Network Flows

Discrete Math, Spring 2025

Konstantin Chukharev

Network Flows

Motivation

TODO: a picture with a graph and a question about the flow in it

Flow Network

Definition 1: A *flow network* is a directed graph $G = \langle V, E \rangle$ with:

- a *source* $s \in V$, a vertex without incoming edges,
- a *sink* $t \in V$, a vertex without outgoing edges,
- a *capacity* function $c : E \to \mathbb{R}_+$ that assigns a non-negative capacity to each edge $e \in E$.

The flow network is denoted as $N = \langle V, E, s, t, c \rangle$.

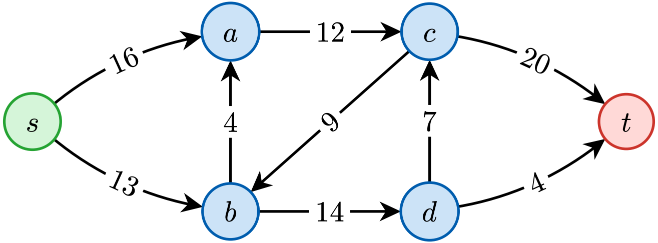
Note: We require that E never contains both edges (u, v) and (v, u) for any $u, v \in V$.

Note: If $(u, v) \notin E$, then $c(u, v) = 0$.

Note: The graph is connected, *i.e.*, every node has at least one incident edge.

Flow Network Example

Example: Very meaningful example of a flow network with annotated capacities:



Flow

Definition 2: Given a flow network N, a *flow* is a function $f : E \to \mathbb{R}_+$ that satisfies the following *feasibility* conditions:

1. *Capacity constraint*: $0 \leq f(e) \leq c(e)$ for each edge $e \in E$.
2. *Flow conservation (balance constraint)*: for each node $v \in V$, except for s and t,

$$\underbrace{\sum_{e \in \mathrm{in}(v)} f(e)}_{\text{flow into } v} = \underbrace{\sum_{e \in \mathrm{out}(v)} f(e)}_{\text{flow out of } v}$$

Note: If $(u, v) \notin E$, then $f(u, v) = 0$.

Flow Value

Definition 3: The *value* $|f|$ of a flow f is the total amount of flow that leaves the source s:

$$|f| = \sum_{e \in \operatorname{out}(s)} f(e) - \underbrace{\sum_{e \in \operatorname{in}(s)} f(e)}_{\text{commonly } 0}$$

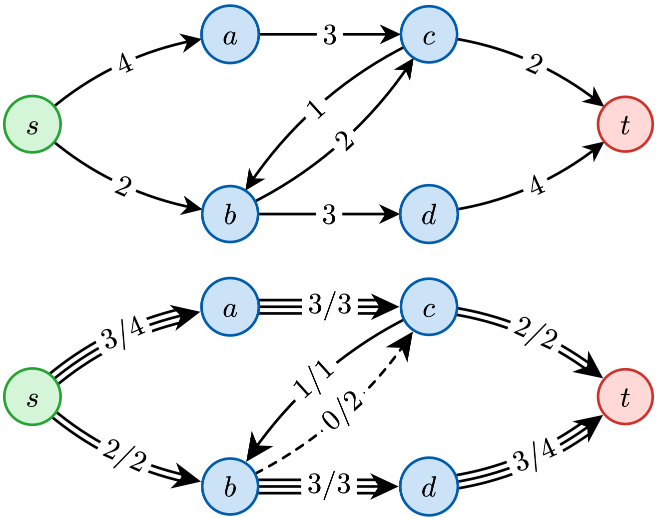
Note: $f^{\text{in}}(v) := \sum_{e \in \operatorname{in}(v)} f(e)$

Note: $f^{\text{out}}(v) := \sum_{e \in \operatorname{out}(v)} f(e)$

Definition 4 (Maximum Flow Problem): Given a flow network N, the *maximum flow problem* is to find a flow f that maximizes the value $|f|$.

Max Flow Example

Example: Yet another meaningful example.



Flow Conservation

Theorem 1: For any feasible flow f, the net flow out of s is equal to the net flow into t:

$$|f| = \sum_{e \in \mathrm{out}(s)} f(e) = \sum_{e \in \mathrm{in}(t)} f(e)$$

Proof: This follows directly from the flow conservation condition.

$$\begin{aligned} |f| = \sum_{e \in \mathrm{out}(s)} f(e) = \\ &= \sum_{e \in \mathrm{out}(s)} f(e) - \sum_{v \in V \setminus \{s,t\}} \left[\sum_{e \in \mathrm{in}(v)} f(e) - \sum_{e \in \mathrm{out}(v)} f(e) \right] = \\ &= \sum_{e \in \mathrm{in}(t)} f(e) \end{aligned}$$

$\square$

Residual Capacity

Definition 5: The *skew-symmetry* convention defines the flow in the opposite direction of an edge $e = (u, v)$ as $f(v, u) = -f(u, v)$.

Definition 6: Given a flow f in a flow network N, the *residual capacity* c_f of an edge e is the amount of flow that can be sent through the edge in addition to the flow already in it:

$$c_f(e) := c(e) - f(e)$$

Residual Network

Definition 7: The *residual network* N_f for a flow f is a flow network with the same vertices as N, constructed as follows:

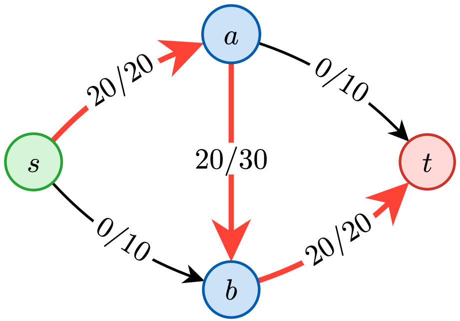
- *Forward edges*: For each edge $e = (u, v)$ of N, if $f(e) < c(e)$, add an edge $e' = (u, v)$ to N_f with capacity $c(e) - f(e)$.
- *Backward edges*: For each edge $e = (u, v)$ in N, if $f(e) > 0$, add a reversed edge $e' = (v, u)$ to N_f with capacity $f(e)$.

In other words, a residual network is a directed graph with *all* edges with *positive* residual capacity.

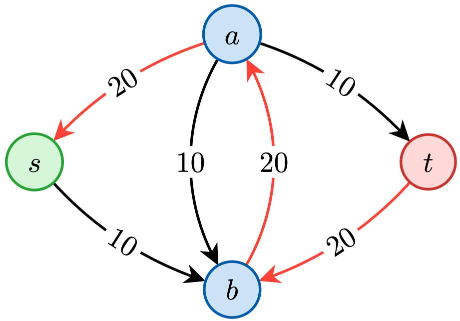
Residual Network Example

- *Remaining capacity:* If $f(e) < c(e)$, add edge e to N_f with capacity $c(e) - f(e)$.
- *Can erase up to $f(e)$ capacity:* If $f(u, v) > 0$, add reversed edge (v, u) to N_f with capacity $f(e)$.

Network N with flow f



Residual Network N_f



Augmenting Paths

Definition 8: An *augmenting path* in the residual network N_f is an s-t path (a path from s to t) such that all edges in the path have positive capacity. The *bottleneck* of an augmenting path is the minimum capacity of the edges in the path.

Theorem 2: If *bottleneck* is positive, then the flow can be increased by that amount along the path.

Ford-Fulkerson Algorithm

Input: A flow network N with source s and sink t.

Output: Maximum flow f from s to t.

Initialize $f(e) = 0$ for all $e \in E$

while there is an augmenting path P in the residual network N_f **do**

  Let $b = \min_{e \in P} c'(e)$ in N_f along P

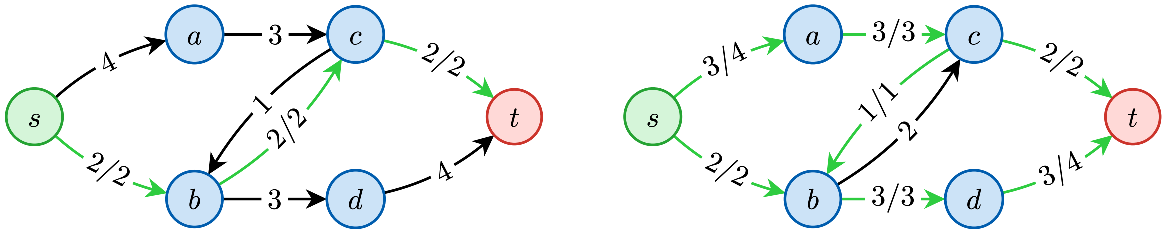
  for each edge $e \in P$ **do**

    Update flow: $f(e) := f(e) + b$

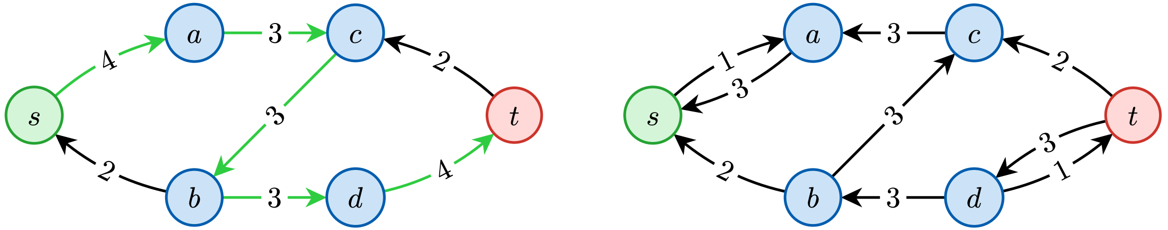
  Rebuild the residual network N_f

return f

Example



Networks (above) and residual networks (below) after pushing the flow with $|f| = 2$ along the path $s - b - c - t$ (left), and then after pushing the flow with $|f| = 3$ along the path $s - a - c - b - d - t$ (right).



Cuts

Definition 9: An *s-t cut* is a set of edges whose removal disconnects t from s.

Formally, a *cut* is a partition of the vertices $V = A \cup B$ such that $s \in A$ and $t \in B$. The edges of the cut are the edges that go from A to B.

Definition 10: The *capacity* of a cut (A, B) is the sum of the capacities of the edges leaving A.

$$c(A, B) = \sum_{a \in A, b \in B} c(a, b)$$

Definition 11: Given a flow f in N, the *net flow* across a cut (A, B) is defined as

$$f(A, B) = \sum_{e \in \operatorname{out}(A)} f(e) - \sum_{e \in \operatorname{in}(A)} f(e)$$

Cut Theorem 1

Theorem 3: Let f be a flow and (A, B) be an s-t cut. Then:

$$f(A, B) = |f|$$

Cut Theorem 2

Theorem 4: Let f be a flow and (A, B) be an s-t cut. Then:

$$f(A, B) \leq c(A, B)$$

Proof:

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

□

Max-Flow Min-Cut Theorem

Theorem 5: Given a flow network N and a flow f, the following are equivalent:

1. f is a *maximum flow* in N.
2. There is *no augmenting path* in the residual network N_f.
3. $|f| = c(A, B)$ for some s-t cut (A, B) in N.

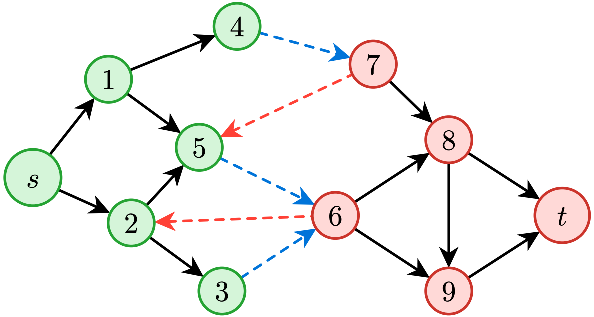
If one (and hence all) of these conditions hold, then (A, B) is a *minimum cut*.

Proof *(1 → 2)*: An augmenting path in the residual network N_f would allow us to increase the flow f. □

Proof *(3 → 1)*: No flow can exceed the capacity of a cut (by Theorem 4) □

Proof *(2 → 3)*: Let S be the set of vertices reachable from s in the residual network N_f. Since there is no augmenting path in N_f, S does not contain t. Then (S, T) is a cut of N, where $T = V \setminus S$. Moreover, for any $u \in S$ and $v \in T$, the residual capacity $c_f(e)$ must be zero (otherwise, the path $s \rightsquigarrow u$ in N_f could be extended to a path $s \rightsquigarrow u \to v$ in N_f). Thus, $f(S, T) = f^{\text{out}}(S) - f^{\text{in}}(S) = f^{\text{out}}(S) - 0 = c(S, T)$. □

Max-Flow Min-Cut Theorem [2]



- Cut (S, T) with $s \in S, t \in T$.
- Blue edges must be saturated.
- Red edges must be empty (zero).