Report on Network Simulation using ns3

1705003, Mahdi Hasnat Siyam

Description of Network Topology :

According to my student id, I had to simulate two network topologies. Description is given below.

High rate wireless mobile network:

In this scenario every node is equipped with a wifi network device. Each wifi device maintains IEEE 802.11b standard. Every node operates as AdhocWifi mode. To maintain connection within a range, I have installed RangePropagationLossModel in the wifi channel.

In the network layer, Internet protocol version 4 is used. Ad-hoc On demand Distance Vector (AODV) routing protocol is installed in the network layer.
In the application layer, UdpEchoServer and UdpEchoClient are used for generation of packets. Client and server applications are bound to ports such that there is a specific number of flow generated in the simulation.

Each node is equipped with a RandomWayPointMobility model such that every node can move to a random direction at defined speed and is bound to a fixed rectangle. Nodes are initially allocated a random position in the fixed rectangle

Low rate wireless mobile network:

Here every node has a device that supports low rate wireless personal area communication. Each device maintains IEEE 802.15.4 standard. In the network layer I have used Internet protocol version 6. So I used 6LoPan network devices as an adapter (for supporting Ipv6 communication on traditional LrWpan devices).

In the Application layer I have used the same method as earlier. For mobility and position allocation the previous mobility model is used.

Variation of Parameters

- Number of node

  This is achieved by creating a desired number of node at the beginning of the simulation.
- Number of flows

  A flow consists of 4 tuple (sender ip, sender port, receiver ip, receiver port)

Half the number of echo client , echo server pairs is created. So when a client sends a packet to the server it is recognised as a flow,on the other hand if the server replies back to the client it is also considered a different flow. By this way desired number of flow is generated

- Number of packet per second

  In my implementation packets are generated by application. Let's say there are *k* applications, each generates packets after *t* time interval, so packet rate is *k*/*t*. Now given the packet rate, we can set *t* equal to *k* /packet rate.

- Speed of Node

  Desired speed of the node is set in the mobility model.

Overview of Proposed Algorithm:

Temporally-Ordered Routing Algorithm (TORA) is a distributed routing protocol for mobile wireless networks.

Properties of TORA:

- Reactive protocol
- Source Initiated
- Highly adaptive to topological change
- Loop free Multi path Routing

How TORA Works?

Prerequisites:

- Global synchronized timer
- Link aware protocol on lower layer (tora needs to know active/inactive status of each link to neighbours)
- Unique Id assigned to every node.

First of all, this algorithm is described for a single destination. So multiple processes are run for each destination in the network. Each node is assigned a unique id. In my implementation I have assumed Ip address as unique for each node.

Each node other than the destination will maintain a *height* data structure. Height consists of 5 tuples namely (τ, oid , r ,δ , i). Here τ is time , oid is Id from which this height is known, r is reference bit, δ is defined such that nodes can be ordered from destination, i is the unique id of that node.

There are three phases of this algorithm.

- Route Creation:

  When a route to a destination is needed, the source node sends a *Query* packet to neighboring nodes, these *Query* packets are further flooded to other nodes.

When a node has a route to the destination node and receives a *Query* packet, then that packet floods back with an *Update* packet containing height information of that node. In this way the route to destination is created. Each node compares height between that node and neighboring nodes, link between them is directed from higher height to lower height.
So data packets are sent from higher height nodes to lower height nodes. Height of the destination is zero (0, 0, 0, 0, destination_id)

- Route Maintenance:
  Let's say there are three available downward link to the neighbour node. Now if any of the link become inactive, tora takes no action. Because there are still two downward link to reach destination. But If all the downward link becomes inactive, tora takes action upon condition of current height and height of other neighbours. There are 5 case described in the paper

- Route Erasing:
  If there are no way to re-orient link to the destination, tora removes all link with the help of *Clear* packet.

My Implementation of Task-A:

- Simulation code for high rate wireless network is available in `manat.cc`
- Simulation code for high rate wireless network is available in `low-rate-network.cc`

My Implementation of TORA in ns3:

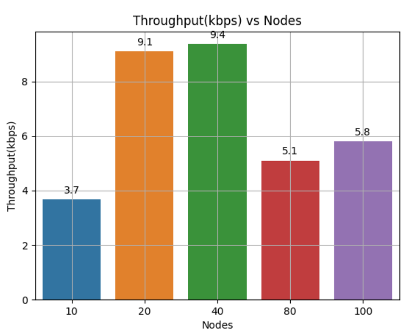
- I created `tora` module in the contrib folder.
- For establishment of link awar protocol I added Hello packet defined in tora-packet.h
- When a route to unknown destination is required, A QRY packet is broadcasted from that node and data packet is queued for future sending.
- Added tora-rqueue for queueing deferred packet
- Completed Qry Packet processing stage
- Tested code for two nodes
- Added Upd Packet Processing(all 5 case)
- Added clear packet

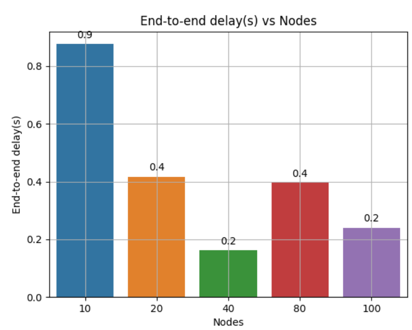
<u>Results & Explanation of Task-A:</u>

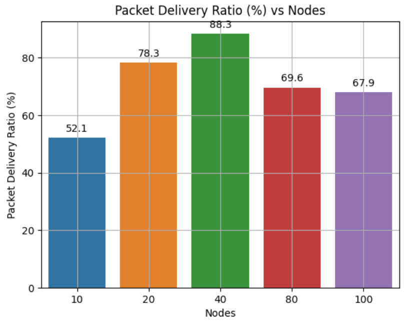
Results for varying node count in High rate wireless network:

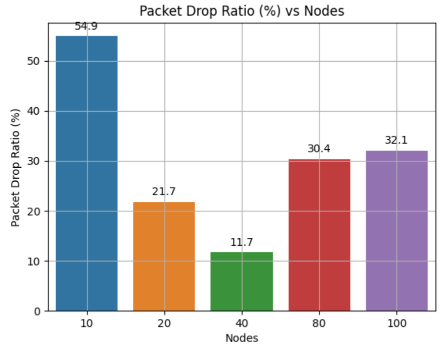
Parameters in this scenario:

Total flows added = 20
Node speed = 10 m/s
Area 400 m x 600 m
Packet Rate =10 packet / s
Maximum packet for each echo server = 5
SimulationTime = 100 s
Routing Algorithm = AODV







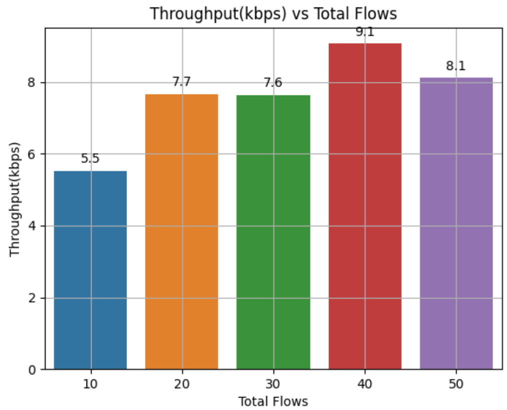


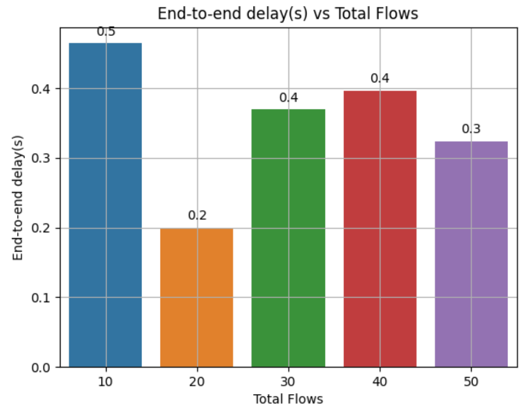
Findings: Here all nodes are distributed within region of same area. So when we increase node number, nodes become densely connected. So There will be better ways to transfer packets from source node to destination node. However for node count 80 and 100, we can see a performance drop. This may occur because of too many nodes congested within small space.

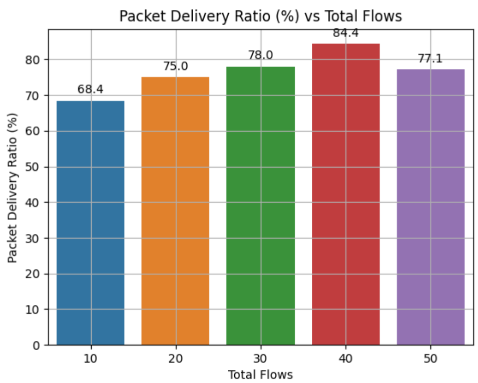
Results for varying flow count in High rate wireless network:

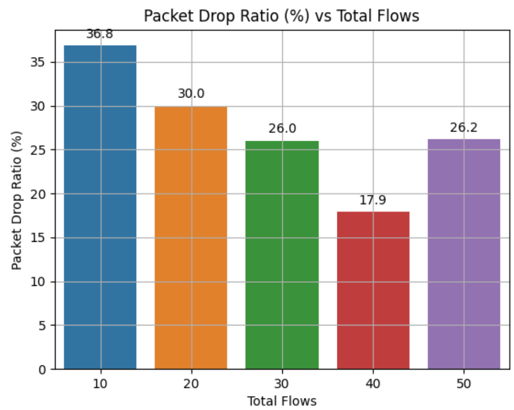
Parameters in this scenario:

Total Node = 20
Node speed = 3m/s
Area 400 m x 600 m
Packet Rate =10 packet / s
Maximum packet for each echo server = 5
SimulationTime = 100 s
Routing Algorithm = AODV









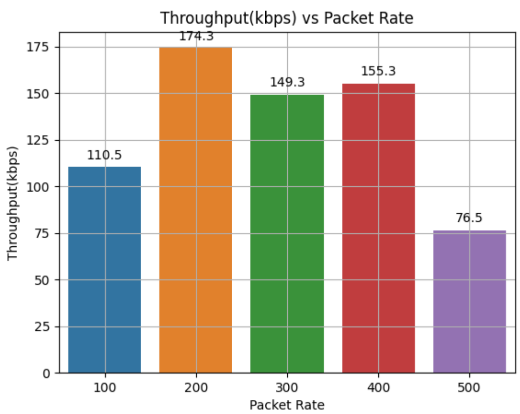
Here packet drop ratio tends to go low with increase in total flows.

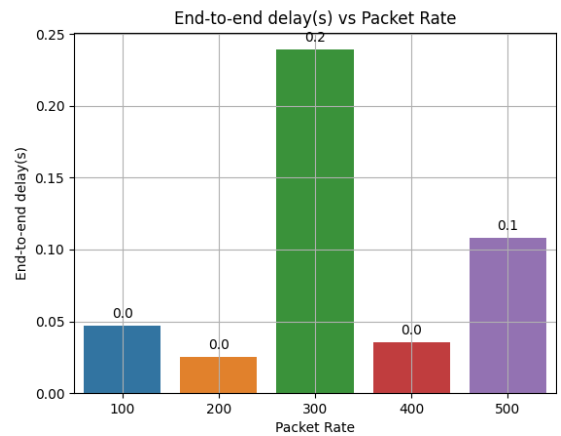
Results for varying packet rate in High rate wireless network:

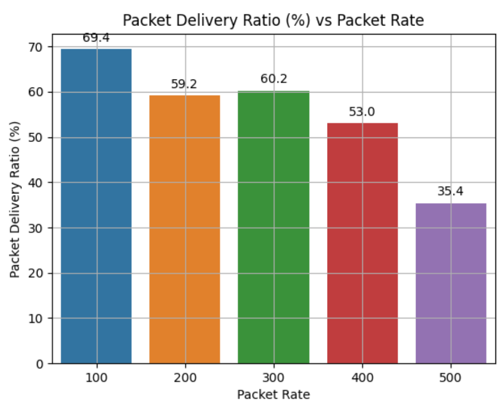
Parameters in this scenario:

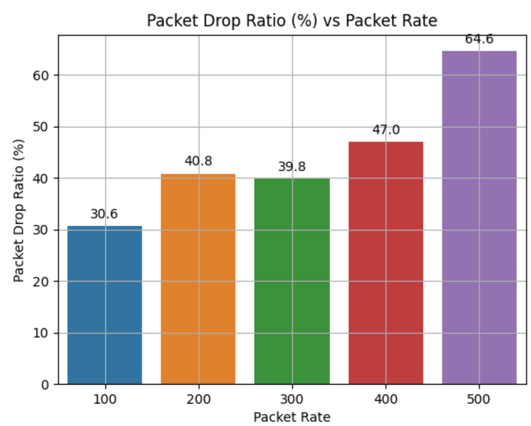
Total node = 20
Total number of flows = 20
Node speed = 3m/s
Area = 400 m x 600 m
Maximum packet for each echo server = 100
SimulationTime = 100 s

Routing Algorithm = AODV









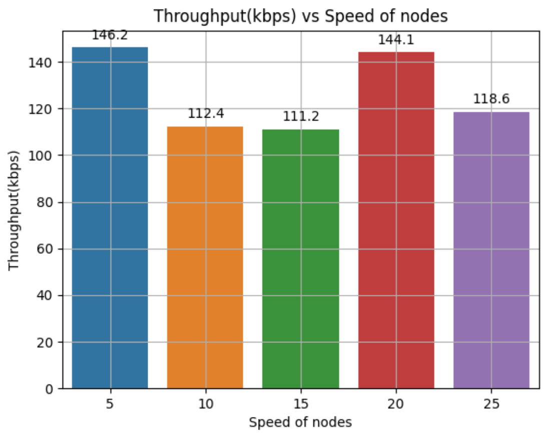
As expected, increase in packet also increases packet drop ratio due to heavy congestion.

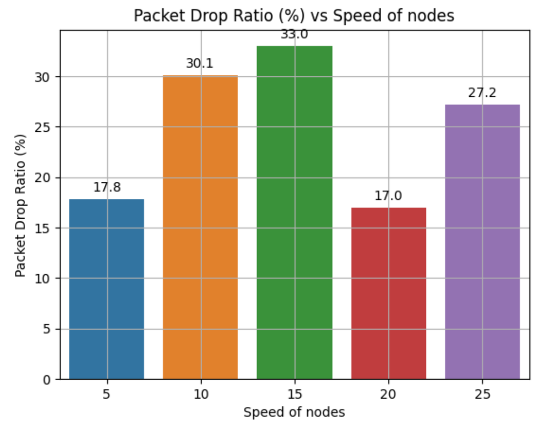
Results for varying node speed in High rate wireless network:

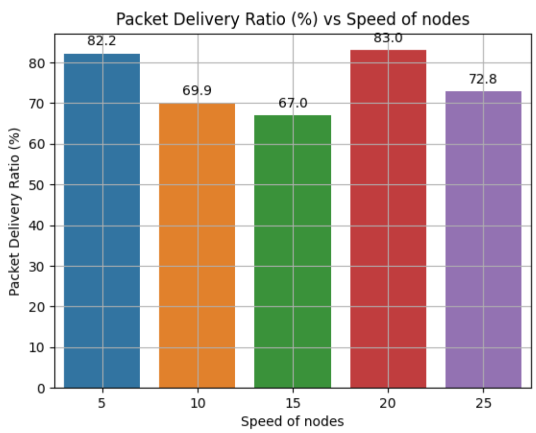
Parameters in this scenario:

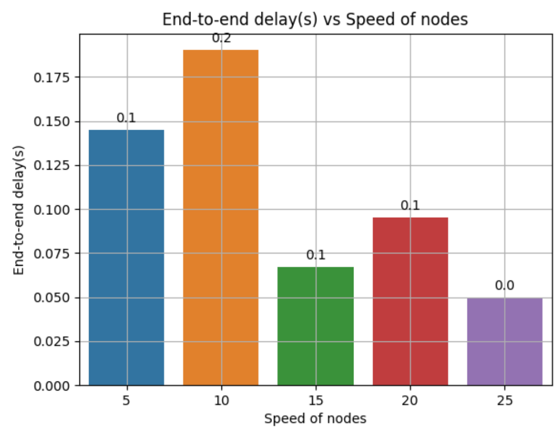
Total node = 20
Total number of flows = 20
Area = 400 m x 600 m
Packet Rate = 100 packet per sec
Maximum packet for each echo server = 100
SimulationTime = 300 s

Routing Algorithm = AODV







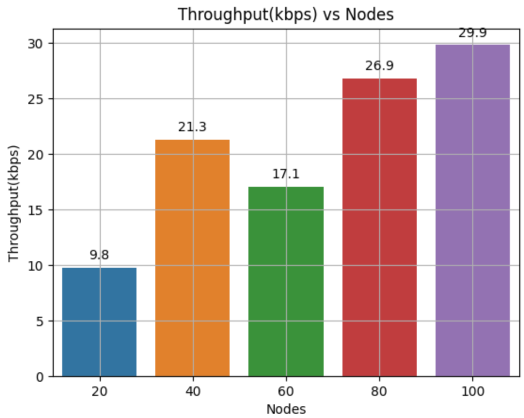


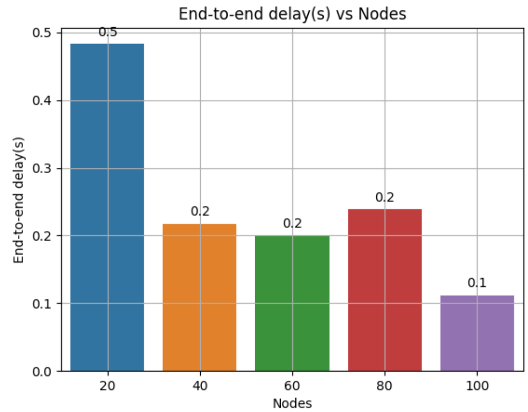
Increase in speed generally increases packet drop ratio, because due to instability of topology packet may not reach destination. But in this experiment nodes can move inside a bounded rectangular area. So destination nodes are bounced back to source nodes. That's why there are low packet drop ratio during relatively higher node speed.

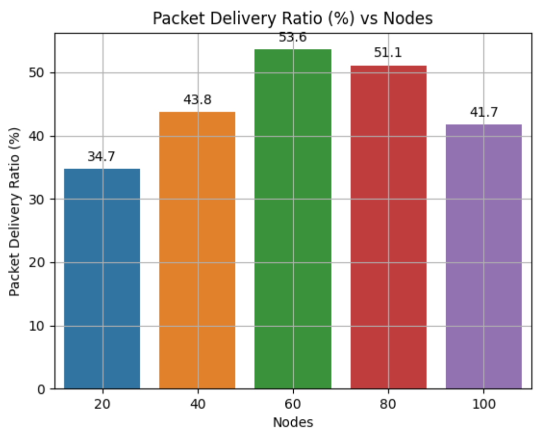
Results for varying node count in Low rate wireless network:

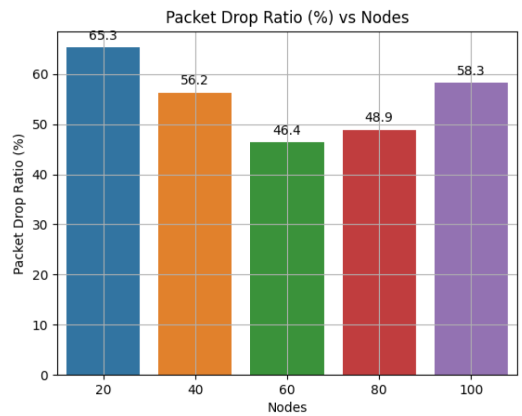
Parameters in this scenario:

Total flows added = 20
Node speed = 2 m/s
Area 100 m x 200 m
Packet Rate =100 packet / s
Maximum packet for each echo server = 10
SimulationTime = 150 s







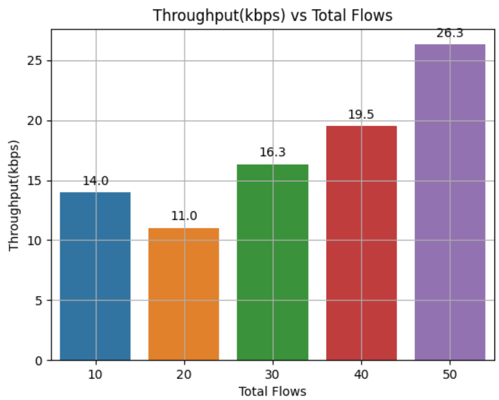


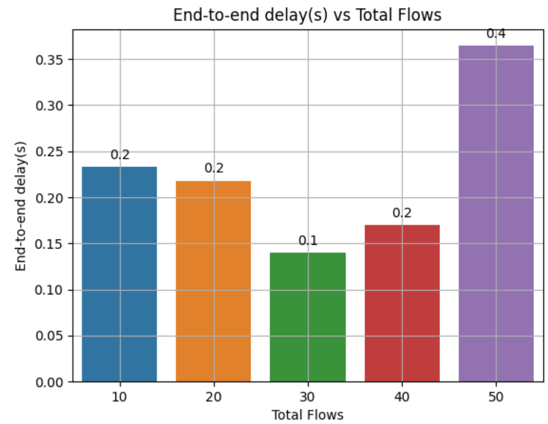
Generally an increased number of nodes within a fixed area should increase better packet delivery. Because nodes will be closely located. Thats why packet drop ratio decreases in first three simulation. But too much crowded node may infer rapid change in topology for which drop ratio increases.
Higher number of nodes ensure faster transmission (dense topology, so almost straight path possible)

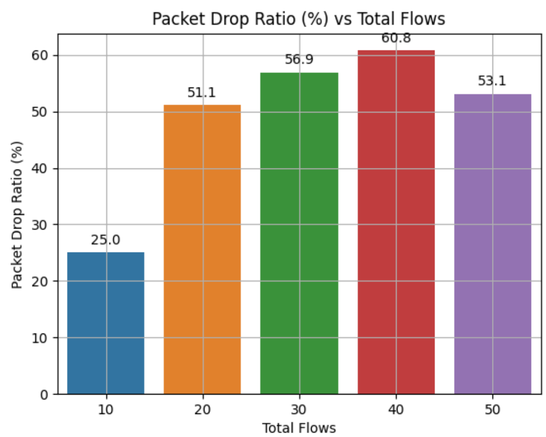
Results for varying flow count in Low rate wireless network:

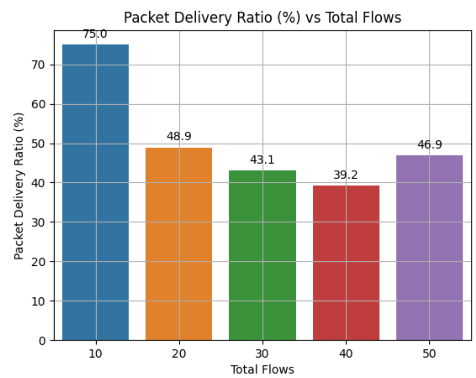
Parameters in this scenario:

Total Node = 40
Node speed = 1 m/s
Area 100 m x 200 m
Packet Rate =50 packet / s
Maximum packet for each echo server = 10
SimulationTime = 200 s







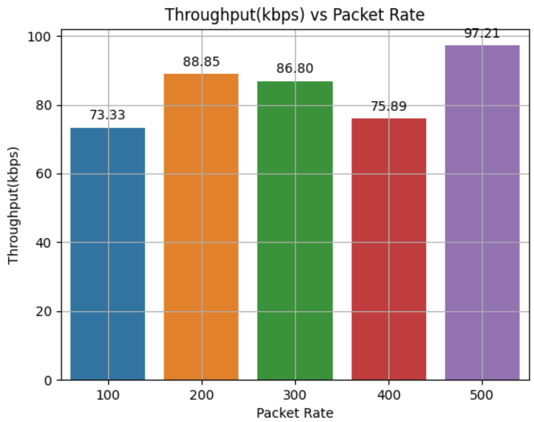


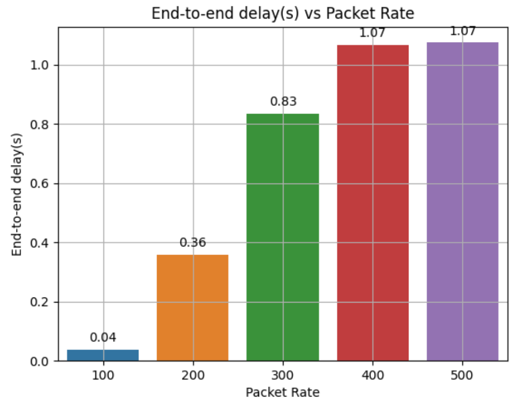
Here added flow has higher impact on total data transferred rather total transfer time. For this reason throughput is increasing with respect to total flows.

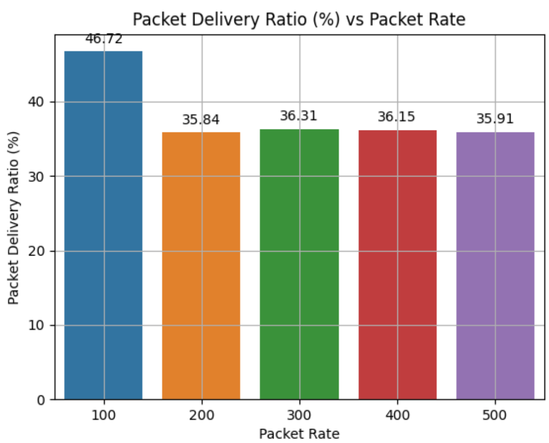
Results for varying packet rate in Low rate wireless network:

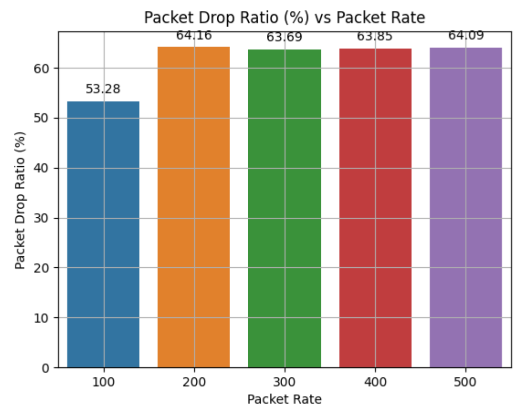
Parameters in this scenario:

- Total node = 20
- Total number of flows = 20
- Node speed = 1 m/s
- Area = 100 m x 200 m
- Maximum packet for each echo server = 100
- SimulationTime = 300 s







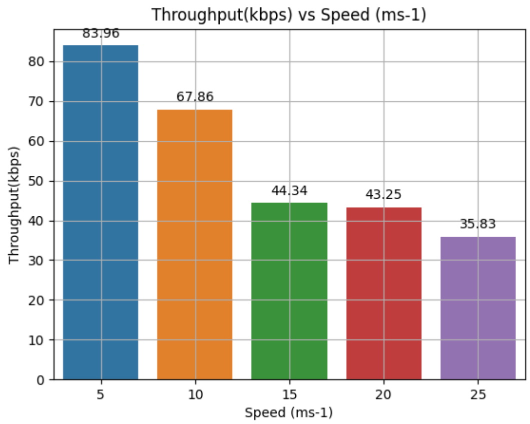


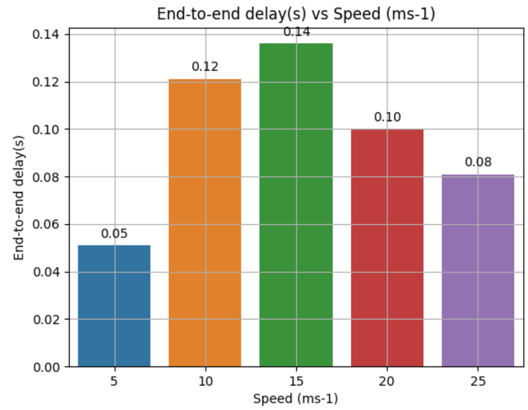
Here end-to-end delay time is increasing due to high congestion at nodes of the routing path. Though the fall of packet delivery ratio is intuitive, we can see packet delivery and drop ratio converges to specific value.

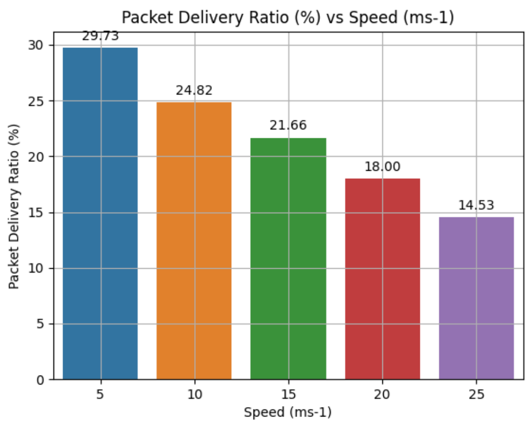
Results for varying node speed in Low rate wireless network:

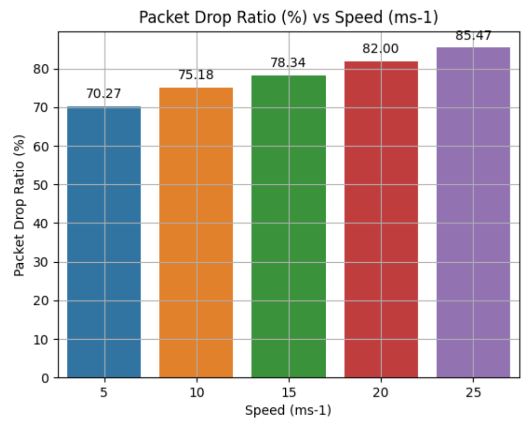
Parameters in this scenario:

Total node = 40
Total number of flows = 70
Area = 400 m x 500 m
Packet Rate = 200 packet per sec
Maximum packet for each echo server = 100
SimulationTime = 300 s









Here the total area is chosen large enough such that high speedy nodes go out of transmission range. So packet drop ratio increases with node speed. Since most of the packets are dropped , throughput is going low. End-to-end delay is calculated for only non-dropped packets, so it improves as speed increases.