IEEE 802.11 Performance in an Ad-Hoc Environment[1]

Craig Sweet, Vijay Devarapalli, and Deepinder Sidhu

Maryland Center for Telecommunications Research
Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County
1000 Hilltop Circle
Baltimore, MD 21250

{sweet, vdevar1, sidhu}@mctr.umbc.edu

Abstract

IEEE 802.11 is the new standard for communication in a wireless LAN. Its need arose from the many differences between traditional wired and wireless LANs. We briefly introduce the protocol and present the results of our simulation of the Distributed Coordination Function (DCF). Our results show how the various options available with the protocol interact and how the maximum bandwidth can be attained.

Introduction

Since traditional Ethernet has been in existence for quite some time, much research has been done studying its attributes under various conditions. Wireless networking, on the other hand, represents a new shift in networking and is targeted after a significantly different market. It is unlikely that wireless LANs will be used in the same manner as their traditional wired counterparts, at least not in the near term. Current medium access technology effectively limits the amount of traffic, the distance, and/or the number of participating stations. One area where wireless technology stands out is ad-hoc networking. Ad-hoc networks are typically small, on the fly networks that serve a particular purpose and then are dismantled. An example is a conference room or on-site training lab. Wireless networks can provide accessibility and convenience unmatched by other, more traditional, technology.

When assessing the performance of a wireless LAN, it is especially important to consider how this LAN will be used. Previous research has focused mainly on the impact of mobile and/or hidden stations on the performance of the protocol. It can be argued that this is not always the most important attribute. Consider the conference room scenario where a set of wireless terminals is temporarily connected to distribute information to participants. Here we are less concerned with mobility support as with the efficiency of the protocol. Hidden terminals are likely to be classified as an error condition.

The goal of this paper is to determine the efficiency of the IEEE 802.11 wireless LAN standard. We assume near perfect conditions to determine the best performance that could ever be achieved. It is expected that this information could serve as selection criteria when all other metrics are assumed to be equal. We model the MAC-level protocol and describe its performance in the absence of transmission errors, hidden terminals, and mobility. By combining these results with existing experiments, it is hoped that we can determine key characteristics that lend themselves to mobile wireless vs. fixed wireless systems. The results may suggest that instead of making a logical distinction between fixed and wireless systems, one needs to be made between fixed, mobile wireless, and stationary wireless systems.

IEEE 802.11 Wireless LAN Standard

Stations participating in a wireless LAN have fundamental differences from their traditional wired counterparts. Despite these differences, 802.11 is required to appear to higher layers (LLC) as a traditional 802 LAN. All issues concerning these differences must be handled within the MAC layer. This section presents the concepts and terminology used within an 802.11 implementation. For a more detailed description of the 802.11 specification the reader is referred to [IEE97].

One major difference is the wireless station's lack of a fixed location. In a wireless LAN, a station is not assumed to be fixed to a given location. Users are grouped into two classifications, mobile and portable. Portable users are those that move around while disconnected from the network but are only connected while at a fixed location. Mobile users are those users that remain connected to the LAN while they move. IEEE 802.11 is required to handle both types of stations.

Due to differences in the physical medium, wireless LANs also employ a much different physical layer. The physical medium has no fixed observable boundaries outside of which the station cannot communicate. Outside signals are also a constant threat. The end result is that the medium is considerably less reliable.

[1] This research was supported in part by the Department of Defense at the Maryland Center for Telecommunications Research, University of Maryland Baltimore County. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of Defense or the U.S. Government.

Also, the assumption of full connectivity will not always hold true. A station may come into or go out of contact with other stations without leaving the coverage area of the physical layer.

Attributes of Wireless LAN's

Wireless LANs must adhere to the many of the same rules as traditional wired LANs, including full connectivity to stations, the ability to broadcast, high capacity, etc. In addition, wireless LANs have some special requirements unique to their form of communication [STA97]. A few of these follow:

- **Throughput** - Due to the decreased bandwidth of radio and IR channels, the Medium Access Control (MAC) protocol should make as efficient use of this available bandwidth as possible.

- **Backbone Connectivity** - In most cases, wireless LANs connect to some sort of internal (wired) network. Therefore, facilities must be provided to make this connection. This is usually one station that also serves as the Access Point (AP) to the wired LAN for all stations

- **Power Considerations -** Often times, wireless stations are small battery powered units. Algorithms that require the station to constantly check the medium or perform other tasks frequently may be inappropriate.

- **Roaming -** Wireless stations should be able to move freely about their service area.

- **Dynamic** - The addition, deletion, or relocation of wireless stations should not affect other users

- **Licensing** - In order to gain widespread popularity, it is preferred that FCC licenses not be required to operate wireless LAN's.

Physical Medium Specification

As mentioned previously, the wireless physical medium is considerably different than that of traditional wired LANs. Well-defined coverage areas do not exist. The propagation characteristics between stations are dynamic and unpredictable and this drastically influenced the design of the MAC layer. The Physical layer of the IEEE 802.11 specification provides for stations communicating via one of three methods:

- **Infrared (IR)** - Transmits the signal using near-visible light in the 850-nanometer to 950-nanometer range. This is similar to the spectral range of infrared remote controls, but unlike these devices, wireless LAN IR transmitters are not directed.

- **Direct Sequence Spread Spectrum (DSSS) -** Transmits the signal simultaneously over a broad range of frequencies operating at 2.4 GHz. DSSS system uses baseband modulations of differential binary phase shift keying (DBPSK) and differential quadrature phase shift keying (DQPSK) to provide the 1 and 2 Mbps data rates, respectively.

- **Frequency Hopping Spread Spectrum (FHSS) -** Transmits the signal across a group of frequency channels by hopping from frequency to frequency after a given dwell time. This operates at 2.4 GHz. Frequency hopping uses 2-4 level Gaussian FSK as the modulation signaling method. This form of spread spectrum is more immune to jamming.

Distributed Coordination Function

IEEE 802.11 uses a system known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as its Distributed Coordination Function (DCF). All stations participating in the network use the same CSMA/CA system to coordinate access to the shared communication medium.

A station that wishes to transmit must first listen to the medium to detect if another station is using it. If so it must defer until the end of that transmission. If the medium is free then that station may proceed.

Two mechanisms are included to provide two separate carrier sense mechanisms. The traditional physical carrier sense mechanism is provided by the physical layer and is based upon the characteristics of the medium. In addition, the Medium Access Control (MAC) layer also provides a virtual mechanism to work in conjunction with the physical one. This virtual mechanism is referred to as the Network Allocation Vector (NAV). The NAV is a way of telling other stations the expected traffic of the transmitting station. A station's medium is considered busy if either its virtual or physical carrier sense mechanisms indicate busy.

Before a station can transmit a frame, it must wait for the medium to have been free for some minimum amount of time. This amount of time is called the Inter-frame Space (IFS). This presents an opportunity to establish a priority mechanism for access to the shared medium. Depending upon the state of the sending station, one of four Inter-Frame spaces is selected. These are Short IFS (SIFS), PCF IFS (PIFS), DCF IFS (DIFS), and Extended IFS (EIFS) in increasing order of length. The MAC protocol defines instances where each IFS is used to support a given transmission priority.

A station wishing to transmit either a data or management frame shall first wait until its carrier sense mechanism indicates a free medium. Then, a DCF Inter-Frame Space will be observed. After this, the station shall wait an additional random amount of time before transmitting. This time period is known as the backoff interval. The purpose of this additional deferral is to minimize collisions between stations that may be waiting to transmit after the same event.

Before a station can transmit a frame it must perform a backoff procedure. The station first waits for a DIFS time upon noticing that the medium is free. If, after this time gap, the medium is still free the station computes an additional random amount of time to wait, called the Backoff Timer. The station will wait either until this time has elapsed or until the medium becomes busy, whichever comes first. If the medium is still free after the random time period has elapsed, the station begins transmitting

its message. If the medium becomes busy at some point while the station is performing its backoff procedure, it will temporarily suspend the backoff procedure. In this case, the station must wait until the medium is free again, perform a DIFS again, and continue where it left off in the backoff procedure. Note that in this case it is not necessary to re-compute a new Backoff Timer.

Upon the reception of directed (not broadcast or multicast) frames with a valid CRC, the receiving station will respond back to the sending station an indication of successful reception, generally an acknowledgement (ACK). This process is known as positive acknowledgement. A lack of reception of this acknowledgement indicates to the sending station that an error has occurred. Of course, it is possible that the frame may have been successfully delivered and the acknowledgement was unsuccessful. This is indistinguishable from the case where the original frame itself is lost. As a result, it is possible for a destination station to receive more than one copy of a frame. It is therefore the responsibility of the destination to filter out all duplicate frames.

802.11 provides a request-to-send procedure, which is intended to reduce collisions. Stations gain access to the medium in the same way but instead of sending its first frame, the station first transmits a small Request-to-Send (RTS) frame. The destination replies with a Clear-to-Send (CTS) frame. The NAV setting within both the RTS and CTS frames tell other stations how long the transmission is expected to be. By seeing these frames, other stations effectively turn on their virtual carrier sense mechanism for that period of time. While there may be high contention for the medium while the RTS frame is attempted, the remainder of the transmission should be relatively contention-free. This improves the performance of the protocol because all collisions occur on the very small RTS frames and not on the substantially larger data frames. The use of the RTS/CTS mechanism is not mandatory and is activated via a Management Information Base (MIB) variable.

When beginning a transmission that will include more than one fragment, known as a fragment burst, the rules change slightly. Initially it appears identical to a single fragment transmission. The backoff and carrier sense procedures are the same. The difference lies in the IFS used between fragments. Only a SIFS is required between fragments during a fragment burst. The reason for this is to give the sender the highest priority when transmitting a fragment burst. Consider two examples where this may come into play. In the first example, a station with no knowledge of the NAV, perhaps having recently joined the network must try to wait a DIFS before transmitting. After a shorter SIFS the original station takes over the medium with its next fragment and this other station, upon noticing a busy medium, must defer. As a second example a point coordinator, described in the next section, which must wait a PIFS wants to take control of the medium. Since the Point Coordinator observes a shorter IFS than other stations and a longer one than the station transmitting a fragment burst, it (the point coordinator) must defer until after the fragment burst.

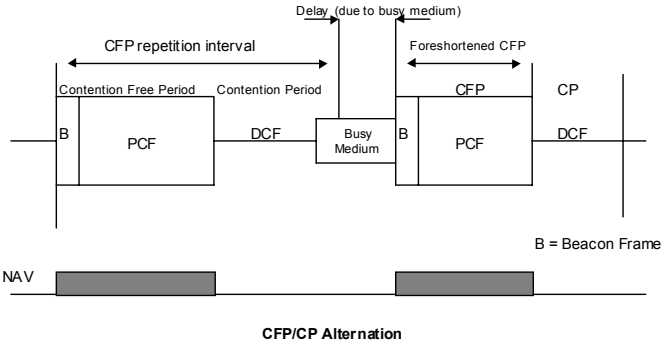
When transmitting broadcast or multicast frames, only the basic transfer mechanism is used. No RTS/CTS mechanism is used regardless of the size of the frame. Additionally, no receiving station will ever respond with an ACK to a broadcast or multicast frame.

Point Coordination Function

The PCF provides contention free frame transfer. The basic principles of the PCF work on top of the mechanisms already provided by the DCF. Thus, all stations inherently coexist with other stations utilizing the PCF function, whether or not they themselves utilize this optional function. The contention free frame transfer is controlled by the Point Coordinator (PC), which normally resides in the AP. The PC maintains a list of all pollable stations and controls access to the medium by polling each station in turn.

CFP Structure and Timing

The Contention Free Period alternates with the Contention Period. Each CFP begins with a Beacon frame, which contains a Delivery Traffic Indication Message (DTIM).



CFP/CP Alternation

The PC generates CFPs at the contention free repetition rate (CFPRate defined as a number of DTIM intervals). The CFPRate is determined by the PC from the CFPRate parameter in the CF Parameter set. The CF Parameter set is present only in the Beacon frames. The length of the CFP is controlled by the PC, with the maximum duration specified by the value of the CFPMaxDuration Parameter in the CF Parameter Set at the start of the CP.

Depending on the available traffic and the size of the polling list, the PC can terminate a contention free period earlier than that specified by the CFPDurationRemaining value in the beacon frames. A CFP can be shortened, if the beacon that signals the starts of the CFP is delayed due to a busy medium. Such CFPs are called foreshortened CFPs.

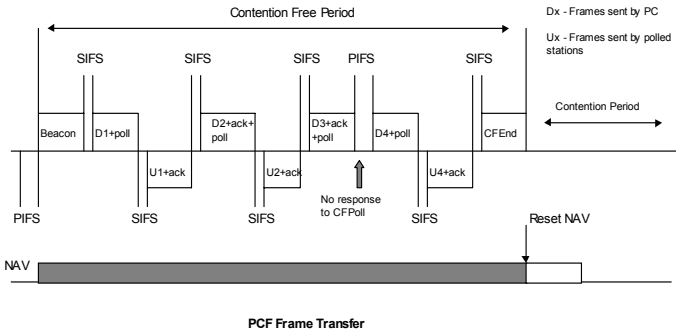
PCF access procedure

The PC maintains a list of all pollable stations. A STA indicates its CF-Pollability using the CF-Pollable subfield of the Capability Information field of the Association Request and Reassociation Request frames. The PC polls at least one STA from the polling list, during the CFP.

The PC senses the medium at the nominal beginning of the CFP. If the medium is idle for a PIFS period, it transmits a beacon frame containing a CF Parameter Set element.

Thereafter, the PC retains control of the medium by waiting only for PIFS period, while all other stations wait for a DIFS period. If there is no station to be polled or no traffic buffered, the PC sends a CF-End frame to terminate the CFP. STAs receiving error free frames from the PC, reply within a SIFS period. If there is no reply in response to a CFPoll, the PC polls the next stations in its polling list after waiting for a PIFS period.

Every STA, except the PC sets its NAV to CFPMaxDuration value at the start of the CFP. During the CFP, they update their NAVs using the CFPDurRemaining value in the beacon frames. This prevents STAs from gaining access to the medium during the CFP, unless when polled by the PC. The NAV operation also facilitates the operation of overlapping CFP coordinated infrastructure BSSs. When a STA receives a CFEnd frame, it resets its NAV.



PCF Frame Transfer

Frame transfers under the PCF typically consist of frames alternately sent from the PC and sent to the PC. In response to a directed CFPoll, a STA sends a data frame within a SIFS period. If it has nothing to send, it responds with a null frame. If the polled STA does not respond within a SIFS period, the PC polls the next station in its polling list after waiting for a CFPoll timeout period.

Acks and CFPolls are piggybacked with data frames, if possible. This reduces the number of frames transmitted and improves throughput.

Modeling and Simulation

In our experiments, our goal was to explore the efficiency of the MAC protocol under ideal conditions. While many of these conditions may be unrealistic, the end result is useful in telling us the highest performance that can be expected from the protocol. This section describes some of the assumptions and limitations assumed in our system. Also, the simulation model and computation variables are described.

Assumptions

All stations are assumed to be using a Direct Sequence Spread Spectrum (DSSS) radio. The operation of Frequency Hopping Spread Spectrum (FHSS) and Infrared (IR) radios had too much of an impact on a given transmission to study the aspects of the protocol itself. Additionally, it is assumed that there are no power considerations for either the radios or the wireless stations that could interfere with the operation of the protocol.

All stations are assumed to be transmitting using the same data rate. In this analysis, speeds of 1, 2, and 10 Megabits per Second (Mbps) are used. While only 1 and 2 Mbps are listed in the 802.11 specification, 10 Mbps was included since current research is aimed at providing radios that work at this and higher speeds.

A significant aspect of any transmission protocol is how it handles transmission errors. In order to focus on the core MAC protocol, we assumed error-free channels. Additionally, all stations have unobstructed access to all other stations and thus can hear all transmissions.

To minimize complexity, we chose to model our wireless LAN as an ad-hoc network, also known as an Independent Basic Service Set (IBSS). This is the simplest type of wireless LAN defined in the standard. There is no Access Point and therefore no connection to a wired LAN.

Description of Simulation Model

To perform this analysis, we constructed a discrete-event simulation of the MAC portion of the IEEE 802.11 protocol. A complete description of simulation techniques can be found in [BAN84]. This simulation is software that has been tested to conform to all aspects of the DCF portion of the protocol. To eliminate any initialization biases, we allowed the simulation to run for approximately 10 seconds before collecting data. After initialization, we allowed the system to run for another 60 seconds. Our tests showed no significant differences in runs longer than 60 seconds.

For all experiments, each station is assumed to have one MAC Service Data Unit (MSDU) buffer. An MSDU is the basic unit delivered between two compatible MAC sub-layers. For uniformity all MSDUs transmitted are of equal size. Initially, each station is given one MSDU to transmit. Upon completing the transmission attempt, another MSDU is assigned for transmission after some exponentially distributed inter-arrival time. In this manner, changing the mean inter-arrival time between MSDUs can be used to alter system load.

Offered Load Computation

Upon transmitting a message, the station generates the next message with an inter-arrival time exponentially distributed with mean θ. Additionally, each station is sending the same size packets, in bytes P, for the duration of a run. Let S denote the set of all stations in the system and N=|S|. The offered load of station $i \in S$, G_i, is defined as in [GON87] to be the throughput of station i if the network had infinite capacity, i.e.,

Upon transmitting a message, the station generates the next message with an inter-arrival time exponentially distributed with mean θ. Additionally, each station is sending the same size packets, in bytes P, for the duration of a run. The offered load of station i, G_i, is defined as in [GON87] to be the throughput of station i if the network had infinite capacity, i.e.,

$$G_i = T_p / \vartheta_i$$

where T_p = P/C and C is the transmission speed in Mbps. The total offered load can thus be computed to be

$$G = \sum_{i=1}^{N} Gi$$

In our system, all stations are assumed to have the same load. This is not unreasonable when considering an environment where each station performs roughly the same duties, as is commonly found in ad-hoc networks.

Let $P_k(t)$ denote the probability that k events will occur during any particular interval of time τ in a Poisson distributon. In [DEV95] it is shown that

$$Pk(t) = e^{-\alpha\tau}(\alpha t)^k / k!$$

where α is the rate by which new transmissions are scheduled. Since we are modeling a perfect system, we assume that all stations are visible to all others. Therefore, a transmission at time *t* is considered successful if no other transmission begins in the interval $(t-\beta, t+\beta)$, where β is the normalized propagation delay. Since all stations are assumed to obey the DCF transmission rules, collisions can only occur during this period. The probability that the current transmission attempt is successful (no collisions) can therefore be expressed as:

$$P_{success} = P_0(2\beta) = e^{-[2\beta G]}$$

This equation gives us a way to express the probability of a collision based upon the system load G and the distance between the stations.

We now wish to obtain an expression for the expected throughput. We define system throughput as the total number of data bytes transmitted divided by the length of a transmission. Let δ denote the number of fragments transmitted, then:

$$\delta = \left\lceil \frac{DATA + OVERHEAD}{Frag.Threshold} \right\rceil$$

Let T_s be the expected length of a successful frame. Close observation reveals that:

$$T_s = 1/G + DIFS + l_{data+overhead} + 2\delta\beta + \delta l_{ack} + (2\delta\text{-}1)SIFS$$

Where 1/G is the expected time until the transmission is scheduled. System throughput can now be defined as:

$$S = \frac{e^{-[2\beta G]}\, l_{data}}{T_s}$$

Performance Analysis

In this analysis, we performed six experiments measuring various aspects of the MAC protocol. Each of these experiments was conducted at several transmission speeds. 1 and 2 Mbps were selected because they are explicitly supported in the specification. 10 Mbps was selected to provide a comparison at traditional LAN speeds. The results of these experiments are the topic of this section. Current research is aimed at providing 802.11 operation at 10 and 20 Mbps.

Experiment 1: Variable Load

In our first experiment we wanted to see what effect the total load on the system played on performance. This experiment is similar to one found in [GON87]. Figure 1 shows the variation of total throughput with total offered load G for various message sizes P at 1 Mbps. In this experiment, the fragmentation threshold has been set to 2346 bytes and the RTS threshold has been set to 3000 bytes. The experiment was run with 32 stations and an active PC.

We can see that with an offered load of about 70% or less virtually no collisions occur and throughput and load are approximately equal. The throughput of the system increases with increasing load. Once the system load increases beyond 90-100 % we see the impact of collisions. There is no increase in throughput due to the large number of collisions.

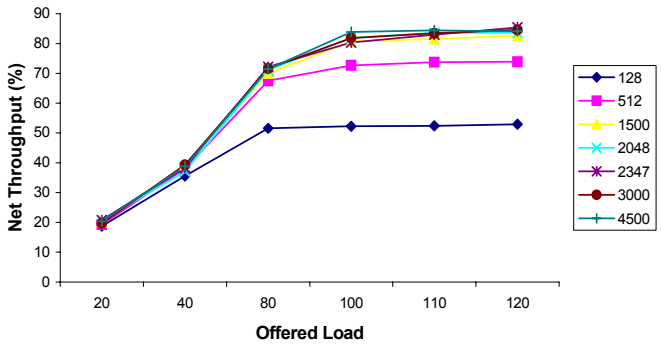


Fig 1. Throughput Vs Offered Load at 1Mbps with variable message size

As can be expected, greater throughput is achieved via a greater packet size. Due to the overhead present in the protocol, especially the polling overhead during the CFP, acceptable throughput was not seen with packet sizes below 2000 bytes. Packet sizes above and below the fragmentation threshold did not yield much difference. Even then, it all but disappeared with loads in excess of 100%. While increasing the number of packets per message produces more overhead, it also reduces the collision probability.

In this example, the RTS threshold played a crucial role in the performance of the protocol. The throughput peaked out at approx. 80% for all packet sizes below 3000 bytes. As explained earlier, the RTS threshold acts as a medium reservation mechanism. Collisions, and subsequent retransmissions, can occur on the smaller RTS frames but not normally on the longer data frames. The result is a better utilization of the bandwidth.

Our results were similar for transmission speeds of 2 and 10 Mbps. Table I summarizes some of these results. What we saw was that as the transmission speed increased, the net throughput dropped. This can be attributed to the fact that the inter-frame spaces are independent of transmission speed. At higher speeds, since it takes less time to send the same packet, an IFS of 50 μs has more of an impact than at lower speeds.

Table I
Simulation Results at 200% Offered Load for Various Packet Sizes and Transmission Speeds

Mbps	Packet Size	Throughput %
1	4500	96.61
	2800	76.54
	2347	71.52
2	4500	96.11
	2800	76.07
	2347	73.08
10	4500	91.80
	2800	73.17
	2347	68.87

Figure 2 compares the performance with and without an active PC while increasing the load. This experiment was performed with 32 stations and with packet size of 512 bytes.

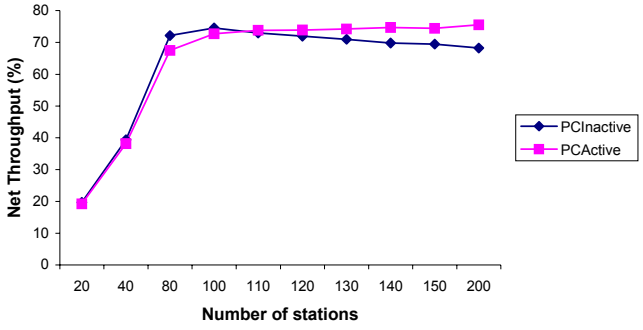


Fig 2 Performance comparison with and without PCF

At or below 100% load, the overhead associated with an active PC degrades performance. This is due to the high overhead that the PCF imposes on the MAC protocol. At higher loads the performance of the DCF protocol deteriorates because of increased contention, whereas with an active PC, the performance improves. This is because the PCF reduces contention between stations. For loads higher than 120%, the net throughput with the PC active remains constant and doesn't decrease in spite of increasing load.

Experiment 2: Variable Stations

The number of stations in an IBSS directly influences the system operation and throughout. During DCF, stations contend for access to the medium and contention increases with an increase in the number of stations. Under PCF, more stations means a longer polling list, which means longer delays for stations wishing to transmit. On the other hand, when the number of stations is very few, the system is underutilized resulting in very poor throughput. The peak performance is achieved with an optimum number of stations, which can be determined experimentally. This optimum number depends on the system load, data rate, and the PCF parameters.

This experiment has two parts. In the first part of the experiment, we observed the protocol performance while varying the number of stations. The PC was turned off to analyze the DCF part of the protocol. Figure 3 shows the effect on throughput with an increasing number of stations at a constant Offered Load of 100%.

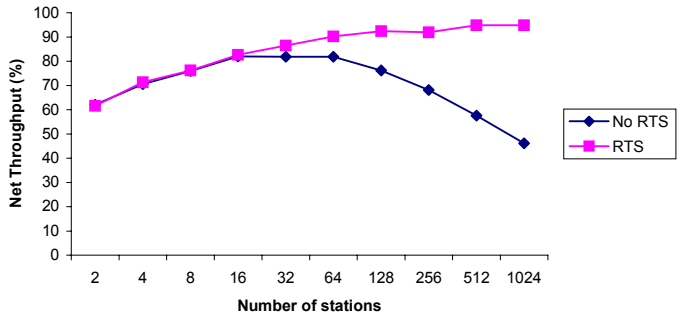


Fig 3. Throughput Vs Number of stations with packet size above and below RTS threshold (1Mbps)

Results are shown both with and without the RTS mechanism implemented. For all runs, the message size was set to 3000 bytes and the fragmentation threshold was set to 2346 bytes.

Without RTS enabled, we can see that the maximum throughput reached was approx. 82% with 16 participating stations. In fact, with few stations (below 16), we see that there is not much difference in performance with and without RTS enabled.

As more stations are added to the simulation the probability that two or more stations will calculate the same backoff window is increased. Thus, the chance for collision increases. This can be seen by the large differences between the RTS and No-RTS runs with higher station counts, above 64.

Since IEEE 802.11 uses CSMA/CA, collisions are expensive. The transmitting station must continue to transmit the entire message and wait a minimum amount of time before determining that the transmission was in error. With RTS enabled, the collisions occur on smaller RTS frames, allowing for a quicker turn-around time. We can see that with RTS enabled, the system stabilized to approx. 92% or higher with 128 or more stations.

As in the previous experiment, we saw similar results in our 2 and 10 Mbps experiments. As the data rate of the medium increased there was still the same pattern between RTS and No RTS results. The only difference was that higher transmission speeds yielded lower average net throughput results. Table II summarizes some of the results from these experiments with and without RTS enabled.

Table II
Simulation Results at 100% Offered Load with Variable Number of Stations

Mbps	# of Stations	Throughput with RTS	Throughput without RTS

1	16	82.59	81.93
	128	92.38	76.22
	1024	94.81	46.13
2	16	82.83	82.98
	128	93.04	73.28
	1024	94.07	53.91
10	16	80.34	78.01
	128	88.6	70.04
	1024	89.95	57.84

In the second part of this experiment, we wanted to compare the protocol performance with and without the PC. The result is shown in Fig. 4. When the PC is inactive, the throughput is due to the DCF part of the protocol alone. When the PC is active the throughput is due to both the DCF and the PCF part of the protocol.

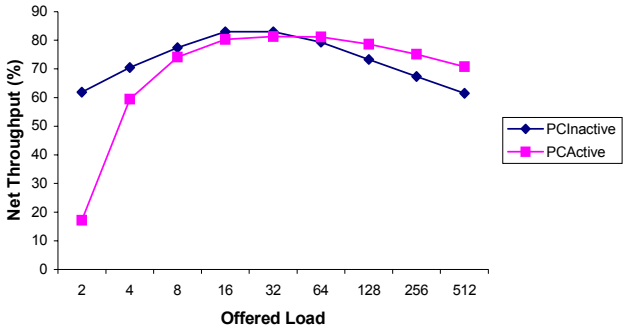


Fig 4. Throughput Vs Number of stations

With fewer than 8 stations, the performance with the PCF turned on is very poor compared to the DCF performance. This is because of the high overhead imposed by the PCF part of the protocol. The performance is comparable when the number of stations is between 16 and 64. The DCF performance decreases with a large number of stations due to more contention. The polling scheme of the PCF allows it to do better by eliminating contention.

Experiment 3: Variable Fragmentation

In our third experiment, our goal was to determine what effect the fragment size played on system performance. The simulation was run with 32 stations at 200% load with varying fragmentation thresholds. Each message sent was 3000 bytes long. Therefore, the fragmentation threshold merely determined how many fragments the 3000 byte messages were broken up into.

Intuitively, advantages can be gained by both increasing and decreasing the fragmentation threshold. Smaller thresholds limit the loss of performance due to retransmissions but come with an increase in overhead. This is important because we have already shown that the 802.11 protocol has considerable overhead. On the other hand large fragmentation thresholds, while limiting the overhead, become expensive in the event of a collision.

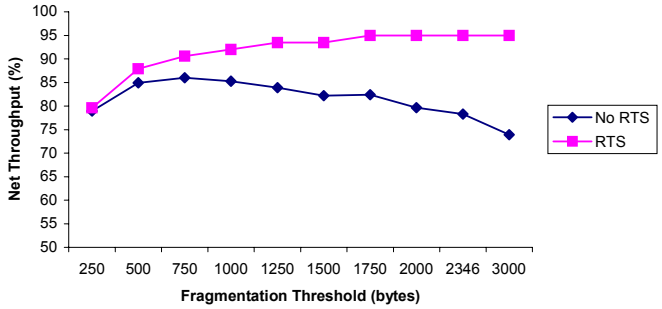


Fig 5. Throughput Vs Fragmentation Threshold with packet size above and below RTS Threshold (1Mbps)

Figure 5 shows the results of this experiment run at 1 Mbps. As predicted, the RTS mechanism does a great deal to improve the performance of this aspect of the protocol. The reason can be attributed to the reduction in collisions that it provides. At smaller thresholds, there is little difference between the RTS and No RTS figures. There is nearly a balance between three factors: the overhead provided by the RTS mechanism, the smaller fragment sizes that are retransmitted in the event of a collision, and the overhead provided by multiple smaller fragments.

It is not until the fragmentation threshold increases that we see the largest variation in performance. As was expected, with larger fragments comes a decrease in performance. Each collision requires retransmission of a much larger fragment. Since 802.11 does not have a collision detection mechanism the entire fragment must be transmitted before success or failure of that fragment can be determined.

This experiment has also shown that, in this specific case, little improvement can be seen with fragments above 1000 bytes when the RTS mechanism is used. While this may be true in this experiment, note that we are assuming that all fragments are transmitted error-free. This assumption will certainly not hold in a real-world case. In fact, performance may decrease as the bit-error rate increases. The probability of each fragment being successfully delivered will decrease as the fragment size increases and results will most certainly differ.

The results for 1, 2, and 10 Mbps experiments are summarized in Table III. We can see that the same pattern is exhibited regardless of the transmission speed. As we have seen in the previous experiments, the constant inter-frame space times effectively reduce the system performance at higher speeds.

Table III
Simulation Results at 200% Offered Load with Variable Fragmentation Threshold and Transmission Speed

Mbps	Frag. Threshold	Throughput with RTS	Throughput without RTS
1	250	79.61	78.93
	1250	93.46	83.90
	3000	94.96	73.89
2	250	78.44	77.72
	1250	92.68	83.49
	3000	94.27	73.38

10	250	70.47	70.09
	1250	88.79	80.56
	3000	89.45	70.50

Experiment 4: Variable Propagation Delay

In our previous experiments, we assumed a constant delay of 1 µs between stations. This allowed us to measure the protocol performance without respect to the interoperability in a real-life situation. In our fourth experiment, our goal was to determine how far apart stations can be from one another, in terms of propagation delay, before system throughput degrades. In a real-world wireless network, some stations may be constantly moving while others are stationary for periods of time.

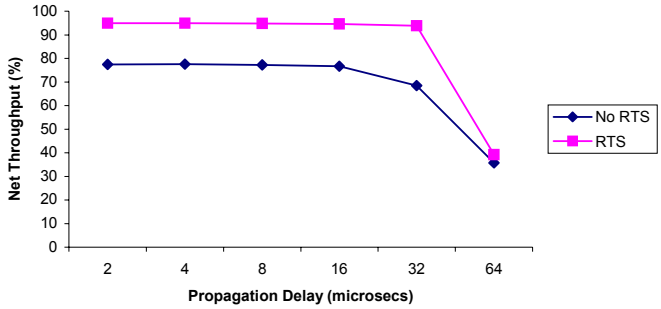


Fig 6. Throughput Vs Propagation Delay with packet size above and below RTS threshold (1Mbps)

In this experiment, we set the fragmentation threshold to 2346 bytes and the message size to 3000 bytes. The system is run at 100% Offered Load. Figure 4 shows the results of increasing the propagation delay between any two wireless stations operating at 1 Mbps.

We can see that, with the current fragmentation threshold and a 50 µs DIFS, throughput drops when the propagation delay between stations exceeds 50 µs.

Recall that when a station transmits a message, it waits only a finite amount of time for the response. If this response does not arrive in time, it will retransmit the message. This timer begins immediately after the sender finishes transmitting the message. If the receiver is sufficiently far away from the sender, much of this time is taken up by twice the delay between the stations, once for the message to reach the recipient and once for the response to arrive at the source.

If the distance between two stations becomes too large, it will be impossible for the sender to hear the acknowledgement from the receiver. In this case, it becomes increasingly difficult for messages to be received correctly. The result is increased retransmissions and decreased throughput.

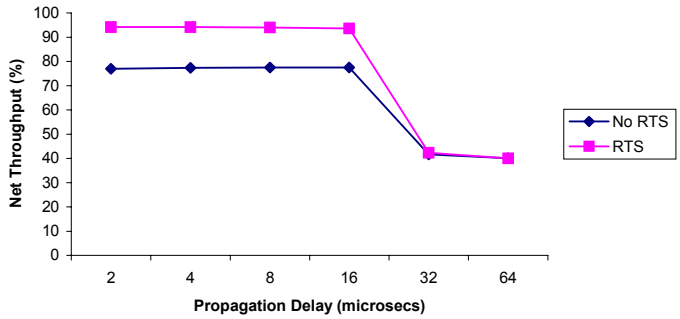


Fig 7. Throughput Vs Propagation Delay with packet size above and below RTS Threshold (2Mbps)

Unfortunately the problem only compounds itself as the transmission speed increases. Figure 5 shows the same experiment run at 2 Mbps. Here we see that the same drop off in throughput occurs with stations only 32 µs apart. The reason is that the initial transmission is shorter at the higher speed, which forces the station to begin its waiting period earlier. Therefore, this timer can expire with a shorter propagation delay.

As can be expected, the results are even worse for transmissions at 10 Mbps. These results are shown in figure 6. An interesting point in all three graphs is that the RTS mechanism can do little to improve this performance. This assures us that the loss in throughput is not attributed to collisions but rather to too much distance between stations. In fact, the added overhead of the RTS mechanism slightly reduces the performance once this problem occurs.

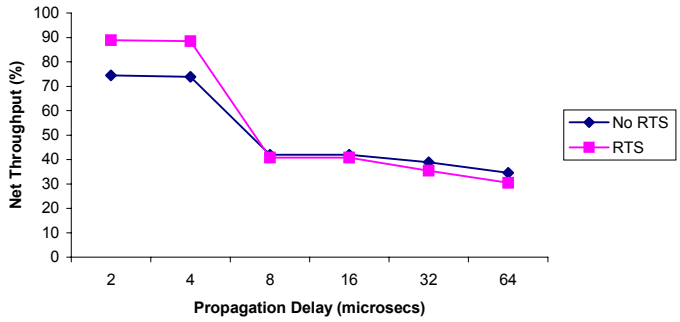


Fig 8. Throughput Vs Propagation Delay with packet size above and below RTS threshold (10Mbps)

It a reasonable assumption that there is a limit to the distance that any two communicating stations can be from one another before system performance suffers. This limit is based upon the attributes of the communication medium and the protocol. From this experiment we can see that the transmission speed also plays a crucial role.

Experiment 5: Length of the Contention Free Period

The length of the Contention Free Period determines how long the PC controls access to the medium during a CFP repetition interval. A smaller CFP means the access to the medium is mostly controlled by DCF. In this case, the system load and number of stations will directly influence the throughput. A

larger CFP means the access to the medium is mostly controlled by the PCF. Virtually all contention is eliminated in this case. The system performance due to PCF varies from that due to DCF under different system parameters. Length of the CFP is one such parameter.

Our goal in this experiment was too see how the throughput is effected when the length of the Contention Free Period is changed. The CFP repetition interval was fixed, and the length of the contention free period was increased from a minimum of zero to the maximum as defined in [IEEE97].

This experiment was performed at varying loads. Figure 9 shows the graph for the experiment.

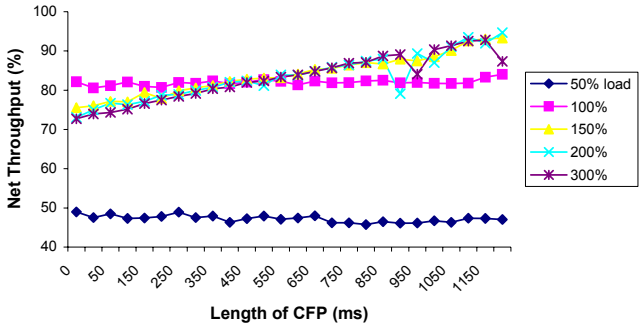


Fig 9. Throughput Vs Length of the Contention Free Period

At loads of 50 % and 100 %, the length of the CFP has no influence on throughput. DCF does as well as PCF because there is not much contention for the medium. A 150 percent load is just enough to cause DCF performance to decrease due to increased contention. At Loads of 150 % and more, PCF helps in improving the throughput. There is an increase in throughput as the length of the CFP is increased.

Experiment 6: Length of the CFP Repetition Interval

The CFP Repetition Interval determines how frequent the CFP alternates with the CP. The overhead associated with this alternation is quite significant and affects the throughput. When the system is run without the PC, this overhead is absent.

The aim of this experiment was to study the protocol performance when the length of the CFP repetition interval is changed. The CFP repetition interval is 6 times the Beacon Interval with the length of the CFP and CP being equal. This experiment was performed with 32 stations and a packet size of 2048 bytes at 100 % load.

Figure 10 shows the graph for this experiment. The straight line corresponds to the protocol performance with only the DCF. The other line corresponds to the PC being active.

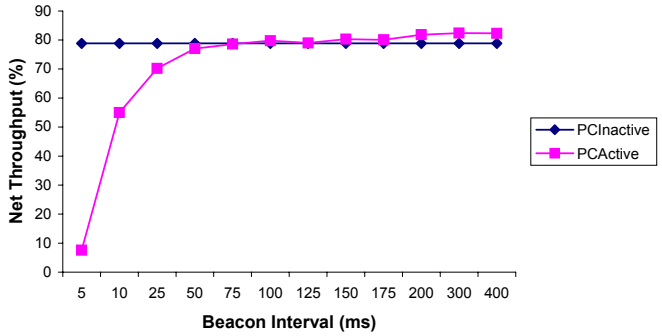


Fig. 10. Throughput Vs Length of Beacon Interval at 100% load

With a small beacon interval of 5 ms, the system performance is very poor. A beacon interval of 5 ms corresponds to a CFP and a CP alternating every 15 ms. It is just enough for one station to transmit during each period. The polling and beacon overheads are significant enough to reduce the throughput drastically. The throughput is also affected by switching between DCF and PCF. The throughput increases quite rapidly with the increase in the length of the beacon interval. At beacon intervals of 200 ms and more, PCF performs better than just DCF alone. When the beacon interval is 200 ms, all the 32 stations are able to transmit a packet each during the CFP and the CP.

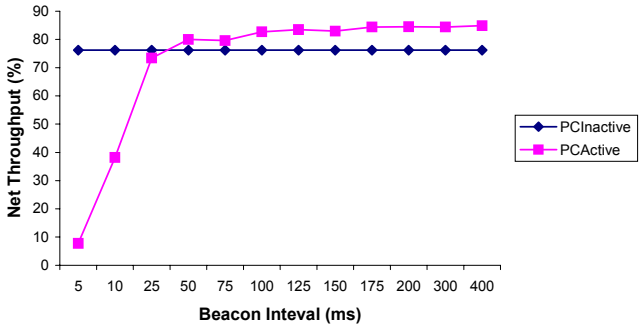


Fig 11. Throughput Vs length of the Beacon Interval at 150 % load

Figure 11 shows the graph for the same experiment at 150% load. The difference is more pronounced at this load. PCF starts performing better at beacon intervals of 50 ms itself. This is because of poorer performance by DCF at increased contention.

Conclusion

While the experiments described in this paper do not reflect any real-life scenario, they are useful in determining the maximum system performance under a variety of conditions. Our goal has been to see what the maximum performance we can expect out of the protocol is and what it takes to reach it.

We see from our experiments that Ethernet speeds are possible but only with the RTS mechanism that is built into the 802.11 MAC protocol. This mechanism, while adding some overhead, offers considerable improvement in most highly loaded systems.

We found that the best performance can only be achieved in systems with relatively slow transmission speeds. Transmission speed and throughput were inversely proportional. This is due

to the constant delays and timers used in the protocol, which are not altered as transmission speed increases.

Future Work

Currently our research does not take into account the transmission errors that are inherent in all forms of communication. One area of research will be to incorporate a bit-error rate into the simulation, based upon the transmission device, and see how the system performance is affected.

Our system did not allow for a subset of stations to be hidden from the others. We assumed that all stations could hear all transmissions from all others. With this medium, stations can be obstructed from some other stations in the network. This would prevent them from reading all of the NAV values that are transmitted. Future research could take this into account.

References

[BAN84] Banks, J. and J. S. Carson, "Discrete-Event System Simulation," Prentice-Hall, Englewood, NJ, 1984.

[BUX81] W. Bux, "Local-area subnetworks: A performance comparison," *IEEE Trans. Commun*., vol COM-29, pp. 1465-1473, 1981.

[DEV95] J. Devore, "Probability and Statistics for Engineering and the Sciences, 4th Edition", Brooks/Cole Publishing, 1995.

[GON83] T. A. Gonsalves, "Performance characteristics of 2 Ethernets: An experimental study," ACM SIGCOMM *Symp. On Commun. Architectures and Protocols*, Austin, TX, Mar. 1983, pp. 178-185.

[GON87] T. A. Gonsalves, "Measured Performance of the Ethernet," in *Advances in Local Area Networks*, Kummerle, K., Tobagi, F., and Limb, J.O. (Eds.), New York: IEEE Press, 1987.

[IEE97] IEEE Std 802.11-1997, "IEEE Standard for Local and Metropolitan Area Networks: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification."

[STA97] Stallings, W, "Data and Computer Communications, 5th Edition," Prentice-Hall, Upper Saddle River, NJ, 1997.

TOB80] F. A. Tobagi and V. B. Hunt, "Performance analysis of carrier sense multiple access with collision detection," *Comput. Networks*, vol. 4, Oct./Nov. 1980.