

Mobile Adhoc Networks (MANETs)

Contents

1	Introduction	3
2	Simulation GUI.....	3
	2.1 Fast Configuration	4
	2.2 Create Scenario.....	6
	2.2.1 Multiple MANETs	6
	2.3 Set Node, Link and Application Properties.....	9
	2.4 Enable Packet Trace, Event Trace & Plots (Optional).....	11
	2.5 Run Simulation	11
3	Model Features	12
	3.1 Adhoc On Demand Distance Vector Routing	12
	3.2 Link Layer Acks and Network Layer Acks in DSR.....	12
	3.2.1 Using Link-Layer Acknowledgements.....	12
	3.2.2 Using Network-Layer Acknowledgements	13
	3.2.3 AODV/DSR Metrics.....	14
	3.3 Mobility models in Netsim	15
	3.3.1 Random Walk Mobility Model.....	15
	3.3.2 Random Waypoint Mobility Model	15
	3.3.3 Group Mobility Model	16
	3.3.4 File Based Mobility Model	16
	3.4 Visualizing node movement in NetSim animator	18
	3.5 How are packet collisions modelled in NetSim?.....	19
	3.6 How is packet error decided in NetSim?	19
4	Featured Examples.....	20
	4.1 AODV Routing	20
	4.2 ZRP-IARP.....	22
	4.3 ZRP-IERP.....	26
	4.4 MANET-OLSR	27
	4.4.1 Optimized Link State Routing Protocol.....	27
	4.4.2 Neighbor detection	28
	4.4.3 Topology discovery	28

4.4.4 TC message.....	28
4.5 Multiple MANETS	30
4.6 MANET- Battery lifetime analysis.....	31
4.7 MANET File Based Placement	34
5 Reference Documents.....	36
6 Latest FAQs	36

1 Introduction

NetSim MANET supports the following protocols

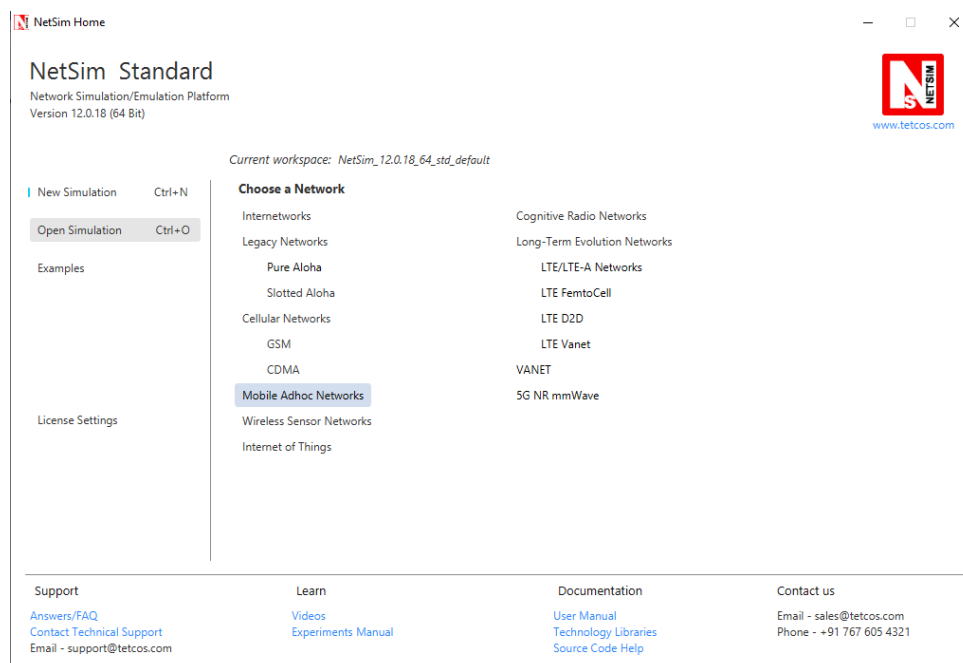
- Layer 3
 - Dynamic Source Routing (DSR)
 - Adhoc On demand Distance Vector Routing (AODV)
 - Optimized Link state Routing (OLSR)
 - Zone Routing Protocol (ZRP)
- MAC / PHY
 - 802.11 a, b, g, n, ac and e

NetSim MANETs component can be interfaced with

- Component 6 (IOT) module to run 802.15.4 in MAC/PHY
- Component 9 (VANETs) module to run IEEE 1609 WAVE in MAC/PHY
- Military Radio Add on to run TDMA/DTDMA in MAC/PHY

2 Simulation GUI

In the Main menu select →**New Simulation**→**Mobile Adhoc networks**



2.1 Fast Configuration

Grid Settings and Device Placement

Total Grid Settings

Grid Length (m)

Grid Length should be between 50 m to 1,000,000 m

Device Grid Settings

Side Length (m)

Side length should be a multiple of 50 but less than or equal to Grid Length. This is a sub-grid in which the devices would be placed automatically

Device Placement Strategy

☐ Automatic Placement

☒ Uniform Placement

☐ Random Placement

☐ File Based Placement

Number of Devices

☒ Manually Via Click and Drop

OK Cancel

Fast Config window allows users to define device placement strategies and conveniently model large network scenarios especially in network such as MANET, WSN and IoT. The parameters associated with the Fast Config Window is explained below:

Grid length: It is the area of simulation environment. Users can change the length of the grid in the range of 50-1000000m

Side length: It specifies the area in the grid environment within which the devices will be placed. User can change the side length of the grid in the range of 50 - 1000000m. Side length should be multiple of 50. Side length should always be set to a value less than or equal to the Grid length.

Device Placement:

Automatic Placement:

1. Uniform Placement: Devices will be placed uniformly with equal gap between the devices in area as specified in side length. This requires users to specify the number of devices as square number. For Eg: 1, 4, 9, 16 etc
2. Random Placement: Devices will be placed randomly in the grid environment within the area as specified in side length.
3. File Based Placement: Inorder to place devices in user defined locations file based placement option can be used. The file has the following general format:

<DEVICE_NAME>,<DEVICE_TYPE>,<X_COORDINATE>,<Y_COORDINATE>

Where,

DEVICE_NAME – is any name that will be assigned to the device.

DEVICE_TYPE – is the unique Device Identifier specific to each type of device in NetSim.

Following table provides the DEVICE_TYPE's of all possible devices for networks with support for Device Fast Configuration:

NETWORK	DEVICE_TYPE
Multiple manets	1. WIRELESSNODE 2. BRIDGE_WIREDNODE 3. BRIDGE_WIRELESSNODE 4. WIREDNODE 5. ACCESSPOINT 6. ROUTER 7. L2_SWITCH
WSN	1. Sensors 2. SinkNode
IOT	1. IOT_Sensors 2. LOWPAN_Gateway 3. WIREDNODE 4. IOT_ROUTER

	5. ACCESSPOINT
	6. L2_SWITCH

X_COORDINATE – the value of X coordinate of the device

Y_COORDINATE – the value of Y coordinate of the device

Eg:

Wireless_Node,WIRELESSNODE,100,400

Wireless_Node,WIRELESSNODE,400,100

Wireless_Node,WIRELESSNODE,100,100

Wireless_Node,WIRELESSNODE,400,400

4. Number of Devices

It is the total number of devices that is to be placed in the grid environment. It should be a square number in case of Uniform placement.

5. Manually Via Click and Drop

Selecting this option will load an empty grid environment where users can add devices by clicking and dropping the devices as required.

2.2 Create Scenario

2.2.1 Multiple MANETs

Select “Manually via Click and Drop” option in the Multiple MANETs fast config window as shown below and click on OK button.

Grid Settings and Device Placement

Total Grid Settings

Grid Length (m)

Grid Length should be between 50 m to 1,000,000 m

Device Grid Settings

Side Length (m)

Side length should be a multiple of 50 but less than or equal to Grid Length. This is a sub-grid in which the devices would be placed automatically

Device Placement Strategy

☐ Automatic Placement
☒ Uniform Placement
☐ Random Placement
☐ File Based Placement

Number of Devices

☒ Manually Via Click and Drop

OK

Cancel

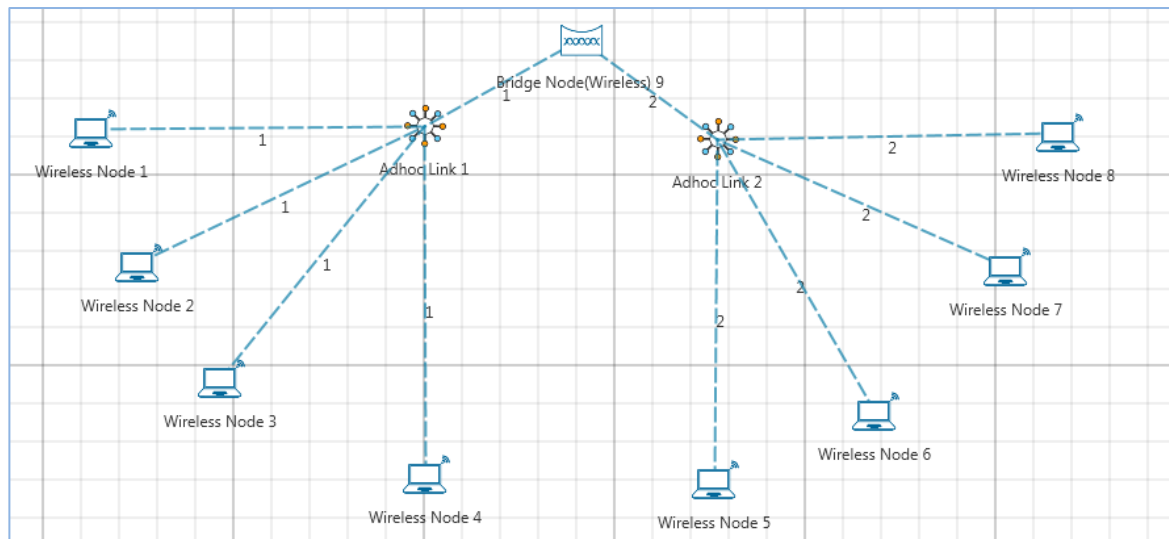
Multiple MANETs allows users to interconnect two or more MANETs using a bridge node. Click and drop Wireless nodes, Adhoc links and Bridge Node onto the grid environment as shown below:

The diagram illustrates a grid environment for network simulation. It features a central 'Bridge Node(Wireless) 9' at the top, which is a blue icon with multiple connection points. Below the bridge node are two 'Adhoc Link' nodes, also represented by blue icons with multiple connection points. Eight 'Wireless Node' icons, each represented by a blue laptop icon, are distributed across the grid. The nodes are labeled 'Wireless Node 1' through 'Wireless Node 8'.

Ver 12.0

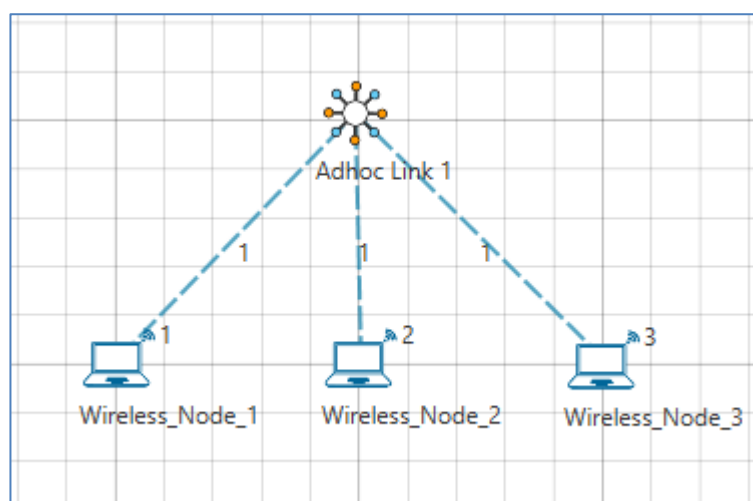
7

Click on Adhoc Link icon (in ribbon). Then click on adhoc link icon (in grid) and on Wireless node (in grid) to connect Adhoc link to Wireless Node. Similarly connect Wireless Nodes to Adhoc links to form two different MANET's using Adhoc links. Further connect the two MANET's using a bridge node as shown below:



Note:

1. Multiple MANETs do not support OLSR and ZRP protocols which is available in single MANETs
2. Bridge Node is not required for creating Single Manet scenario and a sample single MANET scenario would look like:



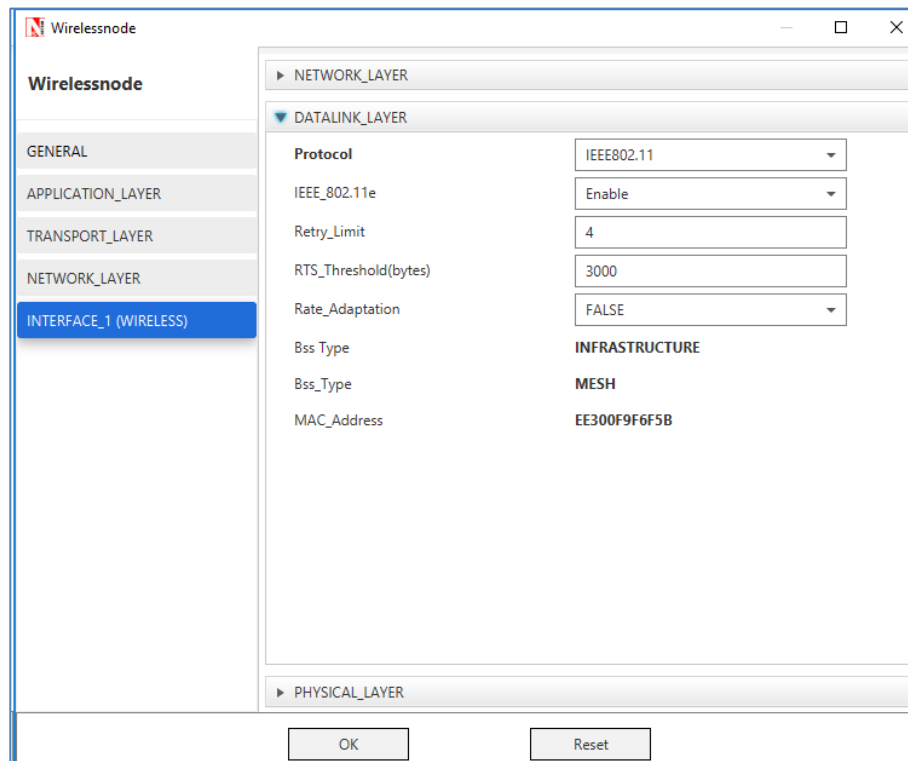
2.2.1.1 Bridge Node

Bridge Node acts as a bridge/interface/gateway between multiple MANETs. Packets coming from MANET-1 are routed to MANET-2 via the Bridge Node. Each Bridge Node has 24 interfaces. If the bridge node is connected via a wired interface, then the velocity in mobility model is automatically set to 0 and cannot be modified. When connecting bridge nodes to one another or to routers care should be taken to ensure that the static route must be set accurately. This is required since the bridge nodes do not run any routing protocol.

2.3 Set Node, Link and Application Properties

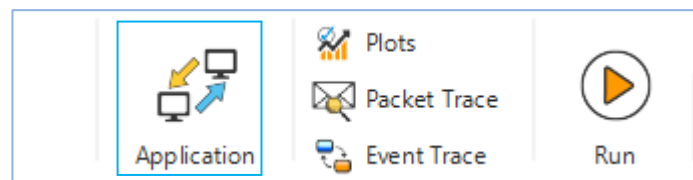
Right click on the appropriate node or link and select Properties. Then modify the parameters according to the requirements.

- Global Properties: Certain properties are global in nature, i.e changing properties in one node will automatically reflect in the others in that network.
- In case of MANET, in Wireless Node, Routing Protocol in Network Layer and all user editable properties in DataLink Layer, Physical Layer and Power are Global
- The following are the main properties of wireless node in Datalink and Physical layers



The image shows the 'Wirelessnode' configuration window. On the left, a sidebar lists layers: GENERAL, APPLICATION_LAYER, TRANSPORT_LAYER, NETWORK_LAYER, and INTERFACE_1 (WIRELESS). The main area is divided into two sections. The top section, 'PHYSICAL_LAYER', includes settings for Protocol (IEEE802.11), Standard (IEEE802.11b), Transmission_Type (DSSS), Trmitter_Power(mW) (100), Antenna Gain (0), Antenna Height(m) (1), SIFS(Microsec) (10), Frequency_Band(GHz) (2.4), Bandwidth(MHz) (20), CCA Mode, SlotTime(Microsec) (20), Standard_Channel(MHz) (1_2412), CW_Min (31), and CW_Max (1023). The bottom section, 'POWER', includes settings for Reference Distance d0(m) (1), Connection_Medium (WIRELESS), PowerSource (Battery), EnergyHarvesting (On), RechargingCurrent(mA) (0.4), InitialEnergy(mAH) (0.5), TransmittingCurrent(mA) (8.8), IdleModeCurrent(mA) (3.3), Voltage(V) (3.6), ReceivingCurrent(mA) (9.6), and SleepModeCurrent(mA) (0.237). At the bottom are 'OK' and 'Reset' buttons.

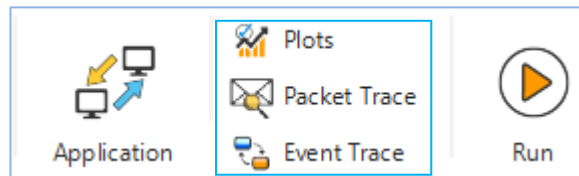
- Click on the Application icon present on the ribbon and set properties. Multiple applications can be generated by using add button in Application properties.



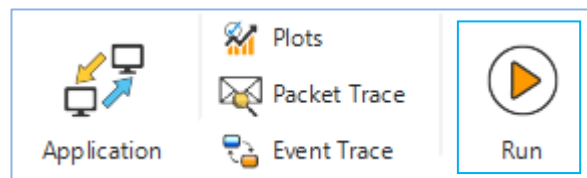
- Set the values according to requirement and click Accept.

2.4 Enable Packet Trace, Event Trace & Plots (Optional)

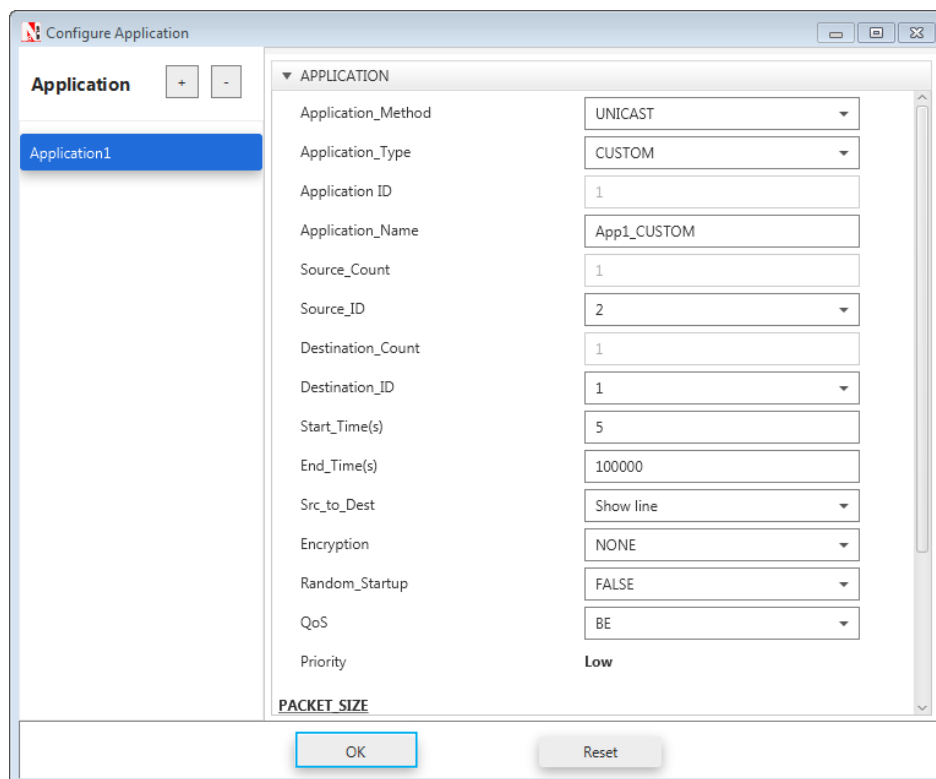
Click Packet Trace / Event Trace icon in the tool bar and check Enable Packet Trace / Event Trace check box and click OK. To get detailed help, please refer sections 7.5 and 7.6 in User Manual. Select Plots icon for enabling Plots and click OK.



2.5 Run Simulation



Click on Run Simulation icon on the top toolbar. Set the Simulation Time and click on OK.



Note on MANET implementation in NetSim:

- If user wants to implement HTTP application among Nodes, TCP must be enabled in Source Node as TCP is set to disable by default.

- OLSR is a proactive link-state routing protocol, which uses hello and topology control (TC) messages to discover and then disseminate link state information throughout the mobile ad hoc network.
- Individual nodes use this topology information to compute next hop destinations for all nodes in the network using shortest hop forwarding paths. For topology control (TC) messages to disseminate throughout, it requires 5 or more seconds depending upon the network size. In general, it is $(5.5 \text{ secs} + \text{Tx_Time} * \text{network size})$.
- Hence, when simulating OLSR in MANET the Application Start Time must be greater than 5s (preferably greater than 10s) because in OLSR Topology Control (TC) messages start at 5s. Once the TC messages are sent, some further time will be required for OLSR to find the route. This can be done by setting the “Starting time” parameter in Application.

3 Model Features

3.1 Adhoc On Demand Distance Vector Routing

Implemented Features

- RREQ, RREP and RERR messages
- Hello message
- Interface with other MAC/PHY protocols such as 802.15.4, TDMA / DTDMA

3.2 Link Layer Acks and Network Layer Acks in DSR

Route Maintenance is the mechanism by which a source node S is able to detect, while using a source route to some destination node D, if the network topology has changed such that it can no longer use its route to D because a link along the route no longer works.

3.2.1 Using Link-Layer Acknowledgements

If the MAC protocol in use provides feedback as to the successful delivery of a data packet (such as is provided for unicast packets by the link-layer acknowledgement frame defined by IEEE 802.11), then the use of the DSR Acknowledgement Request and Acknowledgement options is not necessary. If such link-layer feedback is available, it SHOULD be used instead of any other acknowledgement mechanism for Route Maintenance, and the node SHOULD

NOT use either passive acknowledgements or network-layer acknowledgements for Route Maintenance.

When using link-layer acknowledgements for Route Maintenance, the retransmission timing and the timing at which retransmission attempts are scheduled are generally controlled by the particular link layer implementation in use in the network. For example, in IEEE 802.11, the link-layer acknowledgement is returned after a unicast packet as a part of the basic access method of the IEEE 802.11 Distributed Coordination Function (DCF) MAC protocol; the time at which the acknowledgement is expected to arrive and the time at which the next retransmission attempt (if necessary) will occur are controlled by the MAC protocol implementation.

3.2.2 Using Network-Layer Acknowledgements

When a node originates or forwards a packet and has no other mechanism of acknowledgement available to determine reachability of the next-hop node in the source route for Route Maintenance, that node **SHOULD** request a network-layer acknowledgement from that next-hop node. To do so, the node inserts an Acknowledgement Request option in the DSR Options header in the packet. The Identification field in that Acknowledgement Request option **MUST** be set to a value unique over all packets recently transmitted by this node to the same next-hop node.

When using network-layer acknowledgements for Route Maintenance, a node **SHOULD** use an adaptive algorithm in determining the retransmission timeout for each transmission attempt of an acknowledgement request. For example, a node **SHOULD** maintain a separate round-trip time (RTT) estimate for each node to which it has recently attempted to transmit packets, and it should use this RTT estimate in setting the timeout for each retransmission attempt for Route Maintenance.

While simulating certain network configurations, users may see that packets received are more than packets sent. This is because:

- This is being measured as part of our UDP protocol metrics in layer 4 in the source and in the destination
- Let us say UDP protocol at source node A sends a datagram. At the MAC - WLAN send the frame and starts a re-transmission timer.
- If no Ack is received within this timer period it would initiate a re-transmission (consider cases where the WLAN Ack has a collision or is errored)
- As the destination the MAC (WLAN) layer would send up to UDP both the first packet it received and the re-transmitted packet it received.

- UDP protocol in the destination would count both the packets received

The screenshot shows the 'Application' configuration window. On the left, under 'Application', 'Application1' is selected. The main configuration area is titled 'APPLICATION' and contains the following fields:

- Application_Method: UNICAST
- Application_Type: CUSTOM
- Application_ID: 1
- Application_Name: App1_CUSTOM
- Source_Count: 1
- Source_ID: 2
- Destination_Count: 1
- Destination_ID: 1
- Start_Time(s): 0
- End_Time(s): 100000
- Src_to_Dest: Show line
- Encryption: NONE
- Random_Startup: FALSE
- QoS: BE
- Priority: Low

Below these fields is a section titled 'PACKET_SIZE' with a 'Distribution' field set to 'CONSTANT'. At the bottom of the window are 'ACCEPT' and 'RESET' buttons.

3.2.3 AODV/DSR Metrics

AODV or DSR Metrics table will be part of the results dashboard if routing protocol in the network layer is set to either AODV or DSR in atleast one device in the network scenario simulated.

Parameter	Description
Device Id	It is the unique ID of the wireless node
RREQ sent	It is the number of Route Request packets sent by wireless node during Route Discovery process
RREQ forwarded	It is the number of Route Request packets forwarded by wireless node during Route Discovery process
RREP sent	It is the number of Route Reply packets sent by wireless node when route is found during Route Discovery process
RREP forwarded	It is the number of Route Reply packets forwarded by a wireless node when route is found during Route Discovery process

RERR sent	It is the total number of Route Error packets sent by a wireless node during Route Maintenance
RERR forwarded	It is the total number of Route Error packets forwarded by a wireless node during Route Maintenance
Packet originated	It is the total number of packets originated in a source node
Packet transmitted	It is the total number of packets transmitted by a source node and intermediate device (LoWPAN Gateway or sink node)
Packet dropped	It is the total number of packets dropped by a wireless node

3.3 Mobility models in Netsim

Mobility models represent the movement of mobile user, and how their location, velocity and acceleration change over time. Such models are frequently used for simulation purposes when new communication or navigation techniques are investigated, or to evaluate the performance of mobile wireless systems and the algorithms and protocols at the basis of them. Typical mobility models provided in NetSim are as follows:

3.3.1 Random Walk Mobility Model

It is a simple mobility model based on random directions and speeds. In this mobility model, a mobile node moves from its current location to a new location by randomly choosing a direction and speed in which to travel. The new speed and direction are both chosen from pre-defined ranges. Each movement in the Random Walk Mobility Model occurs in either a constant time interval or a constant distance traveled, at the end of which a new direction and speed are calculated.

3.3.2 Random Waypoint Mobility Model

It includes pause time between changes in direction and/or speed. A mobile node begins by staying in one location for a certain period of time (i.e., a pause time). Once this time expires, the mobile node chooses a random destination in the simulation area and a speed that is uniformly distributed between [minspeed, maxspeed]. The mobile node then travels toward the newly chosen destination at the selected speed. Upon arrival, the mobile node pauses for a specified time period before starting the process again.

3.3.3 Group Mobility Model

It is a model which describes the behavior of mobile nodes as they move together i.e. the devices having common group id will move together.

3.3.4 File Based Mobility Model

In File Based Mobility, users can write their own custom mobility models and define the movement of the mobile users. The name of the trace file generated should be kept as mobility.txt and it should be in the **NetSim Mobility File format**.

The user can also generate the mobility files using external tools like SUMO (Simulation of Urban Mobility), Vanet MobiSim etc.

The NetSim Mobility File format is as follows

Step 1: Open node (Wireless_Node(MANET) or UE(LTE Networks)) properties -> General Properties -> select mobility model as File_Based_Mobility and click on Open Mobility file.

Step 2: Inside the text file and write the code in format shown below

To write comments, use # tag

User needs to specify the total number of Nodes and Environment size as shown below

#nodes: <No. of Nodes> max x = <X_Environment_Size>, max y: <Y_Environment_Size>

#First specify the location of all the devices as per their X, Y and Z Axis

\$node_(<Node _ID - 1>) set X_ <Initial X_Coordinate>

\$node_(<Node _ID - 1>) set Y_ <Initial Y_ Coordinate >

\$node_(<Node _ID - 1>) set Z_ <Initial Z_ Coordinate >

#Specify the new location of the specific device at the specific time

\$time <Time_in_Secs> "\$node_(<Node_ ID - 1>) <X_Coordinate><Y_ Coordinate ><Z_ Coordinate >"

A sample file based mobility experiment is present at <NetSim Installed Directory> \Docs\ Sample_Configuration\ MANET.

A sample mobility.txt file for a MANET network containing 2 nodes is shown below

#

#nodes: 2, max x = 500.0, max y = 500.0

#

\$node_(0) set X_ 70.0

\$node_(0) set Y_ 70.0

\$node_(0) set Z_ 0.0

\$node_(1) set X_ 150.0

\$node_(1) set Y_ 150.0

\$node_(1) set Z_ 0.0

\$time 0.0 "\$node_(0) 70.00 70.00 0.00"

\$time 0.0 "\$node_(1) 150.0 150.0 0.0"

\$time 5.0 "\$node_(0) 100.00 70.00 0.00"

\$time 5.0 "\$node_(1) 150.0 160.0 0.0"

\$time 10.0 "\$node_(0) 130.00 70.00 0.00"

\$time 10.0 "\$node_(1) 150.0 170.0 0.0"

\$time 15.0 "\$node_(0) 160.00 70.00 0.00"

\$time 15.0 "\$node_(1) 150.0 180.0 0.0"

\$time 20.0 "\$node_(0) 190.00 70.00 0.00"

\$time 20.0 "\$node_(1) 150.0 190.0 0.0"

\$time 25.0 "\$node_(0) 220.00 70.00 0.00"

\$time 25.0 "\$node_(1) 150.0 200.0 0.0"

\$time 30.0 "\$node_(0) 250.00 70.00 0.00"

\$time 30.0 "\$node_(1) 150.0 210.0 0.0"

\$time 35.0 "\$node_(0) 280.00 70.00 0.00"

\$time 35.0 "\$node_(1) 150.0 220.0 0.0"

\$time 40.0 "\$node_(0) 310.00 70.00 0.00"

\$time 40.0 "\$node_(1) 150.0 230.0 0.0"

\$time 45.0 "\$node_(0) 340.00 70.00 0.00"

\$time 45.0 "\$node_(1) 150.0 240.0 0.0"

\$time 50.0 "\$node_(0) 370.00 70.00 0.00"

\$time 50.0 "\$node_(1) 150.0 250.0 0.0"

3.4 Visualizing node movement in NetSim animator

NetSim animator animates the flow of packets in the network along with device mobility if configured. Viewing of node movement in NetSim animator is impacted by:

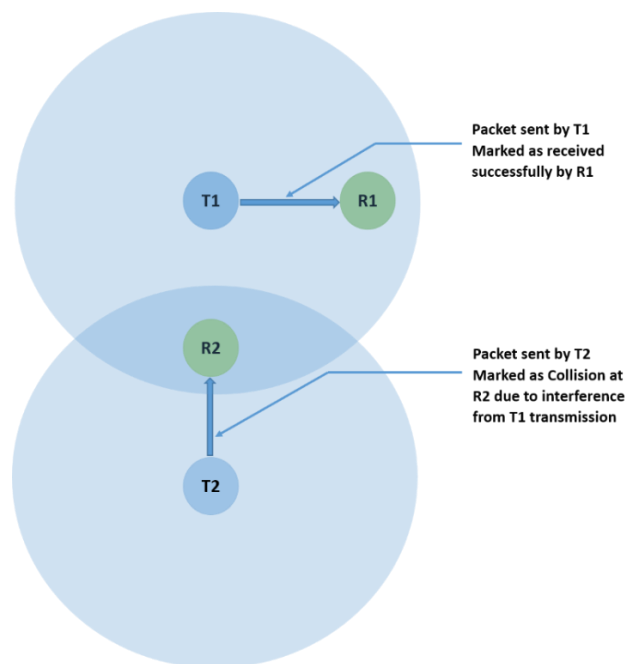
- **Time Units** - movement is based on update interval which by default is set to 1 second in the UI. In case of file based mobility, for example there are movement entries at 1.25 seconds and 1.5 seconds. However, the animator progresses time in the order of microseconds. This sometimes means that users would need to wait for a long time to view node movement.
- **Application start time** - It is generally set to 5s but can be set to other values. Packet generation will start only at the application start time
- **Grid size** - If the grid size is set in thousands of meters whereas the movement of the devices is only in tens of meters then movement may not be visible even though it is happening.
- **Relation between time set in mobility file and time set for simulation** - In case of File Based Mobility where a movements file (mobility.txt) is provided as input to NetSim, if the mobility file contains entries beyond the simulation time, only those entries that lie within the simulation start and end time will be considered for the simulation and corresponding movement can be viewed in the animation.
- **Channel characteristics** - If the devices communicating in the network aren't within the range of one another or if there is no successful data transmission in the network during the simulation, animation may not show any device movement.

3.5 How are packet collisions modelled in NetSim?

Packet Collision is a theoretical concept. In the real world, a packet is not a "physical" entity that can collide. Therefore a "collision" is the interference of two radio waves (of the packets being sent).

In NetSim the collision model is as follows:

- At the transmitter, when a packet is being transmitted, if the sum of received power from all other nodes is greater than the receiver sensitivity then that packet is marked as collided
- At the receiver, when a packet is being received, if the sum of the received power from all other nodes, except the node from where this packet is being transmitted, is greater than the receiver sensitivity then that packet is marked as collided.
- There is also a special case where a Single Packet can be marked as collided as explained below:



Where,

T1 and **T2** are transmitters

R1 and **R2** are receivers

3.6 How is packet error decided in NetSim?

The way the packet error is decided is given as follows:

- Compute Rx power at the receiver based on Tx Power, fewer losses due to Propagation. Propagation models cover HATA Urban, Sub Urban, COST 231 HATA, Indoor Home / Office, Factory, Lognormal, Friis Free space. Apart from path loss, fading and shadowing can also be modeled
- $Rx \text{ Power} / (\text{Background Noise} + \text{Interference noise}) = \text{SINR}$. Here Interference noise is due to other transmissions within the network
- Based on the SINR, BER is got from the respective modulation curve/table.
- Based on the BER, the packet error rate is decided
- NetSim does not perform bit-wise simulation since it would be too computationally intensive.

4 Featured Examples

Sample configuration files for all networks are available in Examples Menu in NetSim Home Screen. These files provide examples on how NetSim can be used – the parameters that can be changed and the typical effect it has on performance.

4.1 AODV Routing

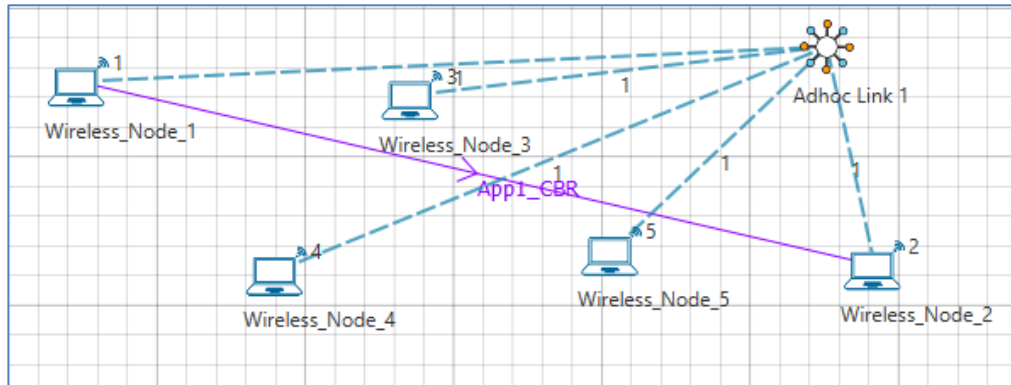
Adhoc On Demand Distance Vector Routing (AODV) protocol defines 3 message types:

- **Route Requests (RREQs)** - RREQ messages are used to initiate the route finding process.
- **Route Replies (RREPs)** - RREP messages are used to finalize the routes
- **Route Errors (RERRs)** - RERR messages are used to notify the network of a link breakage in an active route.

The route discovery process involves ROUTE REQUEST (RREQ) and ROUTE REPLY (RREP) packets. The source node initiates the route discovery process using RREQ packets. The generated route request is forwarded to the neighbors of the source node and this process is repeated till it reaches the destination. On receiving a RREQ packet, an intermediate node with route to destination or the destination node generates a RREP containing the number of hops required to reach the destination. All intermediate nodes that participates in relaying this reply to the source node creates a forward route to destination.

RERR message processing is initiated when: – Node detects a link break for the next hop of an active route, or receives a data packet destined for a node for which it has no (active) route.

Open NetSim, Select Examples->Mobile-Adhoc-Networks->MANET-AODV

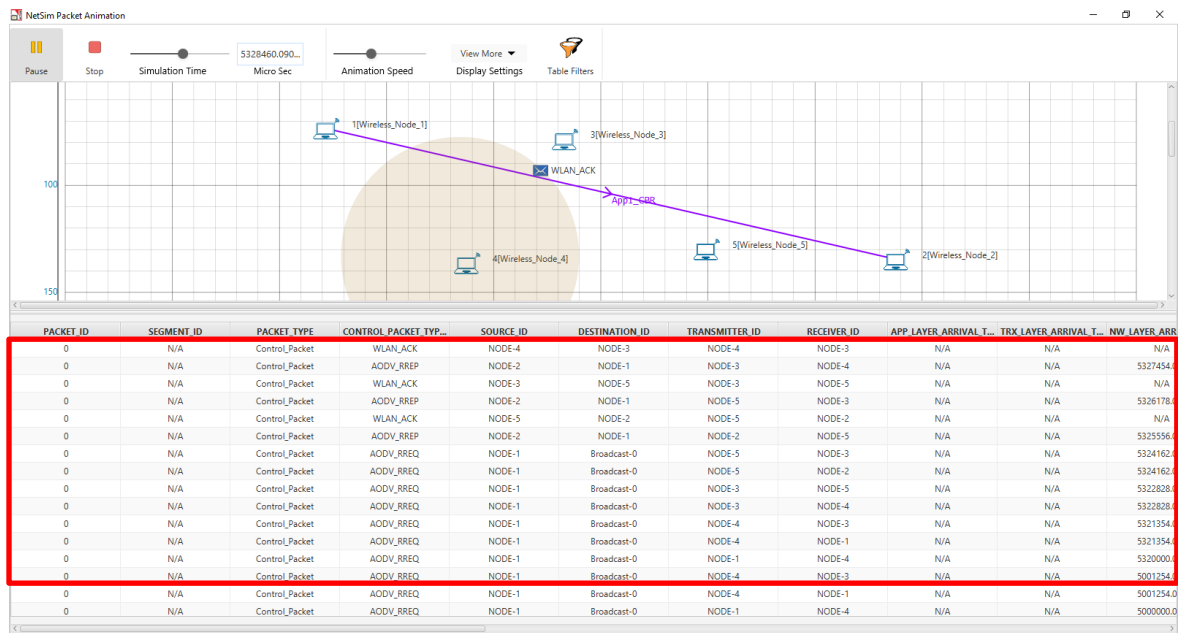


Settings done in the Network:

1. Grid length: 500m*500m
2. Disable TCP in all nodes and disable mobility in all nodes
3. Configure CBR application with default properties
4. Channel characteristics: Path loss only, Path loss model: Log Distance, Path loss exponent: 4
5. Set AODV routing protocol under network layer properties in all nodes
6. Enable Packet Trace
7. Run simulation for 10 seconds

Output:

Open Packet animation and you will be able to see that Source Node 1 initiates route discovery process using RREQ packets. The generated route request is broadcasted to the neighbours of the source node i.e. Node3 and Node4. Node4 broadcasts the RREQ packet to its neighbouring nodes i.e. Node3, Node1 since Node4 does not have route to destination Node2. Similarly, Node3 broadcasts the RREQ packet to Node5, and Node4 and then Node5 broadcasts to Node 2 and Node 3. On receiving a RREQ packet, the Destination Node2 generates a RREP packet and transmits to Node5, Node5->Node3, Node5->Node3, and Node4->Node1. Then the Source Node1 starts sending Data packets.



The same process can be observed in Packet trace file by filtering Control_Packet_Type/Application_Name to AODV_RREQ and AODV_RREP as shown below:

1	PACKET ID	SEGMENT ID	PACKET TYPE	CONTROL_PACKET_TYPE/APP_NAME	SOURCE ID	DESTINATION ID	TRANSMITTER ID	RECEIVER ID	APP_LAYER_ARRIVAL_TIME(US)	TRX_LAYER_ARRIVAL_TIME(US)
2	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-1	NODE-4	N/A	N/A
3	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-4	NODE-1	N/A	N/A
4	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-4	NODE-3	N/A	N/A
5	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-1	NODE-4	N/A	N/A
6	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-4	NODE-1	N/A	N/A
7	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-4	NODE-3	N/A	N/A
8	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-3	NODE-4	N/A	N/A
9	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-3	NODE-5	N/A	N/A
10	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-5	NODE-2	N/A	N/A
11	0	N/A	Control_Packet	AODV_RREQ	NODE-1	Broadcast-0	NODE-5	NODE-3	N/A	N/A
12	0	N/A	Control_Packet	AODV_RREP	NODE-2	NODE-1	NODE-2	NODE-5	N/A	N/A
14	0	N/A	Control_Packet	AODV_RREP	NODE-2	NODE-1	NODE-5	NODE-3	N/A	N/A
16	0	N/A	Control_Packet	AODV_RREP	NODE-2	NODE-1	NODE-3	NODE-4	N/A	N/A
18	0	N/A	Control_Packet	AODV_RREP	NODE-2	NODE-1	NODE-4	NODE-1	N/A	N/A
185	0	N/A	Control_Packet	AODV_RREP	NODE-2	Broadcast-0	NODE-2	NODE-5	N/A	N/A
425	0	N/A	Control_Packet	AODV_RREP	NODE-5	Broadcast-0	NODE-5	NODE-2	N/A	N/A
426	0	N/A	Control_Packet	AODV_RREP	NODE-5	Broadcast-0	NODE-5	NODE-3	N/A	N/A
428	0	N/A	Control_Packet	AODV_RREP	NODE-2	Broadcast-0	NODE-2	NODE-5	N/A	N/A
456	0	N/A	Control_Packet	AODV_RREP	NODE-3	Broadcast-0	NODE-3	NODE-4	N/A	N/A
457	0	N/A	Control_Packet	AODV_RREP	NODE-3	Broadcast-0	NODE-3	NODE-5	N/A	N/A
491	0	N/A	Control_Packet	AODV_RREP	NODE-1	Broadcast-0	NODE-1	NODE-4	N/A	N/A
695	0	N/A	Control_Packet	AODV_RREP	NODE-2	Broadcast-0	NODE-2	NODE-5	N/A	N/A
702	0	N/A	Control_Packet	AODV_RREP	NODE-3	Broadcast-0	NODE-3	NODE-4	N/A	N/A
703	0	N/A	Control_Packet	AODV_RREP	NODE-3	Broadcast-0	NODE-3	NODE-5	N/A	N/A
708	0	N/A	Control_Packet	AODV_RREP	NODE-5	Broadcast-0	NODE-5	NODE-2	N/A	N/A

4.2 ZRP-IARP

Zone routing protocol

The ZRP is based on two procedures:

1. IntraZone Routing Protocol (IARP) and
2. InterZone Routing Protocol (IERP).

Through the use of the IARP, each node learns the identity of and the (minimal) distance to all the nodes in its routing zone. The actual IARP is not specified and can include any number of protocols, such as the derivatives of Distance Vector Protocol (e.g., Ad Hoc On-Demand Distance Vector, Shortest Path First (e.g., OSPF). In fact, different portions of an ad hoc network may choose to operate based on different choice of the IARP protocol. Whatever the

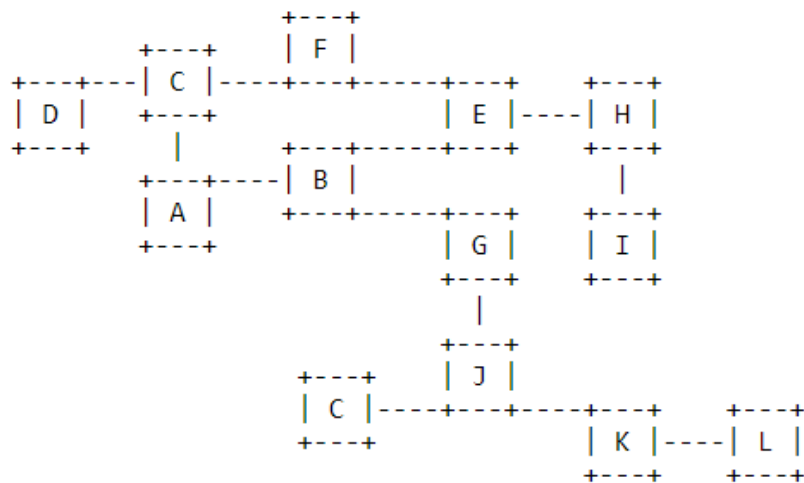
choice of IARP is, the protocol needs to be modified to ensure that the scope of this operation is restricted to the zone of the node in question.

Note that as each node needs to learn the distances to the nodes within its zone only, the nodes are updated about topological changes only within their routing zone. Consequently, in spite of the fact that a network can be quite large, the updates are only locally propagated.

IERP

While IARP finds routes within a zone, IERP is responsible for finding routes between nodes located at distances larger than the zone radius. IERP relies on bordercasting. Bordercasting is possible as any node knows the identity and the distance to all the nodes in its routing zone by the virtue of the IARP protocol.

The IERP operates as follows: The source node first checks whether the destination is within its routing zone. (Again, this is possible as every node knows the content of its zone). If so, the path to the destination is known and no further route discovery processing is required. If, on the other hand, the destination is not within the source's routing zone, the source bordercasts a route request (referred to here as a "request") to all its peripheral nodes. Now, in turn, all the peripheral nodes execute the same algorithm: check whether the destination is within their zone. If so, a route reply (referred to here as a "reply") is sent back to the source indicating the route to the destination. If not, the peripheral node forwards the query to its peripheral nodes, which, in turn, execute the same procedure. An example of this Route Discovery procedure is demonstrated in the figure below.

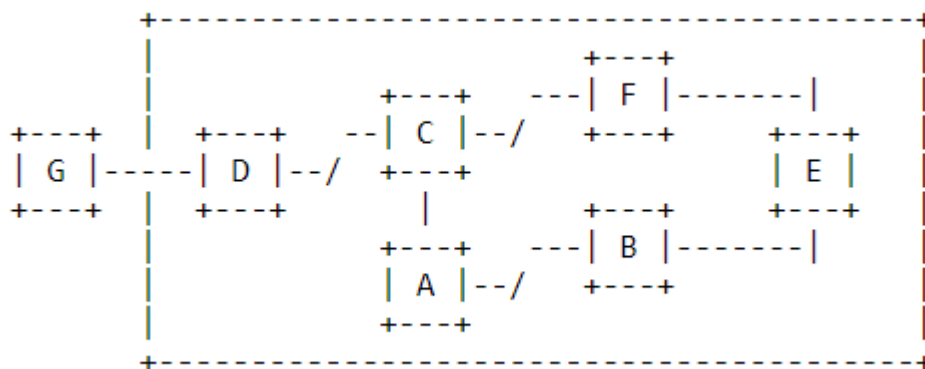


The node A has a datagram to node L. Assume routing zone radius of 2. Since L is not in A's routing zone (which includes B,C,D,E,F,G), A bordercasts a routing request to its peripheral nodes: D,F,E, and G. Each one of these peripheral nodes check whether L exists in their routing zones. Since L is not found in any routing zones of these nodes, the nodes bordercast the request to their peripheral nodes. In particular, G bordercasts to K, which realizes that L is in its routing zone and returns the requested route (L-K-G-A) to the query source, namely A.

IARP

In Zone Routing, the Intrazone Routing Protocol (IARP) proactively maintains routes to destinations within a local neighborhood, which we refer to as a routing zone. More precisely, a node's routing zone is defined as a collection of nodes whose minimum distance in hops from the node in question is no greater than a parameter referred to as the zone radius. Note that each node maintains its own routing zone. An important consequence is that the routing zones of neighboring nodes overlap.

An example of a routing zone (for node A) of radius 2 is shown below.



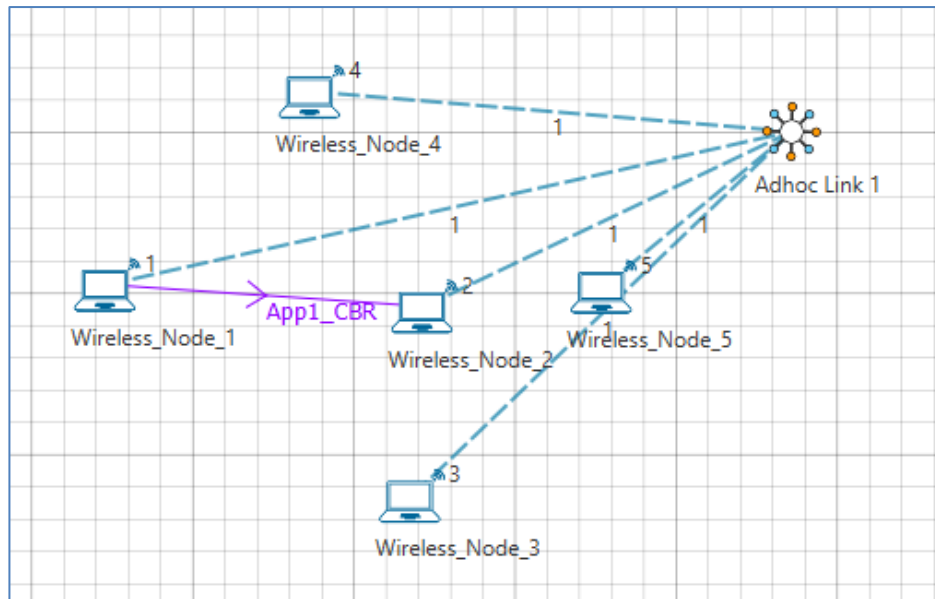
Note that in this example nodes B through F are within the routing zone of A. Node G is outside A's routing zone. Also note that E can be reached by two paths from A, one with length 2 hops and one with length 3 hops. Since the minimum is less or equal than 2, E is within A's routing zone.

Peripheral nodes are nodes whose minimum distance to the node in question is equal exactly to the zone radius. Thus, in the above figure, nodes D, F, and E are A's peripheral nodes.

The construction of a routing zone requires a node to first know who its neighbors are. A neighbor is defined as a node with whom direct (point-to-point) communication can be established and is, thus, one hop away. Identification of a node's neighbors may be provided directly by the media access control (MAC) protocols, as in the case of polling-based protocols. In other cases, neighbor discovery may be implemented through a separate **Neighbor Discovery Protocol (NDP)**. Such a protocol typically operates through the periodic broadcasting of "hello" beacons. The reception (or quality of reception) of a "hello" beacon can be used to indicate the status of a connection to the beaconing neighbor.

Neighbor discovery information is used as a basis for the IARP. IARP can be derived from globally proactive link state routing protocols that provide a complete view of network connectivity.

Open NetSim, Select Examples->Mobile-Adhoc-Networks->ZRP-IARP



Settings done in the Network:

1. Grid length: 500m*500m
2. Disable TCP in all nodes and disable mobility in all nodes
3. Configure CBR application with default properties
4. Channel characteristics: Path loss only, Path loss model: Log Distance, Path loss exponent: 3
5. Set ZRP routing protocol under network layer properties in all nodes
6. Enable Packet Trace
7. Run simulation for 10 seconds

Output: Open Packet trace and observe Control_Packet_Type/App_Name

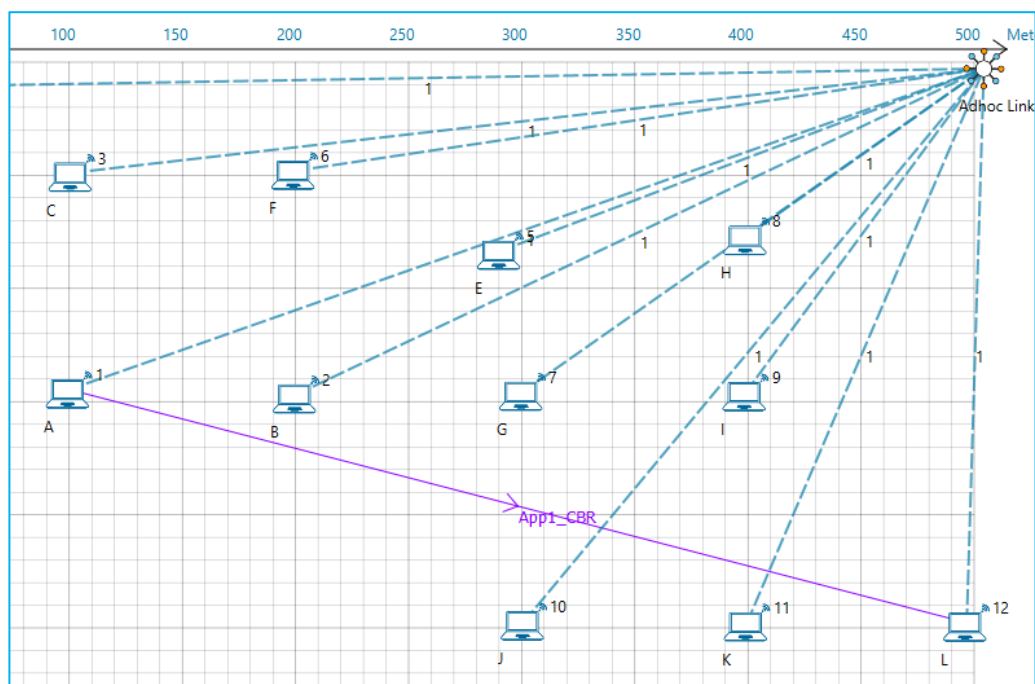
Packet Trace - Excel (Product Activation Failed)

PACKET_ID	SEGMENT_ID	PACKET_TYPE	CONTROL_PACKET_TYPE/APP_NAME	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID
1	0	N/A	Control_Pe	NODE-4	Broadcast-0	NODE-4	NODE-1
2	0	N/A	Control_Pe	NODE-4	Broadcast-0	NODE-4	NODE-2
3	0	N/A	Control_Pe	NODE-4	Broadcast-0	NODE-4	NODE-3
4	0	N/A	Control_Pe	NODE-4	Broadcast-0	NODE-4	NODE-5
5	0	N/A	Control_Pe	NODE-2	Broadcast-0	NODE-2	NODE-1
6	0	N/A	Control_Pe	NODE-2	Broadcast-0	NODE-2	NODE-3
7	0	N/A	Control_Pe	NODE-2	Broadcast-0	NODE-2	NODE-4
8	0	N/A	Control_Pe	NODE-2	Broadcast-0	NODE-2	NODE-5
9	0	N/A	Control_Pe	NODE-3	Broadcast-0	NODE-3	NODE-1
10	0	N/A	Control_Pe	NODE-3	Broadcast-0	NODE-3	NODE-2
11	0	N/A	Control_Pe	NODE-3	Broadcast-0	NODE-3	NODE-4
12	0	N/A	Control_Pe	NODE-3	Broadcast-0	NODE-3	NODE-5
13	0	N/A	Control_Pe	NODE-1	Broadcast-0	NODE-1	NODE-2
14	0	N/A	Control_Pe	NODE-1	Broadcast-0	NODE-1	NODE-3
15	0	N/A	Control_Pe	NODE-1	Broadcast-0	NODE-1	NODE-4
16	0	N/A	Control_Pe	NODE-1	Broadcast-0	NODE-1	NODE-5
17	0	N/A	Control_Pe	NODE-5	Broadcast-0	NODE-5	NODE-1
18	0	N/A	Control_Pe	NODE-5	Broadcast-0	NODE-5	NODE-2
19	0	N/A	Control_Pe	NODE-5	Broadcast-0	NODE-5	NODE-3
20	0	N/A	Control_Pe	NODE-5	Broadcast-0	NODE-5	NODE-4
21	0	N/A	Control_Pe	NODE-5	Broadcast-0	NODE-5	NODE-5
22	0	N/A	Control_Packet	NODE-5	Broadcast-0	NODE-5	NODE-1
23	0	N/A	Control_Packet	NODE-5	Broadcast-0	NODE-5	NODE-2

Users can see only NDP_Hello_Message since the destination node is within the zone. The ZRP framework proactively maintains local routing information (routing zones) based on periodic exchanges of neighbor discovery messages.

4.3 ZRP-IERP

Open NetSim, Select Examples->Mobile-Adhoc-Networks->ZRP-IERP



Settings done in the Network:

1. Grid length: 500m*500m
2. Disable TCP in all nodes and disable mobility in all nodes
3. Configure CBR application with default properties
4. Channel characteristics: Path loss only, Path loss model: Log Distance, Path loss exponent: 3.3
5. Set ZRP routing protocol under network layer properties in all nodes
6. Enable Packet Trace
7. Run simulation for 10 seconds

Output:

Open Packet trace and filter Control_Packet_Type/App_Name to IERP_Route_Request

1	PACKET_ID	SEGMENT_ID	PACKET_TYPE	CONTROL_PACKET_TYPE/APP_NAME	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID
502	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-8	NODE-1	NODE-6
508	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-10	NODE-1	NODE-2
514	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-8	NODE-1	NODE-6
518	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-10	NODE-1	NODE-2
520	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-9	NODE-1	NODE-5
524	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-11	NODE-1	NODE-5
530	0	N/A	Control_Packet	IERP_ROUTE_REQUEST	NODE-1	NODE-9	NODE-1	NODE-5

In the above figure, Node-1 is sending IERP_Route_Request packets to its peripheral nodes 8, 9, 10 and 11 since the destination is not present in Node1's zone. Similarly, Node4 transmits IERP_Route_Request packets to its peripheral nodes 2 and 6 and so on. Node7 sends IERP_Route_Reply packet in response to the IERP_Route_Request. And the data packet routes through Node1->Node2->Node10->Node12. This can be observed in Packet animation also.

4.4 MANET-OLSR

4.4.1 Optimized Link State Routing Protocol

OLSR is developed for mobile ad hoc networks. It operates as a table driven, proactive protocol, i.e., exchanges topology information with other nodes of the network regularly. Each node selects a set of its neighbor nodes as "multipoint relays" (MPR). In OLSR, only nodes, selected as such MPRs, are responsible for forwarding control traffic, intended for diffusion into the entire network. MPRs provide an efficient mechanism for flooding control traffic by reducing the number of transmissions required.

Nodes, selected as MPRs, also have a special responsibility when declaring link state information in the network. Indeed, the only requirement for OLSR to provide shortest path routes to all destinations is that MPR nodes declare link-state information for their MPR selectors. Additional available link-state information may be utilized, e.g., for redundancy.

Nodes which have been selected as multipoint relays by some neighbor node(s) announce this information periodically in their control messages. Thereby a node announces to the network, that it has reachability to the nodes which have selected it as an MPR. In route calculation, the MPRs are used to form the route from a given node to any destination in the network. Furthermore, the protocol uses the MPRs to facilitate efficient flooding of control messages in the network.

4.4.2 Neighbor detection

Neighbor detection populates the neighborhood information base and concerns itself with nodes and node main addresses. The mechanism for neighbor detection is the periodic exchange of HELLO messages.

A node maintains a set of neighbor tuples, based on the link tuples. This information is updated according to changes in the Link Set. The Link Set keeps the information about the links, while the Neighbor Set keeps the information about the neighbors. There is a clear association between those two sets, since a node is a neighbor of another node if and only if there is at least one link between the two nodes.

The "Originator Address" of a HELLO message is the main address of the node, which has emitted the message. Upon receiving a HELLO message, a node SHOULD first update its Link Set and then update its Neighbor Set

4.4.3 Topology discovery

The link sensing and neighbor detection part of the protocol basically offers, to each node, a list of neighbors with which it can communicate directly and, in combination with the Packet Format and Forwarding part, an optimized flooding mechanism through MPRs. Based on this, topology information is disseminated through the network.

4.4.4 TC message

In order to build the topology information base, each node, which has been selected as MPR, broadcasts Topology Control (TC) messages. TC messages are flooded to all nodes in the network and take advantage of MPRs. MPRs enable a better scalability in the distribution of topology information.

The list of addresses can be partial in each TC message (e.g., due to message size limitations, imposed by the network), but parsing of all TC messages describing the advertised link set of a node MUST be complete within a certain refreshing period (TC_INTERVAL). The information diffused in the network by these TC messages will help each node calculate its routing table.

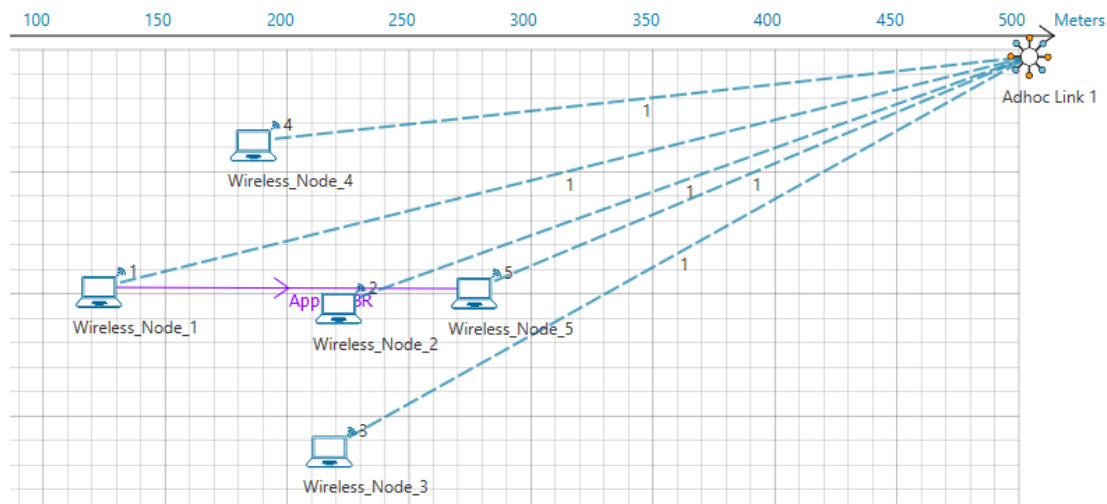
When the advertised link set of a node becomes empty, this node SHOULD still send (empty) TC-messages during the duration equal to the "validity time" (typically, this will be equal to TOP_HOLD_TIME) of its previously emitted TC-messages, in order to invalidate the previous

TC-messages. It SHOULD then stop sending TC-messages until some node is inserted in its advertised link set.

A node MAY transmit additional TC-messages to increase its reactivity to link failures. When a change to the MPR selector set is detected and this change can be attributed to a link failure, a TC-message SHOULD be transmitted after an interval shorter than TC_INTERVAL.

Example:

Open NetSim, Select Examples->Mobile-Adhoc-Networks->MANET-OLSR



Settings done in the Network:

1. Grid length: 500m*500m
2. Disable TCP in all nodes and disable mobility in all nodes
3. Configure CBR application with default properties
4. Channel characteristics: Path loss only, Path loss model: Log Distance, Path loss exponent: 3
5. Set OLSR routing protocol under network layer properties in all nodes
6. Enable Packet Trace
7. Run simulation for 30 seconds

Output:

Open Packet trace and filter Control_Packet_Type/App_Name to NDP_HELLO_MESSAGE, Source_ID and Transmitter_ID to Node-1. All the nodes in the network exchange the NDP_HELLO_MESSAGES periodically (for every 2 seconds- check NETWORK_LAYER_ARRIVAL_TIME column) to detect the neighbours. In the below screenshot, Node-1 is broadcasting NDP_HELLO_MESSAGES to its neighbours 2, 3, 4 and 5 for every 2 seconds

	PACKET_ID	CONTROL_PACKET_TYPE/App	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID	NW_LAYER_ARRIVAL TIME(US)	MAC_U
30	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	1719373	
31	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	1719373	
32	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	1719373	
33	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	1719373	
54	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	3642780	
55	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	3642780	
56	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	3642780	
57	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	3642780	
238	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	5589414	
239	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	5589414	
240	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	5589414	
241	0	NDP_HELLO_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	5589414	

Now filter Control_Packet_Type/App_Name to OLSR_TC_MESSAGE, Source_ID and Transmitter_ID to Node-1. All nodes in the network broadcasts Topology Control (TC) messages (for every 5 seconds - check NETWORK_LAYER_ARRIVAL_TIME column) in order to build the topology information base. In the below screenshot, Node-1 is broadcasting OLSR_TC_MESSAGE to its neighbours 2, 3, 4 and 5 for every 5 seconds

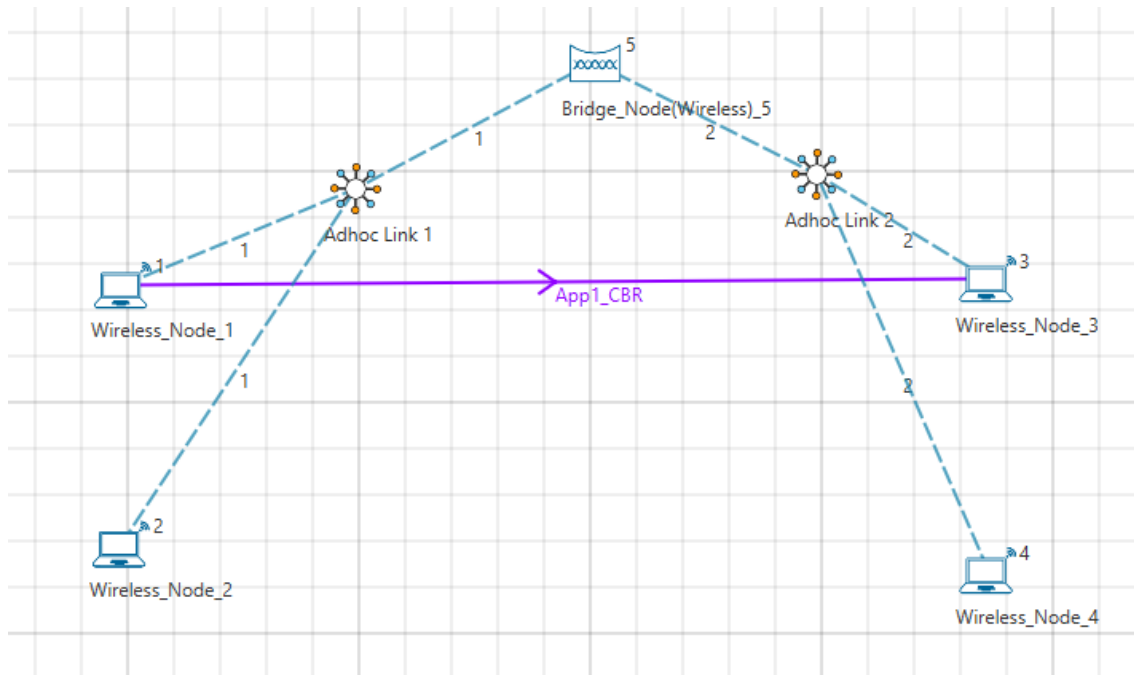
	PACKET_ID	CONTROL_PACKET_TYPE/App	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID	NW_LAYER_ARRIVAL TIME(US)	MAC
78	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	4633567	
79	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	4633567	
80	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	4633567	
81	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	4633567	
720	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	9573939	
721	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	9573939	
722	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	9573939	
723	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	9573939	
1296	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	14167133	
1297	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	14167133	
1298	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	14167133	
1299	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	14167133	
1916	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-2	18883077	
1917	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-3	18883077	
1918	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-4	18883077	
1919	0	OLSR_TC_MESSAGE	NODE-1	Broadcast-0	NODE-1	NODE-5	18883077	

4.5 Multiple MANETS

Example:

Open NetSim, Select Examples->Mobile-Adhoc-Networks->Multiple-MANETs

Multiple MANETs allows users to interconnect two or more MANETs using a bridge node. Click and drop Wireless nodes, Adhoc links and Bridge Node onto the grid environment and connect Wireless Nodes to Adhoc links to form two different MANET's using Adhoc links. Further connect the two MANET's using a bridge node as shown below:



Settings done in the Network:

1. Grid length: 500m*500m
2. Disable TCP in all nodes
3. Configure CBR application from Node1 to Node3 with default properties
4. Enable Packet Trace
5. Run simulation for 10 seconds

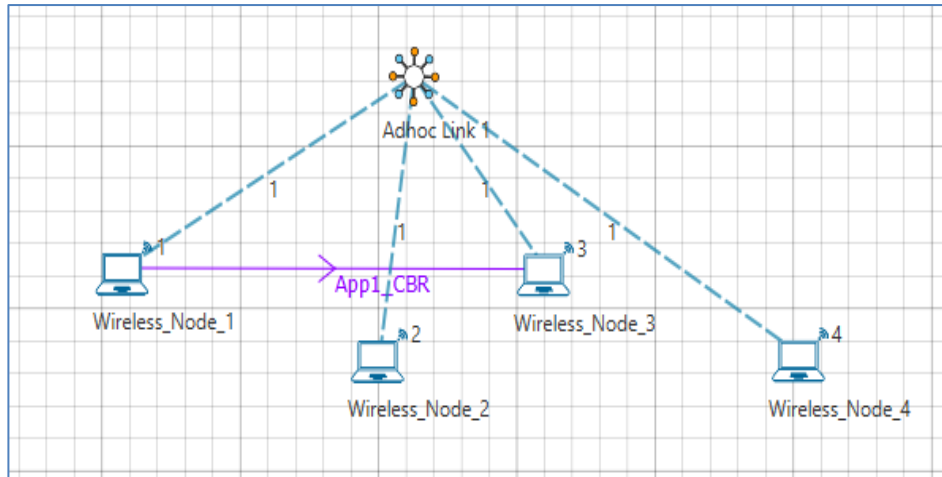
Output:

After simulation, open packet trace and filter CONTROL_PACKET_TYPE//APP_NAME to App1_CBR. In the below screenshot, users can observe that the packets from Source Node-1 are going to Destination Node-3 via Bridge Node-5.

PACKET_ID	SEGMENT_ID	PACKET_TYPE	CONTROL_PACKET_TYPE/APP_NAME	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID	APP_LAYER
1	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
1	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-5	NODE-3	
2	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
2	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-5	NODE-3	
3	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
3	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-5	NODE-3	
4	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
4	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-5	NODE-3	
5	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
5	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-5	NODE-3	
6	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
6	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-1	NODE-5	
6	0	CBR	App1_CBR	NODE-1	NODE-3	NODE-5	NODE-3	

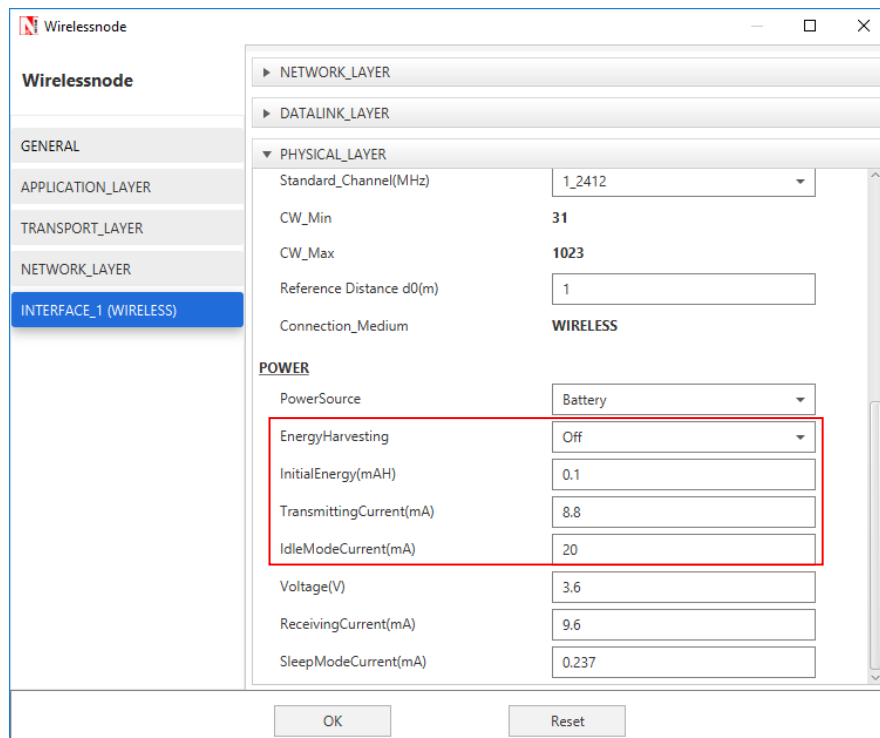
4.6 MANET- Battery lifetime analysis

Open NetSim, Select Examples-> MANET-> Battery-Lifetime-Analysis



Settings done in sample network:

1. Grid length → 500m*500m
2. Open Wireless node Interface properties and set the following properties under Physical Layer as shown below
 - a. Idle Mode Current (mA) = 20,
 - b. Initial Energy (mAH) = 0.1 and
 - c. Energy harvesting to Off



3. Set application start time as 0s and accept default for other parameters
4. Enable Packet Trace

5. Run simulation for 50 secs

Theoretical calculation for battery life:

The battery life can be calculated by using the formula below:

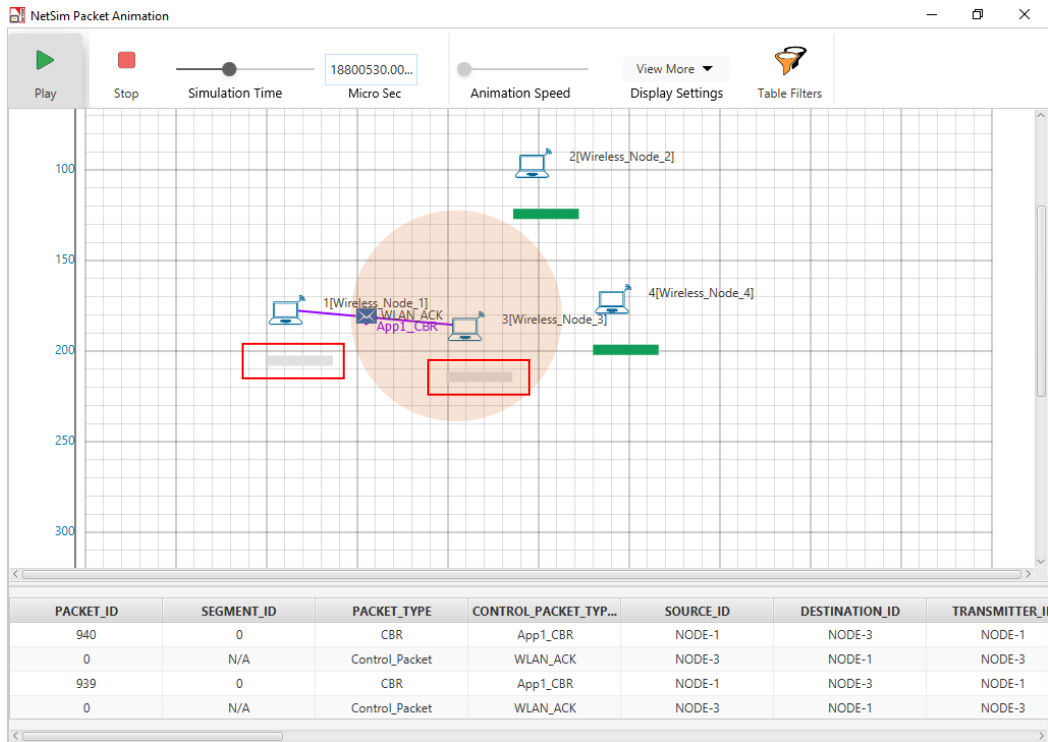
$$\text{Battery life(Hours)} = \frac{\text{Energy(mAH)}}{\text{Load current(mA)}}$$
$$= \frac{0.1}{20} \text{ Hours} = 0.005 \text{ Hours or 18 secs}$$

Output:

After simulation, open Battery model metrics and observe the consumed energy would be approximately equal to initial energy for Wireless node_1 and Wireless node_3 since these nodes are communicating with each other and hence there is battery consumption. We can also observe that the Remaining energy is approximately 0 for Wireless node_1 and Wireless node_3. i.e., the battery is completely drained

Battery model_Table							
Battery model		☒ Detailed View					
Device Name	Initial energy(mJ)	Consumed energy(mJ)	Remaining Energy(mJ)	Transmitting energy(mJ)	Receiving energy(mJ)	Idle energy(mJ)	Sleep energy(mJ)
WIRELESS_NODE_1	1296.000000	1294.661386	1.338614	39.128014	9.895130	1245.638243	0.000000
WIRELESS_NODE_2	1296.000000	0.063262	1295.936738	0.000000	0.026542	0.036720	0.000000
WIRELESS_NODE_3	1296.000000	1296.000000	0.000000	9.070196	42.639832	1244.289972	0.000000
WIRELESS_NODE_4	1296.000000	0.063262	1295.936738	0.000000	0.026542	0.036720	0.000000

Open Packet animation and check the battery level of wireless nodes 1 and 3 would reduce with time and packet animation stops at 18.8 seconds approximately since the batteries of wireless nodes 1 and 3 dies as shown in the below screenshot:



4.7 MANET File Based Placement

File based placement option can be used to place devices in user defined locations. Create a text file as per the following or use the file present in the Docs folder of NetSim Install Directory < C:\Program Files\NetSim Standard\Docs\Sample_Configuration\MANET>

MANET_File_based_placement.txt

Wireless_Node,WIRELESSNODE,100,150

Wireless_Node,WIRELESSNODE,150,100

Wireless_Node,WIRELESSNODE,100,100

Wireless_Node,WIRELESSNODE,50,50

Open NetSim, in the Main menu select **New Simulation→Mobile Adhoc networks**. Select File Based Placement option under Automatic Placement and give the path of the text file as shown below:

Grid Settings and Device Placement

Total Grid Settings

Grid Length (m)

Grid Length should be between 50 m to 1,000,000 m

Device Grid Settings

Side Length (m)

Side length should be a multiple of 50 but less than or equal to Grid Length. This is a sub-grid in which the devices would be placed automatically

Device Placement Strategy

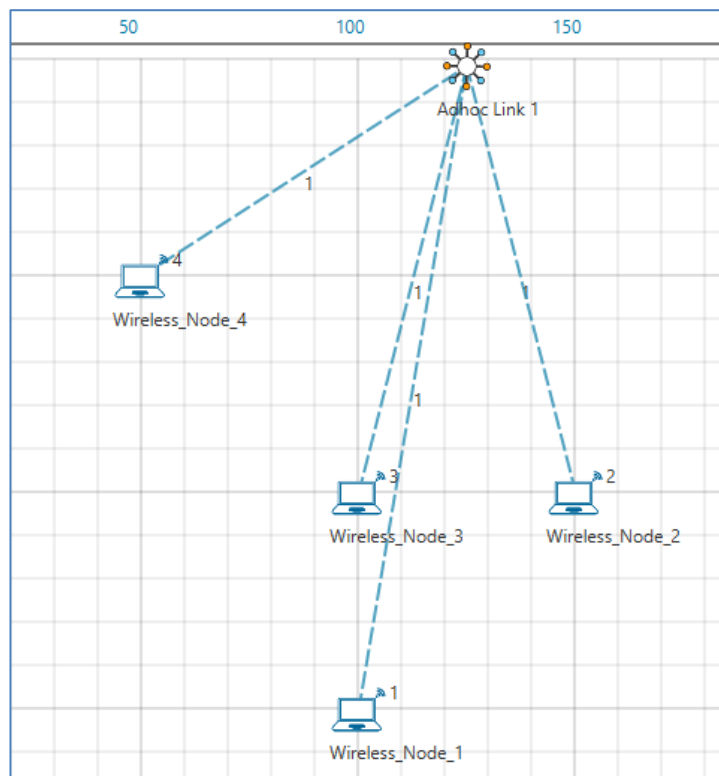
☒ Automatic Placement
☐ Uniform Placement
☐ Random Placement
☒ File Based Placement



☐ Manually Via Click and Drop

OK Cancel

After giving the path, Click on OK will display the MANET network shown below, where all devices are placed as per the positions given in the text file.



Please refer section “2.1 Fast configuration” for more information

5 Reference Documents

IEEE 802.11 - 2012 Standard for Wireless LAN

OLSR - <https://tools.ietf.org/html/rfc3626#section-5>

ZRP - <https://tools.ietf.org/html/draft-ietf-manet-zone-zrp-04#section-1>

ZRP - <https://tools.ietf.org/html/draft-ietf-manet-zone-zrp-00>

6 Latest FAQs

Up to date FAQs on NetSim's MANET library is available at <https://tetcos.freshdesk.com/support/solutions/folders/14000110331>