Discrete Mathematics and Its Applications

Lecture 7: Graphs: Graph Models and Graph Types

MING GAO

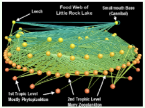
DaSE@ECNU
(for course related communications)
mgao@dase.ecnu.edu.cn

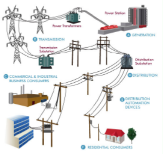
Dec. 25, 2018

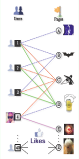
Outline

Graphs - why should we care?

Motivation







Graphs in real world

- “YahooWeb graph”: 1B vertices(Web sites), 6B edges (links)
- Facebook, Twitter, etc: more than 1B users
- Food Web: all biologies, food chain
- Power-grid: vertices (plants or consumers), edges (power lines)
- Airline route: vertices (airports), edges (flights)
- Adoption: users purchase products, adopt services, etc.

Motivation questions

Questions

- What do real graphs look like?
  - What properties of vertices, edges are important to model?
  - What local and global properties are important to measure?
- Are graphs helpful to understand the real world?
  - Social influence
  - Recommendation
  - Information propagation
  - Human behaviors
- Is a sub-graph "normal" (Water army, fraud detection, spam filtering, etc)?
- How to generate realistic graphs?
- How to get a "good" sample of a network?
- How to design an efficient algorithm to handle large-scale graphs?

Graphs

Definition

A **graph** $G = (V, E)$ consists of V , a nonempty set of vertices (or nodes) and E, a set of edges. Each edge has either one or two vertices associated with it, called its endpoints. An edge is said to connect its endpoints.

Graphs

Definition

A **graph** $G = (V, E)$ consists of V , a nonempty set of vertices (or nodes) and E, a set of edges. Each edge has either one or two vertices associated with it, called its endpoints. An edge is said to connect its endpoints.

- The set V of a graph G may be infinite (infinite graph).
- In this book we will usually consider only finite graphs.
- A graph in which each edge connects two different vertices and where no two edges connect the same pair of vertices is called a **simple graph**.
- Graphs that may have multiple edges connecting the same vertices are called **multigraphs**.
- To model this network we need to include edges that connect a vertex to itself, such edges are called **loops**.

Directed graphs and graph terminology

Definition

A **directed graph** (or **digraph**) (V, E) consists of a nonempty set of vertices V and a set of directed edges (or arcs) E. Each directed edge is associated with an ordered pair of vertices. The directed edge associated with the ordered pair (u, v) is said to start at u and end at v.

Directed graphs and graph terminology

Definition

A **directed graph** (or **digraph**) (V, E) consists of a nonempty set of vertices V and a set of directed edges (or arcs) E. Each directed edge is associated with an ordered pair of vertices. The directed edge associated with the ordered pair (u, v) is said to start at u and end at v.

TABLE 1 Graph Terminology.

Type	*Edges*	*Multiple Edges Allowed?*	*Loops Allowed?*
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Simple directed graph	Directed	No	No
Directed multigraph	Directed	Yes	Yes
Mixed graph	Directed and undirected	Yes	Yes

Graph models I

Social networks

Graphs are extensively used to model social structures based on different kinds of relationships between people or groups of people. These social structures are known as **social networks**.

Graph models I

Social networks

Graphs are extensively used to model social structures based on different kinds of relationships between people or groups of people. These social structures are known as **social networks**.

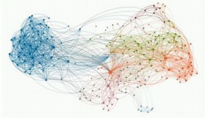
We can use a simple graph to represent whether two people know each other.

Graph models I

Social networks

Graphs are extensively used to model social structures based on different kinds of relationships between people or groups of people. These social structures are known as **social networks**.

We can use a simple graph to represent whether two people know each other.



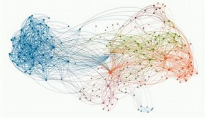
In studies of group behavior, certain people can influence the thinking of others.

Graph models I

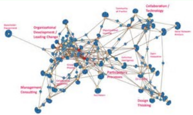
Social networks

Graphs are extensively used to model social structures based on different kinds of relationships between people or groups of people. These social structures are known as **social networks**.

We can use a simple graph to represent whether two people know each other.



In studies of group behavior, certain people can influence the thinking of others.



The graph is to model that two people are related by working together in a particular way.

Graph models II

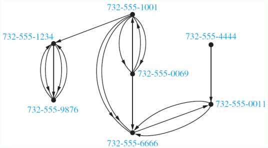
Communication networks

We can model different **communications networks** using vertices to represent devices and edges to represent the particular type of communications links of interest.

Graph models II

Communication networks

We can model different **communications networks** using vertices to represent devices and edges to represent the particular type of communications links of interest.

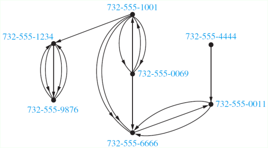


Graphs can be used to model telephone calls made in a network, such as a long distance telephone network.

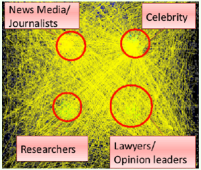
Graph models II

Communication networks

We can model different **communications networks** using vertices to represent devices and edges to represent the particular type of communications links of interest.



Graphs can be used to model telephone calls made in a network, such as a long distance telephone network.



In Sina Weibo, via observing the retweet behavior, Guo Meimei scandal cascades in the network.

Graph models III

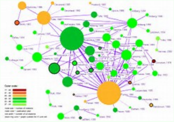
Information networks

Graphs can be used to model various networks that link particular types of information.

Graph models III

Information networks

Graphs can be used to model various networks that link particular types of information.

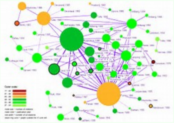


Graphs can be used to represent citations in different types of documents, including academic papers, patents, and legal opinions.

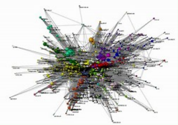
Graph models III

Information networks

Graphs can be used to model various networks that link particular types of information.



Graphs can be used to represent citations in different types of documents, including academic papers, patents, and legal opinions.



The World Wide Web can be modeled as a directed graph where each Web page is represented by a vertex and where an edge starts at the Web page a and ends at the Web page b if there is a link on a pointing to b.

Graph models IV

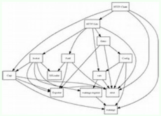
Software design applications

Graph models are useful tools in the design of software. We will briefly describe two of these models here.

Graph models IV

Software design applications

Graph models are useful tools in the design of software. We will briefly describe two of these models here.

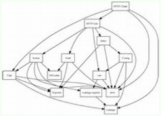


A module dependency graph provides a useful tool for understanding how different modules of a program interact.

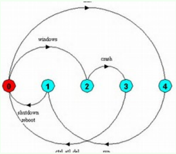
Graph models IV

Software design applications

Graph models are useful tools in the design of software. We will briefly describe two of these models here.



A module dependency graph provides a useful tool for understanding how different modules of a program interact.



Computer programs can be executed more rapidly by executing certain statements concurrently. The dependence of statements can be represented by a directed graph.

Graph models V

Transportation networks

We can use graphs to model many different types of transportation networks, including road, air, and rail networks, as well shipping networks.

Graph models V

Transportation networks

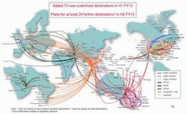
We can use graphs to model many different types of transportation networks, including road, air, and rail networks, as well shipping networks.

The resulting graph will generally be a directed multigraph, as there may be multiple flights from one airport to some other airport during the same day.

Graph models V

Transportation networks

We can use graphs to model many different types of transportation networks, including road, air, and rail networks, as well shipping networks.



The resulting graph will generally be a directed multigraph, as there may be multiple flights from one airport to some other airport during the same day.

In such models, vertices represent intersections and edges represent roads.

Graph models VI

Biological networks

Many aspects of the biological sciences can be modeled using graphs.

Graph models VI

Biological networks

Many aspects of the biological sciences can be modeled using graphs.

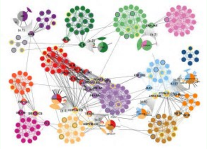
Each species is represented by a vertex. An undirected edge connects two vertices if the two species represented by these vertices compete, i.e., some of the resources they use are the same.

Graph models VI

Biological networks

Many aspects of the biological sciences can be modeled using graphs.

Each species is represented by a vertex. An undirected edge connects two vertices if the two species represented by these vertices compete, i.e., some of the resources they use are the same.



A protein interaction in a living cell occurs when two or more proteins in that cell bind to perform a biological function.

Basic concepts for undirected graphs

Neighbors

Two vertices u and v in an undirected graph G are called **adjacent (or neighbors)** in G if u and v are endpoints of an edge e of G.

Basic concepts for undirected graphs

Neighbors

Two vertices u and v in an undirected graph G are called **adjacent (or neighbors)** in G if u and v are endpoints of an edge e of G.

Neighborhood

The set of all neighbors of a vertex v of $G = (V, E)$, denoted by $N(v)$, is called the **neighborhood** of v. If A is a subset of V , we denote by $N(A)$ the set of all vertices in G that are adjacent to at least one vertex in A. So, $N(A) = \bigcup_{v \in A} N(v)$.

Basic concepts for undirected graphs

Neighbors

Two vertices u and v in an undirected graph G are called **adjacent (or neighbors)** in G if u and v are endpoints of an edge e of G.

Neighborhood

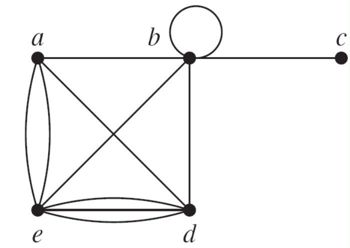
The set of all neighbors of a vertex v of $G = (V, E)$, denoted by $N(v)$, is called the **neighborhood** of v. If A is a subset of V , we denote by $N(A)$ the set of all vertices in G that are adjacent to at least one vertex in A. So, $N(A) = \bigcup_{v \in A} N(v)$.

Degree

The **degree** of a vertex in an undirected graph is # edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex. The degree of the vertex v is denoted by $deg(v)$.

Example

What are the degrees and what are the neighborhoods of the vertices in the graph displayed in the figure?



$N(a) = \{b, d, e\},$ $deg(a) = 4$

$N(b) = \{a, b, c, d, e\},$ $deg(b) = 6$

$N(c) = \{b\},$ $deg(c) = 1$

$N(d) = \{a, b, e\},$ $deg(d) = 5$

$N(e) = \{a, b, d\},.$ $deg(e) = 6.$

Some important conclusions in undirected graphs

The Handshaking theorem

Let $G = (V, E)$ be an undirected graph with m edges. Then

$$2m = \sum_{v \in V} deg(v).$$

(Note that this applies even if multiple edges and loops are present.)

Some important conclusions in undirected graphs

The Handshaking theorem

Let $G = (V, E)$ be an undirected graph with m edges. Then

$$2m = \sum_{v \in V} deg(v).$$

(Note that this applies even if multiple edges and loops are present.)

An undirected graph has an even number of vertices of odd degree.
Proof: Let V_1 and V_2 be the set of vertices of even and odd degrees in an undirected graph $G = (V, E)$ with m edges. Then

$$2m = \sum_{v \in V} deg(v) = \sum_{v \in V_1} deg(v) + \sum_{v \in V_2} deg(v).$$

Some important conclusions in undirected graphs

The Handshaking theorem

Let $G = (V, E)$ be an undirected graph with m edges. Then

$$2m = \sum_{v \in V} deg(v).$$

(Note that this applies even if multiple edges and loops are present.)

An undirected graph has an even number of vertices of odd degree.

Proof: Let V_1 and V_2 be the set of vertices of even and odd degrees in an undirected graph $G = (V, E)$ with m edges. Then

$$2m = \sum_{v \in V} deg(v) = \sum_{v \in V_1} deg(v) + \sum_{v \in V_2} deg(v).$$

Thus, there are an even number of vertices of odd degree.

Basic concepts for undirected graphs

Adjacent

When (u, v) is an edge of the graph G with directed edges, u is said to be adjacent to v and v is said to be adjacent from u. The vertex u is called the initial vertex of (u, v), and v is called the terminal or end vertex of (u, v). The initial vertex and terminal vertex of a loop are the same.

Basic concepts for undirected graphs

Adjacent

When (u, v) is an edge of the graph G with directed edges, u is said to be adjacent to v and v is said to be adjacent from u. The vertex u is called the initial vertex of (u, v), and v is called the terminal or end vertex of (u, v). The initial vertex and terminal vertex of a loop are the same.

Degree

In a graph with directed edges the in-degree of a vertex v, denoted by $deg^{-}(v)$, is # edges with v as their terminal vertex. The out-degree of v, denoted by $deg^{+}(v)$, is # edges with v as their initial vertex. (Note that a loop at a vertex contributes 1 to both the in-degree and the out-degree of this vertex.)

Degrees on directed graphs

Theorem

Let $G = (V, E)$ be a directed graph, Then

$$\sum_{v \in V} deg^{-}(v) = \sum_{v \in V} deg^{+}(v) = |E|.$$

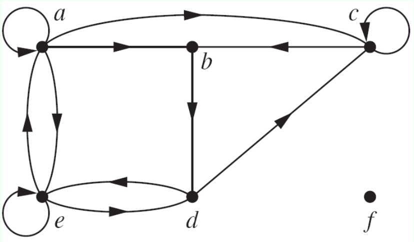
Degrees on directed graphs

Theorem

Let $G = (V, E)$ be a directed graph, Then

$$\sum_{v \in V} deg^-(v) = \sum_{v \in V} deg^+(v) = |E|.$$

What are in-degree and out-degree of the vertices in the graph displayed in the figure?



$deg(a)^+ = 4, deg(a)^- = 2$

$deg(b)^+ = 1, deg(b)^- = 2$

$deg(c)^+ = 2, deg(c)^- = 3$

$deg(d)^+ = 2, deg(d)^- = 2$

$deg(e)^+ = 3, deg(e)^- = 3$

$deg(f)^+ = 0, deg(f)^- = 0$

Complete graphs

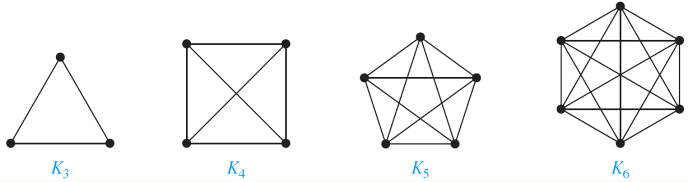
Definition

A **complete graph** on n vertices, denoted by K_n, is a simple graph that contains exactly one edge between each pair of distinct vertices.

Complete graphs

Definition

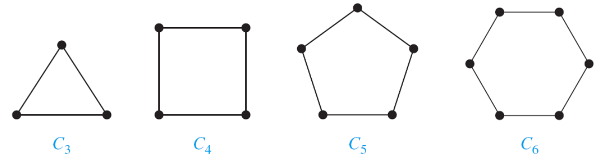
A **complete graph** on n vertices, denoted by K_n, is a simple graph that contains exactly one edge between each pair of distinct vertices.



- A complete graph is also called **clique**.
- A simple graph for which there is at least one pair of distinct vertex not connected by an edge is called non-complete.

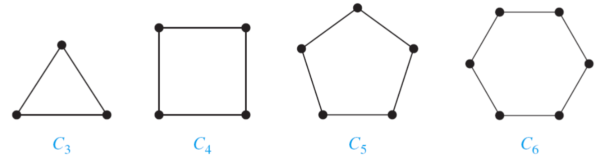
Cycles and wheels

A cycle C_n, $n \geq 3$, consists of n vertices $v_1, v_2, \cdots, v_n$ and edges $\{v_1, v_2\}, \{v_2, v_3\}, \cdots, \{v_{n-1}, v_n\}$, and $\{v_n, v_1\}$.

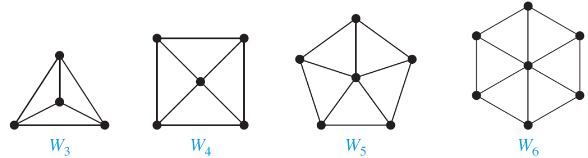


Cycles and wheels

A cycle C_n, $n \geq 3$, consists of n vertices $v_1, v_2, \cdots, v_n$ and edges $\{v_1, v_2\}, \{v_2, v_3\}, \cdots, \{v_{n-1}, v_n\}$, and $\{v_n, v_1\}$.



We obtain a wheel W_n when we add an additional vertex v to a cycle C_n, for $n \geq 3$, and connect v to each of the n vertices in C_n.

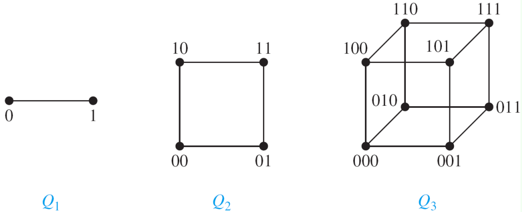


n-Cubes

An n-dimensional hypercube, or n-cube, denoted by Q_n, is a graph that has vertices representing the 2^n bit strings of length n. Two vertices are adjacent if and only if the bit strings that they represent differ in exactly one bit position.

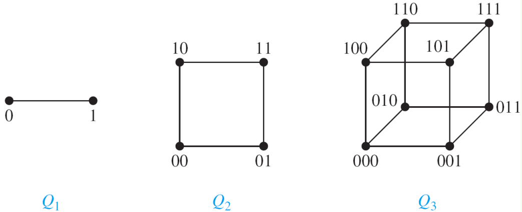
n-Cubes

An n-dimensional hypercube, or n-cube, denoted by Q_n, is a graph that has vertices representing the 2^n bit strings of length n. Two vertices are adjacent if and only if the bit strings that they represent differ in exactly one bit position.



n-Cubes

An n-dimensional hypercube, or n-cube, denoted by Q_n, is a graph that has vertices representing the 2^n bit strings of length n. Two vertices are adjacent if and only if the bit strings that they represent differ in exactly one bit position.



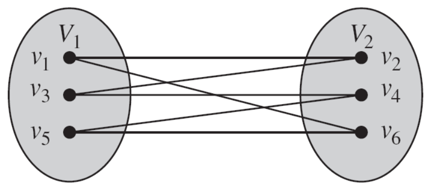
We can construct the $(n+1)$-cube Q_{n+1} from Q_n by making two copies of Q_n, prefacing the labels on the vertices with a 0 in one copy of Q_n and with a 1 in the other copy of Q_n, and adding edges connecting two vertices that have labels differing only in the first bit.

Bipartite graphs

A simple graph G is called **bipartite** if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2). When this condition holds, we call the pair (V_1, V_2) a bipartition of the vertex set V of G.

Bipartite graphs

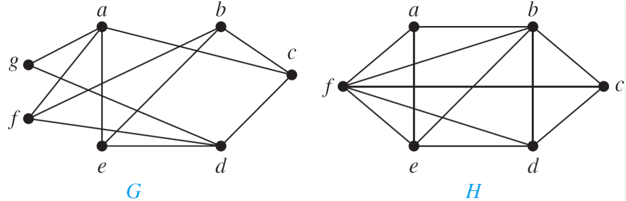
A simple graph G is called **bipartite** if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2). When this condition holds, we call the pair (V_1, V_2) a bipartition of the vertex set V of G.



C_6 is bipartite because its vertex set can be partitioned into the two sets $V_1 = \{v_1, v_3, v_5\}$ and $V_2 = \{v_2, v_4, v_6\}$, and every edge of C_6 connects a vertex in V_1 and a vertex in V_2.

Examples

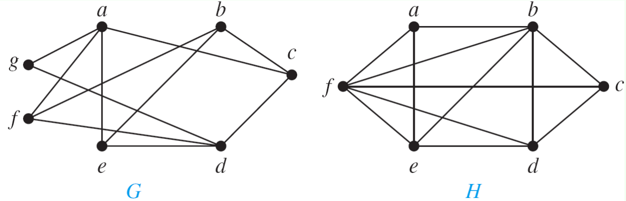
Are the graphs G and H displayed in the figure bipartite?



Is there an efficient way to determine whether a graph is a bipartite?

Examples

Are the graphs G and H displayed in the figure bipartite?



Is there an efficient way to determine whether a graph is a bipartite?

Theorem

A simple graph is bipartite if and only if it is possible to assign one of two different colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.

Proof

Proof.

$\Rightarrow$:

First, suppose that $G = (V, E)$ is a bipartite simple graph. Then $V = V_1 \cup V_2$, where V_1 and V_2 are disjoint sets and every edge in E connects a vertex in V_1 and a vertex in V_2. If we assign one color to each vertex in V_1 and a second color to each vertex in V_2, then no two adjacent vertices are assigned the same color.

Proof

Proof.

$\Rightarrow$:

First, suppose that $G = (V, E)$ is a bipartite simple graph. Then $V = V_1 \cup V_2$, where V_1 and V_2 are disjoint sets and every edge in E connects a vertex in V_1 and a vertex in V_2. If we assign one color to each vertex in V_1 and a second color to each vertex in V_2, then no two adjacent vertices are assigned the same color.

$\Leftarrow$:

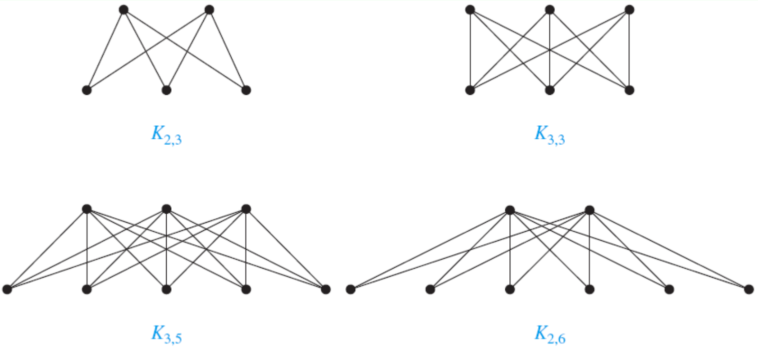
Now suppose that it is possible to assign colors to the vertices of the graph using just two colors so that no two adjacent vertices are assigned the same color. Let V_1 be the set of vertices assigned one color and V_2 be the set of vertices assigned the other color. Then, V_1 and V_2 are disjoint and $V = V_1 \cup V_2$. Furthermore, every edge connects a vertex in V_1 and a vertex in V_2 because no two adjacent vertices are either both in V_1 or both in V_2. Consequently, G is bipartite. $\square$

Complete bipartite graphs

A complete bipartite graph $K_{m,n}$ is a graph that has its vertex set partitioned into two subsets of m and n vertices, respectively with an edge between two vertices if and only if one vertex is in the first subset and the other vertex is in the second subset.

Complete bipartite graphs

A complete bipartite graph $K_{m,n}$ is a graph that has its vertex set partitioned into two subsets of m and n vertices, respectively with an edge between two vertices if and only if one vertex is in the first subset and the other vertex is in the second subset.



Matching

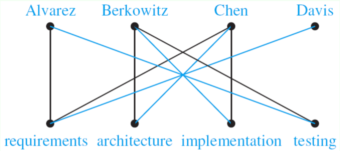
Definition

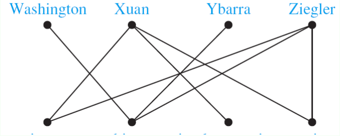
A matching M in a simple graph $G = (V, E)$ is a subset of the set E of edges of the graph such that no two edges are incident with the same vertex.

Matching

Definition

A matching M in a simple graph $G = (V, E)$ is a subset of the set E of edges of the graph such that no two edges are incident with the same vertex.





- A maximum matching is a matching with the largest number of edges.
- A matching M in a bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) is a complete matching from V_1 to V_2 if every vertex in V_1 is the endpoint of an edge in the matching, or equivalently, if $|M| = |V_1|$.

Hall'S Marriage theorem

Theorem

The bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a complete matching from V_1 to V_2 if and only if $|N(A)| \geq |A|$ for all subsets A of V_1.

Hall'S Marriage theorem

Theorem

The bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a complete matching from V_1 to V_2 if and only if $|N(A)| \geq |A|$ for all subsets A of V_1.

Proof.

$\Rightarrow$:

Suppose that there is a complete matching M from V_1 to V_2. Then, if $A \subset V_1$, for every vertex $v \in A$, there is an edge in M connecting v to a vertex in V_2. Consequently, there are at least as many vertices in V_2 that are neighbors of vertices in V_1 as there are vertices in V_1. It follows that $|N(A)| \geq |A|$.

$\Leftarrow$:

Basis step: If $|V_1| = 1$, it follows since V_1 contains a single vertex. $\square$

Proof of Hall'S Marriage theorem

Proof:

Inductive step: Let k be a positive integer. If $G = (V, E)$ is a bipartite graph with bipartition (V_1, V_2), and $|V_1| = j \leq k$, then there is a complete matching M from V_1 to V_2 whenever the condition that $|N(A)| \geq |A|$ for all $A \subset V_1$ is met.

Proof of Hall'S Marriage theorem

Proof:

Inductive step: Let k be a positive integer. If $G = (V, E)$ is a bipartite graph with bipartition (V_1, V_2), and $|V_1| = j \leq k$, then there is a complete matching M from V_1 to V_2 whenever the condition that $|N(A)| \geq |A|$ for all $A \subset V_1$ is met.

Now suppose that $H = (W, F)$ is a bipartite graph with bipartition (W_1, W_2) and $|W_1| = k + 1$.

Proof of Hall'S Marriage theorem

Proof:

Inductive step: Let k be a positive integer. If $G = (V, E)$ is a bipartite graph with bipartition (V_1, V_2), and $|V_1| = j \leq k$, then there is a complete matching M from V_1 to V_2 whenever the condition that $|N(A)| \geq |A|$ for all $A \subset V_1$ is met.

Now suppose that $H = (W, F)$ is a bipartite graph with bipartition (W_1, W_2) and $|W_1| = k + 1$.

Case I: Suppose $\forall j \in Z^+$ with $1 \leq j \leq k$, the vertices in every subset of j elements from W_1 are adjacent to at least $j+1$ elements of W_2. Then, we select a vertex $v \in W_1$ and an element $w \in N(\{v\})$ with $|N(\{v\})| \geq |\{v\}| = 1$. We delete v and w and all edges incident to them from H. This produces a bipartite graph H' with bipartition $(W_1 - \{v\}, W_2 - \{w\})$. Because $|W_1 - \{v\}| = k$, the inductive hypothesis tells us there is a complete matching from $W_1 - \{v\}$ to $W_2 - \{w\}$. Adding the edge from v to w to this complete matching produces a complete matching from W_1 to W_2.

Proof of Hall'S Marriage theorem Cont'd

Proof:

Case II: Suppose that for some j with $1 \leq j \leq k$, there is a subset W_1' of j vertices such that there are exactly j neighbors of these vertices in W_2. Let W_2' be the set of these neighbors. Then, by the inductive hypothesis there is a complete matching from W_1' to W_2'. Remove these $2j$ vertices from W_1 and W_2 and all incident edges to produce a bipartite graph K with bipartition $(W_1 - W_1', W_2 - W_2')$.

Proof of Hall'S Marriage theorem Cont'd

Proof:

Case II: Suppose that for some j with $1 \leq j \leq k$, there is a subset W_1' of j vertices such that there are exactly j neighbors of these vertices in W_2. Let W_2' be the set of these neighbors. Then, by the inductive hypothesis there is a complete matching from W_1' to W_2'. Remove these $2j$ vertices from W_1 and W_2 and all incident edges to produce a bipartite graph K with bipartition $(W_1 - W_1', W_2 - W_2')$. We will show that the graph K satisfies the condition $|N(A)| \geq |A|$ for all subsets A of $W_1 - W_1'$. If not, there would be a subset of t vertices of $W_1 - W_1'$ where $1 \leq t \leq k + 1 - j$ such that the vertices in this subset have fewer than t vertices of $W_2 - W_2'$ as neighbors. Then, the set of $j + t$ vertices of W_1 consisting of these t vertices together with the j vertices we removed from W_1 has fewer than $j + t$ neighbors in W_2, contradicting the hypothesis that $|N(A)| \geq |A|$ for all $A \subset W_1$.

Proof of Hall'S Marriage theorem Cont'd

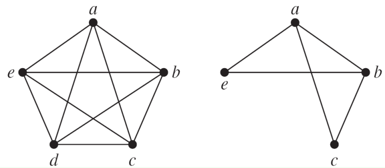
Proof:

Case II: Suppose that for some j with $1 \leq j \leq k$, there is a subset $W_1^{'}$ of j vertices such that there are exactly j neighbors of these vertices in W_2. Let $W_2^{'}$ be the set of these neighbors. Then, by the inductive hypothesis there is a complete matching from $W_1^{'}$ to $W_2^{'}$. Remove these $2j$ vertices from W_1 and W_2 and all incident edges to produce a bipartite graph K with bipartition $(W_1 - W_1^{'}, W_2 - W_2^{'})$. We will show that the graph K satisfies the condition $|N(A)| \geq |A|$ for all subsets A of $W_1 - W_1^{'}$. If not, there would be a subset of t vertices of $W_1 - W_1^{'}$ where $1 \leq t \leq k + 1 - j$ such that the vertices in this subset have fewer than t vertices of $W_2 - W_2^{'}$ as neighbors. Then, the set of $j + t$ vertices of W_1 consisting of these t vertices together with the j vertices we removed from W_1 has fewer than $j + t$ neighbors in W_2, contradicting the hypothesis that $|N(A)| \geq |A|$ for all $A \subset W_1$.

Hence, we obtain a complete matching from W_1 to W_2.

Subgraph

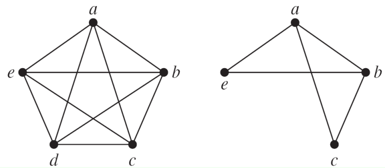
Subgraph



A **subgraph** of a graph $G = (V, E)$ is a graph $H = (W, F)$, where $W \subseteq V$ and $F \subseteq E$. A subgraph H of G is a proper subgraph of G if $H = G$.

Subgraph

Subgraph



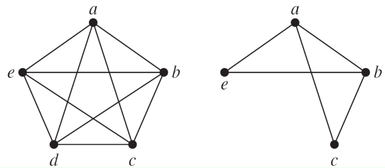
A **subgraph** of a graph $G = (V, E)$ is a graph $H = (W, F)$, where $W \subseteq V$ and $F \subseteq E$. A subgraph H of G is a proper subgraph of G if $H = G$.

Induced subgraphs

Let $G = (V, E)$ be a simple graph. The **subgraph induced** by a subset W of the vertex set V is the graph (W, F), where the edge set F contains an edge in E if and only if both endpoints of this edge are in W.

Subgraph

Subgraph



A **subgraph** of a graph $G = (V, E)$ is a graph $H = (W, F)$, where $W \subseteq V$ and $F \subseteq E$. A subgraph H of G is a proper subgraph of G if $H = G$.

Induced subgraphs

Let $G = (V, E)$ be a simple graph. The **subgraph induced** by a subset W of the vertex set V is the graph (W, F), where the edge set F contains an edge in E if and only if both endpoints of this edge are in W.

Remove or adding edges of a graph

Remove

Given a graph $G = (V, E)$ and an edge $e \in E$, we can produce a subgraph of G by removing the edge e. The resulting subgraph is denoted $G - e = (V, E - \{e\})$.
Similarly, if $E^{'}$ is a subset of E, we can produce a subgraph of G, denoted as $(V, E - E^{'})$, by removing the edges in $E^{'}$ from the graph.

Remove or adding edges of a graph

Remove

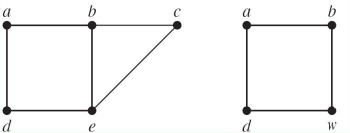
Given a graph $G = (V, E)$ and an edge $e \in E$, we can produce a subgraph of G by removing the edge e. The resulting subgraph is denoted $G - e = (V, E - \{e\})$.
Similarly, if $E^{'}$ is a subset of E, we can produce a subgraph of G, denoted as $(V, E - E^{'})$, by removing the edges in $E^{'}$ from the graph.

Adding

We can also add an edge e to a graph to produce a new larger graph when this edge connects two vertices already in G. We denote by $G + e$ the new graph, denoted as $(V, E \cup \{e\})$, produced by adding a new edge e, connecting two previously non-incident vertices, to the graph G.

Edge contractions and vertices remove

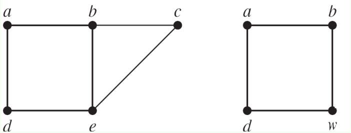
Edge contractions



An **edge contraction** which removes an edge e with endpoints u and v and merges u and v into a new single vertex w, and for each edge with u or v as an endpoint replaces the edge with one with w as endpoint in place of u or v and with the same second endpoint.

Edge contractions and vertices remove

Edge contractions



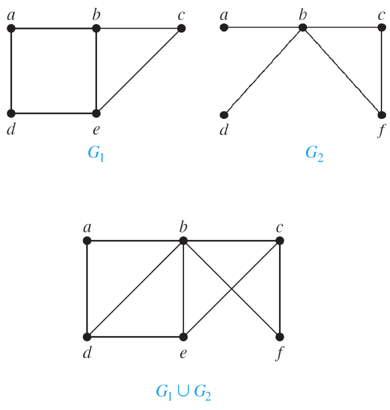
An **edge contraction** which removes an edge e with endpoints u and v and merges u and v into a new single vertex w, and for each edge with u or v as an endpoint replaces the edge with one with w as endpoint in place of u or v and with the same second endpoint.

Vertex removing

When we remove a vertex v and all edges incident to it from $G = (V, E)$, we produce a subgraph, denoted by $G - v = (V - \{v\}, E')$, where E' is the set of edges of G not incident to v.
If V' is a subset of V, then graph $G - V'$ is subgraph $(V - V', E')$, where E' is the set of edges of G not incident to a vertex in V'.

Graph union

Definition



The union of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$. The union of G_1 and G_2 is denoted by $G_1 \cup G_2$.

Take-aways

Conclusions

- Graphs
- Graph models
- Graph terminology and special types of graphs
  - Basic terminology
  - Some special simple graphs
  - New graphs from old