Course Name:	MCAN Laboratory	Semester:	VI
Date of Performance:	07 / 04 / 25	Batch No.:	B - 2
Faculty Name:	Dr. Rajashree Daryapurkar	Roll No.:	16014022050
Faculty Sign & Date:		Grade/Marks:	___ / 25

Experiment No.: 8
Title: Experiment on OLSR Routing in MANET

Aim and Objective of the Experiment:

To study and analyze the performance of the OLSR (Optimized Link State Routing) protocol in a Mobile Ad Hoc Network (MANET) environment.

COs to be achieved:

CO3: Understand the current topics in MANETs and WSNs, both from an industry and research point of views.
CO4: Analyze how proactive routing protocols function and their implications on data trans0mission delay and bandwidth consumption.

Books/Journals/Websites referred:

NetSim User Manual

Tools required:

NetSim software

Theory:

OLSR (Optimized Link State Routing) is a **proactive routing protocol** designed for **Mobile Ad Hoc Networks (MANETs)**. Unlike reactive protocols that discover routes on-demand, OLSR continuously maintains and updates routing information between all nodes in the network, allowing for immediate route availability when data transmission is required.

OLSR optimizes the traditional link state algorithm by using **Multipoint Relays (MPRs)**. Each node selects a set of MPRs from its neighbors, which are responsible for forwarding control messages. This significantly reduces the overhead of flooding control messages across the network. The protocol periodically exchanges **Hello messages** for link sensing and **Topology Control (TC) messages** for route calculation. By maintaining up-to-date topology information, OLSR ensures efficient and loop-free routing in dynamic MANET environments.

In this experiment, OLSR is implemented or simulated to analyze its behavior, efficiency, and performance in terms of routing overhead, packet delivery, and delay under different network scenarios.

Implementation details:

1. **Enlist all the Steps followed and various options explored.**
    a. Opened NetSim and selected the MANET simulation environment.
    b. Chose OLSR (Optimized Link State Routing) as the routing protocol.
    c. Configured simulation parameters such as:
        - Number of nodes
        - Mobility model (e.g., Random Waypoint)
        - Simulation duration
        - Transmission range, data rate, and node speed
        - Application type (e.g., CBR – Constant Bit Rate traffic)
    d. Explored different node densities and mobility patterns to observe the behavior of OLSR.
    e. Compared OLSR performance with other routing protocols like AODV and DSR under similar conditions.
    f. Analyzed output metrics such as Packet Delivery Ratio (PDR), End-to-End Delay, and Routing Overhead.

2. **Explain your program logic and methods used.**
    a. OLSR Protocol is proactive; it continuously updates routing tables at every node using periodic control messages.
    b. The simulation logic works as follows:
        - Hello Messages are exchanged to detect and maintain link status with neighbors.
        - Multipoint Relays (MPRs) are selected to minimize the number of transmissions during route updates.
        - Topology Control (TC) Messages are used to disseminate link state information throughout the network via MPRs.
        - Each node uses this information to compute the shortest path routes to all other nodes in the network.
    c. In NetSim, OLSR operations such as neighbor discovery, MPR selection, and routing table updates are handled internally.
    d. Performance data was extracted from trace files and simulation logs using NetSim's analysis tools for result interpretation.

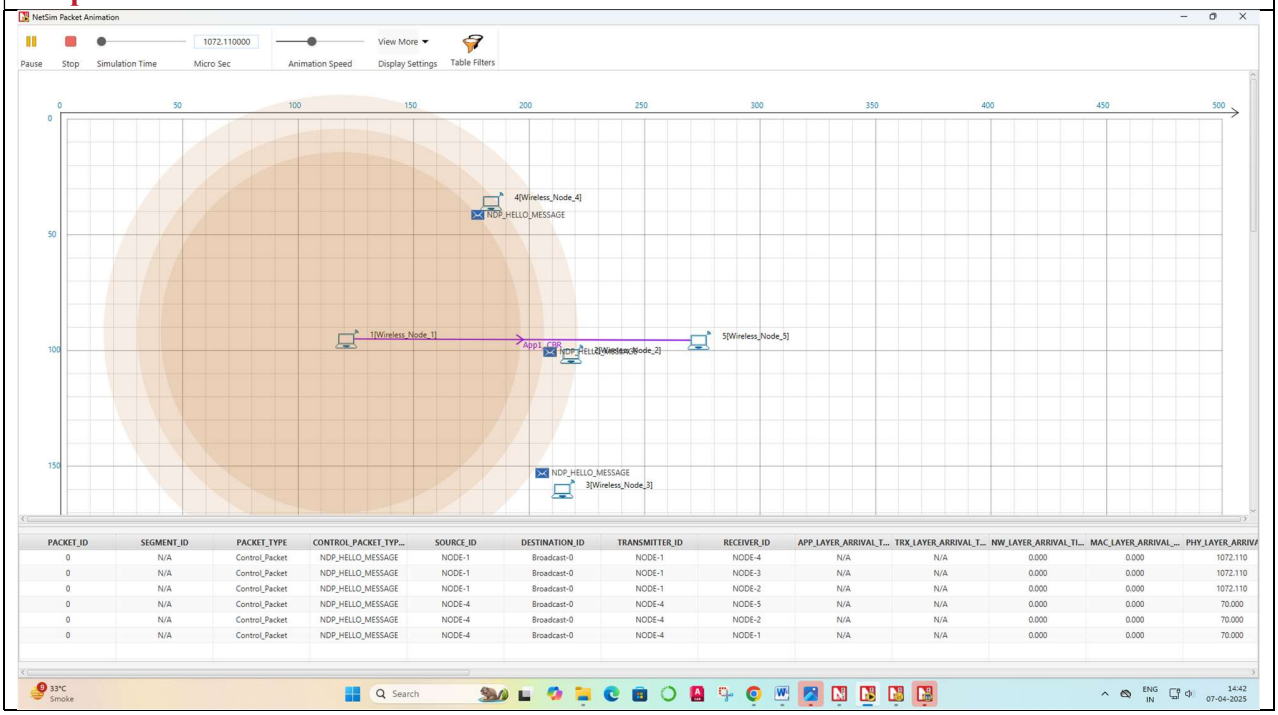
Procedure:

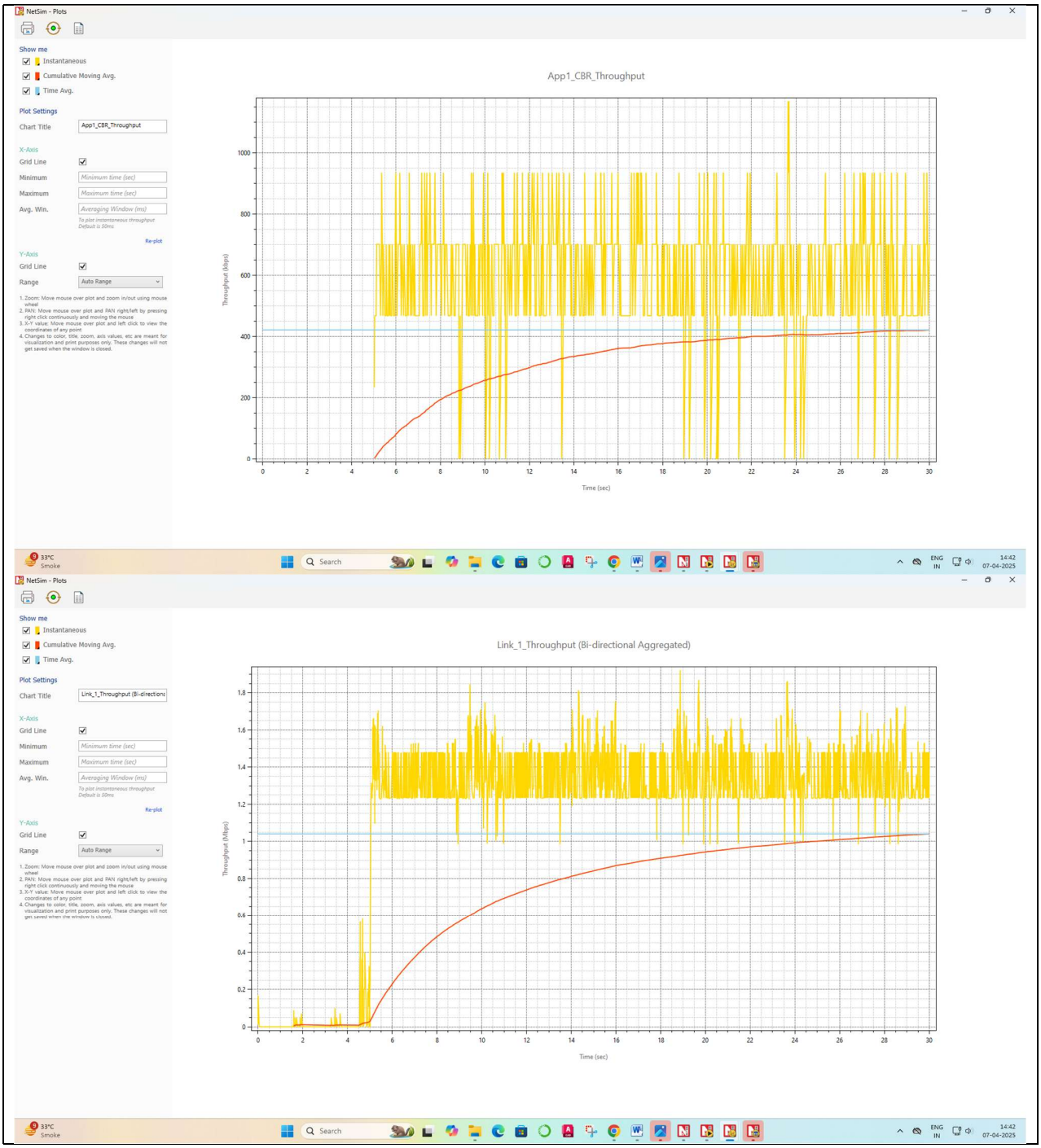
1. Opened NetSim and selected the MANET simulation environment.
2. Chose OLSR (Optimized Link State Routing) as the routing protocol.
3. Configured simulation parameters such as:
    - Number of nodes

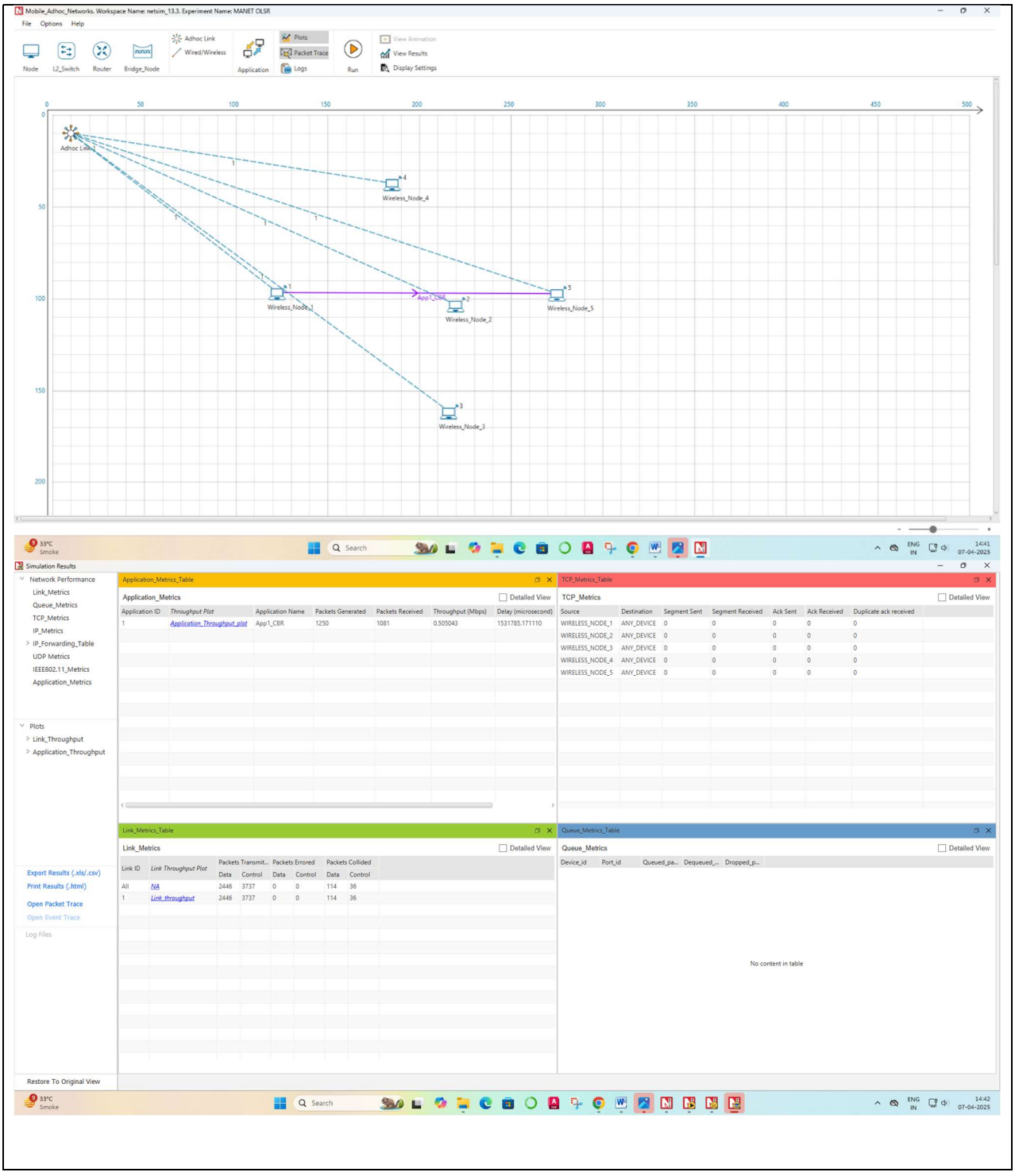
- Mobility model (e.g., Random Waypoint)
- Simulation duration
- Transmission range, data rate, and node speed
- Application type (e.g., CBR – Constant Bit Rate traffic)

4. Explored different node densities and mobility patterns to observe the behavior of OLSR.
5. Compared OLSR performance with other routing protocols like AODV and DSR under similar conditions.
6. Analyzed output metrics such as Packet Delivery Ratio (PDR), End-to-End Delay, and Routing Overhead.

Output/results after execution:




NetSim - Plots
Show me
Instantaneous
Cumulative Moving Avg.
Time Avg.
Plot Settings
Chart Title
App1_CBR_Throughput
X-Axis
Grid Line
Minimum
Maximum
Avg. Win.
Re-plot
Y-Axis
Grid Line
Range
Auto Range
App1_CBR_Throughput
Throughput (Mbps)
Time (sec)
NetSim - Plots
Show me
Instantaneous
Cumulative Moving Avg.
Time Avg.
Plot Settings
Chart Title
Link_1_Throughput (Bi-directional Aggregated)
X-Axis
Grid Line
Minimum
Maximum
Avg. Win.
Re-plot
Y-Axis
Grid Line
Range
Auto Range
Link_1_Throughput (Bi-directional Aggregated)
Throughput (Mbps)
Time (sec)


Mobile_Adhoc_Networks. Workspace Name: netsim_13.3. Experiment Name: MANET OLSR
File Options Help
Node L2_Switch Router Bridge_Node
Adhoc Link Wired/Wireless
Application
Plots Packet Trace Logs
Run
View Animation View Results Display Settings
Adhoc Link
Wireless_Node_4
Wireless_Node_1
App1_CBR
Wireless_Node_2
Wireless_Node_5
Wireless_Node_3
Simulation Results
Network Performance
Link_Metrics
Queue_Metrics
TCP_Metrics
IP_Metrics
IP_Forwarding_Table
UDP Metrics
IEEE802.11_Metrics
Application_Metrics
Plots
Link_Throughput
Application_Throughput
Application_Metrics_Table
Application_Metrics
Detailed View
Application ID | Throughput Plot | Application Name | Packets Generated | Packets Received | Throughput (Mbps) | Delay (microsecond)
1 | Application_Throughput_plot | App1_CBR | 1250 | 1081 | 0.505043 | 1531785.171110
TCP_Metrics_Table
TCP_Metrics
Detailed View
Source | Destination | Segment Sent | Segment Received | Ack Sent | Ack Received | Duplicate ack received
WIRELESS_NODE_1 | ANY_DEVICE | 0 | 0 | 0 | 0 | 0
WIRELESS_NODE_2 | ANY_DEVICE | 0 | 0 | 0 | 0 | 0
WIRELESS_NODE_3 | ANY_DEVICE | 0 | 0 | 0 | 0 | 0
WIRELESS_NODE_4 | ANY_DEVICE | 0 | 0 | 0 | 0 | 0
WIRELESS_NODE_5 | ANY_DEVICE | 0 | 0 | 0 | 0 | 0
Link_Metrics_Table
Link_Metrics
Detailed View
Link ID | Link Throughput Plot | Packets Transmit... Data | Packets Transmit... Control | Packets Errored Data | Packets Errored Control | Packets Collided Data | Packets Collided Control
All | NA | 2446 | 3737 | 0 | 0 | 114 | 36
1 | Link_throughput | 2446 | 3737 | 0 | 0 | 114 | 36
Queue_Metrics_Table
Queue_Metrics
Detailed View
Device_id | Port_id | Queued_pa... | Dequeued_... | Dropped_p...
No content in table
Export Results (.xls/.csv)
Print Results (.html)
Open Packet Trace
Open Event Trace
Log Files
Restore To Original View

Post Lab Subjective/Objective type Questions:

1. **Explain difference between LSR and OLSR.**

Aspect	LSR (Link State Routing)	OLSR (Optimized Link State Routing)
Type	Traditional link-state protocol (used in wired networks)	Modified link-state protocol optimized for MANETs
Flooding	Floods link state information to all nodes	Uses **Multipoint Relays (MPRs)** to reduce flooding overhead
Control Overhead	High, due to global broadcasting of link-state packets	Lower, due to selective forwarding via MPRs
Mobility Support	Designed for static or low-mobility networks	Specifically designed for **mobile ad hoc networks**
Routing Updates	Periodic full updates	Periodic updates using **Hello** and **TC messages** with optimization

2. **Explain the following terms with reference to OLSR.**
   - **Neighbor detection**
     1. OLSR uses Hello messages exchanged periodically between nodes.
     2. These messages help identify one-hop and two-hop neighbors.
     3. Based on this information, each node selects a subset of one-hop neighbors as Multipoint Relays (MPRs).
     4. This detection process helps in maintaining an updated view of the local topology.

   - **Topology discovery**
     1. After neighbors are identified and MPRs are selected, MPR nodes generate Topology Control (TC) messages.
     2. TC messages are broadcasted across the network only by MPRs, not all nodes.
     3. These messages carry information about the node's MPR selectors (i.e., nodes that chose it as an MPR).
     4. Using TC messages, all nodes build a partial link state table, sufficient to compute routes to all nodes.

Conclusion:

The OLSR routing protocol efficiently maintains up-to-date routes in MANETs using proactive link state updates. Its use of Multipoint Relays (MPRs) reduces control overhead and improves scalability in dynamic networks.

Signature of faculty in-charge with Date: