Social Network Analysis

Sakthi Balan M

LNMIIT

Social Network Analysis

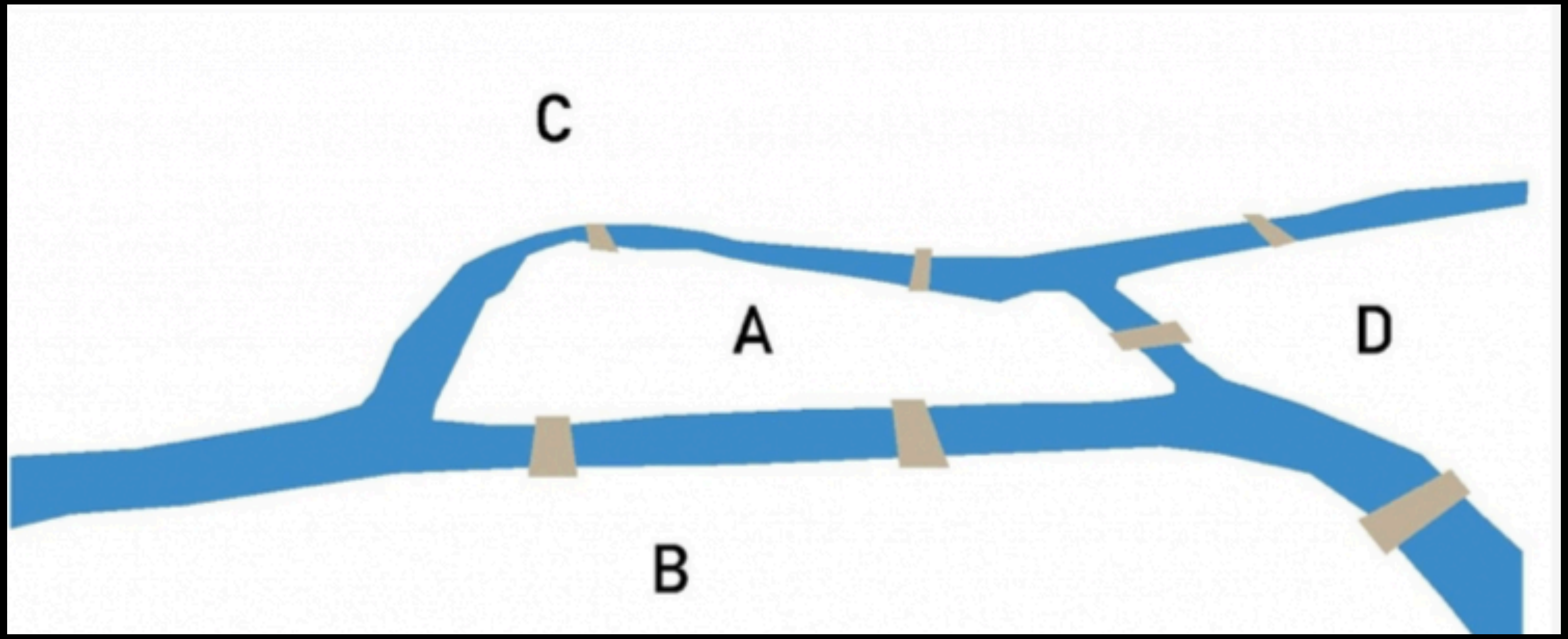
Basic Graph Theory Concepts and Algorithms

LNMIIT

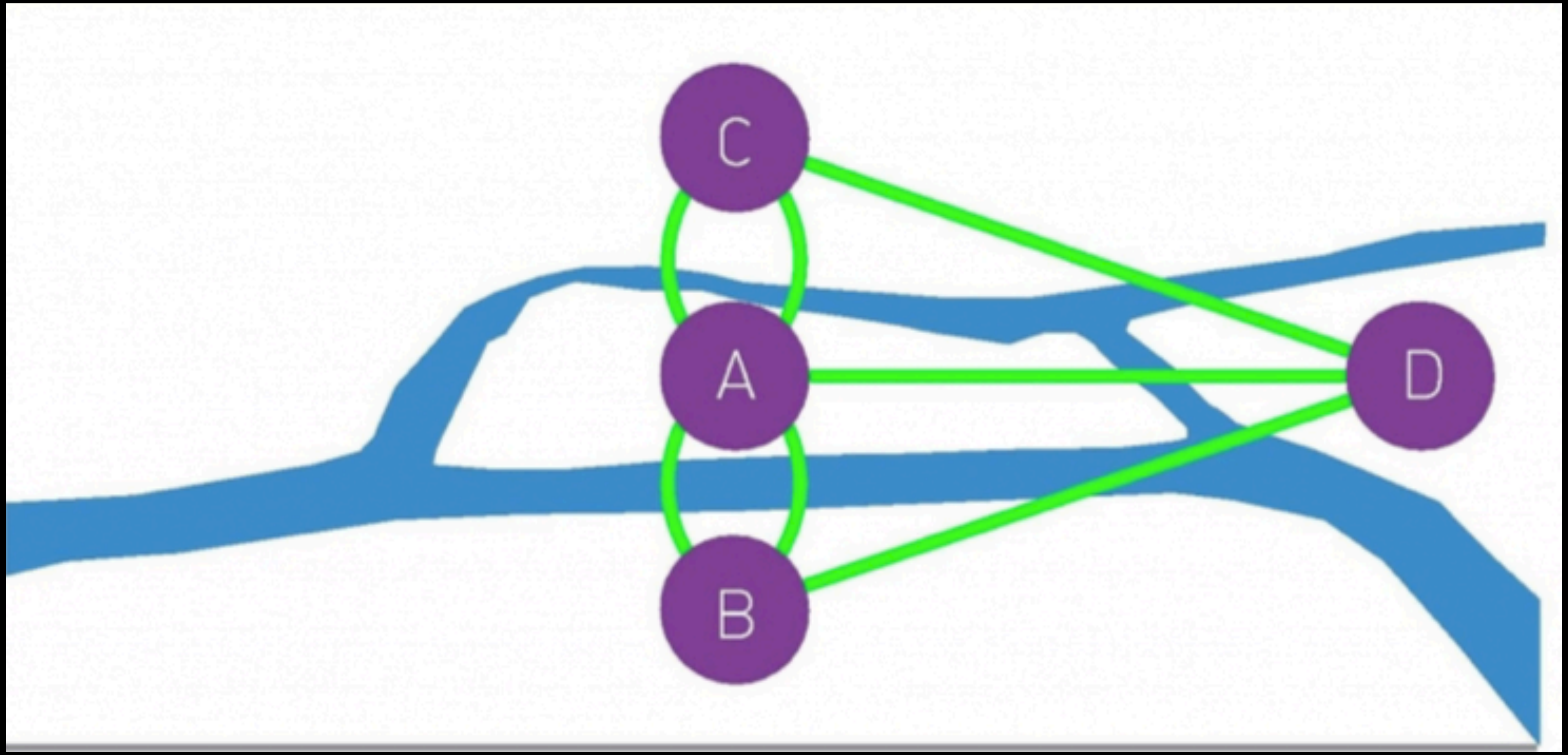
Some pictures have been used from the following books:
(This is only for learning purpose)

Social Media Mining by Zafaranai et al,
Network Science by Barabasi
Networks, Crowd, Behaviour by Kleinberg et al

The Bridges of Königsberg



The Bridges of Königsberg



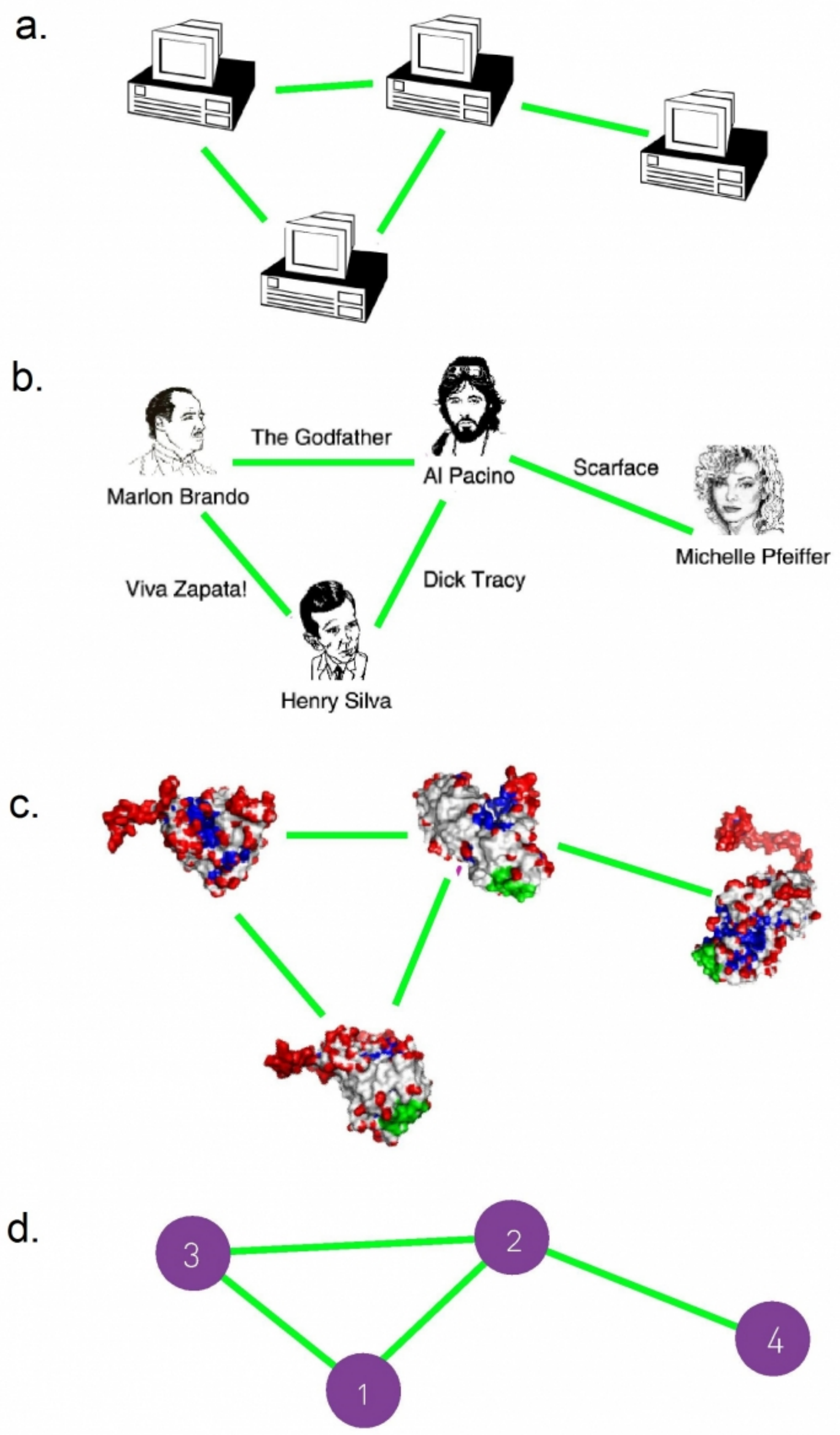
Graph

- If we want to study about a group of actors we need to study about their network not about actors as such!

- A Graph is represented as $G = (V, E)$ where V is the set of all nodes / vertices and E is the set of all edges / links

- The two parameters that help to describe the graph are the number of vertices, say n and the number of edges, say m

LNMIIT



Picture from
Network Science
by
C. Barabasi

Different networks but same Graph!

Graph is an abstraction

We may leave out the specifics and concentrate on the essential thing, i.e., connectivity

A graph *G* can be drawn in many ways!

A graph with directed edges is called a digraph

Unless specified a graph in general is an undirected graph

LNMIIT

Graph

- Degree of a vertex — number of edges incident on a vertex
- Average degree of a node $\langle k \rangle$ *in an undirected graph* = $\frac{1}{N}\sum_{i=1}^{N} k_i$
    - k_i is the degree of the vertex *i*
- *For directed graph* $\langle k \rangle = \frac{1}{N}\sum_{i=1}^{N} k_i^{in} = \frac{1}{N}\sum_{i=1}^{N} k_i^{out}$

Some Networks of our interest

Network	Nodes	Links	Directed / Undirected	N	L	‹K›
Internet	Routers	Internet connections	Undirected	192,244	609,066	6.34
WWW	Webpages	Links	Directed	325,729	1,497,134	4.60
Power Grid	Power plants, transformers	Cables	Undirected	4,941	6,594	2.67
Mobile-Phone Calls	Subscribers	Calls	Directed	36,595	91,826	2.51
Email	Email addresses	Emails	Directed	57,194	103,731	1.81
Science Collaboration	Scientists	Co-authorships	Undirected	23,133	93,437	8.08
Actor Network	Actors	Co-acting	Undirected	702,388	29,397,908	83.71
Citation Network	Papers	Citations	Directed	449,673	4,689,479	10.43
E. Coli Metabolism	Metabolites	Chemical reactions	Directed	1,039	5,802	5.58
Protein Interactions	Proteins	Binding interactions	Undirected	2,018	2,930	2.90

Picture from Network Science by C. Barabasi

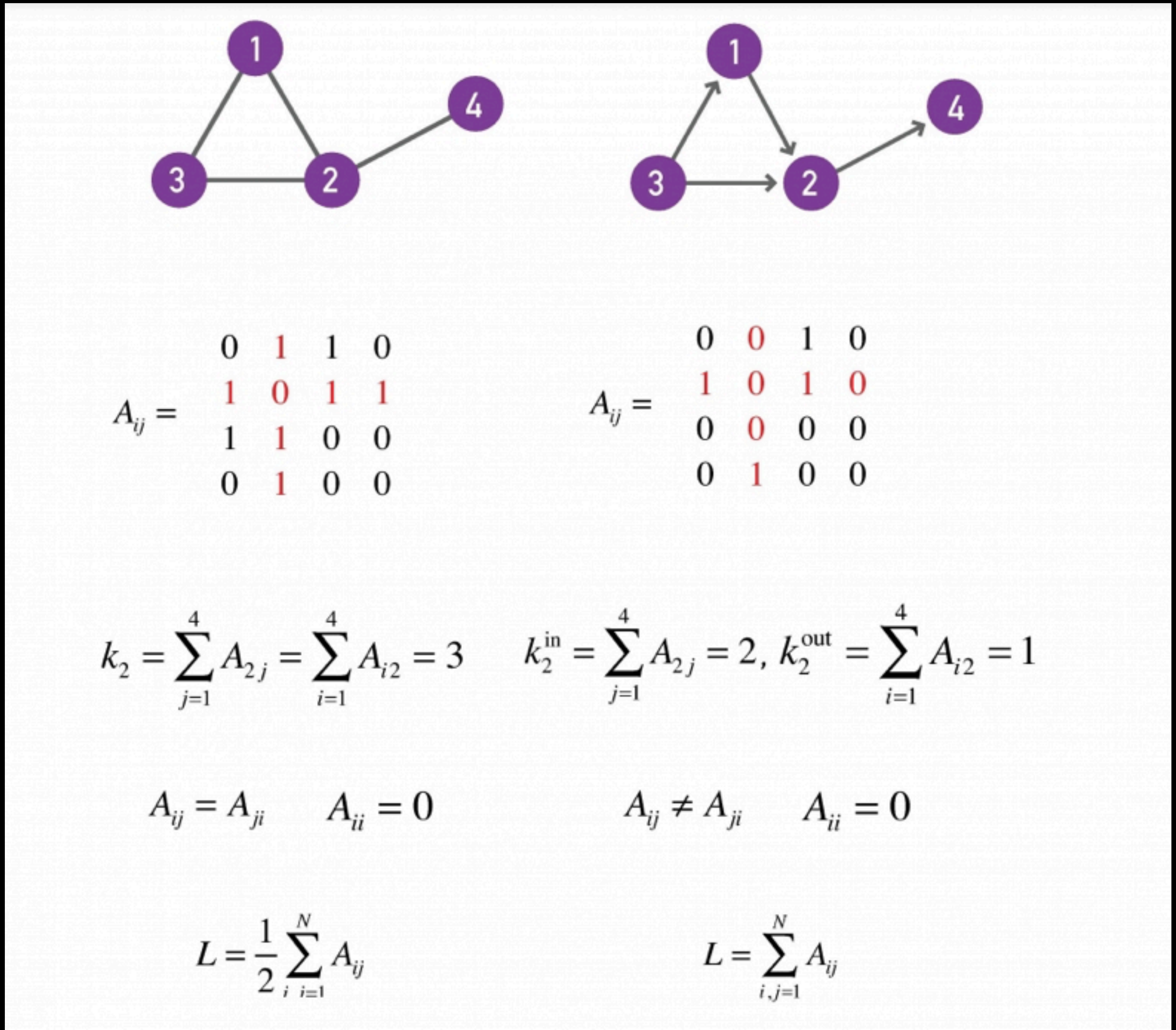
LNMIIT

Graph Representation

- Adjaceny Matrix, A — $n \times n$ matrix
  - if nodes i and j are connected by an edge in G then

$$A_{ij} = 1 \text{ **else** } 0$$

Some intances (sparse graphs) it will be good to use adjacency list



Picture from Network Science by C. Barabasi

Degree Distribution

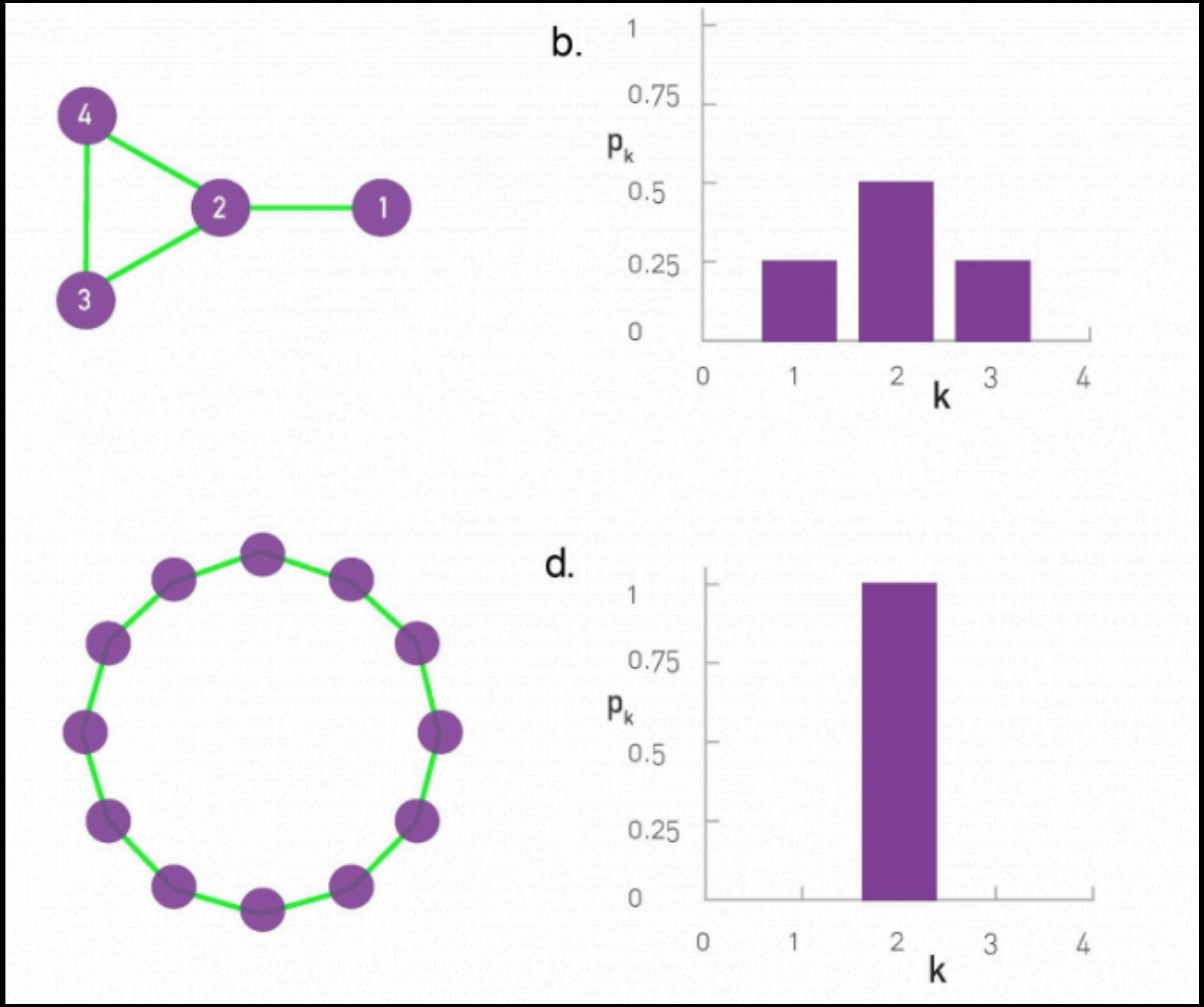
- The degree distribution of a graph G
  - p_k is the probability that a randomly selected node in the network has degree k.

$$p_k = \frac{N_k}{N}$$

- Number of degree-k nodes can be obtained from the degree distribution as $N_k = Np_k$
- The average degree of the network is

$$\sum_{k=0}^{\infty} kp_k$$

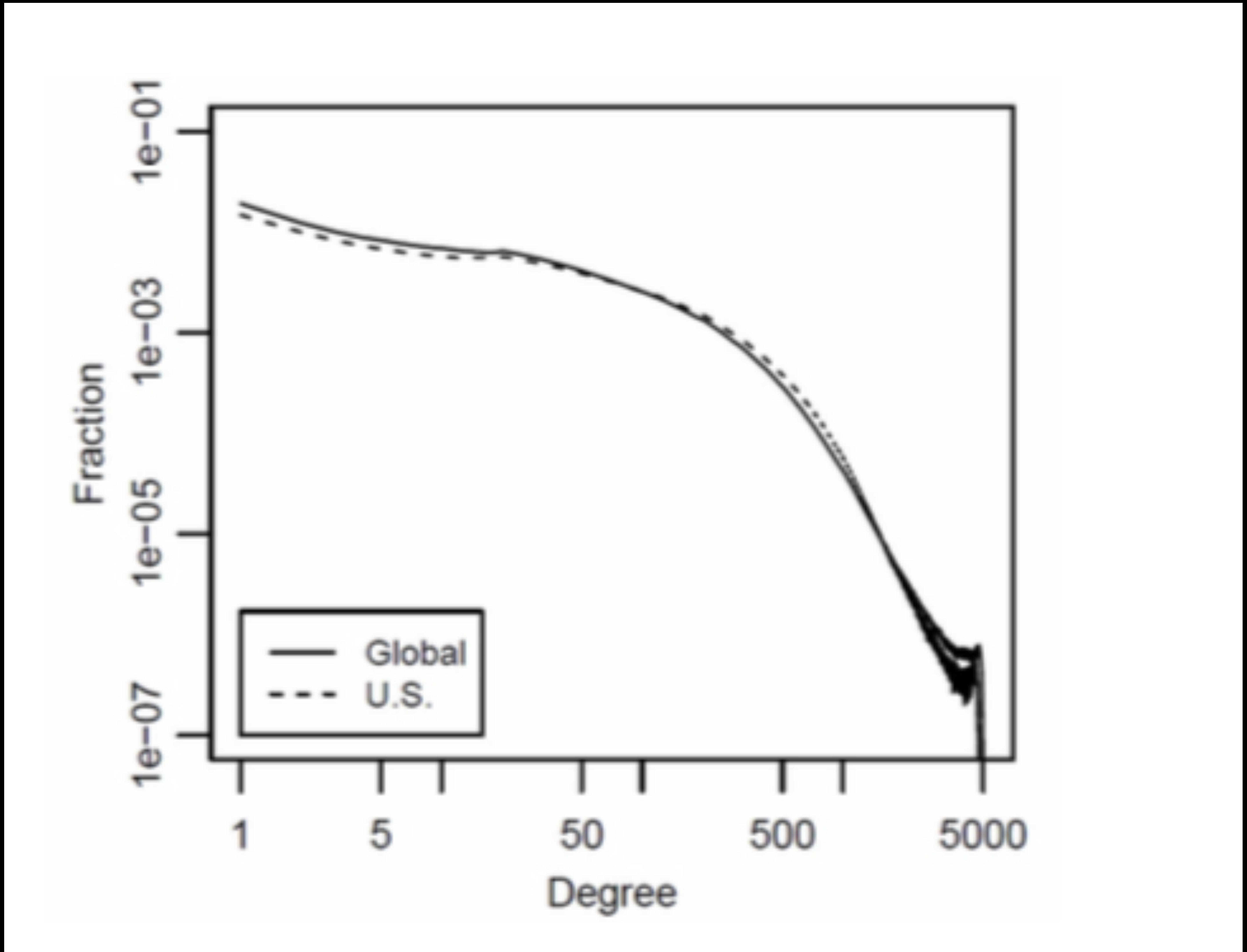
Picture from Network Science by C. Barabasi



LNMIIT

Facebook Degree Distribution

Users with lots of friends is so less and users with minimum friends is huge! Its a power law distribution.



Picture taken from Social media mining by R. Zafarani et al

LNMIIT

Weighted Graphs

- Graph with weights for its edges is called as weighted graph
- In a graph we assume that all the edges have equal weights, i.e. 1
  - Many scenarios equal weight is not appropriate — some edges will be used more than other edges — rail network, roads and so on
- For a weighted graph G the adjacency matrix takes care of the weights, i.e., $A_{ij} = w_{ij}$

Metcalfe’s Law: the Value of a Network

- Value of network is proportional to square of the number of nodes
  - What is the *max* and *min* number of edges a network can have?
- More individuals use a network, the more valuable it becomes. Example: WWW, cellular network and so on
- Cost associated with the network increases linearly with number of nodes but the value will be square of it!

LNMIIT

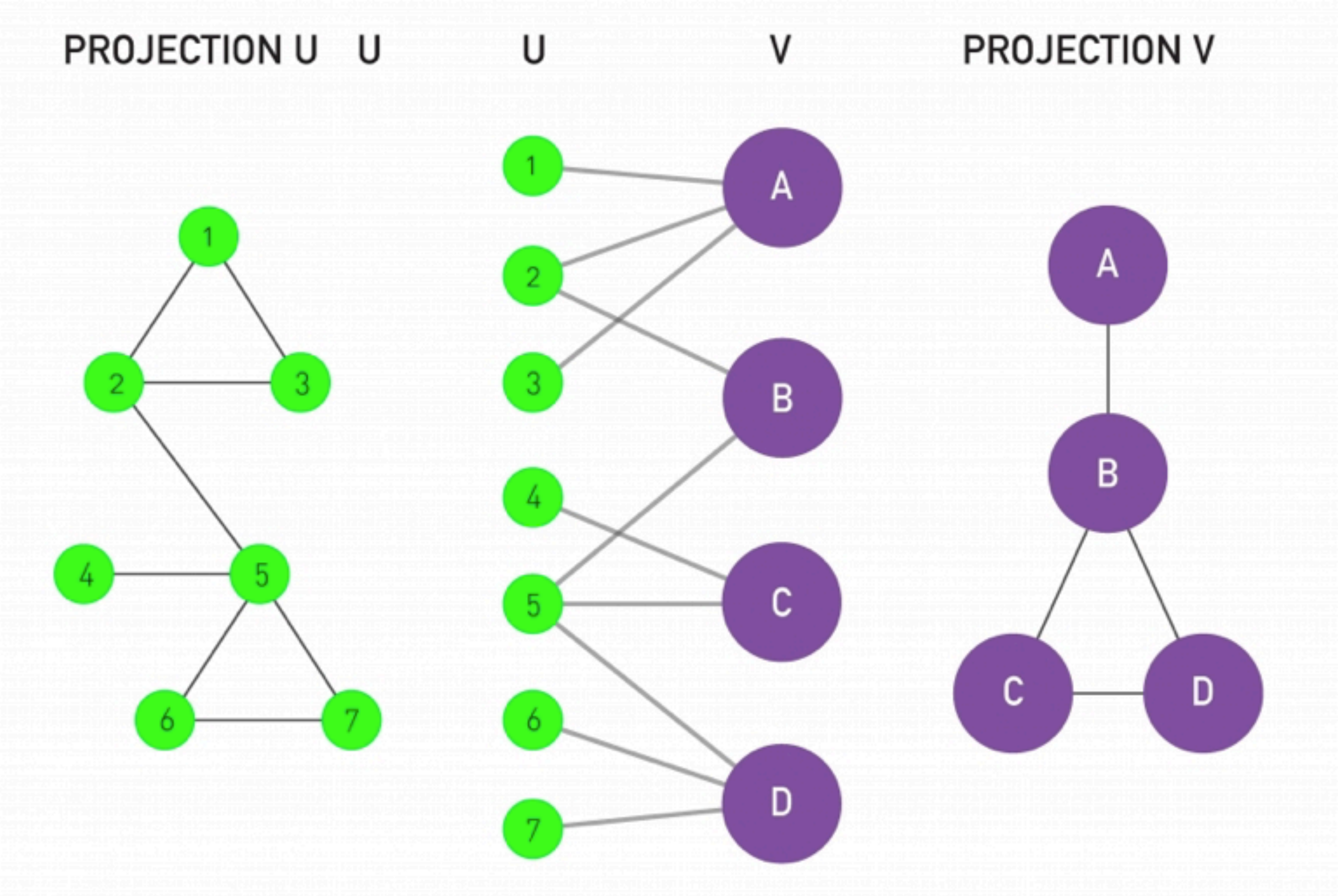
Bipartite Networks

- A bipartite graph is a network whose nodes can be divided into two disjoint sets *U* and *V* such that each link connects a U-node to a V-node.

- We can generate two projections for each bipartite network.

    - The first projection connects two U-nodes by a link if they are linked to the same V-node in the bipartite representation.

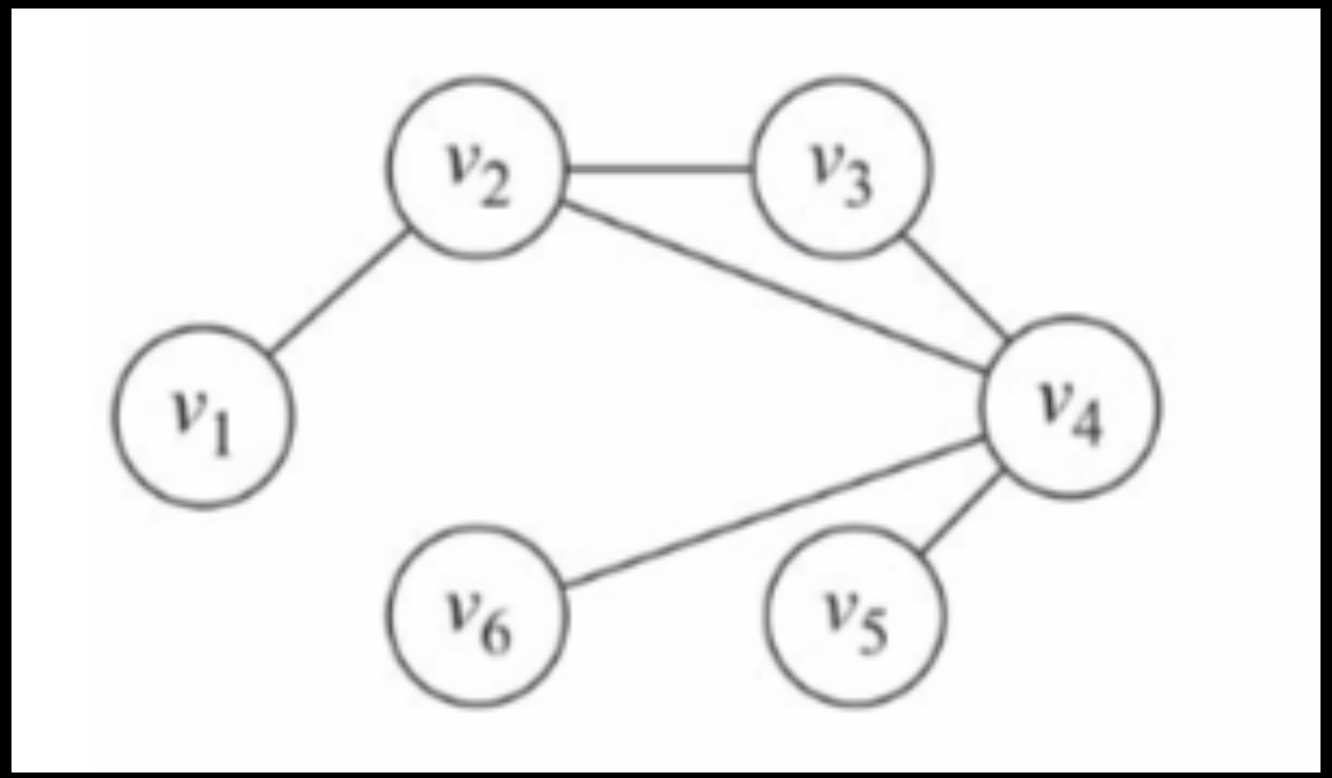
    - The second projection connects the V-nodes by a link if they connect to the same U-node



Picture from Network Science by C. Barabasi

Some basic terms in Graph Theory

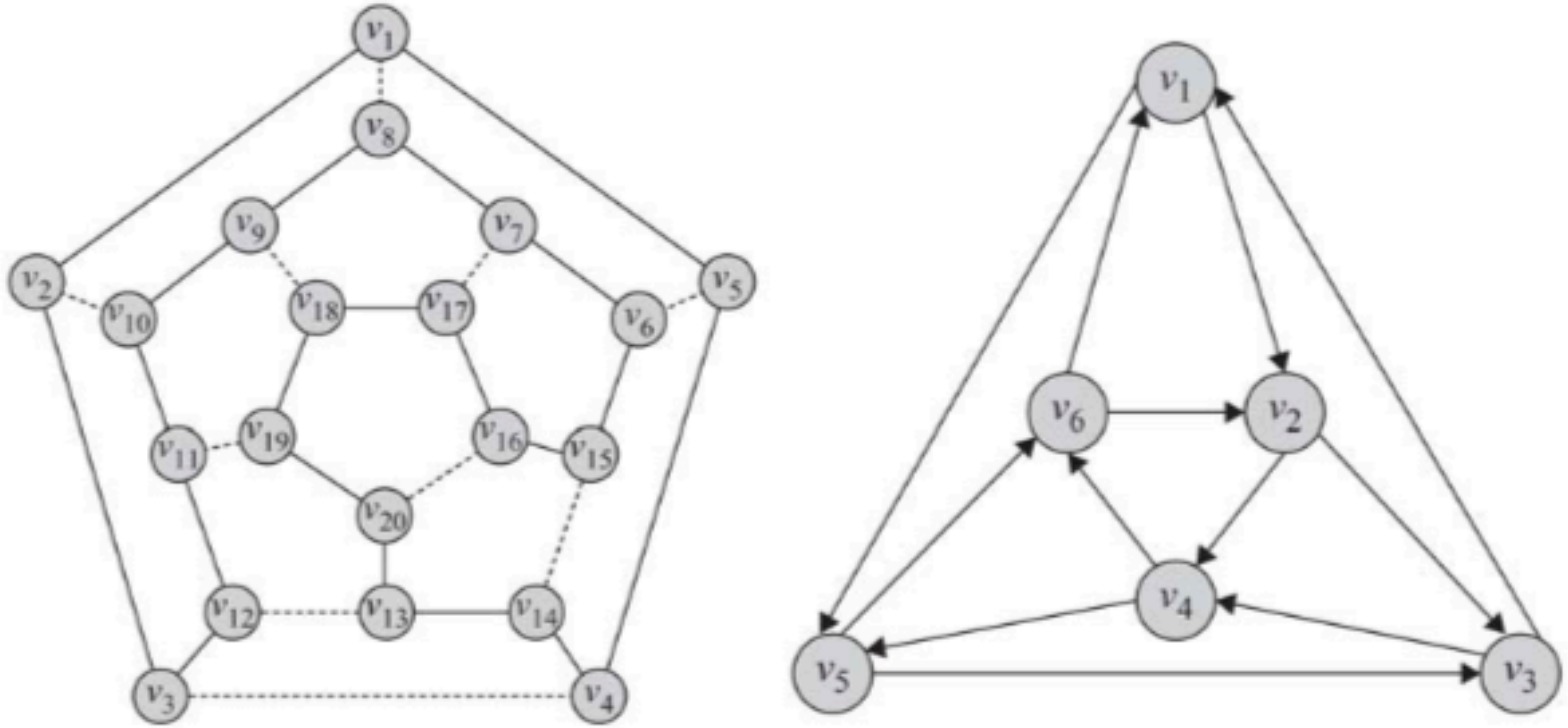
- Null Graph — both V and E are empty
- Empty Graph — E is empty
- Simple graphs — only one edge between any pair of nodes
- Multigraphs — multiple edges may be there between a pair of nodes



- Complete graph — Graph where we have edges between any pair of nodes — K_n
- Walk — A walk is a sequence of incident edges traversed one after the another — open walk (starting and end nodes not same) — closed walk when starting and end nodes are the same)
- Trail — A walk where no edges are repeated more than once. A closed trail is called a tour or a circuit
- Path — A walk where no edges and vertices are repeated — a closed path is a cycle
- Hamiltonian Cycle — A cycle which covers all vertices of the graph
- Eulerian Tour — A tour where all edges are covered
- Planar Graph — A graph that can be drawn without edges crossing over each other

Example

Picture taken from Social media mining by R. Zafarani et al



(a) Hamiltonian Cycle

(b) Eulerian Tour

Connectivity in a graph

- We say v_i is connected to v_j if there is a path between them. We also say they both are reachable

- A graph is said to be connected if every part of vertices are connected to each other

- In a directed graph if every pair of vertices are connected with respect to direction then it is called as strongly connected

- In a directed graph if we replaces all directed edges by undirected edges and the graph is connected then it is called as weakly connected

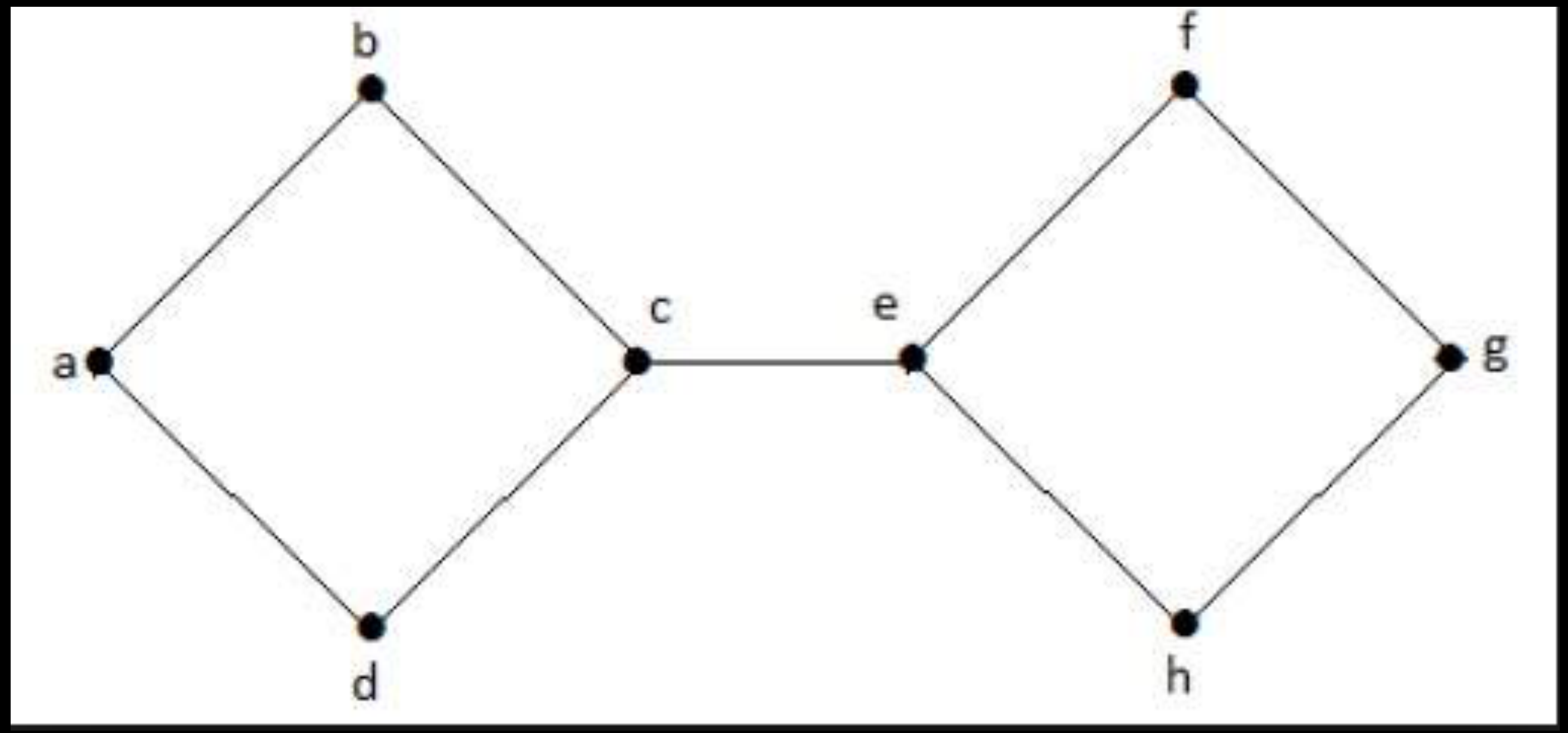
- In a connected graph there may be many paths between a pair of vertex. Shortest path is a path that contains minimum number of edges in it

- Diameter of a graph is the length of the longest shortest path between any two pair of vertices

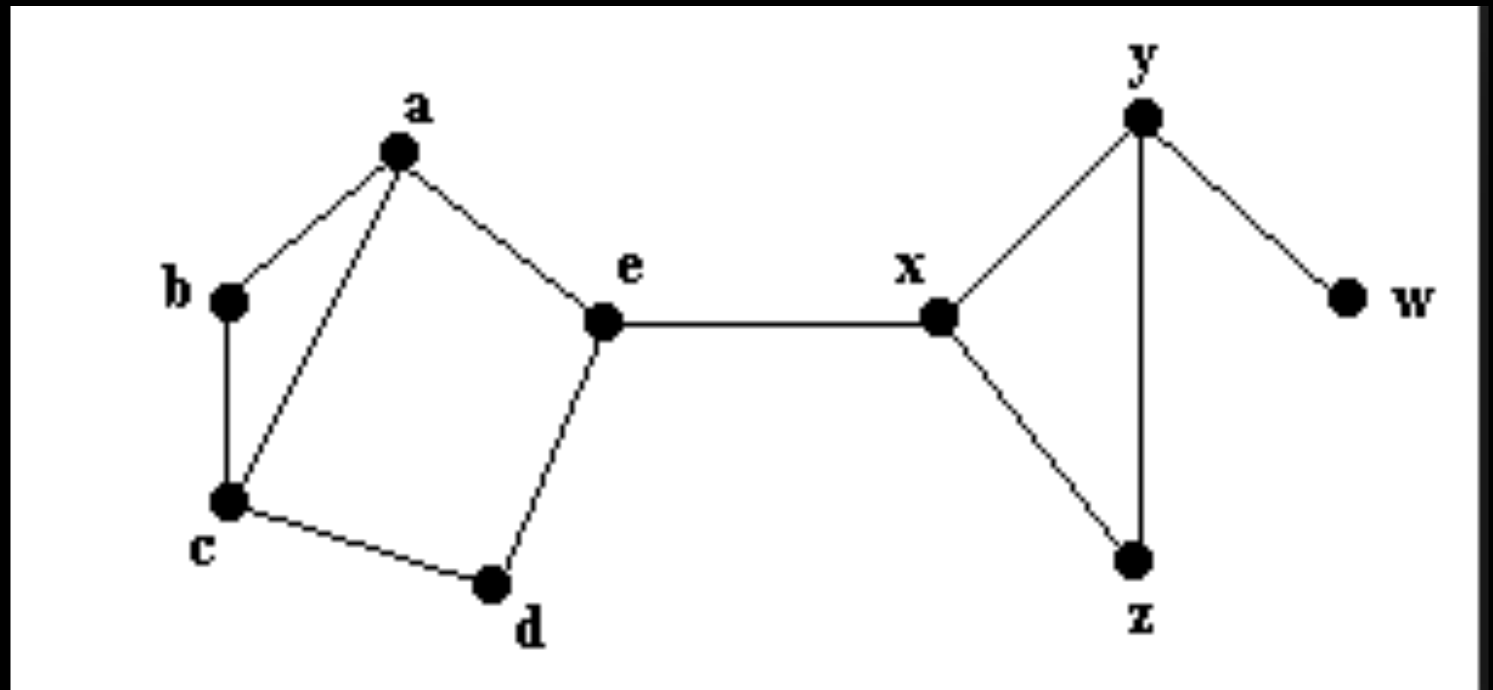
Tree and Forest

- Tree is a special graph which is connected and there are no cycles in it
- A graph which is not connected but that has many trees is called a forest
- In a tree $G=(V,E)$, E has always $n-1$ edges
- Spanning tree of a graph: Spanning tree is a subgraph of G which is a tree and that spans all the vertices of G
- Minimum Spanning tree: For a weighted connected graph there may be many spanning trees we choose the minimum one (according to the weights)
- Steiner Tree: For a weighted connected graph and a subset V' of V we choose the minimum spanning tree that spans all the vertices of V'

On Connectivity



- Cut edge — also called as bridge
- Cut vertex — also called articulation points / vertex cut / cut points



Cutpoints: e, x and y

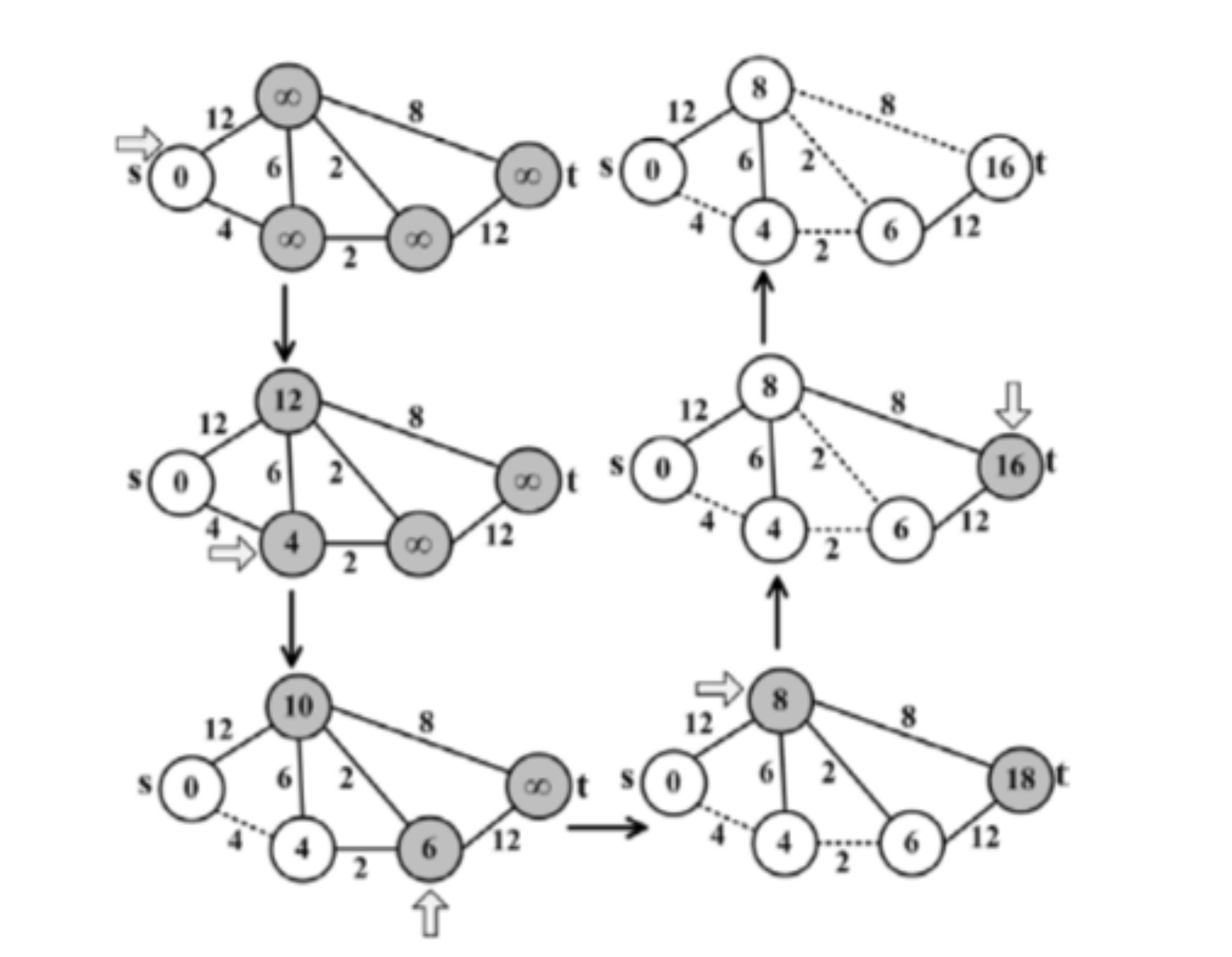
Graph Algorithms

- DFS traversal algorithm
- BFS traversal algorithm
- Dijkshtra’s shortest path algorithm
- Prim’s algorithm for MST
- Network flow algorithm — Ford-Fulkerson algorithm

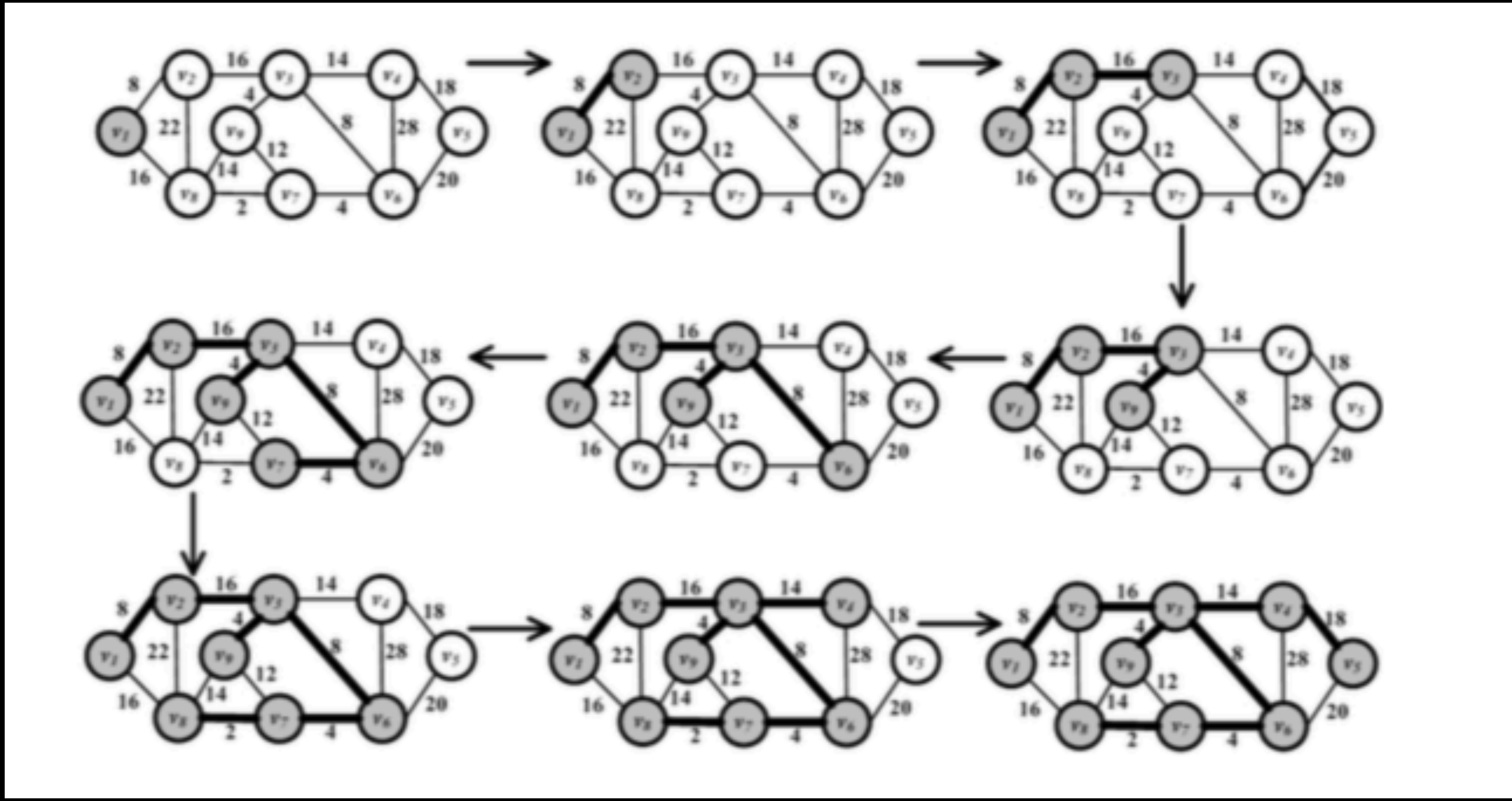
Graph Algorithms

- Traversal Algorithms — Suppose we want to find the follower graph we can use any of the traversal strategy to build the network
- Dijkshtra's Algorithm — Suppose we want to find the diameter of the graph to see how much is the spread of the graph
- MST algorithms — Ensuring the minimum connectivity of a group of actors to minimise the connections
- Flow algorithms — Suppose you want to transfer some bits of data to another system through various hops (through various bandwidths) and also in parallel. What will be the maximum bits that can be transferred at any point of time in the network?

Dijkshtra's Algorithm



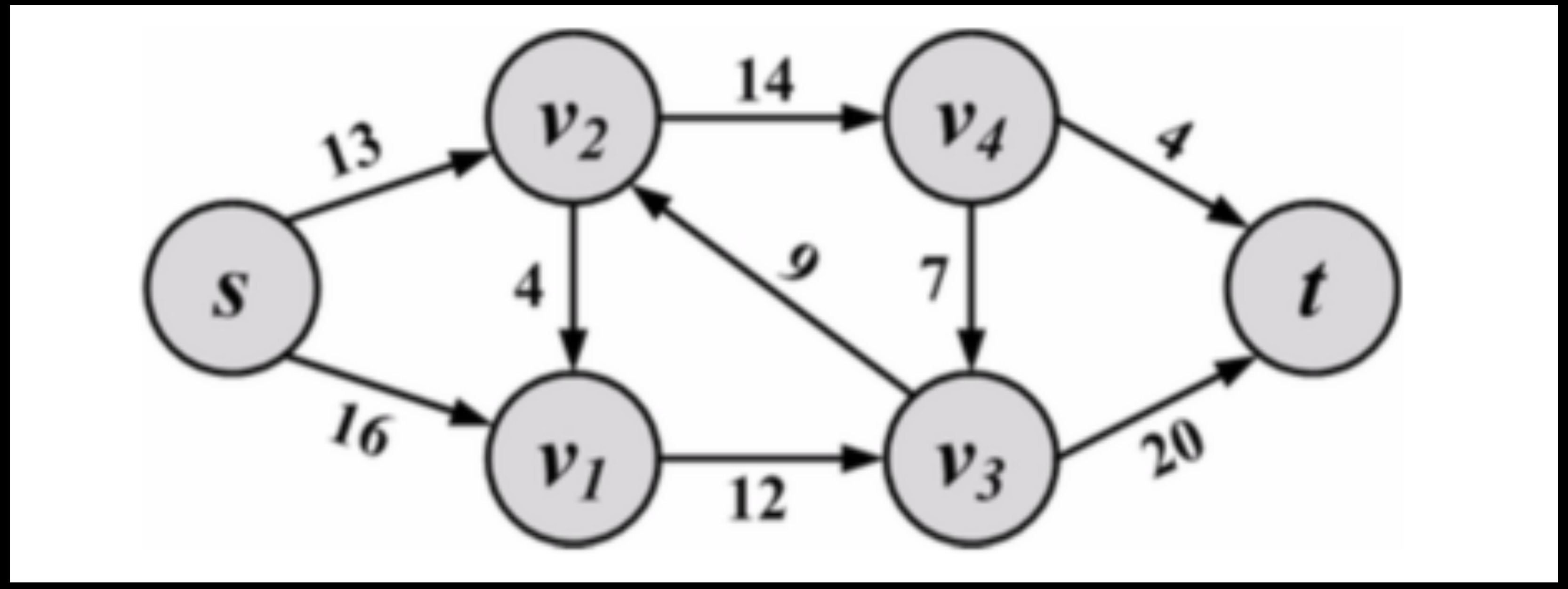
Prim's Algorithm - MST



LNMIIT

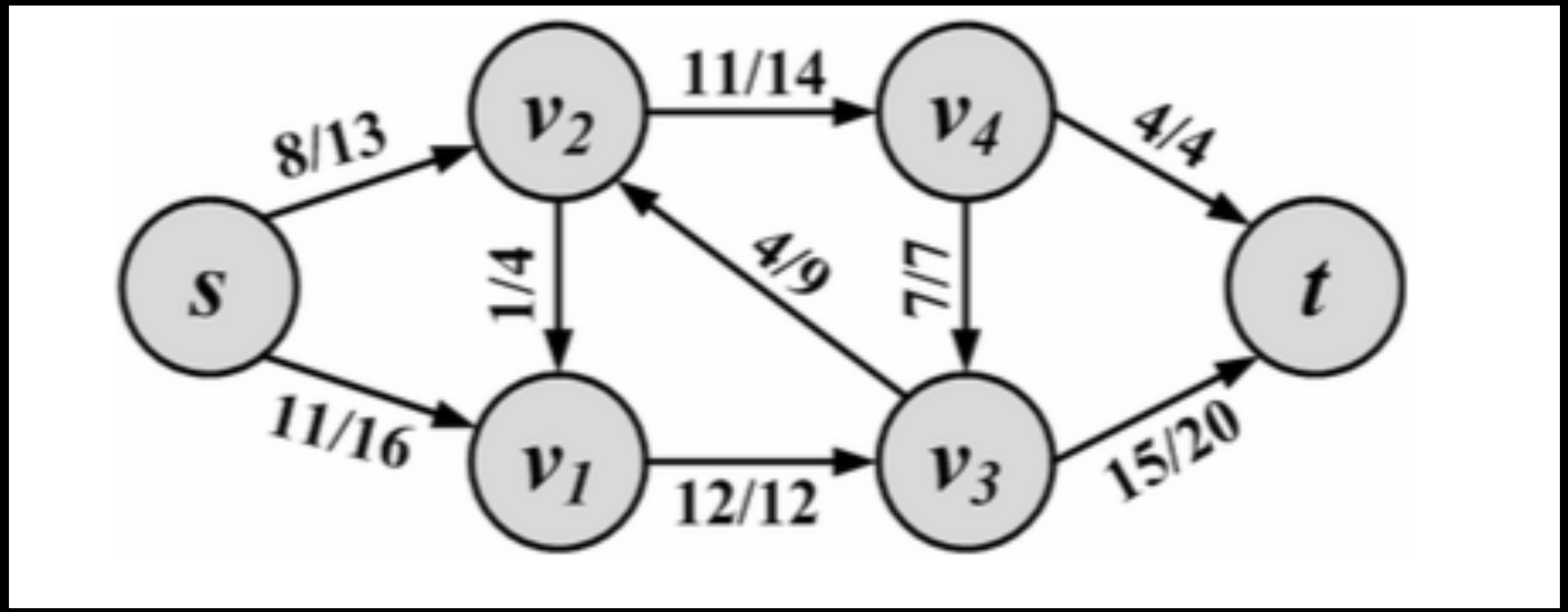
Flow Network

- A flow network is a weighted graph represented as $G=(V,E,c)$
  - where for all (u,v) in E, $c(u,v) \geq 0$ is the capacity of the edge
  - when (u,v) in E, (v,u) is not in E
  - s is the source node and t is the sink node



LNMIIT

Flow in a Flow Network

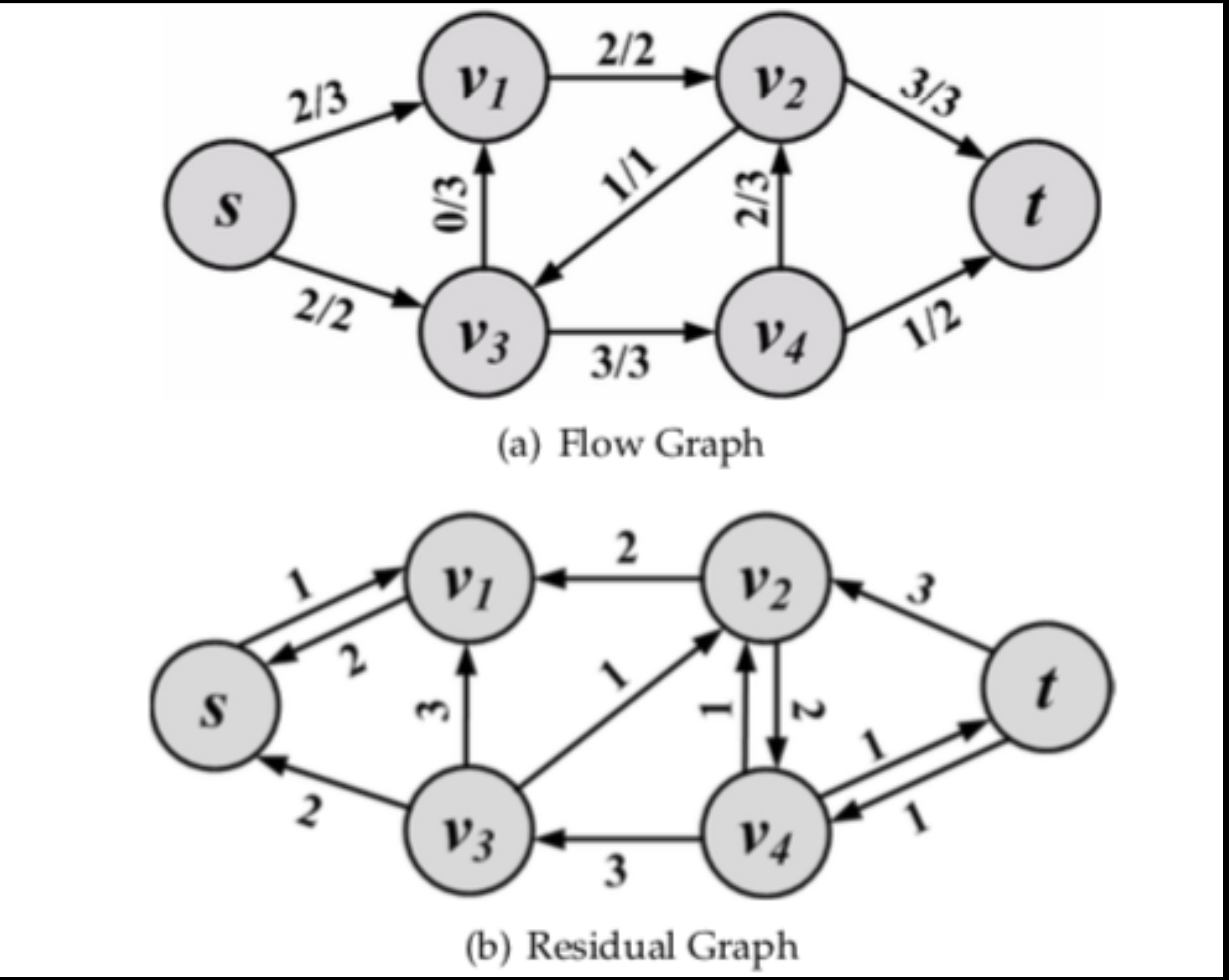


- For all *(u,v) in E, f(u,v) ≥ 0* is the flow through the edge
- *For all (u,v) in E, 0 ≤ f(u,v) ≤ c(u,v) is called capacity constraint*
- *Following is called flow conservation constraint*

$$\forall v \in V, v \notin \{s, t\}, \sum_{k:(k,v)\in E} f(k, v) = \sum_{l:(v,l)\in E} f(v, l)$$

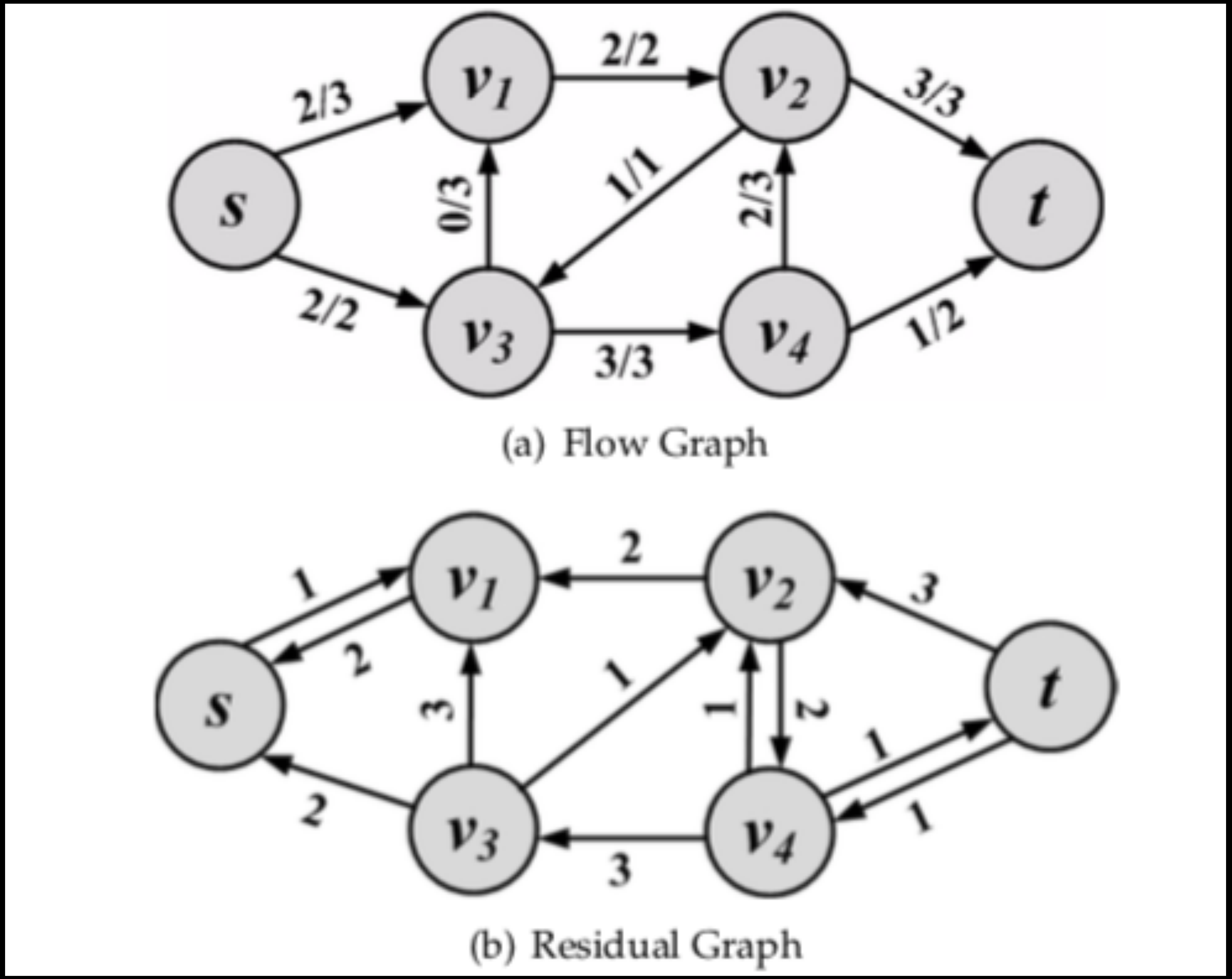
LNMIIT

Flow and Residual Network



(a) Flow Graph

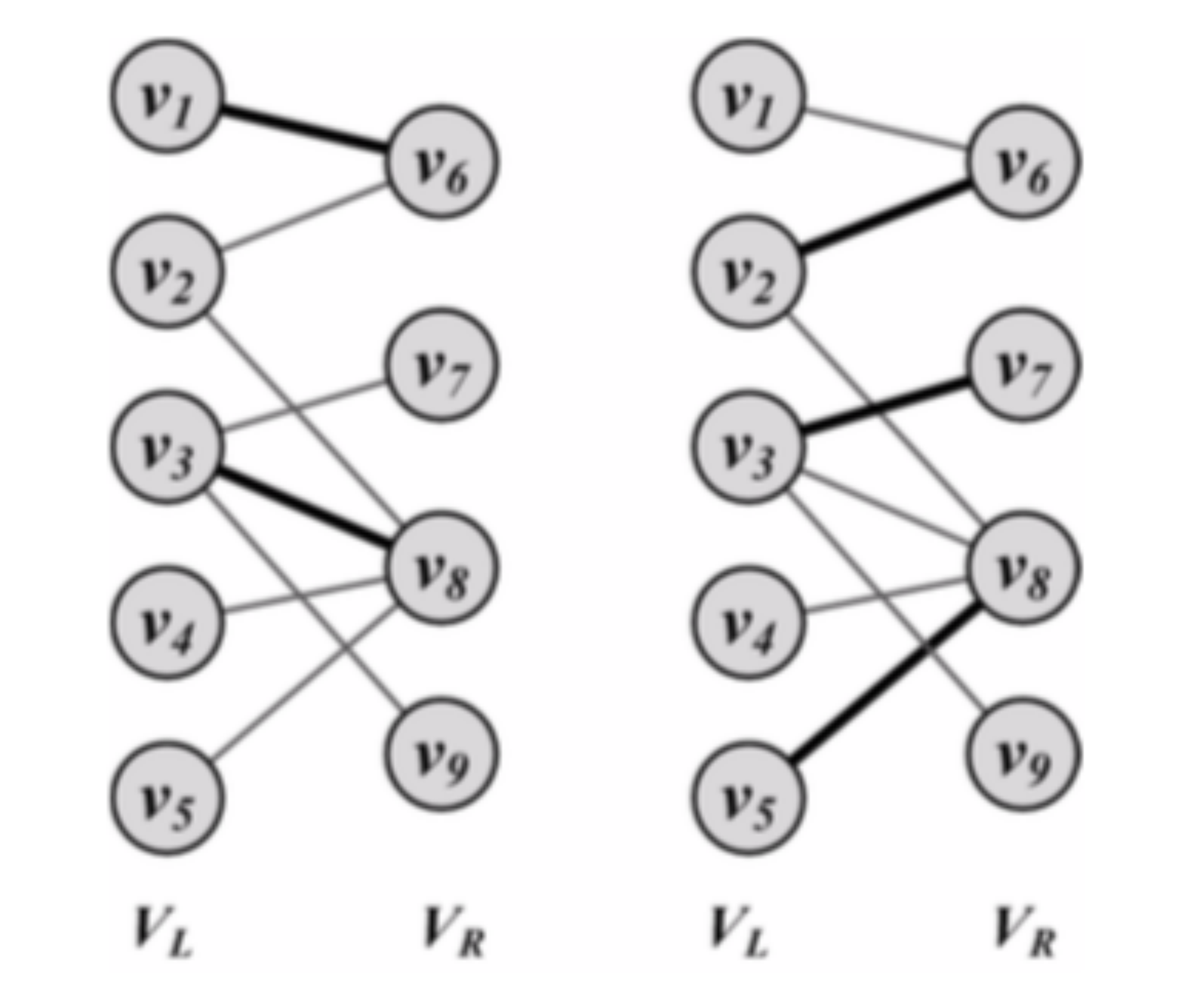
(b) Residual Graph



(a) Flow Graph

(b) Residual Graph

Maximum Bipartite Matching



LNMIIT