Northeastern University

Wireless Sensor Networks

Charalampos Papadopoulos

001187462

HW1

Report

We use ns-2 to study the throughput of a multi-hop IEEE 802.11 network. In the course, we listed several factors that would affect the throughput, 1) number of nodes, 2) the back-of mechanism for contention resolution, 3) use of ACK and 4) packet size and use of RTS/CTS etc. The purpose of this assignment is to evaluate the impact of some of these factors.

We construct a simulation scenario with (n = 49) nodes in a multi-hop network. A constant bit rate (CBR) source on top of UDP transport is used as the source. For this experiment, we set the bit rate of the source to 80 kbps. We use CBR source, set the packet size to 500 bytes and set the packet interarrival time to yield 80 kbps. We choose AODV routing protocol and set the data rate at the physical layer to 2 Mbps. Dimension of the terrain is 490X490 and the sensors are located 70 meters apart at x and y coordinates. We create a grid with sensors located 70 meters apart from one another. Throughput is defined as the total number of application layer bits received. We measure the throughput and end-to-end delay with respect to the following parameters. We vary the number of sender/receiver pairs (with all sources transmitting at the same rate).

Question A

We randomly choose sender/receiver pairs among 49 sensors and vary the number of sender/receiver pairs from 1 to 20. All the sources start transmitting at 5 s and stop at 25 s. For each randomly chosen pairs we run the experiment with the parameters as above for 10 times (we do not use different random seed values due to lack of time) and we compute the average to get the final results. We plot the throughput and average delay versus the number of pairs below.

We need to keep the rate of generating cbr packet (R_cbr) equal to 80kbps in all simulations.

Supposing that our packet size is x, that means it contains x*8 bits. Therefore the interval must be T = x*8/R_cbr. For our case, x is equal 500 B and R_cbr = 80kbps then T = 500*8/80000 = 0.05 second.

Each simulation runs automatically with a script and lasts for about 10 minutes. We obtain the following table. Delay is measured in seconds and throughput in kbits. Delay is computed as the summation of travel times ($t_{receive} - t_{transmit}$) of the packets that managed to arrive at the destination nodes, divided by that number of packets. Packets that are dropped are not taken into account for the scope of this exercise. This could make a difference, especially when comparing using RTS/CTS with not using it. However, most scripts for computing the delay online did not take into account dropped packets.

Throughput is computed as the total number of bits received at the receive nodes.

#pairs	delay	throughput
1	0.0959	3264
2	0.0924	6509
3	0.1414	9564
4	0.1810	12472
5	0.2130	15306
6	0.3800	18624
7	0.4336	20588
8	1.1255	23342
9	1.4763	23733
10	1.9363	27145
11	2.1623	27439
12	2.3827	28846
13	2.8327	30291
14	3.2285	32055
15	3.3662	31952
16	3.7628	34553
17	4.1332	35632
18	3.8374	37683
19	4.1892	39847
20	4.3531	40182

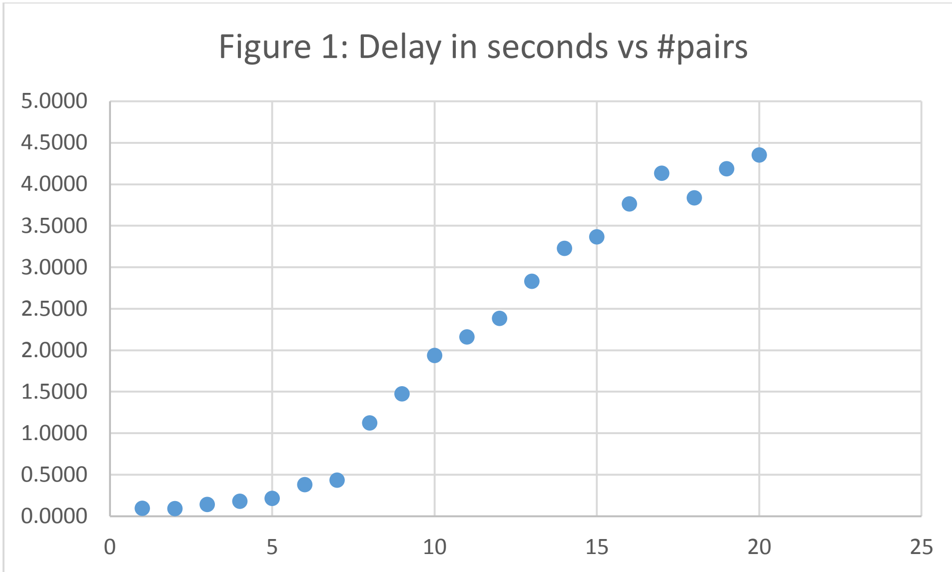


Figure 1: Delay in seconds vs #pairs
5.0000
4.5000
4.0000
3.5000
3.0000
2.5000
2.0000
1.5000
1.0000
0.5000
0.0000
0
5
10
15
20
25

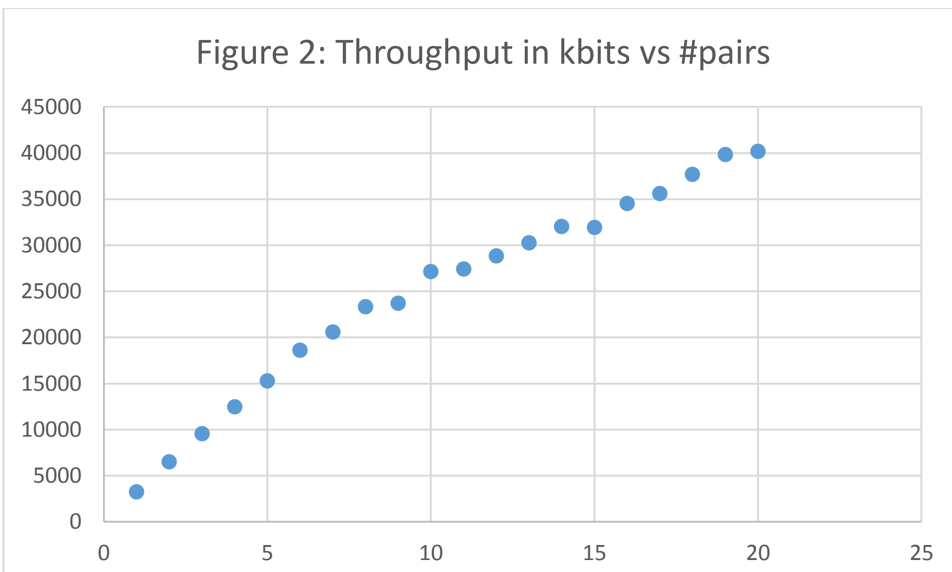


Figure 2: Throughput in kbits vs #pairs
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0
5
10
15
20
25

Question B

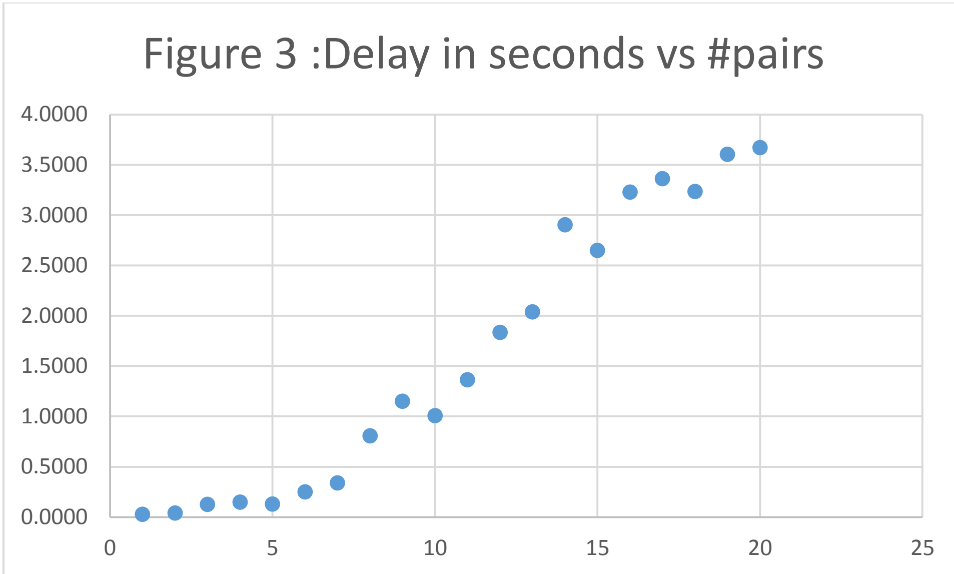
We repeat the experiment a. with RTS/CTS disabled. We do that by inserting this line in our script:

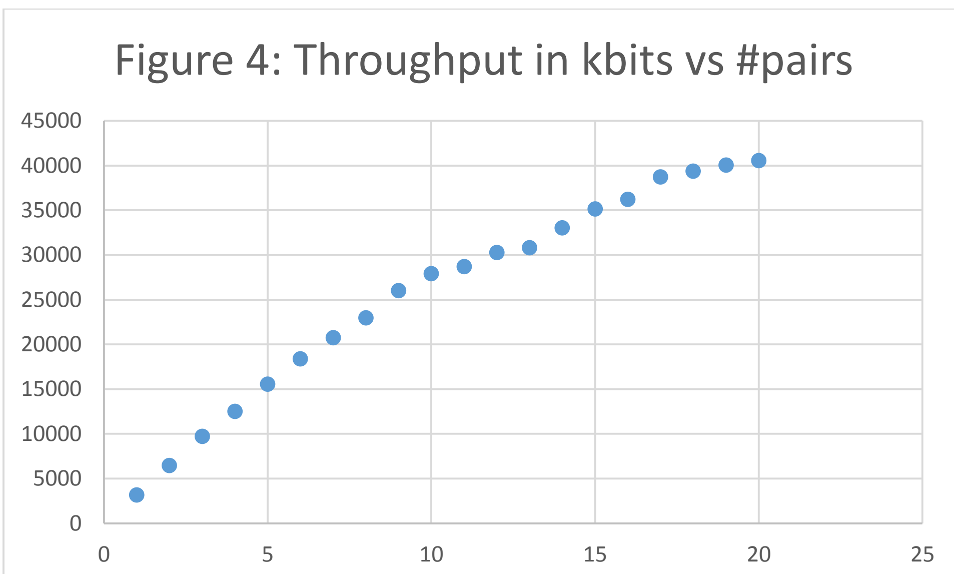
Mac/802_11 set RTSThreshold_   3000

Our new table is shown below:

#nodes	delay	throughput
1	0.0284	3181
2	0.0415	6470
3	0.1252	9714
4	0.1470	12511
5	0.1308	15551
6	0.2509	18406
7	0.3404	20764
8	0.8066	22974
9	1.1501	26020
10	1.0080	27911
11	1.3656	28721
12	1.8357	30275
13	2.0398	30833
14	2.9032	33031
15	2.6510	35136
16	3.2279	36243
17	3.3613	38720
18	3.2367	39366
19	3.6052	40065
20	3.6713	40557

We plot again throughput and average delay versus the number of pairs below.





Conclusion: We may observe some enhancements in both throughput and delay, related to question a. But there are other advantages in using RTS/CTS that are not shown from these graphs.

Question C

We repeat the experiment a. and b. by varying the size of application packets. We use packet size (100; 300; 1000) and we set the packet interrarival time to yield 80kbps. We did not simulate again for packet size equal to 500 because we did that in the two previous questions.

Packet size 100 with enabled RTS/CTS

#nodes	delay	throughput
1	0.0082	3171
2	0.0402	6859
3	0.3755	9225
4	0.5655	10916
5	0.6390	12745
6	0.9803	14572
7	1.1214	15890
8	1.3076	17196
9	1.5124	18746
10	1.6587	20554
11	1.7569	21347
12	1.9241	23442
13	2.1602	25002
14	2.3539	26279
15	2.4496	27345
16	2.5844	29537
17	2.8121	30750
18	2.9150	32417
19	3.0007	34233
20	2.9868	35379

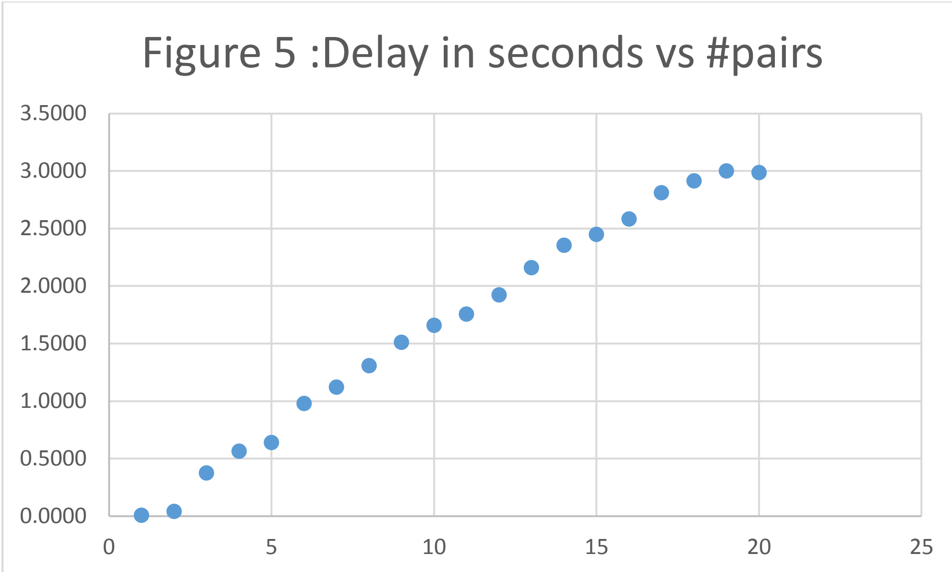


Figure 5 :Delay in seconds vs #pairs
3.5000
3.0000
2.5000
2.0000
1.5000
1.0000
0.5000
0.0000
0
5
10
15
20
25

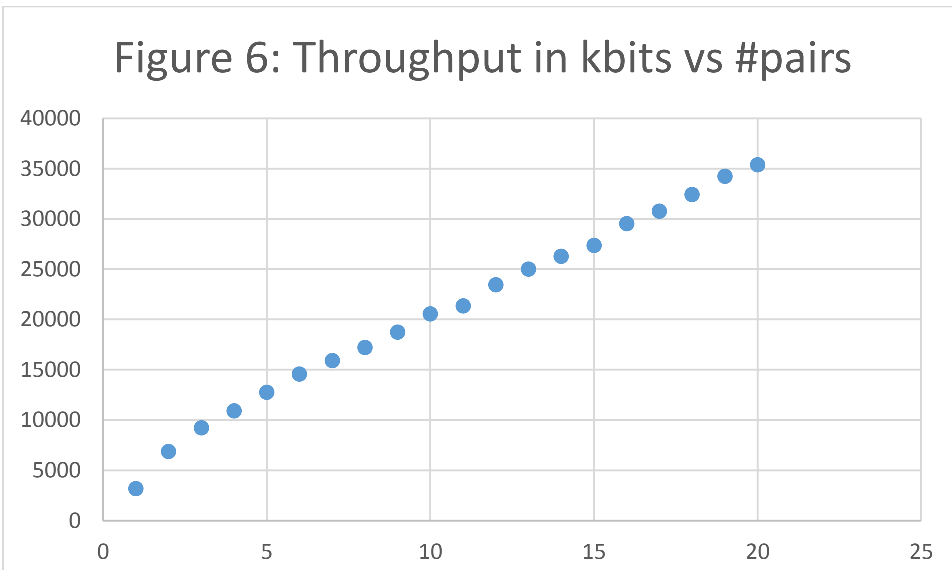


Figure 6: Throughput in kbits vs #pairs
40000
35000
30000
25000
20000
15000
10000
5000
0
0
5
10
15
20
25

Packet size 100 with disabled RTS/CTS

#nodes	delay	throughput
1	0.0068	3462
2	0.0130	6470
3	0.1045	9946
4	0.0931	12960
5	0.2155	15865
6	0.5590	17004
7	0.7445	17984
8	0.8993	18987
9	0.9907	20921
10	1.1319	23005
11	1.2756	23802
12	1.3430	25787
13	1.6171	26542
14	1.8798	27629
15	1.5278	30466
16	1.7498	31636
17	1.8514	33033
18	1.8381	34981
19	2.0493	35445
20	2.5505	36755

Figure 7 :Delay in seconds vs #pairs

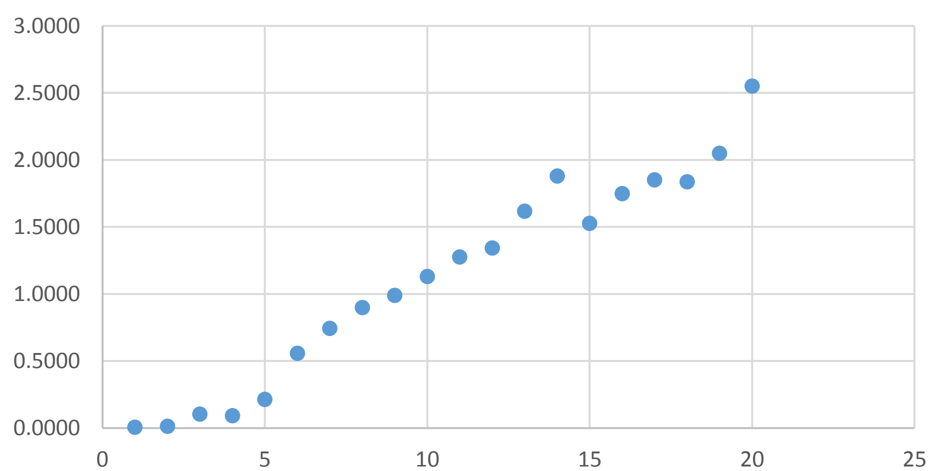
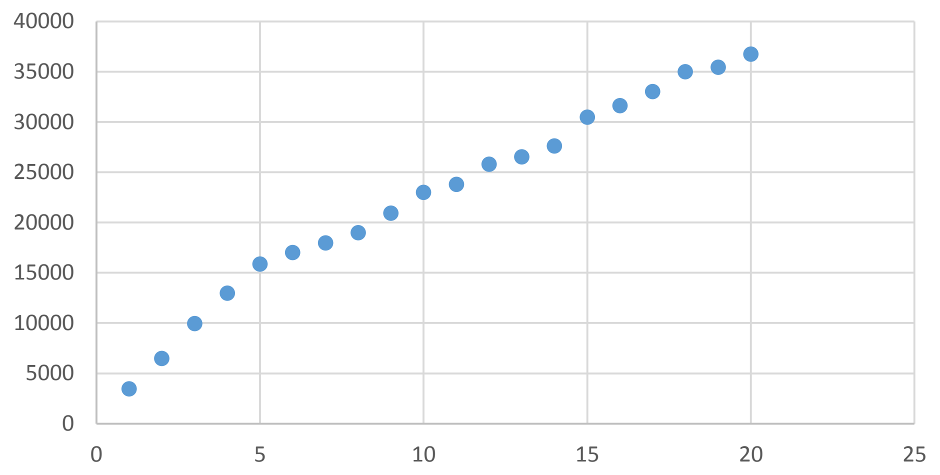


Figure 8: Throughput in kbits vs #pairs



Packet size 300 with enabled RTS/CTS

#nodes	delay	throughput
1	0.0355	3247
2	0.0396	6555
3	0.0754	9657
4	0.0924	12068
5	0.3727	15174
6	0.4091	17524
7	0.7935	20231
8	1.7218	19912
9	1.4696	22413
10	1.7505	24645
11	1.9786	25658
12	2.1309	27206
13	2.7796	27738
14	2.5493	29955
15	2.7740	31628
16	3.3618	33054
17	3.0744	34901
18	3.7254	35033
19	3.4997	37096
20	3.9924	38357

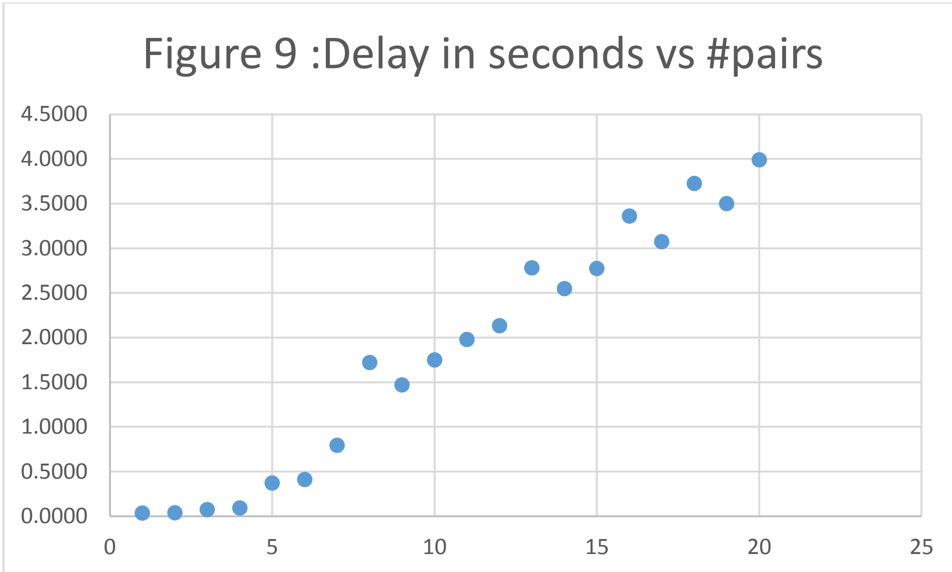


Figure 9 :Delay in seconds vs #pairs
4.5000
4.0000
3.5000
3.0000
2.5000
2.0000
1.5000
1.0000
0.5000
0.0000
0
5
10
15
20
25

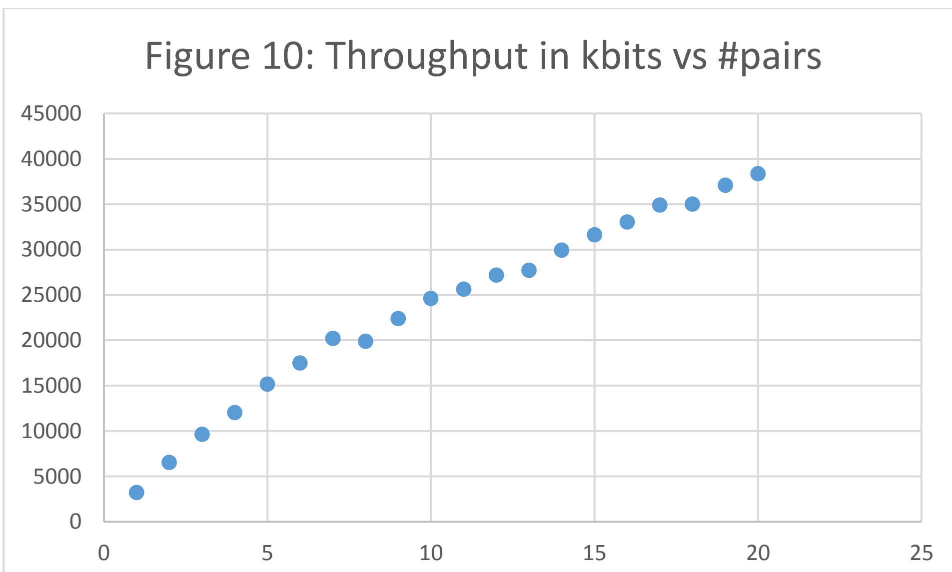
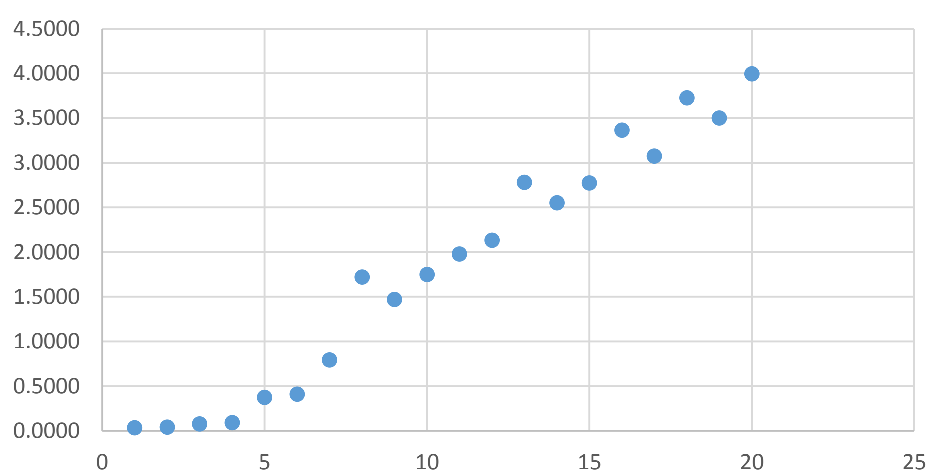


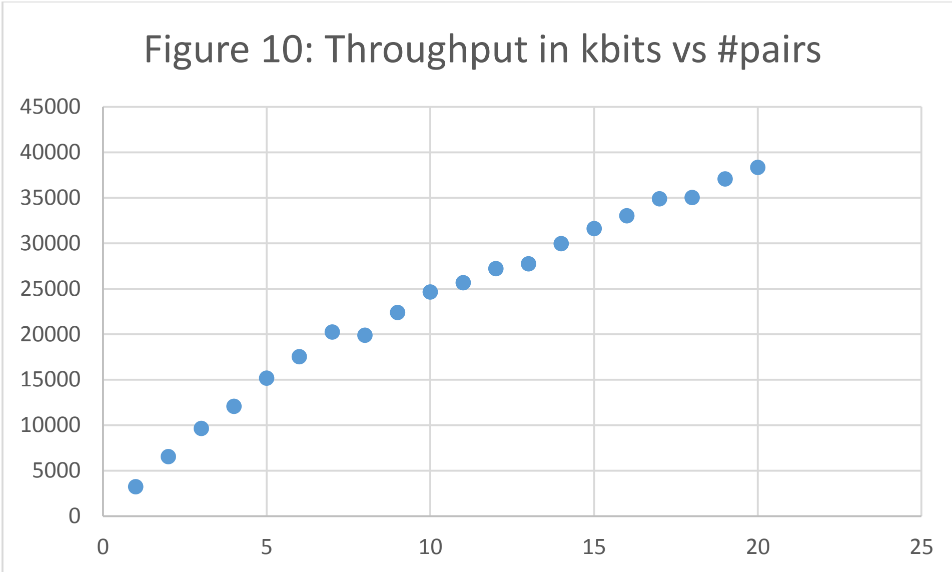
Figure 10: Throughput in kbits vs #pairs
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0
5
10
15
20
25

Packet size 300 with disabled RTS/CTS

#nodes	delay	throughput
1	0.0355	3247
2	0.0396	6555
3	0.0754	9657
4	0.0924	12068
5	0.3727	15174
6	0.4091	17524
7	0.7935	20231
8	1.7218	19912
9	1.4696	22413
10	1.7505	24645
11	1.9786	25658
12	2.1309	27206
13	2.7796	27738
14	2.5493	29955
15	2.7740	31628
16	3.3618	33054
17	3.0744	34901
18	3.7254	35033
19	3.4997	37096
20	3.9924	38357

Figure 9 :Delay in seconds vs #pairs





<u>Packet size 1000 with enabled RTS/CTS</u>

#nodes	delay	throughput
1	0.0177	3278
2	0.0436	6539
3	0.0498	8916
4	0.0953	12821
5	0.1024	15389
6	0.3309	18370
7	0.5847	20990
8	0.3899	22945
9	0.7915	25349
10	1.2021	26105
11	1.1207	28403
12	1.6969	28654
13	2.0445	29913
14	2.0168	32923
15	2.4442	33000
16	2.2589	35073
17	2.2318	37260
18	2.7744	37501
19	3.0213	38477
20	2.8630	40207

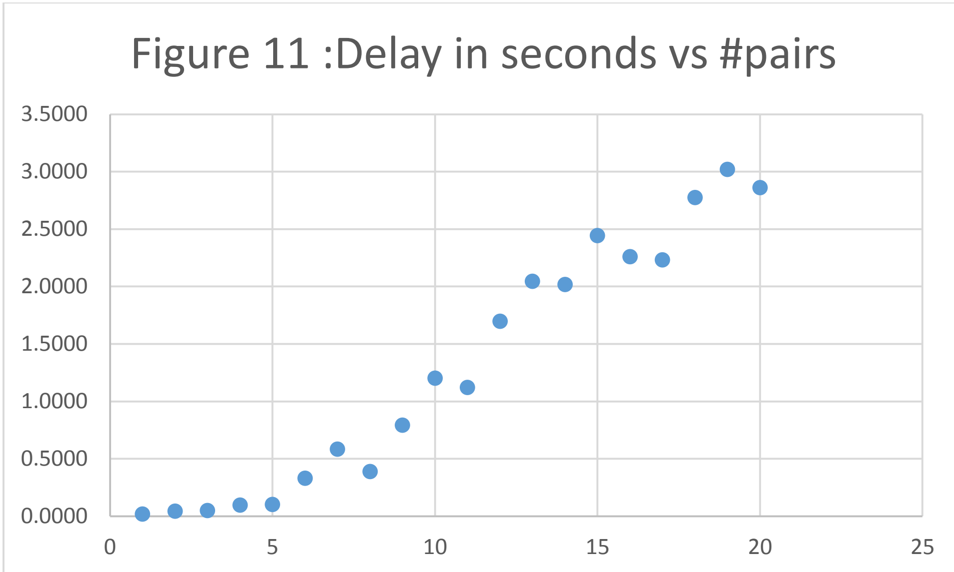


Figure 11 :Delay in seconds vs #pairs
3.5000
3.0000
2.5000
2.0000
1.5000
1.0000
0.5000
0.0000
0
5
10
15
20
25

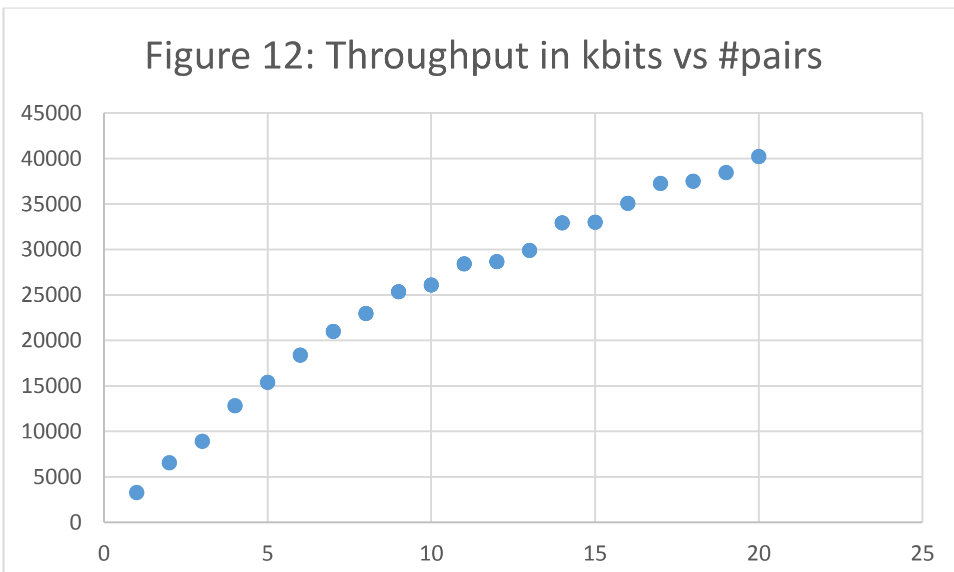
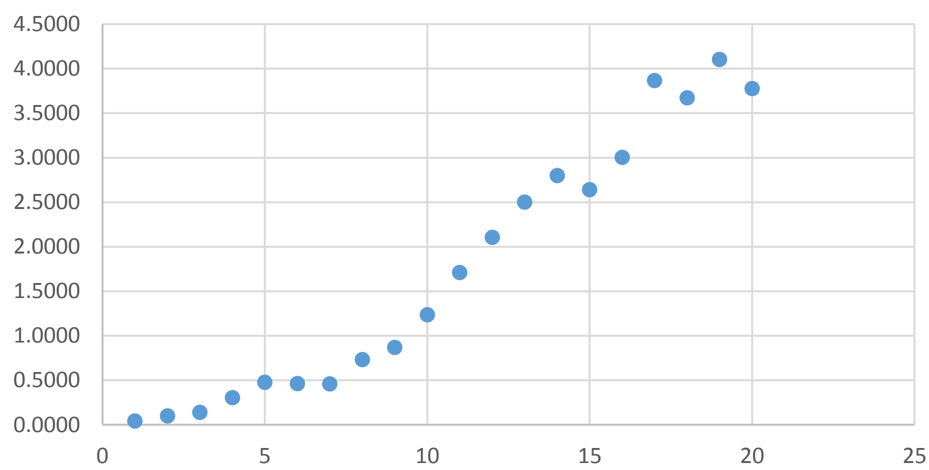


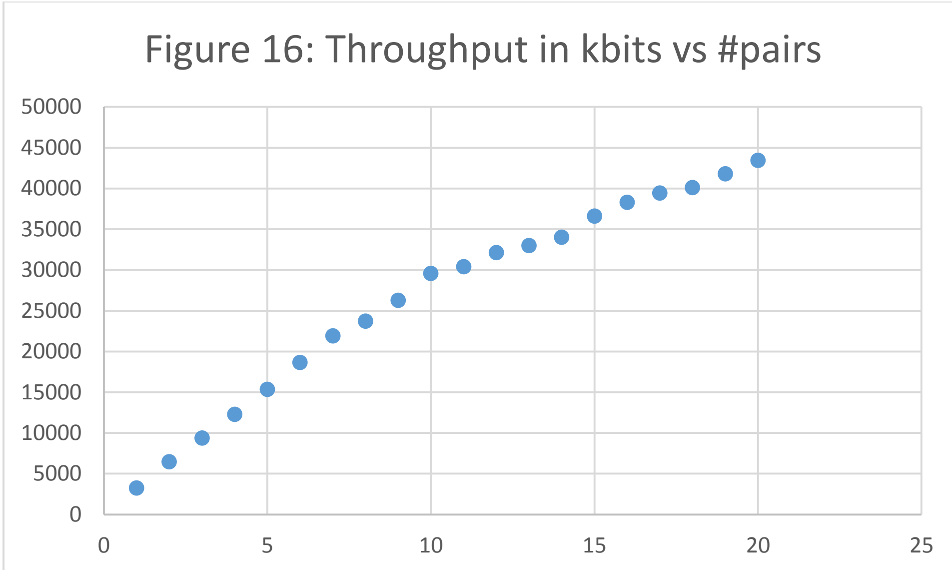
Figure 12: Throughput in kbits vs #pairs
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0
5
10
15
20
25

Packet size 1000 with disabled RTS/CTS

#nodes	delay	throughput
1	0.0452	3232
2	0.1013	6464
3	0.1414	9370
4	0.3042	12284
5	0.4794	15339
6	0.4643	18639
7	0.4596	21927
8	0.7352	23706
9	0.8706	26282
10	1.2356	29574
11	1.7114	30400
12	2.1071	32140
13	2.5014	32996
14	2.7991	34040
15	2.6431	36623
16	3.0042	38287
17	3.8689	39460
18	3.6756	40094
19	4.1042	41808
20	3.7767	43437

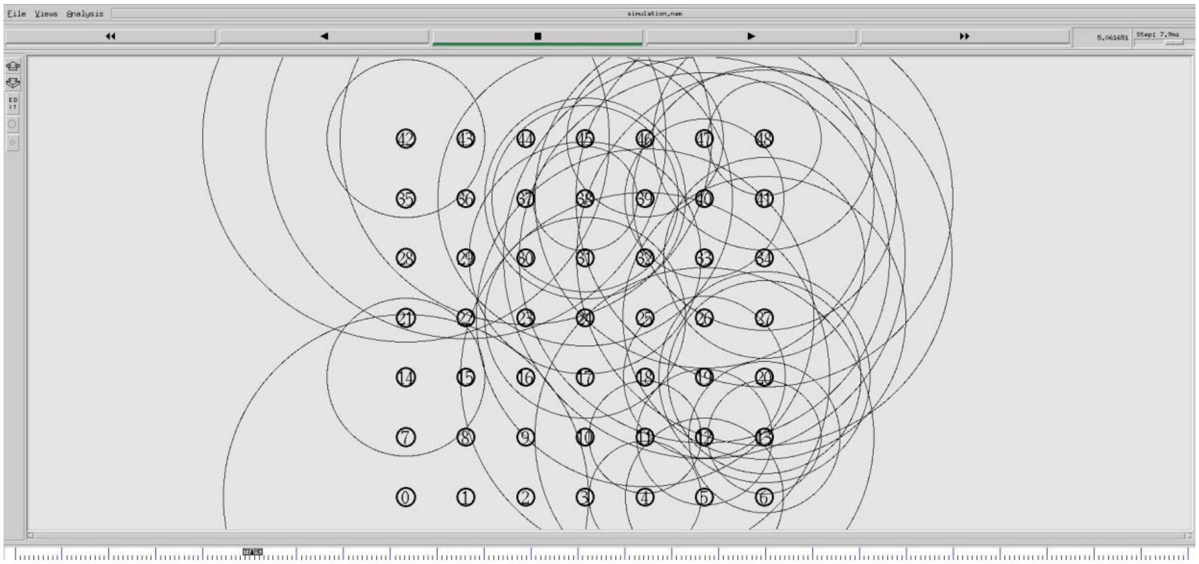
Figure 15 :Delay in seconds vs #pairs





Conclusions

- From all the above simulations we may see that enabling RTS/CTS decreases the throughput and increases the delay times.
- We may observe that the relationship between throughput and number of nodes is linear, but the slope is reduced approximately above 10 pairs.
- We can conclude that delay grows slowly, linearly, up to 5 pairs. Then it starts to increase faster, linearly again.
- From the above we can say that this network is starting to get saturated somewhere between 5-10 nodes.
- We may observe that the bigger the packet size, the greater delays we experience, for a stable packet interarrival time to yield 80kbps.
- However, the bigger the packet size, the more throughput we achieve.
- A nice picture of our simulation topology can be seen below.



File Views Analysis
simulation.nam