

Network Layer: Link State Routing

Formation of Shortest Path Tree: Dijkstra's Algorithm

In practice, each switch computes its routing table directly from the LSPs it has collected using a realization of Dijkstra's algorithm called the *forward search* algorithm. Specifically, each switch maintains two lists, known as **Tentative** and **Confirmed**. Each of these lists contains a set of entries of the form **(Destination, Cost, NextHop)**. The algorithm works as follows:

1. Initialize the **Confirmed** list with an entry for myself; this entry has a cost of 0.
2. For the node just added to the **Confirmed** list in the previous step, call it node **Next** and select its LSP.
3. For each neighbor (**Neighbor**) of **Next**, calculate the cost (**Cost**) to reach this **Neighbor** as the sum of the cost from myself to **Next** and from **Next** to **Neighbor**.
 1. If **Neighbor** is currently on neither the **Confirmed** nor the **Tentative** list, then add **(Neighbor, Cost, NextHop)** to the **Tentative** list, where **NextHop** is the direction I go to reach **Next**.
 2. If **Neighbor** is currently on the **Tentative** list, and the **Cost** is less than the currently listed cost for **Neighbor**, then replace the current entry with **(Neighbor, Cost, NextHop)**, where **NextHop** is the direction I go to reach **Next**.
4. If the **Tentative** list is empty, stop. Otherwise, pick the entry from the **Tentative** list with the lowest cost, move it to the **Confirmed** list, and return to step 2.

Dijkstra's algorithm - Steps for Building Routing Table for Node D.

Step	Confirmed	Tentative	Comments
1	(D,0,-)		Since D is the only new member of the confirmed list, look at its LSP.
2	(D,0,-)	(B,11,B) (C,2,C)	D's LSP says we can reach B through B at cost 11, which is better than anything else on either list, so put it on Tentative list; same for C.
3	(D,0,-) (C,2,C)	(B,11,B)	Put lowest-cost member of Tentative (C) onto Confirmed list. Next, examine LSP of newly confirmed member (C).
4	(D,0,-) (C,2,C)	(B,5,C) (A,12,C)	Cost to reach B through C is 5, so replace (B,11,B). C's LSP tells us that we can reach A at cost 12.
5	(D,0,-) (C,2,C) (B,5,C)	(A,12,C)	Move lowest-cost member of Tentative (B) to Confirmed , then look at its LSP.
6	(D,0,-) (C,2,C) (B,5,C)	(A,10,C)	Since we can reach A at cost 5 through B, replace the Tentative entry.
7	(D,0,-) (C,2,C) (B,5,C) (A,10,C)		Move lowest-cost member of Tentative (A) to Confirmed , and we are all done.

Dijkstra algorithm

