Contention Resolution in Random-Access Wireless Networks Based on Orthogonal Complementary Codes

Xiaohua Li, *Member, IEEE*

***Abstract*—This paper proposes a new method for contention resolution in random-access wireless networks. Using orthogonal complementary codes to design access-request packets, users can reserve channel access successfully, even in severe contentions. Collisions among access-request packets can be resolved and exploited, whereas collisions among data packets are avoided. System throughput and delay performance can be enhanced, because random-access contention becomes transparent. Specifically, system throughput approaches the offered load up to the maximum value one with improved average packet delay performance. A joint layer design approach is proposed with both the physical layer signal-detection algorithm and the medium access-control layer random-access protocol. The performance is analyzed with the consideration of signal detection errors. Simulations are performed to demonstrate its superior performance.**

***Index Terms*—Contention resolution, random access, reservation ALOHA, wireless network.**

I. Introduction

SINCE WIRELESS spectrum is becoming a scarce resource, it is critical to make wireless networks optimal in spectrum efficiency. One of the difficulties for enhancing efficiency comes from the fact that traffic is heterogeneous, which requires sophisticated channel-access schemes.

Multiuser channel-access schemes include static channel assignment, such as time-division multiple access (TDMA), code-division multiple access (CDMA), and random-access channel assignment [1], [2], such as the ALOHA algorithm, carrier-sense multiple access (CSMA), the tree algorithm [3], and other adaptive algorithms [4]. For bursty data traffic, random-access schemes are usually more efficient. Unfortunately, they suffer greatly from access contention.

In wired networks, CSMA with collision detection (CSMA/CD) is widely used [2] for contention resolution. However, its efficiency degrades greatly in wireless networks because carrier sense and collision detection become difficult in a multipath fading environment [1].

On the other hand, ALOHA-based random-access protocols, although with efficiency problems, are still widely investigated due to their simplicity and robustness. Typical examples include [5], [6], and the slotted ALOHA used in global systems for mobile communications (GSM) and general packet radio system (GPRS) cellular systems [1]. To improve efficiency, reservation techniques may be integrated with ALOHA. There are various such reservation-based implementations, e.g., packet reservation multiple access (PRMA) [7], [8], and request to send/clear to send (RTS/CTS) [9], [10]. A typical reservation ALOHA is that some slots are subdivided into several minislots, which are used for channel reservation purposes [1], [11]–[13]. However, although contention among data packets can be reduced with reservation techniques, contention in the minislots is increased. This still degrades system performance, sometimes even more severely.

With the traditional contention-resolution techniques, system performance may still be severely degraded by packet collisions, especially when the traffic load is heavy. More important, quality of service (QoS) [14] is difficult to keep. One objective of this paper is to develop random-access schemes which are not only more efficient, but also have priority scheduling capability.

Traditional random-access schemes are addressed in the medium access-control (MAC) layer only, where collided packets are simply discarded. If considering jointly the physical layer and the MAC layer, more effective ways are available for contention resolution. Some approaches were proposed which utilize signal separation techniques for collision resolution [15], [16]. However, they suffer from many practical problems, such as ill channel condition, difficult order determination, and high computational complexity [17], [18]. Therefore, our another objective is to develop robust and computationally efficient algorithms for collision resolution.

In this paper, we present a new method which jointly designs the physical layer signal-separation algorithm and the MAC layer random-access protocol to resolve contentions. Specifically, signal-separation techniques are applied in the physical layer for collision resolution, which is simplified and made robust by the MAC layer transmission scheduling protocol. In addition, the MAC layer scheduling protocol is assisted by the physical layer signal separation. This way, packet collision can be avoided or utilized. System performance can be much enhanced. Furthermore, because random-access contention becomes transparent, i.e., known to the receivers/transmitters, it is more effective for QoS support.

This paper is organized as follows. In Section II, the contention system model is described. In Section III, we present the new method, which is then analyzed in Section IV. In Section V,

Paper approved by M. Zorzi, the Editor for Multiple Access of the IEEE Communications Society. Manuscript received September 30, 2002; revised February 14, 2003 and April 28, 2003. This paper was presented in part at the IEEE International Conference on Communications, Anchorage, AK, May 11–15, 2003.

The author is with the Department of Electrical and Computer Engineering, State University of New York at Binghamton, Binghamton, NY 13902 USA (e-mail: xli@binghamton.edu).

Digital Object Identifier 10.1109/TCOMM.2003.822173

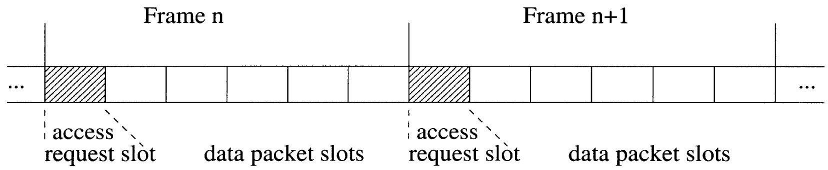


Fig. 1. Structure of packet slots.

simulations are performed to demonstrate its performance. Finally, conclusions are presented in Section VI.

II. System Models

A. Wireless Random-Access Network

The wireless network considered in this paper has a central controller and multiple mobile users, such as cellular systems or wireless LAN with access points [10]. We consider only the case that all mobile users communicate with the central controller. We do not consider peer-to-peer communication such as that defined in IEEE 802.11, although the proposed scheme can be modified to support it.

Each mobile user has a unique ID after registering into the network. The central controller schedules random access of all users to a single slotted channel and maintains slot synchronization. Mobile users send access-request packets during the access-request slots to request channel access.

As shown in Fig. 1, the channel is subdivided into a sequence of frames. Each frame contains one access-request slot and multiple data packet slots whose number may change according to the number of access requests. We define the users who hold data packets for transmission in a frame as *active users*. In the beginning of each frame, the central controller broadcasts a beacon (short message) to all users asking for access request. Every active user then sends an access-request packet to the central controller during the access-request slot. From the received collided signal, the central controller detects all active users, and assigns them data packet slots. Therefore, the active users can transmit data packets without collisions.

Fig. 1 illustrates only the uplink channel from mobile users to the central controller. Downlink transmission can be performed in data slots or in another channel, which is contention-free and thus omitted in this paper. In addition, information about how many packets each active user has can be piggybacked in the data packets. Although we consider the case where only one user is selected to transmit in each data packet slot, the proposed method can be easily extended to support multiple packet reception.

B. Access-Request Packets

One of the major differences between the proposed method and the traditional reservation-based approaches is how the access-request slot is used. Traditionally, this slot is subdivided further into several minislots, which are randomly selected by active users to transmit access-request packets [11]–[13]. However, in wireless networks, since the length of each minislot should be long enough for reliable signal detection, the number of minislots in each frame is limited. Collisions in the minislots may severely degrade system performance.

In contrast, the access-request slot in the proposed method is not further subdivided. All active users simply transmit their access-request packets at the same time. In order for the central controller to detect active users from the collided signal, the access-request packets should be properly designed so that sufficient processing gain can be achieved for every user within a short slot. We will show that this task can be done with orthogonal complementary codes.

C. Orthogonal Complementary Codes

Orthogonal complementary (OC) codes [19], [20] are code set $\{c_{i,j}(\ell)\}$ with the following parameters:

$i = 0, \cdots, I-1$, where I is the flock size;
$j = 0, \cdots, J-1$, where J is the family size;
$\ell = 0, \cdots, L_c - 1$, where L_c is the code length.

Each code set contains I flocks. Each flock contains J family members, whereas each family member is a code sequence with length L_c. We define the family members as code vectors $\mathbf{c}_{i,j} \triangleq [c_{i,j}(0), \cdots, c_{i,j}(L_c-1)]^T$, where $(\cdot)^T$ denotes transpose.

The autocorrelation of each flock is zero except at zero delay, whereas the crosscorrelation between different flocks is zero. For example, for the OC code with $i = j = 4$ [19], we have $[\mathbf{c}_{0,0}^T, \mathbf{c}_{0,1}^T, \mathbf{c}_{0,2}^T, \mathbf{c}_{0,3}^T][\mathbf{c}_{0,0}^T, \mathbf{c}_{0,1}^T, \mathbf{c}_{0,2}^T, \mathbf{c}_{0,3}^T]^T = 64$ and $[\mathbf{c}_{0,0}^T, \mathbf{c}_{0,1}^T, \mathbf{c}_{0,2}^T, \mathbf{c}_{0,3}^T][\mathbf{c}_{1,0}^T, \mathbf{c}_{1,1}^T, \mathbf{c}_{1,2}^T, \mathbf{c}_{1,3}^T]^T = 0$.

Define the $L_c \times L_c$ shifting matrix as

$$\mathbf{J} \triangleq \begin{bmatrix} 0 & & & \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix}. \tag{1}$$

Then the OC codes satisfy the correlation property

$$\sum_{j=0}^{J-1} \mathbf{c}_{i,j}^H \mathbf{J}^n \mathbf{c}_{k,j} = J L_c \delta(i-k)\delta(n),$$
$$n = 0, 1, \cdots; \quad i = 0, \cdots, I-1; \quad k = 0, \cdots, I-1 \tag{2}$$

where $(\cdot)^H$ denotes Hermitian, and $\delta(\cdot)$ is the Dirac-delta function. The processing gain is JL_c, which is achieved only if $n = 0$ and $i = k$.

III. Access Request and Detection

We propose to design the access-request packets with the OC codes. This way, sufficient processing gain can be achieved for all users, and multiaccess interference (MAI) can be completely eliminated, which is in the same spirit as multiuser detection [21]. Transmission can be asynchronous with near–far (NF) propagation and frequency-selective fading. Channels are assumed time invariant within each access-request slot.

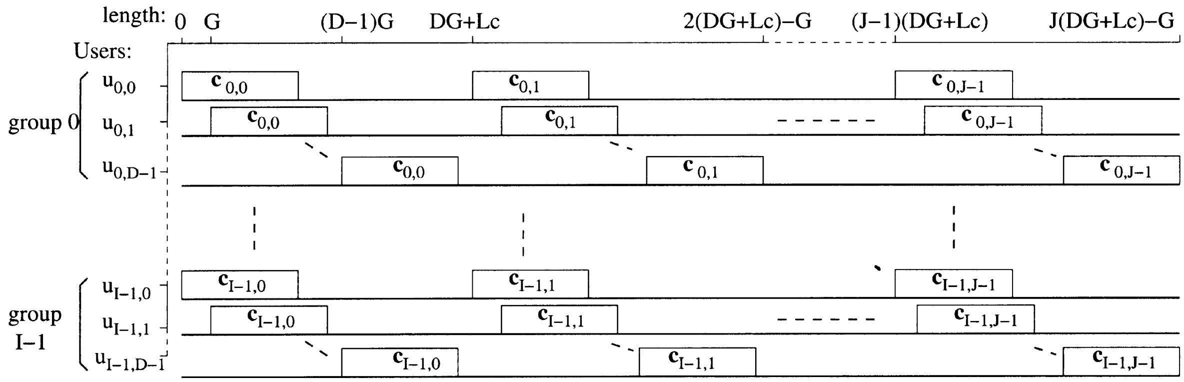


Fig. 2. Structure of access-request packets. Each row shows an access-request packet corresponding to the user marked in the left.

A. Access-Request Packet Design

Assume there are, altogether, U users in the network. We subdivide them into I groups corresponding to I flocks of codes. Each group contains, at most, $D = \lceil U/I \rceil$ users, where $\lceil x \rceil$ denotes the minimum integer not less than x. We denote the users uniquely as $u_{i,d}$, where $i = 0, \cdots, I-1$, and the integer d satisfies $0 \le d \le D-1$.

The access-request packet for the user $u_{i,d}$ is designed as the sequence

$$\mathbf{b}_{i,d} = \big[\mathbf{0}_{dG},\ \mathbf{c}_{i,0}^T,\ \mathbf{0}_{DG},\ \cdots,\ \mathbf{c}_{i,J-2}^T,\ \mathbf{0}_{DG},\ \mathbf{c}_{i,J-1}^T,\ \mathbf{0}_{(D-d-1)G}\big]^T \quad (3)$$

where $\mathbf{0}_k$ is a zero vector with length k (whether it is a column or row vector is obvious and thus omitted), and G is some integer parameter. We will see later that G is the length of the guard interval determined by delays and channel lengths. The length of the access-request packet $\mathbf{b}_{i,d}$ is then

$$L_p = (JD-1)G + JL_c. \quad (4)$$

Equation (3) shows that all the users in the same group i transmit the same code $\mathbf{c}_{i,j}, j = 0, \cdots, J-1$, as access-request packets, with different delay dG. The access-request packets are illustrated in Fig. 2.

B. Access-Request Detection

Consider access-request detection performed at the central controller in Frame n (c.f., Fig. 1). First, we set up the model of the received signal. Each user $u_{i,d}$ is with a finite impulse response (FIR) channel of length $L_h(i,d)$, and begins transmission after an integer delay $\tau_{i,d}$ (due to asynchronous transmission). We can choose the guarding length G in (3) to satisfy $G \ge \max_{\forall i,d} L_h(i,d) + \tau_{i,d}$. The channel length and delay can be overestimated as $L_h \ge \max_{\forall i,d} L_h(i,d)$, $\tau \ge \max_{\forall i,d} \tau_{i,d}$. Then we choose

$$G = L_h + \tau. \quad (5)$$

Let the channel of the user $u_{i,d}$ be $\mathbf{h}_{i,d} = [h_{i,d}(0), \cdots, h_{i,d}(L_h - 1)]^H$. Define $L_p + G - 1$ dimensional vector $\mathbf{s}_{i,d}(\ell) \triangleq [\mathbf{0}_\ell, \mathbf{b}_{i,d}^H, \mathbf{0}_{G-\ell-1}]^H$, $0 \le \ell \le G-1$. Then in Frame n, the noiseless received signal from the active user $u_{i,d}$ is

$$\mathbf{x}_{i,d}^H(n) = \mathbf{h}_{i,d}^H \begin{bmatrix} \mathbf{s}_{i,d}^H(\tau_{i,d}) \\ \vdots \\ \mathbf{s}_{i,d}^H(\tau_{i,d} + L_h - 1) \end{bmatrix} \triangleq \mathbf{h}_{i,d}^H \mathbf{S}_{i,d}(\tau_{i,d}). \quad (6)$$

The received signal at the central controller is then

$$\mathbf{y}^H(n) = \mathbf{x}^H(n) + \mathbf{v}^H(n) = \sum_{i=0}^{I-1}\sum_{d=0}^{D-1} A_{i,d}\mathbf{x}_{i,d}^H(n) + \mathbf{v}^H(n) \quad (7)$$

where $A_{i,d}$ is an indication function with value one if the user $u_{i,d}$ is active, and zero otherwise. The elements in the noise vector $\mathbf{v}(n)$ are assumed additive white Gaussian noise (AWGN) with zero mean and variance σ_v^2.

Next, we develop the user-detection method. Consider noiseless signal $\mathbf{x}(n)$ at first. In order to detect whether a user $u_{m,k}$, $0 \le m \le I-1$, $0 \le k \le D-1$, is active, we design the $L_p + G - 1$ dimensional detectors for the user $u_{m,k}$ as

$$\mathbf{f}_{m,k}(\ell) = \frac{1}{JL_c}\begin{bmatrix} \mathbf{0}_\ell \\ \mathbf{b}_{m,k} \\ \mathbf{0}_{G-\ell-1} \end{bmatrix}, \quad \ell = 0, \cdots, G-1. \quad (8)$$

Using (8) as a bank of correlators, we have the detector matrix

$$\mathbf{F}_{m,k} = [\mathbf{f}_{m,k}(0), \cdots, \mathbf{f}_{m,k}(G-1)]. \quad (9)$$

Proposition 1: In the noiseless case, we have

$$\mathbf{x}^H(n)\mathbf{F}_{m,k} = A_{m,k}\left[\mathbf{0}_{\tau_{m,k}} \quad \mathbf{h}_{m,k}^H \quad \mathbf{0}_{G-\tau_{m,k}-L_h}\right] \triangleq A_{m,k}\tilde{\mathbf{h}}_{m,k}^H. \quad (10)$$

If $\mathbf{x}^H(n)\mathbf{F}_{m,k} \ne \mathbf{0}$, then the user $u_{m,k}$ is active. Otherwise, it is inactive.

Proof: From (2), (7), and (8), if $k \ge d$, we have

$$\mathbf{x}^H(n)\mathbf{f}_{m,k}(\ell) = \frac{1}{JL_c}\sum_{i=0}^{I-1}\sum_{d=0}^{D-1} A_{i,d}\sum_{q=0}^{L_h-1} h_{i,d}(q)\sum_{j=0}^{J-1}\mathbf{c}_{i,j}^H\mathbf{J}^w\mathbf{c}_{m,j}$$

where $w = \ell - q - \tau_{i,d} + (k-d)G$. We can rewrite it as $(d-k)G + (q+\tau_{i,d}) = \ell$. Because $0 \le \ell \le G-1$ and $0 \le q + \tau_{i,d} \le G-1$, we find that $d = k$. Hence

$$\begin{aligned}\mathbf{x}^H(n)\mathbf{f}_{m,k}(\ell) &= \sum_{i=0}^{I-1} A_{i,k} \sum_{q=0}^{L_h-1} h_{i,k}(q)\delta(i-m)\delta(\ell - q - \tau_{i,k}) \\ &= \begin{cases} A_{m,k}h_{m,k}(\ell - \tau_{m,k}), & \text{if } 0 \le \ell - \tau_{m,k} \le L_h - 1 \\ 0, & \text{else.}\end{cases}\end{aligned}$$

It is then straightforward to obtain (10). Similar results hold if $k \le d$. □

For noisy signals, we have

$$\mathbf{y}^H(n)\mathbf{F}_{m,k} = A_{m,k}\tilde{\mathbf{h}}_{m,k}^H + \mathbf{v}^H(n)\mathbf{F}_{m,k}. \quad (11)$$

It is easy to show that each of the noise part $\mathbf{v}^H(n)\mathbf{f}_{m,k}(\ell)$ is AWGN with zero mean and variance $\sigma_v^2/(JL_c)$.

Detection of $A_{m,k}$ can be made from the strongest output, i.e., $\max_\ell \|\mathbf{y}^H(n)\mathbf{f}_{m,k}(\ell)\|^2$. Or, a better way is to use all the received signal energy. Specifically, if the channel is known, a decision can be made by comparing the decision metric

$$z_{m,k}(n) = \max_{0 \le \ell \le G-1} \frac{\mathbf{y}^H(n)\mathbf{F}_{m,k}}{\|\mathbf{h}_{m,k}\|^2} \begin{bmatrix} \mathbf{0}_\ell \\ \mathbf{h}_{m,k} \\ \mathbf{0}_{G-\ell-L_h} \end{bmatrix} \quad (12)$$

with threshold $\gamma = 1/2$. If the channel is unknown, we can simply use

$$z_{m,k}(n) = \left\|\mathbf{y}^H(n)\mathbf{F}_{m,k}\right\|^2 \quad (13)$$

and choose γ according to the received signal energy.

C. Algorithms

As described in Section II-A, in the beginning of each frame, the central controller asks all active users to transmit access-request packets. After receiving the collided signals, it detects all active users as in Section III-B, and then assigns data packet slots to active users. Since all users can request access and be detected at the same time, access contentions become transparent or known to the central controller. Therefore, it is convenient for the central controller to manage channel access according to various QoS or packet priority criteria.

The proposed method consists of a signal-separation (active user detection) algorithm implemented in the physical layer, and a random-access scheduling protocol implemented in the MAC layer.

Algorithm 1. Signal Separation (Active User Detection)

1) Central controller receives signal $\mathbf{y}(n)$ (7).
2) Uses detector $\mathbf{F}_{m,k}$ (9) to calculate decision metrics $z_{m,k}(n)$ (12), (13).
3) Determines whether each user $u_{m,k}$ is active by comparing the metrics with threshold γ.

Algorithm 2. Random-Access Scheduling Protocol

1) Central controller broadcasts initializing beacon.
2) Active users transmit access-request packets. Central controller detects active users.
3) Central controller schedules channel access and informs active users. Active users transmit data packets in the assigned slots.

The algorithms are efficient in computation. The computational complexity is $O(L_p)$ for each user. From (4), L_p is determined by the product of the number of users U and the *guard length* G. Note that in most other reservation-based techniques, the slot length is the product of U and the *processing gain* JL_c. In practice, $G \ll JL_c$, especially if transmissions can be synchronous or quasi-synchronous. In addition, if channels have been estimated, then G can be further reduced, even to one [19]. Therefore, we can use much shorter access-request slots, or more users can share a slot.

From (3) and Fig. 2, information on user ID is embedded in the access codes. Each user need register with the central controller only once to get the access code, which can be performed with another random-access channel, or simply use a code in (3) reserved for this purpose. The key point here is that each user need register only once, not once for each data packet or each access demand. First, because of the efficient code structure, many users can remain registered even when they are not active. Second, the central controller knows each user's ID during access-request detection. It is not necessary for the user to transmit any ID information, which improves efficiency and is convenient for packet priority scheduling.

Compared with some CDMA-based random-access schemes [11], [12], the advantage of our method is due to the signal-separation capability of the OC codes. From Section III-B, MAI is eliminated in a computationally efficient manner under asynchronous, NF, and multipath environments. Our method is more suitable for distributed packet networks without effective power control and accurate channel estimation.

IV. Performance Analysis

A. Probability of Detection Error

We consider the problem with known channels first. From (11) and (12), decision metrics can be generalized to $z_{m,k}(n) = A_{m,k} + v(n)$, where $v(n)$ is AWGN with zero mean and variance $\sigma^2 = \sigma_v^2/(JL_c\|\mathbf{h}_{m,k}\|^2)$. Decisions can be made as

$$\text{H1}: z_{m,k}(n) \ge \gamma, \quad \text{H2}: z_{m,k}(n) < \gamma \quad (14)$$

where H1: the user $u_{m,k}$ is active, and H2: the user $u_{m,k}$ is not active. It is a standard binary detection problem [13]. With optimal threshold $\gamma = 1/2$, the detection error rate (DER) is

$$P_{\epsilon 1} \triangleq P(\epsilon|\text{H1}) = Q\left(\frac{1-\gamma}{\sigma}\right), \quad P_{\epsilon 2} \triangleq P(\epsilon|\text{H2}) = Q\left(\frac{\gamma}{\sigma}\right). \quad (15)$$

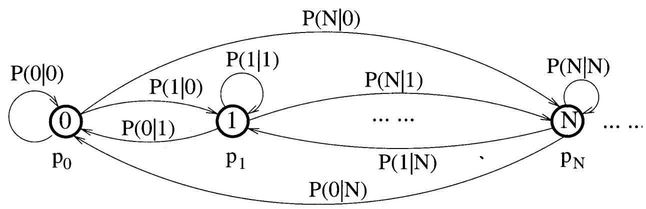


Fig. 3. Illustration of the Markov chain.

A special property in our case is that $P_{\epsilon 1} < 1$ does not reduce system throughput because all affected users will transmit again in the following frames. It may only increase the average packet delay.

If the channels are unknown, from (13), we can still analyze $P_{\epsilon 1}$ and $P_{\epsilon 2}$. It is skipped here to save space.

B. System Throughput

System throughput is defined as the ratio of the number of successfully transmitted data packets to the total number of slots. Let the length of data packet slots and access-request slots be L_d and L_a, respectively. The traffic load is Poisson distributed with an average λ_u packets per slot per user. Then, the overall average traffic load is $\lambda = U\lambda_u$ per slot.

We first consider the case without detection error. From Fig. 1, assume that there are j data packet slots in Frame n. Then the probability that there are i data packets in Frame $n+1$ is

$$P(i|j) = \frac{[(j+1)\lambda]^i}{i!} e^{-(j+1)\lambda}. \quad (16)$$

Define the system state as the number of data packet slots in a frame. The system can be modeled as a Markov chain with state probability p_j and transitional probability $P(i|j)$, $i, j = 0, 1, \cdots$, as illustrated in Fig. 3.

Proposition 2: Without detection error, the average throughput of the system is

$$R = \frac{\sum_{j=0}^{\infty} jL_d p_j}{\sum_{j=0}^{\infty}(jL_d + L_a)p_j} \quad (17)$$

where p_j are the solutions to the linear equation array

$$p_j \sum_{i \neq j} P(i|j) = \sum_{i \neq j} P(j|i)p_i, \quad j = 0, 1, \cdots. \quad (18)$$

Proof: The steady state of the Markov chain (c.f., Fig. 3) satisfies (18) [22]. We consider the first N states p_j, $j = 0, \cdots, N$, for some sufficiently large N, so that $p_j \approx 0$ for all $j > N$. From the $N+1$ equations in (18), $N+1$ variables p_j are readily available. The accurate solutions can be approximated as $N \to \infty$. The expected total number of data packets is $\sum_{j=0}^{N} jp_j$. Since each state is corresponding to j data packet slots and one access-request slot, the throughput can be calculated by (17). □

With detection errors, the analysis becomes more involved. Since accurate analysis is out of the scope of this paper, we derive only approximate results.

Consider $P_{\epsilon 1}$ only. If in Frame n there are j data slots, then there are, on average, $j/(1-P_{\epsilon 1})$ data packets requesting for channel access, among which $jP_{\epsilon 1}/(1-P_{\epsilon 1})$ data packets are delayed until Frame $n+1$, due to wrong detections. Similarly, if in Frame $n+1$ there are i data slots, then there are a total of $i/(1-P_{\epsilon 1})$ data packets. Therefore, from state j to state i, there should be $i/(1-P_{\epsilon 1}) - jP_{\epsilon 1}/(1-P_{\epsilon 1})$ newly generated data packets. Hence, the transitional probability can be approximated as

$$P_1(i|j) = P\left(\left.\frac{i - jP_{\epsilon 1}}{1 - P_{\epsilon 1}}\right| j\right) \quad (19)$$

with $i - jP_{\epsilon 1}$ rounded to the nearest integer (16), and $P_1(i|j) = 0$ if $i - jP_{\epsilon 1} < 0$. The Markov chain in Fig. 3 should then be modified by substituting $P(i|j)$ with $P_1(i|j)$. Denote the steady-state probability as p_{1j}. From (17) and (18), the throughput becomes

$$R_1 = \frac{\sum_{j=0}^{\infty} jL_d p_{1j}}{\sum_{j=0}^{\infty}(jL_d + L_a)p_{1j}}. \quad (20)$$

Next, consider $P_{\epsilon 2}$ only. In this case, we can use state transition probability $P_2(i|j) = P(i - iP_{\epsilon 2}|j)$ to calculate steady-state probability p_{2j}, because there are $i(1-P_{\epsilon 2})$ newly generated data packets. The throughput becomes

$$R_2 = \frac{\sum_{j=0}^{\infty} j(1-P_{\epsilon 2})L_d p_{2j}}{\sum_{j=0}^{\infty}(jL_d + L_a)p_{2j}}. \quad (21)$$

However, (21) is an (approximate) lower bound because we have overestimated the errors.

Finally, consider both decision errors. For simplicity, we assume that they are independent from each other. For state j, there are approximately a total of $j(1-P_{\epsilon 2})/(1-P_{\epsilon 1})$ data packets, among which $jP_{\epsilon 1}(1-P_{\epsilon 2})/(1-P_{\epsilon 1})$ delayed. Then from state j to state i, the number of newly generated packets is $(i - jP_{\epsilon 1})(1-P_{\epsilon 2})/(1-P_{\epsilon 1})$. Therefore, the state transition probability can be approximated as

$$P_3(i|j) = P\left(\left.\frac{(i - jP_{\epsilon 1})(1-P_{\epsilon 2})}{1 - P_{\epsilon 1}}\right| j\right) \quad (22)$$

with which the steady-state probability p_{3j} can be calculated. The throughput can be estimated as

$$R_3 = \frac{\sum_{j=0}^{\infty} j(1-P_{\epsilon 2})L_d p_{3j}}{\sum_{j=0}^{\infty}(jL_d + L_a)p_{3j}} \quad (23)$$

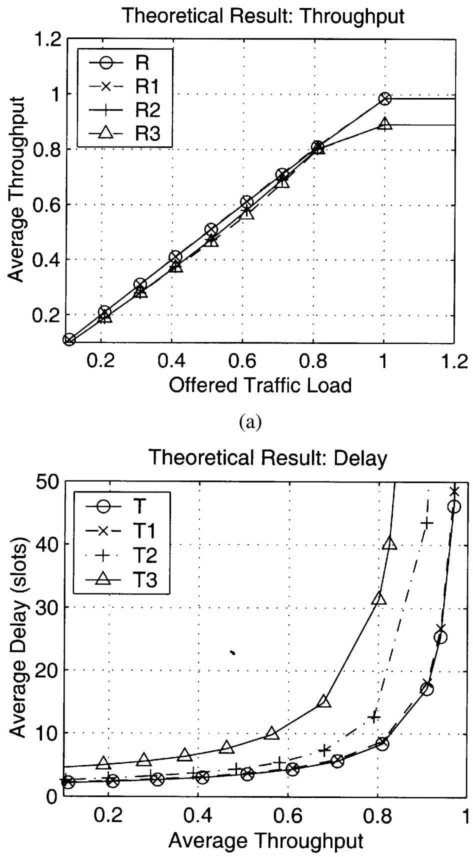
which is again a lower bound only.

Some numerical results are shown in Fig. 4(a). We choose $N = 100$ and $L_d = L_a$ to calculate the state probabilities and throughput. It shows that the method can always achieve high throughput, even in the case of high detection errors. Especially, there is no throughput loss caused by $P_{\epsilon 1}$.

C. Average Packet Delay

Assume $L_d = L_a$. The average packet delay equals the average number of waiting slots of each data packet plus two (the access-request slot and the transmission slot in the frame it is transmitted).

We first consider the decision error-free case. If in Frame n there are j data packet slots, then there are j data packets trans-



(a)

(b)

Fig. 4. Results of throughput and delay analysis. (a) Throughput versus offered load in case of various DERs. R, R_1, R_2, R_3 are throughputs obtained with $P_{\epsilon 1} = P_{\epsilon 2} = 0$ (no error), $P_{\epsilon 1} = 0.1$, $P_{\epsilon 2} = 0.1$, and $P_{\epsilon 1} = P_{\epsilon 2} = 0.1$, respectively. (b) Delay versus throughput. T, T_1, T_2, T_3 are delays obtained with $P_{\epsilon 1} = P_{\epsilon 2} = 0$ (no error), $P_{\epsilon 1} = 0.1$, $P_{\epsilon 2} = 0.1$, and $P_{\epsilon 1} = P_{\epsilon 2} = 0.1$, respectively.

mitted and an average of $(j+1)\lambda$ data packets generated. The data packet transmitted in the first data slot waits for 0 data slot, whereas the one transmitted in the last slot waits for $j-1$ data slots. The total number of waiting data slots equals $j(j-1)/2$. For the newly generated data packets, we assume that they are generated uniformly among slots, i.e., λ packets generated in each slot. Then, the total number of waiting slots for them is $j\lambda + (j-1)\lambda + \cdots + 0\lambda = j(j+1)\lambda/2$.

Therefore, the total waiting time is $j(j-1)/2 + j(j+1)\lambda/2$ in state j. The average delay can be obtained as

$$T = \frac{\sum_{j=0}^{\infty} \left[\frac{j(j-1)}{2} + \frac{j(j+1)\lambda}{2}\right] p_j}{\sum_{j=0}^{\infty} j p_j} + 2 \quad (24)$$

where state probabilities p_j can be solved, according to *Proposition 2*.

Next, consider decision error rate $P_{\epsilon 1}$. For a frame with j data slots, the transmission of j data packets introduces $j(j-1)/2$ waiting slots. In addition, there are $(j+1)\lambda$ new packets, which generate $j(j+1)\lambda/2$ waiting slots. Furthermore, there are $jP_{\epsilon 1}/(1-P_{\epsilon 1})$ delayed packets due to decision errors, which have $(j+1)jP_{\epsilon 1}/(1-P_{\epsilon 1})$ waiting slots. Therefore, the average delay can be approximated as

$$T_1 = \frac{\sum_{j=0}^{\infty} \left[\frac{j(j-1)}{2} + \frac{j(j+1)\lambda}{2} + \frac{j(j+1)P_{\epsilon 1}}{1-P_{\epsilon 1}}\right] p_{1j}}{\sum_{j=0}^{\infty} j p_{1j}} + 2. \quad (25)$$

Third, consider $P_{\epsilon 2}$ only. The number of successfully transmitted data packets reduces to $j(1-P_{\epsilon 2})$. Therefore, the average delay can be overestimated as

$$T_2 = \frac{\sum_{j=0}^{\infty} \left[\frac{j(j-1)}{2} + \frac{j(j+1)\lambda}{2}\right] p_{2j}}{\sum_{j=0}^{\infty} j(1-P_{\epsilon 2}) p_{2j}} + 2. \quad (26)$$

Finally, consider both decision errors. For a frame with j data slots, the number of transmitted data packets is approximately $j(1-P_{\epsilon 2})$, which generates $j(j-1)(1-P_{\epsilon 2})/2$ waiting slots. The $(j+1)\lambda$ newly generated packets contribute $j(j+1)\lambda/2$ waiting slots. In addition, there are $j(j+1)P_{\epsilon 1}(1-P_{\epsilon 2})/(1-P_{\epsilon 1})$ waiting slots due to the delayed packets. Hence, the average delay is overestimated as

$$T_3 = 2 + \frac{\sum_{j=0}^{\infty} \left[\frac{j(j-1)(1-P_{\epsilon 2})}{2} + \frac{j(j+1)\lambda}{2} + \frac{j(j+1)P_{\epsilon 1}(1-P_{\epsilon 2})}{1-P_{\epsilon 1}}\right] p_{3j}}{\sum_{j=0}^{\infty} j(1-P_{\epsilon 2}) p_{3j}}. \quad (27)$$

Some numerical results are shown in Fig. 4(b) with $N = 100$, $L_d = L_a$. It shows that the method can always achieve sufficiently small average delay, even with high detection error. Especially, $P_{\epsilon 1}$ causes only a small increase in delay.

V. Simulations

We first investigate the performance of the access-request detection algorithm (*Algorithm 1*, Section III-C). The measurement is the DER, which is defined as the ratio of total number of wrong detections to total number of detections. Then we compare the new protocol (*Algorithm 2*) with slotted ALOHA (ALOHA), reservation-based ALOHA (R-ALOHA) ([13, p. 683]) and a time-division duplexing (TDD)/CDMA protocol [12] in terms of throughput and delay.

We used the OC code with $i = j = 4$ [19]. Channels were randomly generated for each user with maximum length $L_h = 5$. Transmission delays were also randomly generated from 0 to $\tau = 5$. Therefore, the guard interval length was $G = 10$. We assumed $L_d = L_a = 800$, and $U = 60$ users sharing the access-request slots. To calculate DER, decision threshold γ was set as 1/2 of the received signal energy for each user, which was estimated as the mean value during 10 requests. We used 100 Monte Carlo runs to evaluate the average DER, throughput, and delay.

In the first experiment, we studied the performance of the access-request detection algorithm. Each of the 60 users had a high traffic load of 0.75. The transmitting power of each user was randomly changed to generate the NF propagation ratio. The DER as a function of signal-to-noise ratio (SNR) is shown in Fig. 5(a) with NF ratio 0, 5, and 10 dB, respectively. It shows that sufficiently low DER can be obtained, especially if SNR is larger than 10 dB and NF is not over 10 dB. In this case, DER can be well below 0.1.

The second experiment is to study the effect of DER on system throughput. We carefully adjusted SNR and NF to

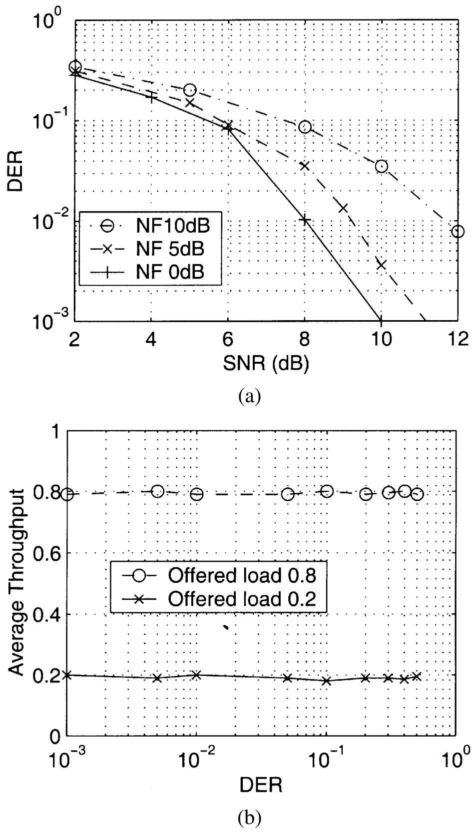


Fig. 5. Performance of access-request detection algorithm in terms of DER. (a) DER versus SNR, which shows that sufficiently low DER can be obtained with low SNR and a high NF ratio. (b) Average throughput versus DER, which shows that system throughput is robust to detection error.

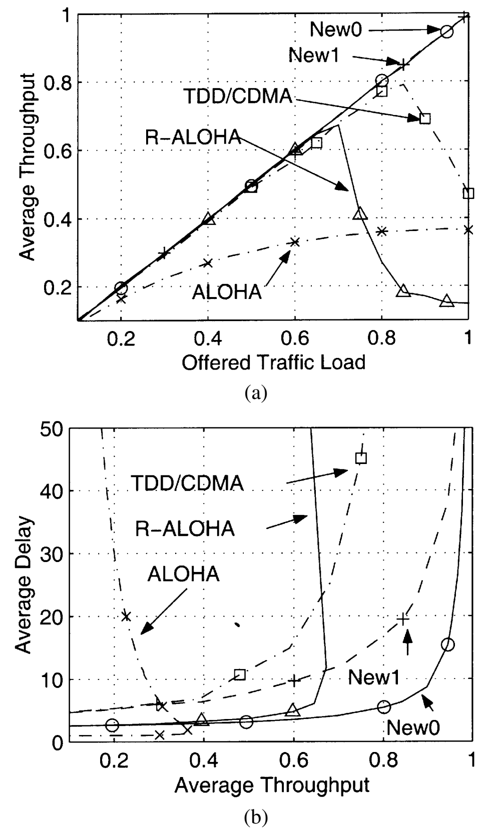


Fig. 6. Compare our method with other random-access protocols in terms of (a) average throughput, and (b) average delay. "New0 ∘:" our method with zero DER (for optimal performance). "New1 +:" our method with high DER $= 0.1$. "TDD/CDMA:" CDMA-based contention resolution. "R-ALOHA △:" reservation ALOHA. "ALOHA ×:" slotted ALOHA.

obtain the required DER. Then we evaluated system throughput for the offered traffic loads 0.2 and 0.8. The results are shown in Fig. 5(b), which shows that the throughput is robust to DER.

Finally, we compare our new method with other random-access protocols. For TDD/CDMA, due to the requirement of transmitting user ID, we used pseudonoise (PN) codes with length (and processing gain) 100. For R-ALOHA, we used eight reservation minislots to achieve the same processing gain as our method within the same access-request slot length. For our method, we calculated the average throughput and delay with a high DER 0.1, and also the results without DER to see the optimal performance. With the same SNR (approximately 8 dB) and NF (approximately 10 dB) as our method, the TDD/CDMA suffered much higher DER, even above 0.5 in the case of intensive traffic load. R-ALOHA and ALOHA were simulated without detection error.

As shown in Fig. 6(a), our method can always achieve almost the offered traffic load up to the maximum value one, which fits the theoretical analysis results as shown in Fig. 4(a). The ALOHA has the worst performance. When the traffic load is light (less than 0.6), the other three protocols do not suffer too much throughput loss. However, when the offered traffic load is high (higher than 0.7), throughput of R-ALOHA degrades rapidly, due to severe collisions in the reservation minislots. TDD/CDMA suffers greatly from detection errors.

The average delay is shown in Fig. 6(b). High DER limitedly increases the average delay of our method, which has also been predicted by the analysis shown in Fig. 4(b). For R-ALOHA and ALOHA, delay increases with packet collisions. For TDD/CDMA, delay increases with both detection errors and packet collisions. In summary, our method outperforms the other three at mid-to-high system traffic load.

VI. Conclusions

In this paper, a new method is proposed to resolve contentions in the random-access wireless networks, with the help of OC codes. Collisions can be separated and used by multiuser detection principles. Theoretical analysis, as well as simulations, show that the new method achieves throughput almost equal to the offered traffic load up to the maximum value one, with limited increase in delay. The computational complexity is as efficient as that of the traditional methods, such as ALOHA.

Compared with other random-access methods, it requires a central controller to manage channel access. When the number of mobile users is too large, then the access-request slot should be long, which reduces efficiency, and the complexity of the central controller becomes also high. In this case, it can be further improved with the consideration of the traffic intensity [23].

References

[1] K. Pahlavan and P. Krishnamurthy, *Principles of Wireless Networks*. Upper Saddle River, NJ: Prentice-Hall, 2002.

[2] F. A. Tobagi, "Multiaccess protocols in packet communication systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 468–489, Apr. 1980.

[3] J. I. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 505–515, Sept. 1979.

[4] L. Kleinrock and Y. Yemini, "An optimal adaptive scheme for multiple access broadcast communication," in *Proc. Int. Conf. Communications (ICC)*, June 1978, pp. 721–725.

[5] L. Musumeci, P. Giacomazzi, and L. Fratta, "Polling- and contention-based schemes for TDMA-TDD access to wireless ATM networks," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1597–1607, Sept. 2000.

[6] A. Polydors and J. Sylvester, "Slotted random-access spread-spectrum networks: An analytical framework," *IEEE J. Select. Areas Commun.*, vol. SAC-5, pp. 989–1002, July 1987.

[7] D. J. Goodman, R. A. Valenzuela, K. I. T. Gayliard, and B. Bamamurthi, "Packet reservation multiple access for local wireless communications," *IEEE Trans. Commun.*, vol. 37, pp. 885–890, Aug. 1989.

[8] G. Benelli, R. Fantacci, G. Giambene, and C. Ortolani, "Voice and data transmission with a PRMA-like protocol in high propagation delay cellular systems," *IEEE Trans. Veh. Technol.*, vol. 49, pp. 2126–2147, Nov. 2000.

[9] Z. J. Haas and J. Deng, "Dual busy-tone multiple access (DBTMA)—a multiple-access control for ad hoc networks," *IEEE Trans. Commun.*, vol. 50, pp. 975–985, June 2002.

[10] *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11, 1997.

[11] I. Akyildiz, D. Levine, and I. Joe, "A slotted CDMA protocol with BER scheduling for wireless multimedia networks," *IEEE/ACM Trans. Networking*, vol. 7, pp. 146–158, Apr. 1999.

[12] V. Huang and W. Zhuang, "Optimal resource management in packet-switching TDD CDMA systems," *IEEE Pers. Commun.*, vol. 17, pp. 26–31, Dec. 2000.

[13] B. Skalar, *Digital Communications, Fundamentals and Applications*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 2001.

[14] J. L. Sobrinho and A. S. Krishnakumar, "Quality-of-service in ad hoc carrier-sense multiple-access wireless networks," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1353–1368, Aug. 1999.

[15] M. K. Tsatsanis, R. Zhang, and S. Banerjee, "Network-assisted diversity for random-access wireless networks," *IEEE Trans. Signal Processing*, vol. 48, pp. 702–711, Mar. 2000.

[16] Q. Zhao and L. Tong, "Semi-blind collision resolution in random-access wireless ad-hoc networks," *IEEE Trans. Signal Processing*, vol. 48, pp. 2910–2920, Oct. 2000.

[17] J. Tugnait, L. Tong, and Z. Ding, "Single-user channel estimation and equalization," *IEEE Signal Processing Mag.*, vol. 17, pp. 17–28, May 2000.

[18] X. Li, "Blind sequence detection without channel estimation," *IEEE Trans. Signal Processing*, vol. 50, pp. 1735–1746, July 2002.

[19] H. Chen, J. Yeh, and N. Suehiro, "A multicarrier CDMA architecture based on orthogonal complementary codes for new generations of wideband wireless communications," *IEEE Commun. Mag.*, vol. 39, pp. 126–135, Oct. 2001.

[20] N. Suehiro and M. Hatori, "N-shift cross-orthogonal sequences," *IEEE Trans. Inform. Theory*, vol. 34, pp. 143–146, Jan. 1988.

[21] M. Honig, U. Madhow, and U. Verdu, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 944–960, July 1995.

[22] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*, 2nd ed. Reading, MA: Addison-Wesley, July 1993.

[23] X. Li and W. Liu, "Smart contention resolution in random-access wireless networks with unknown multiple users: a joint layer design approach," in *Proc. IEEE Int. Conf. Communications*, Anchorage, AK, May 11–15, 2003, pp. 1243–1247.

Xiaohua Li (M'00) received the B.S. and M.S. degrees from Shanghai Jiao Tong University, Shanghai, China, in 1992 and 1995, respectively, and the Ph.D. degree from the University of Cincinnati, Cincinnati, OH, in 2000.

He has been an Assistant Professor with the Department of Electrical and Computer Engineering, State University of New York at Binghamton, Binghamton, NY, since 2000. His research interests are in the fields of adaptive and array signal processing, blind channel equalization, and digital and wireless communications.