Maximum flows, minimum cuts, and longest paths

Steph van Willigenburg
Math 442-201 2019WT2

2 April 2020

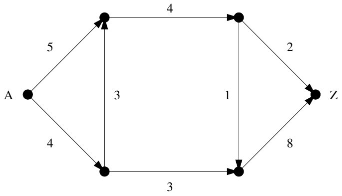
Networks

Definition

Let D be a connected digraph. Then it is a network if each arc $a \in A(D)$ has a weight, called capacity $c(a)$.

Let v be a vertex. The sum of capacities of arcs coming in is the in-flow, $inflow(v)$. The sum of capacities of arcs going out is the out-flow, $outflow(v)$.

Example Below we have $inflow(v) =$ $outflow(v) =$



Flows

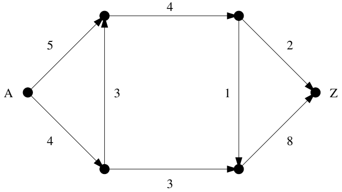
Definition

Let D be a network with exactly one source A and one sink Z. Then a flow is a function that assigns a flow in a, $\phi(a) \geq 0$ to every $a \in A(D)$ so

1. $\phi(a) \leq c(a)$
2. $inflow(v) = outflow(v)$ for all $v \in V(D)$ and $v \neq A, Z$.

Note: $\phi(a) = 0 \ \forall a$ is a zero-flow; $\phi(a) = c(a)$ is a saturated arc.

Example



Flow value

Definition

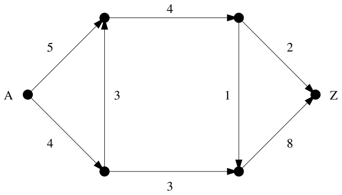
Let D be a network with exactly one source A, one sink Z, and a flow ϕ. We call

$$\mathit{inflow}(Z) = \mathit{outflow}(A)$$

the flow value.

Goal: We want to maximize this - find the maximum flow.

Example Below we have flow value = max. flow value =



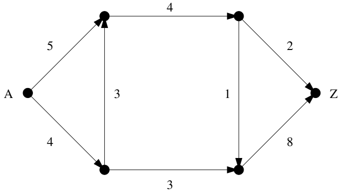
Cuts

Definition

Let D be a network with exactly one source A and one sink Z. A set of arcs S whose deletion disconnects A and Z is a cut. The sum of capacities of $s \in S$ going from component with A to component with Z is capacity of cut.

Goal: We want to minimize this - find the minimum cut.

Example



Max flow-Min cut

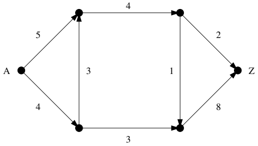
Theorem

Let D be a network with exactly one source A and one sink Z.

$$\begin{array}{c} \textit{The maximum} \\ \textit{flow value} \end{array} = \begin{array}{c} \textit{The capacity} \\ \textit{of a} \\ \textit{minimum cut.} \end{array}$$

Note: Find any flow $=$ any cut to answer. Proof is about 45 mins!

Example max. flow $=$ $\qquad$ $=$ min. cut.



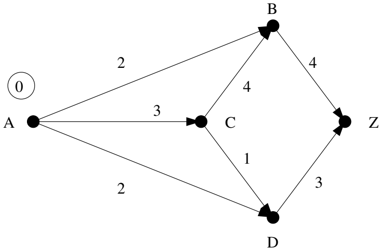
Algorithm for longest path

Let D be an acyclic network with source A and sink Z and $\ell(v)$ denote the label of v. Compute the longest distance from vertices A to Z.

- Let $\ell(A) = 0$. Make it permanent. Choose a vertex v all of whose arcs coming in have permanent labels $v_1, \ldots, v_k$.

Note: Permanent labels cannot be changed.

Example



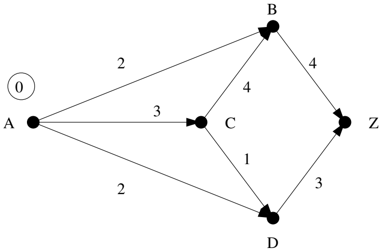
Algorithm for longest path

Let D be an acyclic network with source A and sink Z and $\ell(v)$ denote the label of v. Compute the longest distance from vertices A to Z.

- Find $\max\{\ell(v_i) + d_i\}_{i=1}^{k}$ where $v_i \overset{d_i}{\to} v$ and make this permanent.
- Repeat until all vertices have permanent labels.

Note: Permanent labels cannot be changed.

Example



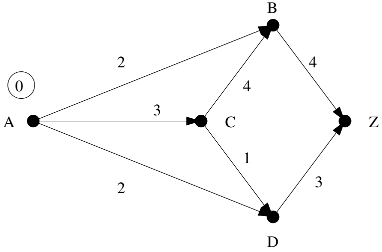
Algorithm for longest path

Let D be an acyclic network with source A and sink Z and $\ell(v)$ denote the label of v. Compute the longest distance from vertices A to Z.

- The longest distance from A to Z is label at Z. The longest path found by starting at Z, include arc capacity d between v, w if

$$\ell(w) - \ell(v) = d.$$

Example Longest distance = Longest path =



What we would normally do final class...

This can take all class and is a fun application of our theory:

In summary

- Studied networks
- Studied flows and maximum flows
- Studied cuts in networks
- Max-flow min-cut theorem
- Algorithm for longest path

Thanks for a great Math 442!