Dijkstra's algorithm for single source shortest paths

\[ BF : O(|V||E|) \] which can be \( \Omega(n^3) \) runtime.

Please use 'O', 'Ω', 'o', 'ω' appropriately.

\[ o : \text{small } O \]

\[ \omega : \text{small } \Omega \]

\[
\frac{n}{\sqrt{n \log n}} \rightarrow 0 \quad \text{as } n \rightarrow \infty
\]

\[ \leq, \leq, > \]

Basic operation: Relax \((u, v)\)

\[
\begin{cases}
\text{if } d(v) < d(u) + w(u, v) \\
\text{then } d(v) \leftarrow d(u) + w(u, v) \\
\text{Link } u \text{ as predecessor of } v
\end{cases}
\]

\[
\text{dist} \{ S : \text{set of vertices } v, \; d(v) = D(v) \}
\]

\[
T : \quad \begin{cases}
\text{dist} \{ S : \text{set of vertices } v, \; d(v) > D(v) \}
\end{cases}
\]

Initially \( d(s) = D(s) = 0 \) other \( d(v) = \infty \)

Try to include vertices \( w \notin T - S \) into \( S \)

Pick the smallest value of \( d(v) \) for \( T \)
Once w moves to S, relax all edges \((w, x) \quad (x, f) \in E\)

until \(T = \emptyset\)

Running Time: \((|V| - 1)\) iterations

1. Pick the smallest label. Heap on \(T\) \(O(\log n)\) to pick the smallest

2. Relax all edges \((w, x)\)

\[
\begin{align*}
W & \quad x_1 \quad x_2 \quad x_3 \\
\text{neighbors of } W & \quad \text{degree of } W \\
S(x_1) & \quad S(x_2) \quad S(x_3)
\end{align*}
\]

must be updated

Decrease key: \(O(\log n)\)

Cost: \(O(\text{deg}(w) \cdot \log n)\)

Total cost for a single iteration \(O(\log n + \text{deg}(w) \cdot \log n)\)

Over \(|V|\) iterations \(O(|V| \log n + |V|^2 \log n)\)
\begin{align*}
\sum_{i=1}^{n} \left( \log n + \deg(w) \cdot \log n \right) \\
\leq n \log n + \log n \sum_{w \in V} \frac{\deg(w)}{n} \\
\leq n \log n - |E| \log n \\
= (n + |E|) \log n
\end{align*}

Possibility of improvement: Perhaps decrease key can improve

Why in Dijkstra's algorithm always succeed?

Claim: When we move \( w \in T \) to \( S \),
\( s(w) = D(w) \)

Observation: The labels of vertices in successive iterations included in \( S \) are in non-decreasing order.

Proof by contradiction: Suppose \( D(w) \) is not the shortest path distance of \( w \). \( s(w) \) was the next smallest label in \( T \).
This is the shortest path from $s$ to $w$:

$$s \rightarrow x_1 \rightarrow x_2 \rightarrow \cdots \rightarrow x_k \rightarrow w = x_k$$

Clearly, $x_{k-1} \not\in S$ and $y$ is an edge.

Case: no zero-length path from $x_{k-1}$ to $w$.

$$D(y) = d(y) > d(x_k)$$

Clearly, $w$ doesn't have the smallest label: contradiction.

What about shortest paths rather than shortest path distances $D(v)$?

$$s \rightarrow \cdots \rightarrow \varnothing \rightarrow v$$

All pairs shortest paths

Invoke $SSSP(v)$ for all $v \in V$. 

pre+/v
A single path can have length \( |V|-1 \)

What is the structure of the shortest path (SSSP)

\[ V_{i-1} \rightarrow V_i \]