Report
NS2 Term Project : Own Modifications on elected paper (TCP-Constant)

Submitted by:

Kazi Ababil Azam

1805077

L-3/T-2

CSE-B1

Modifications made:

The modification I implemented have the following changes to the paper:

1. The paper suggests an RTT calculation before the change of the cwnd. I have omitted the calculation as the Westwood algorithm recalculates the minimum RTT at every dupack.

2. The major change is by not keeping it constant, rather I am increasing it as long as the network scenario is unchanged. So, it defeats the naming, but works better in a few cases.
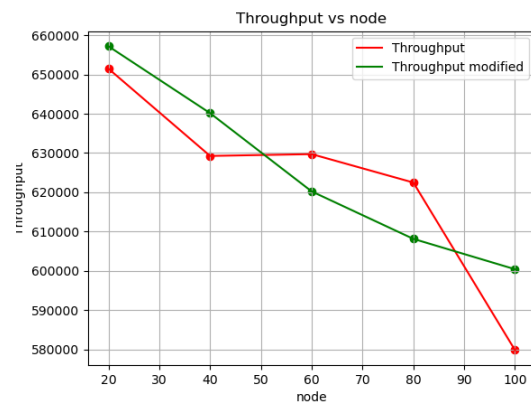
```
        double rtt_change = rtt_archive - rtt_estimate;
        double beta_ = 0.015;

        if ((rtt_archive != 0 && abs(rtt_change)/rtt_archive) > beta_) {
            cwnd_ = (((current_bwe_*(min_rtt_estimate))/((double)(size_*8.0))));
            if (cwnd_ < 1) cwnd_ = 1;
            rtt_archive = rtt_estimate;

        } else cwnd_++;



    // }
```
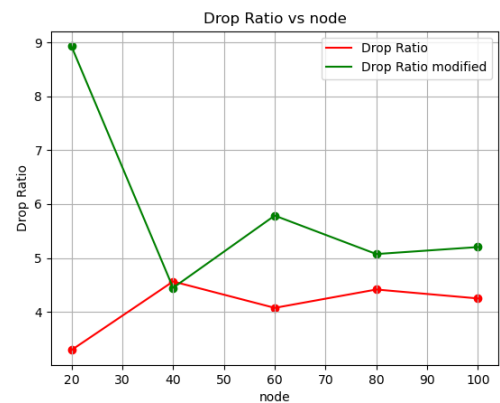
Results with graphs:

1. ## Wireless 802.11 static nodes:
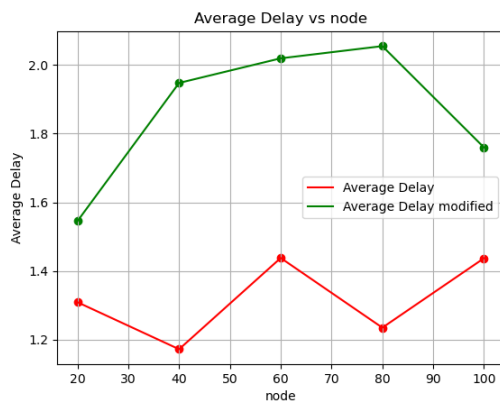
With respect to Number of Nodes:

Network Throughput:

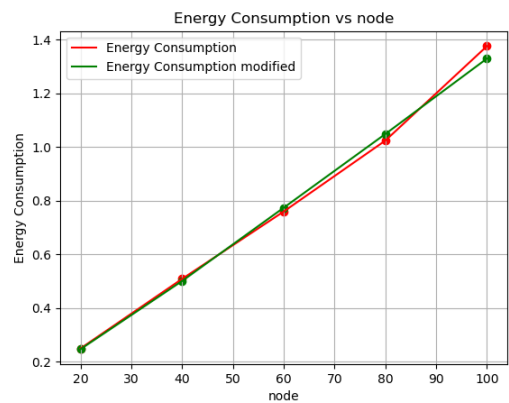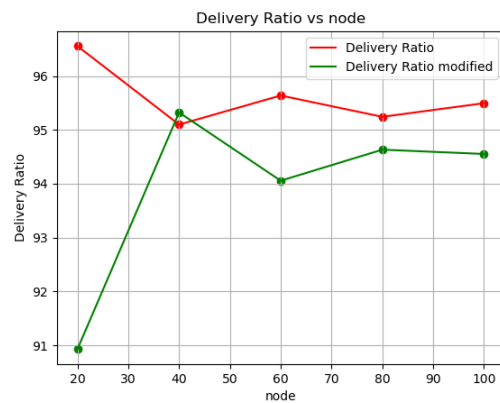

Packet Drop Ratio:
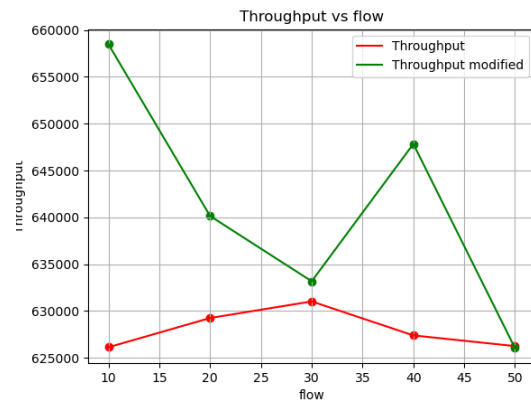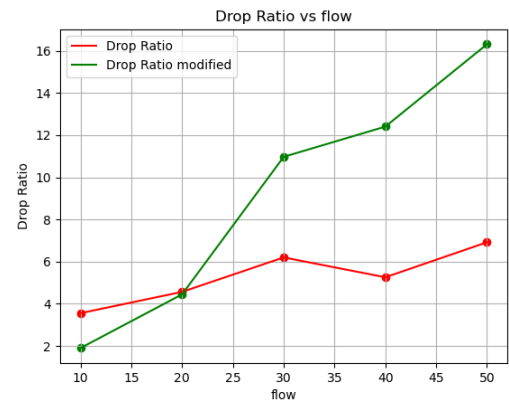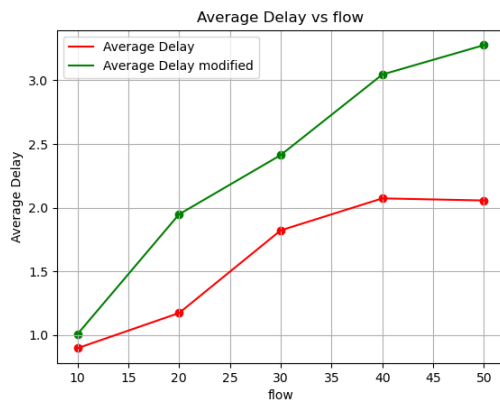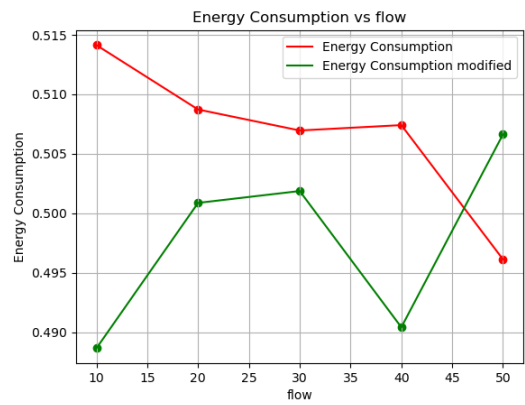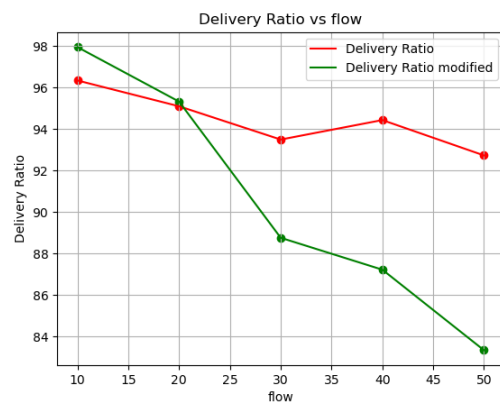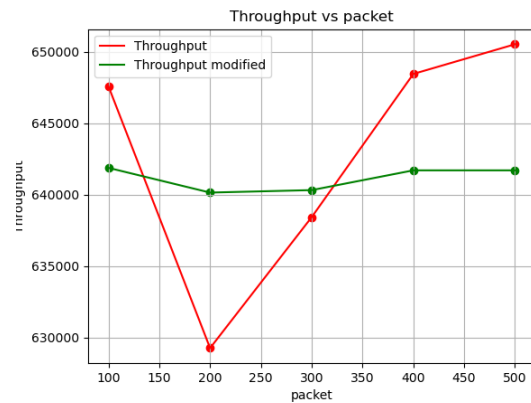

End-to-End Delay:

Energy Consumption:

Packet Delivery Ratio:
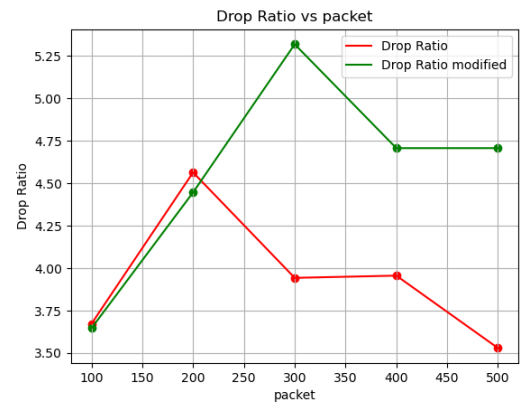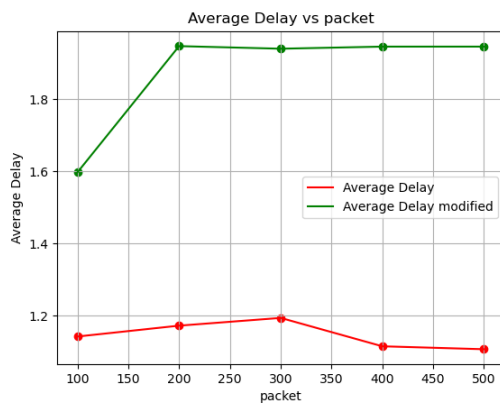
With respect to Number of Flows:

Network Throughput:

Packet Drop Ratio:

End-to-End Delay:

Energy Consumption:

Packet Delivery Ratio:

With respect to Number of Packets per second:
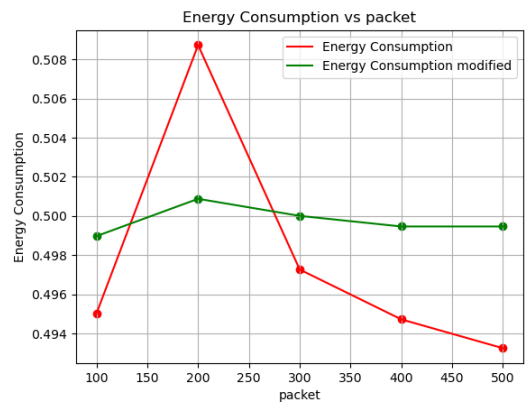
Network Throughput:

Packet Drop Ratio:

End-to-End Delay:
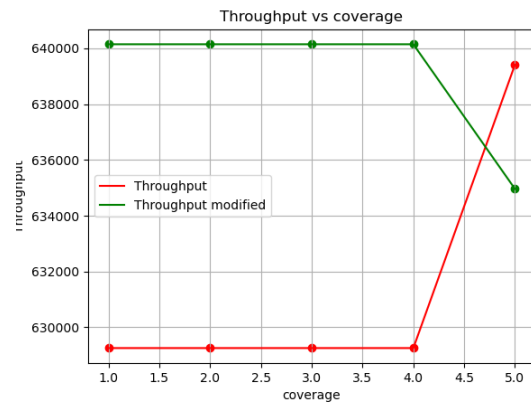

Energy Consumption:
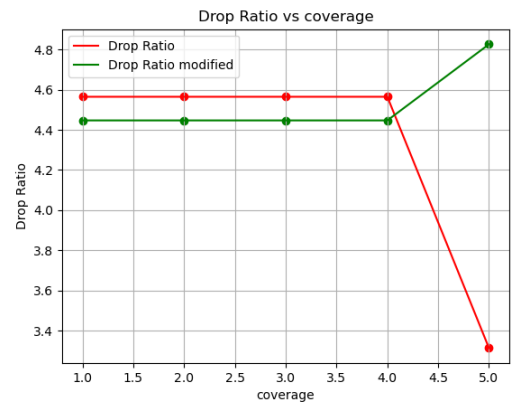

Packet Delivery Ratio:

With respect to Coverage Area:

Network Throughput:

Packet Drop Ratio:

End-to-End Delay:

Energy Consumption:

Packet Delivery Ratio:

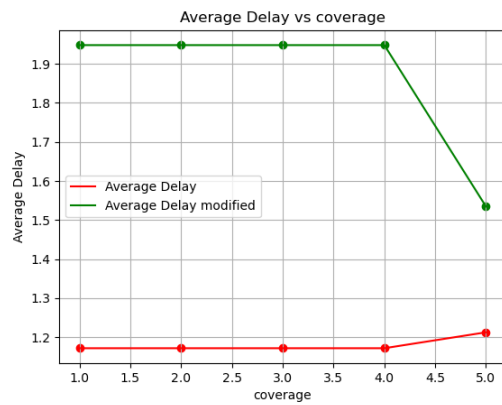## 2. Wireless 802.15.4 mobile nodes:

With respect to Number of Nodes:

Network Throughput:
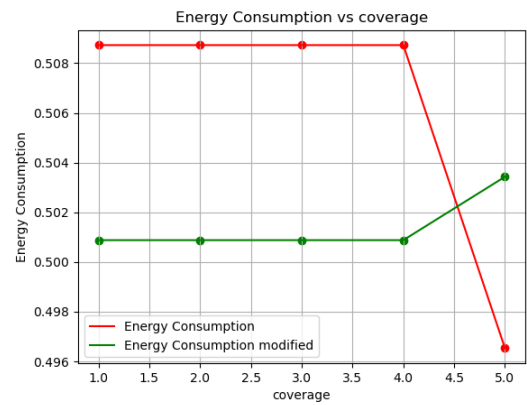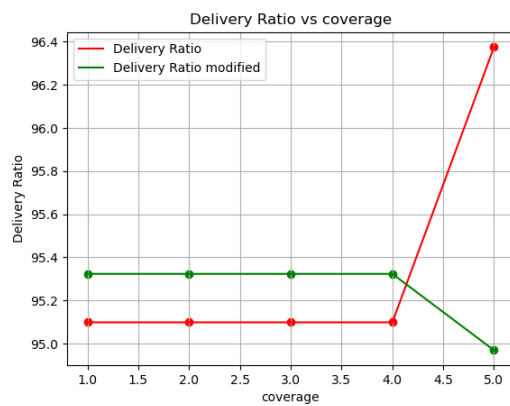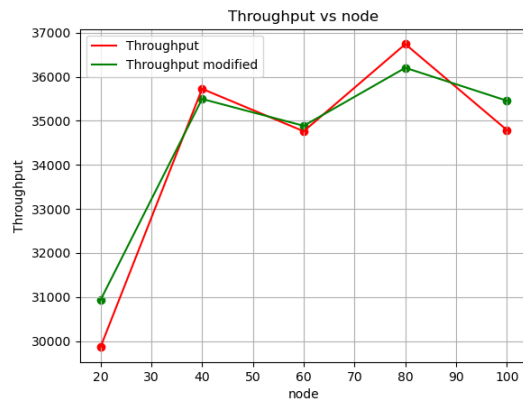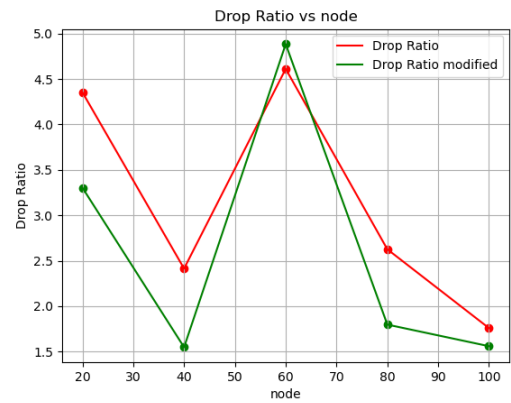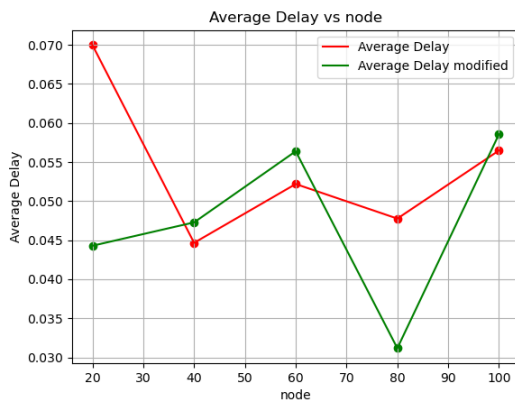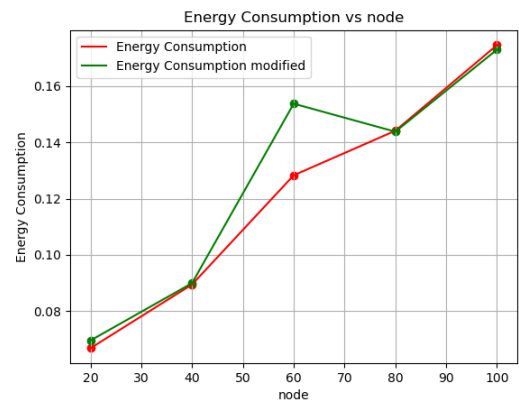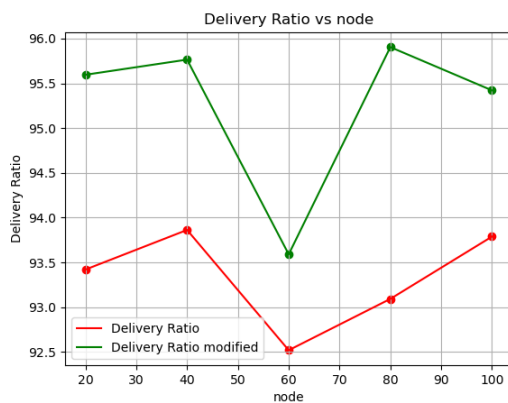

Packet Drop Ratio:

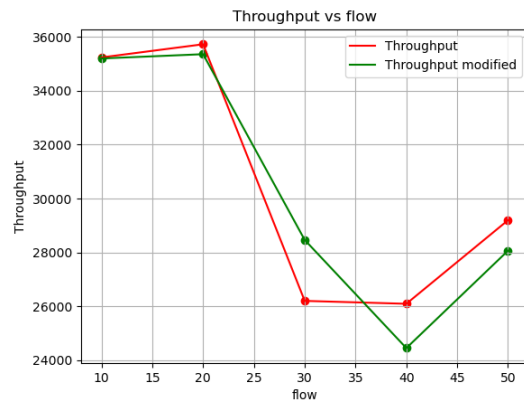
End-to-End Delay:

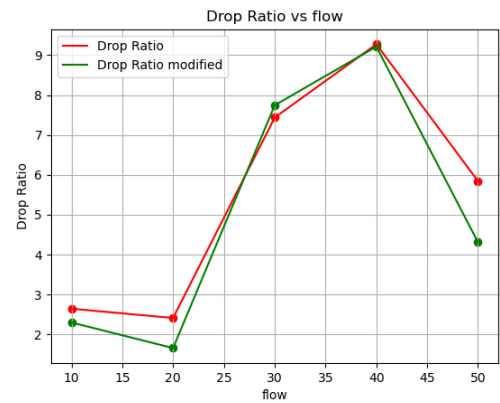Energy Consumption:

Packet Delivery Ratio:

With respect to Number of Flows:

Network Throughput:

Packet Drop Ratio:

End-to-End Delay:

Energy Consumption:

Packet Delivery Ratio:

With respect to Number of Packets per second:

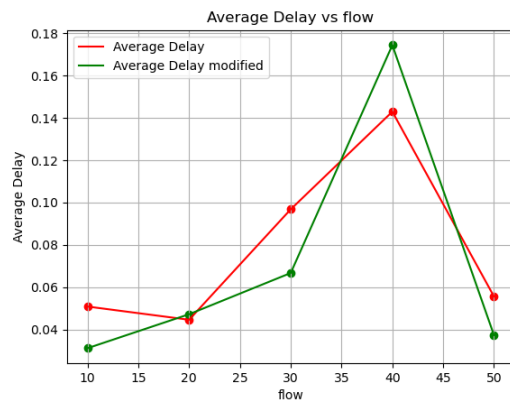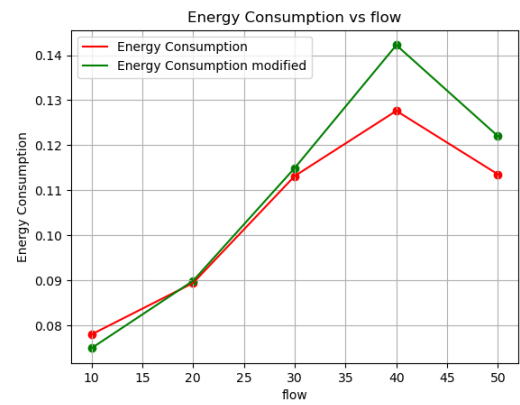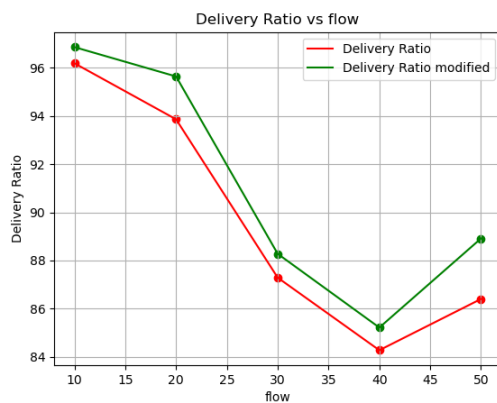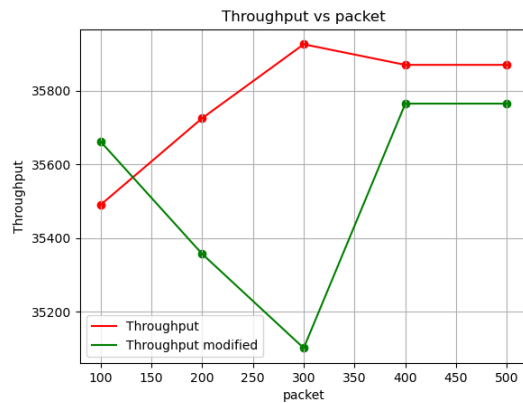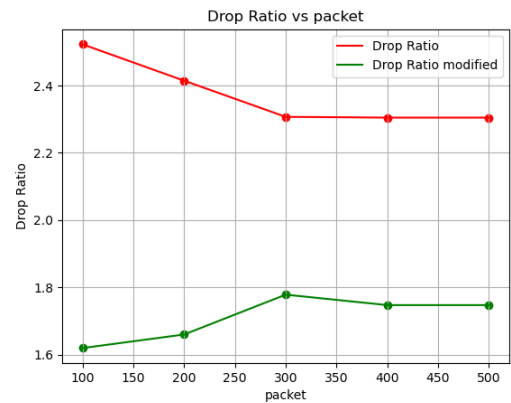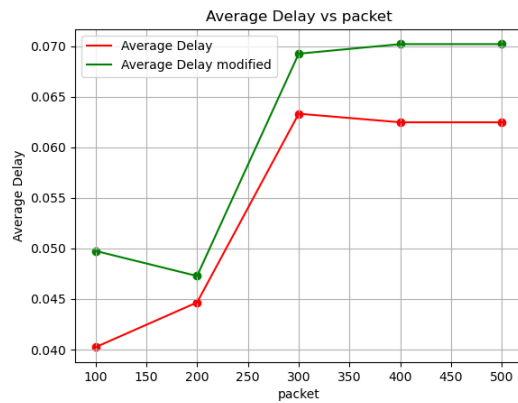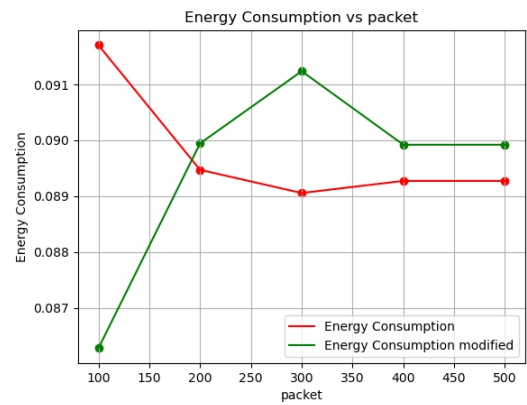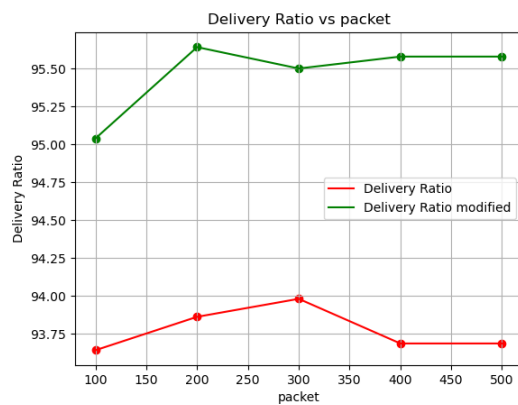## Network Throughput:

Packet Drop Ratio:
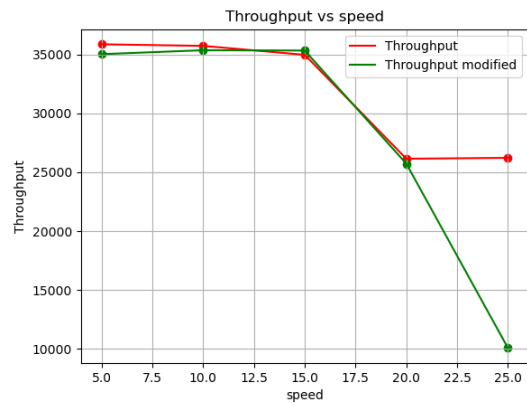

End-to-End Delay:
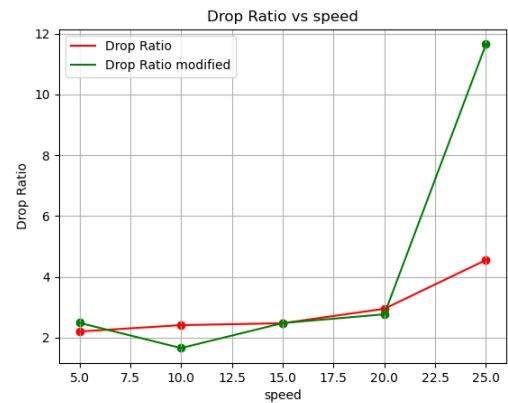

Energy Consumption:
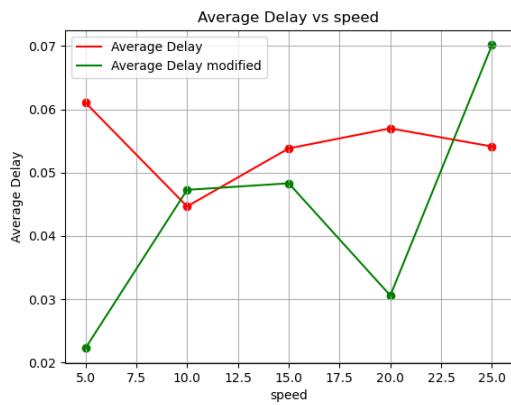

Packet Delivery Ratio:
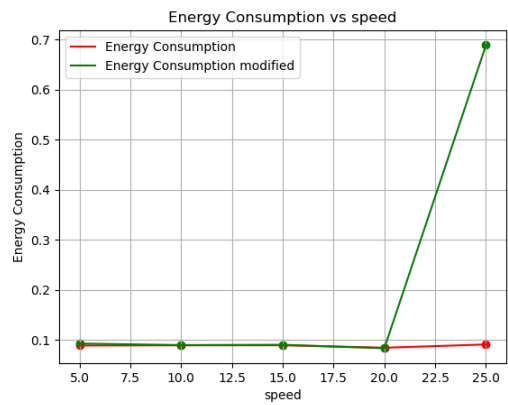
With respect to Node Speed:

Network Throughput:

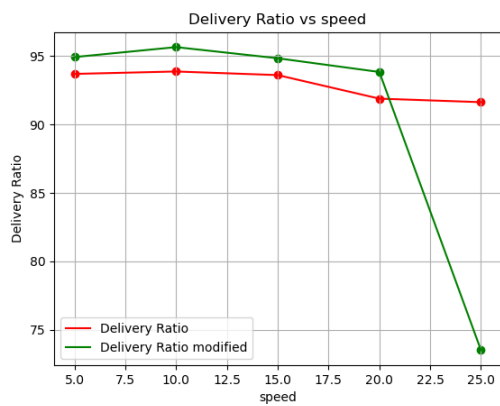## Packet Drop Ratio:

End-to-End Delay:

Energy Consumption:

Packet Delivery Ratio:

Summary findings after own modifications:

The modifications applied by myself from the intuition that as long as the fair share is allotted for the TCP sender, it can use the full potential of the allotted bandwidth by increasing the window. The increment will go on until the network scenario is changed due to the congestion created with the increasing windows of each sender.

My simulations prove my intuition wrong, as the performances still degrade for higher parameters. It also deals with the fact that all the senders increasing the cwnd is also causing continuous changes in the network. So the recalculation is done more often than the increment, causing a sort of deadlock.

Scenario-specific modifications deal with a lot of fixed parameters, which do not ensure improvement in the case of even a slight change in those parameters. The conclusion of the project is that the proposed congestion control mechanism fails to show improvement in general cases of wireless networks.