Graph Algorithms: Undirected Graphs

Unit 4: Lecture 06

Graph Matching

Graph **matching** refers to the **subset of edges** without common vertices.

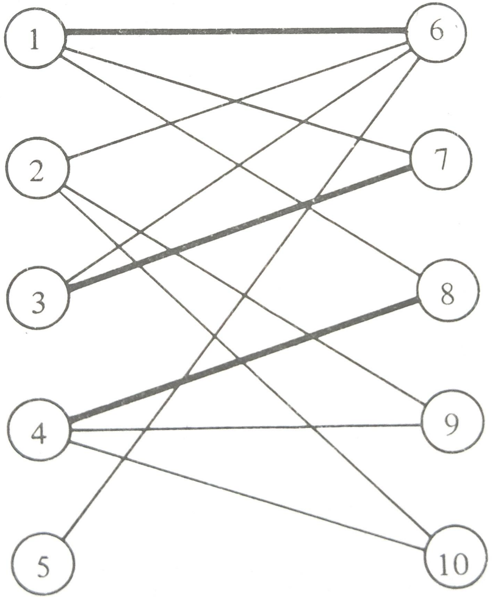
Mathematically, let G=(V,E) be an undirected graph. A **matching M is a subset** of edges E (M⊆E) such that each vertex v∈V is incident to **at most one edge** from M.

The task of **selecting a maximum subset** of such edges is called the **maximal matching** problem.

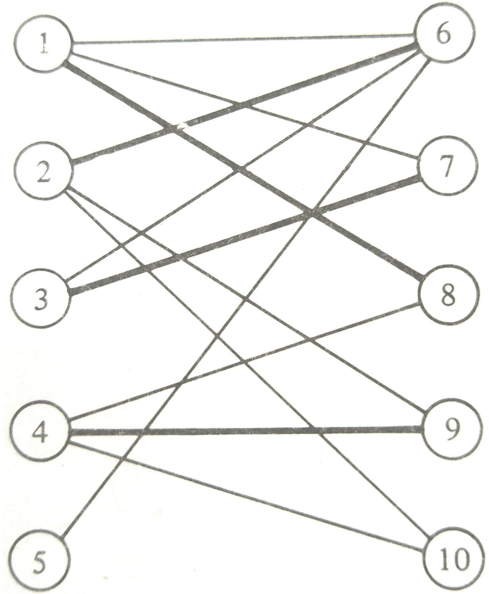
A **complete matching** is a matching in which every vertex is an end point of some edge in the matching.

Graph Matching (contd.)

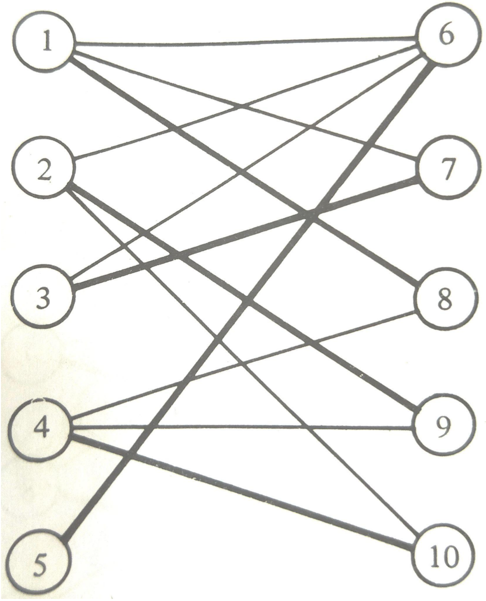
Example: V={1,2,...,10} of a **bipartite graph**



A Matching



Another Matching



A Maximal Matching

Graph Matching (contd.)

How to find a maximal matching?

The elementary approach to find a maximal matching is to generate all matchings and pick with the largest number of edges! (is it feasible?)

Other approaches to find a maximal matching include:

- Augmenting path finding
- Using maximum flow (not discussed here)

Augmenting Path

In graph matching, a vertex is called **matched** if it is the endpoint of an edge in M.

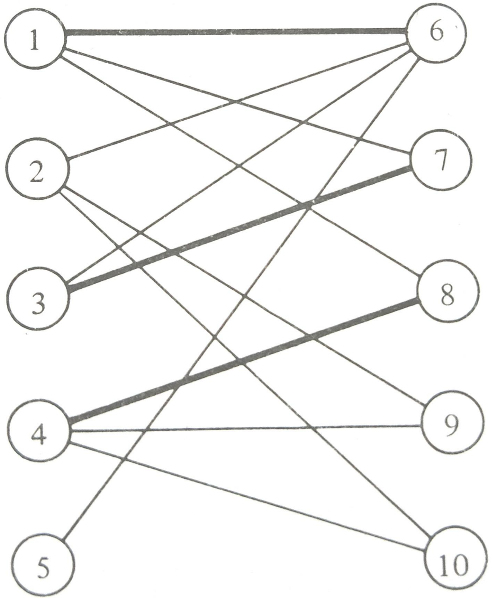
A path connecting two unmatched vertices in which alternate edges in the path are in M is called an **augmenting path** relative to M.

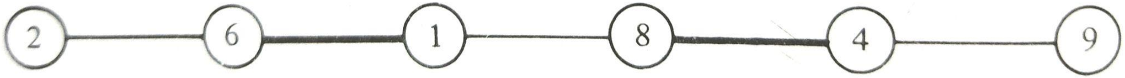
An augmenting path must be of **odd length** which must begin and end with edges not in M.

Augmenting Path (contd.)

Example:

An augmenting path 2,6,1,8,4,9 corresponding to the matching (1,6),(3,7),(4,8)





Augmenting Path (contd.)

An augmenting path P provides a **bigger matching** by removing the edges of P from M and then adding to M the edges of P which were initially not in M (equivalent to M **XOR** P)

M is a **maximal matching iff** there is no augmenting path relative to M!

Augmenting Path (contd.)

Procedure:

a. Start with M=∅
b. Find augmenting path P relative to M and replace M by M **XOR** P
c. Repeat **b** until no further augmenting paths exits
d. M is a maximal matching

Analysis:

O(ne) time is required- constructing the **augmenting path graphs** for a given matching takes O(e) time: at most n/2 augmenting paths are constructed to find a maximal matching.

Exercise

1. Find the maximal matching for a given graph using augmenting path.
2. How many maximal matchings can be obtained for a given graph?

References

Data Structures and Algorithms: Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, 10th Impression, Pearson Education, New Delhi

https://www.cs.dartmouth.edu/~ac/Teach/CS105-Winter05/Notes/kavathekar-scribe.pdf