NS2 Project Report

Submitted by:

Name: Shamim Al Mamun Student ID: 1805060

Lab Group: A2

1

LEVEL - 3, TERM - 2

StudentID % 8 : 4

Tasks Performed:

- Simulated networks in ns2.
- Computed and plotted results of simulation in ‘pyplot’.
- Modified source files in simulators to observe impact on performance metrics.

Networks under simulation:

- Wired
- Wireless 802.15.4 (mobile)

Parameters under variation:

Wired	Wireless 802.15.4 (Mobile)
I. Number of Nodes II. Number of flows III. Number of packets per second	I. Number of Nodes II. Number of flows III. Number of packets per second IV. Speed of each node

Metrics variation Graphs observed for each parameter variation:

Wired	Wireless 802.15.4 (mobile)
● Network throughput ● End-to-end delay ● Packet delivery ratio ● Packet drop ratio	● Network throughput ● End-to-end delay ● Packet delivery ratio ● Packet drop ratio ● Energy consumption (Total)

Modifications made in the simulator:

1. Change in Droptail Queue :

Affected Network:

Wired, Wireless 802.15.4

Intuition:

In our simulation both networks use Droptail queue which drops the packet arrived first when the queue becomes full.

I have modified the dropping mechanism so that, instead of dropping the packet arrived first, an index into the queue is randomly selected and packet at that index is dropped.

As a result, when network become too congested, packets from varying instance of time will be dropped instead of dropping serially.

Modified files:

drop-tail.h, **drop-tail.cc**

Consequences:

Decrease in drop-ratio in some case, increase in other cases.

2. Change in DSDV protocol:

Affected Network:

Wireless 802.15.4

Intuition:

Lost_Link is a very common phenomenon in case of mobile wireless networks. I tried to improve the existing lost_link_handling process of DSDV routing protocol, taking idea from this paper.

Steps followed to modify the function `lost_link` for route repair in case of lost link packets:

1. Broadcast a Route Error message (RERR) to inform other nodes in the network of the lost link. The RERR message was broadcast to inform all nodes about lost link.
2. When a node receives a RERR message, it should update its routing table to reflect the lost link, which was done by calling function bCastPacket().
3. Trigger a new route discovery process to find an alternate path to the destination, which was done by calling processUpdate() function.

Moreover, I have changed the values of “Period of Update” to 10 from 15; and “Minimum number of update before discarding stall routes” to 2 from 3, in order to quick update of routing table and fast convergence by discarding unnecessary routing informations.

Modified files:

dsdv.h, dsdv**.cc**

Consequences:

- Throughput increased in all cases.
- End-to-end delay sometimes decreased.
- Energy consumption has decreased.
- Drop ratio has increased.

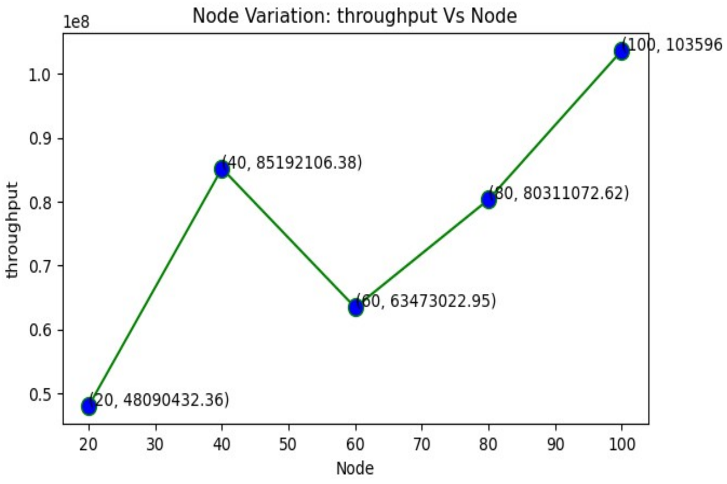
Results with Graphs

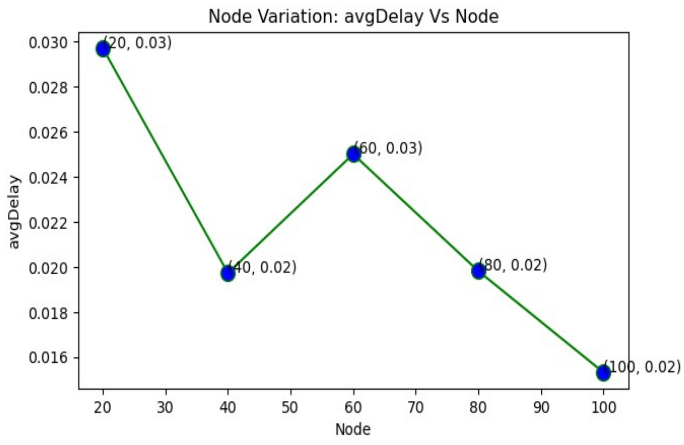
Graphs reflecting variation of metrics with respect to variation of parameters has been attached below:

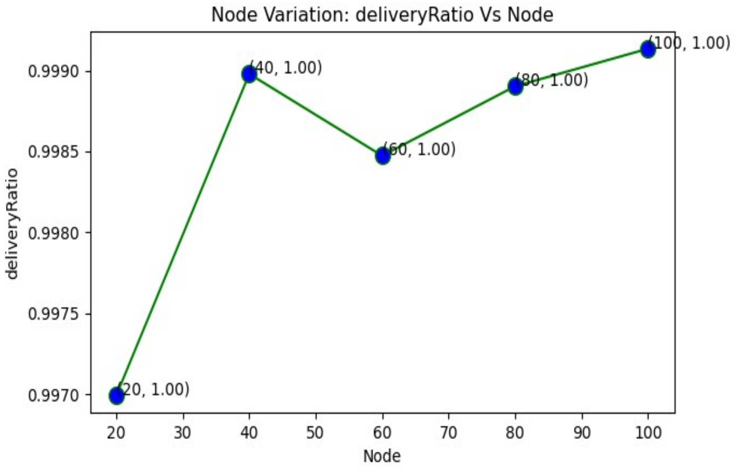
Without Modification:

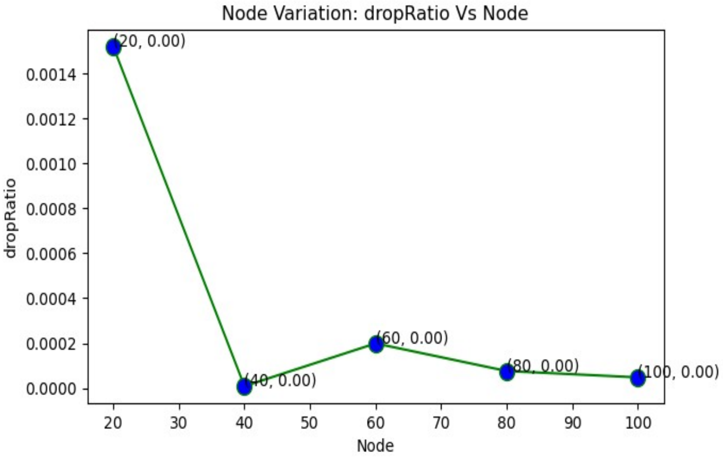
Wired

Node Variation:

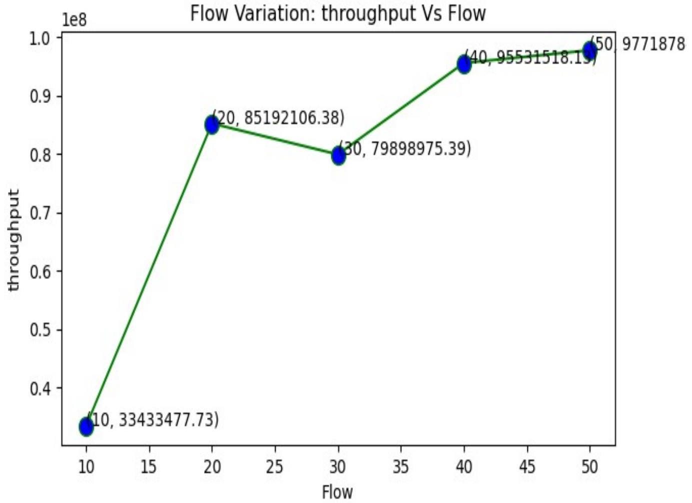


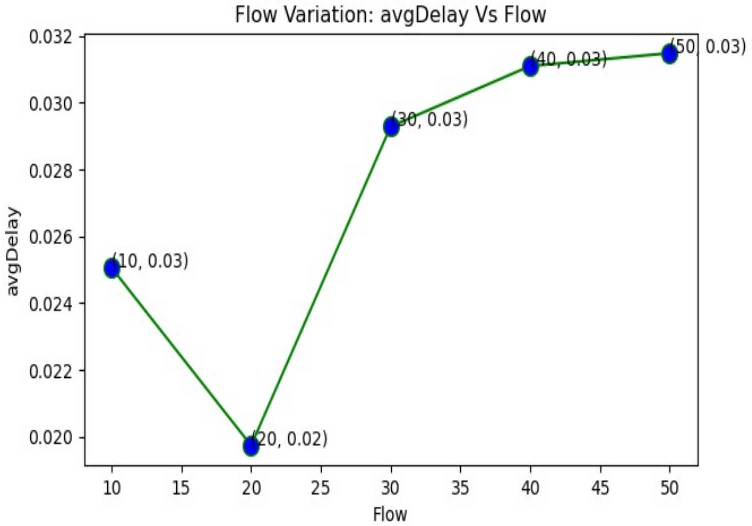


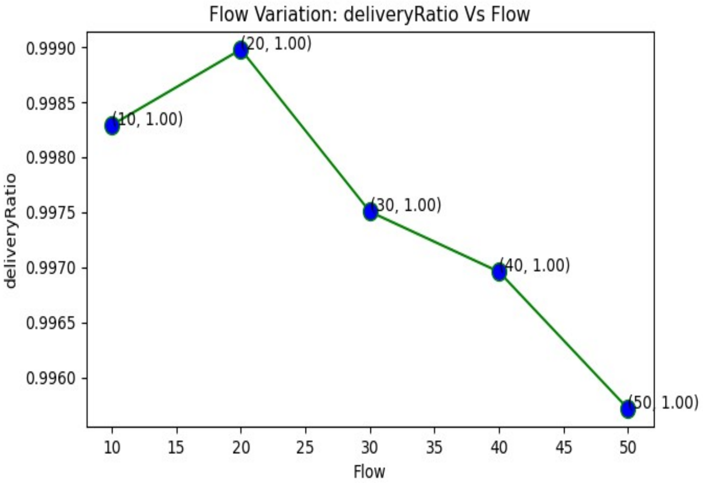


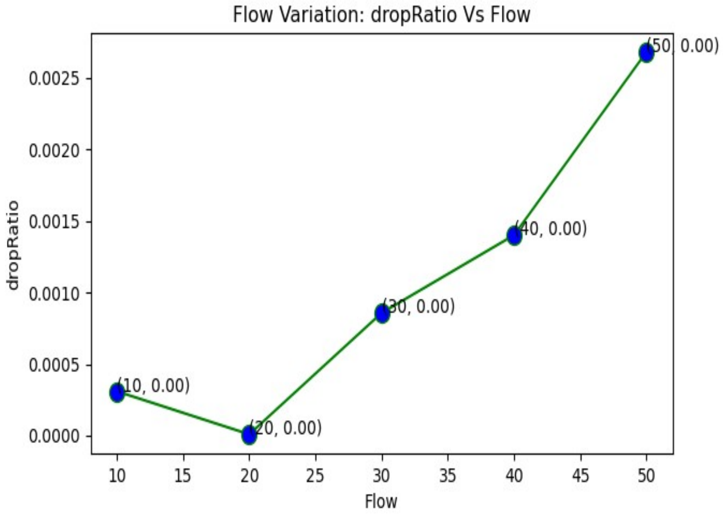


Flow Variation:

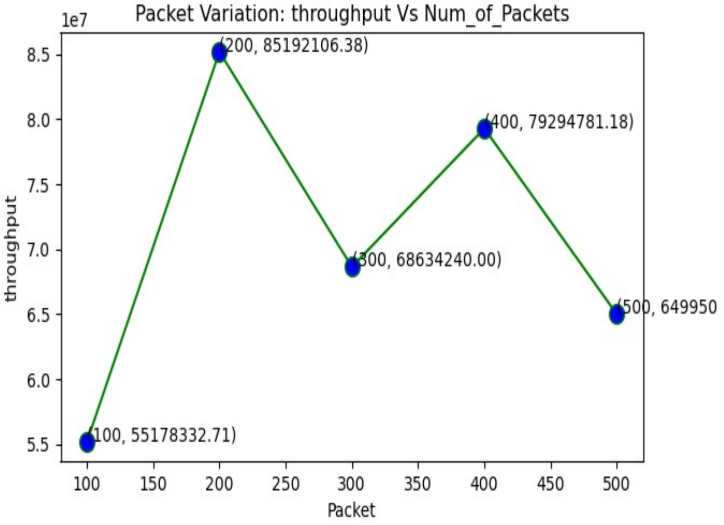


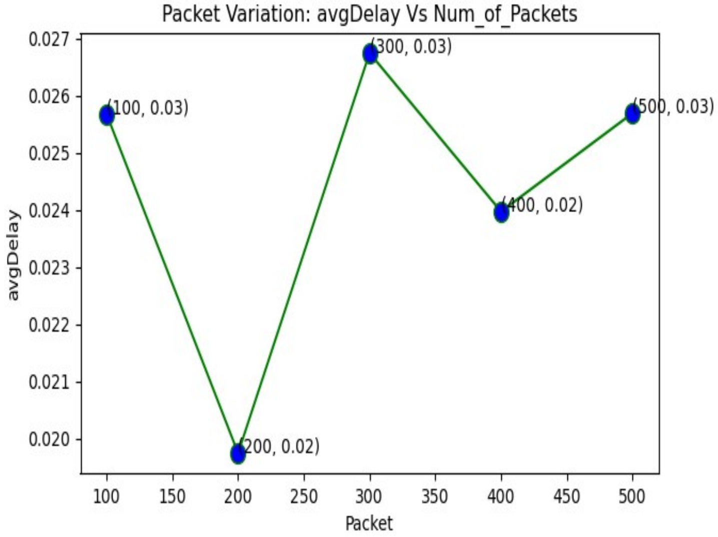


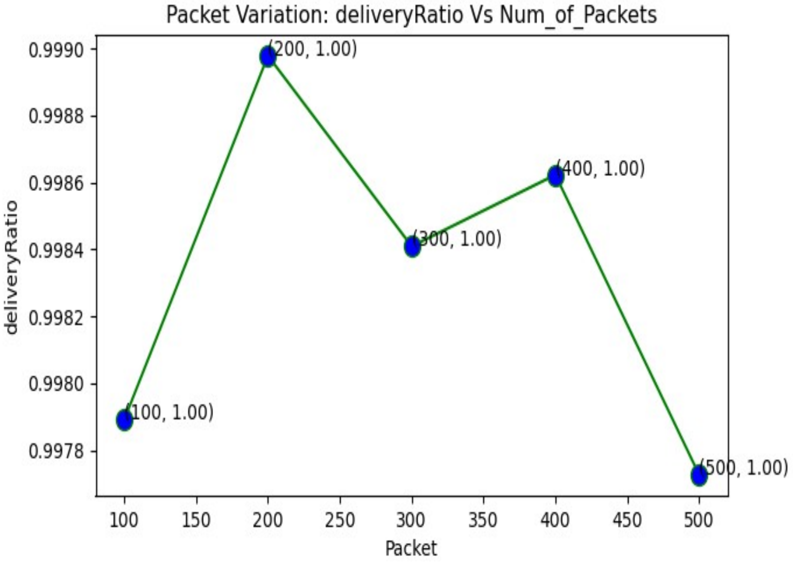


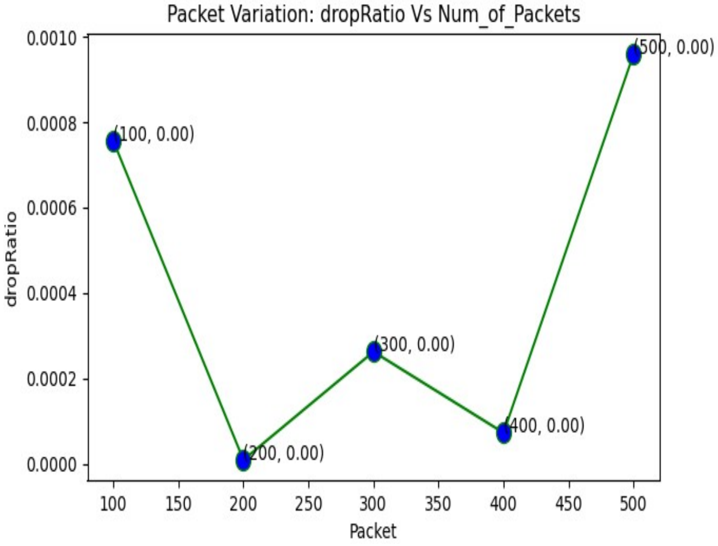


Packet Variation:



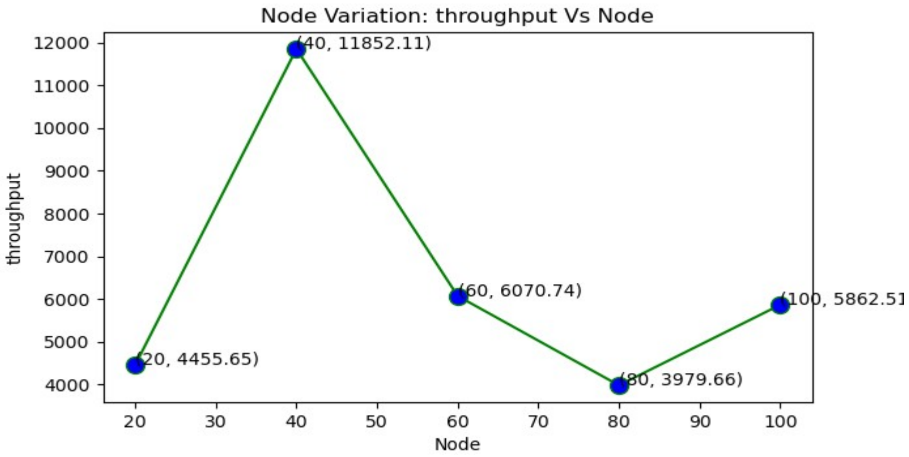


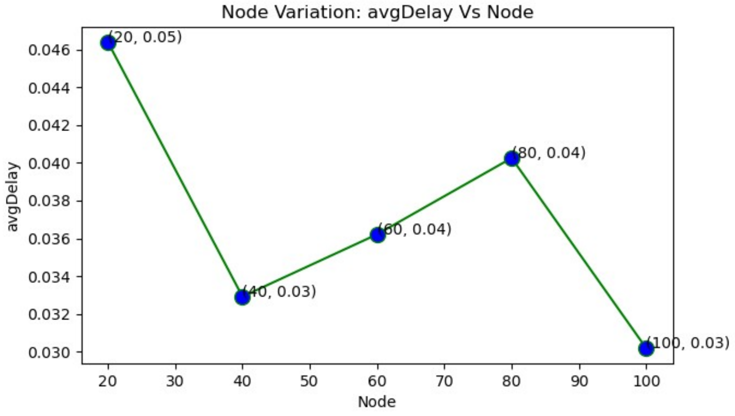


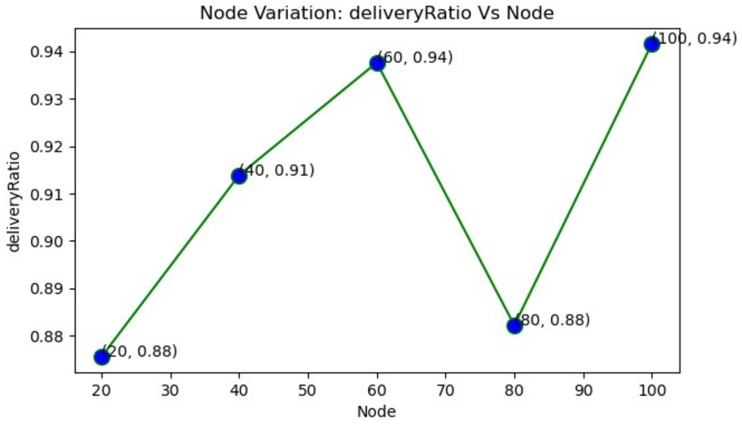


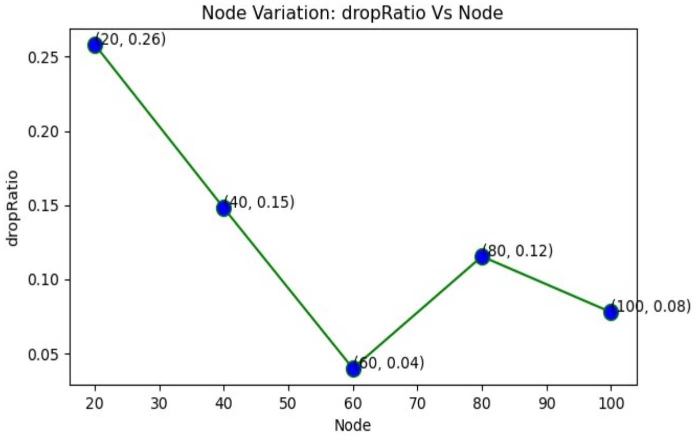
Wireless(802.15.4 – mobile)

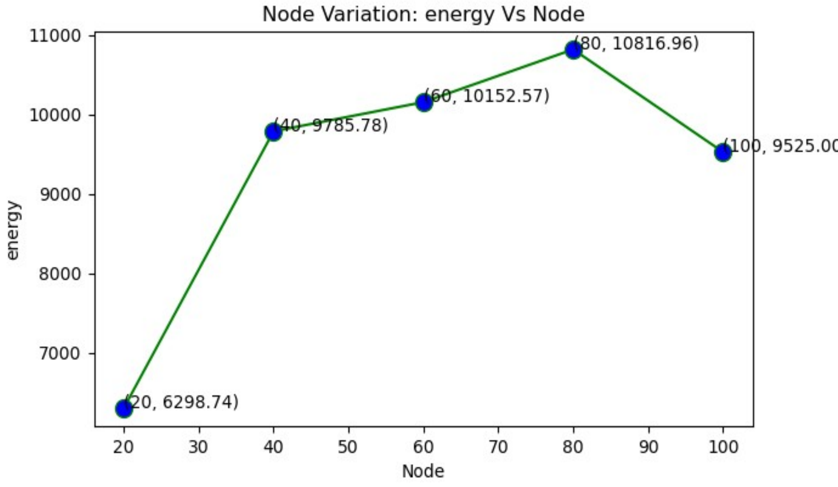
Node Variation:



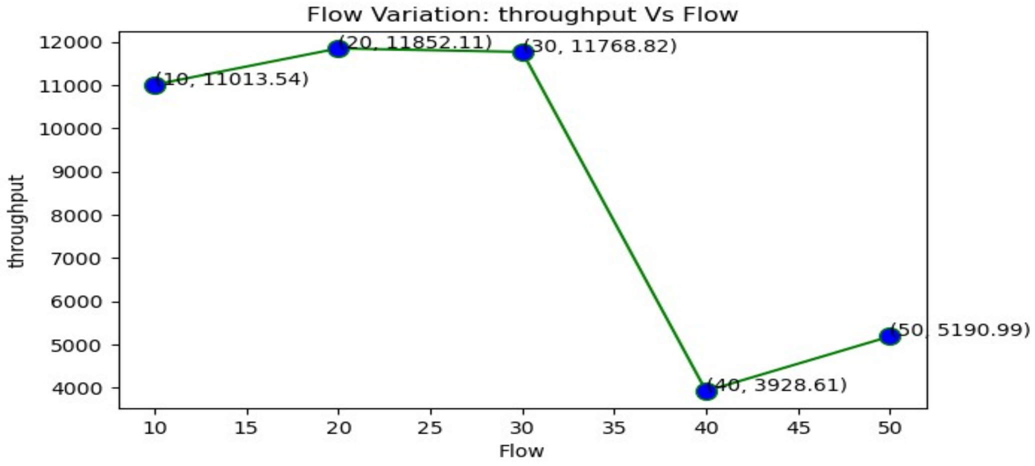


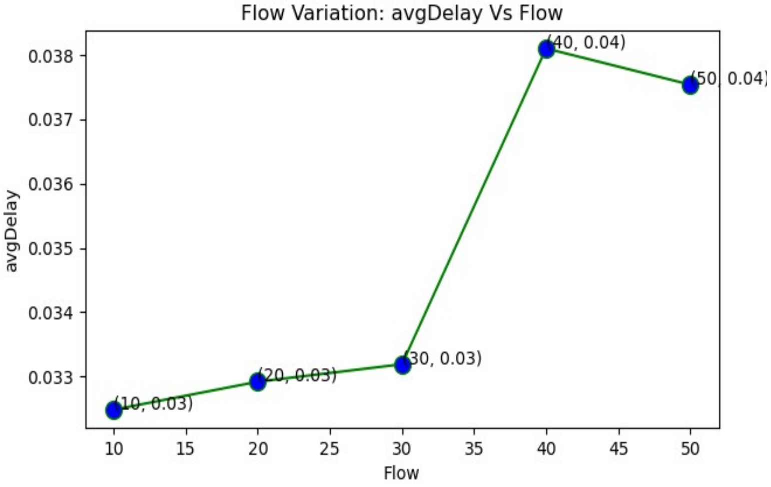


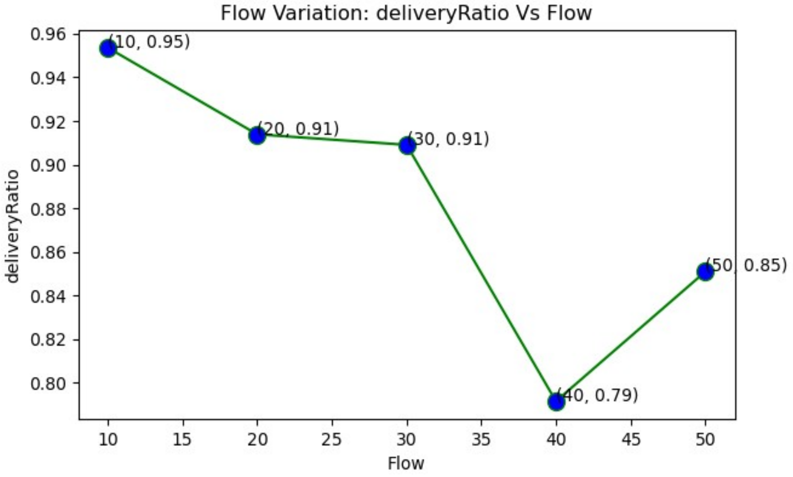


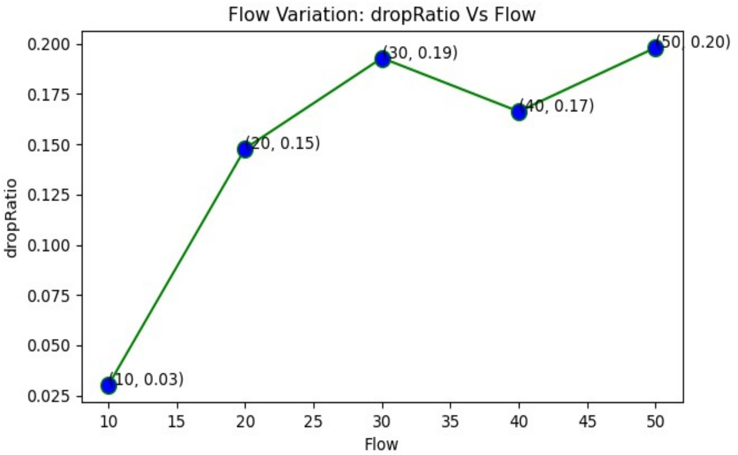


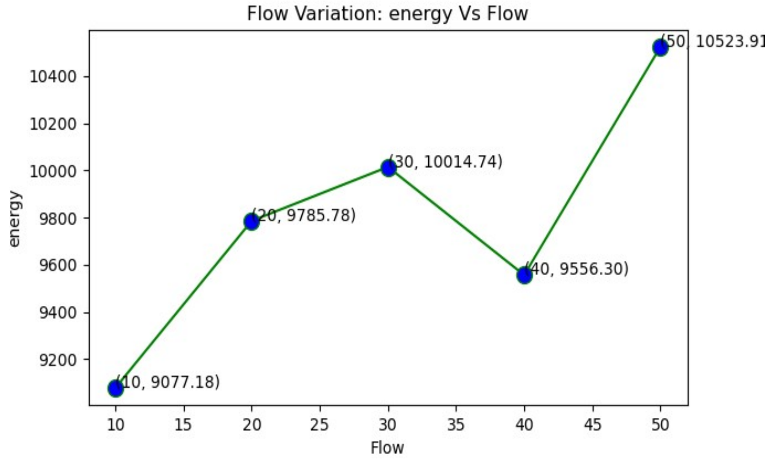
Flow Variation:



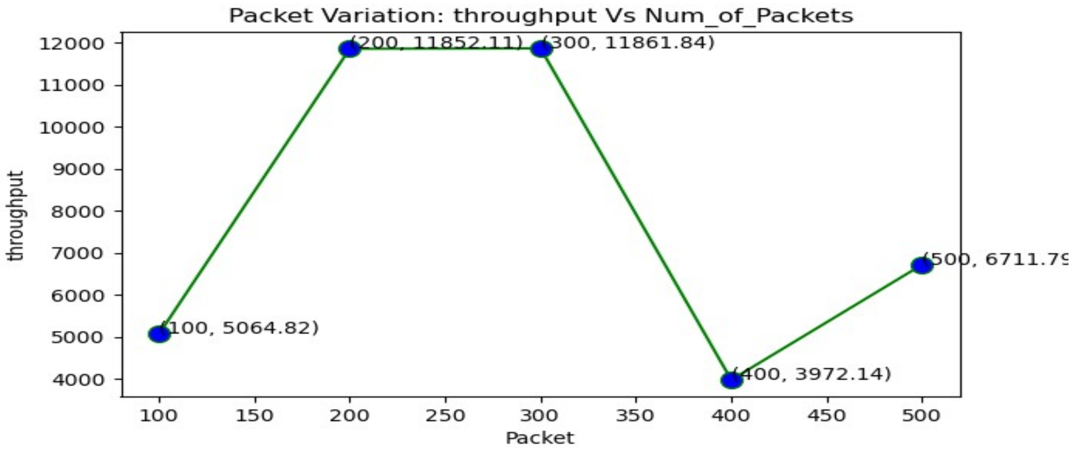


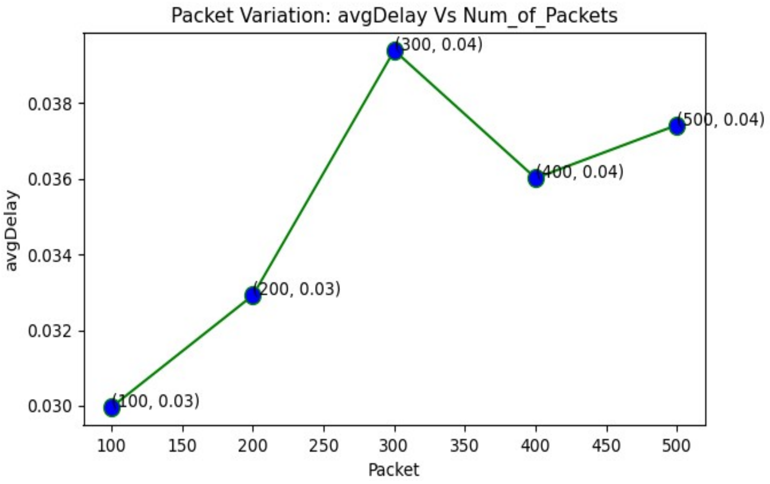


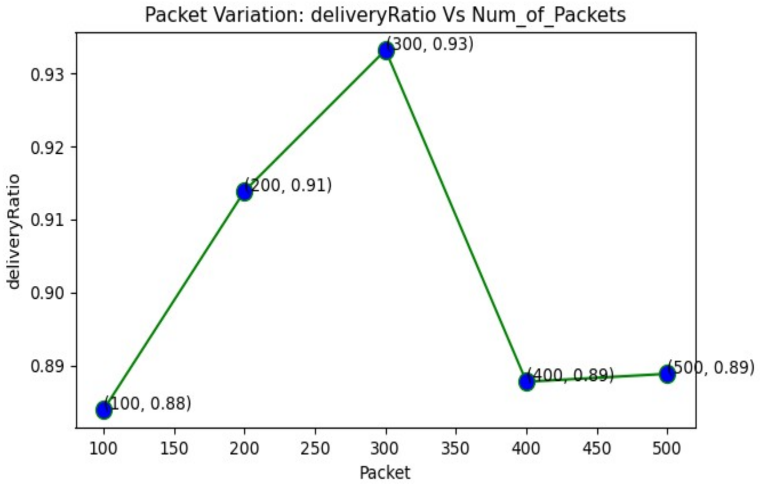


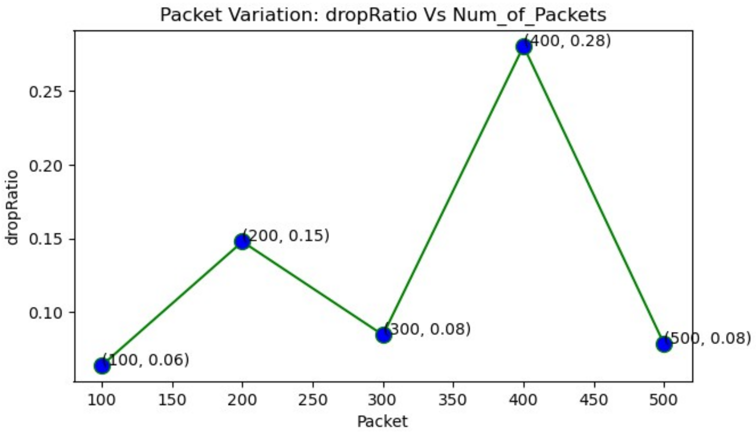


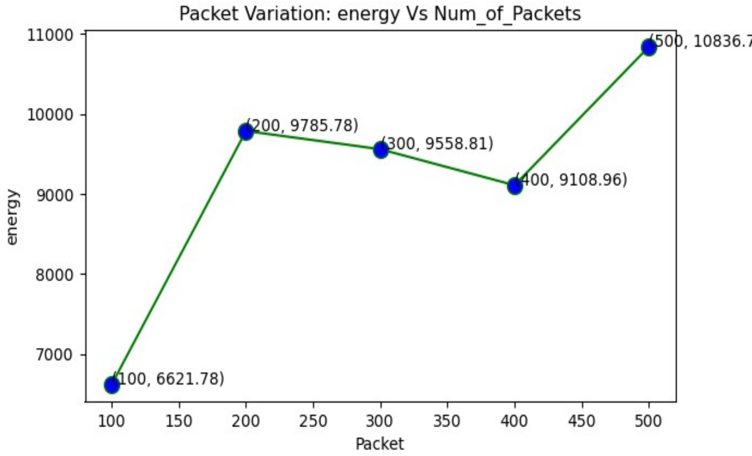
Number of Packet Variation:



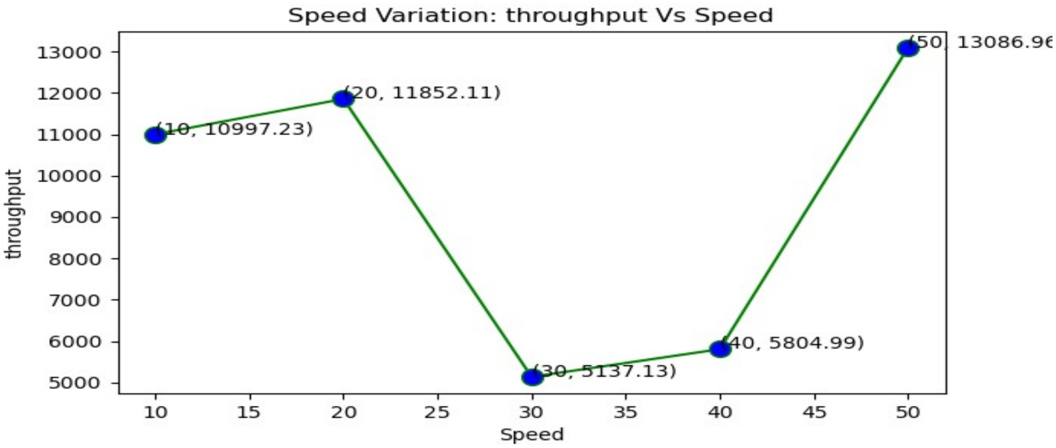


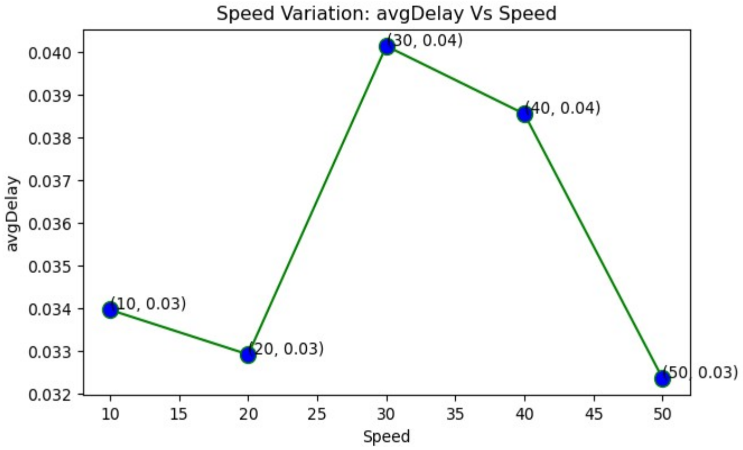


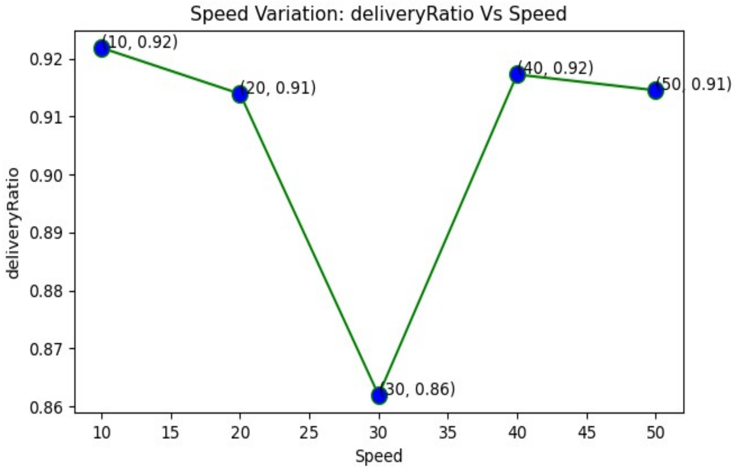


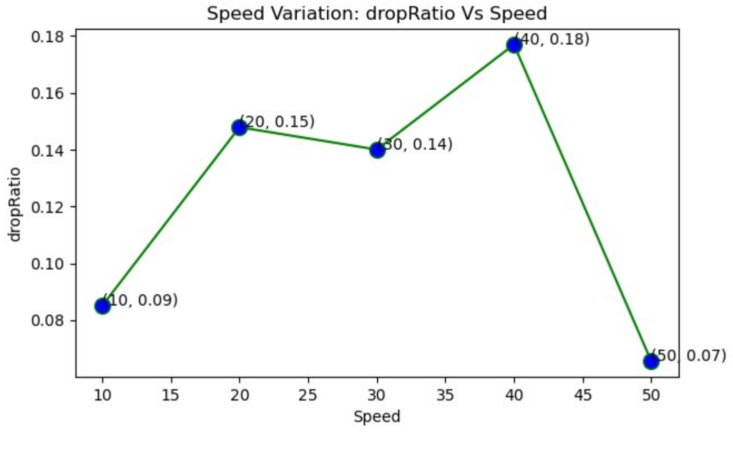


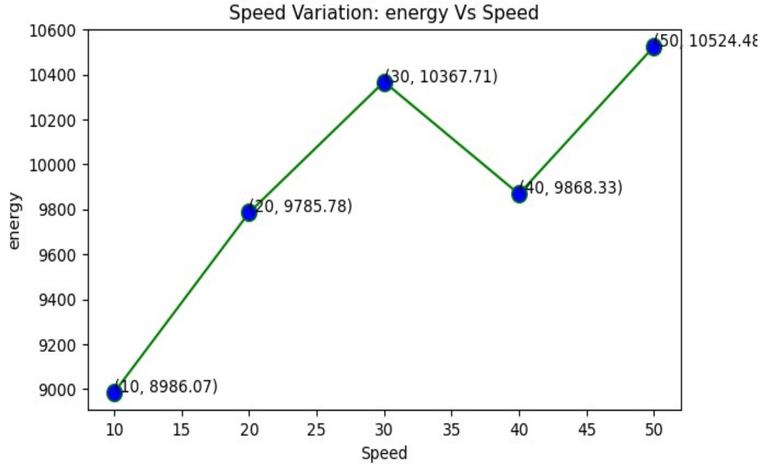
Speed of mobile node Variation:





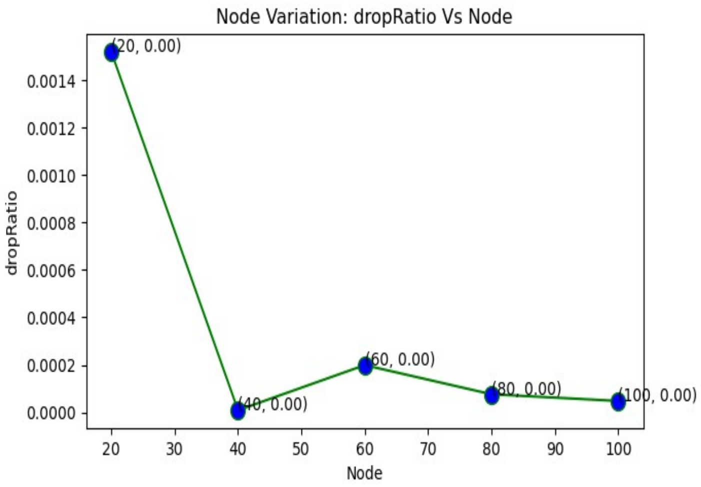




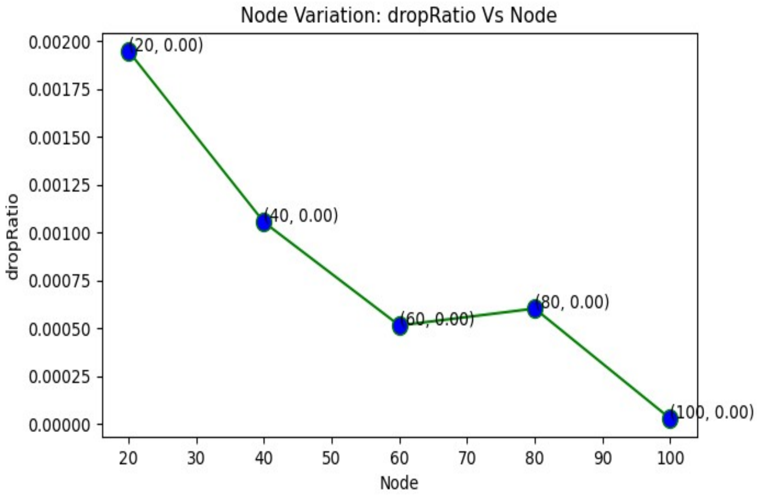


Effect of Modification-1

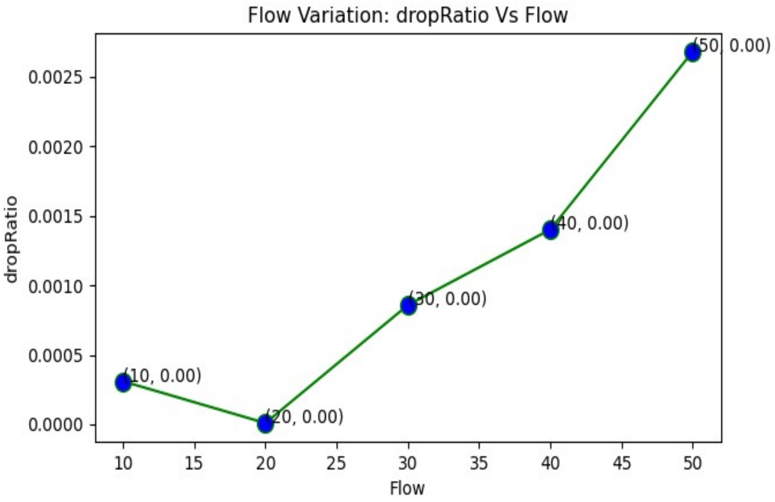
Wired Network:



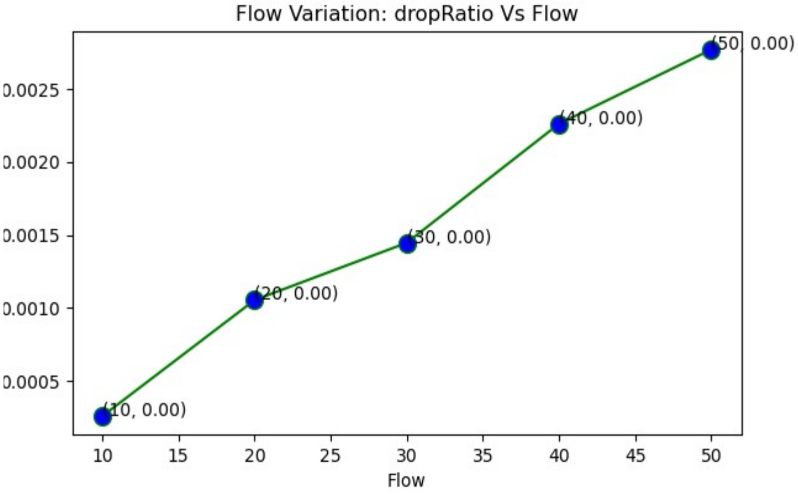
Original



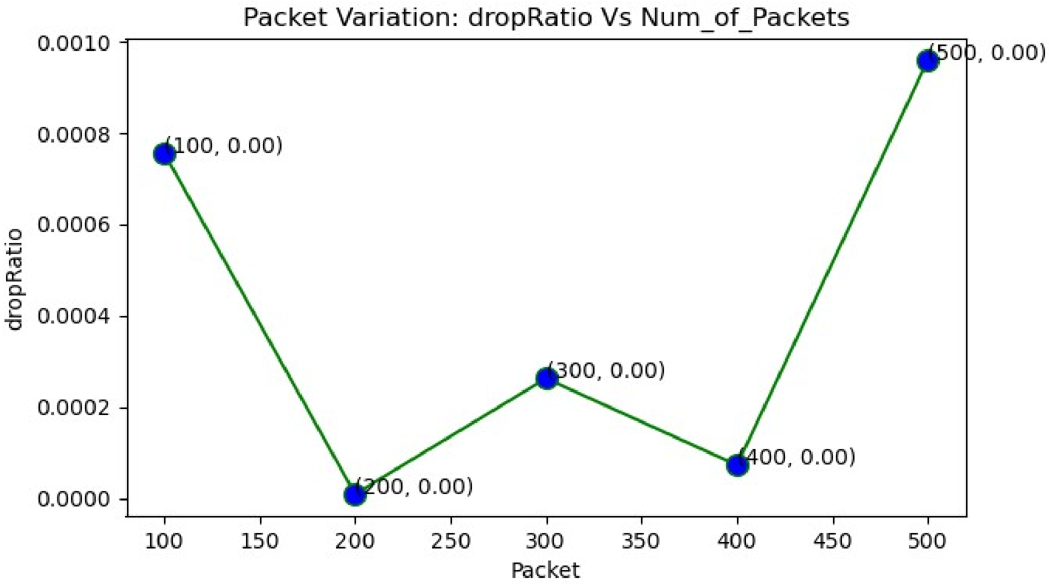
After Modification



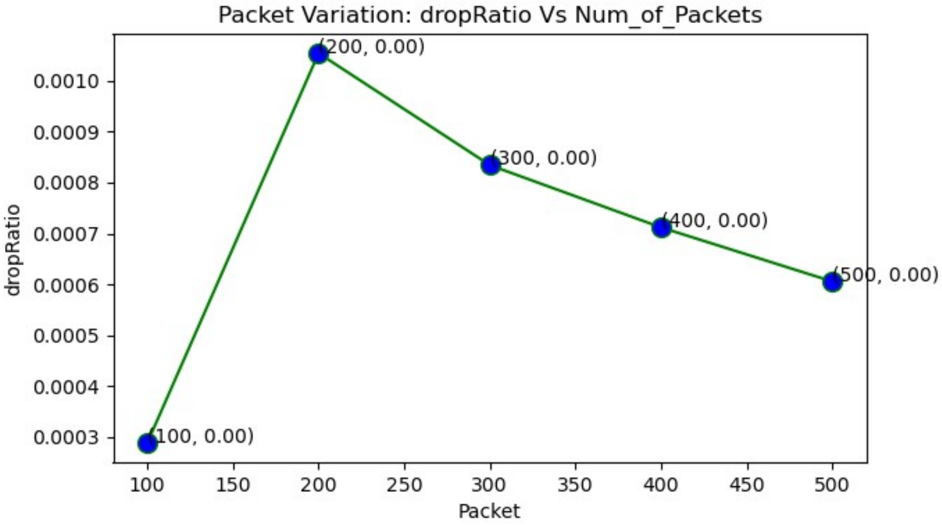
Original



After Modification

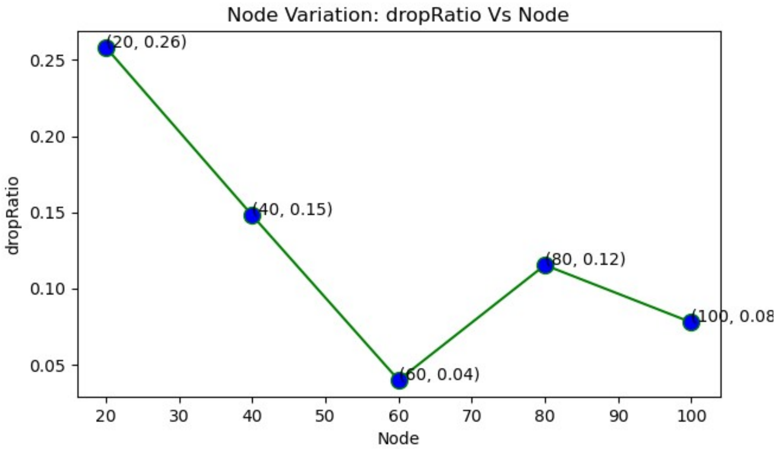


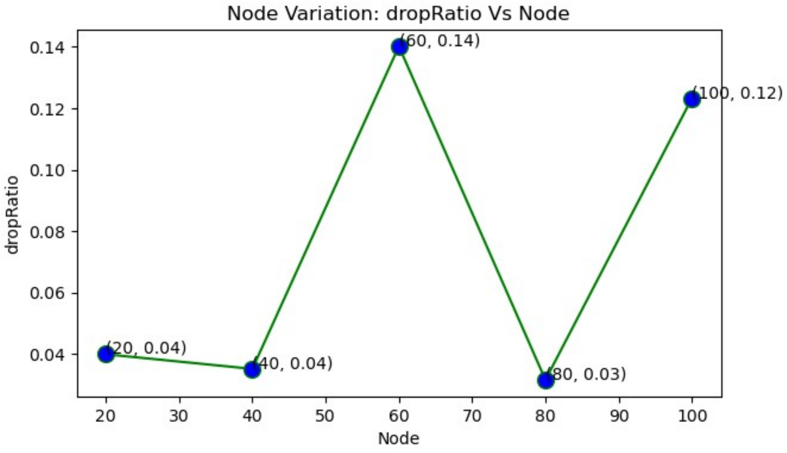
Original



After Modification

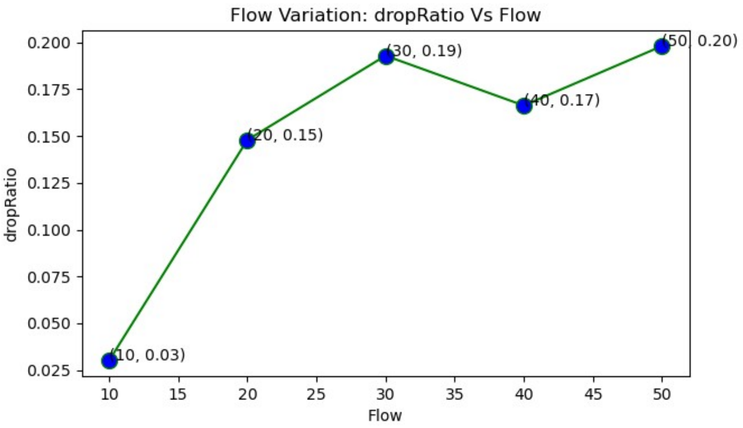
Wireless Network:

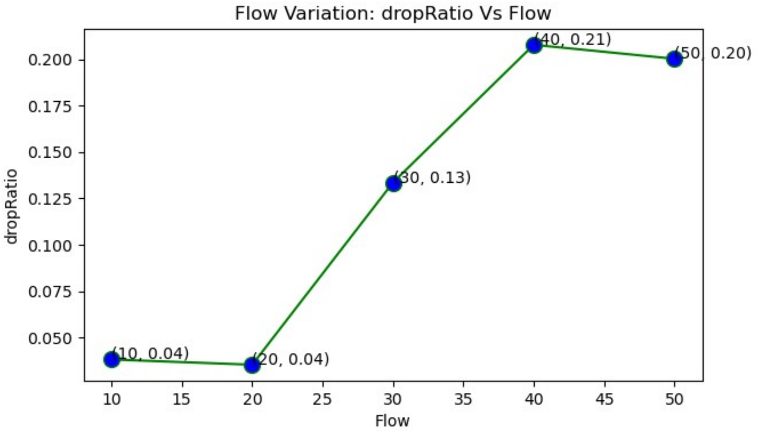




Original

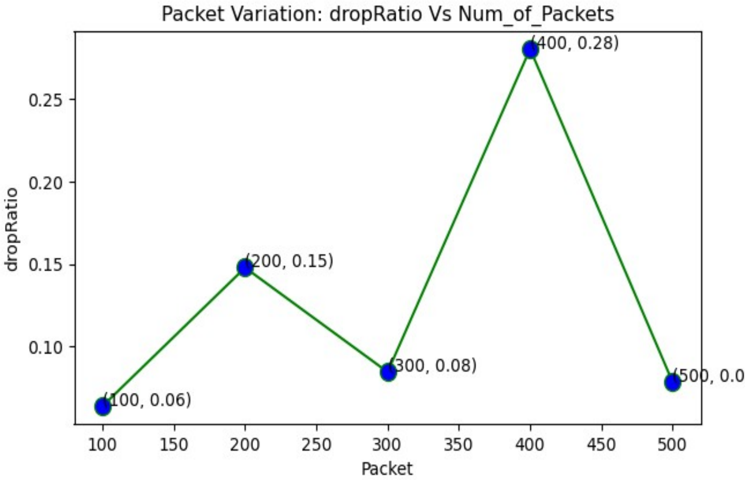
After Modification



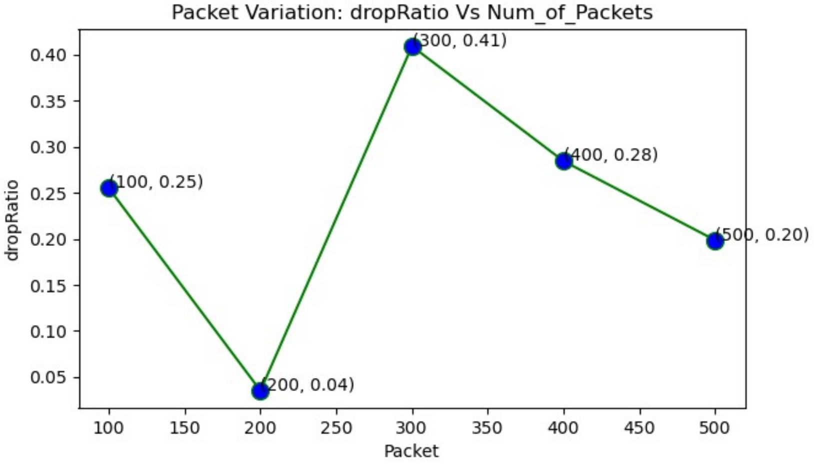


Original

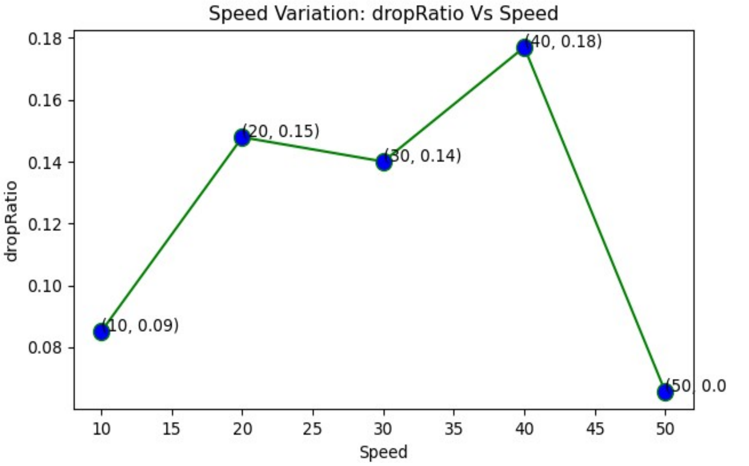
After Modification



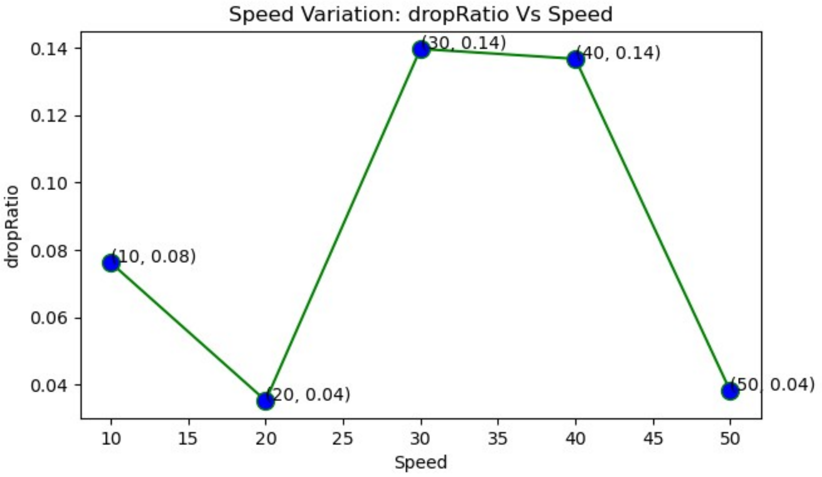
Original



After Modification



Original



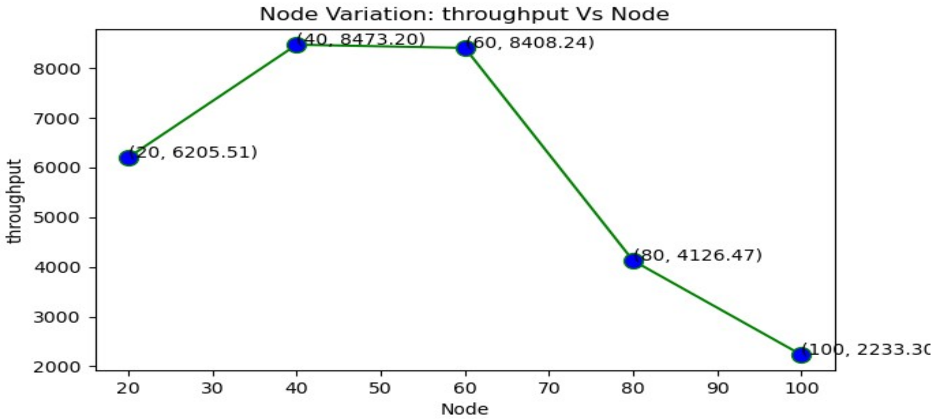
After Modification

Effect of Modification-2

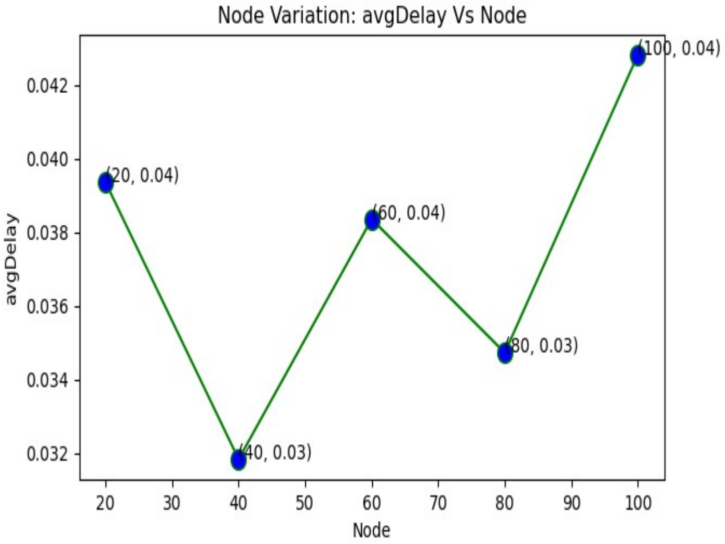
Wireless Network:

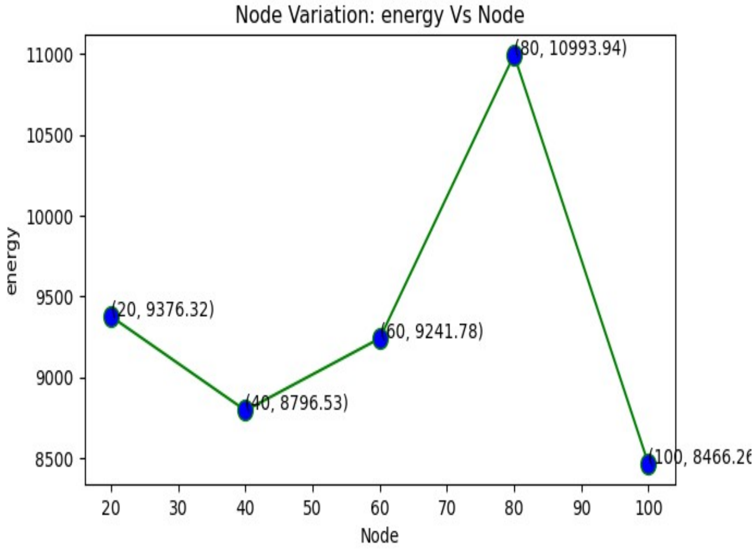
Modified Graphs (802.15.4)

Node Variation:



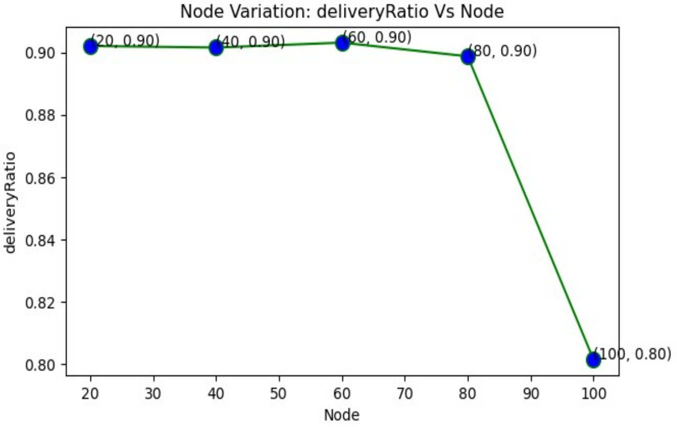
Increased



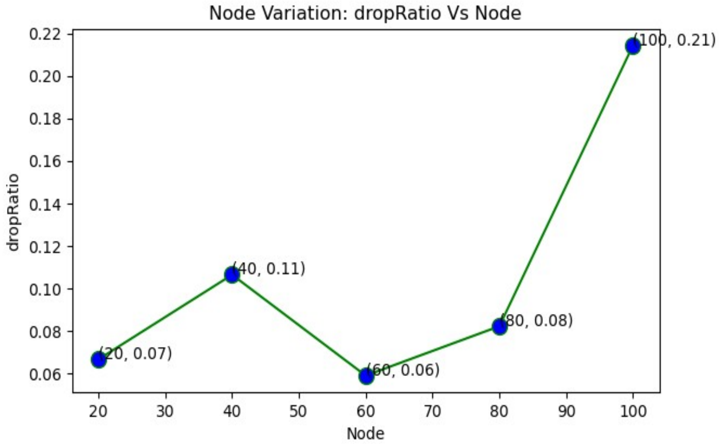


decreased

Intact

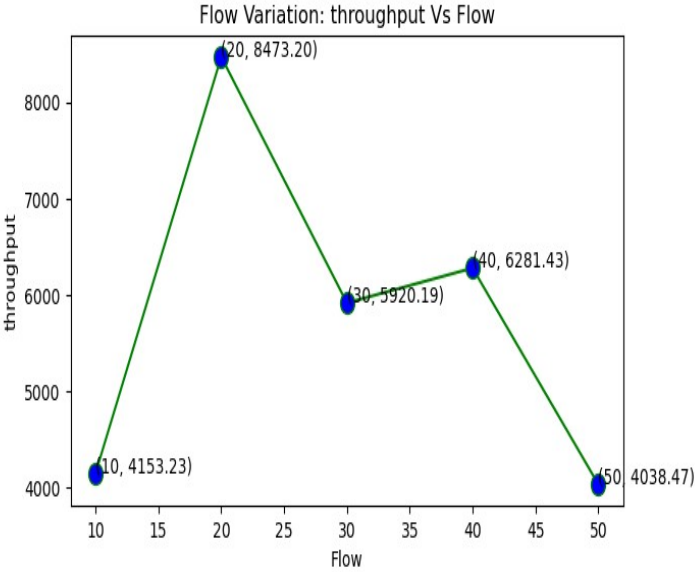


decreased

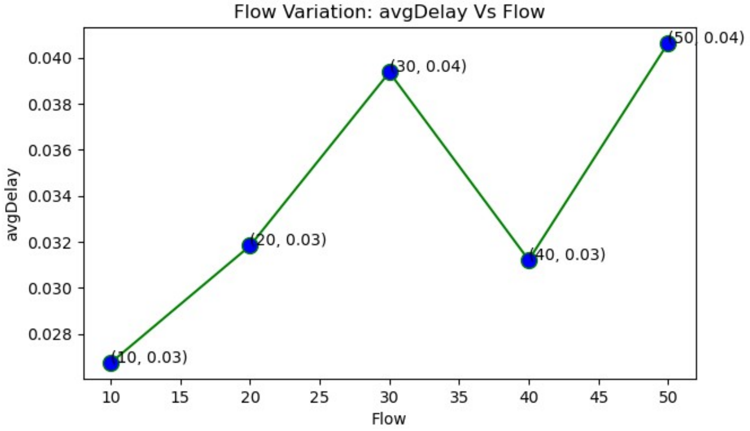


Increased

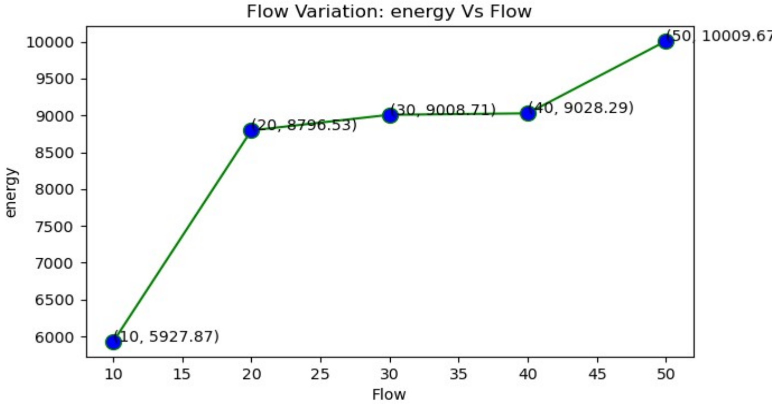
Flow Variation:



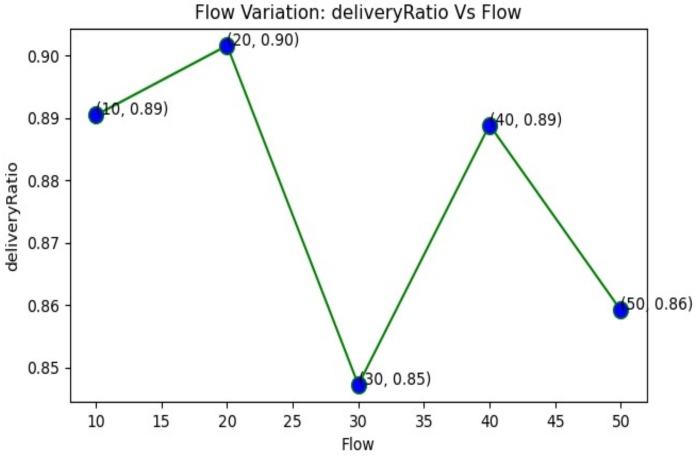
Increased



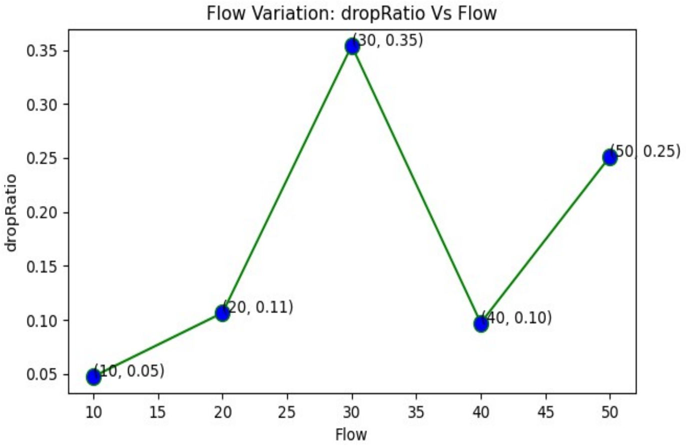
Intact



decreased

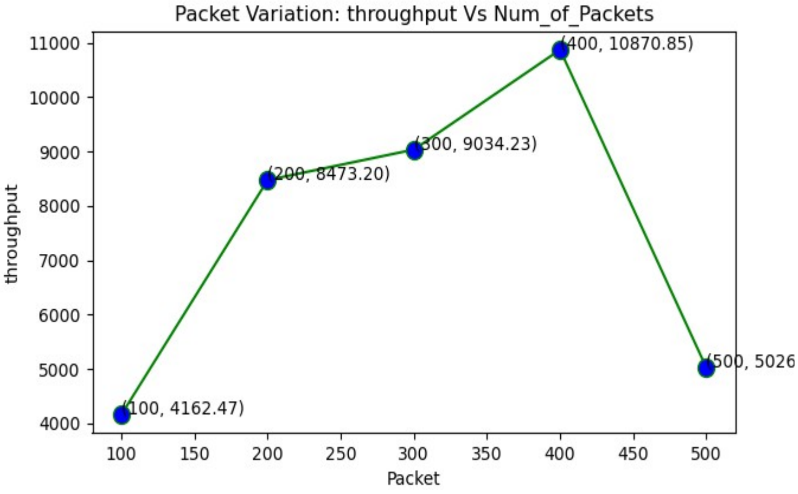


decreased

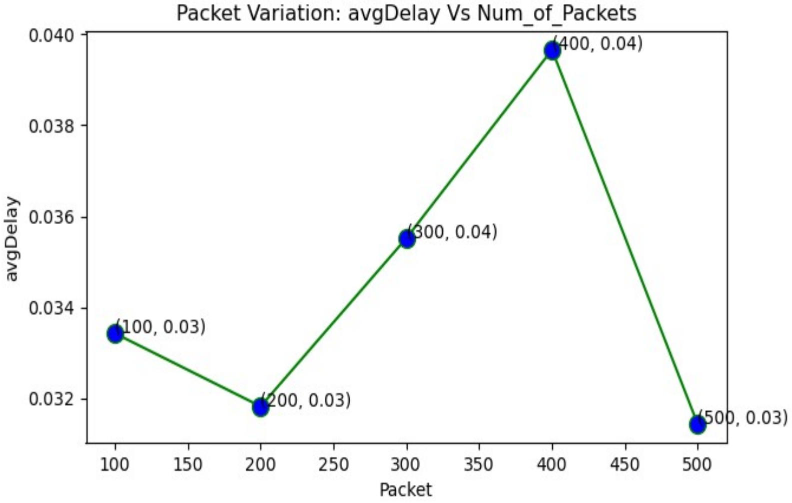


Increased

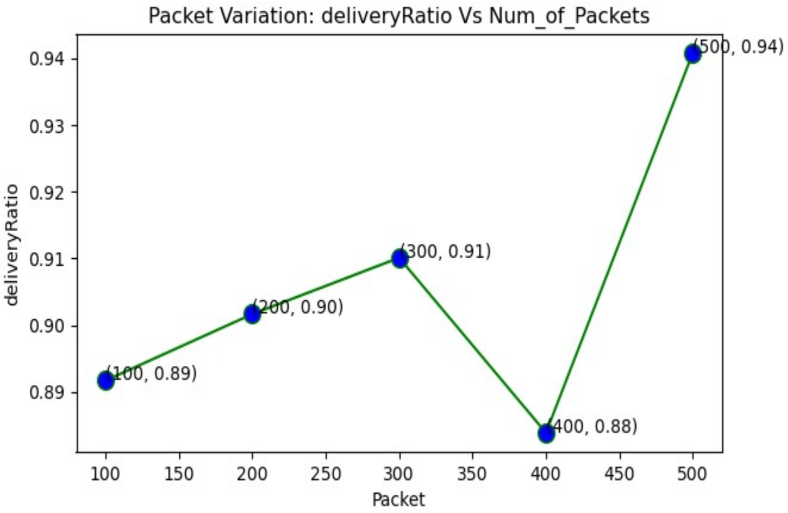
Number of Packet Variation:



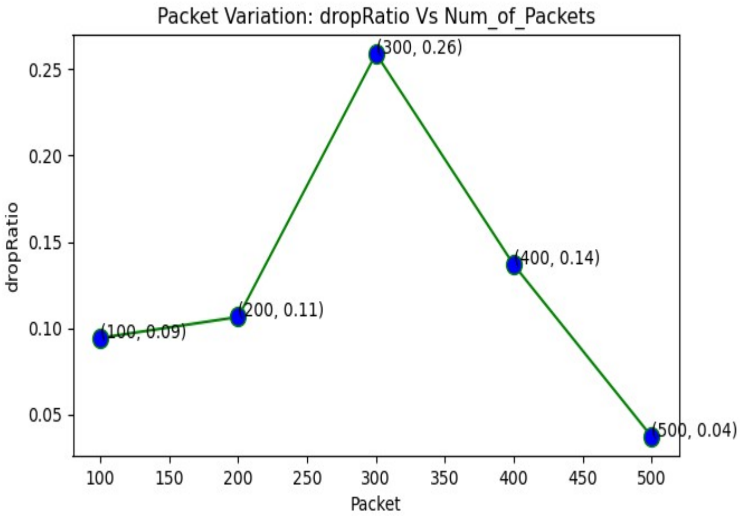
Increased



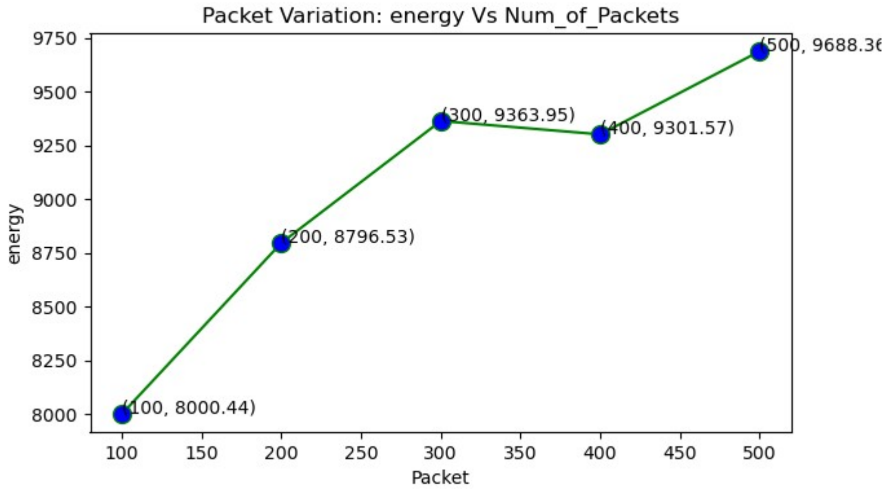
Increased(decreased for 500)



Intact

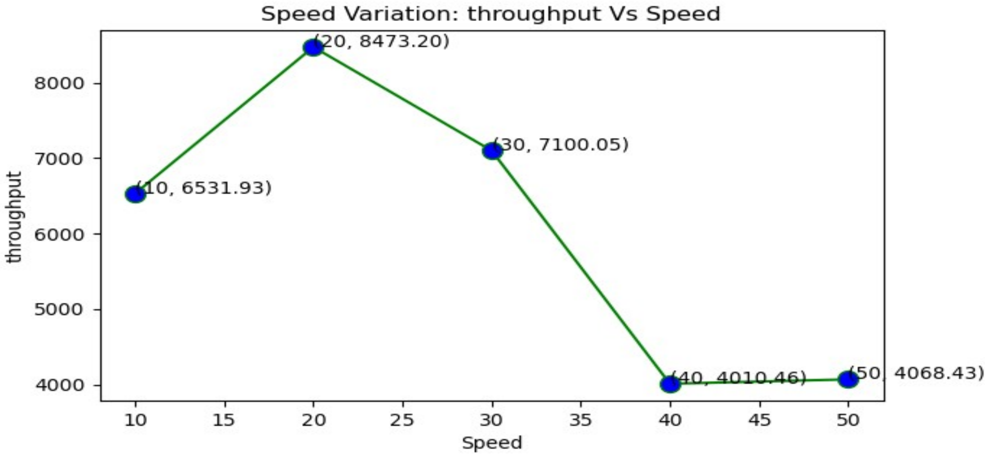


Intact

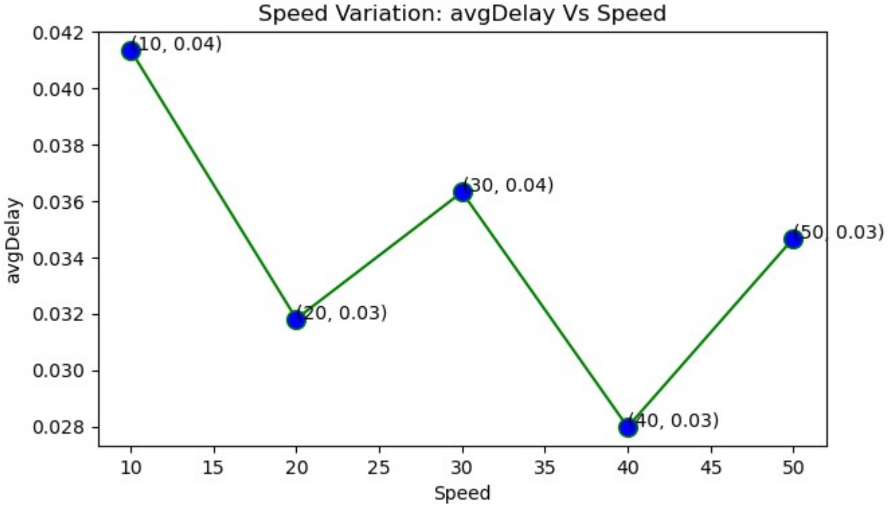


Decreased

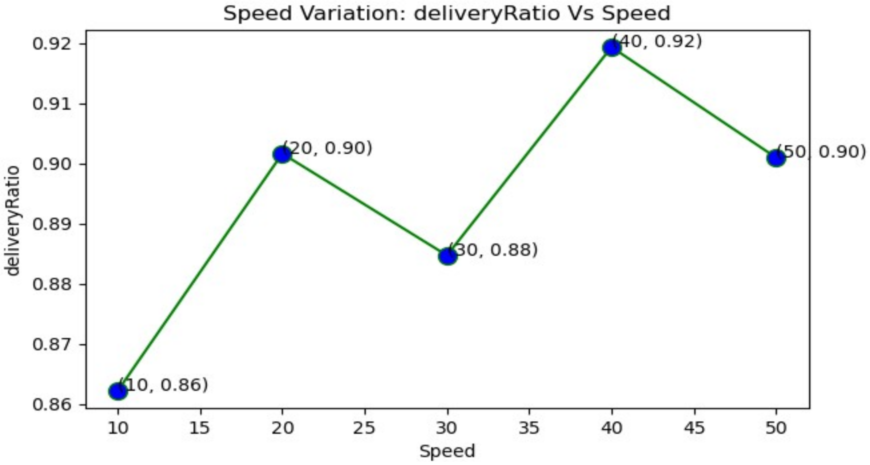
Speed of mobile node Variation:

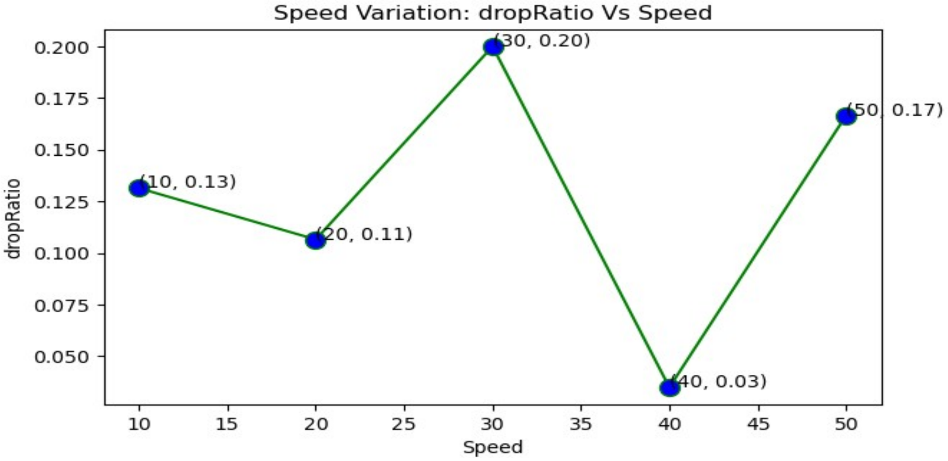


Increased

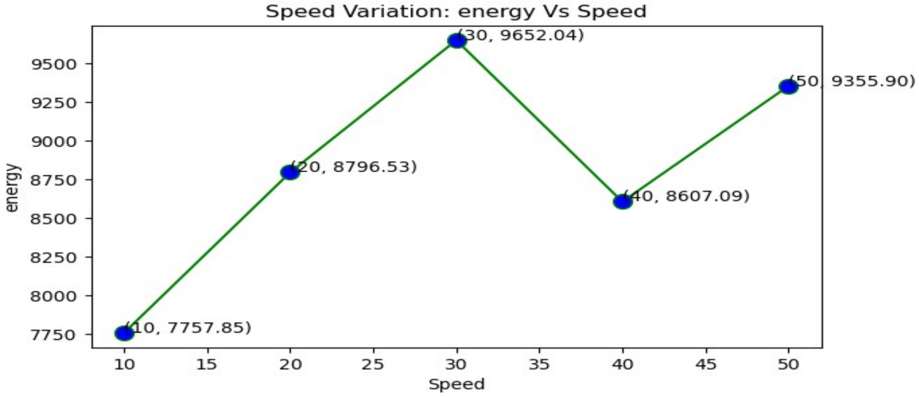


Increased





Increased



Decreased

Summary Findings:

- Packet delivery ratio and throughput are much higher in Wired network than 802.15.4.
- Drop Ratio is much higher in Wireless network than wired networks.
- Dropping randomly instead of from head of the queue does not consistently increase performance(i.e. decrease in drop ratio)
- Change in drop-tail queue could not affect 802.15.4 drop ratio mentionably because most of the packets dropped in 802.15.4 was not due to queue.
- Decreasing the value of “perup_ (time period of update)” and “min_update_period (minimum number of period after which stale routes will be discarded)” has increased performance in respect of delivery ratio and throughput but at cost of energy because of increase in the frequency of updates.
- Modification in the lost_link(Packet* p) function could not improve overall performance significantly.
- Since nodes frequently change position in mobile wireless networks, lost link is very common. So drop ratio is significantly higher.
- After getting a packet with lost link, we are broadcasting a message to all nodes and calling a processUpdate with this packet. It helps convergence faster, so energy consumption reduces and throughput increases. But, since lot of packets are being broadcast, so congestion occurs and drop ratio increases.