Matchings

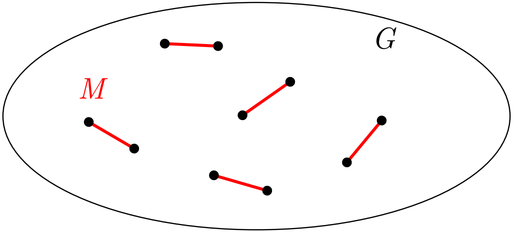
Graph theory

Lecture 11

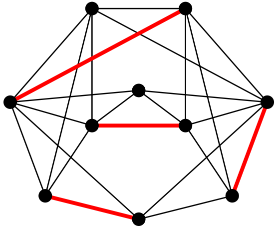
SZTE Bolyai Institute

Szeged

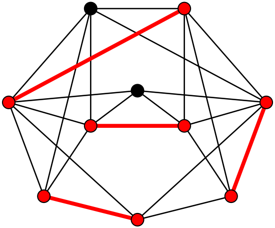
A set of edges $M \subseteq E(G)$ in a multigraph G is called a matching, if no two edges of M have a common endpoint, and there are no loops in M.



A set of edges $M \subseteq E(G)$ in a multigraph G is called a matching, if no two edges of M have a common endpoint, and there are no loops in M.

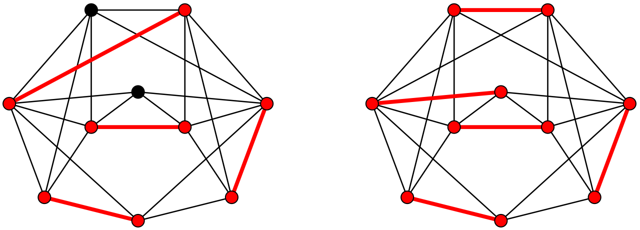


A set of edges $M \subseteq E(G)$ in a multigraph G is called a matching, if no two edges of M have a common endpoint, and there are no loops in M.



For a matching M, the end vertices of the edges in M are called the matched or covered vertices. (Their number is $2|M|$.)

A set of edges $M \subseteq E(G)$ in a multigraph G is called a matching, if no two edges of M have a common endpoint, and there are no loops in M.



For a matching M, the end vertices of the edges in M are called the matched or covered vertices. (Their number is $2|M|$.)

A matching in G is perfect, if it covers all vertices of G. (The matching in the left figure is not perfect, the matching in the right figure is perfect.)

If G contains a perfect matching then G must have an even number of vertices.

Def. $\nu(G) := \max\{|M| : M \text{ is a matching in } G\}$,

where $|M|$ is the number of edges in M. So $\nu(G)$ is the size of a largest matching in G.

Def. $$\nu(G) := \max\{|M| : M \text{ is a matching in } G\},$$

where $|M|$ is the number of edges in M. So $\nu(G)$ is the size of a largest matching in G.

Remark.
$$\nu(G) \leq \frac{|V(G)|}{2},$$

because any matching M covers $2|M|$ distinct vertices for any matching M, which means that $2|M| \leq |V(G)|$.

Moreover, $\nu(G) = \frac{|V(G)|}{2}$ holds if and only if G contains a perfect matching.

Def. $$\nu(G) := \max\{|M| : M \text{ is a matching in } G\},$$

where $|M|$ is the number of edges in M. So $\nu(G)$ is the size of a largest matching in G.

Remark.

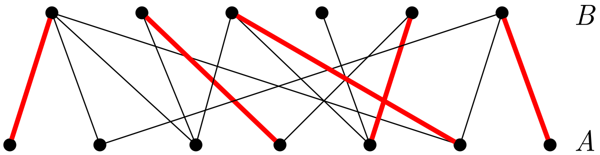
$$\nu(G) \leq \frac{|V(G)|}{2},$$

because any matching M covers $2|M|$ distinct vertices for any matching M, which means that $2|M| \leq |V(G)|$.

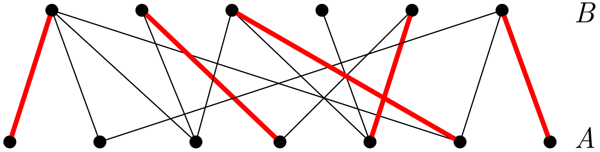
Moreover, $\nu(G) = \frac{|V(G)|}{2}$ holds if and only if G contains a perfect matching.

Remark. We will see that the parameter $\nu(G)$ can be computed efficiently (in polynomial time) by computer. In honor of the Hungarian mathematicians Dénes Kőnig and Jenő Egerváry, the algorithm for finding a maximum size matching in bipartite graphs is called “Hungarian method”.

In bipartite graphs, each edge of a matching M covers one vertex from the lower class and one vertex from the upper class:



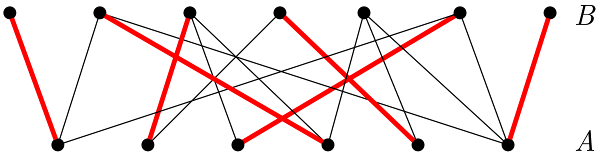
In bipartite graphs, each edge of a matching M covers one vertex from the lower class and one vertex from the upper class:



This implies that every matching contains at most as many edges as the number of vertices in A. So in case of bipartite graphs, we have $\nu(G) \leq |A|$.

As the role of partite classes A and B are interchangeable, here and henceforth A can be replaced to B.

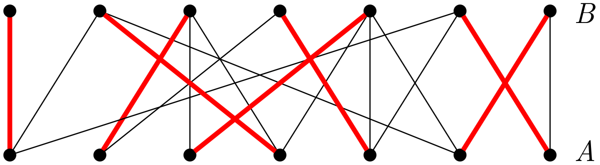
In bipartite graphs, each edge of a matching M covers one vertex from the lower class and one vertex from the upper class:



This implies that every matching contains at most as many edges as the number of vertices in A. So in case of bipartite graphs, we have $\nu(G) \leq |A|$.

The equality $\nu(G) = |A|$ is equivalent to the fact that G has a matching that covers all vertices of A.

In bipartite graphs, each edge of a matching M covers one vertex from the lower class and one vertex from the upper class:



This implies that every matching contains at most as many edges as the number of vertices in A. So in case of bipartite graphs, we have $\nu(G) \leq |A|$.

The equality $\nu(G) = |A|$ is equivalent to the fact that G has a matching that covers all vertices of A.

Finally consider a perfect matching in a bipartite graph (see the new figure). As every matching covers the same number of points in A as in B, a necessary (but not sufficient) condition for the existence of a perfect matching is $|A| = |B|$.

Definition. Let G be a bipartite graph with classes A, B.

• The neighborhood $N(X)$ of a set $X \subseteq A$ is the collection (union) of the neighbors of vertices in X, that is,

$$N(X) := \{v \in B : v \text{ is adjacent to some vertex in } X\}.$$

Definition. Let G be a bipartite graph with classes A, B.

- The neighborhood $N(X)$ of a set $X \subseteq A$ is the collection (union) of the neighbors of vertices in X, that is,

$$N(X) := \{v \in B : v \text{ is adjacent to some vertex in } X\}.$$

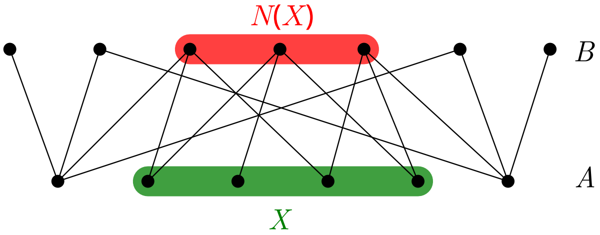
- The set $X \subseteq A$ is a Kőnig set, if $|N(X)| < |X|$.

Definition. Let G be a bipartite graph with classes A, B.

- The neighborhood $N(X)$ of a set $X \subseteq A$ is the collection (union) of the neighbors of vertices in X, that is,

$$N(X) := \{v \in B : v \text{ is adjacent to some vertex in } X\}.$$

- The set $X \subseteq A$ is a Kőnig set, if $|N(X)| < |X|$.

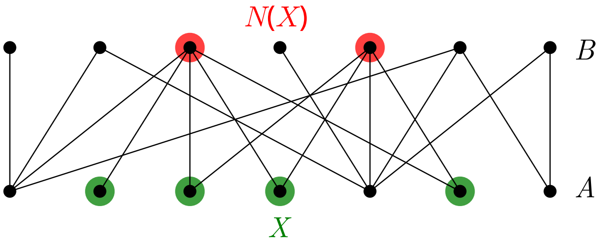


Definition. Let G be a bipartite graph with classes A, B.

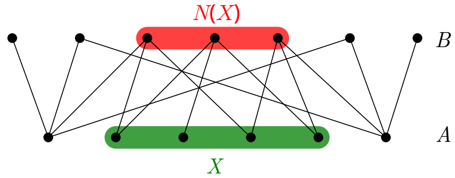
- The neighborhood $N(X)$ of a set $X \subseteq A$ is the collection (union) of the neighbors of vertices in X, that is,

$$N(X) := \{v \in B : v \text{ is adjacent to some vertex in } X\}.$$

- The set $X \subseteq A$ is a Kőnig set, if $|N(X)| < |X|$.

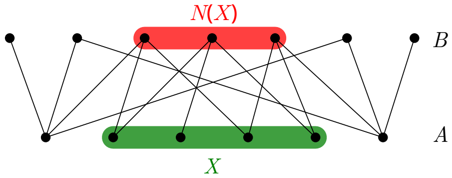


Recall. The set $X \subseteq A$ is a Kőnig set, if $|N(X)| < |X|$.



Claim. If a Kőnig set exists in G, then G does not have a matching that covers A (and so G has no perfect matching either). This is because for any matching M, there are at least $|X| - |N(X)| > 0$ unmatched vertices in A.

Recall. The set $X \subseteq A$ is a Kőnig set, if $|N(X)| < |X|$.



Claim. If a Kőnig set exists in G, then G does not have a matching that covers A (and so G has no perfect matching either). This is because for any matching M, there are at least $|X| - |N(X)| > 0$ unmatched vertices in A.

Proof. It is enough to prove the second statement. By the definition of neighborhood, all edges starting from a Kőnig set X go to $N(X)$, so every vertex of X can be matched to a vertex of $N(X)$ only. As distinct vertices of X are matched to distinct vertices of $N(X)$ in any matching M, this means that at most $|N(X)|$ vertices of X can be matched, and so the number of unmatched vertices in X is at least $|X| - |N(X)|$. □

The next deep theorem shows that the fact "there exists no matching that covers A" can be always justified by finding a Kőnig set.

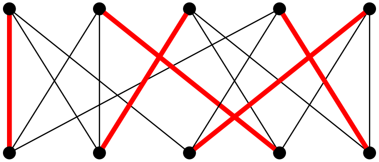
Marriage theorem. Let G be a bipartite graph with classes A, B.

a) The bipartite graph G contains a matching covering A if and only if there is no Kőnig set in G, i.e. if $|N(X)| \geq |X|$ for all $X \subseteq A$.

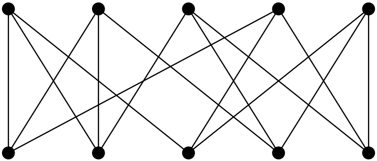
b) The bipartite graph G contains a perfect matching if and only if $|A| = |B|$ and there is no Kőnig set in G (i.e. $|N(X)| \geq |X|$ for all $X \subseteq A$).

Remark. The second statement is the corollary of the first one. We have already seen that $|A| = |B|$ is a sufficient condition for the existance of perfect matching, and in case of $|A| = |B|$, the perfect matchings are precisely those matchings that cover A.

Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.



Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.

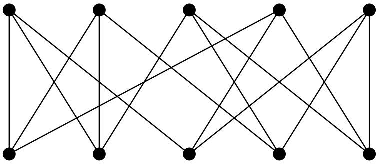


Proof. Let G be a d-regular bipartite graph with classes A, B, where $d \geq 1$. In order to prove that G contains a perfect matching, it is enough to check that the conditions of the marriage theorem hold:

1. The classes A and B contain the same number of vertices.

2. $|N(X)| \geq |X|$, for all sets $X \subseteq A$.

Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.



Proof. Let G be a d-regular bipartite graph with classes A, B, where $d \geq 1$. In order to prove that G contains a perfect matching, it is enough to check that the conditions of the marriage theorem hold:

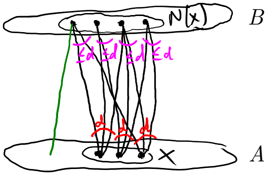
1. The classes A and B contain the same number of vertices.

This follows from the fact that the sum of degrees of vertices of A is equal to the sum of degrees of vertices of B in any bipartite graph, so in the d-regular case we have that

$$|A| \cdot d = |B| \cdot d,$$

and a division by d $(d \neq 0)$ yields that $|A| = |B|$. ✓

Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.

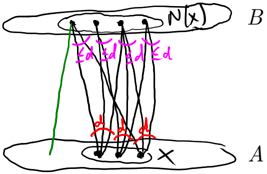


2. $|N(X)| \geq |X|$, for all sets $X \subseteq A$.

Let $X \subseteq A$ be arbitrary. Let $e(X, N(X))$ denote the number of edges between X and $N(X)$ in G. Then

$$|X| \cdot d = e(X, N(X)) \leq |N(X)| \cdot d.$$

Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.



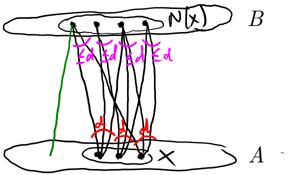
2. $|N(X)| \geq |X|$, for all sets $X \subseteq A$.

Let $X \subseteq A$ be arbitrary. Let $e(X, N(X))$ denote the number of edges between X and $N(X)$ in G. Then

$$|X| \cdot d = e(X, N(X)) \leq |N(X)| \cdot d.$$

The reason for the first equality: For every vertex v of X, there are exactly d edges going from v to $N(X)$, because there are exactly d edges incident to v by the d-regularity, and all of these edges go to $N(X)$ by the definition of the neighborhood. This gives altogether $|X| \cdot d$ edges between X and $N(X)$.

Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.



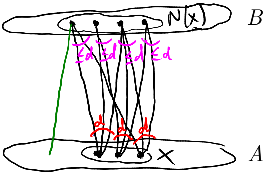
2. $|N(X)| \geq |X|$, for all sets $X \subseteq A$.

Let $X \subseteq A$ be arbitrary. Let $e(X, N(X))$ denote the number of edges between X and $N(X)$ in G. Then

$$|X| \cdot d = e(X, N(X)) \leq |N(X)| \cdot d.$$

The reason for the left inequality: For every vertex w of $N(X)$, there are at most d edges going from w to X, because there are exactly d edges incident to w by the d-regularity, and some of these edges go to X (cf. the green edge in the fig.). This gives altogether at most $|N(X)| \cdot d$ edges between X and $N(X)$.

Claim. Every d-regular bipartite graph contains a perfect matching, if $d \geq 1$.



2. $|N(X)| \geq |X|$, for all sets $X \subseteq A$.

Let $X \subseteq A$ be arbitrary. Let $e(X, N(X))$ denote the number of edges between X and $N(X)$ in G. Then

$$|X| \cdot d = e(X, N(X)) \leq |N(X)| \cdot d.$$

Combining the two sides, after a division by d $(d > 0)$ we obtain that

$$|N(X)| \geq |X|. \checkmark$$

As this reasoning works for any set $X \subseteq A$, the proof is complete. □

M is a matching in the (not necessarily bipartite) graph G. We say that a path

$$P : (v_0, e_1, v_1, e_2, v_2, e_3, v_3, \ldots, v_{2k}, e_{2k+1}, v_{2k+1})$$

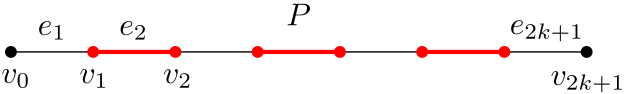
in G is an augmenting path w.r.t. M, if P satisfies the following conditions:

(i) the start vertex v_0 is unmatched,

(ii) $e_1, e_3, e_5, \ldots, e_{2k+1} \notin M$,

(iii) $e_2, e_4, e_6, \ldots, e_{2k} \in M$,

(iv) the end vertex v_{2k+1} is unmatched.

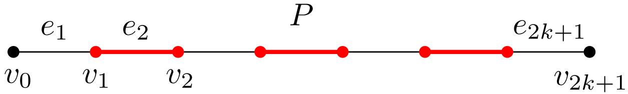


M is a matching in the (not necessarily bipartite) graph G. We say that a path

$$P : (v_0, e_1, v_1, e_2, v_2, e_3, v_3, \dots , v_{2k}, e_{2k+1}, v_{2k+1})$$

in G is an augmenting path w.r.t. M, if P satisfies the following conditions:

(i) the start vertex v_0 is unmatched,
(ii) $e_1, e_3, e_5, \dots, e_{2k+1} \notin M$,
(iii) $e_2, e_4, e_6, \dots, e_{2k} \in M$,
(iv) the end vertex v_{2k+1} is unmatched.



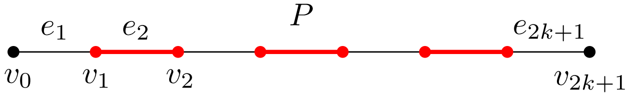
Remark. The conditions (i) and (iv) imply that the first and last edges of P are not in M, so the length of an augmenting path is always odd, and it contains one more "black" edges than "red" edges.

M is a matching in the (not necessarily bipartite) graph G. We say that a path

$$P : (v_0, e_1, v_1, e_2, v_2, e_3, v_3, \ldots, v_{2k}, e_{2k+1}, v_{2k+1})$$

in G is an augmenting path w.r.t. M, if P satisfies the following conditions:

(i) the start vertex v_0 is unmatched,
(ii) $e_1, e_3, e_5, \ldots, e_{2k+1} \notin M$,
(iii) $e_2, e_4, e_6, \ldots, e_{2k} \in M$,
(iv) the end vertex v_{2k+1} is unmatched.



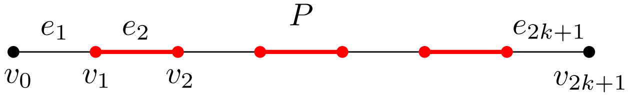
We say that P is a partial augmenting path w.r.t. M if it satisfies the conditions (i)-(iii). (The length of a partial augmenting path is allowed to be even.) The attribute 'partial' in the name reflects that a partial augmenting path might be extendable to an augmenting path.

M is a matching in the (not necessarily bipartite) graph G. We say that a path

$$P : (v_0, e_1, v_1, e_2, v_2, e_3, v_3, \ldots , v_{2k}, e_{2k+1}, v_{2k+1})$$

in G is an augmenting path w.r.t. M, if P satisfies the following conditions:

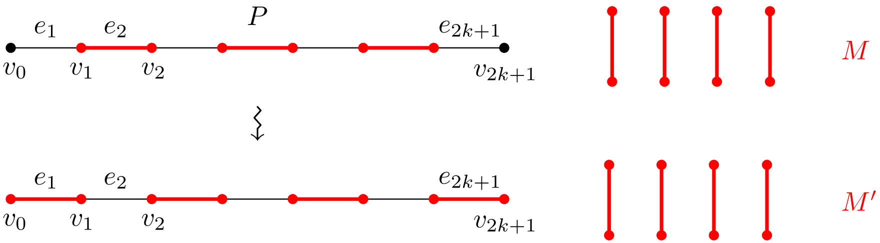
(i) the start vertex v_0 is unmatched,
(ii) $e_1, e_3, e_5, \ldots , e_{2k+1} \notin M$,
(iii) $e_2, e_4, e_6, \ldots , e_{2k} \in M$,
(iv) the end vertex v_{2k+1} is unmatched.



We say that P is a partial augmenting path w.r.t. M if it satisfies the conditions (i)-(iii). (The length of a partial augmenting path is allowed to be even.) The attribute 'partial' in the name reflects that a partial augmenting path might be extendable to an augmenting path.

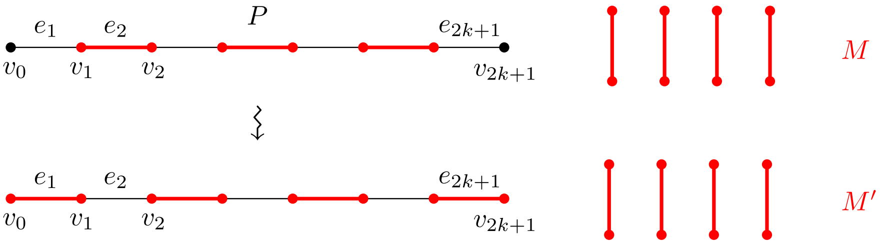
Let M be a matching in the graph G.

Observation. If G contains an augmening path with w.r.t M, then M can be "augmented along P" to obtain a matching M' that has one more edges than M has: Just interchange the "black" and "red" edges inside P.



Let M be a matching in the graph G.

Observation. If G contains an augmening path with w.r.t M, then M can be "augmented along P" to obtain a matching M' that has one more edges than M has: Just interchange the "black" and "red" edges inside P.



Theorem. The matching M is not of maximum size in G if and only there exists an augmenting path w.r.t. M.

Proof. The direction $\Leftarrow$ is just the observation above. The direction $\Rightarrow$ follows from the analysis of the augmenting path finding algorithms discussed later. $\square$

Theorem. The matching M is not of maximum size in G if and only there exists an augmenting path w.r.t. M.

Using this, it is easy to find a maximum size matching in an input graph G in theory:

(1) Start from an arbitrary matching M of G. For example, M can be a trivial matching that contains only one (non-loop) edge of G.

(2) Find an augmenting path w.r.t. M. If an augmenting path is found, then augment M using it, and repeat step (2) for the obtained matching Otherwise, i.e. if there is no augmenting path, then the actual M has maximum size, terminate.

We will discuss polynomial time algorithms for finding an augmenting path (if it exists) in the 'Matching algorithms' presentation. This means that the above simple method can be implemented in polynomial time to find a maximum size matching in G, and so determine the parameter $\nu(G)$.