Ve203 Discrete Mathematics

Runze Cai

University of Michigan - Shanghai Jiao Tong University
Joint Institute

Spring 2023

Part III

Selected Topics in Graph Theory

Table of Contents

Partial Order

Definition

An ***ordered set*** (or ***partially ordered set*** or ***poset***) is an ordered pair $(P, \leq)$ of a set P and a binary relation $\leq \subset P \times P$, called the ***order*** (or the ***partial order***) on P such that $\leq$ is

- reflexive: $(\forall x \in P)(x \leq x)$.
- antisymmetric: $(\forall x, y \in P)(x \leq y \wedge y \leq x \rightarrow x = y)$.
- transitive: $(\forall x, y, z \in P)(x \leq y \wedge y \leq z \rightarrow x \leq z)$.

We write $x < y$ if $x \leq y$ and $x \neq y$. (Other notation: $\preceq$ and $\prec$)

Definition

If $(P, \leq)$ is a poset, and for all $x, y \in P$, either $x \leq y$ or $y \leq x$, then it is a ***total order*** or ***linear order***.

Example of linear/total order

- $\mathbb{Z}$
- $\mathbb{Q}$
- $\mathbb{N}$
- $\mathbb{R}$

Representation of Finite Poset

Let $P = (X, \leq_P)$ be a nonempty finite poset, with the ground set $X = \{x_1, x_2, \ldots, x_n\}$. A total order $L = (X, \leq_L)$ is a ***linear extension*** of P if $x_i \leq_P x_j$ implies $x_i \leq_L x_j$ for all $x_i, x_j \in X$.

Matrix Representation

Suppose $x_1 \leq_L x_2 \leq_L \cdots \leq_L x_n$, then the matrix representation of the poset P given by $M_P = (m_{ij})$ is upper-triangular.

Example

Consider $X = \mathcal{P}([2]) = \{\varnothing, \{1\}, \{2\}, \{1,2\}\}$, ordered by inclusion $\subset$. Thus

$$M_X = \begin{array}{c} \\ \varnothing \\ \{1\} \\ \{2\} \\ \{1,2\} \end{array} \begin{array}{c} \begin{array}{cccc} \varnothing & \{1\} & \{2\} & \{1,2\} \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

Partial Order

Pre-order/Quasi-order

- reflexive: $(\forall x \in P)(x \leq x)$
- transitive: $(\forall x, y, z \in P)(x \leq y \wedge y \leq z \rightarrow x \leq z)$

Partial Order

- antisymmetric: $(\forall x, y \in P)(x \leq y \wedge y \leq x \rightarrow x = y)$

Total/Linear Order

- total: $(\forall x, y \in P)(x \leq y \vee y \leq x)$

Covers in a Poset

Definition

Let P be an ordered set. Then $y \in P$ is called a cover of $x \in P$ if $x < y$ and for all $z \in P$, $x \leq z \leq y$ implies $z \in \{x, y\}$. We also say that y covers x, or x is covered by y. Such x and y are called ***adjacent***.

Examples

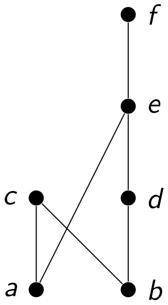
- In $(\mathcal{P}([6]), \subset)$, $\{1, 3\}$ is covered by $\{1, 3, 5\}$, but not covered by $\{1, 2, 3, 4\}$.
- In $\mathbb{Z}$, each $k \in \mathbb{Z}$ is covered by $k + 1$, and covers $k - 1$.
- In $(\mathbb{N}, |)$, 15 is covered by 105, 14 is not covered by 84.
- In $\mathbb{R}$ and $\mathbb{Q}$, no two elements are covers of each other.

Hasse Diagram (Bottom-up)

Hasse/Order Diagram (Idea: keep the most essential component.)

- Edges are the cover pairs (x, y) with x covered by y;
- Edges are drawn such that x is below y;
- Edges are monotone vertically.

Example



What relation does the Hasse diagram on the left corresponds to?

$$\begin{aligned}\leq = \{&(a,a),(b,b),(c,c),(d,d),(e,e),(f,f),\\ &(a,e),(a,f),(b,d),(b,e),(b,f),(b,c),\\ &(d,e),(d,f),(a,c),(e,f)\}\end{aligned}$$

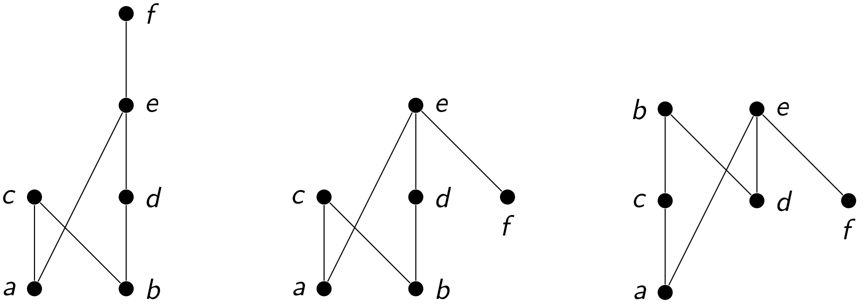
Hasse Diagram (Top-down)

To construct a Hasse or order diagram for a poset $(P, \leq)$

- construct a digraph of the poset $(P, \leq)$ so that all arcs point up (except the loops).
- Eliminate all loops.
- Eliminate all arcs that are redundant because of transitivity.
- Eliminate the arrows on the arcs

Hasse Diagrams of Three Different Posets

Example

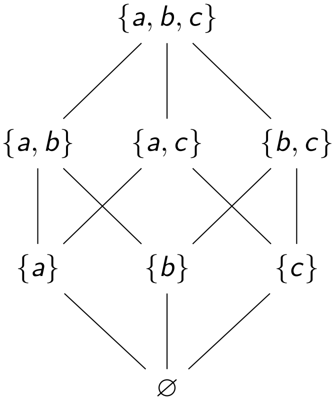


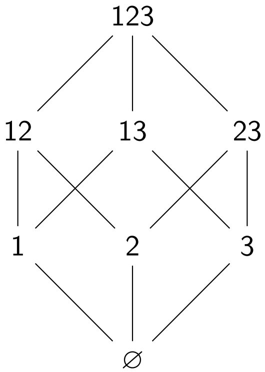
Note that all three are the same as graphs, but not as posets.

Partial Order

Examples

- Power set/Boolean lattice $(\mathbf{2}^{[n]}, \subseteq)$. $[n] = \{1, \ldots, n\}$, subsets of $[n]$ ordered by inclusion.

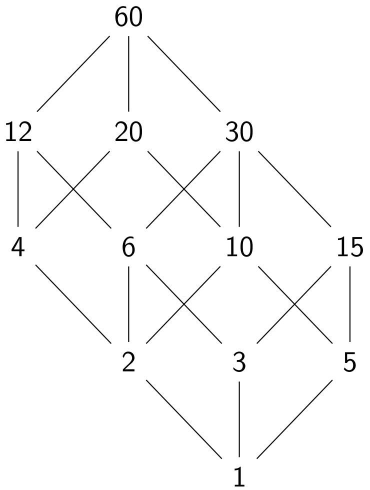




Partial Order

Examples

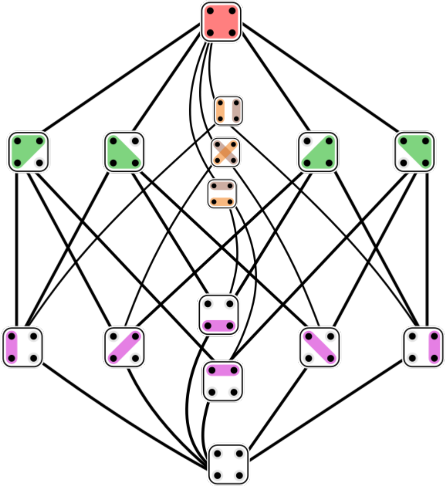
- Divisors of $n \in \mathbb{N}$. $(\mathbb{N}, |)$. Ordered by divisibility. $n = 60$.



Partial Order

Examples

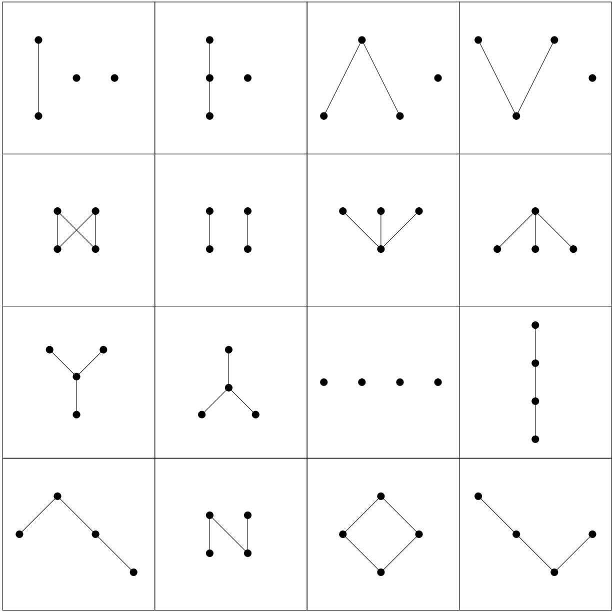
- Partition of $[n] = \{1, \ldots, n\}$, ordered by refinement.



Partial Order

Examples

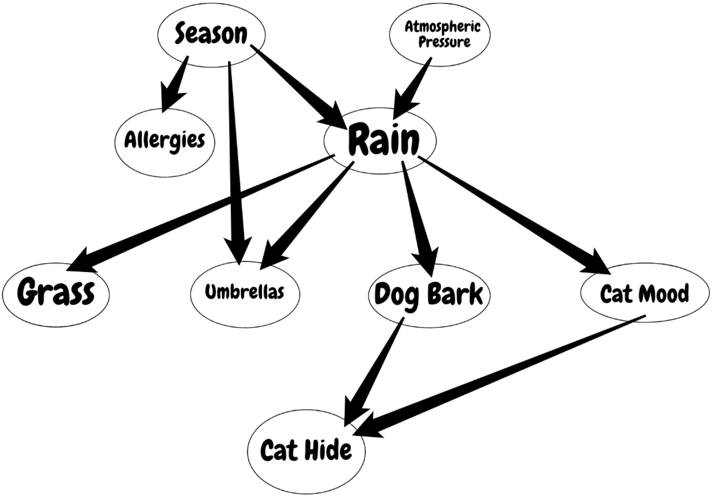
- All posets on a set with 4 elements (up to relabeling of the points).



Partial Order

Examples

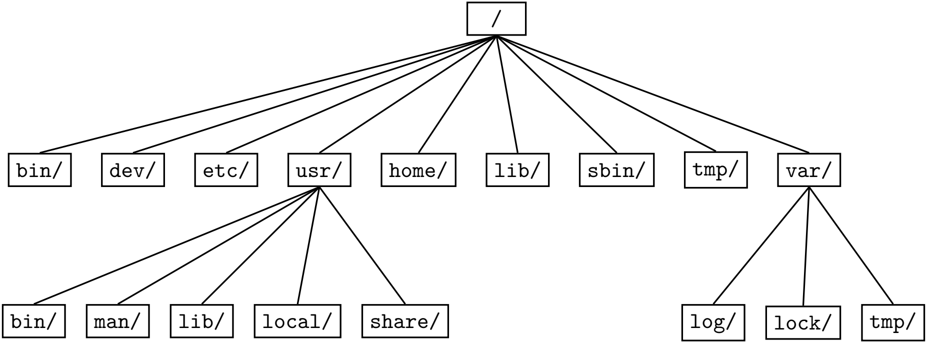
- Any directed acyclic graph (DAG), e.g., Bayesian network.



Partial Order

Examples

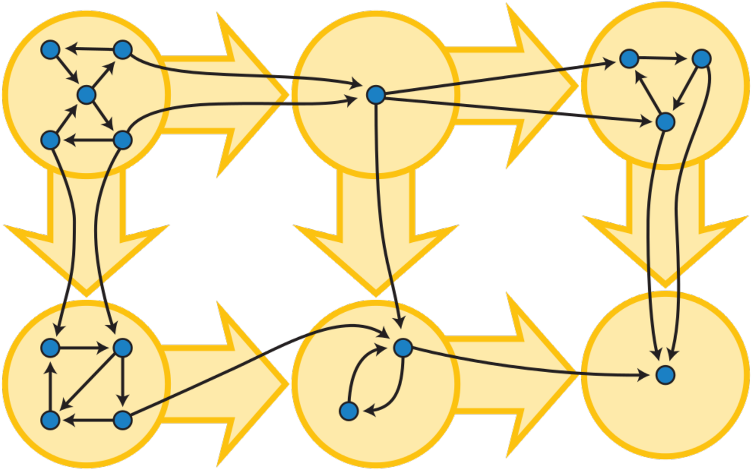
- Vertices in a rooted tree (e.g., computer directory structure, family tree).



Partial Order

Examples

- Strongly connected components in a directed graph. (cf., preorder)



Partial Order

Examples

- sub-trees/graphs/groups/vector spaces of a trees/graphs/groups/vector spaces.

Non-example

- $(\mathbb{Z}, |)$. $-1|1$ and $1|-1$, but $1 \neq -1$.

More Definitions

Definition

Let $(P, \leq)$ be a poset, and $a, x, y, z \in P$.

- If $a \in P$ but $\nexists x \in P$ such that $x < a$, then a is a ***minimal element***.
- If $a \leq x$ for all $x \in P$, then a is the ***minimum element***.
- If $z \in P$ but $\nexists x \in P$ such that $z < x$, then z is a ***maximal element***.
- If $x \leq z$ for all $x \in P$, then z is the ***maximum element***.
- If either $x < y$ in P or $y < x$ in P, then x and y are ***comparable*** in P, otherwise x and y are ***incomparable***.

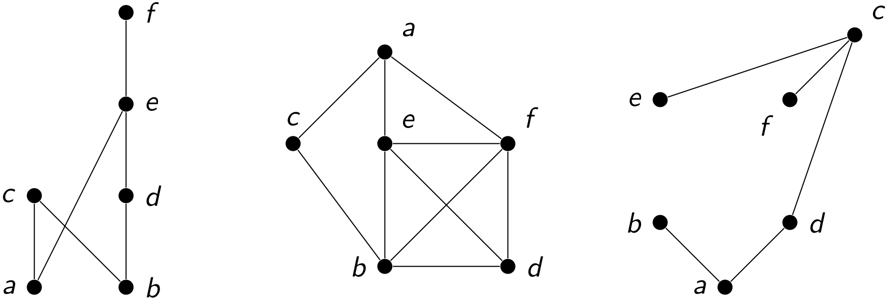
Definition

Given a poset $(P, \leq_P)$ and $Q \subset P$, then the (binary) relation $\leq_Q = \leq_P|_{Q \times Q}$ is a partial order on Q. The induced poset $(Q, \leq_Q)$ is called ***subposet*** of $(P, \leq_P)$.

Comparability and Incomparability Graphs

With a poset $(P, \leq)$, we associate a ***comparability graph*** $G_1 = (P, E_1)$ and an ***incomparability graph*** $G_2 = (P, E_2)$, where
$E_1 = \{\{x, y\} \in \binom{P}{2} \mid x, y \text{ comparable}\}$ and
$E_2 = \{\{x, y\} \in \binom{P}{2} \mid x, y \text{ incomparable}\}$.

Example

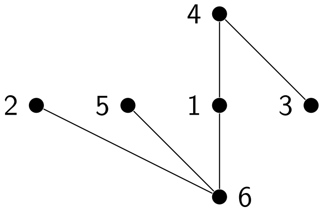


Note that a comparability graph and an incomparability graph are complement graph of each other. The ***complement*** of graph $G = (V, E)$ is
$\overline{G} = \left(V, \binom{V}{2} - E\right)$.

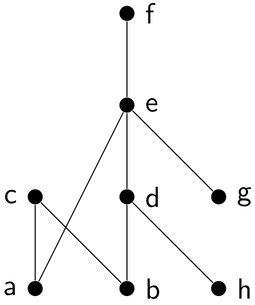
An Example

Let $P = \{1, 2, 3, 4, 5, 6\}$, and $\leq = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (6, 1), (6, 4), (1, 4), (6, 5), (3, 4), (6, 2)\}$. Then

- 6 and 3 are minimal elements.
- 2, 4, and 5 are maximal elements.
- 4 is comparable to 6.
- 2 is incomparable to 3.
- 1 covers 6, and 3 is covered by 4.
- $4 > 6$ but 4 does not cover 6.



Another Example



- c and f are maximal elements.
- a, b, g, and h are minimal elements.
- a is comparable to f.
- c is incomparable to h.
- e covers a, and h is covered by d.
- $e > h$ but e does not cover h.

Chains and Antichains

Definition

Given $(P, \leq)$ poset,

- A ***chain*** in a poset is a subset $C \subset P$ such that any two elements are comparable.
- An ***antichain*** in a poset is a subset $A \subset P$ of incomparable elements.

Definition

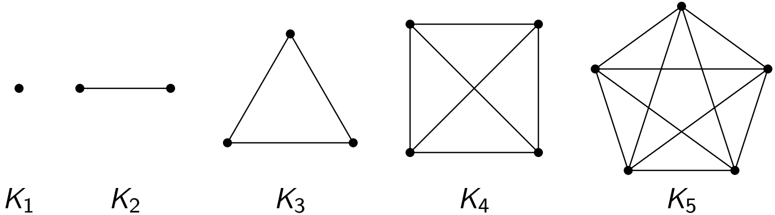
A graph $G = (V, E)$ is called a ***clique*** or ***complete graph*** if $E = \binom{V}{2}$. Conversely, the complement graph of $G = (V, \binom{V}{2})$, given by $\overline{G} = (V, \varnothing)$, is called an ***independent graph*** or ***independent set***.

Remark

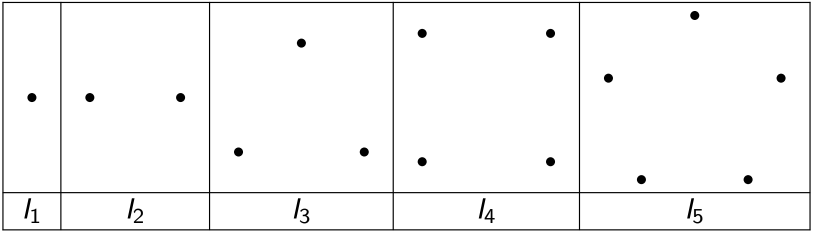
- The comparability graph of a chain is a complete graph.
- The comparability graph of an antichain is an independent graph.

Complete Graphs and Independent Graphs

Complete Graphs K_n



Independent Graphs I_n



Chains and Antichains

Lemma

Given a chain C and an antichain A of a poset, $|A \cap C| \leq 1$.

Proof.

If $|A \cap C| \geq 2$, then we can find two elements that are both comparable and incomparable. Contradiction. □

Chains and Antichains

Definition

A chain C in P is

- ***maximal*** if there exists no chain C' such that $C \subsetneq C'$.
- ***maximum*** if for all chain C', $|C| \not< |C'|$.

The ***height*** (not *length*) of a poset P, denoted by $h(P)$, is the maximum size of a chain in P.

Definition

An antichain A in P is

- ***maximal*** if there exists no antichain A' such that $A \subsetneq A'$.
- ***maximum*** if for all chain A', $|A| \not< |A'|$.

The ***width*** of a poset P, denoted by $w(P)$, is the maximum size of an antichain in P.

Remark

A maximal chain or maximal antichain CANNOT be prolonged by adding a new element.

Chains and Antichains

Observation

By pigeonhole principle,

- If P can be partitioned into t antichains, then the height of P is at most t.
- If P can be partitioned into s chains, then the width of P is at most s.

Observation

The set of maximal (or minimal) elements is an antichain.

Theorem (Mirsky's Theorem, 1971)

A poset of height h can be partitioned into h antichains.

Proof.

Recursively remove the set of maximal (or minimal) elements. $\square$

Mirsky's Theorem (dual Dilworth)

Proof. (a little more detail).

Denote the set of minimal elements of $(P, \leq)$ by $\mathrm{Min}(P)$. Similarly for $\mathrm{Max}(P)$. Thus we have a partition of P into antichains $A_1, \ldots, A_k$, $k \in \mathbb{N}$. Since $|A_i \cap C| \leq 1$ for any chain $C \subset P$, then (recall observation)

$$\begin{aligned} k &\geq \max\{|C| : C \text{ is a chain in } P\} \\ &= h(P) \end{aligned}$$

Input: A partial order $(P, \leq)$
Output: An antichain partition of $(P, \leq)$

```
i ← 1
while P ≠ ∅ do
    A_i ← Min(P)
    P ← P − A_i
    i ← i + 1
end
return {A_1, ..., A_{i−1}}
```

Claim: a chain of length k can be traced back from A_k.
Indeed, choose $x_k \in A_k$, then $\exists x_{k-1} \in A_{k-1}$ such that $x_{k-1} < x_k$, and so on.
Eventually, we have $x_1 < x_2 < \cdots < x_{k-1} < x_k$. Therefore $h(P) = k$. $\square$

Dilworth's Theorem

Theorem (Dilworth's Theorem, 1950)

A poset of width w can be partitioned into w chains.

Remark

Dilworth theorem holds if the size of the poset is infinite, however, the width w needs to be finite, i.e., $w \in \mathbb{N}$.

Dilworth's Theorem

Proof. (Perles, 1963).

We use induction on the size of the poset P.

- True when $|P| = 1$.
- Assume the theorem is true when $|P| \leq k$, then consider a poset P with $|P| = k + 1$, then for each maximal antichain A, define the downset of A

$$D(A) := \{x \mid x < a \text{ for some } a \in A\}$$

and the upset of A

$$U(A) := \{x \mid x > a \text{ for some } a \in A\}$$

Dilworth's Theorem

Proof (Cont.)

Case I. Assume there exists a maximum antichain A with $D(A) \neq \varnothing$ and $U(A) \neq \varnothing$.
Claim: $\{A, D(A), U(A)\}$ form a partition of P.
It suffices to show that $D(A) \cap U(A) = \varnothing$. Indeed, otherwise let $x \in D(A) \cap U(A)$, then $\exists y \in A$ with $x < y$, and $\exists z \in A$ with $z < x$, resulting $y, z \in A$ comparable, contradiction.
Let $A = \{a_1, \ldots, a_w\}$, note that $|A \cup D(A)| \leq k$ and $|A \cup U(A)| \leq k$, thus by induction hypothesis, we obtain a chain partition $\{D_1, \ldots, D_w\}$ of $A \cup D(A)$ with maximal elements $a_1, \ldots, a_w$.
Similarly we can obtain a chain partition $\{U_1, \ldots, U_w\}$ of $A \cup U(A)$ with minimal elements $a_1, \ldots, a_w$.
Glue the chains resepctively, we have a chain partition $\{D_1 \cup U_1, \ldots, D_w \cup U_w\}$ of P.

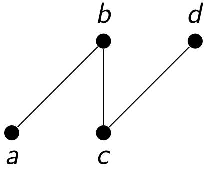
Dilworth's Theorem

Proof (Cont.)

Case II. Otherwise for every maximum antichain A, either $D(A)$ or $U(A)$ is empty. (Or equivalently, either $D(A) \cup A = P$ or $U(A) \cup A = P$ for every maximum antichain A). Hence each maximum antichain is either the set of minimal or maximal elements of P.

Choose $x \in \text{Min}(P)$ and $y \in \text{Max}(P)$ with $x \leq y$ (Possibly $x = y$), then $\{x, y\}$ is a chain.

Now $|P - \{x, y\}| \leq k$ and $P - \{x, y\}$ is of width $w - 1$ (since each antichain of size k contains x or y), hence by induction hypothesis, $P - \{x, y\}$ can be partitioned into $w - 1$ chains. Add chain $\{x, y\}$ to obtained the w-chain partition of P. $\square$



An Application of Dilworth's Theorem

Theorem (Erdős–Szekeres, 1935)

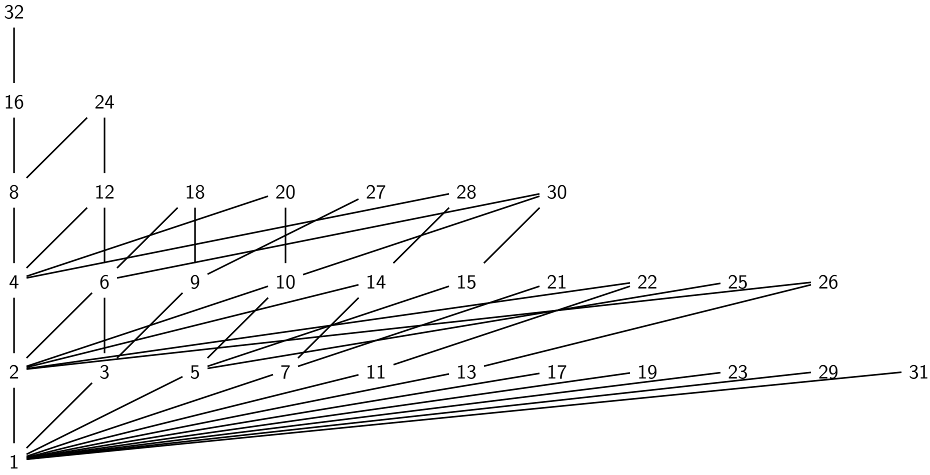
*Let $A = (a_1, ..., a_n)$ be a sequence of n **different** real numbers. If $n \geq sr + 1$ then either A has an increasing subsequence of $s + 1$ terms or a decreasing subsequence of $r + 1$ terms (or both).*

Proof by Dilworth's Theorem.

Define the partial order $\preceq$ on A by $a_i \preceq a_j$ iff $a_i \leq a_j$ and $i \leq j$. (Check it!)
Then we can observe that an increasing subsequence of A corresponds to a chain in $(A, \preceq)$, and an decreasing subsequence in A corresponds to an antichain in $(A, \preceq)$.
Assume that there is no decreasing subsequence of length $r + 1$, then by Dilworth's Theorem, the poset $(A, \preceq)$ can be ***partitioned*** into k chains $C_1, \ldots, C_k$, with $k \leq r$. Therefore $|C_1| + \cdots + |C_k| = n \geq sr + 1$.
By pigeonhole principle, there exsits a chain C_j with $|C_j| \geq s + 1$, which corresponds to an increasing subsequence of length at least $s + 1$. □

Divisibility Revisited

Consider the set $[32] = \{1, 2, \ldots, 31, 32\}$, ordered by divisibility.



Several Equivalent Major Theorems in Combinatorics

- Kőnig's Theorem
- Menger's Theorem (1929)
- Max-Flow Min-Cut theorem
- König-Egerváry theorem (1931)
- Birkhoff-Von Neumann Theorem (1946)
- ***Hall's Theorem***
- ***Dilworth's Theorem***

and duality in linear programming.

Table of Contents

Graphs

Definition

A ***graph*** G consists of a set of ***vertices***, denoted by $V(G)$, a set of edges, denoted by $E(G)$, and a relation called ***incidence*** so that each edge is incident with either one or two vertices, called ***ends*** (or ***endpoints***). For convenience, we sometimes write $G = (V, E)$ to indicate that G is a graph with ***vertex set*** V and ***edge set*** E.

Definition

Two distinct vertices u, v in a graph G are ***adjacent*** if there is an edge with ends u, v. We also call u, v neighbors in G.

Remark

- Vertices are also called nodes, points, locations, stations, etc.
- Edges are also called arcs, lines, links, pipes, connectors, etc.

Loops, Parallel Edges, and Simple Graphs

Definition

An edge with just one end is called a ***loop***. Two distinct edges with the same ends are are ***parallel*** (called "parallel edges" or "multiple edges"). A graph without loops or parallel edges is called ***simple***.

Remark

We specify a simple graph (V, E) by its ***vertex set*** V, and ***edge set*** E, where $E \subset \binom{V}{2}$. We write $e = uv$ or $e = vu$ for an edge $e \in E$ with ends $u, v \in V$. (That is, $e = \{u, v\}$.)

Isomorphism

Definition

An ***isomorphism*** from a simple graph G to a simple graph H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ iff $f(u)f(v) \in E(H)$. We say "G is ***isomorphic*** to H", denoted $G \cong H$, if there is an isomorphism from G to H.

Remark

The ***relation isomorphism***, consisting of the set of ordered pairs (G, H) such that G is isomorphic to H is an equivalence relation on the class of simple graphs.

Representing Graph

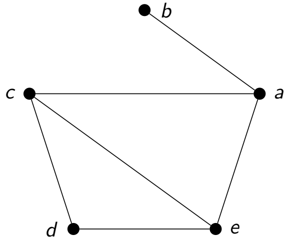
Example

- By specifying the vertex and edge sets of the graph.
- By using database structure.
- By showing a “drawing” of the graph.

Adjacency Tables

An adjacency table lists all the vertices of the graph and the vertices adjacent to them. Consider $G = (V, E)$ with $V = \{a, b, c, d, e\}$ and $E = \{\{a, b\}, \{a, c\}, \{a, e\}, \{c, d\}, \{c, e\}, \{d, e\}\}$.

Vertex	Adjacent Vertices
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d



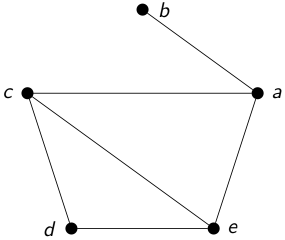
Using Graphs as Models

Example

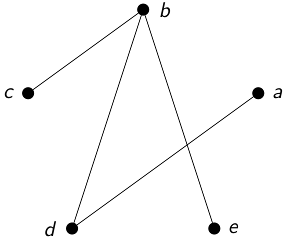
- Acquaintance relations.

Definition

The ***complement*** $\overline{G}$ of a simple graph G is the simple graph with vertex set $V(G)$ defined by $uv \in E(\overline{G})$ iff $uv \notin E(G)$. Note that given graph $G = (V, E)$, we have $\overline{G} = \left(V, \binom{V}{2} - E\right)$.



G



$\overline{G}$

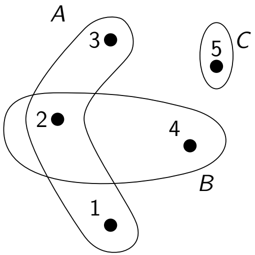
Using Graphs as Models

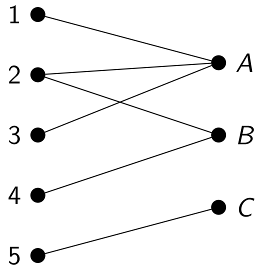
Example

- Job assignments.

Definition

A graph (not necessarily simple) is ***bipartite*** if $V(G)$ is the union of two disjoint (possibly empty) independent sets (i.e., a set of pairwise nonadjacent vertices), called ***partite sets*** of G.





	A	B	C
1	1	0	0
2	1	1	0
3	1	0	0
4	0	1	0
5	0	0	1

Using Graphs as Models

Example

- Maps and coloring.
- Routes in road networks.
- ...

Standard Graphs

Definition

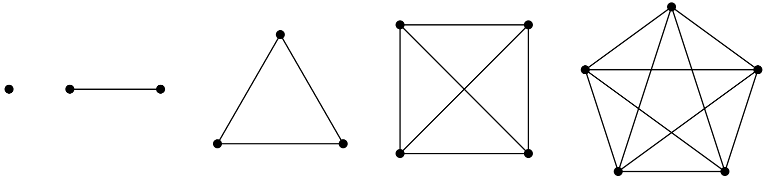
The ***null graph*** is the graph whose vertex set and edge set are empty.

Definition

A graph G is ***complete*** if it is simple and all pairs of distinct vertices are adjacent. A complete graph on n vertices is denoted by K_n.

Definition

A ***clique*** in a graph is a set of pairwise adjacent vertices.

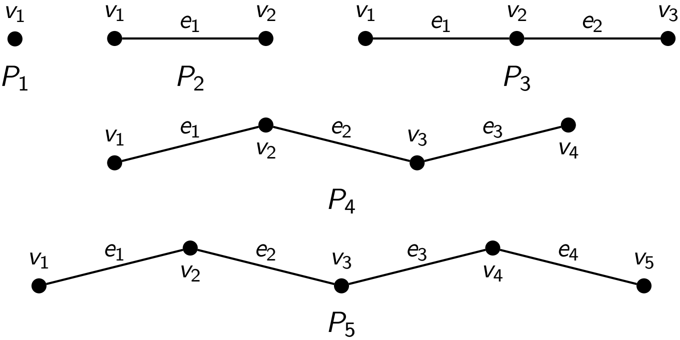


K_1 K_2 K_3 K_4 K_5

Standard Graphs

Definition

A graph G is called a ***path*** if the vertices can be ordered as $v_1, \ldots, v_n$, and edges can be ordered as $e_1, \ldots, e_{n-1}$ such that $e_i = v_i v_{i+1}$, $i = 1, \ldots, n$. A path on n vertices is denoted by P_n.



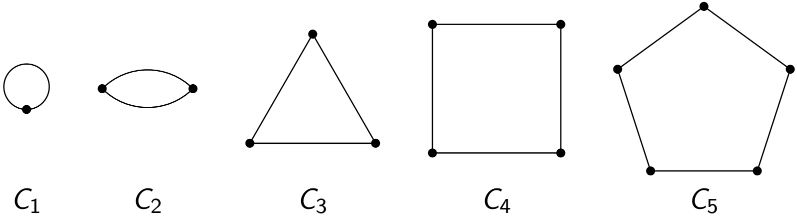
Standard Graphs

Definition

A graph G is a ***cycle*** if $V(G)$ can be ordered as $v_1, \ldots, v_n$, and $E(V)$ can be ordered as $e_1, \ldots, e_n$, where

$$e_i = \begin{cases} v_i v_{i+1}, & 1 \le i \le n-1 \\ v_n v_1, & i = n \end{cases}$$

A cycle on n vertices is denoted by C_n.



Subgraphs

Definition

If G, H have $V(H) \subset V(G)$, and $E(H) \subset E(G)$ with incidence in H the same as G, then H is a ***subgraph*** of G, denoted by $H \subset G$.

Obviously, given $H_1, H_2 \subset G$, then

- $H_1 \cap H_2 \subset G$, with

$$V(H_1 \cap H_2) = V(H_1) \cap V(H_2)$$
$$E(H_1 \cap H_2) = E(H_1) \cap E(H_2)$$

- $H_1 \cup H_2 \subset G$, with

$$V(H_1 \cup H_2) = V(H_1) \cup V(H_2)$$
$$E(H_1 \cup H_2) = E(H_1) \cup E(H_2)$$

Subgraphs

Remark

When we name a graph without naming its vertices, we often mean its isomorphsim class. Technically, "H is a subgraph of G" means that some subgraph of G is isomorphic to H (we say "G contains a **copy** of H"). That is, $H \subset G$ if $\exists G'$ such that $H \cong H' \subset G$.

Example

- C_3 is a subgragh of K_5.
- P_1 is a subgragh of K_5.
- K_5 is a subgragh of K_6.

Degree of Vertices

Definition

The ***degree*** of a vertex v in a graph G, denoted $\deg(v)$ is the number of incident edges (loops counted twice). We write $\deg_G(v)$ in case G is not clear (e.g., when G is a subgraph of some other graph).

Theorem

For all finite graph $G = (V, E)$,

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

Corollary (Handshaking lemma/degree sum formula)

Every graph has an even number of odd degree vertices.

Degree of Vertices

Proof.

By double counting.

$$\begin{aligned}
2|E| &= \sum_{e \in E} (\text{number of vertices } v \text{ incident to } e) \\
&= \sum_{e \in E} \sum_{v \in V} \begin{cases} 1, & \text{if } e \text{ is incident to } v \\ 0, & \text{o/w} \end{cases} \\
&= \sum_{v \in V} \sum_{e \in E} \begin{cases} 1, & \text{if } e \text{ is incident to } v \\ 0, & \text{o/w} \end{cases} \\
&= \sum_{v \in V} \deg(v)
\end{aligned}$$

$\square$

Table of Contents

Walks

Definition

A **walk** W in a graph G is a sequence $v_0, e_1, v_1, \dots, e_n, v_n$ such that every e_i has ends v_{i-1} and v_i. If $v_0 = v_n$, we say that W is closed.

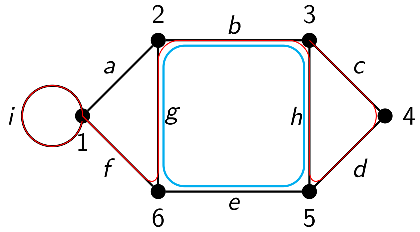
Definition

The **length** of a walk, path, or cycle is its number of edges.

Remark

- A walk is **NOT** a graph in general.
- A path is a graph.
- If $v_0, \dots, v_n$ in a walk are distinct, we also call this walk a path.
- A walk with only 1 vertex has length 0.

Walks



Example

- $W = 3, c, 4, d, 5, h, 3, b, 2, g, 6, f, 1, i, 1$. W is NOT a closed walk (b/c 3 is not the same vertex as 1). The length of W is 7.
- $W' = 3, b, 2, g, 6, e, 5, h, 3$. W' is a closed walk, with with length 4. Note that $W' \cong C_4$.

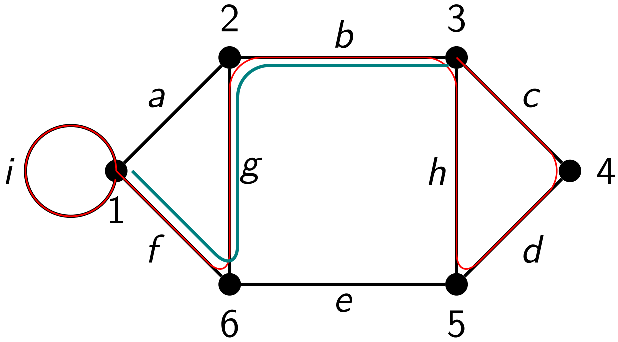
Connected Graph

Definition

A graph G is **connected** if for all $u, v \in V(G)$, there is a walk from u to v (also called a u, v-walk). Otherwise, G is **disconnected**.

Theorem

If there is a walk from u to v, then there is a path from u to v.



Connected Graph

Proof.

Claim: The path from u to v is the shortest walk from u to v (i.e., the walk of minimum length.)

Indeed. Let W be a walk of minimum length from u to v, say $v_0 e_1 v_1 e_2 v_2 \cdots e_n v_n$, with $u = v_0$ and $v = v_n$.

Suppose this is is **NOT** a path, then there exists $v_i = v_j$ such that $0 \leq i < j \leq n$. Therefore $v_0 e_1 v_1 \cdots v_i e_{j+1} v_{j+1} \cdots e_n v_n$ is a shorter walk from u to v, which is a contradiction. $\square$

Connected Graph

Theorem

G is disconnected iff there is a partition $\{X, Y\}$ of $V(G)$ such that no edge has an end in X and an end in Y.

Proof.

($\Leftarrow$) True by definition of connectivity.

($\Rightarrow$) Choose $x, y \in V(G)$ such that no walk from x to y exists, define

$$X := \{z \mid \exists \text{ a walk from } x \text{ to } z\}$$
$$Y := V(G) \setminus X$$

Claim: no edge has an end in X and an end in Y, which is obvious. $\square$

Connected Graph

Theorem

Given $H_1, H_2 \subset G$, H_1, H_2 connected graphs, and $V(H_1) \cap V(H_2) \neq \varnothing$, then $H_1 \cup H_2$ is connected.

Proof.

Let $u, v \in V(H_1 \cup H_2)$. Choose $w \in V(H_1 \cap H_2)$ $(\neq \varnothing)$, note that u, v is either in H_1 or H_2, w.l.o.g., let $u \in V(H_1)$, $v \in V(H_2)$. For $i = 1, 2$, H_i is connected, so there is a u, w-walk W_i. Now concatenate W_1 and W_2, we have a u, v-walk. Since u, v are arbitrary, therefore $H_1 \cup H_2$ is connected. □

A Result From Analysis (iii)⇒(ii)⇒(i)⇒(iii)

Let U be an open subset of a normed space over $\mathbb{R}$, TFAE,

(i) U is connected.

(ii) Any two points of U can be joined by a path in U (path connected).

(iii) Any two points of U can be joined by a polygonal path in U.

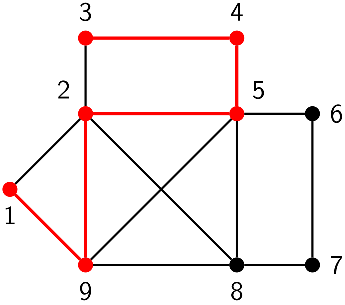
Connected Graph

Definition

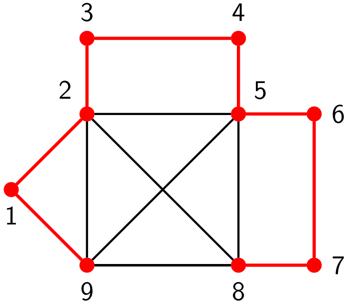
A maximal connected subgraph of G is a subgraph that is connected and is ***not*** contained in any other connected subgraph of G.

Remark

A path/subgraph in G is maximal if it cannot be enlarged.



a path that is maximal



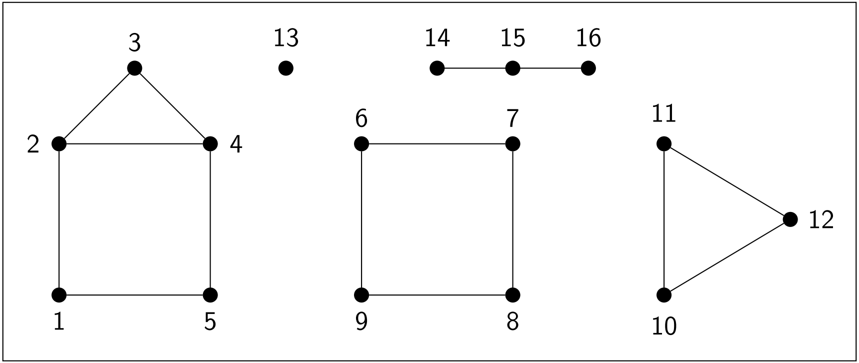
a path that is both maximal and maximum

Connected Graph

Definition

A **component** of a graph G is a maximal non-empty connected subgraph of G.
The number of components of G is denoted $\mathrm{comp}(G)$.

Example



$$\mathrm{comp}(G) = 5$$

Connected Graph

Theorem

Every vertex is in a ***unique*** *component.*

Proof.

Let $v \in V(G)$. Note that v is in a connected subgraph $(\{v\}, \varnothing)$, which consists of only v and no other vertices or edges. If H_1 and H_2 are connected subgraphs containing v, then $H_1 \cap H_2 \neq \varnothing$, thus $H_1 \cup H_2$ is connected. Therefore v is in a unique component. $\square$

Remark

- Components are pairwise disjoint;
- No two components share a vertex;
- Adding an edge with endpoints in distinct components combine the two components into one.
- Adding/Deleting an edge decreases/increases the number of components by at most 1.

Connected Graph

Deleting Edges

Given graph G, $S \subset E(G)$, then $G - S$ is the graph obtained from G by deleting S.

Deleting Vertices

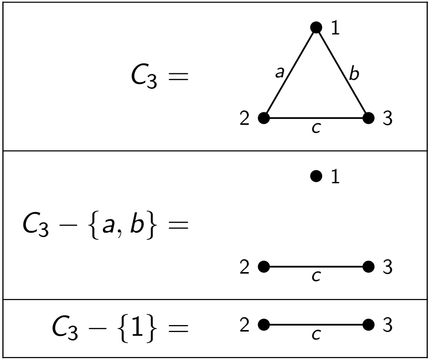
Given graph G, $X \subset V(G)$, then $G - X$ is the graph obtained from G by deleting every vertex in X and every edge incident to a vertex in X.

Notation

If $e \in E(G)$ or $v \in V(G)$, we define

- $G - e := G - \{e\}$;
- $G - v := G - \{v\}$.

For example,
$G - v - w = G - \{v, w\}$.



Connected Graph

Definition

An edge $e \in E(G)$ is called a ***cut-edge*** or ***bridge*** if no cycle contains e.

Theorem

Given graph G and $e \in E(G)$, then

- *either e is a cut-edge and* $\text{comp}(G - e) = \text{comp}(G) + 1$;
- *or e is NOT a cut-edge and* $\text{comp}(G - e) = \text{comp}(G)$.

Proof.

Let u, v be the ends of e ($u = v$ if e is a loop). Note that G has a cycle containing e, iff $G - e$ contains a path from u to v, iff u, v are in the same component of $G - e$. Now

- If u, v are in the same component H of $G - e$, then $H + e$ is a component of G, so $\text{comp}(G - e) = \text{comp}(G)$.
- If u, v are in distinct components, say H_1, H_2 of $G - e$, then $H_1 \cup H_2 + e$ is a component of G, so $\text{comp}(G - e) = \text{comp}(G) + 1$. $\square$

Connected Graph

Definition

A vertex $v \in V(G)$ is called a ***cut-vertex*** whose deletion increases the number of components.

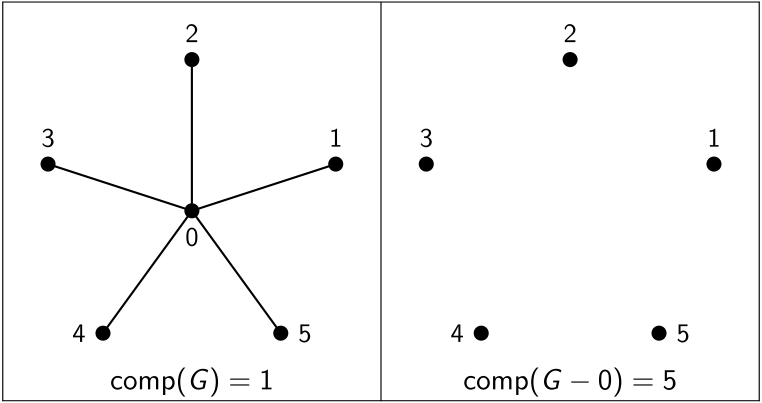


Table of Contents

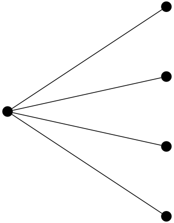
Bipartition

Definition

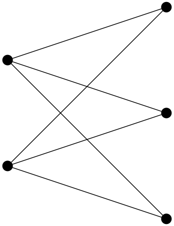
A ***bipartition*** of a graph G is a pair (A, B) where $A, B \subset V(G)$ with $A \cap B = \varnothing$, $A \cup B = V(G)$ such that every edge has an end in A and an end in B. G is ***bipartite*** if it admits a bipartition.

Definition

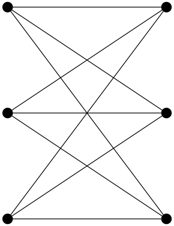
A ***complete bipartite graph*** or ***biclique***, denoted $K_{m,n}$, is a simple bipartite graph with bipartition (A, B) with $|A| = m$ and $|B| = n$ such that every vertex in A is adjacent to every vertex in B.



$K_{1,4}$



$K_{2,3}$



$K_{3,3}$

Bipartition

Theorem

For every graph G, TFAE

(i) *G is bipartite.*

(ii) *G has no cycle of odd length.*

(iii) *G has no closed walk of odd length.*

Proof.

(i)$\Rightarrow$(ii): Assume that $G = (A \cup B, E)$ is bipartite and let $C \subset G$ be a cycle. Then every other vertex of C is in A and every other vertex is in B, hence C must have even length. (It takes even number of steps in a bipartite graph to return to the starting point.)

Bipartition

Proof (Cont.)

(ii)⇒(iii): We show the contrapositive, i.e., $\neg$(iii) $\Rightarrow \neg$(ii). Let G have a closed walk of odd length, and choose such a walk $v_0, e_1, v_1, \ldots, v_n$ of minimum length. Claim: this walk is a cycle (of odd length).
If not, suppose there exist $1 \leq i < j \leq n$ with $v_i = v_j$, then

- either $j - i$ is odd and $v_i, e_i, \ldots, v_j$ is a shorter closed walk of odd length,
- or $j - i$ is even and $v_0, e_1 \ldots v_i, e_{j+1}, v_{j+1}, \ldots, v_n$ is a shorter closed walk of odd length.

It follows that $v_1, \ldots, v_n$ must be distinct $(v_0 = v_n)$, hence $(\{v_1, \ldots, v_n\}, \{e_1, \ldots, e_n\})$ is an odd cycle.

Bipartition

Proof (Cont.)

(iii)⇒(i): Let G be a graph with no closed walk of odd length, w.l.o.g., we may assume that G is connected. Choose a "base point" $u \in V(G)$, observe that for every vertex $v \in V(G)$,

- either all u, v-walks have even length,
- or all all u, v-walks have odd length.

(Note that otherwise can concatenate and odd and an even walk to form a closed walk of odd length.) Now define

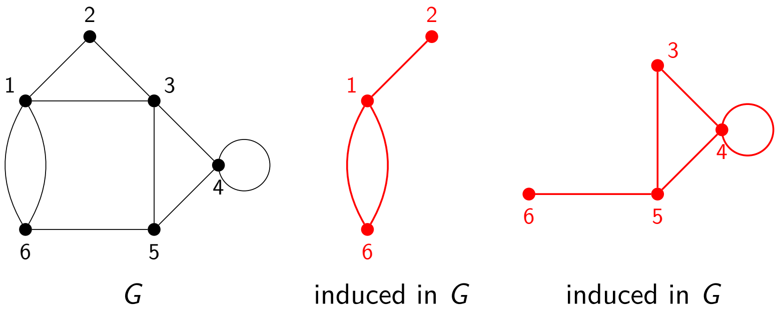
$$A := \{v \in V(G) \mid \exists u, v\text{-walk of even length}\}$$
$$B := \{v \in V(G) \mid \exists u, v\text{-walk of odd length}\}$$

It follows that $A \cap B = \varnothing$. Since G is connected, we have $A \cup B = V(G)$. It follows that (A, B) is a bipartition of G, hence (i) is satisfied. □

Induced Subgraph

Definition

A subgraph $H \subset G$ is **induced** if every edge of G with both ends in $V(H)$ is in $E(H)$. Equivalently, H is induced if $H = G - (V(G) \setminus V(H))$.

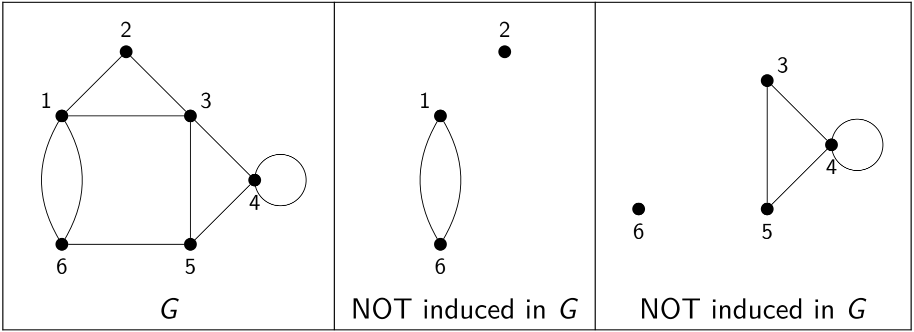


Remark

- An ***induced path*** is sometimes called a ***snake***.
- An ***induced cycle*** is sometimes called a ***chordless cycle*** or a ***hole***.

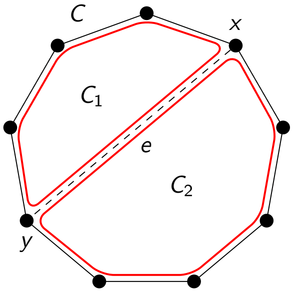
Induced Subgraph

NOT induced subgraph



G | NOT induced in G | NOT induced in G

Bipartition



Theorem

For every graph G, TFAE

(i) *G is bipartite.*

(ii) *G has no cycle of odd length.*

(iii) *G has no closed walk of odd length.*

(iiii) *G has no induced cycle of odd length.*

Proof.

(ii)⇒(iiii). Immediate.

(iiii)⇒(ii). We show the contrapositive, i.e., $\neg$(ii) $\Rightarrow$ $\neg$(iiii). Suppose G has a cycle of odd length, choose a shortest cycle $C \subset G$. Note that C is induced, otherwise $\exists e \in E(G) \setminus E(C)$, with ends x, y. But now either C_1 or C_2 is is an odd cycle of shorter length, contradiction. □

Remark

The ***girth*** of a graph with a cycle is the length of its shortest cycle. A graph with no cycle has infinite girth.

Table of Contents

Matching

Definition

A **matching** in a graph $G = (V, E)$ is a subset of edges M such that M does not contain a loop and no two edges in M are incident with a common vertex. (i.e., the graph (V, M) has all vertices of degree < 2)

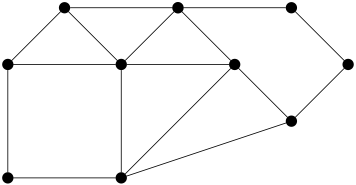
Definition

- A matching M is maximal if there is no matching M' such that $M \subsetneq M'$.
- A matching M is maximum if there is no matching M' such that $|M| < |M'|$.
- A ***perfect matching*** is a matching M such that every vertex of G is incident with an edge in M.

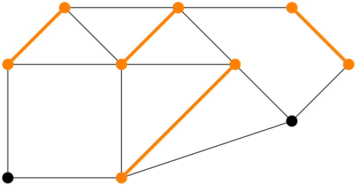
Example

- $K_{n,n}$ has $n!$ perfect matchings.
- K_{2n+1} has 0 perfect matchings.
- K_{2n} has $(2n-1)(2n-3)\cdots(3)(1) = (2n-1)!!$ perfect matchings.

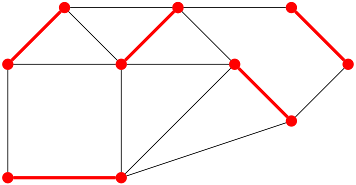
Matching



G



The matching M is maximal in G

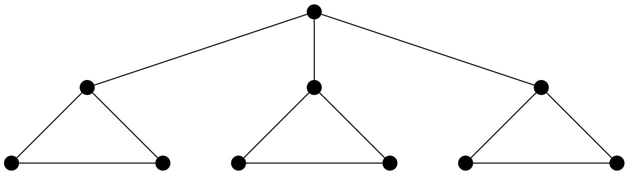


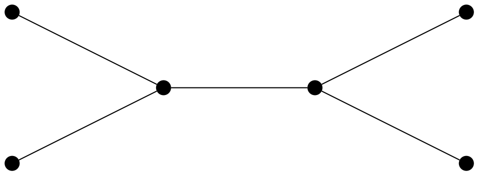
The matching M' is maximal, maximum, and perfect in G

Matching

Remark

A necessary but not sufficent condition for a graph G to have a perfect matching is that $|V(G)|$ is even.





Hall's Theorem

Definition

If $X \subset V(G)$, the ***neighbors*** of X is

$$N(X) \coloneqq \{v \in V(G) \setminus X \mid v \text{ is adjacent to a vertex in } X\}$$

For simplicity, we write $N(x) \coloneqq N(\{x\})$.

Definition

The edges $S \subset E(G)$ ***covers*** $X \subset V(G)$ if every $x \in X$ is incident to some $e \in S$.

Definition

The vertices $X \subset V(G)$ ***covers*** $S \subset E(G)$ if every $e \in S$ is incident to some $v \in X$.

Hall's Theorem

Theorem (Hall)

Let G be a finite bipartite graph with bipartition (A, B). There exists a matching covering A iff there does not exist $X \subset A$ with $|N(X)| < |X|$.

Remark

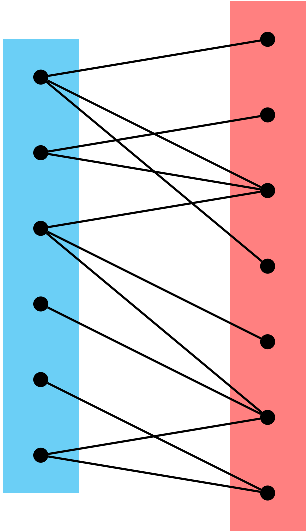
The condition

$$\nexists X \subseteq A \text{ with } |N(X)| < |X|$$

is equivalent to

$$|N(X)| \geq |X| \; \forall X \subseteq A$$

This is called the ***Hall's condition***.



Hall's Theorem

TONCAS

It is straightforward that Hall's conditions is necessary, Hall's theorem states that it turns out to be sufficient as well. The phenomenon is in fact prevalent, and is called ***TONCAS (The Obvious Necessary Condition is Also Sufficient)*** in combinatorics.

The matching covering A induces an injection $f : A \to B$. The Cantor-Bernstein Theorem can be reformulated as

Theorem

If A and B are subsets of the two respective sides of a bipartite graph, and if there exist two matchings covering A and B respectively, then there is a matching covering $A \cup B$.

Note that the trivial generalization of Hall's therem to infinite graph is not true.

Hall's Theorem

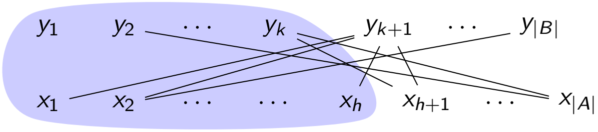
Proof via Dilworth's Theorem.

Necessity. Immediate by pigeonhole principle.

Sufficiency. Let $G = (A \cup B, E)$ be a bipartite graph satisfying Hall's condition that $|N(X)| \geq |X|$ for all $X \subset A$. Define a poset $(P, \leq)$ by letting $P = A \cup B$, and $x < y$ if $x \in A$, $y \in B$, and $xy \in E$. Suppose that the largest antichain is $S = \{x_1, \ldots, x_h, y_1, \ldots, y_k\}$, then

$$N(\{x_1, \ldots, x_h\}) \subset B \setminus \{y_1, \ldots, y_k\}$$

(for otherwise S would not be an antichain if $y \in \{y_1, \ldots, y_k\}$ were the neighbor of some $x \in \{x_1, \ldots, x_h\}$.) Thus Hall's condition implies $|B| - k \geq h$, i.e., $|B| \geq k + h$.



Hall's Theorem

Proof (Cont.)

By Dilworth's theorem, P can be partitioned into $k+h$ chains, denote the matching by M, then

$$|M| + (|A| - |M|) + (|B| - |M|) = k + h \leq |B|$$

that is,

$$|A| + |B| - |M| \leq |B|$$

thus

$$|M| \geq |A|$$

i.e., there is a matching M covering A. $\square$

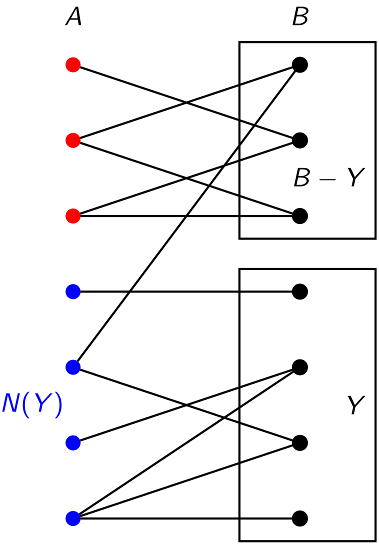
Hall's Theorem

Theorem (Hall, balanced version, see Lovász)

Let G be a finite bipartite graph with bipartition (A, B), then G has a perfect matching iff $|A| = |B|$ and $|N(X)| \geq |X|$ for all $X \subset A$.

Proof. (sufficiency).

First we show that Hall's condition is symmetric, i.e., $|N(Y)| \geq |Y|$ for all $Y \subset B$. Indeed, take $Y \subset B$. Note that $N(A - N(Y)) \subset B - Y$, thus $|N(A - N(Y))| \leq |B - Y|$. By Hall's condition,
$|A - N(Y)| \leq |N(A - N(Y))| \leq |B - Y|$, thus $|A| - |N(Y)| \leq |B| - |Y|$, and it follows that $N(Y) \geq |Y|$.



Hall's Theorem

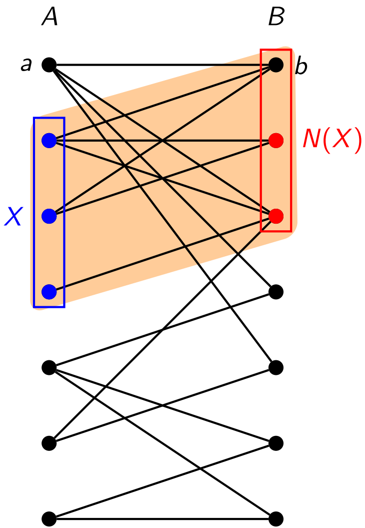
Proof. (sufficiency, cont.)

We proceed by induction on $|A|$.
Base case: $|A| = |B| = 1$. Trivial.
Inductive case: Assume the IH that a perfect matching exists for $|A| = |B| < n$. Let $|A| = |B| = n$, take $a \in A$ and $b \in B$ that are connected by an edge. If $G - \{a, b\}$ satisfies Hall's condition, we are done.

Hall's Theorem

Proof (Cont.)

Otherwise, we can find a subset $X \subset A - a$ such that $|N(X) - b| < |X|$, and thus $|N(X)| = |X|$. Let H and H' be the subgraphs induced by $X \cup N(X)$ and $(A - X) \cup (B - N(X))$, respectively.
Note that both H and H' are balanced bipartite graphs (of smaller size). Now H satisfies Hall's condition by restriction, and H' satisfies Hall's condition by argument prior to the induction. That is, $N(B - N(X)) \subset A - X$ and satisfies Hall's condition. This completes the proof. $\square$



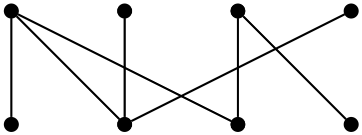
Kőnig-Egerváry Theorem

Definition

A **vertex cover** of a graph G is a set $X \subset V(G)$ if every $e \in E(G)$ is incident with a vertex in X. The vertices in X ***cover*** $E(G)$.

Remark

The size of the smallest vertex cover is denoted $\beta(G)$, and the size of the largest matching is denoted $\alpha'(G)$.



Kőnig-Egerváry Theorem

Theorem (Kőnig-Egerváry)

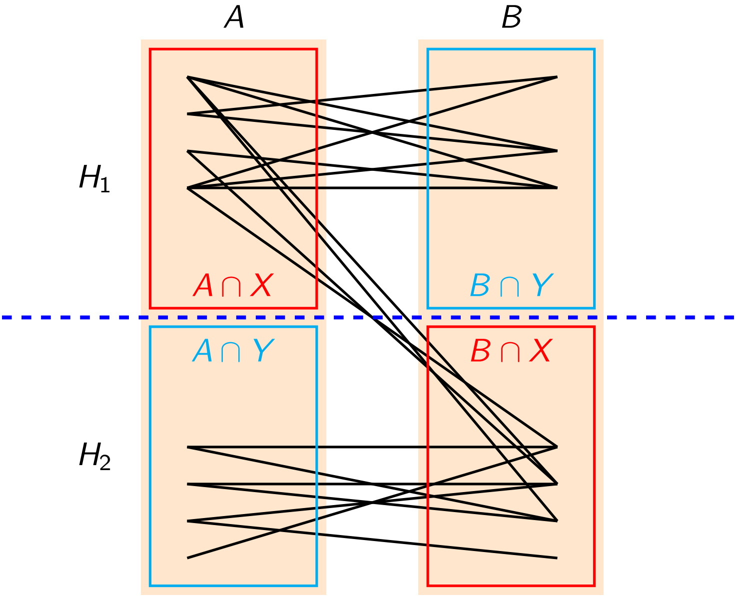
Given a finite bipartite graph G, $\alpha'(G) = \beta(G)$.

Proof.

First of all, it is clear that $\alpha'(G) \leq \beta(G)$.
For the other direction. Let (A, B) be a bipartition of G, and X a vertex cover of minimum size, and $Y = V(G) - X$. Let H_1 and H_2 be the subgraphs induced by $(A \cap X) \cup (B \cap Y)$ and $(A \cap Y) \cup (B \cap X)$ respectively. Note that H_1 and H_2 have bipartitions $(A \cap X, B \cap Y)$ and $(A \cap Y, B \cap X)$ respectively, and there is no edge between $A \cap Y$ and $B \cap Y$. Note that there is no edge between $A \cap Y$ and $B \cap Y$ (otherwise that edge is not covered). Now we claim that

- H_1 has a matching covering $A \cap X$, and
- H_2 has a matching covering $B \cap X$.

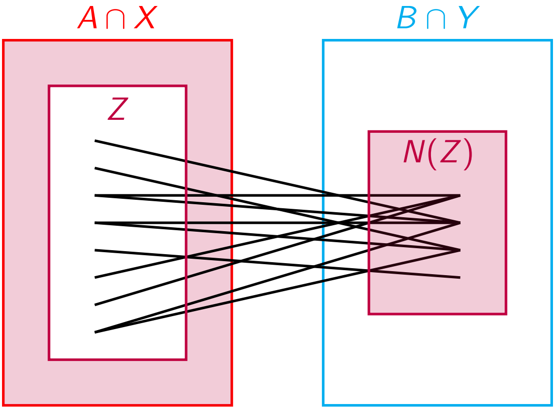
Kőnig-Egerváry Theorem



Kőnig-Egerváry Theorem

Proof (Cont.)

Indeed, consider H_1, suppose there is no such matching, i.e., $|N(Z)| < |Z|$ for some $Z \subset A \cap X$, then we can switch Z and $N(Z)$ for a smaller vertex cover $X' = ((A \cap X) - Z) \cup N(Z)$, a contradiction. It remains to show that X' is indeed a vertex cover. The part for H_2 is similar. $\square$



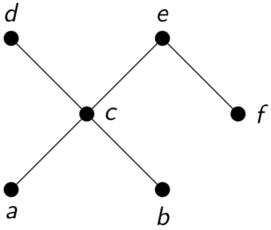
Kőnig-Egerváry implies Dilworth

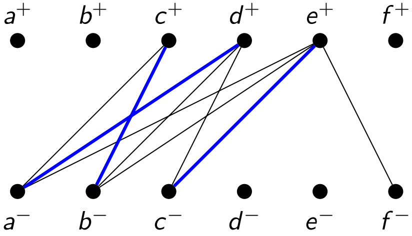
Theorem (Fulkerson, 1956)

Kőnig-Egerváry theorem implies Dilworth theorem (and vice versa).

Proof.

Given a finite poset $P = (X, \leq)$, define a bipartite graph $B_P = (X^- \cup X^+, E)$ with bipartition (X^-, X^+), where X^- and X^+ are copies of X, and $x^- y^+ \in E$ iff $x < y$ in P.





Kőnig-Egerváry implies Dilworth

Proof (Cont.)

For every matching M of B_P (not necessarily maximum or maximal), we can associate with it a chain partition $\mathcal{C}_M$ of P, then $|\mathcal{C}_M| = |X| - |M|$.
Take a minimum vertex cover R of B_P, let

$$A_R := \{x \in X \mid \{x^-, x^+\} \cap R = \varnothing\}$$

then

- A_R is an antichain.
- $\{x^-, x^+\} \not\subset R$ for all $x \in X$.

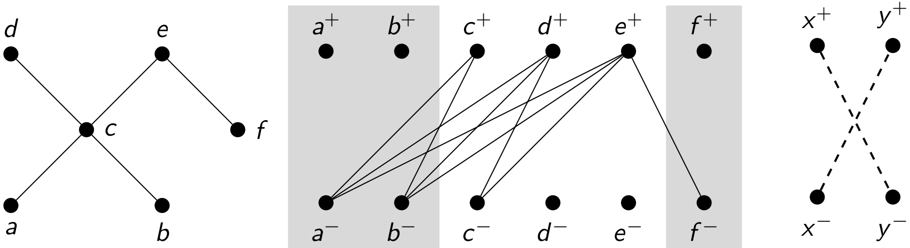
As a consequence, $|A_R| = |X| - |R|$. Take M and R of the same size (in B_P), we get $\mathcal{C}_M$ and A_R of the same size (in P). This is Dilworth theorem. $\square$

Kőnig-Egerváry implies Dilworth

A_R is an antichain

Take any $x, y \in A_R$, we need to show that x and y are incomparable. Indeed, we know that

$$\begin{array}{lll} & \{x^-, x^+, y^-, y^+\} \not\subseteq R & \text{(by definition of } A_R\text{)} \\ \Rightarrow & \{x^- y^+, y^- x^+\} \not\subseteq E & \text{(no edge necessary to cover)} \\ \Rightarrow & x \text{ and } y \text{ are incomparable} & \text{(by definition of } B_P\text{)} \end{array}$$



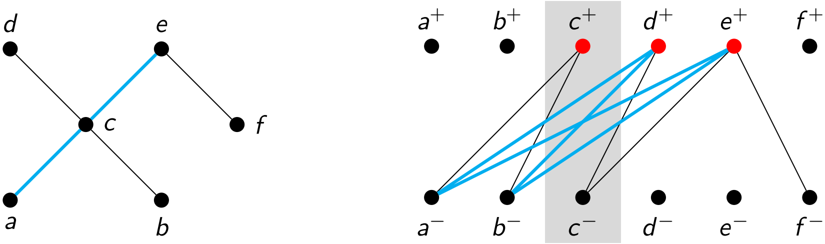
Kőnig-Egerváry implies Dilworth

$\{x^-, x^+\} \not\subset R$ for all $x \in X$

That is, at most one of x^- or x^+ is in the minimum vertex cover R. Indeed, consider $x \in P$, note that the subgraph of B_P with vertices $D(x)^- \cup U(x)^+$ is ***complete bipartite***, thus either its downset $D(x)^-$ xor its upset $U(x)^+$ is in R. Since R is minimum, we can skip x^- or x^+. Now for each $x \in X$, we have two classes,

- either none of $\{x^-, x^+\}$ is in R, i.e., $x \in A_R$.
- or one of $\{x^-, x^+\}$ is in R.

hence $|A_R| = |X| - |R|$.



Graph Homomorphism

Definition

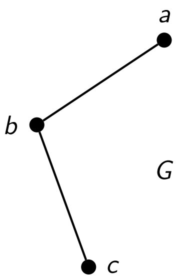
Given graphs G and H, a homomorphism from G to H is a map from $V(G)$ to $V(H)$ which takes edges to edges. That is, a graph homomorphsm $f : G \to H$ is a pair of functions $f = (f_V, f_E)$ such that

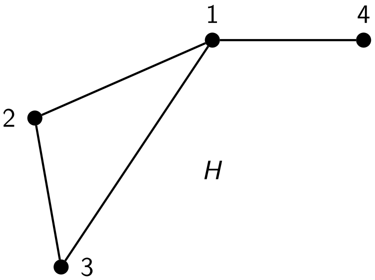
- $f_V : V(G) \to V(H)$, $u \mapsto f_V(u)$
- $f_E : E(G) \to E(H)$, $uw \mapsto f_V(u)f_V(w)$

Remark

The homomorphsm $f : G \to H$ could map a nonedge to a single vertex, a nonedge, or an edge.

Graph Homomorphism



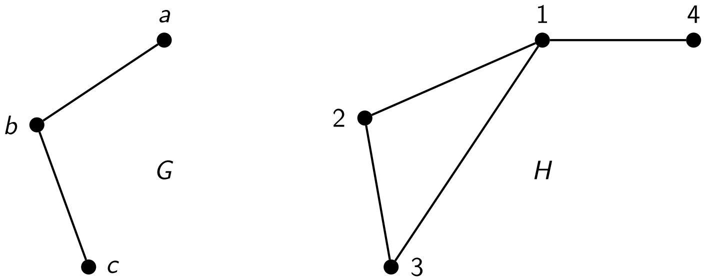


f is a homomorphism

$$f_V : V(G) \to V(H)$$
$$a \mapsto 2$$
$$b \mapsto 1$$
$$c \mapsto 4$$

$$f_E : E(G) \to E(H)$$
$$ab \mapsto 12$$
$$bc \mapsto 14$$

Graph Homomorphism

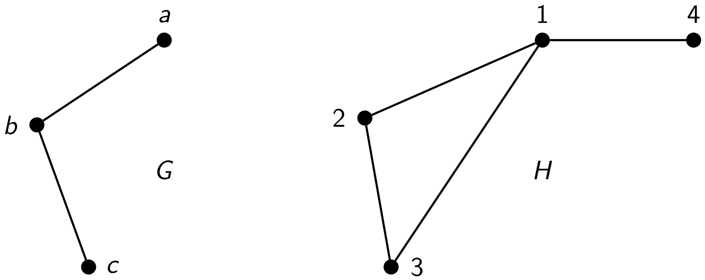


f is a homomorphism

$$\begin{aligned} f_V : V(G) &\to V(H) \\ a &\mapsto 2 \\ b &\mapsto 3 \\ c &\mapsto 2 \end{aligned}$$

$$\begin{aligned} f_E : E(G) &\to E(H) \\ ab &\mapsto 23 \\ bc &\mapsto 23 \end{aligned}$$

Graph Homomorphism

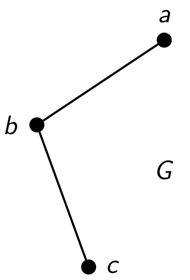


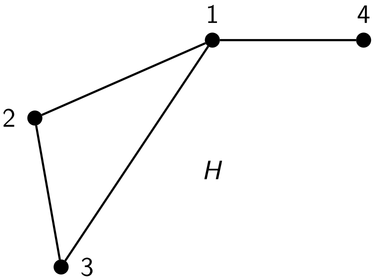
f is NOT a homomorphism

$$\begin{aligned} f_V : V(G) &\to V(H) \\ a &\mapsto 3 \\ b &\mapsto 4 \\ c &\mapsto 1 \end{aligned}$$

$$\begin{aligned} f_E : E(G) &\to E(H) \\ ab &\mapsto 34 \notin E(H) \end{aligned}$$

Graph Homomorphism





f is a NOT homomorphism

$$f_V : V(G) \to V(H)$$
$$a \mapsto 2$$
$$b \mapsto 2$$
$$c \mapsto 1$$

$$f_E : E(G) \to E(H)$$
$$ab \mapsto 22 \notin E(H)$$

Graph Homomorphism

Theorem

A graph G bipartite iff there exists a graph homomorphism $f : G \to K_2$.

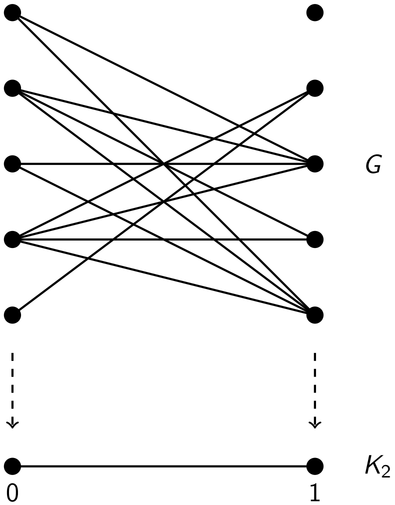
Proof.

($\Leftarrow$) Suppose $f : G \to K_2$ is a graph homomorphism, say we have

$$f_V : V(G) \to V(K_2) = \{0, 1\}$$

Let $A = f_V^{-1}(0)$ and $B = f_V^{-1}(1)$, then we claim $V(G) = (A, B)$ is the desired partition.
(First of all, it is indeed a partition, i.e., the partition of the domain induced by f_V.)
Suppose $\exists uw \in E(G)$ and $u, w \in A$, then

$$f_E(uw) = f_V(u)f_V(w) = 00 \notin E(K_2)$$



Graph Homomorphism

Proof. (Cont).

($\Rightarrow$). Let G be a bipartite graph with bipartition (A, B). We define

$$f_V : V(G) \to V(K_2) = \{0, 1\}$$
$$f_V(u) = \begin{cases} 0, & \text{if } u \in A \\ 1, & \text{if } u \in B \end{cases}$$

Now if $u \in A$, $w \in B$, then

$$f_E(uw) = f_V(u)f_V(w) = 01 \in E(K_2) = \{01, 10\}$$

Therefore $f = (f_V, f_E)$ is the desired homomorphism. $\square$

Remark

A graph G is bipartite if $E(G) = \varnothing$, where f_E is the empty function.

Graph Isomorphism

Definition

Graphs G and H are ***isomorphic*** if there exists $f : G \to H$ and $g : H \to G$ such that $g \circ f = \mathrm{id}_G$ and $f \circ g = \mathrm{id}_H$, where the identity and composition are given by

- Identity: $\mathrm{id}_G := (\mathrm{id}_{V(G)}, \mathrm{id}_{E(G)}) : G \to G$
- Composition: $f \circ g := (f_V \circ g_V, f_E \circ g_E)$

Theorem

The two definitions of graph homomorphisms are equivalent.

Proof.

Check definitions. $\square$

Table of Contents

Trees

Definition

A **forest** is a graph with no cycles. A **tree** is a connected forest.

Theorem

If G is is a forest, then $\mathrm{comp}(G) = |V(G)| - |E(G)|$. *In particular, if T is a tree, then* $|V(T)| = |E(T)| + 1$.

Proof.

Induction on $|E(G)|$.
Base case. If $|E(G)| = 0$, G has no edge, thus $\mathrm{comp}(G) = |V(G)|$.
Inductive case. $|E(G)| > 0$. Choose $e \in E(G)$, since G has no cycle, then e is a cut-edge, thus

$$\begin{aligned} \mathrm{comp}(G) &= \mathrm{comp}(G - e) - 1 \\ &= |V(G - e)| - |E(G - e)| - 1 \qquad \text{(by IH)} \\ &= |V(G)| - |E(G)| \end{aligned}$$

□

Trees

Definition

A **leaf** is a vertex of degree 1.

Theorem

Let T be a tree with $|V(T)| \geq 2$, then T has at least 2 leaves, and if there are only 2 leaves, then T is a path.

Proof.

Note that $2 = 2|V(T)| - 2|E(T)| = \sum_{v \in V(T)}(2 - \deg(v))$. Since T is connected, and $|V(T)| \geq 2$, all vertices have degree > 0. This means there are at least 2 leaves.

Further if there are exactly 2 leaves, then all other vertices have degree 2, therefore T is a path. (Take any maximal path in T, note that any extra edge would increase the degree of interior vertices to 3, or increases the degree of leaves to 2) □

Trees

Lemma
If T is a tree and v is a leaf, then $T - v$ is a tree.

Proof.
Observe that $T - v$ has no cycle and is connected. $\square$

Theorem
If T is a tree, and $u, v \in V(T)$, then there is a unique u, v-path.

Proof.
Induction on $|V(T)|$.
Base case. $|V(T)| = 1$. We have $u = v$.
Inductive case.

- If there is a leaf $w \neq u, v$, apply induction to $T - w$;
- Otherwise T is a path with ends u, v, which is unique by IH. $\square$

Trees

Theorem

Let T be a graph with n vertices. TFAE

(i) *T is a tree;*

(ii) *T contains no cycles, and has $n-1$ edges;*

(iii) *T is connected, and has $n-1$ edges;*

(iv) *T is connected, and each edge is a bridge;*

(v) *any two vertices of T are connected by exactly one path;*

(vi) *T contains no cycles, but the addition of any new edge creates exactly one cycle.*

Table of Contents

Spanning Trees

Definition

If T is a subgraph of a graph G, and T is a tree with $V(T) = V(G)$, then we call T a ***spanning tree*** of G.

Notation

If G is a graph, $H \subset G$ and $e \in E(G)$, then we define $H + e$ to be the subgraph of G obtained from H by adding e and its ends.

Remark

- Removing edges leads to spanning subgraphs.
- Removing vertices leads to induced subgraphs.

Spanning Trees

Theorem

Let G be a finite connected graph with $|V(G)| \geq 2$. If $H \subseteq G$ such that

(i) *either H is minimal such that $V(H) = V(G)$ and H is connected,*

(ii) *or H is maximal such that H has no cycles,*

then H is a spanning tree of G.

Proof.

(i) It suffices to show that H is a tree. Suppose that H has a cycle C, choose $e \in E(C)$. Now $H - e$ is connected (b/c e is not a cut-edge), but this contradicts that H is minimal.

(ii) Note that $V(H) = V(G)$ by maximality of H. It remains to show that H is connected. Suppose not, choose a partition $\{X, Y\}$ of $V(H) = V(G)$ such that no edge of H has one end in X and the other in Y. Choose $e \in E(G)$ such that e has one end in X and the other in Y (b/c G connected), but now $H + e$ contradicts that H is maximal. □

Spanning Trees

Theorem
If $|V(G)| = |E(G)| + 1$, and

(i) *either G has no cycles,*
(ii) *or G is connected,*

then G is a tree.

Proof.

(i) Since G is a forest, then $1 = |V(G)| - |E(G)| = \text{comp}(G)$.
(ii) Choose a spanning tree T of G (possible b/c G connected), then

$$|E(G)| = |V(G)| - 1 = |V(T)| - 1 = |E(T)|$$

thus $G = T$, so G is a tree. $\square$

Remark
Note that (i) and (ii) together implies that G is a tree.

Spanning Trees

Definition

Let G be a graph, and $T \subset G$ a spanning tree of G, $f \in E(G) \setminus E(T)$. If C is a cycle of G such that $C - f \subset T$, we call C a ***fundamental cycle*** of f w.r.t. T.

Observation

For every edge $f \in E(G) \setminus E(T)$, there is a **unique** fundamental cycle of f w.r.t. T.

Spanning Trees

Theorem

Let G be a graph, and $T \subset G$ a spanning tree of G, $f \in E(G) \setminus E(T)$, $e \in E(T)$.

(i) *If e is in the fundamental cycle of f w.r.t. T, then $T - e + f$ is a spanning tree.*

(ii) *If f has one end in each component of $T - e$, then $T - e + f$ is a spanning tree.*

Spanning Trees

Proof.

(i) Note that $T + f$ is connected and e is not a cut-edge of $T + f$, so $T + f - e$ is a connected graph with n vertices and $n - 1$ edges, so it is a tree, and in particular, a spanning tree of G.

(ii) If f has one end in each component of $T - e$, then $T - e + f$ is a forest (why?) with n vertices and $n - 1$ edges, so it is a tree, and in particular, a spanning tree of G. □

Table of Contents

Weighted Graph

Definition

A ***weighted graph*** is a graph G with a weight function $w : E(G) \to \mathbb{R}$. A ***minimum-cost tree*** (or ***minimum-weight spanning tree***) of G is a spanning tree T for which

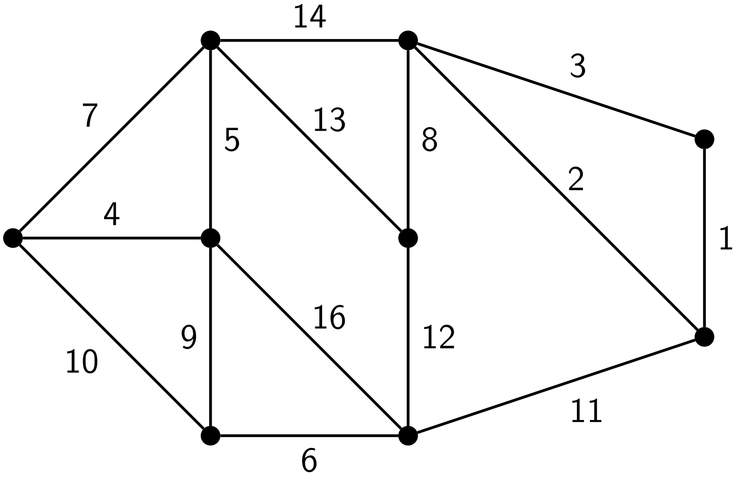
$$\sum_{e \in E(T)} w(e)$$

is minimum.

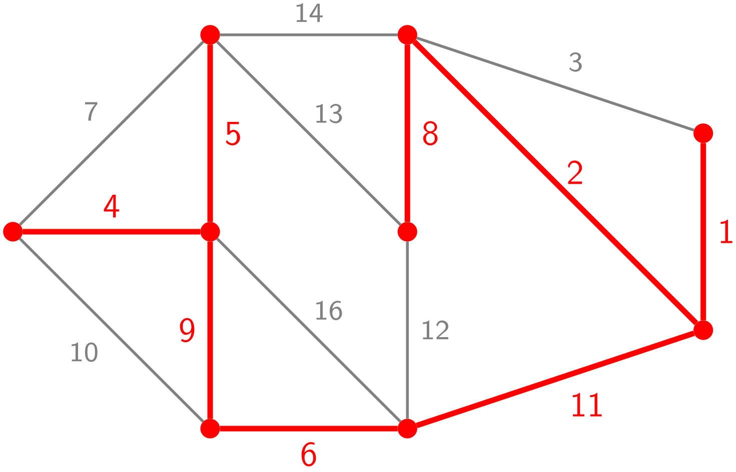
Kruskal's Algorithm

- Input: A connected weighted graph $G = (V, E)$.
- Output: A minimum-cost tree T.
- Procedure: Choose a sequence of edges $e_1, e_2, \ldots, e_m$ according to the rule that e_i is an edge of minimum weight in $E(G) \setminus \{e_1, \ldots, e_{i-1}\}$ so that $\{e_1, \ldots, e_{i-1}\}$ does not contain the edge set of a cycle. When no such edge exists, stop and return the subgraph $T = (V, \{e_1, \ldots, e_m\})$.

Kruskal's Algorithm



Kruskal's Algorithm



Spanning Trees v. Vector Space Bases

Finite dimensional vector spaces

A basis for a finite dimensional vector space is any of the following

- A minimal spanning/generating set.
- A maximal linearly independent set.
- Every element of the vector space is uniquely represented by as a linear combination of the basis vectors.

Finite connected graphs

A spanning tree is any of the following

- A minimal subgraph maintaining the same vertex set and connectedness.
- A maximal subgraph without cycles.
- For any two vertices, there is a unique path between them in the tree.

Table of Contents

Distance Function

Definition

Given graph G, and $u, v \in V(G)$, the distance from u to v, denoted $\text{dist}(u, v)$, is the shortest length of a walk from u to v in G.

Remark

For $u, v, w \in V(G)$, the triangle inequality holds,

$$\text{dist}(u, v) + \text{dist}(v, w) \geq \text{dist}(u, w).$$

Adjacency Matrix

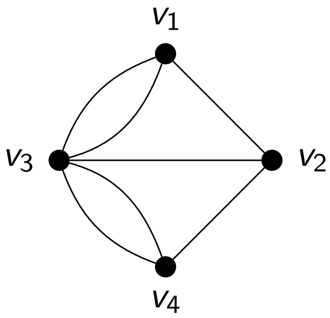
Definition

Let G be a undirected graph with vertices ordered as $v_1, \ldots, v_n$. Define a matrix $A = (a_{ij}) \in M_n(\mathbb{N})$ by setting a_{ij} to be the number of edges between v_i and v_j. In particular, if the graph is simple,

$$(A)_{ij} = \begin{cases} 1, & \text{if } v_i \text{ is adjacent to } v_j \\ 0, & \text{otherwise} \end{cases}$$

The matrix A is called an adjacency matrix for the graph G.

$$A_G = \begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 0 & 1 & 1 \\ 2 & 1 & 0 & 2 \\ 0 & 1 & 2 & 0 \end{bmatrix}$$



Adjacency Matrix

Theorem

Let A be an adjacency matrix for a graph G with vertices ordered as $v_1, \ldots, v_n$, and $K \geq 0$. The for every $v_i, v_j \in V(G)$, the number of walks of length K from v_i to v_j is $(A^K)_{ij}$

Proof. (sketch).

Induction on K. It is clear for $K = 0$. Suppose the result holds for $K \geq 0$, then

$$(A^{K+1})_{ij} = (A^K A)_{ij} = \sum_{\ell=1}^{n} (A^K)_{i\ell} A_{\ell j}$$

The rest follows by noting that

- $(A^K)_{i\ell} A_{\ell j}$ is the number of walks from length $K + 1$ from v_i to v_j that passes v_ℓ right before v_j.
- $(A^K)_{i\ell}$ is the number of walks from v_i to v_j of length K by IH.
- $A_{\ell j}$ is the number of edges from v_ℓ to v_j. $\square$

Adjacency Matrix

Example

Given a graph G with adjacency matrix $A_G \in M_n(\mathbb{N})$, the number of ***closed walks*** of length K is given by $\operatorname{Tr} A_G^K$. If the eigenvalues of A_G are given by $\lambda_1, \ldots, \lambda_n$, then

$$\operatorname{Tr}(A_G)^K = \sum_{j=1}^{n} (A_G^K)_{jj} = \sum_{j=1}^{n} \lambda_j^K$$

Note that the formula also works when A_G is nondiagonalizable.

Weighted Distance

Definition

Given a simple connected graph with weight function $w : E(G) \to \mathbb{R}_{\geq 0}$, the length of a walk $v_1 e_1 v_2 e_2 \cdots e_k v_{k+1}$ is given by

$$w(e_1) + w(e_2) + \cdots + w(e_k).$$

Then the distance from u to v is the length of the shortest walk from u to v.

Definition

Given graph G, and $r \in V(G)$, a tree $T \subset G$ with $r \in V(T)$ is a ***shortest path tree*** or ***shortest path spanning tree*** for r if

$$\mathrm{dist}_G(r, v) = \mathrm{dist}_T(r, v)$$

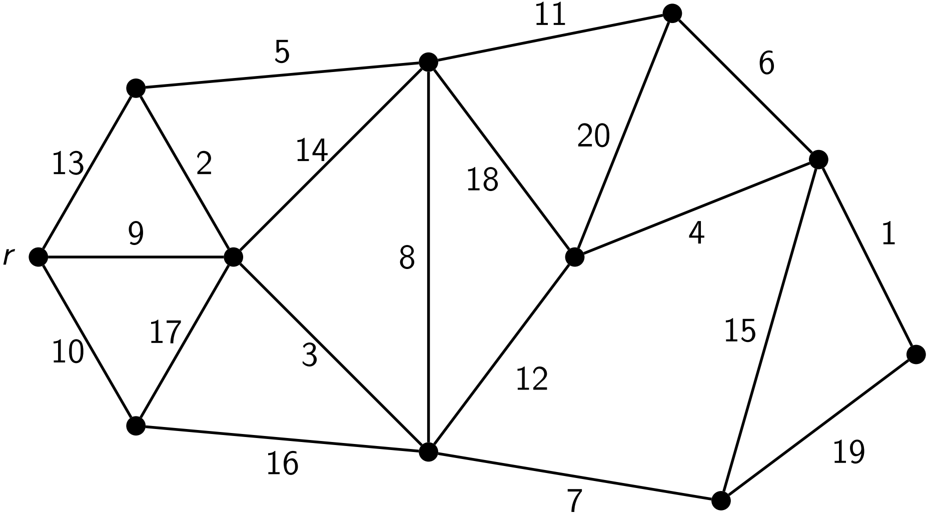
for every $v \in V(T)$.

Dijkstra's Algorithm

Dijkstra's Algorithm

- Input: A simple connected graph $G = (V, E)$ with root vertex r and nonnegative weight function $w : E(G) \to \mathbb{R}_{\geq 0}$.
- Output: A shortest path spanning tree for r.
- Procedure:
  1. $i = 1$. Set T_1 to be the tree consisting of only the root vertex r.
  2. $i \geq 2$. Choose an edge uv such that $u \in V(T_{i-1})$, $v \in V(G) \setminus V(T_{i-1})$, and $\mathrm{dist}_T(r, u) + w(uv)$ is minimum. Let $T_i := T_{i-1} + uv$. If no such choice is possible, return the present tree.

Dijkstra's Algorithm



Dijkstra's Algorithm