Today’s Outline

Shortest path algorithms
1. Unweighted graphs: BFS
2. Weighted graphs without negative cost edges: Dijkstra’s Algorithm
3. Negative cost edges but no negative cost cycles

Aside: Graph Connectivity

Undirected graphs are connected if there is a path between any two vertices.

Directed graphs are strongly connected if there is a path from any one vertex to any other.

Directed graphs are weakly connected if there is a path between any two vertices, ignoring direction.

A complete graph has an edge between every pair of vertices.

The Shortest Path Problem

Given a graph G, edge costs $c_{ij}$, and vertices $s$ and $t$ in G, find the shortest path from $s$ to $t$.

For a path $p = v_0, v_1, v_2, \ldots, v_k$
- unweighted length of path $p = k$ (a.k.a. length)
- weighted length of path $p = \sum_{i=0}^{k-1} c_{v_i v_{i+1}}$ (a.k.a. cost)

Path length equals path cost when?

Single Source Shortest Paths (SSSP)

Given a graph G, edge costs $c_{ij}$, and vertex $s$, find the shortest paths from $s$ to all vertices in G.

- Is this harder or easier than the previous problem?

All Pairs Shortest Paths (APSP)

Given a graph G and edge costs $c_{ij}$, find the shortest paths between all pairs of vertices in G.

- Is this harder or easier than SSSP?

- Could we use SSSP as a subroutine to solve this?
Variations of SSSP

- Weighted vs. unweighted
- Directed vs undirected
- Cyclic vs. acyclic
- Positive weights only vs. negative weights allowed
- Shortest path vs. longest path
- ...

Applications

- Network routing
- Driving directions
- Cheap flight tickets
- Critical paths in construction management (see textbook)
- ...

SSSP: Unweighted Version

Ideas?

Weighted SSSP:
The Quest For Food

Can we calculate shortest distance to all nodes from Allen Center?

Dijkstra, Edsger Wybe

Legendary figure in computer science; was a professor at University of Texas.

Supported teaching introductory computer courses without computers (pencil and paper programming).

Supposedly wouldn’t (until very late in life) read his e-mail; so, his staff had to print out messages and put them in his box.

Dijkstra’s Algorithm: Idea

Adapt BFS to handle weighted graphs

Two kinds of vertices:
- Finished or known vertices
  - Shortest distance has been computed
- Unknown vertices
  - Have tentative distance
Dijkstra’s Algorithm: Idea

At each step:
1) Pick closest unknown vertex
2) Add it to finished vertices
3) Update distances

Dijkstra’s Algorithm: Pseudocode

Initialize the cost of each node to \( \infty \)
Initialize the cost of the source to 0
While there are unknown nodes left in the graph
    Select an unknown node \( b \) with the lowest cost
    Mark \( b \) as known
    For each node \( a \) adjacent to \( b \)
        \( a \)’s cost = min(\( a \)’s old cost, \( b \)’s cost + cost of \( (b, a) \))

Dijkstra’s Algorithm in Action

Time to play at home…

1. Use Dijkstra’s algorithm to find the shortest path from \( H \) to every node in the graph below

2. Under what conditions will Dijkstra’s algorithm fail?

3. What data structures should you use to best implement this algorithm? What running time does that yield?

Dijkstra’s Alg: Implementation

Initialize the cost of each node to \( \infty \)
Initialize the cost of the source to 0
While there are unknown nodes left in the graph
    Select the unknown node \( b \) with the lowest cost
    Mark \( b \) as known
    For each node \( a \) adjacent to \( b \)
        \( a \)’s cost = min(\( a \)’s old cost, \( b \)’s cost + cost of \( (b, a) \))

What data structures should we use?

Running time?

Dijkstra’s Algorithm: Summary

• Classic algorithm for solving SSSP in weighted graphs without negative weights
• A greedy algorithm (irrevocably makes decisions without considering future consequences)

• Intuition for correctness:
  – shortest path from source vertex to itself is 0
  – cost of going to adjacent nodes is at most edge weights
  – cheapest of these must be shortest path to that node
  – update paths for new node and continue picking cheapest path
Correctness: The Cloud Proof

Next shortest path from inside the known cloud
Better path to V? No!

- If path to V is shortest, path to W must be at least as long
- So the path through W to V cannot be any shorter!

Dijkstra’s vs BFS

At each step:
1) Pick closest unknown vertex
2) Add it to finished vertices
3) Update distances

Dijkstra’s Algorithm

At each step:
1) Pick vertex from queue
2) Add it to visited vertices
3) Update queue with neighbors

Breadth-first Search

Some Similarities:

The Trouble with Negative Weight Cycles

What’s the shortest path from A to E?

Problem?

Back to an Application:
Moving Around Washington

What’s the fastest way from Seattle to Spokane?

Answer:

To Do

- Read Chapter 9