Andrew Camps

Derek Paris

October 10, 2017

ECE 478 Project 1

Loukas Lazos

**The Distributed Coordination Function of 802.11**

As a team Derek Paris and Andrew Camps studied the performance of multiple access protocols in a wireless setting. The project studied two different topological orderings of nodes, two different setups of Carrier Sensing Multiple Access (CSMA), all at different packet rate distributions. To complete this project as a team we decided to meet up on multiple occasions and work on creating a simulation for each test. We worked together to list out the many possible edge cases, then design a simulation which would deal with each case appropriately. After working out much of the needed logic, Andrew began to code the simulations for the first few scenarios. Derek and Andrew peer programmed many of the more logically intense edge cases. Derek code reviewed the rest of the code periodically as it was written.

In order to test all of the different scenarios in the project our team decided that we would develop this simulation is C/C++. The simulation consisted of using a single main.cpp file which, when compiled and run, outputted to the terminal the necessary data to compile the graphs in the figures below. Each time the program was run, the simulation was conducted 1000 times for each of the many scenarios. The data retrieved from each of these trials was then averaged and outputted. This data included: throughput averages, fairness index, collision average, total slots used averages, and success averages. The program was broken down into 4 major functions, where both topologies, node in either the same or different collision domains, each used CSMA and CSMA (with VCS) functions. Each of these 4 functions was run at different transmission rates. These rates we tested in two different sets. The first transmission rate set was a set of 50, 100, 200, and 300 pkts/sec for both nodes A and C. The second set doubled the transmission rates of node A, but left the rates for node C unchanged. See below for the graphical representation of each simulation.

Figure 1- Average Throughput of Node A While λA = λC

Figure 2- Average Throughput of Node A While λA = 2λC

Figure 3- Average Throughput of Node C While λA = λC

Figure 4- Average Throughput of Node C While λA = 2λC

Figure 5- Average Collisions of Node A While λA = λC

Figure 6- Average Collisions of Node A While λA = 2λC

Figure 7- Average Collisions of Node C While λA = λC

Figure 8- Average Collisions of Node B While λA = 2λC

Figure 9- Fairness Index While λA = λC

Figure 10- Fairness Index While λA = 2λC

Figure 11-

Figure 12-

Figure 13-

Figure 14-

Figure 15-

Figure 16-

Figure 17-

Figure 18-

Figure 19-

Figure 20-