

Ethernet Simulator Results

The purpose of this project is to simulate an Ethernet. This Ethernet network can have a varying number of nodes, referred to as N . All of the nodes will have the same average number of frames available to transmit per second, λ (lambda). This document will provide an explanation of what happens when either N changes or λ changes. The raw data from the simulations can be viewed in a separate document (RawData.pdf).

In this project two sets of simulations were run. The first was a set of simulations that kept λ the same, but increased the number of nodes connected to the network. The number of nodes connected to the network began at only two and increased by a node until there were twenty nodes connected to the network. The second set of simulations had a constant number of nodes, but λ would increase. The initial simulation had each node operating with a λ of 100. Each simulation in the set increased λ by 100 until it reached 2000.

Increasing Nodes and Network Performance

First, a look at how the network performs as the number of nodes connected to it changes. Knowing how an Ethernet is supposed to behave can lead one to make assumptions on network performance with an increasing number of nodes. One would assume that as the number of nodes on the network increases there will be an increase in the number of collisions. One would also assume that as the number of nodes increases there will be an increase in throughput of the entire network for a while. There would come a point in which the throughput of the network will peak or plateau when either the throughput equals approximately 100% of the bandwidth or when there are too many collisions on the network to allow nodes to efficiently send frames.

When looking at the actual results from the simulation, the reported throughput seems reasonable. With only two nodes on the network, throughput is approximately 10%. As the number of nodes increases to twenty, the throughput increases in a linear fashion. With twenty nodes on the network the throughput is approximately 95%.

The collision percentage is more confusing and is not consistent with how one would assume the Ethernet to perform. With only two nodes on the network there is a less than 1% collision percentage, which seems reasonable. As the number of nodes increases to eight, the collision percentage also increases, to about 12.5%. As the number of nodes increases from eight to twenty the collision percentage actually decreases in an exponential fashion. With twenty nodes on the network the collision percentage is slight more than 1%. Again, these results are not consistent with our assumptions of Ethernet performance.

In attempting to determine why the collision percentage starts decreasing information about each node had to be analyzed. When a collision was detected on my simulated Ethernet I would have each node involved in the collision output to the screen it's back-off wait time, the amount of time it would wait until it would attempt to resend the frame. What I discovered was that a number of nodes would have back-off values of more than 150,000 microseconds. Due to these nodes having such large back-off values they would be idle for a significant period of time, essentially reducing the number of active nodes on the network. This analysis makes sense and explains what at first is incorrect results.

The scope of this project was to look at the network's performance as the number of nodes increases from two nodes to twenty. With the somewhat unusual collision percentage, I wondered if the exponential decrease is consistent as the number of nodes continues to increase. With 100 nodes the collision percentage is at 4.5%. So it does start to increase at some point. Further simulations are needed to see at what point the collision percentage increases and to what degree the increase takes.

Increasing λ and Network Performance

The second set of simulations looked at how the Ethernet performs as a constant number of nodes increases in the average number of frames that are ready to be transmitted every second. In thinking about this situation theoretically results would be similar as in the first part. Initially as the average number of frames to be sent increases the throughput of the network would increase as will the number of collisions. This increase would be finite. At some point, one would believe that the throughput will peak or plateau either because the maximum throughput of the network has been reached or there are so many frames that need to be transmitted many nodes' frames begin colliding bringing down the throughput.

In making these theoretical assumptions that are so similar to the first set of tests, perhaps the assumptions of how the network will behave will turn out just as wrong as in the first set of tests. As a matter of fact, the Ethernet's performance as λ increases looks very much similar to the results of if the number of nodes on the network increases.

The tests involving increasing λ held the number of nodes constant at 10. When each of these nodes has an average number of frames per second to send, λ , the throughput is at about 5%. As λ increases, from 100 to 2000, so does the throughput. The increase in throughput appears linear in nature and at λ of 2000 the throughput is at just over 90%.

Just as in the first set of simulations, the collision percentage doesn't come out as expected. Initially, as λ grows, from 100 to 600, so do the percentage of collisions. The collision percentage peaks at almost 14% when λ is between 600 and 700. As λ grows beyond 700 frames per second the collision percentage has an exponential decrease. By the time λ is 2000 frames per second the collision percentage is under 1%.

The explanation for why the collision percentage decreases as λ increases is much the same as when the number of nodes on the network increases. There comes a point in which a number of nodes has frequent collisions causing their back-off values to reach high amounts. The nodes essentially become idle on the network. This can explain why the collision percentage decreases and the throughput continues to increase even though the nodes are trying to send more frames every second.

As in the previous section, the question is if this trend continues as λ increases further. A spot check was made and it appears that as λ reaches 4000 frames per second the throughput continues to increase, reaching 99%, and the collision percentage continues to decrease, to 0.4%. This would be consistent with an exponential decline in the number of collisions. More tests would be needed at higher rates of λ to see where the throughput may level off or if the collisions may increase.