Shortest Paths Problem

- Given a weighted directed graph, let $\ell(e) > 0$ denote the length of edge $e$ and for a path $P$ consisting of edges $e_1, e_2, \ldots, e_k$ we denote the length of this path as $\ell(P) = \ell(e_1) + \ell(e_2) + \ldots + \ell(e_k)$.
- Fix a node $s$ and let $d(v)$ be the length of shortest $s \rightarrow v$ path.
- **Problem:** Can we efficiently find $d(v)$ for all nodes $v \in V$?

Dijkstra’s Algorithm

- **Initialize:** Let $S = \{s\}$ be set of “explored nodes” and $d(s) = 0$.
- **While** $S \neq V$:
  - Find node $v \notin S$ that minimizes $\pi(v) = \min_{(u,v) \in E : u \in S} (d(u) + \ell(u,v))$.
  - Add $v$ to $S$ and set $d(v) = \pi(v)$.

**Running Time Analysis:** The while loop occurs $n - 1$ times and in each iteration finding $v$ can be done in $O(m)$ time. So total run time of a naive implementation is $O(mn)$ but a more clever implementation exists that uses $O(m \log n)$ time.

Proof of Correctness

- We prove by induction on $|S|$ that for all $u \in S$, $d(u)$ is the length of the shortest $s \rightarrow u$ path.
- **Base case:** When $|S| = 1$, it’s obvious since $s$ is only node in $S$ and $d(s) = 0$.
- **Inductive hypothesis:** Assume true for $|S| = k \geq 1$.
- Let $v$ be next node added to $S$ and let $(u, v)$ be preceding edge.
- Shortest $s \rightarrow u$ path plus $(u, v)$ is $s \rightarrow v$ path of length $\pi(v)$.
- Consider any $s \rightarrow v$ path $P$. We will show $\ell(P) \geq \pi(v)$.
- Let $(x, y)$ be the first edge in $P$ that leaves $S$, and let $P'$ be the subpath from $s$ to $x$.
- Then,
  \[ \ell(P) \geq \ell(P') + \ell(x, y) \geq d(x) + \ell(x, y) \geq \pi(y) \geq \pi(v) \]