Understanding the FICA MAC Protocol in High Data Rate WLANs

Fatima Zarinni Samir R. Das
Computer Science Dept., Stony Brook University, USA

***Abstract*—Recently wireless radios and hardware technologies have been developed that allow operation on very wide channels and that can support very high physical layer data rates (1Gbps and up). However, it is proven that the conventional 802.11 CSMA/CA MAC protocol causes a drastic under-utilization of such high-speed channels. In order to improve channel utilization, a new scheme, called, *Fine-Grained Channel Access (FICA)*, has been recently put forward. While the FICA approach appears more promising than the other proposed schemes for high data rate WLANs, it has not been studied extensively before.**

Hence, in this paper, we focus on the FICA MAC protocol, and we study this protocol thoroughly in different traffic scenarios and network topologies. We identify, for the first time, some of the serious problems that can arise with the FICA MAC protocol. We call these problems *deafness*, *muteness* and a form of *hidden terminal problem*. We quantify the impact of these problems on the performance of FICA via extensive simulations. Our results show that under some typical scenarios, FICA can perform even worse than the conventional 802.11 DCF, in terms of channel utilization, per-user-throughput or fairness. The insights obtained in this paper motivates the need for addressing FICA's problems and paves the path for future development of better new MAC protocols for high data rate WLANs.

I. Introduction

The advancements in MIMO and OFDM physical layer technologies, and the development of tranceivers that can operate on very wide channels, have turned the once dream of having high speed wireless links, into a reality. The forthcoming successor of the current 802.11n standard, the 802.11ac standard, is envisioned to support physical layer data rates even higher than 1Gbps at even long distances, by using 8 MIMO antennas and channels as wide as 160 MHz [1], [2], [3], [4].

However, unfortunately, the conventional 802.11 DCF[1] running at the MAC layer, causes the channel utilization[2] to drop drastically at such high PHY data rates. This is proven analytically, experimentally and via simulations [4], [5], [3], [6], [7]. For example, in [3] it is shown that when we shift from 1Mbps to 1Gbps PHY data rate, the 802.11 DCF causes the channel utilization to drop from 75% to as low as 6%. This undesired phenomenon arises, because the entire channel is treated as a single entity and 802.11 DCF performs channel contention in the time domain. The time-domain contention usually incurs channel idle time before every packet transmission, due to nodes being in their backoff periods. As we shift to higher data rates, this channel idle time due to contention remains unchanged, while the packets take a proportionately smaller transmission time. Thus, clearly, the 802.11 channel contention overheads become substantial at high PHY data rates, which leads to poor channel utilization [4]. Hence, it becomes important to develop new random access protocols for high data rate WLANs, that will provide a better usage of the underlying channel.

For this purpose, one work that is recently proposed is the *Fine-Grained Channel Access (FICA)* technique [3]. FICA attempts to improve the channel utilization by using two main ideas: (1) performing contention and backoff on the *frequency-domain*[3] instead of the time-domain and (2) dividing the wide channel into smaller subchannels of equal and fixed width, and allowing packet transmissions by different nodes on different subchannels, simultaneously. (Hence, the term *Fine-grained* channel access.)

On each of the subchannels we will have a proportionately slower data rate, than the data rate supported on the entire wide channel. Hence, the same packet will have a proportionately longer transmission time on a subchannel, than when it is transmitted on the entire wide channel. Every node can contend for and access any number of subchannels. Hence, in essence, FICA causes relatively short frequency-domain contention periods to be followed by long periods of data transmissions on the subchannels. Clearly, this approach should be effective in reducing the impact of contention overheads and improving channel utilization in high data rate WLANs.

In [3], it is also argued that FICA is more promising and practical than even 802.11 with frame aggregation[4] [8], [1] enabled. The effectiveness of 802.11 frame aggregation reduces, as we shift to higher data rates. This is because it is usually not possible for each sender to individually have enough packets of suitable sizes to aggregate, in order to enhance the overall efficiency. Also, the presence of delay-sensitive packets (e.g., VoIP packets) makes the frame aggregation scheme unsuitable [3], [7], [4]. In contrast, FICA allows in a sense, an "aggregation" of packets across different senders, after the shared frequency-domain contention period, and hence, is much more practical in enhancing channel utilization.

[1]We assume that the readers are familiar with the 802.11 DCF MAC protocol, and RTS/CTS and NAV concept in the 802.11 standard.

[2]*Channel Utilization* is defined as the ratio of the network throughput achieved to the physical layer data rate. We use the terms *Channel Utilization* and *Efficiency* interchangeably.

[3]In frequency-domain contention, nodes compete for the medium by sending signals on randomly chosen OFDMA subcarriers.

[4]This option of 802.11 allows the same sender to send multiple MAC packets back-to-back, after winning the channel, in order to improve efficiency.

978-1-4673-1544-9/12/$31.00 ©2012 IEEE

While the FICA approach appears appealing for improving efficiency in high data rate WLANs, the scenarios used in previous literatures [3], for evaluating the performance of FICA, are very limited. Hence, the **goal** of this paper is to study the FICA MAC protocol thoroughly, with realistic traffic scenarios and network topologies. Since in real-world settings, senders typically have packets of different sizes at the MAC layer [7] [9], we find it important to also analyze FICA's performance under such settings.

To the best of our knowledge, this is the first work that identifies the problems that can arise with the FICA MAC protocol, when packets of *different* sizes are present in the network. We refer to these problems as *deafness*[5], *muteness* and a certain form of *hidden terminal problem*. Our careful analysis and simulation results show that these problems can be sufficiently serious to cancel out the advantages of fine-grained channel access. In fact, we show that, these problems can cause FICA to perform worse than the plain 802.11 DCF without packet aggregation, in terms of channel utilization, per-user throughput or fairness. We also show that in some occasions, where FICA provides high efficiency, it does so, at the cost of starving several nodes in the network.

Our above findings motivate the need to address the problems that arise with the FICA MAC protocol, in order to truly benefit from frequency-domain contention and fine-grained channel access. The insights obtained in this paper, led us to the development of a new MAC protocol, btFICA, for high data rate WLANs. Our btFICA protocol preserves the positive features of the FICA scheme, while solving its issues. We have described the details of btFICA in a companion paper [12] appearing elsewhere in this proceedings.

II. Description of the FICA Scheme

FICA defines both a new PHY and MAC scheme for high data rate WLANs. The FICA MAC protocol is a carrier sensing based, random access scheme. The FICA MAC protocol is designed for WLANs where each AP and each client is equipped with a single *half-duplex* radio [3]. The radios are capable of operating on very wide channels, and can support data transmission/reception at very high rates. If a sender has packets to send, the sender has to pass through two phases: (1) *contention phase* and (2) *DATA/ACK phase*. We further briefly describe the FICA PHY/MAC scheme below:

A. The FICA Physical Layer Architecture

At the PHY layer, FICA makes use of the OFDMA [13] technology. In OFDMA, the wide channel is divided into many narrow-band orthogonal smaller channels, called *subcarriers*. In simple words, the FICA PHY layer is designed to just provide the following capabilities: (1) Allow the frequency-domain contention to take place, while preserving orthogonality amongst subcarriers.(2) Allow the subcarriers to be grouped into subchannels, and allow different nodes to transmit frames on different subchannels, simultaneously, while preserving orthogonality amongst subchannels. The receiver can listen on the entire wide channel and can receive frames arriving on different subchannels, simultaneously.

Now, during the DATA/ACK phase, the nodes involved divide the entire wide channel into M subcarriers. After accounting for guardband on the two ends of the wide channel, the remaining set of subcarriers is partitioned into smaller groups of K subcarriers, called *subchannels.* The nodes will treat each of these subchannels as a unit, i.e., senders will be transmitting each frame on a subchannel.

Now, during the *contention phase* at a node, the FICA PHY divides the entire wide channel into 2M subcarriers. After accounting for the same amount of guardband, half of the remaining subcarriers here are used to represent all the subcarriers of all of the subchannels of the the DATA/ACK phase. The other half is used to send control information, such as, NAV, etc. For more details in regards to the FICA PHY layer, we refer interested readers to [3].

B. The FICA MAC Protocol

The FICA MAC protocol allows different senders to share the medium. We describe the main parts of the MAC protocol below:

1) Frequency-domain Contention: Briefly, a sender, first, begins to carrier sense on the entire wide channel for a certain specified period of time[6]. If it finds the entire wide channel idle during this entire time, it contends for randomly chosen subchannels, by immediately sending a special symbol called M-RTS. After an SIFS period, the sender expects to hear back another symbol called M-CTS, that contains the winner information for each subchannel. Note that the M-RTS/M-CTS handshake uses the entire channel. Until this point we have the contention phase taking place. Now, if the sender wins on any of the subchannels for which it contended, then after an SIFS period, the sender will begin its DATA/ACK phase, and will send its data packets on all the wun subchannels. After it finishes sending all its data packets, it waits to receive ACKs on the corresponding subchannels. If a sender does not receive an ACK back on any of its subchannels after an SIFS duration, then it will assume that a collision has occured on that subchannel. A receiver will generate an ACK packet for each subchannel on which it received an intended data frame successfully. The ACKs are sent by the receiver, after an SIFS period, after receiving all intended data frames arriving on different subchannels.

Note that an *M-RTS symbol* just consists of a set of tones sent on subcarriers. In order to contend for a subchannel, a sender randomly selects one of the K subcarriers that represent this subchannel, and sends a tone on it. Now, if we have multiple senders contending, then all of them will simultaneously send out their M-RTS symbols.

[5] We have identified a new form of deafness which is different from the type of deafness that is previously discovered in the contexts of directional antenna networks [10] and multichannel networks described in [11].

[6] This period is DIFS for clients and Long-DIFS or Short-DIFS for APs [3].

The nodes that hear an M-RTS symbol, can in reality be hearing a combination of individually transmitted M-RTS symbols. From these nodes, only the nodes that are indicated as *potential receivers*[7] will be involved in resolving contention. Each potential receiver selects the highest active subcarrier that it sees on each subchannel as the *winning subcarrier* on that subchannel. All potential receivers will simultaneously broadcast their contention results in their M-CTS symbols. Upon hearing the arriving M-CTS symbol, if the sender finds its randomly chosen subcarrier selected for that subchannel, then the sender will consider itself a winner on that subchannel.

2) Frequency-Domain Backoff: Every node, i, also maintains a local *Contention Window* variable, CW_i. CW_i represents the maximum number of subchannels for which the node i is allowed to contend. Initially, CW_i is set to the total number of subchannels in the network. In [3], the authors provide two different algorithms, for updating CW_i, namely, $RMAX$ and $AIMD$[8]. $AIMD$ stands for the Additive Increase and Multiplicative Decrease strategy. Here, if a sender i receives ACKs for all the packets that it had transmitted, then, it increases CW_i by 1 (Additive Increase). However, if the sender concludes a collision on $p\%$ of the subchannels on which it had transmitted data packets, then the sender will reduce CW_i by $p\%$ (Multiplicative Decrease). The AIMD algorithm is said to perform *frequency-domain backoff*.

3) Network Allocation Vector (NAV) Band: In the M-RTS and M-CTS symbols, there exists a reserved set of subcarriers, called the NAV Band. This is used for reducing hidden terminal problems that can cause collisions at receivers. Each subcarrier in the NAV band stands for a specific data packet transmission time. Senders when sending an M-RTS, will also specify the longest data transmission time that they might require by sending a tone on the corresponding subcarrier in the NAV band. Each potential receiver will then echo back the longest transmission time it hears, in the M-CTS symbol which it sends. The nodes receiving an M-CTS, will use the information in the NAV band to defer contention for the longest time needed.

III. Performance Issues with the FICA MAC scheme

Our studies show that the FICA MAC protocol performs well, in terms of channel utilization and fairness, when all nodes in the network have packets of the same size to transmit. However, in real-world settings, senders usually have packets of different sizes at the MAC layer, depending upon upper layer protocols and applications in operation. For example, in [7], the authors collected packet traces in a real-world WLAN deployment at Duke University. Their study shows that the frame sizes generated by Skype, Web browsing and HD streaming sessions were on average, 511, 1063 and 1424 bytes, respectively. Also, we can see from the packet traces collected in the Sigcomm 2008 conference [9], that senders in WLANS can usually have frames of different sizes at the MAC layer. Thus, having a MAC protocol that provides high channel utilization and a good level of fairness amongst flows, even when frames of different sizes are present in the network, becomes important.

[7] *Potential receivers* are those nodes for whom the senders are contending.
[8] In this paper we focus on FICA's $AIMD$ scheme , which is shown to be the better of the two schemes for FICA [3].

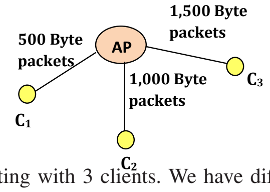


Fig. 1. Single cell setting with 3 clients. We have different packet sizes for each link.

However, we find that three new problems can arise with the FICA MAC scheme, when packets of different sizes are present in the network. We call these problems (1) *Deafness* (2) *Muteness* and (3) a certain form of *hidden terminal problem*. As we show in Section IV, these problems can degrade FICA's performance drastically. We explain each of these issues in detail below:

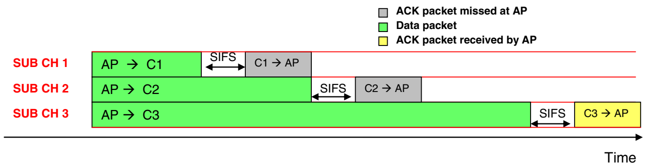
A. The Deafness Problem

We say that *deafness* occurs when a sender finishes successful transmissions to some of its receivers on some of the subchannels, but *cannot hear the ACKs* intended back for it, because it is still busy transmitting to receivers on other subchannels. Here, we say that the *sender* is *deaf* to the ACKs coming from its receivers. The deafness problem can *reduce channel utilization* significantly, as shown by the following example:

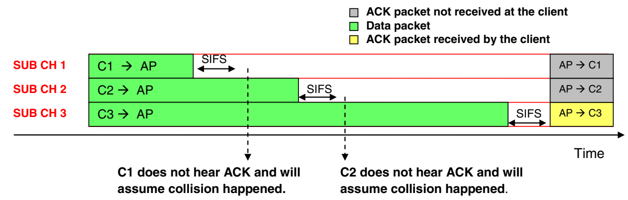
Let us consider the scenario of a single cell with one AP and 3 clients as shown in figure 1. For simplicity, let us assume only downlink traffic. For clients c_1, c_2 and c_3, the AP always has packets of sizes 500, 1000 and 1500 bytes, respectively, to send. It is clear that after any M-RTS/M-CTS handshake, the AP always wins on all the subchannels that it had chosen randomly. To give a fair channel access opportunity for all flows, the AP serves the data packets for each client in a round robin fashion. Also, it is clear that we will not have any collisions occuring on any of the subchannels.

Let us assume that the AP has started transmitting data to all its clients, as shown in figure 2(a). Now, c_1 will be the first to finish receiving all its intended data packets, arriving on different subchannels, because c_1's packets are smaller than the packets of all other clients. After SIFS, c_1 transmits ACKs on its respective subchannels, however, the AP is deaf to these ACKs, because it is busy transmitting to c_2 and c_3 (figure 2(a)). Similarly, when c_2 finishes receiving its data packets and transmits its ACKs, the AP is again deaf to these ACKs also, because, it is busy transmitting to c_3 (figure 2(a)).

After the AP finishes all its transmissions, the AP switches to receive mode and waits for ACKs on all the subchannels it used. However, it will only hear the ACKs coming from



(a) Illustration of the deafness problem. Activities on only 3 of the subchannels are shown.



(b) Illustration of the muteness problem. Activities on only 3 of the subchannels are shown.

Fig. 2. FICA MAC operations for the single cell setting of Figure 1.

c_3.Hence, despite that c_1 and c_2 received their packets successfully, the AP will incorrectly conclude that collisions occurred at these clients.

This has adverse effects, because, (1) the AP will incorrectly assume that there is high congestion in the network and thus, it will needlessly reduce its CW in the same way as described in Section II-B2. Thus, the AP will contend for a lesser number of subchannels in the next round, leaving several subchannels, (on which otherwise successful transmissions could have taken place), empty. (2) The AP will cause further channel wastage by retransmitting already successful packets for c_1 and c_2, on the already limited number of subchannels in the next round.

Now, deafness again repeats in future rounds, which will eventually shrink the CW of the AP to very small values. This will cause the AP to access only a very small number of subchannels (1 or 2 out of a 128, in our analysis), during data transmission rounds, despite that it has high demands. Additionally, we also find that the deafness problem also causes many of the already successful packets for c_1 and c_2, to be needlessly retransmitted multiple times. This example clearly shows how deafness can lead to a very inefficient usage of the channel.

One can argue that deafness would not have happened if the AP had picked and transmitted packets of the same size in each transmission round. However, this may not be an effective approach, because the AP might not have enough packets of the same size to send on all the available subchannels. Thus, again we can still have unused subchannels which will cause the efficiency to drop. Also, some packets can be delay-sensitive and hence, should be transmitted immediately.

Note that, deafness can also occur at the client, when the client has packets of different sizes to transmit to the AP. However, this case is rare. This can only happen when all the largest packets sent by the client face a collision at the AP, but AP finishes receiving all its other intended data packets, while the client is still busy transmitting its largest frames.

B. The Muteness Problem

Muteness occurs when a *client* finishes successful packet transmission(s) to the AP, and waits to receive ACK(s) from the AP after an SIFS period, however, the AP *cannot transmit back ACKs* during this time, because it is busy *receiving* from its other clients on other subchannels. We say that, here, the AP is *mute* for this client. When the client does not receive the ACK(s), after an SIFS period, the client incorrectly assumes that collision(s) have occured on its respective subchannels. The muteness problem can cause *starvation of nodes* and *unfairness*, as shown by the following example:

We now again consider the same scenario as shown in figure 1, however, this time with only uplink traffic. c_1, c_2 and c_3 can hear each other and they always have packets of sizes 500, 1000 and 1,500 bytes, respectively, to transmit to the AP. Initially, the CW of all the clients is set to the total number of subchannels. To avoid distracting details during explanation, let us also assume that during the M-RTS/M-CTS handshake no two clients win on the same subchannel and hence, we have no collisions on any subchannels. All the clients start their data transmissions at the same time on different subchannels.

Since there is a single half duplex radio on the AP, it has to finish receiving on all the subchannels, before it can send back ACKs on these subchannels. c_1's packets are the smallest in size (500 bytes), hence, c_1 finishes its data transmissions first, and waits to hear its ACKs. However, the AP is mute for c_1 because it is still busy receiving data from c_2 and c_3 (figure 2(b)), which are sending packets of larger sizes than that of c_1. Since, c_1 does not receive an ACK after an SIFS period, on any of its subchannels, c_1 incorrectly concludes that collisions occured on all its subchannels, and thus incorrectly suspecting high congestion, c_1 aggresively reduces its CW to 1. Similarly, c_2 will also face a mute AP, after it finishes sending its data packets, and thus, will needlessly reduce its CW to 1. Now, when the AP finishes receiving from c_3, the AP switches to transmit mode and transmits ACKs for all the packets that were successfully decoded, including c_1's and c_2's packets(figure 2(b)). However, these ACKs cannot be decoded by c_1 and c_2.[9] Only c_3 can correctly decode its ACKs on all its subchannels, and hence, unlike c_1 and c_2, c_3 will keep its CW at the maximum size.

Thus, clearly, in the next round, each of c_1 and c_2 will contend for only 1 subchannel, but c_3 will contend for all the subchannels and will win on almost all of them. Moreover, if c_1 and c_2 win the contention and start their data transmissions, then unlike c_3, they will waste the channel resources by retransmitting already successfully received packets. Furthermore, for c_1 and c_2, the muteness problem occurs repeatedly, which will not only cause the CW for the two clients to remain only one, but will also cause multiple retransmissions of the same successful packets in future rounds. On the otherhand,

[9]This is because c_1 and c_2 by now have switched to an FFT size that is twice the IFFT size of the AP, in order to hear new M-RTS/M-CTSs.

c_3 will again receive ACKs back on all its subchannels and hence, will maintain a large CW.

Hence, it is clear that the clients c_1 and c_2 will starve, until c_3 finishes its transmissions. Note that the channel utilization is enhanced in this example, however, at the cost of starving the clients with smaller packet sizes. This example clearly shows how muteness can cause unfairness in the network.

Note that, one way of solving the deafness and muteness problems, and allowing senders to be correctly informed of their successful transmissions, might be to replace the half-duplex FICA radio by a full-duplex radio. Now, the nodes can decode (transmit) ACKs arriving on some subchannels, while transmitting (receiving) data packets on other subchannels. However, full-duplex radios are not only costly, but also they cannot solve the hidden terminal problem described in Section III-C, below.

C. The Hidden Terminal Problem

The hidden terminal problem that we are focusing on, in this Section, is different than the hidden terminal problem that causes a collision at a receiver when it is receiving its intended data packets. The NAV concept in FICA is sufficient to reduce this form of hidden terminal problems. In this Section we are dealing with the *hidden terminal problems that can cause collision(s) at the sender when it is busy receiving its intended ACK(s).* With FICA, this form of hidden terminal problem can occur consistently, and thus if not addressed, can lead to *poor network thoughput* and *starvation*.

We further explain this via the following example. Let us consider the scenario shown in figure 3. Such topologies are also observed in real-world deployments [14]. For the sake of simplicity, let us assume only downlink traffic. AP_1 and AP_2 always have packets of sizes 500 bytes and 1500 bytes, respectively, to transmit to their clients. Initially, each AP has a CW of the maximum size. In the beginning, AP_1 and AP_2 listen on the entire wide channel for a period of Long-DIFS, and they both transmit their M-RTSs simultaneously. Clearly, both of them will win on all the subchannels after the M-RTS/M-CTS handshake, and they will both begin their transmissions.

All of these transmissions arrive at the respective clients successfully. AP_1 finishes its transmissions first because it has packets of smaller sizes than AP_2. However, the ACKs arriving back to AP_1 will collide with AP_2's on going transmissions. Hence, AP_1 will incorrectly conclude collisions on all subchannels, and will reduce its CW to 1, and will needlessly retransmit all of these successful packets in the future rounds. Now, AP_1 will find the channel busy for the entire time that AP_2 is transmitting. AP_1 starts seeing a clear channel after AP_2 finishes its transmissions. When c_2 begins sending its ACKs after an SIFS period, these ACKs cannot be heard at AP_1 because AP_1 and c_2 are hidden terminals. Hence, AP_1 continues to sense the spectrum idle for a Long-DIFS period and *starts its M-RTS transmission, which causes collisions with the ACK(s) arriving at* AP_2.This

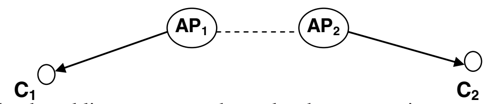


Fig. 3. The dotted lines represent the nodes that can carrier sense each other. Solid lines represent client-AP associations.

will cause AP_2 to also incorrectly conclude collisions on all its subchannels. Thus, AP_2 will also needlessly reduce its CW to 1 and will wastefully retransmit already successful packets in future rounds.

We can see that the above hidden terminal problem can easily occur, by observing the PHY parameters used in FICA [3]. In FICA, we have $SIFS = 16\mu$sec; $slottime = 9\mu$sec; $DIFS = SIFS + 2 * slottime = 34\mu$sec; Long-DIFS $= DIFS + slottime = 43\mu$sec; and the smallest preamble time is $preambletime = 46.8\mu$sec. Hence, when AP_2 finishes its data transmissions, AP_1 will start its M-RTS transmission, after 43μ seconds. This clearly causes collisions at AP_2 which is still receiving the preamble of the ACK packets from c_2.

Note that, the main purpose of having SIFS periods between entities involved in a dialogue[10], and a form of DIFS period before a new dialogue can begin is to give all the entities, including ACKs, involved with the on-going dialogue priority in transmission. While this proves effective in allowing an ongoing dialogue to complete before a new one could begin in single collision domain networks, where all nodes can hear each other, this is not the case in networks where we have hidden terminals.

Now, because AP_1 started its idle Long-DIFS interval earlier than AP_2, it so happens that for the next round of transmissions, AP_1 manages to finish its contention phase earlier than AP_2, and begins its data transmission. AP_2 upon seeing the channel busy defers from contending. Moreover, AP_1 not only accesses only 1 of the many subchannels present, but also wastes it, by retransmitting an already successful packet. Note that, the same form of hidden terminal problem occurs again, because now AP_2's M-RTS will collide with the ACKs arriving at AP_1.

This problem occurs consistently, which will lead to both APs maintaining a CW of 1, and both APs performing unneeded retransmissions.Furthermore, AP_1 and AP_2 cannot share the entire wide channel at the same time, i.e., AP_1 and AP_2 take rounds in accessing a small portion of the wide channel, which causes an inefficient usage of the channel. Clearly, this causes the overall channel utilization to drop, as well as, starvation of both the APs. Hence, it becomes important to address this problem of FICA.

Recall that, in 802.11 DCF, this form of hidden terminal problem that can cause collisions at the sender is addressed by using RTS packets and the NAV concept. Here, the nodes receiving the RTS also update their NAV accordingly. Note that, in 802.11 DCF, even if RTS/CTS is not used, then this type of hidden terminal problem may not have severe effects.

[10]By a *dialogue* we mean the *M-RTS/M-CTS/DATA Packets/ACKs* exchange.

This is because, unlike FICA, in 802.11 DCF, defering nodes do not immediately transmit after hearing the channel idle for a DIFS period. They go into a random time-domain backoff, which reduces the chances of such collisions at senders.

Note that in [3] nothing is mentioned about whether, with FICA, the nodes in the vicinity of the sender that *receive the M-RTS*, use the NAV band information here to defer their contention, or not. However, even if such NAV information is not ignored in FICA, then this will still not prevent the consistent hidden terminal problem of this example and of similar scenarios. This is because, here, the APs can't even receive each other's M-RTS, because at each AP, the arriving M-RTS constantly gets collided with the ACK reception here. Moreover, when two neighboring senders simultaneously transmit their M-RTS, they will miss each other's M-RTSs thus, will not be able to update their NAV accordingly in order to prevent such hidden terminal problems.

Also, it can be argued that, this form of hidden terminal issue with FICA would not have occured if, instead of giving each AP the entire wide channel to operate on, we had divided the spectrum amongst the APs, and then used the FICA scheme for each of the cells. However, we argue that one of the implicit benefits of the FICA scheme, when all nodes operate on the same entire wide channel, is that FICA can adaptively and efficiently distribute the available wide channel amongst flows, based upon the traffic demands of the flows, in a completely distributed fashion. (Note that an intelligent spectrum assignment to links can enhance network throughput [15] [16].) When a node's traffic demand changes or when interference levels change in the network, FICA can quickly adapt the spectrum distribution amongst links. This is unlike previous channel assignment schemes [15] [16], that can incur relatively large adaptation overheads when interference levels and traffic demands change in the network.

Hence, we find it important to address such hidden terminal problems that arise with FICA, while allowing all nodes in the network to be assigned the entire wide channel.

IV. Performance Evaluation

We now evaluate and compare the performance of the FICA MAC protocol against the conventional 802.11 DCF, under a variety of network topologies and traffic scenarios.

A. Simulation Methodology

We have implemented a detailed event-based simulator for both the FICA MAC protocol and the 802.11 DCF. Our simulators also carefully capture the details of the FICA PHY layer, such as the CP lengths, subcarrier widths, etc. Note that the 802.11 standard has a different PHY layer than that of FICA, which will naturally cause a slight mismatch between the PHY data rates of the two schemes on a specified channel [3]. However, since our goal is to isolate and compare the benefits that can be provided by the two *MAC* protocols over the *same* high PHY data rate, we find it important to maintain the same PHY characteristics, (FICA PHY), for both the MAC schemes. Note that, for 802.11 DCF, every node always treats the entire wide channel as a single entity.

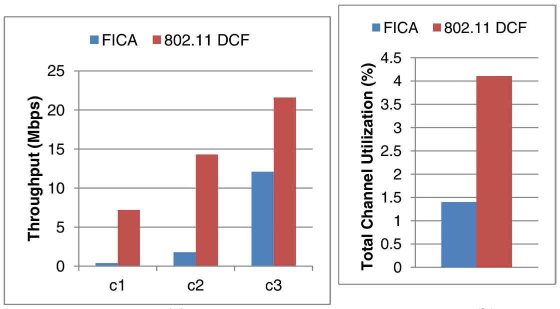
We have a 160 MHz wide channel. We use the QPSK modulation with 1/2 coding rate on each subcarrier, with 8 MIMO antennas, to give us a PHY data rate of 1.05 Gbps on the entire wide channel. For FICA we use 16 data subcarriers per subchannel as in [3], which gives us a total of 128 subchannels. Also in FICA PHY, we have constant transmit power per active subcarrier across all nodes. Hence, unlike 802.11 PHY [15], [17], the interference/transmission range of nodes do not change even if they access portions of the wide channel. We have used the free-space path loss model and we have an interference range and transmission range of 50m and 45m, respectively, for each node. We use the same timing parameters as in [3]. To make conditions favorable for the FICA MAC protocol, we have used the $AIMD$ backoff scheme. To give a fair chance to 802.11 DCF and enhance its efficiency, we have turned off RTS/CTS. This is the preferred option to take for 802.11 DCF in high data rate networks, because now packets take a smaller transmission time [18]. The maximum number of restransmissions for a packet is 7.

B. Simulation Results - Sample Networks

In order to quantify the isolated impact of each of the three problems described in Section III, on the performance of the the FICA MAC protocol, in this Section, we use sample scenarios in which only the problem under consideration exists and the other two problems do not occur. For completeness, we later on also show results for single-cell and multi-cell *random* networks.

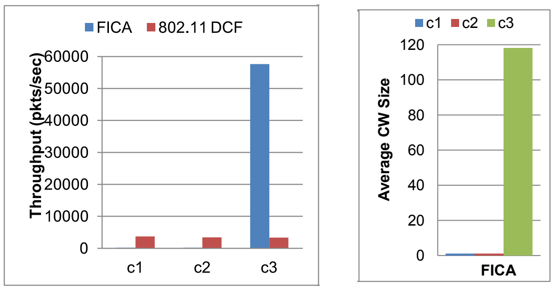
1) Quantifying the Impact of Deafness: We consider the exact scenario described in Section III-A. Here, it is clear that only the deafness problem occurs with the FICA MAC protocol. From figure 4(b) we can see that, this deafness problem of the FICA MAC protocol is detrimental enough to cause the channel utilization to drop to as low as **1.4%**. Even the worst case of 802.11 DCF, i.e., 802.11 DCF without packet aggregation, provides a **3** times higher channel utilization than FICA. This is because, unlike FICA, 802.11 DCF does not waste the channel with needless retransmissions. Moreover, with 802.11 DCF, the AP keeps its time-domain CW to the minimum value (16) and when it transmits a packet, it makes good use of the entire channel.

In figure 4(a), we show the throughput achieved for every downlink flow. We can see that, with FICA every flow not only achieves a lower throughput than that of 802.11 DCF, but also the flow towards c_1, that contains the smallest sized packets, starves. This is because for the flow towards c_1, deafness occurs more often, leading to more needless retransmissions of c_1's packets. In contrast, with 802.11 DCF we are achieving better per-flow-throughput, and no starvation of flows. This is because, here, not only is every transmission of the AP successfully received at the intended client, but also, the ACK coming back to AP is successfully received. This causes the AP to always serve the next *new* packet for each client upon



(a) (b)

Fig. 4. Results for the scenario described in Section III-A. The impact of deafness on the performance of the FICA MAC protocol is shown.



(a) (b)

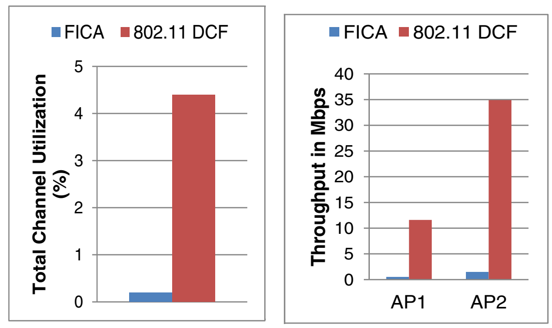
Fig. 5. Results for the scenario described in Section III-B. The impact of muteness on the performance of the FICA MAC protocol is shown.

the client's turn, hence, enhancing the per-user-throughput.

2) Quantifying the Impact of Muteness: Now we consider the scenario described in Section III-B, in which FICA only faces the muteness problem. It is clear from figure 5(a), that because of the muteness problem of FICA, the clients with smaller packet sizes, i.e., c_1 and c_2, severely starve, but, only one client that has the largest packet size, is given a very high throughput. Clearly in this scenario, FICA performs well in terms of channel utilization, but at the cost of *starving* all but one client in the network. An ideal MAC protocol should avoid starvations and provide a fair but as high as possible throughput to all clients. We can see that 802.11 DCF performs better than FICA in avoiding starvations, and providing fairer throughput for all the clients. However 802.11 achieves a low network throughput and hence, low channel utilization. In figure 5(b), we show finer results for FICA. Clearly, with FICA the average CW size for c_3 is significantly larger than that of c_1 and c_2, which shows that on average, c_3 accesses almost all subchannels, but c_2 and c_1 access only one subchannel. This shows the unfairness that arises with the FICA MAC protocol due to the muteness problem.

3) Quantifying the Impact of the Hidden Terminal Problem: Now, we focus on the scenario described in Section III-C. Here, the deafness and muteness problems are not arising, but the hidden terminal problem, described in Section III-C, is occurring with FICA. It is clear from figure 6(a), that the impact of the hidden terminal problem is severe enough to cause the efficiency of FICA to drop **very close to 0**. We have already explained the reasons for why we get such results in Section III-C. We see that 802.11 DCF, even without RTS/CTS, can still give a channel utilization of 4%, which shows that this form of hidden terminal problem does not affect 802.11 performance as severly as it does FICA's.

In figure 6(b), we plot the per-AP-throughput that is achieved with both the MAC schemes. As we expect, we can see that with FICA both APs are starving, despite that they have high demands. However, we are achieving better per-AP throughput with 802.11 DCF. The two APs starve with FICA, because, the consistent occurance of the hidden terminal problem, not only causes many needless retransmissions, but also it forces each AP to access only 1 subchannel, from the 128 free subchannels, in each transmission round.



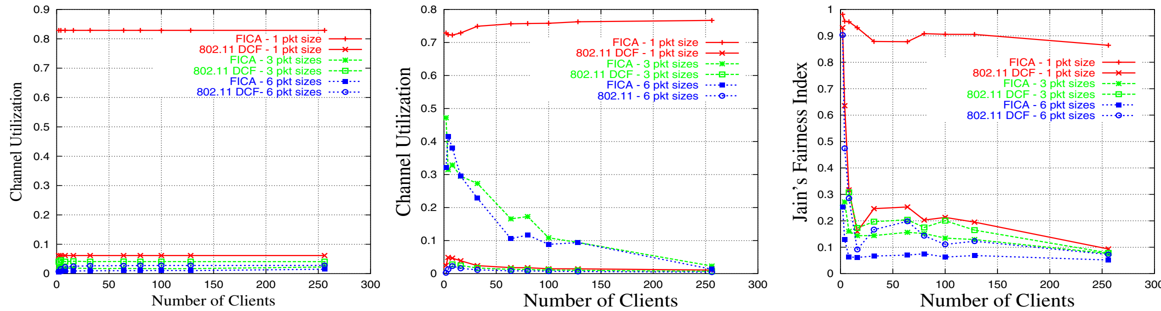
(a) (b)

Fig. 6. Results for the scenario described in Section III-C. Impact of hidden terminal problem on the performance of FICA is shown.

C. Simulation Results - Single Cell Random Networks

We now consider random network topologies and random traffic settings. In this Section we have one AP and we place clients on random locations that are within the the AP's transmission range. We consider cases with one packet size, of 1500 bytes, in the entire network, as well as cases with 3 , 6 and 12 different packet sizes in the network, respectively. All packet sizes are in the range of 100 bytes to 1500 bytes. We assume that the AP is always backlogged, which means that the AP always has packets to send to all its clients. For the scenarios where uplink traffic is present, we assume that every client is backlogged. We find it important to evaluate performance under such settings, because, it tells us how well the MAC protocols make use of the available channel, and how well do they serve all the flows, when there is a high need for the available bandwidth.

In figure 7, for every link in the network, we randomly choose a *fixed* packet size from the set of packets sizes for the network. Now, in figure 7(a), we only have downlink traffic in the network, and we plot the efficiency for both MAC protocols, over varying number of clients. We see that when we have one packet size for all flows, FICA provides an efficiency of 83%, which is a significant improvement over 802.11 DCF. However, when we have different packet sizes involved in the network, the efficiency for FICA drops drastically, due to the deafness problem. For example, clearly with as little as 3 different packet sizes in the network, FICA's channel utilization drops to approximately 1%, which is worse than the 6% channel utilization that we have with 802.11 DCF. Similar results hold when we increase the different number of

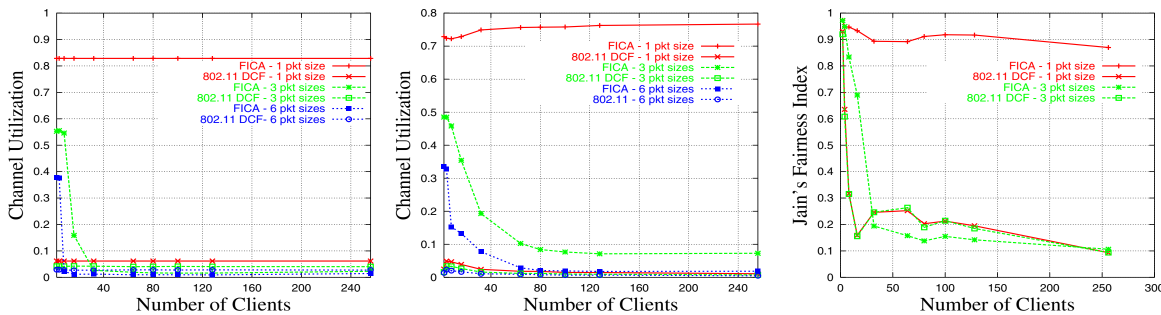


(a) Only downlink traffic is present.

(b) Both Downlink and Uplink traffic is present.

(c) Jain's fairness Index for when both Uplinks and Downlinks are active.

Fig. 7. Results for one AP and randomly placed clients. For every flow we randomly choose a *fixed* packet size.



(a) Only downlink traffic is present.

(b) Both Downlink and Uplink traffic is present.

(c) Jain's fairness Index for when both Uplinks and Downlinks are active.

Fig. 8. Results for one AP and randomly placed clients. For each flow we have *variable* packet sizes that are randomly chosen.

packet sizes in the network to 6 and 12, respectively. Note that we are not showing our results for 12 packet sizes in the network, in order to avoid cluttering of the figures.

In figures 7(b) and 7(c), we use the same topologies as in figure 7(a), however, now we have both uplink and downlink traffic. The packet size for each downlink and each uplink is randomly chosen and fixed. In figure 7(b) we again plot the efficiency over varying number of clients. We observe that when we have different packet sizes, FICA still performs better in efficiency than 802.11 DCF. However, our careful analysis shows that FICA gives such improvement, at the cost of *starving* uplink flows that have smaller packet sizes, but allowing the uplink flows that have the largest packet size to get an unfair larger share of the available channel. We can also see that for a given number of packet sizes, as we increase the number of clients, FICA's efficiency starts to drop, because now we have more clients with smaller packet sizes winning a subchannel in a round, thus causing more subchannels to be wasted due to the muteness problem, during uplink data frame transmissions. Moreover, the deafness problem also reduces the efficiency of FICA.

In figure 7(c), we use the Jain's Fairness Index [19] to compute the level of throughput-fairness amongst all flows in the network, that is achieved with the two MAC protocols. Here, a value close to 1 indicates a high level of fairness.Note that, since we are dealing with a single cell network, with all backlogged flows, an ideal fair MAC protocol should provide all flows equal throughput. We see that FICA, with one packet size in the network, performs very good interms of fairness. However, as we expect, FICA's throughput-fairness amongst flows drops drastically, when we have different packet sizes in the network. For example, for the case of 32 clients and 6 packet sizes in the network, we can see that FICA's fairness index drops to as low as 0.08. Even 802.11 DCF performs slightly better interms of fairness, in these scenarios. Note that, here we are considering single cell *random* networks, in which hidden terminal scenarios can naturally arise. However, we have noticed that for single collision domain networks, 802.11 DCF can provide even better fairness results.

The results in figure 7 were for the case, where the sender for each link had packets of a *fixed* size to transmit. However, in real-world settings a flow in the network can have packets of *different* sizes. Thus, in figure 8, we evalute both schemes under traffic settings, where for every flow we have packets of variable sizes. The packet sizes are chosen randomly from the set of packet sizes for the network. We see that, in these cases also, the deafness and the muteness problems are sufficient to degrade the performance of the FICA MAC protocol.

D. Simulation Results - Multi-Cell Random Networks

We have also evaluated both the MAC protocols in a wide variety of multiple AP settings and traffic scenarios, and we show some of our results here. Note that in [3], FICA has not been evaluated in multiple AP settings with multiple collision domains. In figure 9, we have 6 APs randomly deployed within a 200m x 200m area, and we change the number of randomly placed clients in the network, and plot the efficiency. Each client is associated with that AP from which it had the

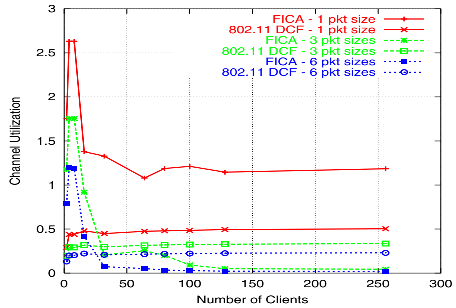


Fig. 9. Results for WLAN with 6 APs deployed in a 200m x 200m area.

strongest received signal. We assume only downlink traffic and that the AP is always backlogged. Each flow can contain different packet sizes that are chosen randomly, from the set of packet sizes for the network. We can see that FICA gives very high efficiency (greater than 100%), when we have only one packet size in the network. We can achieve an efficiency of greater than 100%, because in multiple collision domain networks, non-interfering links can *reuse* any portion of the entire wide channel. This is a significant improvement over what can be achieved with 802.11 DCF. However, as we expect, we can see that, with packets of different sizes present in the network, the efficiency of FICA reduces to low values, as we increase the number of clients. Note that, when we have small number of clients in the network then FICA still achieves high channel utilization. For example, when we have 6 packet sizes in the network and 8 clients, then FICA achieves an efficiency of approximately 125%. This is because in these scenarios, on average, the APs will usually have only one client associated to it, and thus, non of the issues arise over here. However, as we increase the number of clients in the network, we find that the problems faced by FICA causes FICA's efficiency to drop.

V. Conclusion

In this paper, we have extensively studied the FICA MAC protocol, which is one of the leading schemes designed for the purpose of improving efficieny in high data rate WLANs. We have identified, for the first time, the problems that can easily arise with the FICA MAC protocol, when packets of different sizes are present in the network. We have quantified the impact of these problems on the performance of FICA via extensive simulations. We showed that these problems can severely degrade channel utilization and fairness in the network, if left unaddressed.

The insights achieved in this paper motivated us to the develop a new MAC protocol, btFICA, for high data rate WLANs, that is based upon the FICA framework. We describe btFICA in a companion paper [12] appearing elsewhere in this proceedings. btFICA effectively and comprehensively addresses all the three problems that arise with the FICA MAC protocol, while maintaining the positive features of the original FICA scheme.

References

[1] "IEEE 802.11 TGac," http://mentor.ieee.org/802.11/dcn/10/11-10-1361-03-00ac-proposed-tgac-draft-%amendment.docx.

[2] "802.11ac: Gigabit Wi-Fi Devices will be shipping in 2012," http://www.zdnet.com/blog/networking/80211ac-gigabit-wi-fi-devices-will%-be-shipping-in-2012/1867.

[3] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang, "Fine-grained Channel Access in Wireless LAN," in *Proc. of the SIGCOMM Conference*. ACM, 2010.

[4] E. Magistretti, K. K. Chintalapudi, B. Radunovic, and R. Ramjee, "Wifi-nano: reclaiming wifi efficiency through 800 ns slots," in *Proceedings of the MOBICOM Conference*, 2011.

[5] R. Maheshwari and et. al, "Adaptive channelization for wireless networks," WINGS Lab, Stony Brook University, Tech. Rep., 2009.

[6] V. Shrivastava, S. Rayanchu, J. Yoonj, and S. Banerjee, "802.11n Under the Microscope," in *Proceedings of the 8th ACM SIGCOMM conference on IMC*, 2008.

[7] S. Sen, R. Roy Choudhury, and S. Nelakuditi, "No Time to Countdown: Migrating Backoff to the Frequency Domain," in *Proc. of MOBICOM*, 2011.

[8] "IEEE 802.11n Specification," http://standards.ieee.org/getieee802/download/802.11n-2009.pdf.

[9] "Packet Traces Collected in the Sigcomm 2008 Conference," http://www.cs.umd.edu/projects/wifidelity/sigcomm08_traces/.

[10] R. Choudhury and N. Vaidya, "Deafness: A MAC Problem in Ad Hoc Networks when using Directinal Antennas," in *ICNP*, 2004.

[11] R. Maheshwari, H. Gupta, and S. Das, "Multichannel MAC Protocols for Wireless Networks," in *SECON*, 2005.

[12] F. Zarinni and S. Das, "btFICA MAC Protocol for High Data Rate WLANs," in *Proc. of ICCCN*, 2012.

[13] H. Yin and S. Alamouti, "OFDMA: A Broadband Wireless Access Technology," in *Proc. IEEE Sarnoff Symposium*, 2006.

[14] V. Shrivastava, N. Ahmed, S. Rayanchu, S. Banerjee, S. Keshav, K. Papagiannaki, and A. Mishra, "CENTAUR: realizing the full potential of centralized wlans through a hybrid data path," in *MOBICOM*, 2009.

[15] S. Rayanchu, V. Shrivastava, S. Banerjee, and R. Chandra, "FLUID: Improving Throughputs in Enterprise Wireless LANs Through Flexible Channelization," in *MOBICOM*, 2011.

[16] T. Moscibroda, R. Chandra, Y. Wu, S. Sengupta, P. Bahl, and Y. Yuan, "Load-aware spectrum distribution in wireless LANs," in *Proc. IEEE ICNP Conference*, 2008.

[17] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl, "A Case for Adapting Channel Width in Wireless Networks," *SIGCOMM Comput. Commun. Rev.*

[18] G. Bianchi, "Performance analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communication*, vol. 18, no. 3, pp. 535–547, 2000.

[19] R. Jain, D.-M. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared systems." Eastern Lab, Washington University, Tech. Rep., 1984.