Information Networks

Introduction to networks

Lecture 1

Announcement

No Lecture this Thursday

Welcome!

- Introductions
  - My name in finnish: Panajotis Tsaparas
    - I am from Greece
    - I graduated from University of Toronto
      - Web searching and Link Analysis
    - In University of Helsinki for the past year
  - Tutor: Evimaria Terzi
    - also Greek

- Knowledge of Greek is not required

Course overview

- The course goal
  - To read some recent and interesting papers on information networks
  - Understand the underlying techniques
  - Think about interesting problems
- Prerequisites:
  - Mathematical background on discrete math, graph theory, probabilities
  - The course will be more “theoretical”, but your project may be more “practical”
- Style
  - Both slides and blackboard

Topics

- Measuring Real Networks
- Models for networks
- Scale Free and Small World networks
- Distributed hashing and Peer-to-Peer search
- The Web graph
  - Web crawling, searching and ranking
- Temporal analysis of data
- Gossip and Epidemics
- Clustering and classification
- Biological networks

Homework

- Two or three assignments of the following three types
  - Reaction paper
  - Problem Set
  - Presentation
- Project: Select your favorite network/algorithm/model and
  - do an experimental analysis
  - do a theoretical analysis
  - do a in-depth survey
- No final exam
- Final Grade: 50% assignments, 50% project (or 60%,40%)
- Tutorials: will be arranged on demand

Web page

- Web page has been (partially) updated

http://www.cs.helsinki.fi/u/tsaparas/InformationNetworks/

What is an information network?

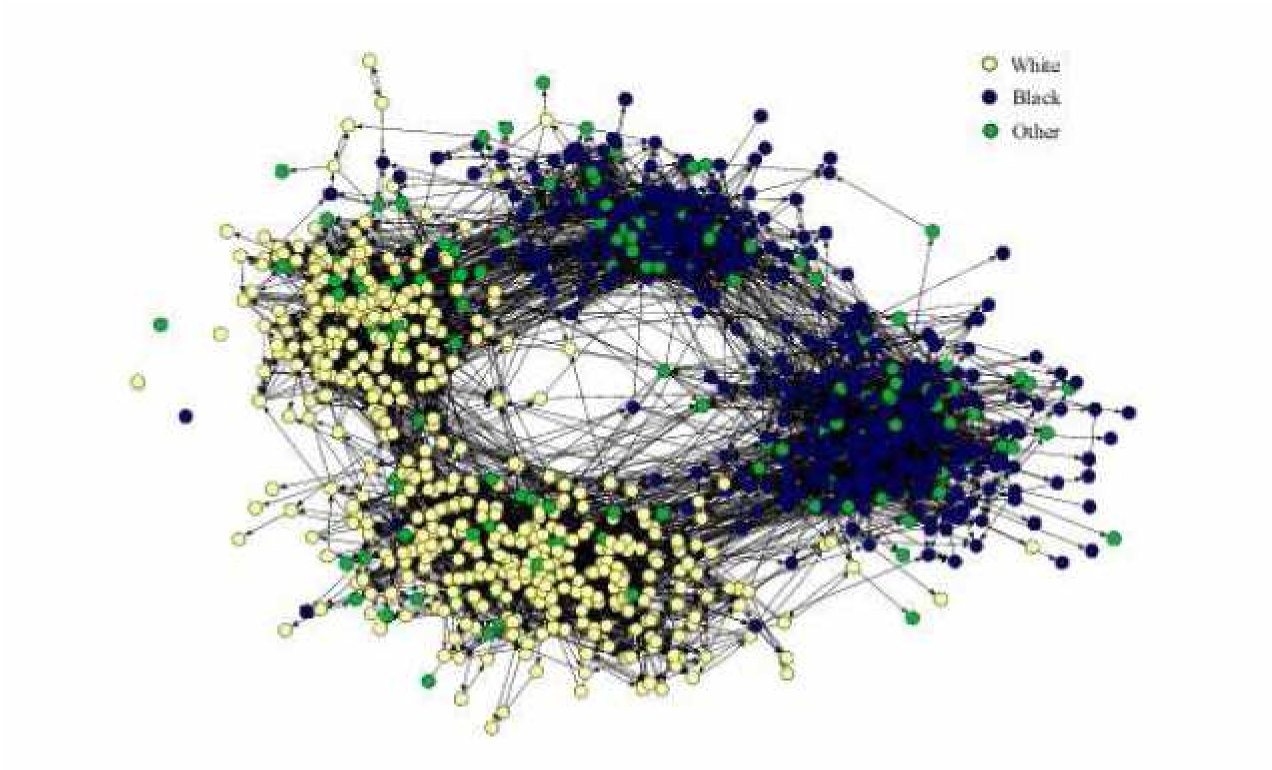
- Network: a collection of entities that are interconnected
  - A link (edge) between two entities (nodes) denotes an interaction between two entities
  - We view this interaction as information exchange, hence, Information Networks
  - The term encompasses more general networks

Why do we care about networks?

- Because they are everywhere
  - more and more systems can be modeled as networks
- Because they are growing
  - large scale problems
- Because we have the computational power to study them
  - task: to develop the tools

Social Networks

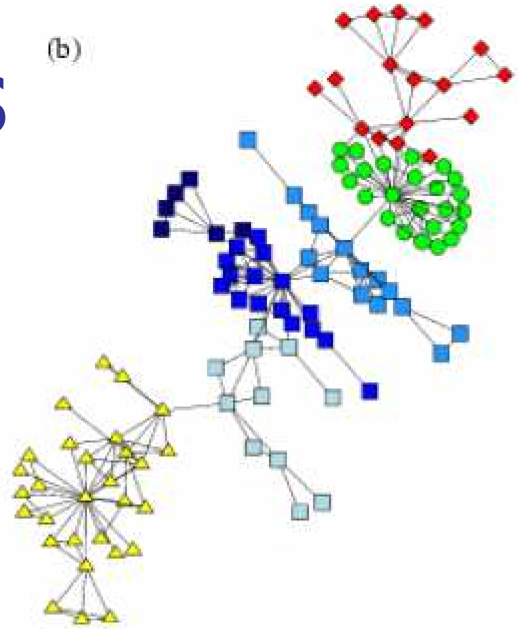
- Links denote a social interaction
  - Networks of acquaintances



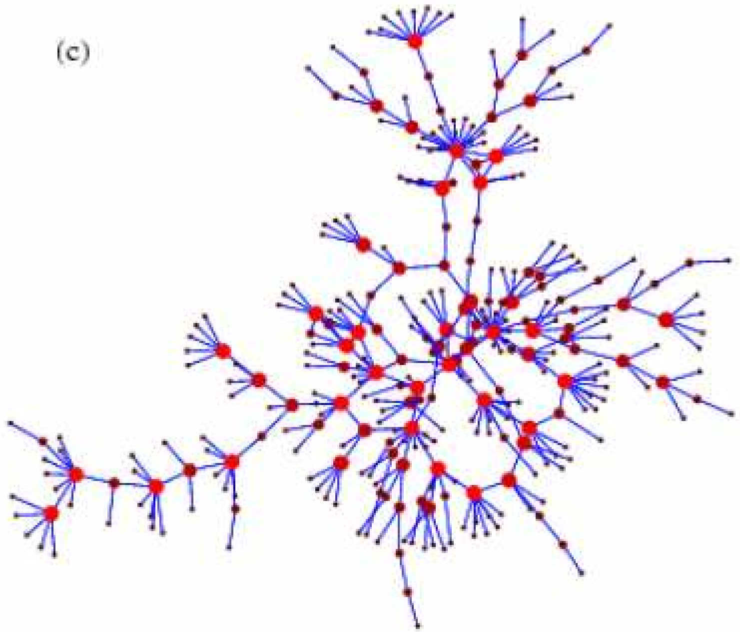
Other Social networks

- actor networks
- co-authorship networks
- director networks
- phone-call networks
- e-mail networks
- IM networks
  - Microsoft buddy network
- Bluetooth networks
- sexual networks

(b)

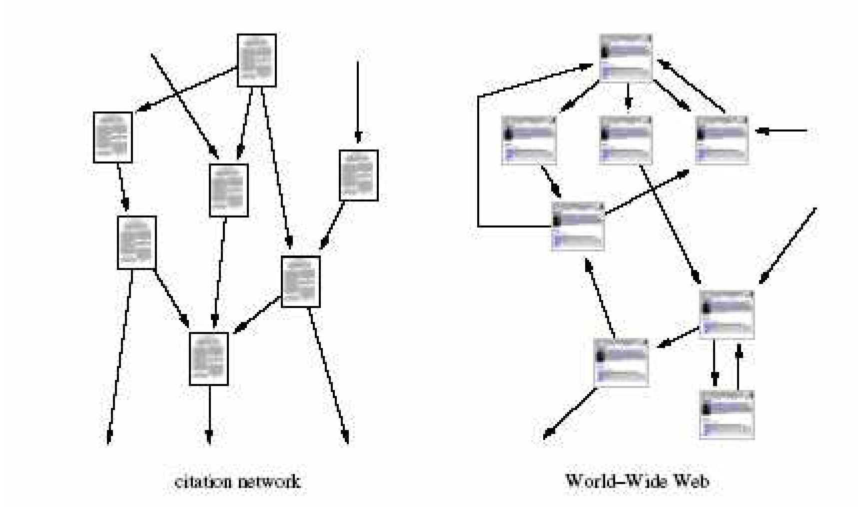


(c)



Knowledge (Information) Networks

- Nodes store information, links associate information
  - Citation network (directed acyclic)
  - The Web (directed)

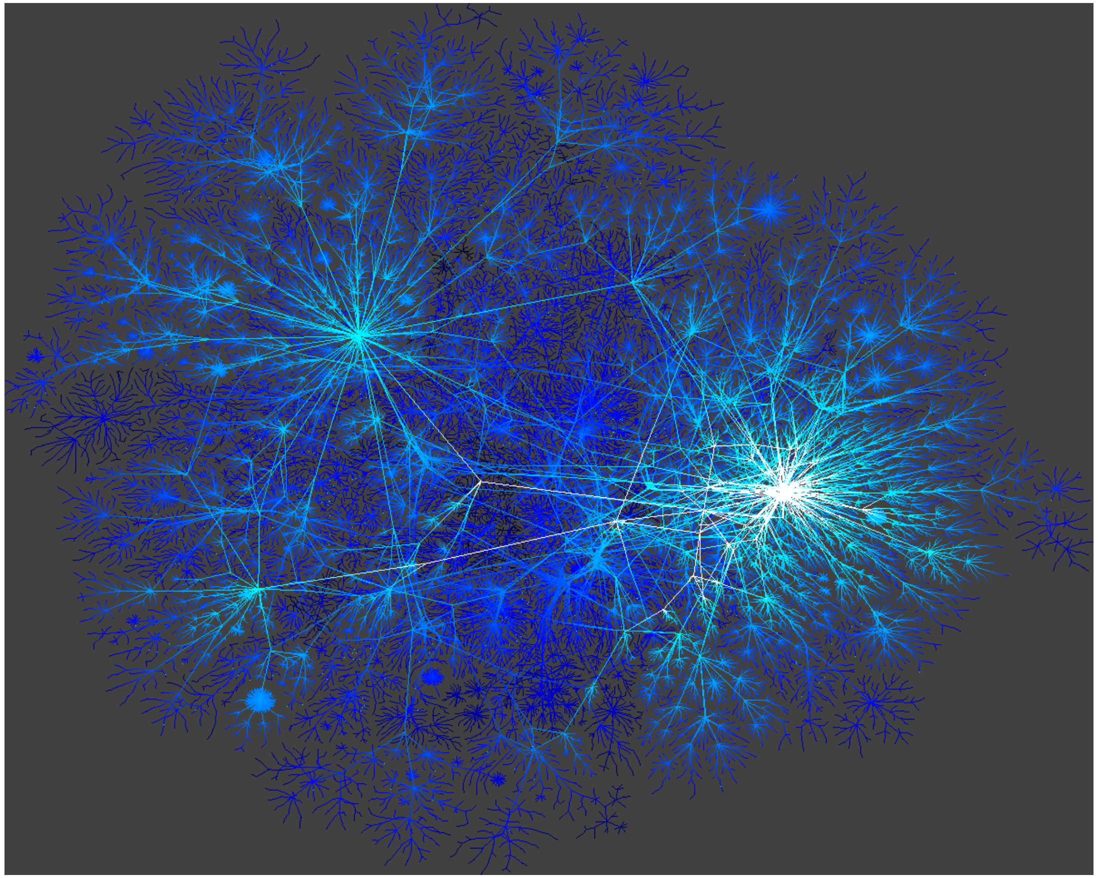


Other Information Networks

- Peer-to-Peer networks
- Word networks
- Networks of Trust
  - epinions

Technological networks

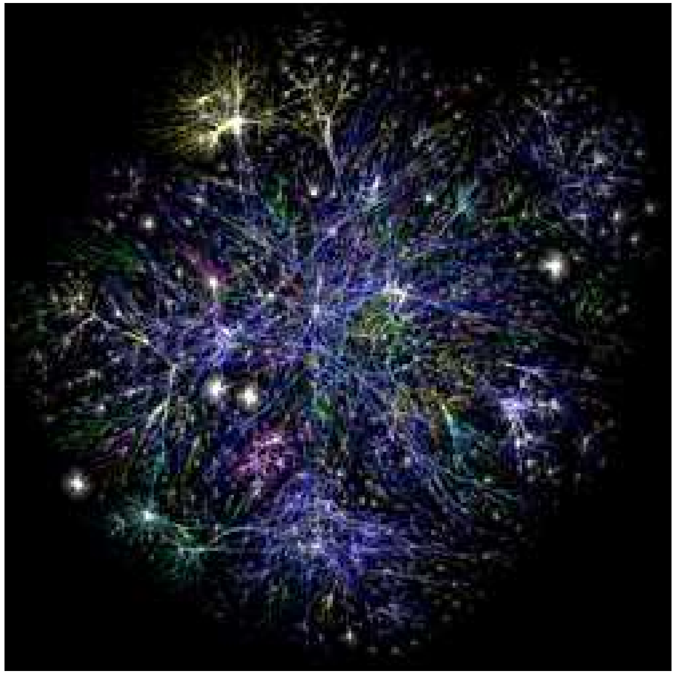
- Networks built for distribution of commodity
  - The Internet
    - router level
    - AS level



ISP network

The Internet

The Opte Project

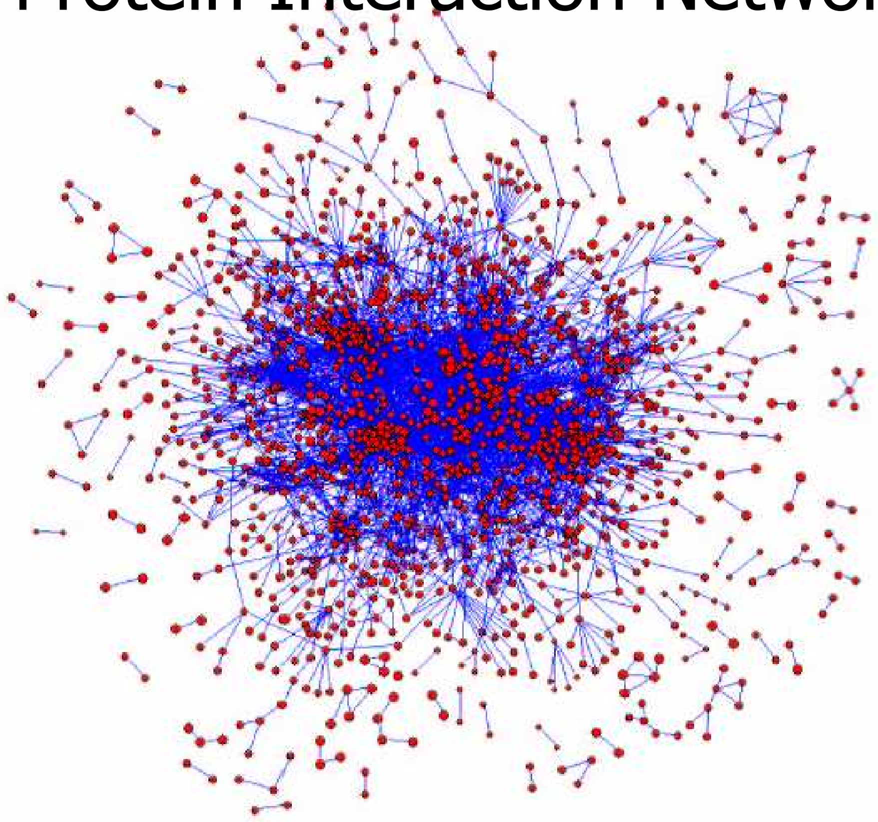


Other Technological networks

- Power Grids
- Airline networks
- Telephone networks
- Transportation Networks
  - roads, railways, pedestrian traffic
- Software networks

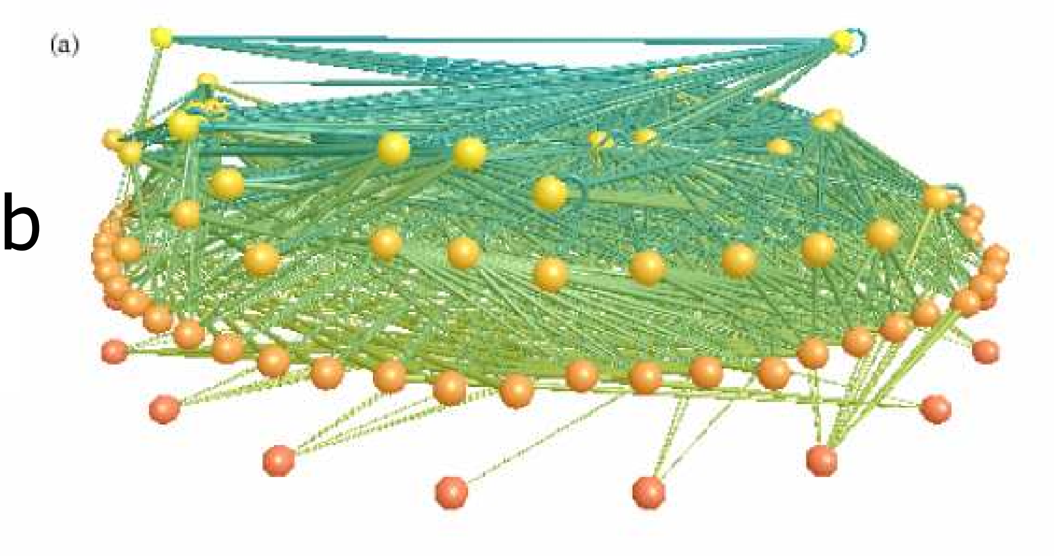
Biological networks

- Biological systems represented as networks
  - Protein-Protein Interaction Networks

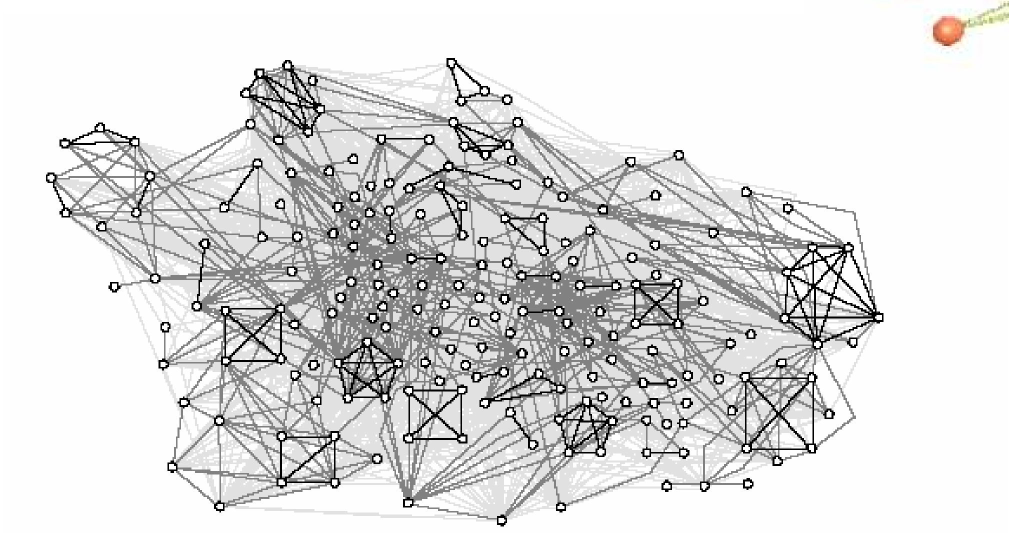


Other Biological networks

- Gene regulation networks



- The Food Web



- Neural Networks

Now what?

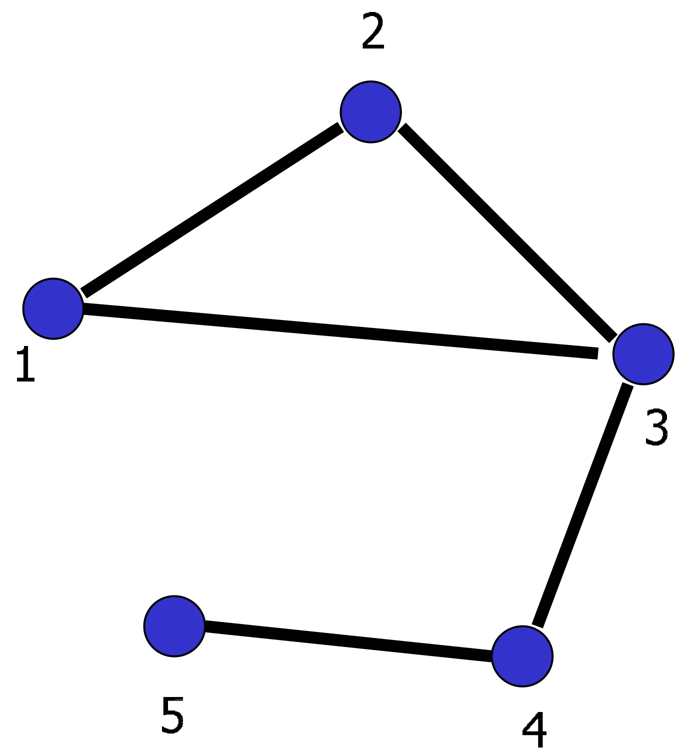
- The world is full with networks. What do we do with them?
  - understand their topology and measure their properties
  - study their evolution and dynamics
  - create realistic models
  - create algorithms that make use of the network structure

Mathematical Tools

- Graph theory
- Probability theory
- Linear Algebra

Graph Theory

- Graph G=(V,E)
  - V = set of vertices
  - E = set of edges

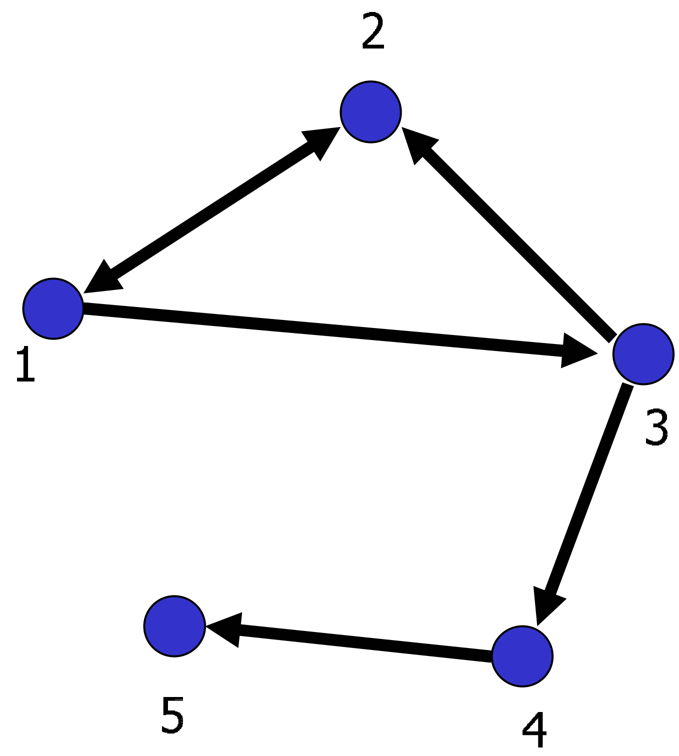


undirected graph

E={(1,2),(1,3),(2,3),(3,4),(4,5)}

Graph Theory

- Graph G=(V,E)
  - V = set of vertices
  - E = set of edges

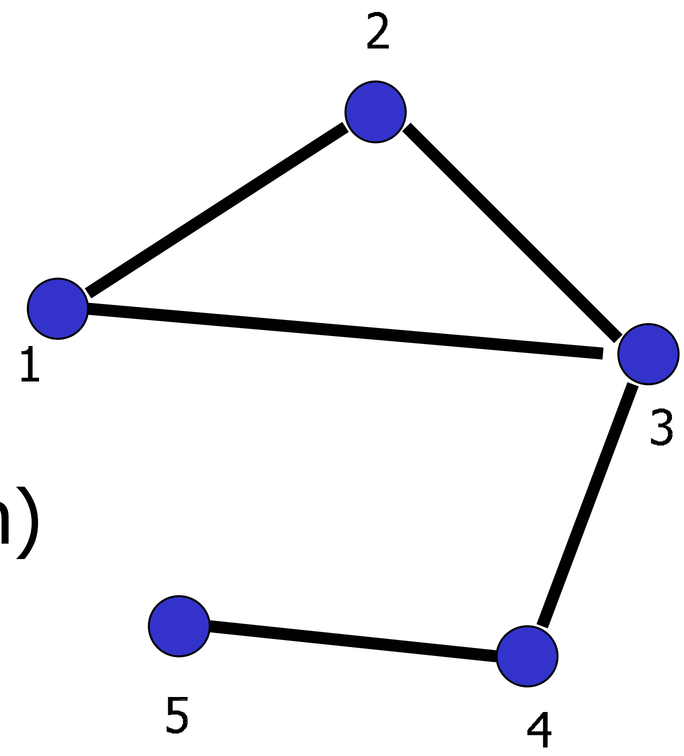


directed graph

E={‹1,2›, ‹2,1› ‹1,3›, ‹3,2›, ‹3,4›, ‹4,5›}

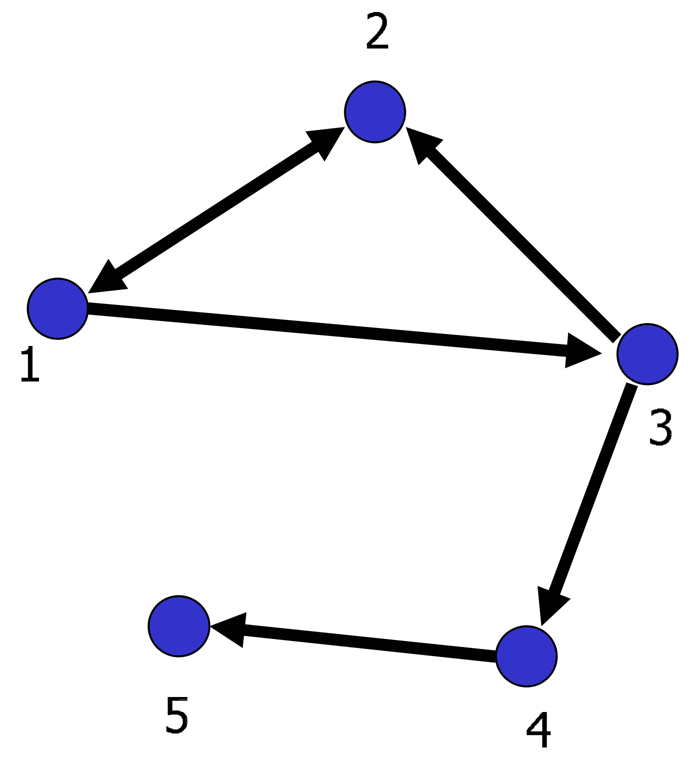
Undirected graph

- degree d(i) of node i
  - number of edges incident on node i
- degree sequence (distribution)
  - [d(i),d(2),d(3),d(4),d(5)]
  - [2,2,2,1,1]



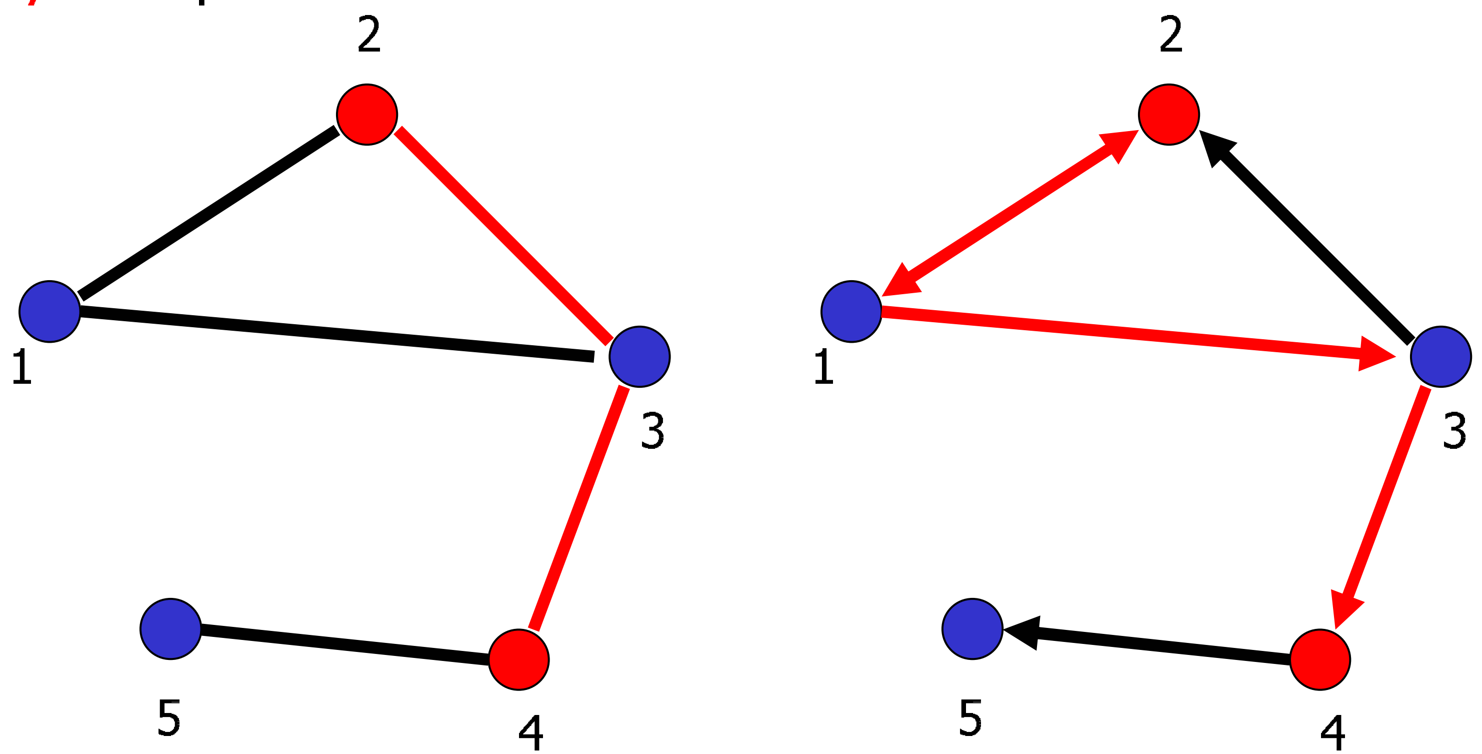
Directed Graph

- in-degree $d_{in}(i)$ of node i
  - number of edges pointing to node i
- out-degree $d_{out}(i)$ of node i
  - number of edges leaving node i
- in-degree sequence (distribution)
  - [1,2,1,1,1]
- out-degree sequence (distribution)
  - [2,1,2,1,0]



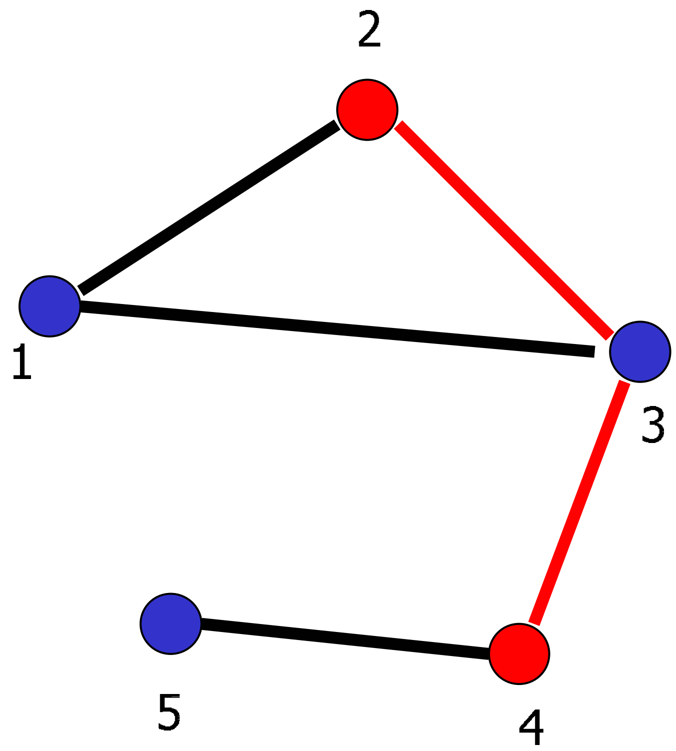
Paths

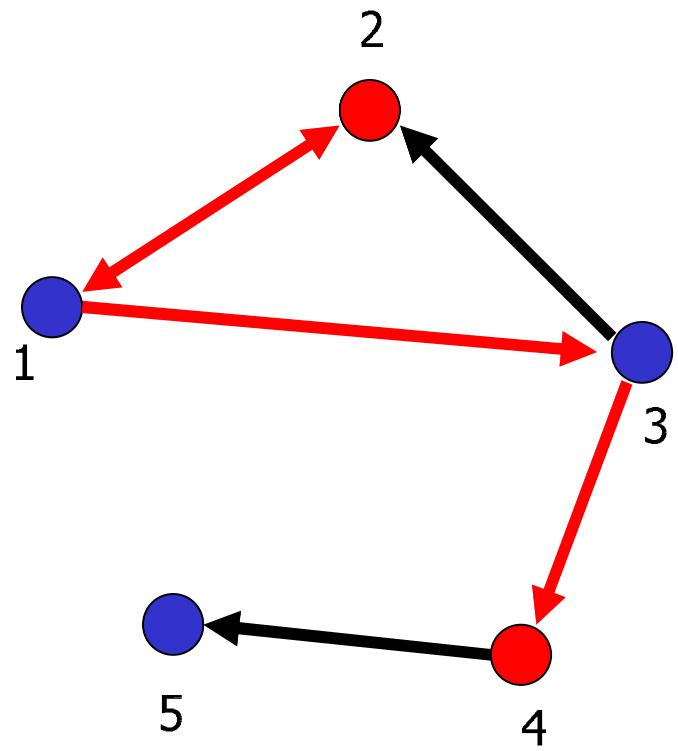
- Path from node i to node j: a sequence of edges (directed or undirected from node i to node j)
  - path length: number of edges on the path
  - nodes i and j are connected
  - cycle: a path that starts and ends at the same node



Shortest Paths

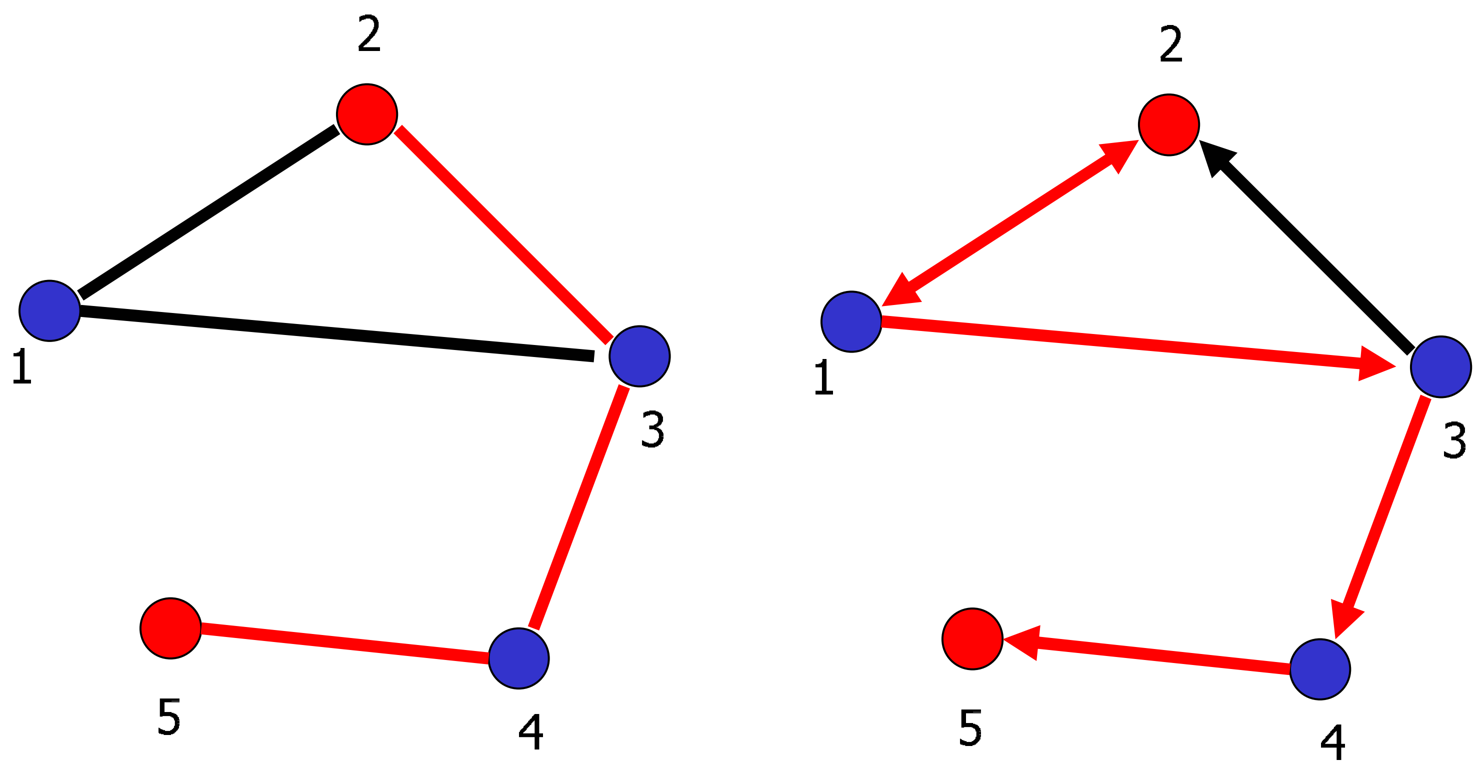
- Shortest Path from node i to node j
  - also known as BFS path, or geodesic path





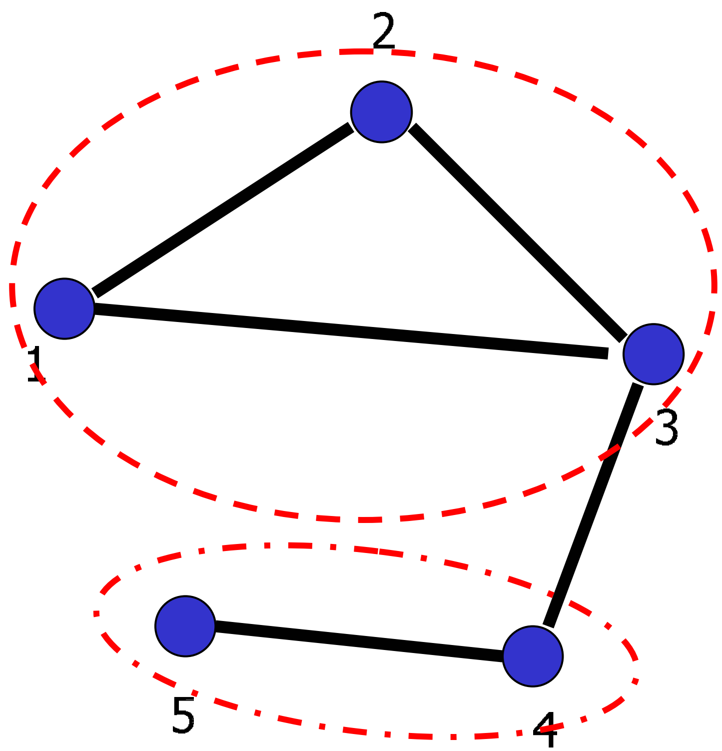
Diameter

- The longest shortest path in the graph



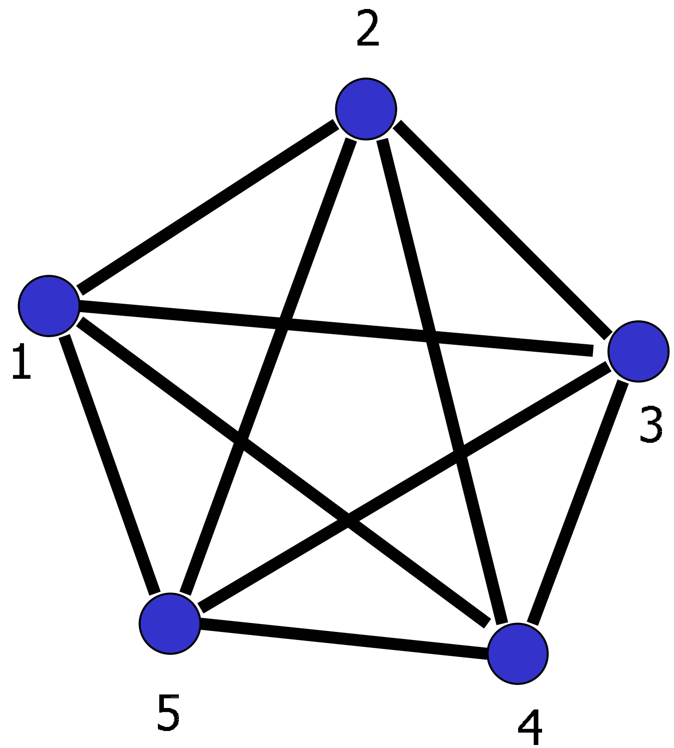
Undirected graph

- Connected graph: a graph where there every pair of nodes is connected
- Disconnected graph: a graph that is not connected
- Connected Components: subsets of vertices that are connected



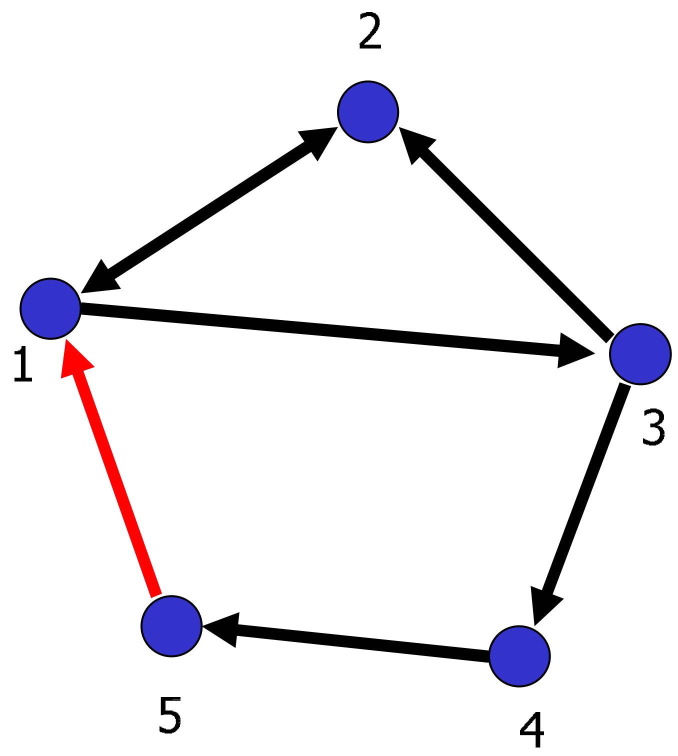
Fully Connected Graph

- Clique K_n
- A graph that has all possible n(n-1)/2 edges



Directed Graph

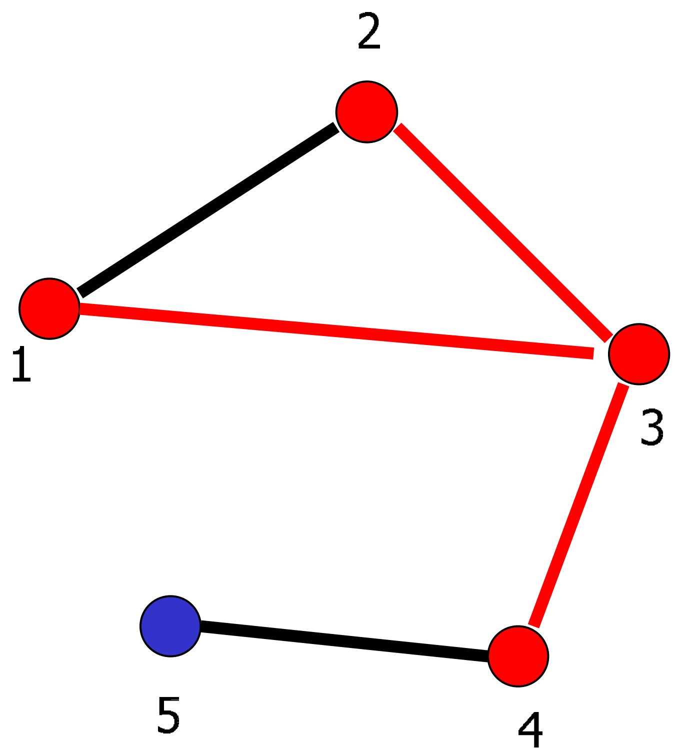
- Strongly connected graph: there exists a path from every i to every j
- Weakly connected graph: If edges are made to be undirected the graph is connected



Subgraphs

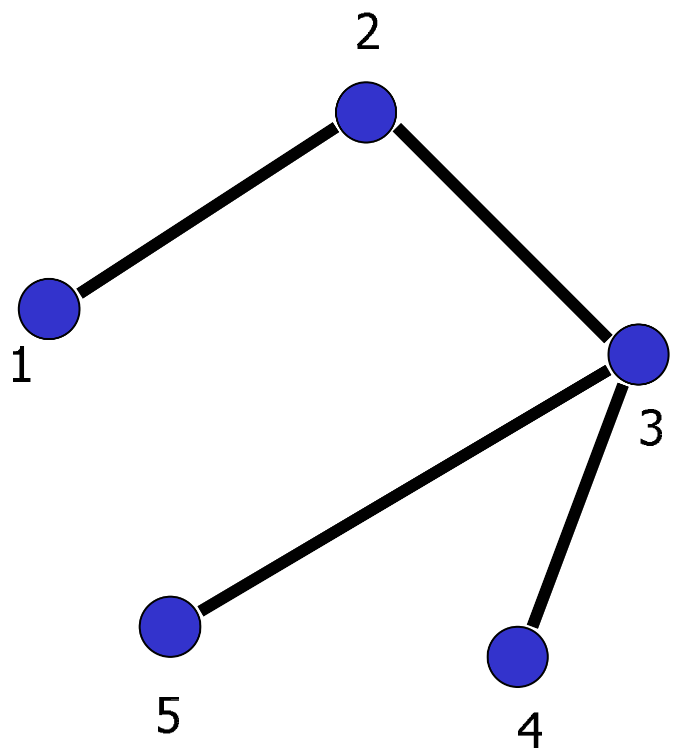
- Subgraph: Given $V' \subseteq V$, and $E' \subseteq E$, the graph $G'=(V',E')$ is a subgraph of G.

- Induced subgraph: Given $V' \subseteq V$, let $E' \subseteq E$ is the set of all edges between the nodes in V'. The graph $G'=(V',E')$, is an induced subgraph of G



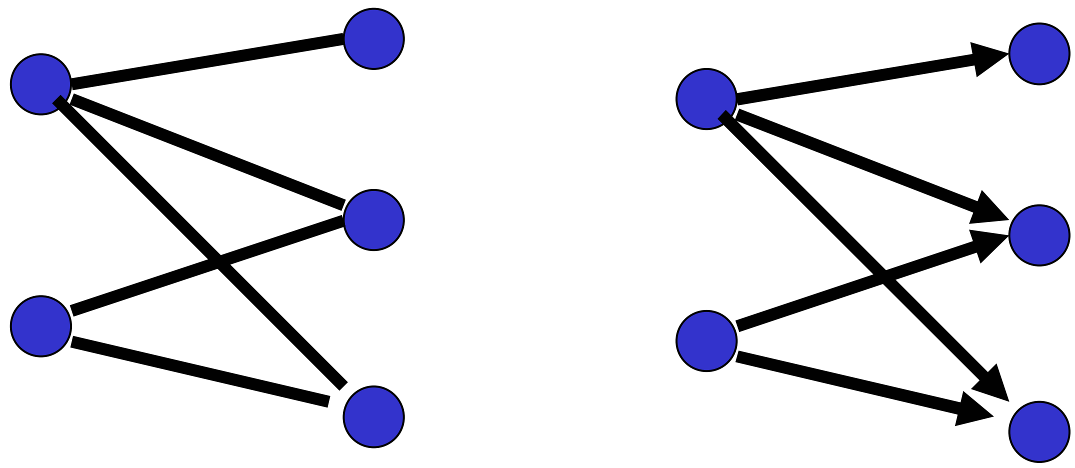
Trees

- Connected Undirected graphs without cycles



Bipartite graphs

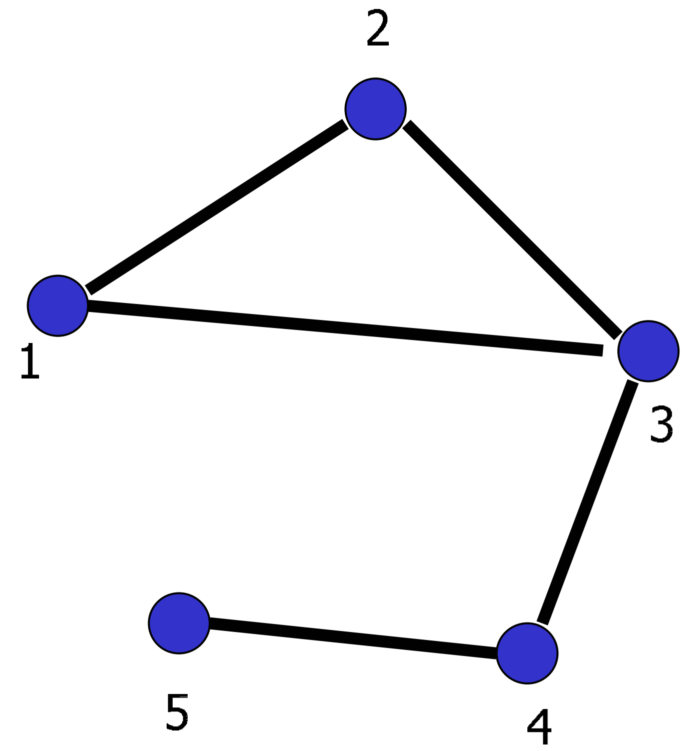
- Graphs where the set V can be partitioned into two sets L and R, such that all edges are between nodes in L and R, and there is no edge within L or R



Linear Algebra

- Adjacency Matrix
  - symmetric matrix for undirected graphs

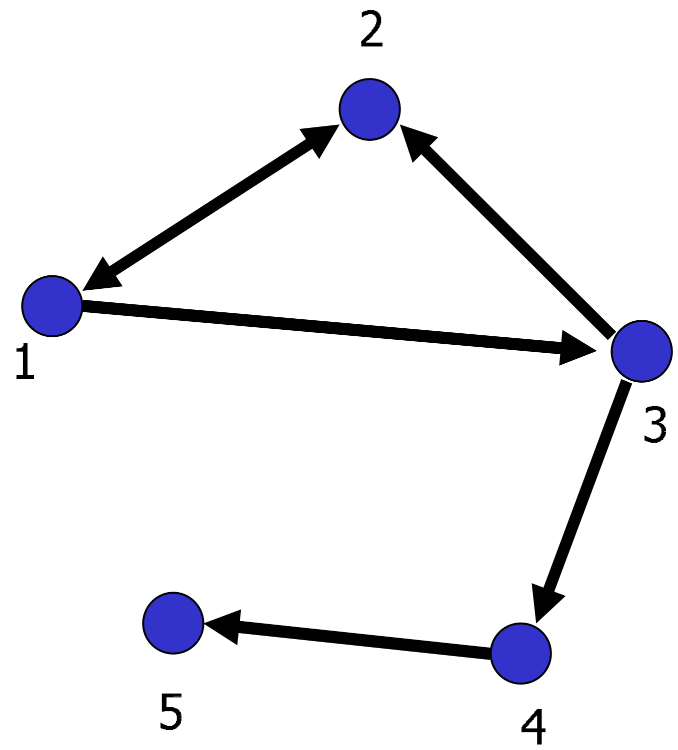
$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$



Linear Algebra

- Adjacency Matrix
  - unsymmetric matrix for undirected graphs

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



Eigenvalues and Eigenvectors

- The value λ is an eigenvalue of matrix A if there exists a non-zero vector x, such that $Ax=\lambda x$. Vector x is an eigenvector of matrix A
  - The largest eigenvalue is called the principal eigenvalue
  - The corresponding eigenvector is the principal eigenvector
  - Corresponds to the direction of maximum change

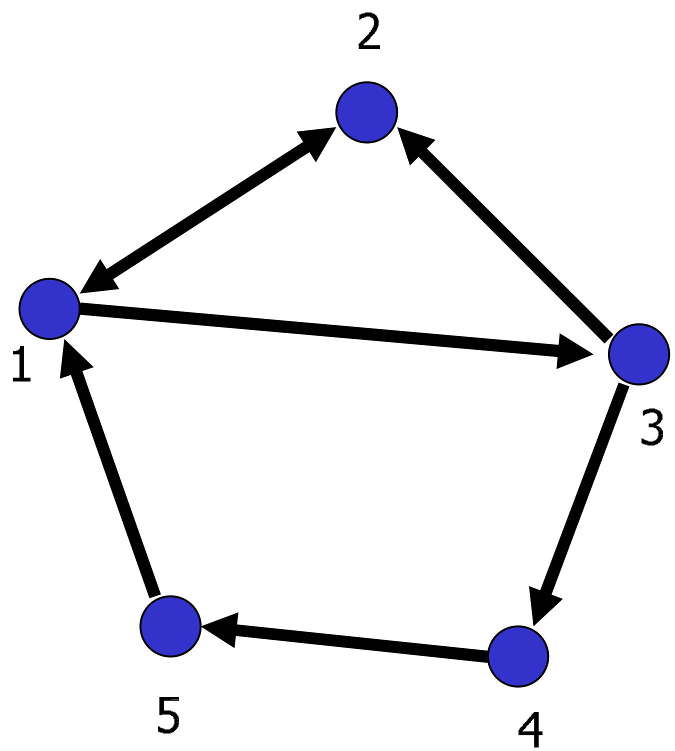
Random Walks

- Start from a node, and follow links uniformly at random.
- Stationary distribution: The fraction of times that you visit node i, as the number of steps of the random walk approaches infinity
  - if the graph is strongly connected, the stationary distribution converges to a unique vector.

Random Walks

- stationary distribution: principal left eigenvector of the normalized adjacency matrix
  - x = xP
  - for undirected graphs, the degree distribution

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



Probability Theory

- Probability Space: pair ‹Ω,P›
  - Ω: sample space
  - P: probability measure over subsets of Ω
- Random variable X: Ω→R
  - Probability mass function P[X=x]
- Expectation

$$E[X] = \sum_{x \in \Omega} x P[X = x]$$

Classes of random graphs

- A class of random graphs is defined as the pair $\langle G_n, P \rangle$ where G_n the set of all graphs of size n, and P a probability distribution over the set G_n
- Erdös-Renyi graphs: each edge appears with probability p
  - when $p=1/2$, we have a uniform distribution

Asymptotic Notation

- For two functions f(n) and g(n)
  - $f(n) = O(g(n))$ if there exist positive numbers c and N, such that $f(n) \leq c\, g(n)$, for all $n \geq N$
  - $f(n) = \Omega(g(n))$ if there exist positive numbers c and N, such that $f(n) \geq c\, g(n)$, for all $n \geq N$
  - $f(n) = \Theta(g(n))$ if $f(n)=O(g(n))$ and $f(n)=\Omega(g(n))$
  - $f(n) = o(g(n))$ if $\lim f(n)/g(n) = 0$, as $n \to \infty$
  - $f(n) = \omega(g(n))$ if $\lim f(n)/g(n) = \infty$, as $n \to \infty$

P and NP

- **P**: the class of problems that can be solved in polynomial time
- **NP**: the class of problems that can be verified in polynomial time
- **NP**-hard: problems that are at least as hard as any problem in **NP**

Approximation Algorithms

- NP-optimization problem: Given an instance of the problem, find a solution that minimizes (or maximizes) an objective function.
- Algorithm A is a factor c approximation for a problem, if for every input x,

  $A(x) \leq c\ OPT(x)$ (minimization problem)

  $A(x) \geq c\ OPT(x)$ (maximization problem)

References

- M. E. J. Newman, The structure and function of complex networks, SIAM Reviews, 45(2): 167-256, 2003