

# Introduction to Network Model

*Wan Dang*



# Contents

<b>1</b>	<b>Motivation, and introduction to graph theory</b>	<b>5</b>
1.1	Aspects of Networks . . . . .	5
1.1.1	Overview: The Connected World . . . . .	5
1.1.2	Structure and Behavior . . . . .	5
1.2	Central Themes and Topics . . . . .	7
1.2.1	Graph Theory (Theories of Structure) . . . . .	7
1.2.2	Game Theory (Theories of Behavior) . . . . .	7
1.2.3	Markets and Strategic interaction on Networks . . . . .	8
1.2.4	Information Networks . . . . .	8
1.2.5	Dynamics on Networks . . . . .	9
1.2.6	Institutions and Aggregate Behavior . . . . .	11
1.3	Introduction to Graph Theory . . . . .	11
1.3.1	Basic Definitions and Components . . . . .	11
1.3.2	Local Connectivity and Degree . . . . .	14
1.3.3	Global connectivity and Path . . . . .	16
1.3.4	Specialized Graph Structures . . . . .	18



# Chapter 1

## Motivation, and introduction to graph theory

### 1.1 Aspects of Networks

#### 1.1.1 Overview: The Connected World

A network is fundamentally defined as a pattern of interconnections among a set of objects. There has been a rising public interest in the complex “connectedness of modern society”, driven by this idea.

##### Areas where network appear

1. Social networks: collections of social ties (e.g., friendships, business relationships) have increased in complexity due to technological advances like global communication and digital interaction, weakening traditional geographic limitations.
2. Information networks: the information consumed by people has a networked structure. Understanding any piece of information requires knowing how it refers to or is endorsed by other pieces within a large network of links.
3. Technological and economic systems: these systems rely on networks of enormous complexity, making their behavior hard to predict and susceptible to disruptions that can spread, turning localized breakdowns into cascading failures or financial crises.
4. Operational networks: many organizations depend on complex networks of people, resources, and processes to function effectively. Understanding and optimizing these networks is crucial for efficiency and resilience. Examples include networks of suppliers for global manufacturing, networks of users for websites, and networks of advertisers for media companies.

#### 1.1.2 Structure and Behavior

##### Network structure

In its simplest form, a network is any group of objects where some pairs are connected by *link*. The objects are often shown as small circles, and the links connecting pairs are shown as lines between the circles. Such a representation is called a *graph* in mathematics, and the objects and links are called *nodes* (or vertices) and *edges*, respectively.

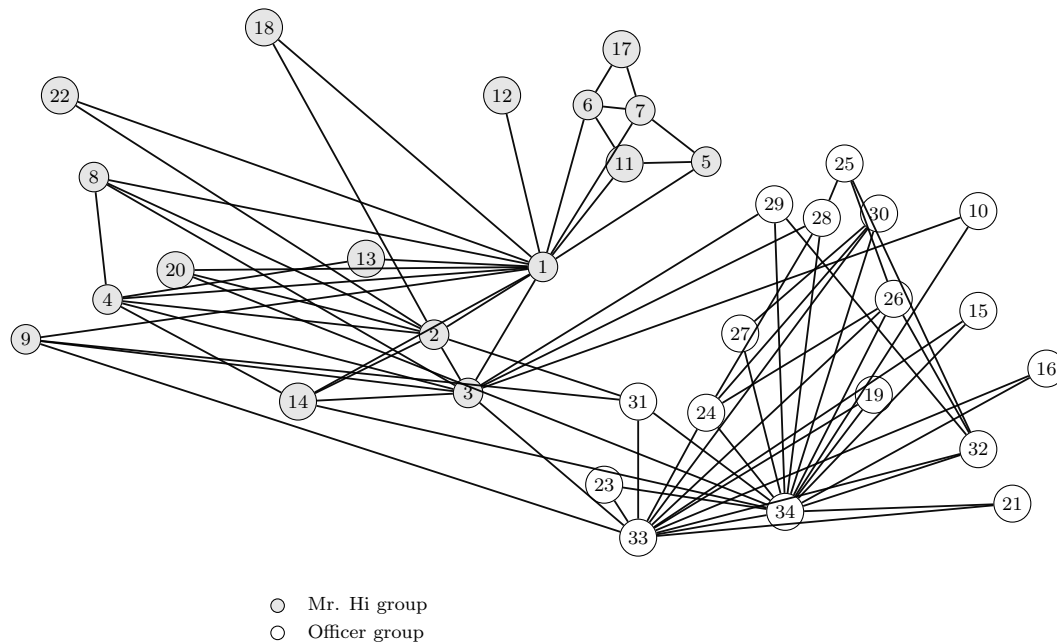


Figure 1.1: A social network of 34 members divided into two groups: Mr. Hi and Officer. Edges represent friendships between members.

### Tip 1.1.1: Network Drawings

Networks often show great visual complexity, sometimes featuring central “cores”, and sometimes splitting into multiple tightly-linked regions. Participants can be central or peripheral, or they may straddle the boundaries between tightly-linked regions. Understanding the structure of a network can provide insights into its function and behavior.

## Behavior and dynamics

The “connectedness” of a complex system involves two related issues:

1. Connectedness of Structure: who is linked to whom.
2. Connectedness of Behavior: the fact that an individual’s action have implicit consequences for everyone else in the system.

In network settings, individual actions must be evaluated with the expectation that the world — the interconnected system — will react. When individuals have incentives to achieve good outcomes, they must take into account the strategic behavior of others when planning their own actions.

### Tip 1.1.2: Aggregate Effects

When large groups are tightly interconnected, they respond in complex ways that are only visible at the population level, even if the underlying network is implicit (not directly observed). The rise in popularity of new products like YouTube or Flickr illustrates these collective feedback effects.

## Confluence of ideas

Understanding highly connected systems requires combining ideas for reasoning about network structure, strategic behavior, and population feedback effects. This synthesis draws perspectives from

several scientific disciplines:

- Computer Science, Applied Mathematics, and Operations Research: Provide the language to discuss the complexity of network structure and systems with interacting agents.
- Economics: Contributes models for the strategic behavior of interacting individuals.
- Sociology: Offers theoretical frameworks, particularly mathematical social network analysis, for the structure and dynamics of social groups.

## 1.2 Central Themes and Topics

### 1.2.1 Graph Theory (Theories of Structure)

Graph theory is defined as the study of network structure. Concepts drawn from social network analysis include:

#### Strong ties vs. Weak ties

- **Strong ties:** represent close and frequent social contacts and tend to be embedded in tightly-linked network regions.
- **Weak ties:** represent more casual and distinct social contacts and typically cross between tightly-linked regions. Weak ties can act as global “short-cuts”, which rise to the phenomenon known as *six degrees of separation*.

#### Structural holes

These are gaps between parts of a network that interact very little, which suggests a strategic opportunity for navigating a social landscape. An individual who bridges a structural hole can access diverse information and resources, potentially gaining a competitive advantage.

#### Network Fissures

Networks can capture sources of conflict within a group. The theory of *structural balance* is used to reason about how conflicts or antagonism at a local level can cause fissures in a network structure.

### 1.2.2 Game Theory (Theories of Behavior)

Game theory provides a framework for reasoning about behavior when outcomes depend on the joint decisions made by all involved parties. The core framework is: individuals choose a *strategy* to maximize their own *payoff*, taking into account the strategies chosen by others.

#### Example 1.2.1: Driving Routes

The strategy is the choice of route, and the payoff is based on the resulting travel time, which is affected by traffic congestion from all drivers. A counter-intuitive outcome is *Braess's Paradox*, where adding resources to a network can actually decrease efficiency.

### Example 1.2.2: Auctions

The strategy is how to bid, and the payoff is the difference between the value of the goods received and the price paid.

Equilibrium concepts are used to predict the outcome of strategic interactions. The most common is the *Nash equilibrium*, where no individual can improve their payoff by unilaterally changing their strategy.

### 1.2.3 Markets and Strategic interaction on Networks

Economics activity and trade naturally form networks where participants are linked by relationships (e.g., borrower-lender, trading partners).

- Network constraints: Network structures sometimes reflect constraints, such as institutional restrictions (regulations) or physical limitations (like geography, as seen in Medieval trade routes: Fig 1.2) which limit access between participants.
- Network position and power: The level of success for participants is affected by their positions in the network. Power depends on both the number of connections and the power of the other individuals connected to oneself.

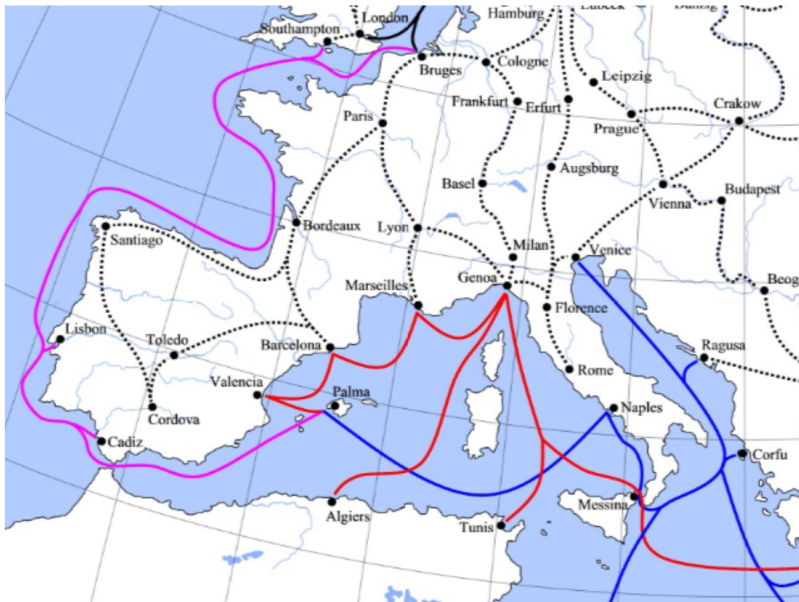


Figure 1.2: In some settings, such as this map of Medieval trade routes, physical networks constrain the patterns of interaction, giving certain participants an intrinsic economic advantage based on their network position.

### 1.2.4 Information Networks

Online information systems, such as the Web, possess a fundamental network structure.

- Community structure: link between Web pages reveal how they cluster into different communities. For instance, political blogs before the 2004 election separated into two distinct clusters corresponding to liberal and conservative perspectives.



- Prominence and ranking: Search engines (like Google) utilize network structure to evaluate page quality. Prominence is often recursively defined: a page is considered more prominent if it receives links from pages that are *themselves* prominent. This circular definition can be resolved during the concept of *equilibrium* in the link structure.
- Strategic interaction: The relationship between search engines and content creators is game-theoretic. Content creators constantly optimize their Web pages to achieve a high rank under the engines's current evaluation methods; thus, search methods must be developed considering these human feedback effects.

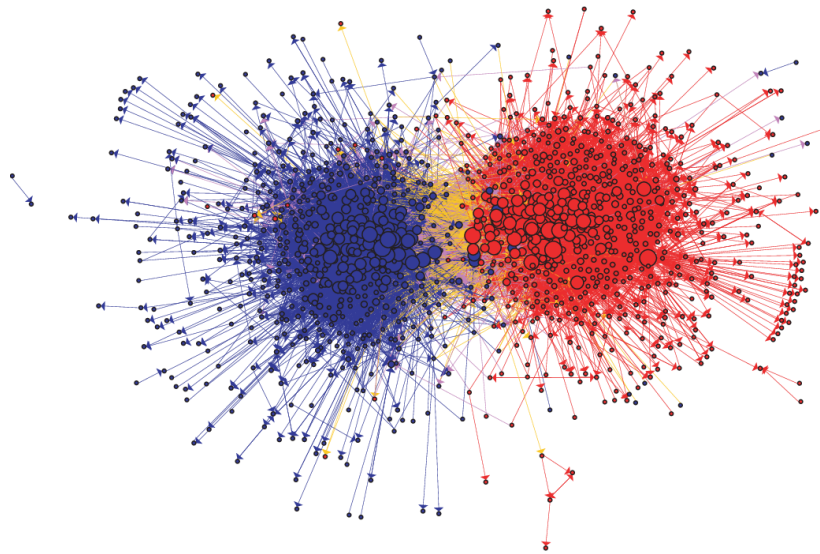


Figure 1.3: The links among Web pages can reveal densely-knit communities and prominent sites. In this case, the network structure of political blogs prior to the 2004 U.S. Presidential election reveals two natural and well-separated clusters.

## 1.2.5 Dynamics on Networks

### Population Effects

Collective phenomena, such as the spread of new *social practices* (new beliefs, opinions, or technologies), following recurring patterns. We called that *mechanism of influence*, which describes how individuals change their behavior based on the actions of others in their network.

- Information: People may copy others because observed behavior conveys information. If many people use a product (like YouTube), it suggests they know something about its quality. This can lead to *information cascades*, where rational individuals follow the crowd, abandoning their private information.
- Direct benefit (network effects): There is a direct benefit to aligning one's behavior with others, regardless of whether the decision is optimal. For social media sites, value increases as more people join (more content, wider audience). *Network effects* amplify success, creating a “rich-get-richer” feedback process that is characteristic of the aggregate distribution of popularity.



Figure 1.4: Cascading adoption of a new technology or service (in this case, the social-networking site MySpace in 2005–2006) can be the result of individual incentives to use the most widespread technology — either based on the informational effects of seeing many other people adopt the technology, or the direct benefits of adopting what many others are already using.

## Structural Effects

When individuals are motivated to adopt the behavior of their immediate neighbors, *cascading* effects can result, spreading outward from a small set of initial adopters.

- Diffusion barriers: The diffusion of technologies can be blocked by the boundary of a *densely-connected cluster* (a “closed community”) that is highly resistant to outside influences. See Fig 1.5.

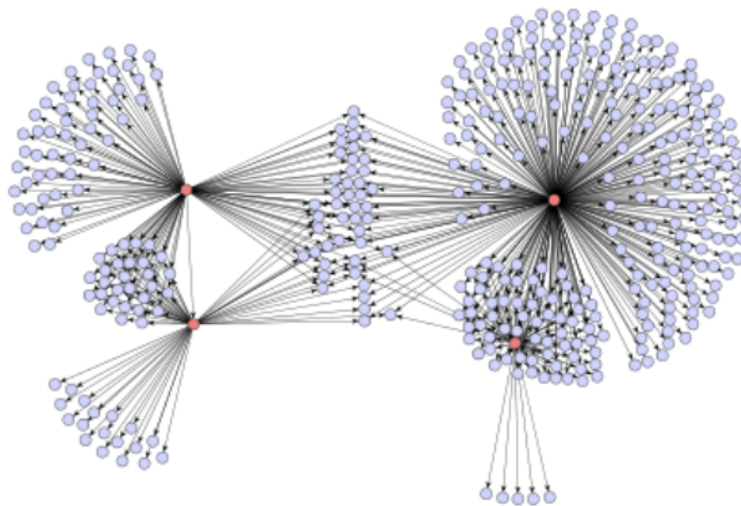


Figure 1.5: When people are influenced by the behaviors their neighbors in the network, the adoption of a new product or innovation can cascade through the network structure. Here, e-mail recommendations for a Japanese graphic novel spread in a kind of informational or social contagion.

- Social contagion: Cascading behavior is often referred to as *social contagion*, analogous to a biological epidemic. While social contagion involves decision-making and biological contagion involves pathogen transmission, the network-level dynamics are similar. See Fig 1.6.

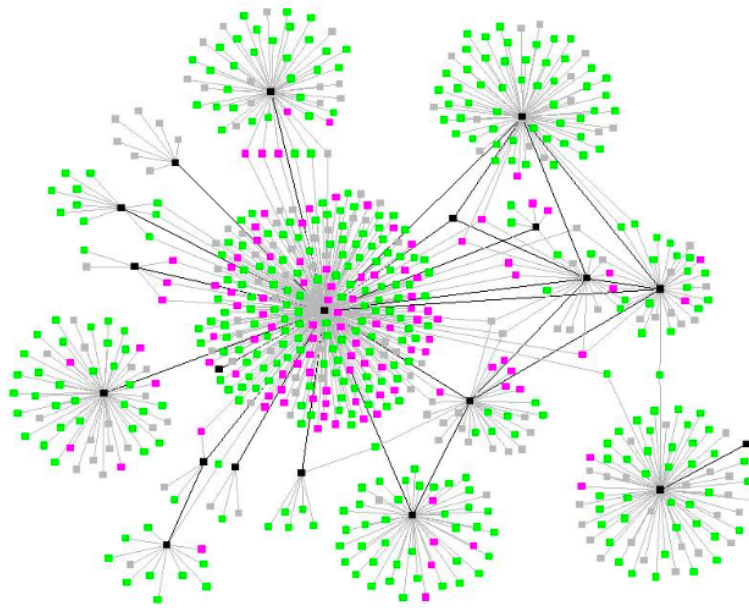


Figure 1.6: The spread of an epidemic disease (such as the tuberculosis outbreak shown here) is another form of cascading behavior in a network. The similarities and contrasts between biological and social contagion lead to interesting research questions.

### 1.2.6 Institutions and Aggregate Behavior

Institutions are the rules, conventions, or mechanisms designed by a society to synthesize individual actions into a pattern of aggregate behavior.

#### Example 1.2.3: Markets as Aggregators

Markets synthesize individuals' beliefs. For example, the price in a financial market aggregates beliefs about the value of assets. *Prediction markets* use a market mechanism where asset prices reflect an aggregate estimate for the probability of future events (e.g., elections).

Voting systems aggregate individual preferences over subjective choices. The process of producing a cumulative social preference from conflicting individual priorities is inherently difficult, as formalized by *Arrow's Impossibility Theorem*.

## 1.3 Introduction to Graph Theory

### 1.3.1 Basic Definitions and Components

#### The Graph Structure

A *graph* is a mathematical model for specifying relationships among a collection of items. It consists of two sets:  $G(V, E)$ .

- Vertices or nodes ( $V$ ): the set of objects in the graph, often represented by small circles.
- Edges or links ( $E$ ): connections between pairs of vertices. Edges are typically drawn as lines connecting the nodes.

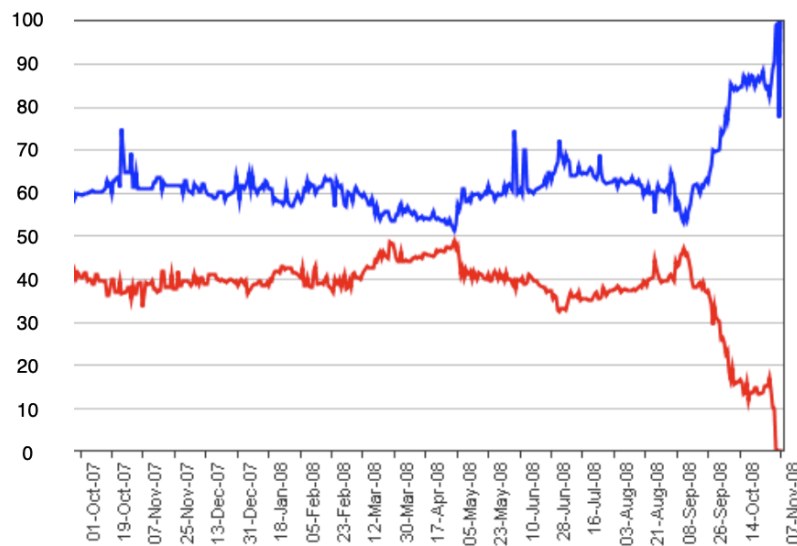


Figure 1.7: The plot here depicts the varying price over time for two assets that paid \$1 in the respective events that the Democratic or Republican nominee won the 2008 U.S. Presidential election.

### Graph Types based on Edge direction

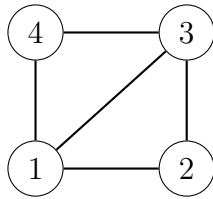
Graphs are categorized based on whether the relationships they model are symmetric or asymmetric. See table 1.1 for a summary.

Graph Type	Definition	Edge Pairs	Example
<b>Undirected Graph</b>	The relationship is symmetric; the edge simply connects the nodes to each other. Elements of $E$ are unordered pairs.	$(u, v) = (v, u)$	Facebook friendships.
<b>Directed Graph (Digraph)</b>	The relationship is asymmetric, meaning a link goes from one node <i>to</i> another, and direction matters. Elements of $E$ are ordered pairs $(u, v)$ . By convention, $(u, v)$ points to $v$ .	$(u, v)$ is distinct from $(v, u)$	Twitter follower networks; who-calls-whom phone networks.

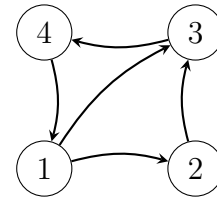
Table 1.1: Types of Graphs based on Edge Direction

#### Tip 1.3.1: Mutual edges

In directed graphs, it is possible for two nodes to have edges pointing to each other, known as *mutual edges*. This indicates a bidirectional relationship between the nodes, such as mutual friendships on social media platforms.



(a) Undirected graph on 4 nodes

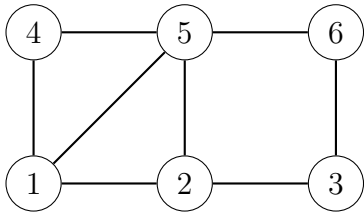


(b) Directed graph on 4 nodes

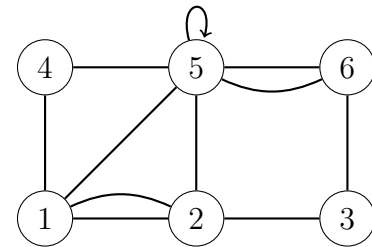
Figure 1.8: Examples of (a) undirected and (b) directed graphs on four nodes.

### Graph Types based on Edge properties

- Simple graph: Graphs that do not contain self-loops (an edge connecting a node to itself) or multi-edges (multiple edges between the same two nodes). Most analysis focuses on simple graphs.
- Multigraph: Graphs that may contain self-loops and/or multi-edges. Multi-edges are often encoded as edge weights (counts).
- Weighted graph: Graphs where edges are labeled with numerical values. The length of a path in a weighted graph is the sum of weighted of the traversed edges.



(a) Simple graph on 6 nodes



(b) Multigraph on the same 6 nodes

Figure 1.9: A simple graph versus a multigraph on the same vertex set. The multigraph includes parallel edges and a self-loop.

#### Example 1.3.1: Network Representations as Graphs

Different types of networks can be represented using various graph structures. For instance, the World Wide Web is often modeled as a directed multigraph with loops, while citation networks are directed and acyclic. Collaboration networks are typically undirected and unweighted. Table 1.2 summarizes some common examples.

Network	Graph Representations
WWW	Directed multi-graph (with loops), unweighted
Citation Network	Directed, unweighted, acyclic
Collaboration Network	Undirected, unweighted
Mobile Phone Calls	Directed, weighted
Protein Interaction	Undirected multi-graph (with loops), unweighted

Table 1.2: Examples of networks and common graph representations.

### 1.3.2 Local Connectivity and Degree

#### Adjacency and Incidence

Two nodes are *adjacent* if they are directly connected by an edge. The set of nodes adjacent to a given node  $u$  is called the *neighborhood* of  $u$ , denoted  $N(u)$ .

An edge is *incident* to a node if the node is one of the edge's endpoints. For an edge  $(u, v)$ , it is incident to both  $u$  and  $v$ . See Fig 1.10 for an illustration.

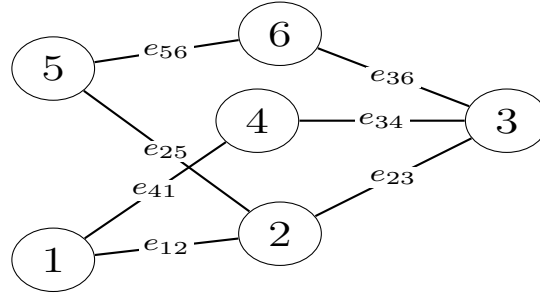


Figure 1.10: A simple undirected graph with vertex set  $V = \{1, 2, 3, 4, 5, 6\}$  and edge set  $E = \{e_{12}, e_{23}, e_{34}, e_{41}, e_{25}, e_{56}, e_{36}\}$ . **Adjacency (vertex–vertex):** vertices 1 and 2 are adjacent because  $e_{12}$  joins them; vertices 1 and 3 are not adjacent because there is no edge between them. **Incidence (edge–vertex):** edge  $e_{12}$  is incident to vertices 1 and 2 (and is not incident to 3); the edges incident to vertex 2 are  $e_{12}$ ,  $e_{23}$ , and  $e_{25}$ .

#### Vertex degrees

The *degree* ( $d_v$ ) of a vertex  $v$  is its number of incident edges. High-degree vertices are often considered influential, central, or prominent. See Fig 1.11 for an illustration.

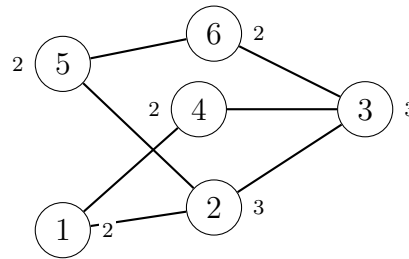


Figure 1.11: A simple undirected graph on  $V = \{1, 2, 3, 4, 5, 6\}$  with vertex degrees shown. For example,  $d(2) = 3$  because vertex 2 is adjacent to  $\{1, 3, 5\}$ , while  $d(1) = 2$  because vertex 1 is adjacent to  $\{2, 4\}$ .

#### Definition 1.3.1: Neighborhood

The *neighborhood*  $\mathcal{N}_i$  of a node  $i$  is the set of all its adjacent nodes.  
The size of the neighborhood is equal to the degree, i.e.,  $|\mathcal{N}_i| = d_i$ .

#### Degree formulas and properties

The following properties hold for an undirected graph:

### Properties 1.3.1: Degree Properties

- Degree range: For any vertex  $v$ ,  $0 \leq d_v \leq n - 1$ , where  $n$  is the total number of vertices in the graph.
- Sum of degrees: The sum of the degrees of all vertices is equal to twice the number of edges:

$$\sum_{v=1}^{|V|} d_v = 2|E|.$$

- Odd degree count: The number of vertices with an odd degree must be an even number.

*Proof.* Consider a finite undirected graph  $G = (V, E)$  (allowing neither direction nor multiple counting beyond multiplicity). Each edge  $e = \{u, w\} \in E$  is incident to exactly two vertices, namely  $u$  and  $w$ .

Now count the number of *incidences* (vertex–edge pairs  $(v, e)$  where  $e$  is incident to  $v$ ) in two ways:

- Fix a vertex  $v$ . There are exactly  $d(v)$  edges incident to  $v$ , so the total number of incidences is  $\sum_{v \in V} d(v)$ .
- Fix an edge  $e$ . It contributes exactly 2 incidences (one for each endpoint), so the total number of incidences is  $2|E|$ .

Both expressions count the same set of incidences, hence

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

□

### Degrees in Directed Graphs

In directed graphs (digraphs), vertices have two types of degrees:

- **In-degree** ( $d_v^i$ ): the number of edges pointing *to* vertex  $v$ .
- **Out-degree** ( $d_v^o$ ): the number of edges pointing *away* from vertex  $v$ .

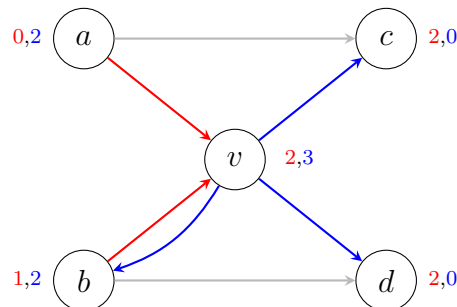


Figure 1.12: A directed graph where each vertex is annotated with its **in-degree** and **out-degree** as an ordered pair  $(d^-, d^+)$ . Incoming edges are shown in **red** and outgoing edges in **blue**.



### 1.3.3 Global connectivity and Path

#### Movement in a Graph

##### Definition 1.3.2: Path, Walk, Cycle

A *path* in a graph is a consecutive sequence of distinct vertices  $\{v_0, v_1, \dots, v_l\}$  such that  $v_i$  and  $v_{i+1}$  are adjacent. The length of the path is  $l$ .

A *walk* is similar to a path, but vertices and edges can be repeated. A walk that starts and ends at the same vertex is called a *closed walk*, or a *cycle* if no other vertices are repeated.

These definition generalize naturally to directed graphs.

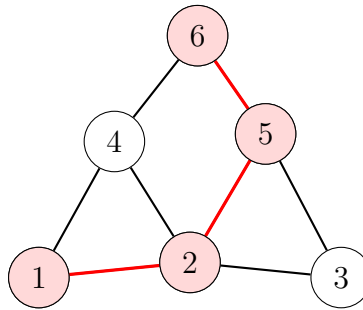


Figure 1.13: A graph with the path  $1 \rightarrow 2 \rightarrow 5 \rightarrow 6$  highlighted in red.

#### Distance and Diameter

##### Definition 1.3.3: Distance and Diameter

The *distance* between two vertices  $u$  and  $v$ , denoted  $d(u, v)$ , is the length of the shortest path connecting them. If no path exists, the distance is defined to be infinite.

*Diameter* of a graph is the maximum distance between any pair of vertices in the graph:

$$\text{diameter}(G) = \max_{u, v \in V} d(u, v).$$

Efficient algorithms exist for computing shortest paths in both undirected and directed graphs, such as Dijkstra's algorithm for graphs with non-negative edge weights, Floyd-Warshall algorithm for dense graphs, and Bellman-Ford algorithm for graphs with negative edge weights.

#### Connectivity

Vertex  $v$  is *reachable* from vertex  $u$  if there exists a path from  $u$  to  $v$ .

##### Definition 1.3.4: Connected graphs (Undirected)

A graph is *connected* if every vertex is reachable from every other vertex. Removing a “bridge edge” (an edge whose removal increases the number of connected components) can disconnect the graph.



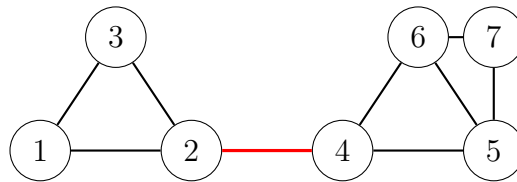


Figure 1.14: A connected undirected graph on 7 vertices. The red edge  $(2, 4)$  is a *bridge*: removing it disconnects the graph into two components (the triangle on  $\{1, 2, 3\}$  and the subgraph on  $\{4, 5, 6, 7\}$ ).

### Definition 1.3.5: Connected components

A *component* is a maximally connected subgraph. A maximal subgraph is one where adding any other vertex would ruin the connectivity. Disconnected graphs have two or more components.

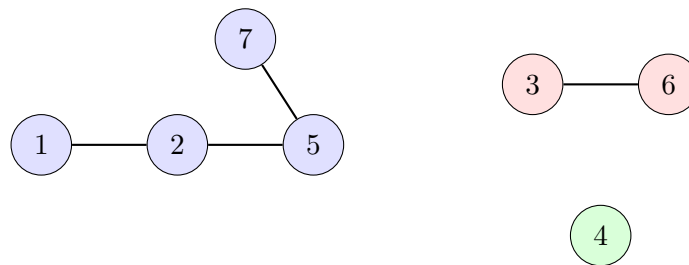


Figure 1.15: A disconnected graph with three connected components:  $\{1, 2, 5, 7\}$ ,  $\{3, 6\}$ , and  $\{4\}$ . The subgraph on  $\{3, 4, 6\}$  is not connected because vertex 4 is isolated from  $\{3, 6\}$ , while the subgraph on  $\{1, 2, 5\}$  is not maximal since it can be enlarged to the connected component  $\{1, 2, 5, 7\}$ .

Disconnected graphs have 2 or more components, largest component often called giant component. Large real-world networks often exhibit one giant component.

### Tip 1.3.2: Why do we expect to find a single giant component?

In many real-world networks, the presence of a single giant component can be attributed to the underlying connectivity patterns and the nature of interactions among nodes. Factors such as preferential attachment, where new nodes are more likely to connect to already well-connected nodes, and the small-world phenomenon, where most nodes can be reached from every other by a small number of steps, contribute to the formation of a giant component. Additionally, real-world networks often exhibit clustering and community structures that facilitate connectivity within the giant component.

## Connectivity in Directed Graphs

Connectivity is more complex in directed graphs and has two main notions:

1. Strongly connected: A digraph is strongly connected if for every pair of vertices  $u$  and  $v$ ,  $u$  is reachable from  $v$  (via a directed path) and  $v$  is reachable from  $u$ .
2. Weakly connected: A digraph is weakly connected if it remains connected after disregarding the edge directions (i.e., its underlying undirected graph is connected). Strongly connectivity implies weak connectivity.

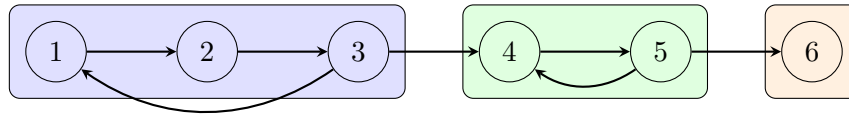


Figure 1.16: A directed graph that is *weakly connected* (ignoring arrow directions, all vertices lie in one connected component) but *not strongly connected* (there is no directed path from 6 back to earlier vertices, and no directed path from  $\{4, 5\}$  back to  $\{1, 2, 3\}$ ). The strongly connected components are highlighted:  $\{1, 2, 3\}$ ,  $\{4, 5\}$ , and  $\{6\}$ .

### 1.3.4 Specialized Graph Structures

#### Complete Graphs and Cliques

A *complete graph* is a simple undirected graph in which every pair of distinct vertices is connected by a unique edge.

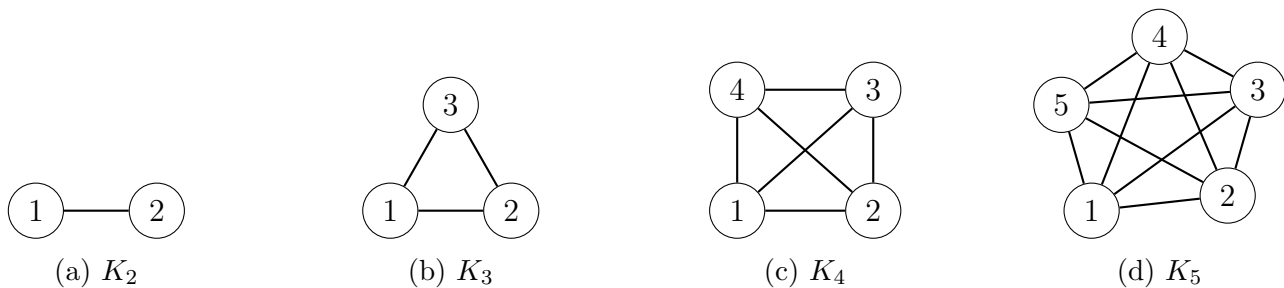


Figure 1.17: Complete graphs  $K_n$  for  $n = 2, 3, 4, 5$ .

#### Properties 1.3.2: Number of edges in complete graphs

The number of edges is equal to the number of vertex pairs:

$$\text{Number of edges in } K_n = \binom{n}{2} = \frac{n(n-1)}{2}$$

*Proof.* In the complete graph  $K_n$ , every pair of distinct vertices is connected by exactly one edge. Thus, counting edges is the same as counting unordered pairs of distinct vertices.

The number of ways to choose 2 vertices from  $n$  (order does not matter) is

$$\binom{n}{2} = \frac{n(n-1)}{2}.$$

Therefore  $|E(K_n)| = \binom{n}{2} = \frac{n(n-1)}{2}$ . □

A *clique* in a graph is a subset of vertices such that every two distinct vertices are adjacent. In other words, a clique is a complete subgraph. The size of the largest clique in a graph is called the *clique number*.

#### Regular graphs

A *regular graph* is a graph where every vertex has the same degree  $k$ . Such a graph is called a  $k$ -regular graph.

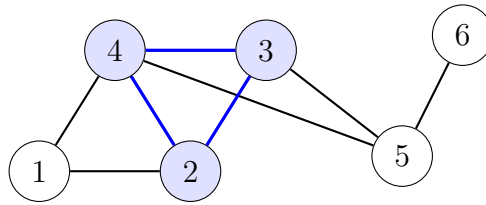
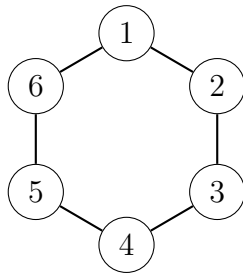
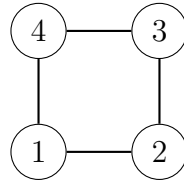


Figure 1.18: An example of a clique: vertices  $\{2, 3, 4\}$  form a 3-clique (a  $K_3$  subgraph), since every pair among them is connected by an edge.

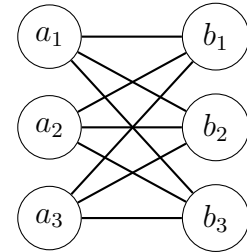
The complete graph  $K_n$  is an example of a  $(n - 1)$ -regular graph, since every vertex is connected to all other  $n - 1$  vertices. Cycles are examples of 2-regular graphs, as each vertex is connected to exactly two other vertices.



(a)  $C_6$  (2-regular)



(b)  $C_4$  (2-regular)



(c)  $K_{3,3}$  (3-regular)

Figure 1.19: Examples of regular graphs:  $C_6$  and  $C_4$  are 2-regular;  $K_{3,3}$  is 3-regular.

Regular graphs frequently arise in the study of crystal structures (physics/chemistry), pixel adjacency models in image processing (geo-spatial settings), and modeling opinion formation or information cycles.

## Trees and Acyclic Graphs

### Definition 1.3.6: Trees and Forests

A connected graph that is acyclic (contains no cycles) is called a *tree*.  
A *forest* is a disjoint set of trees.

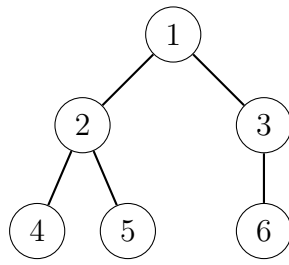
A *directed tree* (or *arborescence*) is a directed graph where there is a unique directed path from a designated root node to every other node.

### Definition 1.3.7: Directed Acyclic Graph (DAG)

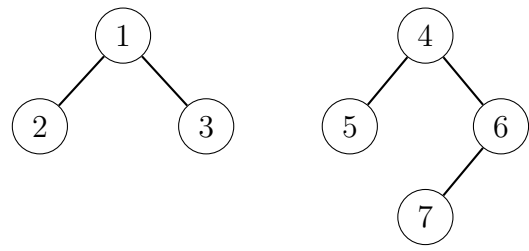
A *directed acyclic graph* (DAG) is a directed graph with no directed cycles. The underlying graph of a DAG need not be a tree. DAGs are commonly used to represent hierarchical structures, such as task scheduling or data processing pipelines.

### Example 1.3.2: Hierarchical Structures as Trees and DAGs

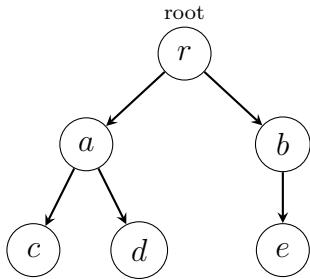
Hierarchical structures, such as organizational charts or file directory systems, can be effectively modeled using trees and DAGs. In an organizational chart, each employee reports to a single



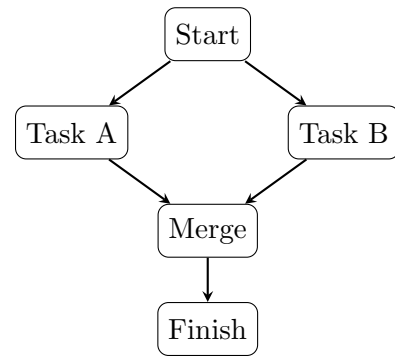
(a) Tree



(b) Forest (disjoint trees)



(c) Directed tree (arborescence)



(d) DAG (no directed cycles)

Figure 1.20: Examples: (a) a tree (connected and acyclic), (b) a forest (disjoint set of trees), (c) a directed tree/arborescence (unique directed path from root to each node), and (d) a DAG (directed acyclic graph).

*(continued)*

manager, forming a tree structure. In a file directory system, folders can contain subfolders and files, creating a DAG where cycles are avoided to prevent infinite loops in navigation.