Fine-Grained Channel Access in Wireless LAN

Ji Fang, *Student Member, IEEE*, Kun Tan, *Member, IEEE*, Yuanyang Zhang, Shouyuan Chen, Lixin Shi, Jiansong Zhang, Yongguang Zhang, *Senior Member, IEEE*, and Zhenhui Tan, *Member, IEEE*

***Abstract*—With the increasing of physical-layer (PHY) data rate in modern wireless local area networks (WLANs) (e.g., 802.11n), the overhead of media access control (MAC) progressively degrades data throughput efficiency. This trend reflects a fundamental aspect of the current MAC protocol, which allocates the channel as a single resource at a time. This paper argues that, in a high data rate WLAN, the channel should be divided into separate subchannels whose width is commensurate with the PHY data rate and typical frame size. Multiple stations can then contend for and use subchannels simultaneously according to their traffic demands, thereby increasing overall efficiency. We introduce FICA, a fine-grained channel access method that embodies this approach to media access using two novel techniques. First, it proposes a new PHY architecture based on orthogonal frequency division multiplexing (OFDM) that retains orthogonality among subchannels while relying solely on the coordination mechanisms in existing WLAN, carrier sensing and broadcasting. Second, FICA employs a frequency-domain contention method that uses physical-layer Request to Send/Clear to Send (RTS/CTS) signaling and frequency domain backoff to efficiently coordinate subchannel access. We have implemented FICA, both MAC and PHY layers, using a software radio platform, and our experiments demonstrate the feasibility of the FICA design. Furthermore, our simulation results show FICA can improve the efficiency of WLANs from a few percent to 600% compared to existing 802.11.**

***Index Terms*—Cross-layer, fine-grained channel access, media access control (MAC), orthogonal frequency division multiple access (OFDMA), wireless.**

Manuscript received September 19, 2011; revised April 16, 2012; accepted July 10, 2012; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor T. Javidi.

J. Fang is with the Institute of Broadband Wireless Mobile Communications, Beijing Jiaotong University, Beijing 100044, China, and also with the State Key Laboratory of Railway Traffic Control and Safety, Beijng Jiaotong University, Beijing 100044, China; the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China; and Microsoft Research Asia, Beijing 100080, China (e-mail: imfangji@gmail.com).

K. Tan, J. Zhang, and Yo. Zhang are with Microsoft Research Asia, Beijing 100080, China (e-mail: kuntan@microsoft.com; jiazhang@microsoft.com; ygz@microsoft.com).

Yu. Zhang is with the University of California, Santa Barbara, Goleta, CA 93106 USA, and also with Microsoft Research Asia, Beijing 100080, China (e-mail: zhangyy1209@gmail.com).

S. Chen is with the Chinese University of Hong Kong, Hong Kong, China, and also with Microsoft Research Asia, Beijing 100080, China (e-mail: sychen@cse.cuhk.edu.hk).

L. Shi is with the Massachusetts Institute of Technology (MIT), Cambridge, MA 02139 USA, and also with Microsoft Research Asia, Beijing 100080, China (e-mail: lixshi@mit.ed).

Z. Tan is with the Institute of Broadband Wireless Mobile Communications, Beijing Jiaotong University, Beijing 100044, China, and also with the State Key Laboratory of Railway Traffic Control and Safety, Beijng Jiaotong University, Beijing 100044, China, and the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China (e-mail: zhhtan@bjtu.edu.cn)

Digital Object Identifier 10.1109/TNET.2012.2212207

I. INTRODUCTION

MODERN communication technologies are steadily advancing the physical-layer (PHY) data rates in wireless local area networks (WLANs). This capacity growth is achieved primarily through wider channel widths and advanced PHY techniques like multiple-input–multiple-output (MIMO). The latest ratified 802.11n [1] has boosted data rates to 600 Mb/s, and future standards, like IEEE 802.11ac/ad, are already poised to provide even faster ($>$ 1 Gb/s) PHY rates.

The data throughput efficiency, i.e., the ratio between the network throughput and the PHY data rate, however, may degrade rapidly as the PHY data rate increases. For example, if the payload size of a media access control (MAC) frame is 1500 B, i.e., common maximal transmit unit (MTU) size of IP packets, the efficiency ratio in an 802.11n network at 300 Mb/s is only 20%. That is, the 300-Mb/s data rate sustains an actual throughput of only 60 Mb/s.

The fundamental reason for this inefficiency is that the current MAC allocates the entire channel to one station as a single resource. This allocation strategy can become too coarse-grained when the channel width increases or PHY data rate increases. Even if a sender has a small amount of data to send, it still needs to contend for the entire channel. The contention resolution time therefore introduces an overhead to the channel time used for data. Unfortunately, this overhead cannot easily be reduced due to constraints of current electronics and physical laws. As a result, the higher the PHY data rate, the lower the throughput efficiency may become.

One way to improve the throughput efficiency is to extend the useful channel time for data transmissions by sending larger frames. Indeed, IEEE 802.11n allows frame aggregation, i.e., sending multiple frames together in one contention period. However, when the PHY data rate increases, the aggregated frame size needs to increase as well, e.g., achieving an efficiency of 80% in a 300-Mb/s network would require frames to be as large as 23 kB. Such large aggregation size usually means a long delay as the sender must wait to collect enough application data before transmission, resulting in adverse effects to real-time applications like VoIP and video conferencing. Moreover, applications involving chatty protocols or short-lived sessions, e.g., Web browsing, may not even benefit from frame aggregation as all data they can generate are small.

We argue that a better way to improve WLAN efficiency is to effectively reduce the channel width and create more channels, where the channel width is commensurate with the PHY data rate and typical frame size. Multiple stations can then contend for and use these smaller channels simultaneously according to their traffic demands, thereby amortizing MAC coordination and increasing overall efficiency. We call this method *fine-grained channel access* (FICA) for WLANs.

It is, however, nontrivial to divide a wide channel band into multiple subchannels without losing useful frequency spectrum. One common practice is to allocate both edges of two adjacent subchannels as "guard bands" to properly space the transmissions to avoid mutual interference. These guard bands can add up to significant overhead, though, especially if the number of subchannels is large. For example, 802.11a uses a 1.875-MHz guard band at both edges of every channel. If a 20-MHz channel is divided into four 5-MHz subchannels, the overhead will amount to 75% of the total bandwidth. Furthermore, the guard band width cannot be easily reduced due to power mask requirements and the difficulty of filter designs.

Orthogonal frequency division multiplexing (OFDM) is a well-understood PHY-layer technology that can eliminate the need of guard bands if the frequency and width of subchannels are strategically picked and transmission on each subchannel is synchronized in a way to become "orthogonal," and hence noninterfering, to one another. Although some cellular networks (e.g., WiMAX [2] and 3 GPP LTE [3]) have proposed using OFDM in channel multiaccess (OFDMA), doing so requires tight synchronization among user handsets, and they cannot support random access. It thus remains a new technical challenge for how to use OFDM-type channelization for fine-grained channel access among distributed and asynchronous stations in a random access WLAN, where it is impractical and unnecessary to achieve similar tight synchronization.

In this paper, we present the design and implementation of FICA, a novel cross-layer architecture based on OFDM that enables fine-grained subchannel random access in a high data rate WLAN. FICA introduces two key techniques to address the aforementioned challenges.

- FICA proposes a new PHY architecture based on OFDM. Solely relying on the coordination mechanisms provided by existing WLANs, carrier sensing and broadcasting, FICA retains orthogonality among subchannels with low overhead.
- FICA employs a novel *frequency-domain contention method* that uses physical-layer Request to Send/Clear to Send (RTS/CTS) signaling and *frequency-domain backoff* for contending subchannels. We show that frequency-domain contention is much more efficient than the conventional time-domain contention mechanism in a fine-grained channel access environment.

We have implemented a FICA prototype on the Sora software radio platform [23]. Our implementation demonstrates the feasibility of our key techniques for both PHY and MAC design. We further use ns-3 simulation to evaluate FICA in large-scale wireless environments under different traffic patterns. Our results show that FICA has up to a 6-fold gain in efficiency compared to existing 802.11n with all its optimizations.

In summary, this paper makes the following contributions. 1) We describe and examine the efficiency issue of current MAC protocols in the context of high-speed WLANs and argue that this issue can be resolved by fine-grained channel access. 2) We design and implement FICA, a protocol that enables fine-grained subchannel random access in WLANs. 3) We demonstrate the feasibility of FICA with a prototype implementation on a software radio platform and evaluate its performance using detailed simulation. To the best of our knowledge, FICA is the first system that enables fine-grained channel access in WLANs.

The rest of the paper is organized as follows. Section II provides a detailed analysis of the source of inefficiency in current MAC protocols. We then describe the design of FICA in Section III and evaluate its performance using simulation in Section IV. After describing the implementation of a FICA prototype using a software radio platform in Section V, we evaluate its performance in Section VI. Finally, Section VII discusses related work, and Section VIII concludes.

II. Background and Challenges

A. Inefficiency of Current WLANs

State-of-the-art MAC protocols in wireless LANs manage the whole channel (e.g., 20/40 MHz width) as a single resource. The MAC protocol arbitrates access among multiple potential senders and selects one as the winner, which then consumes the whole channel resource to transmit. If multiple senders transmit at the same time, collisions may happen, and receivers will likely fail to decode the transmissions.

Current 802.11 WLANs use carrier sensing multiple access with collision avoidance (CSMA/CA) for their MAC protocol. When the channel is busy, all contending nodes wait until the channel becomes free. The MAC employs a random backoff scheme to avoid having multiple nodes transmitting simultaneously. Each node will randomly choose a number b within a contention window $[0, CW)$, and wait for b time-slots before it starts transmitting. If a node detects a transmission during its backoff period, it will freeze the backoff counter until the channel is free again. If two nodes randomly choose the same backoff time, their transmissions will eventually collide. A collision is usually detected by a missing acknowledgement (ACK) from the receiver. When a collision is detected, a sender will double its contention window CW according to the binary exponential backoff (BEB) algorithm to further reduce the collision probability for the next transmission.

Fig. 1 illustrates the channel access timing diagram of the 802.11 MAC. Fig. 1(a) is the basic access method, and Fig. 1(b) shows channel access with the optional RTS/CTS handshake to handle hidden terminals. The Short Interframe Space (SIFS) is the shortest time interval required for a receiver to return a message to a sender. It is determined by (1), where $t_{\mathrm{rf\_delay}}$ is the delay incurred to transfer digital signals from the RF antenna to the processing unit, t_{proc} is the time needed for the processing unit to operate on the incoming signals, and t_{TxRx} is the time needed for the RF front end to switch from receiving mode to transmitting. Normally, SIFS is about 10–16 μs. The Distributed Interframe Space (DIFS) is determined based on SIFS and the backoff slot time, as shown in (2). DIFS is defined to support priorities in CSMA/CA and should be larger than SIFS. The backoff slot time is critical. It is the minimal time needed for a node to sense the channel condition and acquire the channel.

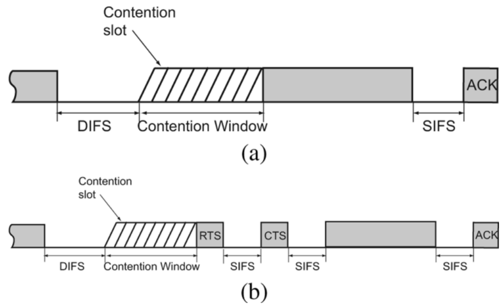


Fig. 1. Illustration of CSMA/CA access method. (a) Basic access. (b) With RTS/CTS handshake.

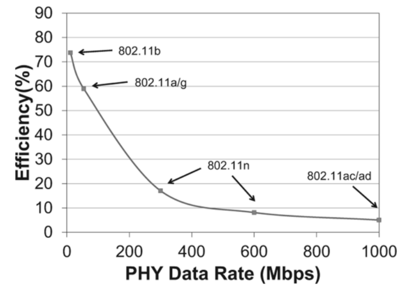


Fig. 2. Inefficiency of 802.11 MAC at high data rates with a typical Ethernet MTU (1500B).

Slot time is determined by (3), where $t_{\rm cca}$ is the time for a node to measure the channel energy to decide the channel status, and $t_{\rm prop}$ is the time for the radio signal to reach the maximal distance of the network

$$t_{\rm sifs} = t_{\rm rf\_delay} + t_{\rm proc} + t_{\rm TxRx} \quad (1)$$
$$t_{\rm difs} = t_{\rm sifs} + 2 \cdot t_{\rm slot} \quad (2)$$
$$t_{\rm slot} = t_{\rm cca} + t_{\rm TxRx} + t_{\rm prop} + t_{\rm proc}. \quad (3)$$

Using these values, we can build a simple analytical model to compute the efficiency ratio for CSMA/CA. Since a node chooses a random number uniformly from the contention window $[0, CW)$, the expected number of backoff slots is $W = \frac{CW}{2}$. The following equation gives the efficiency ratio for the basic access of CSMA/CA:

$$\eta = \frac{t_{\rm data}}{t_{\rm slot} \cdot W + t_{\rm difs} + t_{\rm preamble} + t_{\rm sifs} + t_{\rm ack} + t_{\rm data}} \quad (4)$$

where $t_{\rm data}$ is the time used for data transmission, $t_{\rm preamble}$ is the time used to transmit per-frame training symbols, and $t_{\rm ack}$ is the time used for the ACK frame.

Only $t_{\rm data}$ is used for transmitting application data, while all other times are overheads. Some overheads are constrained by physical laws and current limitations in state-of-the-art radio electronics. For example, you cannot reduce $t_{\rm prop}$ less than 1 μs to cover a network with a radius of a few hundreds of meters. It is also difficult to reduce $t_{\rm TxRx}$ since the RF circuit requires a few microseconds to settle down for sending or receiving. Others are needed for the correct operation of the protocol. For example, we need training symbols for reliable estimation of the wireless channel for each frame, thus $t_{\rm preamble}$ is essential. The average number of backoff slots, denoted by W, reflects the ability of CSMA/CA to avoid collisions. Thus, to work well in normal network settings, we need a reasonably large W. ACKs are also needed to detect collisions and other losses, thus in general we do not want to remove $t_{\rm ack}$. Table I outlines some timing parameters defined in 802.11. They remain similar across the different standards of 802.11a/g/n except for the preamble; since 802.11n uses MIMO, it requires more training symbols in its preamble.

Therefore, when the PHY data rate increases, only $t_{\rm data}$ will be reduced proportionally, while the other parameters remain largely unchanged. As a consequence, the efficiency ratio η decreases inversely proportionally. Fig. 2 illustrates such a phenomenon: The efficiency quickly decreases from 60% at 54 Mb/s (802.11a/g) to less than 10% at 1 Gb/s (future 802.11ac/ad).

TABLE I
TIMING PARAMETERS OF 802.11

Parameter	Value
$t_{\rm slot}$	$9\mu s$
$t_{\rm sifs}$	10–$16\mu s$
$t_{\rm cca}$	$4\mu s$
$t_{\rm TxRx}$	$\leq 5\mu s$
$t_{\rm prop}$	$\leq 1\mu s$
$t_{\rm preamble}$	20–$56\mu s$

As mentioned in Section I, transmitting larger frames will improve the efficiency ratio, but such a frame-aggregation approach has practical limitations. Fine-grained channel access will be a better approach if we can divide the whole channel into smaller subchannels efficiently and allow different nodes to access different subchannels simultaneously. Enabling concurrent transmissions across subchannels is in effect an aggregation and opportunity to amortize the MAC overhead across different nodes.

B. OFDM Primer

Orthogonal frequency division multiplexing (OFDM) has become increasingly popular in modern wireless communications [17]. It has been embraced by many existing wireless standards like IEEE 802.11a/g/n, WiMAX [2], and by future standards like 3 GPP LTE [3]. Cognitive radio technologies also mainly rely on OFDM to use noncontiguous spectrum bands for communication [19].

OFDM divides a spectrum band into many small and partially overlapping signal-carrying frequency bands called subcarriers. The subcarrier frequencies are chosen so that they are "orthogonal" to one another, meaning that crosstalk between subcarriers sums up to zero even though they are overlapping (Fig. 3). OFDM can therefore pack subcarriers tightly together without intercarrier interference, eliminating the need to have guard bands.

OFDM can be efficiently implemented using (inverse) fast Fourier transform (iFFT/FFT). In an OFDM system with FFT size N, each subcarrier has exactly the same width of $\frac{B}{N}$, and the subcarrier central points are located at frequencies of $f_{\rm c} + \frac{2\pi n B}{N}$, $n = -\frac{N}{2}..(\frac{N}{2} - 1)$, where $f_{\rm c}$ is the central frequency of the channel and B is the channel width. Different modulations (e.g., BPSK, QPSK, etc.) can be applied to each subcarrier independently. After modulating information onto each subcarrier, the

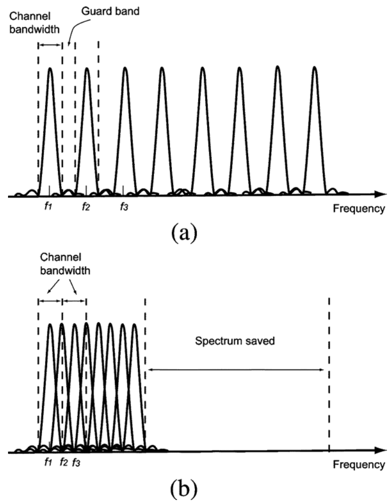


Fig. 3. OFDM achieves higher spectrum efficiency. (a) Normal frequency division multiplexing. (b) OFDM.

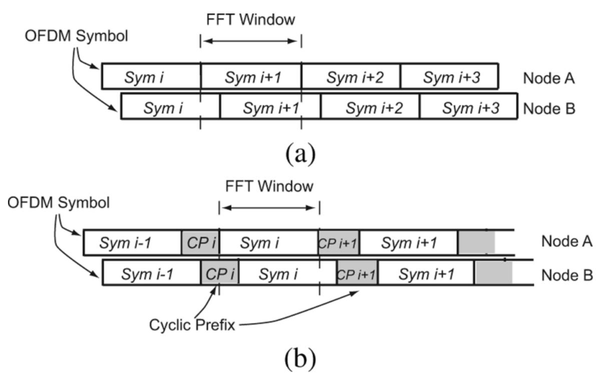


Fig. 4. Misaligned OFDM transmissions. (a) Loss of orthogonality due to misaligned symbols. (b) Retaining orthogonality with a proper size CP.

sender performs an iFFT to convert the frequency domain presentation to N time-domain samples that can be sent over the air. The time needed to transmit these N samples is usually called the FFT period, which is equal to $\frac{N}{B}$ seconds. Thus, given a fixed channel width, a larger N means a longer FFT period. Then, at the receiver side, the signal can be converted back to the frequency domain using the FFT, where each subcarrier can be demodulated independently.

When OFDM is used as a multiaccess technology where multiple stations share the same channel, symbol timing alignment will be a critical issue. As shown in Fig. 4(a), if OFDM symbols from two nodes misalign, the receiver may not be able to pick up an FFT window containing the same samples across all senders. Orthogonality will be lost, and signals from both nodes will cause mutual interference. To ensure perfect symbol alignment, a multiaccess technology called OFDMA has been proposed for OFDM cellular networks like WiMAX and LTE. OFDMA requires all mobile stations to maintain tight timing synchronization with the cellular base station (usually hundreds of nanoseconds). It requires a complex ranging scheme to measure the propagation delay and fine-tune each mobile station's timing offset at the sample level granularity.

OFDM further has a built-in robustness mechanism called the cyclic prefix (CP) [25] to guard against symbol misalignment due to multipath echoes. Each OFDM symbol is prefixed with a copy of the symbol's tailing samples so that the receiver can still find a proper FFT window as long as the misalignment is within a CP length [Fig. 4(b)]. CP is intrinsic to any OFDM system; in 802.11, the CP-to-symbol length ratio is 1:4 (0.8 μs to 3.2 μs).

C. Fine-Grained Channel Access in WLAN

We propose to use fine-grained channel access to improve throughput efficiency in a high-data-rate WLAN. We divide the channel width into appropriately sized subchannels commensurate with the PHY data rate and typical frame size, and further use OFDM on the whole channel to avoid wasting bandwidth on guard bands. The fundamental challenge with this approach is coordinating random access among multiple distributed and asynchronous nodes in a WLAN [potentially with multiple access points (APs)], without resorting to cellular-style tight timing synchronization.

Because coordination in a WLAN is distributed and decentralized in nature, it is impractical to have OFDMA-style global time synchronization. Not only would it introduce a great deal of system complexity, it would also likely require new hardware functionality beyond the current or emerging 802.11 standards. Furthermore, OFDMA does not support random access and hence cannot be used directly in a WLAN.

Instead, we should use existing 802.11 coordination mechanisms, such as carrier sensing and broadcast, to establish a rough symbol alignment among concurrent senders. We can leverage OFDM's intrinsic CP mechanism and lengthen it to suit the alignment scale, and further use a longer symbol length to maintain the same CP-to-symbol ratio. This approach calls for a new OFDM architecture specially designed for distributed coordination.

Having a longer symbol length, however, does have a negative impact that makes a conventional time-domain backoff scheme very inefficient. For example, if we can only guarantee a 10-μs symbol alignment under current 802.11 coordination schemes, we will need a 40-μs symbol length to keep the same guard-time overhead ratio. The reserved time-slot for backoff, which has to be at least one OFDM symbol in length, will now increase proportionally. This raises another technical challenge: We need a new efficient MAC contention mechanism and a new backoff scheme. All these are necessary conditions for fine-grained channel access in high-data-rate WLANs.

III. FICA Design

FICA is a cross-layer design that enables fine-grained channel access in high-rate wideband WLANs. It is based on OFDM and divides a wideband channel into a set of orthogonal subchannels, which different nodes can contend for individually. For the sake of clarity, we first assume a scenario consisting of a single access point and multiple mobile stations. We show later that FICA can be extended to the multi-AP case. Also, while in the following discussion we primarily use a 20-MHz channel as an example to explain the operation of FICA, we note that FICA is a scalable design that works for arbitrarily wide channel bands.

Fig. 5 illustrates the basic uplink media access scheme for FICA. In this example, the channel is divided into just four subchannels, and each subchannel contains a number of subcarriers.

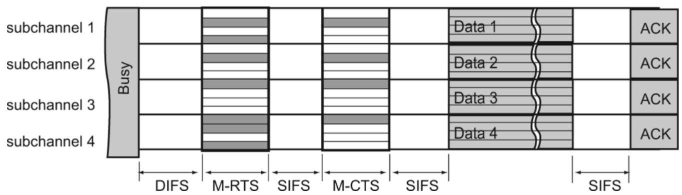


Fig. 5. FICA uplink media access with four subchannels per channel.

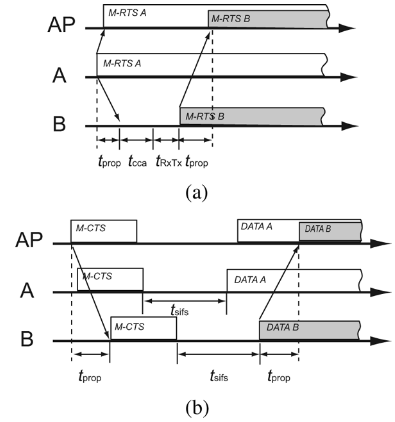


Fig. 6. Worst-case symbol timing misalignment of two concurrent transmissions received at the AP in a CSMA WLAN. (a) Symbol timing misalignment when coordination uses carrier sensing. (b) Reference broadcast synchronizes the senders better, and timing misalignment is much tighter.

FICA follows the basic scheme of CSMA. A new transmission opportunity appears only when the whole channel is idle. Then, all stations try to contend for different subchannels after the channel is sensed idle for a certain amount of time (DIFS). At this time, all nodes will transmit a special RTS signal simultaneously.[1] This RTS signal is a specially designed OFDM symbol, called *Multitone RTS* (M-RTS) (see Section III-C), in which each node embeds its contention information in a set of subcarriers for each subchannel it intends to access. All M-RTS signals are resolved at the AP, and the AP will broadcast the contention results in a corresponding *M-CTS* OFDM signaling symbol. Then, only the nodes assigned subchannels will use them for data transmissions; note that a node may contend for multiple subchannels based on its instantaneous traffic demands. The AP will then generate an acknowledgment on each subchannel where a data frame has been successfully decoded.

Downlink transmissions follow similar steps, but the AP will initiate an M-RTS signal, and receiving stations may return an M-CTS. However, since FICA does not use random time backoff, it needs to separate uplink and downlink transmissions; otherwise, collisions would happen under bidirectional traffic. FICA does so by assigning different DIFS times to uplink and downlink transmissions, described further in Section III-D.

We now present an analysis of the symbol timing misalignment problem in a CSMA-based WLAN. Based on this analysis, in Section III-B we describe the FICA PHY structure. We then describe how FICA achieves *frequency domain contention and backoff* in Section III-C. We finish by discussing several related design issues in Section III-F.

A. Symbol Timing Misalignment in WLANs

In a WLAN, transmissions from distributed nodes are coordinated based on carrier sensing and overhearing broadcast frames (e.g., RTS/CTS). Unlike conventional MACs that use these mechanisms to avoid simultaneous transmissions, FICA exploits simultaneous transmissions to enable concurrent access from different nodes but in orthogonal subchannels.

In FICA, carrier sensing coordinates the transmissions of M-RTS. After they sense an idle channel for a fixed DIFS time, two nodes A and B may transmit their M-RTS symbols simultaneously. However, since there is always a delay to sense the channel and for a signal to propagate from one node to another, these two M-RTS symbols cannot be transmitted and received at exactly the same time. Fig. 6(a) shows the worst-case analysis of the symbol timing difference of two such M-RTS symbols received by the AP. Assume A senses the channel idle for $t_{\rm difs}$ seconds, and A transmits M-RTS first. It will take at most $t_{\rm prop}$ for the signal to arrive at node B. However, it may take B at least $t_{\rm cca}$ time to reliably sense the busy channel. If the signal arrives at node B after B starts a sensing slot, B may not be able to assess a busy channel.[2] Thus, after the sensing slot, B still declares an idle channel and sends its M-RTS as well. The radio signal of B's M-RTS is actually emitted after $t_{\rm RxTx}$. It will take at most another $t_{\rm prop}$ for the M-RTS to arrive at the AP. The total misaligned time is characterized by (5). The bound is tight in the worst case

$$t_{\rm err\_rts} \leq t_{\rm cca} + t_{\rm RxTx} + 2 \times t_{\rm prop}. \tag{5}$$

If we use the values of the parameters listed in Table I, we find that $t_{\rm err\_rts}$ can be as large as 11 μs.

If coordination is performed by overhearing a broadcast frame,e.g., M-CTS or DATA, the timing misalignment can be shorter because the two senders are synchronized better by a reference broadcast [6]. Fig. 6(b) shows the worst case of symbol timing misalignment of data frames after receiving an M-CTS broadcast. Assuming the jitter for a local timer can be neglected since the waiting time is usually very small (a few microseconds), the timing misalignment is bounded by twice the propagation time, as shown in

$$t_{\rm err} \leq 2 \times t_{\rm prop}. \tag{6}$$

Using the parameter values in Table I, $t_{\rm err}$ is about 2 μs in a WLAN.

B. PHY Architecture

Based on the analysis in Section III-A, FICA needs to provide a guard time sufficiently long to handle the symbol timing misalignment in a WLAN based on carrier sensing (11 μs) and broadcasting (2 μs).

We further need to include an additional 800 ns guard time to account for the typical spreading delay in indoor environments [1].

[1] If a node senses other transmissions during this DIFS, it will give up contention and wait until the medium becomes idle again.

[2] This case depends on the way the node implements carrier sensing and threshold setting. Here we consider only the worst case.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE II
OFDM SYMBOL TIMINGS IN FICA

Parameter	Value
$N_{\text{fft\_data}}$	256 points
$t_{\text{fft\_data}}$	12.8 μs
$N_{\text{fft\_mrts}}$, $N_{\text{fft\_mcts}}$	512 points
$t_{\text{fft\_mrts}}$, $t_{\text{fft\_mcts}}$	25.6 μs
$t_{\text{long\_cp}}$	11.8 μs
$t_{\text{short\_cp}}$	2.8 μs
$t_{\text{data\_sym}}$	15.6 μs
$t_{\text{mrts\_sym}}$	37.4 μs
$t_{\text{mcts\_sym}}$	28.4 μs

We design two guard time sizes tailored to each coordination situation: a long CP of 11.8 μs and a short CP of 2.8 μs. The long CP is attached to M-RTS only, while a short CP is attached to every M-CTS, DATA, and ACK OFDM symbols, and therefore is the major overhead of concern.

To amortize the short CP overhead, we need a longer OFDM symbol for data. Recall from Section II-B that a longer OFDM symbol is achieved by applying a larger FFT size N. Although in theory one can choose any large FFT size N, there are a few practical considerations. First, a large N requires more computational power to calculate the transform since the complexity of FFT is $O(N \log(N))$. Second, with large N, the adjacent subcarriers are spaced very closely. Therefore, it is more sensitive to the frequency offsets of different nodes. Since multiple nodes may always have some small frequency offset (a few hundred Hertz as we show later), the adjacent subcarriers should have enough separation to accommodate such an offset. In practice, subcarriers are usually separated by at least 15 kHz [2], [3].

In FICA, we choose the FFT size of the DATA OFDM symbol to be 256 points in a 20-MHz channel (subcarrier width is 78.12 kHz). Its FFT period is 12.8 μs, yielding a cyclic-prefix overhead of 18%, which is comparable to the CP overhead of 802.11 (20%). ACK symbols have the same structure as DATA symbols. Since the M-RTS uses the long CP, it should contain as few symbols as possible. Thus, we choose a larger FFT size (512 points) for M-RTS to embed all necessary control information, which otherwise would require multiple OFDM symbols for a smaller FFT size. FICA applies the same FFT size to the M-CTS for format consistency. Table II summarizes the detailed time parameters of the OFDM symbol structure in FICA.

In principle, FICA can allocate each subcarrier independently to provide maximal flexibility. In practice, however, it is difficult for a node to use only a single subcarrier to transmit data for two reasons. First, using only one subcarrier suffers from frequency selective fading: If that subcarrier encounters deep fading, all data will be lost. Thus, it is essential for a wireless PHY to code across multiple subcarriers to achieve a spectrum diversity gain [17]. Second, although a preamble may be used for channel estimation and compensation, the wireless channel may change during the period of data transmission. It is essential for the receiver to track the changes in the wireless channel. This tracking is typically done by adding an additional training subcarrier (pilot) along with other data subcarriers.

Therefore, following typical practice, FICA groups a set of subcarriers into a subchannel and uses it as the basis for channel access.[3] Each subchannel contains 16 data subcarriers and one pilot subcarrier. Thus, a single subchannel in FICA is 1.33 MHz wide with a 6% pilot overhead, which is comparable to 802.11 (7%). With different modulation modes, the PHY data rate of a single subchannel can range from 512 kb/s (BPSK, 1/2 coding on each subcarrier) to 20.25 Mb/s (64 QAM, 5/6 coding and four spatial multiplexing streams on four antennas).

TABLE III
EXAMPLE PHY DATA RATES IN FICA VERSUS 802.11n

Configuration	FICA (Mbps)	802.11n (Mbps)
20MHz channel	71.8	72.2
40MHz channel	148.7	150
40MHz channel, 2xMIMO	297.4	300
40MHz channel, 4xMIMO	594.8	600

With a 1.33-MHz subchannel, a 20-MHz 802.11 channel contains 14 orthogonal subchannels. FICA uses the remaining spectrum as guard bands separating adjacent wideband channels. Note that it is also straightforward for FICA to support wider band channels, e.g., 40–100 MHz or wider. To support a 40-MHz channel, for instance, we simply double the FFT size for all OFDM symbols; since our sampling rate is also doubled, the symbol period does not change. When using a 40-MHz channel, we can save the guard bands that would be necessary to separate two 20-MHz channels. Thus, we can have 29 orthogonal FICA subchannels with a 40-MHz channel. Table III shows some example PHY data rates of FICA and compares them to 802.11n. As we will show later, although FICA has slightly lower data rates than 802.11n, the effective throughputs are actually dramatically higher due to diminished overheads.

C. Frequency Domain Contention

FICA uses the M-RTS/M-CTS signal exchange to avoid subchannel collisions. M-RTS/M-CTS use simple *binary amplitude modulation* (BAM) to modulate a single bit on each subcarrier. Specifically, BAM uses ON–OFF keying that maps a binary "0" to zero amplitude in a subcarrier, and it uses a random complex number on the unit circle ($e^{j\theta}$) for a binary "1." Receivers can easily detect BAM symbols by comparing the energy on a subcarrier against a threshold, without the need to recover the symbol phase.

The basic idea of frequency domain contention works as follows. In M-RTS, a group of K subcarriers are allocated for each subchannel, called a *contention band*. When a node contends for a subchannel, it will first randomly pick a subcarrier from the contention band and send a signal "1" using BAM. At the AP side, there may be multiple M-RTS signals superposed, and the AP may detect multiple ones on different subcarriers of the contention band. The AP then can arbitrate a winning node by selecting a subcarrier based on some predefined rules, e.g., the one with highest frequency. Then, the AP sends the identification of the winning subcarrier for each subchannel in an M-CTS signal. After receiving the M-CTS, each node compares the subchannel allocation broadcast by the AP to the subcarrier

[3]The grouping of subcarriers may be arbitrary, and not necessarily contiguous. In our current implementation, however, we only group adjacent subcarriers into a subchannel.

it has transmitted for the subchannel. If they match, the node will transmit data symbols on that corresponding subchannel after a SIFS delay.

There are a few issues that need to be considered.

What if two nodes transmit on the same contention subcarrier? It is quite possible for two nodes to choose the same random number and transmit on the same contention subcarrier. Thus, their transmitted energy is additive at the AP side. If the AP picks this subcarrier as the winner, a collision will happen since both nodes will consider themselves the winner and send data symbols on the same subchannel simultaneously. It is also possible (although the possibility is small) that two nodes' signals are destructive and cancel each other, so the AP may not be able to detect the transmissions. In this case, though, the nodes involved will conclude that they were not assigned the subchannels they requested.

How large should K be? Clearly, a large K will have fewer collisions in a subchannel. A larger K, though, means more subcarriers are used and result in a larger signaling overhead: More subcarriers need a larger FFT size and therefore a longer FFT period. In FICA, we set $K = 16$, the initial contention window size in 802.11.

Who is responsible for returning the M-CTS? Any node that overhears M-RTS transmissions can arbitrate subchannel access. Simultaneous transmissions of M-CTS from different nodes are also allowed since, in a single broadcast domain, these M-CTS symbols are likely to have the same contention results and their energy is additive on each subcarrier.

However, it is still reasonable that only the potential receivers should return an M-CTS for an M-RTS based on the following considerations: 1) the irrelevant nodes can use power-save mode since it is unfair for them to spend power responding to M-RTS signals not for them; 2) nonreceiving nodes may unnecessarily create more exposed terminals if they reply with an M-CTS and disrupt transmissions that could otherwise happen without interference.

How can we specify receivers in an M-RTS? It is nontrivial since multiple nodes may transmit M-RTSs simultaneously to different receivers and the receiver information may be mixed. FICA resolves this issue by using a membership vector of m subcarriers to represent receiver information in the M-RTS. Before transmitting an M-RTS, a node will hash the receiver's ID into a value between 0 and $(m-1)$. Then, the corresponding subcarrier will carry a "1" bit. Any node receiving an M-RTS will check if the subcarrier corresponding to its ID has been set. If true, it should return an M-CTS. It is possible that multiple nodes have hash value collisions. However, with a reasonably sized m—we use 40 bits in the current design—the number of station collisions should be small: 2–3 in a typical WLAN setting where one AP may be associated with dozens of stations [14].

1) M-RTS/M-CTS Format: Table IV summarizes the subcarrier allocation for the M-RTS and M-CTS symbols. We explain the M-RTS/M-CTS format using a 20-MHz channel as a concrete example. M-RTS/M-CTS use a 512-point FFT in a 20-MHz channel (for wider channels, a proportionally longer FFT point size is used).

Tag: The tag band contains a predefined binary sequence to identify the type of this OFDM symbol. The tag band is interleaved on selected subcarriers whose indices are multiples of four. Thus, the tag band creates a repeating pattern in time-domain samples. The receiver can use this pattern to detect the symbol using correlation. In a 20-MHz channel, the tag band contains 32 subcarriers. For M-RTS, the tag sequence starts with zero, $\{0, 1, 0, 1, \dots, 0, 1\}$, while for M-CTS the sequence becomes $\{1, 0, 1, \dots, 0, 1, 0\}$.

TABLE IV
NUMBER OF SUBCARRIERS ALLOCATED FOR THE M-RTS AND M-CTS SYMBOLS IN A 20-MHZ CHANNEL

M-RTS		***M-CTS***	
Tag	32	Tag	32
Contention band	16×14	Resolution band	24×14
NAV	64	NAV	64
Receiver band	160	Reserved	48
Guard band	32	Guard band	32

Contention Band and Resolution Band: For M-RTS, FICA allocates 16 subcarriers to each subchannel for contention. Thus, a total of 224 subcarriers are allocated for a contention band for all 14 subchannels arbitrated using an M-RTS symbol.

Contention results are embedded in the *resolution band* in M-CTS. The resolution band occupies more subcarriers compared to the contention band in the M-RTS. There are 24 subcarriers for each subchannel. However, since BAM is less robust compared to phase-shift keying modulation (e.g., BPSK), we improve BAM's reliability by replicating on multiple subcarriers and using soft decoding. It can be shown that, by replicating twice, BAM can achieve the same performance of BPSK [24]. In FICA, we replicate a bit on four subcarriers when we require high reliability. Thus, with four-time replication, we can encode 6 bits (24/4) to represent the resolution results for each channel.

Currently, 6 bits can represent the winner number (1–16) and two other special states. If these 6 bits are all zero, it means the subchannel is not used (e.g., no node has contended for this subchannel). Alternately, if the bits are all ones, the subchannel is reserved by the receiver; thus, no transmission is allowed on this subchannel in this contention period.

Receiver Band: The receiver band contains the hash vector for intended receivers across all subchannels. It contains 40 bits, and each bit again is replicated on four subcarriers for reliability, for a total of 160 subcarriers.

Network Allocation Vector (NAV): The NAV band encodes the expected number of OFDM symbols in the following data frame. It contains 64 subcarriers and, also due to four times replication, can modulate 16 bits. There could be several ways to encode the data symbol number in the NAV band. One simple method is to use each single bit to present a certain number of OFDM symbols. Thus, 16 bits can present numbers from 3 to 48, with a step of 3. As we discuss later, such a range is sufficient for FICA. In an M-CTS, only the highest bit is echoed back, and all nodes that overhear the M-CTS should defer their contention for a corresponding period.

2) Frequency-Domain Backoff: In FICA, nodes will choose how many subchannels to request via contention based on their traffic requirements. If its traffic requirements are light, a node may request fewer subchannels, while a heavily loaded node may contend for as many as all subchannels for most efficient

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.


```
Initialize Cmax:
Cmax = Ctotal;

Update1: Reset to max
if collision detected in any subchannel then
    Cmax = max(Cmax/2, 1);
else
    Cmax = Ctotal;
end if

Update2: AIMD
if p% subchannels have collisions and (p > 0) then
    Cmax = max(Cmax × (1 − p/100), 1);
else
    Cmax = min(Cmax + 1, Ctotal);
end if
```

Fig. 7. Pseudocode of the frequency domain backoff algorithms. *Update1* emulates the behavior of 802.11 binary exponential backoff. *Update2* uses an AIMD strategy.

communication. However, if there are many nodes in a network contending for many subchannels, the collision avoidance mechanism provided by M-RTS/C-RTS may not be sufficient to represent all contenders. Using multiple M-RTS symbols can further scale collision avoidance, but at the cost of more signaling overhead as described above.

Instead, we use a novel *frequency-domain backoff* scheme to scale FICA in a heavily contended dense network. The basic idea is to control the maximum number of subchannels that one node can access based on the observed collision level. It is similar to existing widely used congestion control mechanisms. Each node maintains a state variable, $C_{\max}$, which controls the maximum subchannels the node can access in the next transmission opportunity. Thus, when the channel is idle for DIFS, a node may pick up to n subchannels to contend for, where $n = \min(C_{\max}, l_{\text{queue}})$ and l_{queue} is the number of fragments in the node's local sending queue.

The maximum subchannel count $C_{\max}$ is updated based on the contention situation on the channel. There can be multiple update strategies. For example, we can emulate the behavior of BEB used in 802.11. When a collision is detected on a subchannel that a node has transmitted on, it will reduce $C_{\max}$ by half. Once all transmissions are successful, the node resets $C_{\max}$ to the total number of subchannels.

Alternatively, we can use an additive increase/multiplicative decrease (AIMD) strategy. Assume after transmission, a node detects that p% of channels have collisions. The node then decreases $C_{\max}$ by p% (multiplicative decrease). This reaction is reasonable since p reflects an estimation of the contention level in all subchannels. When all accesses to subchannels succeed, the node increases $C_{\max}$ by one (additive increase).

It is straightforward to prove that both strategies converge if all contending nodes are within a single broadcasting domain. Fig. 7 shows the pseudocode of the adaptive $C_{\max}$ adjustment algorithms. Later, we evaluate these two strategies and find that AIMD is slightly better than the reset-to-max strategy.

D. Multiple Access Points and Two-Way Traffic

Until now, we have only concentrated on uplink access. Downlink access follows the same process with roles reversed. In this case, the AP will send out an M-RTS. In FICA, it is possible for the AP to transmit simultaneously to multiple clients with a single transmission burst (but different frames to different subchannels). The receiver IDs are encoded in the *receiver band* of M-RTS. All receiving clients should return M-CTS to the AP. It is necessary for the downlink transmissions to go through such a contention process since in practice there can be multiple APs located nearby on the same wideband channel. Thus, these APs should use M-RTS and frequency-domain backoff to contend for each subchannel for transmission. The contention result is resolved by the receiving stations and fed back to each AP by M-CTS broadcasts.

There is another issue in the presence of two-way traffic. Assume that an AP and one client have frames to exchange. Since FICA does not use time backoff by default, both nodes may send an M-RTS simultaneously. This may cause a failure if there are no other nodes that can send back an M-CTS. Furthermore, even if there is a third node that receives these M-RTSs and sends an M-CTS back with the contention resolution, it is still possible that both the AP and the client have been granted some winning subchannels. Therefore, they will both transmit data simultaneously, but neither of them can receive its frame due to the simplex radio used.

To address this issue, FICA separates uplink and downlink traffic by assigning different DIFS times to the AP and stations. The one with a short DIFS has priority to access the channel by sending an M-RTS earlier. To ensure fairness between uplink and downlink traffic, we use a simple dynamic DIFS assignment strategy. We assign a fixed DIFS time to all mobile stations, and the AP has two different DIFS time settings. One is shorter than the DIFS of mobile stations, and the other one is longer. The AP chooses a DIFS time based on the following rules: 1) once an AP accesses the channel with a short DIFS, it will use the long DIFS for its next access; and 2) if an AP receives an M-RTS from stations, it will use a short DIFS for its next access. Note that this simple strategy ensures the fair interleaving of uplink and downlink traffic, but not the fairness among all nodes as 802.11 currently does. In effect, it gives the AP many more chances to transmit—which may be an appropriate strategy given the asymmetrical nature of WLAN application workloads. It is an interesting open question of what sort of fairness is best provided in FICA and remains future work.

E. Analysis

We analyze the performance of FICA in this section. We use the same values of t_{sifs} and t_{difs} as in 802.11. Similar to 802.11n, the preamble design of FICA depends on the transmission mode. It requires three OFDM symbols for single and 2× MIMO and four for 4× MIMO (we will detail the preamble design in Section V). Using the three-symbol preamble as an example, $t_{\text{preamble}} = 46.8\ \mu$s, and counting another OFDM symbol for the ACK, i.e., $t_{\text{ack}} = 15.6\ \mu$s, the subtotal per-access MAC overhead of FICA is 186.2 μs. Note that although FICA uses the M-RTS/M-CTS handshake, the overhead is comparable to that of 802.11 (160 μs with *minimal* contention window) due to the use of the PHY signaling mechanism. The following equation gives a simple model for FICA's access efficiency:

$$\eta_{\text{fica}} = \frac{t_{\text{data}}}{t_{\text{difs}} + t_{\text{rts/cts/ack}} + t_{\text{preamble}} + 3 \times t_{\text{sifs}} + t_{\text{data}}} \quad (7)$$

where $t_{\text{rts/cts/ack}} = t_{\text{mrts}} + t_{\text{mcts}} + t_{\text{mack}}$.

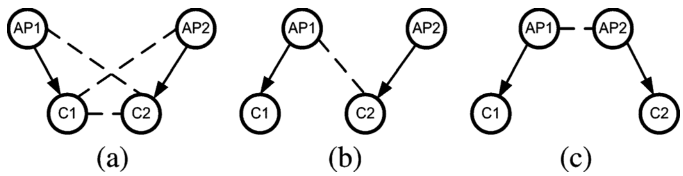


Fig. 8. Hidden terminal and exposed terminal cases. Arrows present the communication links. Dash link means the two nodes can hear each other. (a), (b) Hidden terminals. (c) Exposed terminals.

Based on this model, we find 40 DATA OFDM symbols are needed to achieve an efficiency ratio of 80%. We use this number as a rule of thumb to guide FICA transmission on each subchannel. Based on current data rate, a FICA node fragments the upper-layer frame into segments. Each segment contains 40 DATA OFDM symbols or 400 B, whichever is larger. Then, the node may transmit one segment on each subchannel after it wins the contention in these subchannels. Note that a fragment may also need a header to identify itself—containing the sender and receiver addresses, length, and sequence control—as well as a CRC checksum. This additional header overhead is minor, up to 5%.

F. Discussion

We end the description of FICA with a few additional points of consideration. For the issues raised, we broadly describe potential approaches for addressing them, but in general leave an exhaustive discussion as future work.

Multiple Contention Domains and Hidden Terminals: Until now, we have only focused on FICA within a single broadcast domain. How does FICA operate in a network with hidden terminals, and therefore when there are multiple overlapped contention domains? Fig. 8(a) and (b) illustrates two typical hidden terminal cases. Two APs hidden to each other send frames to two stations. As hidden terminals cannot be coordinated by carrier sensing, they may cause collisions. However, since FICA involves M-RTS/M-CTS handshakes (similar to RTS/CTS in 802.11), this issue can be mitigated. For example, in Fig. 8(a), if AP1 sends out M-RTS first, M-CTS from C1 will block AP2 from a contending channel, and therefore avoid collisions at C1. In Fig. 8(b), AP2 may still be able to send M-RTS to C2. However, since C2 cannot return a proper M-CTS, AP2 will soon detect M-RTS collision at C2 and abort transmission. In this case, the frame reception on station C1 is not affected. To prevent persistent M-RTS collisions, once a node misses an M-CTS or it receives an invalid M-CTS after it sends an M-RTS, the node should wait for a random time before sending an M-RTS again. This random waiting time is necessary to desynchronize the hidden terminals and reduce the collisions.

There are also cases that these two hidden terminals are synchronized, e.g., by a transmission of C2, and thus they may send M-RTS simultaneously. Then, there is a chance that a contending node may receive inconsistent resolution results from M-CTSs from different nodes. We will show that FICA can still work well when facing these inconsistent resolutions. For example, in Fig. 8(b), assume both AP1 and AP2 are contending subchannel i, and the contention numbers of AP1 and AP2 are 3 and 1, respectively. Since C1 cannot hear AP2, it will report, in its M-CTS, that the winner is 3 (AP1), while C2 will report that the winner is 1 (AP2).[4] In this case, AP2 will find itself the winner of the subchannel and send the frame. However, AP1, when receiving the combination of M-CTS from both C1 and C2, will conclude it is not the winner (Section III-C), and therefore abort transmission. Note that NAV in the combined M-CTS (no less than the maximum transmission time of the two frames) will prevent AP1 from colliding AP2's transmission. In general, only when the node is the winner in all contention domains in which it participates should it be allowed to access the subchannels (implying all returned M-CTSs contain the same resolution results for intended subchannels). We believe such behavior is reasonable and follows the general principle of wireless system designs.

With frequency domain contention and backoff, FICA may also mitigate the exposed terminal issue as shown in Fig. 8(c). FICA enables the two APs, which are exposed to each other in 802.11, to send M-RTSs and their following data frames to both stations simultaneously, and therefore improves the overall network performance.

Finally, we note that current FICA is mainly designed for WLAN where AP infrastructure is deployed. Applying FICA in a pure ad hoc network, where network topology may change very dynamically, may impose additional challenges. For example, frequency synchronization will become much more difficult in such a dynamic environment. However, a systematic study of FICA in an *ad hoc* setting may go beyond the scope of this paper. We put it as our future work.

Multiuser Diversity: FICA also enables an opportunity to exploit multiuser diversity in WLANs [25]. When a node chooses subchannels to access, it may also consider the quality of each subchannel. Moreover, heterogeneous modulation methods can be applied to different subchannels to match the conditions on that specific channel band. There is already much research on resource allocation for multiuser diversity in both single- and multichannel cases [7], [12], [15], [21], and applying these ideas in a distributed system like FICA remains interesting open challenges.

Backwards Compatibility: It is also possible for FICA to coexist with current 802.11. Since FICA is still based on CSMA, FICA nodes will defer if they sense a transmission of 802.11 nodes, and vice versa. It is also possible to retain fairness among these two types of nodes. For example, since FICA has a fixed access pattern with the M-RTS/M-CTS handshake, it is possible to choose an appropriate DIFS time (e.g., equal to half of the $CW_{\min}$ of 802.11) for FICA to be fair to 802.11 nodes with random time-domain backoff.

Cellular Versus WLAN Environments: Finally, we note that the mechanisms designed in FICA may unlikely be applicable in cellular networks because coverage of a cellular base station is large (e.g., a few kilometers). Thus, the propagation time is large as well, at least an order of magnitude larger than that in a WLAN. Consequently, even using broadcasting, the synchronization accuracy is too coarse (e.g., several tens of microseconds). Current OFDM-based WWANs already employ a relative long cyclic prefix (4.69 μs) and FFT period (66.67 μs) to handle a large delay spread due to multipath fading in the wide area. A

[4]We assume the one with smaller contention number wins.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

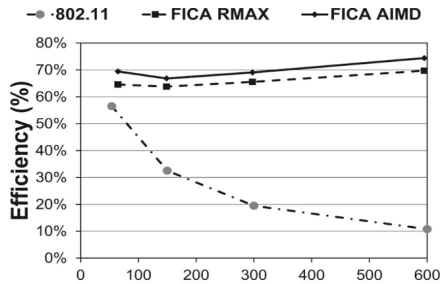


Fig. 9. Efficiency ratio of 802.11 and FICA with different PHY data rates. No frame aggregation is enabled.

low-precision synchronization method in OFDMA will further enlarge the symbol FFT size, adding substantial engineering complexity to control frequency offsets and undermining the ability to handle Doppler effects in a mobile environment.

IV. Simulation

We have implemented the FICA protocol on ns-3 to study its performance in large-scale wireless networks and compare its performance with 802.11n. We use a simple PHY model where only collisions will cause frame reception failures. This is a reasonable simplification here as our focus is the performance of the MAC design. We defer the evaluation of FICA PHY design in Section VI. We divide the whole channel into subchannels, and each node can transmit data on any subchannels following the FICA protocol as described in earlier sections. Our performance study is primarily under a single AP network with varying number of stations, except for the last experiment, which contains two APs. We also focus on only the uplink transmissions (the downlink behavior in this setting is analogous), and we apply various traffic patterns in a wide 40-MHz channel with high data rates. For 802.11n, we simulate MAC Service Data Units (MSDU) aggregation, which is the most efficient aggregation method defined in 802.11n [1].

No Aggregation: In this scenario, we first disable the frame aggregation of 802.11n as a lower bound. Fig. 9 shows the throughput efficiency of 802.11n and FICA with two different frequency-backoff schemes: AIMD and Reset-to-Max (RMAX) (Section III-C.2). The scenario simulates 10 concurrent nodes where each node transmits UDP traffic corresponding to 1/10 of the PHY data rate with a frame size of 1500 B. As expected, with 1500-B frames, current 802.11a/g rates only provide around a 56% efficiency ratio, and this ratio decreases rapidly with the increase of the PHY data rate. However, by enabling fine-grained channel access, FICA can achieve a much higher efficiency ratio in the same situation. This benefit is because different stations can access different subchannels simultaneously. Thus, the per-access MAC overhead is amortized among all concurrent nodes. Also, we find that FICA AIMD has slightly better performance than FICA RMAX. As we will see in subsequent experiments, FICA AIMD consistently performs better. We hypothesize that this is because FICA AIMD adjusts $C_{\max}$ much smoother compared to FICA RMAX. However, a deep analysis on the optimal frequency-domain backoff strategy remains future work.

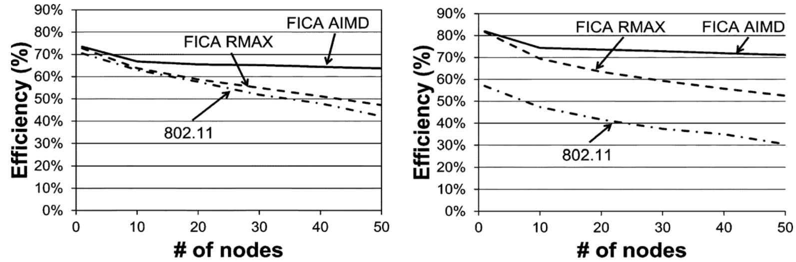
This scenario is the worst case for 802.11n. We show this case to demonstrate how significant the MAC overhead can be at high PHY data rates, and that techniques like FICA or frame aggregation are indeed necessary for efficiency.

Full Aggregation: Here, we show the best case of 802.11n with frame aggregation. In this experiment, all nodes are saturated so that frame aggregation can work most efficiently. Fig. 10 shows throughput efficiency with different numbers of contending nodes at two PHY data rates, 150 and 600 Mb/s, respectively. In both cases, the efficiency of 802.11n has been significantly improved due to frame aggregation. Since all nodes are saturated, the aggregation level is very high: 12 frames (or 18 kB) on average.

FICA still has slightly better performance than 802.11n even in this case, though, because FICA has slightly fewer collisions compared to 802.11n. To understand why, consider the operation of frequency domain contention. When there are many stations contending for a subchannel, if two stations happen to pick up the same subcarrier to send their signals, it does not necessarily result in a collision. A collision occurs only when the collided subcarrier is also chosen as the winner as nodes contend for subchannels. In the next contention period, all stations will pick a different random number again. This situation is unlike time-domain backoff used in 802.11: When two stations pick the same backoff slots, they will eventually collide with each other.

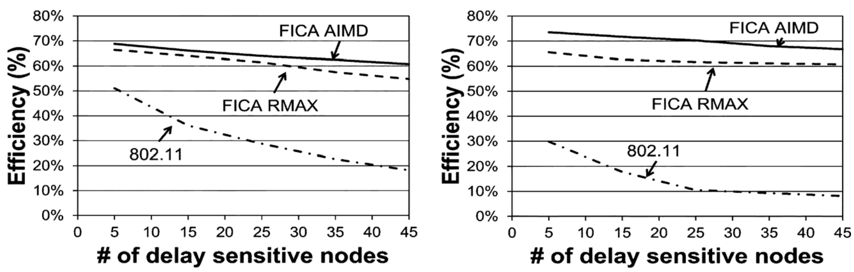
Mixed Traffic: Here, we evaluate situations in between the two extremes. We have five saturated stations that always have full-sized frames to transmit. In addition, there are a variable number of nodes that have small but delay-sensitive traffic representing, for instance, video conferencing or Web browsing. We choose the load of this delay sensitive traffic uniformly from 800 kb/s to 5 Mb/s, and the packet size from 800–1300 B. Fig. 11 shows the efficiency results of this scenario as a function of the number of delay-sensitive nodes. With a few delay-sensitive nodes, the throughput efficiency of the network is significantly reduced for 802.11n. Since the delay sensitive flows cannot be aggregated, their access to the channel is much less efficient. Thus, the overall channel utilization is reduced. However, with FICA, such nodes can request access to fewer subchannels, leaving the other subchannels for use by other nodes. Consequently, the overall network efficiency remains at a high level, improving upon 802.11 from 35% up to six times better at the high PHY data rates.

By synchronizing their transmissions, FICA allows multiple users accessing the channel simultaneously. However, if the segments transmitted on different subchannels have different sizes, there will be a waste of air time for the subchannels with short segments since a new transmission opportunity appears only when all transmissions have finished. Fig. 12 evaluates this effect. In this experiment, we have five saturated stations that send full-sized frames. Then, we enable a variable number of nodes that send small packets only. We ensure these nodes always have a small packet (64 B) to send each time a transmission opportunity occurs. Clearly, this is a worst-case situation. From Fig. 12, we see that when the number of nodes sending small packets increases, the overall network efficiency indeed reduces. For FICA, the efficiency ratio reduces from 70% (no



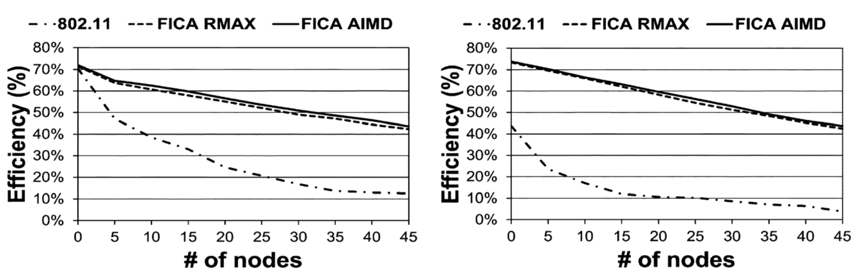
(a) (b)

Fig. 10. Full aggregation case. For 802.11, the maximal aggregated frame size is 18 kB. All nodes are saturated. (a) 802.11 PHY 150 Mb/s; FICA 148 Mb/s. (b) 802.11 PHY 600 Mb/s; FICA 594 Mb/s.



(a) (b)

Fig. 11. Mixed traffic. Five nodes are fully saturated. All other nodes have delay-sensitive traffic with a uniform distribution between 800 kb/s to 5 Mb/s. (a) 802.11 PHY 150 Mb/s; FICA 148 Mb/s. (b) 802.11 PHY 600 Mb/s; FICA 594 Mb/s.



(a) (b)

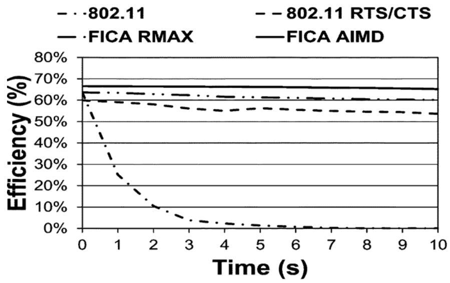
Fig. 12. Small segment size. Five nodes are fully saturated with full-sized packets. All other nodes send only small 64-B packets. (a) 802.11 PHY 150 Mb/s; FICA 148 Mb/s. (b) 802.11 PHY 600 Mb/s; FICA 594 Mb/s.

small packet node) to 43% (45 small packet nodes). However, FICA still performs much better than 802.11n (up to $8\times$) in the same situation. Therefore, we conclude FICA can effectively improve the network efficiency, while its simple design allows a practical implementation in a random access WLAN.

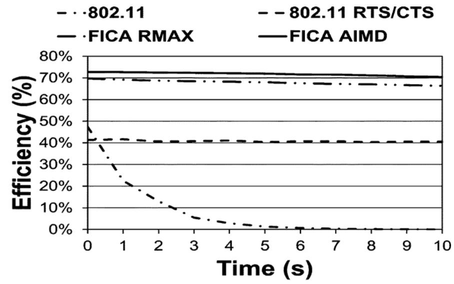
Hidden Terminals: In this experiment, we evaluate the performance of FICA in a scenario when there are hidden terminals and compare its performance to 802.11n, with RTS/CTS enabled and disabled, respectively. We have two APs, each of which connects to 10 stations. Each station can hear both APs and other stations connecting to the same AP. However, stations connecting to different APs are hidden to one another. We assume all nodes have saturated traffic. We enable frame aggregation for 802.11n and the packet size is 1500 B. We let 10 nodes of one AP transmit first. Then, every second, we start one station of the other AP, which is a hidden terminal for the first 10 nodes. Fig. 13 shows the throughput efficiency results of the network. We can see that without RTS/CTS, the efficiency drops quickly, and no packet can be successfully delivered when there are a few hidden nodes. However, the RTS/CTS handshake can effectively mitigate the hidden terminal issue in this case. Similarly, FICA also performs well due to the M-RTS/M-CTS exchange. FICA has higher performance compared to 802.11n with RTS/CTS simply because the M-RTS/M-CTS signaling has less overhead compared to RTS/CTS frames.

V. Implementation

We have implemented the basic mechanisms of FICA using Sora, a fully programmable software radio platform based on commodity general-purpose PC architectures [23]. Our FICA implementation is based on SoftWiFi, a software implementation of the 802.11a/b/g PHY/MAC [23]. In summary, we make the following modifications: 1) we change the FFT size from 64- to 256-point for DATA/ACK symbols and 512-point for M-RTS/M-CTS symbols; 2) we employ convolutional coding in each subchannel and decode data in each subchannel individually using the Viterbi algorithm; 3) we remove the random time-domain backoff in the CSMA MAC and implement the M-RTS/M-CTS handshake after the channel is sensed idle. In



(a)



(b)

Fig. 13. Hidden terminals. Ten nodes of one AP start transmissions first. Then, every second, a node of the other AP starts sending packets, which is hidden to the first 10 nodes. (a) 802.11 PHY 150 Mb/s; FICA 148 Mb/s. (b) 802.11 PHY 600 Mb/s; FICA 594 Mb/s.

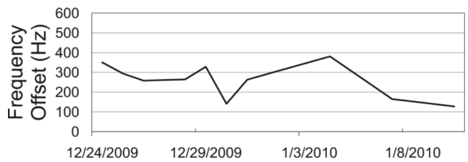


Fig. 14. Measured frequency offsets of two nodes over two weeks starting from a single calibration.

the following, we elaborate some design details and algorithms implemented in our system.

Frequency Offset Calibration: In OFDM-based multiaccess networks, the frequency offset among simultaneous senders should be controlled within a certain range. Otherwise, it will undermine the orthogonality among subcarriers since the concurrent senders are not actually transmitting on orthogonal frequency due to the offset. Thus, in FICA, all nodes should have their frequency calibrated.

Fortunately, FICA can use the AP's frequency as a reference. Every station can hear the beacon from the AP and adjust its local numerically controlled oscillator (NCO) to match the frequency of the AP. We note that frequency calibration is much simpler than time synchronization since the frequency offset does not accumulate over time. Fig. 14 shows the measured absolute value of the frequency offset of two Sora nodes over two weeks after a single calibration. The frequency offset is within a limited range of 500 Hz. This range is quite small compared to the data subcarrier width (0.63% of 80 KHz), and its impact to orthogonality can be neglected in practice. Thus, we conclude that an infrequent frequency calibration process can support FICA very well.

Frame Detection and Synchronization: Similar to 802.11, FICA exploits the periodic property of the signal to detect a valid frame. All FICA data frames are preceded with a preamble, which contains a short training symbol (STS), a long training symbol (LTS), and a signal symbol. The STS has a self-repeating pattern in the time domain, so that the receiver can detect it using auto-correlation. The LTS is used for channel estimation as discussed later. The signal symbol encodes the Physical Layer Convergence Protocol (PLCP) header using BPSK and 1/2 convolutional coding. The PLCP header contains the modulation mode used in the following DATA symbols for the receiver to set the proper demodulating parameters of the following frame, like frame size and modulation rate.

Both M-RTS and M-CTS do not have a preamble. However, they can be easily detected by determining the repeating pattern of the Cyclic Prefix. If the auto-correlation—normalized by the signal average energy—is higher than a threshold, the receiver decides an M-RTS or M-CTS is received. It will perform 512-point FFT on the following samples and demodulate the tag band using BAM to decide the symbol type. Note that it may raise one question: Since M-RTS may be sent by different node in a coarsely synchronized way, does this time misalignment affect the frame detection accuracy? The answer is no. It is because the signal received at the AP is the sum of all M-RTSs from different nodes and the summation of periodic functions are still periodic. Therefore, they can be detected using the same auto-correlation algorithm.

Channel Estimation and Tracking: After detecting a data frame, the FICA receiver uses the long training symbol to estimate the wireless channel. The channel estimation is performed on each subchannel independently. The long training symbol contains a series of known samples on each subcarrier, denoted by x_i, and the estimated channel coefficient is $h_i = y_i/x_i$, where y_i is the received sample on that subcarrier i. Although FICA requires all nodes to calibrate their frequency, there may still have residual frequency offset between the receiver and each sender. This residual frequency offset is corrected using the Pilot on each subchannel. The receiver can track the phase rotation of the pilot samples and use this information to update the channel coefficient on each data subcarrier. Note that when MIMO is enabled, FICA requires more long training symbols to train the channels between different sending antennas and receiving antennas, just like 802.11n.

Carrier Sensing: FICA removes the random time-domain backoff, but still relies on carrier sensing to coordinate the transmission of M-RTS. FICA divides the carrier sensing period, i.e., *DIFS*, into m slots. In our implementation, each slot is 5.6 μs. FICA computes the average energy during a slot and compares it to a threshold. A busy channel is assessed if the average energy is higher than the threshold. Once a busy channel is detected, the FICA node will defer its M-RTS and reset its timer. It will start counting time when the channel becomes clean again. In our implementation, we choose the energy threshold to be 5 dB higher than the noise floor, which is estimated periodically. There is one main difference between the carrier sensing method in FICA and typical 802.11. Instead of sending M-RTS after sensing m clean slots, a FICA node

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

decides to send M-RTS after $m - 1$ clean slots. However, it will still wait for one additional slot before it really sends out M-RTS. The rationale behind this design is to encourage the concurrent transmissions of M-RTS from different senders, which may have misaligned slot start-points. As discussed in Section III, the choice of the CP for M-RTS can well handle this timing mismatch.

BAM Threshold Setting: With physical-layer signal symbols (M-RTS/M-CTS), we need a proper energy threshold to reliably detect the bit modulated on each subcarrier. Since we target WLANs, where the signal-to-noise ratio (SNR) is usually high, setting such a threshold is not a difficult task [5]. In FICA, we further use a self-calibration method to facilitate this threshold setting. We use blank subcarriers (carrying a bit "0") in the M-RTS/M-CTS tag band to calibrate the noise floor. Then, we choose a threshold that is a constant multiple of the noise floor. In our experiments, we find that a threshold 2–5 dB higher than the noise floor works reliably in WLAN settings.

VI. Evaluation

In this section, we evaluate the feasibility of FICA using our prototype implementation on four Sora nodes: One serves as the AP, and the others are stations. We first demonstrate how well concurrent transmissions can be coordinated in a WLAN. We evaluate the maximal symbol timing misalignment at the receiver for two concurrent transmissions coordinated by means of broadcasting and carrier sensing. We show that even with our existing software radio implementation, we can bound the maximal symbol timing misalignment within the range discussed in Section III-A. Then, we evaluate the efficiency and reliability of detecting BAM-modulated PHY signals in M-RTS/M-CTS. Finally, we show the decoding performance of our FICA decoder for two concurrent FICA senders. We perform our experiments in the 2.4-GHz band with 802.11b/g-compatible RF front ends. Due to timing constraints, we have prestored all needed PHY frame samples on the radio control board (RCB) first. We also conduct the experiments late at night to minimize interference from other traffic in the same frequency band.

A. Symbol Timing Misalignment in a WLAN

As discussed in Section III-A, there are two ways to coordinate concurrent transmissions in a WLAN: using a reference broadcast (e.g., M-CTS) or carrier sensing on an idle channel. These two methods result in different performance requirements in aligning symbol times.

To measure the symbol timing misalignment with broadcasting, the AP broadcasts an M-CTS symbol, and two stations, when they detect the M-CTS, wait for a fixed 200 μs and return two different pseudo-noise (PN) signals. We use PN signals so that we can easily separate the two signals by cross correlation and precisely measure the time difference of the arrivals of the two signals at the AP by counting the number of samples between the two correlation peaks. Fig. 15 shows the cumulative distribution function (CDF) of 1000 tests. In over 95% of the cases, the symbol timing difference of these two transmissions is less than 1 μs, and in 99% cases, the timing difference is less than 2 μs.

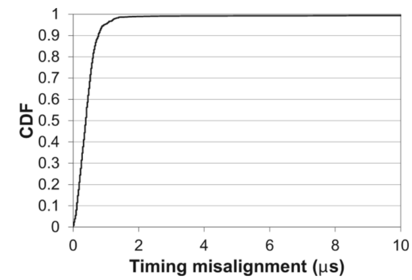


Fig. 15. CDF of symbol timing mismatch with broadcasting.

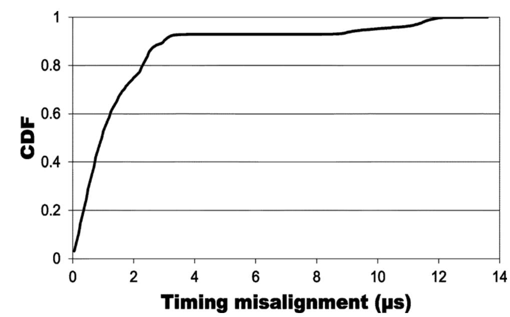


Fig. 16. CDF of timing misalignment with carrier sensing.

To characterize the maximal symbol time difference when coordinating with carrier sensing, we let two stations perform carrier sensing before they transmit their signals. When they detect the channel is clean, they will transmit two different PN signals to the AP. If one station senses a busy channel, it will cancel its transmission, and thus only one PN signal will be received at the AP. We again use cross correlation to identify the two PN signals and measure the time difference between the arrivals of the two PN signals at the AP.

Fig. 16 shows the CDF of 100 000 tests. The x-axis shows the relative delay of two stations. In 96.33% of the cases, the symbol timing difference of these two transmissions is less than 11 μs, and in 99.23% cases, the timing difference is less than 11.8 μs, confirming our timing analysis in Section III-A and the long CP design in Section III-B. We see that there is a flat zone about 5 μs from 3.65 to 8.7 μs. This reflects that two stations can transmit concurrently if their time misalignment is less than one slot. It also confirms the design presented in Section V.

B. Reliability of PHY Signaling

In this experiment, we evaluate the reliability of the PHY signaling implementation. Two nodes transmit the same M-RTS symbol simultaneously so that every subcarrier contains a superposition of the signals transmitted by the two senders. The AP detects the information on each subcarrier. Since we know what is exactly transmitted, we can detect false positives (i.e., the AP detects one on a subcarrier that should be zero) and false negatives (i.e., the AP detects zero where it should be one). We use the self-calibrated method to set detection threshold as described in Section V: We measure the maximal energy N_f on the blank subcarriers and set the threshold to be αN_f, varying α. We also experiment with different degrees of bit replication on subcarriers.

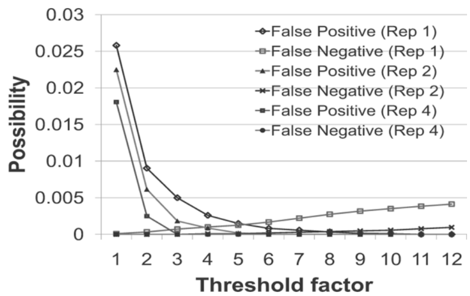


Fig. 17. Reliability of PHY M-RTS/M-CTS signaling.

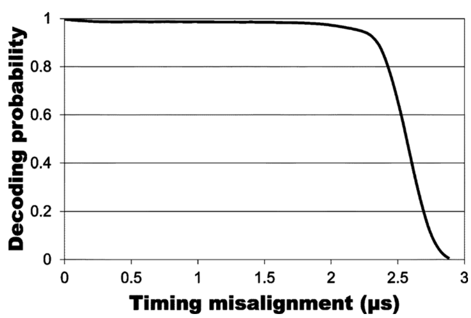


Fig. 18. Decoding probability of FICA under different timing misalignment.

Fig. 17 shows the error rates for 1000 tests at various degrees of replication as we vary the threshold α. With a large threshold, false positives decrease while false negatives increase slightly (although still less than 0.5%). Overall, though, there is quite a large space for threshold setting to provide good performance. In particular, when a bit is replicated on a few subcarriers (e.g., 4), both false positive and negative rates are close to zero when α is in the range 3–5. We hypothesize that, with more simultaneous transmissions, the detection should be more reliable as BAM is essentially energy detection. With more transmissions superposed, the energy is additive at the receiver side. However, due to limited hardware availability, we leave this for future study.

C. Demodulation Performance

We first evaluate the impact of timing misalignment on the FICA decoder performance. In our experiment, we have one FICA node access all odd-numbered subchannels and another FICA node access all even-numbered subchannels. Under this access pattern, the two stations maximally interleave their subchannels and should be more sensitive to intersubchannel interference. The frames are of length 400 B, modulated by BPSK, and encoded with 1/2 convolutional code. To precisely control the timing misalignment, we record the signal transmitted by each node separately at the AP and manually mix the two signal traces with different sample offset. We evaluate the successful decoding rate based on 100 000 such mixed signal traces.

Fig. 18 shows the results. The x-axis shows the timing misalignment of two frames. The decoding probability is over 97.28% when the timing misalignment is less than 2 μs. When the timing misalignment is in the range of 2–2.8 μs, the decoding probability drops sharply from 97.28% to 0.0043%. This is expected behavior as the timing difference is already very close to the short CP length of the data symbols. As shown

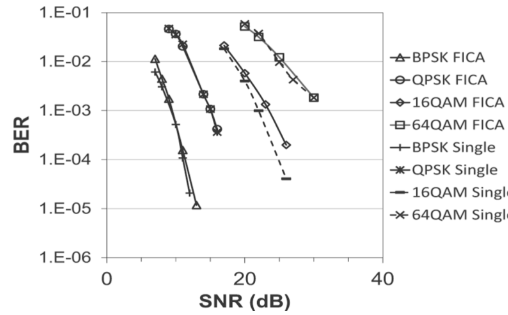


Fig. 19. Demodulation performance of FICA compared to conventional WLAN where only a single node can access channel at a time.

in our previous experiments, FICA can well control the timing misalignment within 2 μs with broadcasting, and the overall decoding performance should not be affected.

Next, we compare the demodulation performance of FICA under different modulation schemes, where multiple nodes are allowed to simultaneously access different subchannels, to the conventional WLAN, where only a single node can access the whole channel. The frame settings are the same as above. We fix the position of the AP and two stations and adjust the transmission power to get different SNRs. For each SNR setting, we evaluate four different modulation schemes. We schedule the transmissions in the following way. For each transmission power setting, we let two stations access the channel simultaneously using FICA first, and then immediately one station transmits alone. Since these two transmissions are back to back, we assume their channel conditions should be similar. For each power setting, we send 1400 frames. Each frame is 400 B and uses one subchannel.

We use the classic bit error rate (BER)-to-SNR plot to illustrate the demodulation performance. Fig. 19 shows the results for FICA as well as the case where only one node accesses the channel. The BER value shown is measured before the Viterbi decoder; after Viterbi, most of the errors are corrected. Clearly, all curves are very close to each other, including the high rate modulations like 64 QAM, which are very sensitive to interference. Thus, we conclude that with FICA, different nodes can transmit on different subchannels simultaneously without interfering with each other.

VII. Related Work

The application of OFDM for multiple access in WLANs is limited. In [18], Rahua, *et al.*, developed FARA that implements downlink OFDMA in a WLAN and per-subcarrier rate adaptation. However, since there is only one transmitter (the AP), symbol alignment is not an issue. In [5] and similarly in [11], OFDM has been used as a simple form of concurrent channel access. Nodes may modulate one bit of ACK information on different subcarriers after receiving a broadcast frame. However, FICA is a new PHY/MAC framework for WLANs that enables data communication over fine-grained subchannels to improve overall network efficiency.

Physical-layer signaling, usually with binary amplitude modulation, has been used previously to assist MAC protocols. In [8] and [13], busy tones are used to indicate channel occupancy to mitigate the hidden terminal problem. Recently, SMACK [5] uses a physical-layer ACK, and MCBC [20] uses a PHY RTS/CTS handshake to facilitate a multiround leader election protocol in a heavy-contention environment: After the protocol completes, the winner obtains access to the medium. FICA similarly shares the idea with SMACK and MCBC to apply PHY signaling based on simple BAM modulation, but FICA has the broader goal of enabling fine-grained channel access in high-data-rate WLANs.

Coordination using broadcast in local area networks has been previously exploited for time synchronization in reference-broadcasting synchronization, which provides microsecond-level synchronization precision [6]. Our results further confirm that microsecond-level coordination accuracy is practical in WLANs.

The overhead of 802.11 MAC has been carefully studied before, which motivates the frame aggregation in 802.11n [22], [27]. There is also extensive work to improve 802.11 MAC performance by tuning the backoff scheme [9], [10]. However, these approaches still consider the channel as one resource unit where only one radio can work on one channel at a time. Multichannel MAC protocols [16] have been studied to improve wireless network performance by using more orthogonal channels that are separated by guard bands. In contrast, FICA improves the performance of WLANs by increasing channel access efficiency. Using OFDM, FICA creates a fine-grained structure (i.e., subchannels) for multiaccess inside a wideband channel without guard bands. Thus, a FICA node can adjust the portion of the spectrum it accesses based on its traffic demands, while other nodes can use the remaining spectrum simultaneously. This property shares some similarity to the adaptive channel width demonstrated in [4]. FICA is complementary to that work by providing a concrete means for adaptive fine-grained subchannel access in WLANs.

The inefficiency of the 802.11 MAC has also been discussed before for supporting VoIP traffic [26]. In [26], a TDMA approach is used to reduce the contention overhead for CSMA in 802.11. In this paper, we argue that the inefficiency of 802.11 MAC is a fundamental bottleneck as the PHY data rate increases for all traffic, not just VoIP traffic. We further argue that this inefficiency issue should be resolved by enabling fine-grained channel access.

VIII. Conclusion

This paper addresses the inefficiency issue of MAC protocols in current WLANs as the PHY data rate increases. The fundamental reason of this inefficiency lies in the fact that the current MAC protocol allocates the entire wideband channel as a single resource. Such allocation becomes too coarse-grained for general traffic demands as the channel width or the modulation rate increases.

We argue that this inefficiency issue should be resolved using fine-grained channel access in high-data-rate WLANs. We present the design of FICA, a new cross-layer design that enables fine-grained subchannel random access based on OFDM. FICA addresses challenges in both PHY and MAC design due to the asynchronous and distributed nature of WLANs. First, FICA proposes a new PHY architecture based on OFDM that retains orthogonality among subchannels solely relying on the coordination mechanisms in existing WLANs. Second, FICA employs frequency-domain contention that uses physical-layer RTS/CTS signaling and frequency domain backoff to efficiently coordinate subchannel access.

We have implemented FICA on the Sora software radio platform. With our prototype, we validate the feasibility of the FICA cross-layer design. Our simulation results further indicate that FICA can outperform 802.11 with frame aggregation from a few percent to 600% under different traffic settings.

References

[1] *IEEE Standard for Local and Metropolitan Area Networks Part 11; Amendment 5: Enhancements for Higher Throughput*, IEEE Std 802. 11n-2009 , 2009.

[2] *IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, ANSI/IEEE Std 802.16-2004, 2004.

[3] 3 GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Long Term Evolution (LTE) physical layer; general description," TS 36.201-820, Dec. 2008.

[4] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl, "A case for adapting channel width in wireless networks," *Comput. Commun. Rev.* vol. 38, no. 4, pp. 135–146, 2008.

[5] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "SMACK: A SMart ACKnowledgment scheme for broadcast messages in wireless networks," in *Proc. SIGCOMM*, 2009, pp. 15–26.

[6] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. 5th OSDI*, 2002, pp. 147–163.

[7] A. Goldsmith, *Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[8] Z. J. Haas and J. Deng, "Dual busy tone multiple access (DBTMA)—A multiple access control scheme for ad hoc networks," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 975–985, Jun. 2002.

[9] Y. He, R. Yuan, J. Sun, and W. Gong, "Semi-random backoff: Towards resource reservation for channel access in wireless LANs," in *Proc. IEEE ICNP*, 2009, pp. 21–30.

[10] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle sense: An optimal access method for high throughput and fairness in rate diverse wireless LANs," *Comput. Commun. Rev.*, vol. 35, no. 4, pp. 121–132, Aug. 2005.

[11] B.-S. Kim, S. W. Kum, and R. L. Ekl, "OFDMA-based reliable multicasting MAC protocol for WLANs," *IEEE Trans. Veh. Technol.*, vol. 57, no. 5, pp. 3136–3145, Sep. 2008.

[12] D. Kivanc, G. Li, and H. Liu, "Computationally efficient bandwidth allocation and power control for OFDMA," *IEEE Trans. Wireless Commun.*, vol. 2, no. 6, pp. 1150–1158, Nov. 2003.

[13] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part I—Carrier sense multiple-access modes and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. COM-23, no. 12, pp. 1400–1416, Dec. 1975.

[14] D. Kotz and K. Essien, "Analysis of a campus-wide wireless network," *Wireless Netw.*, vol. 11, no. 1–2, pp. 115–133, 2005.

[15] X. Liu, E. K. P. Chong, and N. B. Shroff, "Opportunistic transmission scheduling with resource-sharing constraints in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 10, pp. 2053–5064, Oct. 2001.

[16] J. Mo, H.-S. So, and J. Walrand, "Comparison of multichannel MAC protocols," *IEEE Trans. Mobile Comput.*, vol. 7, no. 1, pp. 50–65, Jan. 2008.

[17] J. G. Proakis and M. Salehi, *Digital Communications*. New York: McGraw-Hill, 2008.

[18] H. Rahul, F. Edalat, D. Katabi, and C. Sodini, "Frequency-aware rate adaptation and MAC protocols," in *Proc. Mobicom*, 2009, pp. 193–204.

[19] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat, "Learning to share: Narrowband-friendly wideband networks," *Comput. Commun. Rev.*, vol. 38, no. 4, pp. 147–158, 2008.

[20] B. Roman, F. Stajano, I. Wassell, and D. Cottingham, "Multi-carrier burst contention (MCBC): Scalable medium access control for wireless networks," in *Proc. IEEE WCNC*, 2008, pp. 1667–1672.

[21] Z. Shen, J. G. Andrews, and B. L. Evans, “Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints,” *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 2726–2737, Nov. 2005.

[22] D. Skordoulis, Q. Ni, H.-H. Chen, A. Stephens, C. Liu, and A. Jamalipour, “IEEE 802.11n MAC frame aggregation mechanisms for next-generation high-throughput WLANs,” *IEEE Wireless Commun.*, vol. 15, no. 1, pp. 40–47, Feb. 2008.

[23] K. Tan, J. Zhang, J. Fang, H. Liu, Y. Ye, S. Wang, Y. Zhang, H. Wu, W. Wang, and G. M. Voelker, “Sora: High performance software radio using general purpose multi-core processors,” in *Proc. NSDI*, 2009, pp. 75–90.

[24] Q. Tang, S. Gupta, and L. Schwiebert, “BER performance analysis of an on-off keying based minimum energy coding for energy constrained wireless sensor applications,” in *Proc. IEEE ICC*, 2005, vol. 4, pp. 2734–2738.

[25] D. Tse and P. Vishwanath*, Fundamentals of Wireless Communications*. New York: Plenum, 2005.

[26] P. Verkaik, Y. Agarwal, R. Gupta, and A. C. Snoeren, “Softspeak: Making VoIP play well in existing 802.11 deployments,” in *Proc. NSDI*, 2009, pp. 409–422.

[27] Y. Xiao and J. Rosdahl, “Throughput and delay limits of IEEE 802.11,” *IEEE Commun. Lett.*, vol. 6, no. 8, pp. 355–357, Aug. 2002.

Ji Fang (S’12) received the B.E. degree in computer science and engineering from Beijing Jiaotong University, Beijing, China, in 2003, and is currently pursuing the Ph.D. degree in electrical engineering at Beijing Jiaotong University.

He has worked as a research intern with Microsoft Research Asia, Beijing, China, since March 2008. His research interests include wireless communication, software radio, and parallel computing.

Kun Tan (M’03) received the B.E., M.E., and Ph.D. degrees in computer science and engineering from Tsinghua University, Beijing, China, in 1997, 1999, and 2002, respectively.

He is a Lead Researcher with the Wireless and Networking Group, Microsoft Research Asia, Beijing, China. He developed the Compound TCP and led the development of the Sora software radio system. His research interests include transport protocols, congestion control, and wireless networks and systems.

Yuanyang Zhang received the B.S. degree in electrical engineering from Beihang University, Beijing, China, in 2010, and has been a graduate student with the University of California, Santa Barbara, since 2010.

His present research interests are networking and data mining.

Shouyuan Chen received the B.E. degree in computer science and engineering from Tsinghua University, Beijing, China, in 2010, and is currently pursuing the Ph.D. degree in computer science and engineering at the Chinese University of Hong Kong, Hong Kong.

He worked as an intern with Microsoft Research Asia, Beijing, China, from 2009 to 2010.

Lixin Shi received the B.S. degree in computer science from Tsinghua University, Beijing, China, in 2011, and is currently pursuing the Ph.D. degree in computer science at the Massachusetts Institute of Technology, Cambridge.

From December 2009 to June 2011, he was a research intern with the Wireless and Networking Group, Microsoft Research Asia, Beijing, China, mainly focusing on wireless networking topics, e.g., software defined radio, wireless protocol design, etc.

Jiansong Zhang received the B.S. and M.E. degrees in automation from Tsinghua University, Beijing, China, in 2004 and 2006, respectively, and is currently pursuing the Ph.D. degree in computer science and engineering at the Hong Kong University of Science and Technology, Hong Kong.

He is now an Associate Researcher with the Wireless and Networking Group, Microsoft Research Asia, Beijing, China. His research interest is computer wireless systems, including networking protocol design, baseband design, and radio hardware.

Yongguang Zhang (M’94–SM’11) received the Ph.D. degree in computer science from Purdue University, West Lafayette, IN, in 1994.

He is a Research Manager with Microsoft Research Asia, Beijing, China, in the areas of mobile systems and networking. Before joining Microsoft in early 2006, he was a Senior Research Scientist with HRL Labs, Malibu, CA. He has made technical contributions in internetworking techniques, system developments, and security mechanisms for satellite networks, ad hoc networks, and 3G wireless systems. He was also an Adjunct Professor of computer science with the University of Texas at Austin from 2001 to 2003.

Zhenhui Tan (M’94) received the Master’s degree in wireless communications from Bejing Jiaotong University, Beijing, China, in 1981, and the Ph.D. degree from Southeast University, Nanjing, China, in 1987.

He is a Professor with Beijing Jiaotong University. His research interests are digital mobile communication technology, spread spectrum communications, adaptive filtering algorithm, and digital signal processing techniques.