Advanced algorithms and data structures

Lecture 2: Max Flow 2

Jacob Holm (jaho@di.ku.dk)

November 22nd 2023

Good Afternoon.

Today's Lecture

- Max flow
  - Recap
  - Ford-Fulkerson analysis
  - Edmods-Karp
  - Integrality Theorem
- Summary

Recap 1

Flow network (G, s, t, c), no self-loops or antiparallel edges.

Flow $f : V \times V \to \mathbb{R}$ satisfies:

1. $\forall u, v \in V : 0 \leq f(u, v) \leq c(u, v)$ (capacity constraints)
2. $\forall u \in V \setminus \{s, t\} : \sum_{v \in V} f(u, v) = \sum_{v \in V} f(v, u)$ (flow conservation)

Value $|f| = \sum_{v \in V} \big(f(s, v) - f(v, s)\big)$.

Residual capacity $c_f : c_f(u, v) = \begin{cases} c(u, v) - f(u, v) & \text{if } (u, v) \in E \\ f(v, u) & \text{if } (v, u) \in E \\ 0 & \text{otherwise} \end{cases}$

Residual network (G_f, s, t, c_f) where $G_f = (V, E_f)$ and
$E_f = \{(u, v) \in V \times V \mid c_f(u, v) > 0\}$. This is a flow network.

Given flow f in G and f' in G_f,
$(f \uparrow f')(u, v) = \begin{cases} f(u, v) + f'(u, v) - f'(v, u) & \text{if } (u, v) \in E \\ 0 & \text{otherwise} \end{cases}$

Recap 2

function FORD-FULKERSON($G = (V, E)$, s, t, c)
  $f \leftarrow 0$
  while $\exists$ (augmenting) path p from s to t in G_f **do**
    Find a max flow f_p along p in G_f.
    $f \leftarrow f \uparrow f_p$
  return f

A cut is a partition of V into sets $S \ni s$ and $T \ni t$.

$f(S, T) = \sum_{u \in S} \sum_{v \in T} f(u, v) - f(v, u)$.

$c(S, T) = \sum_{u \in S} \sum_{v \in T} c(u, v)$.

Lemma 1: $f \uparrow f'$ is a flow in G of value $|f \uparrow f'| = |f| + |f'|$.

Lemma 2: $\forall$ flows f and cuts (S, T), $|f| = f(S, T)$.

Corollary: $\forall$ flows f and cuts (S, T), $|f| \leq c(S, T)$.

Max flow/Min cut Theorem

Given a flow f in G, the following 3 statements are equivalent:

1. f is a max flow.
2. There is no augmenting path (a path $s \rightsquigarrow t$ in G_f).
3. $\exists$ cut (S, T) such that $|f| = c(S, T)$.

(1) $\implies$ (2).

Assume for contradiction f is a max flow and there exists an augmenting path p. Then by Lemma 1, $f \uparrow f_p$ is a flow in G of value $|f \uparrow f_p| = |f| + |f_p| > |f|$, which is a contradiction. $\square$

Max flow/Min cut Theorem

Given a flow f in G, the following 3 statements are equivalent:

1. f is a max flow.
2. There is no augmenting path (a path $s \rightsquigarrow t$ in G_f).
3. $\exists$ cut (S, T) such that $|f| = c(S, T)$.

(2) $\implies$ (3).

Let $S = \{v \in V \mid v \text{ is reachable from } s \text{ in } G_f\}$, $T = V \setminus S$. Then S, T partition V, $s \in S$ (why?) and $t \in T$ (why? no augmenting path), so (S, T) is a cut.
Now let $u \in S, v \in T$. If $(u, v) \in E$ then $f(u, v) = c(u, v)$ (why?), otherwise $c_f(u, v) > 0$ so $(u, v) \in E_f$. Since u is reachable from s in G_f that implies v is reachable from s in G_f thus $v \in S$ contradicting $v \in T = V \setminus S$.
If $(v, u) \in E$ then $f(v, u) = 0$ (why?), otherwise $c_f(u, v) > 0$ and same problem.
Thus

$$\begin{aligned}
|f| &= f(S, T) && \text{(By Lemma 2)} \\
&= \sum_{u \in S} \sum_{v \in T} (f(u, v) - f(v, u)) && \text{(Definition of } f(S, T)\text{)} \\
&= \sum_{u \in S} \sum_{v \in T} (c(u, v) - 0) && \text{(Above argument)} \\
&= c(S, T) && \text{(Definition of } c(S, T)\text{)}
\end{aligned}$$

$\square$

$s \in S$ because obviously s is reachable from itself.
$t \in T$ because otherwise t would be reachable from s by some path p, but this would be a augmenting path contradicting 2.

Max flow/Min cut Theorem

Given a flow f in G, the following 3 statements are equivalent:

1. f is a max flow.
2. There is no augmenting path (a path $s \rightsquigarrow t$ in G_f).
3. $\exists$ cut (S, T) such that $|f| = c(S, T)$.

(3) $\implies$ (1).

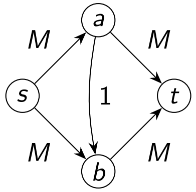
Let (S, T) be the cut from (3), and f' be any other flow in G. By the Corollary, $|f'| \leq c(S, T) = |f|$, so f is a max flow. $\square$

Ford-Fulkerson worst case analysis

In general Ford-Fulkerson is not guaranteed to terminate,
i.e. there exists flow networks and bad choices of augmenting paths such that it never terminates.
For this to happen, some capacities need to be irrational.

If all capacities are integers, F.F. does at most $|f^*|$ iterations, where f^* is a max flow (why?).
Assuming each iteration can be done in $\mathcal{O}(E)$ time, that gives a running time of $\mathcal{O}(E \cdot |f^*|)$.

Bad case example:



Edmonds-Karp algorithm avoids the bad case by always choosing the shortest augmenting path.

Edmonds-Karp Algorithm

function $\text{EDMONDS-KARP}(G = (V, E), s, t, c)$
  $f \leftarrow 0$
  while $\exists$ (augmenting) path from s to t in G_f **do**
    $p \leftarrow$ shortest such path.
    Find a max flow f_p along p in G_f.
    $f \leftarrow f \uparrow f_p$
  return f

Theorem

The number of iterations of Edmonds-Karp is $\mathcal{O}(V \cdot E)$.

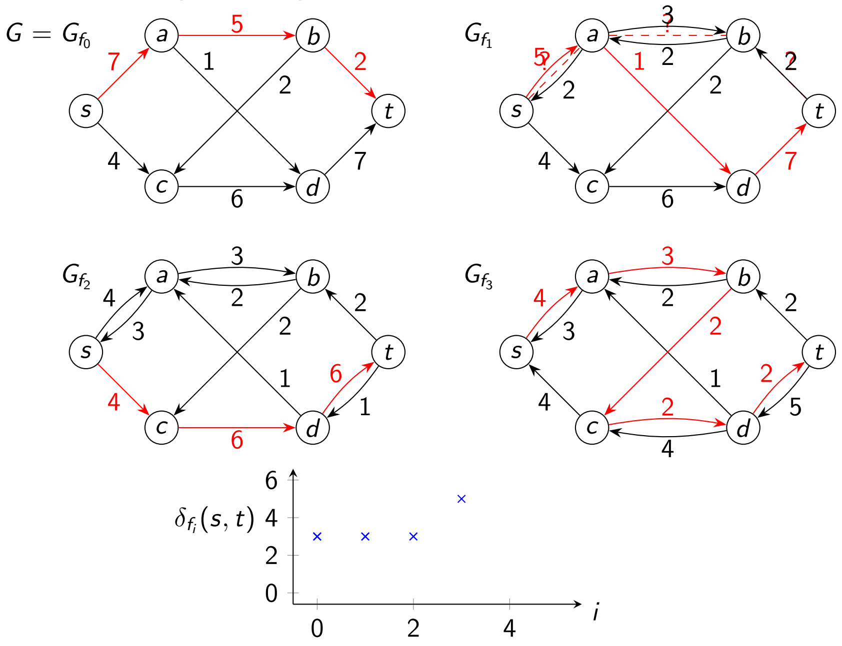
Corollary

Edmonds-Karp can be implemented to run in $\mathcal{O}(V \cdot E^2)$ time.

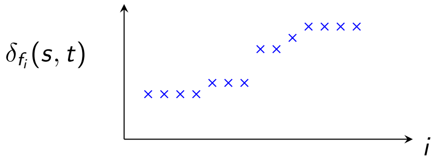
Thus, Edmonds-Karp is a polynomial-time algorithm for max flow.

Break

Edmonds-Karp Example



Edmonds-Karp Proof sketch of #iterations



We will show that the distance $\delta_{f_i}(s,t)$ is nondecreasing, and that it can only stay the same for at most E consecutive iterations.

This implies that #iterations in Edmonds-Karp is $\mathcal{O}(V \cdot E)$, why?

Each time the $s \rightsquigarrow t$ distance changes, it is increased by at least 1, and if a path exists the distance is at most $V - 1$. $\implies$ distance increases $\mathcal{O}(V)$ times.

So we have $\mathcal{O}(V)$ "runs", each of at most E consecutive iterations where the $s \rightsquigarrow t$ distance is unchanged. $\implies \mathcal{O}(V \cdot E)$ iterations in total.

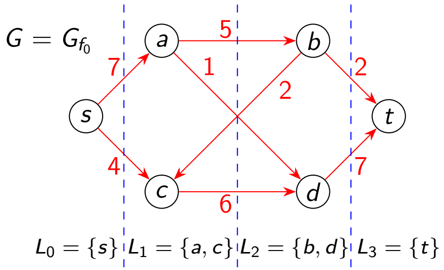
Edmonds-Karp Level sets and forward/backward edges

Consider consecutive flows $f_0, \ldots, f_k$ found by Edmonds-Karp, where $\delta_{f_0}(s,t) = \delta_{f_1}(s,t) = \ldots = \delta_{f_k}(s,t)$.

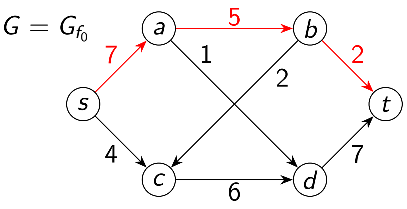
For $d = 0, 1, \ldots$ let
$L_d = \{v \in V \mid \delta_{f_0}(s,v) = d\}$, i.e.
L_d is the "BFS layer" consisting of vertices at distance d from s in G_{f_0}.

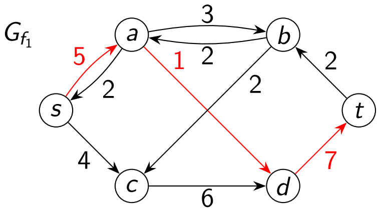
A forward edge (of G_{f_i}) is an edge $(u,v) \in G_{f_i}$ such that $u \in L_d$ and $v \in L_{d+1}$ for some d.

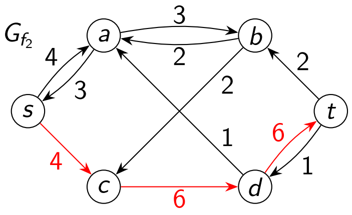
A backward edge (of G_{f_i}) is an edge $(u,v) \in G_{f_i}$ such that $u \in L_{d+1}$ and $v \in L_d$ for some d.



Edmonds-Karp Claim 1







Claim 1: For $i = 0, \ldots, k$, Edmonds-Karp finds an augmenting path in G_{f_i} consisting only of edges that are forward edges in G_{f_0}.

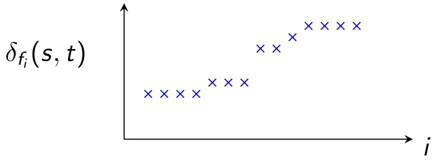
Note that Claim 1 implies $k = \mathcal{O}(E)$, why?

At least one forward edge gets saturated in iteration i, which removes it from $G_{f_{i+1}}$.

Edmonds-Karp Claim 2

Claim 2: If there is an augmenting path in $G_{f_{k+1}}$, then $\delta_{f_{k+1}}(s,t) \geq \delta_{f_k}(s,t)$.

Note that Claim 1 and Claim 2 together gives the claimed #iterations for Edmods-Karp.



Edmonds-Karp Proof sketch for Claim 1 & 2

Claim 1: For $i = 0, \ldots, k$, Edmonds-Karp finds an augmenting path in G_{f_i} consisting only of edges that are forward edges in G_{f_0}.

Proof sketch.

This is clear for $i = 0$, because a shortest path can only use forward edges.
So suppose $1 \leq i \leq k$. G_{f_i} is obtained from $G_{f_{i-1}}$ by removing forward edges (at least one!) and adding backward edges.
Since $\delta_{f_i}(s,t) = \delta_{f_0}(s,t)$ any shortest $s \rightsquigarrow t$ path can only use forward edges, which must have stayed forward edges since G_{f_0}.
The claim follows by induction. □

Claim 2: If there is an augmenting path in $G_{f_{k+1}}$, then $\delta_{f_{k+1}}(s,t) \geq \delta_{f_k}(s,t)$.

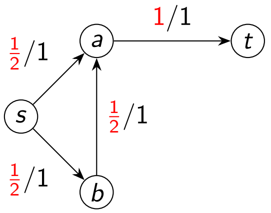
Proof sketch.

$G_{f_{k+1}}$ is obtained from G_{f_0} by removing forward edges and adding backward edges. This can never reduce the distance. □

Integrality Theorem

Integrality Theorem: Given integer capacities, Ford-Fulkerson (and therefore Edmonds-Karp) will find an integer-valued flow $f : V \times V \to \mathbb{Z}_{\geq 0}$ with $|f| \in \mathbb{Z}_{\geq 0}$ an integer.

Note that not all max flows in a network with integer capacities have to be integer-valued. Q: Can you find a max flow in this example that is not integer-valued?



Summary

This finished the topic on Max Flow. We have covered

- Proof of Max flow/Min cut Theorem
- Worst case analysis of Ford-Fulkerson
- Edmonds-Karp Algorithm
- Integrality Theorem

Next time:

- Linear Programming