

Experiment 4 – Implement Link State / Distance Vector Routing

Aim

Write a program to implement Distance Vector or Link State routing to find suitable path for transmission.

Objectives

Understand how Distance Vector routing protocol works and how routing algorithms find shortest paths.

Important Oral Topics & Answers

1 What is Distance Vector Routing?

Each router maintains a **table of distances** (called distance vector) showing the best known distance to every other node.

Uses **Bellman-Ford Algorithm**.

Routers exchange information with neighbors periodically and update their tables.

Problems:

- Count-to-infinity problem
- Loops in routing tables

Solutions: Split horizon, route poisoning.

2 Bellman-Ford Algorithm

Used to find shortest paths from a single source node to all others in a weighted graph.

Formula:

$$D_x(y) = \min [C(x,v) + D_v(y)]$$

Where:

- $D_x(y)$: cost from node x to y
- $C(x,v)$: cost to reach neighbor v
- $D_v(y)$: cost from v to y

3 What is Link State Routing?

Each router knows the full network topology.

It sends information about its neighbors to all routers using **flooding**.

Uses **Dijkstra's Algorithm** to find shortest path.

Advantages: Faster convergence, accurate routes.

Disadvantages: More processing and memory usage.

4 Difference between Distance Vector and Link State

Feature	Distance Vector	Link State
Algorithm	Bellman-Ford	Dijkstra
Information shared	To neighbors only	To all routers
Update type	Periodic	Event-driven
Speed	Slower	Faster
Traffic	Low	High (flooding)

Outcome

Understood and implemented Distance Vector algorithm to find suitable path for data transmission.