Year/Sem/Batch: III /V / P

Date: 24-Sep-2026

Course Instructor : Dr. M.S.Karthika Devi, Asst. Prof, DCSE

Distance Vector and Link State Routing

A university has deployed a backbone network connecting six locations through routers R1, R2, R3, R4, R5, and R6. The cost of each link represents the communication cost based on factors such as delay, bandwidth utilization, and network congestion.

The initial network topology and link costs are given below:

Link	Cost
R1 – R2	4
R1 – R3	2
R2 – R3	1
R2 – R4	5
R2 – R5	8
R3 – R4	8
R3 – R5	10
R4 – R5	2
R4 – R6	6
R5 – R6	3

The network administrator needs to determine optimal routes between the different locations and ensure that the routing tables are updated dynamically when the network conditions change.

Question

Implement and analyze Distance Vector and Link State routing algorithms for the given university network.

(a) Distance Vector Routing

Implement the Distance Vector Routing algorithm using the Bellman-Ford approach.

1. Initialize the routing table at each router using information about its directly connected neighbors.
2. Exchange distance vectors between neighboring routers.
3. Update the routing tables based on the Bellman-Ford equation.
4. Repeat the process until all routing tables converge.
5. Display the routing table after each iteration.

For every routing-table entry, display the following information:

Destination | Minimum Cost | Next Hop

(b) Link State Routing

Implement the Link State Routing algorithm using Dijkstra's shortest-path algorithm.

Consider R1 as the source router and perform the following:

1. Construct the link-state database representing the network topology.
2. Execute Dijkstra's algorithm iteratively.
3. For each iteration, display:
    - Confirmed nodes
    - Tentative nodes
    - Minimum cost
    - Predecessor
4. Construct the final shortest-path tree rooted at R1.
5. Generate the routing table for R1 containing the destination, minimum cost, and next-hop router.

(c) Network Cost Change

After the routing tables have converged, the network administrator observes heavy traffic on the R2–R4 link. Therefore, its cost increases from 5 to 12.

Modify the network accordingly and:

1. Identify the routes that may be affected by the change.
2. Recalculate the routing tables using the Distance Vector algorithm.
3. Recalculate the shortest-path tree using the Link State algorithm.
4. Compare the routing information before and after the cost change.
5. Determine whether the route from R1 to R6 changes after the increase in link cost.

(d) Link Failure

Subsequently, the R4–R5 link fails completely and becomes unavailable.

Modify the network topology to represent the link failure and:

1. Update the routing information in the Distance Vector approach.
2. Determine the new route from R1 to R6.
3. Display the routing-table changes during Distance Vector convergence.
4. Re-run Dijkstra's algorithm using the updated topology.
5. Construct the new shortest-path tree rooted at R1.
6. Determine the number of iterations required for the Distance Vector algorithm to converge after the link failure.