Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2008, Article ID 970842, 12 pages
doi:10.1155/2008/970842

Research Article

A Multigroup Priority Queueing MAC Protocol for Wireless Networks with Multipacket Reception

Wen-Fang Yang, Jwo-Yuh Wu, Li-Chun Wang, and Ta-Sung Lee

Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan

Correspondence should be addressed to Wen-Fang Yang, elvanyang.cm91g@nctu.edu.tw

Received 16 January 2008; Accepted 8 May 2008

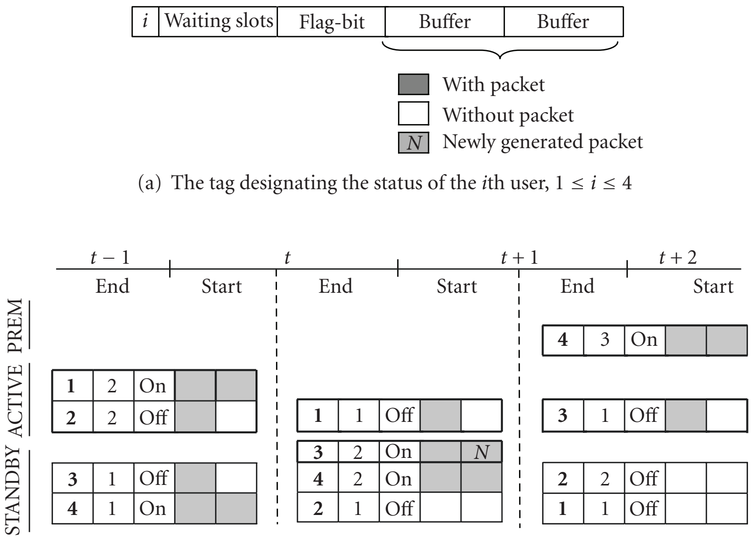
Recommended by Yuh-Shyan Chen

Relying on a simple flag-assisted mechanism, a multigroup priority queueing (MGPQ) medium access control (MAC) protocol is proposed for the wireless networks with multipacket reception (MPR). The proposed MGPQ scheme is capable of overcoming two major performance bottlenecks inherent in the existing MPR MAC protocols. First, the proposed solution can automatically produce the list of active users by observing the network traffic conditions, remove the need of active user estimation algorithm, and thus can largely reduce the algorithm complexity. Second, the packet blocking constraint imposed on the active users for keeping compliant with prediction is relaxed. As a result, the proposed MGPQ is not only applicable to both homogeneous and heterogeneous cases, but also outperforms the existing MPR MAC protocols. Simulation results show that the network throughput can be improved by 40% maximum and 14% average as compared with the well-known dynamic queue (DQ) MAC protocol.

1. INTRODUCTION

1.1. Overview

An efficient medium access control (MAC) mechanism is characterized by high throughput and low delay. Traditionally, the design of MAC protocols is based on the so-called collision channel model, that is, a transmitted packet is successfully received only when no concurrent transmission occurs. Such a paradigm, however, ignores the multipacket reception (MPR) capability at the physical layer, for example, multiuser detection [1]. Recently MAC protocols with the MPR capability draw increasing attention. Several proposals have been reported in the literatures [2–11], almost all of which are devised for the homogeneous environment, that is, all users are associated with the same packet generating probability. An initial attempt to reflect the MPR facility is the channel model with capture effect characterized via the probability of successful reception [2]. The impact of capture effects on various existing MAC protocols such as slotted ALOHA, and FCFS has been addressed in [3–5]. However, the capture model overall remains a simplified representation of the actual channel characteristics and does not explicitly account for the MPR capability. This thus motivates the development of more realistic MPR channel model [6], based on which several MAC protocols have been proposed for realizing various system-wide performance requirements [7–11]. The multiqueue service room (MQSR) protocol [7] is, to the best of our knowledge, the first proposal which relies on the MPR model [6] for user scheduling. It calls for active user prediction via an exhaustive search over all the available network-traffic and physical layer channel capacity information up to the current slot. However, as the total number of users increases, the number of search states grows exponentially thereby incurring high-computational complexity. Moreover, the transmission of the newly generated packets of selected users is not allowed in order to maintain the active user prediction determined via the previous network traffic, inevitably resulting in throughput degradation. The dynamic queue (DQ) protocol introduced in [8] delivers a large portion of performance gain attained by MQSR solution but at reduced complexity. By viewing the traffic as a flow of transmission periods (TP), the DQ protocol otherwise aims for minimization of the expected TP duration by exploiting the MPR property. To further reduce the idle period of users with empty buffer, a modification of DQ scheme that includes active user identification at the receiver is subsequently introduced in [9]. In [10],



(a) The tag designating the status of the ith user, $1 \leq i \leq 4$

(b) The priority grouping process within three consecutive time slots

Figure 1: An illustrative example of MGPQ with four users.

a predictive multicast polling (PMP) scheme was proposed for the general finite buffer size. This approach relies on active user prediction slot by slot, and can significantly improve system throughput since packet blocking is no longer necessary. However, the computational complexity is still a concern. The bit-map assisted dynamic queue (BMDQ) protocol [11], which is essentially a modified DQ scheme, inserts an extra TDMA slot at the head of each TP for channel access/reservation request. However, such an overhead will reduce the bandwidth efficiency, especially when the number of users is large. The two major performance bottlenecks inherent in the existing multipacket reception (MPR) MAC protocols are the computational complexity and the packet blocking constraints. In order to optimize the number of concurrent transmissions, the central controller may rely on an exhaustive search to estimate the buffer status of each user, thus resulting in a high-computational load. Second, the newly generated packets are not allowed to enter the buffer (hence blocked) for maintaining a static buffer status during each processing round.

1.2. Paper contributions

Relying on a simple flag-assisted mechanism and on an associated multipriority user grouping strategy, this paper proposes a new MPR MAC protocol which is applicable to both the homogeneous and heterogeneous cases. Compared with existing protocols, the proposed scheme through grouping users according to the prescribed service priority has two unique advantages. First, it avoids the search for the active users and thus reduces computational complexity [7, 10]. Second, it is free from the packet blocking constraint and can further improve the throughput performance [7–9]. Also, unlike [11] in which a longer overhead is appended for aiding protocol design, the proposed approach relies only on a single bit redundancy and hence minimizes bandwidth expansion. The novel aspects of the proposed MPR MAC scheme can be specially summarized as follows.

(1) A single flag-bit is appended on the tail of the transmitted packet for indicating the existence of the following packet in the buffer. This scheme provides the central controller with the certain partial knowledge about the subsequent network traffic in a *deterministic* fashion. The flag-assisted information can greatly simplify the channel access which can be reserved directly for the users with packets ready to transmit. Note that the *deterministic* knowledge is only available for those users whose packets are successfully received by the base station. Although the mechanism similar to the flag-bit may be available in the existing network protocol such as IEEE 802.11 [12], it is never exploited for facilitating the MPR MAC protocol design.

(2) By exploiting the on-off flag signature, we propose to classify the users into *three* groups with different service priorities: the ACTIVE group consisting of the users with packets to send, the STANDBY group consisting of those with empty buffers, and the PRe-EMptive (PREM) group accommodating those who have stayed in the STANDBY or the ACTIVE group longer than certain waiting period. The users in the ACTIVE group are guaranteed to have packets waiting for transmission. However, those users in the STANDBY group are NOT guaranteed to have no packets waiting for transmission, because there may be packets generated after last successful transmission (note that the successful transmission is the only way for the users to convey the flag-bit information to the base station). The inclusion of the complementary PREM group is to avoid unfair scheduling that can occur in a binary grouping strategy. (If there are merely two groups, users in the STANDBY group would suffer an unlimited service delay since the channels could be constantly reserved for some ACTIVE

links with heavy traffic.) With the trigroup user classification scheme, the priorities of service (from high to low, resp.) are PREM, ACTIVE, and STANDBY. The proposed method integrates the deterministic knowledge of those users in the ACTIVE group and the estimated states of those users in the STANDBY group to derive the optimal waiting period for the PREM group.

(3) Through a Markov chain model of the proposed protocol and an associated analysis of the steady-state transition probabilities, we propose a method for determining the optimal waiting period, subject to the constraint that a uniform mean delay requirement among all users must be met.

(4) In the proposed scheme, the number of users permitted for channel access is deterministically set to be that attaining the MPR channel capacity. This prevents the channel from being overloaded and hence avoids irrecoverable packet collision in a heavy traffic environment.

The rest of paper is organized as follows. Section 2 introduces the MPR channel model. Section 3 describes the proposed MGPQ protocol. The problem of optimal waiting period selection is addressed in Section 4. Simulation results are given in Section 5. Finally, Section 6 concludes this paper.

2. MPR CHANNEL MODEL

Following [8], the MPR channel matrix for M users is described as

$$\mathbf{C} = \begin{bmatrix} C_{1,0} & C_{1,1} & & & \\ C_{2,0} & C_{2,1} & C_{2,2} & & \\ \vdots & \vdots & \vdots & & \\ C_{M,0} & C_{M,1} & C_{M,2} & \cdots & C_{M,M} \end{bmatrix}, \qquad (1)$$

where

$$C_{n,k} = P_r\{k \text{ packets correctly received} \mid n \text{ packets transmitted}\} \qquad (2)$$

for $1 \leq n \leq M$ and $0 \leq k \leq n$. Denote $C_n = \sum_{k=1}^{n} kC_{n,k}$ the expected number of the correctly received packets when n packets are concurrently transmitted. The capacity of an MPR channel is defined as $\eta = \max_{n=1,\dots,M} C_n$. Note that the numbers of simultaneously transmitted packets to achieve the channel capacity may not be unique. Let

$$n_0 = \min\left\{\arg\max_{n=1,\dots,M} C_n\right\} \qquad (3)$$

be the minimum amount of capacity-achieving packets. Hence the maximal number of users permitted to access the channel should be n_0, since there will be no further improvement in system capacity if more than n_0 users are simultaneously served. Note that the MPR matrix (1) can be determined via the physical layer performance metric such as bit error rate; an illustrative example based on CDMA communication can be found in [8].

3. MULTIGROUP PRIORITY QUEUEING PROTOCOL

3.1. System description

A centralized network typically involves two-side communications, namely, downlink and uplink. The former is the transmission from the central controller to users, and the latter is the transmission from users to the central controller. As all the packets of downlink are stored at the buffer of the central controller, MAC can easily exploit the multipacket capability of PHY layer due to the full knowledge about the packet status for all users. Nevertheless, there must be some specially designed mechanism for scheduling the uplink transmission due to the lack of full knowledge about the status of users' buffers in which the packets are stored. We will focus on the uplink in this paper. In the proposed system model, all accesses to the common wireless channel are controlled by the central controller. At the beginning of each slot, the central controller broadcasts an access set to inform the users who are allowed to access the channel in the current slot. Upon reception, the central controller acknowledges the users whose packets are successfully received. Users who transmit packets but do not receive the acknowledgments assume their packets are lost, and will retransmit whenever they are enabled. At the end of this slot, the central controller updates the access set by the proposed multipriority grouping strategy. In this paper, it is assumed that feedback acknowledgement channel (from the central controller to the users) is error free and the incurred time delay is negligible. As in [8], we assume that each user has a buffer of size two. We propose to append one flag bit on the tail of the transmitted packet for indicating if there is a following packet in the buffer. The extra flag bit has the advantage to provide explicit information about the incoming traffic condition, as discussed next.

3.2. An illustrative example

Figure 1 shows an illustrative example for the proposed MGPQ protocol, where the total number of users is $M = 4$ and $n_0 = 2$ users are selected to simultaneously access the channel. In MGPQ, all users are classified into three different priority groups (PREM, ACTIVE, and STANDBY). The condition of the user i is summarized in a tag as shown in Figure 1(a), in which the first field represents user ID, second field is the count of waiting slots, third field marks the on/off status of the flag bit, fourth and fifth fields represent the contents of the buffer. Figure 1(b) depicts the operation of the proposed protocol during three consecutive time slots. At the end phase of slot $t-1$, there is no user in the PREM group, user 1 with two packets and user 2 with one packet are in the ACTIVE group, and user 3 with one packet and user 4 with two packets are in the STANDBY group. The detailed operations of the proposed MGPQ are described as follows.

(1) At the start phase of slot t, with empty PREM group, users 1 and 2 in the ACTIVE group are selected for transmitting packets.

(2) At the end phase of slot t,

(i) upon successful packet reception, user 1 with flag bit on in the start phase is retained in the ACTIVE group; the flag bit is then switched off since there is no packet in the second buffer. User 2 is moved to the tail of the STANDBY group since the flag bit is off;

(ii) the waiting slots of both users 1 and 2 are reset to 1, and the waiting slots of the yet-to-be-served users 3 and 4 are increased to 2;

(iii) user 3 has a newly generated packet in the second buffer, and the associated flag bit is switched on.

(3) At the start phase of slot $t + 1$, there is no user in the PREM group and there is only one user in the ACTIVE group, so users 1 and 3 are selected.

(4) At the end phase of slot $t + 1$,

(i) upon successful packet reception, user 1 is moved to the tail of the STANDBY group (flag bit off). User 3 is moved into the ACTIVE group, and then flag-bit is switched off;

(ii) both the waiting slots of users 1 and 3 are reset to 1, and the waiting slots of the yet-to-be-served users 2 and 4 are increased to 2 and 3 respectively;

(iii) because user 4 has stayed in the STANDBY group for a certain waiting period $S = 3$ (to be specified later), it is moved into the PREM group.

(5) At the start phase of slot $t + 2$, there is one user in the PREM group and one user in the ACTIVE group, so users 4 and 3 will be selected.

3.3. Proposed MGPQ algorithm

The proposed MGPQ protocol is now stated as follows, and the resulting state transition conditions are summarized in Table 1.

(I) Put all users into the PREM group.

(II) Select first n_0 users (by the order of PREM, ACTIVE, and then STANDBY group) to access the channel.

(a) If the packet of a certain user is received successfully, then put the user to the tail of the ACTIVE (if the flag-bit is on) or STANDBY group (if the flag-bit is off). And reset its count of waiting slots to zero.

(b) If, for a certain user, the buffer is empty (no packet sent) or there is packet transmitted but not successfully received, and then put the user back to the tail of the STANDBY or ACTIVE group in which the user originally stayed.

(III) Increase waiting slots of all users by one.

(IV) Move those users with waiting slots equal to S to the PREM group.

(V) Repeat steps (II) to (IV).

We note that, in the initial step, all users should be put in the PREM group rather than the STANDBY group. The rationale behind this choice is to avoid unfair scheduling when the packet generating probability is high. Indeed, if the protocol starts with all users in the STANDBY group, the first-selected n_0 users are likely to stay ACTIVE for a long time. The channel will thus be reserved for such ACTIVE users (with higher service priority), and those in the STANDBY group will then suffer a long delay.

3.4. Stability

System stability in the MAC design is extremely important since it guarantees all users with acceptable delays. A fixed packet arrival rate vector is stable if a transmission probability vector can be found to make all the queues in the corresponding system are stable [13]. However, it is difficult to derive the stability region for MPR protocols due to the complicated interactive queue behavior. Another approach to characterize the stability in the systems with finite buffer size is the absence of deadlock [14], or equivalently, all packets will be successfully received with finite delay. In this section, instead of finding the stability region, we will prove that the MGPQ MAC protocol is stable in terms of the finite-delay criterion.

According to the proposed protocol, the worst case occurs when a certain user is assigned with the *lowest* service priority in the STANDBY group while having two packets in the buffer. In this case, the second buffered packet will experience the longest service delay $d_{\max}$. To prove that the average of $d_{\max}$ is finite, we need the following two lemmas (the detailed proofs can be found in Appendices A and B, resp.).

Lemma 1. *Let p_{ms} be the minimal probability that a packet can be successfully received. Then p_{ms} is bounded away from zero. That is, there exists $\delta > 0$ such that $p_{ms} \geq \delta > 0$.*

Lemma 2. *Let t_k be the total time slots elapsed after k rounds of channel access ($k \geq 1$), and let $t_{\max}$ denote the maximal waiting slots for each access. Then we have*

$$t_k \leq k t_{\max}, \tag{4}$$

where

$$t_{\max} = \begin{cases} \left\lceil \dfrac{M}{n_0} \right\rceil, & \text{if } 1 \leq S \leq \dfrac{M}{n_0}, \\ S, & \text{if } \dfrac{M}{n_0} < S < \infty, \end{cases} \quad \text{and } S \text{ is the waiting period.} \tag{5}$$

Based on the above two lemmas, the following theorem can be sustained.

Theorem 1. *The mean worst-case delay $E[d_{\max}]$ satisfies*

$$E[d_{\max}] \leq t_{\max}\delta^{-1} + \frac{M}{n_0}\delta^{-1} < \infty. \tag{6}$$

Proof. The mean worst-case delay can be expressed as $E[d_{\max}] = E[d_1] + E[d_2]$, where $E[d_1]$ and $E[d_2]$ are the averaged delays upon which the first and the second packets

Table 1: Transition conditions among three different priority groups.

From	To		
	PREM (1st priority)	ACTIVE (2nd priority)	STANDBY (3rd priority)
PREM (1st priority)	– without getting permission to access the channel	– transmitted packet (flag-bit = 1) being successfully received – transmitted packet not being successfully received, and previous flag-bit = 1	– transmitted packet (flag-bit = 0) being successfully received – transmitted packet not being successfully received, and previous flag-bit = 0 – no packet for transmission
ACTIVE (2nd priority)	– without getting permission to access the channel for S slots	– transmitted packet (flag-bit = 1) being successfully received – transmitted packet not being successfully received – without getting permission to access the channel for less than S slots	– transmitted packet (flag-bit = 0) being successfully received
STANDBY (3rd priority)	– without getting permission to access the channel for S slots	– transmitted packet (flag-bit = 1) being successfully received	– transmitted packet (flag-bit = 0) being successfully received – transmitted packet not being successfully received – no packet for transmission – without getting permission to access the channel for less than S slots

associated with the last-to-be-served user are successfully received, respectively. We first observe that

$$\begin{aligned} &E[d_1] \\ &\quad = \sum_{k=1}^{\infty} t_k \Pr\{\text{1st packet successfully received in the } k\text{th round}\} \\ &\quad \le t_{\max} \sum_{k=1}^{\infty} k \Pr\{\text{1st packet successfully received in the} \\ &\qquad\qquad k\text{th round}\} \\ &\quad \le t_{\max} \left\{ \underbrace{p_{ms}}_{k=1} + \underbrace{2p_{ms}(1-p_{ms})}_{k=2} + \underbrace{3p_{ms}(1-p_{ms})^2}_{k=3} + \cdots \right\} \\ &\quad = t_{\max} \sum_{k=1}^{\infty} k p_{ms}(1-p_{ms})^{k-1} = t_{\max} p_{ms}^{-1}. \end{aligned} \tag{7}$$

We note that the considered user will be moved to the ACTIVE group when the first packet is successfully received. In the worst-case, d_2 will incur when all the M users are in the ACTIVE group. Therefore, the central controller will assign users to access the channel in a round-robin way, and the average time slots elapsed per service round is thus M/n_0. Thus, it is implied that

$$\begin{aligned} E[d_2] &\le \underbrace{\left(\frac{M}{n_0}\right) p_{ms}}_{k=1} + \underbrace{2\left(\frac{M}{n_0}\right) p_{ms}(1-p_{ms})}_{k=2} \\ &\quad + \underbrace{3\left(\frac{M}{n_0}\right) p_{ms}(1-p_{ms})^2}_{k=3} + \cdots \\ &= \frac{M}{n_0} \sum_{k=1}^{\infty} k p_{ms}(1-p_{ms})^{k-1} \\ &= \frac{M}{n_0} p_{ms}^{-1}. \end{aligned} \tag{8}$$

Combining (7) and (8), we obtain

$$\begin{aligned} E[d_{\max}] &= E[d1] + E[d2] \le t_{\max} p_{ms}^{-1} + \frac{M}{n_0} p_{ms}^{-1} \\ &\le t_{\max} \delta^{-1} + \frac{M}{n_0} \delta^{-1} < \infty. \end{aligned} \tag{9}$$

□

Note that for those protocols with more than n_0 users allowed to access the channel simultaneously, deadlock may occur if $C_{n_0+i,0} = 1$ for $i \ge 1$. With the benefit from the fixed n_0 accesses, MGPQ is more robust in such a channel environment.

4. OPTIMAL WAITING PERIOD SELECTION

In the proposed protocol, the number of users permitted for channel access is fixed to be n_0, namely, the one attaining the MPR channel capacity. A natural criterion for determining the waiting period S is to maximize the probability that *each* of the selected n_0 users has a packet to send. We first note the probability of the user i (selected from PREM) with a packet to transmit after waiting a period of S is at least [15]

$$\widetilde{p}_i := 1 - \left(1 - p_i\right)^S, \quad i \in \{1, 2, \dots, M\}, \tag{10}$$

where p_i denotes the packet generating probability of the user i. This implies that the larger the waiting period S, the more likely the users in the PREM group have packets to send. As a result, S should be kept as large as possible. However, the unlimited increase in S may incur severe delay penalty. Particularly if $S \to \infty$, the transition from STANDBY to PREM is prevented and the proposed trigroup priority queuing protocol degenerates into a bigroup scheme. To determine an S for striking a balance between large $\widetilde{p}_i$ and small delay, we propose to seek the optimal S_{opt} with which the following set of constraints on the mean delay per user is satisfied:

$$D_i(S) \le D_r, \quad 1 \le i \le M, \tag{11}$$

where $D_i(S)$ stands for the mean delay of the user i and D_r is a uniform delay requirement.

To find the desired S from (11), one crucial step is to determine an explicit expression of $D_i(S)$ in terms of S. Toward this end, we shall determine all the possible transitions of states (an exact definition of a "state" will be specified later) in the proposed protocol. This can be solved by applying Markov chain analysis shown below.

4.1. Markov chain

Associated with the user i ($1 \le i \le M$), we define $x_i(t)$, $y_i(t)$, and $z_i(t)$ to be the assumed value of the waiting slots, the indication of the flag, and the number of packets in the buffer at the tth time slot, respectively. Hence we have $x_i(t) \in \{1, \dots, S\}$, $y_i(t) \in \{0, 1\}$, and $z_i(t) \in \{0, 1, 2\}$. (The waiting period $S \ge \lceil M/n_0 \rceil$ and the buffer of size two are assumed hereafter if not specified otherwise.) Let us further collect $x_i(t)$, $y_i(t)$, and $z_i(t)$ for all users to form $X(t) = (x_1(t), \dots, x_M(t))$, $Y(t) = (y_1(t), \dots, y_M(t))$, and $Z(t) = (z_1(t), \dots, z_M(t))$. The proposed protocol can be modeled by a Markov chain with state space

$$\Omega := \{E(t) \mid E(t) = (X(t), Y(t), Z(t)),\ t \ge 1\}. \tag{12}$$

We note that the number of states is at most $(S \cdot 2 \cdot 3)^M$. However, since in each time slot, exact n_0 users can simultaneously access the channel, it follows that (i) the number of "1" in $X(t)$ must be equal to n_0; (ii) no more than n_0 entries in $X(t)$ will assume the same value. Taking the above constraints into account and using the permutation and combination theory, the number of distinct outcomes of $X(t)$ is (see Appendix C for proof)

$$N_C = \frac{M!}{n_0!(M - n_0)!} \cdot \sum_{m_i} \frac{(S-1)!}{\prod_{i=0}^{n_0}(m_i!)} \cdot \frac{(M - n_0)!}{\prod_{i=0}^{n_0}(i!)^{m_i}}, \tag{13}$$

where the integers $m_i' s$ are found as the solutions to the following equations:

$$\begin{aligned} \sum_{i=0}^{n_0} i \cdot m_i &= M - n_0, \\ \sum_{i=0}^{n_0} m_i &= S - 1. \end{aligned} \tag{14}$$

With (13) and the constraint that there must be packet(s) in the buffer for the users in the ACTIVE group (i.e., $(y_i, z_i) \ne (1, 0)$), the total number of possible states in the system can be reduced to

$$N_S = N_C \cdot 5^M. \tag{15}$$

If there exists some $p_i = 0$ or 1, the total number of states will be further reduced.

4.2. State transition probability

We proceed to compute the state transition probabilities as follows. Assuming that the events of packet generation among users are independent, we have

$$\begin{aligned} &P_r\{E(t+1) = (\widetilde{X}, \widetilde{Y}, \widetilde{Z}) \mid E(t) = (X, Y, Z)\} \\ &\quad = \prod_{i=1}^{M} P_x(\Delta x_i) P_y(\Delta y_i) P_z(\Delta z_i), \end{aligned} \tag{16}$$

where $\widetilde{X} - X = (\Delta x_1, \dots, \Delta x_M)$, $\widetilde{Y} - Y = (\Delta y_1, \dots, \Delta y_M)$, and $\widetilde{Z} - Z = (\Delta z_1, \dots, \Delta z_M)$; $P_x(\Delta x_i)$, $P_y(\Delta y_i)$, and $P_z(\Delta z_i)$ are the probabilities of the increment of state components given (X, Y, Z) (see Appendix D for details). Based on the state transition probabilities (16), we can immediately construct the transition matrix $\mathbf{T}_{N_S \times N_S}$, with which the steady-state probability π_j, $1 \le j \le N_S$, can be readily obtained by

$$\lim_{t \to \infty} \mathbf{T}^t = \begin{bmatrix} \pi_1 & \pi_2 & \cdots & \pi_{N_S} \\ \pi_1 & \pi_2 & \cdots & \pi_{N_S} \\ \vdots & \vdots & \ddots & \vdots \\ \pi_1 & \pi_2 & \cdots & \pi_{N_S} \end{bmatrix}. \tag{17}$$

In this paper, we assume that the above limit exists, and the assumption is justified by numerical results. The mean delay $D_i(S)$ can be then determined as follows.

4.3. Computation of the mean delay

According to Little's law [16], we have

$$D_i(S) = N_i(S)/\lambda_i(S), \tag{18}$$

where $N_i(S)$ is the average number of packets in the buffer of the user i, and $\lambda_i(S)$ is the packet departure rate (i.e., throughput) of the user i. Let $z_{i,j}$ be the number of buffered packets of the user i in the jth state, then we have

$$N_i(S) = \sum_{j=1}^{N_S} \pi_j z_{i,j}. \tag{19}$$

Also, denoted by $p_{B,i}(S)$ the packet blocking probability of user i, therefore

$$p_{B,i}(S) = \sum_{1\le j'\le N_S,\ z_{i,j'}=2,\ i\notin A} \pi_{j'} + \sum_{1\le j'\le N_S,\ z_{i,j'}=2,\ i\in A} \pi_{j'}\left(1-P_S\right), \tag{20}$$

where access set A and success probability P_S are defined in Appendix D. Then it follows that

$$\lambda_i(S) = p_i\left(1 - p_{B,i}(s)\right). \tag{21}$$

Substituting (19) and (21) into (18), we can obtain a functional relation of $D_i(S)$ in terms of S. The solution to (11) can then be computed via numerical search.

4.4. Homogeneous case

In the homogeneous environment, that is, the packet generating probabilities of all users are identical, it can be shown that the mean delay in (18) is independent of waiting period S (the detailed proof is referred to Appendix E). An intuitive explanation of this phenomenon is that, when subject to the same packet generating probability, all users tend to share the same service priority, and hence experience the same average service delay irrespective of the choice of S.

4.5. Extension to finite buffer size

Although the previous derivation is obtained under the assumption that each user has a buffer of size two, it can be easily extended to the case with finite buffer size B by allowing $z_i(t) \in \{0, 1, 2, \ldots, B\}$. The N_S in (15) must also be increased to $N_C(2B+1)^M$ accordingly. This case will be simulated and compared with other MPR MACs in the next section.

5. NUMERICAL RESULTS

In this section, simulations are carried out by Matlab and we first compare the results with the theoretical analysis for a simple scenario to validate the derivation in Section 4. In this paper, throughput is defined as the average of successful packet transmissions per slot; delay is defined as the average elapsed time slots for a packet to be successfully received by central controller; packet loss ratio (PLR) is defined as the average ratio of the number of blocked packets to the number of generated packets. Then in the heterogeneous case, the individual delay curves with increasing S are plotted to show the effect of S on system performance. In the homogeneous case, throughput, delay, and PLR of MGPQ are further compared with those of DQ. Finally, the throughput performance with more users and finite buffer size of MGPQ, predictive multicast polling (PMP) [10], and DQ [8] are compared to verify their scalability.

5.1. Validation of analytical results

This simulation aims at validating the analytical performance results in Section 4. The test system is a CDMA network with random spreading; the packet length, spreading gain, number of correctable errors in a packet, and noise variance are, respectively, 200, 6, 2, and 10 dB as adopted in [8]. The capacity of such an MPR channel in this scenario is 1.7925, which is attained by $n_0 = 2$ concurrent transmissions in each time slot. The total number of users is set to be $M = 3$. We note that the incurred overhead due to the insertion of a flag bit is $1/201 < 0.005$, which is rather small and is thus neglected in the performance evaluation. Figures 2 and 3, respectively, show the mean throughput and mean delay curves for the two scenarios: (i) the heterogeneous case with packet generating probabilities $[p_1, p_2, p_3] = [0.1, 0.9, 0.9]$, and (ii) the homogeneous case with an equal packet generating probability $p_1 = p_2 = p_3 = 0.5$. As we can see from the figures, in both cases the theoretical results well predict the corresponding simulated outcomes. It can also be seen that, in the homogeneous environment, the mean throughput and mean delays remain unchanged as the waiting period increases: this confirms the assertion in Section 4.4. For the heterogeneous case, we impose the mean delay requirement of each user to be less than 4 time slots; by using the results in Section 4.3, the optimal waiting period is computed to be $S_{\text{opt}} = 7$. Figure 4 depicts the mean delay of each user. It can be seen that the delays of all the three users are indeed kept below 4 when $S = S_{\text{opt}} = 7$. We also note from Figure 4 that users with large (or small, resp.) packet generating probabilities p_i experience less (or more) delay. This is not unexpected since, if p_i is large, the flag bit will be on with a high probability and the user will be allowed for accessing the channel more frequently.

5.2. Comparison with previous work of [8]

This simulation further compares the proposed MGPQ scheme with the DQ protocol [8]. We will consider the homogeneous case since the DQ protocol is exclusively tailored for this scenario. The respective throughput curves, including the slotted ALOHA with optimal retransmission probability [8], are plotted in Figure 5. As we can see, the proposed solution can outperform the DQ protocol over a wide range of the packet generating probabilities. The maximal achievable throughput improvement is about 40% for $p = 0.25$. Also, the proposed approach almost achieves the channel capacity 1.7925 whenever $P \ge .3$, whereas the DQ protocol can attain at most 96% of the capacity for $P \ge .8$. Figure 6 shows the delay performances (measured via time slots per packet) of the two schemes. As shown, the proposed method yields a smaller mean delay with light traffic ($P < .4$). This is because the MGPQ method tends to reserve the channel access for those who are more likely to

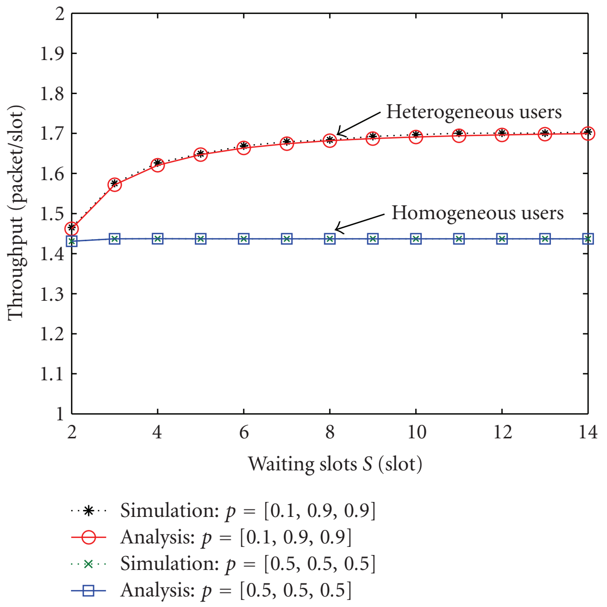


Figure 2: Mean throughput performance of the proposed MGPQ.

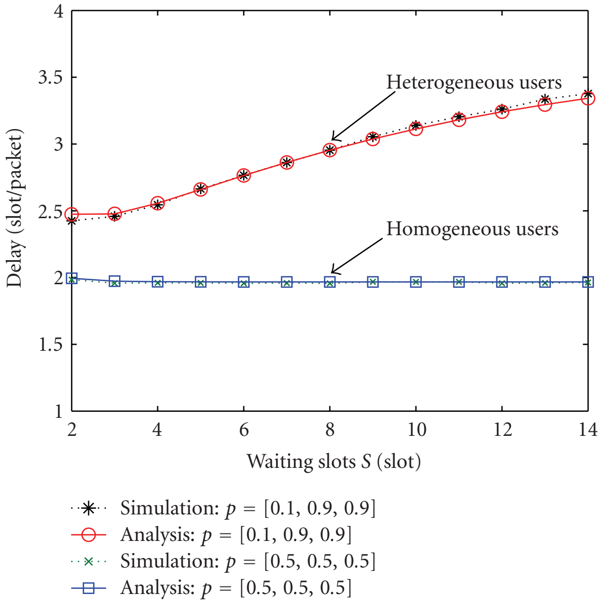


Figure 3: Mean delay performance of the proposed MGPQ.

have packets to send, thus avoiding the time latency incurred by the procedure of network-wide active user prediction. In a heavy-traffic environment, the DQ protocol will block the incoming packets, thereby reduce the mean delay. However, this comes at the expense of a larger PLR, as evidenced in Figure 7.

5.3. General case

In this simulation, we test the proposed protocol with finite buffer size, and compare the performance with the DQ [8]

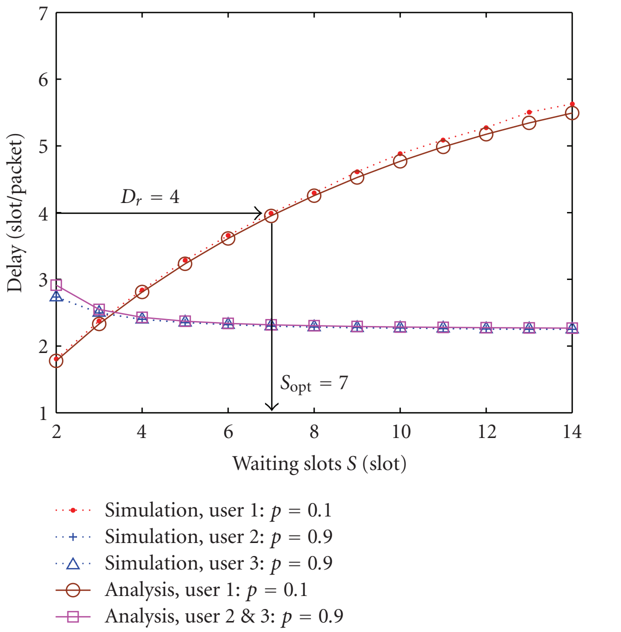


Figure 4: Delay performance of individual users.

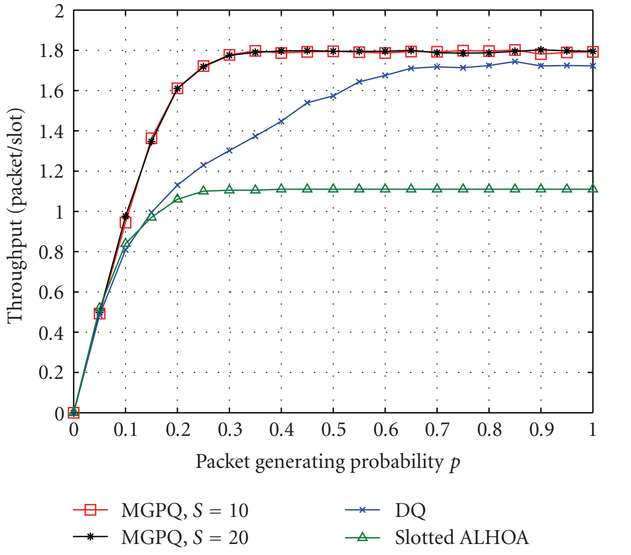


Figure 5: Throughput performance comparison between MGPQ and DQ.

and PMP [10] methods (the latter is specifically devised for the case with finite buffer size). We consider the system setup as in [10] which is described in terms of the MPR matrix as

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & & & \\ 0 & 0 & 1 & & \\ 1 & 0 & 0 & & 0 \\ \vdots & \vdots & \vdots & & \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}, \qquad (22)$$

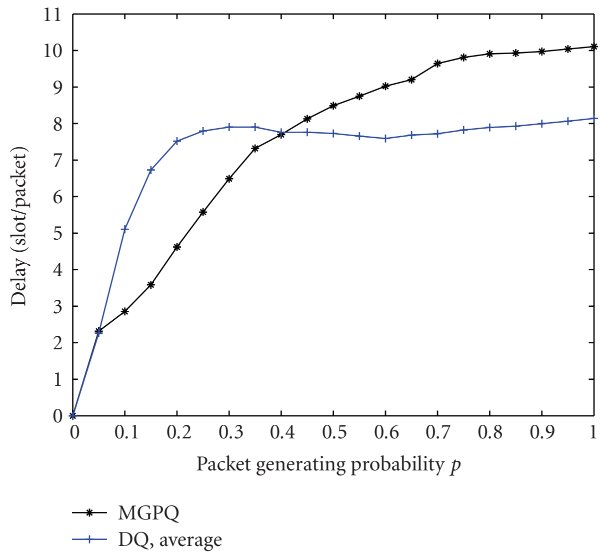


FIGURE 6: Delay performance comparison between MGPQ and DQ.

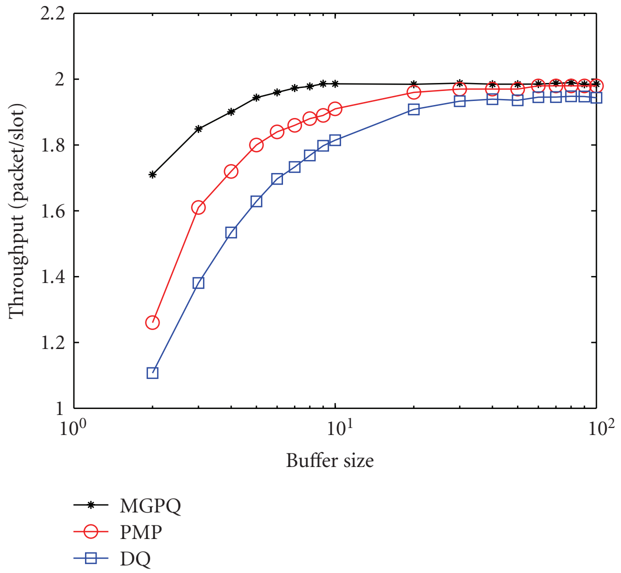


FIGURE 8: Throughput comparison between MGPQ, PMP, and DQ for different buffer sizes.

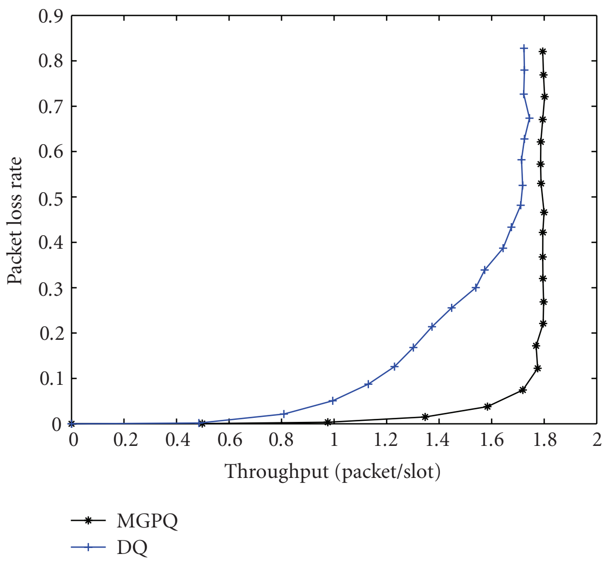


FIGURE 7: Packet loss ratio performance comparison between MGPQ and DQ.

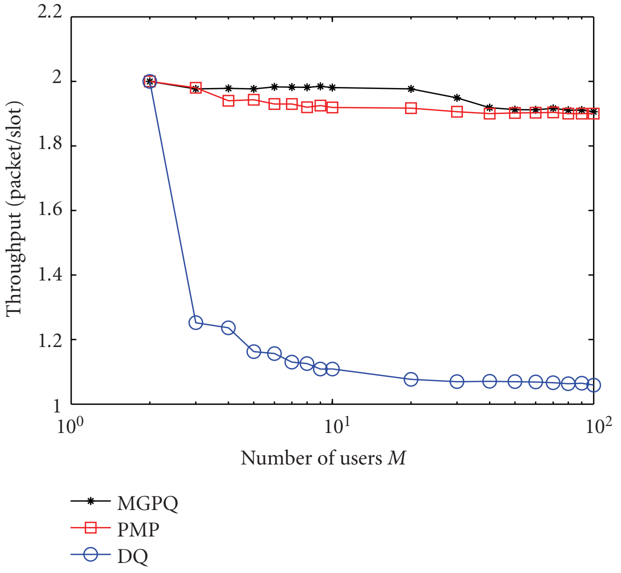


FIGURE 9: Throughput comparison between MGPQ, PMP, and DQ for different number of users.

thus with $n_0 = 2$, $\eta = 2$, and set the total traffic load to be the same with channel capacity. Figure 8 shows the throughput curves of the three methods as the buffer size increases from 2 to 100. It is seen that the DQ scheme results in the lowest throughput, mainly due to the packet blocking constraint. The proposed MGPQ protocol outperforms the PMP solution, thanks to the benefits from the priority mechanism which can reduce the blocking rate especially when the buffer size is small. Figure 9 further depicts the respective throughput performance as the number of user increases from 2 to 100. The result shows that the DQ protocol degrades the performance severely when there are more than two users. This is mainly because in the DQ protocol all users, no matter with packet or not, will be served *continually* until their packets are received successfully or empty slot occurs. With more than n_0 users in the system, the probability of serving idle users is definitely increased.

6. CONCLUSION

In this paper, we proposed a new approach to design the MAC protocol for wireless networks with multipacket reception (MPR) capability. The proposed approach relies on the flag-bit-assisted knowledge about the presence of buffered packets as well as a multipriority user grouping strategy. The advantages of the proposed method are three folds: (1) it is applicable to both the heterogeneous and homogeneous environments, whereas almost all existing protocols developed for the MPR channel are exclusively tailored for the latter case; (2) the insertion of a single bit facilitates the acquisition of network traffic condition with minimal bandwidth expansion; (3) the adopted user grouping policy avoids computationally intensive search for the active users as required in the existing protocols. To prevent an infinitely long service delay in the heterogeneous environment, the waiting period of those yet-to-be-served users can be determined subject to a specified delay requirement. Simulation results show that, compared with the DQ protocol, the proposed scheme achieves higher throughput, reduces the mean delay penalty in light traffic condition, and yields a smaller packet loss ratio. Also, the proposed MGPQ protocol outperforms the predictive multicast polling (PMP) protocol for the general case with finite buffer size. Future work will focus on generalizing the result in this paper to the more realistic generalized MPR channel model [17].

APPENDICES

A. PROOF OF LEMMA 1

According to the definition of η in Section 2, we have

$$\eta = \sum_{k=1}^{n_0} k\cdot C_{n_0,k} = \sum_{k=1}^{n_0-1} k\cdot C_{n_0,k} + n_0\cdot C_{n_0,n_0}. \tag{A.1}$$

If $C_{n_0,n_0} = 0$, then (A.1) becomes

$$\eta = \sum_{k=1}^{n_0-1} k\cdot C_{n_0,k} \leq \sum_{k=1}^{n_0-1} k\cdot C_{n_0-1,k} = \eta', \tag{A.2}$$

where η' corresponds to a higher or equal channel capacity but achieved by sending $n_0 - 1$ packets simultaneously. Note that the inequality in (A.2) holds because the success probability of transmitting more packets simultaneously is less than or equal to that of transmitting less packets under the same channel condition, that is, $C_{n_0,k} \leq C_{n_0-1,k}$. Because (A.2) conflicts with the definition of channel capacity, we conclude that $C_{n_0,n_0} > 0$ with proof by contradiction. Thus, we have $p_{ms} \geq C_{n_0,n_0} = \delta > 0$.

B. PROOF OF LEMMA 2

We first derive the $t_{\max}$ as follows.

For $1 \leq S \leq M/n_0$, do the following.

Let n_{PREM}, n_{ACTIVE}, and n_{STANDBY} denote the number of users in the PREM, ACTIVE, and STANDBY groups, respectively, and then we have

$$n_{\text{PREM}} + n_{\text{ACTIVE}} + n_{\text{STANDBY}} = M. \tag{B.1}$$

Because the user with waiting slots equal to S will be moved to the PREM group, the waiting slots of the users in the ACTIVE and STANDBY groups must be less than S, that is, equal to $1, 2, \ldots,$ or $S-1$. Besides, as n_0 users are selected to access the channel in each slot, the maximal number of users with the same waiting slots must be less than or equal to n_0. Therefore, it can be seen that

$$n_{\text{ACTIVE}} + n_{\text{STANDBY}} \leq (S-1)n_0 \leq \left(\frac{M}{n_0} - 1\right)n_0 = M - n_0. \tag{B.2}$$

Combining (B.1) and (B.2), we have

$$n_{\text{PREM}} = M - \left(n_{\text{ACTIVE}} + n_{\text{STANDBY}}\right) \geq M - \left(M - n_0\right) = n_0. \tag{B.3}$$

Equation (B.3) shows that there will always be at least n_0 users in the PREM group waiting for channel access, which implies that all users will be selected (n_0 users per slot) to access the channel in turn in the PREM group, that is,

$$t_{\max} = \left\lceil \frac{M}{n_0} \right\rceil. \tag{B.4}$$

For $M/n_0 < S < \infty$, the following hold.

According to the MGPQ protocol defined in Section 3.3, all users are in the PREM group initially. After $\lfloor M/n_0 \rfloor$ slots, there will be less than n_0 users left in the PREM group because $M - \lfloor M/n_0 \rfloor n_0 < n_0$; and no user reenters the PREM group because $\lfloor M/n_0 \rfloor \leq M/n_0 < S$. Hereafter, the input rate of the PREM group is less than or equal to the output rate (n_0/slot) of the PREM group, which implies that the users entering the PREM group will be immediately selected to access the channel, that is, $t_{\max} = S$.

C. PROOF OF N_C IN (13)

It is known from the multinomial theorem that [15]

$$\binom{n}{k_1, k_2, \ldots, k_m} = \frac{n!}{k_1!k_2!\cdots k_m!}, \quad \text{where } \sum_{i=1}^{m} k_i = n. \tag{C.1}$$

The above multinomial coefficient can be interpreted as the number of distinct ways to permute a multiset of n elements, and k_i's are the multiplicities of each distinct element. According to the MGPQ protocol defined in Section 3.3, there will be always exactly n_0 users whose waiting slots are one. However, there may be 0 to n_0 users with the same waiting slots ranging from 2 to S, because the users in the ACTIVE group may be selected with higher priority than those in the STANDBY group. Let m_i stand for the number of distinct waiting slots which i users have waited for, $0 \leq i \leq n_0$. Then we have

$$\begin{gathered} N_C = \underbrace{\frac{M!}{n_0!(M-n_0)!}}_{(a)} \cdot \sum_{m_i} \underbrace{\frac{(S-1)!}{\prod_{i=0}^{n_0}(m_i!)}}_{(b)} \cdot \underbrace{\frac{(M-n_0)!}{\prod_{i=0}^{n_0}(i!)^{m_i}}}_{(c)}, \\ \text{where } \underbrace{\sum_{i=0}^{n_0} m_i = S-1}_{(d)}, \quad \underbrace{\sum_{i=0}^{n_0} i\cdot m_i = M-n_0}_{(e)}. \end{gathered} \tag{C.2}$$

In (C.2), (a) is the possible combinations for distinct n_0 users whose waiting slots equal 1; (b) accounts for possible combinations of m_i's in the remaining $S-1$ waiting slots; (c) accounts for possible combinations of i's in the remaining $M-n_0$ users; (d) is the constraint for multinomial coefficient (b), that is, summation of m_i's must equal $S-1$; and (e) is the constraint for multinomial coefficients (c), that is, summation of users in each m_i's must equal $M-n_0$.

D. DESCRIPTION OF STATE TRANSITION PROBABILITY IN SECTION 4.2

Denoted by $A=\{a_1,a_2,\dots,a_{n_0}\}$ the index set of the users who are allowed to access the channel. Also, let n_a be the number of nonzero elements in $\{z_{a_1},z_{a_2},\dots,z_{a_{n_0}}\}$, that is, the number of packets that will be sent simultaneously. Define P_S as the success probability of selected user with packet to send in each slot, then

$$P_S=\sum_{k=1}^{n_a}\frac{k}{n_a}\cdot C_{n_a,k}. \tag{D.1}$$

Thus, the probabilities of the increment of state for X, Y, Z components, that is, $P_x(\Delta x_i)$, $P_y(\Delta y_i)$, and $P_z(\Delta z_i)$ can be calculated by (D.2) according to current state (X,Y,Z).

$$\begin{gathered}
P_x(\Delta x_i=1-x_i)=1,\quad i\in A,\\
P_x(\Delta x_i=1)=1,\quad i\notin A,\\
P_y(\Delta y_i=0)=\begin{cases}1\begin{cases}i\in A, & y_i=0,\ z_i<2,\\ i\in A, & y_i=1,\ z_i=2,\\ i\notin A,\end{cases}\\ 1-P_S,\quad i\in A,\ y_i=0,\ z_i=2,\end{cases}\\
P_y(\Delta y_i=1)=P_S,\quad i\in A,\ y_i=0, z_i=2,\\
P_y(\Delta y_i=-1)=1,\quad i\in A,\ y_i=1, z_i=1,\\
P_z(\Delta z_i=0)=\begin{cases}p_iP_S+(1-p_i)(1-P_S), & i\in A,\ z_i=1,\\ p_iP_S+(1-P_S), & i\in A,\ z_i=2,\\ 1-p_i\begin{cases}z_i=0,\\ i\notin A,\ z_i=1,\end{cases}\\ 1,\quad i\notin A,\ z_i=2,\end{cases}\\
P_z(\Delta z_i=1)=\begin{cases}p_i(1-P_S), \quad i\in A,\ z_i=1,\\ p_i\begin{cases}i\notin A,\ z_i=1,\\ z_i=0,\end{cases}\end{cases}\\
P_z(\Delta z_i=-1)=(1-p_i)P_S,\quad i\in A,\ z_i>0.
\end{gathered}\tag{D.2}$$

E. PROOF OF STATEMENT IN SECTION 4.4

If each user has equal packet generating probability, without loss of generality, we can write the transition matrix as $\mathbf{T}=\mathbf{G}\otimes\mathbf{H}$ by appropriate ordering of states, where $\mathbf{G}$ is the $N_C\times N_C$ transition matrix of state $X(t)$, $\mathbf{H}$ is the $5^M\times 5^M$ transition matrix of state $(Y(t),Z(t))$, and $\otimes$ stands for Kronecker product. Note $\mathbf{G}$ (including size and contents) is the function of the waiting period selection S, and $\mathbf{H}$ is the function of packet generating probability.

To compute the steady-state probabilities, let

$$\begin{gathered}
\mathbf{G}=\begin{bmatrix} g_{1,1} & g_{1,2} & \cdots & g_{1,N_C}\\ g_{2,1} & g_{2,2} & \cdots & g_{2,N_C}\\ \vdots & \vdots & \ddots & \vdots\\ g_{N_C,1} & g_{N_C,2} & \cdots & g_{N_C,N_C}\end{bmatrix},\\
\mathbf{H}=\begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,5^M}\\ h_{2,1} & h_{2,2} & \cdots & h_{2,5^M}\\ \vdots & \vdots & \ddots & \vdots\\ h_{5^M,1} & h_{5^M,2} & \cdots & h_{5^M,5^M}\end{bmatrix}.
\end{gathered}\tag{E.1}$$

According to the property of Kronecker product, we have $\mathbf{T}^\infty=\mathbf{G}^\infty\otimes\mathbf{H}^\infty$, in which

$$\begin{gathered}
\mathbf{G}^\infty=\begin{bmatrix} g_1 & g_2 & \cdots & g_{N_C}\\ g_1 & g_2 & \cdots & g_{N_C}\\ \vdots & \vdots & \ddots & \vdots\\ g_1 & g_2 & \cdots & g_{N_C}\end{bmatrix},\quad \text{where } \sum_{\alpha=1}^{N_C}g_\alpha=1;\\
\mathbf{H}^\infty=\begin{bmatrix} h_1 & h_2 & \cdots & h_{5^M}\\ h_1 & h_2 & \cdots & h_{5^M}\\ \vdots & \vdots & \ddots & \vdots\\ h_1 & h_2 & \cdots & h_{5^M}\end{bmatrix},\quad \text{where } \sum_{\beta=1}^{5^M}h_\beta=1.
\end{gathered}\tag{E.2}$$

Now, (17) can be written as

$$\begin{bmatrix} \pi_1 & \pi_2 & \cdots & \pi_{N_S}\\ \pi_1 & \pi_2 & \cdots & \pi_{N_S}\\ \vdots & \vdots & \ddots & \vdots\\ \pi_1 & \pi_2 & \cdots & \pi_{N_S}\end{bmatrix}=\begin{bmatrix} g_1 & g_2 & \cdots & g_{N_C}\\ g_1 & g_2 & \cdots & g_{N_C}\\ \vdots & \vdots & \ddots & \vdots\\ g_1 & g_2 & \cdots & g_{N_C}\end{bmatrix}\otimes\begin{bmatrix} h_1 & h_2 & \cdots & h_{5^M}\\ h_1 & h_2 & \cdots & h_{5^M}\\ \vdots & \vdots & \ddots & \vdots\\ h_1 & h_2 & \cdots & h_{5^M}\end{bmatrix}. \tag{E.3}$$

Substituting (E.3) into (19), we have

$$\begin{aligned}
N_i(S)&=\sum_{j=1}^{N_S}\pi_j z_{i,j}=\sum_{\alpha=1}^{N_C}\sum_{\beta=1}^{5^M}\left(g_\alpha h_\beta z_{i,\beta}\right)\\
&=\sum_{\alpha=1}^{N_C}g_\alpha\left(\sum_{\beta=1}^{5^M}h_\beta z_{i,\beta}\right)=\sum_{\beta=1}^{5^M}h_\beta z_{i,\beta}\triangleq N_B.
\end{aligned}\tag{E.4}$$

Substituting (E.3) into (20), we have

$$\begin{aligned}
p_{B,i}(S) &= \sum_{1\leq j'\leq N_S,\, z_{i,j'}=2,\, i\notin A} \pi_{j'} + \sum_{1\leq j'\leq N_S,\, z_{i,j'}=2,\, i\in A} \pi_{j'}(1-P_S)\\
&= \sum_{\alpha=1}^{N_C}\sum_{\beta=1,\, i\notin A}^{5^M} g_\alpha h_\beta \delta(z_{i,\beta}-2)\\
&\quad + \sum_{\alpha=1}^{N_C}\sum_{\beta=1,\, i\in A}^{5^M} g_\alpha h_\beta \delta(z_{i,\beta}-2)(1-P_S)\\
&= \sum_{\alpha=1}^{N_C} g_\alpha \Bigg(\sum_{\beta=1,\, i\notin A}^{5^M} h_\beta \delta(z_{i,\beta}-2)\\
&\qquad + \sum_{\beta=1,\, i\in A}^{5^M} h_\beta \delta(z_{i,\beta}-2)(1-P_S)\Bigg)\\
&= \sum_{\beta=1,\, i\notin A}^{5^M} h_\beta \delta(z_{i,\beta}-2) + \sum_{\beta=1,\, i\in A}^{5^M} h_\beta \delta(z_{i,\beta}-2)(1-P_S)\\
&\triangleq P_B.
\end{aligned} \tag{E.5}$$

Substituting (E.5) into (21), we have

$$\lambda_i(S) = p_i(1-p_{B,i}(S)) = p(1-P_B) \triangleq \Lambda. \tag{E.6}$$

Note that the $z_{i,j}$ in (E.4) and (E.5) is replaced with $z_{i,\beta}$, because it is not related with g_α. Substituting (E.4) and (E.6) into (18), we have

$$D_i(S) = N_i(S)/\lambda_i(S) = N_B/\Lambda \triangleq D. \tag{E.7}$$

The above derivations prove the throughput (E.6), mean delay (E.7), and blocking probability (E.5) of the system with equal packet generating probability are the functions of packet generating probabilities, but independent of S.

ACKNOWLEDGMENTS

This work is sponsored by the National Science Council of Taiwan under grant NSC 95-2752-E-002-009 and 96-2628-E-009-003-MY3, by the Ministry of Education of Taiwan under the MoE ATU Program, and by MediaTek research center at National Chiao Tung University, Taiwan.

REFERENCES

[1] L. Tong, Q. Zhao, and G. Mergen, "Multipacket reception in random access wireless networks: from signal processing to optimal medium access control," *IEEE Communications Magazine*, vol. 39, no. 11, pp. 108–112, 2001.

[2] M. Sidi and I. Cidon, "Splitting protocols in presence of capture," *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 295–301, 1985.

[3] I. M. I. Habbab, M. Kavehrad, and C.-E. W. Sundberg, "ALOHA with capture over slow and fast fading radio channels with coding and diversity," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 1, pp. 79–88, 1989.

[4] M. Zorzi, "Mobile radio slotted ALOHA with capture and diversity," *Wireless Networks*, vol. 1, no. 2, pp. 227–239, 1995.

[5] M. Zorzi and R. R. Rao, "Capture and retransmission control in mobile radio," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 8, pp. 1289–1298, 1994.

[6] S. Ghez, S. Verdu, and S. C. Schwartz, "Optimal decentralized control in the random access multipacket channel," *IEEE Transactions on Automatic Control*, vol. 34, no. 11, pp. 1153–1163, 1989.

[7] Q. Zhao and L. Tong, "A multiqueue service room MAC protocol for wireless networks with multipacket reception," *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 125–137, 2003.

[8] Q. Zhao and L. Tong, "A dynamic queue protocol for multiaccess wireless networks with multipacket reception," *IEEE Transactions on Wireless Communications*, vol. 3, no. 6, pp. 2221–2231, 2004.

[9] M. Realp and A. I. Pérez-Neira, "PHY-MAC dialogue with multi-packet reception," in *ETSI Workshop on Broadband Wireless Ad-Hoc Networks and Services*, Sophia Antipolis, France, September 2002.

[10] R.-H. Gau and K.-M. Chen, "Predictive multicast polling for wireless networks with multipacket reception and queuing," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 725–737, 2006.

[11] X. Wang and J. K. Tugnait, "A bit-map-assisted dynamic queue protocol for multiaccess wireless networks with multiple packet reception," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2068–2081, 2003.

[12] ANSI/IEEE Standard 802.11b–1999 (R2003), "Wireless lan medium access control (MAC) and physical layer (PHY) specifications: higher-speed physical layer extension in the 2.4ghz band," IEEE SA Standards Board, Tech. Rep., 1999.

[13] M. X. Cheng, Y. Li, and D.-Z. Du, Eds., *Combinatorial Optimization in Communication Networks*, Springer, New York, NY, USA, 2006.

[14] H. M. Deitel, *An Introduction to Operating Systems*, chapter 6, Addison-Wesley, Reading, Mass, USA, 1984.

[15] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, NY, USA, 2nd edition, 1984.

[16] L. Kleinrock, *Queueing Systems. Volume 1: Theory*, John Wiley & Sons, New York, NY, USA, 1975.

[17] S. Adireddy and L. Tong, "Exploiting decentralized channel state information for random access," *IEEE Transactions on Information Theory*, vol. 51, no. 2, pp. 537–561, 2005.