Variants

- Known algorithms better than running single-source for each vertex.
- All-pairs shortest paths
  - But single-source algorithms have same worse-case asymptotic running time as
  - Seems like it should be easier than single-source
    - From u to v
  - Single-source shortest path
    - Just transpose the graph
    - To a particular point from all other points
- Single-destination shortest paths
  - From a particular point to all other points
- Single-pair shortest path
  - Seems like it should be easier than single-source
    + But all known algorithms have same worse-case asymptotic running time as
- Known algorithms better than running single-source for each vertex

Single-Source Shortest Paths (Chapter 24)

- Find shortest path
  - Useful for navigation
  + Each intersection, onramp, exit is a vertex

Weighted directed graph $G = (V, E)$ with $w : E \rightarrow \mathbb{R}$

- Weight of a path $p = \langle v_0, v_1, ..., v_k \rangle$
  $$\text{weight}(p) = \sum_{i=1}^{k} w(v_{i-1}, v_i)$$

- Shortest-path weight $\delta(u, v)$ from $u$ to $v$
  $$\delta(u, v) = \begin{cases} 
\min \{ \text{weight}(p) : u \rightarrow_p v \} & \text{if there is a path from } u \text{ to } v \\
\infty & \text{otherwise}
\end{cases}$$

- Similar to shortest path for breadth-first search (where weights are all 1)
  - Shortest-path weight from $a$ to $n$
    $$\delta(a, n) = \infty$$

- Weight of a graph $G = (V, E)$ with $n$ vertices and $m$ edges where
  + Each intersection, onramp, exit is a vertex
  + Edges between distinct edges, which have a distance
- Find shortest path
  - Useful for navigation
Negative Weight edges

• Each time through a negative weight cycle gives a lower weight.
  • Paths that can include a negative weight cycle undefined.
    • Weight is $-\infty$.
  • Some algorithms (Dijkstra's) assume weights are non-negative.

Example: source is $s$.

Optimal Substructure

Lemma 24.1

Given a weighted directed graph, if there is a shortest path from $x$ to $n$, and it goes through $x$ and $n$, the subpath from $x$ to $n$ is a shortest path from $x$ to $n$. 

Optimal Substructure
Representing Shortest Paths

- Each vertex has a predecessor \( v.\pi \): a vertex or nil

Similar to breadth-first search

Cycles

- Can shortest path contain a cycle?
- Can shortest path contain a cycle with 0 weight?
- Can shortest path contain a cycle with positive weight?
- Can shortest path contain a cycle with negative weight?

Can restrict ourselves to paths of at most \( |V| - 1 \) edges