Introduction to Graphs

1 Graphs

What are graphs useful for?

- Representation of connectivity (network (social), travel, problem solution(s))
- Language definition
- Games (state/room connectivity)

1.1 Terminology

A *graph* consists of a set V, called *vertices* (or nodes), and a set number of pairs from V (*edges* or *arcs*).

There are two types of graphs:

- Undirected
- Directed (also called *digraphs*)

In a directed graph, there is an order to the pairs.

1.2 The Seven Bridges of Königsberg

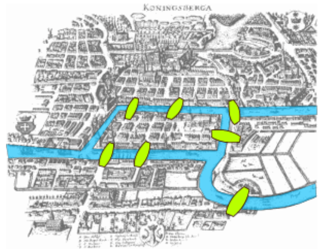
Original graph problem solved by Leonard Euler in 1735.

The city of Königsberg in Prussia (now Kaliningrad, Russia) was set on both sides of the Pregel River, and included two large islands which were connected to each other and the mainland by seven bridges.

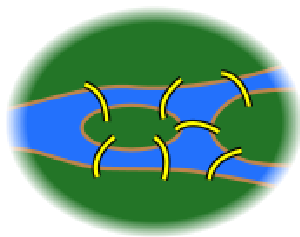
The problem

Determine if a person could walk through the city that would cross each bridge once and only once. The islands could not be reached by any route other than the bridges, and every bridge must have been crossed completely every time (one could not walk half way onto the bridge and then turn around and later cross the other half from the other side).

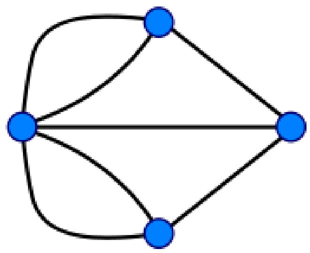
Euler proved that the problem has **no** solution.



$\downarrow$



$\downarrow$



`http://en.wikipedia.org/wiki/Seven_Bridges_of_Konigsberg`

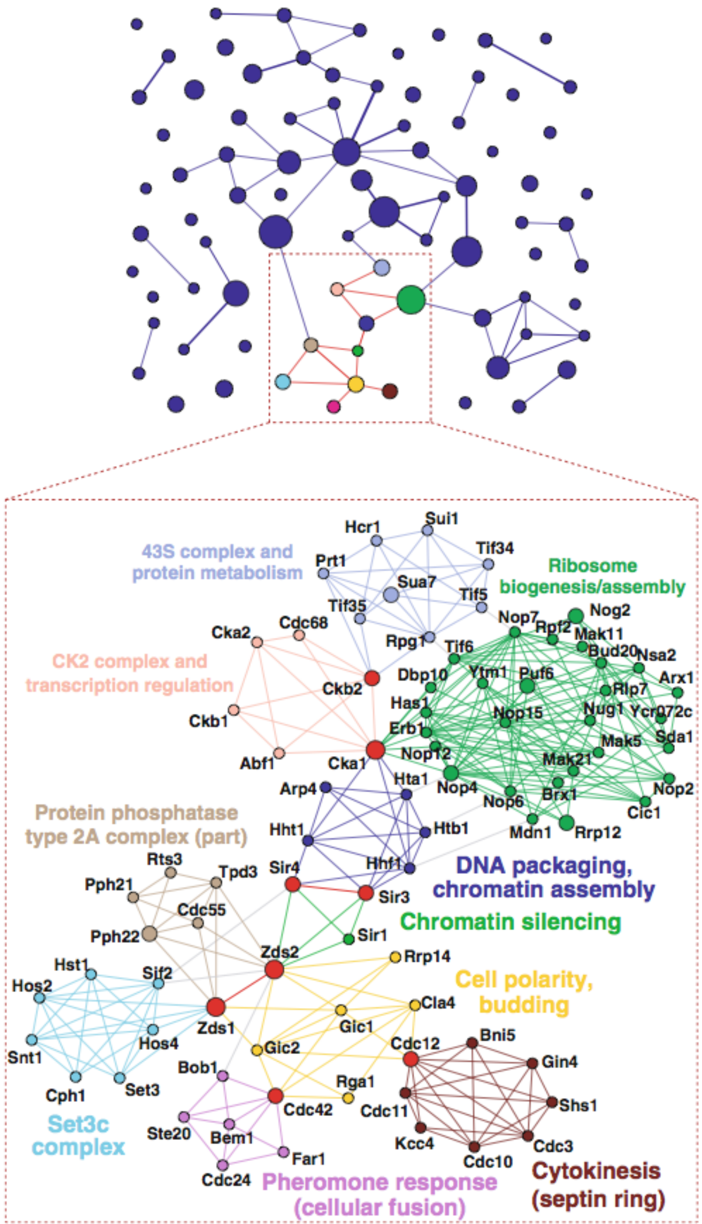


Fig. 1.2 A protein interaction network, showing a complex interplay between highly connected hubs and communities of subgraphs with increased densities of edges (from Palla et al. 2005)

1

[1] Claudius Gros, *Complex and Adaptive Dynamical Systems, Third Edition*, page 4.

1.3 More Terminology

Adjacent	Vertices are *adjacent* if in a pair.
Path	A sequence of edges that begin at one vertex and end at another vertex.
Simple Path	May not pass through same vertex twice.
Cycle	A path that begins and ends at same vertex and doesn't pass through the same vertex twice.
Rooted graph	Has a unique node, the *root*, such that there is a path from the root to all nodes within the graph.
Free tree	Connected, undirected graph with no cycles. Trees cannot have cycles.

Undirected graph examples

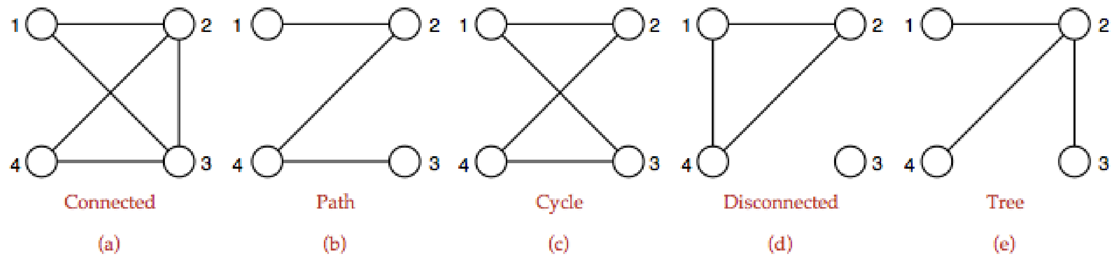
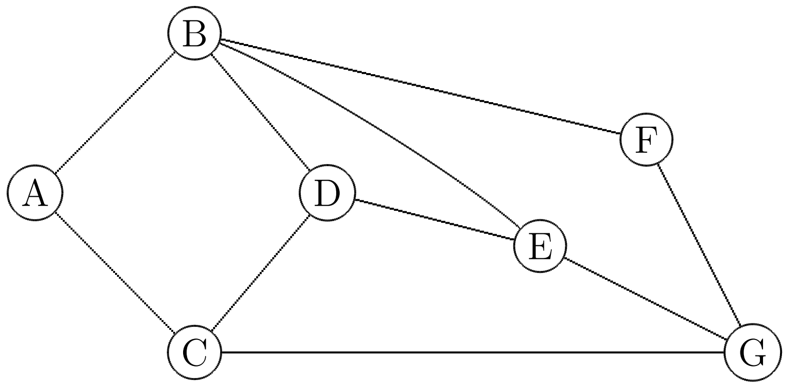


Figure 12.2. Various kinds of undirected graphs

1.4 State Space Representations of Problems

The nodes of a graph correspond to partial problem solution *states* and the edges correspond to steps in a problem-solving process. One or more *initial states*, corresponding to the given information in a problem instance, form the root of the graph. The graph also defines one or more *goal* conditions, which are solution the problem. *State space search* characterizes problem solving as the process of finding a *solution path* from the start state to a goal state.

Game trees and shortest path problems are two examples that are frequently solved using a state space representation.



Sample Graph

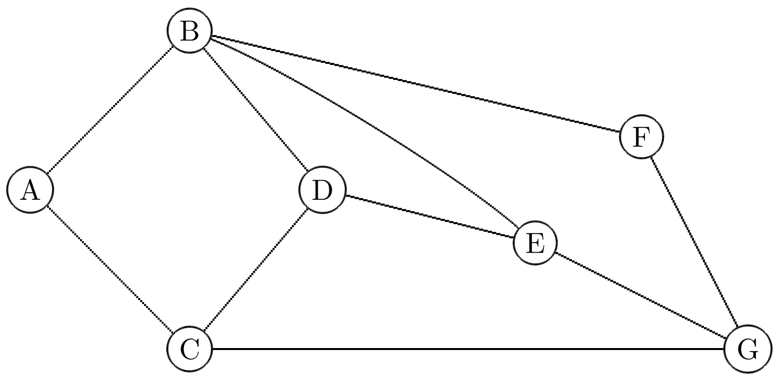
1.5 Adjacency Matrix

An **adjacency matrix** is a square grid of true/false (1/0) values that represent the edges of a graph. If the graph contains n vertices, then the grid contains n rows and n columns. For the two vertex numbers i and j, the component at row i and column j is true (1) if there is an edge from vertex i to vertex j; otherwise the component is false (0).

The adjacency matrix for the sample graph we saw earlier is given by:

	A	B	C	D	E	F	G
A	0	1	1	0	0	0	0
B	1	0	0	1	1	1	0
C	1	0	0	1	0	0	1
D	0	1	1	0	1	0	0
E	0	1	0	1	0	0	1
F	0	1	0	0	0	0	1
G	0	0	1	0	1	1	0

	A	B	C	D	E	F	G
A	0	1	1	0	0	0	0
B	1	0	0	1	1	1	0
C	1	0	0	1	0	0	1
D	0	1	1	0	1	0	0
E	0	1	0	1	0	0	1
F	0	1	0	0	0	0	1
G	0	0	1	0	1	1	0



1.6 Edge Lists

A directed graph with n vertices can be represented by n different linked lists. List number i provides the connection for vertex i. Specifically, for each entry j in list number i, there is an edge from i to j.

The **edge lists** for the sample graph we saw earlier is given by:

A $\rightarrow$ B $\rightarrow$ C

B $\rightarrow$ A $\rightarrow$ D $\rightarrow$ E $\rightarrow$ F

C $\rightarrow$ A $\rightarrow$ D $\rightarrow$ G

D $\rightarrow$ B $\rightarrow$ C $\rightarrow$ E

E $\rightarrow$ B $\rightarrow$ D $\rightarrow$ G

F $\rightarrow$ B $\rightarrow$ G

G $\rightarrow$ C $\rightarrow$ E $\rightarrow$ F

The lists could be linked together.

$$
\begin{array}{l}
A \rightarrow B \rightarrow C \\
\downarrow \\
B \rightarrow A \rightarrow D \rightarrow E \rightarrow F \\
\downarrow \\
C \rightarrow A \rightarrow D \rightarrow G \\
\downarrow \\
D \rightarrow B \rightarrow C \rightarrow E \\
\downarrow \\
E \rightarrow B \rightarrow D \rightarrow G \\
\downarrow \\
F \rightarrow B \rightarrow G \\
\downarrow \\
G \rightarrow C \rightarrow E \rightarrow F
\end{array}
$$

Remind you of anything we have seen?

A $\rightarrow$ B $\rightarrow$ C

B $\rightarrow$ A $\rightarrow$ D $\rightarrow$ E $\rightarrow$ F

C $\rightarrow$ A $\rightarrow$ D $\rightarrow$ G

D $\rightarrow$ B $\rightarrow$ C $\rightarrow$ E

E $\rightarrow$ B $\rightarrow$ D $\rightarrow$ G

F $\rightarrow$ B $\rightarrow$ G

G $\rightarrow$ C $\rightarrow$ E $\rightarrow$ F