Graphs and Networks (I)

Discrete Mathematics and Optimization
Bioinformatics

Outline

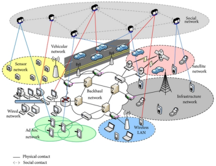
- Basic definitions
- Degrees, distance, connectivity
- Trees
  - Characterization of trees
  - Counting trees: Prüfer code

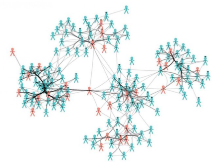
Graphs

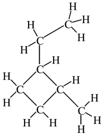
Graph $G = (V, E)$

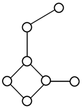
- V set of nodes or vertices
- E set of edges (pairs of nodes) (can be directed, multiple, loops,...)

They can model biological, communication, internet,... networks.



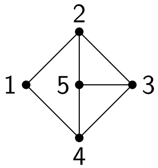






G_1

Specifying a Graph



- Adjacency matrix: A, $n \times n$ $(0,1)$–matrix $A_{i,j} = \begin{cases} 1 & ij \in E(G) \\ 0 & ij \notin E(G). \end{cases}$

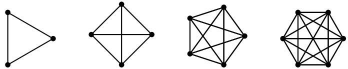
$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

- Adjacency lists:

1	2,4
2	1,3,5
3	2,4,5
4	1,3,5
5	2,3,4

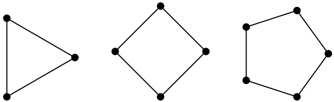
Examples of graphs

- complete graph K_n: all pairs of edges are present.

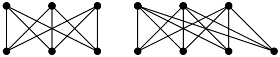


- Path P_n

- Cycle C_n



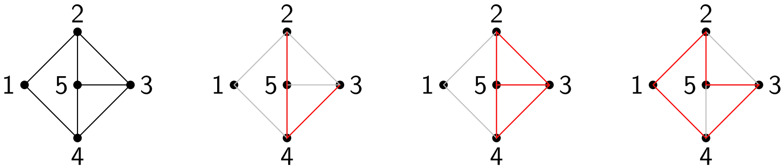
- Complete bipartite graph $K_{n,m}$



Subgraphs

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

- A subgraph of G is $H = (V', E')$ with $V' \subset V$ and $E' \subset E$.
- A induced subgraph of G is a subgraph $H = (V', E')$ which contains all edges of G joining vertices in V'
- A spanning subgraph of G is a subgraph $H = (V', E')$ with $V' = V$ and $E' \subset E$.
- A Hamiltonian cycle is a spanning subgraph which is a cycle



Degrees

- The neighborhood $N(x)$ of a vertex $x \in V(G)$ is the set of vertices adjacent to x
- The degree $d(x)$ of a vertex $x \in V(G)$ is the number of vertices adjacent to x (the number of edges incident to x).
- The maximum degree of G is $\Delta(G) = \max\{d(x) : x \in V\}$.
- The minimum degree of G is $\delta(G) = \min\{d(x) : x \in V\}$.
- The average degree of G is $d(G) = \frac{1}{|V|}\sum_{x \in V} d(x)$
- A graf is d–regular if all vertices have degree d.
  - 1–regular: collection of edges (matching)
  - 2–regular: collection of cycles
  - 3–regular: cubic graph.

Lemma (Handshaking Lemma)

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

- Every cubic graph has an even number of vertices.
- The average degree is $d(G) = \frac{2|E|}{|V|}$

Distance and diameter

- The distance between two vertices $u, v \in V(G)$ of a connected G is the length of the shortest path connecting them.
  - $d(u, v) = 0$ if and only if $u = v$.
  - $d(u, v) = d(v, u)$
  - $d(u, v) \leq d(u, w) + d(w, v)$.
- The diameter of a graph is the maximum distance between pairs of vertices.
  - $D(G) = 1$ if and only if G is a complete graph.
  - $D(G) = 2$ if and only if every pair of nonadjacent nodes have a common neighbour.
  - $D(G) = |V| - 1$ if and only if G is a path

Connectivity

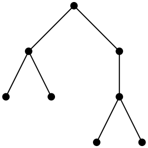
- A graph G is connected if every pair of vertices is connected by a path in G.
- A connected component of a graph is a maximum subgraph which is connected

Lemma

A graph G is connected if and only if, for every $X \subsetneq V$, there is an edge of G with an end vertex in X and one in $V \setminus X$.

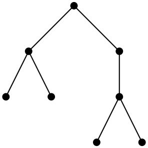
Trees

A tree $T = (V, E)$ is a connected and acyclic graph.



Trees

A tree $T = (V, E)$ is a connected and acyclic graph.



A leaf in a tree is a vertex of degree 1.

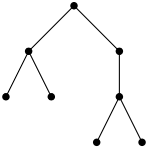
Lemma

Every tree has at least two leaves.

By removing a leaf from a tree T we obtain a subtree of T

Trees

A tree $T = (V, E)$ is a connected and acyclic graph.



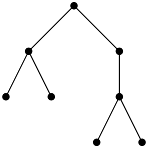
Proposition

The following are equivalent:

- T *is a tree.*
- T *is edge–minimal connected graph.*
- T *is edge–maximal acyclic graph.*
- T *is connected with* n *vertices and* $n - 1$ *edges.*
- T *is acyclic with* n *vertices and* $n - 1$ *edges.*

Trees

A tree $T = (V, E)$ is a connected and acyclic graph.



A spannning tree of a graph G is a subgraph T which is a tree and $V(T) = V(G)$.

Proposition

A graph is connected if and only if it has a spanning tree.

Counting Trees

A tree $T = (V, E)$ is a connected and acyclic graph.

Theorem (Cayley)

There are n^{n-2} labeled trees on n vertices.

Prüfer Code

- Input: A labelled tree T with n vertices
- Output: A sequence of length $n-2$ on n letters
  - $T_1 = T$, $b = []$
  - for i from 1 to $n-2$
    - Choose the leaf with smallest label in T_i
    - Append b_i, the label of the vertex adjacent to that leaf, to b.
    - Remove the leave to obtain T_{i+1}

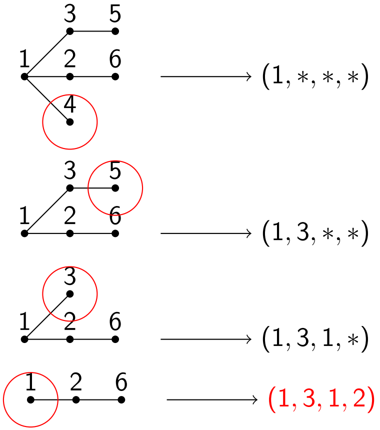
Counting Trees

A tree $T = (V, E)$ is a connected and acyclic graph.

Theorem (Cayley)

There are n^{n-2} labeled trees on n vertices.

Prüfer Code



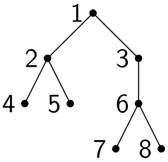
Counting Trees

A tree $T = (V, E)$ is a connected and acyclic graph.

Theorem (Cayley)

There are n^{n-2} labeled trees on n vertices.

Prüfer Code



$\leftarrow (2, 2, 1, 3, 6, 6)$

A Prüfer code uniquely determines a labeled tree.

Counting Trees

A tree $T = (V, E)$ is a connected and acyclic graph.

Theorem (Cayley)

There are n^{n-2} labeled trees on n vertices.

Prüfer Code

The reverse algorithm for recovering the tree from its Prüfer code:

- Input: A word $b = (b_1, \ldots, b_{n-2})$ of length $n-2$ on an alphabet of n letters.
- Output: A labeled tree with n vertices.
  - $F_1 = ([n], \emptyset)$, $w = b$
  - for i from 1 to $n-2$
    - Find the smallest label x_i not in w
    - Add an edge joining x_i with b_i to F_{i-1} to get F_i
    - Update w by removing b_i and appending x_i
  - Add to F_{n-2} an edge among the two labels not appearing in w