The Design and Analysis of Algorithms

Lecture 14 Network Flow I

Zhenbo Wang

Department of Mathematical Sciences, Tsinghua University

Content

Maximum Flow and Minimum Cut

Ford-Fulkerson Algorithm

Maximum Flow and Minimum Cut

- Max flow and min cut.

  Two very rich algorithmic problems.

  Cornerstone problems in combinatorial optimization.

  Beautiful mathematical duality.

- Applications.

  Network reliability.

  Data mining.

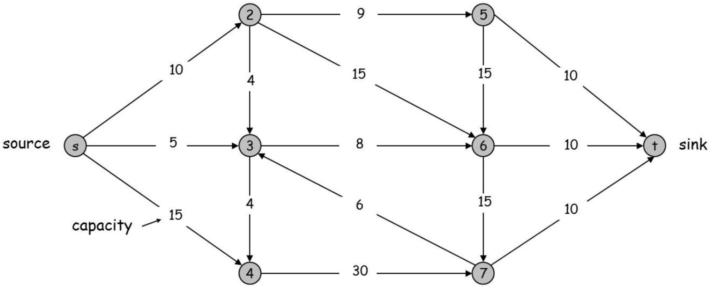
  Airline scheduling.

  Image segmentation.

  Bipartite matching $\cdots$

Flow Network

- Abstraction for material flowing through the edges.
- Digraph $G = (V, E)$ with source $s \in V$ and sink $t \in V$.
- Nonnegative integer capacity $c(e)$ for each $e \in E$.

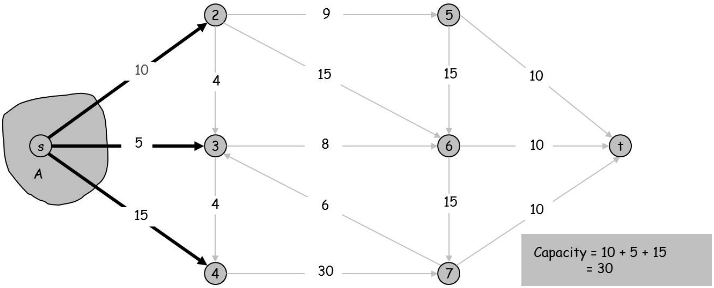


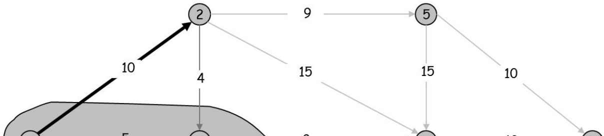
Minimum Cut Problem

Def. An $s-t$ cut is a partition (A, B) of V with $s \in A$ and $t \in B$.

Def. The capacity of a cut (A, B) is: $cap(A, B) = \sum_{\text{e out of A}} c(e)$.

Min-cut Find a cut of minimum capacity.





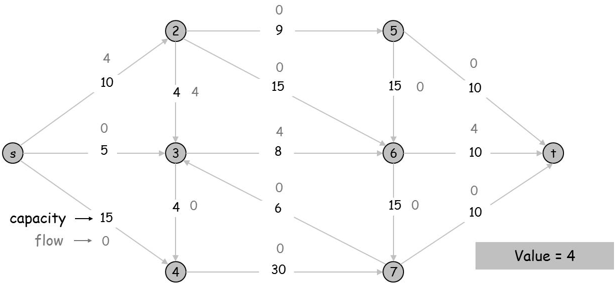
Maximum Flow Problem

Def. An $s - t$ flow is a function that satisfies:

- For each $e \in E$: $0 \leq f(e) \leq c(e)$ (capacity)
- For each $v \in V - \{s, t\}$: $\sum_{\text{e in to v}} f(e) = \sum_{\text{e out of v}} f(e)$ (conservation)

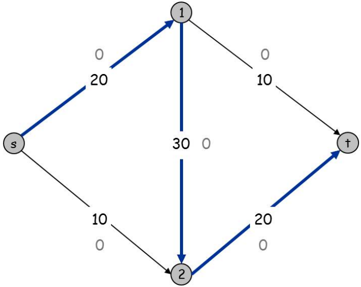
Def. The value of a flow f is: $v(f) = \sum_{\text{e out of s}} f(e)$.

Max-flow Find a flow of maximum value.

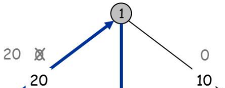


Greedy Algorithm

- Start with $f(e) = 0$ for all edge $e \in E$.
- Find an $s - t$ path P where each edge has $f(e) < c(e)$.
- Augment flow along path P.
- Repeat until you get stuck.

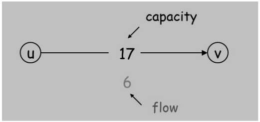


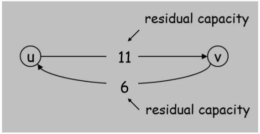
Flow value = 0



Residual Graph

- Original edge: $e = (u, v) \in E$, flow $f(e)$, capacity $c(e)$.
- Residual edge, $e = (u, v)$ and $e^R = (v, u)$.
- Residual capacity: $c_f(e) = \begin{cases} c(e) - f(e) & \text{if} \quad e \in E \\ f(e) & \text{if} \quad e^R \in E \end{cases}$
- Residual graph: $G_f = (V, E_f)$.
- Residual edges with positive residual capacity.





Augmenting Path

Def. An augmenting path is a simple $s - t$ path P in the residual graph G_f.

Def. The bottleneck capacity of an augmenting P is the minimum residual capacity of any edge in P.

- Let f be a flow and let P be an augmenting path in G_f . Then f' is a flow and $val(f') = val(f) + bottleneck(G_f, P)$.

$AUGMENT(f, c, P)$

```
b ← bottleneck capacity of path P.
for edge e ∈ P do
    if e ∈ E then
        f(e) ← f(e) + b.
    else
        f(e^R) ← f(e^R) − b.
    end if
end for
return f.
```

Ford-Fulkerson Algorithm

- Start with $f(e) = 0$ for all edge $e \in E$.
- Find an augmenting path P in the residual graph G_f
- Augment flow along path P.
- Repeat until you get stuck.

$FORD - FULKERSON(G, s, t, c)$

for edge $e \in P$ **do**
    $f(e) \leftarrow 0$.
end for
while (there exists an augmenting path P in G_f **do**
    $f \leftarrow AUGMENT(f, c, P)$.
    Update G_f.
end while
return f.

Relationship between Flows and Cuts

Flow value lemma

Let f be any flow and let (A, B) be any cut. Then, the net flow across (A, B) equals the value of f, i.e.

$\sum_{\text{e out of A}} f(e) - \sum_{\text{e in to A}} f(e) = v(f).$

Pf.

$$\begin{aligned} v(f) &= \sum_{\text{e out of s}} f(e) \\ &= \sum_{v \in A} \Big(\sum_{\text{e out of v}} f(e) - \sum_{\text{e in to v}} f(e) \Big) \\ &= \sum_{\text{e out of A}} f(e) - \sum_{\text{e in to A}} f(e). \quad \square \end{aligned}$$

Relationship between Flows and Cuts

Weak duality

Let f be any flow and (A, B) be any cut. Then, $v(f) \leq cap(A, B)$.

Pf.

$$\begin{aligned} v(f) &= \sum_{\text{e out of A}} f(e) - \sum_{\text{e in to A}} f(e) \\ &\leq \sum_{\text{e out of A}} f(e) \\ &\leq \sum_{\text{e out of A}} c(e) \\ &= cap(A, B). \quad \square \end{aligned}$$

Max-Flow Min-Cut Theorem

Augmenting path theorem

Flow f is a max flow *iff* there are no augmenting paths.

Max-flow min-cut theorem

The value of the max flow is equal to the value of the min cut.

Pf. The following three conditions are equivalent for any flow f:

i. There exists a cut (A, B) such that $cap(A, B) = val(f)$.

ii. f is a max-flow.

iii. There is no augmenting path with respect to f.

Max-Flow Min-Cut Theorem–Con't

$i \Rightarrow ii$ Via weak duality lemma.

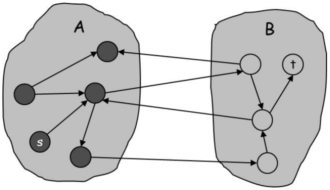
$ii \Rightarrow iii$ Can improve flow f if a augmenting path exists.

$iii \Rightarrow i$ Let f be a flow with no augmenting paths.

Let A be set of vertices reachable from s in residual graph.

Then $s \in A$ and $t \notin A$.

$$
\begin{aligned}
v(f) &= \sum_{\text{e out ot A}} f(e) - \sum_{\text{e in to A}} f(e) \\
&= \sum_{\text{e out ot A}} c(e) \\
&= cap(A, B). \quad \square
\end{aligned}
$$



original network

Running Time

- Assumption. Capacities are integers between 1 and C.

Theorem 1

The algorithm terminates in at most nC iterations.

Corollary 2

The running time of Ford-Fulkerson is $O(mnC)$.

Corollary 3

If $C = 1$, the running time of Ford-Fulkerson is $O(mn)$.

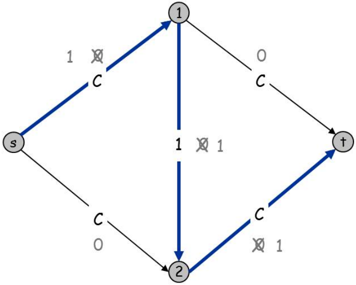
Integrality theorem

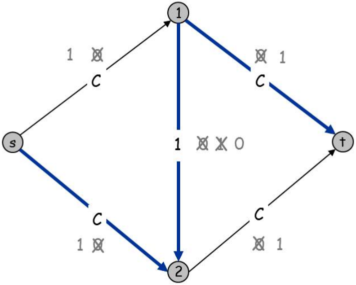
Then exists a max-flow for which every flow value is an integer.

Exponential Number of Augmentations

Q. Is Ford-Fulkerson algorithm polynomial in input size?

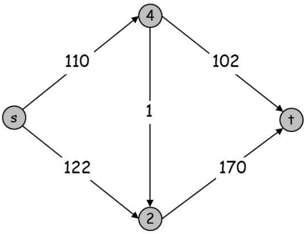
A. No.



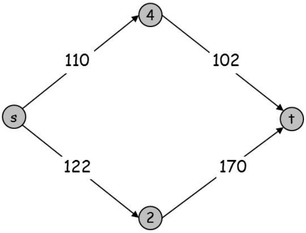


Capacity Scaling Algorithm

- *Intuition*. Choosing path with highest bottleneck capacity.
- Maintain scaling parameter Δ.
- Let $G_f(\Delta)$ be the subgraph of the residual graph consisting of only arcs with capacity at least Δ.



G_f



$G_f(100)$

Capacity Scaling Algorithm

$CAPACITY-SCALING(G, s, t, c)$

$\Delta \leftarrow$ largest power of $2 \leq C$.
for edge $e \in E$ **do**
    $f(e) \leftarrow 0$.
end for
while $\Delta \geq 1$ **do**
    $G_f(\Delta) \leftarrow \Delta$–residual graph.
    while there exists an augmenting path P in $G_f(\Delta)$ **do**
        $f \leftarrow AUGMENT(f, c, P)$.
    end while
    $\Delta \leftarrow \Delta/2$.
end while
return f.

Homework

- Read Chapter 7 of the textbook.
- Exercises 4 & 5 in Chapter 7.