CSE 322: Computer Networks Sessional

NS3 Final Project Report

Student

Ataf Fazledin Ahamed
1705066
1705066@ugrad.cse.buet.ac.bd

Supervisor

Md. Tareq Mahmood
Lecturer
Department of CSE, BUET

Assignment Task

Task A
Simulate the following network and measure different metrics.

Network
Wireless High Rate (802.11) - Static
Wireless Low Rate (802.15.4) - Mobile

Metrics
Network Throughput
End-to-end Delay
Packet Delivery Ratio
Packet Drop Ratio

Task B
Implement modification of an existing system. And show metrics difference before and after the change.

Modification
Using Minimum Spanning Tree instead of Dijkstra's Algorithm to find shortest path in Dynamic Source Routing.

Task A - Wireless High Rate (Static)

Topology

In this network simulation, we used a Wifi Adhoc Network. We used Wifi 802.11b standard for our network. YansWifiHelper was used to facilitate the WiFi configuration. As we simulated for static nodes, **ns3::RangePropagationLossModel** was used. **ns3::GridPositionAllocator** was used to position the nodes on a square grid.

Basic Parameters

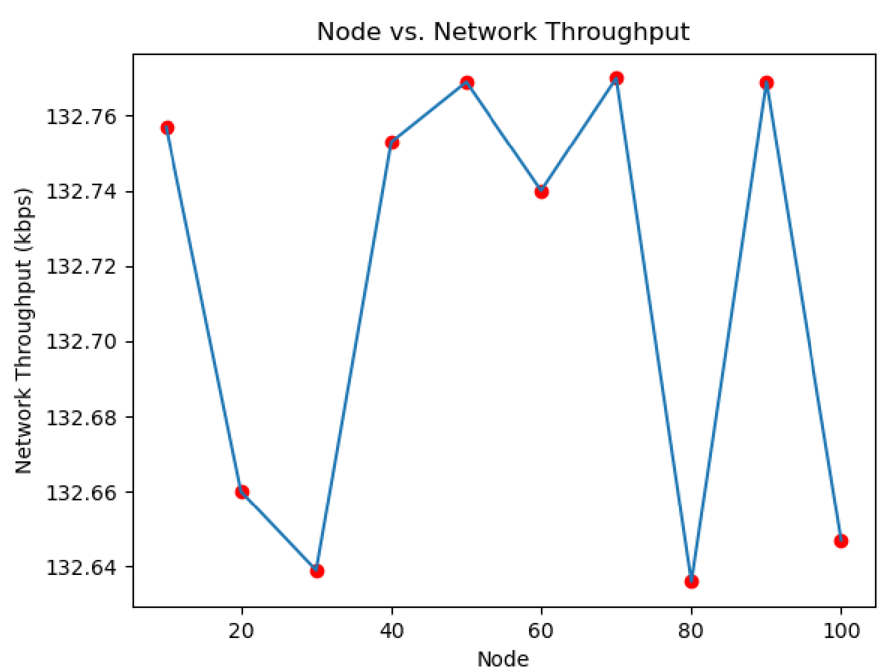
Property	Packet Size	Data Rate
Value	10 KB	240 kbps

Parameters Under Variation

Variaton: Number of Nodes

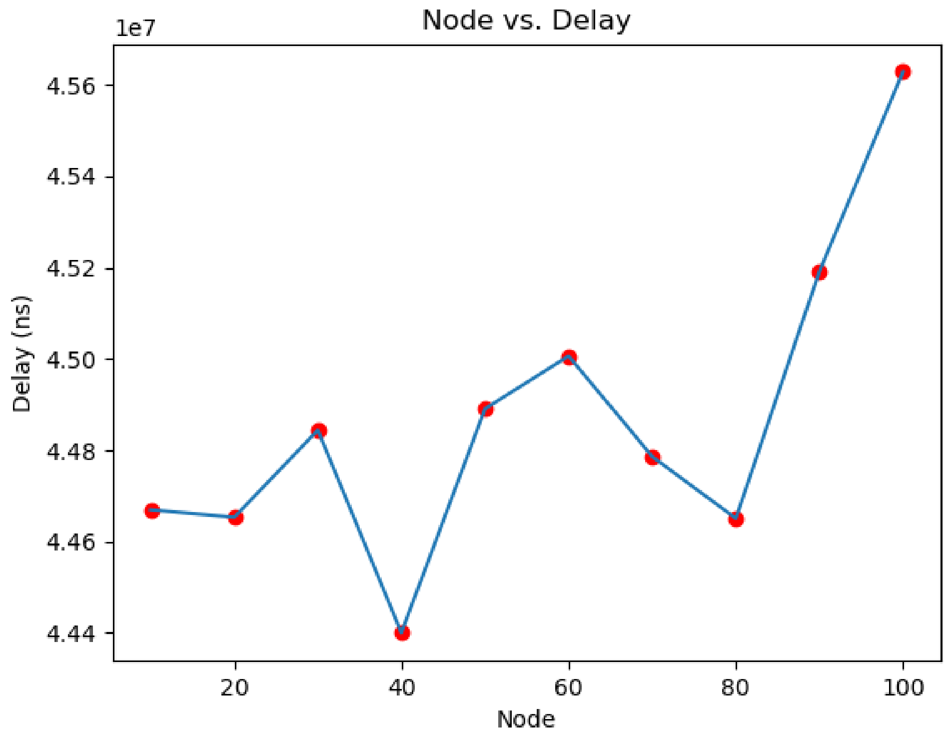
Property	Number of Nodes	Number of Flows	Packets per second	Coverage Area
Value	-	2	3	0.5 x 0.5

Node vs. Throughput



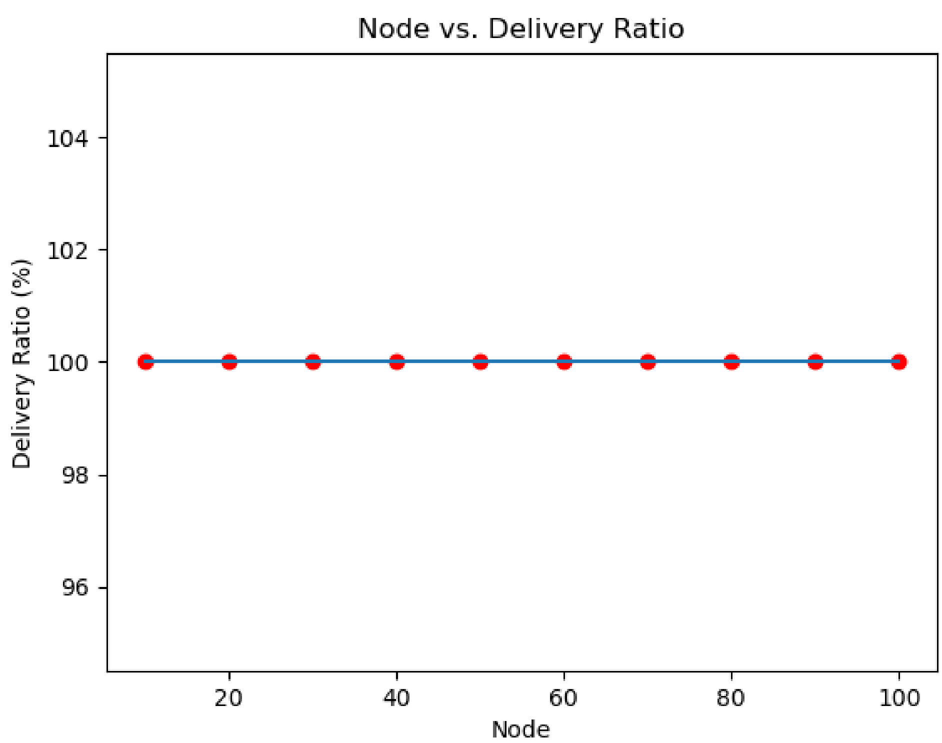
For each data point (Number of Nodes), the server and client were chosen randomly. Which sometimes led to random position on the grid. The low throughput data point to the nodes that were closed to each other- resulting in less bandwidth transfer between them and other nodes. As they didn't need to send out more packets for finding the destination. The high throughput data points to those nodes which were placed far from each other, hence consuming more and more bandwidth and increasing the throughput of the network.

Node vs. Delay



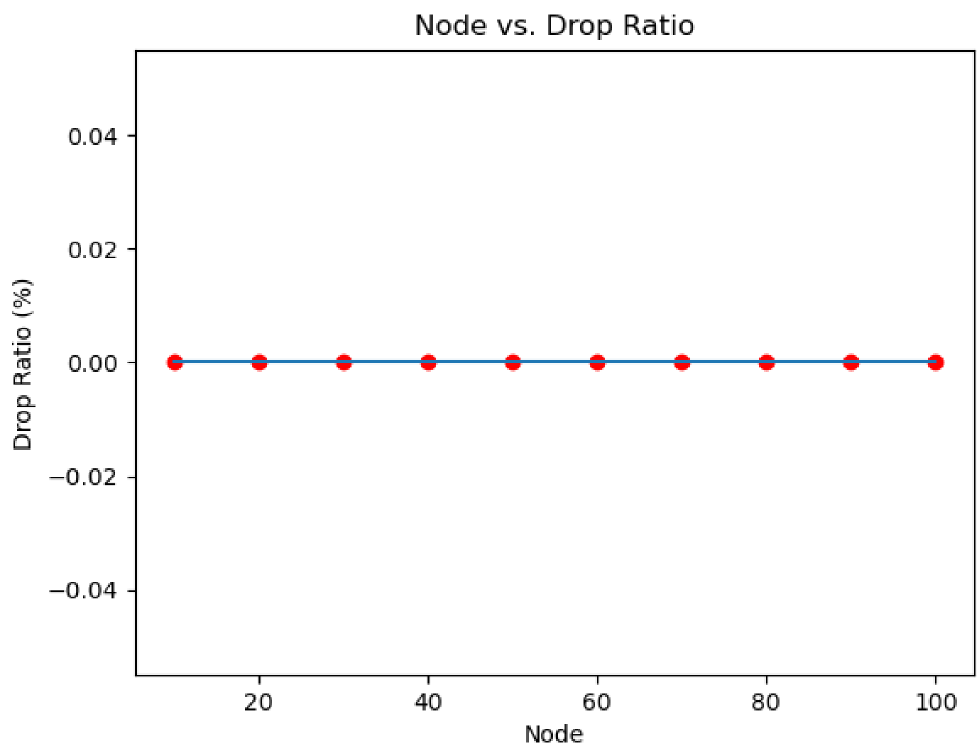
In this graph, most of the points indicate to an increase of delay as number of nodes increases. However, there are few negligible points where the delay becomes less than previous. Considering that as two points randomly placed nearby, it can be ignored. As a result, we find a proportional relation between delay and number of nodes.

<u>Node vs. Delivery Ratio</u>



This graph maintains a straight line indicating that all packets reached the destination no matter how many nodes were present in the network. From the previous graph, we can debate that the packets take more time to reach the destination- but they don't drop or get lost.

Node vs. Drop Ratio

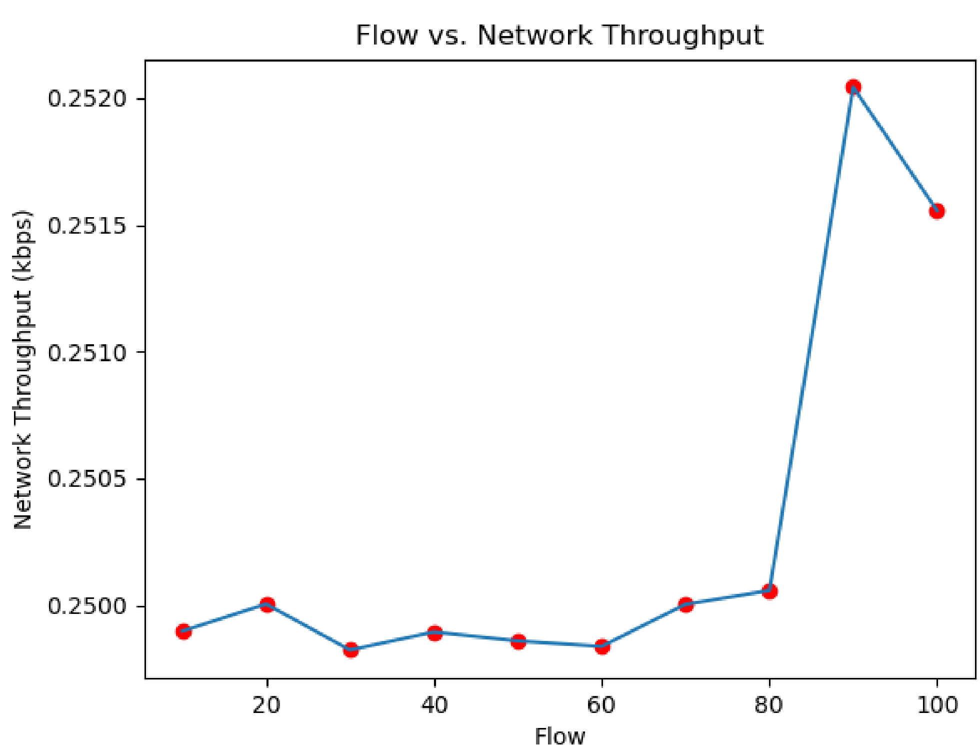


This graph also validates our argument that the number of nodes in the network do not have any connection to packet drop ratio. The insight found through this graph perfectly aligns with the previous one.

Variaton: Number of Flows

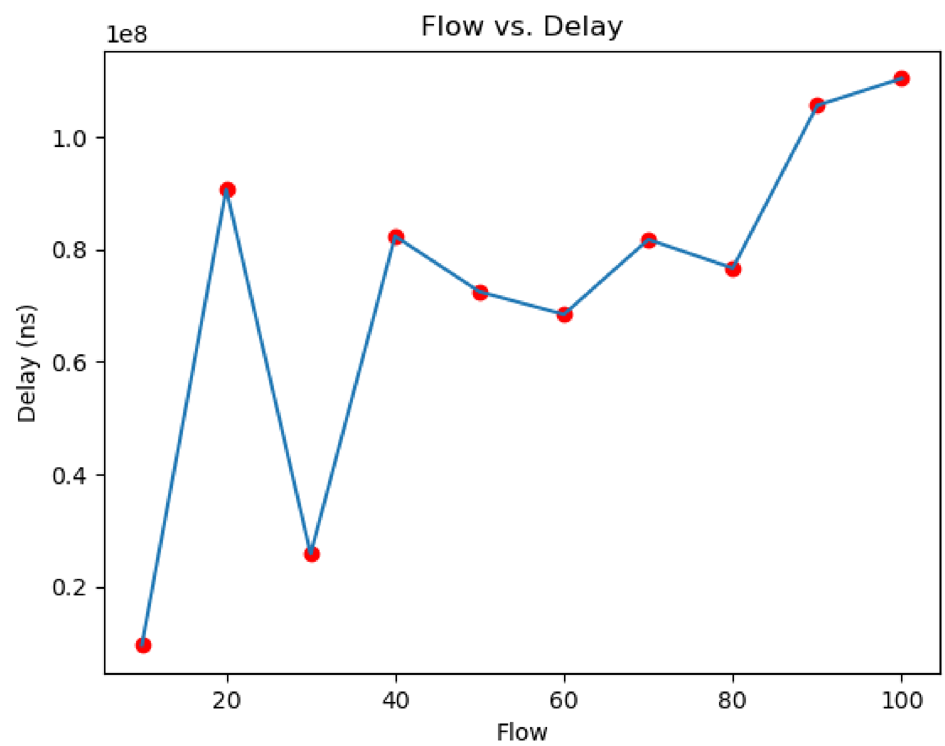
Property	Number of Nodes	Number of Flows	Packets per second	Coverage Area
Value	50	-	3	0.5 x 0.5

Flow vs. Throughput



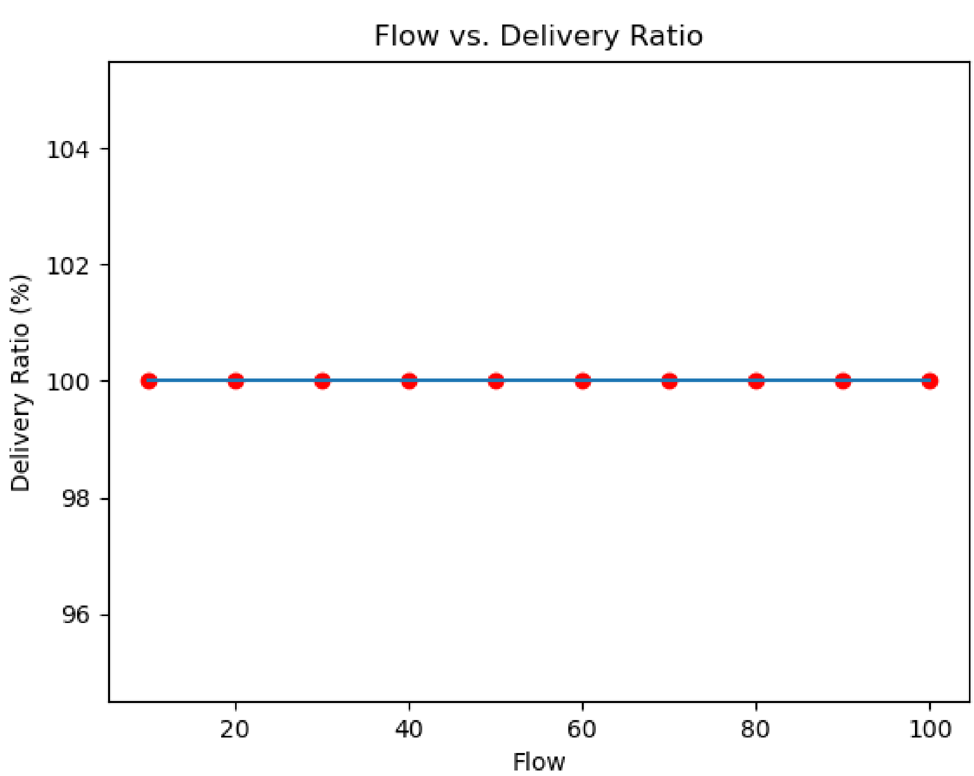
In this graph, the value of Network Throughput fluctuates around the values. However, when the number of flow increases to a value of 90, the network throughput increases drastically. Another value at 100 nodes, the throughput remains high. More observation could have been conducted to see the pattern. Due to time and resource constraints, it couldn't be done.

Flow vs. Delay



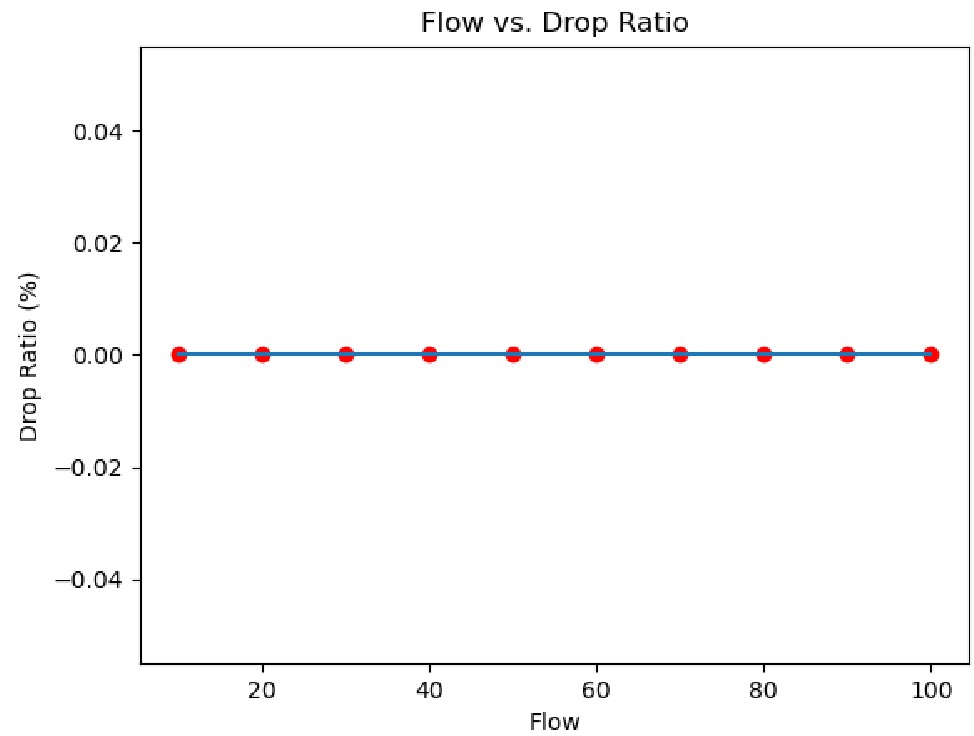
This graph clearly shows that the increase in flow results in an increase of delay too. There's only one data point that shows otherwise. Such reason could be the nodes maintaining the flow remain in a closer position than other data points.

Flow vs. Delivery Ratio



Similar to Node vs. Delivery Ratio, this graph shows us that the number of flows has no effect on delivery ratio.

<u>Flow vs. Drop Ratio</u>

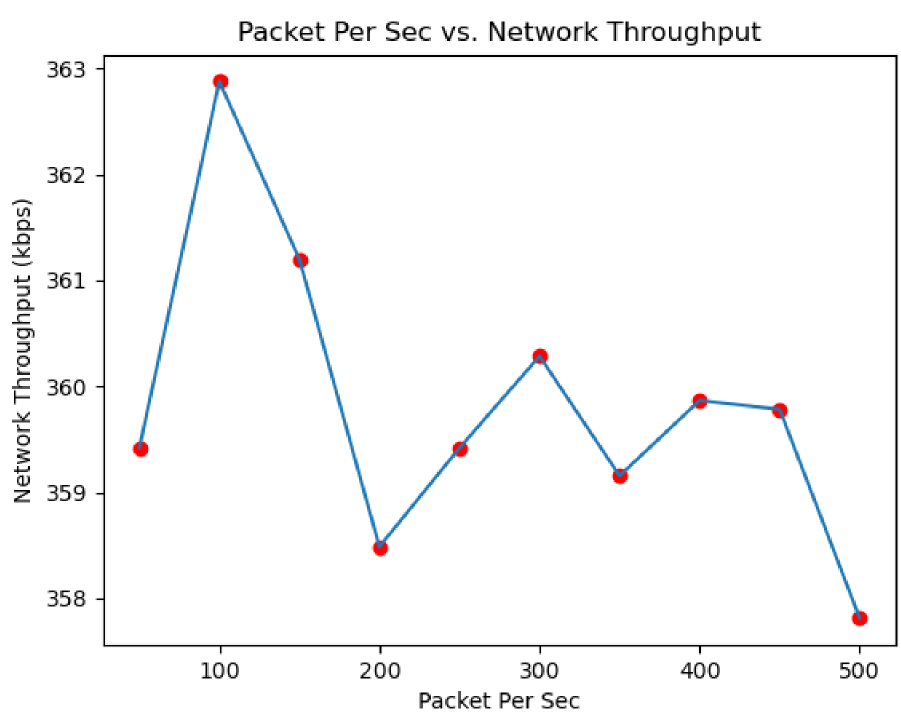


This graph also resembles to the one of Node vs. Drop Ratio. The number of flows has no effect over drop ratio.

Variaton: Number of Packets per second

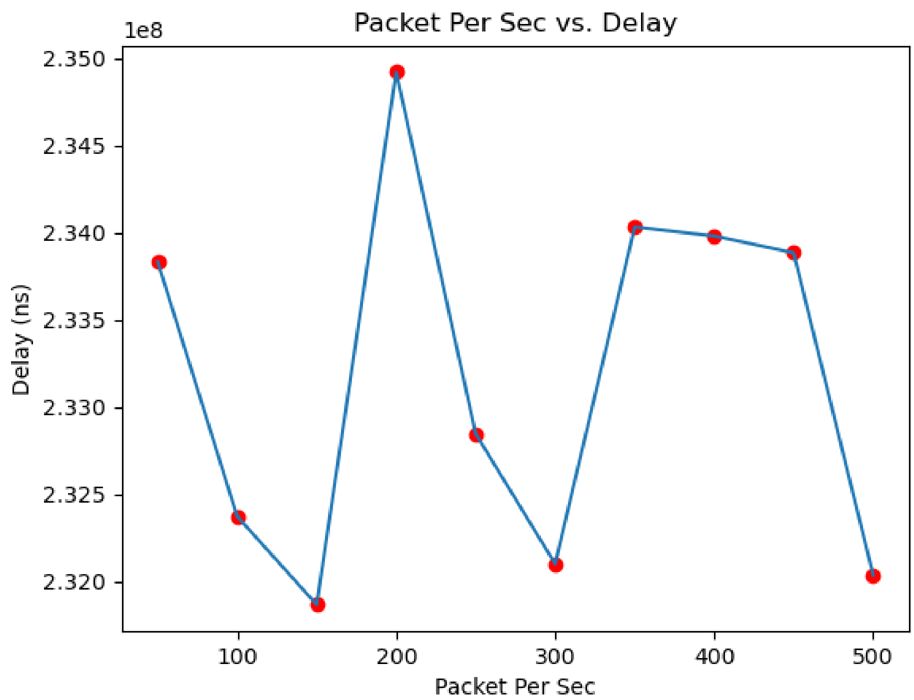
Property	Number of Nodes	Number of Flows	Packets per second	Coverage Area
Value	20	2	-	0.5 x 0.5

<u>Packet per second vs. Throughput</u>



At a glance, this graph may look as if the throughput increases with packet/s. But after observing few more data points, it becomes clear that the throughput fluctuates between a range. However, due to the randomness of our simulation, the data point at 200 p/s and 500 p/s has lower throughput value. Which indicates that the nodes were closer with each other.

Packet per second vs. Delay

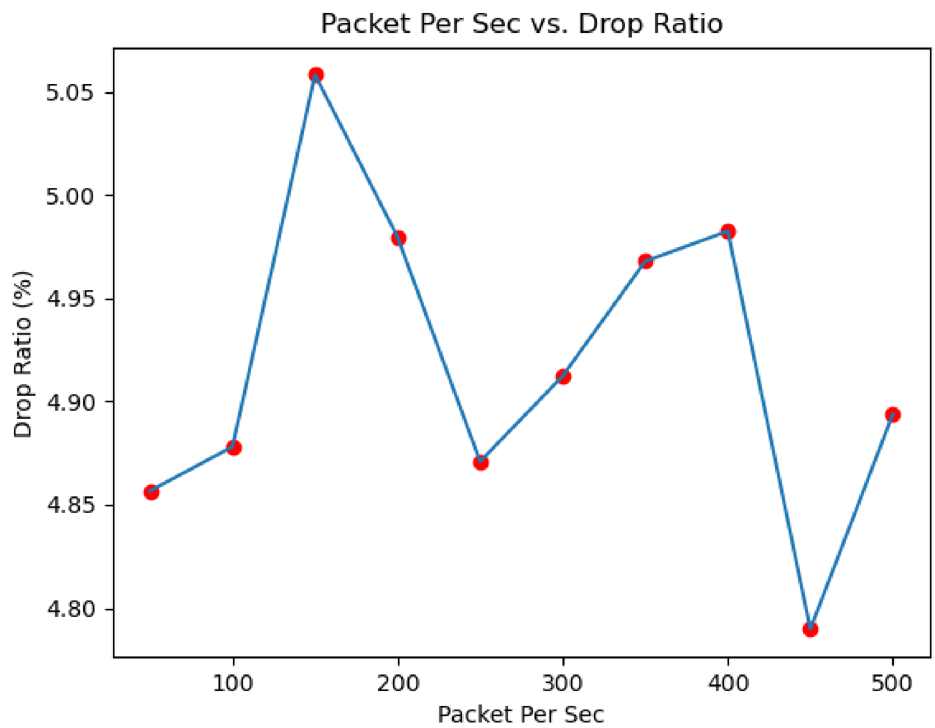


Same as the previous graph, the median value of delay fluctuates between a range. Also due to the randomness, the delay increases/decreases. The high value of delay points to the nodes being distant with each other while the rest indicate the opposite.

Packet per second vs. Delivery Ratio

This graph resonances with previous ones. The value being fluctuated. The delivery ratio decreases whenever high value of packet/s approaches. Possibly, it could be related to the random positioning of nodes.

Packet per second vs. Drop Ratio

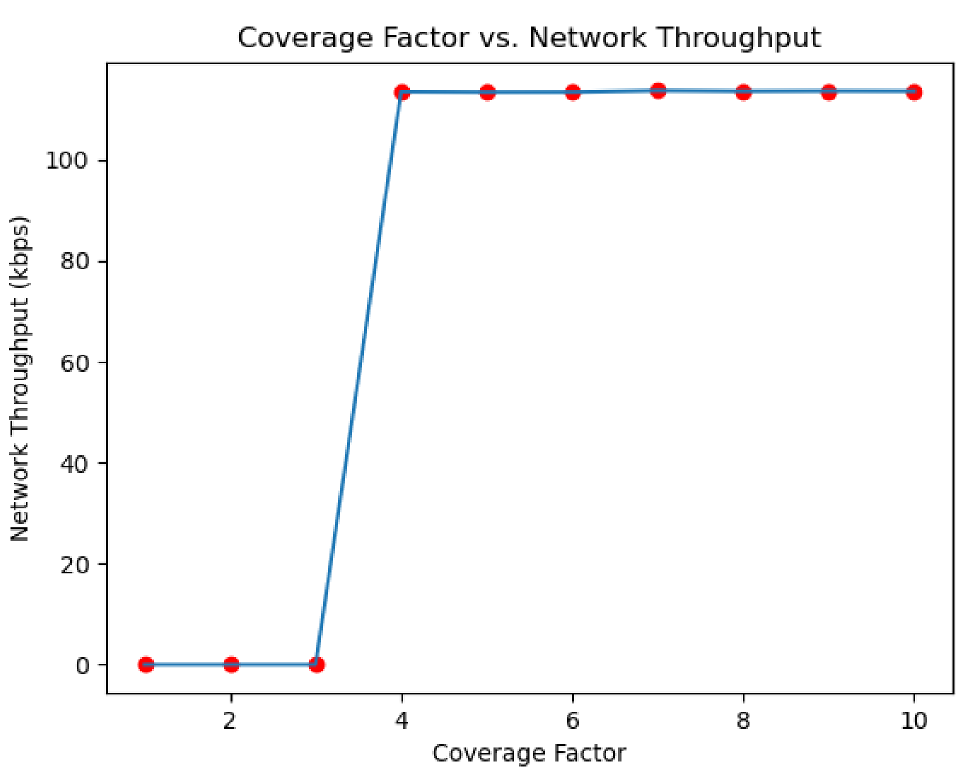


This graph is the opposite of the previous one. The drop ratio increases whenever a high packet/s value arises. After observing this graph, it can be debated that the last two value at 450 p/s and 500 p/s was low due to the random positioning.

Variaton: Coverage Area

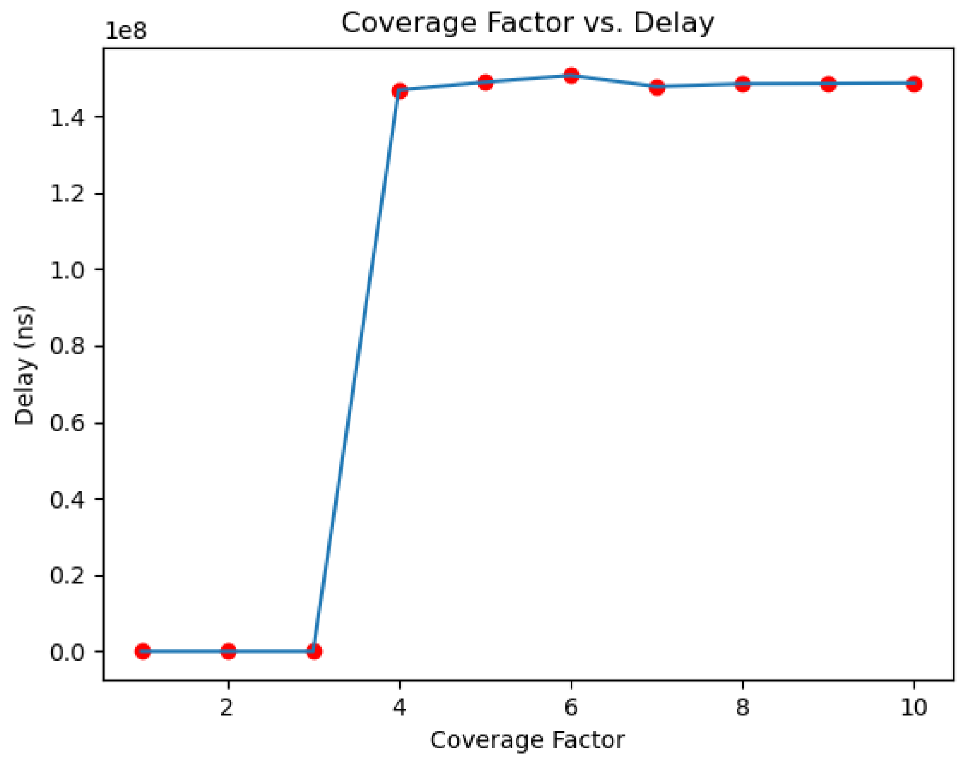
Property	Number of Nodes	Number of Flows	Packets per second	Coverage Area
Value	50	2	3	-

Coverage Area vs. Throughput



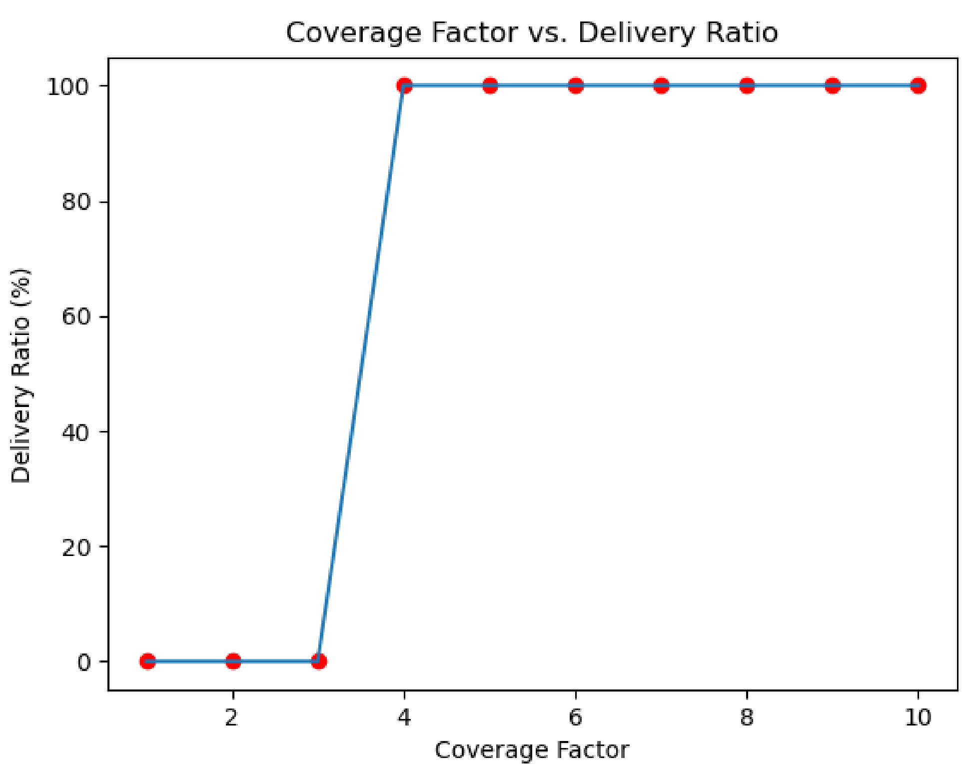
Among all, the graph of this series represent a crystal clear idea about the effect of coverage area on network throughput. In this graph, we see that the network throughput starts coming in after the coverage is 4x of Tx_range. This is because the nodes were places 3 grid spaces away from each other. In case of 2, 1, grid spaces, the coverage needs to be 3x and 2x respectively.

Coverage Area vs. Delay



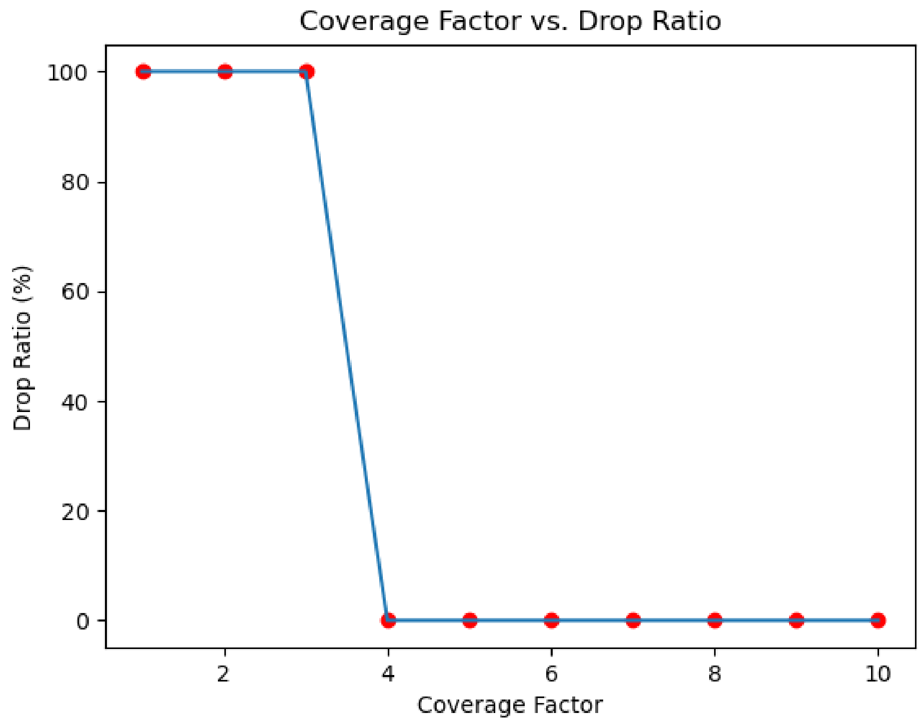
In this graph, when the coverage area falls under the destination node, the delay starts coming up. However, increasing coverage area after that doesn't have any effect on delay, since the signal remains strong.

Coverage Area vs. Delivery Ratio



Before getting inside the coverage area, the destination node and source have no way to communicate over the channel. As a result, the initial delivery ratio are 0%. But after that, the delivery ratio become 100%.

Coverage Area vs. Drop Ratio



Similar to the previous one, the drop ratio is initially 100% because there is no way to reach a destination node out of the range. But once it comes within the range, the drop ratio drops down to 0%.

Task A - Wireless Low Rate (Mobile)

Topology

In this network simulation, we used 6LoPAN Helper classes to create a Wireless Adhoc network over IPv6 layer. During the simulation, we used a custom application for sending and receiving packets over TCP socket. Finally, the overall analytics were captured using the FlowMonitor module.

Basic Parameters

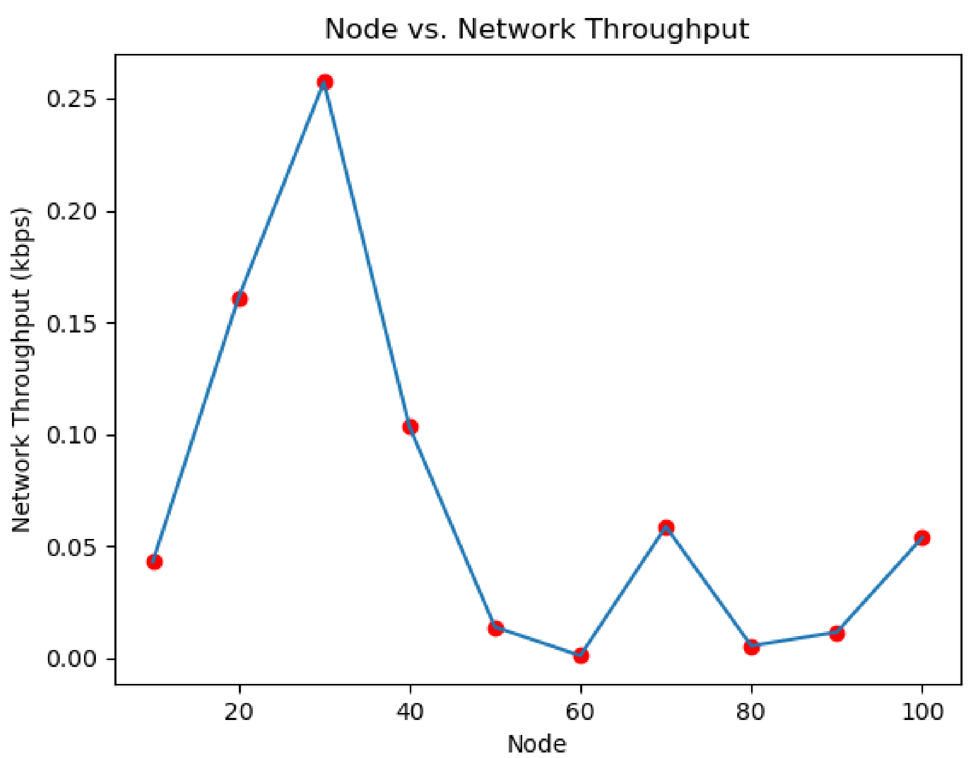
Property	Packet Size	Data Rate
Value	10 KB	240 kbps

Parameters Under Variation

Variaton: Number of Nodes

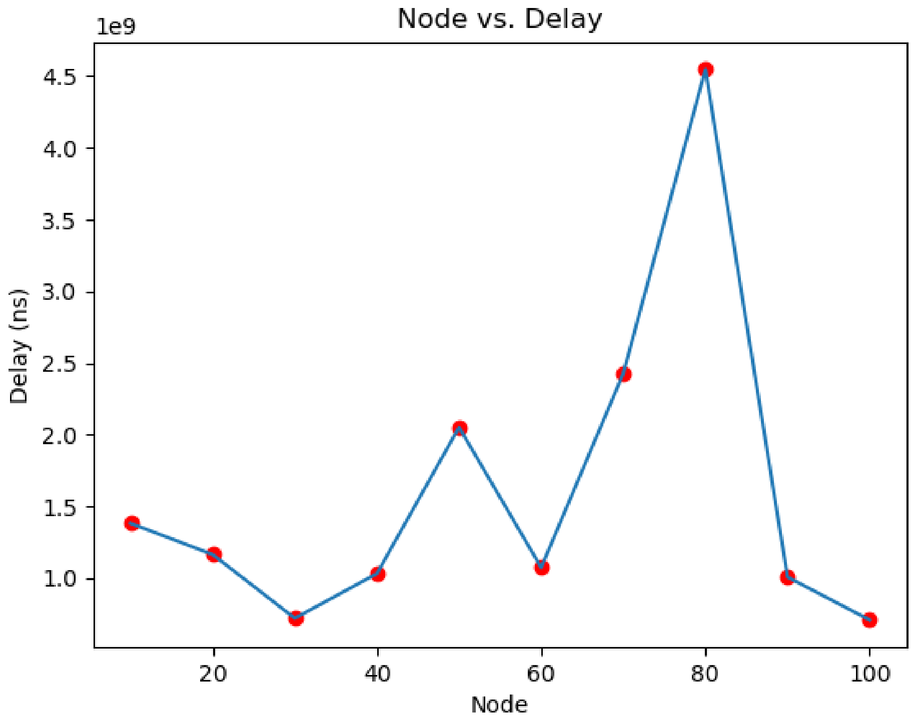
Property	Number of Nodes	Number of Flows	Packets per second	Speed of Nodes
Value	-	20	3	0.1

Node vs. Throughput



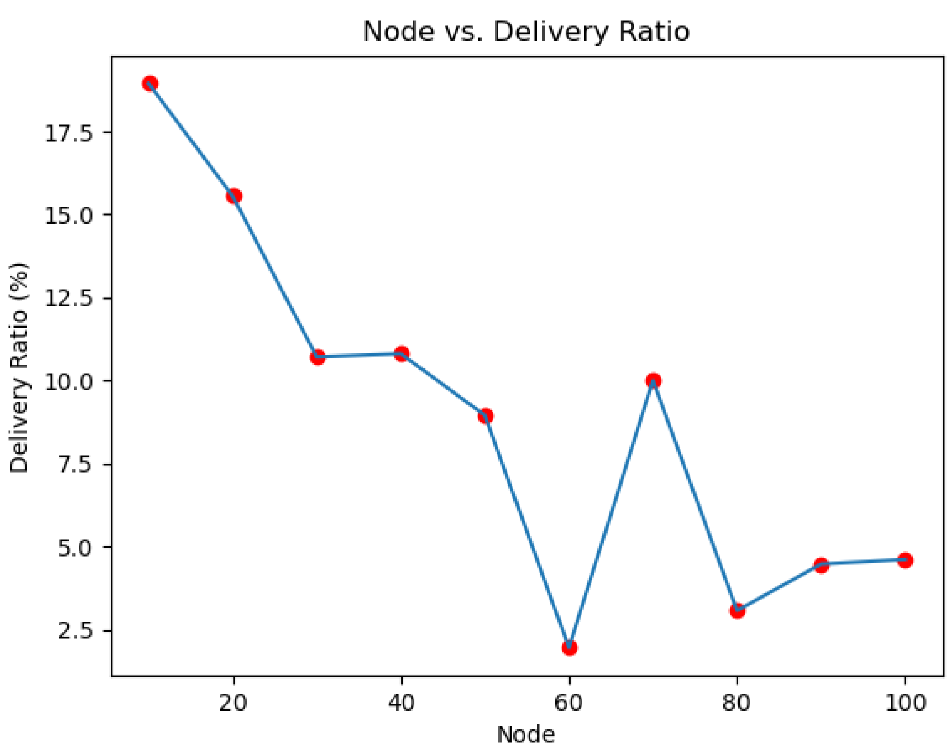
In terms of mobile wireless network, the value of throughput decreases once it reaches a certain threshold. Possibly because an increase in node refers to a strong connection, since this is an Adhoc network. And less path needs to be travelled before finding the destination of the packet.

Node vs. Delay



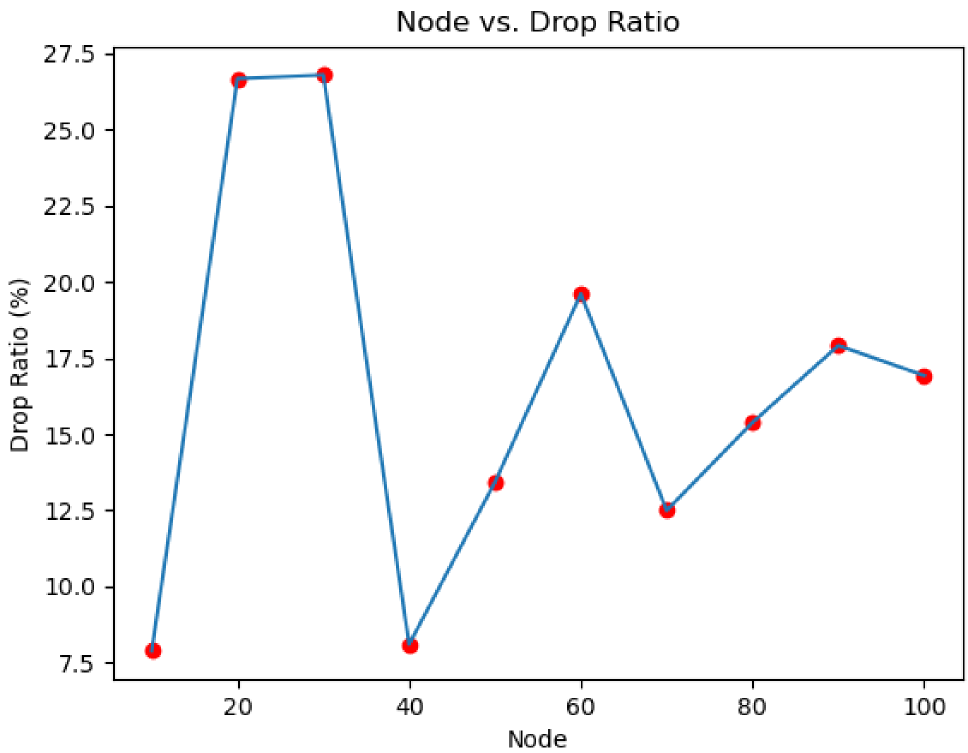
Apart from a single data point at 80, the delay fluctuates between a range- no matter what the number of nodes is. The data point at 80 basically is an erroneous one. Several simulation have been tried and yet, almost the same outcome has been observed. It is to believe that there could be another variable at play for the result.

<u>Node vs. Delivery Ratio</u>



In this graph, we see that the delivery ratio starts decreasing once the number of nodes increase. This is due to the 'lost' packets, not 'drop' packets. Since the nodes in the network increase, packets start travelling from device to device and gets lost.

<u>Node vs. Drop Ratio</u>

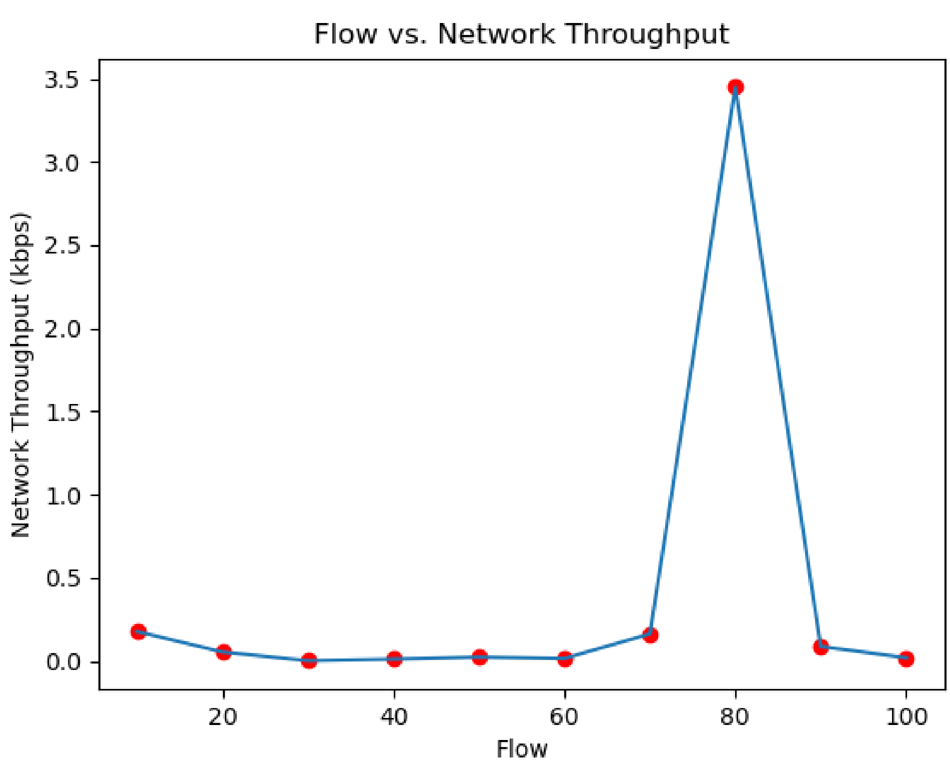


In this graph, the drop ratio fluctuates once the nodes are increased. However, this does not complement the delivery ratio graph. Which leads us to the conclusion that more and more packets are lost.

Variaton: Number of Flows

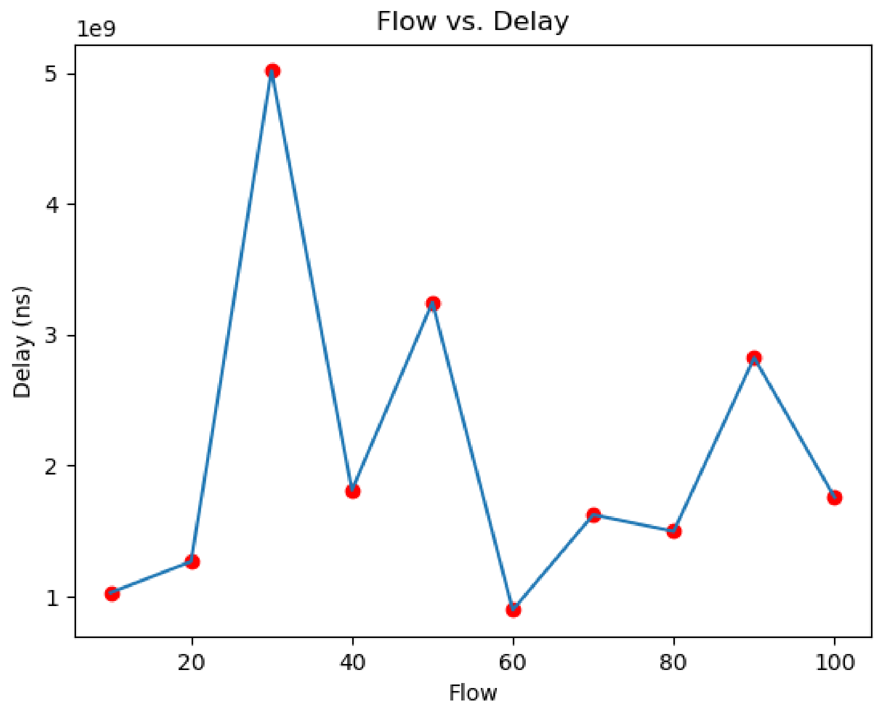
Property	Number of Nodes	Number of Flows	Packets per second	Speed of Nodes
Value	50	-	3	0.1

Flow vs. Throughput



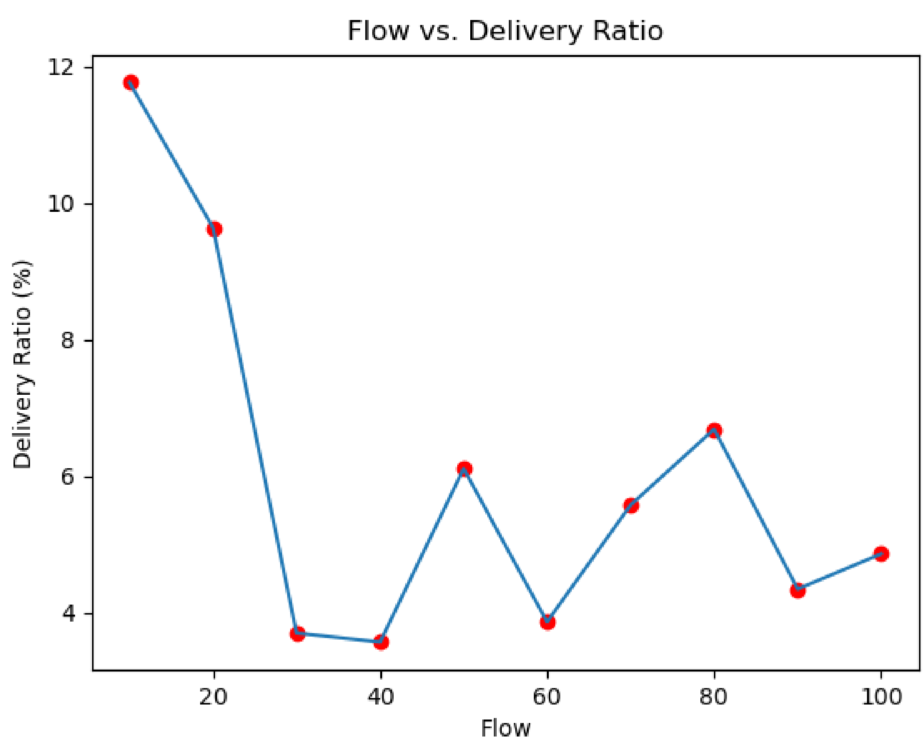
Except for one data point at flow 80, the rest of the data points indicate that the increase of flow doesn't quite affects network throughput. The value of throughput fluctuates in a range.

Flow vs. Delay



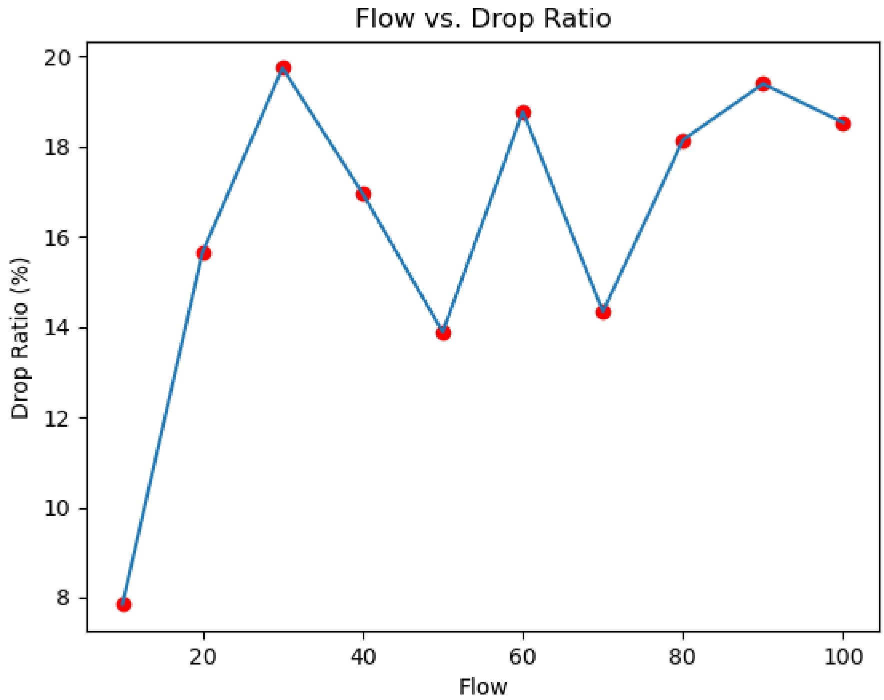
Increase of flow indicates to an increase in delay with some drastic fluctuations in between. Probably due to the mobile nature of the network- since the nodes are moving.

Flow vs. Delivery Ratio



As flow increases, the delivery ratio decreases and one point fluctuates around a range. This is to be noted that most of the packets are actually lost- since in the next graph, the drop ratio doesn't complement the delivery ratio.

Flow vs. Drop Ratio

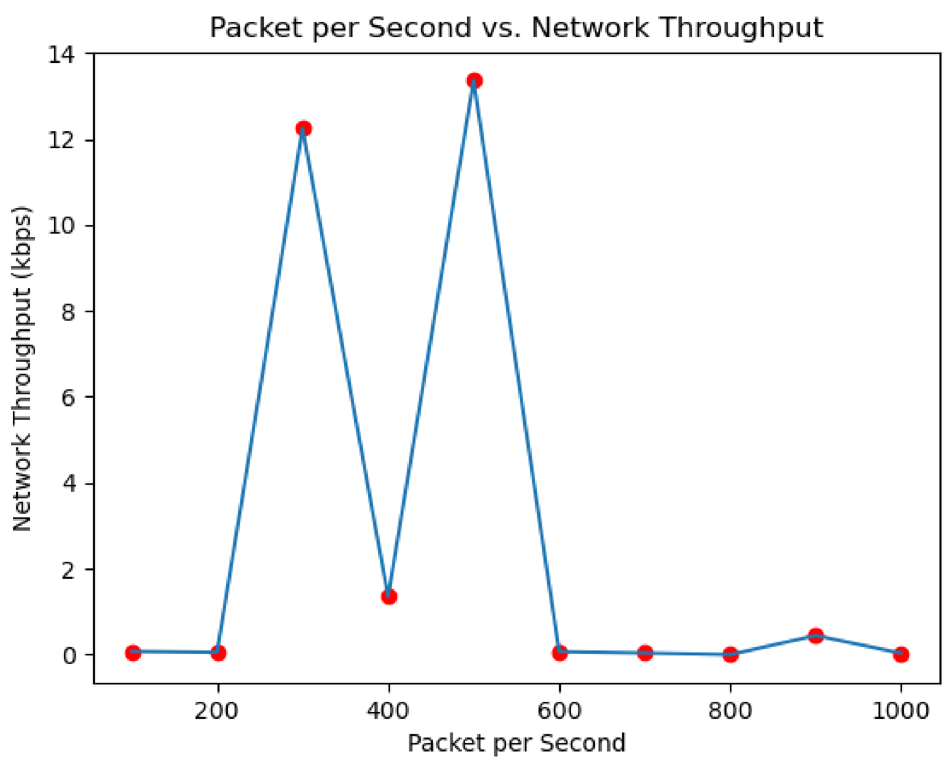


As stated in the previous graph, the drop ratio increases as flow increases and fluctuates around a range. A significant portion of the packets are lost.

Variaton: Number of Packets per second

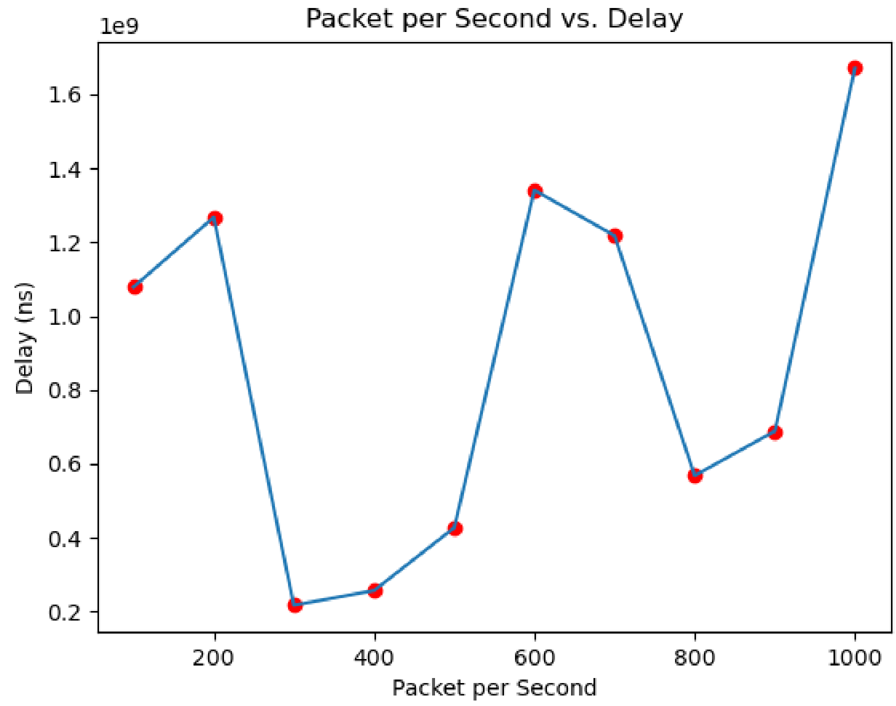
Property	Number of Nodes	Number of Flows	Packets per second	Speed of Nodes
Value	50	20	-	0.1

<u>Packet per second vs. Throughput</u>



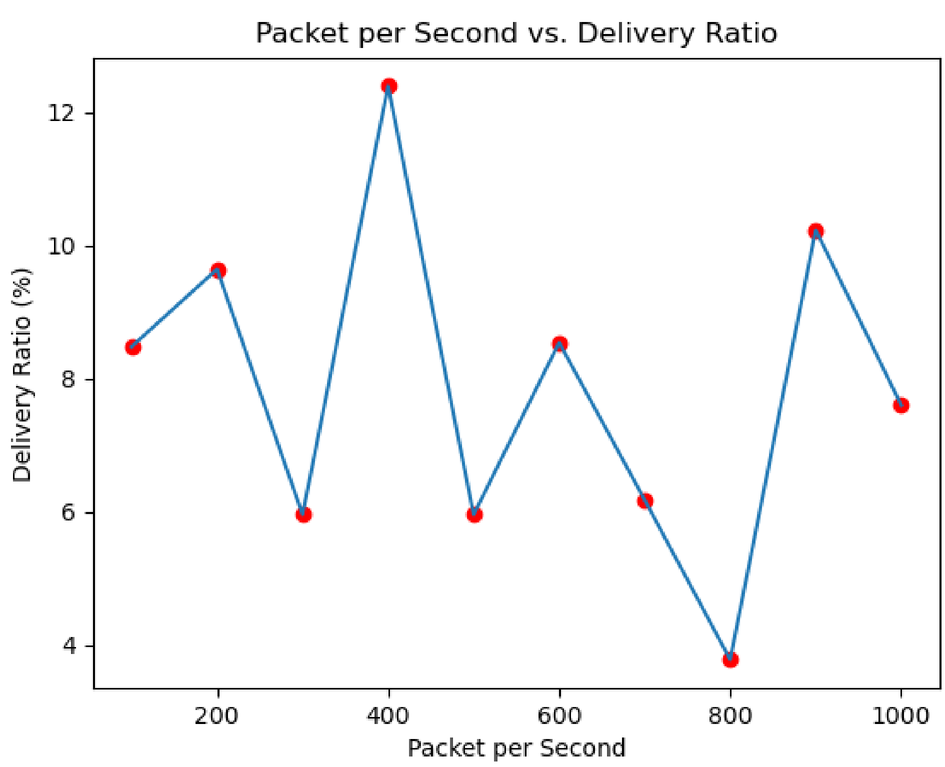
Except for two data points at 300 and 500, the value of network throughput doesn't increase. The abnormal points could be due to an error introduced by the randomness of the node position.

<u>Packet per second vs. Delay</u>



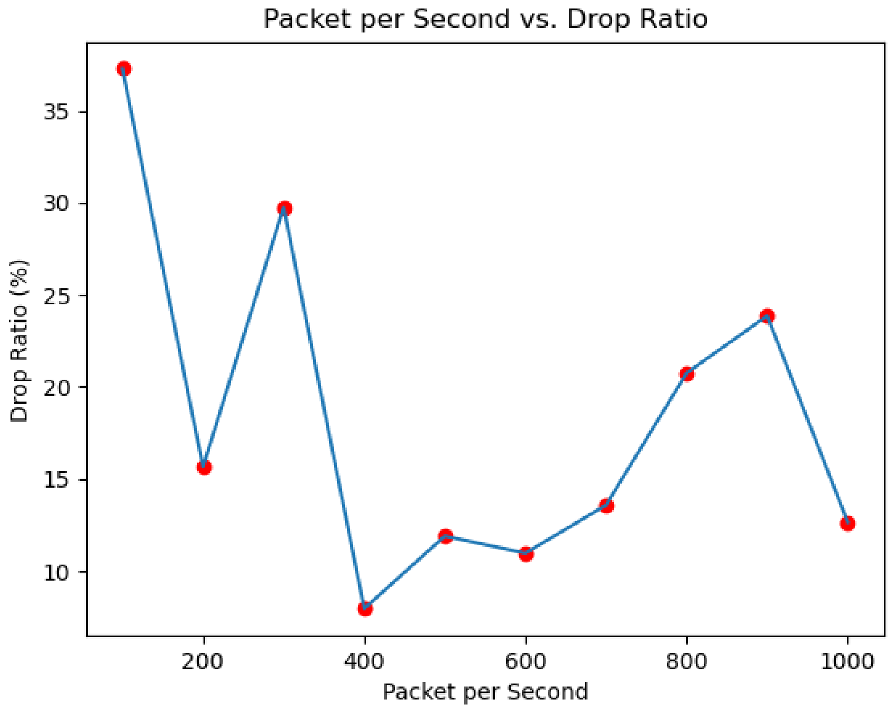
Ignoring some data points in this graph leads us to an insight that the delay increases as packet/s increases.

<u>Packet per second vs. Delivery Ratio</u>



The delivery ratio fluctuates between a range as packet/s increases. This is a lot similar to the previous instance. Such similarity could mean that the reason behind packet loss is due to the mobility nature of the adhoc network.

<u>Packet per second vs. Drop Ratio</u>

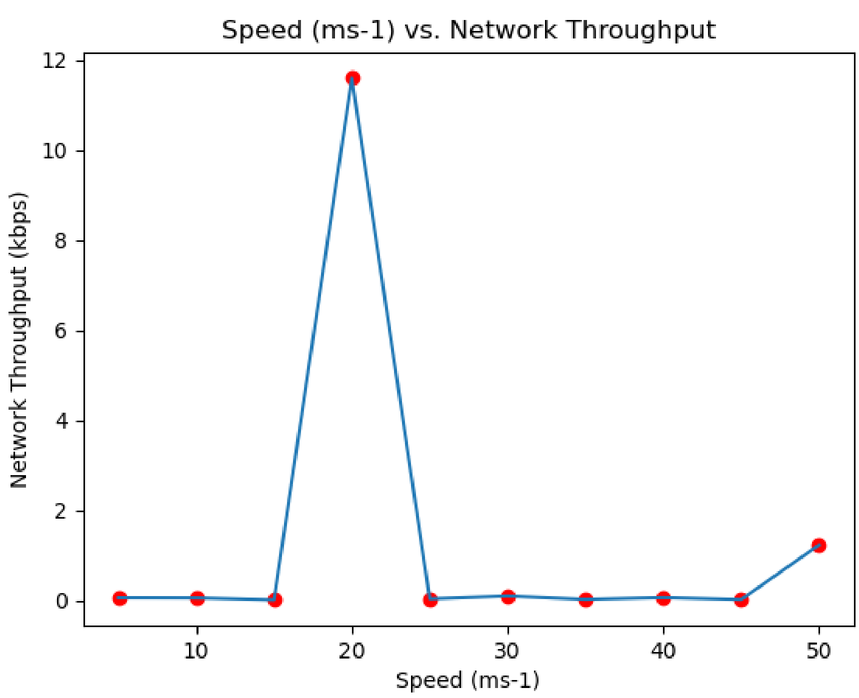


This graph also resembles the previous instance. Even when drop ratio increases as packet/s increases, most of the packets are lost.

Variaton: Speed of Nodes

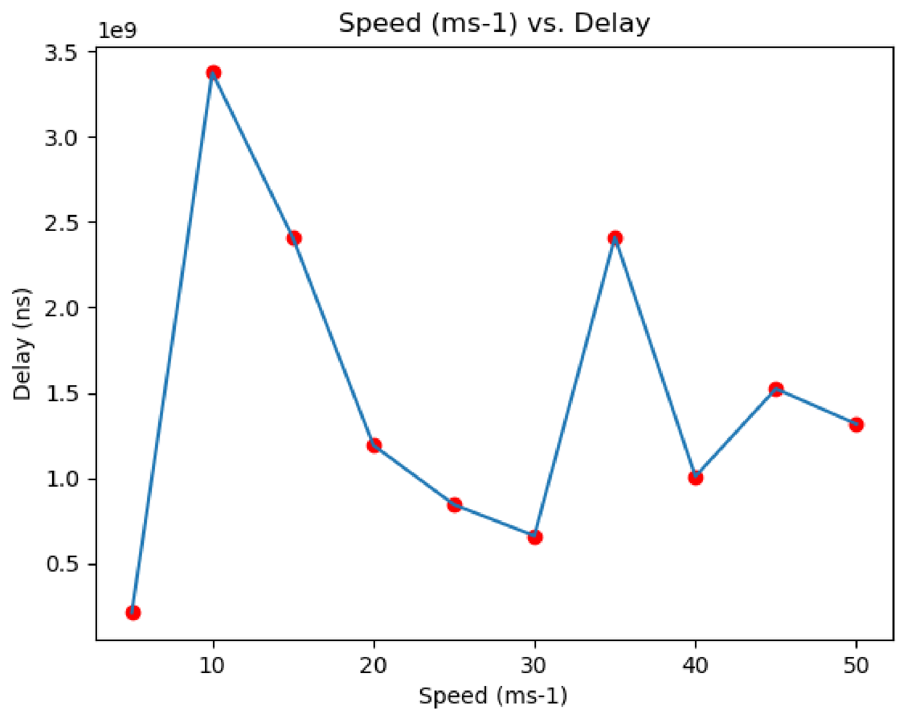
Property	Number of Nodes	Number of Flows	Packets per second	Speed of Nodes
Value	50	20	3	-

Speed vs. Throughput



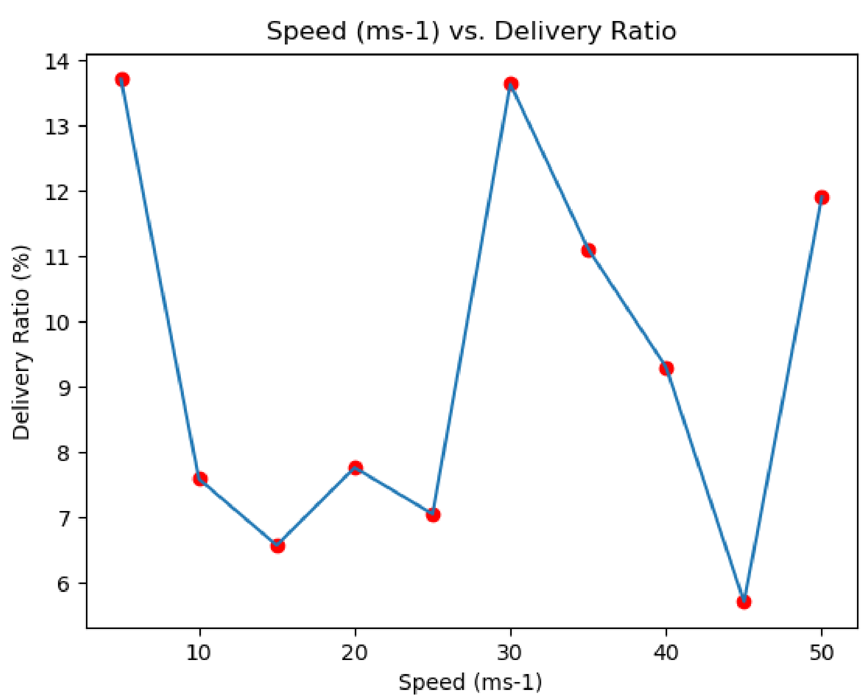
Except for one data point at flow 80, the rest of the data points indicate that the increase of speed doesn't quite affects network throughput. The value of throughput fluctuates in a range.

Speed vs. Delay



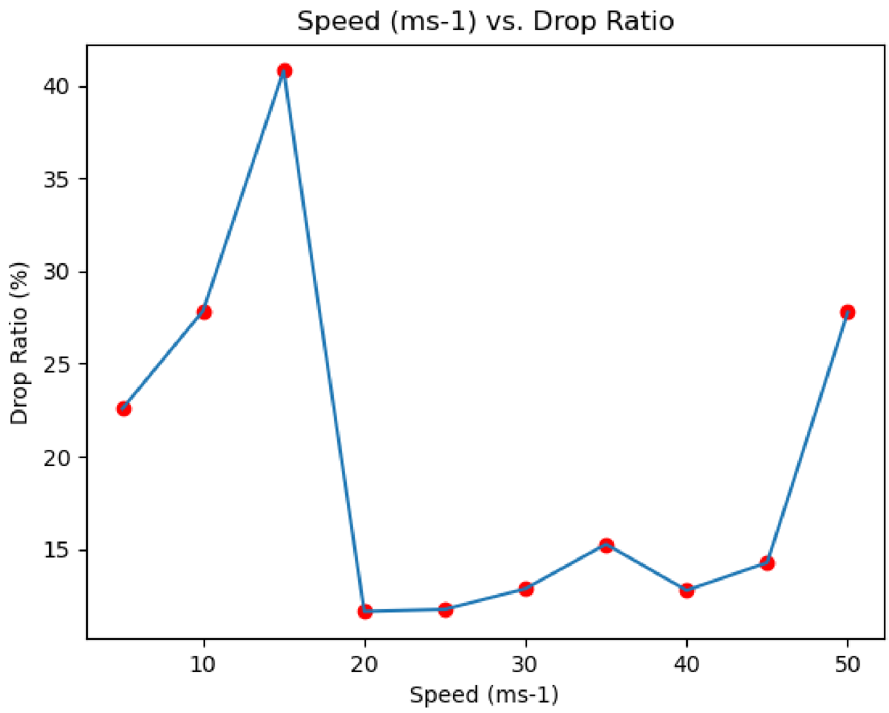
The value of delay fluctuates in a range. And at some region, it decreases as the speed increases.

<u>Speed vs. Delivery Ratio</u>



This graph can be divided into two segments. And both segment indicate that with increase in speed, the delivery ratio of the network decreases rapidly. There could be a high chance that plotting more data could end up showing a repetitive pattern in the graph.

<u>Speed vs. Drop Ratio</u>



Ignoring initial values, the value of drop ratio increases after a while. An increase in speed increases the drop ratio also. And since drop ratio and delivery ratio do not complement each other, most of the packets are lost.

Task B

Using Minimum Spanning Tree instead of Dijkstra’s Algorithm for finding shortest path in Dynamic Source Routing

Modification of Dynamic Source Routing- that’s used for Wireless Adhoc Networks to use Minimum Spanning Tree instead of Dijkstra’s algorithm.

Implementation Change Location

The use of Dijkstra’s Algorithm is set in DSR module file called **dsr-rcache.cc**. The file is then edited to our need and replaced with the original one.

Remarks

Due to shortage of time and resources, no baseline performance setting/configuration could be set for the original Dyanmic Source Routing. Thus, the modification in the source file was delayed.

The source file being too obfuscated made it extremely difficult for the modification to take place. As a result, the task could not be completed.

--- End ---