MATH 239 Introduction to Graph Theory

Jude Gao
University of Waterloo

2018 FALL

Contents

Chapter 1

Basic Concepts

1.1. What is a Graph?

Definition 1.1 (Graph)**.** A **graph** G is a pair (V, E) such that

- V is a set whose elements are called **vertices** of G, and
- E is a set whose elements are pairs of distinct vertices, those are called **edges**.

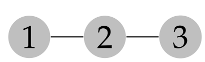
We will denote the vertex set as $V(G)$ and the edge set as $E(G)$. In Computer Science terms, V and E are operators.

Example 1.2. Consider a graph G where

$$V(G) = \{1, 2, 3\},$$

and

$$E(G) = \{\{1, 2\}, \{2, 3\}\}.$$



is a **drawing** of G.

Definition 1.3 (Drawing)**.** A **drawing** of a graph is a pictorial representation where vertices are denoted with points and a curve connects every pair of points that form an edge.

Remark 1.4. Note that we may use curves to connect points, not necessarily straight lines. Also, a drawing is not necessarily 2-dimensional. Hence, we can allow edges to "cross". It is helpful to think of edges as string. Also, we may have different drawings of a graph G.

1.2. Different Kinds of Graphs

Definition 1.5 (Planar Graph)**.** A graph is **planar** if it can be drawn in the plane without crossing.

Remark 1.6. We will study planar graphs in the future, but not every graph is planar!

$E(G)$ is a set, not a multiset. However, there is a more general concept, called multiset, in which $E(G)$ is a multiset and can consist of pairs and/or singletons. There are even more generalizations of graphs, which we will not study. They are

- **Weighted Graph**: edges are assigned **weights**, usually integer or real numbers.
- **Directed Graph**: edges are ordered pairs.
- **Hyper-graphs**: edges can be of any size.

The goals for this course are

- to learn basic graph terminology,
- to understand basic types or classes of graphs,
- to prove structure theorems for these classes, and
- to teach you how to write a Graph Theory proof.

Example 1.7.

- **Complete Graphs**: a family of graphs, denoted

$$K_n : V(K_n) = \{1, 2, \ldots, n\}, E(K_n) = \{\{i, j\} : i \neq j, i, j \in \{1, \ldots, n\}\}$$

Fun Fact: K_n is planar for $n \leq 4$, not planar for all $n \geq 5$.

- **Paths**: a family of graphs for each n, denoted

$$P_n : V(P_n) = \{1, 2, \ldots, n\}, E(P_n) = \{\{1, 2\}, \{2, 3\}, \ldots, \{n-1, n\}\}$$

Fun Fact: $P_1 = K_1$, $P_2 = K_2$, $P_3 \neq K_3$. P_n is planar $\forall n$.

- **Cycles**: a family of graphs for each $n \geq 3$, denoted

$$C_n : V(C_n) = \{1, \ldots, n\}, E(C_n) = \{\{1, 2\}, \ldots, \{n-1, n\}, \{1, n\}\}$$

Fun Fact: C_n is planar $\forall n$. C_n is also referred as an $n-$cycle.

- **Bipartite Graph**: a graph is bipartite if there exists a partition of $V(G)$ into parts A and B such that for each $\{u, v\} \in E(G)$,

$$|\{u, v\} \cap A| = |\{u, v\} \cap B| = 1.$$

  Every edge has one end point in A and the other in B.

1.3. Introduction to Complexity Theory

To study the theorems throughout the course, we want to talk about them in the context of decision problems with their complexities. In this section, we will introduce several basic definitions in Complexity Theory with respect to the following questions:

- It is **easy** to **see** that a graph is bipartite?
- Is it **easy** to **see** that a graph is **NOT** bipartite?

We will make clear what we mean by "easy" and "see" during the discussion in Complexity Theory.

Definition 1.8 (Decision Problem)**.** A **decision problem** is a problem that has a **YES** or **NO** answer.

Example 1.9. The examples of decision problems are

- Is G planar?
- Is G bipartite?

Definition 1.10 (P)**.** A decision problem is in P if there exists a **polynomial** time algorithm to solve it.

Definition 1.11 (NP)**.** There are two possible definitions:

- A decision problem is in NP if there exists a polynomial time algorithm in the input to verify that a **YES**-solution is actually a solution.
- A decision problem is in NP if there exists a polynomial time algorithm to decide the decision problems if you are the "luckiest possible guessers".

Obviously, we know that $P \subset NP$, but whether $P \neq NP$ is still a conjecture.

Definition 1.12 (NP-Complete)**.** A decision problem is NP-complete if it being in P implies that every NP-problems is in P.

Definition 1.13 (Co-NP)**.** A problem is in co-NP if there exists a polynomial algorithm to verify a **NO**-solution.

$NP =$co-NP is also a conjecture.
Let's return to our questions:

- It is **easy** to **see** that a graph is bipartite?
- Is it **easy** to **see** that a graph is **NOT** bipartite?

Now, this can be rephrased as

- Is the Bipartite Decision Problem in NP?
- Is the Bipartite Decision Problem in co-NP?
- Is the Bipartite Decision Problem in P?

Theorem 1.14. *Bipartite Decision Problem is in NP.*

Proof. Given a solution, i.e. a bipartition A, B, of the graph, it is linear time, i.e. polynomial time, to decide if it is correct: Check for edge $e \in E(G)$, if one end is in A, and the other is in B. □

However, is it in co-NP?

Naively, we need to show that every bipartition has an edge with ends on the same side. Since there are exponential number of bipartitions, we would think that Bipartite Decision Problem was not in co-NP.

Theorem 1.15. *A graph is bipartite if and only if it contains no odd cycles as a* ***subgraph****.*

Definition 1.16 (Subgraph)**.** A **subgraph** H of G is a graph that satisfies

- $V(H) \subset V(G)$
- $E(H) \subset V(G)$

Theorem 1.17. *The Bipartite Decision Problem is in co-NP.*

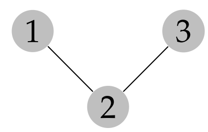
Proof. As a **NO**-solution is given as an **ordered list** of vertices $v_1, v_2, \ldots, v_k$, there exists a polynomial time algorithm to verify, namely, to check $\{v_1, v_2\}$,$\{v_2, v_3\}, \ldots, \{v_{k-1}, v_k\}, \{v_k, v_1$ are connected and that k is odd. □

1.4. Isomorphism

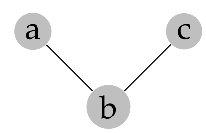
Definition 1.18 (Isomorphism)**.** An **isomorphism** from a graph G to graph H is a bijection $V(G)$ to $V(H)$ that preserves edges. More formally, if f is the bijection, then $\forall u, v \in V(G)$, there exist $u, v \in E(G)$ such that $\{f(u), f(v)\} \in E(H)$.

Example 1.19.

G



H



Here is a possible isomorphism from G to H:

$$f(a) = 3, f(b) = 1, f(c) = 2$$

Check that edges are preserved:

$$\{1,2\} \iff \{b,c\}$$

$$\{1,3\} \iff \{b,a\}$$

$$\{2,3\} \iff \{c,a\}$$

Definition 1.20. A graph G and H are isomorphic if there exists an isomorphism from G to H or from H to G.

We only care about properties of graphs that hold up to isomorphisms. We do not care about labels. We are interested in whether a graph is bipartite, or planar.

In the context of Complexity Theory,

- Is it easy to see if two graphs are isomorphic?

  YES, Graph Isomorphism is in NP. We check the bijections.

- Is it easy to decide if two graphs are isomorphic?

  State-of-the-art: Graph Isomorphism is either not known to be in P, or to be in NP-complete.

  However, (2015) there exists a quasi-polynomial time algorithm $(2^{\text{polylog}(|V(G)|)})$.

1.5. Basic Terminology

Definition 1.21 (Adjacent)**.** We say two vertices u, v are **adjacent** if $\{u, v\} \in E(G)$.

Definition 1.22 (Neighbour)**.** We say an edge u is a **neighbour** of an edge v, if $\{u, v\} \in E(G)$.

Definition 1.23 (Neighbourhood)**.** The **neighbourhood** of an edge v, denoted $N(v)$ is the set of neighbours of v.

Definition 1.24 (Degree)**.** The **degree** of a vertex v, denoted $\deg(v)$, is the number of neighbours of v

$$\deg(v) = |N(v)|.$$

Definition 1.25. We say a vertex v and an edge e are **incident**, written $e \sim v$, if $v \in e$.

Definition 1.26. We say two edges b, f are incident, if there exists a vertex v such that $v \sim e, v \sim f$.

If two graphs G, H are isomorphic,

$$N(v) \leftrightarrow N(f(v)), \deg_G(v) = \deg_H(f(v)).$$

Definition 1.27 (k-regular)**.** A graph G is k-regular, if every vertex has degree k.

Example 1.28.
- Peterson is 3-regular.
- K_n is $n-1$-regular.
- C_n is 2-regular.

Definition 1.29. A graph is **regular** if it is k-regular for some k.

Example 1.30. P_n is not regular for $n \geq 3$.

1.6. Our First Lemma

Lemma 1.31 (Handshaking Lemma)**.** *If G is a graph, then*

$$2|E(G)| = \sum_{v \in V(G)} \deg(v).$$

Proof.

$$\begin{aligned} 2|E(G)| &= \sum_{e \in E(G)} 2 \\ &= \sum_{e \in E(G)} \sum_{\substack{v \in V(G) \\ v \sim e}} 1 \\ &= \sum_{v \in V(G)} \sum_{\substack{e \in E(G) \\ e \sim v}} 1 \\ &= \sum_{v \in V(G)} \deg(v) \end{aligned}$$

□

Corollary 1.32. *The average degree of a graph G is*

$$\frac{2|E(G)|}{|V(G)|}.$$

Proof.

$$\text{Average Degree} = \frac{\sum \deg(v)}{|V(G)|} = \frac{2|E(G)|}{|V(G)|}.$$

□

Corollary 1.33. *If G is k-degree, then*

$$|E(G)| = \frac{k|V(E)|}{2}.$$

Proof. The average degree of any k-degree graph is, clearly, k. By the theorem above,

$$\text{Average Degree} = k = \frac{2|E(G)|}{|V(G)|} \implies |E(G)| = \frac{k|V(E)|}{2}.$$

□

Example 1.34.
- Peterson has $\frac{(3)(10)}{2} = 15$ edges, since $|V(G)| = 10$ and it is 3-regular.

- $|E(K_n)| = \frac{(n-1)n}{2} = \binom{n}{2}$, since K_n are $(n-1)$-regular.

Corollary 1.35. *The number of vertices in G that have odd degree is even.*

Proof. $2|E(G)|$ is even. By the **Handshaking Lemma**,

$$2|E(G)| = \sum_{v \in G} \deg(v),$$

and hence latter is even.

Let

$$O_G = \{v \in V(G) : \deg(V) \text{is odd}\}$$

and

$$E_G = \{v \in V(G) : \deg(v) \text{is even}\}.$$

O_G and E_G partition $V(G)$.

Hence,

$$\underbrace{\sum_{v \in V(G)} \deg(v)}_{\text{even}} = \underbrace{\sum_{v \in O_G} \deg(v)}_{\textbf{even}} + \underbrace{\sum_{v \in E_G} \deg(v)}_{\text{even}}$$

□

Chapter 2

Fundamental Notions

2.1. Paths and Walks

Definition 2.1 (Walk)**.** A **walk** W in a graph G is an alternating sequence of vertices and edges

$$v_0e_1v_1e_2v_2\cdots e_{k-1}v_{k-1}e_kv_k$$

such that

$$e_i = v_{i-1}v_i \ \ \forall i, 1 \le i \le k$$

Example 2.2. An example of a walk:

$$W = 1, 12, 2, 23, 3, 31, 1, 13, 3, 34, 3$$

The v_i need not be distinct. The length of the walks is the number of edges. The example above is a walk with length 5.

Definition 2.3 (Path)**.** A **path** in a graph G is a walk W where all vertices are distinct.

Again, the length of a path is the number of edges.

Definition 2.4 (v_0v_n-walk)**.** We say $W = v_0 \cdots v_n$ is a v_0v_n-walk.

If we reverse the sequence in W, we get a v_nv_0-walk. Hence we often talk of there existing a walk between v_0 and v_k, not caring about the direction.

Definition 2.5 (Closed Walk)**.** We call a walk **closed** if $v_0 = v_k$.

The closed walk has the start and the end the same vertex. There can be a path/walk of length 0, namely one vertex v, no edges.

We are interested in answering the following question:

When does there exist a path/walk from vertex x to vertex y?

Of course, if there exists a path from x to y, then there exists a walk from x to y, namely the path itself. However, the converse is also true.

Theorem 2.6. *If there exists a walk from vertex x to vertex y in a graph G, then there exists a path from x to y in G.*

Proof. Idea: if a walk is not a path, shorten it.

First Proof: Proof by algorithm. Algorithm to turn a walk $w = v_0 \cdots v_k$ with $v_0 = x$ and $v_k = y$ into a path from x to y.

while ($\exists i < j$ such that $v_i = v_j$)

  $W = v_0 \ldots v_i v_{j+1} \ldots v_k$

return W

You are not done yet! This is not a proof. We still have to

- Prove correctness.
- Prove algorithm terminates.

Proof of correctness:

- It returns a path since the while-loop failed to execute, when $\nexists i < j$ such that $v_i = v_j$.
- The start is x and the end is y, because W always preserves this property.

Proof of termination: In every execution of the loop, the length of W decreases. The length is finite, so the algorithm terminates. □

Proof. Second Proof: Proof by the Minimum Counterexample. Idea: suppose existence of counterexample minimum with respect to some parameter(s). Then, derive contradiction. Given an algorithmic proof or idea, construct a minimum counterexample proof by 'skipping to the end'.

Suppose not. Let $W = v_0 \cdots v_n$ be a walk with $v_0 = x, \ldots, v_n = y$, and, subject to that, W has minimum length. W exists because

- $\exists$ a walk from x to y by assumption.
- The length of walks is finite in G, since G is finite.

Since W is not a path, $\nexists$ a path from x to y. Then $\exists i < j$ such that $v_i = v_j$. We have $w = v_0 \cdots v_i v_j v_{j+1} \cdots v_k$. But now we shorten it to

$$w = v_0 \cdots v_i v_{j+1} \cdots v_k.$$

This is a shorter length from x to y of shorter length than W, a contradiction.

□

Corollary 2.7. *If there exists a path from x to y, and a path from y to z in a graph G, then there exists a path from x to z.*

Proof. Let P_1 be a path from x to y and P_2 be a path from y to z.

Now P_1P_2, the sequence formed by concatenation, is a walk from x to z. This might not be path because vertices may repeat. Hence there exists a walk from x to z by the theorem above, then there exists a path from x to z. □

2.2. Connected Components

Definition 2.8. A graph G is **connected** if $\forall x, y \in V(G)$, $\exists$ a path from x to y in G.

Question: Is deciding if a graph connected in NP? Co-NP? P?

Theorem 2.9. *The Deciding Connectedness is in NP.*

Proof. The prover provides a verifier with a path from x to y, $\forall x, y \in V(G)$. It is $O(V(G))$ for the verifier to check each part is a path from x to y. Since there are $|V(G)|^2$ paths given in total, this is $O(|V(G)|^3|$ time. □

Question: How to show it's in co-NP?

Suffice to examine that $\exists x, y \in V(G)$ and no path from x to y. How to verify **no path**? We would need to show all paths but there are maybe exponentially many paths in G, so it is not efficient.

Definition 2.10 (Equivalence Relation)**.** A relation $\sim_R$ on a set S is an **equivalence relation** if all the following hold:

- $\forall x \in S, x \sim_R x$. (Reflexive)
- $\forall x, y \in S, x \sim_R Y \implies Y \sim_R X$. (Symmetric)
- $\forall x, y, z \in S, x \sim_R y, y \sim_R z \implies x \sim_R z$. (Transitive)

Definition 2.11 (Equivalence Class)**.** An **equivalence class** under $\sim_R$ is a maximal subset of S whose elements are all pair-wise related.

The equivalence class under $\sim_R$ containing $x \in S$ is

$$\{y \in S : x \sim_R y\}.$$

Lemma 2.12. *If $\sim_R$ is an equivalence relation, then S **partitions** into equivalence classes.*

Definition 2.13 (Path Relation)**.** Let G be a graph. We define a relation $\sim_{\text{Path}}$ on $V(G)$ as follows:

$$\forall x, y \in V(G), x \sim_{\text{Path}} y \iff \exists \text{ a path } x \text{ to } y \text{ in } G.$$

Lemma 2.14. *If G is a graph, then $\sim_{Path}$ is a equivalence relation on $V(G)$.*

Proof. It suffices to check three properites:

- Reflexive: $\forall x \in V(G)$, $\exists$ a path from x to x, because x is itself a path from x to x.
- Symmetric: $\forall x, y \in G$, $\exists$ a path from x to y, then $\exists$ a path from y to x, because we can reverse the path.
- Transitive: $\forall x, y, z \in V(G), \exists$ a path from x to y, a path from y to z, then $\exists$ a path from x to z. (By the Theorem)

□

Definition 2.15 (Component)**.** An equivalence classes under $\sim_{\text{Path}}$ in a graph G is called a **component** of G. Similarly, a vertex $v \in V(G)$, then equivalence class under $\sim_{\text{Path}}$ containing vertex v is called the component of G containing v.

Corollary 2.16. *Our earlier proposition about equivalence relation gives the following: If G is a graph, then $V(G)$ partitions into its components.*

Theorem 2.17. *A graph G is connected iff $V(G)$ is a component, or equivalently, G is disconneted iff G has at least 2 components.*

2.3. Cut and $CO - NP$ Characterization of Connectedness

Definition 2.18 (Cut)**.** Let G be a graph, and $X \subset V(G)$. The (edge) cut induced by X, denoted $\delta(X) = \{e = xy \in E(G) : x \in X, y \notin X\}$. (I.e. the list of edges exactly one ending X)

Lemma 2.19. *$\delta(X) \neq \emptyset$ iff $\exists$ a component C of a graph G such that $C \cap X \neq \emptyset$ and $C \cap (V(G) - X) \neq \emptyset$.*

Proof. Assume $\delta(X) \neq \emptyset$. In particular, $\exists e = xy \in \delta(X)$ and $x \in X, y \notin X$. Let C be the component containing x. Since xy itself is a path, we have $y \in C$. Hence $x \in C \cap X$, and $y \in C \cap (V(G) - X)$, as desired.

Assume $\exists$ a component C such that $C \cap X \neq \emptyset$, and $C \cap (V(G) - X) = \emptyset$. Let $x \in C \cap X$ and $y \in C \cap (V(G) - X)$. Since C is a component, and $x, y \in C$, $\exists$ a path $P = v_0 v_1 \cdots v_k$ in C such that $v_0 = x, v_k = y$. Let i be maximum such that $v_i \in X$. This exists because $v_0 = x \in X$. However $i < k$ since $v_k = y \notin X$. Then, $v_i v_{i+1} \in \delta(X)$.

□

Corollary 2.20. *Let G be a graph, $X \subset V(G)$. Then $\delta(X) = \emptyset$ iff $\exists$ components $C_1, C_2, \ldots, C_k$ of G such that $X = C_1 \cup C_2 \cup \cdots \cup C_k$*

Corollary 2.21. *Let G be a graph. G is connected iff $\exists X \subsetneq V(G)$, $X \neq \emptyset$ such that $\delta(X) \neq \emptyset$. G is disconnected iff $\exists X \subsetneq V(G)$, $X \neq \emptyset$ such that $\delta(X) = \emptyset$.*

Theorem 2.22. *Deciding if a graph is connected is in co-NP.*

Proof. The prover gives the verifier a set $X \subsetneq V(G), X \neq \emptyset$, such that $\delta(X) = \emptyset$. Then, the verifier checks this. □

2.4. Trees and Forests

Definition 2.23 (Forest)**.** A graph G is a forest if G does NOT contain a cycle as a subgraph.

Definition 2.24 (Tree)**.** A connected forest is called a **tree**.

Lemma 2.25. *Every component of a forest is a tree.*

Proof. Every component has no cycles, since the forest does not. Also, component is connected. Hence every component of a forest is a tree. □

Do remember that every tree is a forest but not every forest is a tree.
Question: Is deciding if a graph is a forest in NP? Co-NP? P?
The answers are all yes, but one is obvious.

Theorem 2.26. *Deciding if a graph is a forest is in co-NP.*

Proof. The prover gives a verifier a cycle. □

How to show it is in NP? How to show that there is no cycle?

Theorem 2.27. *Deciding if a graph is a tree is in co-NP.*

Proof. A graph is not a tree means that it is not a forest or it is disconnected.
To show that it is not a forest, we show it has a cycle.
To show that it is disconnected, we show $\exists X \subsetneq V(E), X \neq \emptyset, \delta(X) = \emptyset$. □

Definition 2.28 (Leaf)**.** Let G be a graph. A **leaf** in G is a vertex of degree exactly 1.

A graph with one vertex is a tree, but has no leaves.

Lemma 2.29. *Let G be a graph. At least one of the following holds:*

- $deg(v_0) = 1$
- $\exists$ *a neighbour w not in P.*
- *G has a cycle.*

Theorem 2.30. *If T is a tree on ≥ 2 vertices, then T has a leaf.*

Proof. Let P be the longest path in $T = v_0 v_1 \ldots v_k$. By the theorem above, we consider

- if $deg(v_0) = 1$, then v_0 is a leaf.

- if $\exists$ a neighbour w not in P, then $Q = wv_0v_1 \cdots v_k$ is a longer path than P, contradicting to the choice of P.
- 'G has a cycle' cannot hold, since T is a tree.

$\square$

Actually if T is a tree on ≥ 2 vertices, then T has at least 2 leaves, because we can apply the same theorem on v_k as well. Therefore, v_0, v_k are leaves. More importantly, we need to show that $v_0 \neq v_k$. Since T is connected and $|V(T)| \geq 2$, T has an edge. $v_0 \neq v_k$, since $k \geq 1$.

Theorem 2.31. *If T is a tree and v is a leaf of T, then $T - v$ is a tree.*

Proof. It suffices to prove

- $T - v$ is a forest, and
- $T - v$ is connected.

Proof that $T - v$ is a forest: $T - v$ is a forest since it is a subgraph of T which has no cycles.

Proof that $T - v$ is connected: Let $x, y \in V(T - v)$. Since T is connected, $\exists$ a path $v_0v_1 \cdots v_k$ such that $v_0 = x, v_k = y$. Note $v \neq x, y$ since $x, y \in V(T - v)$. Moreover, $v \neq v_i$ for any $1 \leq i \leq k - 1$, since $\deg(v_i) \geq 2$, and $\deg(v_0) = 1$. Hence $v \notin V(P)$. This P is a path in $T - v$ from x to y. Since x, y are arbitrary, $T - v$ is connected. $\square$

Theorem 2.32. *T is a tree iff $\exists$ an ordering on n vertices $v_1, v_2, \ldots, v_n$ of $V(T)$ such that $\forall 2 \leq i \leq n$, v_i is a leaf on $G[\{v_1, \ldots, v_i\}]$, that is a graph induced by $v_1, \ldots, v_i$.*

Proof. By induction on $|V(T)|$. If $|V(T)| = 1$, this is trivial. So we may assume $|V(T)| \geq 2$. By lemma, T has a leaf, call it $v_{|V(T)|}$. By lemma, $T' = T - v_{|V(T)|}$ is a tree. Hence, by induction $\exists$ an ordering $v_1, \ldots, v_{|V(T')|}$ such that $\forall v_i$ is a leaf in $G[\{v_1, \ldots, v_i\}]$ $\forall 2 \leq i \leq |V(T)|$. But now, $v_1, v_2, \ldots, v_{|V(T')|}, v_{|V(T)|}$ is the desired ordering.

By induction on $|V(T)|$. If $|V(T)| = 1$, then T is a tree, as desired. So we may assume $|V(T)| \geq 2$. Let $T' = T - v_{|V(T)|}$. Now $v_1, \ldots, v_{|V(G)|-1}$ is an ordering of $V(T')$. By induction, T' is a tree.

Now claim that

1. T is a forest, and
2. T T is connected.

Proof that T is a forest, that is, T does not have a cycle.

Suppose not, that is, $\exists$ a cycle C in T. Since T' is a tree, meaning no cycle. It implies that $V(C) - V(T') \neq \emptyset$. But $V(T) - V(T') = \{v_{|V(T)|}\}$ and hence $v_{|V(T)|} \in V(C)$. Yet then $\deg(v_{|V(T)|}) \geq 2$ contradicting that $\deg(v_{|V(T)|}) = 1$.

Proof that T is connected.

Let $w \in N(v_{|V(T)|})$. Since T' is a tree, $V(T')$ are all in one component of T, yet since $v_{|V(T)|} \in E(T)$, we have that w is in the same component as $v_{|V(T)|}$. Hence all of $V(T)$ is in one component, and T is connected. □

Theorem 2.33. *Deciding if a graph is a tree is in NP.*

Proof. The prover provides such ordering, and the verifer checks each $\deg v_i = 1$. □

Lemma 2.34. *If T is a tree, then $|E(T)| = |V(T)| - 1$.*

Proof. Proof by induction on $|V(T)|$. If $|V(T)| = 1$, then $|E(T)| = 0 = 1 - 1 = (V(T) - 1)$, as desired. We may assume that $|V(T)| \geq 2$. By lemma, T has a leaf v. By other lemma, $T' = T - v$ is a tree. By induction, $|E(T')| = |V(T')| - 1$. Note that $|V(T)| = |V(T')| + 1$, since $V(T) - V(T') = \{v\}$. Since v is a leaf, $N(v) = \{w\}$, hence $E(T) - E(T') = \{vw\}$, $|E(T)| = |E(T')| + 1$. Hence, $|E(T)| = |E(T')| + 1 = |V(T')| - 1 + 1 = |V(T)| - 1$. □

Corollary 2.35. *If T is a tree, then*

$$\sum_{v \in V(T)} \deg(v) = 2|E(T)| = 2|V(T)| - 2$$

Proof. It follows directly from the Handshaking Lemma, and the theorem above. □

2.5. Spanning Subgraph

Definition 2.36. A subgraph H of a graph G is spanning in G if $V(H) = V(G)$.

A nice object to study is a spanning tree of a graph, that is a tree and spanning. Question: When does a graph have a spanning tree?

Lemma 2.37. *If a graph G has a spanning tree, then G is connected.*

Proof. $\forall x, y \in V(G)$, $x, y \in V(T)$ because T is a spanning tree, but then $\exists$ a path P in T from x to y. Since T is connected, P is a path in G. Hence G is connected. □

The converse is also true!

Lemma 2.38. *If G is connected, then G has a spanning tree.*

Proof. Idea: If not a tree, then $\exists$ a cycle, delete edge in C and iterate.

Proof by minimality:

Let H be a spanning, connected subgraph, such that $|E(H)|$ is minimized. Note that such H exists because

1. it can be minimized because the graph is finite
2. G itself is a connected subgraph of G. Hence set of such subgraph is non-empty.

Claim: H has no cycle.

Suppose not, that is, $\exists$ a cycle C in H. Let $e \in E(C)$, $H' = H - e$. Note that $V(H') = V(H) = V(G)$, and hence H' is a spanning subgraph of G, with smaller number of edges, contradicting the choice of H.

Subclaim: H' is connected. Let $x, y \in V(H')$. Since H is connected, $\exists$ a path H from x to y in H. If $e \notin E(P)$, then P is a path, as desired. So we may assume $e \in E(P)$, but then $P = P_1 e P_2$. But now $P_1, C - e, P_2$, then is a walk from x to y in H. But then $\exists$ a path from x to y in H' so H' is connected.

□

2.6. Bridges

Definition 2.39 (Bridges)**.** An edge e in a graph G is a bridge if $G - e$ has strictly more components than G.

Question: Deciding if an edge is a bridge is in NP? Co-NP? Or P?

It is in NP, we only need partition into components and for each component, we need the paths or the spanning trees. The verifier verifies that the cuts are empty, i.e. components connected, paths are valid paths to show that the number of components in NP. Put simply, deciding the number of components is in NP.

How about Co-NP? Is it easy to see that an edge is not a bridge?

Theorem 2.40. *e is a bridge of a graph G iff e is not in a cycle of G.*

Proof. Show that if e is a bridge, then e is not in a cycle.

Suppose not, that is, e is a bridge and still in a cycle C of G.

Since e is a bridge, by definition of bridges, $G - e$ has more components than G.

Let C_1 be the component of $G - e$ but not in G. C_1 exists since $G - e$ has more components.

Since C_1 is a component of $G - e$, we have that

$$\delta_{G-e}(C_1) = \emptyset.$$

However, since C_1 is not a component of G, it follows that

$$\delta_G(C_1) = \{e\}. \text{ (Why?)}$$

Let x, y be the ends of e such that, WLOG, $x \in C_1$ and $y \notin C_1$.

It turns out that $C - e$ is a path P from x to y in $G - e$. (Note C is that cycle we assumed at the beginning!) Hence x and y are in the same component of $G - e$. A contradiction.

Show that if e is not in a cycle, then e is a bridge.

Let's show contrapositive, that is, if e is not a bridge, then e is in a cycle.

Let $e = xy$. Since e is not a bridge, $G - e$ has the same number of components as G.

Let C be the component of G containing e.

From the above, C is a component of $G - e$. Since $x, y \in C$, $\exists$ a path P from x to y in $G - e$.

$P + e$ is a cycle containing the edge e, as desired. □

Theorem 2.41. *Deciding if e is a bridge is in co-NP.*

Proof. The prover gives the verifier a cycle containing e and the verifier verifies it. □

Lemma 2.42. *If G is a forest, then every edge is a bridge.*

Actually, the converse is also true.

2.7. $CO - NP$ Characterization of Bipartition

Recall that we have seen the following theorem, but this time we prove it!

Theorem 2.43. *A graph G is bipartite, iff G has no odd cycles.*

Proof. Show that if G is bipartite, then G has no odd cycles.

Suppose not, that is, $\exists$ a cycle $C = v_0 v_1 \dots v_k v_0$ of G such that k is even. (Note that for an odd cycle, we need k to be even instead of odd.)

Suppose G is bipartite, that is, $\exists$ a bipartition A, B such that $\forall e = xy, x \in A, y \in B$.

More generally, we find that, $v_i \in A$ if i is even; $v_i \in B$ if i is odd.

By considering the edges $v_i v_{i+1} \forall i$, then $v_0 \in A$ and $v_k \in A$. Thus, $v_0 v_k$ is an edge with both ends in A.

Show that if G has no odd cycles, then G is bipartite.

By induction on $|V(G)|$.

If G is disconnected, by induction, each component C is bipartite.

That is $\exists$ a bipartition A_i, B_i of C_i such that $\forall e \in E(C_i)$ has one end in A_i and the other in B_i. Then, $A = A_1 \cup A_2 \cup \cdots$ and $B = B_1 \cup B_2 \cup \cdots$.

If G is connected, then G has a spanning tree.

Let $v_i \in V(T)$. Partition $V(T)$ into

$$S_i := \{v \in V(T) : D(v, v_0) = i\}.$$

Claim: $\forall e = xy \in E(G)$, e has one end in A, other in B

Suppose not, that is, x and y are either both in A, or both in B.

We may assume, WLOG, $x, y \in A$. (The case B is symmetric)

Since $x, y \in V(G)$ and T is connected, $\exists$ a path $P = v_0 \dots v_k$ in T such that $v_0 = x, v_k = y$.

However, every edge in T has one end in A and the other in B.

Since $v_0 v_1 \in E(T)$, $v_0 \in A \implies v_1 \in B$, inductively, we find that since $v_i v_{i+1} \in E(T), \forall 0 \leq i \leq k-1$, it follows that $v_i \in A$ if i even, $v_i \in B$ if i odd.

Then $C = v_0 v_1 \dots v_k v_0$ ia an odd cycle in G since $y = v_k \in A$. Hence k is even. Hence the cycle is odd. A contradiction.

□

2.8. Minimum Weight Spanning Trees

There are two nice lemmas about how to change a spanning tree to a similar but a different spanning tree.

Theorem 2.44 ('Add and delete edge in cycle'). *Let G be a connected graph, and T be its spanning tree.*

If $e \in E(G) - E(T)$, then $\exists$ a unique cycle C_e in $T + e$.

Moreover, $\forall f \in E(C_e)$, $T + e - f$ is a spanning tree of G.

Proof. First statement:

Let $e = xy$.

Since T is connected, $\exists$ a path $P = v_0 \dots v_k$ such that $v_0 = x, v_k = y$ in T. Moreover, that path is unique. (see course notes)

Therefore, $C = v_0 \dots v_k v_0$ is a cycle in $T + e$.

Let C' be any cycle in $T + e$. Then, C' contains e, since T is acyclic.

However, $C' - e$ is a path P' between x and y. Since that path is unique, we find that $P' = P$. Hence, $C' = C$.

This proves the first part of the statement.

Second statement:

Let $f \in E(C_e)$, and $T' = T + e - f$. T' is spanning in G, since $V(T') = V(T) = V(G)$.

Claim 1: T' is acyclic.

Suppose not, that is, $\exists$ a cycle C'' in T', since $T' = T + e - f \subset T + e$. We have that C'' is a cycle in $T + e$.

By previous statement, $C'' = C_e$, since C_e is the unique cycle in $T + e$.

However, $f \in E(C_e)$ and $f \notin E(T')$. Hence a contradiction.

Claim 2: T' is connected.

Note $T + e$ is connected, and thus has one component.

Since $f \in C_e$, a cycle $T + e$. By theorem, f is not a bridge of $T + e$.

Hence $T + e - f$ has the same number of components as $T + e$. Thus $T' = T + e - f$ has 1 component. Thus T' is a spanning tree of G.

$\square$

Theorem 2.45 ('Delete edge, add edge in cut'). *Let G be a graph and T be a spanning tree of G. If $e \in E(T)$, then $\exists!$(up to complement)$X \subset V(T)$ such that $e \in \delta_T(X), \forall f \in \delta_G(X), T - e + f$ is a spanning tree of G.*

Let G be a connected graph. Let w be a non-negative integer weight function on the edges of G.

Problem. Find a spanning tree of G whose total weight is minimized. This tree we are trying to find is called the minimum weight spanning tree, denoted as MST.

Question. How do we find MST efficiently?

Answer. This is not a decision problem such as **is the MST having weight at most** k**?** but an optimization problem. It usually asks for an instance. Nevertheless, the answer is YES.

Theorem 2.46 (Prim's Algorithm)**.** *Initiate a tree T by starting with an arbitrary vertex v.*

WHILE *T is not a spanning tree.*

- *Choose an edge e in $\delta_G(V(T))$ of minimum weight in the cuts.*
- *Let $T := T + e'$.*

RETURN *T.*

Proof. It is clear that Prim's outputs a spanning tree.

Proof that the output from Prim's is the MST.

Suppose not, that is, there exists a spanning tree with weight **strictly smaller** than the weight of the tree outputted by Prim's. Let's call the tree outputted by Prim's T.

Claim. $\forall 1 \leq i \leq |V(G)|, \exists$ a MST T_i such that $v_1, v_2, \ldots, v_i \in V(T_i)$ and all edges from $E(T)$, where $v_1, v_2, \ldots, v_{|V(G)|}$ is the ordering from Prim's.

Proof of Claim. By induction on i.

- When $i = 1$, there are no edges to agree on, so any MST agrees as desired.
- We may assume that $\exists$ a T_i such that $e_1, e_2, \ldots, e_{i-1} \in E(T_i) \cap E(T)$. We have to prove that $\exists\, T_{i+1}$ such that $e_1, \ldots, e_i \in E(T_{i+1}) \cap E(T)$. If $e_i \in E(T_i)$, then $T_{i+1} = T_i$, which satisfies the claim.
- We may assume that $e_i \notin E(T_i)$. Set $T_{i+1} = T_i - e + e_i$. Note $w(T_{i+1}) \leq w(T_i)$ and hence T_{i+1} is MST. Yet $e_1, \ldots, e_i \in E(T_{i+1})$ as desired.

However the claim with $i = |V(G)|$ says $\exists$ an MST $T_{|V(G)|} = T$, i.e., T is an MST.

□

Chapter 3

Planar Graph

3.1. Planarity

Definition 3.1 (Arc)**.** An arc is a piece-wise linear segment with a finite number of pieces.

Definition 3.2 (Plane Embedding)**.** A plane embedding of a graph G is a map from $V(G)$ to distinct points in $\mathbb{R}^2$ and from $E(G)$ to internally disjoint arcs between the points corresponding to the ends of the edge.

Definition 3.3 (Plane Graph)**.** A plane graph is a graph that has a plane embedding.

Definition 3.4 (Planar Graph)**.** A planar graph is a graph that has a plane embedding. (No particular one specified.)

Definition 3.5 (Face)**.** If G is a plane graph, then a face of G is a connected region of $\mathbb{R}^2 - (V(G) \cup E(G))$.

Definition 3.6 (Boundary of Face)**.** The boundary of a face is the boundary topologically of the region. (i.e. vertices incident with the face)

The boundaries naturally divide into connected components. Moreover, each component of the boundary has a **boundary walk**.

A **boundary walk** is not necessarily a cycle, because an edge or vertex may appear twice in a boundary walk. A face may have more than one boundary components.

Definition 3.7 (Degree of Face)**.** The degree of a face is the sum of the boundary walks of each boundary components of the face.

Hence, an edge appearing twice in a walk counts twice in the degree.

Lemma 3.8 (Faceshaking Lemma)**.** *If G is a planar graph, then*

$$2|E(G)| = \sum_{f \in F(G)} \deg(f)$$

where $F(G)$ is the set of faces and $\deg(f)$ *denotes the degree of face f.*

Definition 3.9 (Dual)**.** If G is a plane graph, then the dual of G, denoted as G^*, is the **multi-graph** where $V(G^*) = F(G)$ and $E(G^*)$ is defined by: $\forall e \in E(G)$, putting an edge e^* between the two faces incident with e. (i.e. its left and right face)

We do this, even if the two faces are infact incident with the same face, in which case e^* is a loop.

Example 3.10. The dual of K_4 is itself.

Lemma 3.11. *If G is a planar graph, G^* is a planar graph and, in fact, a plane graph if we force e and e^* to intersect, $\forall e$.*

Theorem 3.12. *Let G be a plane graph. An edge e of G is incident with only one face iff e^* is a loop, iff e is a bridge.*

Proof. The first "iff" follows by definition. The second "iff" is proven as follows:

'$\implies$': we prove the contrapositive. Assume e is not a bridge. By **Bridge-Cycle Lemma**, e is ain a cycle C.

Note $\mathbb{R}^2 - C$ is disconnected and the left and right faces of e live in different regions of $\mathbb{R} - C$. The face for general curves are called the **Jordan Curve Theorem**.

Hence e is incident with two faces as desired.

'$\impliedby$': If e^* is a loop, then $\mathbb{R}^2 - e^*$ has two regions. Let $e = xy$. The component containing x will live in one region and the component containing y will live in the other.

□

Theorem 3.13.

$$F(G^*) \leftrightarrow V(G)$$

$$E(G^*) \leftrightarrow E(G)$$

$$V(G^*) \leftrightarrow F(G)$$

$$(G^*)^* \leftrightarrow G$$

3.2. Euler's Formula

Theorem 3.14 (Euler's Formula)**.** *If G is a planar graph, then*

$$|V(G)| - |E(G)| + |F(G)| = 1 + c(G)$$

where c is the number of components in G.

Proof. By induction on $|E(G)|$.

Base Case: $|E(G)| = 0$. Here $|F(G)| = 1$. Yet, $c(G) = |V(G)|$. Thus,

$$\begin{aligned} |V(G)| - |E(G)| + |F(G)| &= |V(G)| - 0 + 1 \\ &= 1 + |V(G)| \\ &= 1 + c(G) \end{aligned}$$

Inductive Step: We may assume $|E(G)| \neq 0$. Let $e \in E(G)$. Let $G' = G - e$. G' is planar graph. Moreover, check the condition for induction:

$$|E(G')| = |E(G)| - 1 < |E(G)|$$

By induction,

$$|V(G')| - |E(G')| + |F(G')| = 1 + c(G')$$

Note that $|V(G)| = |V(G')|$. Now, discuss whether e is a bridge or not.

- If e is a bridge, by definition of bridge,

  $$c(G') = c(G - e) = 1 + c(G)$$

  By **Bridge-Face Theorem**, e is incident with exactly one face,

  $$|F(G')| = |F(G)|$$

  $$\begin{aligned} & |V(G')| - |E(G')| + |F(G')| = 1 + c(G') \\ \implies & |V(G)| - (|E(G)| - 1) + |F(G)| = 1 + (1 + c(G)) \\ \implies & |V(G)| - |E(G)| + 1 + |F(G)| = 1 + 1 + c(G) \\ \implies & |V(G)| - |E(G)| + |F(G)| = 1 + c(G) \end{aligned}$$

- If e is not a bridge, $c(G) = c(G')$. By **Bridge-Face Theorem**, e is incident with 2 distinct faces, f_1 and f_2. In G', f_1 and f_2 merge into one face. Thus, $|F(G')| = |F(G)| - 1$.

  $$\begin{aligned} & |V(G')| - |E(G')| + |F(G')| = 1 + c(G') \\ \implies & |V(G)| - (|E(G) - 1) + (|F(G)| - 1) = 1 + c(G') \\ \implies & |V(G)| - |E(G)| + |F(G)| = 1 + c(G) \end{aligned}$$

□

Corollary 3.15. *If G is a connected planar graph, then*

$$|V(G)| - |E(G)| + |F(G)| = 2$$

Theorem 3.16 (Edge Number Upperbound)**.** *If G is a plane graph and* $|V(G)| \geq 3$*, then*

$$|E(G)| \leq 3|V(G)| - 6$$

Proof. Now we only consider G connected. **Since** $|V(G)| \geq 3$**,** $|E(G)| \geq 2$**.** By Euler's formula

$$|V(G)| - |E(G)| + |F(G)| = 2$$

By Faceshaking Lemma,

$$2|E(G)| = \sum_{f \in F(G)} \deg(f)$$

Claim 1: If $|V(G)| \geq 3$ and G is connected, then $\forall f(G), \deg(f) \geq 3$.

Proof. If the boundary walk of f contains a cycle C,

$$\deg(f) \geq E(C) \geq 3, \forall f \in F(G)$$

If the boundary walk contains no cycle, the graph G is a tree. Here, $|F(G)| = 1$. By Euler's Formula, since $|V(G)| \geq 3$

$$|E(G)| = |V(G)| + |F(G)| = |V(G)| - 1 \geq 2$$

By Faceshaking Lemma,

$$\deg(f) = 2|E(G)| \geq 4$$

Combining two cases where $\deg(f) \geq 4$ and $\deg(f) \geq 3$, we can safely conclude that $\deg(f) \geq 3$.

□

By Claim 1, $2|E(G)| = \sum_{f \in F(G)} \deg(f) \geq 3|F(G)| \implies |F(G)| \leq \frac{2}{3}|E(G)|$.
By Euler's Formula,

$$\begin{aligned} |V(G)| - |E(G)| + |F(G)| = 2 \implies |E(G)| &= |V(G)| + |F(G)| - 2 \\ &\geq 3 + |F(G)| - 2 \\ &= |F(G)| + 1 \end{aligned}$$

$$2 = |V(G)| - |E(G)| + |F(G)| \leq |V(G)| - |E(G)| + \frac{2}{3}|E(G)|$$

$$|V(G)| - \frac{1}{3} \geq 2 \implies |E(G)| \leq 3|V(G)| - 6$$

□

Theorem 3.17 (Existence of Degree at most 5). *If G is a plane graph and $|V(G)| \geq 3$, the average degree of vertices < 6. (At least one vertex has degree < 6)*

Proof. If G is a plane graph and $|V(G)| \geq 3$, then

$$2|E(G)| \leq 6|V(G)| - 12$$

Hence, by **Handshaking Lemma**, the average degree of vertices < 6. (For all plane G)

□

Theorem 3.18 (Tighter Edge Number Upperbound). *If G is a plane graph, and every face has degree ≥ 4, then*

$$|E(G)| \leq 2|V(G)| - 4$$

Proof. By Euler's Formula, $|V(G)| - |E(G)| + |F(G)| \geq 2$. By Faceshaking Lemma,$2|E(G)| = \sum_{f \in F(G)} \deg(f)$. Since every face has degree ≥ 4, then $\sum_{f \in F(G)} \deg(f) \geq 4|F(G)|$. Hence $|F(G)| \leq \frac{|E(G)|}{2}$. Plug into Euler's Formula, we get $2 \leq |V(G)| - |E(G)| + |F(G)| \leq |V(G)| - |E(G)| + \frac{|E(G)|}{2} = |V(G)| - \frac{|E(G)|}{2}$. Then, this implies $\frac{|E(G)|}{2} \leq |V(G)| - 2$. Finally, $|E(G)| \leq 2|V(G)| - 4$. □

Theorem 3.19 (Edge Bound for Bipartite Graph). *If G is a bipartite planar graph and $|V(G)| \geq 3$, then*

$$|E(G)| \leq 2|V(G)| - 4$$

3.3. Kuratowski's Theorem

Big Question: which graphs are not planar?

Example 3.20. K_5 is not planar. (Indeed, K_n for any $n \geq 5$.)

Proof. $|V(K_5)| = 5, |E(K_5)| = 10$. If K_5 is planar, then by **Edge Number Upperbound**, $|E(K_5)| \leq 3|V(K_5)| - 6 = 9$, which contradicts to $|E(K_5)| = 10$. □

Definition 3.21 ($K_{m,n}$ complete bipartite graph). G has a bipartition A, B, where $|A| = m, |B| = n$, and all edges are in between.

Example 3.22. $K_{3,3}$ is not planar. ($K_{m,n}$ for any $m, n \geq 3$)

Proof. If $K_{3,3}$ is planar and $K_{3,3}$ is bipartite with $|K_{3,3}| \geq 3$, $|E(K_{3,3})| \leq 2|V(K_{3,3})| - 4 = 8$. □

Lemma 3.23. *If G is a planar graph, and H is a subgraph of G, then H is planar.*

Proof. Take the plane embedding of G. Delete down to H to get a plane embedding of H. □

K_5 and $K_{3,3}$ are minimally non-planar in that $\forall$ proper subgraph of each is planar.

Actually there are infinitely many minimal non-planar graphs.

Definition 3.24 (Subdivide)**.** Let G be a graph and $e \in E(G) = xy$. We can create a new graph G' by "sub-dividing" e as follows:

$$V(G') = V(G) \cup \{z\}$$

a new vertex not in $|V(G)|$.

$$E(G') = (E(G) - \{e\}) \cup \{xy, yz\}$$

We can repeatedly apply this operation to subdivide an edge into a path. We can also do this to different edges as well. Replacing edges of original with new edge-disjoint paths.

Definition 3.25 (Subdivision)**.** Let G be a graph. We say H is a subdivision of H if H can be obtained from G by the use of the sub-division algorithm possibly repeatedly.

Lemma 3.26. *Let H be a subdivision on G. Then G is a planar iff H is planar.*

Proof. "$\implies$": We assume G is planar. It suffices to prove that subdividing edge preserves planarity. Pick a point z on the arc e and add z there.

"$\impliedby$": Assume it is planar. It suffices to prove that reversing one subdivision operation preserves planarity. i.e. If we subdivided $e = xy$ into xz, yz, show reverse if planar. Let $G - e + \{xz, yz\}$ be embedded in the plane. Let A_1 be the arc for xz and A_2 be an arc for zy. Now we embed G as for H, except we don't embed z or xz, yz, instead we embed e into $A_1 A_2$ which is still an arc. □

Note the negation: G is non-planar iff all subdivisions of G are non-planar. Hence every subdivision of K_5 is non-planar. Remark: actually each is minimal.

Theorem 3.27 (Kuratowski's Theorem)**.** *A graph G is planar if and only if G does not contain a subgraph that is a subdivision of* K_5*, or a subdivision of* $K_{3,3}$*.*

Question: is it easy to verify that a graph is (not)planar?
Question: is deciding if a graph is planar in NP**? co-**NP**?** P**?** For all: Yes.
To prove an embedding, it will suffice to prove a combinatorial embedding, which is just all of the boundaries of each face.

Theorem 3.28. *Deciding if a graph is planar is in* NP*.*

Proof. The prover gives a combinatorial embedding. The verifier verifies that every edge appears exactly twice and walks are walks and that Euler's holds. □

Theorem 3.29. *Deciding if a graph is planar is in co-*NP*.*

Proof. The prover gives a $K_{3,3}$ or K_5 subdivision. Each gurantees to exist by Kuratowski's Theorem. The verifier checks subdivision. □

Deciding if a graph is planar is in P, namely $\exists$ an algorithm $O(|V(G)|^2)$ to do.
How to find $K_{3,3}$**-subdivision?**

1. Find a cycle and two connected but internally disjoint crossing paths.
2. Find $K_{3,3}$ with three pairwise disjoint paths.

3.4. Graph Colouring

What is a generalization of bipartite graph?
We have tripartite, i.e. no edge has both ends in the same parts A, B, C. In general, $k-$partite is the same as $k-$parts.

Definition 3.30 (Coloring)**.** A coloring of a graph G assigns a **color** to each vertex such that adjacent vertices do not receive the same color.

But what colors? Any things you want! So we usually use (positive) integers.

Definition 3.31 ($k-$coloring)**.** A $k-$coloring of a graph G is a coloring that was at most $k-$colors.
(usually here we use $\{1, 2, \ldots, 10\}$.
More formally, a $k-$ coloring of G is a map $\phi : V(G) \to \{1, \ldots, k\}$ such that $\forall e = uv \in E(G), \phi(u) \neq \phi(v)$.

What is the smallest number of colors needed to color a given graph G**?**

Definition 3.32 (Chromatic Number)**.** The chromatic number of a graph G is the minimum k such that G has a k-coloring, denoted $\chi(G)$.

Facts

- $\chi(K_n) = n$, because we need at least n and n suffices.
- $\chi(P_n) = \begin{cases} 2 & \text{if } n \geq 2 \\ 1 & \text{if } n = 1 \end{cases}$
- $\chi(C_n) = \begin{cases} 2 & \text{if even} \\ 3 & \text{if odd} \end{cases}$

Lemma 3.33.

$$\chi(G) \leq 2 \iff G \textit{ is bipartite.}$$

$$\chi(G) \leq k \iff G \textit{ is } k-\textit{partite.}$$

Lemma 3.34. *If $H \subset G$, then $\chi(H) \leq \chi(G)$.*

Proof. Let $k = \chi(G)$. By definition, G has a $k-$coloring ϕ. But then $\phi|V(H)$ is a $k-$coloring of H. ☐

Definition 3.35 ($k-$critical)**.** A graph G is $k-$critical if $\chi(G) = k$ and $H \subsetneq G$, $\chi(H) < k$.

Note

- G is $1-$critical iff $G = K_1$.
- G is $2-$critical iff $G = K_2$.
- G is $3-$critical iff G is an odd cycle. (Infinitely many possible odd cycles can be)
- G is $4-$critical iff G is $4-$critical. (No structure!)

Question: Is it easy to verify that $\chi(G) \leq k$, i.e. has a $k-$coloring? Or not? Or to decide between? In other words, is deciding G is $k-$coloring in NP? Co-NP? P?

Theorem 3.36. *Deciding G is $k-$coloring is in NP.*

Proof. The prover gives the coloring $\phi : V(G) \to \{1, \ldots, k\}$. The verifier checks $\forall e = uv \in E(G)$ that $\phi(u) \neq \phi(v)$. This is $O(|E(G)|)$. ☐

Theorem 3.37. *Deciding G is k−coloring is in co-NP if* $k \leq 2$.

Proof. For $k = 0$, the prover gives a vertex. For $k = 1$. the prover gives an edge. For $k = 2$, the prover gives an odd cycle. □

Actually if for any $k \geq 3$**, this problem was in** $co - NP$**, then every** NP **problem is in co-**NP**, i.e.** $P = co - NP$**. (This was conjectured not to be true.)**

Theorem 3.38. $\forall k \geq 3$, *k−coloring decision problem is NP−complete, meaning that if this problem is in P, then every NP problem is in P.*

Bounds on the chromatic number

Definition 3.39 (Clique Number)**.** The clique number of a graph G, denoted $\omega(G)$, is the size of the largest clique. (Clique, i.e. Complete Graph)

Example 3.40.

- $\omega(K_n) = n$
- $\omega(P_n) = \begin{cases} 2 & \text{if } n \geq 2 \\ 1 & \text{if } n = 1 \end{cases}$

Theorem 3.41. *Let G be a graph,*

$$\chi(G) \geq \omega(G)$$

Proof. Let $\omega(G) = k$. By definition, G contains a subgraph $H = K_k$. By the fact that $\chi(K_k) = k$. By proposition, since $H \subsetneq G$, $\chi(G) \geq \chi(H) = k$. □

Definition 3.42 (Maximum Degree of vertex in G)**.** The maximum degree of G, denoted $\Delta(G)$, is defined by

$$\Delta(G) := \max_{v \in V(G)} \deg(v)$$

Theorem 3.43. *Let G be a graph.*

$$\chi(G) \leq \Delta(G) + 1$$

Proof. We use induction on the number of vertices of G, denoted n, to prove the result.

Base Case: If $n = 1$, it is clear that we cannot have any vertices on a graph with only one vertex, so the maximum degree of vertex of this graph is 0, i.e. $\Delta = 0$. We need exactly 1 colour to colour this only vertex, i.e. $\chi(G) = 1$. $\chi(G) = 1 \leq 1 = 0 + 1 = \Delta(G) + 1$. The base case holds.

Inductive Hypothesis: Assume the result holds for every graph with $n-1$ vertices for some $n \geq 2$.

Inductive Conclusion: Let G be a graph on n vertices. Let $v \in V(G)$. Now, $G - v$ is a graph on $n-1$ vertices. By Inductive Hypothesis,

$$\chi(G-v) \leq \Delta(G-v) + 1$$

This means that the graph $G - v$ can be coloured by at most $\Delta(G-v)+1$ colours.

Note that $\deg_G(n) \leq \delta(G)$, by definition. The neighbours of v can be coloured up $\leq \Delta(G)$ colours.

We discuss two cases $\Delta(G) = \Delta(G-v)$ and $\Delta(G) < \Delta(G-v)$.

Consider $\Delta(G) = \Delta(G-v)$.

$$\chi(G-v) \leq \Delta(G-v) + 1 \implies \chi(G-v) \leq \Delta(G) + 1$$

This means $G - v$ can be coloured by at most $\Delta(G)+1$ colours. The neighbours of v can use the number of colours up $\leq \Delta(G)$. Therefore, there is at least 1 colour of $\Delta(G)+1$ colours unused by any neighbours of v. Therefore, for the graph $G = (G-v)+v$, we give that unused colour to v. This gives a $\Delta(G)+1$-colouring of G. Therefore, $\chi(G) \leq \Delta(G)+1$.

Consider $\Delta(G) \neq \Delta(G-v)$. Then, it tells we must have $\Delta(G-v) < \Delta(G)$. $G = (G-v)+v$ requires one more colour than $G-v$. Hence, this gives a $\Delta(G-v)+1+1$-colouring. Since $\Delta(G-v) < \Delta(G)$, $\Delta(G-v) \leq \Delta(G)-1$. $\Delta(G-v)+1+1 = \Delta(G-v)+2 \leq \Delta(G)-1+2 = \Delta(G)+1$. This gives a $\Delta(G)+1$-colouring of G. Therefore, $\chi(G) \leq \Delta(G)+1$.

□

Corollary 3.44. *Let G be a graph.*

$$\omega(G) \leq \chi(G) \leq \Delta(G)+1$$

Definition 3.45 ($d-$degenerate)**.** A graph G is $d-$**degenerate** if in every subgraph H of G, $\exists v \in V(H)$, such that $\deg_H(v) \leq d$.

Theorem 3.46.

- *G is $\delta(G)$-degenerate.*
- *G is 0-degenerate iff $|E(G)| = 0$.*
- *G is 1-degenerate iff G is a forest.*

Remark 3.47. Note that the definition of $d-$generate is of a co-NP characterization. The following lemma will give a P characterization of degeneracy.

Theorem 3.48 (NP Characterization of Degeneracy). *G is $d-$degenerate $\iff$ $\exists$ an ordering $v_1, \ldots, v_n$ of $V(G)$ such that $\forall 1 \le i \le n, \deg(v_i) \le d \in G[\{v_1, \ldots, v_i\}]$.*

Proof. " $\implies$ ": Assume G is $d-$degenerate. By induction on $|V(G)|$. If $|V(G)| = 1$, v_1 is the desired ordering. We may suppose that $|V(G)| \ge 2$. Since G is $d-$degenerate, $\exists v \in V(G)$ such that $\deg_G(v) \le d$. Let $G' = G - v$. Note that G' is $d-$degenerate because G is. By induction, $\exists$ an ordering $v_1, \ldots, v_{n-1}$ of $V(G')$, such that $\forall 1 \le i \le n-1, \deg(v_i) \le d \in G[\{v_1, \ldots, v_i\}]$. $v_1, \ldots, v_{n-1}, v$ is a desired ordering of $V(G)$ since $\forall 1 \le i \le n, \deg(v_i) \le d \in G[\{v_1, \ldots, v_i\}]$.

" $\impliedby$ ": Assume we have an ordering $v_1, \ldots, v_n$ of $V(G)$ such that $\forall 1 \le i \le n, \deg(v_i) \le d \in G[\{v_1, \ldots, v_i\}]$. Suppose for contradiction that G is not $d-$degenerate, i.e., $\exists$ a subgraph H of G such that $\forall v \in V(H)$ we have $\deg_H(v) > d$. Let $j := \max\{i : v_i \in V(H)\}$. By the specification of such ordering, we must have $\deg_{G[\{v_1,\ldots,v_j\}]}(v_j) \le d$. Since $V(H) \subset \{v_1, \ldots, v_j\}$, we find that $\deg_H(v_j) \le d$. A contradiction. $\square$

Theorem 3.49. *Deciding $d-$degeneracy is in co$-NP$.*

Proof. The prover gives the verifier a subgraph of H such that $\deg_H(v) > d, \forall v \in V(G)$. $\square$

Theorem 3.50. *Deciding $d-$degeneracy is in NP.*

Proof. The prover gives the verifier an ordering $v_1, \ldots, v_n$ of $V(G)$ such that $\forall 1 \le i \le n, \deg(v_i) \le d \in G[\{v_1, \ldots, v_i\}]$. $\square$

Theorem 3.51. *If G is $d-$degenerate, then*

$$\chi(G) \le d + 1$$

Proof. By induction on $|V(G)|$. If $|V(G)| = 1$, then $\chi(G) \le 1 \le d + 1$. We may assume $|V(G)| \ge 2$. Since G is $d-$generate, $\exists v \in V(G)$ such that $\deg_G(v) \le d$. Let $G' = G - v$. By induction, $\exists$ a $(d+1)$-coloring ϕ of G'. Let $A = [d+1] - \{\phi(u) : u \in N(v)\}$. Since $N(v)$ is at most d. We have that $A \ne \emptyset$. Let $\phi(v) \in A$. Now, ϕ is a $(d+1)$-coloring ϕ of G. $\square$

3.5. Coloring Planar Graph

Question. What is the maximum of $\chi(G)$ over all planar graphs of G?

Answer. Four Color Conjecture (around 1852)

Theorem 3.52 (Six Color Theorem). *If G is planar, $\chi(G) \le 6$.*

Proof. By **the corollary of Euler's Formula**, $E(G) \leq 3|V(G)| - 6$. We have that every planar graph has a vertex with degree ≤ 5. Since a subgraph of a planar graph is sill planar. It follows that the planar graph is 5-degenerate. By **theorem above**, $\chi(G) \leq 5 + 1 = 6$.

□

We would like to prove the **Five Color Theorem**, but degeneracy would not be enough in a sense that not every planar graph is $4-$degenerate.

Example 3.53. $\exists$ a $5-$regular planar graph called **Icosahedron**.

Theorem 3.54 (Five Color Theorem)**.** *If G is planar, then $\chi(G) \leq 5$.*

Proof. By induction on $|V(G)|$. If $|V(G)| = 1$, $\chi(G) = 1$. The base case holds.

We may assume $|V(G)| \geq 2$. Since G is planar, $\exists\, v \in V(G), \deg_G(v) \leq 5$.

Case 1 : $\deg_G(v) \leq 4$

Let $G' = G - v$. By induction, G' has a 5-coloring.

Extend ϕ to v by letting $\phi(v) \in ([5] - \{\phi(u) : u \in N(v)\})$

Case 2 : $\deg_G(v) = 5$

Since G is planar, $\nexists K_5$ subgraph of G. Hence, $\exists u \neq w \in N(v)$ such that $uw \notin E(G)$. **(Why? Otherwise, there would be a K_5 as a subgraph!)**

Let G' be obtained as follows (It has a name called **contraction**.):

Let x be a new vertex.

- $V(G') = (V(G) - \{u, v, w\}) \cup \{x\}$
- $E(G') = E(G[V(G) - \{u, v, w\}]) \cup \{xy : y \in N(u) \cup N(w)\}$

Claim: G' is planar.

By induction, G' has a $5-$coloring ϕ. Let $\phi(u) = \phi(x)$ and $\phi(w) = \phi(x)$. Then let $\phi(v) \in [5] - \{\phi(z) : z \in N(v)\}$. This will be a 5-coloring of G since $([5] - \{\phi(z) : z \in N(v)\}) \neq \emptyset$ and $\phi(u) = \phi(w)$.

But why can you be sure it is really a $5-$coloring of G?

Claim: ϕ is a $5-$coloring of G.

Proof. Suppose not, that is, $\exists\, a, b \in E(G)$ such that $\phi(a) = \phi(b)$. If $ab \in V(G) - \{u, v, w\}$, then ϕ is not a $5-$coloring to G', contrary to assumption. W.l.o.g., assume $a \in \{u, v, w\}$.

Case 1: $a \in \{u, w\}$ So we may assume, w.l.o.g., that $a = u$. If $b = v$, then $\phi(a) \neq \phi(b)$ by definition of $phi(v)$. Otherwise, $b \in N(u) - \{v\}$, but then $\phi(x) \neq \phi(b)$, $\phi(a) = \phi(u) = \phi(x) \neq \phi(b)$.

Case 2: $a = v$, by definition of $\phi(v)$, $\phi(a) \neq \phi(b)$. □

□

Chapter 4

Matchings

4.1. Matching

Definition 4.1 (Matching)**.** A **matching** M in a graph G is a subset of $E(G)$ such that $\forall e, f \in E(M)$, e and f are not incident with the same vertex.

(Silly) Question. Does every graph have a matching?
Answer. Yes! Empty matching.
Question. For a given graph G, what is the size of a maximum matching of G?

Definition 4.2 (Matching Number)**.** The **matching number** of a graph G, denoted $\nu(G)$, is the size of a largest matching in G.

Definition 4.3 (Perfect Matching)**.** A matching M of G is a **perfect matching** of G if $|M| = \dfrac{|V(G)|}{2}$.

Question. Does every graph have a perfect matching?
Answer.

- If $|V(G)|$ is odd, **no chance!**
- If $|E(G)| = 0$
- Stars
- $|E(G)| < \dfrac{|V(G)|}{2}$
- If G has a component of an odd size

Question.
Is deciding if a graph has a matching of size at least k (for some fixed k) in NP? Co-NP? Or P?

Related Question. (The negation) Is deciding if $\nu(G) \leq k-1$ in NP Co-NP? Or P?

Answer: YES for all!

Theorem 4.4. *Deciding if G has a matching of size at least k is in NP, and hence deciding if $\nu(G) \leq k-1$ is in co$-NP$.*

Proof. The prover gives the verifier k edges and the verifier checks no two have a vertex in common. □

4.2. Covering

Definition 4.5 (Cover)**.** A **cover** of a graph G (often more specifically called a vertex covering) is a subset of vertices C of $V(G)$ such that $\nexists uv \in E(G)$ such that $u, v \in V(G) - C$. (i.e. $G - C$ has no edges. i.e. $|E(G-C)| = 0$)

(Silly) Question: Does every graph have a cover?
Answer: YES. $C = V(G)$ is a cover.
Question: For a graph G, what is the size of a smallest cover?

Definition 4.6 (Cover(ing) Number)**.** The cover(ing) number of a graph G, denoted $\tau(G)$. (i.e. $\tau(G) =$ minimum k of G has a cover of size$=k$)

Question.

Is deciding if a graph has a cover of size at most k (for some fixed k) in NP? Co-NP? Or P?

Or, equivalently, is deciding if $\tau(G) \geq k+1$ (for some fixed k) in NP? Co-NP? Or P?

Theorem 4.7. *Deciding if G has a cover of size at most k is in NP, and hence deciding if $\tau(G) \geq k+1$ is in co$-NP$.*

Proof. The prover gives a cover. The verifier checks that it is a cover. □

Deciding if $\tau(G) \geq k+1$ is $NP-$complete.

4.3. Connection Between Matchings and Covers

Theorem 4.8. *If G is a graph,*

$$\nu(G) \leq \tau(G)$$

Proof. Let C be a minimum cover of G, M be a maximum matching of G. Since C is a cover, for every $e = uv \in M$, at least one of u and v is in C. Since M is a matching, the edges of M are vertex-disjoint. Hence C has more or the same as $|M|$-vertices. Thus, $|C| \geq |M|$. □

Question. For which graphs G are $\nu(G) \leq \tau(G)$?

Definition 4.9 ($M-$alternating path)**.** Let M be a matching in graph G. An M**-alternating path** is a path whose alternate edge is in M.

Definition 4.10 (Saturated by M)**.** We say a vertex v in G is **saturated by** M or $M-$saturated if v is incident with an edge in M. Otherwise, **unsaturated**.

Definition 4.11. An $M-$alternating path P from x to y, where $x \neq y$, is called M**-augmenting** if x and y are unsaturated by M.

Theorem 4.12. *An M-augmenting path has edges of an odd size.*

Definition 4.13 (Symmetric Difference)**.** Let A and B be sets.

$$A \Delta B = (A \setminus B) \cup (B \setminus A)$$

Theorem 4.14. *Let M be a matching in G. $\exists$ an $M-$augmenting path in G. Then M is not a maximum matching.*

Proof. Set $M' = E(M)\Delta E(P)$. Then, $|M'| \geq |M|$ and M' is a matching, so M is not maximum. □

Theorem 4.15 (Lemma)**.** *If M is not a maximum matching in G, then $\exists$ an $M-$augmenting path in G.*

Proof. Since M is not maximum, $\exists$ a maximum matching M' ($\neq M$). Let $H = M\Delta M'$.

Now every component of H has maximum degree at most 2, hence is a path or a cycle.

If a component of H is a cycle C, then C is an even cycle and $|M \cap E(C)| = |M' \cap E(C)|$.

Since M' is maximum, M is not, $|M'| > |M|$, and, in fact $|H \cap M'| > |H \cap M|$. Hence $\exists$ a component P of H such that $|M' \cap E(P)| > |M \cap E(P)|$.

From above, P is not a cycle, hence is a path, and moreover, its first and last edges are in M' and not in M. Since P is a component, both ends of path of P are not in M and hence $M-$unsaturated. Hence, P is the desired M-augmenting path. □

Definition 4.16 (Reachable)**.** Let $v \in V(G)$ and M be a matching of G. We say a vertex u is reachable from v in M if $\exists$ an $M-$alternating path from v to u.

Definition 4.17 (Reachable Vertices)**.** Let $R(v)$ denote the set of reachable vertices from v in M.

Theorem 4.18. *If v is M-unsaturated and $\exists u \in R(v)$ such that $u \neq v$ and u is $M-$unsaturated, then $\exists$ an $M-$augmenting path from v to u.*

Proof. The M-alternating path from v to u is M-augmenting since u, v are M-unsaturated. □

Let $G = (A, B)$ be a bipartite graph, M be a matching of G. Let X_0 be the M-unsaturated vertices in A.

Let $Z = \bigcup_{v \in X_0} R(v)$.

Do XY-construction: we define $X := Z \cap A$ and $Y := Z \cap B$.

Theorem 4.19 (Lemma)**.** *If $\exists$ an M-unsaturated vertex in Y, then $\exists$ an M-augmenting path.*

Proof. Let u be an M-saturated vertex in Y. Since $u \in Y \subseteq Z$, $\exists$ a $v \in X_0$, such that $u \in R(v)$.

Since $u \in Y \subseteq B$, and $v \in X_0 \subseteq A$, we find that $u \neq v$.

By Proposition, $\exists$ an M-augmenting path from v to u as desired. □

Theorem 4.20 (Corollary)**.** *If M is a maximum matching of G, then $\nexists$ an M-unsaturated vertex in Y.*

Proof. By earlier lemma, if M is maximum, $\nexists$ an $M-$augmenting path. By the contrapositive of previous lemma, $\nexists$ an M-unsaturated vertex in Y. □

Theorem 4.21 (König's Theorem)**.** *If G is **bipartite**, then $\nu(G) = \tau(G)$.*

Proof. Let $G = (A, B)$. Define X_0, Z, Y as above for M. By corollary, $\nexists$ an M-unsaturated vertex in Y.

Claim 1. $\nexists$ an edge from x to $B \setminus Y$.

Proof. Suppose not, that is, $\exists ab \in E(G)$ such that $a \in X, b \in B \setminus Y$.

Since $a \in X$, $\exists v \in X_0$ such that $a \in R(v)$. Moreover, $\exists$ an M-alternating path P from v to a. If $b \in V(P)$, then $b \in R(v)$. Hence, $b \in Y$. Contradiction!

So we may assume $b \notin V(P), ab \notin M$, given the parity of P. (i.e. $\exists c \in Y$ such that $ac \in M \cap E(P)$)

Hence $P = Pb$ is an M-alternating path from v to b. So $b \in R(v)$ and hence in Y, a contradiction.

Let $C := Y \cup (A \setminus X)$. By Claim 1, $|E(G - C)| = 0$. Since G is bipartite, hence $|E(A)| - |E(B)| = 0$. So C is a cover.

□

Claim 2. $|M| = |C|$

Proof. Note that every vertex in $A \setminus X$ is M-saturated, because, otherwise, it would be in X_0, and hence in $A \setminus X$.

Since M is maximum, every vertex in Y is M-saturated.

Lastly, note $\nexists ab \in M$ such that $a \in A \setminus X$ and $b \in Y$. Since $b \in Y$, $\exists v \in X_0$ such that $b \in R(v)$ and M-alternating path P from v to b. If $a \in V(P)$, then $a \in R(v)$ and hence $a \in X$.

So we may assume $a \notin V(P)$, but then $P' = Pa$ is an M-alternating path from v to a. So $a \in R(v)$ and hence in X. Hence $|C| = |M|$. But then

$$\nu(G) \geq |M| = |C| \geq \tau(G)$$

By proposition of $V(G) \leq \tau(G)$, so $\nu(G) = \tau(G)$. □

□

4.4. Algorithm from König's Theorem

Let's recall the XY**-construction** as follows:

Definition 4.22 (XY-Construction)**.** Given a matching M, $G = (A, B)$.

- Let $X_0 = M$-unsaturated vertices in A.
- $Z = \bigcup_{v \in X_0} R(v)$
- $X = Z \cap A$
- $Y = Z \cap B$

Let's recall the following theorems.

Theorem 4.23. *If $\exists$ an M-unsaturated vertex in Y, then $\exists$ an M-augmenting path and hence M is not maximum.*

Theorem 4.24. *If $\nexists$ an M-unsaturated vertex in Y, then M is maximum and $C = Y \cup (A \setminus X)$ is a cover such that $|M| = |C|$.*

Note. This words because of the following proposition.

Theorem 4.25. *If M is a matching and C is a cover of a graph G such that $|M| = |C|$, then $|M|$ is a maximum matching and $|C|$ is a minimum cover.*

Proof. An easy corollary of earlier proposition: if M is a matching, C is a cover of G, then $|M| \leq |C|$. (Because C has to cover M)

Assume for contradiction that M is not maximum. $\exists$ a maximum matching M' with $|M'| > |M|$, but then $|M'| > |C|$, a contradiction.

Assume for contradiction that C is not minimum. $\exists$ a minimum cover C' such that $|C'| < |C|$, but then $|C'| < |C| = |M|$, i.e. $|C'| < |M|$, another contradiction. □

Question. Is deciding if G has a matching at least k in NP? Obviously true! Give a matching of size at least k and we check it.

Is it in CO-NP? Is it in P?

Theorem 4.26. *If G is bipartite, then this problem is in CO-NP.*

Proof. Prover gives verifier a cover of size less than k.

Note. Such thing exists by König's Theorem. □

Note. Algorithm also gives algorithm to find minimum cover. We can convert this to an algorithm, i.e. given a maximum matching M, find a minimum cover.

Proof. This is a proof of Termination. **Note.** Initially, $|M| = 0$. After each iteration of the **WHILE**-loop, $|M|$ increases (by 1). Since G is finite, $|M| \leq \frac{|V(G)|}{2}$. There will also be at most $\frac{|V(G)|}{2}$ **WHILE**-iterations. With good implementation, **WHILE**-loop is finite. □

Proof. This is a proof of Correctness and Running time. Supported by König's Theorem. Running time is polynomial, $|V(G)|^2$. □

4.5. Hall's Theorem

Theorem 4.27 (Hall's Theorem)**.** *Let $G = (A, B)$ be a bipartite graph. $\exists$ a matching of G saturating A iff $\forall S \subseteq A, |N(S)| \geq |S|$.*

Remark 4.28. A matching of G saturating A means that every vertex in A is saturated by the matching.

Definition 4.29 (Hall's Condition)**.** $\forall S \subseteq A, |N(S)| \geq |S|$

Remark 4.30.

- If $|A| = |B|$, then M saturating A is a perfect matching of G.
- If $|B| > |A|$, then M saturating A is NOT a perfect matching of G.
- If $|A| > |B|$, then there does not exist a matching saturating A. (Because it violates the Hall's condition at $S = A$)

Proof. **Proof of Hall's Theorem**

Remark 4.31. $N(S)$ here means union neighbourhood, i.e. $\bigcup_{s \in S} N(s)$

Proof of " $\Longrightarrow$ ":

Let S be a subset of A. Let M_S be the set of ends in B where the other end is in S, i.e.

$$M_S := \{u \in B : \exists v \in S \text{ such that } uv \in M\}$$

Since M is a matching saturating S,

$$|M_S| = |S|$$

because M saturates A.

However, $M_S \subseteq N(S)$. Hence, $|N(S)| \geq |S|$.

Remark 4.32. This direction of proof is easier than the other.

There is an even easier version of proof by proving its contrapositive, i.e., if $S \subseteq A$, such that $|N(S)| < |S|$, then there is no "hope" to saturate all of S.

Proof of " $\Longleftarrow$ ":

We assume we satisfy Hall's condition, $\forall\, S \subseteq A, |N(S)| \geq |S|$, we show there is matching saturating A (using König's Theorem, i.e., if G is bipartite, $\tau(G) = \nu(G)$.)

Remark 4.33. "There is a matching saturating A" is equivalent to "$\nu(G) = |A|$" (since you can not have any larger matching)

So suppose for contradiction there is NOT a matching saturating A, equivalently, $\nu(G) < |A|$.

By König's Theorem,

$$\tau(G) = \nu(G) < |A|$$

Let C be a cover of size less than $|A|$, or $\leq |A| - 1$.

Note that $A \setminus C \neq \emptyset$ because $|C| < |A|$. Let $S = A \setminus C$ (a clever choice of S). Since C is a cover and A is edge-less. We find that

$$N(S) \subseteq C \cap B$$

$$|B \cap C| \geq |N(S)| \underbrace{\geq}_{\text{Hall's Condition}} |S|$$

Yet, $|C| = |A \cap C| + |B \cap C| \geq |A \cap C| + |S| = |A \cap C| + |A \setminus C| = |A|$.

Now, $|C| \geq |A|$ contradicts to $|C| \leq |A| - 1$.

□

Corollary 4.34. *Let $G = (A, B)$ be bipartite. Then $|A| = |B|$ and Hall's condition holds iff G has a perfect matching.*

Corollary 4.35. *Let $G = (A, B)$ be bipartite. Then G has a perfect matching iff $\forall\, S \subseteq V(G), |N(S)| \geq |S|$. (Hall's condition for the whole graph.)*

Proof.

$$|N(S \cap A)| \geq |S \cap A|$$

$$|N(S \cap B)| \geq |S \cap B|$$

$$|N(S)| \geq |S|$$

□

Corollary 4.36. *Let $G = (A, B)$ be bipartite. If there is a number $k \geq 1$ such that*

- $\forall\, v \in A, \deg(v) \geq k$, *and*
- $\forall\, v \in B, \deg(v) \leq k$,

then there is a matching saturating A.

Proof. It suffices to show Hall's condition holds for $S \subseteq A$. Let $S \subseteq A$. Let $T = N(S)$. Let $R = E(S,T) = \{uv : u \in S, v \in T\}$

Now

$$\sum_{u \in S} \deg(u) = |R| \leq \sum_{v \in T} \deg(v)$$

Yet by the first condition,

$$k|S| \leq \sum_{u \in S} \deg(u)$$

and, by the second condition,

$$\sum_{v \in S} \deg(v) \leq k|T| = k|N(S)|$$

Altogether,

$$k|S| \leq k|N(S)|$$

and such $k \geq 1$ we find $|S| \leq |N(S)|$ as desired. □

Corollary 4.37. *Let $G = (A, B)$ be bipartite. If G is k-regular for $k \geq 1$, then there is a perfect matching of G.*

Proof. By a previous theorem, there is a perfect matching saturating A. By symmetry, there is a matching saturating B. Hence, $|A| = |B|$, so we get a perfect matching. □

Corollary 4.38. *If G is bipartite k-regular for $k \geq 1$ graph, then $E(G)$ can be partitioned into k perfect matching.*