btFICA MAC Protocol for High Data Rate WLANs

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

***Abstract*—In this paper we develop a new MAC protocol for improving network throughput and hence, channel utilization in wireless LANs that can support very high PHY layer data rates (> 1Gbps). We call our new MAC protocol *Busy Tone Assisted Fine-Grained Channel Access (btFICA)*. btFICA is based upon the framework of a prominent state-of-the-art PHY/MAC scheme for high data rate WLANs, called *Fine-Grained Channel Access (FICA)*. While the rationale behind the FICA scheme appears effective for enhancing channel utilization in high data rate WLANs, a recent study shows that problems, such as deafness, muteness and a form of hidden terminal problem, can easily arise with the FICA MAC protocol. These problems can degrade the network performance, if left unaddressed. This motivates us to develop our btFICA MAC protocol that uses an additional busy tone antenna. btFICA comprehensively *solves* all of the three problems faced by the FICA MAC protocol, while *maintaining* the beneficial aspects of the original FICA scheme. Finally, we show via extensive simulations, that btFICA significantly outperforms the original FICA scheme and 802.11 DCF, in different network topologies and traffic scenarios, in terms of channel utilization, per-user-throughput and fairness.**

I. Introduction

The recent advancements in physical layer (PHY) technologies will very soon allow us to have wireless LANs that can support Multi-Gbps PHY data rates. For example, the forthcoming successor of the current 802.11n standard, the 802.11ac standard, is intended to provide PHY 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 analytically that when we shift to 1Gbps PHY data rate, the 802.11 DCF causes the channel utilization to drop as low as 6%. This happens because, as we shift to higher data rates, the same packets now take a proportionately smaller transmission time. However the channel idle time incurred due to nodes backing off, remains unchanged before every packet transmission. Thus, the 802.11 channel contention overheads become substantial at high PHY data rates, which leads to poor channel utilization [4].

Hence, there is a growing interest in the research community to develop new random access protocols for high data rate WLANs, that will provide a better usage of the underlying channel. To this end, several new PHY/MAC schemes have already been proposed [4] [3] [7] [6]. From these schemes, a recent scheme that appears promising and practical for enhancing channel utilization in high data rate WLANs, 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.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. As discussed in [3] and [8], the FICA approach can also be more practical for enhancing channel utilization than the 802.11 DCF with the frame aggregation technique [9].

While the FICA approach can improve efficiency in high data rate WLANs, a recent work [8] has discovered problems that can easily arise with the FICA MAC protocol when packets of different sizes are present in the network. (Note that in real-world settings, senders usually have MAC frames of different sizes to transmit [7], [10].) The problems are called *deafness*, *muteness* and a certain form of *hidden terminal problem*. As discussed in [8], these problems can degrade the network performance drastically, if left unaddressed. Hence, it becomes important to address the problems that arise with the FICA MAC protocol, in order to be able to reap the true benefits of fine-grained channel access.

Motivated by the above insights, the **goal** of this paper is to develop a new MAC protocol for improving efficiency in high data rate WLANs. We call our new MAC protocol, *Busy Tone Assisted Fine-Grained Channel Access (btFICA)*. btFICA is based upon the FICA framework and uses an additional busy tone antenna. btFICA *comprehensively* solves all three of the problems faced by the FICA MAC protocol, while *preserving* the positive features of the FICA scheme.

[1]We assume that the readers are familiar with the 802.11 DCF MAC protocol, the 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 [3].

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

Note that the extra busy tone antenna used by btFICA, should not be viewed as an extra costly radio meant for data packet reception (transmission). A busy tone antenna is a lot simpler. It is used to just detect (emit) energy on different busy tone channels and there are no modulation/demodulation overheads involved [11], [12], [13], [7].

The rest of the paper is organized as follows. In § II and § III, we briefly describe the FICA PHY/MAC scheme, and we revisit the problems faced by the FICA MAC protocol. In § IV we develop our btFICA MAC protocol. In § V we discuss why we did not consider some other potential solutions for addressing FICA's problems. In § VI we present our simulation results for realistic network topologies and traffic scenarios. Related works are in § VII and we finally conclude our paper in § VIII.

II. Description of the FICA Scheme

FICA defines both a new PHY and MAC scheme for high data rate WLANs, where every AP and every 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.

At the *PHY layer*, FICA makes use of the OFDMA [14] technology, which allows the wide channel to be divided into many, narrow-band orthogonal smaller channels, called *subcarriers*. The FICA PHY is designed to provide capabilities that will allow: (1) frequency-domain contention to take place. (2) subcarriers to be grouped into *subchannels*, and fine-grained channel access to take place, while maintaining orthogonality between subchannels. Note that, each receiver can receive frames arriving on all subchannels, simultaneously.

The FICA *MAC protocol* is a carrier sensing based, random access scheme. If a sender has packets to send, the sender has to pass through two phases: (1) *contention phase* and (2) DATA/ACK phase. The *contention phase* involves the sender sensing entire wide channel to be idle for a certain period of time[4], and then immediately afterwards the M-RTS/M-CTS handshake taking place. This handshake uses the entire wide channel. Note that, M-RTS is a special symbol that is transmitted by the sender, that allows the sender to contend for subchannels. The M-RTS just consists of a set of tones emitted on subcarriers. In order to contend for a randomly chosen subchannel, the sender randomly selects one of the K subcarriers that represent this subchannel and sends a tone on it. Note that, if we have multiple contenders, they can all transmit their M-RTSs simultaneously. Hence, the nodes that hear an M-RTS symbol, can in reality be hearing a combination of individually transmitted M-RTS symbols.

Each *Potential Receiver*[5], sends an M-CTS symbol, after an SIFS period after correctly receiving the M-RTS. The potential receivers locally resolve the contention on each subchannel, and embed the winning subcarrier information for each subchannel in their M-CTS symbols. Like M-RTS, M-CTS is also just a set of tones sent on subcarriers.

After receiving the M-CTS, if the sender wins on any of the subchannels, then after an SIFS period, the sender begins its *DATA/ACK phase*. Here, the sender sends its data packets on its wun subchannels, and after it finishes sending all its data packets, it waits to receive ACKs on the corresponding subchannels. If the sender does not receive an expected ACK after an SIFS duration, then it will assume that its data packet on that subchannel suffered a collision. 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.

Every node, i, also maintains a local variable called the *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. Each node i, updates its CW_i by using the AIMD algorithm.[6] Here, if a sender i receives ACKs for all the packets that it had transmitted, then, it increases CW_i by 1. 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 $p\%$.

Also, in the M-RTS and M-CTS symbols, there exists a reserved set of subcarriers, called the NAV Band. This can be used in order to allow Virtual Carrier Sensing to take place, and thus, reduce hidden terminal problems that can cause collisions at receivers. Senders when sending an M-RTS, will use the the NAV Band to specify the longest data transmission time that they might require. Each potential receiver will then echo back the longest transmission time it hears, in the NAV Band of the M-CTS symbol which it sends. The nodes receiving the M-CTS, will defer contention for the longest time needed as indicated in the NAV band. For more FICA PHY/MAC details, we refer interested readers to [8]and [3].

III. Problems Faced by the FICA MAC Protocol

We now revisit some of the problems that can easily arise with the FICA MAC scheme when we have packets of different sizes present in the network. These problems can degrade network performance as discussed in [8].

A. The Deafness Problem

Deafness occurs at a sender when it finishes successful DATA packet transmissions on some of the subchannels, but cannot hear the ACKs intended back for it on these subchannels, because, it is still busy transmitting on other subchannels. Here, we say that the sender is *deaf* to these ACKs coming from its receiver(s). The sender finishes sending smaller sized packets earlier than the larger sized packets, which consequently leads to deafness.

The sender, upon not hearing the ACKs related to its successful packets, will incorrectly conclude that these packets were not successful due to collisions. This will lead to needless retransmissions of already successful packets, as well as,

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

[5]*Potential receivers* are those nodes for whom a sender is contending.

[6]In this paper we focus on FICA's $AIMD$ scheme , which is shown to be better than the RMAX scheme of FICA for updating Contention Window [3].

unnecessary reduction of the sender's CW. The deafness problem can happen repeatedly at the same sender, which can cause a very inefficient usage of the available channel [8].

One can argue that deafness would not have happened if, for example, an AP picks and transmits packets of the same size from its outgoing queue, in its data transmission rounds. However, this is not a suitable solution, because the AP might not have enough packets of the same size to send on all the available subchannels. Thus, we can still have unused subchannels during data transmission rounds, which would cause the channel utilization to drop.

B. The Muteness Problem

Muteness occurs when a client finishes all its packet transmissions to the AP, and waits to receive ACKs from the AP after an SIFS period, however, the AP cannot transmit back ACKs for the successfully received packets, during this time, because it is still busy receiving from its other clients on other subchannels. We say that, here, the AP is *mute* for this client. The AP sends back ACKs on all the respective subchannels only after it finishes receiving all its intended data packets.

When the client does not receive ACKs for its successful packets, after an SIFS period, then the client incorrectly assumes that collisions have occured on all its respective subchannels. This not only causes the client to aggressively shrink its CW to 1, but it also causes the client to needlessly retransmit already successfully received packets. Moreover, muteness can occur repeatedly at clients, which can lead them to starvation [8], and cause unfairness in the network.

C. The Hidden Terminal Problem

Another problem that FICA faces is a certain form of hidden terminal problem. Here, when a sender S is busy receiving ACK packets from its receiver R, another node X, hidden from R but in the vicinity of S, can sense the channel to be idle, and transmit its M-RTS. This will cause collisions with the ACKs arriving at S. Note that in FICA, nodes do not undergo a random waiting time before sending an M-RTS. Thus, this form of hidden terminal problem can occur consistently, which can degrade the network performance and cause starvations as discussed in [8].

Note that FICA's NAV handling is not sufficient to prevent such hidden terminal problems. [3] does not mention whether a node P in the vicinity of S that receives S's M-RTS also defers its contention based upon the information in the NAV band, or not. However, even if P does not ignore the information in the NAV band, then this will still not be effective in preventing this form of hidden terminal problem always. For example, this approach will not be effective for the cases where the M-RTS sent by S is not received at P, due to collisions or because, P is also busy transmitting its M-RTSs at the same time. Moreover, note that, if a sender sends an M-RTS, then this means that it is contending to get access to some of the subchannels. However, it can happen that the sender does not win on any of the subchannels, but a node in the vicinity of the sender receiving the M-RTS will have to keep quite for the longest period indicated in the NAV. This can suppress otherwise harmless transmissions in the neighborhood.

IV. btFICA: Busy Tone Assisted Fine-Grained Channel Access Scheme

In this Section we develop our btFICA MAC scheme for high data rate WLANs, that solves the problems discussed in Section III, while keeping the strengths achieved with FICA.

A. btFICA - Design

1) Solving the Deafness and the Muteness Problems: From sections III-A and III-B, we make the key observation that *FICA's Acknowledgment scheme cannot fulfill its purpose of informing the senders of successful packet receptions.* If somehow, the senders had correct knowledge about the successful reception of their packets, then the senders would have taken correct corresponding actions, thus saving the channel from an inefficient usage. Hence, we argue that it is necessary to develop a new ACKing scheme for FICA, that prevents deafness and muteness, and that allows the senders to accurately know the state of their transmissions.

For this purpose, we equip every node with one additional, half-duplex, busy tone interface, that is capable of receiving (emitting) energy on multiple busy tone channels, simultaneously.[7] Busy tone (BT) interfaces with such capabilities are implementable [16], [7], [3].

Now, for every subchannel, we have a separate BT channel. *Upon reception of a correct packet, instead of having the receiver send an ACK packet back on the same subchannel, we make the receiver send a tone on the corresponding BT channel, for acknowledgement.* The receiver can continue receiving on the subchannels while sending tone(s) on BT channel(s), hence solving the muteness problem. The sender can also receive tones on BT channels for sent data packets, while it is transmitting on its other subchannels, hence, solving the deafness problem.

We use busy tones because, it is not only simple to implement, but, it also allows the receiver to instantaneously inform the sender of whether its transmission was successful or not. For btFICA, we allocate a portion of the wide channel for the BT channels and the guardbands needed between them. Our results in § VI, show that the impact of this overhead on the performance of btFICA is not significant. We call the rest of the entire wide channel as the *data channel*. We use FICA PHY when operating on the data channel. Like FICA, we use the data channel for sending M-RTS/M-CTS symbols and DATA packets. However, unlike FICA, btFICA does not have explicit ACK packets.

For the correct operation of btFICA, the power level for the BT interface should also be adjusted, so that the channel gain for both the data channel and the BT channels are the same.

Also, note that, unlike previous works that deal with busy tones [15], [11], [12], we are using busy tones in a new context

[7]Busy tone channels are very narrow band channels (in the range of 0.1 to 10 KHz [15]).

to solve new problems that arise with FICA. Also, in contrast to the previous works, we are making use of a BT interface that is capable of operating on multiple BT channels at the same time.

2) Solving the Hidden Terminal Problem and Preserving M-RTS Alignment Amongst Contenders: The solution that we proposed in Section IV-A1 also solves the hidden terminal problem described in Section III-C. This is because, a receiver, R, no longer sends ACK packets on subchannels, that could get collided at the sender, S, due to an M-RTS transmission from a hidden node X.

However, we do observe that there is one issue that is still occurring. S will not begin sensing the channel, in order to participate in the next contention round, before it finishes receiving its busy tones. However, X not hearing the busy tones, can begin sensing and seeing the start of an idle period before S. This can in turn lead X to transmit its M-RTS, earlier than S, and thus, an M-RTS/M-CTS handshake might complete without S even participating. Thus, S will not be able to transmit in the next round of data transmissions within the neighborhood. This can reduce channel utilization because the nodes that got a chance to transmit, might not have enough packets to utilize all the subchannels efficiently.

Hence, to address this we take the following approach. *After the sender finishes transmitting its largest data packets, the sender sends padding bits on those subchannels, for the entire time that it is receiving tone(s) from its receiver(s).* This can increase the chances for all nodes in the vicnity of both the sender and the receiver, along with the sender and the receiver themselves, to begin the next contention round at the same time. This can result in more nodes participating in contention, which can lead to more simultaneous transmissions within a neighborhood, and thus, better channel utilization.

3) Additional Changes from the FICA Scheme: We make use of this opportunity of having a separate BT interface, to also solve the hidden terminal problems that can corrupt data packet receptions at the receiver. Hence, in btFICA, for every intended data packet that the receiver r, started to receive correctly, r emits a tone on the corresponding BT channel for the entire time that it is receiving the packet. This will allow nodes in the vicinity of r that hear the tone(s) to defer contention. Hence, we do not need the NAV Band in the M-RTS and M-CTS symbols anymore.

This approach is better, because it is more accurate in informing contenders of whether there is an actual receiver in their vicinity that is actively involved in packet reception. In FICA, nodes receiving an M-CTS defer contention, but for the longest time that might be needed by a neighboring potential receiver, (i.e., a node that might be a receiver), to finish receiving its packets. Thus, FICA's approach for solving such hidden terminal problems is conservative, which can suppress harmless transmissions in the neighborhood.

Now, for btFICA, we also find it essential to design protocol operations that will protect the M-CTS arriving back at a sender s, from a collision with an M-RTS of a node that cannot hear the M-CTS. Hence, we allocate one more extra BT channel, which we call Q. Right after s finishes sending an M-RTS, s will start emitting a tone on Q and will continue doing so, until s finishes receiving the M-CTS. The nodes in the vicinity of s that hear this tone, will defer beginning a contention.

B. btFICA - Complete MAC Protocol Description

The btFICA MAC protocol, is for the most part similar to the FICA MAC protocol. Whenever the node is idle, it listens on both the wide data channel *and* the BT channels. In order to transmit data packets, a sender s, first starts carrier sensing on *both* the entire wide data channel and all the BT channels, for the same period of time as specified in FICA. If s finds the entire medium to be idle, it sends an M-RTS, and then begins emitting a tone on Q, and waits to receive an M-CTS.

When a node, n, receives an M-CTS, the node defers contention for a period of W_t, where $W_t = SIFS + preambletime + PLCPheaderTime + MACheaderTime$. This is inorder to give enough time for a receiver in the vicinity, to emit a busy tone. If n does not hear any tone after waiting for the W_t period then n resumes participating in contention normally.

A potential receiver, after receiving an M-RTS, will send back an M-CTS, *only* if it is not hearing any tones, and it is not waiting to hear any tones, on any of the BT channels except for Q. After sending the M-CTS, the potential receiver will switch to listening on the data subchannels, in order to receive data packets.

If s does not receive an M-CTS, s will stop the tone on Q and will will repeat the contention process again. If s correctly receives back an M-CTS, then s will stop the tone on Q and will start sending its data packets on the subchannels that it won, while listening on the corresponding BT channels.

If the receiver, r, starts to correctly receive a frame meant for it on a subchannel c, it will emit a tone on the corresponding busy tone channel b_c, for the entire time that it is receiving the frame. r knows if a frame is meant for it, if it can successfully decode the header of the frame. After r receives the entire frame, it checks for errors. If the frame is successfully decoded, then r will continue to transmit the tone on b_c for $A_t = SIFS + slottime$ more, for acknowledgment. Else, r will stop emitting the tone.

If s hears the corresponding busy tone for its frame contineously, since the beginning of its frame payload transmission until the end of its frame transmission, plus an additional A_t period, then the sender will conclude a successful packet reception. Otherwise, s will conclude a collision for this frame.

When s finishes sending its largest data packets, s will start to send *padding bits*, on these subchannels, that will take a maximum time of A_t. The padding bits are just ignored by r. s stops transmission on a subchannel as soon as it stops receiving the corresponding busy tone for that subchannel.

V. Points of Discussion

How can we cope with fadings on the narrow-width busy tone channels? There exists several PHY techniques for reducing the effect of fading on narrow-band channels. For example, one way is to use two or more MIMO antennas with antenna diversity schemes [17], [18], [19], instead of one antenna, when listening on the BT channels. Another technique is to allocate *two* narrow-band BT channels, b_{c1} and b_{c2}, that are spread out in the frequency spectrum, for each subchannel, c. b_{c1} and b_{c2} will face different fading characteristics, and are used as one unit. Hence, at the receiver, if a tone is detected on either b_{c1} or b_{c2}, the receiver will assume that it is receiving a tone for the subchannel c. With channels as wide as 160 MHz, and with each BT channel having a relatively very small width, the overhead of having two BT channels per subchannel should not be significant.

To avoid the muteness problem, why should we not just make the clients (senders) wait for a longer period than SIFS, in order to receive their ACKs? Before starting its data transmissions, a sender, s, usually knows of the longest possible time for which the nodes in its vicinity might be *receiving* data packets, using the NAV Band of its received M-CTS. While the sender s can wait for this entire period of time, to hear back an ACK [20], this may not be a good approach to take. This is because, after s finishes its data transmissions and waits for ACKs, there can be nodes in the vicinity of s that will begin sensing the channel idle and begin data transmissions. These data transmissions can collide with the ACKs that might later arrive at s. Another issue that can arise is that, during the time that s is waiting for its ACKs, potential receivers in the vicinity of s might send an M-CTS, which will not be received at s, since s is in its DATA/ACK phase and hence, is listening with a smaller FFT size[8] [3]. Thus, after s finishes waiting for its ACKs, s will not have a proper NAV set, and thus, can begin a new transmission causing collisions at those receivers. In contrast, our busy tone approach, accurately and instantenously informs s of whether its transmission(s) were successful, without causing any of the above problems.

We can solve the deafness problem if the sender adds padding bits on its smaller sized packets, in order to make its transmissions on all the subchannels take the same time. While this fix is effective in solving the deafness problem, and is more effective than packet fragmentation discussed in [20], it still cannot solve the other two problems in Section III. Our busy tone approach is a simple technique that solves all the problems comprehensively at the same time. It also provides additional benefits, such as reducing the acknowledgment overhead. With btFICA only 1 slot time is spent in signalling an “ack”, which is in contrast to FICA, where the ACK packet transmission spans several time slots.

Why did we not consider the approach where nodes always sense the channel idle for at least *SIFS+entire ACK packet transmission time+defined DIFS*, before sending an M-RTS, so that the hidden terminal problem would not occur? While this approach can solve the hidden terminal problem at the sender, clearly this is inefficient. Also, the M-RTS misalignment discussed, in Section IV-A2 will still occur. btFICA solves the hidden terminal problem at the sender without incurring large overheads.

How will btFICA work in networks where clients experience different SNR from the AP and thus can support different rates? In this case, even if we have one packet size in the entire network, the packets related to clients that support different rates will have different transmission times. Hence, while FICA will again face the same issues as in Section III, btFICA copes well in such settings.

How will btFICA perform in the presence of *packet capture effect*? In our context, *packet capture* refers to the phenomenon where despite of a collision on a subchannel, the receiver is still able to receive and decode the frame corresponding to the stronger signal. In 802.11 networks, the packet capture effect [21] is shown to cause unfairness [22]. In our context, the packet capture effect can sometimes cause false positives, i.e., a sender whose packet is not received at the intended receiver, might receive a busy tone for the needed duration, and thus, falsely conclude that its packet was successfully received. We can avoid such false positives from occuring by preventing packet capture. Note that, the receiver can tell if multiple senders are transmitting on the same subchannel by detecting an increase in the energy level [21] [23]. Thus, if a receiver detects multiple senders on the same subchannel, the receiver can ignore everything on this subchannel, and not send a busy tone. This will prevent packet capture from occuring, and thus will avoid false positives.

Note that, even if packet capture somehow still occurs, it may not be frequent and the impact of false positives may not be severe. Note that, in order for packet capture to occur in the first place, we need a collision on a subchannel. With btFICA, several conditions need to be satisfied before a collision can occur. Two senders, s_1 and s_2, in the vicinity of a receiver r should not only pick the same subchannel c, but should also select the same subcarrier sc on c, *and* sc should be selected as the winner by all the potential receivers in the vicinity of *both* s_1 and s_2, in order for a collision to happen on c at r. Now, it should be noted that not all collisions will necessarily lead to packet capture. For example, for packet capture to occur, the stronger signal must satisfy the SINR requirement, and there should be a slight delay in the arrival times of the two signals at r [23]. Now, even if all the conditions for packet capture are met and a false positive occurs, then this may not have severe consequences.

Firstly, the sender of the weaker signal, seeing a false positive, does not reduce its CW, thus, we will not face channel access unfariness between this sender and the sender of the stronger signal. Secondly, in the next rounds of contention both senders can select and win on different subchannels, thus, the issue of false positive might not occur consistently at a sender. The only concern is that, when the sender sees a false positive, it will not retransmit the packet that requires a retransmission. However, again this may not be an issue, because, some upper layer protocols such as TCP can detect such missed packets and trigger needed retransmissions. Thus, we speculate, that

[8]With FICA PHY, a node during its contention phase will switch to FFT/IFFT size that is twice that of the DATA/ACK phase.

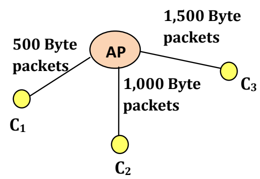


Fig. 1. Single cell setting with 3 clients. We have different packet sizes for each link. All nodes can hear each other.

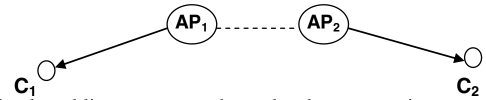


Fig. 2. The dotted lines represent the nodes that can carrier sense each other. Solid lines represent client-AP associations. We assume only downlink traffic. AP_1 and AP_2 have packets of sizes 500 bytes and 1500 bytes, respectively.

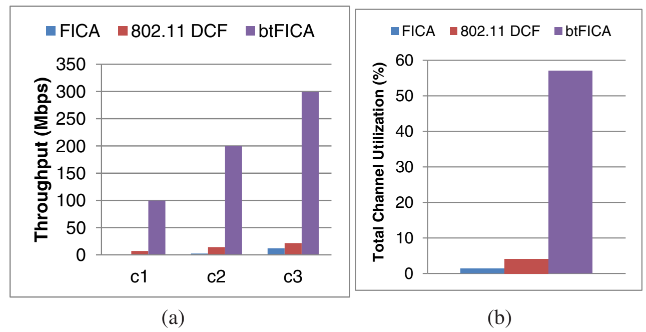


Fig. 3. Results for the scenario described in Section VI-B1. The effectiveness of btFICA in addressing the deafness problem is shown.

even if packet capture can not be completely eliminated, it may be infrequent, and if false positives sometimes occur, then this may not have a severe impact on network throughput or fairness. We leave a full study of this matter as future work.

VI. Performance Evaluation

Now we evaluate and compare the performance of the FICA MAC protocol, btFICA MAC protocol and 802.11 DCF, in high data rate WLAN settings.

A. Simulation Methodology

We have implemented a detailed event-based simulator for each of these 3 protocols. Our simulators also capture the details of the FICA PHY layer. Note that the 802.11 standard has a different PHY layer than that of FICA, which will naturally cause a slight mismatch between the possible PHY data rates of 802.11 and FICA [3]. However, since our goal is to isolate and compare the benefits that can be provided by the *MAC* protocols, over the *same* high PHY data rate, we find it important to maintain the same PHY, (FICA PHY), for all the MAC schemes. Note that, for 802.11 DCF, every node 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 and btFICA, we have 16 data subcarriers per subchannel as in [3], which gives us a total of 128 subchannels. For btFICA, each BT channel and each guardband that goes between adjacent BT channels, is 4.5 KHz wide. Thus for btFICA, we allocate 0.7% of the wide channel (i.e., 1 subchannel) for busy tone operations, and use the remaining 127 subchannels for the data channel.

Also in FICA PHY, we have constant power per active subcarrier across all nodes. Hence, unlike 802.11 PHY [24], [25], in our case, the interference($50m$) and transmission($45m$) ranges of nodes remain the same, even if they access a portion of the wide channel. Our timing parameters, such as SIFS, slot time, etc, are the same as in [3]. To make conditions favorable for the FICA MAC protocol, we have used the $AIMD$ backoff scheme.Since with 802.11 DCF, the packets will take a small transmission time, we turn off RTS/CTS in order to achieve better efficiency [26]. The maximum number of retransmissions for a packet is 7.

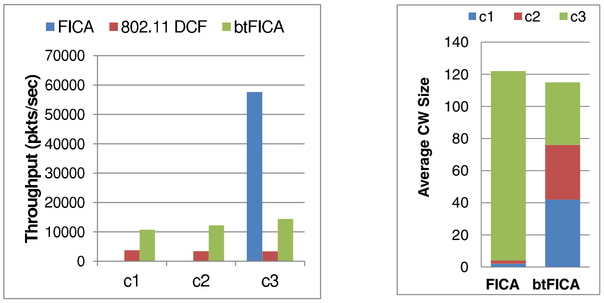
B. Simulation Results - Sample Networks

In this Section we consider 3 different scenarios, in each of which FICA faces only 1 of the 3 problems described in § III. This will allow us to quantify the improvement in performance that btFICA can provide in each of the individual cases. For completeness, we later also show results for single-cell and multi-cell *random* networks.

1) Scenario 1: We consider the scenario shown in fig. 1. Our choices of packet sizes are motivated by a study done in [7]. We assume only downlink traffic, and that the AP always has packets to send to all its clients. Here, FICA only faces the deafness problem. In this scenario, no collisions are happening in the network. From fig. 3(b), we can see that, the deafness problem of the FICA MAC protocol is detrimental enough to cause the efficiency to drop to as low as *1.4%*. We also show that btFICA significantly improves the performance over both FICA and 802.11 DCF. btFICA provides a *40* times improvement in efficiency over FICA, showing its effectiveness in solving the deafness problem faced by FICA. btFICA also performs 9 times better than 802.11 DCF, because btFICA maintains the strengths that are acheived by frequency-domain contention and fine-grained channel access.

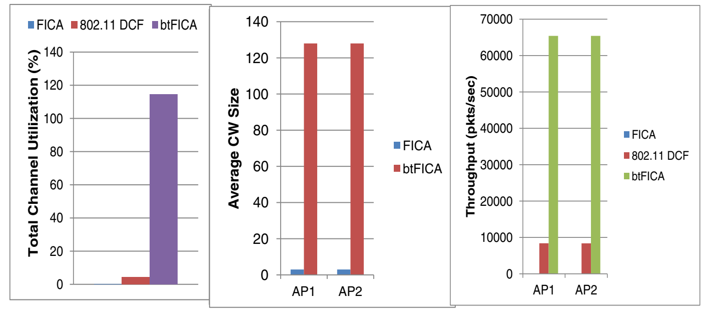
In fig. 3(a), we show the throughput achieved per-downlink flow. As we expect, with FICA, every flow not only achieves a lower throughput that 802.11 DCF, but also, the flow towards c_1, that contains the smallest sized packets, starves. In contrast, with btFICA we can see that here, there is no starvation of flows and the per-flow throughput is improved significantly. Note that, btFICA provides an equal channel access opportunity for all flows.

2) Scenario 2: We again consider the scenario shown in fig. 1, except that now we assume only uplink traffic and that all clients are backlogged, i.e., they always have packets to send. Here, FICA faces only the muteness problem. It is clear from fig. 4(a) that with FICA, all the clients with smaller packet sizes, (c_1 and c_2), starve, but, only one client c_3 that has the largest packet size, is allowed to have a very high throughput. In contrast, btFICA alleviates starvation and



(a) (b)

Fig. 4. Results for the scenario in Section VI-B2. The effectiveness of btFICA in addressing the muteness problem is shown.



(a) (b) (c)

Fig. 5. Results for the scenario described in Section VI-B3. The effectiveness of btFICA in addressing the hidden terminal problem is shown.

provides a fairer throughput for each of the clients. We further show fine-grained results in fig. 4(b). We can see that, with FICA the CW size for c_3 is much 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 very few subchannels. In contrast with btFICA, the contention window sizes for clients are similar in size. Hence, with btFICA, all clients access almost the same number of subchannels, which leads to better fairness in the network. Thus, btFICA is effective in solving the muteness problem faced by FICA.

3) Scenario 3: Now we consider the scenario shown in fig. 2. We assume that both the APs are backlogged. Here, FICA only faces the hidden terminal problem described in § III-C. Both the APs begin contention simultaneously. It is clear from fig. 5(a), that the impact of the hidden terminal problem is severe enough to cause the efficiency of FICA to drop very close to 0. In contrast, with btFICA we achieve an efficiency of approximately 120%, which is a significant improvement over the other two schemes. Note that, in this scenario if both the APs transmit simultaneously, on the same subchannels their transmissions will still be successful, because of the way the clients are positioned. btFICA is achieveing a high efficiency because, btFICA is constantly allowing both the APs to use all the subchannels simultaneously, while correctly informing both the APs of the successful packet receptions. In fig. 5(b), we can see that with btFICA both APs maintain the maximum CW size. In fig. 5(c), we can see that btFICA gives significantly higher per-AP throughput, while maintaining channel access fairness amongst the two flows. In contrast, FICA is starving both flows. Clearly, btFICA is effective here in solving the hidden terminal problem.

C. Simulation Results - Single Cell Random Networks

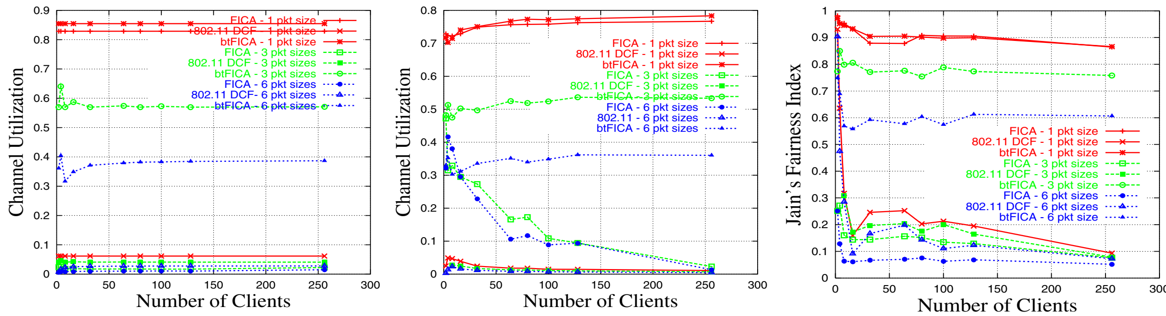
We now study the 3 MAC protocols in random network topologies and random traffic settings. We have one AP and we place clients on random locations that are within the the AP's transmission range. We consider cases with one packet size, of 1500 bytes, in the entire network, as well as cases with 3 , 6 and 12 different packet sizes in the network, respectively. All packet sizes are in the range of 100 bytes to 1500 bytes.

In figures 6 and 7, we assume that the AP is 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 fig. 6 for every link in the network, we randomly choose a *fixed* packet size, from the set of packets sizes for the network. Now, in fig. 6(a), we only have downlink traffic in the network, and we plot the efficiency achieved with each MAC protocol, 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 substantial improvement over 802.11 DCF. btFICA also performs very well here, showing that btFICA *maintains* the positive features of FICA. Note that, here, btFICA performs slightly better than FICA, because, btFICA does not incur overheads due to ACK preambles.

However, we show that when we have different packet sizes in the network, the efficiency for FICA drops drastically, due to the deafness problem. For example, with as little as 3 different packet sizes, FICA's efficiency drops to as low as 1%. On the other hand, btFICA achieves a 57 times improvement over FICA, even with increasing number of clients. btFICA also achieves significant improvement over 802.11 DCF. We get similar results when we increase the different number of packet sizes in the network to 6 and 12, respectively.

In fig. 6(b) and 6(c), we use the same topologies as in fig. 6(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 fig. 6(b) we again plot the efficiency and see that when we have different packet sizes in the network, btFICA provides much better efficiency over FICA and 802.11 DCF, for different number of clients. Note that while FICA provides better efficiency than 802.11 DCF, our analysis shows that FICA gives such improvement, at the cost of starving uplink flows, that contain smaller packet sizes, and giving uplink flows with the largest packet size a larger share of the bandwidth.

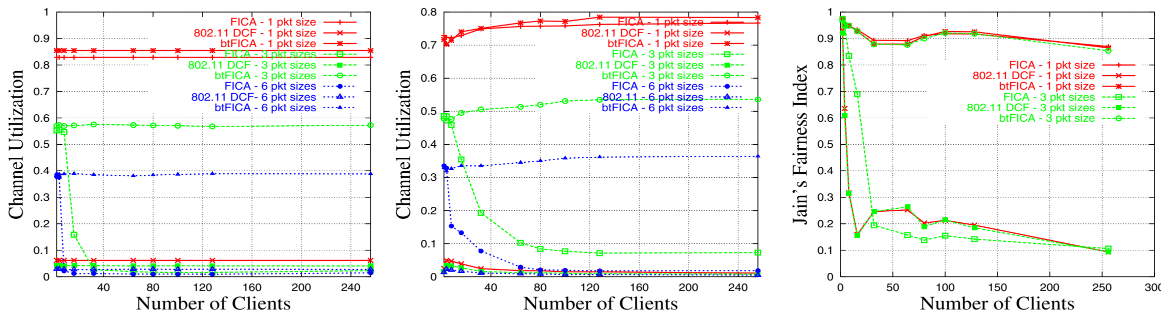


(a) Only downlink traffic is present.

(b) Both Downlink and Uplink traffic is present.

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

Fig. 6. 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. 7. Results for one AP and randomly placed clients. For each flow we have *variable* packet sizes that are randomly chosen.

To further verify our claims, in fig. 6(c), we use the Jain's Fairness Index [27] to compute the level of throughput-fairness amongst all flows in the network. Here, a value close to 1 indicates a high level of fairness. We see that FICA, with one packet size in the network, performs very good interms of fairness. However, as expected, FICA's throughput-fairness drops to very low values, when we have different packet sizes in the network. In contrast, btFICA maintains a much higher level of fairness amongst flows, even in the presence of different packet sizes and even as we increase the number of clients. For example, for the case of 16 clients and 3 different packet sizes in the network, the fairness Index for btFICA, 802.11 DCF and FICA are 0.8, 0.17 and 0.14, respectively. Clearly, btFICA provides noticeable gains in terms of both efficiency *and* fairness.

The results in fig. 6 were for the case where on each link we only had packets of a *fixed* size to transmit. However, in real-world settings a flow can also consist of packets of *different* sizes. Thus, in fig. 7, we evalute the three schemes under such settings as well. Clearly, btFICA substantially outperforms FICA and 802.11 DCF interms of both efficiency and fairness, again. We also find it important to evaluate the performance of the three schemes in settings where we do *not* necessarily have backlogged traffic. In fig. 8, for every uplink and downlink, we choose the data arrival rate for the sender, randomly, from the range of 800 Kbps to 200 Mbps. To every link, we randomly assign a packet size. We also vary the number of clients in the network and we can see that the efficiency of btFICA still remains better than both FICA and 802.11 DCF.

D. Simulation Results - Multi-Cell Random Networks

We have also evaluated the 3 MAC schemes in a wide variety of multiple AP settings and traffic scenarios, and we show some of our results here. In all cases where we had different packet sizes in the network, we found btFICA providing better efficiency than both FICA and 802.11 DCF. In fig. 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 for each case. Each client is associated with that AP from which it received the strongest signal. We assume only downlink traffic with backlogged APs. Each flow contains packets of different sizes that are randomly chosen.Here again, we show that btFICA can provide better efficiency than FICA and 802.11 DCF. For example, for the case for 3 packet sizes in the network and 32 clients, for FICA we acheive an efficiency of 21%. In contrast, with btFICA we achieve an efficiency as high as 90%. Note that btFICA can in some cases give an efficiency that is greater than 100%. This is indeed possible, because in multiple collision domain networks, non-interfering links can reuse any portion of the entire wide channel.

VII. Related Works

In [7], the B2F MAC protocol is proposed for reducing channel contention overheads. While B2F can provide a better efficiency that 802.11 DCF at high PHY data rates, B2F achieves a lower efficiency than btFICA. This is because, unlike btFICA, in B2F, the entire wide channel is treated as a single entity, which causes the DIFS period and the frequency-domain contention period to still be followed by a relatively small period of data transmission. Also, in multiple collision

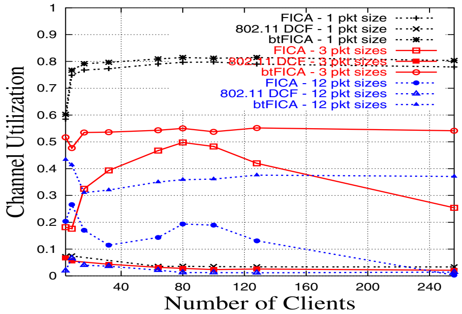


Fig. 8. Single cell setting with arrival rates for flows chosen randomly.

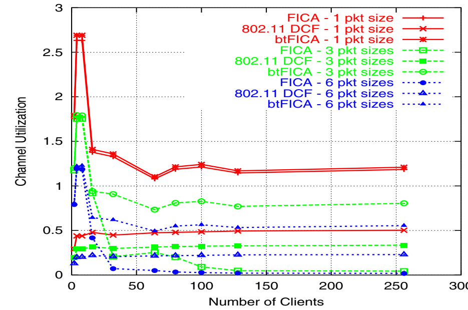


Fig. 9. Results for WLAN with 6 APs deployed in a 200m x 200m area.

domain networks, with B2F we can face starvation of nodes. Moreover, the B2F scheduled transmissions option [7], for improving efficiency, can easily become ineffective in multiple collision domain networks.

In [4], a new PHY/MAC scheme, WiFi-Nano, is proposed, which reduces the slot size from the standard 9 μsec to 800ns, in order to lower overheads due to time-domain contention. However, WiFi-Nano does not increase efficiency by a large amount at high data rates, e.g., efficiency is 16.7% with 600 Mbps PHY data rate [4], whereas btFICA provides a much better efficiency under the same settings. This is because in WiFi-Nano the preamble overheads are still substantial. In contrast, btFICA masks the effect of preamble time by overlapping the preamble transmissions of multiple data packets in time, and following them by relatively long periods of data transmissions on subchannels. Also, there is no preamble overhead with btFICA's ACKing scheme. Also, WiFi-Nano depends upon IdleSense [28], which is not defined for multiple collision domain networks, thus making the performance of WiFi-Nano under such settings, unclear.

Moreover, the Contention Window tuning schemes [28], [29], proposed for 802.11 networks, cannot improve efficiency in high data rate WLANs significantly.

Finally, as discussed in [3] and [8] the 802.11 DCF's packet aggregation scheme is not as practical for improving efficiency in high data rate WLANs as btFICA. This is because, as we shift to higher data rates each individual sender becomes less likely to be able to aggregate enough packets to enhance the overall channel utilization. Also, as shown in [5], even if a sender can transmit many packets back-to-back, 802.11 DCF will enhance efficiency but at the cost of lowering fairness amongst nodes. This is in contrast to btFICA, where we can maintain both high efficiency *and* fairness in the network.

VIII. Conclusion

In this paper we developed a new MAC protocol, btFICA, for the purpose of improving channel utilization in high data rate WLANs. btFICA is based upon a state-of-the-art PHY/MAC scheme, called FICA, and uses an additional busy tone antenna. btFICA effectively and comprehensively solves all the three problems that arise with the FICA MAC protocol, without incurring significant overheads. We have shown, via extensive simulations, the superiority of btFICA over both FICA and 802.11 DCF, in terms of channel utilization, per-user-throughput and fairness. Our results show that btFICA can improve channel utilization in WLANs by upto 40 times compared to the original FICA scheme. As part of our future work, we plan to implement our busy tone approach in a real wireless testbed using the USRP/Gnuradio platform.

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] "Gigabit Devices 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 *et al.*, "Wifi-nano: reclaiming wifi efficiency through 800 ns slots," in *Proceedings of the MOBICOM Conference*, 2011.

[5] R. Maheshwari *et al.*, "Adaptive channelization for high data rate wireless networks," WINGS Lab, SBU, Tech. Rep., 2009.

[6] V. Shrivastava *et al.*, "802.11n Under the Microscope," in *Proceedings of the 8th ACM SIGCOMM conference on IMC*, 2008.

[7] S. Sen *et al.*, "No Time to Countdown: Migrating Backoff to the Frequency Domain," in *Proc. of MOBICOM*, 2011.

[8] F. Zarinni and S. Das, "Understanding the FICA MAC Protocol in High Data Rate WLANs," in *Proc. of ICCCN*, 2012.

[9] http://standards.ieee.org/getieee802/download/802.11n-2009.pdf.

[10] http://www.cs.umd.edu/projects/wifidelity/sigcomm08_traces/.

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

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

[13] S. Sen *et al.*, "CSMA/CN: Carrier Sense Multiple Access with Collision Notification," in *Proceedings of MOBICOM*. ACM, 2010.

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

[15] Z. Hass *et al.*, "Dual busy tone multiple access (DBTMA)-a multiple access control scheme for ad hoc networks," in *IEEE Tran.Commun*, '02.

[16] L. Yang *et al.*, "Supporting Demanding Wireless Applications with Frequency-agile Radios," in *Proc. of NSDI*, 2010.

[17] "Antenna Diversity," http://en.wikipedia.org/wiki/Antenna_diversity.

[18] A. Goldsmith, *Wireless Communications*. Cambridge Uni. Press., 2005.

[19] H. Daniel *et al.*, "802.11 with Multiple Antennas for Dummies," *SIGCOMM Comput. Commun. Rev.*

[20] "Dr. Kun Tan, Microsoft Research Asia, Personal Email Communication, May 2012."

[21] J. Lee *et al.*, "An experimental study on the capture effect in 802.11a networks," in *WinTECH*. ACM, 2007.

[22] A. Kochut *et al.*, "Sniffing out the correct physical layer capture model in 802.11b," in *ICNP*, 2004.

[23] D. Halperin *et al.*, "Taking the sting out of carrier sense: interference cancellation for wireless lans," in *MOBICOM*, 2008.

[24] S. Rayanchu *et al.*, "FLUID: Improving Throughputs in Enterprise Wireless LANs Through Flexible Channelization," in *MOBICOM*, 2011.

[25] R. Chandra *et al.*, "A Case for Adapting Channel Width in Wireless Networks," *SIGCOMM*, 2008.

[26] G. Bianchi, "Performance analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE J. Sel. Areas Commun.*, 2000.

[27] J. R. *et al.*, "A quantitative measure of fairness and discrimination for resource allocation in shared systems." E.Lab,WU, Tech. Rep., '84.

[28] M. Heusse *et al.*, "Idle Sense: An optimal access method for high throughput and fairness in rate diverse wireless LANs," in *Sigcomm*, '05.

[29] F. Cali *et al.*, "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit," in *IEEE/ACM ToN*, 2000.