UC Santa Cruz

UC Santa Cruz Previously Published Works

Title

Collaborative Collision Detection with Half-Duplex Radios

Permalink

<https://escholarship.org/uc/item/81n6083x>

Authors

Garcia-Luna-Aceves, J.J.
Carvalho, Marcelo

Publication Date

2018-04-01

Peer reviewed

Collaborative Collision Detection with Half-Duplex Radios

J.J. Garcia-Luna-Aceves^{*,†} and Marcelo M. Carvalho^{**}

^{*}Computer Engineering Department, UC Santa Cruz, Santa Cruz, CA 95064

[†]PARC, Palo Alto, CA 94304

^{**}Department of Electrical Engineering, University of Brasilia, Brasilia, DF, Brazil 70919-970
jj@soe.ucsc.edu, mmcvalho@ene.unb.br

Abstract—We present and analyze CSMA/CDS, a simple extension of carrier-sense multiple access (CSMA) that attains collision detection (CD) through the collaboration among nodes endowed with half-duplex radios in wireless local area networks (WLAN). CSMA/CDS uses two mechanisms to provide feedback to nodes attempting to transmit data packets concurrently. Using carrier sensing to access the channel, a node transmits a common pilot before sending its data packet, and a designated passive listener sends a collision pilot if it does not hear the common pilot after detecting carrier. This ensures that no collisions of data packets can occur. The throughput of CSMA/CDS is analyzed and compared with the performance of CSMA with priority ACKs and CSMA with collision detection (CSMA/CD), which assumes full-duplex transceivers. The results show that the throughput of CSMA/CDS is much better than the throughput of CSMA and is comparable to that of CSMA/CD.

I. INTRODUCTION

The focus of this paper is contention-based channel access in wireless local area networks (WLAN), where Carrier-Sense Multiple Access (CSMA) [11] is widely used today. Different approaches have been proposed over the years to improve on the performance of CSMA in local area networks, and the most notable schemes are CSMA with collision detection (CSMA/CD) [12], collision resolution techniques in which feedback is provided to senders to control their retransmissions after collisions (e.g., [2], [6]), and the emulation of collision detection using half-duplex radios [5], [13]. Other approaches have focused on improving the performance of CSMA in the presence of hidden terminals and include using collision-avoidance handshakes to avoid collisions [10], [4] and using busy tones in control channels to inform transmitters when receivers are busy (e.g., [14]). In addition, recent work on full-duplex radios has made much progress; however, we focus on the use of half-duplex radios available today. Section II summarizes prior work on channel-access schemes designed to work efficiently in WLANs in which nodes are in radio connectivity with each other.

Section III describes **CSMA/CDS** (Carrier-Sense Multiple Access with Collision Detection Shared), which constitutes a simple extension of CSMA for WLANs, which we call *collaborative collision detection*. Collaborative collision detection consists of two mechanisms to provide feedback to nodes attempting to transmit data packets concurrently. Using carrier

sensing to access the channel, a node transmits a common pilot before sending its data packet, and a designated passive listener echoes the signal it hears. The use of pilots and the pilot feedback from designated passive listeners are designed to ensure that no data packet collides with other transmissions.

Section IV compares the throughput attained with CSMA/CDS against the throughput attained with CSMA with priority ACKs and CSMA/CD with priority ACKs. In contrast to prior studies of CSMA, our analysis takes into account the latencies incurred by the receive-to-transmit turnaround times of half-duplex radios. As the results of the analysis show, CSMA/CDS is far more efficient than CSMA and its throughput is comparable to that of CSMA/CD while requiring much cheaper radios. Section V presents our conclusions.

II. RELATED WORK

Carrier sensing (CSMA) [11] and the many contention-based channel-access methods that have followed were developed to solve the performance problems with ALOHA [1], which stem from the large vulnerability period that any transmitted packet has. The basic approach to improving the performance in ALOHA consists of reducing the vulnerability period of a transmitted packet while incurring as little overhead as possible in doing so.

Several methods have been proposed for resolving collisions immediately after they occur. The basic idea in all these schemes consists of taking advantage of the feedback available to the nodes that transmitted packets to control the way in which they retransmit after being informed that collisions took place. Early proposals were based on the transmission of data packets and ideal feedback on a secondary channel (e.g., [2], [6]) and more recent proposals integrated CA handshakes with collision resolution [7]. The disadvantage of all these approaches is the large overhead incurred in informing the nodes about the collisions and resolving all such collisions.

Few proposals have been reported on how to emulate CSMA/CD using half-duplex radios. Rom [13] proposed a channel-access protocol that detects collisions by means of pauses. A station that senses the channel busy defers transmission as in CSMA, a transmitter that senses the channel idle starts transmitting but pauses during transmission and senses the channel. If the channel is sensed idle, the sender completes

its transmission, otherwise, the sender continues to transmit for a minimum transmission duration to jam the channel. This approach cannot guarantee that data packets will not collide with other transmissions at the receiver if packets start at the same time or the transmit-to-receive turnaround times are not negligible, which is the case in all existing half-duplex radios used in 802.11 WLANs today.

FAMA-PJ [5] emulates CSMA/CD in the context of collision avoidance in a WLAN and prevents data packets from colliding with other transmissions. A transmitter sends an RTS if it detects no carrier in the channel, and listens for a period of time after its RTS to check for jamming signals sent by passive nodes that detected a collision. A passive listener that receives the signal from the one or multiple RTS's sent and is unable to decode an RTS successfully sends a jamming signal for a period of time that is long enough to ensure that active transmitters hear the jamming signals from passive listeners once they can start listening to the channel after sending their RTS's. A remaining limitation of FAMA-PJ is that too many passive nodes end up sending jamming signals.

III. CSMA/CDS

A. Motivation

The design of CSMA/CDS is based on the design of the original ALOHA system [1] in which a central node (called the MENEHUNE) receives the combined signal from all user nodes over an uplink channel and simply rebroadcasts what it receives back to the user nodes over a downlink channel. A user node in the ALOHA system detects that its packet collided with others when the signal it receives over the downlink differs from what it sent in the uplink. Hence, the central node provides the required feedback to user nodes regarding the fate of their transmissions.

The way in which the ALOHA system works is very different from the way in which CSMA/CD operates, where each node detects collisions individually by comparing the signal it sends with the signal it receives from the channel. What is of interest in the ALOHA system is that a node detects the occurrence of a collision of its packet with others based on feedback from another node that simply conveys a signal that differs from the signal that the node itself sent. In a way, the design of the ALOHA system can be viewed as an approach in which user nodes collaborate with the central node to detect collisions, while CSMA/CD distributes collision detection to each node and makes it an individual task.

Our design of CSMA/CDS is based on the notion of collaborative collision detection that, in our opinion, originated with the design of the ALOHA system. However, CSMA/CDS differs from the ALOHA scheme in that carrier sensing is used by nodes to access a single common channel, explicit ACKs are used to inform senders that their data packets have been received correctly, and no central transponder is required. The role of the central node in the ALOHA system is replaced in the CSMA/CDS design by designated passive listeners. The designated passive listener for a given transmission round can be the receiver of the previous data packet that was sent

successfully, the access node of the WLAN, a node selected according to a round-robin schedule or another fair method, or a dedicated proxy collision-detection node that simply provides feedback when collisions occur.

B. State Machine

Fig.1 illustrates one of many possible implementations of non-persistent CSMA/CDS assuming a specific approach to determine designated passive listeners. For simplicity, we assume that the access point of the WLAN or a chosen node is initialized to start in the designated PASSIVE state rather than the ACTIVE state. Every other node that is initialized waits for a period of time equivalent to a DIFS (DCF inter-frame space) as defined in IEEE 802.11. After that time, the node transitions to the ACTIVE state. This waiting period ensures that a node entering the WLAN learns about ongoing packet transmissions if they exist. It is also assumed that a single packet is passed to the MAC layer for transmission.

Fig. 1. Operation of active nodes in non-persistent CSMA/CDS

If a node is in the **ACTIVE** state, there is no carrier in the channel, and the node receives a packet to send, then it starts transmitting a pilot to the intended receiver and transitions to the **SEND** state. Alternatively, if the node detects carrier from a pilot, it transitions to the **REMOTE** state.

A node in the BACK-OFF state computes a random back-off time (TO_B) and transitions to the ACTIVE state after that time has elapsed. The back-off time TO_B is much longer than the time needed for an average data packet and an acknowledgment (ACK) to be exchanged. An exponential back-off discipline can be used to account for unsuccessful retransmission attempts for the same data packet and limit congestion.

A node in the REMOTE state remembers whether it has a local packet to send while in that state. The node transitions to the RECEIVE state if it detects the end of a pilot and the start of a data packet. If the node cannot detect the start of a data packet following the end of carrier for any reason, it transitions to the ACTIVE state if it has no local packet to send, and transitions to the BACK-OFF state if it has a local packet to send.

If a node in the ACTIVE state receives a local packet to transmit and no carrier is detected, the node transitions to the SEND state and transmits a pilot lasting ρ seconds to the intended receiver. After sending its pilot the node listens to the channel to determine if any collision pilot is present. If the node receives no collision pilot while listening to the channel in the SEND state for a pilot-waiting time of W seconds, it determines that the channel is clear, starts transmitting its data packet and transitions to the DATA state. On the other hand, the node determines that a collision occurred and transitions to the BACK-OFF state if it starts receiving a collision pilot or another pilot within the W seconds of the pilot-waiting time.

Once in the DATA state, a node waits for an ACK to its data packet. The node transitions to the BACK-OFF state if it receives no ACK. Alternatively, it transitions to the ACTIVE state after receiving the expected ACK.

If a node in the RECEIVE state cannot decode a data packet, it transitions silently to the ACTIVE state after a short timeout following the end of carrier if it has no local packet to send, or to the BACK-OFF state if it has a local packet to send. If a node in the RECEIVE state is able to decode a data packet for itself (shown simply as “DATA to self” in Fig. 1), it sends an ACK to the sender and transitions to the PASSIVE state, which makes the node become the designated passive listener until the next pilot-DATA-ACK exchange occurs.

A node in the PASSIVE state listens to the channel for pilots. The node transitions to the CLEAR state if it receives a valid pilot and waits to hear a valid ACK. The node transitions to the ACTIVE state if an ACK is received indicating that a new receiver is transitioning to the PASSIVE state. The node goes back to the PASSIVE state if it does not receive an ACK after a timeout long enough to allow the successful reception of the ACK. A node in the PASSIVE state transitions to the COLLISION state if it receives an invalid pilot, which is interpreted as the collision of multiple pilots. Once in the COLLISION state, the node transmits a collision pilot to inform nodes that have sent pilots that a collision occurred. The node transitions back to the PASSIVE state after transmitting its collision pilot.

C. Collaborative Collision Detection with Pilots

There are various ways to implement carrier detection at the physical layer [9], and combining at least two of them can allow passive listeners to determine when a collision has taken place. A pilot is simply a bit pattern that can be used at the physical layer to detect a collision using standard carrier sensing techniques.

A simple approach to carrier sensing based on energy detection consists of comparing the signal strength readings obtained from the radio front end (the Received Signal Strength Indicator or RSSI reading) and a fixed or time-averaged carrier-sense threshold used as the noise floor corresponding to the channel being idle. A node determines that the channel is busy when the instantaneous RSSI reading is larger than its carrier-sense threshold.

Another carrier-sensing approach is based on preamble detection. In a nutshell, a transmission begins with a unique preamble sequence, such that there is a high likelihood that a transmission is taking place if the radio decodes a preamble.

We assume the combined use of preamble detection and energy detection as an example of how collaborative collision detection using pilots can be implemented in practice. A pilot can be defined to be an empty packet, i.e., a preamble at the physical-layer convergence procedure followed by start and end frame delimiters. A typical length of 160 bits would suffice for this purpose according to the IEEE 802.11 standard; however, as we discuss subsequently, we use a much longer duration for pilots for convenience.

A designated passive listener receives the aggregate of the pilots being transmitted. Consequently, if only one pilot is present in the channel, the listener decodes it with high probability into the pre-defined bit sequence for a pilot. Alternatively, if multiple pilots are being transmitted, the listener is either unable to decode the signal or obtains a bit sequence that differs from the bit sequence indicating a pilot. In the latter case, even if a passive listener is unable to decode a pilot, the RSSI reading informs the listener that a transmission is taking place and hence a collision can be inferred after the time corresponding to a pilot has elapsed without a decoded bit sequence and an RSSI reading indicating a busy channel.

Fig. 2. Collaborative collision resolution in CSMA/CDS using pilots

Fig. 2 illustrates an example of how collaborative collision detection works in CSMA/CDS using the approach we summarized above. Nodes a , b , and c transmit pilots at different times and they have different propagation delays between each other. Node p is the designated passive listener in the example, and it is assumed to be at a maximum propagation delay from the three other nodes. We assume that a pilot lasts ρ seconds and p uses preamble detection and energy detection to detect the presence of carrier. Hence, p determines that a collision occurs when its RSSI reading is larger than the carrier-sense threshold for ρ seconds and is unable to decode a pilot.

The common pilot and the interference pilot are of length $\rho = 2(\omega + \tau)$, where τ is the maximum propagation delay

allowed in the WLAN and ω is the maximum transmit-to-receive turnaround time. According to current 802.11 systems ω can be much larger than τ and in such a case makes nodes unable to detect the presence of other pilots immediately after their own pilot transmissions.

The pilot-waiting time in CSMA/CDS is set to $W = \rho$. As Fig. 2 shows, all three nodes a , b and c incur turnaround times of ω seconds, and start receiving the interference pilot from p within $\omega + 2\tau$ seconds from the end of their pilot transmissions, which is well before they are being able to start transmitting data packets. Furthermore, it is apparent from the figure that, even if the turnaround times for a , b and c were all 0 and the turnaround time were ω for node p , the waiting time W imposed on nodes that transmit a pilot ensures that they hear the interference pilot from p and detect the collision of their pilots. The following theorem proves that this is the case in general with the conservative assumption that $\rho = 2(\omega + \tau)$ and the pilot-waiting time is $W = \rho$.

Theorem 1: CSMA/CDS ensures that no data packets collide with other transmissions if pilots last $\rho = 2(\omega + \tau)$ and the pilot waiting time W is ρ , where τ is the maximum propagation delay allowed in the WLAN and ω is the maximum transmit-to-receive turnaround time.

Proof: Assume that nodes $n_1, n_2, \dots, n_k$ transmit pilots concurrently, and let node p be the designated passive listener. Let time t_1 be the time when node n_1 starts transmitting the first pilot in the group of pilots and t_i be the time when node n_i starts transmitting its pilot, where $n_i \in \{n_2, \dots, n_k\}$.

Because the maximum turnaround time is ω and nodes use carrier sensing, $t_1 \leq t_i \leq t_1 + \omega + \tau$.

Before node n_1 can decide whether it can transmit a data packet it must wait until time $W_1 = t_1 + \rho + W = t_1 + 4(\omega + \tau)$. Similarly, node n_i must wait until time $W_i = t_i + \rho + W$ to decide if it can transmit a data packet, and $t_1 + 4(\omega + \tau) \leq W_i \leq t_1 + 5(\omega + \tau)$. Hence, the earliest any node $n_i \in \{n_2, \dots, n_k\}$ can decide to transmit a data packet is W_1 .

Node p starts receiving the pilot from n_i at time S_i^p and receives the entire pilot from n_1 at time $E_1^p = S_1^p + \rho$.

If $\{n_2, \dots, n_k\} \neq \emptyset$, the pilot from n_1 overlaps with the pilot from node $n_i \in \{n_2, \dots, n_k\}$ at node p for $O_i^p = |E_1^p - S_i^p|$ seconds.

Because the maximum propagation delay is τ , we have that $S_i^p \leq t_i + \tau \leq t_1 + (\omega + \tau) + \tau$. On the other hand, because propagation delays are non-zero, $E_1^p > t_1 + 2(\omega + \tau)$. It thus follows that $O_i^p > t_1 + 2(\omega + \tau) - [t_1 + (\omega + \tau) + \tau] = \omega$.

Accordingly, if $\{n_2, \dots, n_k\} \neq \emptyset$, node p must decide that a collision took place and it must transmit a collision pilot after incurring a turnaround delay.

Let T_1 be the time when p detects a collision, then $T_1 \leq t_1 + \tau + \rho = t_1 + 2(\omega + \tau) + \tau$, because the maximum propagation delay is τ . Therefore, the collision pilot from p reaches $n_i \in \{n_1, n_2, \dots, n_k\}$ no later than time $T_{cp}^i \leq T_1 + \omega + \tau \leq t_1 + 3(\omega + \tau) + \tau$. Accordingly, we have that $T_{cp}^i < W_1$ and hence all nodes $n_1, n_2, \dots, n_k$ must receive the collision pilot

from p before any of them is able to transmit a data packet. Therefore, the theorem is true. $\square$

IV. THROUGHPUT ANALYSIS

We analyze the non-persistent versions of CSMA/CDS, CSMA with ACKs, and CSMA/CD using the traffic model first introduced by Kleinrock and Tobagi [11]. We assume the use of priority acknowledgments (ACK) in all protocols, because they are needed to account for transmission errors not due to multiple-access interference.

There is a large number of stations that constitute a Poisson source sending data packets to the channel with an aggregate mean generation rate of λ packets per unit time. Each node is assumed to have at most one data packet to be sent at any time, which results from the MAC layer having to submit one packet for transmission before accepting the next packet. A node retransmits after a random retransmission delay that on the average is much larger than the time needed for a successful transaction between a transmitter and a receiver and such that all transmissions can be assumed to be independent of one another. The channel is assumed to introduce no errors, so multiple access interference (MAI) is the only source of errors. Nodes are assumed to detect carrier perfectly.

We assume that two or more transmissions that overlap in time in the channel must all be retransmitted (i.e., there is no power capture by any transmission), and that any packet propagates to all nodes in exactly τ seconds. The transmit-to-receive and receive-to-transmit turn-around times in the data channel required by a node is ω and is assumed to be larger than the propagation delay τ , which agrees with the parameters assumed in IEEE 802.11 DCF. The transmission time of a data packet is δ and the transmission time of an ACK is α . For the case of CSMA/CD, the time of a jamming bit sequence is η , which has to be larger than the error-checking field of a packet (e.g., 48 bits). We assume that processing delays are negligible, which includes the time ξ to detect carrier or do collision detection. The protocols are assumed to operate in steady state, with no possibility of collapse, and hence the average channel utilization of the channel is given by [11]

$$S = \frac{\bar{U}}{\bar{B} + \bar{I}}. \quad (1)$$

where $\bar{B}$ is the expected duration of a busy period, defined to be a period of time during which the channel is being utilized; $\bar{I}$ is the expected duration of an idle period, defined as the time interval between two consecutive busy periods; and $\bar{U}$ is the time during a busy period that the channel is used for transmitting user data successfully.

A. Non-Persistent CSMA/CDS

Fig. 3 illustrates the idle and busy periods that may occur with CSMA/CDS under the modeling assumption that all nodes are τ seconds from one another. We assume that the length of a pilot ρ and the length of a pilot-waiting period W are $W = \rho = 2(\omega + \tau)$.

Fig. 3. Transmission periods in CSMA/CDS

Theorem 2: The throughput of non-persistent CSMA/CDS is

$$S_{CSMA/CDS} = \frac{\delta}{\delta + \alpha + \tau + e^{\lambda(\omega+\tau)} (5(\omega + \tau) + \frac{1}{\lambda})} \quad (2)$$

Proof: A node with a packet to send must first sense the channel before transmitting a pilot, which forces the node to incur a turnaround time ω during which it is unable to listen to the channel. Therefore, the vulnerability period of a pilot is $\omega + \tau$, and the probability that a pilot is transmitted without interference from other pilots $P_S = e^{-\lambda(\omega+\tau)}$.

If a pilot is sent without multiple access interference, the designated passive listener remains silent and the sender of the pilot transmits its data packet after W seconds. The intended receiver transmits its ACK back to the sender after receiving the data packet and incurring a turnaround time. Accordingly, the length of a successful PILOT-DATA-ACK exchange T lasts $T = \rho + W + \delta + \alpha + \omega + 2\tau$, and with the assumptions we made for W and ρ we have $T = \delta + \alpha + 5\omega + 6\tau$.

On the other hand, if multiple pilots collide with one another, the designated passive listener D determines that there is a collision ρ seconds after it starts receiving the start of the first pilot in the collision interval, incurs a turnaround time and sends a collision pilot that propagates to all nodes, which takes $2\rho + \omega + \tau = 5(\omega + \tau)$ seconds. The last pilot in the collision interval starts arriving at D no later than $\omega + \tau$ seconds after the first pilot in the interval and the end of the last pilot occurs no later than $\omega + \tau + \rho = 3(\omega + \tau)$ seconds. Therefore, the length of a collision interval C equals $C = 5(\omega + \tau)$.

The average length of a busy period is simply

$$\bar{B} = T(P_S) + C(1 - P_S)$$

Substituting the values for P_S , T , and C we have

$$\begin{aligned} \bar{B} &= e^{-\lambda(\omega+\tau)} (\delta + \alpha + 5\omega + 6\tau) + (1 - e^{-\lambda(\omega+\tau)}) (5(\omega + \tau)) \\ &= 5(\omega + \tau) + e^{-\lambda(\omega+\tau)} (\delta + \alpha + \tau) \end{aligned} \quad (3)$$

The average length of an idle period $\bar{I}$ is simply the average inter-arrival time of packets, which are preceded by pilot transmissions, and this equals $1/\lambda$ because inter-arrival times are exponentially distributed with parameter λ .

The average time period used to transmit useful data $\bar{U}$ is the useful portion of a *successful* busy period, which occurs with probability $e^{-\lambda(\omega+\tau)}$. Accordingly, $\bar{U} = \delta e^{-\lambda(\omega+\tau)}$.

Substituting the values of $\bar{U}$, $\bar{B}$, and $\bar{I}$ into Eq. (1) we obtain Eq. (2). $\square$

B. Non-Persistent CSMA/CD

Fig. 4 illustrates the transmission periods for non-persistent CSMA/CD with the assumption that nodes use SIC. No turnaround latencies are incurred in CSMA/CD given that nodes can listen to the channel while they transmit.

Theorem 3: The throughput of non-persistent CSMA/CD is

$$S_{CSMA/CD} = \frac{\delta}{\delta + \alpha - \eta - \frac{1}{\lambda} + e^{\lambda\tau} (\eta + 2\tau + \frac{2}{\lambda})} \quad (4)$$

Proof: The proof is presented in [8]. $\square$

Fig. 4. Transmission periods in non-persistent CSMA/CD

C. Non-Persistent CSMA with Priority ACKs

Figure 5 illustrates the transmission periods in non-persistent CSMA with priority ACKs and the impact of turnaround times.

Kleinrock and Tobagi [15] derived the throughput of CSMA with priority ACKs assuming that all processing delays are negligible. However, with data rates of megabits per second and distances of a few hundred meters, the turnaround times of half-duplex radios are not negligible and must be taken into account, because they increase the vulnerability periods of transmitted packets.

Fig. 5. Transmission periods in non-persistent CSMA with priority ACKs

Theorem 4: The throughput of non-persistent CSMA with priority ACKs is given by

$$S_{CSMA} = \frac{\delta}{\alpha + \omega + \tau + \frac{1}{\lambda} + e^{\lambda(\omega+\tau)} (\delta + \omega + 2\tau)} \quad (5)$$

Proof: The proof is presented in [8]. $\square$

D. Numerical Results

We present numerical results for channel data rates of 1 Mbps for simplicity. An ACK is assumed to be 14 bytes as in the IEEE 802.11 standard. The time needed to send a jamming signal (η) in CSMA/CD is roughly the duration of a jamming signal in Ethernet, or 48-bit time. We normalize the results to the length of a data packet by making $\delta = 1$, $G = \lambda \times \delta$, and $t = \tau/\delta$; and by using the normalized value of each other variable, which equals its ratio with δ .

We compare the throughput (S) versus the offered load (G) attained by CSMA/CDS, CSMA/CD, and CSMA based on

Eqs. (2), (4), and (5), and present results for two scenarios. A local-area scenario consists of a network with 300-meter range and data packets of 1500 bytes, and a geographically-dispersed scenario consists of a network with a 1000-meter range and 100-byte data packets. Figures 6 and 7 show the results. Solid lines show results for turnaround times of $2\mu\text{s}$, and dashed lines show the results for turnaround times of $10\mu\text{s}$.

Fig. 6. S vs. G with short distances and large data packets

Fig. 7. S vs. G with long distances and medium-size data packets

It is clear from the figures that CSMA/CDS provides much better throughput than CSMA with priority ACKs. It is also clear that the throughput of CSMA/CDS approaches that of CSMA/CD as propagation delays and turnaround times decrease, which reduces the overhead incurred using pilots.

CSMA/CD attains higher throughput values than CSMA/CDS does because of the overhead incurred by the use of pilots in CSMA/CDS, which is a necessity in order to enable collision detection with half-duplex radios. However, CSMA/CDS still offers a very good tradeoff between performance and the simplicity of the radios needed. CSMA/CDS can be implemented using existing half-duplex radios and carrier-sense mechanisms, while implementing CSMA/CD in WLANs requires the introduction of radios

capable of enabling collision detection by individual nodes using such physical-layer techniques as self-interference cancellation.

V. CONCLUSIONS AND FUTURE WORK

We introduced CSMA/CDS to enable collision detection in WLANs of nodes with half-duplex radios. CSMA/CDS makes nodes collaborate with each other in detecting collisions using standard carrier-detection techniques used to implement CSMA today. We proved that no data packet collisions can occur with CSMA/CDS, and compared the throughput attained with CSMA/CDS, CSMA/CD, and CSMA with priority ACKs and a non-persistent transmission policy. The results show that collaborative collision detection provides better throughput than basic carrier sensing and offers a good tradeoff between channel performance and the simplicity of radios compared to CSMA/CD, which requires full-duplex radios.

Collaborative collision detection could be used in combination with collision-avoidance handshakes in networks with hidden terminals. The design and analysis of the resulting schemes, and the use of these schemes in the context of the IEEE 802.11 standard are the subject of future work.

ACKNOWLEDGMENTS

This work was supported in part by the Jack Baskin Chair of Computer Engineering at UC Santa Cruz, and by the Research Support Foundation of the Federal District (FAPDF), Brazil.

REFERENCES

- [1] N. Abramson, "The ALOHA System—Another Alternative for Computer Communications," *Proc. Fall Joint Computer Conference '70*, 1970.
- [2] J.I. Capetanakis, "Tree Algorithm for Packet Broadcast Channels," *IEEE Trans. Information Theory*, Sept. 1979.
- [3] K.-C. Chen, "Medium Access Control of Wireless LANs for Mobile Computing," *IEEE Network*, vol. 8, no. 5, pp. 50–63, 1994.
- [4] C. L. Fullmer and J.J. Garcia-Luna-Aceves, "Solutions to Hidden Terminal Problems in Wireless Networks," *Proc. ACM SIGCOMM '97*, 1997.
- [5] C. L. Fullmer and J.J. Garcia-Luna-Aceves, "FAMA-PJ: a channel access protocol for wireless LANs," *Proc. ACM MobiCom '95*, 1995.
- [6] R. Gallager, "Conflict Resolution in Random Access Broadcast Networks," *Proc. AFOSR Workshop on Comm. Theory and applications*, 1978.
- [7] R. Garces, J.J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access with Collision Resolution," *Proc. ACM MobiCom '96*, 1996.
- [8] J.J. Garcia-Luna-Aceves, "Carrier Resolution Multiple Access," *Proc. ACM PE-WASUN '17*, Nov. 2017.
- [9] K. Jamieson et al., "Understanding The Real-World Performance of Carrier Sense," *Proc. ACM E-WIND '05*, 2005.
- [10] P. Karn, "MACA—A New Channel Access Method for Packet Radio," *Proc. ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, 1990.
- [11] L. Kleinrock and F. A. Tobagi, "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Trans. Commun.*, 1975.
- [12] R. M. Metcalfe and D. R. Boggs, "ETHERNET: Distributed packet switching for local computer networks," *CACM*, vol. 19, no. 7, pp. 395 – 403, 1976.
- [13] R. Rom, "Collision Detection in Radio Channels," *Local Area and Multiple Access Networks*, Computer Science Press, 1986.
- [14] F. A. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II - The Hidden Terminal Problem in Carrier Sense Multiple-Access Modes and The Busy-Tone Solution," *IEEE Trans. Commun.*, 1975.
- [15] F. Tobagi and L. Kleinrock, "The Effect of Acknowledgment Traffic on the Capacity of Packet-Switched Radio Channels," *IEEE Trans. Commun.*, 1978.