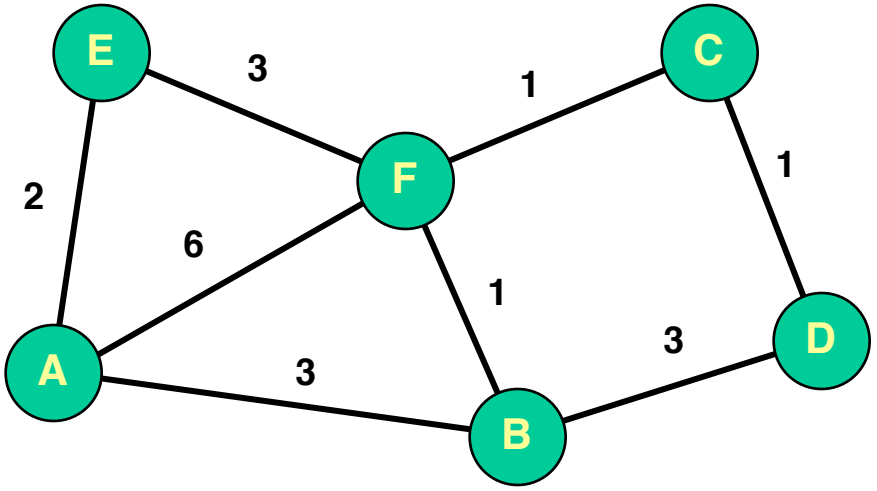
Review: Shortest Path Routing: Dijkstra's Algorithm

ECE 50863 – Computer Network Systems

Graph Model



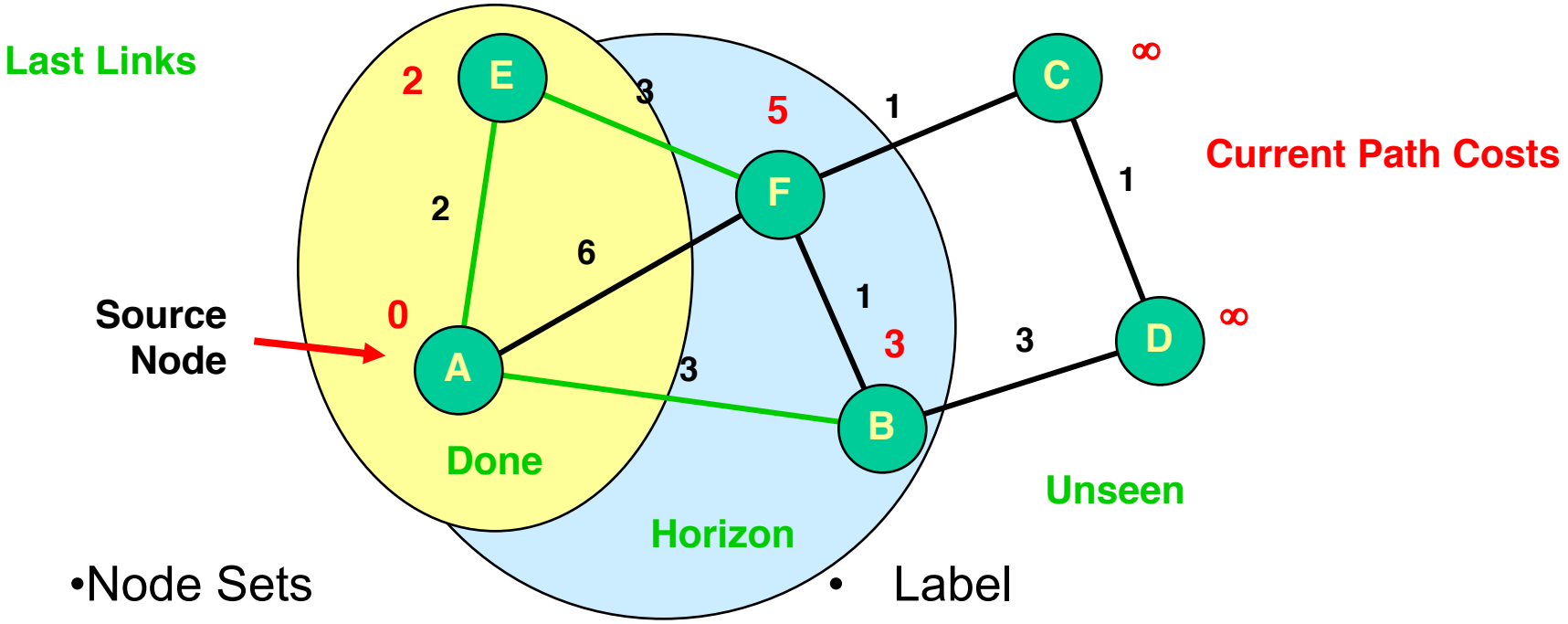
- Represent each router as node
- Direct link between routers represented by edge
  - Symmetric links $\Rightarrow$ undirected graph
- Edge “cost” c(x,y) denotes measure of difficulty of using link

- Task
  - Determine least cost path from every node to every other node
    - Path cost d(x,y) = sum of link costs

Dijkstra's Algorithm

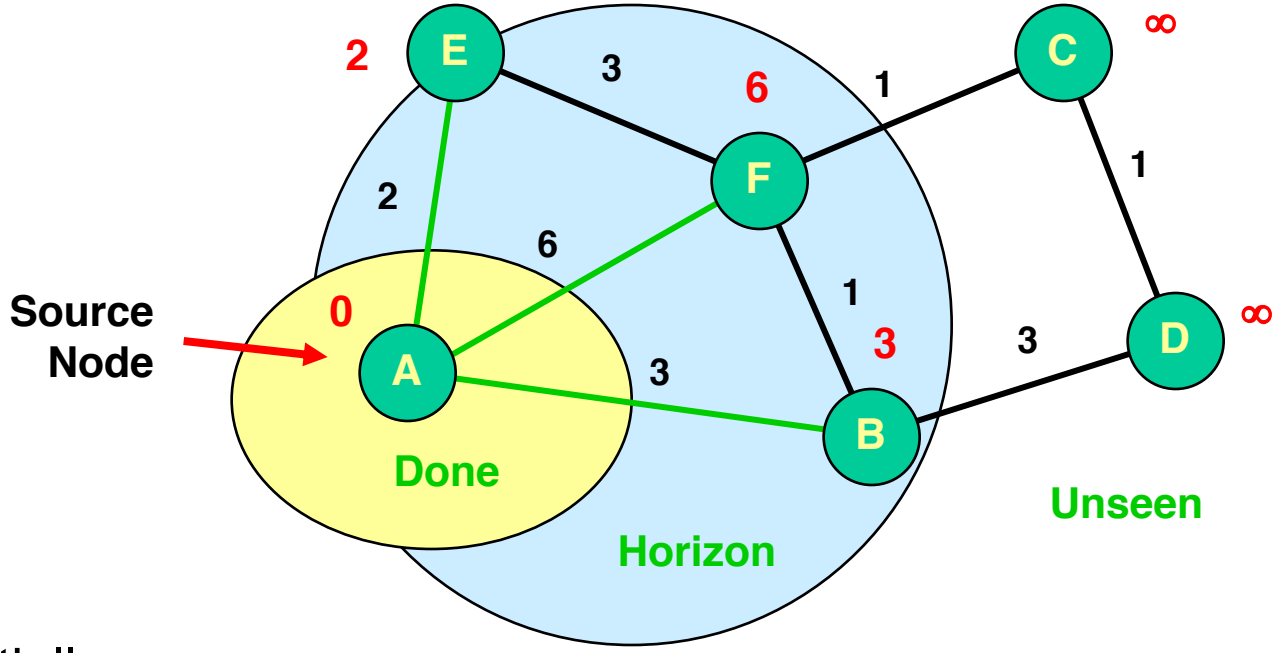
- Given
  - Graph with source node s and edge costs c(u,v)
  - Determine least cost path from s to every node v
- Shortest Path First Algorithm
  - Traverse graph in order of least cost from source

Dijkstra's Algorithm: Concept



- Node Sets
  - Done
    - Already have least cost path to it
  - Horizon:
    - Reachable in 1 hop from node in Done
  - Unseen:
    - Cannot reach directly from node in Done
- Label
  - d(v) = path cost
    - From s to v
    - Optimal for nodes in Done
- Path
  - Keep track of last link in path

Dijkstra's Algorithm



Initially:

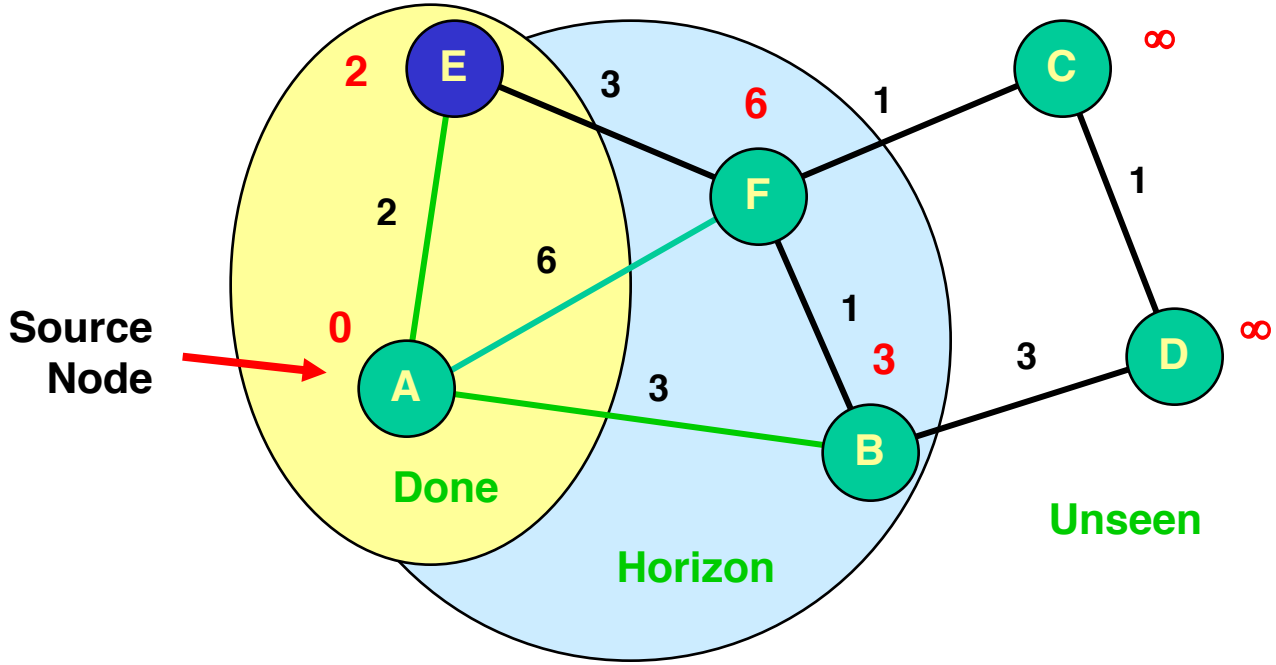
– Source A is in Done.

– Direct neighbors of A are in Horizon.

Next:

– Select node v in horizon with minimum d(v). Move it to Done

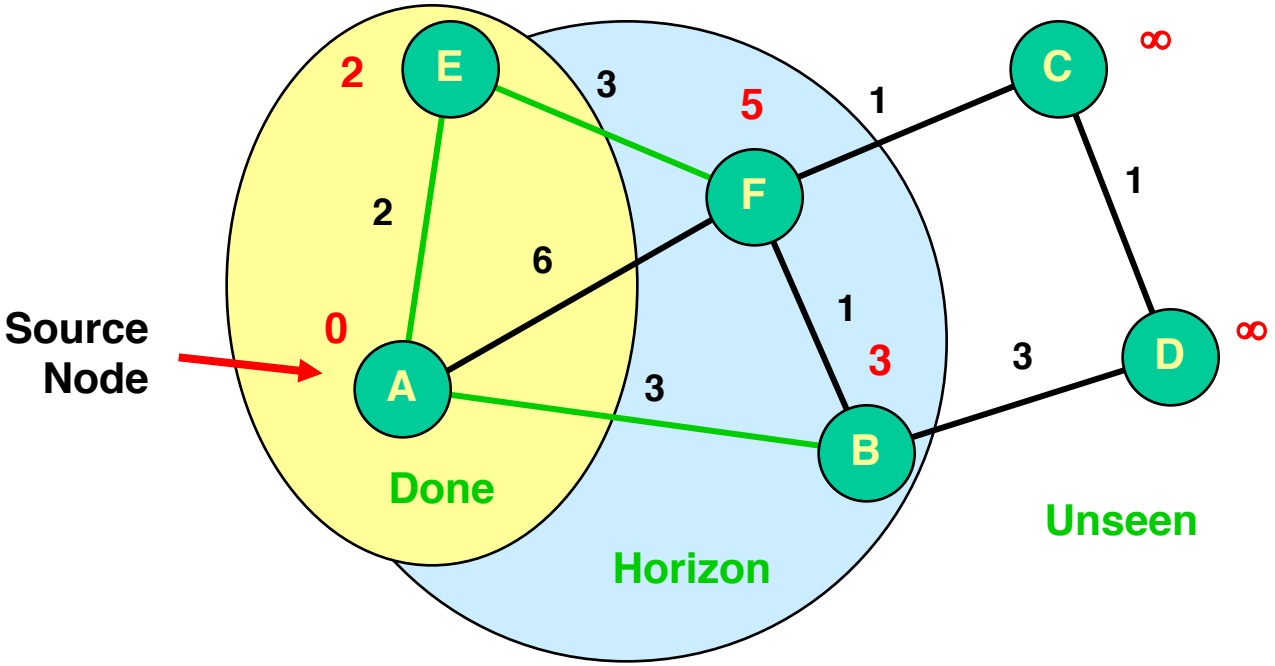
Dijkstra's Algorithm



Next:

- Update which nodes are in Horizon
- Update costs to nodes in Horizon
- Update last link

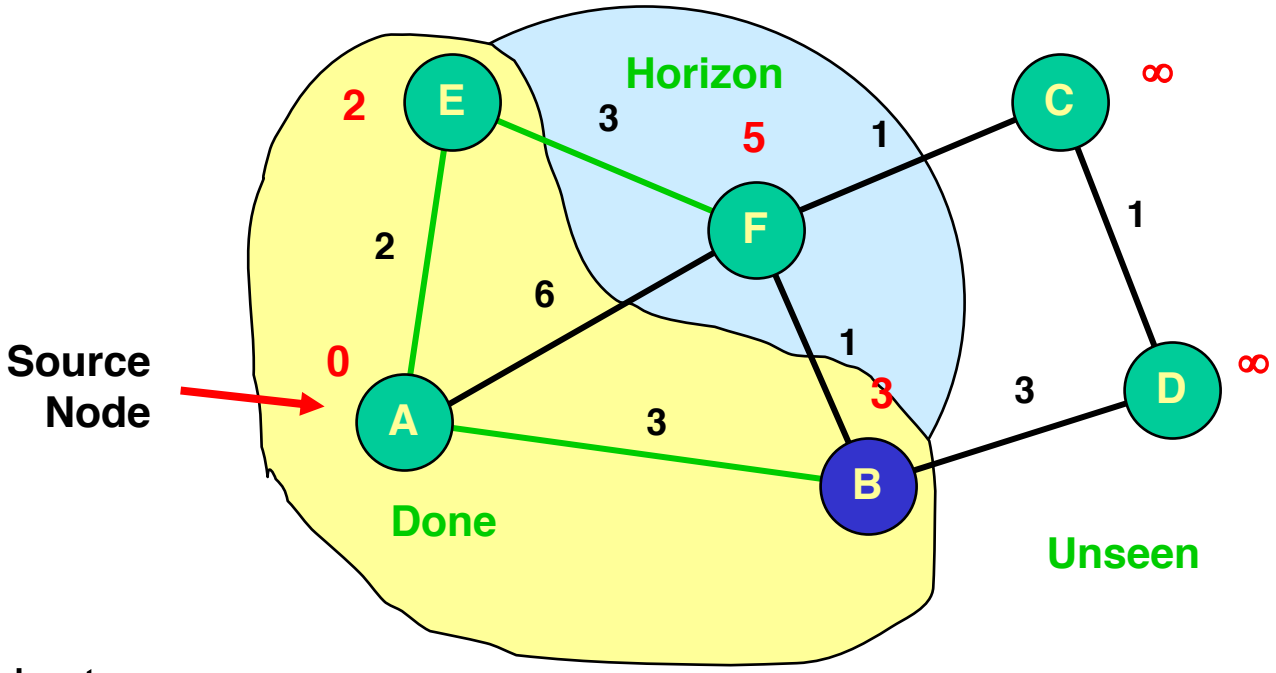
Dijkstra's Algorithm: Selection



Repeat

– Select node v in horizon with minimum d(v). Add to Done

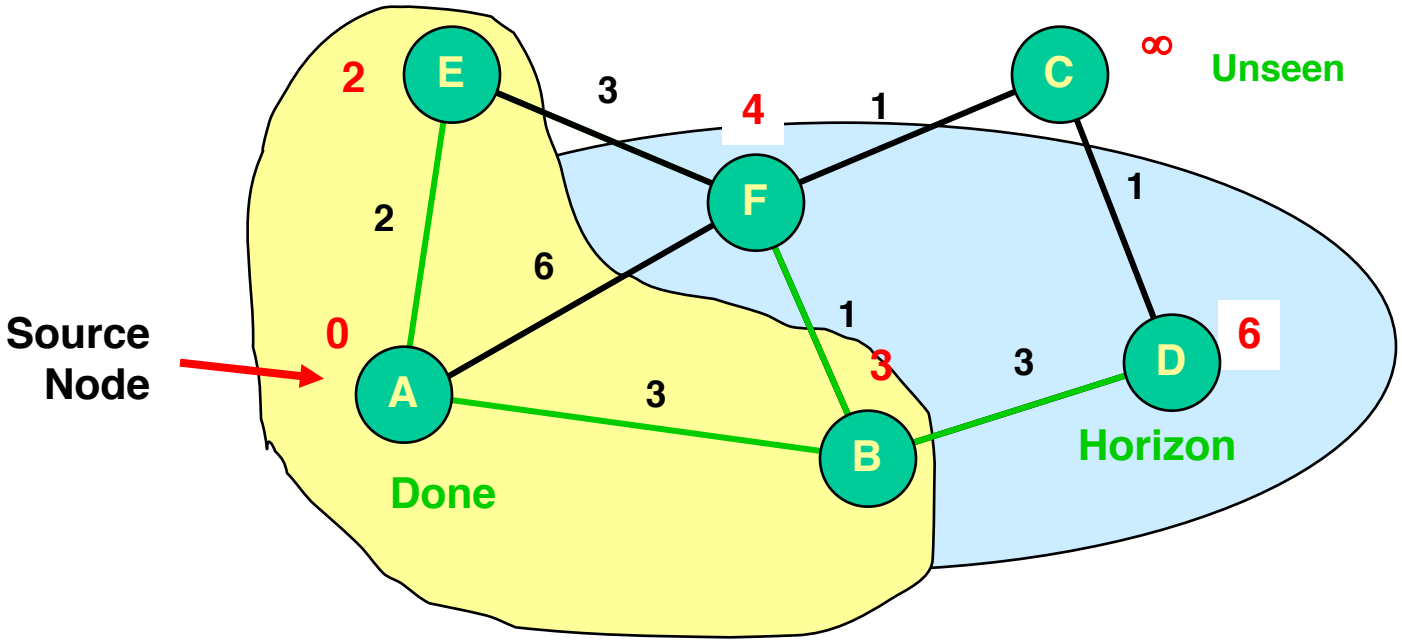
Dijkstra's Algorithm: Selection



Next:

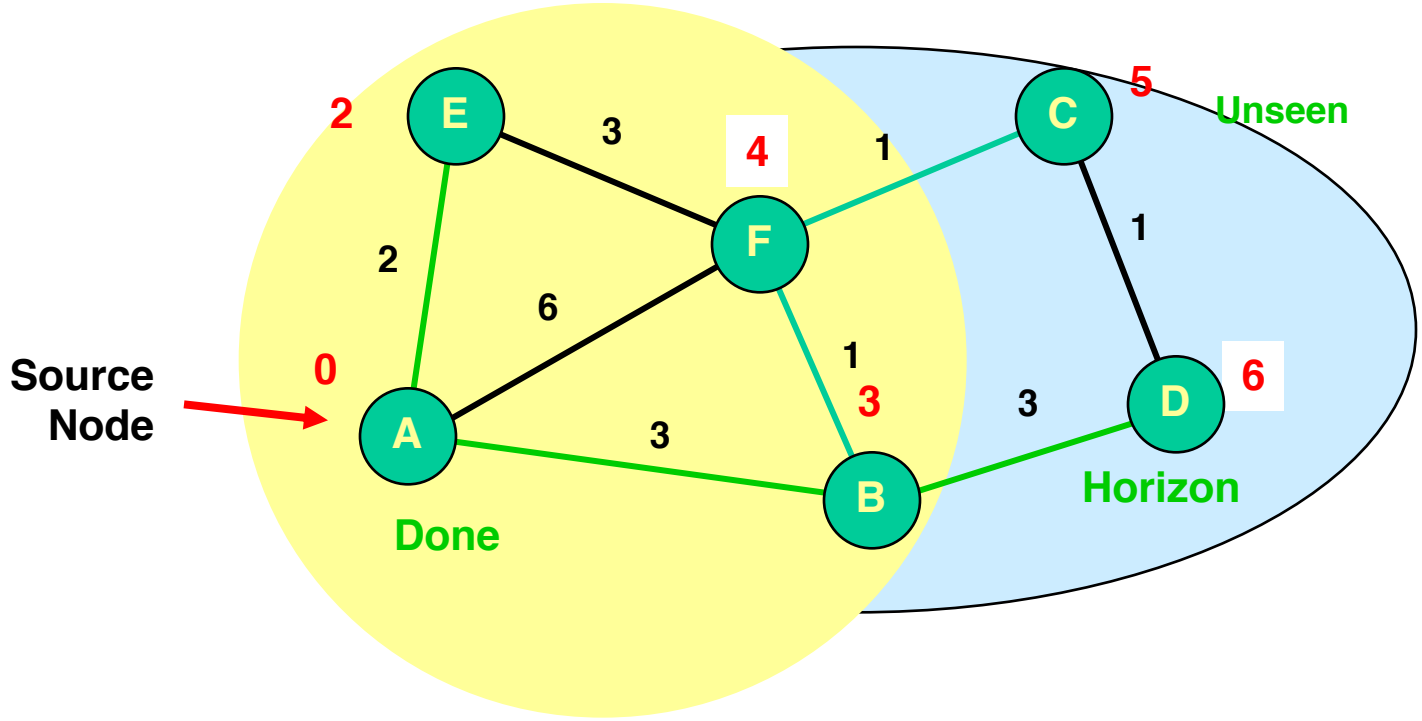
- Update which nodes are in Horizon
- Update costs to nodes in Horizon
- Update last link

Dijkstra's Algorithm: Update



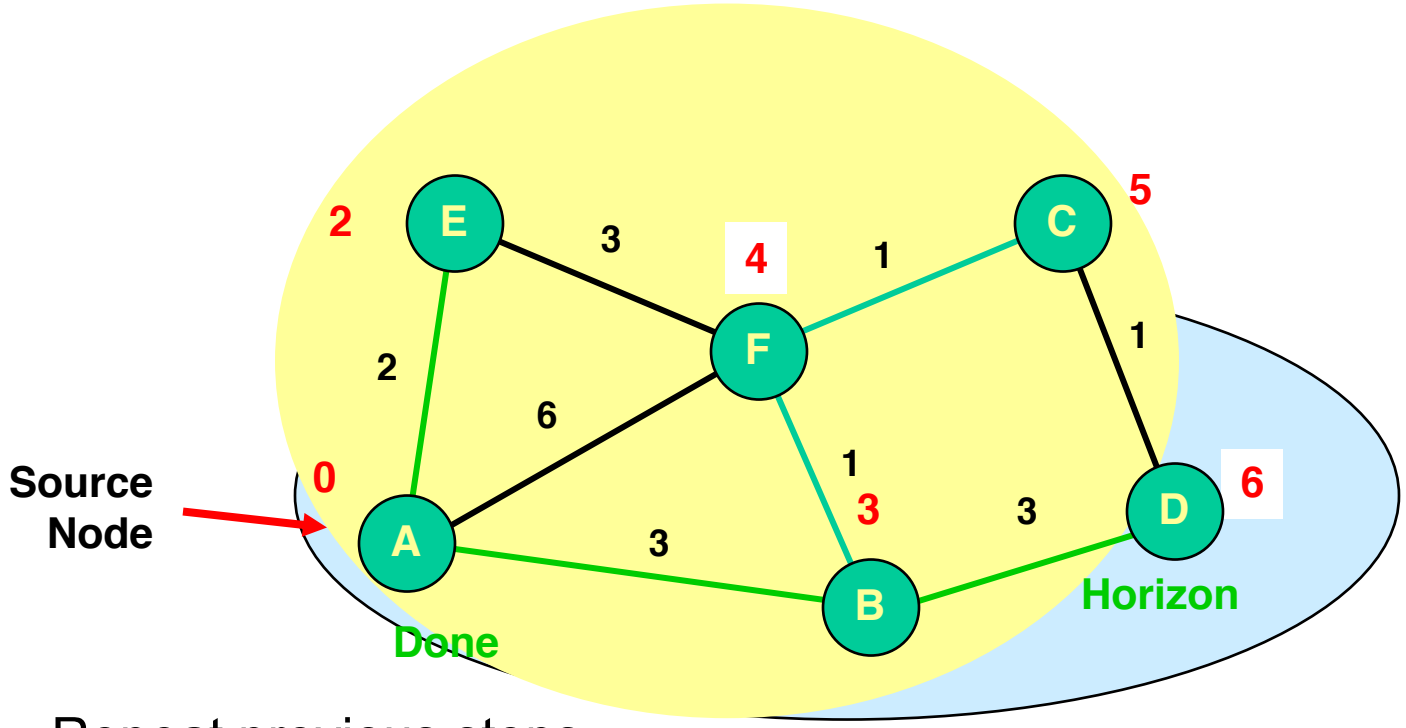
– Repeat previous steps

Dijkstra's Algorithm: Update



– Repeat previous steps

Dijkstra's Algorithm: Update



– Repeat previous steps

Dijkstra's Algorithm: Update