Shortest Reliable Path Dynamic Programming

Given: Graph \( G = (V, E) \), weights \( \lambda(u,v) \)

Nodes \( s, t \in V \), integer \( k \).

Find: Shortest path from \( s \) to \( t \) that uses no more than \( k \) edges.

Why? Edge transitions or "hops" may have inherent costs not measured in edge weights, e.g., airline customer may not want more than 2 flights to get to destination.

Shortest Reliable Path Dynamic Programming

Algorithm:

for \( u \in V \):
   for \( i = 0, k \):
      \( \text{dist}(u, i) = \infty \)
   \( \text{dist}(s, 0) = 0 \)

for \( i = 1, k \):
   for \( (u, v) \in E \):
      if \( \text{dist}(v, i) > \text{dist}(u, i-1) + \lambda(u, v) \):
         \( \text{dist}(v, i) = \text{dist}(u, i-1) + \lambda(u, v) \)

return \( \text{dist}(t, k) \)

Runtime: \( O(k|E|) \)

Shortest Reliable Path Dynamic Programming

Subproblems:

\( \text{dist}(v, i) = \text{shortest distance from } S \text{ to } v \text{ using } \leq i \text{ edges} \)

\( \text{dist}(v, i) = \min \{ \text{dist}(u, i-1) + \lambda(u, v) \} \)

\( (u, v) \in E \)