W1: Hamiltonian Circuits

Objectives: SWBAT

- Formulate an algorithm to find the best Eulerization.
- Explain the difference between an Euler circuit and an Hamiltonian circuit.
- Identify classes of graphs where it is easy to determine whether or not it contains a Hamiltonian circuit.
- Calculate the number of distinct Hamiltonian circuits in a complete graph with a given number of vertices.

How do we find a best eulerization for a given graph? Let’s first examine the case of rectangular networks:

Example:

Start in one corner and walk counterclockwise, whenever you arrive at an odd-valent vertex, link it to the next with an added edge. If the next vertex is now even, skip it and continue around looking for odd vertices, if the next vertex is now odd, link it to the next.

If the graph is not rectangular, we look for all the vertices that have odd valence and pair them up based on the shortest path between them.

Group Work: Worksheet W1.1
Try to come up with a best Eulerization of the given network. Can you prove that your Eulerization is a best Eulerization?

Eulerization Algorithm

1. Find all vertices with odd valence.
2. Pair them up with their nearest neighbor.
3. Find the shortest path between each pair.
4. Duplicate these edges.

Individual Activity: Worksheet W1.2

On Monday we learned about Euler circuits, what is different about what we did in this activity? We wanted to visit each vertex exactly once, not each edge. A circuit that visits every vertex of a graph exactly once is called a Hamiltonian circuit.
Remember: An Euler circuit crosses every edge exactly once, while a Hamiltonian circuit visits every vertex exactly once.

Individual/Group Work: Worksheet W1.3

Were you able to find a way to tell if a general graph has a Hamiltonian circuit or not? No. Remember for Euler circuits we had an easy way to tell, just by checking the valences of the vertices. We’ve changed the problem only slightly, but now there are no such criteria to tell. However, there are classes of graph that cannot have Hamiltonian circuits. What was the difference between graphs E and F? Why did E have a Hamiltonian circuit and F did not? The number of vertices on the left must equal the number of vertices on the right, because the circuit needs to alternately include vertices on the left and vertices on the right. What was the difference between graphs G and H? Color the vertices of the graph, needs to alternately include vertices of each color. Therefore the number of colored vertices much match.

A complete graph with $n$ vertices has edges connecting every pair of vertices. We denote these graphs by $K_n$. When $n \geq 3$, the graph will always have a Hamiltonian circuit.

Examples:

Let’s look back at the first example of vacation planning. We can draw this as a graph with numbers associated to each edge. These numbers are called weights and this graph is an example of a weighted graph. The question of finding a route with minimum distance is a question about finding a minimum-cost Hamiltonian circuit. How do we go about doing this?

**Finding a Minimum-Cost Hamiltonian Circuit**

1. Generate all possible Hamiltonian circuits.
2. Add up the distances along the edges of each circuit.
3. Choose a circuit with total distance being a minimum, that is, as small as possible.
We can use the *method of trees* to generate all the possible Hamiltonian circuits.

Notice that we have repeats. So there are actually only 3 Hamiltonian circuits. We can then add up the weights and compute the minimum-cost Hamiltonian circuit.

That’s not so bad. But what happens if we look at a graph with more vertices? If we start drawing the tree...it quickly becomes complicated and large. Is there a way to find out how large?

The method of trees is a visual representation of the *fundamental principle of counting*.

---

**Fundamental Principle of Counting**

In general, if there are \(a\) ways of choosing one thing, \(b\) ways of choosing a second after the first is chose,..., and \(z\) ways of choosing the last item after the earlier choices, then the total number of choice patterns is \(a \times b \times c \times \cdots \times z\).

---

**Exercises:**

1. In a restaurant, there are 4 kinds of soup, 12 entrees, 6 desserts, and 3 drinks. How many different four-course meals can a patron choose?
2. How many Hamiltonian circuits exist in a complete graph with 4 vertices?

Let’s look more closely at exercise 2. We have \(3 \cdot 2 \cdot 1\). A shorthand way to notate this uses *factorials*, \(3 \cdot 2 \cdot 1 = 3! = 6\). In general, \(n! = n \times (n-1) \times (n-2) \times \cdots \times 3 \times 2 \times 1\). Now, let’s think about the question we are trying to answer.

How many different Hamiltonian circuits exist in a complete graph with \(n\) vertices? \((n-1)!/2\)

If we only have a few cities to visit, enumerating all the paths is not so bad. Like in our example we would get 3 different circuits. Problems arise when we start looking at more cities. Let’s say we have 25 cities to visit. \(24!/2\) is approximately \(3 \times 10^{23}\). Even if we could generate 1 million tours each second, it would take 10 billion years to generate them all! This type of approach is called a *brute force method*. We’d like to find a more efficient method to solve this problem, since it is one of the most common problems in *operations research*, the branch of mathematics concerned with getting governments and businesses to operate more efficiently. It is usually called the *traveling salesman problem (TSP)* and is phrased as follows:
The traveling salesman problem (TSP) involves finding the trip of minimum cost that a salesman can make to visit the cities in a sales territory once and only once (represented by a complete graph with weights on the edges), starting and ending the trip in the same city.

There are lots of situations that in essence involve solving a TSP:

**Examples:**

1. A lobster fisherman has set out traps at various locations and wishes to pick up his catch.
2. The electric company needs to design a route for its meter readers.
3. A limousine service with a van located at the airport must pick up five customers and deliver them to the airport in time to catch their flights.

**Homework:**

Ch 2: 2, 26, 40