RCTC: Rapid Concurrent Transmission Coordination in Full Duplex Wireless Networks

Wenjie Zhou, Kannan Srinivasan and Prasun Sinha
Department of Computer Science and Engineering
The Ohio State University
{zhouwe, kannan, prasun}@cse.ohio-state.edu

***Abstract*—With recent advances in wireless systems, wireless in-band full duplex is proven possible. Prior work primarily allows a full duplex receiver to either send back a packet (*bi-directional mode*) or to forward another packet to its neighbor (*secondary transmission*). In our work, we look beyond a node pair and explore how a network can best utilize the full duplex capability. When a full duplex receiver does not have any packets to send back, concurrent transmissions (*exposed transmissions*) can be initiated. In a distributed channel access protocol, rapid signaling is crucial to identify the best mode for a given pair of transmitter and receiver, and to inform potential exposed terminals of transmission opportunities. In this paper, we present, RCTC, a fast and low overhead signaling mechanism based on Pseudo-random Noise (PN) sequences to enable multi-modal operation of wireless links in a distributed channel access setting to support concurrent transmissions in the neighborhood. Our prototype with USRPs shows up to 78% throughput gain. Extensive simulations over larger networks show a throughput gain of up to 131% for RCTC over the native full duplex scheme and up to 111% over a scheme that enables secondary transmission.**

I. Introduction

In 2010, the feasibility of in-band full duplex communication over a wireless link was shown independently by two research groups [1], [2] using various self-interference cancellation techniques. In full duplex communication, two nodes can simultaneously transmit packets to each other in the same channel, denoted as *bi-directional* mode in this paper. These designs were further improved with fewer antennas and to achieve better performance [3], [4]. But in reality, traffic is often asymmetric. For example, mobile operators and broadband network providers often provision their systems considering the download-heavy traffic pattern that is most prevalent in such networks. As a result, it is difficult to reap the benefits of full duplex in such scenarios. Similarly, *secondary transmission* is another mode of communication when a node is full duplex capable, i.e., capable of self-interference cancellation. In this mode, a full duplex receiver, while receiving a packet, can simultaneously transmit a packet to a node other than its transmitter, called a secondary receiver [5]. In an access point (AP) network, where every node connects to the Internet through an AP, a client cannot avail this mode as each client is connected to one AP at any given time. Even the AP may only have few opportunities to use this mode due to the asymmetry in downlink and uplink traffic. Furthermore, when the receiver does not have any packets to transmit, the transmitter-receiver pair operates in the *unidirectional* mode. In this mode, the receiver sends a busy tone or an artificially created packet with no useful content to protect the reception from hidden terminals. Thus, these current modes of transmission using full duplex may not always be the best way to utilize the potential of full duplex in practical scenarios.

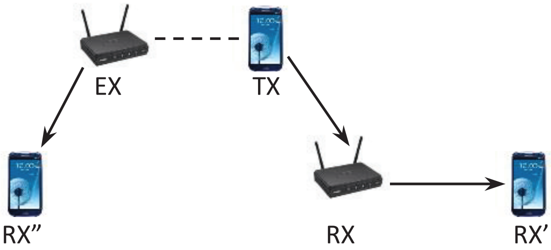


Fig. 1: An example with exposed and secondary transmission opportunities. Three flows ($TX\rightarrow RX, RX\rightarrow RX'$, and $EX\rightarrow RX''$) exist in the network. The dotted line between TX and EX denotes these two nodes interfere with each other. Node pairs with no line between them are out of interference range. $TX\rightarrow RX$ and $EX\rightarrow RX''$ are exposed links while $RX\rightarrow RX'$ forms a secondary transmission for the link $TX\rightarrow RX$.

The three identified modes are based on the ability of a pair of nodes in a network to best utilize the full duplex capability. In this paper, we go beyond a **pair of nodes** and investigate how a **network** can best utilize the full duplex capability. To this end, we propose a medium access control (MAC) protocol for full duplex networks called RCTC that seeks to maximize the overall network throughput: it attempts to identify other transmissions in the vicinity that can be executed concurrently. In other words, RCTC tries to align as many transmissions as possible in a full duplex network. Note that, depending on the mode of operation, the set of concurrent links can be different. For instance, in the *bi-directional* mode, typically the nodes in the vicinity of the transmitter and receiver cannot concurrently transmit, while in the *unidirectional* mode, nodes close to the transmitter and farther from the receiver could transmit concurrently. The mode of operation, in turn, depends on whether the receiver has a packet to send to its transmitter (*bi-directional*) or to another node (*secondary transmission*) or has nothing to send (*unidirectional*). Therefore, RCTC should be able to identify the mode of operation and the concurrent links on-the-fly. We first motivate the need for a **network-centric** full duplex protocol design as opposed to a **node pair-centric** design. Then, we introduce the novel mechanisms in RCTC.

An Example: Figure 1 shows an example scenario. Let us

978-1-4799-1270-4/13/$31.00 ©2013 IEEE

focus only on the case when TX is transmitting a packet to RX. Since node pairs without any line between them are out of interference range, nodes TX and EX form a pair of exposed terminals, and node RX can transmit a secondary packet to node RX' when node TX is transmitting. Let us assume that RX has packets for RX' and EX has packets for RX''. The packets are all of equal size. Let us ignore control and backoff overheads. In the unidirectional mode, the total throughput is 1 unit, as RX and EX are not transmitting. In the bi-directional mode, as RX does not have any packets for TX, the total throughput still remains 1 unit. In the secondary transmission mode, TX and RX can simultaneously send packets to their receivers achieving a throughput of 2 units. Thus, for a node-pair based strategy that tries to best utilize full duplex capability for its transmissions, the net throughput is at most 2 units. However, using a network-centric strategy, EX can concurrently transmit, achieving a net throughput of 3 units. If more exposed non-interfering links are present, the throughput can be even higher.

Challenges: The network-centric design outlined above requires addressing the following challenges. First, it requires the knowledge of whether the receiver (RX) has any packet to send back to the primary transmitter (TX) or has packets to send to a secondary receiver (RX'). Second, it requires that neighboring nodes of TX and RX identify the mode of operation of RX and accordingly decide whether they could transmit simultaneously. These two requirements together imply that the exposed node (EX) should identify if RX is sending a packet back to TX or not, and only align its transmission if there is no return transmission. This decision needs to be made almost instantaneously at EX so that the start times of these transmissions are not too far. Additionally, when more than one exposed transmission is feasible for a primary node pair, one more requirement is to identify *which* of the exposed nodes could transmit concurrently with the primary transmission such that the exposed transmissions do not collide with each other.

Solution Overview: To address the above challenges, RCTC uses pseudo-random noise (PN) sequence-based signatures carried in packets to help quickly identify different modes of full duplex and help exposed nodes to identify *when* they should transmit. Recently, such signatures have been demonstrated to be usable for conveying packet corruption information from the receiver to the sender [6], for identifying the receiver of a packet [7] and for encoding control packets such as RTS, CTS and ACK [8]. Our solution works as follows. Each node uses a random backoff to contend for the channel if it has a packet to transmit. Upon obtaining access to the channel, it starts transmitting two signatures which indicate the IDs of the primary receiver and the transmitter. IDs are locally unique and various mechanisms could be used to select it [8]. Upon receiving these signatures, the receiver looks up in its queue to determine if other transmission modes are feasible. It then selects an appropriate mode and indicates its selection to the transmitter also using signatures. The transmit power of signatures from the primary receiver is controlled to provide a sense of how much interference could be tolerated by the receiver. Upon receiving the mode selection from the primary receiver, the primary transmitter continues to transmit the data packet. The exposed terminals, on the other hand, identify themselves through the signal strength from the signature sent by the receiver.

This paper makes the following contributions:

- We develop a fast and low overhead signaling mechanism to enable multi-modal operation of wireless links in a distributed channel access setting.
- We extend the signaling mechanism to support concurrent transmissions in the neighborhood.
- Using a USRP testbed and simulations we show that our solution RCTC significantly outperforms the state-of-the-art full duplex schemes in practical scenarios. A 79% throughput gain is achieved in our USRP testbed compared with native full duplex solution and as high as 131% average throughput gain without hurting fairness in our simulations for large networks.

The rest of this paper is organized as follows. Section 2 describes the design challenges. Section 3 presents the various components of RCTC and describes how they work together. Results from our USRP prototype and simulations are presented in Sections 4 and 5, followed by discussion, related work and conclusion sections.

II. Design Challenges

Considering the recently proposed full duplex techniques at the physical layer, a practical and efficient distributed channel access scheme has to address the following challenges:

- *Transmission Mode Identification:* The potential receiver of a primary transmission (the first transmission resulting from a successful contention) needs to determine the best option among those three modes based on the available packets in its queue. It then needs to inform the primary transmitter as well as other surrounding nodes of its decision with minimal delay and overhead.
- *Exposed Terminal Identification:* The potential exposed terminal should be able to identify its role in the transmission based on the signal received. The transmission from an exposed terminal should not affect the reception at the primary receiver and the primary transmitter as well. Then it quickly aligns its own transmission with the primary transmission.
- *Picking Exposed and Secondary Receivers:* The potential receiver of an exposed or secondary transmission may be interfered by the signal from the primary transmitter. Randomly picking a receiver could end up in collision, which does not help to increase the system throughput.

These design challenges need to be addressed with a low overhead solution that can rapidly coordinate the transmission activities of nodes without prior knowledge of the information on packets in queues at neighboring nodes and little information about the channel state between different nodes.

III. Design of RCTC Protocol and its Components

In this section, we explain the design details of RCTC. We first define the three transmission modes and the process to identify different modes. Then the technique to identify exposed terminals is elaborated. We next discuss how to pick receivers for secondary and exposed transmissions. To protect bi-directional transmission, a technique called transmitter suppressing is introduced. Finally all the components are combined together.

A. Transmission Modes

When a node accesses the channel and transmits a packet, the receiver responds based on the contents of its queue. There are three possible modes of transmission:

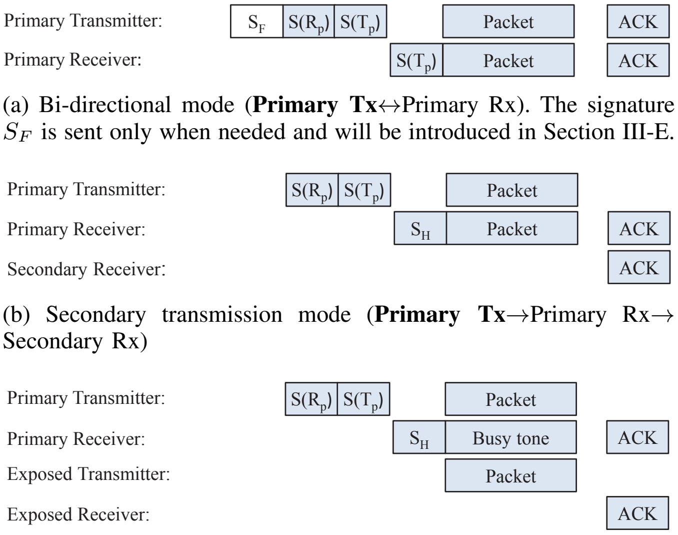
- *Bi-directional Transmission Mode:* The primary receiver uses this mode if a return packet is available for the primary transmitter.
- *Secondary Transmission Mode:* If a bi-directional transmission is not feasible, the primary receiver searches its queue for potential transmissions to other nodes. This secondary transmission receiver should be able to correctly receive the packet under the interference of the primary transmitter.
- *Unidirectional Mode:* This mode is similar to half duplex, except that the receiver sends a busy tone to protect the packet being received from hidden terminals. This mode is used if the above two modes are not feasible.

In all the three modes, exposed terminal transmissions may be feasible in the neighborhood of the transmitter(s). But proper checks must be performed to ensure that the ongoing primary and secondary transmissions are not excessively interfered by the exposed transmissions and vice-versa.

B. Transmission Mode Identification

RCTC makes use of physical layer signatures based on PN sequences that are used for rapid and low-overhead communication and coordination. More details about signature detection are in Appendix A. The coordination process is as follows. The primary transmitter first sends two signatures that encode the locally unique IDs of the receiver and transmitter as shown in Figure 2. The receiver identifies the transmitter using its signature and determines the best mode based on the packets in its queue and their expected physical layer data rates to their next-hop destinations. The sender is notified of the selected mode using different return signatures. The transmitter's signature $S(T_p)$ indicates a bi-directional transmission mode while a special signature S_H is used for unidirectional or secondary transmission mode. We call this process as rapid handshaking.

All nodes have a unique ID (as a signature) within its one-hop neighborhood. The details about how to create and assign the signatures are beyond the scope of this paper. A naive scheme is as follows. Each node randomly selects a signature from a code-book and exchanges this information periodically with its neighbors. Conflicting signatures are resolved by repeating the process. Alternatively, the signatures may be generated using a hashing function of a node's MAC address [8].



(a) Bi-directional mode (**Primary Tx**↔Primary Rx). The signature S_F is sent only when needed and will be introduced in Section III-E.

(b) Secondary transmission mode (**Primary Tx**→Primary Rx→ Secondary Rx)

(c) Unidirectional transmission mode with exposed transmission (Primary Rx←**Primary Tx**   Exposed Tx→Exposed Rx)

Fig. 2: Timing diagrams for various transmission arrangements. $\mathbf{S(T_p)}, \mathbf{S(R_p)}, \mathbf{S_H}, \mathbf{S_F}$ are the signatures of Primary Tx and Primary Rx, a reserved signature for half-duplex transmission and another reserved signature for full duplex transmission.

C. Exposed Terminal Identification

The definition of the exposed terminal suggests that its transmission should not affect packet reception of the primary receiver. Because the packet decoding success probability is positively correlated to the signal-to-interference ratio (SIR), we can use SIR to identify exposed terminals. When the potential SIR at the primary receiver is higher than the minimum requirement to decode the packet correctly, the exposed terminals are allowed to transmit. Assume that the Wi-Fi transmit power is P_0; the channel coefficient between the primary transmitter (node T) and the primary receiver (node R) is h_{TR} and the coefficient between another node E and node R is h_{ER}. Let us denote the received signal strength at node R as P_{TR} from node T and P_{ER} from node E. Node E is categorized as an *exposed terminal* when Equation 1 is satisfied.

$$SIR_R = \frac{P_{TR}}{P_{ER}} = \frac{P_0 |h_{TR}|^2}{P_0 |h_{ER}|^2} = \frac{|h_{TR}|^2}{|h_{ER}|^2} > \Delta_d \qquad (1)$$

Δ_d is a predefined constant related with the transmission data rate d. Large Δ_d results in higher packet reception ratio of the primary transmissions while small Δ_d can support a larger number of simultaneous exposed terminal transmissions. In RCTC, we pick Δ_d for each data rate such that the packet reception ratio (from node T to node R under interference from node E) is above 95%.

Equation 1 requires the value of both h_{ER} and h_{TR} while the latter is not available to the potential exposed terminals. RCTC uses a simple mechanism to ensure that any potential exposed terminal can estimate if it is safe to transmit based on the observed power level of the signature from the primary receiver. The primary receiver, on the other hand, dynamically adjusts its transmit power of the signature according to h_{TR}. Then the received signal strength of the signature at the potential exposed terminal reflects the relationship between h_{ER} and h_{TR}. Specifically, when the primary receiver transmits S_H as shown in Figure 2, it sets the transmit power to

$$P_r = \frac{C_d}{P_{TR}}, \tag{2}$$

where C_d is a constant related to the data rate d. Assume that the wireless channels in the two directions between any pair of nodes are symmetric. Then the received signal power at node E is

$$P_{RE} = P_r |h_{ER}|^2 = \frac{C_d}{P_{TR}} |h_{ER}|^2 = \frac{C_d |h_{ER}|^2}{P_0 |h_{TR}|^2}. \tag{3}$$

From Equations 1 and 3, the following equation is satisfied at the exposed terminal

$$P_{RE} = \frac{C_d |h_{ER}|^2}{P_0 |h_{TR}|^2} < \frac{C_d}{P_0 \Delta_d}. \tag{4}$$

Since it is difficult for the exposed terminal to know the transmission data rate between the primary transmitter and the primary receiver ahead of time, we choose $C_d = C(P_0 \Delta_d)$, where C is a pre-defined constant known to every node. So Equation 4 becomes

$$P_{RE} < \frac{C_d}{P_0 \Delta_d} = C. \tag{5}$$

Then node E is safe to transmit if the received signal strength of the signature is lower than C regardless of the data rate between node T and node R.

D. Selecting Exposed and Secondary Receiver

In the previous section, we introduce the scheme to identify the exposed terminal. However, it only guarantees that the signal from the exposed terminal is not hurting the reception at the primary receiver. It is not clear whether a potential receiver of the exposed terminal could receive correctly or not as it could be close to the primary transmitter. Similarly, in the secondary transmission mode, the reception at the secondary receiver may be affected.

One of the recent works for exploiting exposed terminal transmissions, CMAP [9] maintains a probability map which stores the success reception ratio when two links are transmitting simultaneously. Another work CMAC [10] uses the relationship between the SIR and packet reception ratio (PRR) to check the ability for concurrent transmissions. These schemes, however, only characterize the relationship between *a pair of* links. The *combined* interference from two or more exposed terminals is not considered. Our simulation results in Section V show that the performance of CMAP degrades as the number of exposed terminals increases.

RCTC also utilizes history information to pick the receiver of the exposed transmission and secondary receiver, however, in a different way from prior work. Each node keeps two lists: an *ExMap* for exposed transmissions, and a *SecMap* for secondary transmissions. Both of them consist of a triplet: $\{Tx, Rx, p\}$, where Tx is the primary transmitter, Rx is the potential exposed or secondary receiver, and p is the probability that the reception at Rx could be successful when Tx is the primary transmitter. The probability p is maintained according to the following rules:

- Set p to 1 upon a successful transmission.
- Halve p upon a failed transmission (i.e. ACK timeout).

When an exposed or secondary transmission opportunity appears, the RX with the highest p is selected as the receiver. The exposed transmission, however, is not always carried out. Instead, it happens with the probability p. This scheme helps to resolve the collision between multiple exposed terminals.

E. Transmitter Suppressing

In section III-C, exposed terminals identify themselves using Equation 5. This, however, raises a question: what should the exposed terminals do if they don't receive the return signature from the primary receiver? On one hand, absence of a return signature signal means that the exposed terminal is beyond the communication range of the primary receiver, which suggests that its transmission will not hurt the reception at the primary receiver. On the other hand, the primary receiver may choose to send a return transmission and the exposed transmission may hurt the reception at the primary transmitter.

RCTC uses the following approach to suppress exposed terminal transmissions. The transmitter broadcasts a reserved signature S_F (as shown in Figure 2(a), F denotes full duplex transmissions) when its reception of the return transmission is severely affected by the exposed terminals. Specifically, each primary transmitter Tx keeps a list consisting of: $\{Rx, P_{lost}\}$, where Rx is a potential receiver of Tx, and P_{lost} is the ratio of unsuccessful return transmissions. Tx maintains P_{lost} using the history in a given time window t_w and sets it to 0 when no return transmission happens in t_w. Then, if the ratio P_{lost} is above a threshold, α, the transmitter broadcasts a special suppressing signature, S_F, before it sends the signature of the primary receiver. Exposed terminals should not start a concurrent transmission if S_F is received. Otherwise they could start transmitting when the return signature from the primary receiver is not detected. We study the value of α in Section V and pick $\alpha = 0.1$ as a tradeoff between network throughput and fairness.

F. Putting It All Together

Algorithms 1, 2 and 3 outline the channel access logic of the primary transmitter, receiver and the exposed terminal. Note that these three algorithms are running concurrently at all of the nodes and there is no pre-assigned role for any node. T_P, the primary transmitter, has a packet for R_P, the primary receiver. T_P performs carrier sensing and backoff, and begins transmission upon successful channel contention. Its transmission consists of $\langle S(R_P), S(T_P) \rangle$ where $S(R_P), S(T_P)$ are the signatures of the primary receiver and transmitter, respectively. While waiting for an incoming transmission, R_P continuously correlates the channel with $S(R_P)$. Upon detecting $S(R_P)$, R_P starts correlation with signatures of all neighbors that it communicates with. After detecting $S(T_P)$, R_P identifies the transmitter and searches its outgoing packet queue for packets

Algorithm 1: Operation of a primary transmitter

```
$S(T_p) \leftarrow$ primary transmitter's signature;
$S(R_p) \leftarrow$ primary receiver's signature;
$S_H \leftarrow$ half duplex signature;
$S_F \leftarrow$ full duplex signature;
$t_{return} \leftarrow$ the expected time a return signal should appear;
$P_{lost}(R_p) \leftarrow$ the return packet reception ratio from $R_p$;
while has packet to send do
    if channel access granted then
        if $P_{lost}(R_p) > \alpha$ then
            Send $S_F$;
        Send $\langle S(R_p), S(T_p) \rangle$ in sequence;
        if return signal detected in $t_{return}$ then
            Use $S(T_p), S_H$ to correlate the received signal;
            if signature detected then
                Send the packet;
                Wait for ACK and update $P_{lost}(R_p)$;
            else
                Abort this transmission;
        else
            Abort this transmission;
```

Algorithm 2: Operation of a primary receiver

```
$S(T_p) \leftarrow$ primary transmitter's signature;
$S(R_p) \leftarrow$ primary receiver's signature;
$S_H \leftarrow$ half duplex signature;
while $S(R_p)$ detected do
    Correlate with its neighbors' signatures;
    if $S(T_p)$ detected then
        if packet for $T_p$ exists then
            Send $S(T_p)$ and a return packet in sequence;
        else
            Send $S_H$ with power $P_r$ (Equation 2);
            if a secondary receiver exists then
                Send a packet to the secondary receiver;
                Wait for ACK and update SecMap;
            else
                Send busy tone signal;
```

Algorithm 3: Operation of an exposed terminal

```
$S_H \leftarrow$ half duplex signature;
$S_F \leftarrow$ full duplex signature;
$t_{return} \leftarrow$ the expected time a return signal should appear;
while correlating with neighbors' signatures and $S_F$ do
    if $S_F$ detected then
        Wait for the end of this transmission;
        continue;
    Store the transmitter's ID ($T_p$);
    Keep correlating the received signal with $S_H$;
    if $S_H$ detected in $t_{return}$ then
        if signal strength of $S_H < C$ (Equation 5) then
            Select a potential receiver and send a packet to it;
            Wait for ACK and update ExMap;
    else
        Select a potential receiver and send a packet to it;
        Wait for ACK and update ExMap;
```

destined for T_P. R_P has the following options based on the result of searching its queue.

1) *R_P finds a packet for T_P*: R_P has a *return packet*, and transmits $\langle S(T_P) \rangle$, indicating a bi-directional transmission.
2) *R_P finds no packet for T_P*: R_P searches its queue for packets to other neighboring nodes, and uses the SecMap to check the feasibility of such a transmission. Upon finding a suitable secondary receiver, R_P transmits back S_H and the packet. Otherwise, R_P transmits S_H followed by a busy tone which lasts until the end of the incoming packet.

The exposed terminal keeps track of the transmitter's ID and then keeps correlating S_H. Upon detection of S_H, if the signal strength of S_H satisfies Equation 5, it then selects a suitable receiver and transmits the packet with the specific probability.

IV. Implementation

In this section, we implement RCTC in a testbed with 5 USRPs. We compare the throughput of three different schemes in this section: RCTC, FDNative, and CF [5]. FDNative is the baseline full duplex solution where a primary receiver sends a packet back to the transmitter if a packet is available. If the receiver has no return packet in the queue, a busy tone signal is sent out. In CF, both return and secondary transmissions are enabled.

A. USRP Prototype

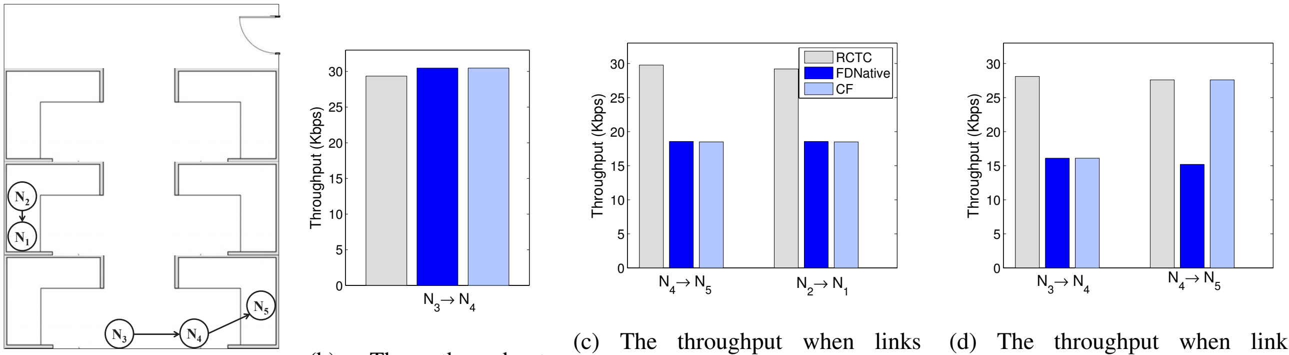
Each USRP in our prototype consists of two antennas working in the same channel, one for transmitting and the other for receiving. Digital cancellation is implemented to cancel self-interference and provides about 27 dB cancellation.

The experiment is conducted in an office environment with 5 USRPs as shown in Figure 3(a). Although all of the nodes are in one collision domain (they can hear each other's transmission), the different distances between nodes and the metallic tables help to create exposed scenarios (link $N_4 \rightarrow N_5$ and $N_2 \rightarrow N_1$) and secondary transmission opportunities (link $N_3 \rightarrow N_4$ and $N_4 \rightarrow N_5$). [1] Because of the large latency between the host computer and USRP, we use small sampling rate (250 KHz in our experiment) and large packet length (3000 Bytes). GMSK with 4 samples per symbol is used as the modulation scheme, which results in a data rate of 62.5 Kbps. However, due to the control latency and backoff overhead, only 30.5 Kbps for FDNative is achieved when there is only one pair of transmitter and receiver as shown in Figure 3(b). RCTC behaves slightly worse with a throughput of 29.35 Kbps because of the signature overhead.

B. Exposed Transmission

Figure 3(c) shows the throughput of two exposed links $N_4 \rightarrow N_5$ and $N_2 \rightarrow N_1$. The aggregate throughput of RCTC is 59.1% higher than that of FDNative, which in theory should be 100% higher. There are two reasons behind this result. First, the control signatures increase the overhead of RCTC as explained in the previous section. Second, these two exposed links also have opportunities to transmit simultaneously and

[1] Note that the rule for exposed transmission (Equation 1) only requires that the interference at the receiver can be tolerated. So the exposed terminal and primary receiver can be in one collision domain. However, we do need to enable capture effect [11], which allows a receiver to re-lock to a stronger signal while it is receiving a weaker one. So the receivers in our experiment could re-lock to the packet from its own transmitter (stronger packet) even if it is already locked to the packet from exposed terminals (weaker packet).



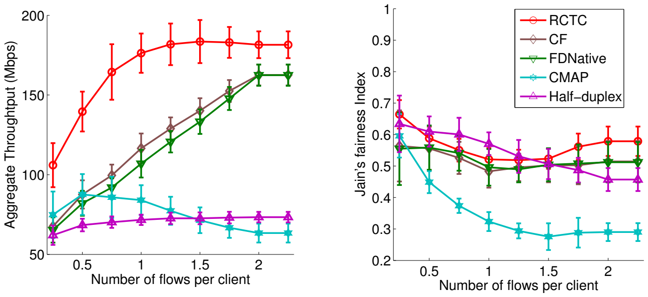
(a) The USRP placement in our experiment. The solid arrows indicate the traffic direction.

(b) The throughput when only link $N_3 \rightarrow N_4$ is enabled.

(c) The throughput when links $N_4 \rightarrow N_5$ and $N_2 \rightarrow N_1$ are enabled. RCTC achieves 59.1% higher throughput than FDNative.

(d) The throughput when links $N_3 \rightarrow N_4$ and $N_4 \rightarrow N_5$ are enabled. RCTC achieves 78% higher throughput than FDNative.

Fig. 3: The USRP testbed topology and throughput of different scenarios. All nodes can hear each other.



(a) Aggregate throughput

(b) Jain's fairness index

Fig. 4: Aggregate throughput and Jain's fairness index for different number of flows per client.

successfully in FDNative if they select the same backoff value. The latter contributes the most because the sum throughput for FDNative is 37 Kbps, which is 21% higher than the throughput when only link $N_3 \rightarrow N_4$ is enabled. Since CF only enables secondary transmission, its performance is the same as FDNative.

C. Secondary Transmission

Figure 3(d) shows the throughput of link $N_3 \rightarrow N_4$ and $N_4 \rightarrow N_5$. In this scenario, $N_3 \rightarrow N_4 \rightarrow N_5$ naturally forms a secondary transmission chain. In addition, when link $N_4 \rightarrow N_5$ wins the contention, link $N_3 \rightarrow N_4$ behaves as an exposed link since its transmission does not hurt the reception of node N_5. So both of the links could transmit concurrently all the time. Compared with FDNative, RCTC achieves 78% higher aggregate throughput. CF, on the other hand, only allows link $N_4 \rightarrow N_5$ to always transmit, resulting in 41% throughput gain.

V. Evaluations

To study the performance of RCTC in larger networks, we present ns-3 based simulation results in this section. The following algorithms are evaluated in this section:

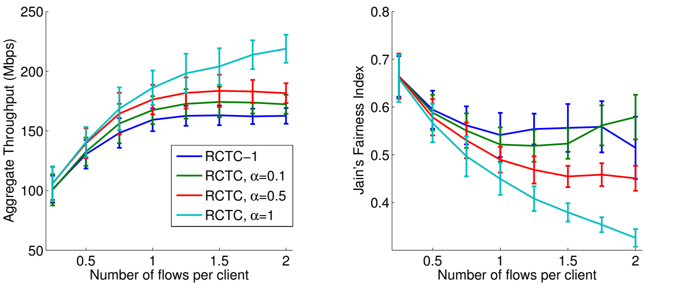
- *Half-duplex*: This represents the Distributed Coordination Function (DCF) mode in the IEEE 802.11 protocol without RTS-CTS control packets.
- *CMAP*: The proposed scheme in [9] which deals with exposed terminals in half duplex.
- *FDNative*: As discussed in Section IV.
- *CF* [5]: As discussed in Section IV.
- *RCTC*: Our proposed solution.

Throughput and fairness are the two metrics used to evaluate the performance of different schemes. The Jain's fairness index [12] is used to calculate the throughput fairness of all of the flows created. We present results from AP based networks and ad hoc networks. The data rate is fixed to 12 Mbps and the packet size is 1500 Bytes. We pick 5 dB as the threshold for Δ_d (Equation 1) because it provides close to 100% packet success ratio with the default error model in ns-3. All flows created are saturated UDP flows.

A. AP Network

The APs are uniformly randomly distributed in an $800 \times 800 m^2$ area. First, the area is divided into small sections, one for each AP. Then each AP is randomly placed within each section. Several clients are randomly placed around each AP. Finally, we randomly create a number of possible flows (uplink and downlink). The default setting is 30 APs with 3 clients per AP and 0.5 flow per client. For each setting, the simulation result is averaged over 20 randomly generated scenarios.

1) Different Number of Flows: Figure 4(a) plots the average aggregate throughput with varying number of flows. The standard deviation is plotted as the vertical bar around the average value. Since return transmission has higher priority than secondary transmission in RCTC, the gain from exposed transmissions decreases when the number of flows increases because more bi-directional transmissions occur. Note that 2 flows per client indicates that all of the flows are bi-directional. However, RCTC could still achieve 12% higher throughput than FDNative because exposed transmissions are still allowed if the primary transmitter has a small P_{lost} as



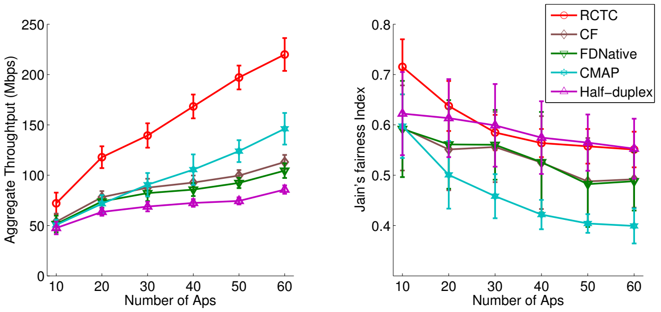
(a) Aggregate throughput (b) Jain's fairness index

Fig. 5: Aggregate throughput and Jain's fairness index with different value of α introduced in Section III-E.

discussed in Section III-E. At 0.75 flows per client, the average throughput gain of RCTC over FDNative reaches the highest point at 79.4%. The fairness of all the full duplex schemes are comparable with RCTC performing the best, indicating that the throughput gain of RCTC does not come with the cost of fairness. CMAP experiences low fairness and even lower throughput than half duplex with more than 1.25 flows per client. We believe this is related to the windowed ACK and backoff scheme in CMAP. In CMAP, when a node has a total of 8 unacknowledged packets, it keeps quiet for a random duration between 40 to 80 ms. In the experiment, we found that the total occurrences of this waiting increases from 1000 to 7000 as the number of flows changes from 0.5 (peek throughput point) to 2 (lowest throughput point) per client during a 2 seconds simulation time. *The high number of ACK timeouts originates from the fact that CMAP uses pair to pair relationship to determine exposed links and is unable to deal with two or more exposed links.*

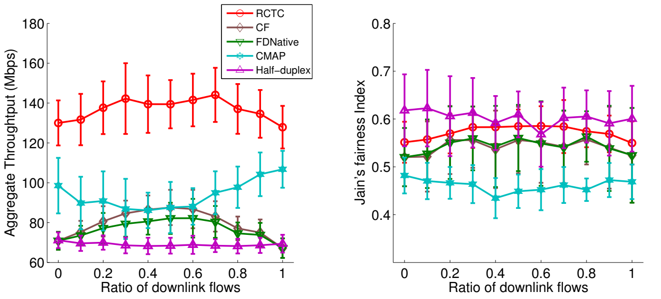
2) Influence of Design Parameter: We keep the same simulation setting as the previous section. Then we change the parameter α discussed in Section III-E. The results are shown in Figure 5. The scheme "RCTC-1" refers to a variant of RCTC where exposed terminals are not allowed to transmit if they do not receive the return signature from the primary receiver. With $\alpha = 1$, the transmitter suppressing is turned off and the exposed terminals are free to transmit in the absence of return signature. This scheme achieves higher throughput, however, at the cost of a sharp decrease in fairness. We pick $\alpha = 0.1$ as the default parameter in our design because it balances the throughput and fairness.

3) Different Number of APs: Figure 6 shows the result with different number of APs. All the schemes show an increase in the aggregate throughput as the density increases. The throughput gain of RCTC increases from 1.41× to 2.31× the throughput of FDNative as higher density has more exposed transmission opportunities. CF only performs 5% to 9% better than FDNative. And RCTC achieves 35% to 111% higher performance than CF. Full duplex is supposed to double the throughput of half duplex. In this experiment, RCTC shows an average of 1.53× to 2.86× the throughput of Half-duplex and 1.43× to 1.67× the throughput of CMAP. Figure 6(b) shows the fairness index for all flows in the network. The results of all full duplex schemes are comparable. Although CMAP achieves higher throughput with increasing number of APs, the fairness is much lower than other schemes.



(a) Aggregate throughput (b) Jain's fairness index

Fig. 6: Aggregate throughput and Jain's fairness index for different number of APs



(a) Aggregate throughput (b) Jain's fairness index

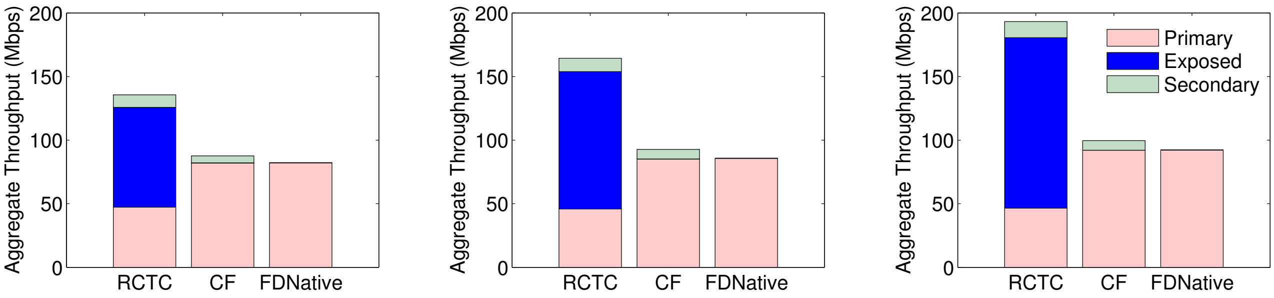
Fig. 8: Aggregate throughput and Jain's fairness index of different ratio of downlink traffic.

4) Analyzing the Throughput Gains: To understand the gains obtained by RCTC, we investigate the contribution of primary, exposed and secondary transmissions in Figure 7 with varying number of APs. In all of the results, the throughput of primary transmissions of RCTC decreases compared with FDNative due to the increase in contention resulting from exposed terminal transmissions. Observe that with increasing node density (more APs), the opportunities for exposed terminal transmissions are increasing in RCTC. The low throughput contribution from secondary transmission also confirms our argument that the opportunities for secondary transmission in AP network are few.

5) Different Ratio of Downlink Traffic: In AP networks, there are two types of traffic. Uplink traffic is from the clients to the AP, and downlink traffic is from AP to clients. AP networks are generally considered as downlink heavy. Figure 8 plots the results with different downlink traffic ratio. As shown in the figure, RCTC outperforms FDNative in case of fairness all of the time. The throughput gain varies from 71.2% to 90.9%, indicating that the performance of RCTC is robust to different traffic patterns.

B. Ad Hoc Network

Nodes are randomly placed in an 800×800m^2 area. One-hop flows are chosen randomly on links that have SNR higher than 10 dB. Figure 9(a) presents the aggregate throughput with varying number of nodes in the network. The throughput gain of RCTC over FDNative increases from 9.1% to 54.2% while the throughput gain of CF changes from 1% to 14%.

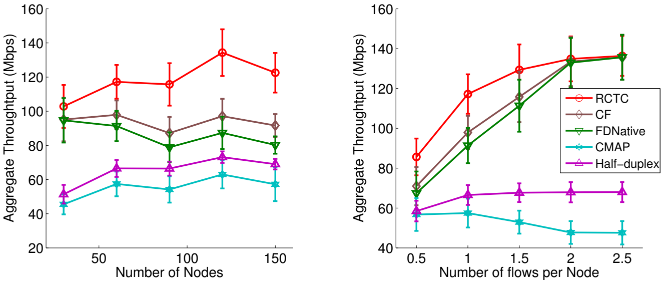


(a) 30 APs

(b) 40 APs

(c) 50 APs

Fig. 7: Throughput of primary, exposed and secondary transmissions with different number of APs.



(a) Different number of nodes

(b) Different number of flows

Fig. 9: Aggregate throughput for different number of nodes and flows per node in an ad hoc network.

Figure 9(b) shows the performance for varying number of flows in the network. The total number of nodes is 60. As shown in Figure 9(b), as the number of flows per link changes, the throughput gain of RCTC varies from 0.6% to 28.9%.

VI. Discussion

Our design and evaluation of RCTC assumes that all packets in the queue are of the same size and all nodes are using the same data transmission rate. The reason behind it is that the transmissions should be all aligned as shown in Figure 2 to protect the ACKs. This assumption raises questions about the performance of RCTC in practice. In this section, we briefly discuss the impact of varying packet size, multiple data rates and bursty traffic, and outline a potential approach to address these issues.

- *Varying packet size:* In a real deployment, packets may not have the same size. The ACK packets in TCP traffic are significantly smaller than the data packets: data packets are as long as 1500 bytes and TCP ACKs are only 40 bytes. However, we can use a two-pronged approach to address such scenarios. First, we can perform packet aggregation like in 802.11n and packet splitting. This allows us to create virtual packets that are of the same length. Second, busy tone signals can be used as padding bits. Finally, the exposed terminals examine the length of the primary transmission and refrain from sending bigger packets.
- *Multiple data rates:* Our solution could be easily extended to multiple data rates by varying the parameters Δ_d and C_d used in Equation 5 for different values of data rate d. Packets of the same length require different durations to transmit using different data rates. However, the solutions for varying packet length could be applied to address this issue.
- *Bursty traffic:* The traffic flows in our evaluation of RCTC are saturated UDP flows. In reality, flows tend to be bursty. However, this could be taken care of through tuning the design parameters t_w and α as discussed in Section III-E. Smaller t_w allows the primary transmitter to quickly stop suppressing exposed terminals when the return flow finishes while larger t_w makes suppressing more robust to fluctuating traffic.

VII. Related Work

Several categories of schemes have been proposed to solve the hidden terminal and the exposed terminal problems.

Busy Tone: In Busy Tone Multiple Access (BTMA)[13], the receiver sends out a busy tone signal using a secondary wireless channel. All of the nodes in the surrounding area of the receiver will hear this busy tone and be prevented from transmission. A drawback is that a second channel and the corresponding guardband create a significant overhead on spectrum.

Request-to-Send/Clear-to-Send (RTS/CTS) [14], [15]: This scheme is proposed to solve the hidden terminal problem and could be extended to the exposed terminal problem. Exposed nodes can send upon hearing the RTS while not receiving the corresponding CTS. To align exposed transmissions, a control time gap is inserted between RTS/CTS and DATA/ACK in [16]. In [17], exposed terminals use the SINR in the CTS message to estimate whether it can start a concurrent transmission or not. To protect the RTS packets from collision, the authors of DBTMA [18] used a second busy tone, which increases the spectrum overhead. In [19], a modification is proposed in which a node desiring to share the channel transmits a Request-To-Send-Simultaneously (RTSS) packet. When an exposed link gets access to the channel, it returns a Clear-To-Send-Simultaneously (CTSS) packet that allows the simultaneous transmission. The RTS/CTS messages are transmitted at the lowest data rate, which creates a significant time overhead. FlashLinQ [20] uses multiple rounds of RTS/CTS OFDM slots to schedule concurrent transmissions, which requires tight synchronization.

Interference Mapping: E-CSMA [21] builds a channel state map using the received signal strength indicator (RSSI) at the transmitter and the transmission success ratio reported by the receiver. Because the observed RSSI is maintained at the transmitter instead of the receiver, there is no real-time knowledge of the interference at the receiver. In CMAP[9], the authors proposed a scheme that uses an online conflict map. Two links that cannot transmit simultaneously form an entry in the conflict map. Potential transmitters monitor ongoing transmissions and check the conflict map to make transmission decision. However, the time required for creating and updating the map at each node grows exponentially with the number of links, and the ability for nodes to monitor multiple overlapping concurrent transmissions is actually impossible. In [10], the authors developed two empirical models based on the relationship between received signal strength, transmit power, SNR, and packet reception ratio. When a potential transmitter overhears a packet, it uses the models to check if it can transmit and select the appropriate transmission power level. An RTS/CTS scheme is used between the exposed node and its receiver. However, the exposed node may not be able to receive the CTS message due to the interference from the ongoing primary transmission, which causes a significant reduction in realization of exposed transmission opportunities.

Tuning Carrier Sensing threshold and transmit power: The number of exposed and hidden link pairs in a given network vary with the transmit power, carrier sensing (CS) threshold, and data rates [19], [22]. A higher transmit power or smaller CS threshold will increase the probability of exposed link pairs while preventing hidden terminal link pairs. Several schemes have been proposed to tune the transmit power and CS threshold to make a tradeoff between hidden terminal collisions and exposed terminal transmissions in order to achieve better network throughput [22]–[24].

Full Duplex Mac: In [25], the transmitters initially send a half duplex packet. Subsequent transmissions between the same transmitter and receiver are full duplex. A scheme called shared random backoff is used to align the full duplex transmissions in time. The authors assume that a node cannot start a transmission while it is in the receiving state, due to the limited coherence time of self interference channel state estimation. In [5], the receiver of a primary transmission can send a packet to a secondary receiver when it does not have return transmission to the transmitter. It also uses a busy tone signal to fill the time gap between the transmission of the transmitter and the receiver to prevent neighboring transmissions. However, it does not take advantage of exposed terminals.

VIII. Conclusion

We have taken the first step in uncovering the ability for full duplex systems to mitigate several practical issues and improve its throughput performance. We believe that the primitives such as the signatures, together with the ability to always overhear, can enable a rethinking of MAC strategies. In this paper, we presented RCTC, a scheme that makes use of signatures to allow fast handshaking for coordinating transmissions in the neighborhood, and allow exposed and secondary transmissions. Compared with native full duplex MAC, our prototype shows as high as 78% throughput gain and evaluation results with larger network show up to 131% gain in throughput. Besides, RCTC also achieves better fairness performance.

Acknowledgements

This material is based upon work partially supported by the National Science Foundation under Grant CNS-1161490.

We thank Sam Cooler for his help in building the ns3 simulations.

References

[1] J. I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving Single Channel, Full Duplex Wireless Communication," in *Proc. of ACM MobiCom*, 2010, pp. 1–12.

[2] M. Duarte and A. Sabharwal, "Full-Duplex Wireless Communications Using Off-The-Shelf Radios: Feasibility and First Results," in *the Forty Fourth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, nov. 2010, pp. 1558 –1562.

[3] M. Jain, J. I. Choi, T. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha, "Practical, Real-time, Full Duplex Wireless," in *Proc. of ACM MobiCom*, 2011, pp. 301–312.

[4] E. M. Dinesh Bharadia and S. Katti, "Full Duplex Radios," in *Proc. of ACM SIGCOMM*, 2013.

[5] N. Singh, D. Gunawardena, A. Proutiere, B. Radunovic, H. V. Balan, and P. B. Key, "Efficient and Fair MAC for Wireless Networks with Self-interference Cancellation," in *International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, 2011, pp. 94–101.

[6] S. Sen, R. Roy Choudhury, and S. Nelakuditi, "CSMA/CN: Carrier Sense Multiple Access with Collision Notification," in *Proc. of ACM MobiCom*, 2010, pp. 25–36.

[7] X. Zhang and K. G. Shin, "E-MiLi: Energy-Minimizing Idle Listening in Wireless Networks," in *Proc. of ACM Mobicom*, 2011, pp. 205–216.

[8] E. Magistretti, O. Gurewitz, and E. W. Knightly, "802.11ec: Collision Avoidance without Control Messages," in *Proc. of ACM MobiCom*, 2012, pp. 65–76.

[9] M. Vutukuru, K. Jamieson, and H. Balakrishnan, "Harnessing Exposed Terminals in Wireless Networks," in *Proc. of USENIX NSDI*, Berkeley, CA, USA, 2008, pp. 59–72.

[10] M. Sha, G. Xing, G. Zhou, S. Liu, and X. Wang, "C-MAC: Model-Driven Concurrent Medium Access Control for Wireless Sensor Networks," in *Proc. of IEEE INFOCOM*, 2009, pp. 1845–1853.

[11] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler, "Exploiting the Capture Effect for Collision Detection and Recovery," in *EEE EmNets Workshop*, 2005, pp. 45–52.

[12] R. Jain, D.-M. Chiu, and W. Hawe, "A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems," *Technical Report, Digital Equipment Corporation, DEC-TR-301*, 1984.

[13] F. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II–The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution," *IEEE Transactions on Communications*, vol. 23, pp. 1417 – 1433, dec 1975.

[14] P. Karn, "MACA: A New Channel Access Method for Packet Radio," in *Proc. of the 9th ARRL Computer Networking Conference*, vol. 9th, 1990, pp. 134–140.

[15] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a Media Access Protocol for Wireless LAN's," in *Proc. of ACM SIGCOMM*, 1994, pp. 212–225.

[16] A. Acharya, A. Misra, and S. Bansal, "MACA-P: a MAC for Concurrent Transmissions in Multi-hop Wireless Networks," in *Proc. of IEEE PERCOM*, 2003, pp. 505–508.

[17] M. Cesana, D. Maniezzo, P. Bergamo, and M. Gerla, "Interference Aware (IA) MAC: an Enhancement to IEEE 802.11b DCF," in *Proc. of IEEE VTC*, vol. 5, oct. 2003, pp. 2799 – 2803.

[18] Z. J. Haas, S. Member, J. Deng, and S. Member, "Dual Busy Tone Multiple Access (DBTMA) - A Multiple Access Control Scheme for Ad Hoc Networks," in *IEEE Transactions on Communications*, 2002, pp. 975–985.

[19] K. Mittal and E. M. Belding, "RTSS/CTSS: Mitigation of Exposed Terminals in Static 802.11-Based Mesh Networks," in *Proc. of IEEE WiMesh Workshop*, Sept. 2006.

[20] X. Wu, S. Tavildar, S. Shakkottai, T. Richardson, J. Li, R. Laroia, and A. Jovicic, "FlashLinQ: A Synchronous Distributed Scheduler for Peer-to-Peer Ad Hoc Networks," in *2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2010, pp. 514–521.

[21] S. B. Eisenman and A. T. Campbell, "E-CSMA: Supporting Enhanced CSMA Performance in Experimental Sensor Networks using Per-neighbor Transmission Probability Thresholds," in *Proc. of IEEE INFOCOM*, 2007, p. 12081216.

[22] T.-S. Kim, H. Lim, and J. C. Hou, "Improving Spatial Reuse Through Tuning Transmit Power, Carrier Sense Threshold, and Data Rate in Multihop Wireless Networks," in *Proc. of ACM MobiCom*, 2006, pp. 366–377.

[23] J. Zhu, S. Roy, X. Guo, and W. S. Conner, "Maximizing Aggregate Throughput in 802 . 11 Mesh Networks with Physical Carrier Sensing and Two-Radio Multi-Channel Clustering," in *Proc. of NSF-RPI Workshop on Pervasive Computing and Networking*, April 2004.

[24] J. Monks, V. Bharghavan, and W.-M. Hwu, "A Power Controlled Multiple Access Protocol for Wireless Packet Networks," in *Proc. of IEEE INFOCOM*, 2001, pp. 219 –228.

[25] A. Sahai, G. Patel, and A. Sabharwal, "Pushing the limits of Full-duplex: Design and Real-time Implementation," *Technical Report TREE1104, Rice University*, 2011.

[26] R. Gold, "Optimal Binary Sequences for Spread Spectrum Multiplexing," *IEEE Transactions on Information Theory*, vol. 13, pp. 619 –621, october 1967.

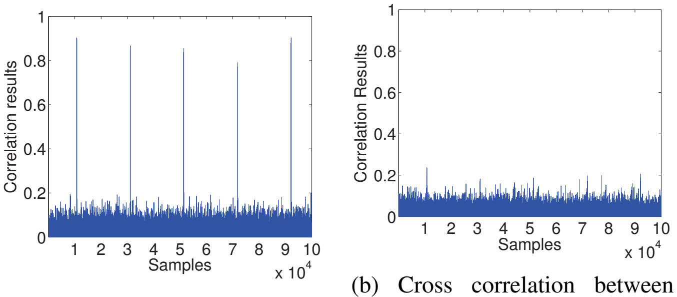
[27] S. W. Golomb, *Shift Register Sequences*. Aegean Park Press, 1981.

[28] "GNU Radio," accessed Jan 2013, http://gnuradio.org.

APPENDIX A
ROBUST SIGNATURES: FORMATION AND DETECTION

The spread spectrum nature of the signatures allows for their detection even in the presence of significant noise and interference (down to -10 dB in our experiments), without requiring precise frequency and timing correction. This enables rapid detection of the signals and facilitates alignment of the full duplex transmissions. Let the signature signal be $x[k]$, $k = 0, 1, 2, ..., N-1$. The corresponding received signal can be represented as $y[k] = e^{-2\pi jk\Delta f/f_s} h[k]x[k] + n[k]$, where Δf is the frequency offset between the transmitter and receiver, f_s is the sampling frequency, $h[k]$ is the channel coefficient, and $n[k]$ is noise. As the duration over which the signature is transmitted is relatively small, we assume that the channel remains unchanged over that period, i.e., $h[k] \approx H$. When the corresponding signature is not received, the cross correlation of x and y, $\mathrm{Corr}(x, y)$, is expected to be low as they are independent. But when the signature appears in the received signal, we have:

$$\begin{aligned} Corr(x,y) &= \Sigma_{k=0}^{k=N-1} x[k]^* y[k] \\ &= \Sigma_{k=0}^{k=N-1} x[k]^* (e^{-2\pi jk\Delta f/f_s} Hx[k] + n[k]) \\ &= H\Sigma_{k=0}^{k=N-1} e^{-2\pi jk\Delta f/f_s} |x[k]|^2 \\ &\quad + \Sigma_{k=0}^{k=N-1} x[k]^* n[k] \end{aligned} \qquad (6)$$



(a) Self correlation of code 0

(b) Cross correlation between code 0 and code 3

Fig. 10: Self correlation and cross correlation for Gold codes

In Equation (6), because $x[k]$ and $n[k]$ are independent,

$$\lim_{N\to\infty} \Sigma_{k=0}^{k=N-1} x[k]^* n[k] = 0 \qquad (7)$$

Observe that

$$\lim_{N\Delta f/f_s \to 0} e^{-2\pi jk\Delta f/f_s} = 1, k = 0, 1, .., N-1 \qquad (8)$$

With a fairly large N, we could still make $N\Delta f/f_s$ a very small value because $\Delta f/f_s$ is close to 0 (no larger than 2.5×10^{-6} for USRP 2). If the length of signatures, N, is selected properly, both Equation (7, 8) could be around their limitations. Then we have,

$$Corr(x,y) \approx H\Sigma_{k=0}^{k=N-1} |x[k]|^2 \qquad (9)$$

To minimize false positives, the cross correlation between different signatures should be low compared with the self correlation. Gold code [26] is a good candidate because the cross correlation of different pairs of codes is low and bounded. To generate Gold codes, two m-sequences [27] of the same length, $(2^L - 1)$, are selected, such that the absolute value of the cross correlation between them is bounded to $2^{\frac{L+2}{2}} + 1$, where L is the length of the m-sequences generator. These two m-sequences are called preferred m-sequences. Using the XOR of one m-sequence with shifted versions of the other m-sequences, we can get $2^L - 1$ new Gold codes. The cross correlation for Gold codes is bounded by $2^{\frac{L+2}{2}} + 1$ for even L and $2^{\frac{L+1}{2}} + 1$ for odd L, while the self-correlation value is $(2^L - 1)$.

Experimental Setup: To evaluate the performance of our signature design, we use two USRP platforms with the GNU Radio[28] software. For the Gold codes, a generator length of 7 is used which results in a total of 129 different codes with a length of 127. It takes 6.35μs in a 20MHz wireless channel and can support up to 129 nodes in one contention domain. The transmitter continuously sends code 0 every 0.01s. The average SNR at the receiver is 7dB. We use codes 0 and 3 to perform the correlation at the receiver side and normalize the correlation values to 1.

The results are shown in Figure 10. We can see distinct self-correlation peeks while the values of cross correlation and correlation with noise are relatively low.