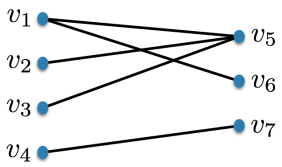
Matchings in Graphs

1 Maximum Matchings

Definition (Matching – maximum, maximal, perfect)**.** A *matching* in a graph $G = (V, E)$ is a subset of the edges that do not share an endpoint. A *maximum matching* in G is a matching with the maximum number of edges among all possible matchings. A *maximal matching* is a matching with the property that if we add any other edge to the matching, it is no longer a matching.[1] A *perfect matching* is a matching that covers all the vertices of the graph.

Example (Examples of matchings)**.** Consider the following graph.



Note that the empty set and a set with only one edge is always a matching. The set $M = \{\{v_1, v_5\}, \{v_4, v_7\}\}$ is a maximal matching with 2 edges, since we if we tried to add another edge to this set, it would no longer be a matching. On the other hand, this maximal matching is not a maximum matching because there is another matching with 3 edges: $M' = \{\{v_1, v_6\}, \{v_3, v_5\}, \{v_4, v_7\}\}$. This graph does not have a perfect matching. One easy way to see this is that it has an odd number of vertices, and any graph with an odd number of vertices cannot have a perfect matching.

[1] Note that a maximal matching is not necessarily a maximum matching, but a maximum matching is always a maximal matching.

Note (Size of a matching)**.** The size of a matching M refers to the number of edges in the matching, and is denoted by $|M|$. Note that this coincides with the size of the set that M represents.

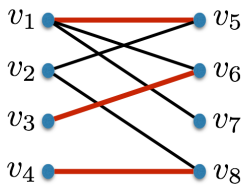
Definition (Maximum matching problem)**.** In the *maximum matching problem* the input is an undirected graph $G = (V, E)$ and the output is a maximum matching in G.

Definition (Augmenting path)**.** Let $G = (V, E)$ be a graph and let $M \subseteq E$ be a matching in G. An *augmenting path* in G with respect to M is a path such that

1. the path is an *alternating path*, which means that the edges in the path alternate between being in M and not in M,
2. the first and last vertices in the path are not a part of the matching M.

Example (Augmenting path example 1)**.** Consider a single edge $\{u, v\}$ in a graph G such that u and v are not matched by a matching M (this means that vertices u and v do not belong to any of the edges in M). Then this edge forms an augmenting path.

Example (Augmenting path example 2)**.** Consider the following graph.

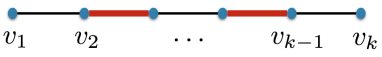


Let M be the matching $\{\{v_1, v_5\}, \{v_3, v_6\}, \{v_4, v_8\}\}$. Then the path (v_2, v_5, v_1, v_7) is an augmenting path with respect to M.

Note (Edge cases for augmenting paths)**.** An augmenting path does not need to contain all the edges in M. It is also possible that it does not contain *any* of the edges of M. A single edge $\{u, v\}$ where u and v are not matched is an augmenting path.

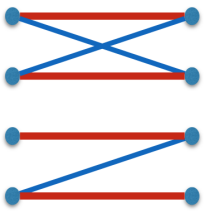
Theorem (Characterization for maximum matchings)**.** *Let $G = (V, E)$ be a graph. A matching $M \subseteq E$ is maximum if and only if there is no augmenting path in G with respect to M.*

Proof. The statement we want to prove is equivalent to the following. Given a graph $G = (V, E)$, a matching $M \subseteq E$ is not maximum if and only if there is an augmenting path in G with respect to M. There are two directions to prove.
First direction: Suppose there is an augmenting path in G with respect to M. Then we want to show that M is not maximum. Let the augmenting path be $v_1, v_2, \ldots, v_k$:



The highlighted edges represent edges in M. By the definition of an augmenting path, we know that v_1 and v_k are not matched by M. Since v_1 and v_k are not matched and the path is alternating, the number of edges on this path that are in the matching is one less than the number of edges not in the matching. To see that M is not a maximum matching, observe that we can obtain a bigger matching by flipping the matched and unmatched edges on the augmenting path. In other words, if an edge on the path is in the matching, we remove it from the matching, and if an edge on the path is not in the matching, we put it in the matching. This gives us a matching larger than M, so M is not maximum.

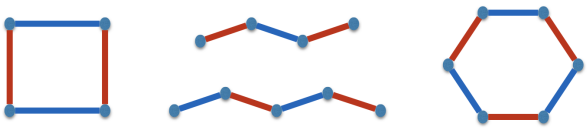
Second direction: We now prove the other direction. In particular, we want to show that if M is not a maximum matching, then we can find an augmenting path in G with respect to M. Since M is not a maximum, there is a matching with larger size. Let M^* be a matching such that $|M^*| > |M|$. We define the set S to be the set of edges contained in M^* or M, but not both. That is, $S = (M^* \cup M) \backslash (M^* \cap M)$. If we color the edges in M with blue, and the edges in M^* with red, then S consists of edges that are colored either blue or red, but not both (i.e. no purple edges). Below is an example:



(Horizontal edges correspond to the red edges. The rest is blue.) Our goal is to find an augmenting path with respect to M in S (i.e., with respect to the blue edges), and once we do this, the proof will be complete.

We now proceed to find an augmenting path with respect to M in S. To do so, we make a couple of important observations about S. First, notice that each vertex that is a part of S has degree 1 or 2 because it can be incident to at most one edge in M and at most one edge in M^*. (If the degree was more than 2, either two edges from M or two edges from M^* would be intersecting, but in a matching, edges cannot intersect.) We now make two claims:

1. Because every vertex has degree 1 or 2, S consists of disjoint paths and cycles (i.e. each connected component is either a path or a cycle).



2. The edges in these paths and cycles alternate between blue and red.

The proof of the first claim is omitted and is left as an exercise for the reader. The second claim is true because if the edges were not alternating, i.e., if there were two red or two blue edges in a row, then this would imply the red edges or the blue edges do not form a matching (remember that in a matching no two edges can share an endpoint).

Since M^* is a bigger matching than M, we know that S has more red edges than blue edges. Observe that the cycles in S must have even length, because otherwise the edges cannot alternate between blue and red. Therefore the cycles have an equal number of red and blue edges. This implies that there must be a path in S with more red edges than blue edges. In particular, this path starts and ends with a red edge. This path is an augmenting path with respect to M (i.e., the blue edges), since it is clearly alternating between edges in M and edges not in M, and the endpoints are unmatched with respect to M. So using the assumption that M is not maximum, we were able to find an augmenting path with respect to M. This completes the proof. □

Exercise (Graphs with max degree at most 2)**.** Let $G = (V, E)$ be a graph such that all vertices have degree at most 2. Then prove that every connected component of G is either a path or a cycle (where we count an isolated vertex as a path of length 0).

Solution. Consider a graph G such that all vertices have degree at most 2. We want to show that it consists of disjoint paths and cycles. We prove this by induction on the number of vertices.

Pick an arbitrary vertex v in the graph. Removing v results in a graph $G - v$ such that every vertex has degree at most 2. Since $G - v$ has one less vertex, by induction hypothesis, $G - v$ consists of disjoint paths and cycles. There are 3 cases to consider: $\deg(v) = 0, 1,$ or 2. It is not hard to see that in each case, adding v back to the graph preserves the property that the graph is a collection of disjoint paths and cycles. (Verify this part for yourself.) ■

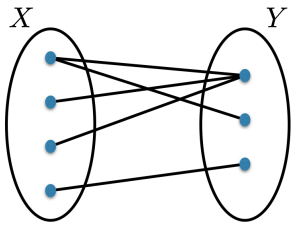
Exercise (A tree can have at most one perfect matching)**.** Show that a tree can have at most one perfect matching.

Solution. The proof is by contradiction, so suppose a tree has two different perfect matchings M and M'. Let S be the symmetric difference between M and M', i.e., $S = (M \cup M')\backslash(M \cap M')$. Since $M \neq M'$, $|S| > 1$. The set S corresponds to a graph in which each vertex has degree at most 2. So the graph consists of disjoint paths and cycles (i.e. each connected component is either a path or a cycle). But a connected component cannot be a cycle since trees are acyclic. It also cannot be a path. This is because the existence of a degree 1 vertex in S implies that this vertex is not covered/matched by either M or M' (verify this yourself), and this would contradict the fact that M and M' are *perfect* matchings covering all vertices. So S must be the empty set, which contradicts our assumption that $|S| > 1$. ■

2 Bipartite Graphs

Definition (Bipartite graph)**.** A graph $G = (V, E)$ is called *bipartite* if there is a partition[2] of V into sets X and Y such that all the edges in E have one endpoint in X and the other in Y. Sometimes the bipartition is given explicitly and the graph is denoted by $G = (X, Y, E)$.

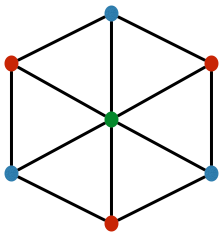
Example (Bipartite graph example)**.** Below is an example of a bipartite graph.



Definition (k-colorable graphs)**.** Let $G = (V, E)$ be a graph. Let $k \in \mathbb{N}^+$. A *k-coloring* of V is just a map $\chi : V \to C$ where C is a set of cardinality k. (Usually the elements of C are called *colors*. If $k = 3$ then $C = \{\text{red}, \text{green}, \text{blue}\}$ is a popular choice. If k is large, we often just call the "colors" $1, 2, \ldots, k$.) A k-coloring is said to be *legal* for G if every edge in E is *bichromatic*, meaning that its two endpoints have different colors. (I.e., for all $\{u, v\} \in E$ it is required that $\chi(u) \neq \chi(v)$.) Finally, we say that G is *k-colorable* if it has a legal k-coloring.

Example (A 3-colorable graph)**.** The graph below is 3-colorable. We can color the vertex at the center green, and color the outer vertices with blue and red by alternating those two colors.

[2] Recall that a *partition* of V into X and Y means that X and Y are disjoint and $X \cup Y = V$.



Note (2-colorability is equivalent to bipartiteness)**.** A graph $G = (V, E)$ is bipartite if and only if it is 2-colorable. The 2-coloring corresponds to partitioning the vertex set V into X and Y such that all the edges have one endpoint in X and the other in Y.

Theorem (Characterization of bipartite graphs)**.** *A graph is bipartite if and only if it contains no odd-length cycles.*

Proof. There are two directions to prove.
($\Longrightarrow$): For this direction, we want to show that if a graph is bipartite, then it contains no odd-length cycles. We prove the contrapositive. Observe that it is impossible to 2-color an odd-length cycle. So if a graph contains an odd-length cycle, the graph cannot be 2-colored, and therefore cannot be bipartite.
($\Longleftarrow$): For this direction, we want to show that if a graph does not contain an odd-length cycle, then it is bipartite. So suppose the graph contains no cycles of odd length. Without loss of generality, assume the graph is connected (if it is not, we can apply the argument to each connected component separately). For $u, v \in V$, let dist(u, v) denote the length of the shortest path from u to v (or from v to u). Pick a starting vertex/root s and consider the "BFS tree" rooted at s. In this tree, level 0 corresponds to s, and level i corresponds to all vertices v with dist$(s, v) = i$. Color odd-indexed levels blue, and color even-indexed levels red.

The proof is done once we show that this is a valid 2-coloring of the graph. To show this, we'll argue that no edge has its endpoints colored the same color. There are two types of edges we need to worry about that could potentially have its endpoints colored the same color. We consider each type below.

First, there could potentially be an edge between two vertices u and v at the same level. Let's assume such an edge exists. Let w be the lowest common ancestor of u and v. Note that dist$(u, w) =$ dist(v, w), so the path from w to u, plus the path from w to v, plus the edge $\{u, v\}$, form an odd-length cycle. This is a contradiction.

Second, we need to consider the possibility that there is an edge between a vertex u at level i and a vertex v at level $i + 2k$ for some $k > 0$. However, the existence of such an edge implies that dist$(s, v) \leq i + 1$, which contradicts the fact that v is at level $i + 2k$. So this type of edge cannot exist either. This completes the proof. □

Theorem (Finding a maximum matching in bipartite graphs)**.** *There is a polynomial time algorithm to solve the maximum matching problem in bipartite graphs.*

Proof. Let $G = (X, Y, E)$ be the input graph. The high level steps of the algorithm is as follows.

- Let $M = \{\{x, y\}\}$ where $\{x, y\} \in E$ is an arbitrary edge.
- Repeat until there is no augmenting path with respect to M:
  - Find an augmenting path with respect to M.
  - Update M according to the augmenting path (swapping matched and unmatched edges along the path).

Every time we find an augmenting path, we increase the size of our matching by one. When there are no more augmenting paths, we stop and correctly output a maximum matching (the correctness follows from Theorem (Characterization for maximum matchings)). The only unclear step of the algorithm is finding an augmenting path with respect to M. And we explain how to do this step below. But before we do that, note that if this step can be done in polynomial time, then the overall running time of the algorithm is polynomial time since the loop repeats $O(n)$ times and the work done in each iteration is polynomial time.

We now show how to find an augmenting path (given $G = (X, Y, E)$ and $M \subseteq E$). We first turn G into a directed graph $\vec{G}$ as follows:

- Direct edges in $E \backslash M$ from X to Y.
- Direct edges in M from Y to X.

Note that an alternating path with respect to M in G corresponds to a directed path in $\vec{G}$. (We leave it to the reader to verify this.) This means that there is an augmenting path with respect to M in G if and only if there is a directed path in $\vec{G}$ from an unmatched vertex x in X to an unmatched vertex y in Y. So to find an augmenting path, we'll search for a directed path in $\vec{G}$ from an unmatched x to an unmatched y, as follows:

- For each unmatched $x \in X$:
  - Run DFS(G, x).
  - If you hit an unmatched $y \in Y$, output the path from x to y.
- Output "no augmenting path found."

The running time is polynomial time since the loop repeats at most $O(n)$ times, and the work done in each iteration is polynomial time. □

Note (Finding a maximum matching in non-bipartite graphs)**.** The high-level algorithm above presented in the proof of Theorem (Finding a maximum matching in bipartite graphs) is in fact applicable to general (not necessarily bipartite) graphs. However, the step of finding an augmenting path with respect to a matching turns out to be much more involved, and therefore we do not cover it in this chapter. See https://en.wikipedia.org/wiki/Blossom_algorithm if you would like to learn more.

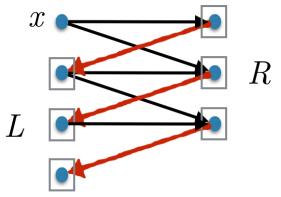
3 Bonus: Hall's Theorem

Theorem (Hall's Theorem)**.** *Let $G = (X, Y, E)$ be a bipartite graph. For a subset S of the vertices, let $N(S) = \bigcup_{v \in S} N(v)$. Then G has a matching covering all the vertices in X if and only if for all $S \subseteq X$, we have $|S| \leq |N(S)|$.*

Proof. There are two directions to prove.
($\Longrightarrow$): For this direction, we need to show that if G has a matching covering all the vertices in X, then every $S \subseteq X$ satisfies $|S| \leq |N(S)|$. We consider the contrapositive. So suppose there is some $S \subseteq X$ such that $|S| > |N(S)|$. The vertices in S can *only* be matched to vertices in $N(S)$, and since $|S| > |N(S)|$, there cannot be a matching that covers every element in S. And this implies there cannot be a matching covering every element of X.
($\Longleftarrow$): For this direction, we need to show that if every $S \subseteq X$ satisfies $|S| \leq |N(S)|$, then there is a matching that covers all the vertices in X. We will prove the contrapositive. So assume there is no matching that covers all the vertices in X. Our goal is to find some $S \subseteq X$ such that $|S| > |N(S)|$.

In order to identify such a set S, we need to make a couple of definitions. Let M be a maximum matching and let $x \in X$ be an element that it does not cover. We turn G

into a directed graph as follows: direct all edges not in M from X to Y, and direct all edges in M from Y to X. We define $L \subseteq X$ to be the set of vertices in X that you can reach by a directed path starting at x (L does not include x). And we define $R \subseteq Y$ to be the set of all vertices in Y that you can reach by a directed path starting at x. Here is an illustration:



We will show that for $S = L \cup \{x\}$, we have $|S| > |N(S)|$. We need two claims to argue this.

Claim 1: $|L| = |R|$.

Proof: Each $\ell \in L$ is matched to some $r \in R$ because the only way we can reach an $\ell \in L$ is through an edge in the matching. Conversely, each $r \in R$ must be matched to some $\ell \in L$ since if this was not true, i.e., if there was an unmatched $r \in R$, that would imply that the path from x to r is an augmenting path, and this would contradict the fact that M is a maximum matching (Theorem (Characterization for maximum matchings)). Since every element of L is matched by M to an element of R and vice versa, there is a one-to-one correspondence between L and R, i.e., $|L| = |R|$.

Claim 2: In the original undirected graph, $N(L \cup \{x\}) \subseteq R$.

(In fact, $N(L \cup \{x\}) = R$ but we only need one side of the inclusion.)

Proof: For any $\ell \in L \cup \{x\}$, we want to argue that $N(\ell) \subseteq R$. First consider the case that $\ell = x$. Then all the neighbors of x must be in R since all the edges incident to x are directed from left to right. So $N(x) \subseteq R$. Now consider any $\ell \in L$. We want to argue that all the neighbors of ℓ must be in R. To argue about the neighbors of ℓ, we look at all the edges incident to ℓ. In this set of edges incident to ℓ, exactly one edge e is in the matching M. Since $e \in M$ is directed from Y to X, it must be incident to some $r \in R$ because the only way to reach ℓ is through some $r \in R$ via e. If we now look at all the other edges incident to ℓ, note that they must be directed from X to Y, and the vertices $K \subseteq Y$ that they are incident to must be in R. This is because by definition of L, $\ell \in L$ is reachable from x, which means the vertices in K would also be reachable from x, and therefore would be in R (by the definition of R). This shows that every neighbor of ℓ is in R, and completes the proof of Claim 2.

Combining Claim 1 and Claim 2 above, we have

$$|L \cup \{x\}| > |R| \geq |N(L \cup \{x\})|,$$

i.e., for $S = L \cup \{x\}$, $|S| > |N(S)|$, as desired. □

Corollary (Characterization of bipartite graphs with perfect matchings)**.** *Let $G = (X, Y, E)$ be a bipartite graph. Then G has a perfect matching if and only if $|X| = |Y|$ and for any $S \subseteq X$, we have $|S| \leq |N(S)|$.*

Note (Hall's Theorem when the two parts have equal size)**.** Sometimes people call the above corollary Hall's Theorem.

Exercise (Practice with perfect matchings)**.**

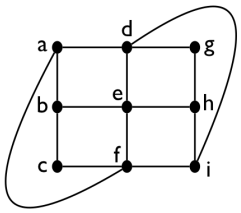
1. Let G be a bipartite graph on $2n$ vertices such that every vertex has degree at least n. Prove that G must contain a perfect matching.
2. Let $G = (X, Y, E)$ be a bipartite graph with $|X| = |Y|$. Show that if G is connected and every vertex has degree at most 2, then G must contain a perfect matching.

Solution. Part 1: If every vertex has degree n, it must be the case that the graph is $G = (X, Y, E)$ where $|X| = |Y| = n$. It must also be the case that the graph is a complete bipartite graph (i.e., every possible edge is present). So clearly Hall's theorem applies and the graph has a perfect matching.
Part 2: A graph with max degree at most 2 consists of disjoint paths and cycles (Exercise (Graphs with max degree at most 2)). Since the graph is connected, it must consist of a single path or a single cycle containing all the vertices. In either case, it is not hard to see that the graph must contain a perfect matching: If the graph is a path, then we can take every other edge (starting with the first edge) along the path to form a perfect matching. If the graph consists of a cycle, it has two different perfect matchings. ■

4 Check Your Understanding

Problem. 1. True or false: A maximum matching is a maximal matching.

2. True or false: A perfect matching is a maximum matching.
3. True or false: There exist graphs with more than one perfect matching.
4. How many perfect matchings are there in a complete bipartite graph (all possible edges are present) where both sides of the bipartition contain exactly n vertices?
5. Suppose a graph with n vertices has a perfect matching. What is the size of the perfect matching?
6. What is the definition of an augmenting path?
7. True or false: Given a matching M, there can be at most one augmenting path with respect to M.
8. True or false: A matching M in a non-bipartite graph G is maximum if and only if there is no augmenting path with respect to M.
9. Describe the high-level steps of a polynomial-time algorithm for finding a maximum matching in a graph.
10. Describe a polynomial-time algorithm that given a bipartite graph and a matching in the graph, determines if an augmenting path exists with respect to the matching.
11. True or false: The graph below is bipartite.



12. True or false: If a graph is k-colorable, then it is $(k+1)$-colorable.
13. True or false: A graph which is not bipartite must contain an odd-length cycle.
14. Explain how one can 2-color a graph that does not contain an odd-length cycle.
15. Describe a polynomial-time algorithm to determine if an input undirected graph is bipartite or not.
16. How do you prove that a tree has at most one perfect matching?

17. True or false: Any graph with more than one perfect matching must contain a cycle.

18. In this chapter we saw the definition of a bipartite graph. We can think of a bipartite graph as a special case of a k-partite graph where $k = 2$. How would you define the general notion of a k-partite graph, and how does it relate to the colorability of the graph?

5 High-Order Bits

Important. Here are the important things to keep in mind from this chapter.

1. As always, all the definitions in this chapter are important (matchings, augmenting path, bipartite graph, k-coloring of a graph).

2. Theorem (Characterization of bipartite graphs) gives an important characterization for bipartite graphs. Make sure to understand its proof.

3. Theorem (Characterization for maximum matchings) gives an important characterization for maximum matchings. The proof is one of the harder proofs we have done in the course so far. A good understanding of the proof is expected and important since it will help you raise your level of mathematical maturity.

4. Make sure to understand how we can use Theorem (Characterization for maximum matchings) to come up with an algorithm for finding a maximum matching in a graph.