Chapter 6 Section 3

MA1020 Quantitative Literacy

Sidney Butler

Michigan Technological University

December 1, 2006

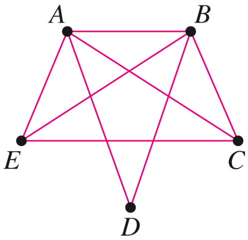
Hamiltonian Paths and Circuits

Definition

A Hamiltonian path is a path that visits each vertex in a graph exactly once. If the Hamiltonian path begins and ends at the same vertex, the path is called a Hamiltonian circuit.

Example

Are the paths below Hamiltonian Paths, Euler paths, both, or neither?



Complete Graphs

Definition

A complete graph is a graph in which *every* pair of vertices is connected by *exactly one* edge.

Example

1. Draw a complete graph with 6 vertices.
2. Count the number of edges in the graph.

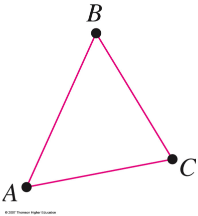
Theorem (Number of Edges in a Complete Graph)

A complete graph with n vertices has $\frac{n(n-1)}{2}$ edges.

Hamiltonian Paths in a Complete Graph

Example

Find all the Hamiltonian paths and all the Hamiltonian circuits in the graph.



Theorem (Number of Hamiltonian Paths in a Complete Graph)

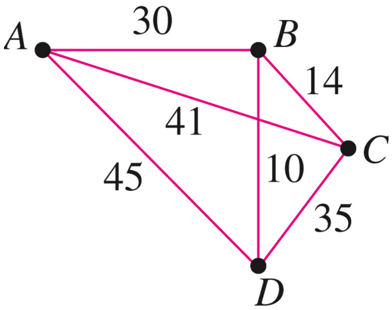
The number of Hamiltonian paths in a complete graph with n vertices is $n!$. The number of Hamiltonian circuits in a complete graph with n vertices is also $n!$.

Traveling-Salesperson Problem

Definition

The cost of a path in a weighted graph is the sum of the weights assigned to the edges in a path. When costs are assigned to each edge in a complete graph, the graph is called a complete weighted graph.

Example #28



1. List all possible Hamiltonian circuits.
2. Compute the cost of each circuit and find the circuit of least cost.
3. Identify all pairs of mirror-image Hamiltonian circuits. How do the costs of mirror-image Hamiltonian circuits compare?

Approximation Algorithms

Definition

An approximation algorithm is on an algorithm that, for most complete weighted graphs, will find a Hamiltonian circuit that is either the least-cost Hamiltonian circuit or is one that is not much more costly than the least-cost Hamiltonian circuit.

- Nearest-Neighbor
- Cheapest-Link

Nearest Neighbor Algorithm

1. Specify a starting vertex.
2. If unvisited vertices remain, go from the current vertex to the unused vertex that gives the least-cost connecting edge.
3. If no unvisited vertex remains, return to the starting vertex to finish forming the low-cost Hamiltonian circuit.

Cheapest-Link Algorithm

1. In the beginning, all edges are acceptable and no edges have been selected.
2. From the set of acceptable edges, select the edge of smallest weight. If there is a tie, select any of the edges with the smallest weight.
3. If the selected edges do not form a Hamiltonian circuit, then determine the set of acceptable edges. Unacceptable edges are those that either share one vertex with two selected edges or that would close a circuit that is not a Hamiltonian circuit. Repeat step 2.
4. If the selected edges form a Hamiltonian circuit, that circuit is your low-cost Hamiltonian circuit.