Simulation of a Wireless Network Using The 802.11 MAC Protocol

Final Report

Pavan Pant
U85160073

Thomas Castelli
U18104565

I. Introduction

What are Wireless Local Area Networks (WLANs), and more importantly, how do they work? Before delving into the mundane technical jargon that is usually associated with wireless networks, we will attempt to provide a more simplistic definition. WLANs are Local Area Networks (LANs) that do not use traditional cable or wired systems. Instead, they use Radio Frequency and Infrared (IR) to connect computers and peripherals over a network.

This project was part of an independent study and simulates an 802.11 wireless network using MATLAB. The purpose of this document is to provide readers an idea of how we approached certain problems and implemented the 802.11 MAC protocol. Additionally, this document contains an explanation of the rationale behind some of our design choices and some interesting data that was extracted from the simulation.

Some of the key terms that will be used throughout the paper and their definitions are as follows:

1) RTS: When a node has a data packet to transmit, it first broadcasts a small packet called a Request-To-Send (**RTS**) packet.

2) CTS: When the intended recipient on the WLAN receives an RTS packet, it broadcasts a small Clear-To-Send (**CTS**) packet.

3) ACK: Acknowledgement packet sent from the receiver to the sender upon successful completion of the data exchange.

4) CSMA/CA: Carrier Sense Multiple Access with Collision Avoidance (**CSMA/CA**) is a protocol in which (a) a carrier sensing scheme is used, (b) a data station that intends to transmit sends a jam signal, (c) after waiting a sufficient time for all stations to receive the jam signal, the data station transmits a frame, and (d) while transmitting, if the data station detects a jam signal from another station, it stops transmitting for a random time and tries again.

II. Background and Motivation

The main motivation for this project is to improve upon the simulation already authored by BU students Keyong Li and Mohammad Sherzai as their final project for this course during the fall semester of 2001. Li and Sherzai discuss some of the problems with the 802.11 protocols and some possible solutions to these problems (such as RTS/CTS and back-off algorithms to deal with hidden and deaf nodes). However, the MATLAB simulation written by Li and Sherzai does not include any implementation of RTS/CTS, or back off, and makes a number of assumptions about the operating environment of the WLAN.

Our primary intent was to relax some of the assumptions made by Li and Sherzai, and to add an implementation of RTS/CTS and back off to their simulation of an 802.11 wireless network. Our current simulator is slightly more comprehensive, considering we have implemented a Collision Avoidance system (CSMA/CA). The idea was to add depth to Li and Sherzai's existing research and we have successfully attained that goal.

We also received a considerable amount of guidance from Professor Starobinski, who provided us with a paper on the deaf node problem in WLANs. The paper resulted from joint research efforts between Professors Starobinski, Carruthers, and Ray. It is hoped that our work will help to guide their research and provide a direction for improvements to the simulator.

III. Approach

We decided to tackle some of the problems with WLANs through a combination of literary research and insights from Li and Sherzai's MATLAB code. Li and Sherzai had implemented CSMA without the Collision Avoidance scheme. We improved the original simulation by including RTS/CTS exchanges, which is part of the CSMA/CA protocol. Using this methodology, we hoped to increase the number of successful transmissions and decrease the number of collisions within the WLAN.

This particular aspect of the simulation is demonstrated in the Analysis section of this paper. The graph clearly shows that our simulation improves performance compared to the original simulation.

Additionally, the simulation makes sure that once an RTS/CTS handshake is completed, all other nodes not included in the data exchange remain silent. This is done to prevent nodes from getting into a continuous cycle of RTS/CTS exchanges.

IV. Implementation

The 802.11 wireless network is simulated in MATLAB using CSMA/CA as the MAC protocol. Li and Sherzai's original code was used as a basis for improvement. Our version of the simulation includes:

- Display of RTS/CTS packets
- Number of DATA and ACK packet collisions, using RTS/CTS
- Number of mobile communication stations
- Transmission range of each station
- Mobility of stations
- Average transmission time (DATA packet size).

In order to simplify our task, we have made certain assumptions about the network environment. Our assumptions are as follows:

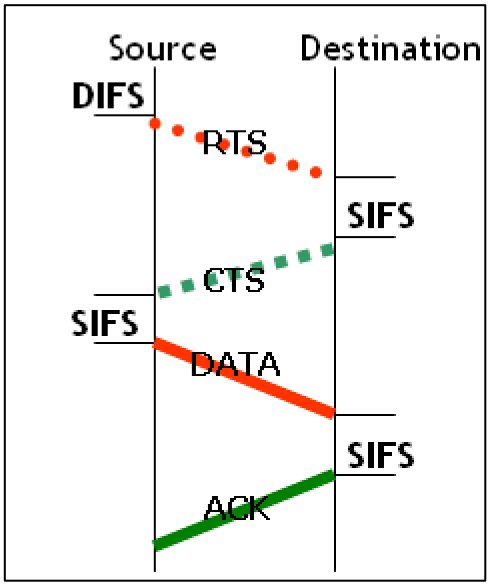
- Distributed Control Function is in use
- Propagation delay is assumed to be zero

According to the IEEE paper on the 802.11 protocol, a coordination function determines when a station is permitted to transmit and may be able to receive data through the wireless channel. A Distributed Coordination Function (DCF) is any member of a class of coordination functions where the same logic is active in every station whenever the network is in operation.

DCF is a basic medium access protocol that allows for automatic sharing between compatible Physical Layers, with the help of CSMA/CA and a random back off interval following any busy medium condition. All directed traffic uses immediate positive acknowledgment (ACK frame) where the sender schedules retransmission if no ACK has been received. CSMA/CA is used to avoid collisions between multiple stations (network nodes) accessing the medium at the point where collisions are most likely -- right after the medium becomes idle following a busy condition. This is also the reason why random back off time intervals have been implemented.

The CSMA/CA protocol specifies that for a node to transmit, it will first sense the medium around it to determine whether another node is transmitting. If the node in question finds that the medium is free, then that node will begin transmission. The CSMA/CA algorithm also mandates a delay between contiguous frame sequences. A transmitting node will make sure that the medium is idle for this required duration of time. Nodes will each select a random back-off interval to initialize a counter, and will decrement this counter while the medium remains idle. Random back off has been implemented in our simulation.

The time interval between frames is known as the Inter Frame Space (IFS). Once a node has determined that the medium is idle, two IFSs (Distributed IFS and Short IFS) are defined, to provide delays between sending and receiving RTS, CTS, DATA and ACK packets. The way CSMA/CA works is illustrated through the diagram shown below:



When a node needs to transmit a DATA packet, and it detects that the medium is idle, it will first broadcast a small packet called a Request To Send (RTS) packet. The RTS packet contains relevant information for the destination node. The receiving node, if it is within range, will wait SIFS units of time, and then respond to the sender with another small packet called Clear To Send (CTS). The CTS packet indicates that the intended recipient of the data is both within range, and ready to receive data. Upon receipt of the CTS packet, the sender waits another SIFS units of time before sending the DATA packet. Once the receiver has heard the DATA successfully, it will wait SIFS units of time, and then send an Acknowledgement (ACK) packet back to the sender. If the sender does not receive an ACK before a set timeout interval, it will try again by sending out another RTS packet.

If a node is within range of the sender, hears an RTS packet, and is not the intended recipient, it will refrain from sending DATA or RTS packets for a random period of time, known as the back-off interval. This is done to prevent packet collisions within the network.

In our simulation, the RTS/CTS exchange is illustrated by the use of concentric circles. Red concentric circles represent an RTS packet, whereas green concentric circles represent a CTS packet. Following this initial exchange of RTS/CTS packets between the sender and the recipient, the sender transmits a DATA packet, which is represented in our simulation by a red line. Once the DATA arrives, the receiver sends an ACK packet back to the sender. A green line in the simulation represents an ACK packet.

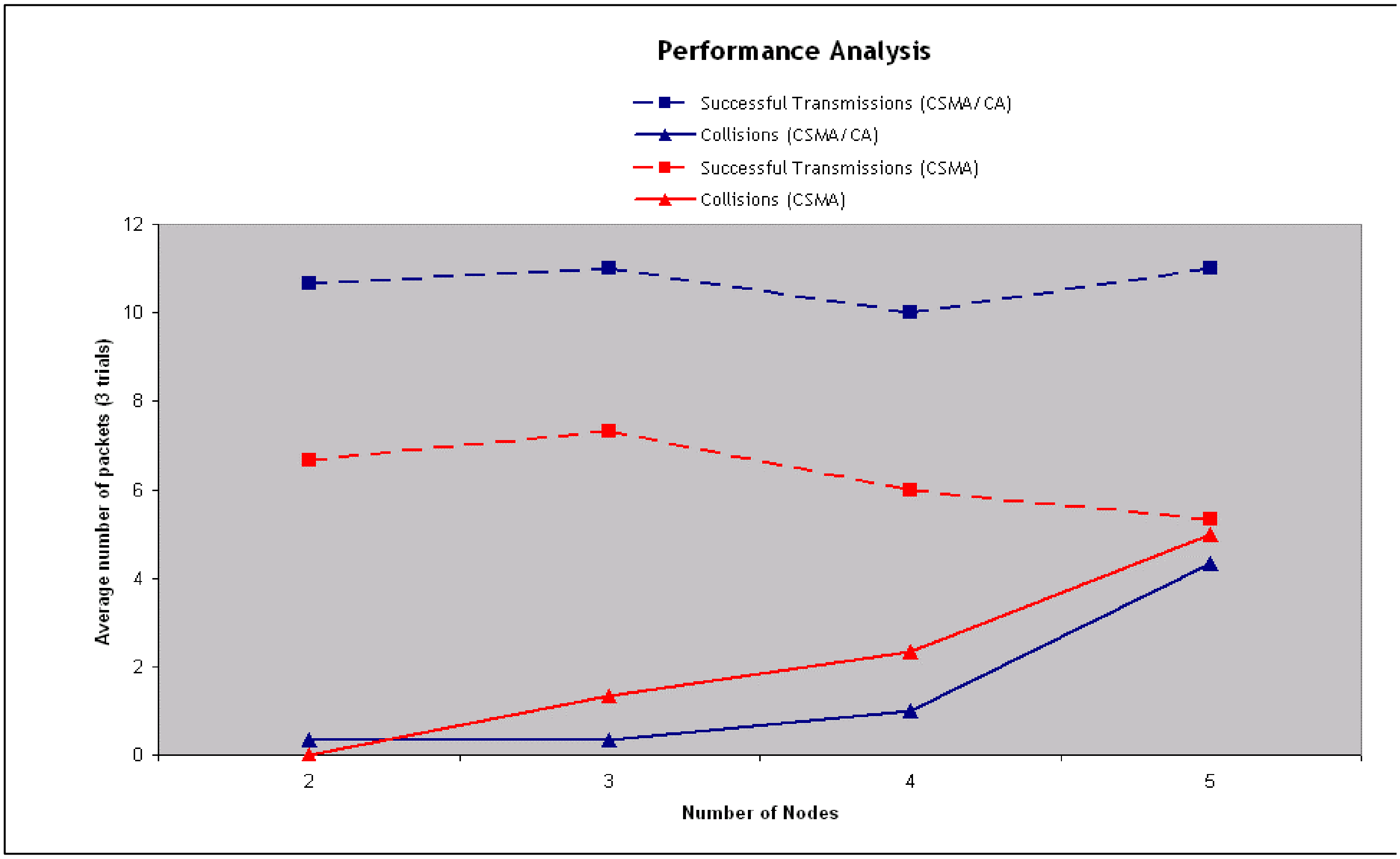
The graphics were set-up using pointers/handles to graphical objects. Initially these objects are all circles.

E.g: h2 (i) = plot (exp (j*ph)*(1+r), 'EraseMode', 'xor');

The MATLAB file "show1.m" changes the properties of these objects to show whether a packet is in range, etc. They get changed to lines to show packet transmission.

V. Analysis

We have found that the inclusion of RTS/CTS handshaking greatly improves the performance of our simulated WLAN. We collected data on the number of successful transmissions and the number of collisions, while varying the number of users on the wireless network using our simulation. Following this we collected the same data using Li and Sherzai's implementation. The performance analysis clearly shows an improvement using CSMA/CA as the protocol.



VI. Summary/Conclusions

Our conclusions based on the improved simulation are as follows:

- Fewer packet collisions and improved collision avoidance.
- More nodes lead to deadlocks where nodes keep exchanging RTS/CTS packets, and no data is exchanged.

All major functionalities are working effectively at this point. There are a few cases of situations where the nodes keep exchanging RTS/CTS packets, and we have yet to explain this situation. One possibility is that of the 'blocked' node, which Professor Starobinski and Ray are currently researching. This concept was introduced to us through Aspian Gazner and Andrew McKnight's presentation.

It is our hope that this simulation model will serve as a tool for analysis of wireless networks in the future, or as a basis for future simulators.

VII References and Acknowledgements

1. ANSI/IEEE Standard 802.11. Piscataway, NJ: LAN MAN society, IEEE Computer Society. Copyright 1999. Available online at http://standards.ieee.org/reading/ieee/std/lanman/802.11-1999.pdf .

2. Carruthers, Jeffrey B., Starobinski, David and Ray, Saikat. The Deaf Node Problem in Ad Hoc Wireless LANs. Not yet published. Contact Professor Starobinski, e-mail staro@bu.edu .

3. Li, Keyong and Sherzai, Mohammed. Wireless MAC 802.11. Unpublished work, final paper for SC 546, Fall 2001. Available online at http://courseinfo.bu.edu/courses/02fallengsc546_a1/coursedocuments/a349b830bc631d6136de00600832de87/paper/finalpaper.doc .

We would like to thank Professor Starobinski and Keyong Li for their insights and constructive criticism, without which this project could not have been completed.