MaxFlow and Reductions

Attribution

- These slides were prepared for the New Jersey Governor’s School course “The Math Behind the Machine” taught in the summer of 2012 by Grant Schoenebeck
- Some of these slides were copied or modified from a previous years’ courses given by Troy Lee in 2010, and the exposition of MaxFlow is based closely upon the exposition in *Algorithm Design* by Kleinberg and Tardos, the exposition of Perfect Matchings is based closely upon Chapter 10 of *Networks, Crowds, and Markets* by Kleinberg and Easley.

Polynomial Time

We have mentioned efficient algorithms several times in this course.

An efficient algorithm is generally taken to mean one whose running time depends polynomially on the size of the input.

Examples

Solving n linear equations in n unknowns.

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_1 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n
\end{aligned}
$$

Gaussian elimination---time about n^3

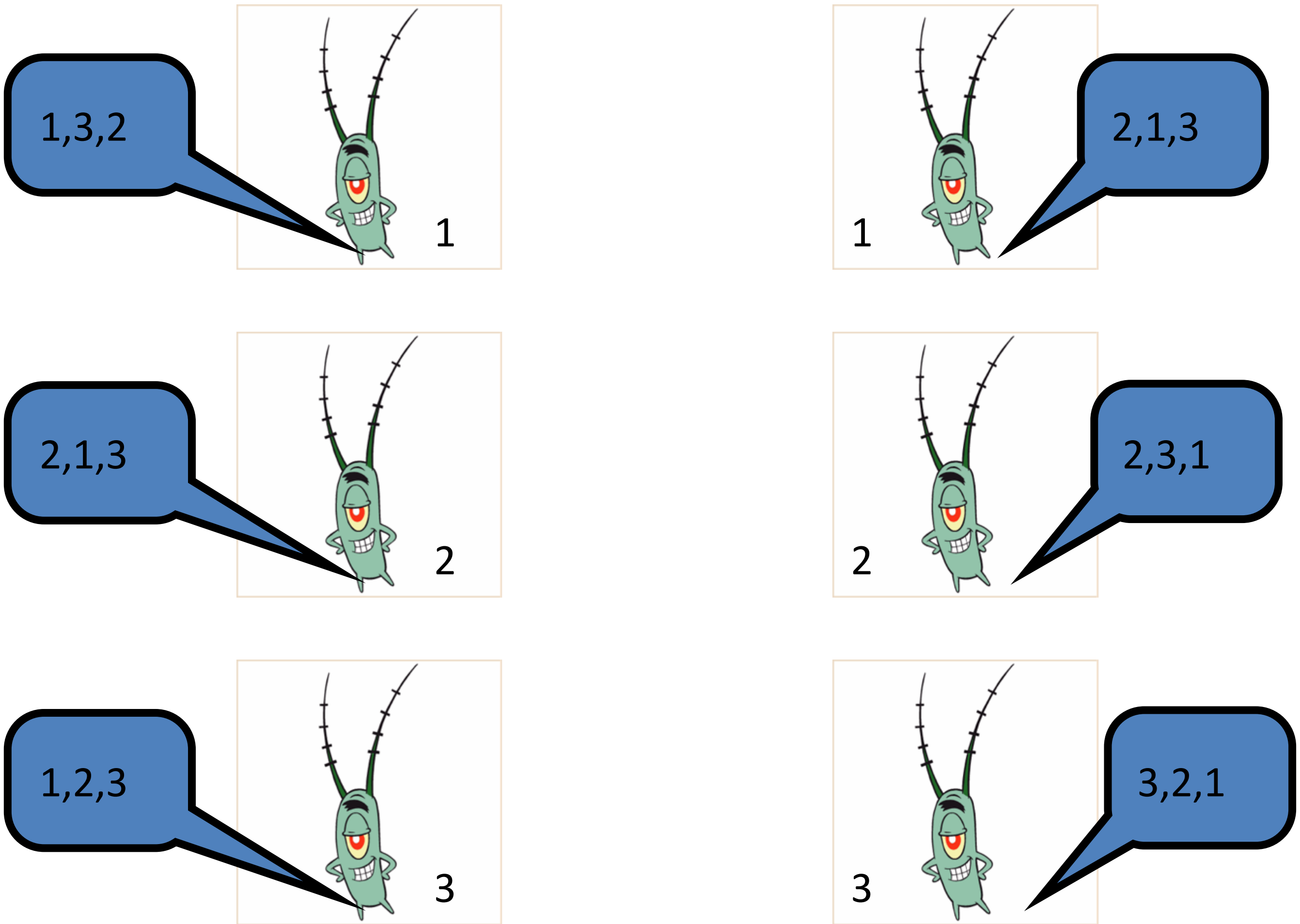
Examples

Linear programming problems.

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &\quad \leq b_1 \\
a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &\quad \leq b_2 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + \ldots + \ldots a_{nn}x_n &\quad \leq b_n \\
x_1, x_2, \ldots, x_n &\quad \geq 0
\end{aligned}
$$

Time about n^4

Examples



Find a stable pairing.

Time about n^3

Chinese Postman Problem

- Given a graph, what is the length of the shortest tour which traverses all edges?
  - Note that here you can---and may have to---use an edge more than once.

More Coming…

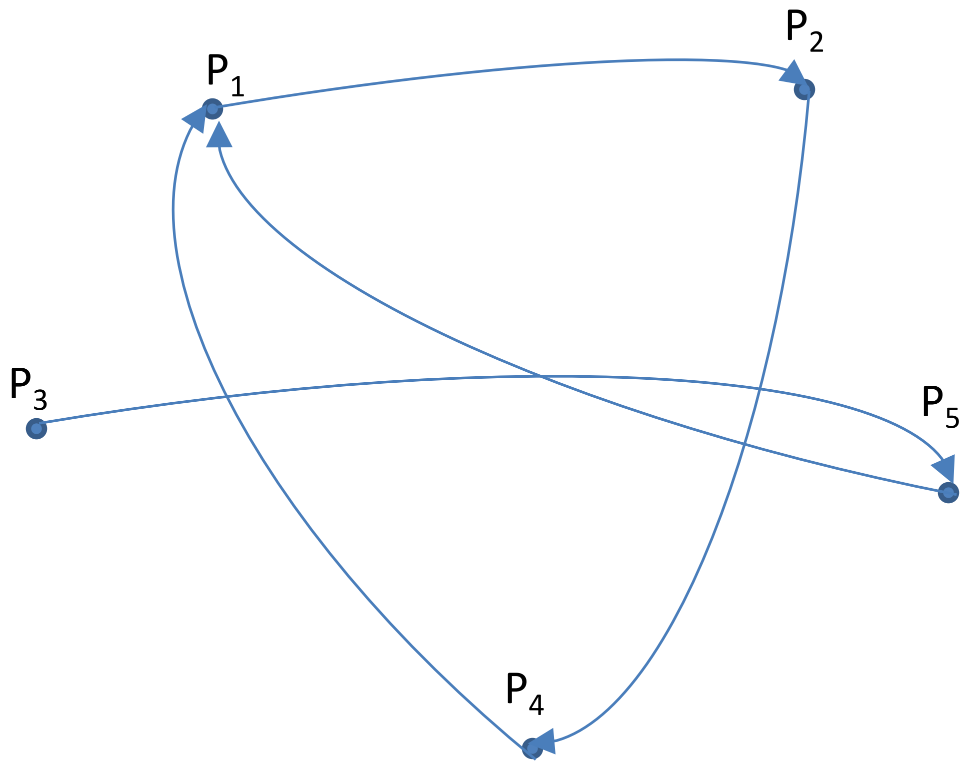
- Today
  - Graph connectivity
  - MaxFlow
  - Perfect Bipartite Matching
- Tomorrow
  - Linear Programming

Reasonable Model?

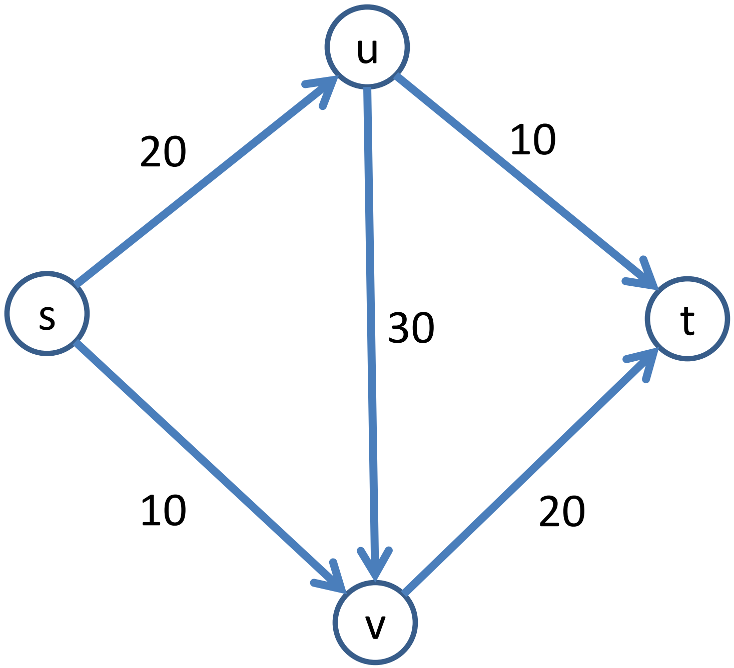
- An algorithm that takes $10^{67}n$ time would not be very practical...
- Even Gaussian elimination is currently not practical with a million unknowns.
- This model does offer very nice mathematical properties: for example, robustness and closure under composition.

Graph Connectivity

- Directed graph $G = (V, E)$, s, t



Network Flow



Network Flow

Network

- Directed graph G=(V, E)
- Associate c_e the capacity of edge e, positive integer.
- Source $s \in V$, sink $t \in V$.

Flow

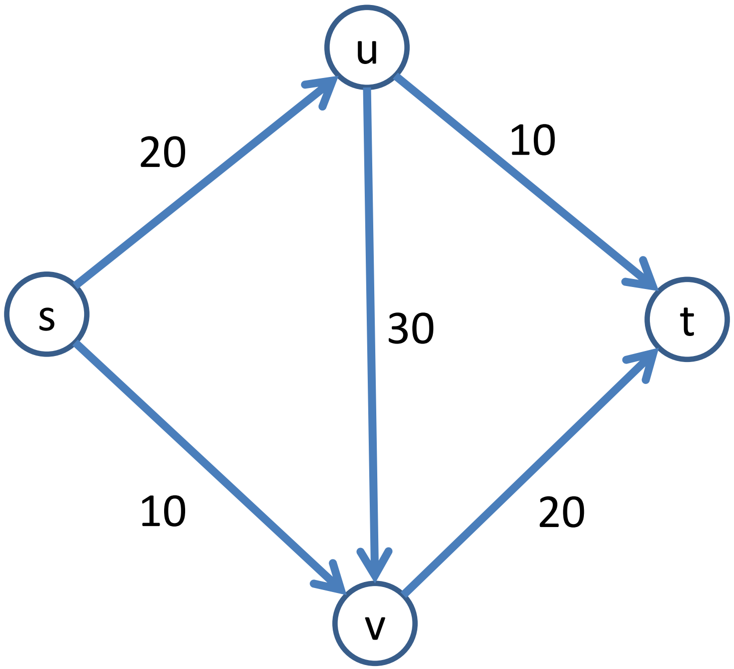
- A flow $f: E \rightarrow R^+$ assigned the amount of flow along each edge such that
  - Flow conserved;
  - Flows obey capacity

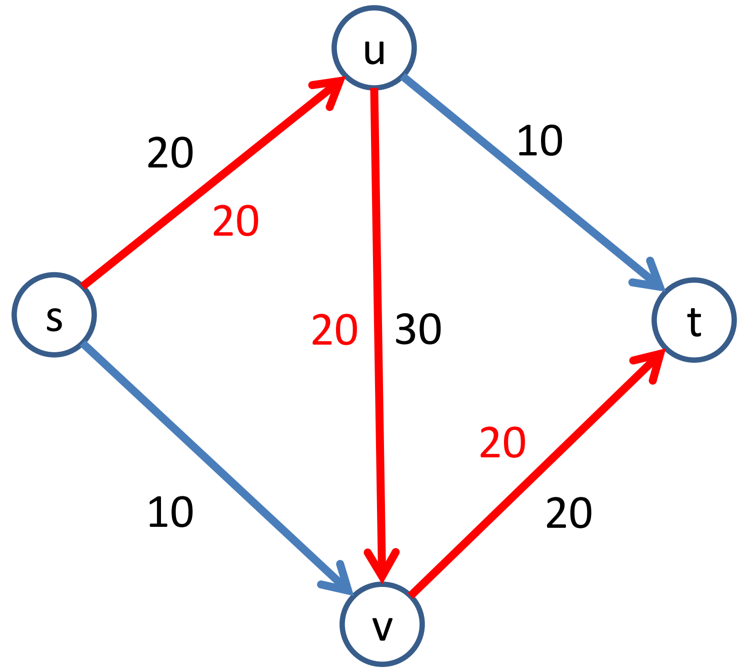
Max Flow

- Given a network, what is the maximum flow (sum of flows entering sink) possible

Max Flow Algorithm

- Try Greedy

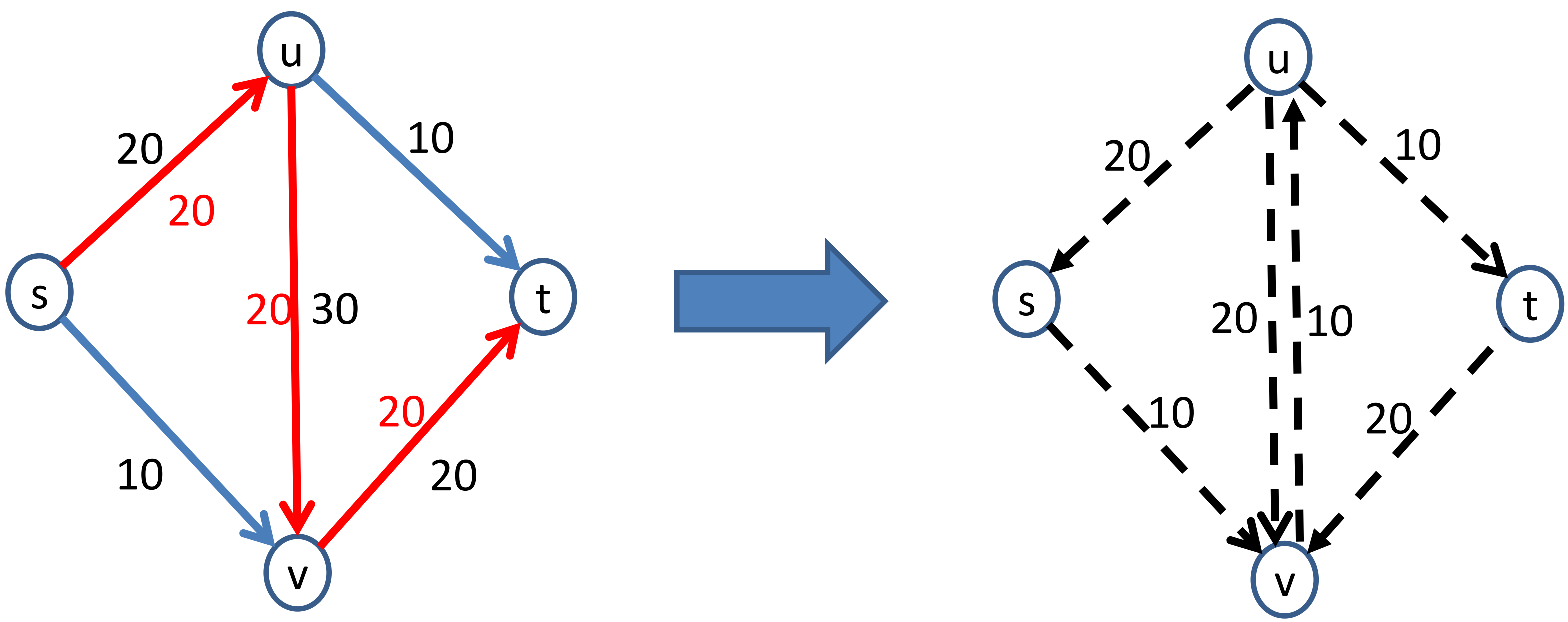




- Now What?

Residual Graph

- Nodes the same
- Edge in if $f(e) < c_e$ (with capacity $c_e - f(e)$)
- Opposite edge in if $0 < f(e)$ (with capacity $f(e)$)



Augmenting Paths

- Augmenting Path
  - A path from source to sink in residual graph
- After adding augmenting paths:
  - Capacity condition satisfied
  - Conservation condition satisfied

Ford Fulkerson Algorithms

- While(there exists augmenting path)
  - Push flow
- Does this terminate?

s-t Cuts

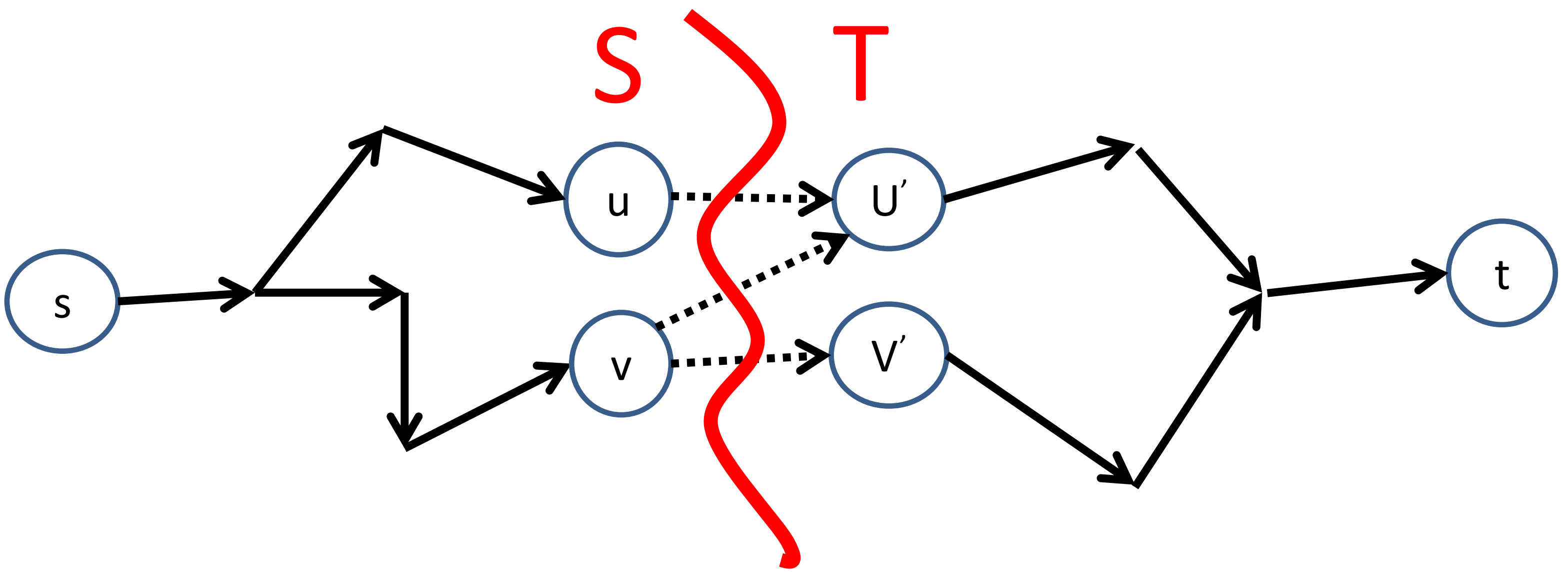
- An s-t Cut (S, T)
  - $S \subseteq V$
  - $T \subseteq V$
  - $s \in S$
  - $t \in T$
  - $S \cup T = V$
  - $S \cap T = \emptyset$
- Capacity of an s-t Cut (S, T) is $\sum_{(u,v)\in E, u\in S, v\in T} c_e = M$
- Impossible to push more than M flow from s to t.

MaxFlow $\leq$ MinCut

- If Capacity of s-t Cut (S, T) is $\sum_{(u,v)\in E, u\in S, v\in T} c_e = M$
- than impossible to push more than M flow from s to t.

MaxFlow ≥ MinCut

- Say the there is no augmenting path
- Let all vertices reachable by s in the residual graph be S.
- Let the rest of the vertices be T.
- No edges from S to T in the residual graph
- So the capacity of the (S, T) cut is equal to the flow!

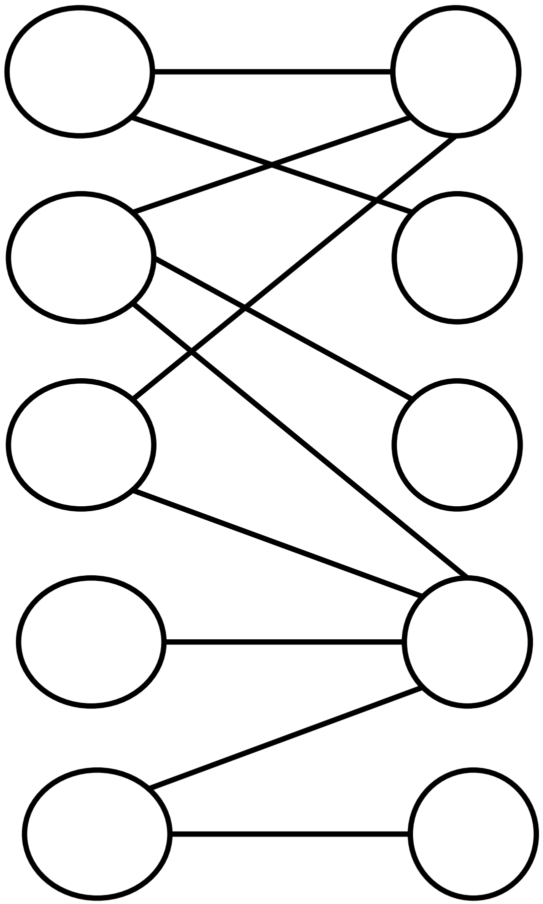


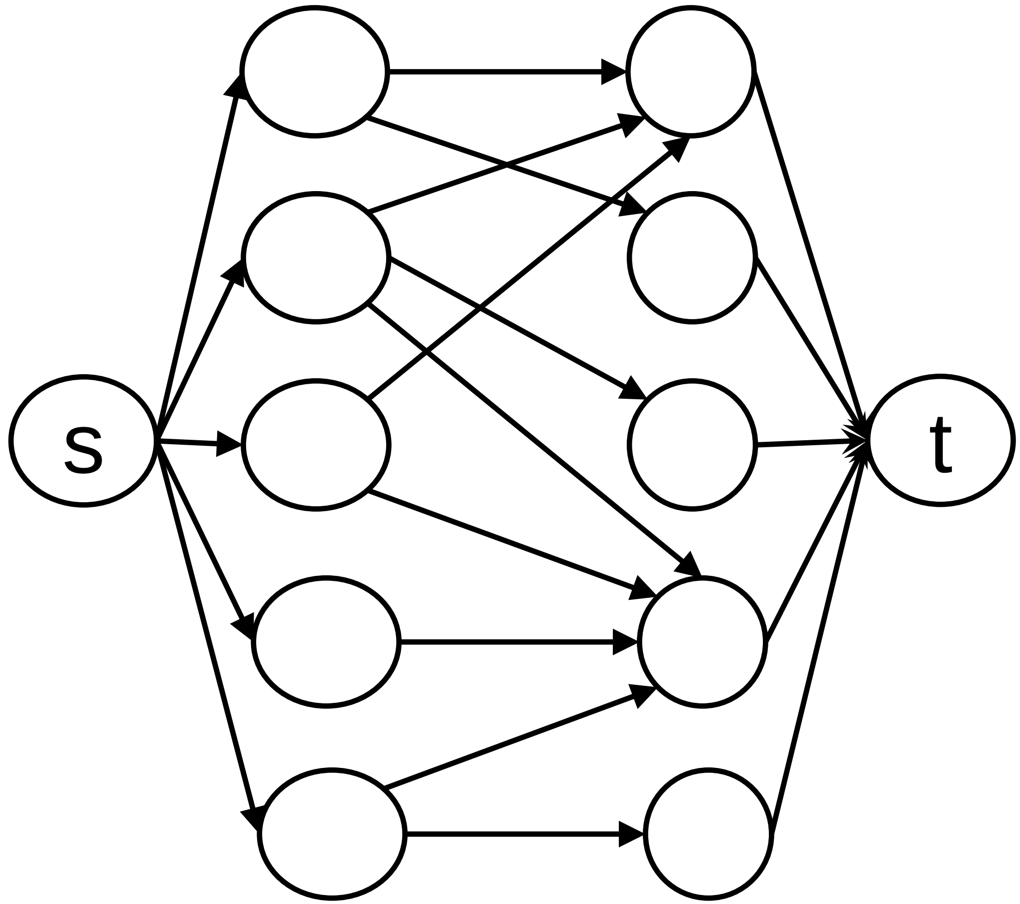
Residual Graph

Reductions!

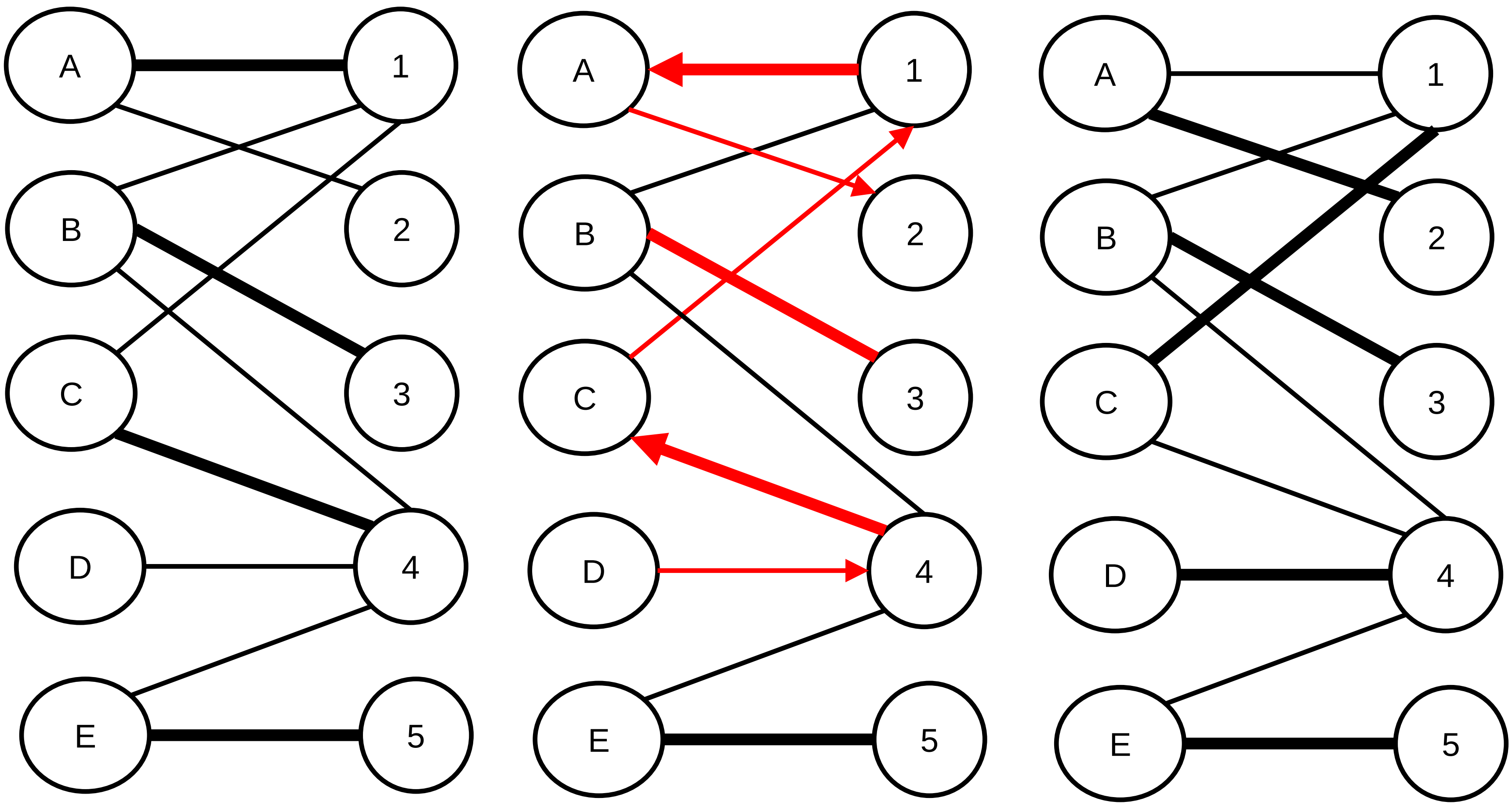
Turning one problem into another

Perfect Bipartite Matching to Max Flow





Proof of Hall's Theorem

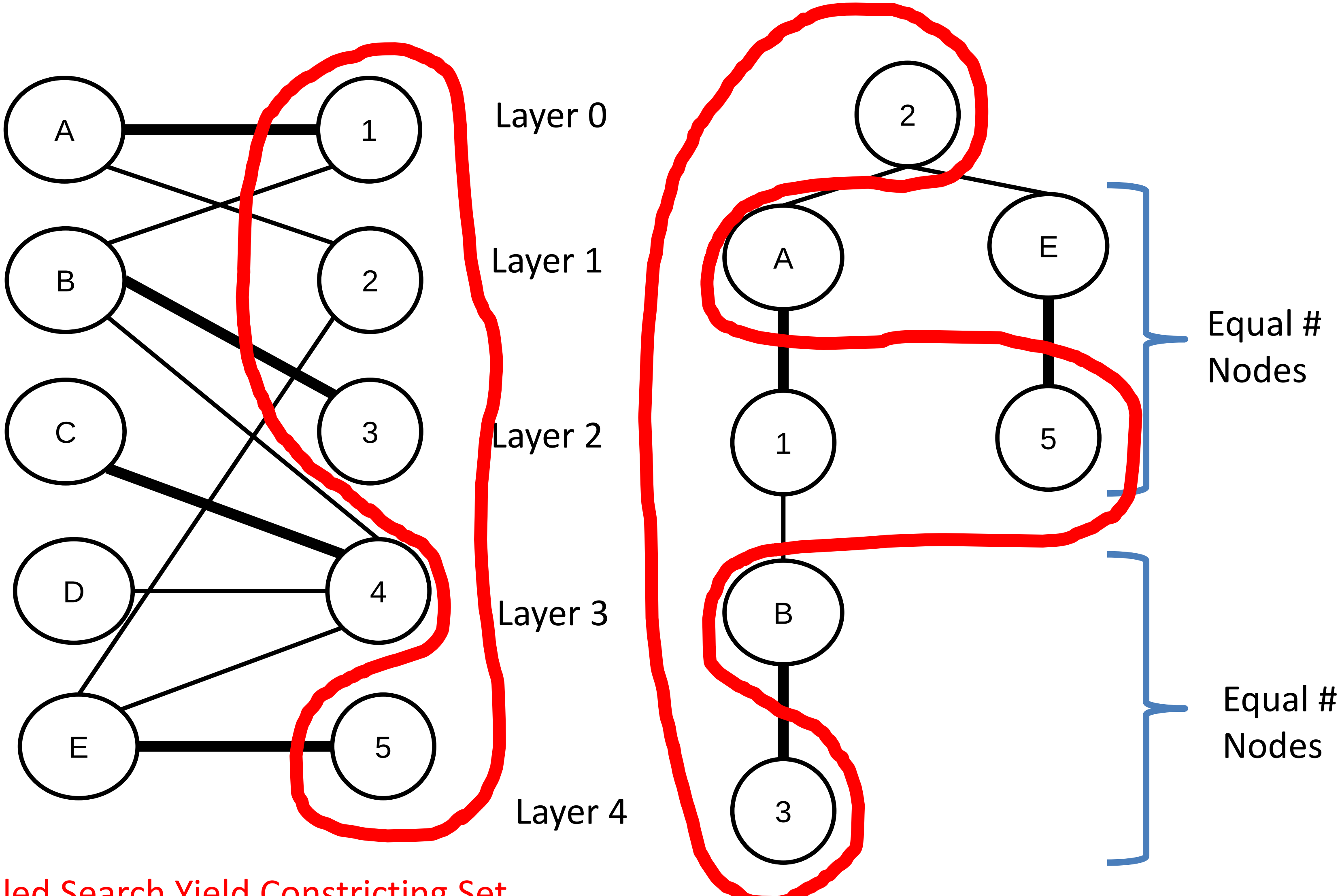


Current Matching

Augmenting Path

New Matching

Proof of Hall's Theorem



Failed Search Yield Constricting Set