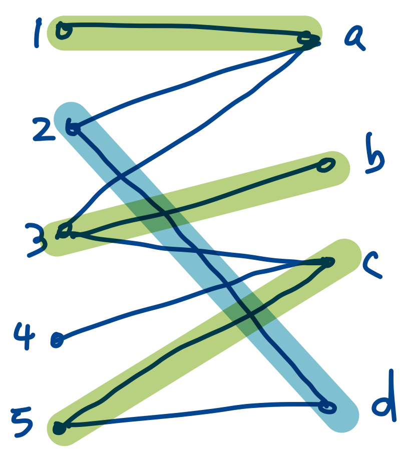
Network Flow [Chapter 7]

Applications of FF algorithm.

Bipartite Matching.

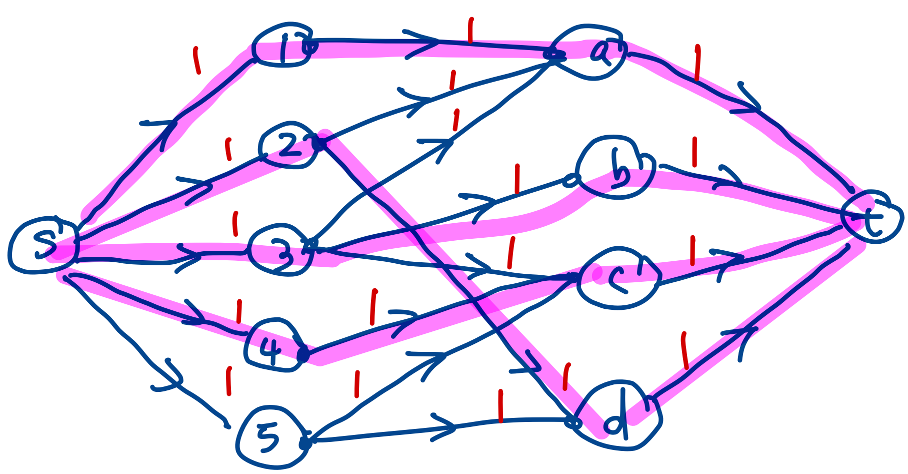
Input : Bipartite Graph $G = (V_1 \cup V_2, E)$



A matching $M \subseteq E$ is collection of edges (pairs) such that any vertex is contained in at most one edge of M.

{1a, 2a, 3b, 5c} is not a matching, a is matched to 2 vertices.

Question: Given a bipartite graph, find a maximum matching (with most edges).



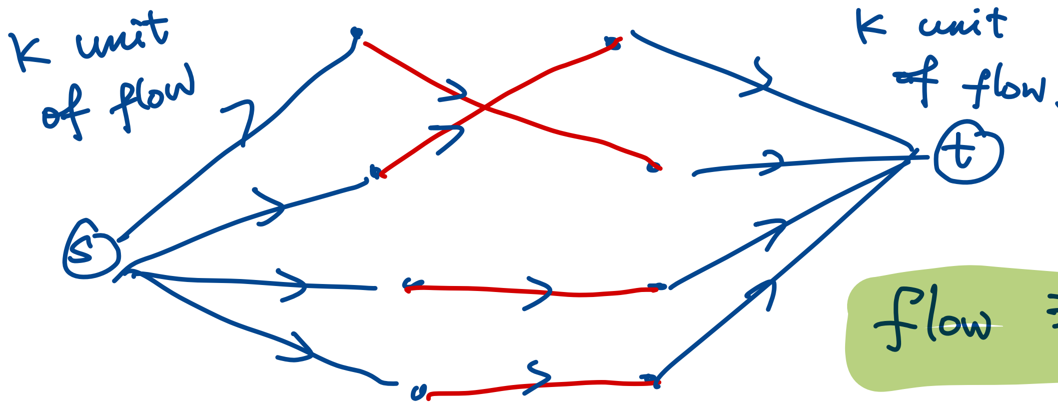
Algorithm:

1) Given BP Graph, construct a flow network.

2) Find the max-flow on this network using FF algorithm.

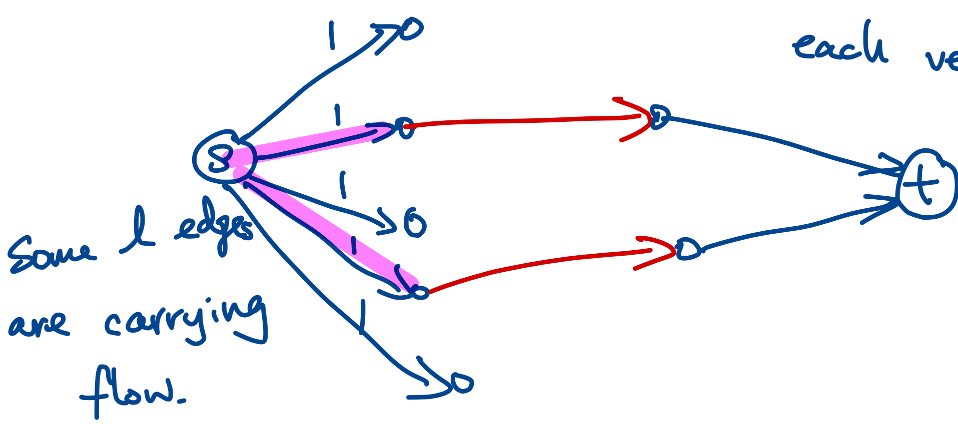
Thm: Size of Max-matching on BP Graph = Value of max-flow on flow network

Pf: Max-Matching size k $\Rightarrow$ Flow of value k.



$\text{flow} \geq |M|$.

Flow of size f $\Rightarrow$ Matching of size f.



each vertex getting a flow from s, must sent out exactly 1 unit of flow.

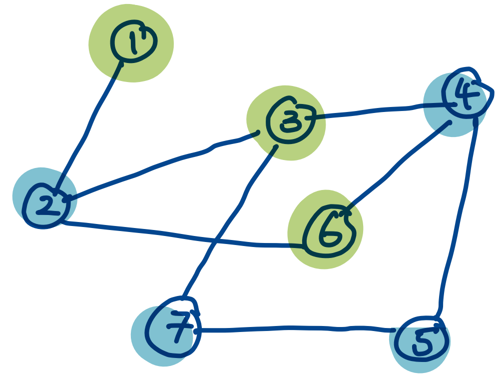
From the flow, we can trace the edges of the max-matching. (edges in BP that have flow on them).

$|M| \geq \text{flow}$.

$\Rightarrow$ Max Flow = Max-matching.

Reductions & NP-completeness [chaper 8]

Max-Independent Set



Given $G = (V, E)$

A set $S \subseteq V$ of vertices is called independent if no two vertices in S have an edge between them in G.

Example: $\{1, 3\}$, $\{1, 3, 6\}$, $\{1, 3, 6, 5\}$,

MIS (Problem): Given $G = (V, E)$, find the largest independent set in G.

Vertex Cover

A set $T \subseteq V$ of vertices is a vertex cover if every edge in G has at least one endpoint in T. $\{2, 4, 5, 7\}$

Minimum Vertex Cover (MVC) Problem

$\{2, 4, 7\}$

Given $G = (V, E)$, find the smallest vertex cover in G.

Thm: If S is independent, then $T = V - S$ is a vertex cover.
If T is a vertex cover, then $S = V - T$ is independent.

$\Rightarrow$ size of MaxIS = size of MinVC

Algo. for $M_{ax}IS \longleftrightarrow$ Algo. for MinVC.

$\underset{ax}{M}IS \leq_p \underset{in}{M}VC$ and $\underset{in}{M}VC \leq_p \underset{ax}{M}IS$.

problem $Y \leq_p$ problem X. [polynomial time reduction]

if X can be solve, then after solving X and doing some poly. work, we can solve Y.

$\Rightarrow$ "Y is poly. time reducible to X".