Matching

Given a graph $G = (V, E)$ a set of edges M is a **matching** in G if

1. $M \subseteq E$;

2. no two edges of M share the same node;

A **maximum** matching in G is a matching with maximal cardinality.

A graph has a **perfect** matching if it has a matching of size $|V|/2$ (i.e. every vertex is covered by the matching).

Building a Matching

Given a matching M in G, a simple path

$$P = v_1, v_2, ..., v_k$$

is an **alternating** path with respect to M if the odd edges in P (the first, third,...) are not in M, and the even edges (second. fourth,...) are in M.

An alternating path P is an **augmenting** path with respect to M if both v_1 and v_k are not covered by M.

An augmenting path has an odd number of edges, with one more edge that is not covered by the matching.

Removing the even edges of P from M and adding the odd edges of P to M increases the size of the matching by one (covering two more vertices).

Matching Characterization

Theorem 1. *A matching M in a graph G is maximum iff there is no augmenting path in G with respect to M.*

Proof.

Augmenting path $\Rightarrow$ not maximum: If there is an augmenting path then M is not maximum since we can use the path to get a larger matching.

Not maximum $\Rightarrow$ augmenting path: Assume that M is not maximum, we need to show that there is an augmenting path with respect to M.

Let M' be a matching in G such that $|M'| > |M|$. Consider the graph $H = (V, M \cup M')$.

The degrees in this graph are $0, 1$ or 2.

The connected components in the graph are either paths or cycles.

There are no cycles with odd number of edges.

There must be at least one connected component (path) with one more edge of M' than M. This component is an alternating path.

$\square$

Matching Algorithm

1. $M \leftarrow \emptyset$;

2. While there are augmenting paths w.r.t. M

   2.1 Extend the matching using an augmenting path;

   How to search for an augmenting path?

Matching in Bi-Partite Graphs

A graph $G = (V, E)$ is **bi-partite** if the set of vertices V can be partitioned to two sets A and B, ($A \cup B = V$ and $A \cap B = \emptyset$), such that every edge in E has one adjacent vertex in A and one in B.

In a bi-partite graph the search for an augmenting path for an unmatched node $x \in A$ can be done by a breadth-first search.

Theorem 2. *The above algorithm finds maximum matching in $O(V \cdot E)$ time.*