CS 312: Algorithms

Shortest Paths

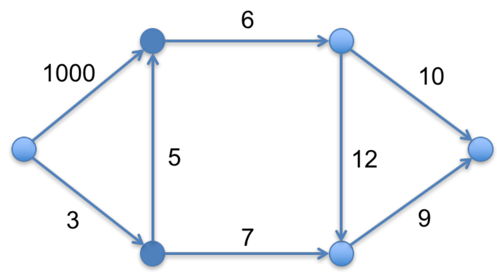
Dan Sheldon

Mount Holyoke College

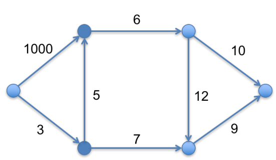
Last Compiled: October 19, 2018

Shortest Paths Problem

Problem: find shortest paths in a directed graph with edge *lengths* (the Google maps problem)



Let's Formalize the Problem



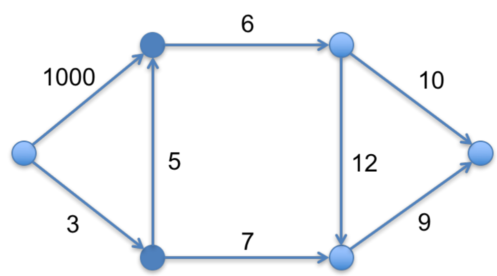
- Directed graph $G = (V, E)$ with edge *lengths* $\ell(e) > 0$
- Define *length* of path P consisting of edges $e_1, e_2, \ldots, e_k$ as

$$\ell(P) = \ell(e_1) + \ell(e_2) + \ldots + \ell(e_k)$$

- Starting node s
- Let $d(v)$ be the length of shortest $s \to v$ path.
- **Problem:** Can we efficiently find $d(v)$ for all nodes $v \in V$?

Shortest Paths Problem

Suppose all edges have integer length. Can we use BFS to solve this problem?



Small example + intuition on board. Derive algorithm on second example.

Dijkstra's Algorithm

Let A be a priority queue
Add s to A with priority $d'(s) = 0$
For all $v \neq s$, add v to A with priority $d'(v) = \infty$
while A not empty **do** ▷ Nodes left to explore
  Extract node $v \in A$ with smallest $d'(v)$ value
  Set $d(v) = d'(v)$ ▷ Wave arrives at v
  for all edges (v, w) where $w \in A$ **do**
    if $d(v) + \ell(v, w) < d'(w)$ **then**
      $d'(w) = d(v) + \ell(v, w)$ ▷ Earlier arrival time
    end if
  end for
end while

Running Time?

Use heap-based priority queue for A

Let A be a priority queue
Add s to A with priority $d'(s) = 0$
For all $v \neq s$, add v to A with priority $d'(v) = \infty$
while A not empty **do**
  Extract node $v \in A$ with smallest $d'(v)$ value ▷ Extract-min
  Set $d(v) = d'(v)$
  for all edges (v, w) where $w \in A$ **do**
    if $d(v) + \ell(v, w) < d'(w)$ **then**
      $d'(w) = d(v) + \ell(v, w)$ ▷ Update-key
    end if
  end for
end while

- n extract-min operations. $O(n \log n)$
- m update-key operations. $O(m \log n)$
- **Total**: $O((m + n) \log n)$

Finding the Actual Path

Keep track of prev(v) = node that last updated arrival time $d'(v)$ = predecessor in shortest path

Let A be a priority queue
Add s to A with priority $d'(s) = 0$
Set prev(s) = null
For all $v \neq s$, add v to A with priority $d'(v) = \infty$
while A not empty **do**
    Extract node $v \in A$ with smallest $d'(v)$ value
    Set $d(v) = d'(v)$
    for all edges (v, w) where $w \in A$ **do**
        if $d(v) + \ell(v, w) < d'(w)$ **then**
            $d'(w) = d(v) + \ell(v, w)$
            prev(w) = v
        end if
    end for
end while

Proof of Correctness

- Let $S = V \setminus A$ be the set of *explored* nodes at any point in the algorithm (i.e., we removed these nodes from queue and assigned $d(v)$)
- **Claim (invariant)**: for $v \in S$, the value $d(v)$ is the length of the shortest $s \to v$-path
- Proof on board, by induction on |S|. See pp.. 139–140.