An Efficient Single-Transceiver CDMA-Based MAC Protocol for Wireless Networks

Xi Zhang, *Senior Member, IEEE*, and Hang Su, *Student Member, IEEE*

Abstract—Applying the code division multiple access (CDMA) techniques, we propose an efficient medium access control (MAC) protocol with single-transceiver for wireless ad hoc networks. Our protocol adopts the time-division method to solve the near-far power control problem inherently associated with the CDMA-based networks. In particular, employing the variable ad-hoc traffic indication messages (ATIM) window to properly determine the required transmission power for data packets, our scheme enables the interference-limited simultaneous transmissions to achieve the high utilization of the limited/precious bandwidth in wireless networks. In addition, our scheme requires only one transceiver per node, which reduces the hardware costs for large scale wireless networks. Using the Markov-chain techniques, we develop an analytical model to evaluate the aggregate throughput under our protocol. Both the analytical and simulation results show that our protocol can improve the network throughput significantly as compared with other existing schemes.

Index Terms—Code division multiple access (CDMA), Markov chain, medium access control (MAC), near-far problem, wireless networks.

I. INTRODUCTION

WIRELESS AD HOC networks have received growing attention in the past a few years. The interest in such networks stems from their ability to provide a temporary wireless networks capability in scenarios where fixed infrastructures are lacking or unavailable (e.g., disaster recoveries and battlefields, etc.). One of the fundamental challenges in wireless ad hoc networks is how to design a medium access control (MAC) protocol to increase the overall network throughput and fully utilize the limited wireless resources (e.g., bandwidth, batteries, etc.). Code division multiple access (CDMA) based MAC is one of the most effective schemes to improve the network throughput because the CDMA techniques enable multiple communicating pairs with the distinct pseudo-random-noise (PN) codes to simultaneously exchange data and occupy the same channel bandwidth. Due to this advantage, CDMA has been chosen as the channel access control technology in the third generation (3G) cellular systems. The authors of [1] showed that in such systems, CDMA can increase the capacity to up to 6 times compared with TDMA- or FDMA-based schemes. In addition, by transmitting spread spectrum signals, CDMA-based communication systems are very effective in tackling signal degradation, multipath fading, and jamming interference.

However, employing CDMA-based MAC for wireless ad hoc networks imposes many new problems. Particularly, in wireless ad hoc networks with the absence of centralized control (e.g., a basestation or an access point), both the CDMA code assignment and inherent near-far problem (see Section II-B) become more challenging to cope with. The CDMA code assignment protocols are used to assign distinct CDMA codes to different communicating pairs, which are usually classified into the following four types. (1) *The receiver-based protocols* let the transmitter use the code of the intended receiver to spread the packets, with an idle node constantly monitoring its own code. (2) *The transmitter-based protocols* require every node to be assigned a distinct transmitter-based code. The transmitter sends its data on its own code. The receiver must monitor that transmitter-based code at the same time in order to despread the received signal and recover the data. (3) *The pairwise-based protocols* assign the distinct CDMA codes to different pairs of nodes. The transmitter will look up a code assignment table to find out the code to communicate with a specific receiver. (4) *The session-based protocols* dynamically assign the distinct codes to the active sessions for the ongoing packets. Theses protocols have their own targeted applications. However, for the large scale networks having more nodes than available CDMA codes, the dynamical reuse of the CDMA codes becomes necessary, which can only be supported by the session-based code-assignment schemes. Several session-based code-assignment schemes [2], [3], [4] were proposed, where the general rules are to assign the codes to nodes such that all neighbors of a node have different codes. Under appropriate code and transmission power assignments, the session-based protocols can guarantee the collision-free transmissions. Thus, our scheme in this paper will focus on the protocols using the session-based code-assignment scheme.

To solve the near-far problem of the CDMA signal interferences caused by the nearby co-existing multiple communicating pairs, the protocol must have a mechanism to allow the senders to negotiate on the transmission codes and power with the intended receivers in the rendezvous, which is separated from where data exchange occurs, either in *time* or *frequency* domain. In the *time-division based schemes*, time axis is divided into cycles consisting of negotiation periods and data exchange periods. During the negotiation phases, every node must stop the data exchange and tune its transceiver to the common channel with same frequency and CDMA code. The transmitter reserves the CDMA code for data transmission by exchanging control packets with the intended receiver. The

other nodes can also update the code assignment information by overhearing the control packets. One unique feature with the time-division method is that each node only needs one set of transceiver. One example of the time-division CDMA-based MAC protocol is the Common-Transmitter-Based (C-T) protocol [5], where the control packets include only the transmitter's code and the receiver's address. Because the negotiation period only lasts for the fixed transmission time of one control packet, the transmitters cannot get any feedback from the receivers. Thus, the transmitter sends data in the data exchange periods regardless the receivers tune their transceivers to the transmitters' codes, resulting in the low throughput and inefficient utilization of resources.

In the *frequency-division* based schemes, the available bandwidth is typically divided into the control channel and data channel. Each node is typically equipped with at least two sets of transceivers: one always monitors the control channel, while the other(s) tuning to the data channel with different CDMA codes. The schemes proposed in [6] and [7] are examples of the frequency-division CDMA-based protocol, where one transceiver listens to the control channel and the other to the data channel. In addition to the time-division and frequency-division based schemes, authors of [8] proposed a scheme, where the control packets can be transmitted with a dedicated CDMA code. In particular, all idle nodes monitor the dedicated CDMA code for any arriving control packets. After the successful negotiation on the CDMA code, the transmitter and the receiver tune their transceivers to the negotiated code to exchange data. While the schemes in [8] need only one set of transceiver per node, it cannot effectively solve the near-far problem. This is because any node may corrupt its neighbors' ongoing data exchange since it cannot continuously monitor the control channel to avoid the severe interference with its neighbors.

To overcome the aforementioned problems, we propose the single-transceiver CDMA-based MAC protocol which uses the session-based CDMA code assignments approach and the time-division based scheme to cope with the near-far problem. Specifically, we divide time into ad-hoc traffic indication messages (ATIM) window and data window. Based on the number of active nodes, all nodes dynamically estimate the length of the ATIM window, which is used to negotiate the CDMA codes using the session-based CDMA code assignments and the transmission power for the data exchange in the data window. Using the channel access control mechanism in our protocol, the communicating pairs select the appropriate power levels such that the new transmission does not interfere the nearby existing communicating pairs. Both the simulation and analytical results show that our proposed scheme with *variable ATIM* windows can significantly improve the throughput, as compared to the *fixed ATIM* window counterparts.

The rest of this paper is organized as follows. Section II briefly describes IEEE 802.11 power saving mode (PSM) and the near-far problem in CDMA systems. Section III develops our CDMA-based single-transceiver MAC protocol. Section IV derives the Markov-chain model to analyze proposed protocol. Section V evaluates our protocol through numerical and simulation solutions. The paper concludes with Section VI.

II. POWER MANAGEMENT FOR MAC PROTOCOLS

A. IEEE 802.11 Power Saving Mechanism (PSM)

We briefly describe IEEE 802.11 PSM to explain how the ATIM window works. A node in the network can save energy by going into the sleep mode, in which the node consumes much less energy than in the idle mode. It is desirable for a node to enter the sleep mode only when there is no need for communications. The IEEE 802.11 PSM conducts the power management by using the ATIM window. Time axis is divided into beacon intervals, and each node in the network is synchronized by periodic beacon transmissions. Thus, all nodes in the network can enter and leave each beacon interval at about the same time.

At the start of each beacon interval, there exists an interval called the ATIM window, where each node is required to stay awake. If a node, e.g., A, has buffered packets targeted to another node, say B, it sends an ATIM frame to node B during the ATIM window. When node B receives this packet, it replies by sending an ACK to node A. Both nodes A and B then stay awake for that entire beacon interval. If any node has not sent or received any ATIM frames during the entire ATIM windows, it enters the sleep mode until the next beacon arrives. One of the most important issues in IEEE 802.11 PSM is how to design the size of ATIM window. A large size of ATIM window leads to a long delay and low throughput, while a small size of ATIM window may result that the nodes do not have sufficient time to finish the exchange of ATIM/ACK frames. Dynamically adjusting the ATIM window size according to the number of active nodes is an effective approach to solve ATIM window control problem [11] [12].

B. Near-Far Problem in CDMA-Based MAC

CDMA is based on the spread spectrum techniques, where each user occupies the entire available bandwidth. At the transmitter, a digital signal of R bandwidth is spread using a PN code of W bandwidth. The ratio of R/W is called processing gain. The PN code is a binary sequence that statistically satisfies the requirement of a random sequence. The intended receiver can use the identical PN code to despread the received signal, which is conceived as background noise by the other unintended receivers with different PN codes. The near-far problem stems from the fact that unlike FDMA and TDMA channels which can be completely orthogonal, CDMA codes suffer from nonzero cross-correlation between codes [14]. When a CDMA receiver despreads a received signal, it calculates the cross-correlation between the signal and a locally generated PN sequence. If this PN sequence is identical to the one used to spread the signal at the transmitter (i.e., the message is intended to this receiver), the cross-correlation calculations restore the original information data. Otherwise, the receiver considers that the received signal is noise and neglects the signal.

In this paper, we consider the asynchronous direct sequence with binary phase shift keying (DS/BPSK) system with rectangular chip pulse. The noise at the receiver detector is due to the interference from the other nodes and a constant background noise with power spectral density $N_0/2$. Denote the signal to interference and noise ratio (SINR) at the detector by

E_b/N_0^{eff}, where E_b is the signal power per bit and $N_0^{eff}/2$ is an equivalent white noise power spectral density for the same signal to noise ratio (SNR) at the detector. If the desired signal received at i-th node ($i = 1, 2, \cdots$) has power $P_0^{(i)}$ and the J interfering nodes have powers P_1, P_2, $\cdots$, P_J, then the average SINR, denoted by γ, at the detector of the i-th node is given by [9]:

$$\gamma \triangleq \frac{E_b}{N_0^{\text{eff}}} = \left(\frac{2Y}{3LP_0^{(i)}} + \frac{1}{\gamma_0} \right)^{-1}, \qquad (1)$$

where $Y = \sum_{j=1}^{J} P_j$, $L = R/W$ is the processing gain, and γ_0 is the SNR (i.e., E_b/N_0) at the detector without the interference. When the desired received signal power increases, γ increases, and then the bit error rate (BER) probability decreases.

To demonstrate the relationship between the BER and average SINR, here we take a DS/BPSK system without the forward error correction (FEC) for example. The system has the following specifications: a rate of 1/2, constraint length of 7, convolutional code with Viterbi decoding, and eight-level soft decisions. A moderate value for the spreading code rate is 1000 chips per data symbol. For a BER of 10^{-6}, the required message bit SINR (E_b/N_0^{eff}) is 10.5 dB and 5.0 dB for uncoded and coded transmission, respectively [10]. The corresponding processing gains are 1000 (30 dB) and 2000 (33 dB) for the uncoded and coded system, respectively. If we neglect the thermal noise, then according to Eq. (1), the total interference power must satisfy $Y/P_0^{(i)} \leq 133.69$ for uncoded case and $Y/P_0^{(i)} \leq 948.68$ for coded case, respectively.

III. THE PROPOSED CDMA-BASED MAC PROTOCOL

Our proposed protocols implement the function of the CDMA codes negotiation in the time-division method, as mentioned in Section I. Similar to the IEEE 802.11 PSM, in our proposed protocols the nodes also use the ATIM window to realize the function of channel negotiation. However, the ATIM window in our proposed protocols is used not only for saving power, but also, more importantly, for avoiding collisions by dynamically reserving CDMA codes and controlling transmission power. The ATIM window can be fixed or dynamically adjustable based on the number of active of nodes. We term the CDMA-based MAC protocol with variable ATIM-window as CMVAW, and the CDMA-based MAC protocol with fixed ATIM-window as CMFAW.

A. Protocol Description

Fig. 1 shows the operating process of control information negotiation and data exchange in our proposed protocol. The time axis is periodically divided into ATIM windows and data windows. The ATIM window is used for CDMA code and transmission power negotiation, while the data packets are exchanged within the data window. The ATIM window begins with the ATIM beacon.[1] The ATIM window consists
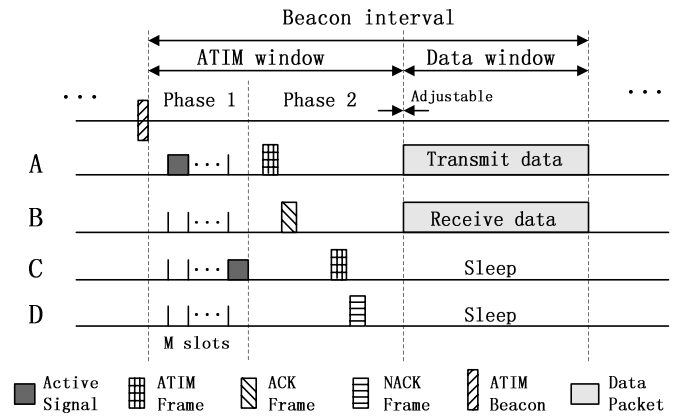
Fig. 1. Illustration of the proposed CMVAW protocol.

of phase 1 and phase 2. Specifically, the phase 1 of ATIM window is fixed and divided into M slots. The CMVAW protocol uses phase 1 to estimate the number of active buffered-nodes that attempt to start transmission in the following data window. Each active node randomly chooses a slot to transmit the active signal. The nodes keep on sensing in the phase 1, then count the number of busy slots to estimate the number of active nodes based on the estimation algorithm (see Section III-B). According to the estimation of the number of active nodes, the nodes decide the length of phase 2. If the nodes sense one of the slot in phase 1 busy, they will be awake in the coming phase 2. Otherwise, the phase 2 will be skipped and the nodes enter the data window directly.

Phase 2 is designed for the negotiation of CDMA codes and transmission power, which is achieved by the exchange of the ATIM/ACK frames. All the nodes in phase 2 tune their transceivers to the common code so that they can hear each other. The active nodes contend for sending the ATIM frames. Upon receiving the ATIM frame, the destination node sends ACK/NACK frame to inform the sender whether it can correctly receive data packets without interfering the current ongoing data exchanges. At the end of phase 2, the nodes, which are allowed to transmit or receive, use the negotiated CDMA codes to exchange data within the data window. At the same time, the other nodes enter the sleep mode until the next ATIM beacon arrives.

On the other hand, in the CMFAW protocol the time axis is also periodically divided into beacon intervals which consists of ATIM windows and data windows. The difference between CMFAW and CMVAW protocols is that the length of the ATIM window under CMFAW is fixed, regardless of the number of the active nodes. Thus, the length of the beacon interval is fixed under CMFAW. The operating process of the codes and power negotiation as well as the data exchange under CMFAW protocol is similar to those in the CMVAW protocol, except that the CMFAW protocol does not need to estimate the number of active nodes.

B. The Estimation of the Number of Active Nodes

We divide the phase 1 of ATIM window into M slots. The length of each slot is σ, which is long enough to determine if the channel is active or not. If a node has buffered packets in

current beacon interval, the source node randomly chooses one of the slots to send an active signal which the other nodes can sense, as shown in Fig. 1. The nodes can estimate the number of nodes which attempt to contend for reserving CDMA codes during the phase 2 of ATIM window based on the observation of the slots. The number of busy slots is fewer than or equal to the number of active nodes because this scheme cannot guarantee that the active nodes choose the distinct slots to send the active signals. Conditioning on the random number, denoted by N_a, of the active nodes, the conditional probability $\Pr\{M_o = m | N_a = n\}$ that the random number, denoted by M_o $(0 \leq M_o \leq M)$, of observed busy slots is equal to m can be expressed as follows:

$$\Pr\{M_o = m | N_a = n\} = (\mathbb{Q}^n)|_{(0,m)}, \qquad (2)$$

where

$$\mathbb{Q} = \begin{bmatrix} 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \frac{1}{M} & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{i}{M} & \frac{i}{M-i} & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & \frac{i+1}{M} & \frac{M-i-1}{M} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & \frac{1}{M} & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}_{(M+1) \times (M+1)}$$

is an $(M+1) \times (M+1)$ upper bidiagonal matrix, and $\mathbb{X}|_{(i,j)}$ represents the element in the position of row i, column j of matrix $\mathbb{X}$. Appendix A details the derivation for $\Pr\{M_o = m | N_a = n\}$. Let $m^{(t)}$ be the observed number of busy slots and $\widehat{p}_{N_a}^{(t)}$ be the estimated probability mass function (pmf) of the number of active nodes at the end of phase 1 of ATIM window in the t-th beacon interval. According to the Bayes' Theorem, we get:

$$\Pr\{N_a = n | M_o = m^{(t)}\}$$
$$= \frac{\Pr\{N_a = n, M_o = m^{(t)}\}}{\sum_{N_a = m^{(t)}}^{N} \Pr\{M_o = m^{(t)} | N_a = n\}\widehat{p}_{N_a}^{(t)}}$$
$$= \frac{\Pr\{M_o = m^{(t)} | N_a = n\}\widehat{p}_{N_a}^{(t)}}{\sum_{N_a = m^{(t)}}^{N} \Pr\{M_o = m^{(t)} | N_a = n\}\widehat{p}_{N_a}^{(t)}}, \qquad (3)$$

where $\Pr\{N_a = n | M_o = m^{(t)}\}$ is the likelihood function. The estimated number of active nodes, denoted by $\widehat{N}_a$, can be obtained through the maximum a posteriori (MAP) estimation, i.e.,

$$\widehat{N}_a^{(t)} = \arg\max_{m^{(t)} \leq N_a \leq N} \Pr\{N_a = n | M_o = m^{(t)}\}$$
$$= \arg\max_{m^{(t)} \leq N_a \leq N} \Pr\{M_o = m^{(t)} | N_a = n\}\widehat{p}_{N_a}^{(t)}. \qquad (4)$$

We can use a simple auto regressive (AR) model to update the $\widehat{p}_{N_a}^{(t)}$ as

$$\widehat{p}_{N_a}^{(t+1)} = \begin{cases} (1-\delta)\widehat{p}_{N_a}^{(t)} + \delta, & \text{if } n = \widehat{N}_a^{(t)}, \\ (1-\delta)\widehat{p}_{N_a}^{(t)}, & \text{otherwise.} \end{cases} \qquad (5)$$

where $\delta > 0$. Clearly, the larger the value of M, the more accuracy the estimation. However, increasing M can cause

higher overhead because the interval of phase 1 of ATIM window will last longer. There is a tradeoff between the accuracy and overhead. Fortunately, the time slot, denoted by σ, can be set to a very small value (e.g., the time slot in IEEE 802.11 DSSS is $20\mu s$), which is enough to determine if the slot is active or not. Let N be the total number of nodes in the system. Based on the extensive simulation experiments (which we omit for lack of space), we observe that $M = 2N$ is enough to get a highly-accurate estimated number of the active nodes. Thus, the interval length for phase 1 of ATIM window, denoted by T_1, can be written as:

$$T_1 = M\sigma = 2N\sigma. \qquad (6)$$

C. Variable Length of ATIM Windows

Based on the estimated number N_a of active nodes that transmit ATIM frames in phase 2 of ATIM window, the length of phase 2 of ATIM window in CMVAW protocol can be adjusted dynamically. In CMVAW, the active nodes adopt p-persistent carrier sense multiple access (CSMA) algorithm to send ATIM/ACK packets to reserve CDMA codes and to negotiate the transmission power. Let σ, T_{succ}, and T_{coll} be the time slot, successful transmission time, and failure transmission time, respectively. We can calculate T_{succ} and T_{coll} by using the following equations:

$$\begin{cases} T_{\text{succ}} = ATIM + SIFS + ACK + DIFS \\ T_{\text{coll}} = ATIM + DIFS. \end{cases} \qquad (7)$$

In the p-persistent CSMA, the probability, denoted by P_{idle}, that the channel is idle is $(1-p)^{N_a}$. The probability, denoted by P_{succ}, that a node successfully transmits an ATIM frame can be determined by $N_a p (1-p)^{N_a - 1}$. The probability, denoted by P_{coll}, that the collision occurs is equal to $(1 - P_{\text{idle}} - P_{\text{succ}})$. Therefore, the average time used for a successful transmission can be expressed as

$$T(p, N_a) = \frac{\sigma P_{\text{idle}} + T_{\text{succ}} P_{\text{succ}} + T_{\text{coll}} P_{\text{coll}}}{P_{\text{succ}}}. \qquad (8)$$

Let p^* be the optimal transmission probability which minimizes $T(p, \widehat{N}_a)$. The value of p^* can be calculated off line and stored in the memory of each node. After estimating the number $\widehat{N}_a$ of active nodes following the phase 1 of ATIM window, the nodes load the corresponding $T(p^*, \widehat{N}_a)$. Then, the nodes can adjust the optimal interval length for phase 2 of ATIM window by using

$$T_2(\widehat{N}_a) = \kappa \sum_{n=0}^{\min\{\widehat{N}_a, u_{max}\}-1} T(p^*, \widehat{N}_a - n), \qquad (9)$$

where u_{max} is the predetermined maximum number of nodes which can successfully transmit in phase 2 of ATIM window, and κ is an adjusting parameter. Clearly, $T_2(\widehat{N}_a)$ is a monotonously increasing function of $\widehat{N}_a$.

D. The Channel Access Control Mechanism

During phase 2 of ATIM window, each node in both CM-VAW and CMFAW protocols tunes its receiver to the common code and monitor the channel. The negotiation is achieved by the exchange of ATIM and ACK frames. First, these packets

allow nodes to estimate the channel gains between communicating pairs. Second, a receiver j can use the ACK frames to inform its neighbors of the additional interference which each neighbor can add to receiver j without interfering its current communication. Finally, each node keeps on overhearing all control packets during the ATIM window to update the number of ongoing data transmissions. All the control frames such as ATIM, ACK, and NACK are transmitted at the maximum transmission power, denoted by P_{max}.

The pseudo code for our proposed protocols to select the CDMA codes and transmission power is listed in **Algorithm 1** as follows:

Algorithm 1 Code for the channel access control mechanism in phase 2 of ATIM window.

1: **Sender i:**
2:   Compute $P_{mstp}^{(i)}$ based on Eq. (10)
3:   Contend in the phase 2 of ATIM window to transmit ATIM frame
4:   Listen on the channel until timeout or receiving replies
5:   **if** receive an ACK frame
6:     exchange data pkt with the receiver-selected code in data window
7:   **elseif** receive a NACK frame
8:     go to sleep mode in data window
9: **Receiver j:**
10:   Compute $\Delta^{(j)}$ according to Eq. (11)
11:   **if** $\Delta^{(j)} > 0$
12:     find an available CDMA code
13:     Send an ACK frame including $\Delta^{(j)}$ and CDMA code
14:     exchange data pkt with the CDMA code in data window
15:   **else** send a negative ACK frame
16: **Neighbor k:**
17:   **if** receive a ATIM frame
18:     compute $\widetilde{\gamma}^{(k)}$ by using Eq. (12)
19:     **if** $\widetilde{\gamma}^{(k)} < \gamma_{th}$
20:       send out a negative ACK
21:   **if** receive a ACK frame from node j
22:     **if** the CDMA code in the ACK frame is identical with mine
23:       send a negative ACK
24:     **else** update the available CDMA codes list.

If sender i has buffered packets targeted to node j, it will notify j by sending an ATIM frame. Let $\mathcal{V}(i)$ be the set of i's neighbors which will receive data in the coming data window, and node k be one of such neighbors (i.e., $k \in \mathcal{V}(i)$). By overhearing the control frames, node i knows the interference margin ($\Delta^{(k)}$) of node k, and the channel gain[2] ($G_{i,k}$) between i and k. Here, the interference margin represents the maximum interference that a receiving node can tolerate. Clearly, in order not to interfere any receiving neighbors, the maximum safe transmission power, denoted by $P_{mstp}^{(i)}$, of node i can be derived as

$$P_{mstp}^{(i)} = \min \left\{ P_{max}, \min_{k \in \mathcal{V}(i)} \left[\frac{\Delta^{(k)}}{|\mathcal{V}(i)|G_{i,k}} \right] \right\}, \quad (10)$$

where $|\mathcal{V}(i)|$ implies the number elements of $\mathcal{V}(i)$. The factor of $1/|\mathcal{V}(i)|$ in Eq. (10) is to prevent one single node from consuming the entire $\Delta^{(k)}$. The value of $P_{mstp}^{(i)}$ is stored in the two-byte field of the ATIM frame, as shown in Fig. 2.

---
[2]We assume that the channel is symmetric, i.e, $G_{i,k} = G_{k,i}$. The sender i can calculate $G_{k,i}$ when it overhears the control packet from node k.

ATIM Frame Format

Frame Control	SA	DA	Data Duration	MSTP	FCS
Bytes: 2	6	6	2	2	4

ACK Frame Format

Frame Control	DA	Data Duration	Interference margin	CDMA Code	FCS
Bytes: 2	6	2	2	4	4

Fig. 2. Format of the ATIM and ACK frames. SA stands for source address and DA stands for destination address.

Upon receiving the ATIM frame, the intended receiver j uses the predetermined P_{max} value and the power of the received signal $P_r^{i,j}$ to estimate the channel gain $G_{i,j} = P_r^{i,j}/P_{max}$ between sender i and receiver j at that time. Then, it checks if P_{mstp} satisfies the minimum required SINR, denoted by γ_{th}. Manipulating Eq. (1) algebraically, we can obtain the interference margin, denoted by $\Delta^{(j)}$, of node j as

$$\Delta^{(j)} = \frac{3L}{2} \left(\frac{P_{mstp}G_{i,j}}{\gamma_{th}} - N_0 \right) - Y^{(j)}, \quad (11)$$

where $Y^{(j)}$ is the total interference by other senders. If $\Delta^{(j)}$ is greater than 0, then the receiver replies by sending an ACK frame (see Fig. 2) including $\Delta^{(j)}$ and selected CDMA code to node i. Otherwise, the node j replies by sending an NACK frame to inform node i that the maximum safe transmission power of i is not enough for correct data decoding.

While the destination j receives the ATIM frame, the neighbor k also can overhear it. The purpose of overhearing is to ensure that the neighbor's new transmission does not interfere its data communication in the coming data window. In particular, it calculates the SINR of the scenario where the sender i transmits packets in the coming data window by using

$$\widetilde{\gamma}^{(k)} = \left[\frac{2 \left(Y^{(k)} + P_{mstp}^{(k)} G_{i,k} \right)}{3L P_r^{(k)}} + \frac{1}{\gamma_0} \right]^{-1}, \quad (12)$$

where $P_r^{(k)}$ is the intended receiving signal power in the data window. If $\widetilde{\gamma}^{(k)}$ is less than the minimum required SINR (γ_{th}), it will transmit an NACK frame to sender i, which will block the data exchange between node i and node j to ensure the successful data transmission of itself.

IV. PROTOCOL MODELING

To make the model tractable, we do not consider the power control issue in the protocol modeling. In other words, the interference margin for every node is large enough to tolerate the interference caused by their neighbors' transmissions. This is reasonable if we adopt a large processing gain. Recalling the example of DS/BPSK system in Section II-B, if all the transmissions are received at a particular receiver with equal power, then the number of possible communicating pairs is about 300 for the coded case.

For both CMVAW and CMFAW protocols, there are five sets of nodes in the t-th beacon interval, denoted by $\mathcal{L}_t$, $\mathcal{C}_t$, $\mathcal{R}_t$, $\mathcal{I}_t$, and $\mathcal{D}_t$, respectively, as shown in Fig. 3. Particularly,
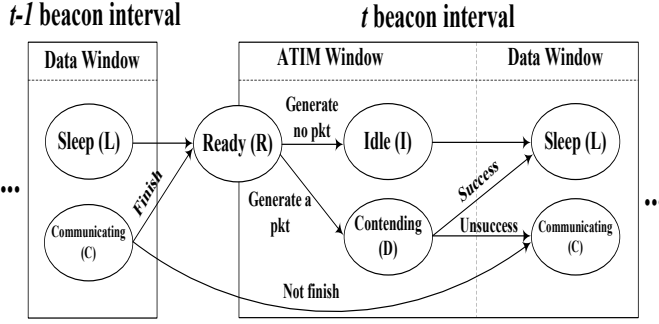
Fig. 3. The diagram of node states transition from $(t-1)$ data window to t-th data window. I represents the set of idle nodes and C denotes the set of communicating nodes.

TABLE I
THE PARAMETERS USED IN OUR PROTOCOLS AND PROTOCOLS MODELING.

N	The number of nodes in the system.
N'	The total number of nodes in $\mathcal{R}_t$, i.e., $|\mathcal{R}_t|$.
u	The number of the $\mathcal{D}_t$ nodes that successfully get ACK frames from the destination nodes, i.e., $|\mathcal{D}_t \cap \mathcal{C}_t|$.
v	The number of communicating pairs of nodes that finish data exchange at the end of $t-1$ beacon interval and become ready at the beginning of the t-th beacon interval, i.e., $|\mathcal{C}_{t-1} \cap \mathcal{R}_t|/2$.
k	The number of communicating pairs that communicate in $t-1$ beacon interval, i.e., $|\mathcal{C}_{t-1}|/2$.
m	The number of communicating pairs that communicate in t-th beacon interval, i.e., $|\mathcal{C}_t|/2$.
w	The number of nodes that are active in the ATIM window in the t-th beacon interval, i.e., $|\mathcal{D}_t|$.
λ	Probability that a node attempts to transmit data.
μ	Probability that a node finishes the transmission.

$\mathcal{L}_t$ is the set of nodes that are in sleep mode during the data window. $\mathcal{C}_t$ is the set of nodes that exchange data during the data window. $\mathcal{R}_t$ is the set of nodes that are ready to transmit or receive data at the beginning of the ATIM window. $\mathcal{I}_t$ is the set of nodes that are idle during the ATIM window. $\mathcal{D}_t$ is the set of nodes that have buffered data packets and contend for transmitting ATIM frames during the ATIM window. As shown in Fig. 3, The set of $\mathcal{R}_t$ includes two parts: i) all the nodes of $\mathcal{L}_{t-1}$, and ii) a fraction of $\mathcal{C}_{t-1}$ nodes that finish the data transmission. We assume that each $\mathcal{R}_t$ node generates a packet at the beginning of ATIM window with probability λ. The destination address of the generated packet is arbitrarily chosen among all the nodes. The nodes that generate packets (i.e., $\mathcal{D}_t$) contend for the right of transmission in data window. If a $\mathcal{D}_t$ node successfully transmits an ATIM frame and receives an ACK from the destination within the ATIM window, then it can exchange data packets in the coming data window.

We assume that the data packet length, denoted by L, in terms of data window size follows the geometrical distribution, i.e., the probability that a data packet has length ℓ is

$$\Pr\{L = \ell\} = \mu(1-\mu)^{\ell-1}. \tag{13}$$

It will take ℓ data-windows to complete the transmission of a data packet with a length of ℓ. If we denote the data transmission rate by R_d and the data window size by T_3, then the average packet length in the unit of byte is $\overline{L} = T_3 R_d/\mu$. Because the geometrical distribution is memoryless, each communicating pair in $\mathcal{C}_{t-1}$ finishes the data transmission with probability μ at the beginning of the ATIM window in the t-th beacon interval.

Clearly, at any given beacon interval, the system state at every interval beacon can be characterized by the number of communicating pairs during the corresponding data window (i.e., $|\mathcal{C}|/2$). We analyze the proposed protocol by using a discrete-time Markov chain to evaluate the performance of our proposed protocols. A transition in the Markov chain from one state to another occurs if i) at least one communicating pair finishes the data transmission, or (and) ii) at least one communicating pair begins the data transmission.

For convenience, we summarize the key parameters for protocol modeling in Table I. The number N' of nodes that is ready to transmit or receive at the beginning of the ATIM

window can be written as:

$$N' = |\mathcal{R}_t| = N - 2(k - v). \tag{14}$$

Because each communicating pair finishes the data transmission with probability μ, given the number k of communicating pairs in the $(t-1)$-th beacon interval, the number v of communicating pairs that become ready at the beginning of t-th beacon interval follows the binomial distribution, i.e.,

$$p(v|k) = \binom{k}{v}\mu^v(1-\mu)^{k-v}. \tag{15}$$

Also, the number w of nodes that is active in the t-th ATIM window follows the binomial distribution conditioning on v and k, i.e.,

$$p(w|k,v) = \binom{N'}{w}\lambda^w(1-\lambda)^{N'-w}. \tag{16}$$

Notice that only the nodes of $\mathcal{I}$ can send the ACK frame when receiving the ATIM frame with their own address. The maximum number, denoted by W, of nodes that can successfully receive ACK frames in phase 2 of ATIM window can be expressed as

$$W = \min\{w, u_{max}, N' - w\}, \tag{17}$$

where u_{max} is the predefined value limiting the length of phase 2 of ATIM window. Because the destination address in the data packet is arbitrary, whether the node, which successfully transmits an ATIM frame, can receive an ACK frame depends on the number of nodes in $\mathcal{I}_t$. Provided that i active nodes have already successfully received ACK frames, the probability, denoted by α_i, that the $(i+1)$-th node also receives an ACK frame from the destination after it successfully sends an ATIM frame can be determined by

$$\alpha_i = \frac{|\mathcal{I}_t| - i}{N-1} = \frac{N' - w - i}{N-1}, \tag{18}$$

where $0 \le i \le W$. The number of nodes that successfully get the ACK frames from destination nodes can be modeled as a Markov chain, as shown in Fig. 4. Thus, we have the one step

transition probability matrix

$$\mathbb{A} = \begin{bmatrix} 1-\alpha_0 & \alpha_0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1-\alpha_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \cdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1-\alpha_i & \alpha_i & \cdots & 0 \\ \vdots & \vdots & \ddots & \cdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 1-\alpha_W \end{bmatrix}_{\substack{(W+1)\times \\ (W+1)}}$$

where $\mathbb{A}$ is a $(W+1) \times (W+1)$ upper bidiagonal matrix. Then, the conditional pmf of u conditioning on k, v, and w can be expressed by:

$$p(u|k,v,w) = (\mathbb{A}^W)|_{(0,u)}, \quad (19)$$

where $\mathbb{X}|_{(i,j)}$ denotes the element located at i-th row and j-th column in matrix $\mathbb{X}$. Since $u = m - k + v$, we have

$$p(m|k,v,w) = (\mathbb{A}^W)|_{(0,m-k+v)}. \quad (20)$$

By removing the conditions on v and w, the transition probability, denoted by ρ_{km}, from state of $|\mathcal{C}_{t-1}|/2 = k$ to state of $|\mathcal{C}_t|/2 = m$ can be derived as

$$\begin{aligned} \rho_{km} &= p(m|k) = \sum_{v=0}^{k} p(m|k,v)p(v|k) \\ &= \sum_{v=0}^{k}\sum_{w=0}^{N'} p(m|k,v,w)p(w|k,v)p(v|k). \quad (21) \end{aligned}$$

Note that in CMVAW protocol the length T_2 of phase 2 of ATIM window can be obtained by substituting $\widehat{N}_a = w$ into Eq. (9). We need to derive the distribution of w conditioning on k and m to calculate the average throughput of CMVAW protocol. After some observations as shown in Appendix B, the conditional pmf $p(w|k,m)$ of w conditioning on k and m can be determined by

$$p(w|k,m) = \frac{\sum_v p(m|k,v,w)p(w|k,v)p(v|k)}{\rho_{km}}. \quad (22)$$

To compute the average throughput, we need to know the steady-state probability that corresponds to each state of the Markov chain. After obtaining the transition probabilities by using Eq. (21), we can derive the steady state distribution by using simple matrix algebra based on transition probability matrices. If we denote the probability for steady state of $|\mathcal{C}_t|/2 = k$ by π_k, then the average aggregate throughput S can be written as:

$$\begin{aligned} S_{\text{CMVAW}} &= \sum_{k=0}^{\lfloor N/2 \rfloor}\sum_{m=0}^{\lfloor N/2 \rfloor}\sum_{w=0}^{N} \frac{p(k,m,w)mT_3R_d}{T_1(N)+T_2(w)+T_3} \\ &= \sum_{k=0}^{\lfloor N/2 \rfloor}\sum_{m=0}^{\lfloor N/2 \rfloor}\sum_{w=0}^{N} \frac{\pi_k\rho_{km}p(w|k,m)mT_3R_d}{T_1(N)+T_2(w)+T_3}, \quad (23) \end{aligned}$$

where $\lfloor x \rfloor$ represents the maximum integer that is not larger than x and R_d is the data transmission rate during the data window. Because there is only a fixed-size phase 2 of ATIM window, and no phase 1 of ATIM window for each beacon interval in CMFAW protocol, the average aggregate
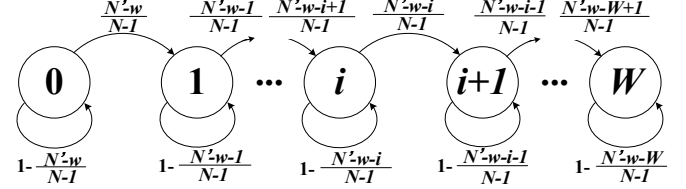

Fig. 4. The state transition diagram for the nodes that get the ACK from destination nodes in phase 2 of ATIM window. The number in the circle represents the number of such nodes.
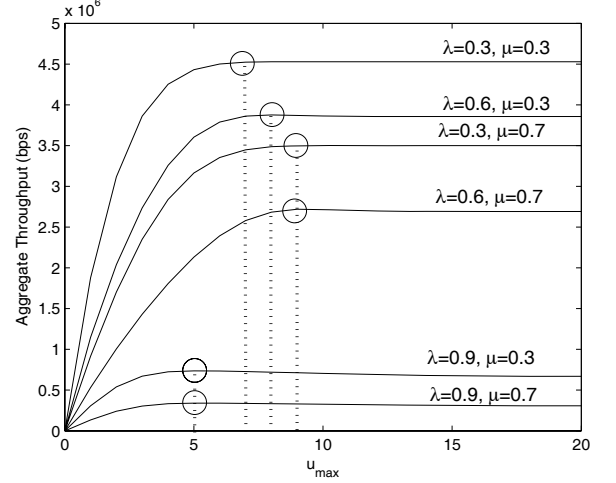

Fig. 5. Aggregate throughput against u_{max} with N=20.

throughput for the fixed-window protocol can be calculated by:

$$S_{\text{CMFAW}} = \sum_{k=0}^{\lfloor N/2 \rfloor}\sum_{m=0}^{\lfloor N/2 \rfloor} \frac{\pi_k\rho_{km}mT_3R_d}{T_2(u_{max})+T_3}. \quad (24)$$

V. PROTOCOL EVALUATIONS

Based on the discussion in Section IV, we know that the throughputs of our proposed protocols depend on various parameters, such as λ, μ, N, u_{max}, etc. In this section, we study the impact of these parameters on the CMVAW and CMFAW protocols. In the following numerical and simulations evaluations, we set the bandwidth $B = 40MHz$, the processing gain $L = 40$, time slot $\sigma = 20\,\mu s$, and $T_3 = 8\,ms$. The data rate $R_d = B/L = 1MHz$.

First, we focus on the CMVAW protocols. Fig. 5 shows that the aggregate throughputs vary with the value of u_{max} under different combinations of λ and μ when the number of nodes is 20. Given λ and μ, each circle in Fig. 5 represents the optimal u_{max}, denoted by u_{max}^*, which is the minimum u_{max} achieving the highest aggregate throughput. The value of u_{max}^* varies with different λ and μ. The aggregate throughput decreases with the increase of u_{max} when $u_{max} > u_{max}^*$. All u_{max}^*'s for different combinations of μ and λ are less than 10 because of the following reasons. First, although the larger u_{max} implies allowing more nodes to transmit successfully in the phase 2 of ATIM window, these nodes receive only NACK instead of ACK because their destination nodes are not ready to receive (i.e., the receivers are also active in the phase 2
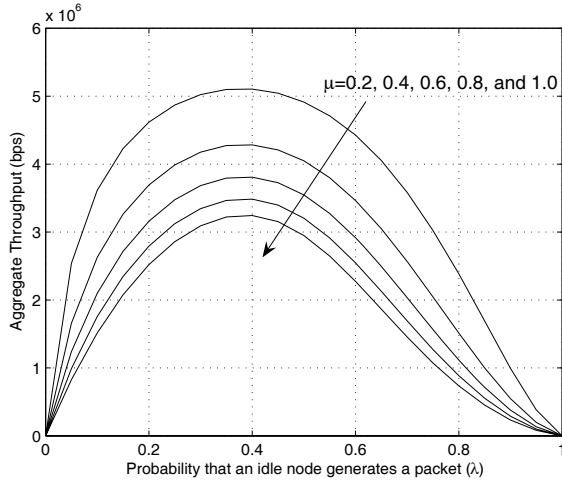
Fig. 6.   Aggregate throughput against λ with different μ's when $N = 20$.

Fig. 7.   Aggregate throughput against T_3 with different λ's. The average packet length is fixed at 10^4 bytes. The number N of nodes is 20.

of ATIM window and attempt to transmit). Second, the larger u_{max} implies the longer phase 2 of ATIM window, resulting in low bandwidth efficiency. We also notice that the aggregate throughput decreases slightly with the increase of u_{max} when $u_{max} > u_{max}^*$. Thus, we set u_{max} to be a constant of 10 for the following evaluations when $N = 20$.

Fig. 6 plots that the aggregate throughput against λ with different μ's when $N = 20$. Given μ, we can find an optimal λ^* that achieves the maximum throughput. The aggregate throughput increases with λ when $\lambda < \lambda^*$, but decreases when $\lambda > \lambda^*$. That is because i) when $\lambda < \lambda^*$, the probability that the nodes attempt to transmit data is small; ii) when $\lambda > \lambda^*$, the probability that the destination node is ready to receive data becomes small and the size of phase 2 of ATIM window increases with the number of active nodes, resulting in the higher overhead for data transmission. Also, when the value of λ is fixed, the smaller the value of μ, the higher the aggregate throughput. The reason behind this is that the larger μ implies that the nodes continue exchanging data packets with larger probability at the next beacon interval, leading to the higher π_k when k is large.

Fig. 7 shows the impact of the data window size (i.e., T_3) on the aggregate throughput with different λ's when the average packet length is fixed at 10^4 bytes. This is expected by observing Eq. (23). Because the actual data exchange takes place only at data window, the ATIM window can be considered as the overhead. A smaller T_3 implies that the overhead incurred by the relatively large $(T_1 + T_2)$ becomes larger, leading to the lower aggregate throughput. On the other hand, the beacon interval becomes larger when T_3 increases, implying that a node attempting to transmit a data packet has to wait for a longer time.

The number of nodes in the system also has impact on the aggregate throughput. Fig. 8 shows the aggregate throughput against λ with $\mu = 1$ when the value of N varies. Under the same λ, the larger the N, the higher the aggregate throughput. In addition, we observe that the aggregate throughput with larger N is more sensitive to λ as compared to that with a smaller N, especially when $\lambda < 0.2$ or $\lambda > 0.6$.

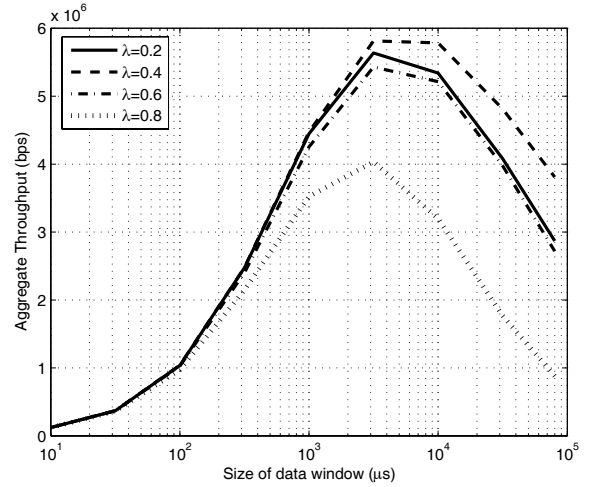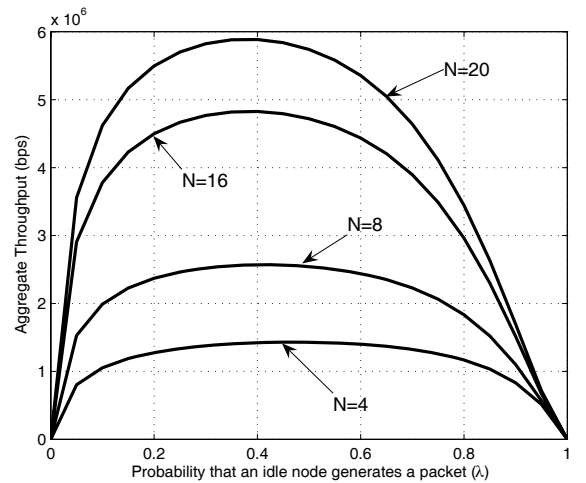Then, we compare the CMVAW and CMFAW protocols.

Fig. 8.   Aggregate throughput against λ with $\mu = 0.1$.

Fig. 9 shows the aggregate throughput versus the average data packet length when the CMVAW and CMFAW protocols are used. When the average data packet length is smaller than 800 Kbit, the aggregate throughput achieved by the CMVAW is largest among the schemes. However, when the average data packet length is larger than 800 Kbit, the throughput of CMFAW with $u_{max} = 1$ is slightly larger than that of CMVAW. This is expected because the large average data packet implies that the data transmission lasts for longer time. There are fewer nodes that need to start the new data transmission and negotiate in the ATIM window. Thus, the longer ATIM window implies the waste of resource, degrading the throughput.

Fig. 10 shows the simulation results and analytical results of the aggregate throughput when CMVAW, CMFAW with $u_{max} = 1$, and C-T protocols [5] are used. In general, the analytical results agree well with the simulation results. The analytical throughput is a little higher than the throughput obtained through simulations because the analytical model does not take the power control into considerations. As the
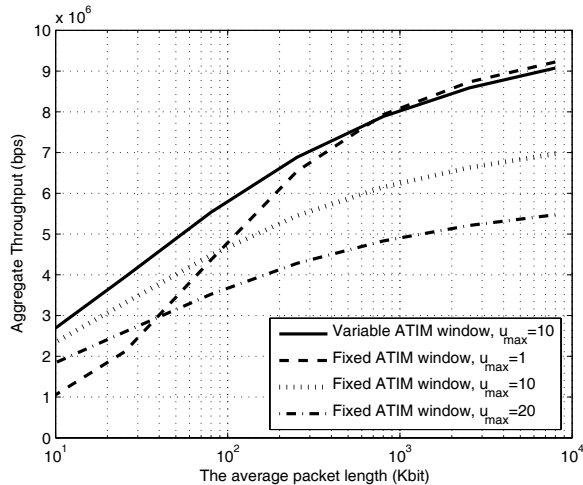
Fig. 9. Aggregate throughput against the average data packet length. The number (N) of nodes is 20.
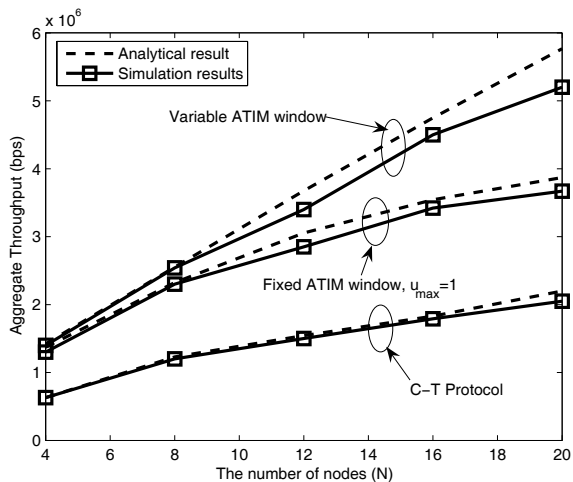

Fig. 10. Comparison of different protocols with $\lambda = 0.4$ and $\mu = 0.1$.

number of nodes increases, the advantage of CMVAW in terms of aggregate throughput becomes stronger as compared to the CMFAW protocol and C-T protocol. The aggregate throughput of C-T protocol is worst, because a node may continue data transmission even if the intended receiver is not ready, leading to the low utilization of the bandwidth.

VI. CONCLUSIONS

Considering the near-far problem, we proposed an efficient single-transceiver CDMA-based MAC protocol with variable ATIM window. In our scheme, time axis is divided into repeated beacon intervals, each of which includes an ATIM window and a data window. The nodes estimate the number of active nodes to determine the length of the ATIM window, and then exchange the ATIM/ACK frames in the ATIM window to choose the appropriate CDMA codes and transmission power for data packet transmissions without interfering the nearby existing communicating pairs. We developed an analytical model to investigate the aggregate throughput of the proposed protocols. Both the simulation and analytical results agree well
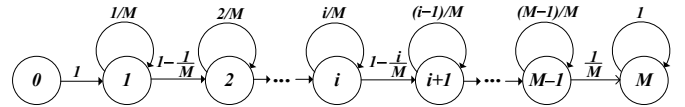

Fig. 11. Markov chain model for the number of active slots.

and also show that our CMVAW scheme can improve the aggregate throughput significantly as compared to the CMFAW and the C-T protocol. In terms of hardware implementations, our scheme is also cost-effective for the large scale wireless networks since each node needs only a single transceiver.

APPENDIX

A. The Derivation of Eq. (2)

Let M be the total number of slots and m be the number of observed busy slots. Notice that each node independently chooses a slot with a probability of $1/M$, given the number n of active nodes, then we can use a Markov chain to derive the conditional pmf of $p(m|n) \triangleq \Pr\{M_o = m|N_a = n\}$. Note that the probability that a node chooses one of the slots is $1/M$. Then, we depict the Markov chain in Fig. 11, where the number in the circle implies the number of observed active slots at the corresponding state. Based on the Markov chain shown in Fig. 11, we get the transition probability as follows:

$$q_{ij} = \begin{cases} \frac{i}{M}, & i = j, \\ 1 - \frac{i}{M}, & j = i + 1, \\ 0, & \text{otherwise,} \end{cases} \quad (25)$$

where $0 \leq i, j \leq M$. Thus, according to Eq. (25), we can obtain the transition probability matrix, denoted by $\mathbb{Q}$, such that $\mathbb{Q}|_{(i,j)} = q_{ij}$. Note that $\mathbb{Q}$ is an $(M+1) \times (M+1)$ upper bidiagonal matrix. The probability that the number of observed active slots is m conditioning on n is equivalent to the n-step transition probability from state of 0-active-slot to state of m-active-slot. Therefore, $\Pr\{M_o = m|N_a = n\} = p(m|n)$ can be expressed as

$$\Pr\{M_o = m|N_a = n\} = p(m|n) = (\mathbb{Q}^n)|_{(0,m)}, \quad (26)$$

which is Eq. (2).

B. The Derivation of Eq. (22)

According to Bayes' Theorem, $p(w|k, m)$ can be expressed as:

$$p(w|k, m) = \frac{p(k, m, w)}{p(k, m)} = \frac{p(w|k)p(m|k, w)}{p(m|k)}. \quad (27)$$

where $p(m|k)$ is known as the transition probability, but $p(w|k)$ and $p(m|k, w)$ are unknown. Further, $p(m|k, w)$ in Eq. (27) can be derived as

$$
\begin{aligned}
p(m|k, w) &= \sum_v p(m|k, w, v)p(v|k, w) \\
&= \sum_v \frac{p(m|k, w, v)p(w|k, v)p(k, v)}{p(k, w)} \\
&= \sum_v \frac{p(m|k, w, v)p(w|k, v)p(v|k)}{p(w|k)} \\
&= \frac{\sum_v p(m|k, w, v)p(w|k, v)p(v|k)}{p(w|k)}. \quad (28)
\end{aligned}
$$

Substituting Eq. (28) into Eq. (27), we obtain

$$
\begin{aligned}
p(w|k, m) &= \frac{\sum_v p(m|k, w, v) p(w|k, v) p(v|k)}{p(m|k)} \\
&= \frac{\sum_v p(m|k, w, v) p(w|k, v) p(v|k)}{\rho_{km}}, \quad (29)
\end{aligned}
$$

which is Eq. (22).

References

[1] K. Gilhousen, I. Jacobs, R. Padovani, A. Viterbi, L. Weaver, and C. Wheatley III, "On the capacity of a cellular CDMA system," *IEEE Trans. Veh. Technol.,* vol. 40, pp. 303–312, 1991.

[2] L. Hu, "Distributed code assignments for CDMA packet radio networks," *IEEE/ACM Trans. Networking,* vol. 1, pp. 668–677, Dec. 1993.

[3] A. Bertossi and M. Bonuccelli, "Code assignment for hidden terminal interference avoidance in multihop packet radio networks," *IEEE Trans. Commun.,* vol. 3, pp. 441–449, Aug. 1995.

[4] J. Garcia-Luna-Aceves and J. Raju, "Distributed assignment of codes for multihop packet-radio networks," in *Proc. 1997 IEEE MILCOM,* vol. 1, pp. 450–454, 1997.

[5] E. Sousa and J. Silvester, "Spreading code protocols for distributed spread-spectrum packet radio networks," *IEEE Trans. Commun.,* vol. 36, no. 3, pp. 272–281, Mar. 1988.

[6] S. Lee and D. Cho, "Distributed reservation CDMA for wireless LAN," in *Proc. 1995 IEEE GLOBECOM,* vol. 1, pp. 360–364, 1995.

[7] A. Muqattash and M. Krunz, "CDMA-based MAC protocol for wireless ad hoc networks," in *Proc. MobiHoc'03.*

[8] M. Joa-Ng and I. Lu, "Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network," in *Proc. INFOCOM'99.*

[9] E. Sousa and J. Silvester, "Optimum transmission range in a direct-sequence spread-spectrum multihop packet radio network," *IEEE J. Sel. Areas Commun.,* vol. 8, no. 5, pp. 762–771, 1990.

[10] C. L. Weber, G. K. Huth, and B. H. Batson, "Performance consideration of code division multiple-access systems," *IEEE Trans. Veh. Technol.,* vol. 30, no. 1, pp. 3–10, 1981.

[11] J. Wang, Y. Fang, and D. Wu, "A power-saving multi-radio multi-channel MAC protocol for wireless local area networks," in *Proc. INFOCOM'06.*

[12] M. Miller and N. Vaidya, "Improving power save protocols using carrier sensing and busy-tone for dynamic advertisement windows," Technical Report, UIUC, 2004

[13] IEEE Standard 802.11 - 1999; Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; Nov. 1999.

[14] G. Stuber, *Priciples of Mobile Communication,* 2nd edition. Kluwer Academic Publishers, 2001.

Xi Zhang (S'89-SM'98) received the B.S. and M.S. degrees from Xidian University, Xi'an, China, the M.S. degree from Lehigh University, Bethlehem, PA, all in electrical engineering and computer science, and the Ph.D. degree in electrical engineering and computer science (Electrical Engineering-Systems) from The University of Michigan, Ann Arbor.

He is currently an Assistant Professor and the Founding Director of the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station. He was an Assistant Professor and the Founding Director of the Division of Computer Systems Engineering, Department of Electrical Engineering and Computer Science, Beijing Information Technology Engineering Institute, Beijing, China, from 1984 to 1989. He was a Research Fellow with the School of Electrical Engineering, University of Technology, Sydney, Australia, and the Department of Electrical and Computer Engineering, James Cook University, Queensland, Australia, under a Fellowship from the Chinese National Commission of Education. He was with the Networks and Distributed Systems Research Department, AT&T Bell Laboratories, Murray Hills, NJ, and with AT&T Laboratories Research, Florham Park, NJ. He has published more than 200 research papers in the areas of wireless networks and communications systems, mobile computing, network protocol design and modeling, statistical communications, random signal processing, information theory, and control theory and systems.

Prof. Zhang received the U.S. National Science Foundation CAREER Award in 2004 for his research in the areas of mobile wireless and multicast networking and systems. He is an IEEE Communications Society Distinguished Lecturer. He received the Best Paper Awards in the IEEE GLOBECOM 2007, IEEE GLOBECOM 2009, and IEEE WCNC 2010, respectively. He also received the TEES Select Young Faculty Award for Excellence in Research Performance from the Dwight Look College of Engineering at Texas A&M University in 2006. He is currently serving as an Editor for IEEE TRANSACTIONS ON COMMUNICATIONS and an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He has served as an Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and an Associate Editor for the IEEE COMMUNICATIONS LETTERS. He is serving as a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS for the special issue on "Broadband Wireless Communications for High Speed Vehicles," a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS for the special issue on "Wireless Video Transmissions," a Guest Editor for *IEEE Communications Magazine* for the special issue on "Advances in Cooperative Wireless Networking," a Guest Editor for *IEEE Wireless Communications Magazine* for the special issue on "Next Generation of CDMA Versus OFDMA for 4G Wireless Applications." He is also serving as an Editor for *Wiley's Journal on Wireless Communications and Mobile Computing,* an Editor for the *Journal of Computer Systems, Networking, and Communications,* and an Associate Editor for *John Wiley's Journal on Security and Communications Networks.*

Prof. Zhang is serving or has served as the TPC Chair for IEEE GLOBECOM 2011, TPC Area Chair for IEEE INFOCOM 2012, General Co-Chair for INFOCOM 2012 - Workshop on Communications and Control for Sustainable Energy Systems: Green Networking and Smart Grids, General Co-Chair for IEEE INOFOCOM 2011 - Workshop on Green Communications and Networking, TPC Co-Chair for the IEEE ICDCS 2011 - Workshop on Data Center Performance, Panels/Demos/Posters Chairs for the ACM MobiCom 2011, TPC Vice-Chair for IEEE INFOCOM 2010, Symposium Co-Chair for IEEE GLOBECOM 2008 – Wireless Communications Symposium, Symposium Co-Chair for IEEE ICC 2008 – Information and Network Security Symposium, Poster Chair for IEEE INFOCOM 2008, Student Travel Grants Co-Chair for IEEE INFOCOM 2007, etc.

Hang Su received B.S. and M.S. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2002 and 2005, respectively. He is currently a research assistant working toward a Ph.D. degree at the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station. He worked as a software engineer with Nokia Research Center, Hangzhou, China, in 2005. His research interests include wireless sensor networks and vehicle ad hoc networks with emphasis on design and analysis of MAC and routing protocols.