NS2 Simulation

Course Name: Computer Networks Sessional (CSE 322)
Lab Group: A2
Zahin Wahab
Student ID: 1505031

Mursalin Habib
Student ID: 1505049

Overview:

We are simulating two different networks in NS2 and measuring different metrics.

Network Topologies under simulation:

1. Wired Network
2. Wireless 802.15.4 (static)

Parameters under variation:

I. Number of nodes (20,40,60,80,100)

ii. Number of flows (10, 20, 30, 40, and 50)

iii. Number of packets per second (100, 200, 300, 400, and 500)

iv. Speed of nodes (5 m/s, 10 m/s, 15 m/s, 20 m/s, and 25 m/s) [Only in case of having mobility]

v. Coverage area (square coverage are varying one side as Tx_range, 2 x Tx_range, 3 x Tx_range, 4 x Tx_range, and 5 x Tx_range) [Only in case of having static nodes only]

Metrics measured:

a. Network throughput

b. End-to-end delay

c. Packet delivery ratio (total # of packets delivered to end destination/total # of packets sent)

d. Packet drop ratio (total # of packets dropped / total # of packets sent)

e. Energy consumption [for wireless nodes]

Extra Metrics Measured:

a. Per Node Throughput (for both wired and wireless simulations)
b. Queue Size variation over time (only wired simulation)

Modifications made in the simulator:

I. RTO (Retransmission Timeout) Calculation:

Instead of using fixed weights for smoothed round trip time and round trip time variance, we use adaptive weights resulting in values that are more robust.

II. Adding a New Congestion Control Mechanism for TCP:

To avoiding packets losses, we included a new mechanism by using previous value of congestion window.

III. Random Early Detection (RED) Queue:

Maximum Drop Probability calculation is changed. Drop probability is calculated using the following linear function:

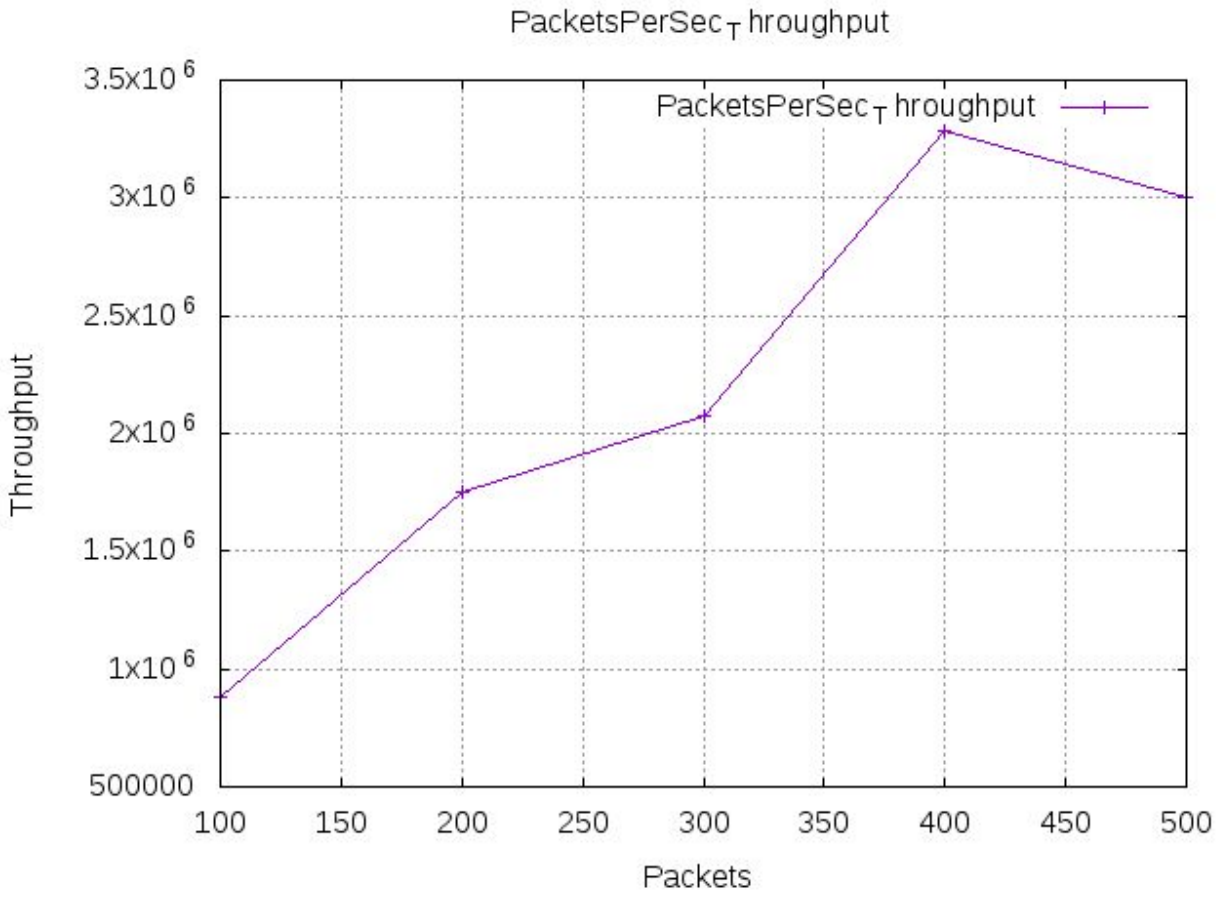
Pb = Pmax*(avg – th_min)/(th_max-th_min)

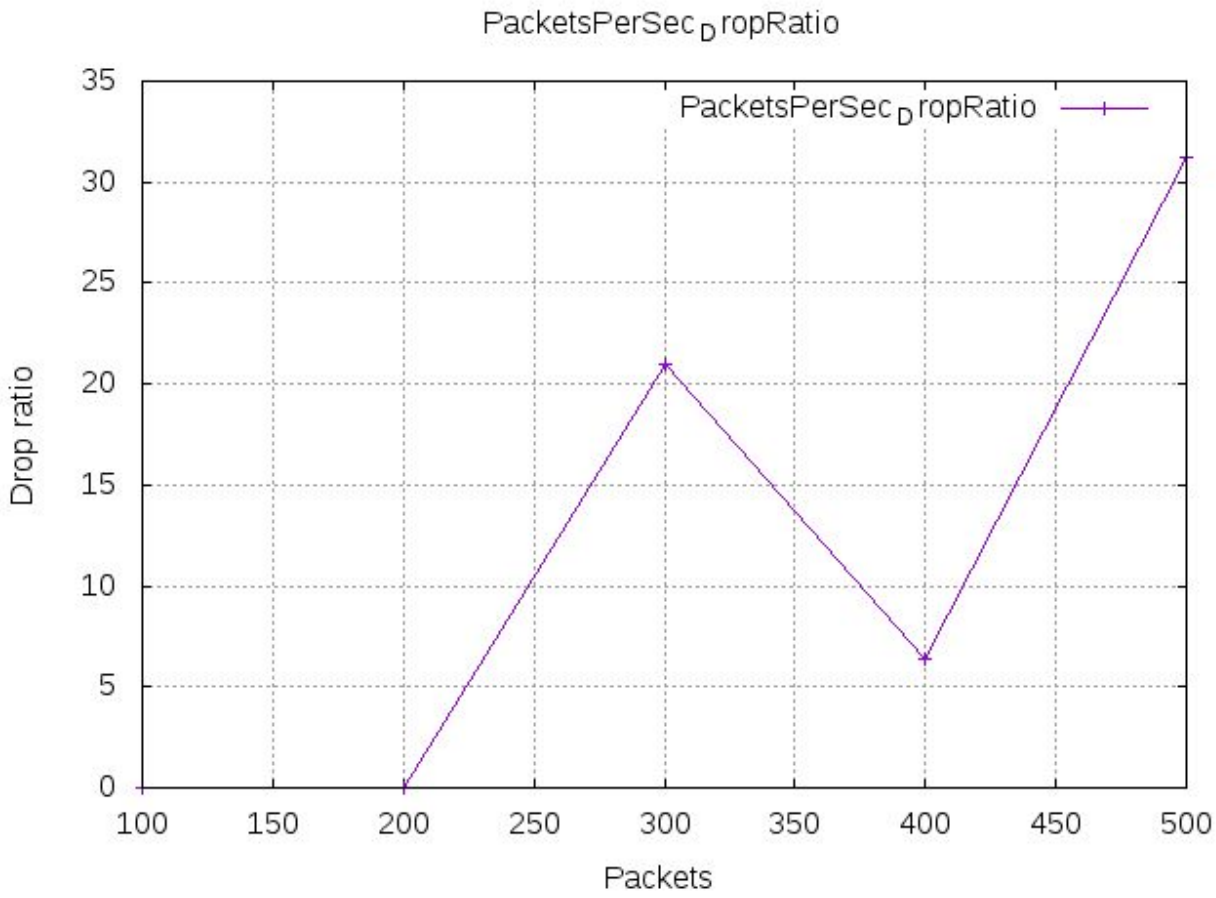
We added a slightly different piecewise function in place of the above function. This keeps avg. queue size less than actual RED Queue.

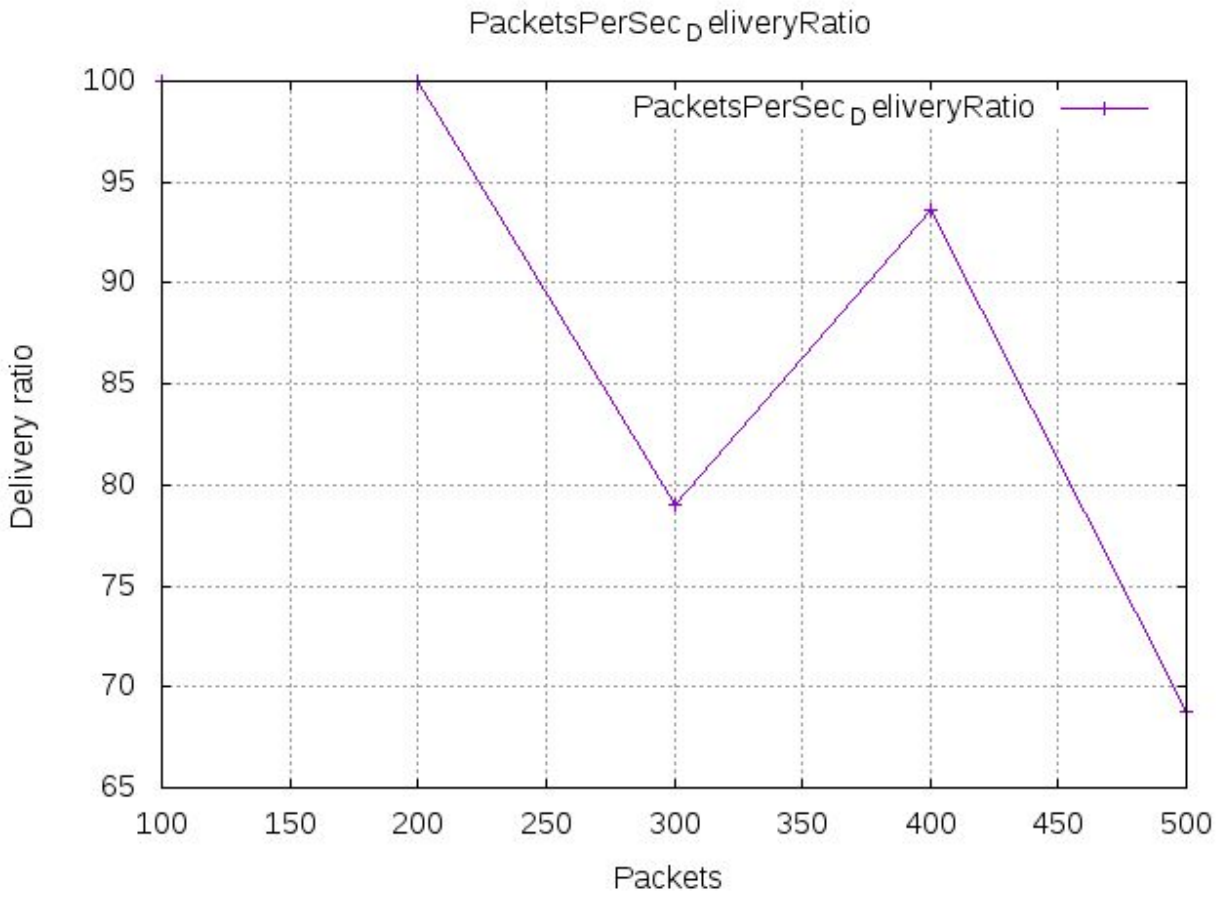
IV. Modifications in the AODV routing protocol:

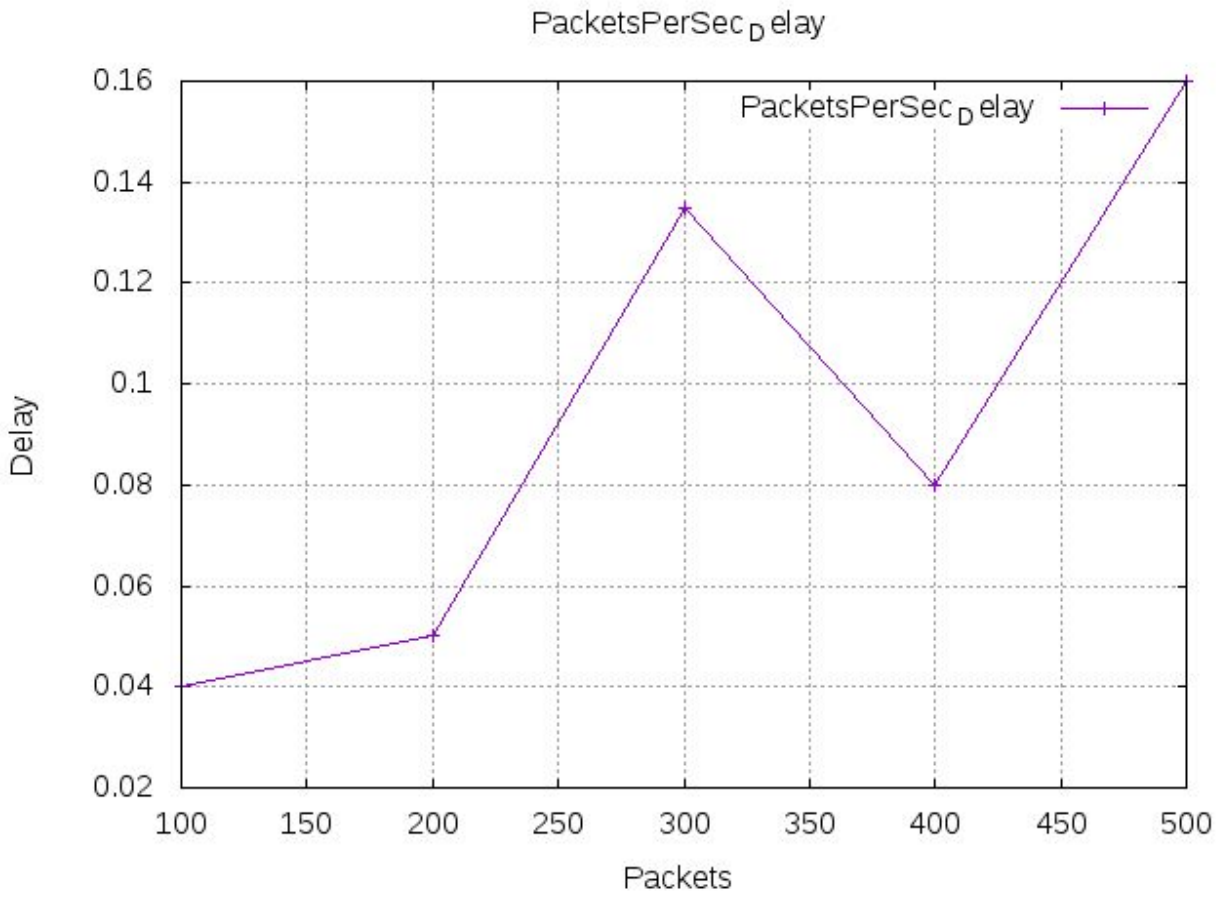
Added support that prevents unnecessary retransmission of RREQ packets. This combats congestion.

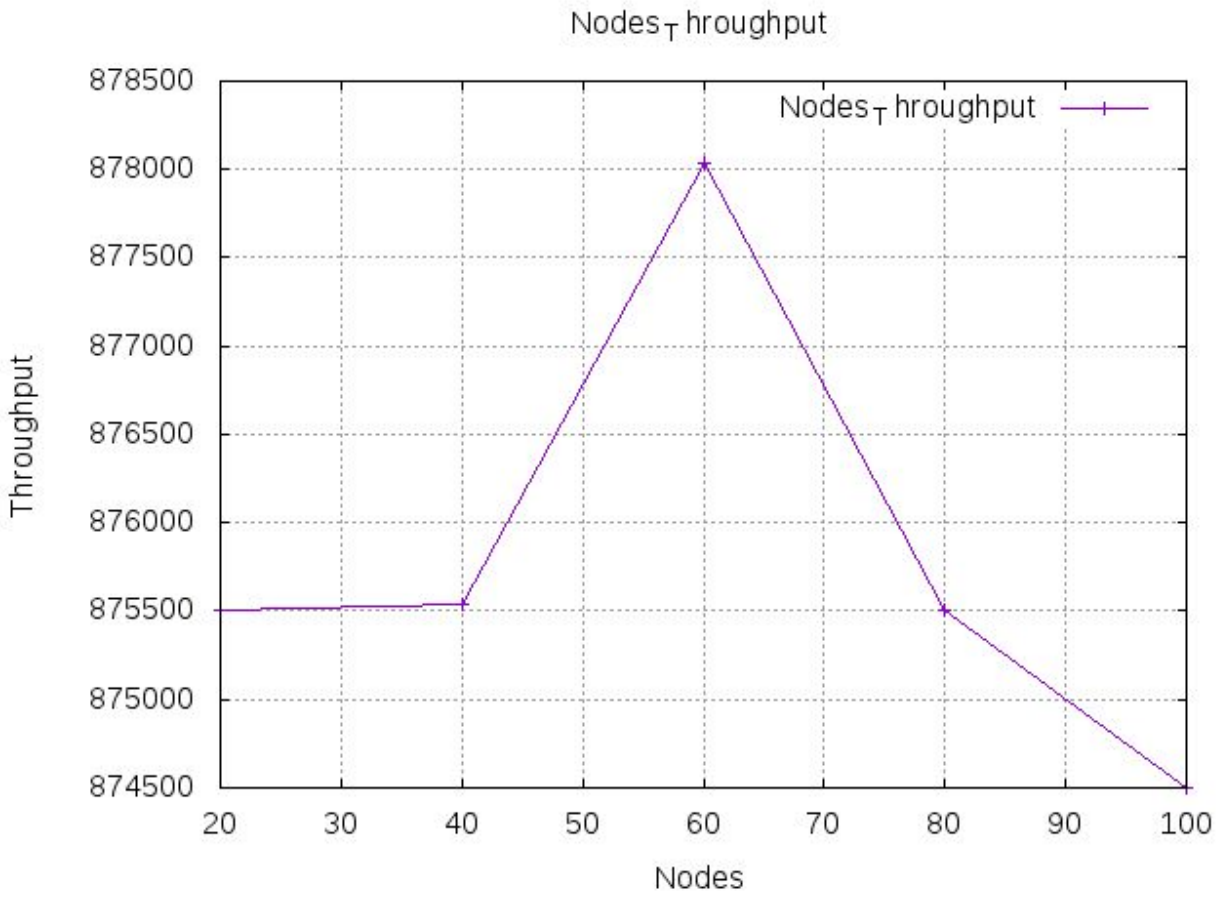
WIRED NETWORK

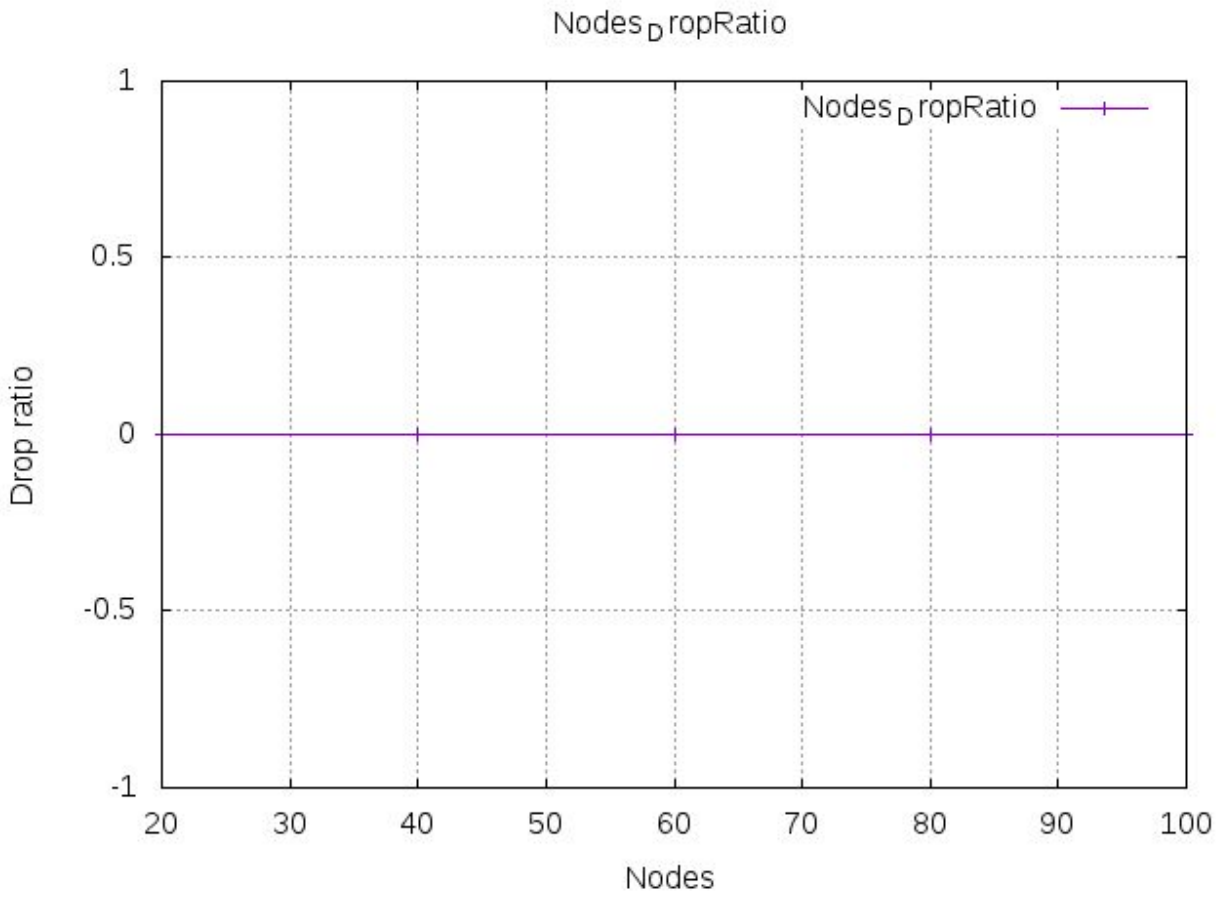

PacketsPerSec T hroughput
PacketsPerSec T hroughput
3.5x10 6
3x10 6
2.5x10 6
2x10 6
1.5x10 6
1x10 6
500000
Throughput
100
150
200
250
300
350
400
450
500
Packets

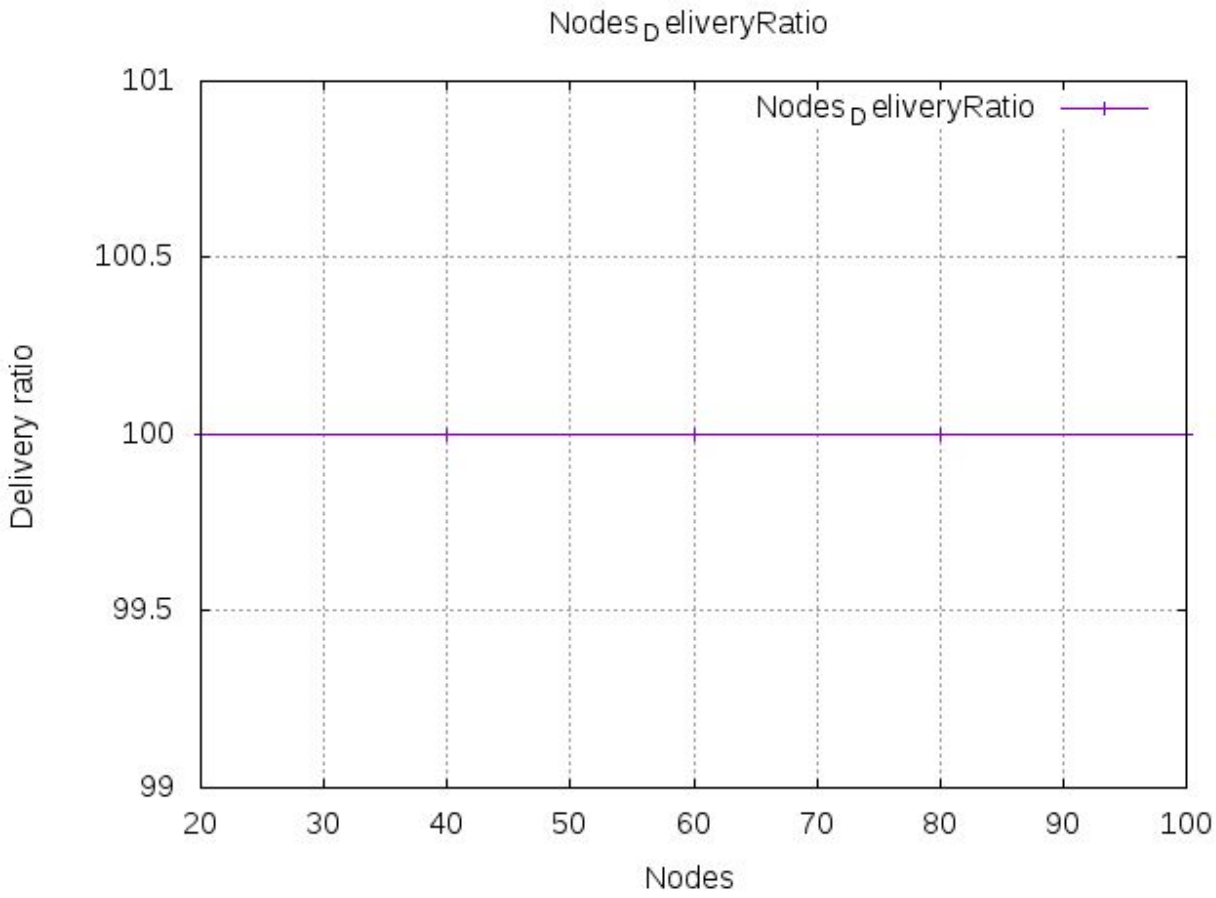

PacketsPerSec D ropRatio
PacketsPerSec D ropRatio
Drop ratio
Packets
0
5
10
15
20
25
30
35
100
150
200
250
300
350
400
450
500

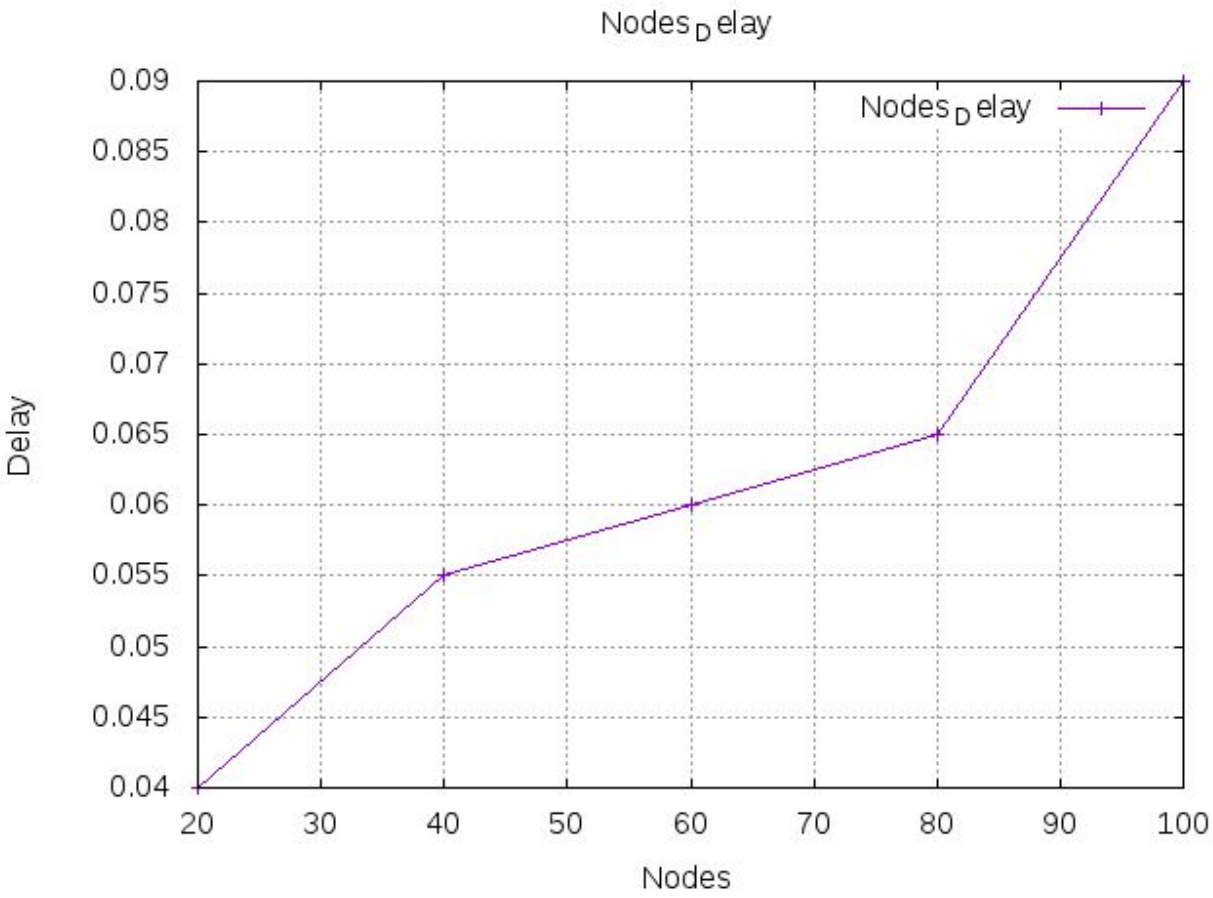

PacketsPerSec D eliveryRatio
PacketsPerSec D eliveryRatio
Delivery ratio
100
95
90
85
80
75
70
65
100
150
200
250
300
350
400
450
500
Packets

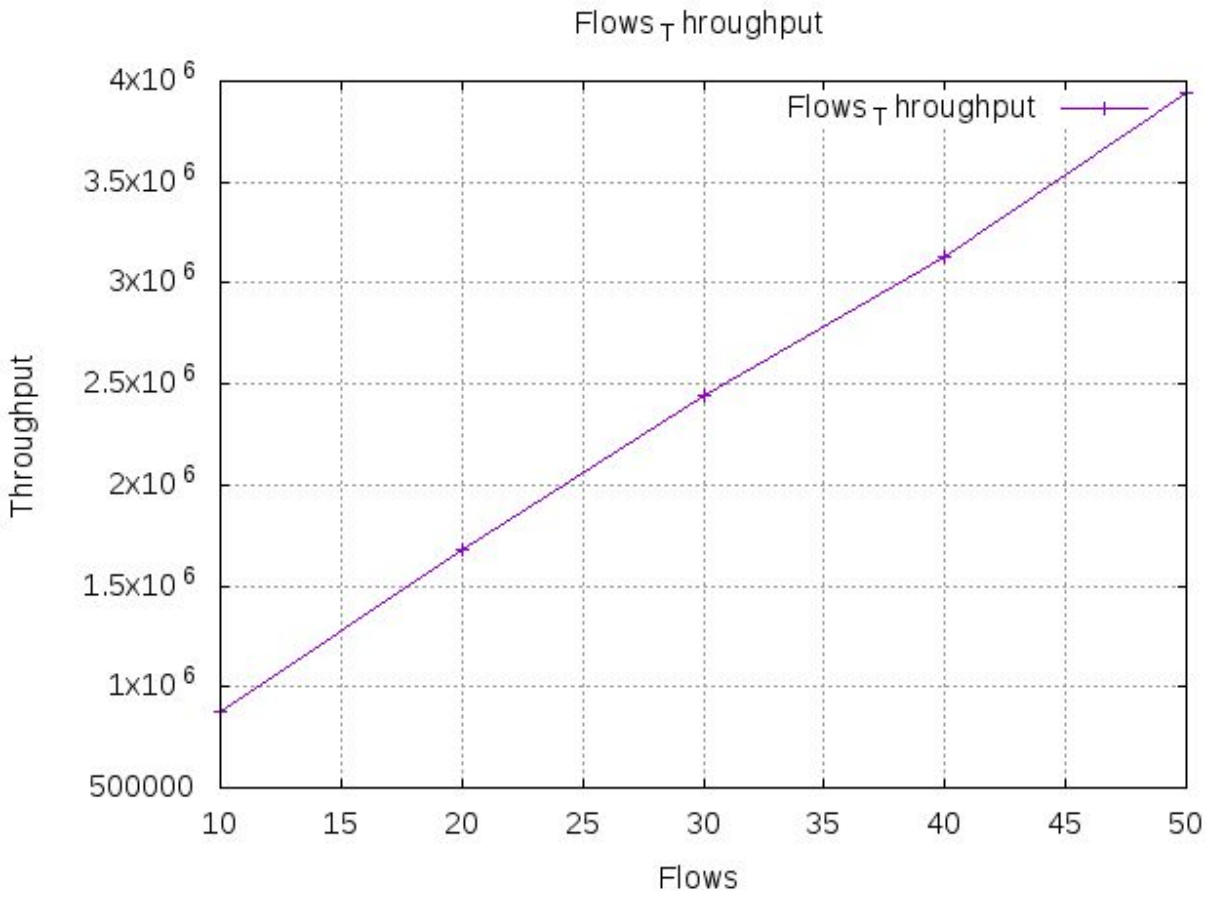

PacketsPerSec D elay
PacketsPerSec D elay
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
Delay
100
150
200
250
300
350
400
450
500
Packets

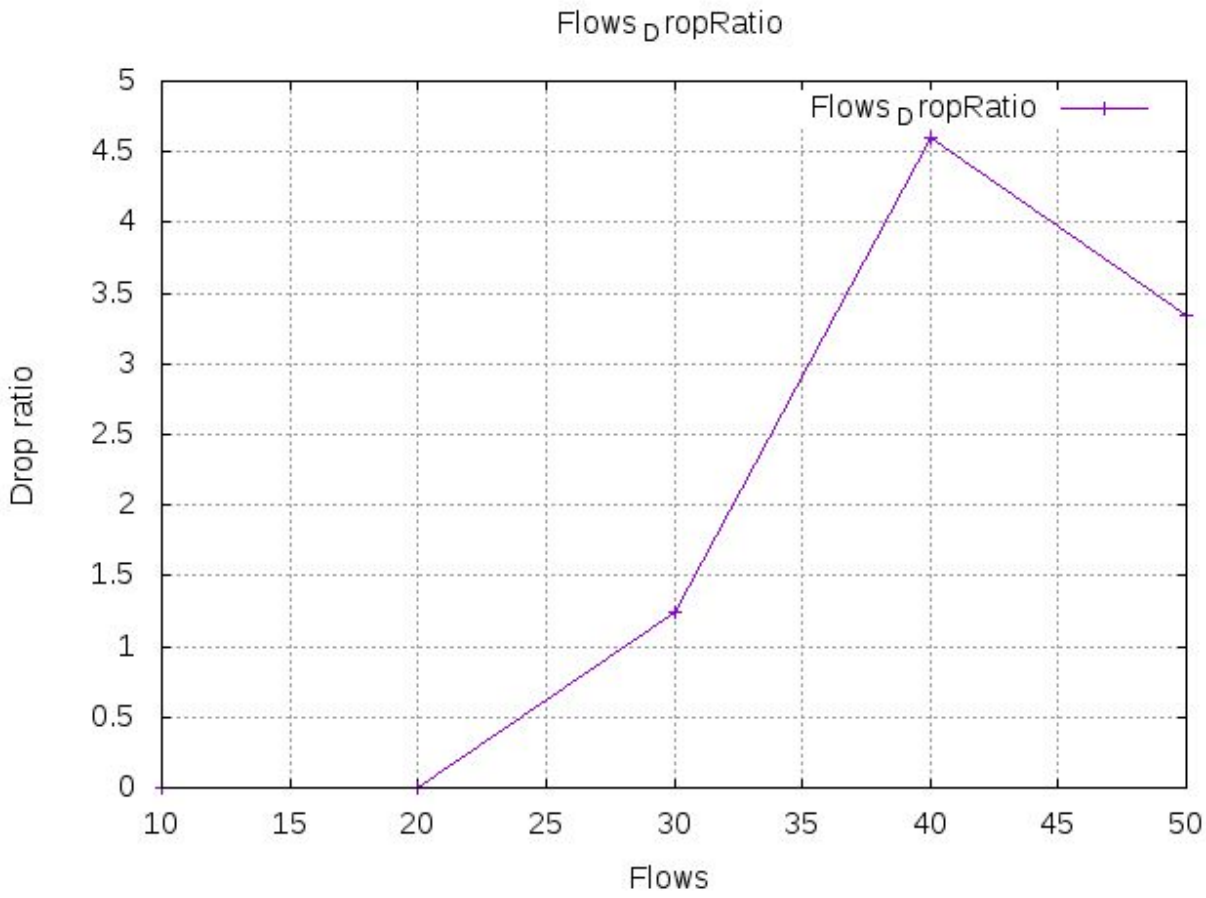

Nodes T hroughput
Nodes T hroughput
878500
878000
877500
877000
876500
876000
875500
875000
874500
Throughput
20
30
40
50
60
70
80
90
100
Nodes

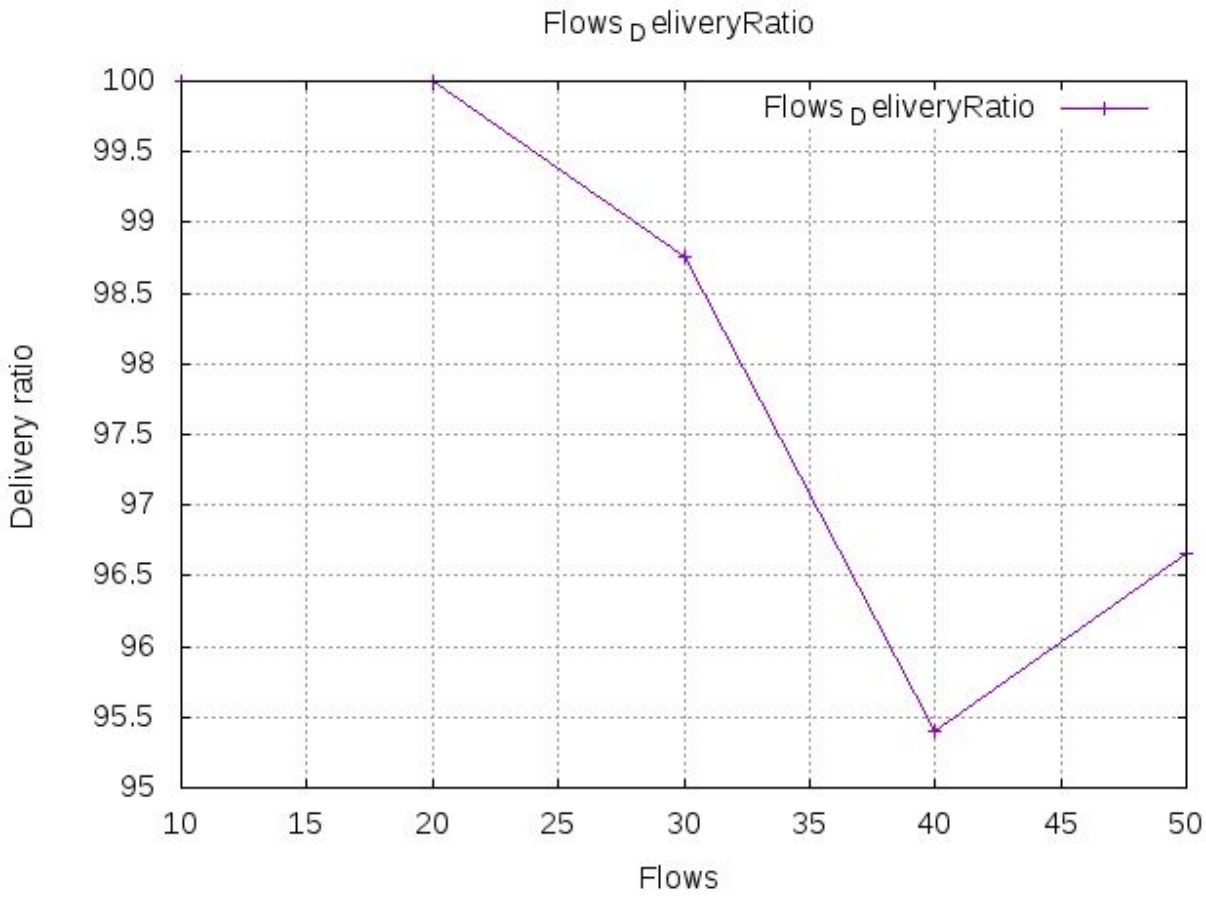

NodesDropRatio
NodesDropRatio
1
0.5
0
-0.5
-1
Drop ratio
20
30
40
50
60
70
80
90
100
Nodes

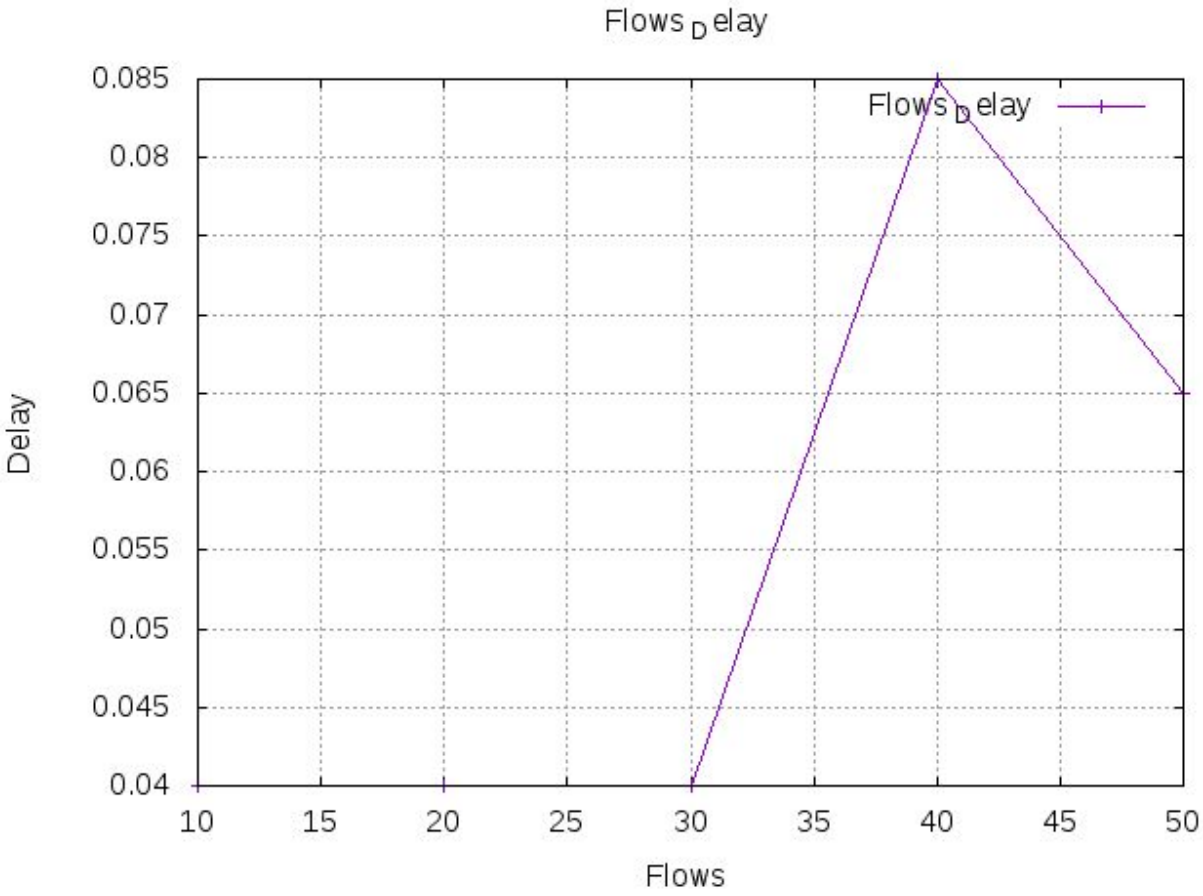

Nodes D eliveryRatio
Nodes D eliveryRatio
101
100.5
100
99.5
99
Delivery ratio
20
30
40
50
60
70
80
90
100
Nodes


Nodes Delay
Nodes Delay
Delay
Nodes
0.09
0.085
0.08
0.075
0.07
0.065
0.06
0.055
0.05
0.045
0.04
20
30
40
50
60
70
80
90
100


Flows T hroughput
Flows T hroughput
Throughput
4x10 6
3.5x10 6
3x10 6
2.5x10 6
2x10 6
1.5x10 6
1x10 6
500000
10
15
20
25
30
35
40
45
50
Flows

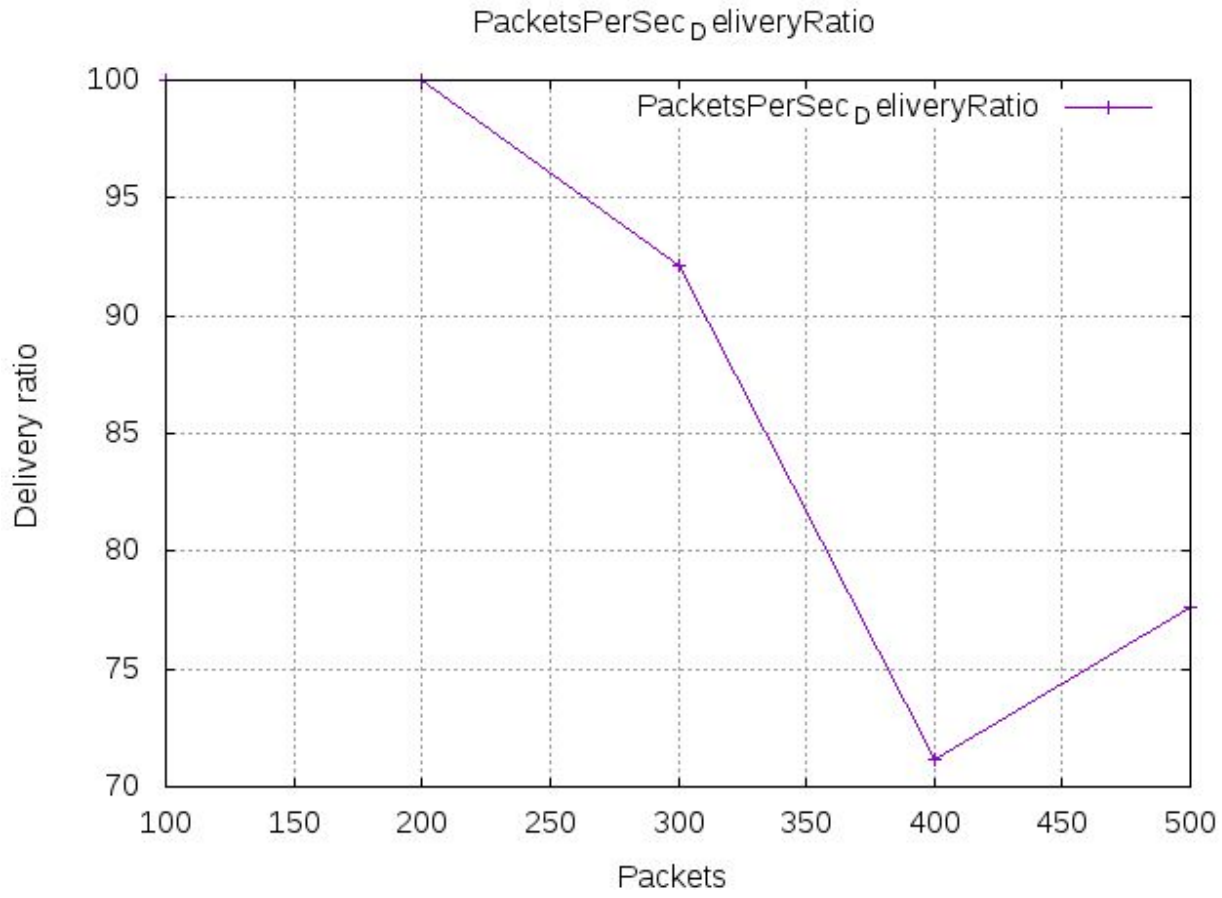

Flows DropRatio
Flows DropRatio
Drop ratio
Flows
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
10
15
20
25
30
35
40
45
50

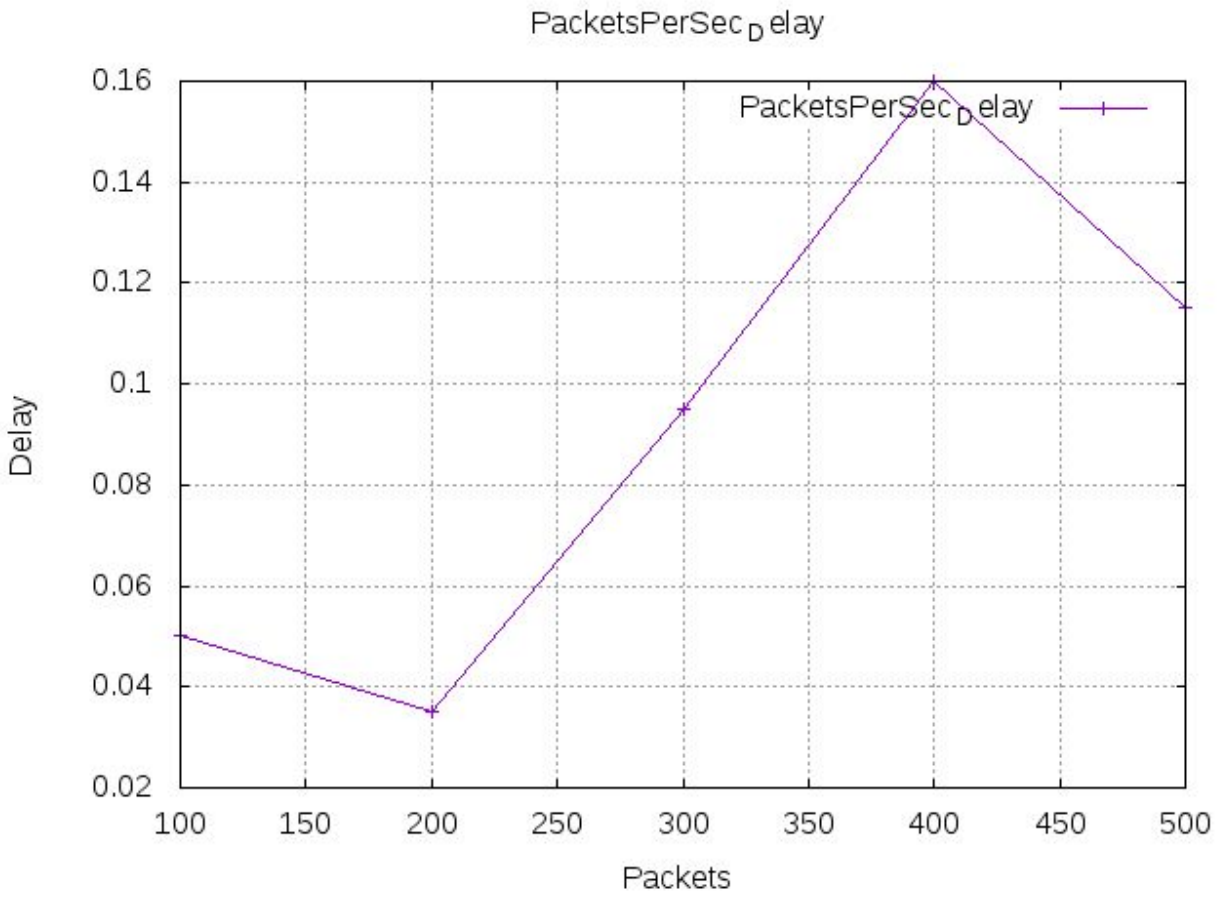

Flows DeliveryRatio
Flows DeliveryRatio
Delivery ratio
100
99.5
99
98.5
98
97.5
97
96.5
96
95.5
95
10
15
20
25
30
35
40
45
50
Flows

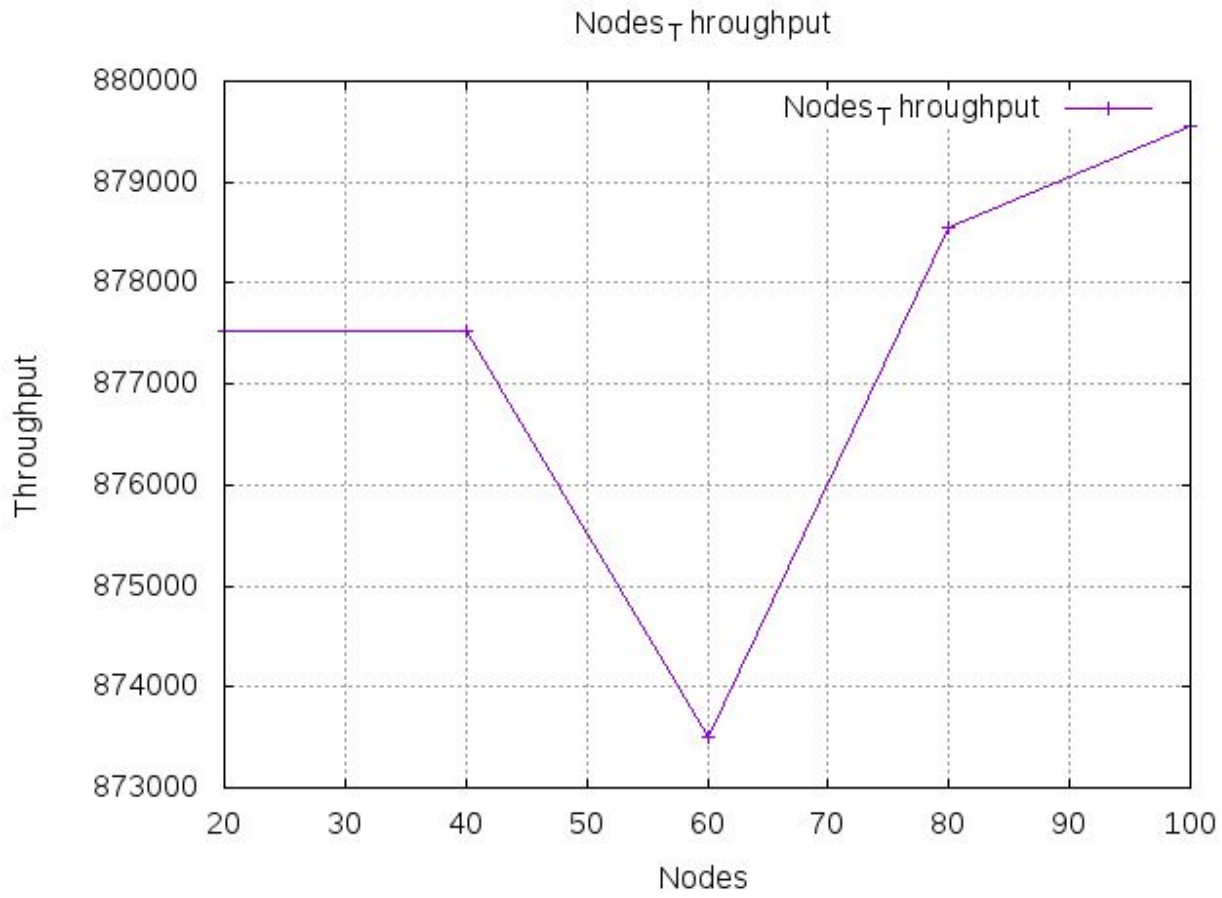

Flows Delay
Flows Delay
Delay
Flows
0.085
0.08
0.075
0.07
0.065
0.06
0.055
0.05
0.045
0.04
10
15
20
25
30
35
40
45
50

After modifications made in the simulator

r


PacketsPerSec D eliveryRatio
PacketsPerSec D eliveryRatio
Delivery ratio
Packets
100
95
90
85
80
75
70
100
150
200
250
300
350
400
450
500

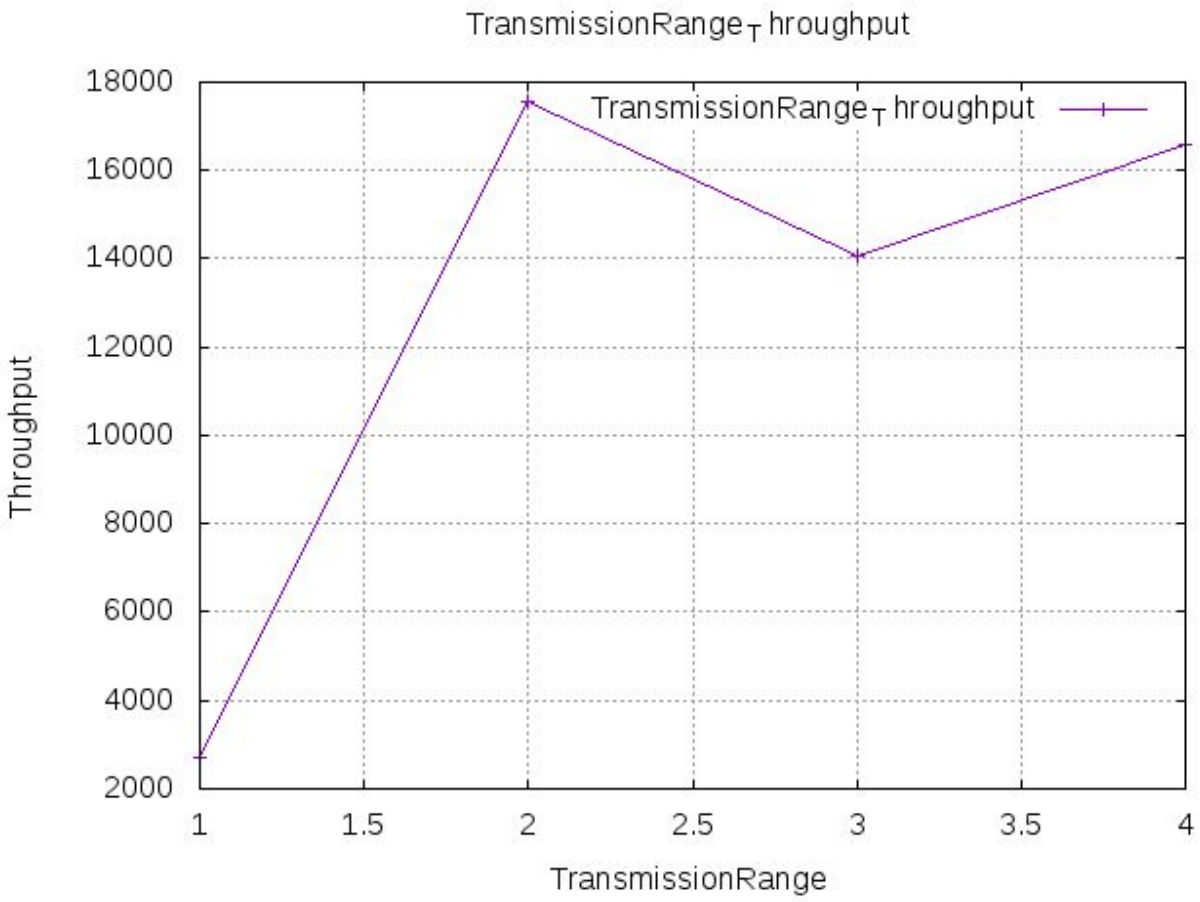

PacketsPerSec_Delay
PacketsPerSec_Delay
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
Delay
100
150
200
250
300
350
400
450
500
Packets

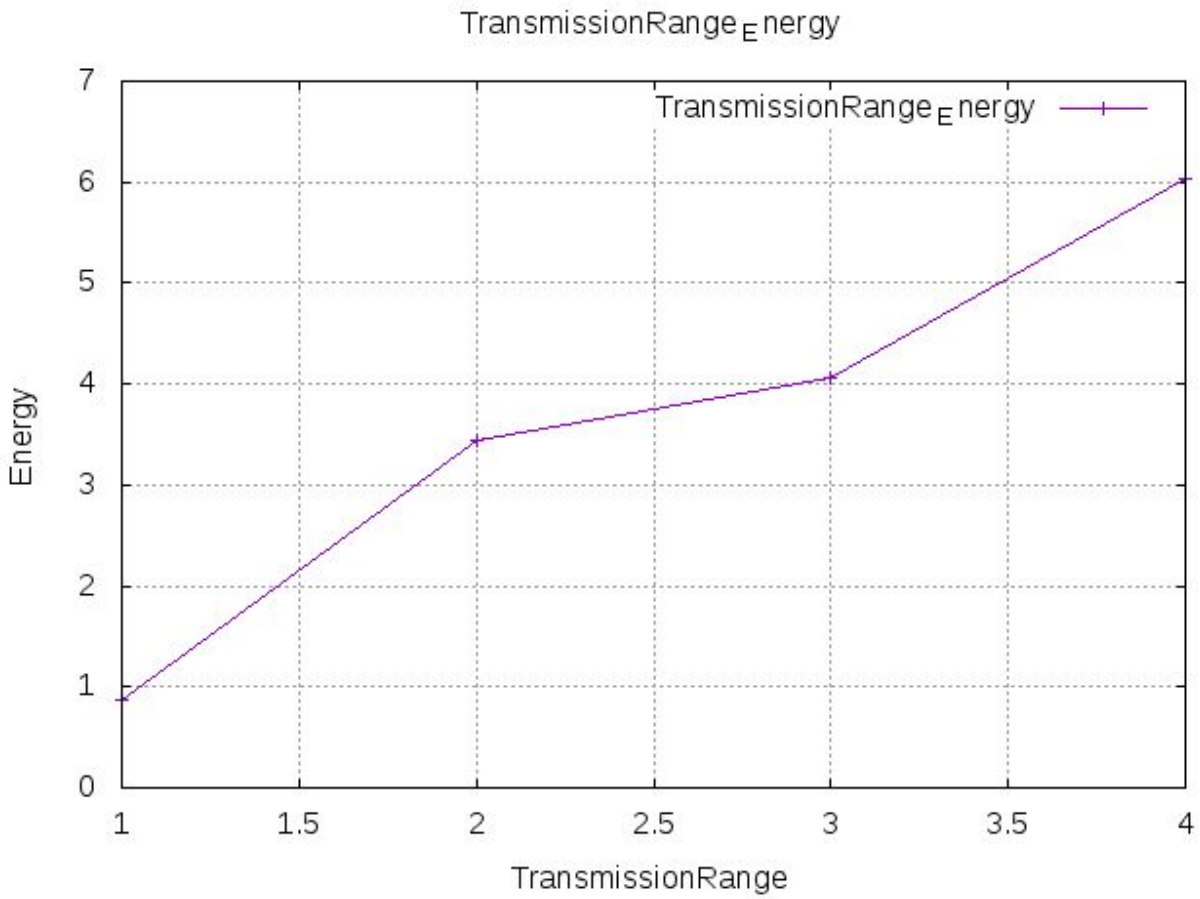


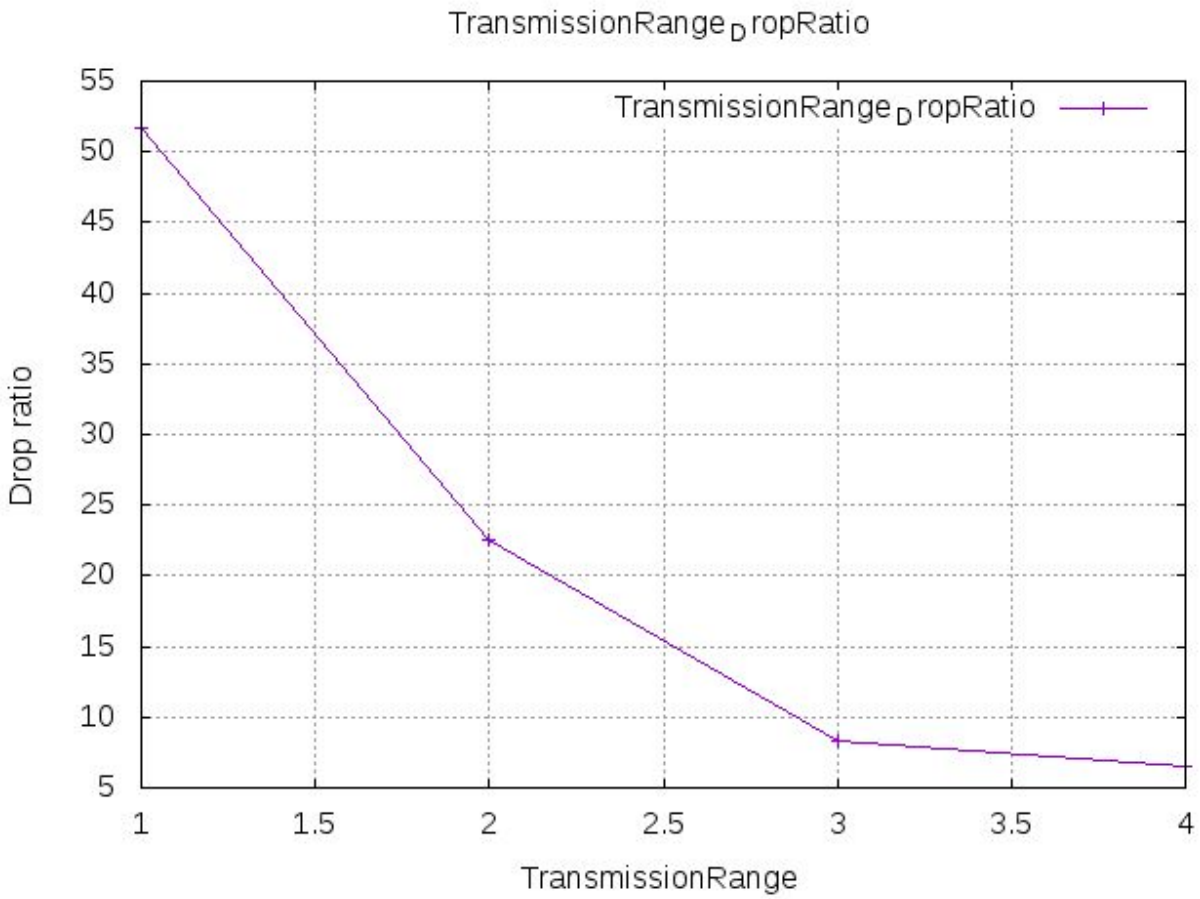
Throughput increases here. Previously we have seen for 20 nodes, throughput starts from 875500 but here it starts from 877500.

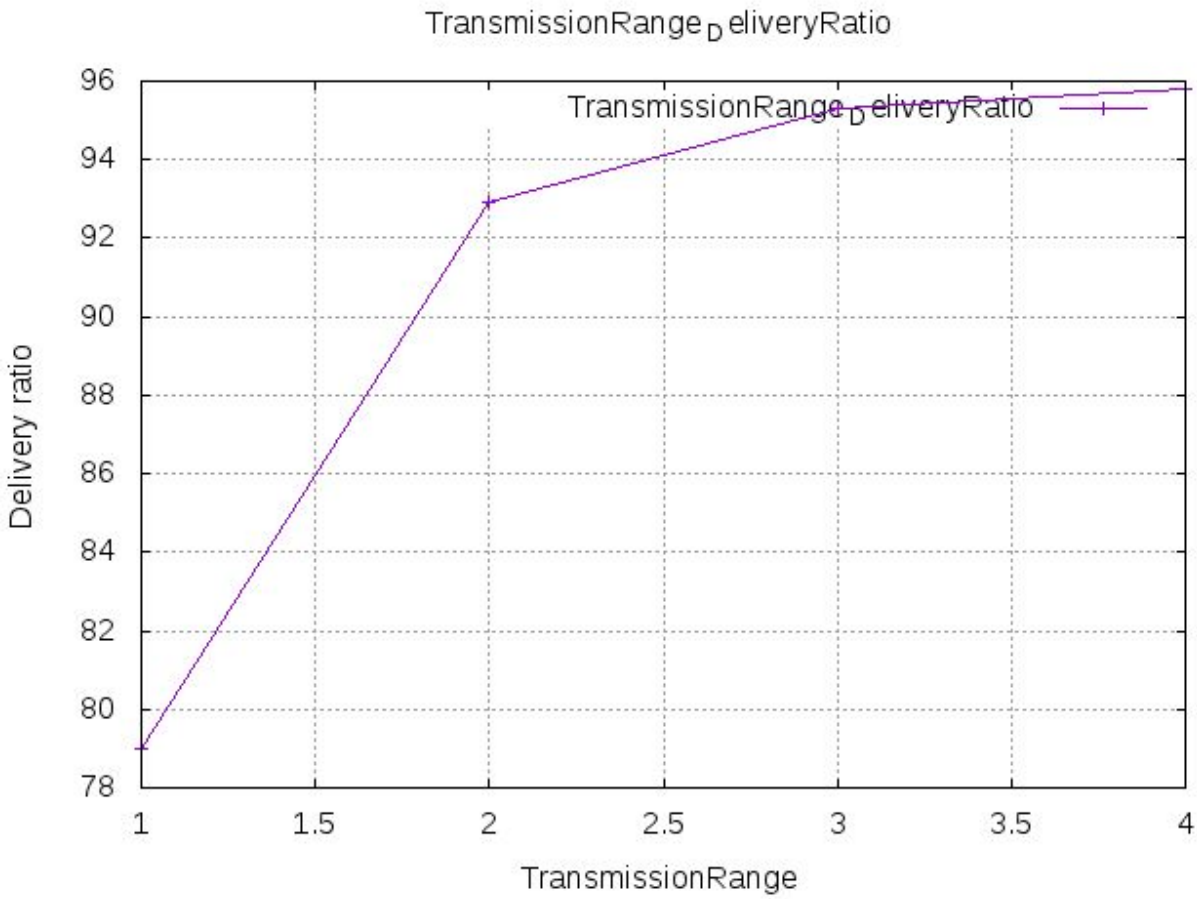
WIRELESS NETWORK

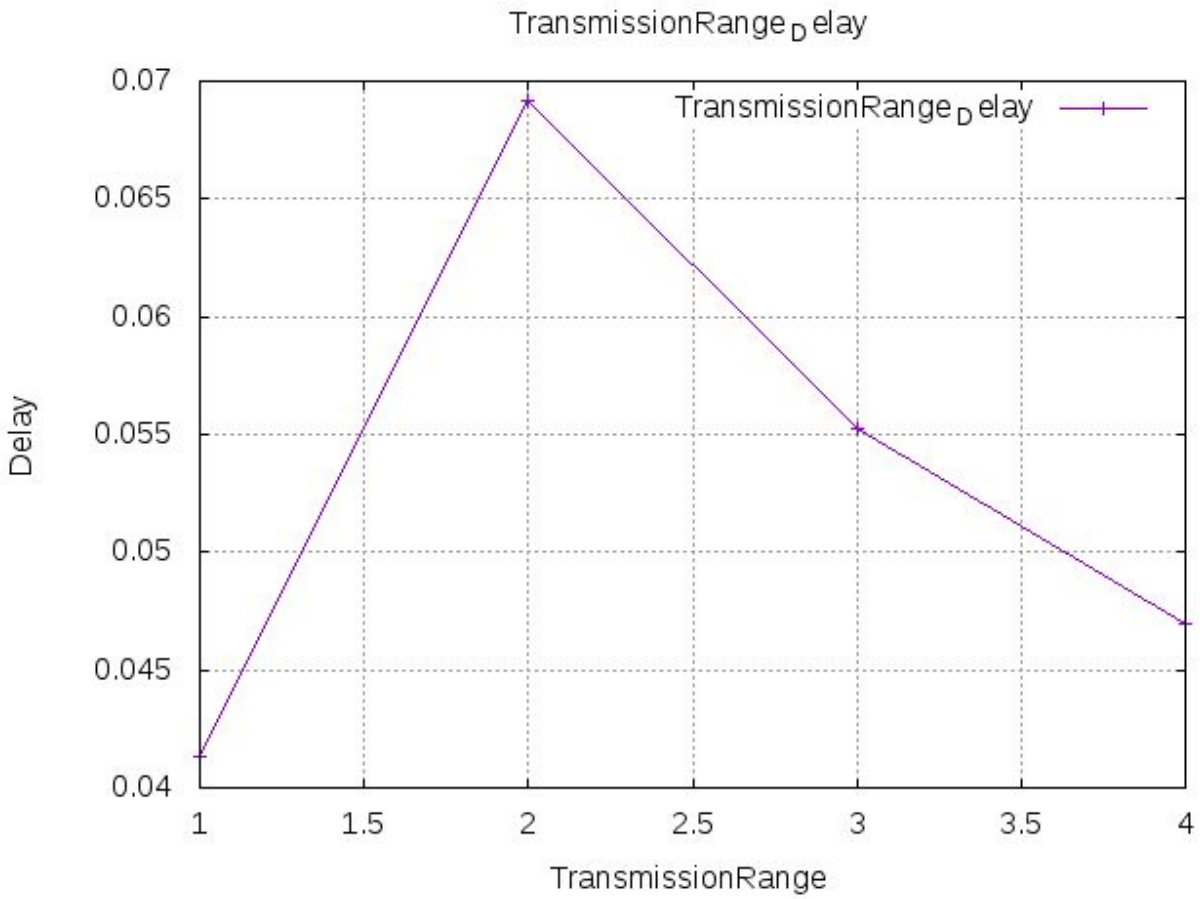
802.15.4

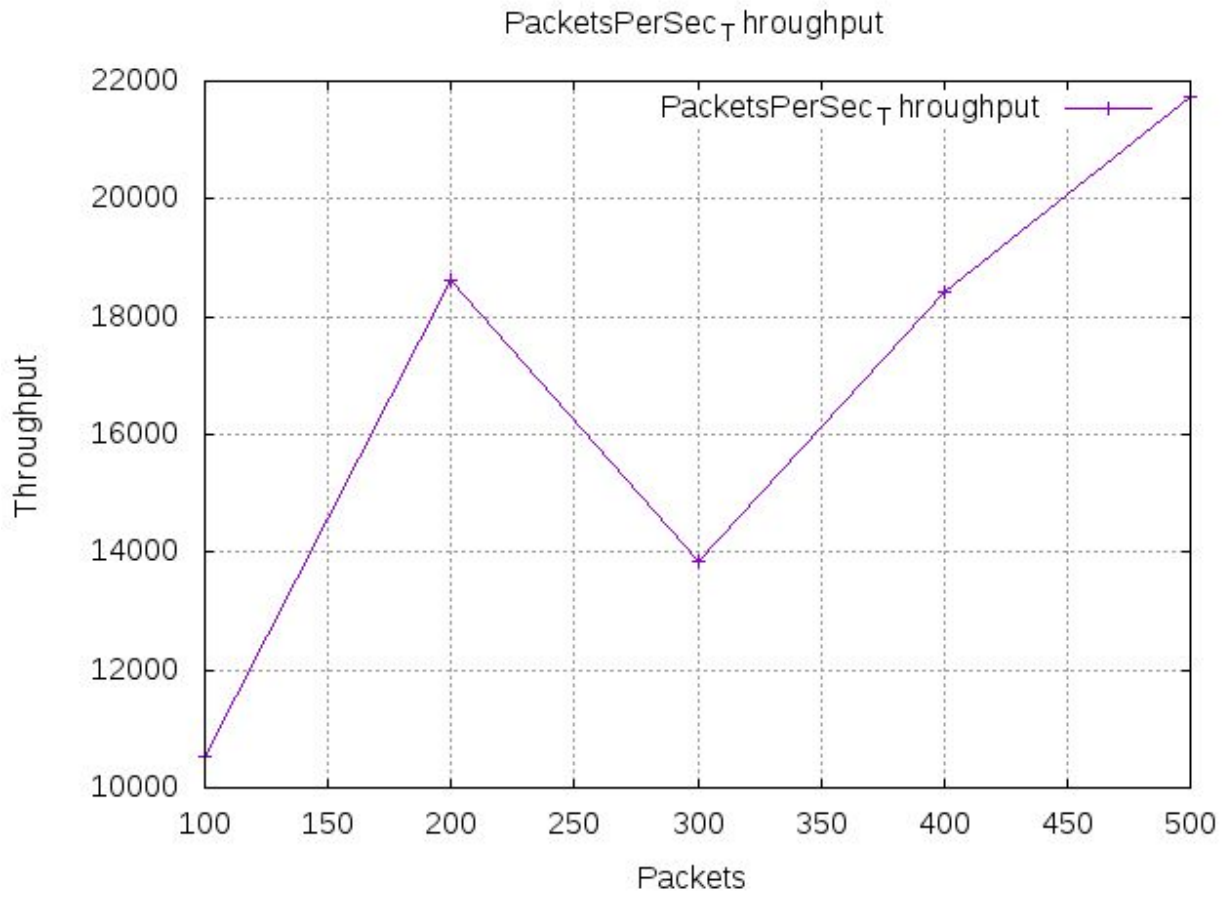

TransmissionRange T hroughput
TransmissionRange T hroughput
Throughput
18000
16000
14000
12000
10000
8000
6000
4000
2000
1
1.5
2
2.5
3
3.5
4
TransmissionRange

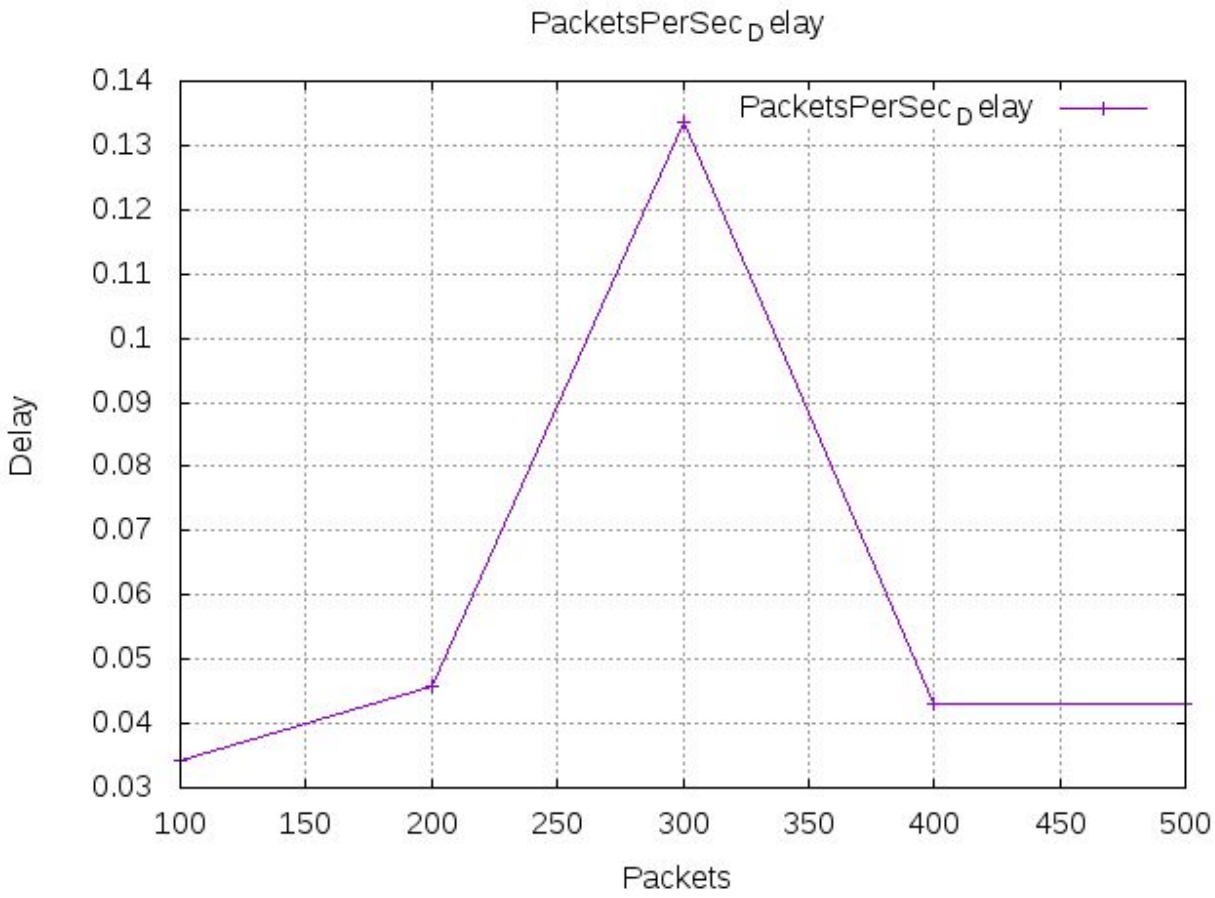

TransmissionRange_Energy
TransmissionRange_Energy
Energy
TransmissionRange
0
1
2
3
4
5
6
7
1
1.5
2
2.5
3
3.5
4

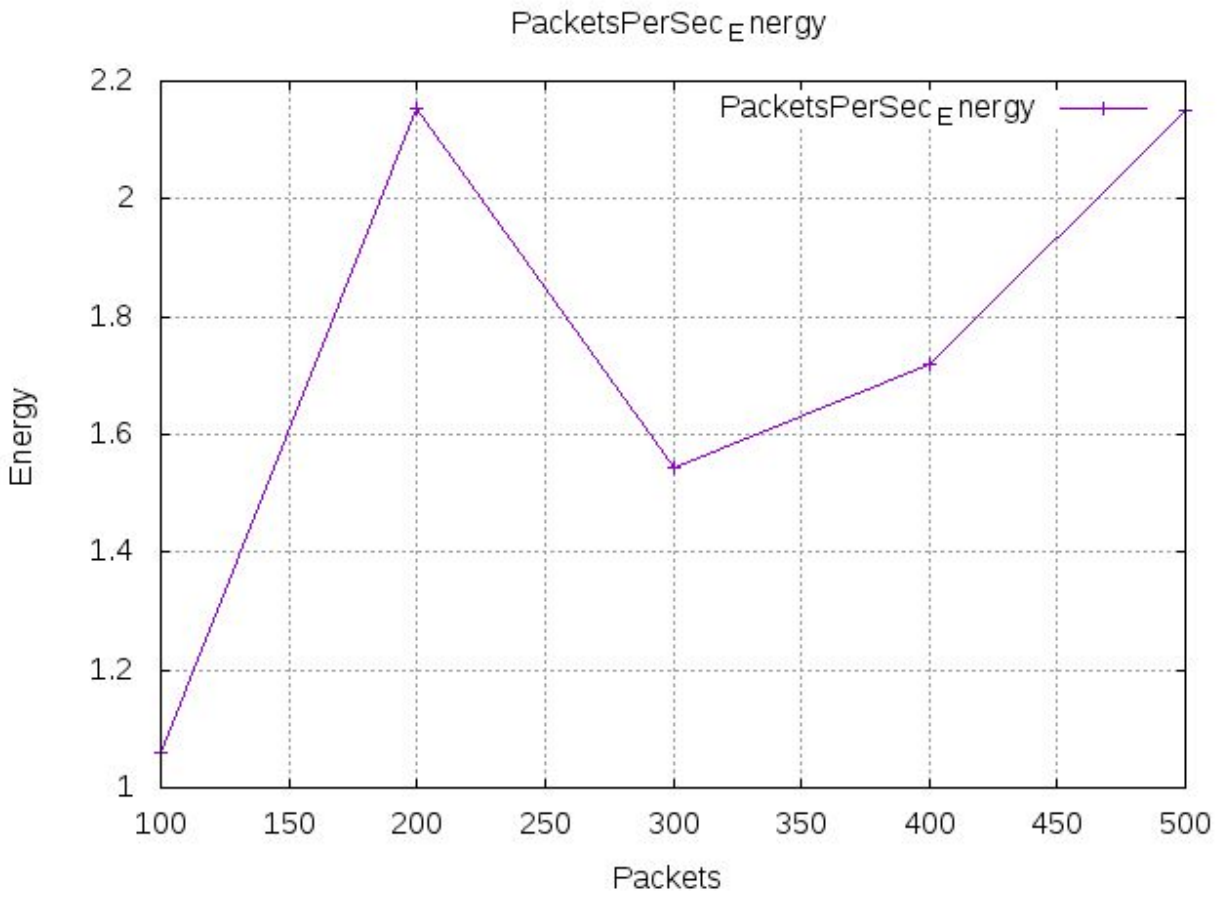

TransmissionRange_DropRatio
TransmissionRange_DropRatio
Drop ratio
TransmissionRange
55
50
45
40
35
30
25
20
15
10
5
1
1.5
2
2.5
3
3.5
4

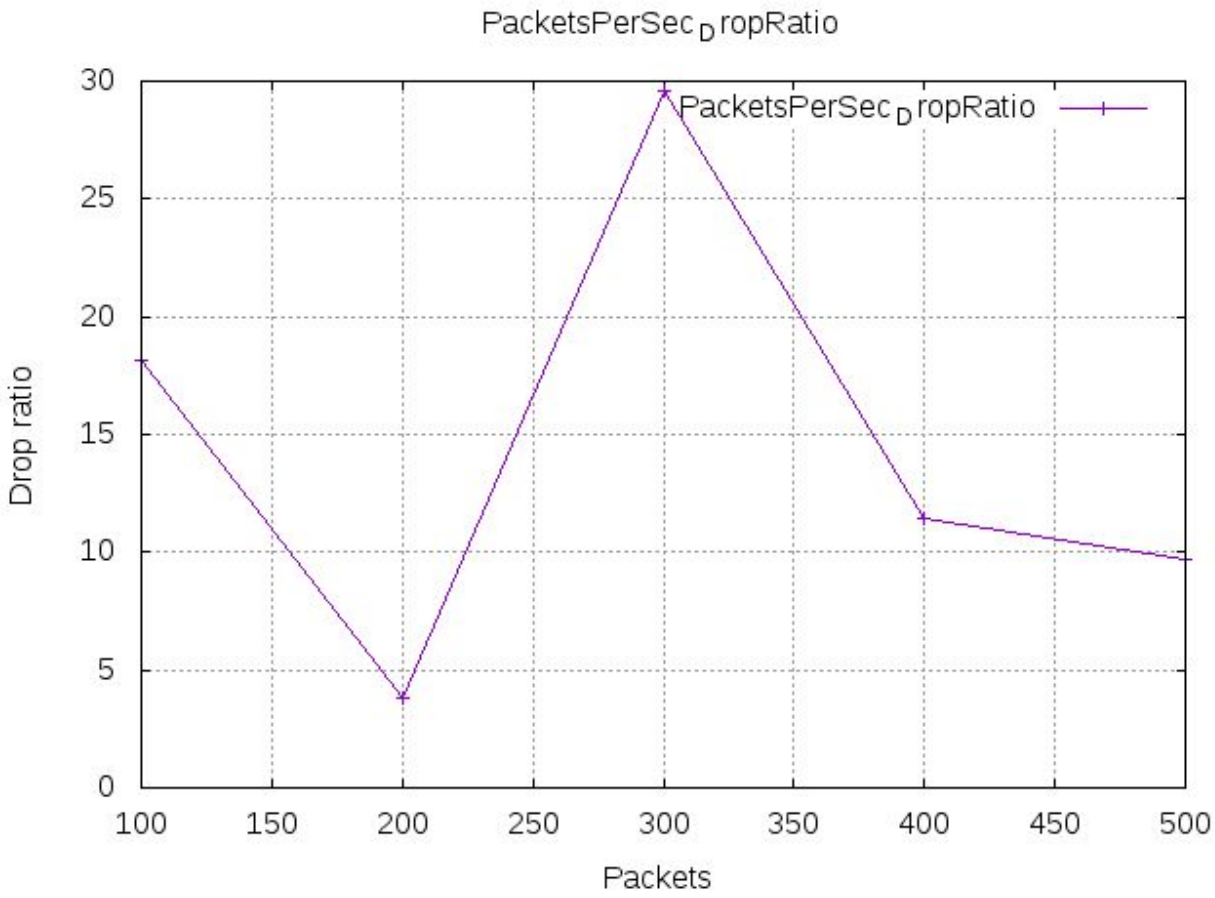

TransmissionRange_DeliveryRatio
TransmissionRange_DeliveryRatio
Delivery ratio
TransmissionRange
96
94
92
90
88
86
84
82
80
78
1
1.5
2
2.5
3
3.5
4

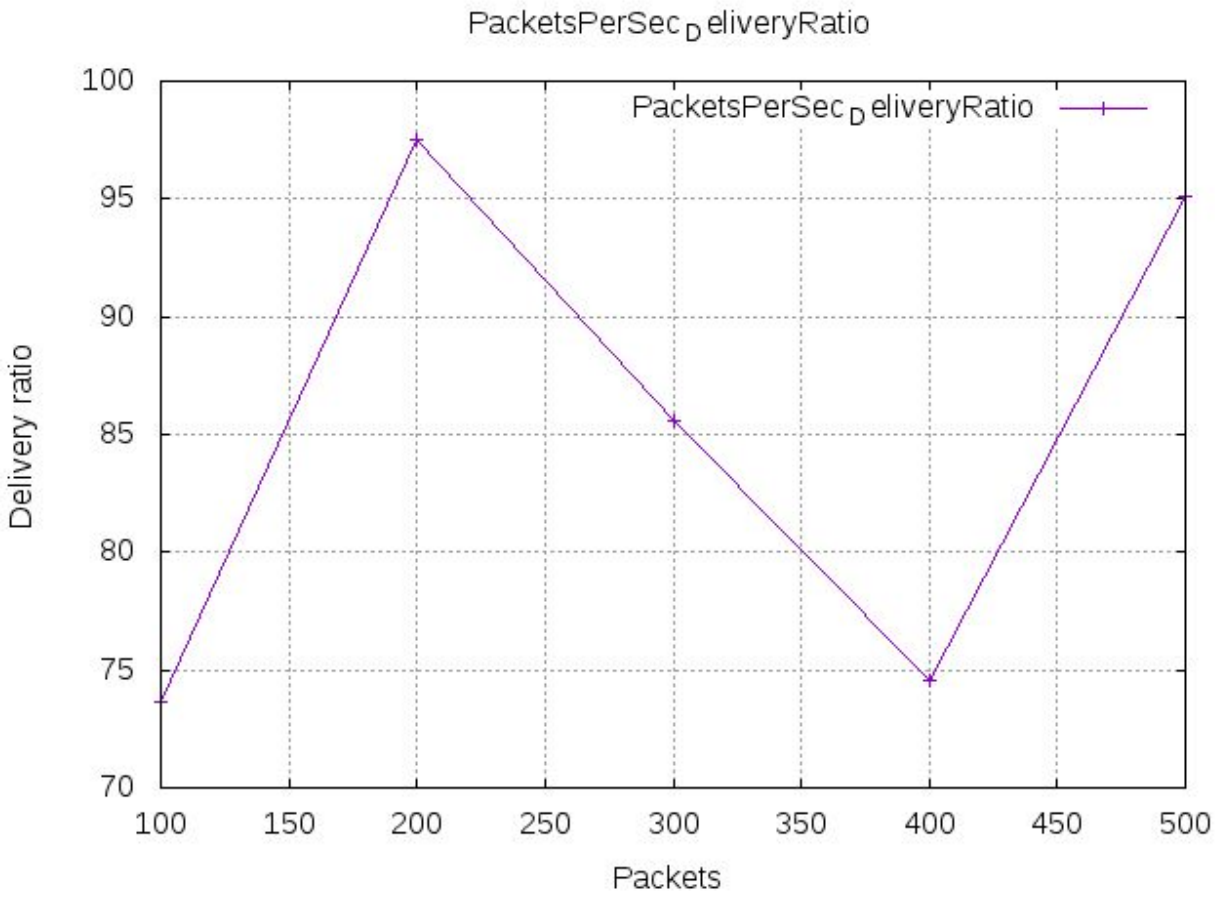

TransmissionRange_Delay
TransmissionRange_Delay
Delay
0.07
0.065
0.06
0.055
0.05
0.045
0.04
1
1.5
2
2.5
3
3.5
4
TransmissionRange

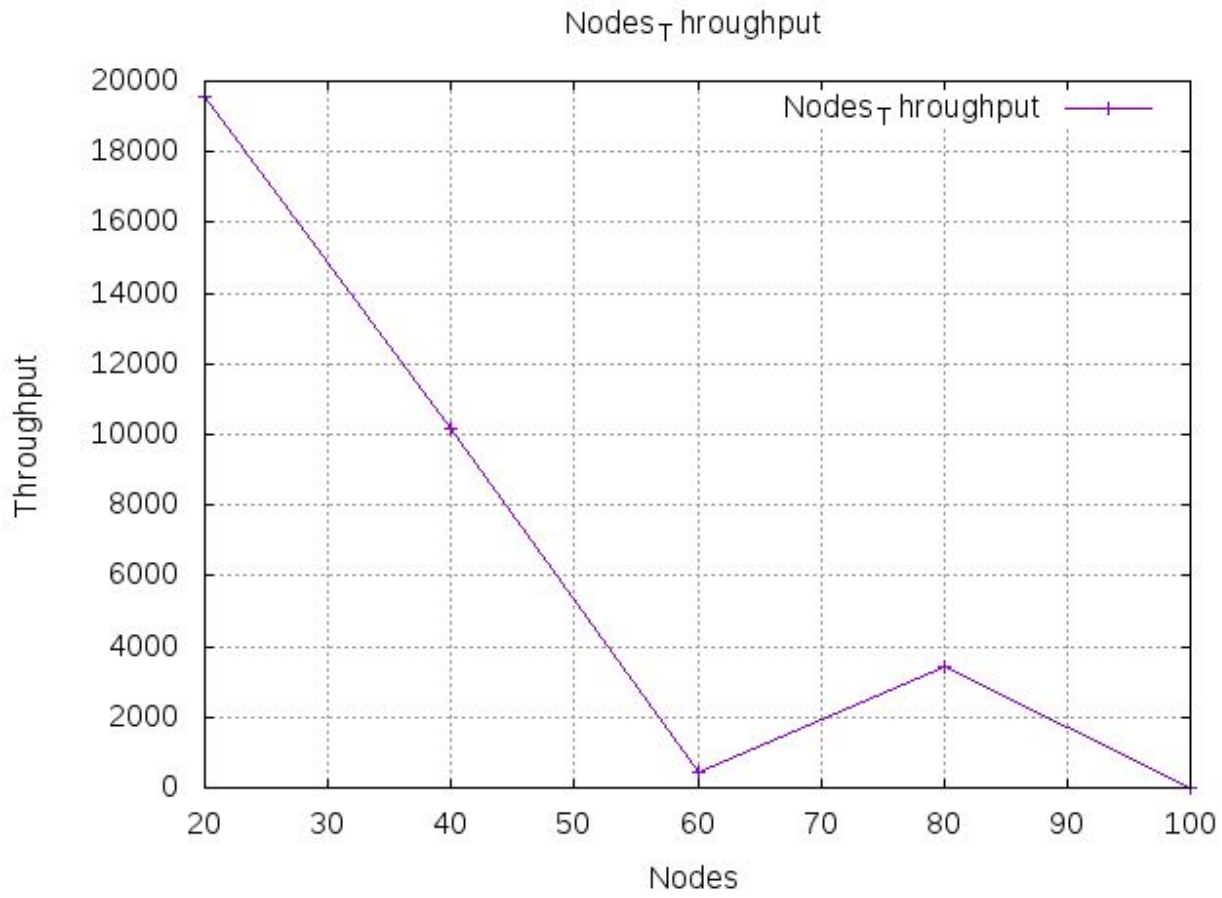

PacketsPerSec T hroughput
PacketsPerSec T hroughput
Throughput
Packets
22000
20000
18000
16000
14000
12000
10000
100
150
200
250
300
350
400
450
500

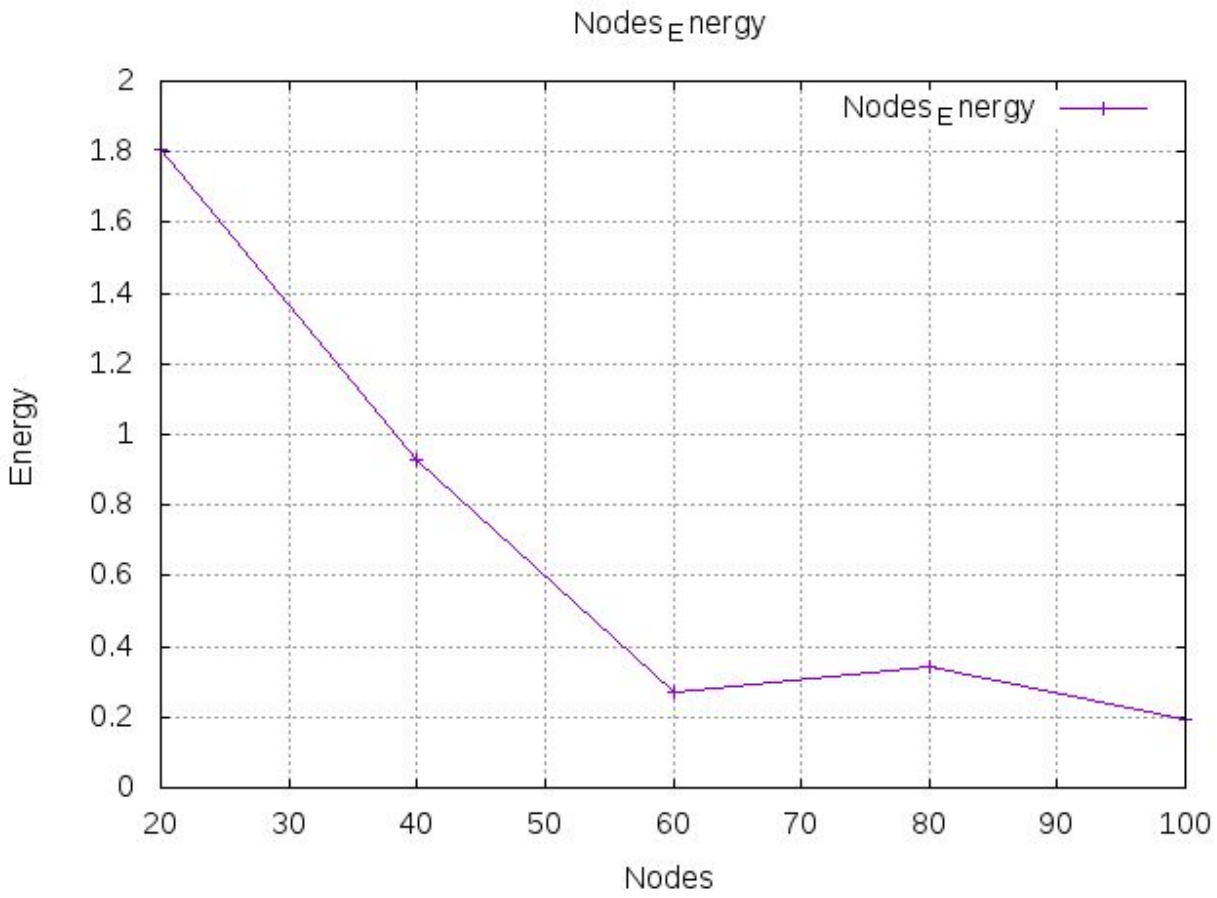

PacketsPerSec_Delay
PacketsPerSec_Delay
Delay
Packets
0.14
0.13
0.12
0.11
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
100
150
200
250
300
350
400
450
500

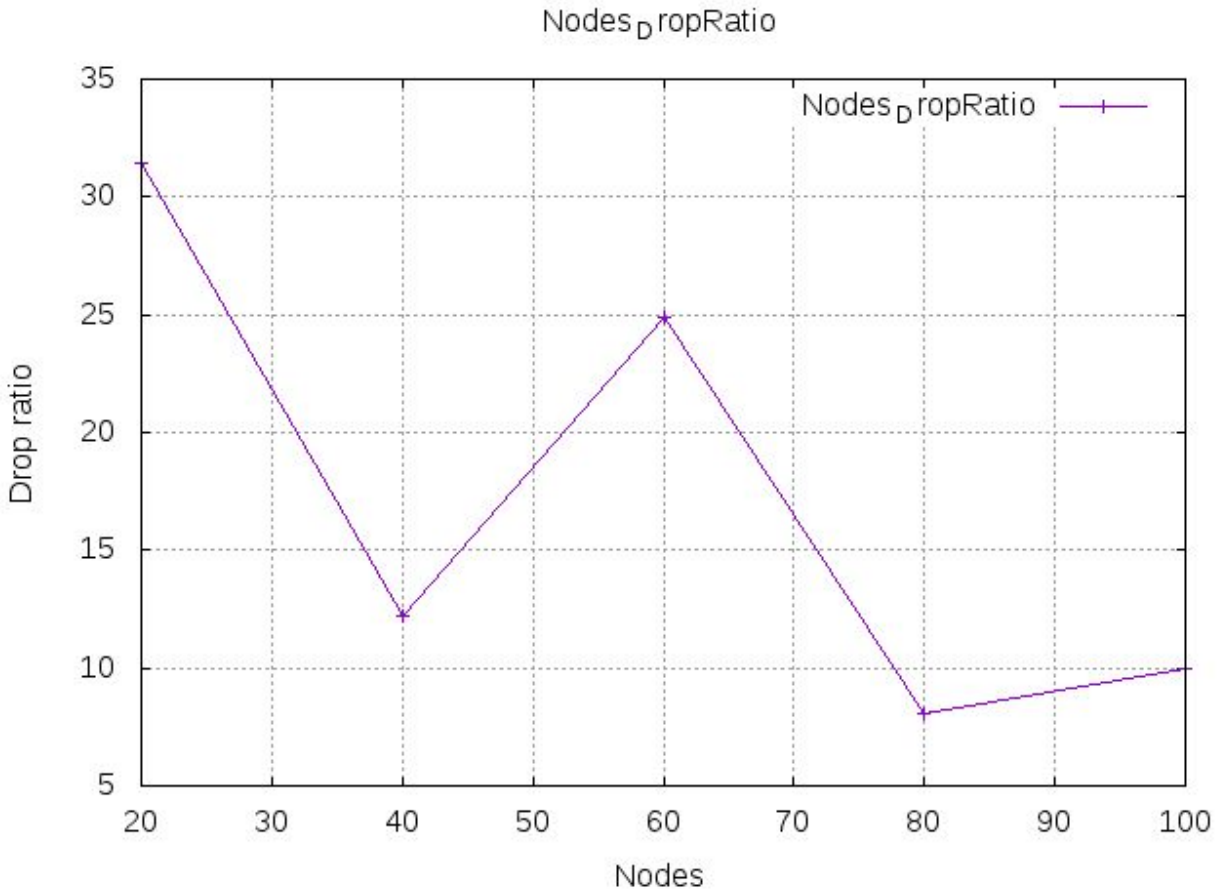

PacketsPerSec Energy
PacketsPerSec Energy
Energy
Packets
2.2
2
1.8
1.6
1.4
1.2
1
100
150
200
250
300
350
400
450
500

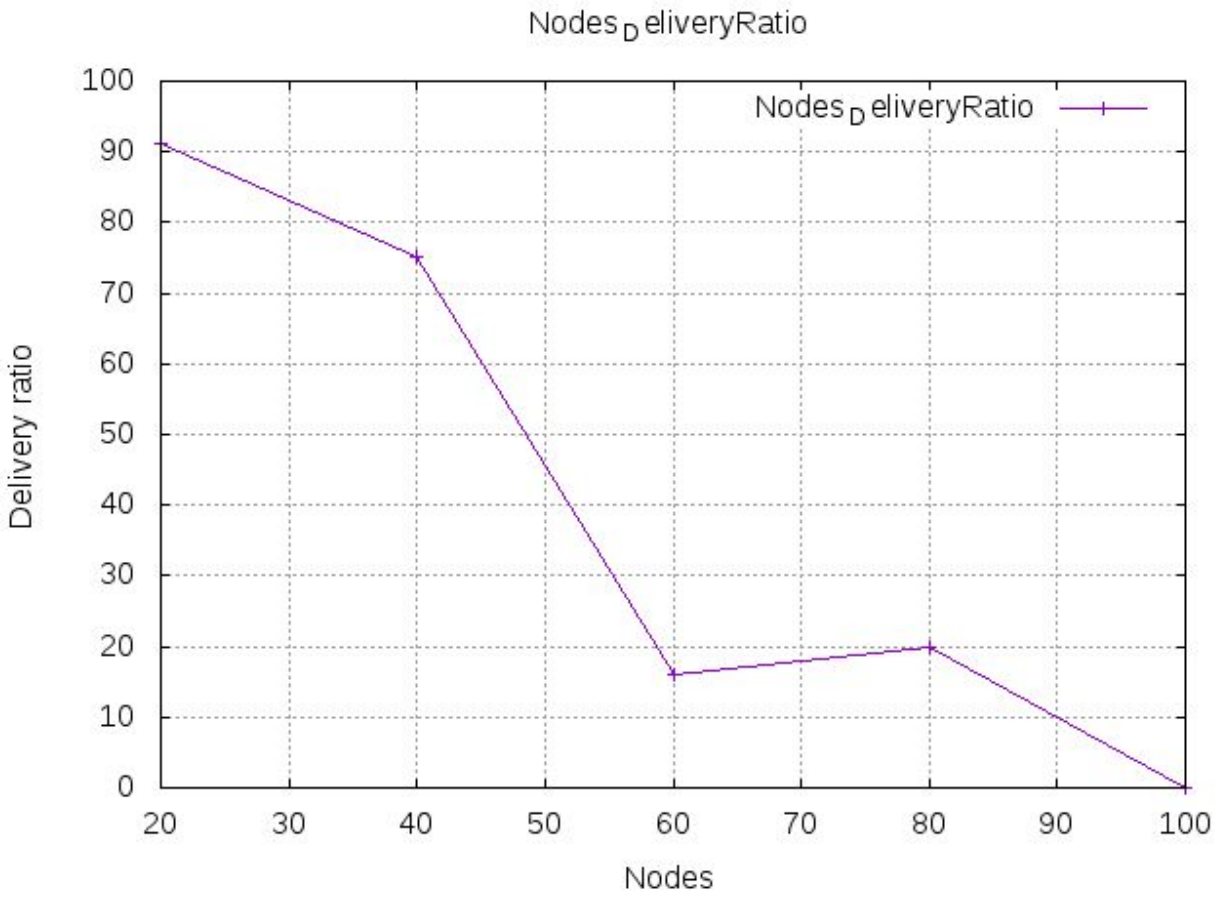

PacketsPerSec D ropRatio
PacketsPerSec D ropRatio
Drop ratio
Packets
0
5
10
15
20
25
30
100
150
200
250
300
350
400
450
500

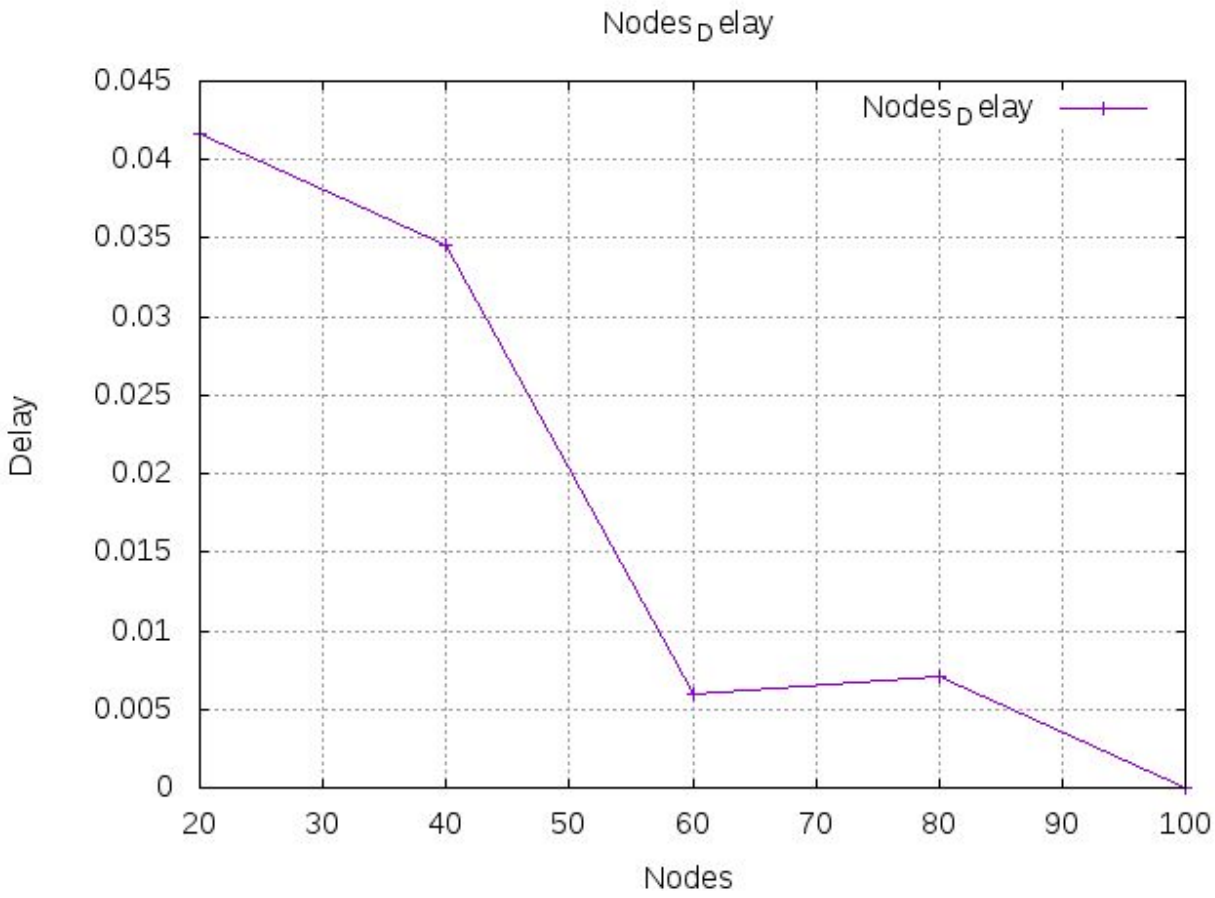

PacketsPerSec D eliveryRatio
PacketsPerSec D eliveryRatio
Delivery ratio
Packets
100
95
90
85
80
75
70
100
150
200
250
300
350
400
450
500

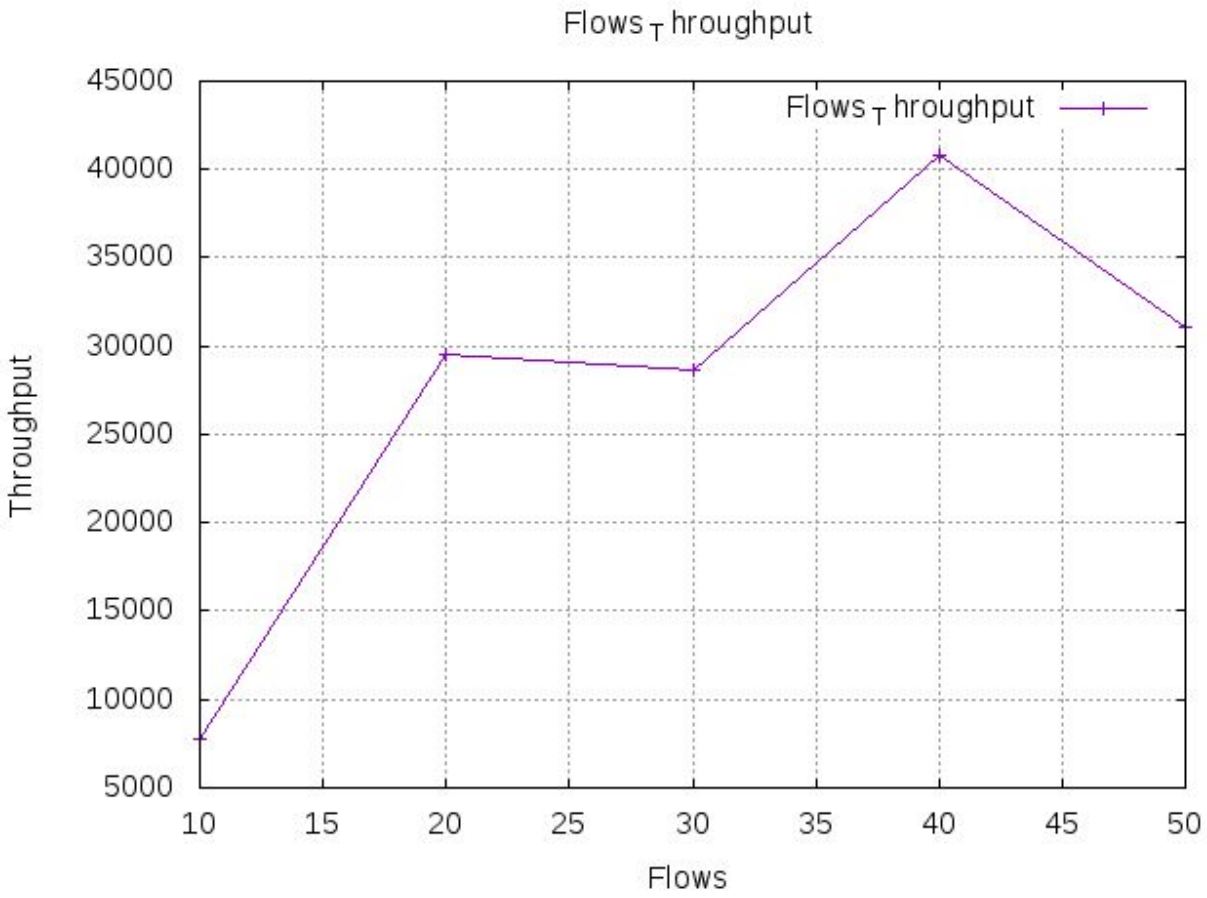

Nodes T hroughput
Nodes T hroughput
Throughput
20000
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
20
30
40
50
60
70
80
90
100
Nodes

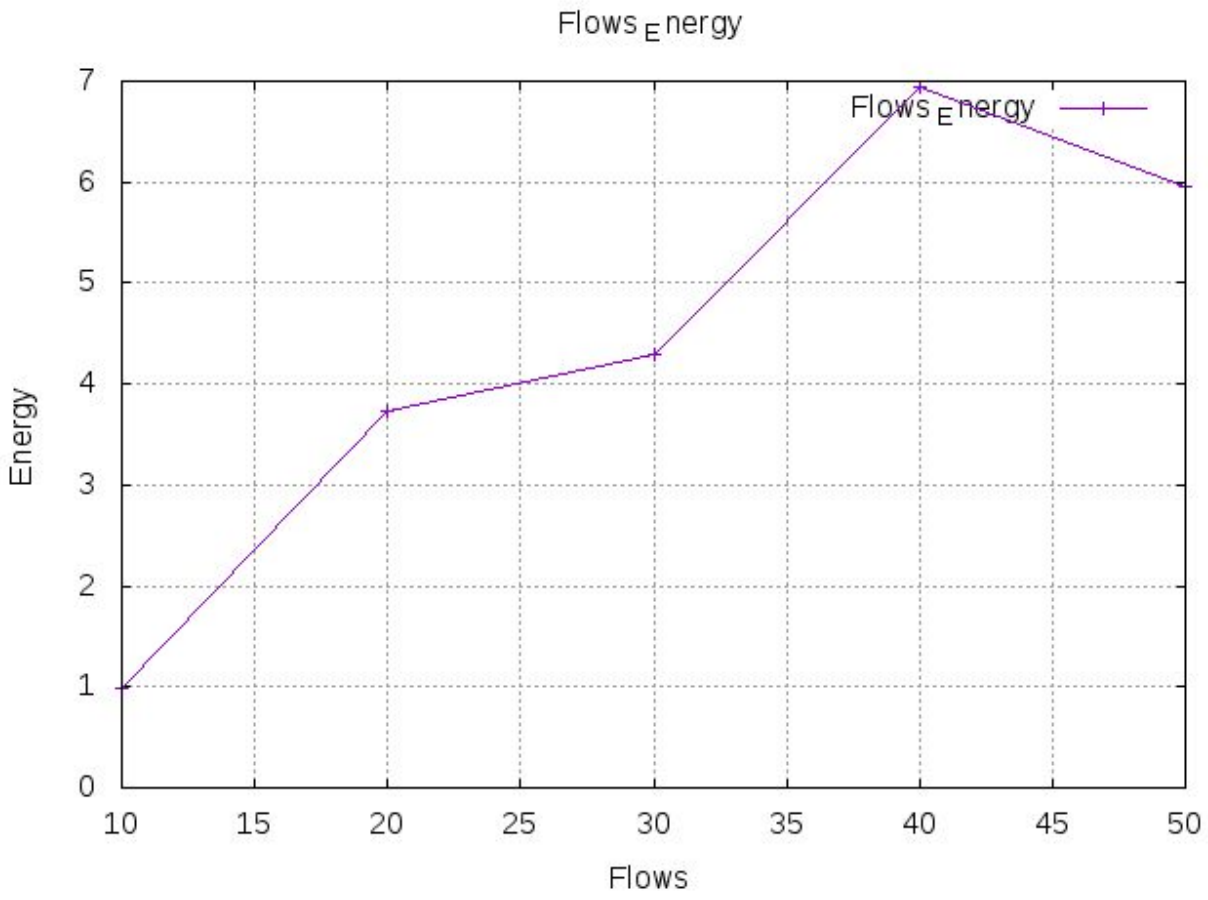

Nodes Energy
Nodes Energy
Energy
Nodes
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
20
30
40
50
60
70
80
90
100

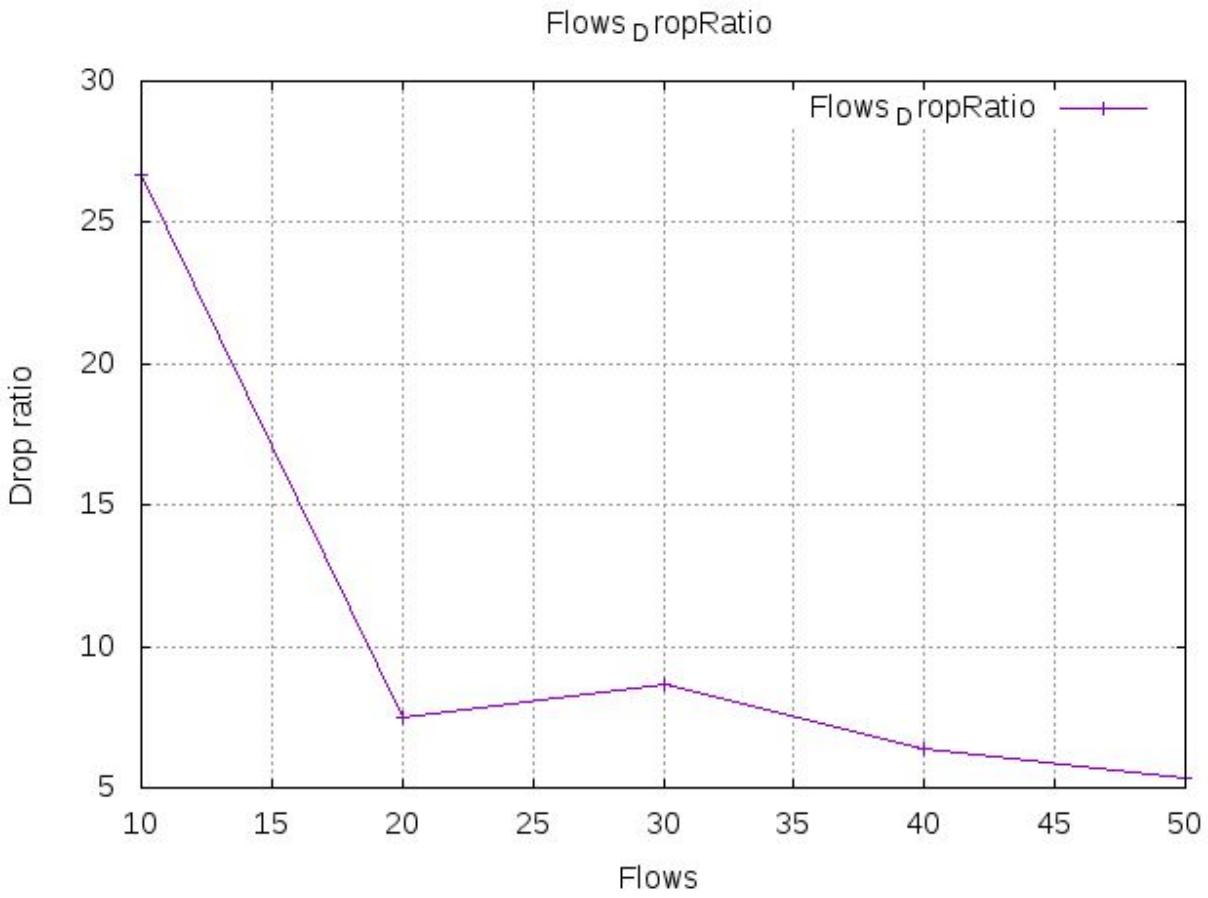

NodesDropRatio
NodesDropRatio
Drop ratio
Nodes
35
30
25
20
15
10
5
20
30
40
50
60
70
80
90
100

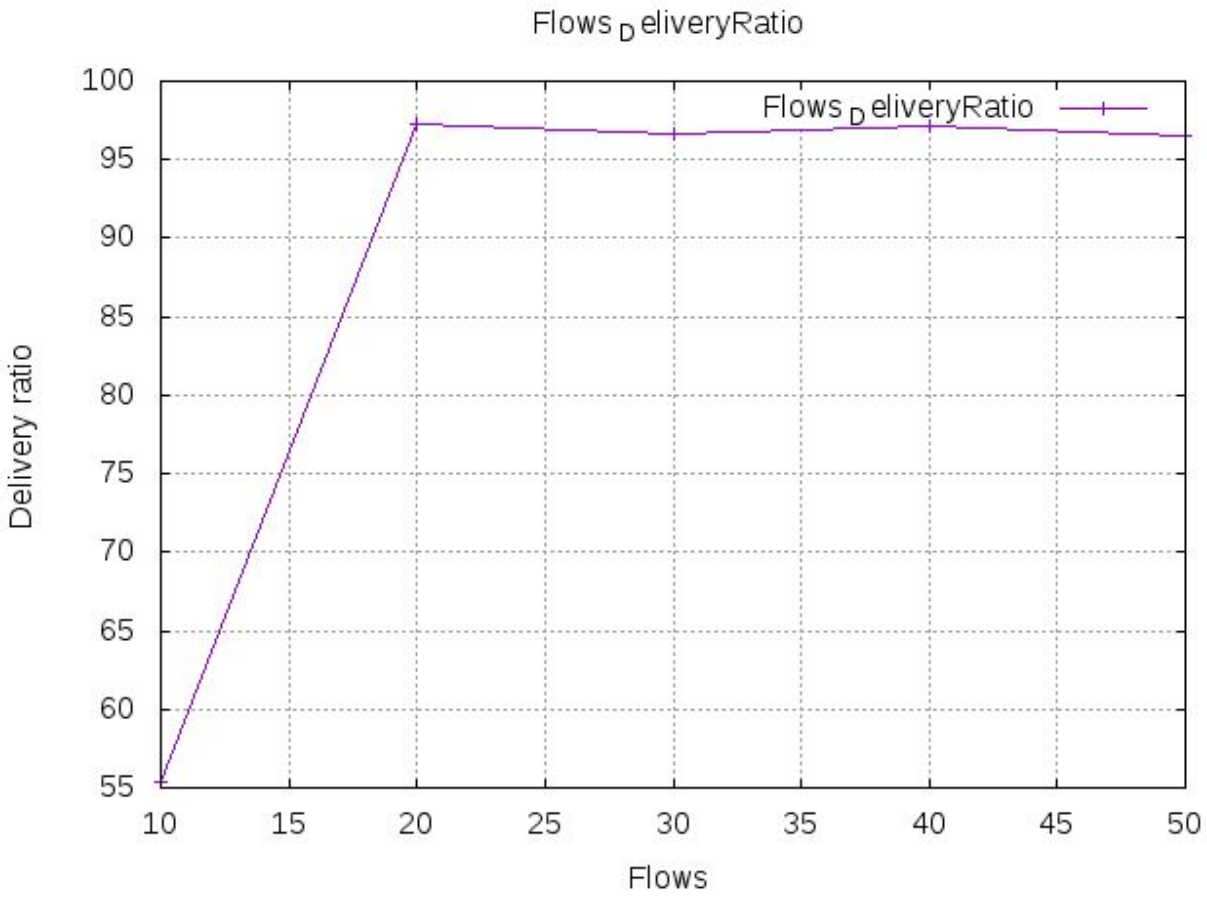

Nodes DeliveryRatio
Nodes DeliveryRatio
Delivery ratio
Nodes
100
90
80
70
60
50
40
30
20
10
0
20
30
40
50
60
70
80
90
100

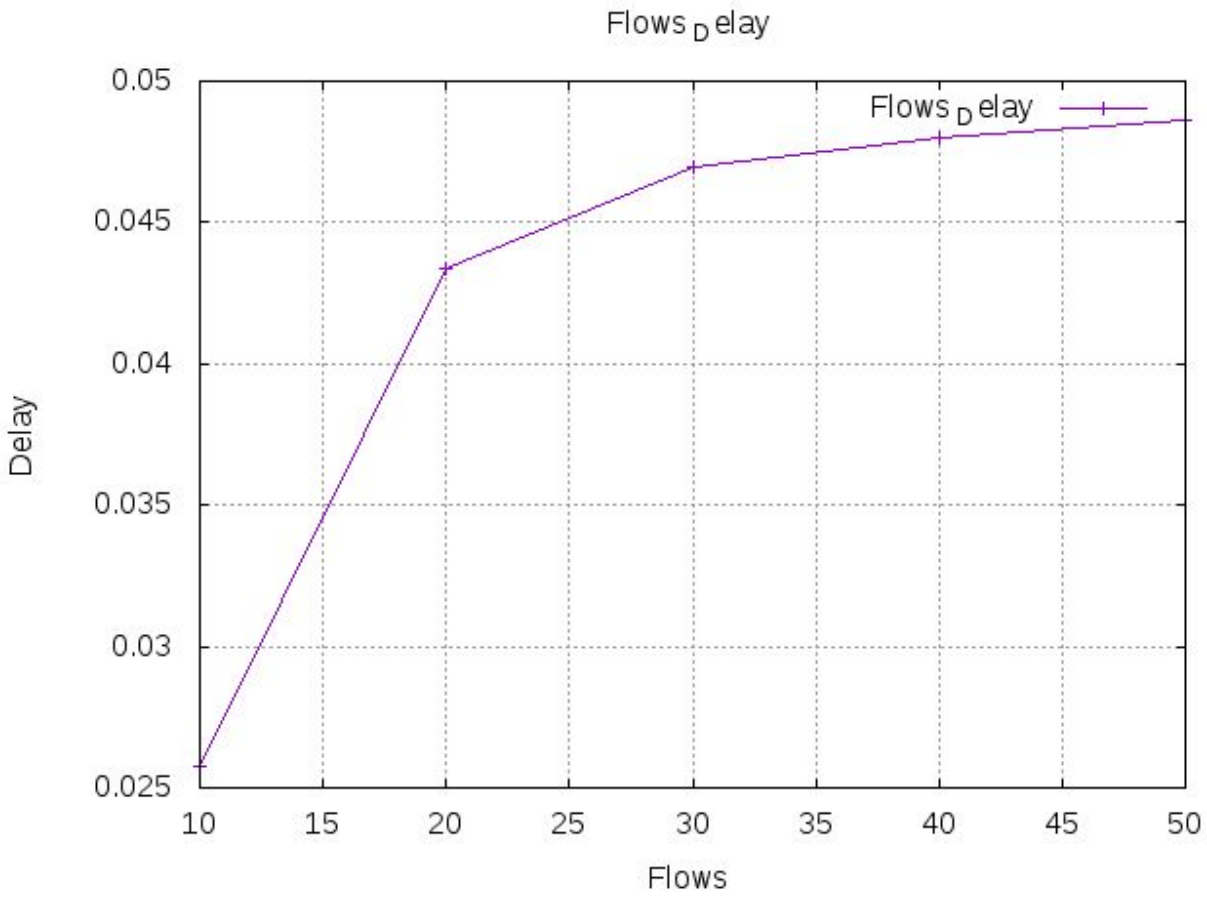

Nodes Delay
Nodes Delay
0.045
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
Delay
20
30
40
50
60
70
80
90
100
Nodes


Flows T hroughput
Flows T hroughput
Throughput
45000
40000
35000
30000
25000
20000
15000
10000
5000
10
15
20
25
30
35
40
45
50
Flows

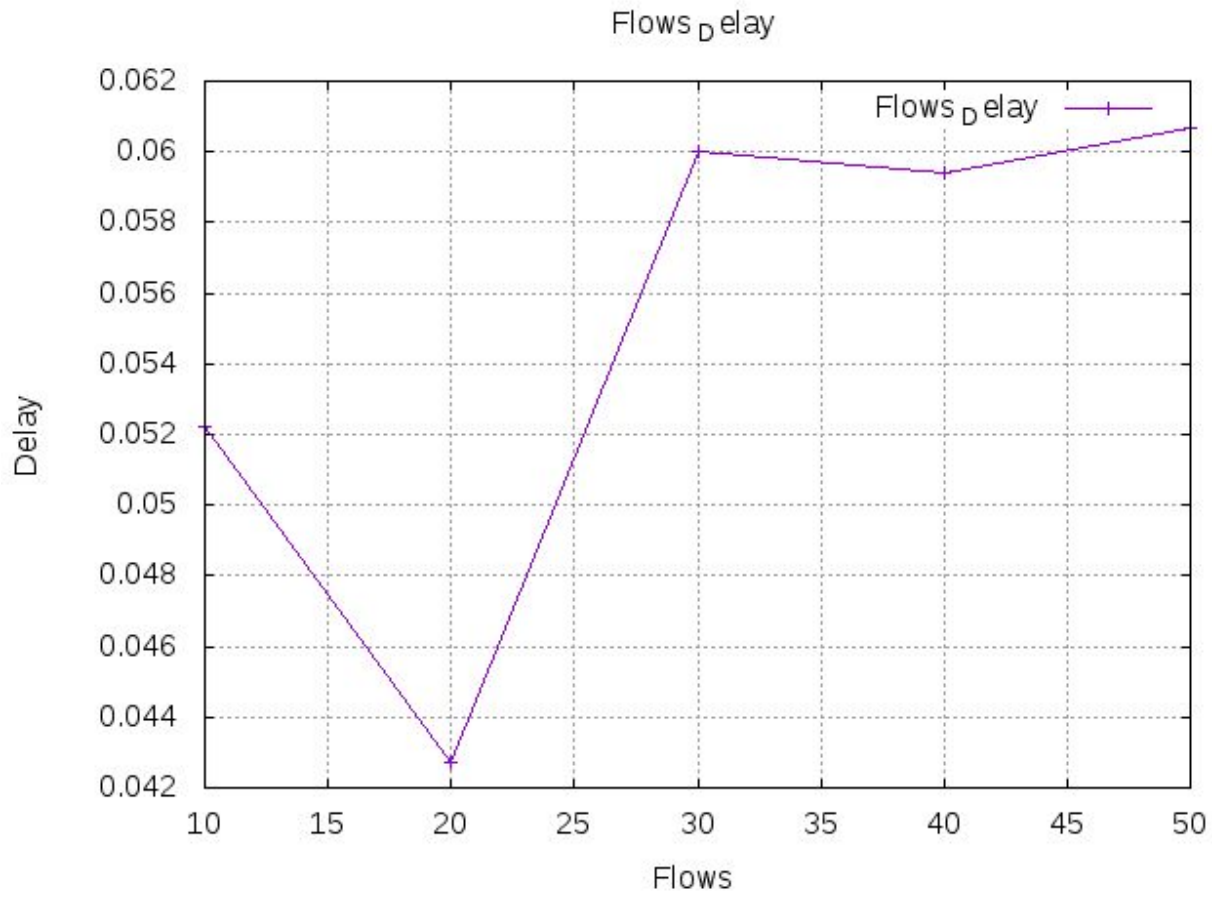

Flows Energy
Flows Energy
Energy
Flows
0
1
2
3
4
5
6
7
10
15
20
25
30
35
40
45
50

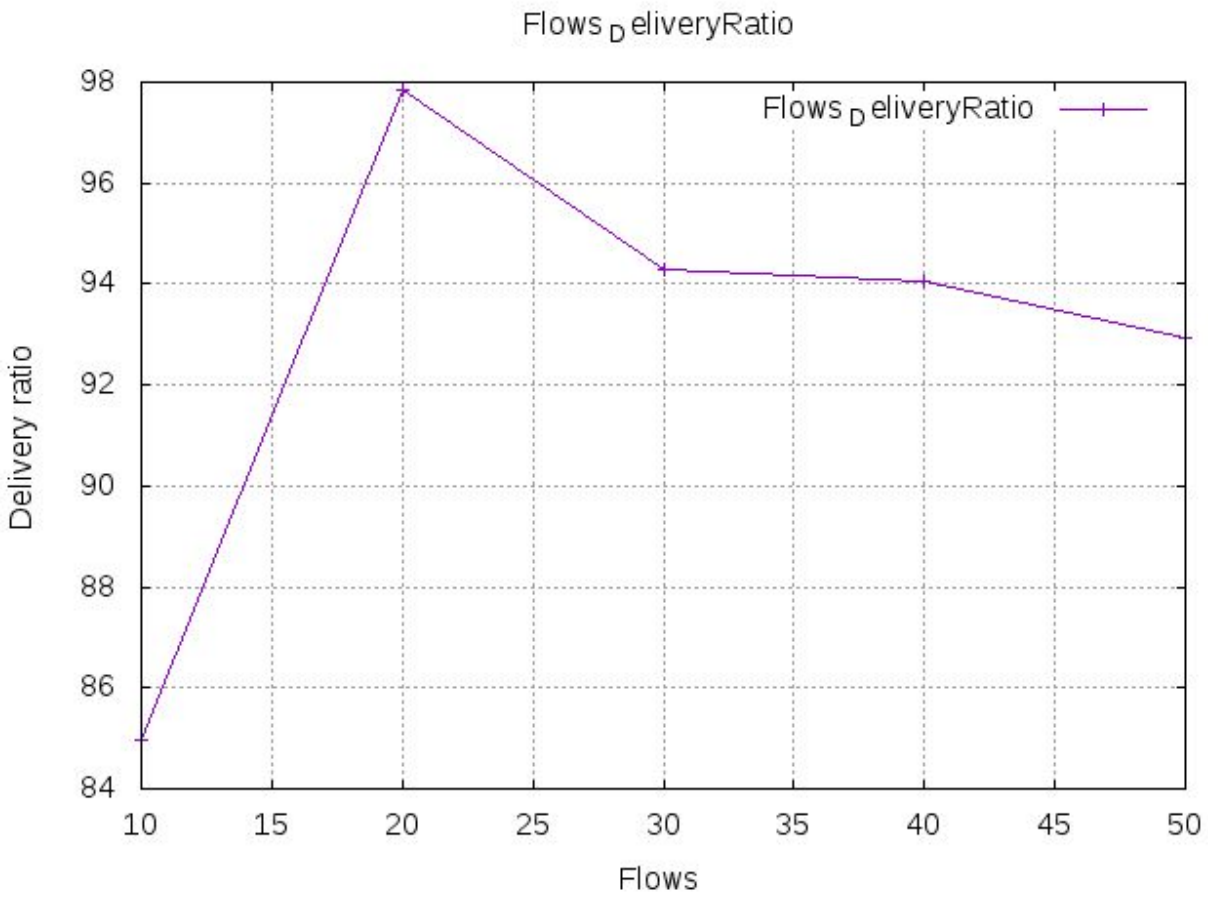

Flows DropRatio
Flows DropRatio
Drop ratio
Flows
30
25
20
15
10
5
10
15
20
25
30
35
40
45
50


Flows DeliveryRatio
Flows DeliveryRatio
Delivery ratio
Flows
100
95
90
85
80
75
70
65
60
55
10
15
20
25
30
35
40
45
50

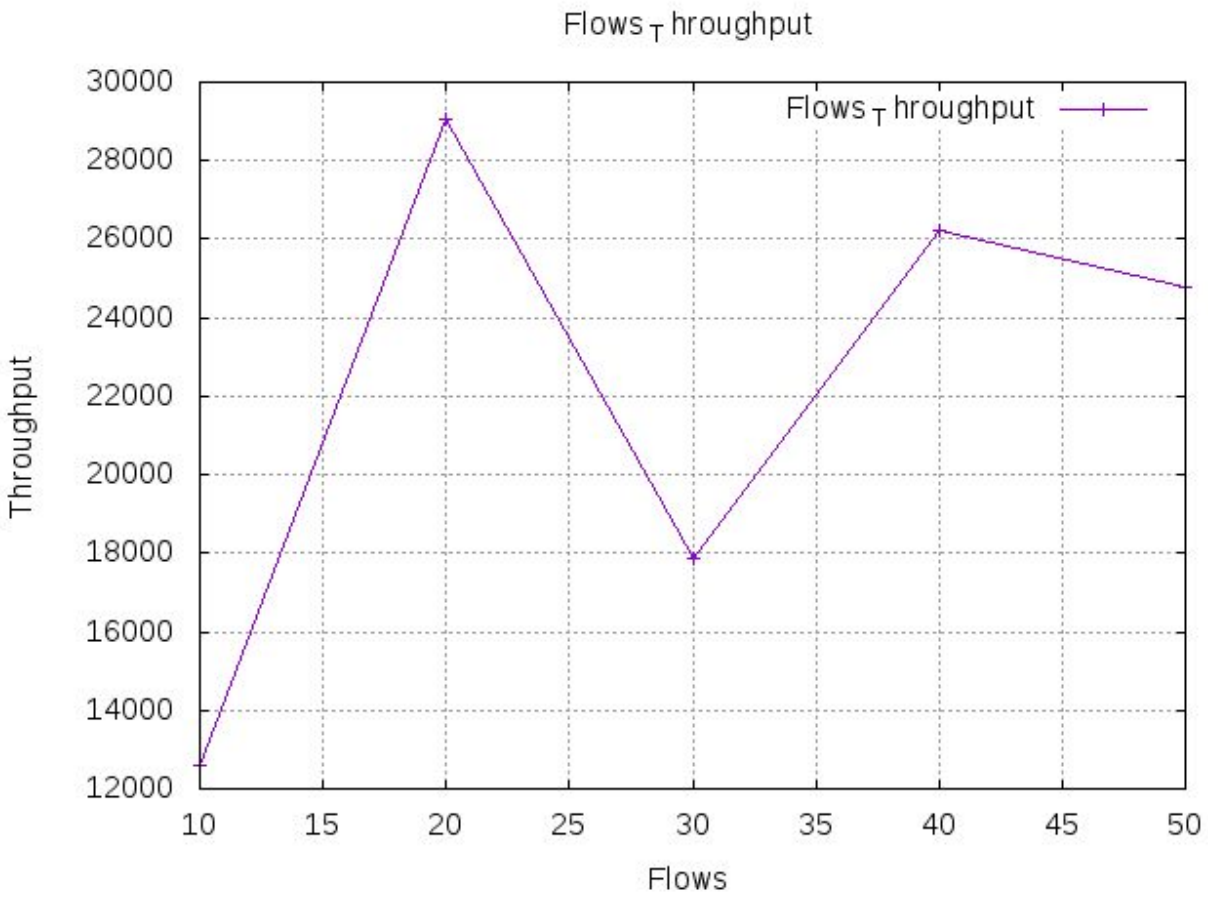

Flows Delay
Flows Delay
0.05
0.045
0.04
0.035
0.03
0.025
Delay
10
15
20
25
30
35
40
45
50
Flows

After modifications made in the simulator

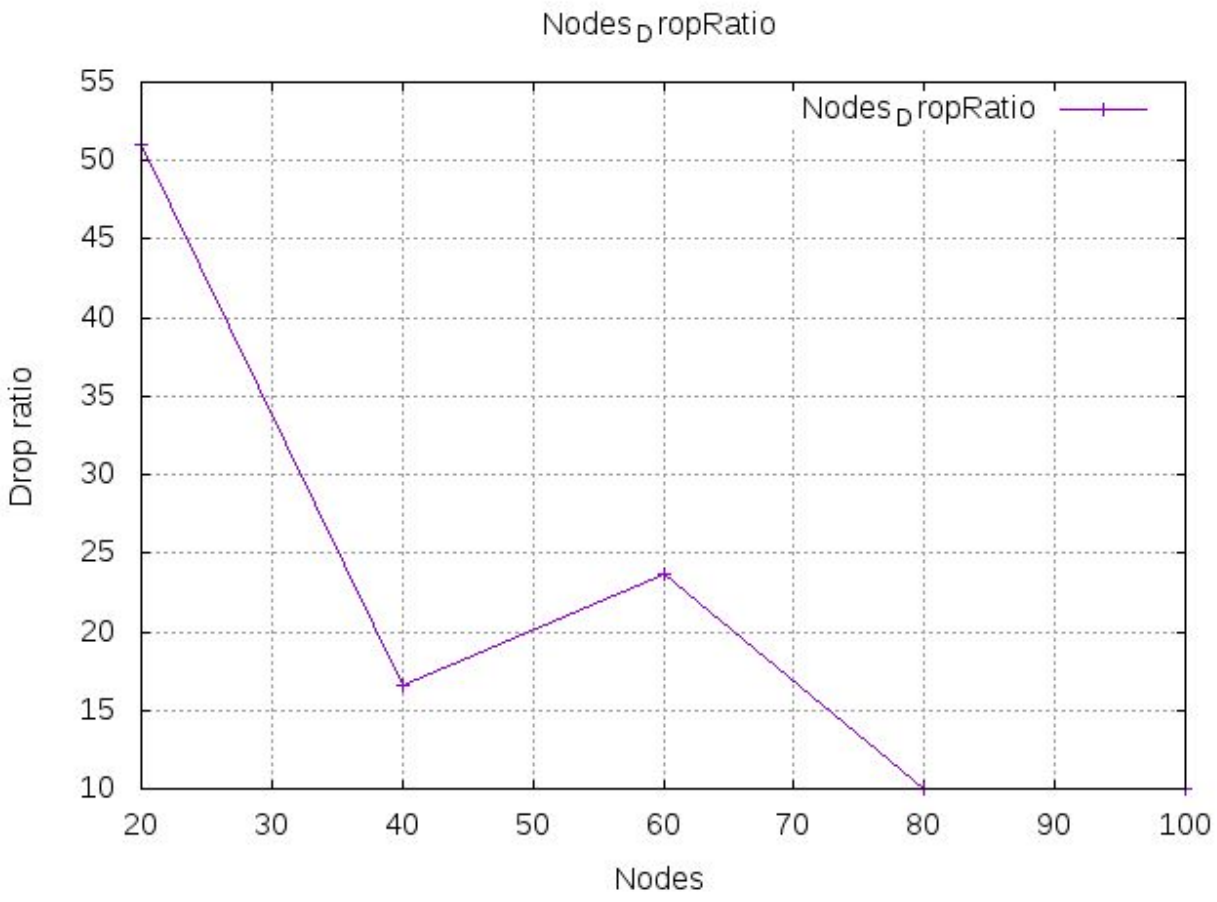


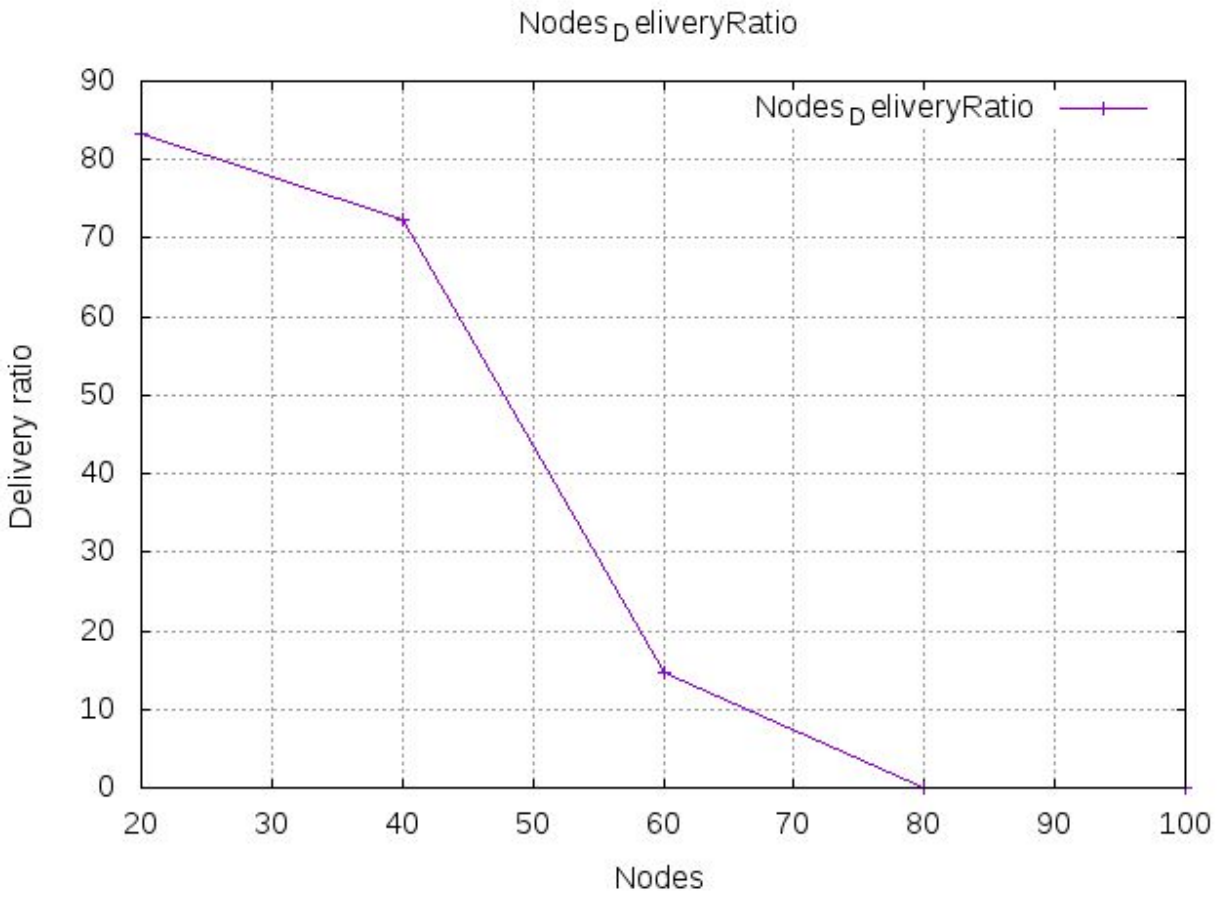


Massive performance changes. Before the modifications were made graph was started from 56 now it is starting from 87.

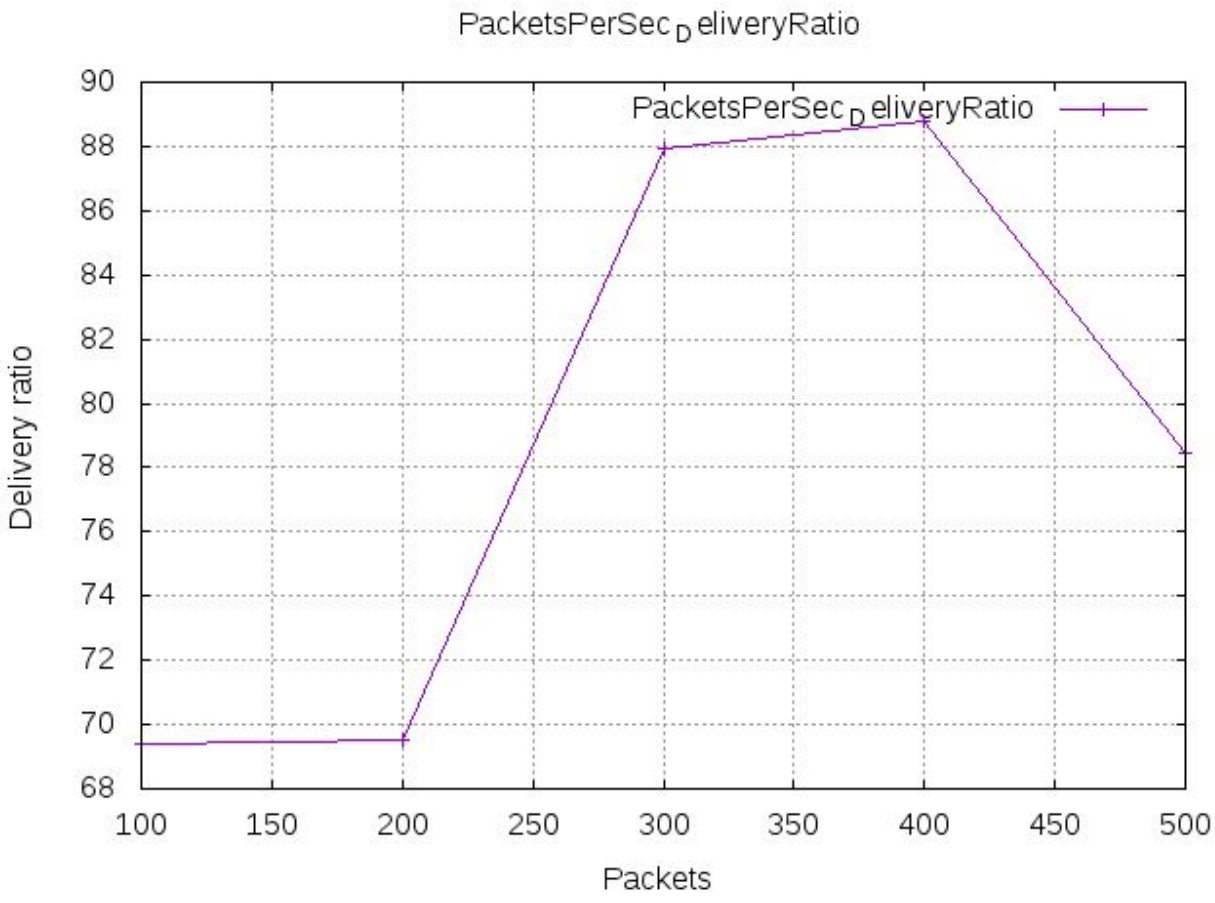


Here throughput is starting from 12750 but before that it was starting from 7000.


NodesDropRatio
NodesDropRatio
Drop ratio
Nodes
55
50
45
40
35
30
25
20
15
10
20
30
40
50
60
70
80
90
100

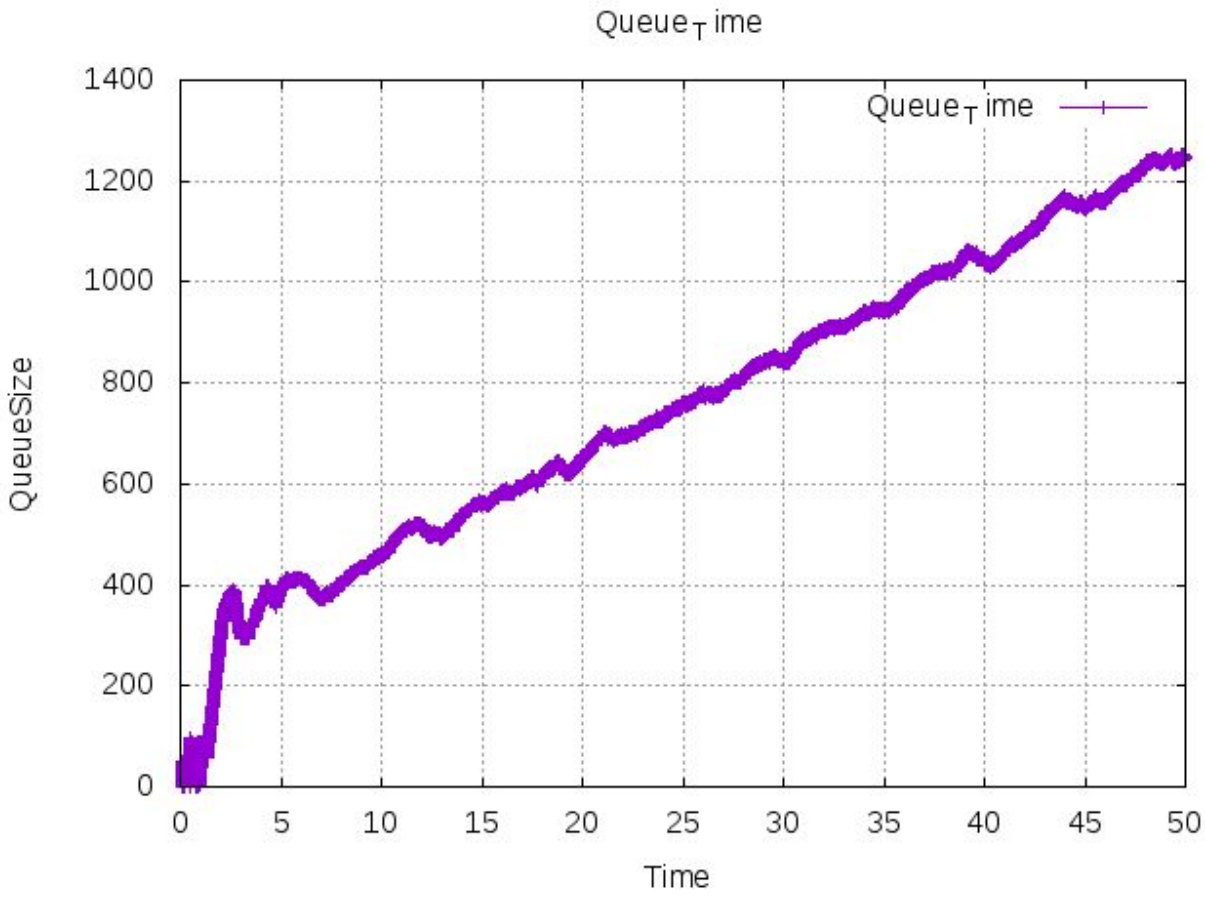


Node vs Delivery ratio is dropping whereas node vs drop ratio is increasing.

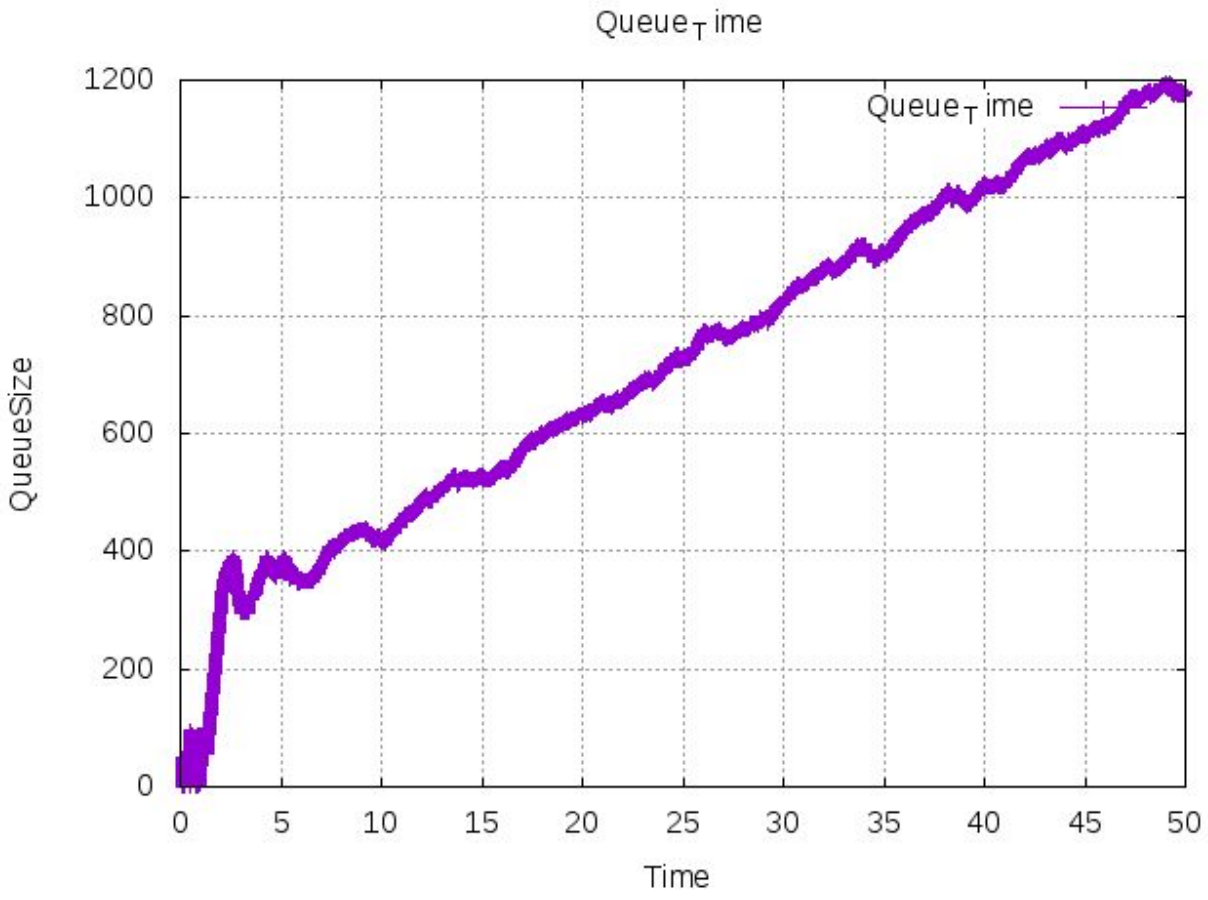


Packet Rate vs Delivery Ratio is providing us better values at some points.

QUEUE SIZE VARIATION OVER TIME (using NS2's default calculate_p_new() function in RED queue)



After using our piecewise function



Average queue size over time is lowered by a great deal.