III. Matching

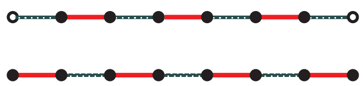
- $G = (V, E)$ undirected graph
- *Matching:* Subset of edges $M \subseteq E$, no two of which share an endpoint.
- *Maximum matching:* Matching of maximum cardinality
- *Perfect matching:* Every vertex in V is matched.

Augmenting paths

- Let M be a matching in $G = (V, E)$.
- A path $P = (v_0, v_1, \dots, v_t)$ in G is called *M-augmenting* if:
  - t is odd,
  - $v_1 v_2, v_3 v_4, v_{t-2} v_{t-1} \in M$,
  - $v_0, v_t \notin \bigcup M = \bigcup_{e \in M} e$.
- If P is an M-augmenting path and $E(P)$ the edge set of P, then

$$M' = M \triangle E(P) = (M \setminus E(P)) \cup (E(P) \setminus M)$$

is a matching in G of size $|M'| = |M| + 1$.



Berge's Theorem

Theorem (Berge 1957)
Let M be a matching in the graph $G = (V, E)$. Then either M is a maximum cardinality matching or there exists an M-augmenting path.

Generic Matching Algorithm

Initialization: $M \leftarrow \emptyset$
Iteration: If there exists an M-augmenting path P, replace $M \leftarrow M \triangle E(P)$.

$\rightsquigarrow$ how can one find an M-augmenting path?

- Difficult in general $\rightsquigarrow$ Edmonds' matching algorithm (Edmonds 1965)
- Easy for bipartite graphs

Bipartite graphs

A graph $G = (V, E)$ is *bipartite* if there exist $A, B \subseteq V$ with $A \cup B = V, A \cap B = \emptyset$ and each edge in E has one end in A and one end in B.

Proposition
A graph $G = (V, E)$ is bipartite if and only if each circuit of G has even length.

Bipartite matching

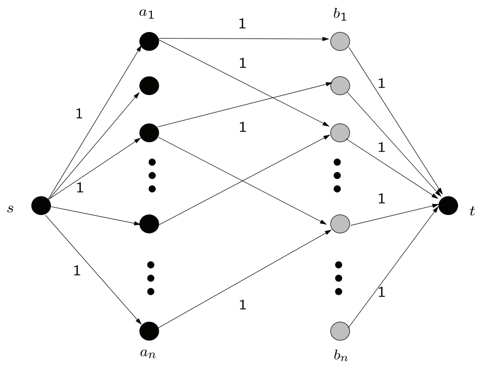
Matching augmenting algorithm for bipartite graphs

Input: Bipartite graph $G = (A \cup B, E)$ with matching M.
Output: Matching M' with $|M'| > |M|$ or proof that no such matching exists.
Description: Construct a directed graph D_M with the same node set as G.
For each edge $e = \{a, b\}$ in G with $a \in A, b \in B$:
if $e \in M$, there is the arc (b, a) in D_M.
if $e \notin M$, there is the arc (a, b) in D_M.
Let $A_M = A \setminus \bigcup M$ and $B_M = B \setminus \bigcup M$.
M-augmenting paths in G correspond to directed paths in D_M starting in A_M and ending in B_M.

Theorem
A maximum-cardinality matching in a bipartite graph $G = (V, E)$ can be found in time $O(|V||E|)$.

Bipartite matching as a maximum flow problem

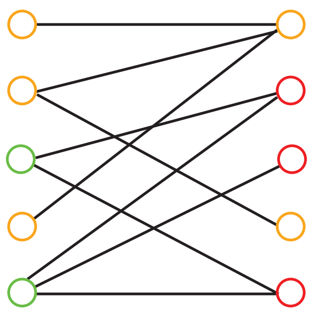
- Add a source s and edges (s, a) for $a \in A$, with capacity 1.
- Add a sink t and edges (b, t) for $b \in B$, with capacity 1.
- Direct edges in G from A to B, with capacity 1.



- Integral flows f correspond to matchings M, with $\mathrm{val}(f) = |M|$.
- Ford-Fulkerson takes time $O(nm)$, since $v^* \leq n$.
- Can be improved to $O(\sqrt{n}\, m)$ (Hopcroft-Karp 1973).

Marriage theorem

Theorem (Hall 1935)
A bipartite graph $G = (A \cup B, E)$, with $|A| = |B| = n$, has a perfect matching if and only if for all $B' \subseteq B$, $|B'| \leq |N(B')|$, where $N(B')$ is the set of all neighbors of nodes in B'.

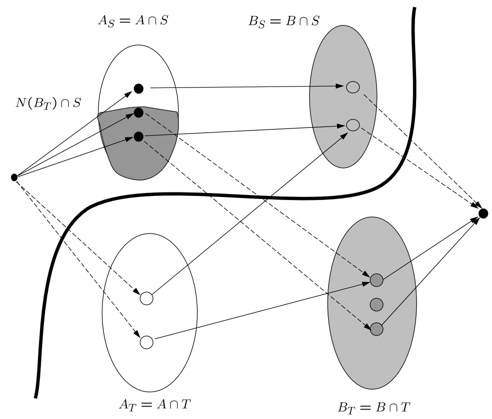


Proof

- Let (S, T) be an (s, t)-cut in the corresponding network.
- Let $A_S = A \cap S, A_T = A \cap T, B_S = B \cap S, B_T = B \cap T$.

$$\begin{aligned}
\mathrm{cap}(S,T) &= \sum_{e \in E \cap S \times T} \mathrm{cap}(e) \\
&= |A_T| + |B_S| + |N(B_T) \cap A_S| \\
&\geq |N(B_T) \cap A_T| + |N(B_T) \cap A_S| + |B_S| \\
&= |N(B_T)| + |B_S| \\
&\geq |B_T| + |B_S| = |B| = n
\end{aligned}$$

- By the max-flow min-cut theorem, the maximum flow is at least n.



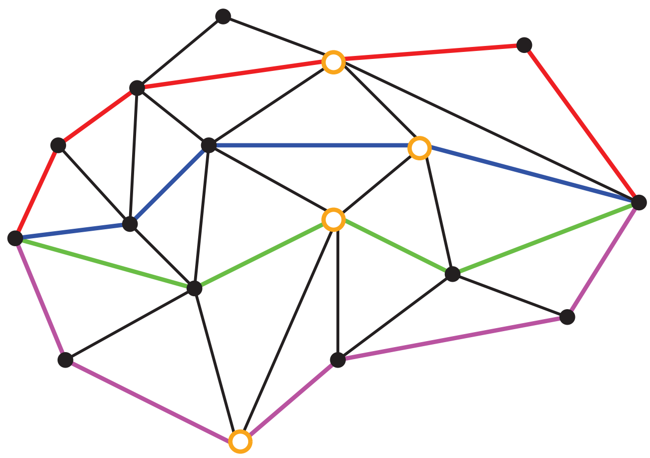
König's theorem

- $G = (V, E)$ undirected graph
- $C \subseteq V$ is a *vertex covering* if every edge of G has at least one end in C.
- **Lemma:** For any matching M and any vertex covering C, we have $|M| \leq |C|$.
- **Theorem** (König 1931) For a bipartite graph G,

$$\max\{|M| : M \text{ a matching }\} = \min\{|C| : C \text{ a vertex covering }\}.$$

Network connectivity: Menger's theorems

- $G = (V, E)$ directed graph, $s, t \in V, s \neq t$ non-adjacent.
- **Theorem** (Menger 1927) The maximum number of *arc-disjoint* paths from s to t equals the minimum number of arcs whose removal disconnects all paths from s to t.
- **Theorem** (Menger 1927) The maximum number of *node-disjoint* paths from s to t equals the minimum number of nodes (different from s and t) whose removal disconnects all paths from s to t.



References

- A. Schrijver: A Course in Combinatorial Optimization, CWI Amsterdam, 2010, `http://homepages.cwi.nl/~lex/files/dict.pdf`
- K. Mehlhorn: Data Structures and Efficient Algorithms, Vol. 2: Graph Algorithms and NP-Completeness, Springer, 1986, `http://www.mpi-sb.mpg.de/~mehlhorn/DatAlgbooks.html`
- S. Krumke and H. Noltemeier: Graphentheoretische Konzepte und Algorithmen. Teubner, 2005
- R. K. Ahuja, T. L. Magnanti and J. L. Orlin: Network flows. Prentice Hall, 1993