Graph Traversals

Major Themes

**Graph traversals:**
- Euler circuit/path
  - Every *edge* exactly once
- Hamilton circuit/path
  - Every *vertex* exactly once
  - Shortest circuit:
    - *Traveling Salesman Problem*
    - Shortest Path:
      - *Dijkstra’s algorithm*

The Bridges of Königsberg Puzzle

18th century map of the city of Königsberg with 7 bridges over the Pregel river

Find a walk through the city that would cross each bridge once and only once before returning to the starting point.

Euler’s Solution to The Bridges of Königsberg

Leonhard Euler (1707-1783) rephrased the question in terms of the *multigraph* (i.e., multiple edges allowed)

Is there a cycle in the multigraph that traverses all its edges exactly once?
Euler cycles (circuits) and paths

**Euler cycle**: a cycle traversing all the edges of the graph exactly once

**Euler path**: a path (not a cycle) traversing all the edges of the graph exactly once

Examples:

- Has an Euler cycle
- Has neither Euler cycle nor Euler path
- Has an Euler path, but no Euler cycle

Necessary and sufficient conditions for Euler cycles

**Theorem 12.3.1 (1)** A connected graph $G$ has an Euler cycle if and only if every vertex of $G$ has even degree.

(⇒) **A necessary condition**

$G$ has an Euler cycle only if every vertex has even degree

Assume $G$ has an Euler cycle. Observe that every time the cycle passes through a vertex, it contributes 2 to the vertex's degree, since the cycle enters via an edge incident with this vertex and leaves via another such edge.

(⇐) **A sufficient condition**

*If every vertex in $G$ has even degree, $G$ has an Euler cycle*

**Lemma** Assume every vertex in a multigraph has even degree. Start at an arbitrary non-isolated vertex $v_0$, choose an arbitrary edge $(v_0, v_1)$, then choose an arbitrary unused edge from $v_1$ and so on. Then after a finite number of steps the process will arrive at the starting vertex $v_0$, yielding a cycle with distinct edges.

**Proof** In the above procedure, once you entered a vertex $v$, there will always be another unused edge to exit $v$ because $v$ has an even degree and only an even number of the edges incident with it had been used before you entered it. The only edge from which you may not be able to exit after entering it is $v_0$ (because an odd number of edges incident with $v_0$ have been used as you didn’t enter it at the beginning), but if you have reached $v_0$, then you have already constructed a required cycle.

A procedure for constructing an Euler cycle

**Algorithm Euler**

//Input: Connected graph $G$ with all vertices having even degrees
//Output: Euler cycle

Construct a cycle in $G$ using the procedure from Lemma
Remove all the edges of cycle from $G$ to get subgraph $H$
while $H$ has edges
  find a non-isolated vertex $v$ that is both in cycle and in $H$
  //the existence of such a vertex is guaranteed by $G$’s connectivity
  construct subcycle in $H$ using Lemma’s procedure
  splice subcycle into cycle at $v$
  remove all the edges of subcycle from $H$
return cycle
Necessary and sufficient conditions for Euler paths

**Theorem 12.3.1 (2)** A connected graph G has an Euler path but not an Euler cycle if and only if it has exactly two vertices of odd degree.

Icosian Game

Icosian Game This puzzle has been invented by the renowned Irish mathematician Sir William Hamilton (1805-1865) and presented to the world under the name of the Icosian Game. The game’s board was a wooden board with holes representing major world cities with grooves representing connections between them. The object of the game was to find a circular route that would pass through all the cities exactly once before returning to the starting point.
Hamilton Circuit and/or Traversal?

C3, K4, G, K6, G

Hamilton Circuits: Sufficient condition

- Let G be a graph with n vertices where n ≥ 3. If the degree of every vertex is more than \( n/2 \), then G has a Hamilton circuit.

- Note: NOT a necessary condition
- No property is known to efficiently verify existence of a Hamilton cycle/path for general graphs. Moreover, the problem is known to be as difficult as the TSP

Traveling Salesman Problem

Image: https://www.wired.com/images_blogs/wiredscience/2013/01/tsp_map.jpg

Shortest Paths

Dijkstra’s Algorithm:
- build a tree of minimal length paths from root to leaves
- similar to Prim’s MST algorithm
Puzzle 1: Seating King Arthur’s knights (and their grudges) on the round table

King Arthur needs to seat 100 knights (including himself) around his round table so that no knight is seated next to his enemy. The king has a list of enemies for each knight, which contains less than 50 persons for each of them. Can the king find an acceptable seating?

Puzzle 2: Weaving an epic tale of King Arthur’s knights (and their grudges)

There is a story – or stories! – between every pair of knights that are enemies. A poet is captivated by these tales and wants to weave them into one great epic poem. The poem’s subject must naturally flow from knight to knight, possibly returning to some of them repeatedly (if they are involved in more than one tale), telling each and every one of these stories, but never jumping arbitrarily from one story to another that does not involve either of the knights in the first story; so the story needs to be connected, somehow. For example, if at a given point the poem is describing a story of how knight 37 and knight 12 became enemies, the next story must be a story involving one of these knights (or both of them, in case there is more than one reason that they are enemies) – it cannot, say, jump to a story about knight 42 and knight 61.

Can you state conditions on the number of enemies for each knight that would guarantee that such a story can be told?