CEE 5690 - Descriptive Modeling of Data

Introduction to Graph Theory
Week 2: Edwards 1.2

January 25, 2008

Definitions

Please be familiar with the definitions of the following terms, as relevant to undirected graphs: *Adjacency, Degree of a vertex, Complete graph, Sub-graph, Clique, Path, Cycle, Connected graph, Tree, Separation, Bipartite graph, Directed Acyclic Graph*

Problems

Typical graph theoretic problems:

- Identifying cliques in graphs
- Existence problems: Does there exist a perfect bipartite graph (matching problem). Does there exist an arrangement of five queens on a chessboard so that every non-occupied square is attached (Queens problem).
- Can we construct a particular graph? A problem of getting out of a maze
- Optimization problem: shortest path between two points, traveling salesman problem

0.1 Some Relevant Data Structures

The following data structures can be used to store G(V,E).

- Adjacency list: representation of all edges or arcs in a graph as a list
- Incidence list: for each vertex a list of edges incident to it
- Adjacency matrix: matrix representing the existence of edges between any two nodes i and j

Covering Problems: Set cover and Vertex Cover

Vertex Cover: Given the graph $G(V, E)$, The set $S \subseteq V$ is a set cover if $\forall\ e \in E$ there is at least 1 end in S. So given a graph G and a number k does G contain a vertex cover of size at most k.

Set Cover: Given U a set of n elements, a collection $S_1, \ldots, S_m$ of subsets of U, and a number k, does there exist a collection of at most k of these sets whose union is equal to U.

For example, if we have m available pieces of software, and a set U of n capabilities that we would like our system to have. The ith piece of software includes the set $S_i \subseteq U$ of capabilities. In the set cover problem, we seek to include a small number of these pieces of software on our system, with the property that our system will have all n capabilities.