UC Santa Cruz

UC Santa Cruz Previously Published Works

Title

Avoiding Interference from Hidden Terminals with Carrier Tones

Permalink

<https://escholarship.org/uc/item/5wk7c9jr>

Author

Garcia-Luna-Aceves, J.J.

Publication Date

2017-12-10

Peer reviewed

Avoiding Interference from Hidden Terminals with Carrier Tones

J.J. Garcia-Luna-Aceves

Computer Engineering Department, University of California, Santa Cruz, CA 95064

PARC, Palo Alto, CA 94304

jj@soe.ucsc.edu

Abstract—A new approach to collision avoidance is described that is based on the use of carrier tones sent over a single control channel. A carrier tone can indicate that the sender is busy transmitting a request-to-send (RTS), the receiver is busy receiving a data packet, or the receiver is sending an acknowledgment (ACK). RTS, data packets, and end of transmission (ET) packets are sent over the primary data channel without using carrier sensing. The throughput of the resulting protocol, which we call Carrier-Tone Multiple Access (CTMA), is compared with the throughput of CSMA with collision avoidance (CSMA/CA) protocol and the dual busy-tone multiple access (DBTMA) protocol. The results of the analysis show that CTMA is a more efficient channel-access scheme than the other two approaches.

I. INTRODUCTION

Carrier sense multiple access (CSMA) [12] is widely used for the sharing of radio channels in ad-hoc networks. While carrier sensing results in a marked improvement over the ALOHA protocol, the performance of CSMA degrades substantially in the presence of hidden terminals [15]. Many approaches have been developed over the years to limit or eliminate multiple-access interference (MAI) due to hidden terminals.

As Section II discusses, the two types of approaches that have been proposed in the past to cope with multiple-access interference from hidden terminals can be classified into collision-avoidance schemes (e.g., [3], [11]) and busy-tone schemes [10], [15]. The focus of this paper is on the combined use of collision-avoidance (CA) handshakes executed over the data channel with busy tones transmitted over a secondary control channel to limit the MAI interference resulting from hidden terminals.

Section III introduces **CTMA** (*Carrier-Tone Multiple Access*), which combines collision-avoidance (CA) handshakes with carrier tones sent by both sender and receiver to eliminate the negative effects of hidden terminals. The channel is partitioned into a data channel and a small control channel, and half-duplex radios are used to access each channel. Nodes use the control channel to detect interference. To transmit a data packet, a node that detects no carrier tones in the control channel sends a short request-to-send (RTS) packet over the data channel and a short sender tone (ST) in the control channel. If the RTS is received correctly, the receiver starts transmitting a receiver tone (RT) over the control channel that lasts until the receiver obtains the entire data packet

from the sender, after which it changes its RT to a short acknowledgment tone (AT). The sender sends its data packet over the data channel after detecting an receiver tone (RT). To allow for variable-length data packets to be transmitted without MAI and without the need for passive listeners to use carrier sensing on the data channel, the sender transmits an end-of-transmission (ET) packet in the data channel after receiving the AT from the receiver.

Section IV shows that CTMA prevents the collision of data packets with other transmissions even in the presence of hidden terminals, and allows receivers to inform senders of the correct reception of data packets.

Sections V and VI analyze the throughput of CTMA and compare it against the throughput of CSMA with collision avoidance (CSMA/CA) and Dual Busy Tone Multiple Access (DBTMA) [10]. The results of the analysis shows that CTMA is more efficient than CSMA/CA and DBTMA. Section VII presents our conclusions.

II. RELATED WORK

The channel-access protocols that have been developed to date to address the negative effect of multiple-access interference (MAI) caused by hidden terminals in CSMA [15] can be classified in two types: protocols based on collision-avoidance (CA) handshakes and protocols based on busy tones sent over secondary channels.

One of the first channel-access protocols based on CA handshakes was Multiple Access with Collision Avoidance (MACA) [11], which consists of a transmitter sending a request-to-send (RTS) packet to an intended receiver and the receiver sending a clear-to-send (CTS) packet if the RTS is successful. MACA does not use carrier sensing and its performance degrades in the presence of hidden terminals. Many subsequent variants of channel-access protocols based on CA handshakes have been proposed and analyzed since the introduction of MACA and have integrated carrier sensing together with CA handshakes. In some schemes the transmitter initiates the handshake [2], [4], [3] and in others the receiver does (e.g., [6], [14]). The IEEE 802.11 distributed coordination function (DCF) is arguably the most popular example of combining carrier sensing with the RTS-CTS handshake followed by a data packet and an ACK in successful cases.

It has been shown that not every CA handshake suffices to avoid MAI from hidden terminals. The only known approach

that has been shown to succeed requires a CTS from a receiver to be much longer than the length of an RTS in order to serve as a busy tone in those cases in which multiple signaling packets are transmitted concurrently [3]. The limitation with this approach is that it works correctly only if either fixed-length data packets are transmitted or overly long back-off periods are applied, which renders lower throughput.

The Busy-Tone Multiple Access (BTMA) [15] eliminates multiple-access interference around a central receiver. The available channel is partitioned into a data channel and the busy-tone channel. The central receiver has radio connectivity with all other nodes in the system and transmits a busy tone over the busy-tone channel as soon as it detects carrier in the data channel resulting from transmissions from any subset of transmitters. A few channel-access protocols based on one or more busy tones have been proposed, and the handshakes used in these protocols eliminate MAI due to hidden terminals, at least to some degree.

In RI-BTMA (Receiver-Initiated Busy Tone Multiple Access) [17], the channel is divided into a data channel and a control channel and both channels are assumed to be time slotted, with each time slot lasting the length of a preamble. When the receiver detects the preamble of the transmission by the sender, it transmits its busy tone in the control channel. A major limitation of this approach is the need to use a time-slotted channel in a contention-based context.

DBTMA (Dual Busy Tone Multiple Access) [10] adopts an approach similar to RI-BTMA but does not require time slotting and uses two busy tones, each sent on a separate control channel. Before sending a data packet, a transmitter sends an RTS over the data channel together with a busy tone of the same duration as the RTS. A transmitter aborts its transmission if it detects a receiver busy tone or a transmitter busy tone. The receiver starts transmitting a busy tone after receiving an RTS, and this busy tone serves as a CTS and lasts as long as needed to protect the reception of a data packet by the receiver. The transmitter sends its data packet after detecting the receiver busy tone. An important feature of DBTMA is that nodes use busy-tone detection over the control channels as the means to determine whether the data channel is busy. However, although DBTMA attempts to solve the hidden-and-exposed-terminal problems, it is an incomplete solution because it does not consider the use of ACKs sent from receivers to transmitters. Another limitation of this approach is that it requires two busy-tone channels, which implies the use of two busy-tone radios operating concurrently in addition to the radio used for the data channel.

III. CTMA

A. Design Objective and Approach

We have recently introduced a variant of CTMA [8] that uses a receive tone and an acknowledgment tone sent by receivers to eliminate MAI from hidden terminals. The advantage of this prior instance of CTMA over all the previous busy-tone schemes is that it does not require time slotting and uses a single busy-tone radio and control channel. However,

nodes are required to use carrier sensing in the data channel and the control channel to determine when the data channel is busy. The design objective of the CTMA protocol described in this paper is threefold: (a) To eliminate the collision of data packets in the presence of hidden terminals by informing transmitters of the presence of interference at the receivers by means of receive tones sent by receivers over the control channel; (b) to inform senders when their packets are received successfully; and (c) to eliminate the need for carrier sensing being used in the data channel, even when variable-length data packets are sent. In contrast to DBTMA, the carrier tones used in CTMA are sent over a single control channel.

A receiver transmits a receiver tone (RT) to protect the reception of a data packet. A receiver transmits its RT as soon as it receives an RTS and transmits a short ACK tone (AT) instead as soon as it receives the data packet correctly. A sender transmits a short sender tone (ST) to enable other nodes to detect the channel being busy without using carrier sensing in the data channel. It serves the purpose of the virtual carrier-sensing mechanism or network allocation vector (NAV) in 802.11. More specifically, a node that detects an ST knows to defer accessing the channel until it hears an end-of-transmission (ET) packet over the data channel, or a long-enough time has elapsed. The sender transmits the short ET as soon as it detects the AT from the receiver.

Fig. 1 illustrates how CTMA works with an example. Sender T_1 transmits its short ST after detecting no carrier tones in the control channel and also sends its RTS. Receiver R_1 transmits its RT after receiving an RTS from transmitter T_1 without MAI. The occurrence of the ST in the control channel blocks node H from attempting to send an RTS and an ST, which would interfere with the reception of the ACK from R_1 . Node H is forced to wait until it hears an ET from T_1 . On the other hand, the presence of the RT in the control channel prevents node T_2 , which is hidden from transmitter T_1 , from attempting to send an RTS to another receiver R_2 or R_1 itself. The figure illustrates the fact that the carrier tones and ET from R_1 and T_1 protect the entire CA handshake between the two nodes from any other transmission that could create MAI. Nodes do not have to use carrier sensing in the data channel to determine that the channel is busy.

Fig. 1. Carrier tones block hidden terminals

B. Operation of Non-Persistent CTMA

The total available bandwidth is partitioned into a data channel and a narrowband control channel. Nodes exchange RTS's, data packets, and ET's in the data channel, and transmit

ST's, RT's, and AT's over the control channel depending on the state they reach in their CA handshakes. Nodes use carrier-tone sensing over the control channel before transmitting RTS's over the data channel, monitor the control channel for the presence of a carrier tone, and use the occurrence of an ST as the virtual carrier-sensing mechanism over the control channel.

Figure 2 illustrates the state machine of non-persistent CTMA, in which a transmitter backs off immediately after detecting a busy tone. It is assumed that a single packet is passed to the MAC layer for transmission at any given time.

Fig. 2. Non-persistent CTMA

A node that is just initialized waits for a period of time before it transitions to the PASSIVE state. This waiting period ensures that a node entering an ad-hoc network learns about ongoing packet transmissions if they exist. Once a node is in PASSIVE state, it listens to the control channel for the presence of a carrier tone.

A node in the BACK-OFF state computes a random back-off time (TO_B) and transitions to the passive state after that time has elapsed. The back-off time TO_B is much longer than the time needed for the longest-allowed data packet and an 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; however, it is not shown for simplicity and the back-off time is simply assumed to be much larger than the time needed for a successful RTS-RT-DATA-AT-ET transaction to take place.

If a node is in the PASSIVE state and detects an ST in the control channel without receiving an RTS for itself transitions to the BACK-OFF state. The same occurs when a node in the PASSIVE state detects an RT, an AT or noise resulting from multiple carrier tones sent concurrently over the control channel. If a node in the PASSIVE receives a local packet to transmit and no carrier tone is detected, the node sends an RTS to the intended receiver, transmits a short ST lasting only the time required for nodes to detect the tone, and transitions to the SEND state.

If a node in the SEND state detects no RT after an RTS timeout or detects a carrier tone other than an RT, the node assumes that a collision occurred and transitions to the BACK-OFF state to schedule its RTS at a future time. On the other hand, the node transitions to the DATA state if it receives an RT in the control channel.

Once in the DATA state, the node waits for d seconds before sending its data packet to the receiver in the data channel and waits for an AT over the control channel from the receiver. The purpose of the delay in sending the data packet is to avoid the possibility of MAI from RTS's sent by neighboring nodes. For this purpose, we assume that $d \geq 2\tau + \sigma$. The node transitions to the BACK-OFF state if it does not receive an AT within an ACK timeout that is long enough to allow the tone to be received and detected. On the other hand, if the node receives an AT over the control channel, it transmits an ET and transitions to the PASSIVE state.

If a node in the PASSIVE state decodes an RTS for itself (shown as "RTS to self" in Fig. 2) and no carrier tone is detected in the control channel, it starts transmitting its RT in the control channel and transitions to the RECEIVE state. A node in the RECEIVE state remembers whether or not it has a local packet to send. If a node in the RECEIVE state receives the expected data packet, it stops transmitting its RT and transmits a short AT instead. The node transitions to the PASSIVE state if it has no local packet to send, or to the BACK-OFF state if it has a local packet to send. Similarly, the node stops transmitting its RT and transitions to the PASSIVE or BACK-OFF state depending on whether it has a local packet to send if a RECEIVE timeout elapses with no data packet being received from the transmitter. The length of the timeout in the RECEIVE state is long enough for the node to be able to start decoding a valid data packet.

CTMA allows concurrent transmissions by neighboring transmitters and concurrent receptions by neighboring receivers to take place, provided that MAI is prevented by the carrier tones sent by senders and receivers engaged in CA handshakes.

Fig. 3 illustrates how two transmitters T_1 and T_2 that are neighbors of each other are allowed to transmit data packets concurrently to receivers R_1 and R_2 , which are hidden from one another. In this example, both transmitters send their RTS's around the same time, such that they do detect their ST's in the control channel before they start sending their RTS's in the data channel.

Fig. 3. Concurrency by neighboring transmitters in CTMA

Fig. 4 illustrates how two neighboring receivers R_1 and R_2 can receive data packets from transmitters T_1 and T_2 that are hidden from one another. In this example, the receivers transmit their RT's around the same time, so that they do not detect the presence of the tones sent from each other.

Fig. 4. Concurrency by neighboring receivers in CTMA

The following section proves that CTMA operates in such a way that concurrency is allowed only when no MAI can occur for data packets and AT's received in the data channel.

IV. CORRECTNESS OF CTMA

The following assumptions are made in discussing the ability of CTMA and other channel-access methods to eliminate multiple-access interference (MAI) for data packets and ACKs: (a) A node knows the addresses of its neighboring nodes through some means external to the channel-access protocol and nodes execute the protocol correctly; (b) the propagation delay t_p between any two neighboring nodes is $0 < t_p \leq \tau$; (c) every node detects the presence of a carrier tone correctly and the time needed to do so is σ ; (d) the transmission time of a data packet is δ , the transmission time of an RTS packet is γ , and the duration of an AT is $\alpha \geq \sigma$; and (e) the transmit-to-receive and receive-to-transmit turn-around times are equal to $\omega > 0$ seconds.

Fig. 5 illustrates the type of interference that a transmitter T and a receiver R may experience from neighboring nodes. Solid lines indicate radio connectivity and dashed lines indicate possible radio connectivity. Node n_R is a neighbor of R other than T that may be connected to or hidden from R , and node n_T is a neighbor of T other than R that may be connected to or hidden from T . Nodes x and y are neighbors of n_T and n_R , respectively, which may cause n_T and n_R to create interference at T or R .

Fig. 5. Nodes that may cause MAI for a receiver R and a transmitter T

For transmitter T to send a data packet to receiver R , the latter must receive the RTS from T free of MAI and T must receive the AT from R free of MAI. Accordingly, to show that CTMA eliminates MAI on data packets and ACKs we can assume that T and R complete an RTS-RT handshake successfully, and must then show that CTMA ensures that

R receives the data packet from T without MAI and that T receives the AT from R without MAI. The following theorem shows that this is the case.

Theorem 1: CTMA ensures that no data packets collide with any other transmissions at their intended receivers and that senders learn that their data packets are received correctly.

Proof: Without loss of generality, assume that T sends an RTS to R and that R sends an RT to T successfully.

After an RTS-RT exchange takes place between T and R , node T starts transmitting a data packet to R and the only way a neighbor n_T of T can interfere with the rest of the exchange between T and R is by interfering with the AT sent to T by R . However, according to the state machine of CTMA, a node that detects any carrier tone must transition to the WAIT state to hear an ET. Once in the WAIT state, n_T must either receive the ET from T or transition to the BACK-OFF state after a timeout interval that is much longer than the time needed for the sender T to receive an AT from R . Therefore, no neighbor of T is able to interfere with the AT from R to T .

Similarly, after the RTS-RT exchange between T and R is successful, the only way for a neighbor n_R of R to disrupt the rest of the exchange is for n_R to interfere with the data packet sent from T to R over the data channel. This would require n_R to send an RTS or a data packet, given that an ET can be sent only after a successful data packet is sent.

Let t_c be the time when R starts transmitting its RT. If n_R sends an RTS at time t_R , then $t_R \leq t_c + \sigma + \tau$ because n_R must transmit its RTS before detecting the RT from R and the propagation delay is at most τ seconds.

By design, an ST lasts only σ seconds. Hence, given that n_R incurs ω seconds of transmit-to-receive turnaround time in the control channel, it is able to hear carrier tones in the control channel at time $t_R + \sigma + \omega \leq t_c + \omega + 2\sigma + \tau$. By design, $\gamma > \omega + 2\sigma + \tau$, which means that n_R can detect the RT from R even before the end of its own RTS. From the state machine of CTMA, this means that n_R must transition to the BACK-OFF state because it must detect the RT from R while transmitting its RTS. Furthermore, the entire RTS from n_R must be received by R at time $t_r^R \leq t_c + \gamma + 2\tau + \sigma$ because n_R must send its RTS without detecting the RT from R and propagation delays are at most τ seconds. On the other hand, the data packet from T must arrive at R at time t_d^R and is $t_d^R > t_c + \gamma + d$, because propagation delays are non-zero and T must wait d seconds before sending a data packet after receiving an RTS. Given that $d \geq 2\tau + \sigma$, it follows that the RTS from n_R cannot interfere with the data packet from T arriving at R . Because n_R must transition to the BACK-OFF if it sends an RTS while R sends an RT successfully, it cannot send a data packet that could interfere with the data packet from T received by R .

Therefore, no neighbor of R is able to interfere with the exchange between T and R and hence it follows from the above that no neighbor n_T of T and no neighbor n_R of R can cause MAI for the data packet or the AT exchanged between T and R and the theorem is true. $\square$

As the previous theorem shows, the integration of CA handshakes with three types of carrier tones sent over a single control channel eliminates MAI for data packets and allows senders to know when their data packets are received successfully. The same is not the case for DBTMA or prior protocol based on CA handshakes, which eliminate MAI for the reception of data packets but do not consider the use of ACKs [3], [10].

V. THROUGHPUT IN FULLY-CONNECTED NETWORKS

We analyze the non-persistent versions of CTMA, CSMA/CA, and DBTMA with ACKs using the traffic model first introduced by Kleinrock and Tobagi [12] assuming a fully-connected network. We consider the use of priority ACKs in all protocols, because they are needed to account for transmission errors not due to multiple-access interference.

Assuming a fully-connected network results in a major simplification of the analytical model, because the channel is utilized in only three ways. The channel may be idle, or it may be busy with either the successful transmissions of data packets or a collision interval. A collision interval consists of overlapping RTS's only and the length of the interval is bounded by the vulnerability period of the first RTS in the interval, after which all nodes detect carrier and must back off. In a fully-connected network, the difference in the performance of the protocols we consider derives from the overhead each incurs during a successful transmission period and how quickly nodes determine that a collision took place.

A. Model and Assumptions

There is a large number of stations that constitute a Poisson source sending RTS's 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 send 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 of RTS's or data packets 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 or busy tones perfectly, and the probability of false busy-tone detection is 0.

The time required to detect any type of tone or the presence of overlapping tones is σ . The transmit-to-receive and receive-to-transmit turn-around times in the data channel and a control channel 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 RTS packet is γ . For the case of CTMA, the waiting time before sending a data packet is $d = 2\tau + \sigma$, the duration of an AT is σ , which is enough for nodes to detect the carrier tones, and an ET message lasts $\epsilon < \gamma$ seconds.

To further simplify the problem, 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 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 [12]

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

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.

B. Non-Persistent CTMA

Fig. 6 illustrates how CTMA uses ST's, RT's, AT's, and ET's as part of CA handshakes. The utilization of the channel in a fully-connected network using CTMA consists only of idle periods, successful busy periods during which data packets are sent as part of successful collision-avoidance handshakes, or collision intervals resulting from the collision of two or more RTS's.

Fig. 6. Transmission periods in non-persistent CTMA

The following theorem provides the throughput of non-persistent CTMA. In our analysis, we assume that data packets have a fixed length δ , which makes the use of ET's unnecessary because a passive listener that receives an RTS for another node knows how long to wait for the entire RTS-RT-DATA-AT to take place.

Theorem 2: The throughput of non-persistent CTMA over a data channel using only ζ (with $0 < \zeta < 1$) of the total available bandwidth is given by

$$S_{CTMA} = \frac{\delta}{\delta + \zeta^{-1} \left(\omega + 3\sigma + 5\tau + \frac{1}{\lambda} \right) + C e^{\lambda(\tau + \sigma)}} \quad (2)$$

where $C = \gamma + \zeta^{-1}(\sigma + 2\tau)$

Proof: According to the operation of CTMA, a node detects channel activity based on the transmissions that occur in the control channel. By assumption, it takes τ seconds for a signal to propagate from transmitters to receivers in the control and data channels, and it takes σ seconds for a node to detect the

presence of a carrier tone or the aggregate of multiple carrier tones in the control channel. Accordingly, a node may start an ST within $\tau + \sigma$ seconds of the first ST in a collision interval, and the vulnerability period of an RTS equals $\tau + \sigma$ seconds. Therefore, the probability with which an RTS is sent into the data channel without MAI is $P_S = e^{-\lambda(\tau+\sigma)}$, because RTS arrivals and hence ST arrivals are Poisson distributed with parameter λ .

The average length of an idle period $\bar{I}$ in CTMA is simply the average inter-arrival time of RTS's into the channel, which 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 an average *successful* busy period. Given that all nodes hear one another, a busy period is successful if an RTS has no MAI from other RTS's and it follows that $\bar{U} = \delta P_S = \delta e^{-\lambda(\tau+\sigma)}$.

If an RTS is sent without MAI, then an RT, a data packet, and an AT follow as part of the successful transmission period, with each AT lasting σ seconds, as can be seen from Fig. 6 with no ET being sent. Sending a data packet in the data channel involves a turnaround delay, packets and carrier tones propagate in τ seconds, and a tone is detected in σ seconds. Furthermore, the transmitter uses a delay $d = 2\tau + \sigma$ to transmit its data packet after detecting an RT (see Section IV). Therefore, the duration of a successful transmission period equals

$$T = \delta + \gamma + \omega + 3\sigma + 6\tau.$$

If an RTS collides with other RTSs, then all RTS's involved in the collision are transmitted in their entirety, but no receiver is able to decode any of them. As Fig. 6 illustrates, the length of a collision interval in CTMA is $Y + \gamma + \tau$, where Y is a random variable that varies from 0 to $\tau + \sigma$ and represents the time between the arrival of the RTS that starts the collision interval and the arrival of the last RTS in the interval. A collision interval occurs with probability $1 - P_S$. Because arrivals of RTS's are assumed to be Poisson distributed, $Y = 0$ implies that an RTS is successful. Accordingly, the length of an average busy period equals

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

If the time period between the start of the first and the last RTS in a collision interval equals y seconds, then there are no more arrivals of RTSs in the remaining time of the vulnerability period of the first RTS of the collision interval, i.e., $\sigma + \tau - y$ seconds. Accordingly,

$$\begin{aligned} P(Y \leq y) &= F_Y(y) = P\{\text{no arrivals in } \sigma + \tau - y\} \\ &= e^{-\lambda(\sigma+\tau-y)} \end{aligned}$$

Therefore, given that Y assumes only non-negative values, the average value of Y equals

$$\bar{Y} = \int_0^\infty (1 - F_Y(t)) dt = \int_0^{\sigma+\tau} (1 - e^{-\lambda(\sigma+\tau-t)}) dt$$

$$= \sigma + \tau - \frac{1 - e^{-\lambda(\sigma+\tau)}}{\lambda} \quad (4)$$

Substituting the value of $\bar{Y}$ in Eq. (3) we obtain the value of $\bar{B}$. The data-channel capacity in CTMA is reduced by the amount of bandwidth needed for the busy-tone channel. To account for this, the transmission time for data and packets must be expanded by the reduction in data-channel capacity. Accordingly, substituting the values of $\bar{U}$, $\bar{B}$, and $\bar{I}$ into Eq. (1) and multiplying each packet length by ζ we obtain Eq. (2). $\square$

C. DBTMA

Fig. 7. Transmission periods in DBTMA

Fig. 7 illustrates the transmission periods involved in DBTMA. As the figure shows, a transmitter waits a round-trip time after detecting the busy tone from its receiver before sending a data packet, which follows the design stated in [10]. We also consider the use of ACKs, and the receive-to-transmit turnaround times in the receiver busy-tone channel and the data channel. We assume that the data channel is assigned a percentage of the total bandwidth equal to $0 < \beta < 1$, with each busy-tone channel having an equal portion of the remaining bandwidth. We also make similar assumptions for busy tones than those made for CTMA.

The time needed for a node to detect the presence of a busy tone from a transmitter or a receiver is σ seconds. We assume that busy tones are detected perfectly, and that the probability of false busy-tone detection is 0. The receive-to-transmit latencies are the same for the data and control channels. A receiver sends an ACK after the successful reception of a data packet. Following the design of DBTMA in [10], a transmitter waits a round-trip time after detecting the busy tone from its receiver before sending a data packet (see Fig. 7).

Theorem 3: The throughput of non-persistent DBTMA over a data channel using only β (with $0 < \beta < 1$) of the total available bandwidth is given by

$$S_{DBT} = \frac{\delta}{\delta + \alpha + \beta^{-1} (2\omega + \sigma + 5\tau + \frac{1}{\lambda}) + H e^{\lambda(\tau+\sigma)}} \quad (5)$$

where $H = \gamma + \beta^{-1}[\sigma + 2\tau]$

Proof: The proof is very similar to the proof of Theorem 2. The vulnerability period of an RTS is $\tau + \sigma$ seconds, because it is the result of the propagation delay and the time needed for a node to detect a busy tone. Hence, a successful

transmission period occurs with probability $P_S = e^{-(\tau+\sigma)}$ and correspondingly a collision interval (CI) occurs with probability $1 - P_S = 1 - e^{-\lambda(\tau+\sigma)}$.

From Fig. 7, it is apparent that a successful transmission period lasts

$$T = \delta + \gamma + \alpha + 2\omega + \sigma + 6\tau$$

On the other hand, the average length of a CI equals $\bar{Y} + \gamma + \tau$, where $\bar{Y}$ is the same as in CTMA as stated in Eq. (4). As it was the case in CTMA, $Y = 0$ can be true only for the case of a successful transmission period because no more than one Poisson arrival can take place at the same instant of time.

Therefore, the average length of a busy period is

$$\bar{B} = \bar{Y} + (1 - e^{-\lambda(\tau+\sigma)})(\gamma + \tau) + T e^{-\lambda(\tau+\sigma)}$$

$$\bar{B} = e^{-\lambda(\tau+\sigma)} \left(\delta + \alpha + 2\omega + \sigma + 5\tau + \frac{1}{\lambda} \right) + \gamma + \sigma + 2\tau - \frac{1}{\lambda} \quad (6)$$

The average length of an idle period $\bar{I}$ is $1/\lambda$, because RTS arrivals are Poisson distributed with parameter λ . On the other hand, the average time period used to transmit useful data $\bar{U}$ is $\delta P_S = \delta e^{-\lambda(\tau+\sigma)}$.

The data-channel capacity in DBTMA is reduced by the amount of bandwidth needed for the two busy-tone channels. To account for this, the transmission time for data packets and signaling packets must be expanded by the reduction in channel capacity. Accordingly, substituting the values of $\bar{U}$, $\bar{B}$, and $\bar{I}$ into Eq. (1) and multiplying each packet length by β we obtain Eq. (5). $\square$

D. Non-Persistent CSMA/CA

Figure 8 presents the transmission periods for non-persistent CSMA/CA with ACKs. The length of a CTS in CSMA/CA must last at least the duration of an RTS plus a round-trip time and transmit-to-receive turn-around time required for the radios [3], which is needed in order to avoid the possibility of collisions of data packets with other transmissions. As Fig. 8 illustrates, if the receive-to-transmit turnaround times are taken into account, the vulnerability period of an RTS is $\tau + \omega$ rather than just τ as is assumed in prior work [3], because a node is unable to hear for ω seconds before transmitting.

Fig. 8. Transmission periods in non-persistent CSMA/CA

The throughput of CSMA/CA with the inclusion of ACKs and turnaround times is derived in [7] for the case of a fully-connected network and the same modeling assumptions we have made. The following theorem states the result.

Theorem 4: The throughput of non-persistent CSMA/CA is given by

$$S_{CA} = \frac{\delta}{\delta + \gamma + \alpha + 4\omega + 5\tau + \frac{1}{\lambda} + e^{\lambda(\omega+\tau)}(\gamma + 2\tau + \omega)} \quad (7)$$

Proof: The proof is presented in [7]. $\square$

VI. PERFORMANCE COMPARISON

A. Modeling Assumptions

We assume a channel data rate of 1 Mbps for simplicity. We assume MAC-level lengths of signaling packets similar to those used in IEEE 802.11 DCF. For simplicity, we assume that an RTS and an ACK is 40 bytes.

To ensure floor acquisition using CTS's in the version of CSMA/CA we use for comparison, the length of a CTS equals the length of an RTS plus a round-trip time and a transmit-to-receive turnaround time ω . We assume that ω is $10\mu s$, similar to the recommendations for IEEE 802.11 DCF.

We assume that the busy-tone detection time σ in CTMA and DBTMA is $100\mu s$, which corresponds to a probability of correct busy-tone detection close to 1 according to the model presented in [15]. We use $\zeta = .95$ and $\beta = .9$ to account for CTMA and DBTMA using most of the available bandwidth for the data channel, and DBTMA using twice the amount of bandwidth for control channels compared to CTMA.

B. Numerical Results

We compare the throughput (S) versus the offered load (G) attained by CTMA, DBTMA and CSMA/CA, based on Eqs. (2), (5), and (7). We present results for a local-area scenario and a geographically-dispersed scenario.

The local-area scenario highlights the performance of the protocols when latencies are very short and signaling overhead is small relative to the time needed to transmit data packets. Physical distances are around 500 meters, and the duration of a data packet is 1500 bytes, which is an average-length IP packet and takes 0.012s to transmit at 1 Mbps. We use a normalized propagation delay of $a = 1 \times 10^{-4}$. The geographically-dispersed scenario was considered to highlight the impact of increasing latencies and signaling overhead in the various protocols. For this scenario we assume that a data packet is only 400 bytes and that distances are around 1000 meters. We thus assume a normalized propagation delay of $a = 1 \times 10^{-3}$. Figures 9 and 10 show the results for these scenarios.

Compared to CSMA/CA, DBTMA and CTMA suffer from the latencies incurred in detecting busy tones or carrier tones and the reduced bandwidth available for the transmission of data packets. This results in lower throughput values at high loads. However, both CTMA and DBTMA provide higher throughput values than CSMA/CA at light loads. Furthermore, DBTMA and CTMA would provide higher throughput values than CSMA/CA in the presence of hidden terminals. The key advantages of CTMA over DBTMA are that it requires only one busy-tone radio and a single control channel, and that it incorporates the use of collision-free ACKs. Although CTMA

uses a single control channel, it incurs less overhead than DBTMA, which results in higher throughput values.

The performance of the version of CTMA presented in [8], which we call CTMA-2T, is slightly better than the performance of CTMA shown in Figures 9 and 10. However, the performance of CTMA-2T comes at the price of requiring more complex radios to implement carrier sensing in both the data and control channels.

Fig. 9. S vs. G with short distances and large data packets

Fig. 10. S vs. G with long distances and medium-size data packets

VII. CONCLUSIONS AND FUTURE WORK

We introduced a specific variant of Carrier-Tone Multiple Access (CTMA) in which carrier tones transmitted over a single control channel are used in coordination with transmissions of data packets and signaling packets in a data channel to eliminate the collision of data packets due to hidden-terminals. In contrast to the original variant of CTMA that we have introduced in [8], the CTMA protocol described in this paper does not require carrier sensing in the data channel.

We showed that no data packets can collide with other transmissions at the intended receivers, and that senders receive

acknowledgments correctly. We compared the throughput attained with CTMA with the throughput of CSMA/CA and DBTMA for the case in which nodes use a non-persistent transmission policy. Our results show that using multiple carrier tones over a single control channel as an integral part of collision-avoidance handshakes has clear advantages over basic collision avoidance and prior busy-tone solutions.

Our future work focuses on: (a) Analyzing the performance of CTMA in networks with hidden terminals; (b) the design of back-off strategies and persistent transmission disciplines similar to those introduced in [5], [9], which are aimed at reducing the average channel-access delays while maintaining high throughput; and (c) using carrier tones as part of receiver-initiated channel access protocols [6] and collision-resolution mechanisms [4].

ACKNOWLEDGMENTS

This work was supported in part by the Jack Baskin Chair of Computer Engineering at UC Santa Cruz.

REFERENCES

- [1] N. Abramson, "The ALOHA System—Another Alternative for Computer Communications," *Proc. Fall Joint Computer Conference '70*, 1970.
- [2] C.L. Fullmer and J.J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access (FAMA) for Packet-Radio Networks," *Proc. ACM SIGCOMM '95*, 1995.
- [3] C. L. Fullmer and J.J. Garcia-Luna-Aceves, "Solutions to Hidden Terminal Problems in Wireless Networks," *Proc. ACM SIGCOMM '97*, 1997.
- [4] R. Garces, J.J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access with Collision Resolution," *Proc. ACM MobiCom '96*, 1996.
- [5] J.J. Garcia-Luna-Aceves and A. Tzamaloukas, "The Effect of Exerting Adequate Persistence in Collision Avoidance Protocols," *Proc. IEEE Mobile Multimedia Communications '99*, Nov. 1999.
- [6] J.J. Garcia-Luna-Aceves and A. Tzamaloukas, "Receiver-Initiated Collision Avoidance in Wireless Networks," *Wireless Networks*, 2002.
- [7] J.J. Garcia-Luna-Aceves, "Carrier-Sense Multiple Access with Collision Avoidance and Detection," *Proc. ACM MSWIM '17*, Nov. 2017.
- [8] J.J. Garcia-Luna-Aceves, "CTMA: A More Efficient Channel Access Method for Networks with Hidden Terminals," *Proc. ACM PE-WASUN '17*, Nov. 2017.
- [9] J.J. Garcia-Luna-Aceves, S. Thompson, and J. Stern, "Time-Based Persistence in Channel-Access Protocols with Carrier Sensing," *Proc. IEEE IPCCC 2017*, Dec. 2017.
- [10] Z. Haas and J. Deng, "Dual Busy Tone Multiple Access (DBTMA)—A Multiple Access Control Scheme for Ad Hoc Networks," *IEEE Trans. Commun.*, June 2002.
- [11] P. Karn, "MACA—A New Channel Access Method for Packet Radio," *Proc. ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, 1990.
- [12] 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.
- [13] R. M. Metcalfe and D. R. Boggs, "ETHERNET: Distributed packet switching for local computer networks," *CACM*, vol. 19, no. 7, pp. 395 – 403, 1976.
- [14] F. Talucci, M. Gerla and L. Fratta, "MACA-BI (MACA by Invitation)—A Receiver Oriented Access Protocol for Wireless Multihop Networks," *Proc. IEEE PIMRC '97*, 1997.
- [15] 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.
- [16] Y. Wang and J.J. Garcia-Luna-Aceves, "Collision Avoidance in Multi-Hop Ad Hoc Networks," *Proc. IEEE MASCOTS '02*, 2002.
- [17] C. Wu and V. Li, "Receiver-Initiated Busy Tone Multiple Access in Packet Radio Networks," *Proc. ACM SIGCOMM '87*, 1987.