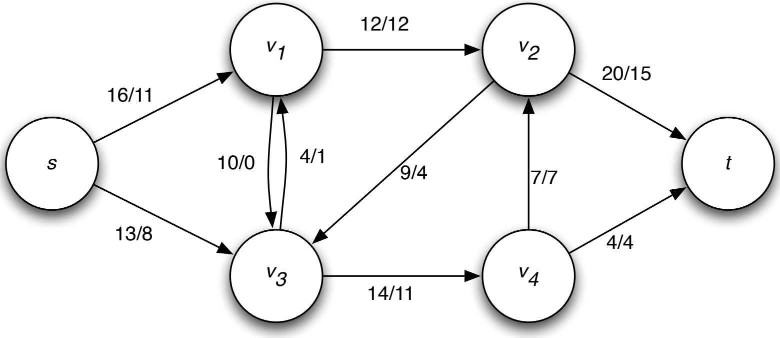
Goal: Maximize *"size of the flow"*, i.e., the total flow coming leaving s:

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

Capacity



Capacity/Flow



Cut Definitions

Definition

An $s - t$ cut of G is a partition of the vertices into two sets A and B such that $s \in A$ and $t \in B$.

Definition

The capacity of a cut (A, B) is

$$C(A, B) = \sum_{u \in A, v \in B} C(u, v)$$

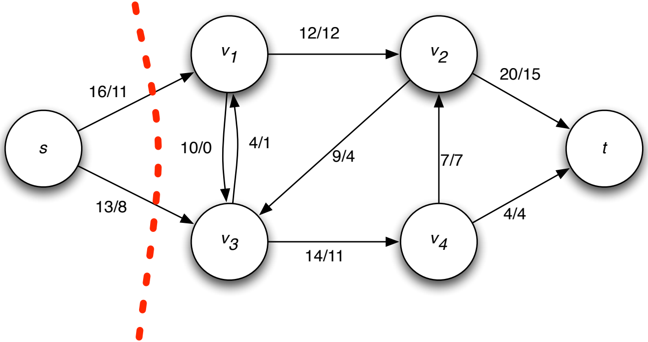
Definition

The flow across a cut (A, B) is

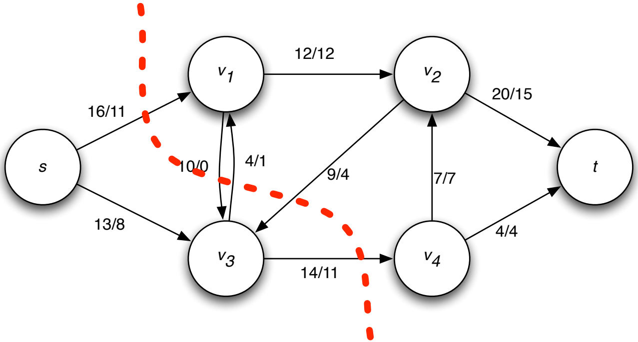
$$f(A, B) = \sum_{u \in A, v \in B} f(u, v)$$

Note that because of capacity constraints: $f(A, B) \leq C(A, B)$

First Cut



Second Cut



Max-Flow Min-Cut

Lemma

For any flow f: for all s-t cuts (A, B), $f(A, B)$ equals $|f|$.

Theorem (Max-Flow Min-Cut)

For any flow network and flow f, the following statements are equivalent:

1. *f is a maximum flow.*
2. *There exists an $s - t$ cut (A, B) such that $|f| = C(A, B)$*

We'll prove both next class.

Residual Networks and Augmenting Paths

Residual network encodes how you can change the flow between two nodes given the current flow and the capacity constraints.

Definition

Given a flow network $G = (V, E)$ and flow f in G, the residual network G_f is defined as

$$G_f = (V, E_f) \text{ where } E_f = \{(u, v) : C(u, v) - f(u, v) > 0\}$$

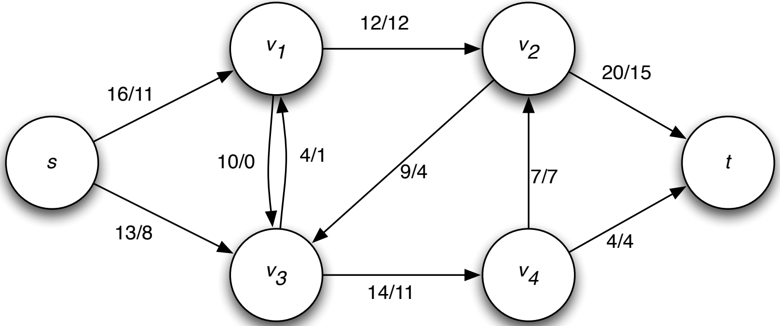
$$C_f(u, v) = C(u, v) - f(u, v)$$

Note that $(u, v) \in E_f$ implies either $C(u, v) > 0$ or $C(v, u) > 0$.

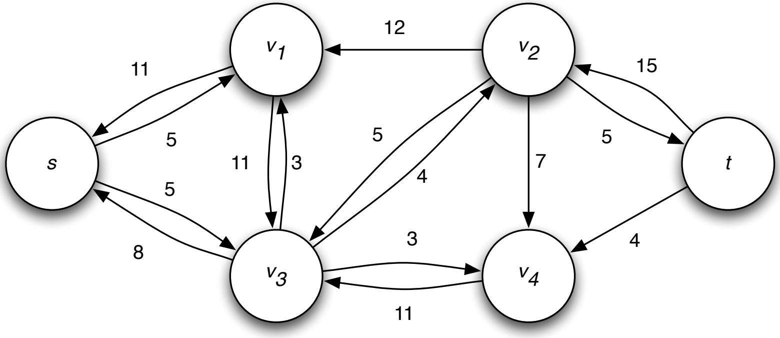
Definition

An augmenting path for flow f is a path from s to t in graph G_f. The bottleneck capacity $b(p)$ is the minimum capacity in G_f of any edge of p. We can increase flow by $b(p)$ along an augmenting path.

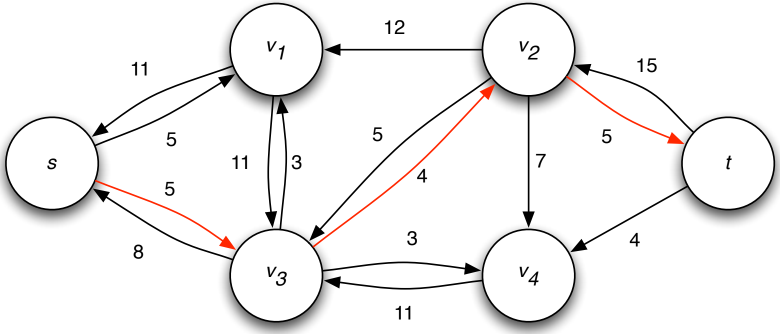
Capacity/Flow



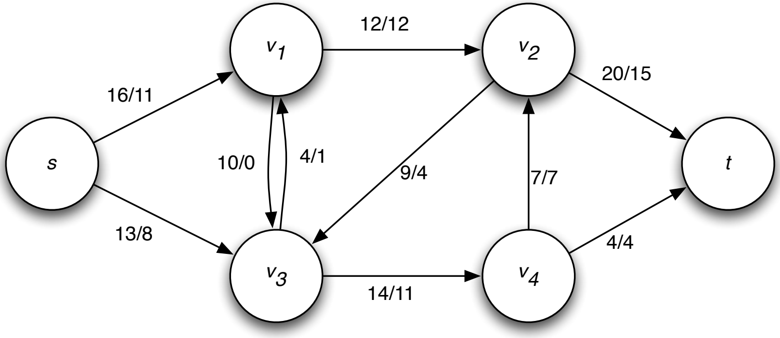
Residual



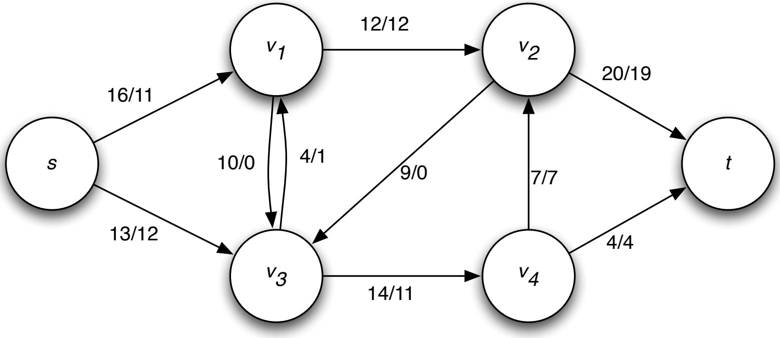
Augmenting Path



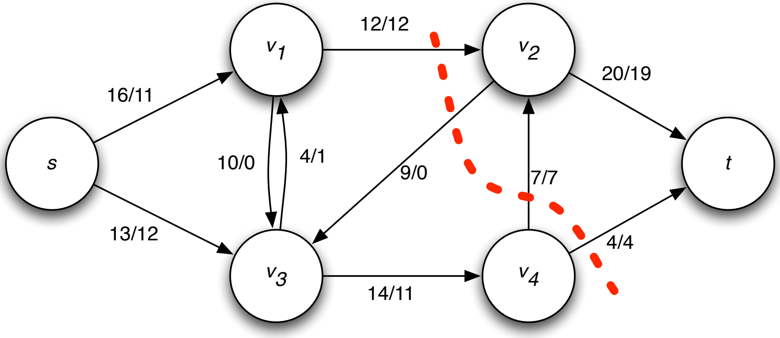
Old Flow



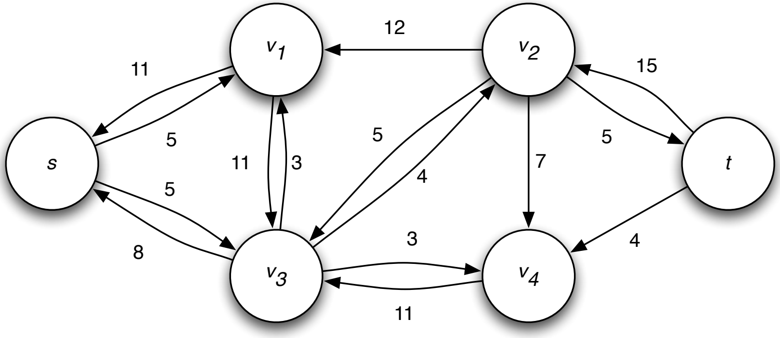
New Flow



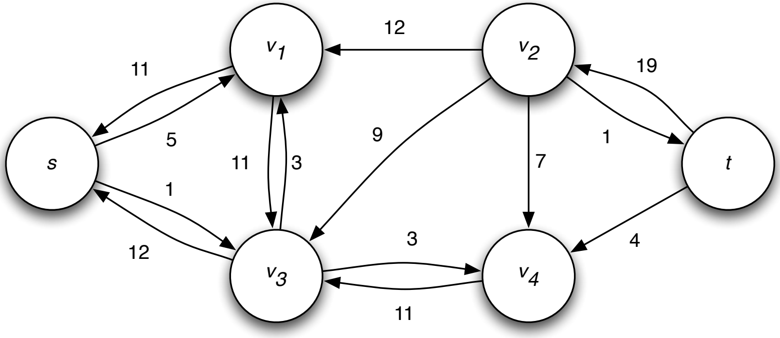
Min Capacity Cut Proves this is Optimal



Old Residual Graph



New Residual Graph



Ford-Fulkerson Algorithm

Algorithm

1. *flow* $f = 0$
2. *while there exists an augmenting path* p *for* f
   2.1 *find augmenting path* p
   2.2 *augment* f *by* $b(p)$ *units along* p
3. *return* f

Theorem

The algorithms finds a maximum flow in time $O(|E||f^*|)$ *if capacities are integral where* $|f^*|$ *is the size of the maximum flow.*

Proof.

$O(|E|)$ time to find each augmenting path via BFS and $|f^*|$ iterations because each augmenting path increases flow by at least 1. □

Blank Slide