Dijkstra’s Algorithm (1959)

Given: a directed graph with non-negative edge costs,
Find: the shortest path from s to every other vertex

- pick the unvisited vertex with the lowest distance to s
- calculate the distance through it to each unvisited neighbour, and update the neighbour's distance if smaller
- mark vertex as visited once all neighbours explored
Example: Dijkstra’s Shortest Path Algorithm

Find shortest path from s to t.
Dijkstra's Shortest Path Algorithm

\[ S = \{ \} \]
\[ PQ = \{ s, 2, 3, 4, 5, 6, 7, t \} \]
Dijkstra's Shortest Path Algorithm

S = { }  
PQ = { s, 2, 3, 4, 5, 6, 7, t }
Dijkstra's Shortest Path Algorithm

\[ S = \{ s \} \]
\[ PQ = \{ 2, 3, 4, 5, 6, 7, t \} \]

distance label 15

distance through vertex to each unvisited neighbour, and update with min if smaller
Dijkstra's Shortest Path Algorithm

\[ S = \{ s \} \]
\[ PQ = \{ 2, 3, 4, 5, 6, 7, t \} \]

pick the unvisited vertex with the lowest distance to \( s \)

Select

Distance label

\( 0 \)

\( s \)
Dijkstra's Shortest Path Algorithm

$S = \{ s, 2 \}$
$PQ = \{ 3, 4, 5, 6, 7, t \}$
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2 \} \]
\[ PQ = \{ 3, 4, 5, 6, 7, t \} \]

compute distance through vertex to each unvisited neighbour, and update with min if smaller
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2 \} \]
\[ PQ = \{ 3, 4, 5, 6, 7, t \} \]

pick the unvisited vertex with the lowest distance to \( s \)
Dijkstra's Shortest Path Algorithm

$S = \{ s, 2, 6 \}$
$PQ = \{ 3, 4, 5, 7, t \}$
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 6 \} \]
\[ PQ = \{ 3, 4, 5, 7, t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 6, 7 \} \]
\[ PQ = \{ 3, 4, 5, t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 6, 7 \} \]
\[ PQ = \{ 3, 4, 5, t \} \]
Dijkstra's Shortest Path Algorithm

S = \{ s, 2, 3, 6, 7 \}
PQ = \{ 4, 5, t \}
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 3, 6, 7 \} \]
\[ PQ = \{ 4, 5, t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 3, 5, 6, 7 \} \]
\[ PQ = \{ 4, t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 3, 5, 6, 7 \} \]
\[ PQ = \{ 4, t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 3, 4, 5, 6, 7 \} \]
\[ PQ = \{ t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 3, 4, 5, 6, 7 \} \]
\[ PQ = \{ t \} \]
Dijkstra's Shortest Path Algorithm

\[ S = \{ s, 2, 3, 4, 5, 6, 7, t \} \]
\[ PQ = \{ \} \]
Dijkstra's Shortest Path Algorithm - solution

S = \{ s, 2, 3, 4, 5, 6, 7, t \}
PQ = \{ \}
More …

• Applications:
  • Snakes in computer vision: dynamic programming on a closed or open chain
  • Detecting human limb layout in images: dynamic programming on trees

• David Mackay’s book for message passing

• More on web page:
  • http://www.robots.ox.ac.uk/~az/lectures/opt