"Sapienza" University of Rome
Faculty of Information Engineering,
Informatics and Statistics
Department of Computer Science

Graph Theory

Lecture notes integrated with the book "Graph Theory", R. Diestel

Author
Simone Bianco

March 10, 2025

Contents

Information and Contacts

Personal notes and summaries collected as part of the *Graph Theory* course offered by the degree in Computer Science of the University of Rome "La Sapienza".

Further information and notes can be found at the following link:
https://github.com/Exyss/university-notes. Anyone can feel free to report inaccuracies, improvements or requests through the Issue system provided by GitHub itself or by contacting the author privately:

- Email: **bianco.simone@outlook.it**
- LinkedIn: **Simone Bianco**

The notes are constantly being updated, so please check if the changes have already been made in the most recent version.

Suggested prerequisites:

Sufficient knowledge of algorithm design, probability and combinatorics

Licence:

1

Introduction to Graph Theory

1.1 Graphs, subgraphs and neighbors

The city of Königsberg (in Prussia), situated along the Pregel River, is divided into four regions: two parts of the mainland and two islands, Kneiphof and Lomse. In the 18th century, these areas were connected by *seven bridges* that spanned the river, crossing it in various directions. As time passed, a fascinating question emerged among residents: could one walk through the city, crossing each of the seven bridges exactly once, and return to the starting point? This became known as the *Seven Bridges of Königsberg problem.*

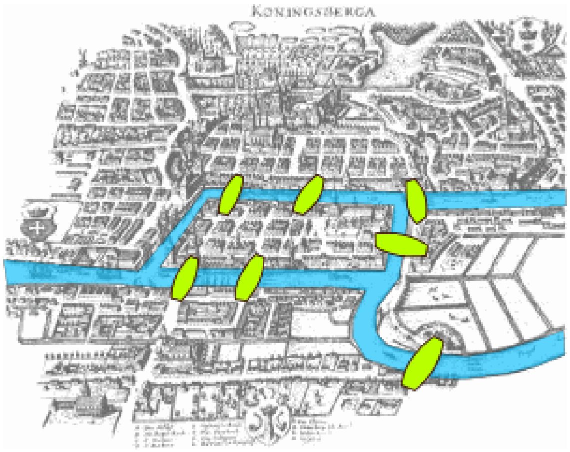

Figure 1.1: The map of Königsberg in the 18th century, showing the layout of the seven bridges.

Though it appeared to be a simple challenge, no one had succeeded in solving it. The riddle became a popular topic of conversation, discussed in markets and taverns alike. Some believed the walk was possible with the right path, while others doubted it could be done. Word of this puzzling problem eventually reached the brilliant Swiss mathematician, Leonhard Euler. Fascinated by the challenge, Euler sought a solution – not by walking the streets himself, but by abstracting the problem into a more general form. Euler realized that the precise layout of the city itself wasn't essential. What really mattered was how the landmasses were **connected** by the bridges. He began by representing each landmass as a dot and each bridge as a line between them. By doing this, he removed unnecessary details and created a simple yet powerful combinatorial structure, which we now call a **graph**.

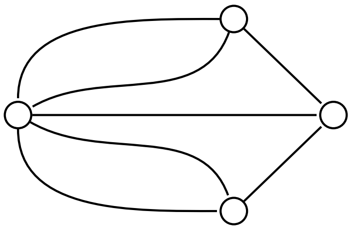

Figure 1.2: The graph drawn by Euler for the Seven Bridges of Königsberg problem.

Through his analysis, Euler discovered a key insight: for a walk to cross each bridge exactly once and return to the starting point, each landmass must be connected to an even number of bridges. In the case of Königsberg, however, every landmass was connected to an odd number of bridges, making such a walk impossible. Euler's solution, published in 1736, was groundbreaking. It not only answered the Königsberg puzzle but also laid the groundwork for an entirely new field of mathematics: *graph theory*. This area of study has since become fundamental to understanding networks, from transportation systems to social media, and even the internet itself. Thus, from a simple riddle about bridges in a small Prussian city, a new mathematical discipline was born—one that continues to influence the world to this day.

Definition 1: Graph

A **graph** is a mathematical structure $G = (V, E)$, where V is the set of vertices (or nodes) of G and E is the set of edges that link the vertices of G.

A graph can be **directed** or **undirected**. In a directed graph the edges are *oriented*, meaning that there is difference between the edge (u, v) – going from u to v – and the edge (v, u) – going from v to u. Formally, we have that:

$$E(G) \subseteq V \times V\{(u, v) \mid u, v \in V(G)\}$$

In an undirected graph, instead, the edges are *not oriented*, meaning that there not is difference between the edges (u, v) and (v, u). Formally, we have that:

$$E(G) \subseteq \binom{V(G)}{2} = \{\{u, v\} \mid u, v \in V(G)\}$$

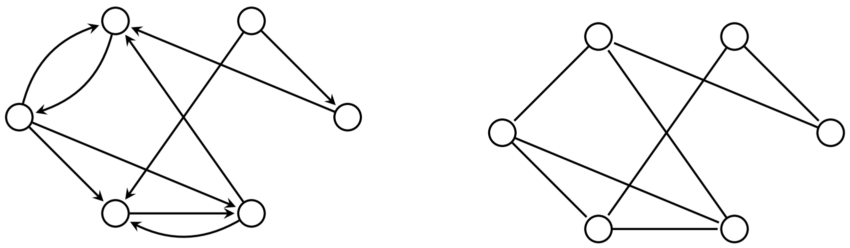

Figure 1.3: A directed graph (left) and an undirected graph (right)

We observe that the definition of graph that we just gave doesn't allow *multiple edges* between two vertices and *loops*, i.e. edges out-going frm and in-going to the same vertex. When this is the case, we say that the graph is **simple**. Generally, simple graphs are enough for any model. Sometimes, however, multiple edges and loops are needed – such as in the Seven Bridges of Königsberg problem. Graphs that allow such edges are called **multigraphs**.

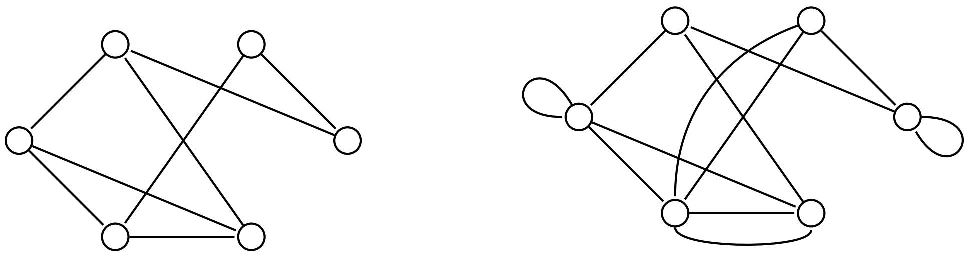

Figure 1.4: A simple graph (left) and a multigraph (right)

From now on, unless stated differently, we'll assume that each graph is simple and undirected. Moreover, to make notation lighter, we will always assume that $|V(G)| = n, |E(G)| = m$ and that $xy = \{x, y\}$ (or $xy = (x, y)$ for directed graphs).

Definition 2: Subgraph

Let G be a graph. If H is a graph such that $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$ then H is a **subgraph** of G, written as $H \subseteq G$. A subgraph $H \subseteq G$ is said to be **induced** when for all edges $xy \in E(G)$ such that $u, v \in V(H)$ it holds that $xy \in E(H)$.

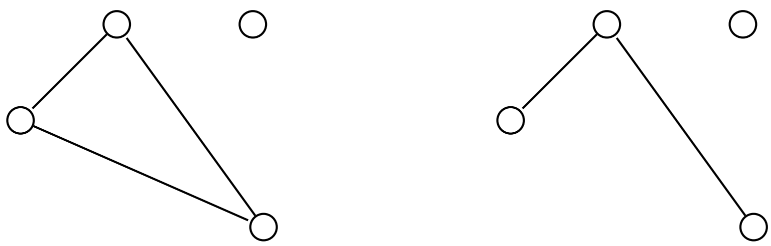

Figure 1.5: Both graphs are a subgraph of the simple graph shown on the left in Figure 1.4. The left subgraph is induced, while the right one is not.

We observe that, by definition, if a subgraph $H \subseteq G$ is induced then H is the unique induced subgraph for the vertices $V(H)$. Hence, when talking about an induced graphs we can consider his set of vertices. Given a subset of vertices $X \subseteq V(G)$, we denote with $G[X]$ the unique induced subgraph of G such that $V(G[X]) = X$, where $E(G[X]) = \{xy \in E(G) \mid x, y \in X\}$.

Definition 3: Adjancency, neighborhood and independence

Given a graph G and two nodes $x, y \in V(G)$, we say that x, y are **adjacent** to each other, written as $x \sim y$, if $xy \in E(G)$. For any edge $xy \in E(G)$, we say that xy is incident to x and y. The set of all vertices adjacent to x in G is called **neighborhood**, written as $N_G(x)$, is defined as:

$$N_G(x) = \{y \in V(G) \mid x \sim y\}$$

A subset of vertices X such that for each $x, y \in X$ it holds that $x \not\sim y$ is called **independent set**.

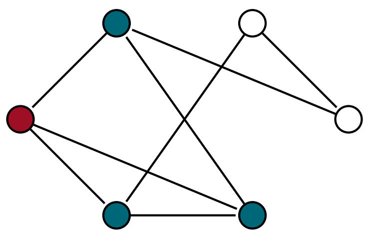

Figure 1.6: The blue nodes form the neighborhood of the red node. The red node and the white nodes form an independent set.

Definition 4: Degree

Let G be a graph. Given a vertex $x \in V(G)$, the **degree** of x over G is defined as $\deg_G(x) = |N_G(x)|$. The minimum and maximum degree of G are respectively denoted as $\delta(G)$ and $\Delta(G)$.

$$\delta(G) = \min_{x \in V(G)} \deg_G(x)$$

$$\Delta(G) = \max_{x \in V(G)} \deg_G(x)$$

When the context makes it clear, we'll simply write $\deg(x)$ instead of $\deg_G(x)$. The minimum and maximum degree of a graph are two very powerful theorem-proving tools: large portion of results are proven by reasoning on the degree of each node, often proving that some condition does or does not hold. The most basic result involving the degree of a graph is known as the **Handshaking lemma**, which can be stated in two equivalent forms.

Lemma 1: Handshaking lemma

For every graph G it holds that:

$$\sum_{x \in V(G)} \deg(x) = 2m$$

Equivalently, for every graph G the number of odd-degree vertices is even.

Proof. It's easy to see that every edge $xy \in E(G)$ is counted exactly two times through $\deg(x)$ and $\deg(y)$, implying that:

$$\sum_{x \in V(G)} \deg(x) = 2m$$

Consider now the subset $X \subseteq V(G)$ containing all the vertices of even degree. We observe that:

$$2m = \sum_{x \in V(G)} \deg(x) = \sum_{x \in X} \deg(x) + \sum_{x' \in V(G)-X} \deg(x')$$

Let $a = \sum_{x \in X} \deg(x)$ and $b = \sum_{x' \in X} \deg(x')$. Since the degrees of the vertices in X are even, we know that a is even. Hence, in order for $2m = a + b$ to hold, b must also be even. However, since each degree in $V(G) - X$ is odd, in order for b to be even it must hold that $|V(G) - X|$ is even. $\square$

1.2 Paths, walks, cycles and trees

After discussing the more general concept of subgraph, we can now focus on particular types of structures that can be usually found in graphs. These sub-structures are the real fundamental tool of reasoning for graph properties.

Definition 5: Path

A **path** is a graph P such that $V(P) = \{x_1, \ldots, x_n\}$ and $E(G) = \{x_0x_1, x_1x_2, \ldots, x_{n-1}x_n\}$. The length of a path is defined as the number of edges that form it. A path with n vertices is denoted as P_n.

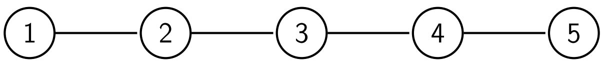

Figure 1.7: The path P_5.

Definition 6: Walk

Let G be a graph. A **walk** on G is defined as a sequence $x_0e_1x_1e_2\dots e_{k-1}e_kx_k$ where $x_0,\dots,x_k \in V(G)$ and $e_1,\dots,e_k \in E(G)$. The length of a walk is defined as the number of edges that form it.

When the first and last vertices are equal, i.e. $x_0 = x_k$, we say that the walk is **closed**. We observe that, by definition, a walk allows edges and vertices to be repeated in the sequence. When no vertices in a walk are repeated, the walk corresponds to a path.

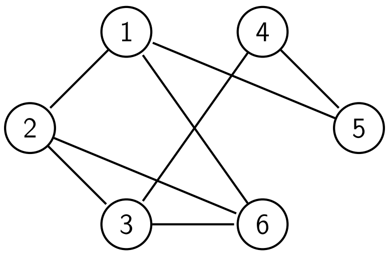

Figure 1.8: The sequence $1\{1,2\}2\{2,6\}6\{6,1\}1\{1,5\}5$ forms a walk on G, but not a path. The sequence $1\{1,2\}2\{2,6\}6\{6,3\}3\{3,4\}4$, instead, forms a path on G.

Proposition 1: Paths and walks

Let G be a graph. Given the vertices $x,y \in V(G)$, there is a path from x to y if and only if there is a walk from x to y.

Proof. Since every path is also a walk, the first direction is trivial. Let $W = x_0e_1x_1e_2\dots e_{k-1}e_kx_k$ be the shortest walk. i.e the one with minimum length, such that $x = x_0$ and $y = x_k$. By way of contradiction, suppose that W is not a path. Hence, at least one edge in W must repeat at least once. Let $i,j \in [k]$ be two indices such that $e_i = e_j$. Then, the following sequence W' is a walk from x to y with fewer edges than W, raising a contradiction. Hence, W must be a path.

$$W' = x_0e_1x_1e_2\dots e_ix_ie_{j+1}x_{j+1}e_{j+2}\dots e_{k-1}e_kx_k$$

□

Proposition 2

The longest path in any graph has length at least δ.

Proof. If $\delta = 1$ then the longest path is trivially made by one single edge. Suppose now that $\delta \geq 2$, implying that there are at least two vertices in G. Let P be the longest path in G and let $x_1,\dots,x_k$ be its vertices.

Claim: $N(x_k) \subseteq \{x_1,\dots,x_{k-1}\}$.

Proof. By way of contradiction, suppose that there is a vertex $x' \in N(x_k)$ such that $x' \notin \{x_1, \dots, x_{k-1}\}$. Then, since $x_k \sim x'$, there must be an edge $x_k x'$, implying that the path $P \cup x_k x'$ is longer than P, raising a contradiction. □

Through the claim we easily conclude that $\delta \leq |N(x_k)| \leq k$, meaning that P has length at least δ. □

Definition 7: Cycle

A cycle is a graph C such that $V(C) = \{x_1, \dots, x_n\}$ and $E(G) = \{x_0x_1, x_1x_2, \dots, x_{n-1}x_n, x_nx_1\}$. The length of a cycle is defined as the number of edges that form it. A cycle with n vertices is denoted as C_n.

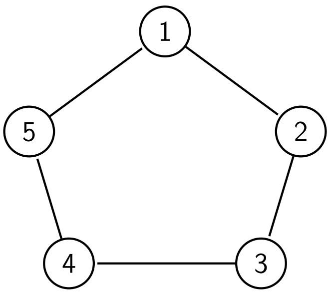

Figure 1.9: The cycle C_5.

Proposition 3

In any graph such that $\delta \geq 2$ there is a cycle of length at least $\delta + 1$.

Proof. Let P be the longest path in G and let $x_1, \dots, x_k$ be its vertices. Through an argument equal to the claim of Proposition 2, we know that $N(x_k) \subseteq \{x_1, \dots, x_{k-1}\}$. Let $i \in [k-1]$ be the minimal index such that $x_i \in N(x_k)$. Since every neighbor of x_k must be inside P and the graph is simple, it must hold that $i \geq \delta$, meaning that the vertices $x_i, x_{i+1}, \dots, x_{k-1}, x_k, x_i$ form the cycle C_{i+1}. □

Definition 8: Connectivity, components and distance

Let G be a graph. Two nodes $x, y \in V(G)$ are said to be linked if there is a path from x to y. If every pair of nodes in G is linked, we say that G is **connected**. If a connected subgraph H of G is maximal – meaning that no other edges can be added to it while preserving connectivity – H is called **component** of G. The **distance** $\text{dist}_G(x, y)$ between two nodes x, y is the length of the shortest path connecting them.

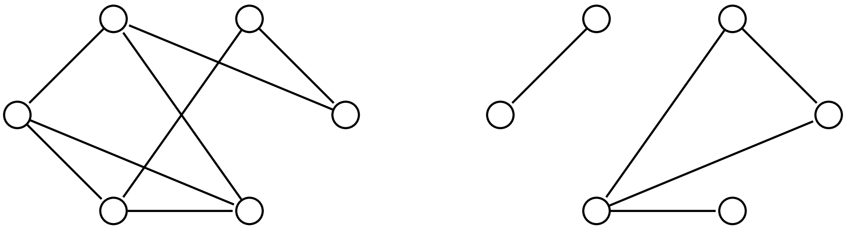

Figure 1.10: A connected graph (left) and a disconnected graph (right). The connected graph has an unique component, while the disconnected graph has two components.

Lemma 2

Let G be a graph. If G is connected and C is a cycle in G then for all $e \in E(C)$ it holds that $G - \{e\}$ is still connected.

Proof. Fix an edge $e \in E(C)$. Let $P = x_0 e_1 x_1 \dots e_k x_k$ be a path in G from x to y. If $e \notin E(P)$ then P is also a path in $G - \{e\}$, preserving the connectivity of x and y. Suppose now that $e \in E(P)$. Given $C = z_0 f_1 z_1 \dots f_\ell z_\ell$, without loss of generality assume that $e = e_i = f_1$. Then, the following sequence W is a walk from x to y in $G - \{e\}$ – we cannot be sure that W is a path since P may intersect C on multiple edges.

$$W = x_0 e_1 x_1 \dots x_i f_\ell z_{\ell-1} \dots f_2 z_2 e_{i+1} x_{i+1} \dots e_k x_k$$

By Proposition 1, we know that since W is a walk from x to y in $G - \{e\}$ there must also be a path from x to y in $G - \{e\}$, preserving connectivity. □

Definition 9: Tree

A **tree** is an connected acyclic subgraph. Any vertex in a tree with degree 1 is called leaf. A rooted tree is a tree with a chosen node called **root**. If every component of a graph is a tree, the graph is referred to as a **forest**.

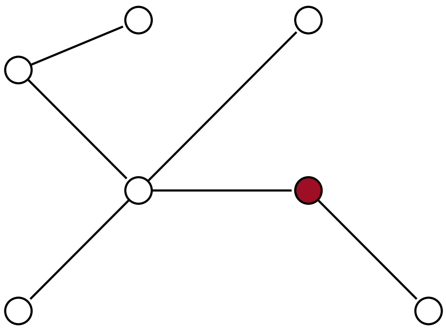

Figure 1.11: A rooted tree. The red node has been chosen as the root.

When the tree is rooted in $r \in V(T)$, we often use the concept of **ancestor** and **parent**. Given two nodes $a, x \in V(T)$, we say that a is an ancestor of x if x lies on a path from r

to x. If p is an ancestor of x and $px \in E(T)$, we say that p is the parent of x. The **least common ancestor (LCA)** between two vertices $x, y \in V(T)$ is the ancestor $z \in V(T)$ shared by x and y that minimizes the value of $\mathrm{dist}(r, z)$. We observe that every pair of vertices of a tree must have a LCA since the root is an ancestor of every node.

Theorem 1: Equivalent definitions of tree

Given a graph T, the following statements are equivalent:

1. T is a tree
2. Every vertex pair of T is connected by an unique path
3. T is minimally connected
4. T is maximally acyclic

Proof. We'll proceed by proving a chain of implications.

$1 \implies 2$ Without loss of generality, assume that every pair of nodes has at least one path, since otherwise T is not connected, hence not a tree. Suppose now that there are two vertices $x, y \in V(T)$ that have at least two different paths P, Q from x to y. While traversing P from x to y, let z be the first node such that $z \in V(P) \cap V(Q)$. Similarly, let $w \in V(P) \cap V(Q)$ be the first node encountered while traversing Q from y to x. We observe that since $x, y \in V(P) \cap V(Q)$, the vertices z, w always exist. Given the subpath $P' \subseteq P$ from w to z and the subpath $Q' \subseteq Q$ from z to w, the graph $Q' \cup P'$ is a cycle in T, meaning that T cannot be a tree. By contrapositive, we get that if T is a tree then every pair of vertices is connected by an unique path.

$2 \implies 3$ Suppose that every vertex pair of T is connected by an unique path. Then, T is clearly connected. By way of contradiction, suppose that there is an edge $e \in E(G)$ such that $T - \{e\}$ is still connected. Then, the edge e must be part of at least one of the unique paths connecting two nodes, meaning that such path cannot exist inside $T - \{e\}$, implying that it is not connected. Hence, T must be minimally connected.

$3 \implies 4$ Suppose that T is minimally connected but not maximally acyclic. By way of contradiction, suppose that T has a cycle. Then, by Lemma 2 we know that we can remove an edge of the cycle from T and keep it connected, contradicting the fact that T is minimally connected. Hence, T must be acyclic. Pick two vertices $x, y \in V(T)$. Since T is connected, we know that there is a path P connecting x to y. Then, if we were to add the edge xy, the subgraph $P \cup \{xy\}$ would be a cycle in $T \cup \{xy\}$. Thus, T is maximally acyclic.

$4 \implies 1$ Suppose that T is maximally acyclic. Fix a pair of vertices $x, y \in V(T)$. Since T is maximally acyclic, we know that adding the edge xy makes $T \cup \{xy\}$ cyclic. Let C be the cycle in $T \cup \{xy\}$ containing xy. Then, $C - \{xy\}$ must be a path in T from x to y.

□

Lemma 3

Let T be a tree. Then:

1. If T has at least two nodes then T has at least a leaf.
2. If $x \in V(T)$ is a leaf then $T - \{x\}$ is still a tree

Proof.

1. By way of contradiction, suppose that T is a tree without leaves. Then, we have that $\delta \geq 2$. However, by Proposition 3, in T there must be a cycle with length at least $\delta + 1$, contradicting the very definition of tree. Hence, T must have at least a leaf.
2. Let x be a leaf of T. Since T is acyclic, $T - \{x\}$ is also clearly acyclic. By way of contradiction, suppose that $T - \{x\}$ is not connected. Then, there are at least two vertices $u, v \in V(T - \{x\})$ for which there is a path P between them in T but not in $T - \{x\}$. Since by removing x the vertices u, v became disconnected, the vertex x must lie on P. Moreover, since $u, v \neq x$, the vertex x must be an internal node of the path, meaning that it must have degree 2, contradicting the very definition of leaf.

□

Definition 10: Spanning tree

Given a graph G, we say that $T \subseteq G$ is a **spanning tree** of G if T is a tree and $V(T) = V(G)$.

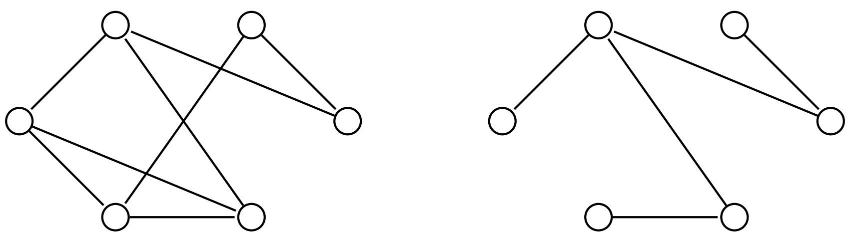

Figure 1.12: A graph (left) and one of its spanning trees (right)

Lemma 4

Every connected graph has a spanning tree.

Proof. If G is a tree then clearly it is its own spanning tree. Suppose now that G is a connected graph that is not a tree, meaning that it is acyclic. Then, through Proposition 3, we can keep removing the edges $e_1, \ldots, e_k$ from every cycle of G until we reach a minimally connected subgraph T such that $V(T) = V(G)$. By Theorem 1, we know that T must be a tree. □

Theorem 2

Let T be a connected graph. Then, T is a tree if and only if it has $n-1$ edges.

Proof. We proceed by induction on n. If $n=1$ then T is trivially the tree with 0 edges. If $n>1$, instead, through the previous lemma we know that T must have at least a leaf $x \in V(T)$ for which $T-\{x\}$ is still a tree. By inductive hypothesis, we know that $T-\{x\}$ has $n-2$ edges. Hence, by adding the unique edge incident to x, we get that T has $n-1$ edges.

Vice versa, by way of contradiction, suppose that T is connected, has $n-1$ edges but that it is not acyclic. Then, through the previous lemma we know that T must have a spanning tree T'. Moreover, since T has a cycle and T' does not, we know that T' must have fewer edges than T. However, we have just proven that every tree must have $n-1$ edges. Hence, we get that $n-1=|E(T')|<|E(T)|=n-1$, which is a contradiction. Hence, T must also be acyclic. $\square$

1.3 Complete graphs and bipartite graphs

Definition 11: Complete graph

A **complete graph** is a graph where every pair of vertices is adjacent to each other. A complete graph with n vertices in denoted as K_n. An induced subgraph that is complete is called **clique**.

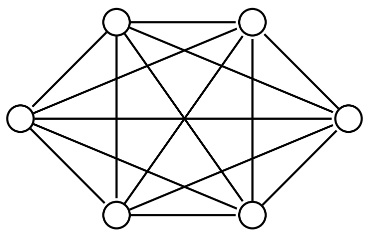

Figure 1.13: The complete graph K_6

Definition 12: Bipartite graph

A **bipartite graph** is a graph with a subset of vertices $X \subseteq V(G)$ such that for every edge $\{u,v\} \in E(G)$ it holds that $u \in X$ and $v \in V(G)-X$. Equivalently, we have that both X and $V(G)-X$ are independent sets. The pair $(X, V(G)-X)$ is called **bipartition** of the graph.

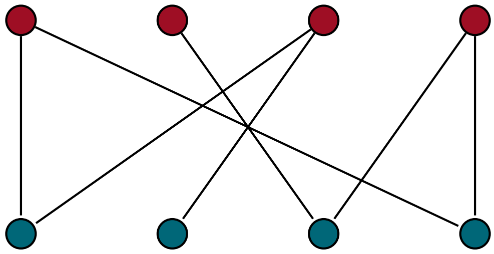

Figure 1.14: The red and blue nodes form a bipartition of the graph.

It's easy to see that no complete graph can be bipartitioned since it's impossible to separates the nodes into two independent sets. In fact, by definition, cliques are the very opposite of the concept of independent set. Through the following lemma we can extend the claim on complete graphs to cliques: if G contains a clique of any cardinality then G cannot be bipartitioned.

Proposition 4

G is bipartite if and only if every subgraph of G is bipartite

Proof. The converse implication trivially holds. Let H be a subgraph of G. Then, any bipartition $(X, V(G) - X)$ of G induces a bipartition $(X \cap V(H), V(H) - X)$ over H. $\square$

Lemma 5

G is bipartite if and only if every component of G is bipartite

Proof. The direct implication trivially follows from the previous proposition. Given the components $H_1, \dots, H_k$ of G, let $(X_1, V(H_1) - X_1), \dots, (X_k, V(H_k) - X_k)$ be the bipartitions of the components. Then, since each component is by definition disjoint from the others, the pair $(X, V(G) - X)$ where $X = \bigcup_{i \in [k]} X_i$ is a bipartition of G. $\square$

We observe that the above characterizations of bipartite graphs are useful but still "weak". A stronger characterization can be achieved through the following theorem. Given a path P and two vertices $x, y \in V(P)$, we denote the subpath of P from x to y with xPy.

Theorem 3: Equivalent defintion of bipartite graphs

G is bipartite if and only if it doesn't contain odd cycles.

Proof. Through the above proposition, it's sufficient to prove that for all $k \in \mathbb{N}$, C_{2k+1} is not bipartite. Fix $k \in \mathbb{N}$. By way of contradiction, suppose that there is bipartition $(X, V(G) - X)$ of C_{2k+1}. Let $x_1, \dots, x_{2k+1}x_1$ be the edges of C_{2k+1}. Without loss of generality, assume that $x_1 \in X, x_2 \notin X, x_3 \in X, \dots, x_{2k} \notin X$. Then, if $x_{2k+1} \in X$ then $x_{2k+1}x_1 \subseteq X$, while if $x_{2k+1} \notin X$ then $x_{2k}x_{2k+1} \subseteq V(G) - X$. In both cases, we get that the pair cannot be a bipartition.

Vice versa, suppose that G is not bipartite. Then, through the previous lemma at least one component H of G is not bipartite. Let T be a spanning tree of H and fix $r \in V(T)$ as its root. Let $X = \{x \in V(T) \mid \mathrm{dist}_T(r,x) \text{ is even}\}$. By definition, $(X, V(T) - X)$ for a bipartition of T. Hence, since T is bipartite but H isn't, there must be an edge $xy \in E(H)$ such that either $x, y \in X$ or $x, y \notin X$. Let P_x and P_y respectively be the paths in T from x to r and from y to r. Let z be the LCA of x and y in T.

Claim: $C = zP_x x \cup zP_y y \cup xy$ is an odd cycle

Proof of the claim. Since either $x, y \in X$ or $x, y \notin X$ holds, we know that $\mathrm{dist}(r,x)$ and $\mathrm{dist}(r,y)$ must be either both even or both odd. Hence, the lengths of P_x and P_y must share the same parity. Moreover, since z is the LCA of x and y w have that $rP_x z = rP_y z$. Thus, the lengths of $zP_x x$ and $zP_y y$ must also share the same parity. This concludes that $zP_x x \cup zP_y y \cup xy$ must be an odd cycle. □

Since C is an odd cycle and it is a subgraph of T, and thus of G, we conclude that G contains an odd cycle. □

1.4 Eulerian tours and Hamiltonian paths

At the start of this chapter, we introduced the Seven Bridges of Königsberg problem, which led to the emergence of graph theory as a branch of combinatorics. In modern graph theory, the problem is formalized through the concept of *Eulerian tour*.

Definition 13: Eulerian tour

An Eulerian tour over a graph G is closed walk that traverses every edge of G exactly once.

To solve the problem, Euler [Eul41] proved the following theorem, which implies that the answer is "no" since the multigraph that models the problem contains some odd-degree vertices.

Theorem 4: Euler's theorem

A graph (or multigraph) has an Eulerian tour if and only if G is connected and every vertex has even degree

Proof. By way of contradiction, suppose that G has an Eulerian tour W. By way of contradiction, suppose that G is not connected. Then, we get an easy contradiction: if G is disconnected then W cannot traverse every edge of the graph. Hence, G must be connected. Again, by way of contradiction suppose that G has at least one odd-degree vertex $x \in V(G)$. Let $\deg(x) = 2k + 1$. Since W is a closed walk, we can assume without loss of generality that x is the first vertex of the walk. When traversing W starting from x, one of the edges incident to x is crossed, hence we have $2k$ incident edges left. Every time the tour returns to x, two edges are crossed – one in-going and one out-going. Hence, in order to cover all the edges, the tour has to return to x for k times. However, this implies that there are no more edges left to close the tour on x, raising a contradiction. Hence, the vertex x cannot exist.

Vice versa, assume that G is connected and every vertex has even degree. Let $W = x_0 e_1 x_1 \dots e_k x_k$ be the longest walk over G with no repeating edges.

Claim 1: $x_k = x_0$, i.e. W is a closed walk

Proof of Claim 1. By way of contradiction, suppose that $x_k \neq x_0$. Let $2\ell + 1$ be the number of edges incident to x_k in W – one edge is given by e_k while 2ℓ edges are given by edges needed to cross the other ℓ the vertices $x_{i_1}, \dots, x_{1_\ell}$ such that $x_i = x_k$. Since x_k has even degree, we know that there must be another edge $\{x_k, y\} \in E(G - W)$, implying that $W \cup \{x_k, y\}$ is walk longer than W, which is absurd. □

Claim 2: W contains every edge of G

Proof of Claim 2. By way of contradiction, suppose that that there is an edge $uv \in E(G - W)$. Fix $i \in [k]$. By connectivity of G, we know that there must be a path P disjoint from x_i to u or from x_i to v. Without loss of generality, assume that P is a path from x_i to u. Since $uv \notin E(W)$, there must be an edge $x_j y \in E(P - W)$ outgoing from W. Since W is a closed walk, we can assume without loss of generality that x_j is the first (and last) vertex of the walk. Then, the walk $W \cup x_j y$ is walk that doesn't repeat any vertices longer than W, raising a contradiction. □

Through the two claims, we conclude that W is an Eulerian tour. □

Bibliography

[Eul41] Leonhard Euler. "Solutio problematis ad geometriam situs pertinentis". In: *Commentarii academiae scientiarum Petropolitanae* (1741).