

Matchings

A **matching** is a set of (non-loop) edges with no shared endpoints. The vertices incident to an edge of a matching M are **saturated** by M , the others are **unsaturated**. A **perfect matching** of G is matching which saturates all the vertices.

Examples. $K_{n,m}$, K_n , Petersen graph, Q_k ; graphs without perfect matching

A **maximal matching** cannot be enlarged by adding another edge.

A **maximum matching** of G is one of maximum **size**.

Example. Maximum \neq Maximal

Characterization of **maximum** matchings_____

Let M be a matching. A path that alternates between edges in M and edges not in M is called an **M -alternating path**.

An M -alternating path whose endpoints are unsaturated by M is called an **M -augmenting path**.

Theorem(Berge, 1957) A matching M is a maximum matching of graph G **iff** G has no M -augmenting path.

Proof. (\Rightarrow) Easy.

(\Leftarrow) Suppose there is no M -augmenting path and let M^* be a matching of maximum size.

What is then $M \Delta M^*$???

Lemma Let M_1 and M_2 be matchings of G . Then each connected component of $M_1 \Delta M_2$ is a path or an even cycle.

For two sets A and B , the symmetric difference is $A \Delta B = (A \setminus B) \cup (B \setminus A)$.

Hall's Condition and consequences_____

Theorem (Marriage Theorem; Hall, 1935) Let G be a bipartite (multi)graph with partite sets X and Y . Then there is a matching in G saturating X iff $|N(S)| \geq |S|$ for every $S \subseteq X$.

Proof. (\Rightarrow) Easy.

(\Leftarrow) Not so easy. Find an M -augmenting path for any matching M which does not saturate X .

(Let U be the M -unsaturated vertices in X . Define

$$T := \{y \in Y : \exists M\text{-alternating } U, y\text{-path}\},$$

$$S := \{x \in X : \exists M\text{-alternating } U, x\text{-path}\}.$$

Unless there is an M -augmenting path, SUU violates Hall's condition.)

Corollary. (Frobenius (1917)) For $k > 0$, every k -regular bipartite (multi)graph has a perfect matching.

Graph parameters — Definitions and simple properties

The size of the **largest independent set** in G is denoted by $\alpha(G)$.

The size of the **largest matching** (*independent set of edges*) in G is denoted by $\alpha'(G)$.

A **vertex cover** of G is a set $Q \subseteq V(G)$ that contains at least one endpoint of every edge. (The vertices in Q *cover* $E(G)$).

The size of the **smallest vertex cover** in G is denoted by $\beta(G)$.

Claim. $\beta(G) \geq \alpha'(G)$.

An **edge cover** of G is a set L of edges such that every vertex of G is incident to some edge in L .

The size of the **smallest edge cover** in G is denoted by $\beta'(G)$.

Claim. $\beta'(G) \geq \alpha(G)$.

Certificates

Suppose we knew that in some graph G with 1121 edges on 200 vertices, a particular set of 87 edges is (one of) the largest matching one could find. How could we convince somebody about this?

Once the particular 87 edges are shown, it is easy to check that they are a matching, indeed.

But why isn't there a matching of size 88? Verifying that none of the $\binom{1121}{88}$ edgesets of size 88 forms a matching could take some time...

If we happen to be so lucky, that we are able to exhibit a vertex cover of size 87, we are saved. It is then reasonable to check that all 1121 edges are covered by the particular set of 87 vertices.

Exhibiting a vertex cover of a certain size **proves** that no larger matching can be found.

Min-max theorems for bipartite graphs_____

Theorem. (König (1931), Egerváry (1931)) If G is bipartite then $\beta(G) = \alpha'(G)$.

Proof. For any minimum vertex cover Q , apply Hall's Condition to match $Q \cap X$ into $Y \setminus Q$ and $Q \cap Y$ into $X \setminus Q$.

Lemma. Let G be any graph. $S \subseteq V(G)$ is an independent set iff \overline{S} is a vertex cover.

Hence $\alpha(G) + \beta(G) = n(G)$.

Proof. Easy.

Theorem. (Gallai, 1959) Let G be any graph without isolated vertices. Then $\alpha'(G) + \beta'(G) = n(G)$.

Corollary. (König, 1916) Let G be a bipartite graph with no isolated vertices. Then $\alpha(G) = \beta'(G)$.

Proof. Put together the previous three statements.