The title of the publication is:

IEE PROCEEDINGS COMMUNICATIONS

The number of words: 7990

Illustrations: eleven figures and one table

Corresponding author:

Zhifei Li

Center for Multimedia and Network Technology (CEMNET),

School of Computer Engineering,

Nanyang Technological University,

Singapore, 639798

A Full Auto Rate (FAR) MAC Protocol for Wireless Ad Hoc Networks

Zhifei Li, Anil K. Gupta, and Sukumar Nandi*

School of Computer Engineering, Nanyang Technological University, Singapore-639798

*Dept. of Computer Science & Engineering, Indian Institute of Technology, Guwahati, India, 781039

E-mail: pg03802331@ntu.edu.sg; asgupta@ntu.edu.sg; sukumar@iitg.ernet.in

Abstract: Emerging IEEE 802.11 standards provide very high raw bandwidth. However, the overheads introduced by the physical and MAC layers are also increasingly substantial. Moreover, the multi-rate capability provided by the various physical layers requires that the MAC layer adapt the transmission rate according to the channel conditions. Recently, several rate-adaptation algorithms have been proposed. However, all the schemes have considered the rate adaptation for the Data frame only, while assuming that the control frames are always transmitted at a low basic rate. Also, the rate adaptation function is performed either at the sender side or at the receiver side. In this paper, we have proposed a new rate-adaptation algorithm, called Full Auto Rate (FAR), which is able to adapt the transmission rate of all the frames (both control and Data frames). Our FAR combines the sender- and receiver-based methods proposed in the literature, that is, the rate adaptation of the RTS/CTS frames is done at the sending side of these frames while that for the Data/ACK frames is done at the receiving side of the frames. Moreover, in order to cope with the issues involved in the virtual carrier sensing (VCS), we have proposed a modified virtual carrier sensing (MVCS) mechanism. Both analytical and simulation results show that the FAR greatly improves the performance of IEEE 802.11.

1 Introduction

Recently, wireless ad hoc networks have been of considerable research interest as they are easy to deploy and maintain. The Distributed Coordination Function (DCF) in IEEE 802.11 [3] is popularly adopted as the MAC protocol for the ad hoc networks. DCF defines two-way and four-way handshakes for accessing the medium. In the two-way handshake, a sequence of Data and ACK frames is used for the transmission of any *single* data packet. On the other hand, in the four-way handshake, two additional frames, Request To Send (RTS) and Clear To Send (CTS), are used before the transmission of the Data frame.

At the physical layer, the IEEE 802.11a [4], 802.11b [5] and 802.11g [6] standards provide the multi-rate capability. For example, data rates defined in IEEE 802.11b are 1, 2, 5.5, and 11 Mbps, while the 802.11a and 802.11g provide a data rate up to 54 Mbps. To make full use of the multi-rate capability, the MAC protocol should choose a rate in an adaptive manner, known as *link adaptation*, *rate adaptation*, or *auto rate*. The basic idea of rate adaptation is to dynamically estimate the channel conditions and then select a transmission rate that will give the optimum throughput under the given channel conditions. In IEEE 802.11 standards, the rate adaptation algorithm has been intentionally left open. Recently, numerous algorithms [7] [8] [9] [10] [11] [12] have been proposed to support multi-rate operation in IEEE 802.11. These algorithms consider the rate adaptation of the Data frame only, while the control frames (RTS/CTS/ACK) are always transmitted at a very low basic rate (e.g., 1 or 2 Mbps in IEEE 802.11b) to ensure that all the nodes in the interference range can overhear the control frames clearly. Based on whether the rate adaptation function for the Data frame is performed at the sender or at the receiver side, these algorithms can be classified into two categories: sender-based and receiver-based. The algorithms in [9] [10] [11] [12] fall into the first category while those in [7] [8] belong to the second category. The sender-based algorithms are simple and easy to incorporate into the standards. However, they cannot reflect the medium state in a timely and precise manner. On the other hand, since it is the receiver that can perceive the channel quality, the receiver-based method adapts better to the channel conditions. However, the receiver-based algorithms rely on the existence of the RTS/CTS handshaking. Specifically, based on the Signal to Noise Ratio (SNR) of the received RTS, the receiver estimates the channel conditions, and then in the CTS it suggests a date rate that the sender should use for the transmission of the Data frame. In ad hoc networks, since the RTS/CTS handshaking is always used to cope with the common hidden-terminal problem, the receiver-based algorithm is more desirable. However, as mentioned, since the control frames (RTS/CTS) are always transmitted at a very low data rate, the bandwidth is greatly wasted by the control frames. Moreover, with the increase of the maximum data rate in the emerging standards (e.g., 54 Mbps in 802.11a and 802.11g), the proportion of the wastage increases. Therefore, the control frames should also be transmitted at the highest attainable rate under the given channel conditions.

In this paper, we have proposed a new rate adaptation algorithm, called Full Auto Rate (FAR), which can adapt the transmission rate of all the frames in the wireless ad-hoc networks. Our FAR combines the sender- and receiver-based methods proposed in the literature. Since the channel conditions may differ in the opposite directions in-between two communicating nodes, and since within a specific frame exchange sequence the RTS and CTS are the first frames transmitted in the corresponding directions, the rate selection has to be done at the sender side of the two frames.

Therefore, any sender-based algorithm (e.g., those in [9] [10] [11] [12]) can be used to adapt the transmission rate of these two frames. On the other hand, since the RTS/CTS frames can probe the channel conditions in the corresponding directions, a receiver-based algorithm (e.g., [7] and [8]) can be used to adapt the transmission rate of the Data/ACK frames. Note that here the 'sender' and the 'receiver' are corresponding to the frame being transmitted, rather than particularly to the Data fame.

Since all the nodes share the medium, when a frame exchange sequence is in progress, it is extremely important to prevent other nodes in the interference range from transmitting. This is achieved through the physical carrier sensing and virtual carrier sensing (VCS) mechanisms in IEEE 802.11. Using VCS, the over-hearing nodes defer their transmission for the impending use of the medium by the ongoing frame exchange sequence. Specifically, a *Duration* field in a frame indicates the time duration that the medium is to be reserved for the transmission of the remaining frames in the sequence. For example, the duration in RTS is equal to the transmission time of CTS, Data, and ACK. If the IEEE 802.11 VCS is used in our FAR, two problems arise. The *first* problem is that the sender may not know the exact transmission rates of all the remaining frames when calculating the duration. For example, when calculating the duration field in RTS, the sender node does not know the actual transmission rate of the Data frame since that rate can be known only at the receiver after it has received the RTS. Therefore, the sender has to assume a rate for the Data frame in the calculation. When the actual rate of Data frame is different from the one assumed by the sender, the reservation made by the RTS becomes inaccurate. The *second* problem is that the VCS relies on the fact that all the nodes in the interference range can overhear the duration filed clearly, which may not be true when the frames are transmitted at a high data rate. In order to cope with the above two problems, we have proposed a modified VCS (MVCS), which reserves the medium only for the *immediate next* frame rather than for *all* the remaining frames in the sequence. The MVCS, though very simple, solves the above two problems with the help of the physical carrier sensing and the Extended Inter Frame Space (EIFS) based deferment defined in IEEE 802.11.

The Full Auto Rate (FAR) with the MVCS greatly improves the performance of IEEE 802.11. We show this using both analytical and simulation methods.

The remainder of the paper is organized as follows. In Section 2, we describe the basic technique of IEEE 802.11and the related work. Then, in Section 3, we present our motivation to transmit the control frames at a higher rate. In Section 4, the FAR and the MVCS algorithms are presented. Sections 5 and 6 present the analytical and simulation results, respectively. The paper is concluded in Section 7.

2 Preliminaries

2.1 Overview of IEEE 802.11 DCF

The IEEE 802.11 DCF adopts the well-known Binary Exponential Back-off (BEB) algorithm as its Contention Resolution (CR) mechanism, which doubles the contention window (CW) whenever a collision occurs and resets the CW whenever a transmission is successful.

The DCF also defines two methods in accessing the medium, namely, the two-way handshake and the four-way handshake. In the two-way handshake, the sender first transmits a Data frame to the receiver, which responds with an ACK frame if it receives the Data frame correctly. On the other hand, in the four-way handshake, a sequence of Request To Send (RTS), Clear To Send (CTS), Data, and ACK frames is used for any *single* data packet. Figure 1 shows an example where the source node *Src* sends a packet to the destination node *Dst*. In the figure, node A is within the transmission range of node *Src* but out of the range of node *Dst*. On the other hand, node B is within the range of *Dst* but out of the range of *Src*. In wireless ad hoc networks, the four-way handshaking is normally used to cope with the common hidden-terminal problem.

When a frame exchange sequence between two nodes (e.g., nodes *Src* and *Dst* in Figure 1) is going on, all the other nodes (e.g., nodes A and B) that are within the range of the sender or the receiver should defer their transmission to prevent interference with the on-going sequence. In order to achieve this, every frame should contain a duration field, which stores the time duration that the medium is to be reserved for the on-going sequence. In IEEE 802.11, the duration value carried in a given frame should be large enough to allow the transmission of *all* the *remaining* frames in the sequence. For example, in the RTS frame, the duration value should be equal to the time needed for the transmission of the CTS, Data, and ACK frames plus three Short Inter-frame Space (SIFS) intervals. This duration is represented by D_{RTS} in Figure 1. The duration values carried in the CTS, Data, and ACK frames are represented by D_{Data}, D_{CTS}, and D_{ACK}, respectively. Note that the D_{ACK} is zero as the ACK is the last frame in the sequence. When an overhearing node (e.g., node A or B) overhears a frame, it will update its variable called Network Allocation Vector (NAV) with the duration value carried in the overheard frame if the following conditions are satisfied: *(i)* the frame is received without error; *(ii)* the frame is not addressed to itself; and *(iii)* the duration value carried in the frame is greater than the current NAV value. The condition *(iii)* is required since a node may overhear many frames corresponding to different exchange sequences, which is very likely in multi-hop ad hoc networks. Since a node with a NAV greater than zero will defer its transmission as if the medium is physically busy, the above mechanism is also called virtual carrier sensing (VCS). In a multi-hop ad hoc networks, it is extremely important for the VCS to ensure that all the overhearing nodes defer properly. If the NAV value is larger than the desired one, the medium is unnecessary idle. On the other hand, when the NAV value is smaller, it will result in collisions with the ongoing transmission. Both the idle time and collisions reduce the utilization of the medium.

In addition to the deferment enforced by VCS, whenever a node detects an *erroneous* frame, the node defers its transmission by a fixed duration indicated by the Extended Inter Frame Space (EIFS) constant [3]. We call this *EIFS deferment*.

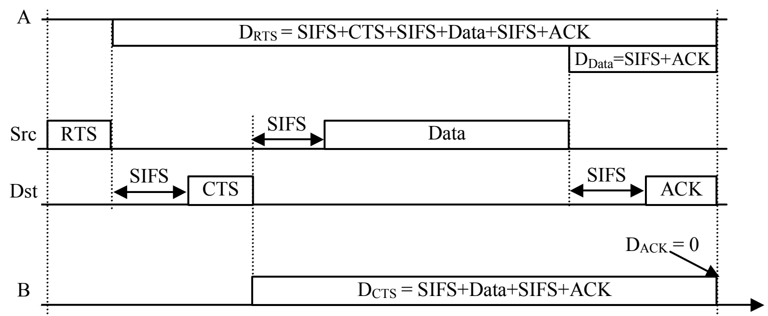

Figure 1: Virtual Carrier Sensing in IEEE 802.11

2.2 Related Work

In [12], the “Auto-Rate Fallback (ARF)” protocol for IEEE 802.11 has been presented. Specifically, if the ACKs for two consecutive Data frames are not received by the sender, the sender reduces the transmission rate to the next lower data rate and starts a timer. When, the timer expires or *ten* consecutive ACKs are received, the transmission rate is raised to the next higher data rate and the timer is cancelled. However, if an ACK is not received for the immediately next data frame, the rate is lowered again and the timer is restarted. The ARF is simple and easy to incorporate into the IEEE 802.11. However, as pointed out in [9], the ARF is purely heuristic and cannot react quickly when the wireless channel conditions (e.g., SNR) fluctuate. Realizing this, [9] [10] and [11] have proposed some enhancements to the ARF.

In all the above algorithms, the rate adaptation is performed at the sender. However, it is the receiver that can perceive the channel quality, and thus determine the transmission rate more precisely. Observing this, the authors in [7] have presented a receiver-based auto-rate (RBAR) protocol assuming that the RTS/CTS mechanism is there. The basic idea of RBAR is as follows. First, the receiver estimates the wireless channel quality using a sample of the Signal to

Noise Ratio (SNR) of the received RTS, then selects an appropriate transmission rate for the Data frame, and piggybacks the chosen rate in the responding CTS frame. Then, the sender transmits the Data frame at the rate advertised by the CTS. The simulation results in [7] show that RBAR can adapt to the channel conditions more quickly and in a more precise manner than the ARF [12] does, and thus it improves the performance greatly. The Opportunistic Auto Rate (OAR) [8] extends the RBAR by giving preference to the nodes who are experiencing good channel conditions. The OAR greatly improves the aggregate throughput of the network. However, it worsens the fairness (in terms of throughput) compared to the RBAR though the temporal fairness is the same as that of RBAR. All the above schemes have considered the rate adaptation of the Data frame only, while the control frames are always transmitted at a low basic rate.

3 Motivation

3.1 Transmission Rate of the Control Frames

In IEEE 802.11, it specifies that all the control frames should be transmitted at one of the rates in the *basic rate set* so that all the potentially interfering nodes can decode the duration information carried in these frames. Obviously, the *basic rate set* contains only one or several lowest transmission rates among all the supported rates. For example, in the IEEE 802.11b that supports rates of 1, 2, 5.5 and 11 Mbps, the *basic rate set* normally contains only 1 and 2 Mbps[1]. Another rule regarding the transmission rate of the control frames is as follows [3]: in order to allow the transmitting node to calculate the duration field value, the responding node should transmit its Control Response frame (either CTS or ACK) at the highest rate in the *basic rate set,* which is less than or equal to the rate of the received frame. Since there is no rule about how to adapt the transmission rate of the RTS frame, a conservative implementation will choose to transmit it at the lowest rate, i.e., 1 Mbps. As a result, the CTS will also be transmitted at 1 Mbps. On the other hand, the ACK frame can be transmitted at 2 Mbps according to the above rule whenever the transmission rate of the Data frame is equal to or greater than 2 Mbps. However, in most of the rate adaptation algorithms [7] [8] [9] [10], they simply assume that all the control frames are always transmitted at the lowest basic rate (i.e., 1 Mbps in IEEE 802.11b). As a result, the bandwidth wastage by the control frames is fixed irrespective of the length and the transmission rate of the Data frame. In the following, we will develop a simple analytical model to show the wastage in IEEE 802.11.

[1] This is also to ensure the coexistence with the legacy IEEE 802.11 products that only support 1 and 2 Mbps.

3.2 Maximum Throughput of IEEE 802.11

If we want to consider the maximum throughput of IEEE 802.11 under the saturation condition, we need to find the probability of collision and that of the medium being idle, as done in [1]. However, since our main objective is to show the overheads introduced by the physical and MAC layers, we develope a simple model under the assumptions that only one active single-hop flow is present in the network and that no wireless errors occur. Under these assumptions, the *average* time required to transmit a packet is:

$$T = DIFS + (CW_{\min}/2) \times T_{slot} + 3 \times SIFS + T_{RTS} + T_{CTS} + T_{Data} + T_{ACK} \qquad (1)$$

where DCF Inter-Frame Space (DIFS) is the time that a node should defer before initiating its transmission. The CW_{min} is the minimum Contention Window (CW) while the T_{slot} is the slot time in microsecond μs. Therefore, $(CW_{min}/2)\times T_{slot}$ is the *average* back-off time. T_{RTS}, T_{CTS}, T_{Data}, and T_{ACK} represent the transmission time of the corresponding *physical layer* frames. For a specific physical layer frame, the transmission time T_{frame} is:

$$T_{frame} = L_{frame} / R_{frame} + PHY_{hdr} \qquad (2)$$

where L_{frame} and R_{frame} are the length of the MAC layer frame and the transmission rate chosen by the MAC layer, respectively. The PHY_{hdr} is the physical layer overhead in units of time. Since $DIFS$, CW_{min}, T_{slot}, $SIFS$, and PHY_{hdr} are fixed for a specific physical layer, we denote the sum of the fields containing them as C_{phy}. That is,

$$C_{phy} = DIFS + (CW_{\min}/2) \times T_{slot} + 3 \times SIFS + 4 \times PHY_{hdr} \qquad (3)$$

Since the lengths of the RTS, CTS, ACK, and the header of Data frame are fixed, i.e., 20, 14, 14, 28 bytes, respectively, equation (1) becomes,

$$T = C_{phy} + 8 \times (20/R_{RTS} + 14/R_{CTS} + (L+28)/R_{Data} + 14/R_{ACK}) \qquad (4)$$

where L is the length of the data packet handed to the MAC layer from the upper layer, while R_{RTS}, R_{CTS}, R_{Data} and R_{ACK} are the transmission rates (in terms of Mbps) chosen by the MAC layer for the corresponding frames. Therefore, the maximum throughput (in terms of Mbps) is as follows:

$$Th = \frac{L \times 8}{T} = \frac{L \times 8}{C_{phy} + 8 \times (20/R_{RTS} + 14/R_{CTS} + (L+28)/R_{Data} + 14/R_{ACK})} \qquad (5)$$

Figure 2 presents the numerical results corresponding to the IEEE 802.11b, where the R_{RTS} and R_{CTS} are equal to 1 Mbps and the R_{ACK} is equal to 1 or 2 Mbps determined by R_{Data}, as discussed in Section 3.1. The C_{phy} corresponding to

IEEE 802.11b is 1168 μs. The results are obtained by varying the packet length L and the transmission rate of the Data frame (i.e., R_{Data}). It is easy to see that the throughput increases with L and R_{Data}. Moreover, the overheard is considerable. For example, when L is equal to 64 bytes and R_{Data} is 11 Mbps, the throughput is about 0.33 Mbps. From equations (3) and (4), it is easy to see that the overheads can be reduced in several ways. First, as done in [2], the contention window can be tuned based on the contention level (e.g., number of active nodes) in the network. The tuning method reduces the time that the medium is idle when the contention level is low, and it will reduce the probability of collisions when the contention level is high. The second method is to reduce the physical layer overheads (e.g., preamble) as done in [16]. Another method is to reduce the usage of RTS/CTS as much as possible. For example, whenever a node gets control of the medium, it should transmit several data packets consecutively (i.e., a burst) in which the RTS/CTS is transmitted only once as done in [8]. In contrast, in this paper, we aim to transmit the control frames at a higher rate to reduce the overheads.

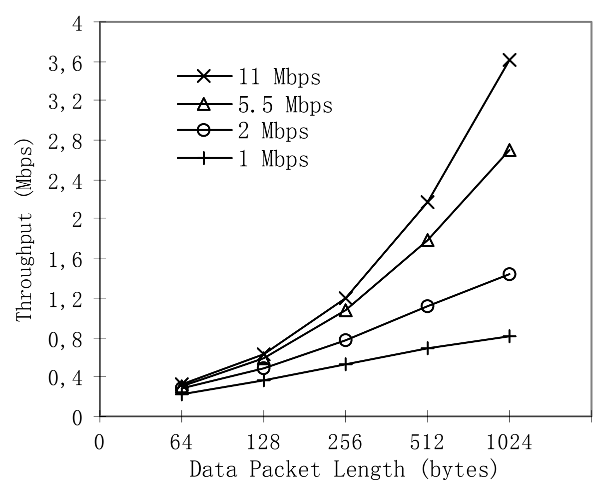

Figure 2: Maximum Throughput in IEEE 802.11b

4 Full Auto Rate (FAR) Algorithm

In this section, we first discuss the VCS issues in the multi-rate scenario, and introduce a modified virtual carrier sensing (MVCS) mechanism assuming that the control frames are always transmitted at the low basic rate. Then, we show that the control frames also can be transmitted at a higher rate with the help of EIFS deferment. Finally, we present our FAR, and discuss how the MVCS is refined in the FAR where the rates of *all* the frames are adapted.

4.1 VCS Issues in Multi-rate IEEE 802.11

VCS in IEEE 802.11 is essentially designed for the single-rate scenario. To calculate the duration value for a frame, the transmission rates and the lengths of *all* the remaining frames in the sequence must be known a priori. For example, to calculate the D_{RTS}, the node *Src* (Figure 1) must know the transmission rates and the lengths of CTS, Data and ACK frames. As the lengths of all the frames are known at the *Src,* and if the transmission rate of the RTS/CTS is assumed to be 1 Mbps, so for the VCS to operate properly, only the transmission rate of the Data frame has to be known before the transmission of the RTS, which is certainly known for the sender-based algorithms [9] [10] [11] [12].

However, in the receiver-based rate adaptation algorithms such as RBAR [7] and OAR [8], since the actual transmission rate of the Data frame is determined by the receiver (e.g., *Dst*) after it has received the RTS frame, the sender (i.e., *Src*) has to calculate the D_{RTS} by assuming a transmission rate. Whenever the rate chosen by the receiver is different from the one assumed by the sender, the reservation made by D_{RTS} is inaccurate. The reservation can be corrected by the D_{CTS} or D_{Data}, which will always be precise if the rate of the control frames follows the rules described in Section 3.1. However, two problems arise. The *first* one is that since the *Data* frame may not be transmitted at the basic rate, it may not be decoded correctly by the overhearing nodes and thus the reservation cannot be corrected by the D_{Data}. The *second* problem occurs as follows. When the actual rate of the Data frame is larger than the rate assumed by the sender, the remaining NAV value previously updated according to D_{RTS} is *larger* than the new duration value contained in the CTS or Data frame. As mentioned before, to ensure the proper operation of the VCS in a multi-hop network, the IEEE 802.11 does not allow reduction in NAV. Therefore, the reservation made by D_{RTS} also cannot be corrected.

To cope with the *first* problem, the RBAR [7], which is the only one that has considered the above problems, defines a sub-header, called the *Reservation SubHeader (RSH)*, to replace the original header of the Data frame. The RSH includes a separate Frame Check Sequence (FCS) and should be transmitted at one of the basic rates to ensure all the overhearing nodes receive it correctly. Moreover, the physical layer frame needs to be modified to include fields indicating the rate of the RSH header and the rate of the actual Data part, and thus the physical layer may be required to switch transmission rates twice during the transmission of the *payload*. To cope with the *second* problem, the VCS is modified to allow the NAV value to be reduced by the frames belonging to the *same* frame exchange sequence. To achieve this, every node has to maintain a history of the NAV updates. Obviously, the RBAR [7] designed as above will increase the complexity as well as overheads, and thus it may not be practically useful as pointed out in [9].

The above discussion assumes that the rate of the control frames will follow the rules discussed in Section 3.1. It is obvious that the situation will be more complex when the transmission rate of the control frames is also made variable, which is one of the objectives in this paper. Moreover, in such a scenario, even the sender-based algorithms need to cope with the problems caused by the inaccurate reservation. Therefore, it is crucial to devise a new VCS mechanism for the multi-rate IEEE 802.11.

4.2 Modified Virtual Carrier Sensing (MVCS)

From Section 4.1, we know that the main problem in the current VCS for multi-rate is that it needs to know the transmission rate of all the remaining frames. However, we must realize that the main purpose of the VCS, a complementary component of the physical carrier sensing, is to allow the *next* frame in the ongoing sequence to go through. Therefore, instead of reserving the medium for all the remaining frames in the sequence, we propose to reserve the medium only for the immediate next frame. We call it Modified Virtual Carrier Sensing (MVCS).

The MVCS can operate properly in a multi-hop network with the help of the physical carrier sensing. In Figure 3, if a node (e.g., node C) is within the range of the sender as well as the receiver, the node can defer properly by using only the physical CS mechanism. Now let us consider the nodes (e.g., node A) that are within the range of the sender but not of the receiver. When node A overhears an RTS frame, under the MVCS, it will defer by a duration to allow the CTS to pass through. Then, when the Src is transmitting the Data frame, the physical CS at node A will make itself to defer. After node Src has transmitted the Data frame, node A will defer for a duration to allow the transmission of the ACK frame. Therefore, the nodes within the range of the sender but not of the receiver can defer properly. This is exemplified in Figure 4. Note that it is not possible for node A to initiate its transmission during the SIFS interval at the *Src* just before transmitting the Data frame. The reason is, after node A has deferred by D_{RTS} it will further defer by a DCF Inter-Frame Space (DIFS), which is larger than the SIFS. One can also verify that the nodes (e.g., node B) within the range of the receiver but not of the sender can defer properly. Therefore, all the nodes that will potentially interfere the ongoing frame exchange sequence can defer properly. So far, we have assumed that all the frames will be overheard clearly by the interfering nodes, which may not be true in a practical scenario. For example, when the sensing range is greater than the transmission range, some nodes can detect the carrier of a frame but cannot interpret the contents. However, with the help of the EIFS deferment, all the nodes will defer properly as will be discussed in Section 4.3.

Clearly, the MVCS reduces the overheads as well as the complexity in the receiver-based rate adaptation algorithms. For example, the Reservation SubHeader (RSH), the maintenance of NAV update history, and the modifications to the physical layer are not required any more. Moreover, since the MVCS modifies the VCS of IEEE 802.11 slightly only, it can be easily incorporated into the receiver-based auto-rate algorithms (e.g., RBAR [7]). More importantly, the MVCS forms the basis to adapt the transmission rates of the control frames.

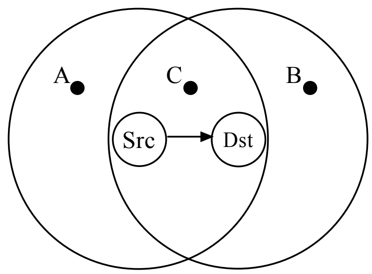

Figure 3: Scenario showing the range of interference

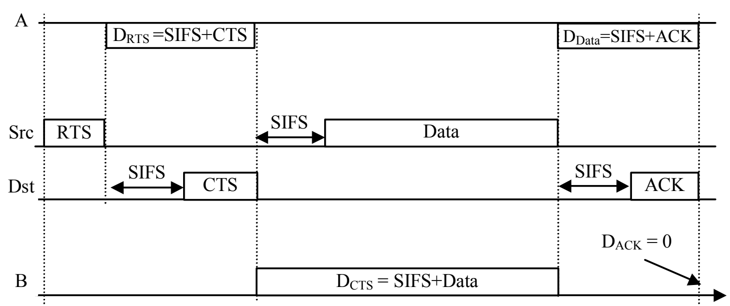

Figure 4: Modified Virtual Carrier Sensing (MVCS)

4.3 Rate Adaptation of Control Frames

As discussed before, the IEEE 802.11 specifies that all the control frames (i.e., RTS, CTS and ACK) should be transmitted at one of the basic rates to ensure that all the nodes in the interference range can interpret them. In fact, it can be easily verified by checking the frame format defined in the standards that in an ad-hoc network the only useful information contained in the RTS/CTS/Data/ACK for the *overhearing* nodes is the duration value. On the other hand, as in the discussion of the MVCS, to prevent potential interference from other nodes, a frame needs to reserve the medium only for the *next* frame in the sequence. Therefore, if somehow the overhearing nodes can defer properly to allow the next frame pass through even when they cannot interpret the contents of the overheard frames, the control frames can also be transmitted at a rate higher than the basic rates. This can be achieved with the help of the EIFS deferment as explained in the following. Since the EIFS is equal to *SIFS + TxTime (ACK) + DIFS* (i.e., 364 μs in DSSS) [3], it is long enough for the complete transmission of any of the control frames at any rate. For example, when a node detects an erroneous frame corresponding to the RTS, it will defer by an EIFS value, which is large enough to allow the

CTS to go through. This is also true when a node detects an erroneous frame corresponding to a Data or ACK. Therefore, we propose that the RTS and ACK frames should be transmitted at the highest attainable rate between the communicating pair under the given channel conditions. However, when the detected erroneous frame corresponds to a CTS frame, since the transmission time required by the Data frame may be greater than the EIFS, the overhearing nodes, relying solely on the EIFS deferment, may not defer properly. Therefore, when the transmission time of the Data frame is greater than the EIFS, the CTS should be transmitted at the lowest basic rate to ensure that the CTS could be clearly understood even by the overhearing nodes. Note that with the increase of the maximum data rate in the emerging standards (e.g., 54 Mbps in IEEE 802.11a and g), the chances of the transmission time of the Data frame being smaller than the EIFS will increase.

4.4 Full Auto Rate (FAR)

Now we present our Full Auto Rate (FAR) algorithm, which integrates the above two proposals, i.e., MVCS and rate adaptation of control frames. Table 1 lists the main operations of the FAR. We first discuss the rate adaptation of the frames (see *second* column of the table). Our FAR combines the sender- and receiver-based methods proposed in the literature. Since the channel conditions may differ in the opposite directions in-between two communicating nodes, and since within a specific frame exchange sequence the RTS and CTS are the *first* frames transmitted in the corresponding directions, the rate selection has to be done at the sender side of these frames. On the other hand, since the RTS/CTS frames can probe the channel conditions in the two directions, the rate selection for the Data/ACK frames can be done at the receiver side of these two frames. Specifically, the transmission rate of the RTS is chosen as follows. Every node will maintain a cache variable, which indicates the rate that should be used for the RTS. The initial value of this cache variable is set to the lowest basic rate (i.e., 1 Mbps). Whenever an attempt of transmitting an RTS or Data frame is successful, the variable will be updated with the rate being used for this successful transmission. Otherwise, whenever the attempt fails, the variable will be reset to the lowest basic rate. The rate selection for the CTS frame is a little bit more complex. If the transmission time of the Data frame (which can be computed at the receiver after it has chosen a rate for the Data frame) is greater than the EIFS value, the CTS frame will be transmitted at the lowest basic rate to ensure that it will be understood by all the nodes in the interference range. Otherwise, the CTS will be transmitted at a rate that is determined by using a similar method as the rate selection for the RTS frame. Obviously, for the RTS/CTS frames, one can adopt any other sender-based algorithm (e.g., [12]) to decide the rate. Now we discuss how the transmission rates of the Data/ACK frames are decided by the receiving side of these frames.

Specifically, based on the SNR of the received RTS, the receiver suggests a transmission rate for the Data frame and piggyback this rate in the CTS as done in the RBAR [7]. Similarly, the receiving side of the CTS frame will also choose a rate for the ACK frame based on the SNR of the CTS and piggyback the rate in the Data frame.

Now we discuss how the MVCS (proposed in Section 4.2) should be refined to operate in our FAR (see *third* and *fourth* columns of Table 1). In the FAR, before the transmission of the RTS frame, the sending node does not know the exact transmission rate of the CTS frame in the sequence. Therefore, the medium should be reserved as if the CTS will always be transmitted at the lowest basic rate. Moreover, as the length of CTS and the lowest basic rate are fixed values, the corresponding duration is also fixed, which can be easily calculated at every node overhearing RTS. Therefore, instead of having the duration value in the RTS frame, it carries the *length* of the Data frame in bytes, which is needed by the receiver to calculate the D_{CTS}. On the other hand, before the transmission of the CTS and Data frames, the corresponding sending node always knows the exact transmission rate of the immediate next frame in the sequence. Therefore, the CTS and Data frames should simply carry the duration in microseconds, which is long enough for the *next* frame in the sequence to pass through. Note that the duration value in the ACK frame is zero. Also note that the rate chosen for the Data and ACK frames can be easily derived from the duration value carried in the CTS and Data frames, respectively. Therefore, it is unnecessary to add an additional filed to store the rate being chosen. As for the overhearing nodes, when they overhear an RTS frame, they will defer by a *fixed* duration that is long enough for the CTS to be transmitted at the lowest basic rate. Otherwise, when they overhear any other types of frame, they will simply defer by a duration advertised in the overheard frame. Note that the deferment when overhearing an RTS may not be precise since the CTS may be transmitted at a higher rate. However, the deviation is very small.

Table 1: Main Operations in FAR

Frame	Transmission Rate	Duration Field	Deferment Time (overhearing nodes)
RTS	Cached rate	Data length (in bytes)	TxTime(CTS, Basic-rate)
CTS	Cached rate or Lowest basic-rate	Data length / Chosen rate	Duration value
Data	Piggybacked in CTS	ACK length / Chosen rate	Duration value
ACK	Piggybacked in Data	0	Duration value

5 Analytical Modeling

In this section, based on the model present in Section 3.2, we will analytically show the advantages of FAR. As clearly shown in [7], in wireless ad hoc networks, the receiver-based algorithm has much better performance than the sender-based algorithms (e.g., [12]). Therefore, here we only compare the maximum throughputs of three schemes: RBAR [7], RBAR with MVCS, and FAR. Note that we do not consider the performance of the OAR [8] since it is mainly based on the RBAR.

RBAR with MVCS versus RBAR: When the original VCS is used, the RBAR transmits the RSH sub-header of the Data frame at the same rate as that of RTS (i.e., 1 Mbps). Moreover, RSH sub-header needs a separate four-bytes FCS. On the contrary, with MVCS all these are unnecessary. Therefore, using the eq. (5), the ratio between the maximum throughputs of the RBAR with MVCS and the RBAR is:

$$Ratio = \frac{C_{phy} + 8 \times (20/R_{RTS} + 14/R_{CTS} + 28/R_{RTS} + (L+4)/R_{Data} + 14/R_{ACK})}{C_{phy} + 8 \times (20/R_{RTS} + 14/R_{CTS} + (L+28)/R_{Data} + 14/R_{ACK})} \qquad (6)$$

FAR versus RBAR with MVCS: Under FAR, we assume that all the control frames are transmitted at the same rate as the Data frame, therefore the ratio is:

$$Ratio = \frac{C_{phy} + 8 \times (20/R_{RTS} + 14/R_{CTS} + (L+28)/R_{Data} + 14/R_{ACK})}{C_{phy} + 8 \times (20/R_{Data} + 14/R_{Data} + (L+28)/R_{Data} + 14/R_{Data})} \qquad (7)$$

FAR versus RBAR: Similarly, the ratio between the maximum throughputs under FAR and RBAR is as follows:

$$Ratio = \frac{C_{phy} + 8 \times (20/R_{RTS} + 14/R_{CTS} + 28/R_{RTS} + (L+4)/R_{Data} + 14/R_{ACK})}{C_{phy} + 8 \times (20/R_{Data} + 14/R_{Data} + (L+28)/R_{Data} + 14/R_{Data})} \qquad (8)$$

Figures 5, 6, and 7 present the results computed from the equations (6), (7), and (8), respectively. Note that in Figure 6 when the R_{Data} is equal to 1 Mbps, the ratio is always one, which is not shown in the figure. From these results, we see that when all the other parameters are fixed, as R_{Data} increase, the throughput ratio also increases. In fact, one can formally prove this by showing that the first-order derivative with respect to R_{Data} is always greater than zero in equations (6), (7), and (8). On the other hand, the first-order derivative with respect to C_{phy} is always *smaller* than zero, implying that a smaller C_{phy} leads to a higher ratio. The above two properties of the ratios are crucial because the C_{phy} will become smaller while the R_{data} will become larger as the IEEE 802.11 evolves. For example, in IEEE 802.11a [4], the C_{phy} is reduced to 226.5 µs, while the R_{data} can reach 54 Mbps. The first-order derivative with respect to L is always *smaller* than zero, implying that a larger L leads to a smaller ratio. However, when the L increases, the throughput also increases as shown in Figure 2. Therefore, though the ratio is decreasing with the increase in L, the absolute throughput improvement (in terms of Mbps) is higher, which is also observed from the simulation results in Section 6.

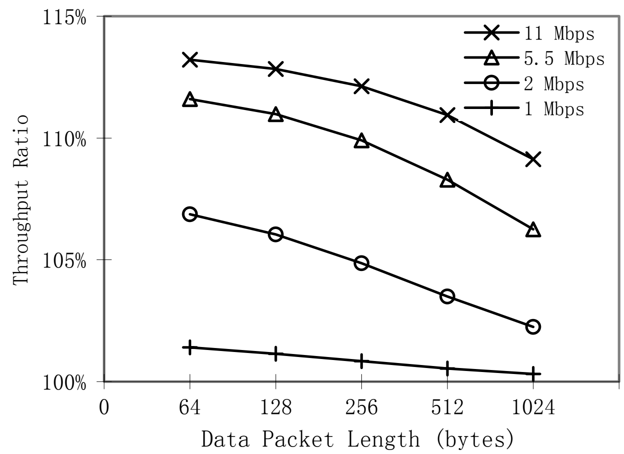

Figure 5: Throughput Improvements of MVCS

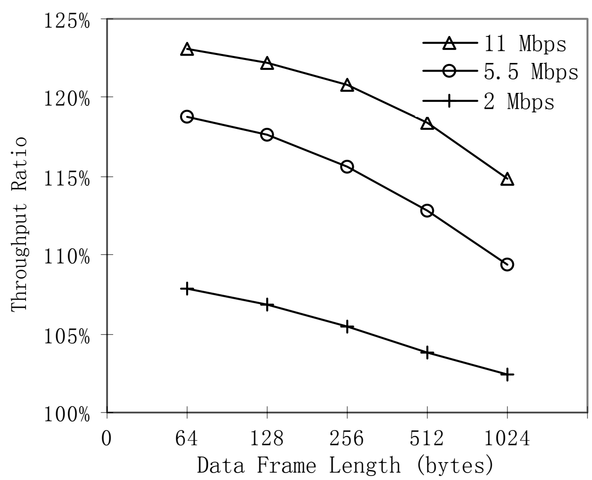

Figure 6: Throughput of FAR versus of RBAR with MVCS

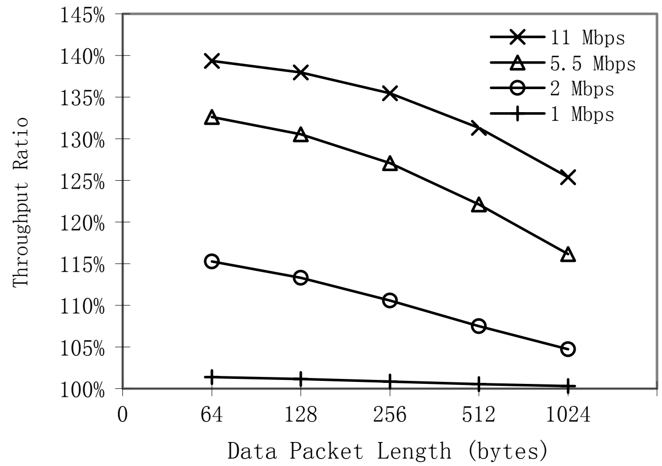

Figure 7: Throughput Improvements of FAR compared to RBAR

6 Simulation Results

The simulations were performed under the NS-2 with CMU wireless extensions [14]. As in [8], we use the fading model with Ricean probability density that has been incorporated into the NS-2 by the authors of [13]. The optimum data rate for a given SNR is selected according to [8]. In the simulation, *seven* different algorithms are considered. Four of them are run at a fixed transmission rate of 1, 2, 5.5, or 11 Mbps as supported in IEEE 802.11b [5]. The other three are rate adaptation algorithms: RBAR, RBAR with MVCS, and FAR. Again, here we do not compare the performance of FAR with the sender-based algorithms (e.g., [12]) and the OAR [8] due to the reasons discussed at the beginning of Section 5. For each of the algorithms, we conduct the simulation with five different packet lengths, i.e., 64, 128, 256, 512, 1024 bytes. For each of the flows, the source rate is large enough to occupy the whole bandwidth. In the single-hop scenario, the static routing is used while in the multi-hop scenario the Ad-hoc On-demand Distance Vector (AODV) [15] is used as the routing protocol. The nominal transmission range as well as the sensing range is 250 meters. The maximum speed of mobility is 5 m/s while the pause time is zero. Each of the simulation is run for 250 seconds.

6.1 Single Flow within Single-hop

In this simulation, there are only two nodes, which are continuously moving within a 200×200 meter arena. Figure 8 presents the average throughputs. It is easy to see that the algorithms with rate adaptation have better performance than all the algorithms with fixed rate. Within the fixed rate algorithms, the one at the rate 5.5 Mbps shows the best performance, implying that the *average* distance of the two nodes in this scenario is close to the optimal distance under which the rate of 5.5 Mbps achieves the best performance. Moreover, for all the fixed rate algorithms, the throughputs conform to the upper bound derived in Section 3.2.

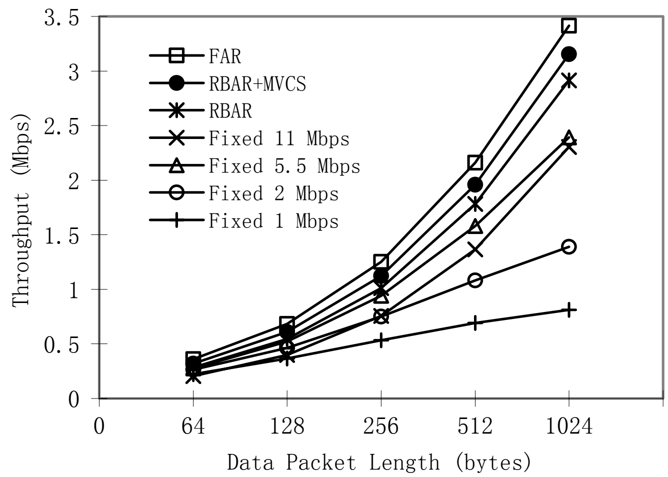

Figure 8: Throughput under single-flow within single-hop

Among the adaptive algorithms, as expected, the FAR shows the best performance while the RBAR with MVCS is the second best. Moreover, the longer the packet length is, the more absolute gain is achieved by the FAR over the RBAR. For example, when the packet length is 1024 bytes, the throughput improves by 0.5 Mbps whereas the improvement is about 0.08 Mbps when the packet length is 64 bytes. The reason is that for a packet of small length, the transmission time reduced by the higher rate is very small compared to the fixed overheads. However, the results are somewhat different from the throughput ratio viewpoint. Figure 9 presents the throughput ratio among the adaptive rate algorithms. The larger the packet length, the smaller is the ratio. This has also been reflected by the analytical results in Section 5. Note that the ratio obtained from the simulation is smaller than that from the analytical results at the rate of 5.5 Mbps. The reason is, in the analysis the rate of RTS/CTS under FAR is assumed equal to the rate of the Data frame, which is not always true in practice.

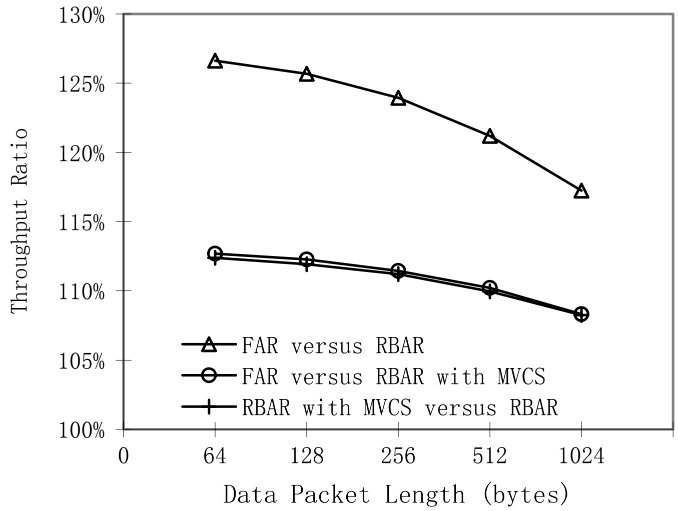

Figure 9: Throughput Ratio under Adaptive Rate Algorithms

6.2 Multiple Flows within Single-hop

In this simulation, there are *ten* nodes with *five* flows and all the nodes are continuously moving within a 200×200 meter arena. Figure 10 presents the aggregate average throughput of all the flows. In general, the performance among the seven algorithms follows the same trend as in the previous scenario except for the following two observations: *(i)* among the fixed rate algorithm, it is the algorithm with 11 Mbps that shows the best performance rather than the one with 5.5 Mbps; and *(ii)* the throughput under RBAR is very close to the fixed rate algorithm with 11 Mbps. The reason for the first observation is as follows. Under the fixed rate algorithm with 11 Mbps, all the transmissions experiencing an SNR that is not large enough to support 11 Mbps will fail, making the flows continuously back off and defer. Moreover, since in this scenario there are 10 nodes within a small area, it is very likely that at least one of the flows can support the 11 Mbps. Therefore, most of the time the medium will be occupied by the flows with 11 Mbps, explaining why the 11 Mbps gets the best performance in this scenario. On the other hand, in the previous scenario in which only two nodes are present, the nodes may not communicate at 11 Mbps during most of the time, explaining why the 5.5 Mbps gets the best performance. Now we explain the second observation. Compared to the fixed rate algorithm with 11 Mbps, the RBAR will allocate more time to the flows that can only support rates lower than 11 Mbps, reducing the aggregate throughput of RBAR. On the other hand, the time wasted by the flows that cannot support 11 Mbps in the fixed rate algorithm is greatly reduced by the RBAR since it can adapt to the channel conditions. Therefore, the above two effects cancel out each other, and finally result in similar performance. However, the RBAR is better as it is fairer than the fixed 11 Mbps algorithm.

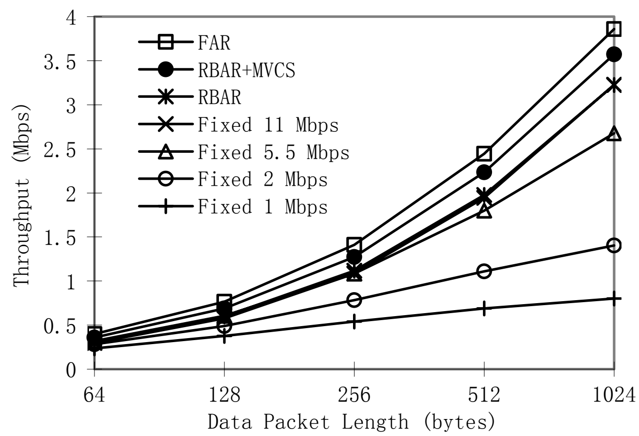

Figure 10: Throughput under multiple flows within single-hop

6.3 Multiple Flows within Multi-hop

In this scenario, there are *twenty* nodes with *five* flows and all the nodes are moving within a 1000×500 meter arena. Figure 11 presents the aggregate average throughput of all the flows. The performance of the adaptive rate algorithms follows the same trend as in the previous examples, i.e., the FAR is the best, followed by RBAR with MVCS, and then by RBAR. However, in terms of the aggregate throughput, the adaptive algorithms do not show any distinct advantage compared to some of the fixed rate algorithms. For example, the fixed rate algorithm with 5.5 Mbps always shows the best performance, while RBAR is inferior to most of the fixed rate algorithms except the one with 1 Mbps. Through detailed analysis of the trace records, we found that the interaction between MAC layer and the routing protocol plays a major rule in determining the performance. Under the fixed rate, whether or not a multi-hop flow can start depends on whether that flow can find a route from the source to the destination with all the hops having that fixed rate. When the fixed rate is very high (e.g., 11 Mbps), it is very likely that sometimes no flows in the network can find a route. On the other hand, when the fixed rate is very low (e.g., 1 or 2 Mbps), all the flows may be active but the medium is used at a very low rate. This explains why the fixed rate algorithm with 5.5 Mbps shows the best performance. Under the adaptive rate algorithm, it is very likely that all the flows can find a route since the nodes can adjust the rate based on the channel conditions. Therefore, though the aggregate throughputs of the adaptive algorithms may be smaller than that under the fixed rate algorithms, the bandwidth is more evenly distributed among the flows.

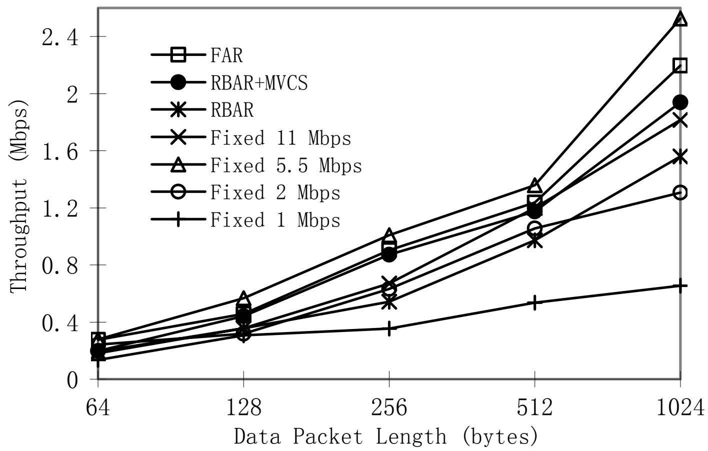

Figure 11: Throughput under multiple flows within multi-hop

6.4 Lessons learnt from the simulation

Three important conclusions are obtained: *(i)* the FAR improves the performance of IEEE 802.11 in all the scenarios; *(ii)* it is not enough to compare the performance only in terms of *throughput* in the scenarios with multiple flows. Fairness should also be considered; *(iii)* In the multi-hop networks, the interaction between the routing protocol and MAC protocol will greatly affect the overall performance. Therefore, it is of great importance to do cross-layer optimization in wireless ad hoc networks.

7 Conclusions

In this paper, we have introduced a new rate adaptation algorithm, called Full Auto Rate (FAR). Different from the schemes in the literature, which only consider the rate adaptation of the Data frame and assume that the control frames are always transmitted at a low basic rate, our FAR aims to transmit *all* the frames at the highest attainable rate under the given channel conditions. Moreover, while the schemes in the literature are either sender- or receiver-based, our FAR is a combination of the two methods, i.e., rate adaptations of RTS/CTS are sender-based and those for Data/ACK are receiver-based. We have also thoroughly addressed the issues involved in the virtual carrier sensing (VCS) in multi-rate ad hoc networks. In order to ensure the correct operation of the VCS, we have proposed a modified virtual carrier sensing (MVCS) mechanism, which addresses the issues very well. To study the performance of FAR, both analytical and simulation methods are used, which show that the FAR greatly improves the performance of IEEE 802.11. Further improvements are possible by refining the routing protocols to enable them to make full use of the multi-rate capability provided by the MAC layer.

Reference:

[1] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," IEEE Journal on Selected Areas in Communications, March 2000, pp.535-547.

[2] F. Cali, M. Conti, E. Gregori. "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit," IEEE Journal on Selected Areas in Communications, December 2000, pp.785-799

[3] IEEE 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," IEEE 802.11 standards, June 1999.

[4] IEEE 802.11a, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer (PHY) in the 5 GHz Band," Supplement to IEEE 802.11 standards, Sep. 1999.

[5] IEEE 802.11b, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer (PHY) Extension in the 2.4 GHz Band," Supplement to IEEE 802.11 standards, Sep. 1999.

[6] IEEE 802.11g, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher Data Rate Extension in the 2.4 GHz Band," Supplement to IEEE 802.11 standards, June 2003.

[7] G. Holland, N. Vaidya, P. Bahl, "A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks," in ACM MOBICOM, July 2001.

[8] B. Sadeghi, V. Kanodia, A. Sabharwal, E. Knightly, "Opportunistic Media Access for Multirate Ad Hoc Networks," in ACM MOBICOM, September 2002.

[9] D. Qiao, S. Choi, K.G. Shin, "Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs," in IEEE Transaction on Mobile Computing, Oct-Dec, 2002.

[10] J. Pavon, S. Choi, "Link Adaptation Strategy for IEEE 802.11 WLAN via Received Signal Strength Measurement," in IEEE ICC, 2003.

[11] P. Chevillat, J.Jelitto, A. Noll Barrerto, H.L. Truong, "A Dynamic Link Adaptation Algorithm for IEEE 802.11a Wireless LANs," in IEEE ICC, 2003.

[12] A. Kamerman, L. Monteban, "WaveLAN-II: A High-performance Wireless LAN for the Unlicensed Band," Bell Labs Technical Journal, pages 118-133, Summer 1997.

[13] R. Punnoose, P. Nikitin, D. Stancil, "Efficient Simulation of Ricean Fading within a Packet Simulator," in IEEE

Vehicular Technology Conference, 2000.

[14] CMU Monarch Group. CMU Monarch Extensions to NS, http://www.monarch.cs.cmu.edu/.

[15] C. E. Perkins and E. M. Royer. “Ad-hoc On-Demand Distance Vector Routing,” in Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90–100, New Orleans, LA, February 1999.

[16] Ka (Gavin) Yeung, “Packet Concatenation (PAC) for Rate Adaptive Mobile Ad Hoc Networks”, http://www.cs.ucla.edu/classes/fall02/cs218/l1/uploads/MACFinalReportPAC218Gerla.pdf