A Novel Algebraic Framework for Analyzing Finite Population DS/SS Slotted ALOHA Wireless Network Systems with Delay Capture

Mun Geon Kyeong

CONTENTS

ABSTRACT

A new analytic framework based on a linear algebra approach is proposed for examining the performance of a direct sequence spread spectrum (DS/SS) slotted ALOHA wireless communication network systems with delay capture. The discrete-time Markov chain model has been introduced to account for the effect of randomized time of arrival (TOA) at the central receiver and determine the evolution of the finite population network performance in a single-hop environment. The proposed linear algebra approach applied to the given Markov problem requires only computing the eigenvector Π of the state transition matrix and then normalizing it to have the sum of its entries equal to 1. MATLAB computation results show that systems employing discrete TOA randomization and delay capture significantly improves throughput-delay performance and the employed analysis approach is quite easily and staightforwardly applicable to the current analysis problem.

I. INTRODUCTION

During last decade, extensive analysis of single-hop packet-switched spread spectrum (SS) random access systems has been introduced in the literature. Most of the analyses thus far have focused on the physical-level performance in terms of the probability of bit error, with only a few papers dealing with medium access control (MAC)-level throughput and delay analyses [1], [2], [3]. In particular, the MAC-level analysis of SS random access systems with delay capture is very rarely found in the literature.

Several researchers have considered the MAC-level performance analysis for some SS multiple access packet radio networks of interest in this paper. Davis and Gronemeyer [1] analyzed a DS/SS slotted ALOHA packet radio system with randomized TOA and delay capture. With the assumption that a perfect error correcting code capability is available at the receiver, they focused on the effect of capture on the network performance in terms of throughput, delay and stability. The slotted ALOHA CDMA system was analyzed by Raychaudhuri [2]. Assuming each arriving packet is always captured with probability one, he emphasized the effect of multiple access coding techniques on the throughput-delay-stability performance. Polydoros and Silvester [3] proposed an analytical framework for the study of single-hop slotted random access SS packet radio networks with various network topologies and channel conditions; their theory is characterized by the identification of a set of probabilistic parameters which serve to efficiently summarize the effect of various network considerations on the MAC-level performance.

In general, to determine the performance of the postulated system, the statistics of throughput and delay are used as figures of merit that characterize system performance. System performance can be evaluated by applying the following three standard analysis approaches:

1. Finite user population Markov chain analysis;

2. Equilibrium contour analysis based on the limiting throughput of an infinite user population;

3. Infinite population $S-G$ analysis, which provides a measure of maximum achievable throughput (channel capacity).

For an infinite user population, we can look into the dynamic behavior of the postulated system from twin viewpoints of equilibrium contour and $S-G$ (throughput versus offered traffic) analyses. In [1], an equilibrium contour analysis has been used for the infinite user population, leading to the limiting output rate (throughput) equation.

Equilibrium contour analysis was first used by Kleinrock and Lam in evaluating the performance of slotted ALOHA multiple access system [4]. This analysis is a fluid-type approximation which applies only to steady state. It assumes that the system is always at an equilibrium point, where the equilibrium point is defined as a point n at which the new packet input rate $S_{\text{in}}(n)$ is equal to the system throughput rate $S_{\text{out}}(n)$. Therefore, in equilibrium contour analysis, it is not necessary to calculate the state transition probabilities $P_{i,j}$, which are essential in the finite population Markov chain analysis.

In equilibrium contour analysis, the globally stable equilibrium point on the load line, denoted by (n_o, S_o), is determined at the intersection of the contour and the load line. This equilibrium point is called the "channel operating point" [4]. Kleinrock and Lam showed that the steady-state throughput-delay performance of a stable channel is closely approximated by its globally stable equilibrium point; that is, $\overline{S}$ and $\overline{N}$ for a stable channel are closely approximated by the equilibrium throughput and delay S_o and n_o at the channel operating point. We will verify this claim through a numerical example.

$S-G$ analysis easily delivers the means to evaluate the performance of the given capture-type slotted ALOHA system in terms of the maximum achievable throughput or channel capacity.

A stability analysis can be performed using a load line approach in an equilibrium contour analysis. The $S-G$ analysis cannot deal with the stability problem because the offered traffic (channel traffic)[1] is modeled as a homogeneous Poisson process. Equilibrium contour analysis is based on a Poisson assumption. However, only the new packet arrival distribution is approximated by this Poisson process for an infinite population. Compared to the equilibrium contour analysis, less computation is required in the $S-G$ analysis.

[1]The offered traffic, often called channel traffic, is referred to as the sum of new packet transmissions and retransmissions in a channel [7].

The major disadvantage of the Markov chain analysis is state space explosion, which limits the size of the network that can be analyzed.

In the Markov model, the new packet input rate $S_{\text{in}}(n)$ in state n is expressed by

$$S_{\text{in}}(n) = (M-n)q_a, \tag{1}$$

where q_a denotes the new packet transmission probability and $0 \le n \le M$, depending on the state of the system.

In this paper, we reconsider the classical Davis and Gronemeyer's work [1] to extend their previous published results into a finite population Markov chain analysis, where the linear algebra approach based on eigenvector-eigenvalue problem is employed to describe the behavior of the postulated DS/SS slotted ALOHA wireless random access network with delay capture. However, this paper majorly concerns about demonstrating the viability of the employed analysis approach, rather than showing the performance itself of a given specific technique.

II. SYSTEM DESCRIPTION

Consider the slotted ALOHA random access system shown in Fig. 1 [5]-[7], in which the ALOHA system consists of M users contending to occupy a single common channel.

Fig. 1 shows that each user is in either the *transmission* (TR) mode or the *retransmission* (RT) mode, depending on whether or not he suffered a channel collision. A user in either the TR or RT mode will enter the RT mode if the packet transmission is not successful, while he will enter the TR mode if the packet is successful. In the TR mode, a user generates and transmits a new packet in a slot with probability q_a. The transmission mode is also referred to the "thinking" or "origination" mode. A user who has suffered a channel collision and is waiting for retransmission is said to be in the RT mode. A user in the RT mode is often said to be "backlogged," and a backlogged user is blocked in the sense that he is inhibited from generating and transmitting a new packet [8], [9]. He simply retransmits a packet in each successive slot with probability q_r until attaining a successful packet transmission; that is, the retransmission delay is geometrically distributed with mean $1/q_r$. Since q_r is the retransmission rate for the input traffic, we note that $0 < q_r < 1$. Similarly, we have $0 < q_a < 1$ for the transmission rate q_a, where $q_r > q_a$ is usually assumed for cases of practical interest. For computer networks, the traffic from individual users is typically characterized by a bursty behavior, that is, a high peak-to-average traffic rate. Because of this characteristic, $q_a \ll 1$ is usually assumed in such a system. For simplicity in modeling the system, it is assumed that acknowledgements are perfect and instantaneous. Zero propagation delay is also assumed for packet transmission. The symbol n in Fig. 1 indicates the backlog number, which describes the state of the system. Thus, n will form the discrete state space consisting of a set of integers, $\{0, 1, 2, \cdots, M\}$.

For the derivation of a tractable solution to the proposed system, a number of modeling assumptions and approximations are necessary.

Basically we assume that all transmitters use a *common* spreading code and that suitable spreading code properties allow for the successful reception of time-overlapping packets. We also assume that the processing gain of the system is high enough to mitigate interference effects and that the repetition period of the code is chosen sufficiently large so as to assure that two packets using the same spreading code are quasi-orthogonal if they arrive with a time offset of at least T_{chip}.

With respect to acknowledgements, a perfect and instantaneous reception of acknowledgements is assumed. It is assumed that collisions are detected by time-out due to the lack of a positive acknowledgement. Zero propagation delay is also assumed.

Perfect synchronization is assumed to be available in slotting. Once a packet is captured, we assume the receiver remains locked onto the packet until the end of reception, regardless of whether or not bit errors occur. Also, all transmitted signals are assumed to have equal received power at the receiver sys-

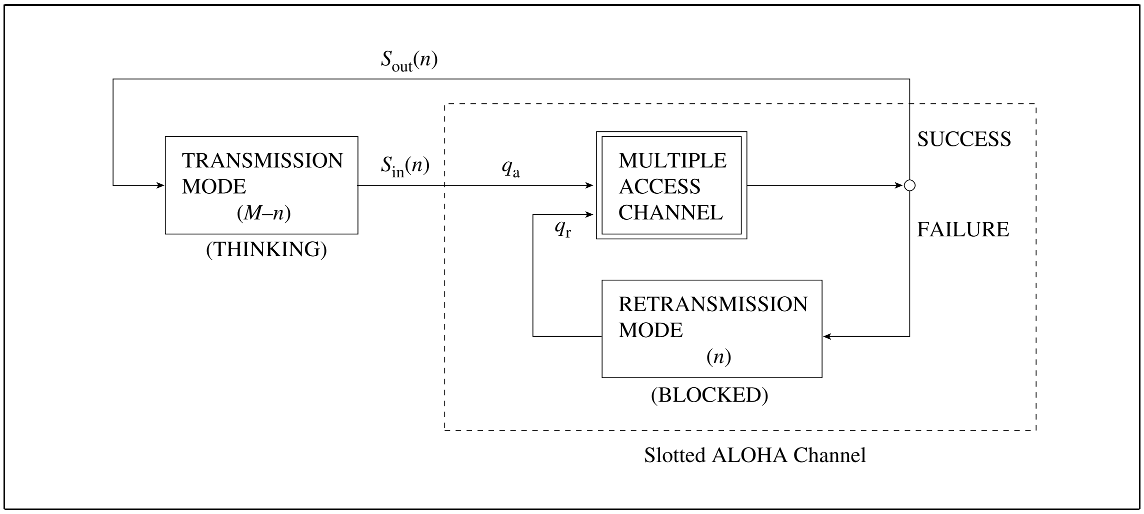

Fig. 1. A model of the slotted ALOHA random access system.

tem. Thus, consideration of the near-far problem is eliminated from the performance analysis.

To make packet captures possible in the presence of collisions, we must specify the design of the packet, the slot and the capture mechanism.

1. Packet Design

In our analysis, we assume that a single packet is comprised of a special preamble code and an address/message portion. In addition, we assume that every packet preamble employs an identical pseudo-noise (PN) code (for more detailed examples, refer to [10]). This preamble code is used for both packet detection and synchronization. For packet detection and synchronization, a matched filter is used at the receiver system [10], [11]. The advantage in using a matched filter lies in the fact that there is no need to search in time for the incoming signal [12]. Since the matched filter will respond whenever the target signal arrives at the receiver, the locations and magnitudes of the peaks in the matched filter output indicate the exact arrival times and signal strengths of all packets that are received during that time slot. The data portion in fact includes an encoded address part as well as an information-bearing part.

2. Slot Design

To support delay capture, the length of the slot, T_s, is extended in order to allow some additional time for the randomization of packet transmission times. Thus, the actual packet time T_p should be made smaller than the slot width by an amount T_u; that is,

$$T_s = T_u + T_p. \qquad (2)$$

In each slot, packet transmission times from each user are then uniformly randomized over the T_u interval. This randomization causes packet arrival times to differ slightly and allows the receiver system to detect and capture the desired packet based on the autocorrelation properties of the preamble PN code.

3. Randomized TOA

In a slotted ALOHA system, all users are synchronized by means of a common clock and are allowed to transmit a packet only at the beginning of a slot. Therefore, the propagation delay will cause the contending packets in a given slot to arrive at the intended receiver in order of increasing link range; the transmitters at shorter distances from the receiver will have a higher chance to eventually deliver their packets to the receiver. Thus, as Abramson has indicated in [13], the transmitters beyond a certain critical distance from the receiver may have no throughput. This range effect is unfair to the transmitters located far from the receiver.

To make the reception of packets fair at the receiver, transmitters must incorporate randomized transmission delays within the slot. In a system with delay capture, the use of this randomization procedure can eliminate the discrimination against the more distant transmitters. If the randomized transmission procedure is implemented at each transmitter, the central receiver at the destination will see the incoming packets with TOAs randomized over the interval $(0, T_u)$.

4. Packet Capture Mechanism

The TOA differences can be used to allow capture of the first arriving packet in a slot. A system which uses DS/SS signaling exhibits the delay capture effect because of the high peak-to-sidelobe autocorrelation property of a PN sequence. With the assumption that the PN sequence does not repeat its pattern within a packet duration, then two packets which use the same sequence and arrive at the same time will be highly correlated over the entire packet duration; whereas, packets arriving with at least chip time offset will demonstrate the quasi-orthogonality of the PN code. As the number of chips in the PN sequence gets larger, the autocorrelation function $R(\tau)$ of the PN code becomes smaller for $|\tau| > 0$. Thus, packets arriving with time offsets are nearly orthogonal and thus do not induce large mutual interference. A minimum time to resolve two packets as separate arrivals will be necessary, however, the orthogonality between packet arrivals depends on the autocorrelation function of the PN code. This minimum time necessary between arrivals is called the "*capture time*," which we denote T_c.

To determine the presence or absence of a signal, a threshold detector located at the output of the matched filter is used. When the detector is triggered by the threshold crossing, the decoder is enabled to lock onto the current

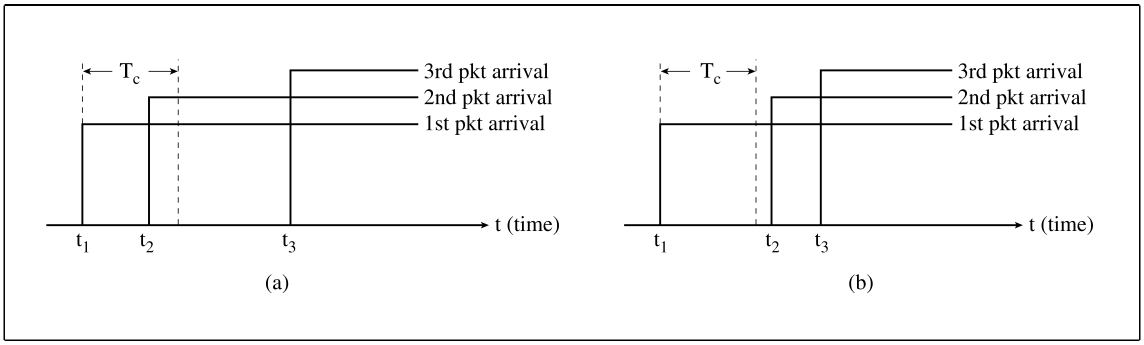

Fig. 2. Illustration of the packet capture mechanism: (a) three packet arrivals, no success; (b) three packet arrivals, first attempt packet captured.

packet. Now we illustrate the concept whereby the first arriving packet will be captured. As an example, consider first the situation shown in Fig. 2(a) in which there are three arriving packets in a slot. The first two packets arrive less than T_c seconds apart, while the third one arrives more than T_c seconds after the second packet. In this case, the output of the matched filter will give three spikes. The receiver will respond to only the first spike and try to decode the first arriving packet. However, the first packet will not be successfully decoded because of the failure to satisfy the minimum requirement for a separation of T_c between the first two arriving packets. The second and third packets will not be captured because the receiver, upon detecting the spike of the first packet, has already switched to the long acquisition and tracking mode and therefore cannot lock onto any other later arriving packets, even if the first packet is not successfully captured. Nevertheless, it is possible for a packet other than the first to capture the receiver. Additional captures may occur due to radio propagation characteristics such as Rayleigh fading that results in power capture at the receiver. Additional captures contribute to the total capture probability. But in [1] the authors assume that only the first packet is eligible for capture. As a result, their equation for the capture probability acts as a lower bound on the true capture probability.

Fig. 2(b) shows the case where the receiver successfully receives the first arriving packet. The receiver will capture the first packet once it detects the first spike. In this case, the first packet will be captured because all subsequent packet arrival times are sufficiently separated from the first packet. For the same reason as in Fig. 2(a), the second and third packets cannot be successfully captured. Accordingly, subsequent spikes will be neglected and only the first packet is captured.

We have now defined the necessary modeling assumptions and approximations, and described how the DS/SS slotted ALOHA receiver system under consideration works. Next, we determine the capture probability, which will later be incorporated into the performance analysis.

III. CAPTURE PROBABILITY

In order to determine the performance of a capture channel, it is necessary to probabilistically describe the capture phenomenon. The capture probability, denoted by C_n, for the capture model considered is

$$C_n = \begin{cases} 0, & n=0; \\ 1, & n=1; \\ (1-Q)^n, & n \geq 2, \end{cases} \quad (3)$$

where Q is the capture ratio, defined by T_c/T_u, and n denotes the number of arriving packets in the given slot at the receiver system.

Equation (3) was previously derived by Davis and Gronemeyer [1]. The derivation method that they used is a random variable approach based on the distribution of TOA occupation. For n packets arriving in a slot with different arrival times, TOA_i, $1 \leq i \leq n$. determining the distribution of the random variable $W = Y - X$, C_n can be defined as the probability that W exceeds T_c. Fig. 3 illustrates the relationships among the variables.

Therefore, given the probability density function of W, $p_W(w)$, the capture probability for $n \geq 2$ is determined as $C_n = \Pr\{W > T_c\} = \int_{T_c}^{T_u} p_W(w)dw$.

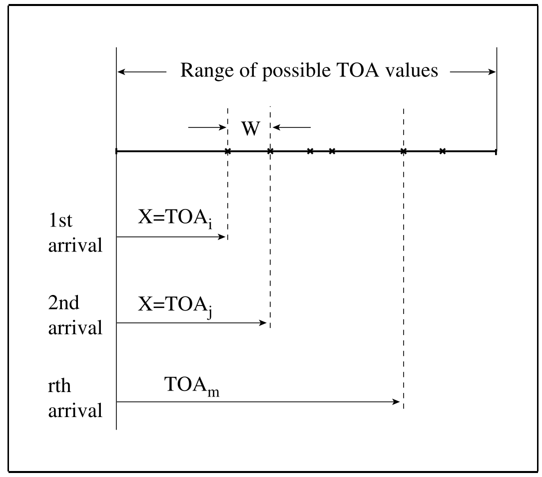

Fig. 3. TOA distribution.

IV. FINITE POPULATION THROUGHPUT-DELAY ANALYSIS

In this section, we describe the discrete-time Markov chain model for a DS/SS, slotted ALOHA system with randomized TOA and delay capture developed by Davis and Gronemeyer [1]. This Markov chain model accounts for the effect of randomized TOA at the central receiver and determines the evolution of the network performance. We first define the probability of success to approximate a slotted ALOHA channel. Subsequently, an analysis of the Markov chain model for the system with delay capture is presented.

1. Success Probability

It is notable that packet capture alone does not guarantee that the packet will be received successfully. Given that a packet is captured, its final destiny depends on the number of other packets present in the slot and the modulation/SS signaling format employed. In fact, the ultimate success probability, Pr{Success}, must be accounted for in the factorization of Pr{Success} = Pr{Capture} Pr{Retain|Capture}. In a real system, an interfering packet can prevent completion of the reception of the captured packet if the system interference margin is exceeded. A DS/SS system can reject later packets as noise up to a certain power level. Davis and Gronemeyer assume, however, that the multi-user threshold condition [14] holds and that a perfect error correcting code capability is available at the decoder [1], [3], i.e., Pr{Retain|Capture} = 1, and thus Pr{Success} = Pr{Capture}. Assume that a packet is reliably captured if it arrives at the receiver T_c seconds before any other packet.

Given that n packets are transmitted in a slot, we now define $P_S(1|n)$ to be the probability that a packet is successfully received:

$$P_S(1|n) = \Pr\{1 \text{ successful packet } | \; n \text{ packets are transmitted}\} = C_n. \quad (4)$$

Correspondingly, we define

$$P_S(0|n) = 1 - P_S(1|n) = 1 - C_n. \quad (5)$$

In practice, the success probability would be affected by the characteristics of the particular DS/SS scheme (i.e., modulation, SS bandwidth, and error control coding). Assuming that a perfect error correcting code capability is available at the receiver, however, we can interchangeably use the two notations, $P_S(1|n)$ and C_n.

Under the assumptions and approximations made thus far, the behavior of the single-hop, DS/SS slotted ALOHA wireless network system can now be described as a Markov model.

2. Markov Chain Analysis

The behavior of the slotted ALOHA system shown in Fig. 1 can now be described by a discrete-time Markov chain whose state transitions occur on a slot-by-slot basis. Let n^t be the number of users in the RT mode, i.e., the number of blocked or backlogged users, at time t. Then, n^t is a discrete-time Markov chain which serves as the state descriptor for the system. Given M users in the network, each of the n^t backlogged users will independently retransmit a packet in each successive slot with probability q_r until that packet is successfully received. Each of $M - n^t$ other users will generate and transmit a new packet in a given slot with probability q_a.

Conditioned on $n^t = i$, we define the following two probabilities:

$$Q_a(k, i) = \Pr\{k \text{ users in TR mode transmit new packets in a slot}\} = \binom{M-i}{k} {q_a}^k (1-q_a)^{M-i-k} \quad (6)$$

and

$$Q_r(k, i) = \Pr\{k \text{ users in RT mode retransmit backlogged packets in a slot}\} = \binom{i}{k} {q_r}^k (1-q_r)^{i-k}. \quad (7)$$

Let S^t be the channel input rate at time t, which is a function of the $M-n^t$ unbacklogged users. Thus, assuming M and q_a to be time-invariant, $S^t = (M-n^t)q_a$, and n^t is a Markov chain with stationary transition probabilities. Then, the one-step steady-state transition probability going from state i to j, $P_{i,j}$, is

$$P_{i,j} = \Pr\{ n^{t+1} = j \mid n^t = i \}, \quad j = 0, 1, \cdots, M, \quad (8)$$

where i and j are the numbers of backlogged users in the successive slots.

Given that the system is in state i at the beginning of a slot, the following four possible state transitions can occur:

1. The state decreases by more than 1. This is not possible because at most one packet can be successfully captured and decoded in a given slot.
2. The state decreases by $i-j=1$. This is possible only when a backlogged packet is successfully transmitted in a given slot.
3. The state does not change, i.e., $i-j=0$.
4. The state increases by $j-i$, $i<j \le M$.

Based on these four cases, the transition probabilities, $P_{i,j}$ are given by

$$P_{i,j} = \begin{cases} 0, & j \le i-2; \\ \Pr\{\text{no TR users transmit}\} & \\ \cdot \Pr\{\text{1 RT user's pkt is captured}\}, & j = i-1; \\ \Pr\{\text{1 TR user transmits}\} & \\ \cdot \Pr\{\text{1 user's pkt is captured}\} & \\ + \Pr\{\text{ no TR users transmit}\} & \\ \cdot \Pr\{\text{ no user's pkt is captured}\}, & j = i; \\ \Pr\{j-i+1 \text{ TR users transmit}\} & \\ \cdot \Pr\{\text{1 user's pkt is captured}\} & \\ + \Pr\{j-i \text{ TR users transmit}\} & \\ \cdot \Pr\{\text{ no user's pkt is captured}\}, & j > i. \end{cases} \quad (9)$$

The corresponding mathematical expressions are then given by

$$P_{i,j} = \begin{cases} 0, & j \le i-2; \\ \sum_{k=1}^{i} Q_a(0,i) Q_r(k,i) P_S(1|k), & j = i-1; \\ \sum_{k=0}^{i} Q_r(k,i)[Q_a(1,i) P_S(1|k+1) & \\ + Q_a(0,i) P_S(0|k)], & j = i; \\ \sum_{k=0}^{i} Q_r(k,i)[Q_a(j-i+1,i) & \\ \cdot P_S(1|k+j-i+1) & \\ + Q_a(j-i,i) P_S(0|k+j-i)], & j \ge i, \end{cases} \quad (10)$$

where $P_{i,j}$ constitutes the (i,j)-th element of the state transition matrix $\quad = [P_{i,j}]$ and

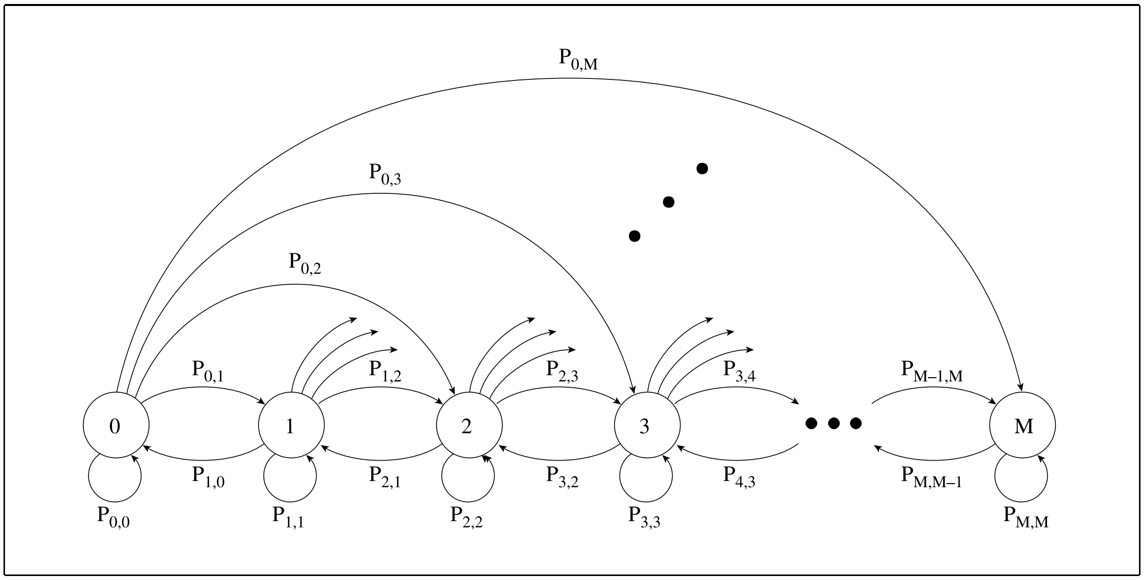

Fig. 4. Markov chain for DS/SS slotted system with delay capture and randomized TOA.

$P_S(m|n)$, $m = 0, 1$ is as previously defined.

The corresponding Markov chain is illustrated in Fig. 4, where the number inside the circle indicates the state, i.e., the size of the backlog. Note that the backlog number can decrease by at most one per transmission, but it can increase by an arbitrary amount. In a DS/SS system with delay capture, one packet can be successfully received even if there is a collision involving two or more packets. Unlike slotted ALOHA, not all packets involved in a collision will become backlogged. The $P_{0,1}$ transition probability accounts for the possibility of single packet capture when the system is initially in a zero backlog state. In standard slotted ALOHA, this state transition is not possible.

For the case of a finite population with M users, the Markov chain in Fig. 4 is irreducible, aperiodic, and ergodic [15], [9]. Thus, a stationary probability distribution vector, $\Pi = [\pi_0, \pi_1, \cdots, \pi_M]$, of n^t exists for which π_n denotes the steady-state probability of being in state n, defined by

$$\pi_n \triangleq \lim_{t\to\infty} \Pr\{n^t = n\}. \qquad (11)$$

The stationary probability distribution vector, Π, can be computed by solving the following $M+1$ linear simultaneous equations,

$$\Pi = \Pi \qquad (12)$$

with the constraint that

$$\sum_{i=0}^{M} \pi_i = 1. \qquad (13)$$

Starting with an arbitrary positive value π_0^*, we may calculate recursively [16]

$$\pi_n^* = \frac{1}{P_{n,n-1}} \left[\pi_{n-1}^* - \sum_{i=0}^{n-1} \pi_i^* P_{i,n-1} \right]. \qquad (14)$$

After normalizing, we get the equilibrium state occupation probabilities given by

$$\pi_n = \frac{\pi_n^*}{\sum_{i=0}^{M} \pi_i^*} \tag{15}$$

to find the Markov chain in state n.

Alternatively, we can also apply linear algebra to determine Π. Since (12) implies that Π is an eigenvector of corresponding to an unity eigenvalue, we need only computing this eigenvector Π and then normalize it to have the sum of its entries equal to 1. We use the second method based on MATLAB computation in our numerical analysis.

Conditioning on $n^t = n$, we can observe the system dynamics by defining the conditional channel throughput to be the average number of successful packets $S_{\text{out}}(n)$, in packets/slot. This channel throughput is the probability of exactly one packet transmission, given that there exist n backlogged users at the beginning of the given slot at time t. Therefore, the conditional throughput can be expressed as

$$\begin{aligned} S_{\text{out}}(n) &= \Pr\{1 \text{ packet is successfully received} \mid \\ &\quad n^t = n\} \\ &= \sum_{l=0}^{M-n} \sum_{k=0}^{n} Q_a(l,n) Q_r(k,n) P_S(1|l+k). \end{aligned} \tag{16}$$

The steady state channel throughput rate is then given by

$$\overline{S} = \sum_{n=0}^{M} S_{\text{out}}(n) \pi_n. \tag{17}$$

Similarly, the expected channel backlog is given by

$$\overline{N} = \sum_{n=0}^{M} n\pi_n. \tag{18}$$

By Little's theorem [17], the average delay or average backlog time is given by

$$\overline{D} = \frac{\overline{N}}{\overline{S}}. \tag{19}$$

V. MATLAB-COMPUTATIONS-BASED NUMBERICAL RESULTS

MATLAB file (.m-file) has been written for computing capture probability (or success probability), binomial coefficient and average channel throughput-delay-backlog, in which equations (3)-(19) were employed.

Based on the equations (16)-(19), Fig. 5 presents typical average performance curves with a small capture ratio and a high retransmission probability for DS/SS slotted ALOHA system, given $Q = 0.01$, $q_r = 0.125$, and $M = \{40, 55\}$. In order to benchmark our results, we have used the same parameter values for the finite population model Davis and Gronemeyer [1] used in the infinite population model. Solid line curves in Fig. 5 represent nominal system operations in which the system supports $M = 40$ users. Assuming that 15 users are unexpectedly added to the system, dotted curves represent the system that supports $M = 55$ users. The two points in Fig. 5 correspond to channel operating points for $M = 40$ and 55, in which $q_a = 0.02$ (refer to Fig. 6 of [1]).

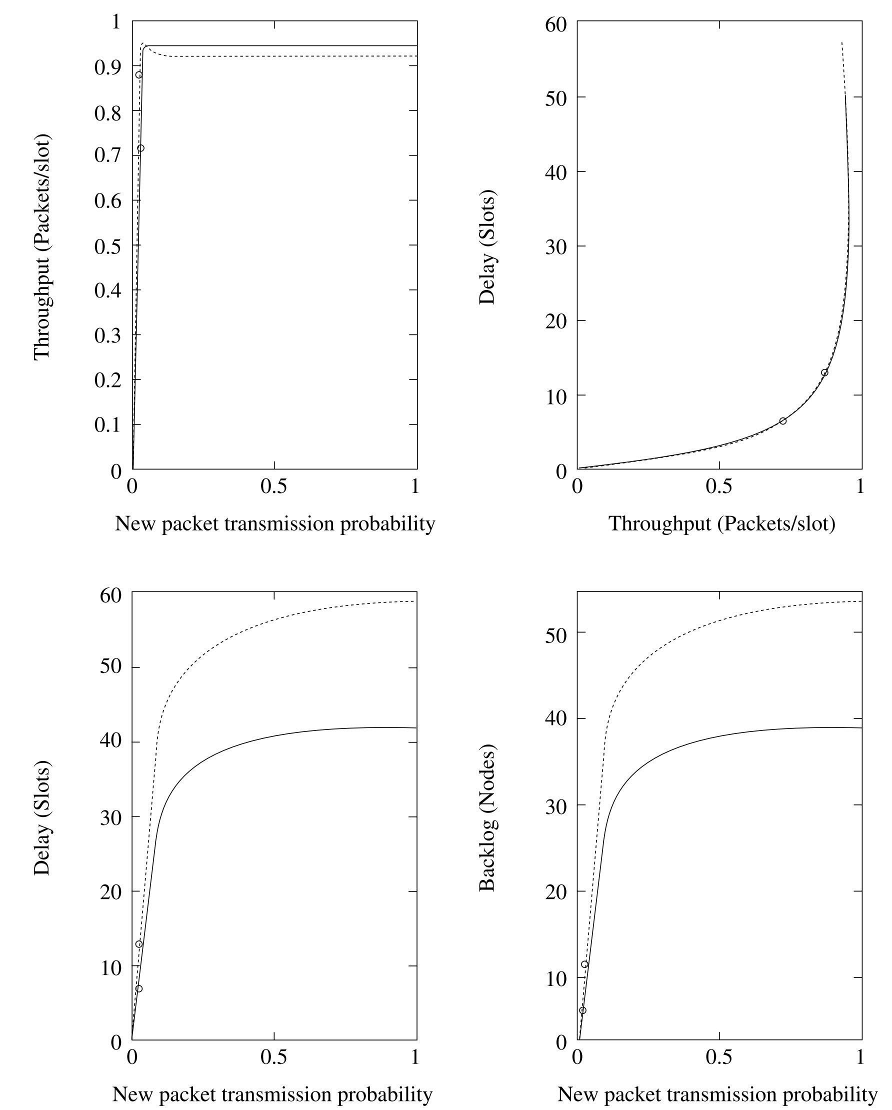

Fig. 5. Average performance for DS/SS slotted ALOHA wireless system employing the first attempt packet capture model. $M=40$ (solid)/$M=55$ (dotted), $Q=0.01$ and $q_r=0.125$: $\overline{S}$ vs. q_a; $\overline{D}$ vs. $\overline{S}$; $\overline{D}$ vs. q_a; $\overline{N}$ vs. q_a.

Under a nominal system operation with $M = 40$ users, the system achieves an average throughput of $\overline{S}=0.712$ packets/slot with an average delay of $\overline{D}=6.203$ slots (average backlog number $\overline{N}=4.415$ nodes) at the channel operating point. In the dotted curves with

$M = 55$ users, the increase in the number of users causes the throughput to increase to $\overline{S} = 0.878$, while the average delay also increases to $\overline{D} = 12.636$ slots, and the average backlog increases to $\overline{N} = 11.096$ nodes. That the average delay $\overline{D}$ doubled due to the additional 15 users means that the system users will experience a slightly larger blocking probability while the extra users remain in the system.

In comparison to the fact that maximum achievable channel throughput for standard slotted ALOHA is just raised to 0.368, it is clear from Fig. 5 that a system with randomized TOA and delay capture significantly improves throughput-delay performance.

As Kleinrock and Lam [4] claimed, we can also easily show that the Markov chain-based calculation for $\overline{S}$ and $\overline{N}$ for a stable channel are closely approximated by the equilibrium contour-based calculation for S_o and n_o at the channel operating point. That is, $(n_o, S_o) \approx (\overline{N}, \overline{S})$.

The application program used for analysis and the computation example for $q_a = 0.125$ are given in Appendix.

VI. SUMMARY

In this paper, we have described and analyzed a single-hop, DS/SS slotted ALOHA wireless communications network system which takes advantage of the pseudo-orthogonality property of a DS/SS waveform and the time capture property of the delay capture mechanism.

For the MAC-level performance analysis, a standard analysis technique, Markov chain analysis has been introduced to investigate the dynamic behavior of the capture-type channel. The MATLAB-computations-based numerical results demonstrated the employed linear algebra analysis approach is viable to the finite population Markov chain problem and also indicated that with some chosen values of the retransmission probability, q_r, and the capture ratio, Q, the system throughput will degrade gracefully while maintaining good delay performance even under heavy load conditions. These attributes can provide significant advantages in supporting a large set of (potentially mobile) users in a ground-based wireless communications environment (especially, in local wireless applications).

APPENDIX

Application MATLAB .m-file and Computational Example for Finite Population Markov Chain Analysis Problem:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ------------------------------------------------------------------------------%
%          <<Application Program for MATLAB                    %
%                Markov Chain Analysis >>                      %
% ------------------------------------------------------------------------------%
%                                                              %
%  Throughput-Delay Calculations versus p for s={0.125}        %
%  (with CAPTURE)                                              %
%                                                              %
%  p=pr=Pr{a terminal retransmits a blocked packet in a        %
%           subsequent slot}                                   %
%  s=sigma=Pr{a terminal transmits a new packet}               %
%  NT = the number of terminals present in the network         %
%  Q = capture ratio                                           %
```

```
%   n(i)t=the number of pkts present in a slot at the i-th      %
%           terminal at time t (0<=n(i)t<=M)                    %
%                                                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear

NT=10;
Q=1/100;

pr=[0.001 0.01 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 0.999];
sigma=[0.125];
m=length(sigma);
mj=length(pr);

Y=[ ];
S=[ ];
D=[ ];
B=[ ];

%–(1st DO LOOP)—
j=0;
while j < m;
j=j+1;

%= = =>
s=sigma(j)

%–(2nd DO LOOP)—
i=0;
while i < mj;
i=i+1;
%= = =>
p=pr(i)

% for the case that j=i :
ii=0;
while ii < NT+1;
ii=ii+1;
im=ii-1;
  k=-1;
  temp1=0;
  temp2=0;
  while k<im;
  k=k+1;
  temp1=temp1+capq(Q,k+1)*bico(im,k)*p^k*(1-p)^(im-k);
  temp2=temp2+(1-capq(Q,k))*bico(im,k)*p^k*(1-p)^(im-k);
  end
P(ii,ii)=(NT-im)*s*(1-s)^(NT-im-1)*temp1+(1-s)^(NT-im)
          *temp2;
end

% for the case that j=i+1 :
ii=0;
while ii < NT;
ii=ii+1;
im=ii-1;
  k=-1;
  temp1=0;
  temp2=0;
  while k<im;
  k=k+1;
  temp1=temp1+capq(Q,k+2)*bico(im,k)*p^k*(1-p)^(im-k);
  temp2=temp2+(1-capq(Q,k+1))*bico(im,k)*p^k*(1-p)
          ^(im-k);
  end
P(ii,ii+1)=bico(NT-im,2)*s^2*(1-s)^(NT-im-2)*temp1
           +(NT-im)*s*(1-s)^(NT-im-1)*temp2;
end

% for the case that j=i-1 :
ii=1;
while ii < NT+1;
ii=ii+1;
im=ii-1;
  k=0;
  temp1=0;
  while k<im;
  k=k+1;
  temp1=temp1+capq(Q,k)*bico(im,k)*p^k*(1-p)^(im-k);
  end
P(ii,ii-1)=(1-s)^(NT-im)*temp1;
end

% for the case that j >= i+2 :
jj=2;
while jj < NT+1;
jj=jj+1;
jm=jj-1;
  ii=0;
  while ii < jj-2;
  ii=ii+1;
  im=ii-1;
  k=-1;
  temp1=0;
  temp2=0;
  while k<im;
  k=k+1;
  temp1=temp1+capq(Q,k+jm-im+1)*bico(im,k)*p^k
          *(1-p)^(im-k);
  temp2=temp2+(1-capq(Q,k+jm-im))*bico(im,k)*p^k
          *(1-p)^(im-k);
  end
P(ii,jj)=bico(NT-im,jm-im+1)*s^(jm-im+1)*(1-s)^(NT-jm-1)
          *temp1+bico(NT-im,jm-im)*s^(jm-im)*(1-s)
          ^(NT-jm)*temp2;
  end
end

% for the case that j <= i-2 :
ii=2;
while ii < NT+1;
ii=ii+1;
  jj=0;
  while jj < ii-2;
  jj=jj+1;
  P(ii,jj)=0;
```

```
  end
end

%= = =>
P
%
[V,E]=eig(P');

%= = =>
V
E
%

ii=0;
while ii < NT+1;
ii=ii+1;
d(ii)=E(ii,ii);
end

%= = =>
d
%

%- - -
%d=diag(E);

%%%%%%%%%%%%%%%%%
%
%   SORT   SORT(X) sorts each column of X in ascending
%          order. [Y,I] = SORT(X) also returns matrix I
%          containing the indexes used in the sort. If X is
%          a vector, Y = X(I). When X is complex, the
%          elements are sorted by ABS(X).
%
%%%%%%%%%%%%%%%%%

[y,z]=sort(d);

%= = =>
y
z
%

ii=0;
c=0;
while ii < NT+1;
ii=ii+1;
c=c+V(ii,z(NT+1));
end

%= = =>
c
V=V/c
%
avg=0;
n=-1;
while n < NT;
n=n+1;
avg=avg+n*V(n+1, z(NT+1));
Y=[Y y(NT+1)];
end

AvgN(i)=avg;
Block(i)=avg/NT;
Sout(i)=(NT-avg)*s;
AvgD(i)=AvgN(i)/Sout(i);
S=[S, Sout(i)];
D=[D, AvgD(i)];
B=[B, Block(i)];

%–(2nd DO LOOP END)—
end

%–(1st DO LOOP END)—
end

end

%%%%%%%%%%%%%%%%%
%
%   capq  capq(Q,n) computes the capture probability given
%         capture ratio Q and the number of packet arrivals, n
%
%%%%%%%%%%%%%%%%%

function yQ = capq(Q,n);

if n<=0,
yQ=0;
elseif n==1,
yQ=1;
else
yQ=(1-Q)^n;
end

end

%%%%%%%%%%
%
%   bico  bico(n,k) computes the value of binomial coefficient.
%
%%%%%%%%%%

function y = bico(n,k);

resd=1.;
resn=1.;
sumlogd=0.;
sumlogn=0.;
loop=n-k;

m=0;
while m<loop;
m=m+1;
i=m-1;
nu=n-i;
```

```
den=loop-i;
resn=resn*nu;
resd=resd*den;
sumlogn=sumlogn+log(resn);
sumlogd=sumlogd+log(resd);
resn=1.;
resd=1.;
end;

if n < k,
y=0;
else
y=exp(sumlogn-sumlogd);
end

end;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                    %
% ————————————————————————————-%
%       << Application Example of MATLAB            %
%                 Computations >>                    %
% ————————————————————————————-%
%                                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

s =
     0.1250

p =
     0.2000

P =
   Columns 1 through 6

    0.6389  0.2368  0.0941  0.0248  0.0047  0.0006
    0.0601  0.6255  0.2176  0.0761  0.0175  0.0028
         0  0.1234  0.6099  0.1946  0.0589  0.0115
         0       0  0.1908  0.5904  0.1682  0.0430
         0       0       0  0.2632  0.5649  0.1389
         0       0       0       0  0.3418  0.5314
         0       0       0       0       0  0.4278
         0       0       0       0       0       0
         0       0       0       0       0       0
         0       0       0       0       0       0
         0       0       0       0       0       0

   Columns 7 through 11

    0.0001  0.0000  0.0000  0.0000  0.0000
    0.0003  0.0000  0.0000  0.0000  0.0000
    0.0015  0.0001  0.0000  0.0000  0.0000
    0.0069  0.0007  0.0000  0.0000  0.0000
    0.0290  0.0037  0.0003  0.0000  0.0000
    0.1076  0.0175  0.0016  0.0001  0.0000
    0.4869  0.0759  0.0088  0.0005  0.0000
    0.5226  0.4282  0.0459  0.0032  0.0001
         0  0.6276  0.3515  0.0203  0.0006
         0       0  0.7446  0.2521  0.0033
         0       0       0  0.8755  0.1245

V =
   Columns 1 through 6

    0.0220   0.1116  -0.2633   0.3729   0.3195  -0.1680
    0.1320   0.4073  -0.4021  -0.1442  -0.6732   0.6122
    0.3583   0.5544   0.1624  -0.7038   0.0051  -0.6983
    0.5712   0.2027   0.6913  -0.0407   0.6076   0.0805
    0.5837  -0.3319   0.3897   0.5230  -0.0195   0.3029
    0.3930  -0.5025  -0.1550   0.2211  -0.2668  -0.0769
    0.1732  -0.3145  -0.2695  -0.1029  -0.0390  -0.0714
    0.0480  -0.1062  -0.1248  -0.0973   0.0465   0.0088
    0.0077  -0.0192  -0.0263  -0.0256   0.0178   0.0089
    0.0006  -0.0016  -0.0024  -0.0025   0.0020   0.0012
    0.0000   0.0000  -0.0001  -0.0001   0.0000   0.0000

   Columns 7 through 11

   -0.0699  -0.0294  -0.0072  -0.0136   0.0042
    0.3499   0.1822   0.0580   0.0987  -0.0373
   -0.6815  -0.4742  -0.2090  -0.3109   0.1482
    0.6073   0.6614   0.4395   0.5556  -0.3494
   -0.1732  -0.5142  -0.5943  -0.6128   0.5409
   -0.0833   0.1970   0.5357   0.4223  -0.5747
    0.0490  -0.0101  -0.3216  -0.1723   0.4243
    0.0066  -0.0152   0.1238   0.0338  -0.2150
   -0.0041   0.0020  -0.0276   0.0000   0.0716
   -0.0007   0.0006   0.0026  -0.0008  -0.0141
    0.0000   0.0000   0.0000   0.0000   0.0013

E =
   Columns 1 through 6

    1.0000       0       0       0       0       0
         0  0.8584       0       0       0       0
         0       0  0.7307       0       0       0
         0       0       0  0.6156       0       0
         0       0       0       0  0.5122       0
         0       0       0       0       0  0.4198
         0       0       0       0       0       0
         0       0       0       0       0       0
         0       0       0       0       0       0
         0       0       0       0       0       0
         0       0       0       0       0       0

   Columns 7 through 11

         0       0       0       0       0
         0       0       0       0       0
         0       0       0       0       0
         0       0       0       0       0
         0       0       0       0       0
         0       0       0       0       0
    0.3378       0       0       0       0
         0  0.2660       0       0       0
         0       0  0.1515       0       0
         0       0       0  0.2040       0
         0       0       0       0  0.1081

d =
   Columns 1 through 6

    1.0000  0.8584  0.7307  0.6156  0.5122  0.4198

   Columns 7 through 11

    0.3378  0.2660  0.1515  0.2040  0.1081
```

y =

Columns 1 through 6

0.1081 0.1515 0.2040 0.2660 0.3378 0.4198

Columns 7 through 11

0.5122 0.6156 0.7307 0.8584 1.0000

z =

11 9 10 8 7 6 5 4 3 2 1

c =

2.2897

V =

Columns 1 through 6

0.0096	0.0487	-0.1150	0.1629	0.1396	-0.0734
0.0576	0.1779	-0.1756	-0.0630	-0.2940	0.2674
0.1565	0.2421	0.0709	-0.3074	0.0022	-0.3050
0.2494	0.0885	0.3019	-0.0178	0.2653	0.0352
0.2549	-0.1450	0.1702	0.2284	-0.0085	0.1323
0.1717	-0.2195	-0.0677	0.0966	-0.1165	-0.0336
0.0756	-0.1373	-0.1177	-0.0449	-0.0170	-0.0312
0.0210	-0.0464	-0.0545	-0.0425	0.0203	0.0039
0.0034	-0.0084	-0.0115	-0.0112	0.0078	0.0039
0.0003	-0.0007	-0.0010	-0.0011	0.0009	0.0005
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Columns 7 through 11

-0.0305	-0.0128	-0.0031	-0.0060	0.0018
0.1528	0.0796	0.0253	0.0431	-0.0163
-0.2977	-0.2071	-0.0913	-0.1358	0.0647
0.2652	0.2889	0.1920	0.2427	-0.1526
-0.0756	-0.2246	-0.2596	-0.2676	0.2363
-0.0364	0.0860	0.2340	0.1844	-0.2510
0.0214	-0.0044	-0.1405	-0.0753	0.1853
0.0029	-0.0066	0.0541	0.0148	-0.0939
-0.0018	0.0009	-0.0121	0.0000	0.0313
-0.0003	0.0002	0.0012	-0.0003	-0.0062
0.0000	0.0000	0.0000	0.0000	0.0005

ACKNOWLEDGMENT

The author wishes to express his sincere gratitude to the reviewers of this paper, Prof. L. Hanzo, Dr. S. Chia, Dr. F. Adachi, Prof. K. D. Letaief, and Dr. L. K. Rasmussen, for their valuable comments and suggestions. Special thank also goes to Dr. E.-H. Lee for his warm and constructive criticism shown for the better presentation of the paper.

REFERENCES

[1] D. H. Davis and S. A. Gronemeyer, "Performance of slotted ALOHA random access with delay capture and randomized time of arrival," *IEEE Trans. Commun*., vol. COM-28, pp. 703-710, May 1980.

[2] D. Raychaudhuri, "Performance analysis of random access packet-switched code division multiple access systems," *IEEE Trans. Commun*., vol. COM-29, pp. 895-901, June 1981.

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

[4] L. Kleinrock and S. S. Lam, "Packet switching in a multiaccess broadcast channel: Performance evaluation," *IEEE Trans. Commun.*, vol. COM-23, pp. 410-422, Apr. 1975.

[5] H. Kobayashi and Y. Onozato, and D. Huynh, "An approximate method for design and analysis of an ALOHA system," *IEEE Trans. Commun.*, vol. COM-25, pp. 148-158, Jan. 1977.

[6] Y. Onozato and S. Noguchi, "On the thrashing cusp in slotted Aloha systems," *IEEE Trans. Commun.*, vol. COM-33, pp. 1171-1182, Nov. 1985.

[7] S. Tasaka, *Performance Analysis of Multiple Access Protocols*. Cambridge, MA: The MIT Press, 1986.

[8] A. B. Carleial and M. E. Hellman, "Bistable behaviour of ALOHA-type systems," *IEEE Trans. Commun.*, vol. COM-23, pp. 401-410, Apr. 1975.

[9] D. Bertsekas and R. Gallager, *Data Network*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[10] R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman, "Advances in packet radio technology," *Proc. IEEE*, vol. 66, pp. 1468-1496, Nov. 1978.

[11] B. Ramamurthi, A. Saleh, and D. J. Goodman, "Perfect-capture ALOHA for local radio communications," *IEEE J. Select. Areas in Commun.*, vol.

SAC-5, pp. 806-814, June 1987.

[12] G. R. Cooper and C. D. McGillem, *Modern Communications and Spread Spectrum*. New York: McGraw-Hill, 1986.

[13] N. Abramson, "The throughput of packet broadcasting channels," *IEEE Trans. Commun.*, vol. COM-25, pp. 117-128, Jan. 1977.

[14] J. A. Silvester, "Performance of spread spectrum networks," *Proc. Allerton Conf.*, pp. 30-39, Oct. 3-5, 1984.

[15] E. Çinlar, *Introduction to Stochastic Processes*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[16] C. Namislo, "ALOHA networks in which collisions do necessarily destroy all packets involved," *Proc. Mediterranean Electrotech. Conf.*, Athens, Greece, pp. A2.09-A2.10, May 1983.

[17] J. D. C. Little, "A proof for the queueing formula: $l = \lambda w$," *Oper. Res.*, vol. 9, pp. 383-387, May 1961.

Mun Geon Kyeong received B.S. and M.S. degrees in electronics engineering from Korea University, Seoul, Korea, in 1980 and 1985, respectively. He received his Ph.D. degree in electrical engineering from Texas A&M University in 1993. From 1982 to 1984, he was an independent researcher at IIT, Chicago, IL under the YOOK YOUNG SOO Memorial Foundation Fellowship. Following a brief period with the OPC Central Research Institute as an assistant manager, he joined ETRI in 1985. He then spent about one and half years (early 1985 - late 1986) with the Radio Communications Development Division and about three years (late 1986 - mid 1989) with TDX Development Division until he was awarded an ETRI Fellowship for pursuing a doctoral degree at The Ohio State University in 1989, and later at Texas A&M University in 1992. His early research was concerned with the design and analysis of new digital communication systems and application of advanced signal processing techniques to new communication system designs. More recently he has been involved with the research on HF-band surveillance system development, development of universal signaling transceiver board assembly for ISDN-type TDX-10 electronic switching system, and IS-95/CDMA digital cellular mobile communications system development, now largely centered on the issues closely related with multiple access technologies, packet radio networks, FPLMTS/IMT-2000 systems, and smart antenna systems. He is a member of KICS, KITE and IEEE Communications Society. He served as Manager for the FPLMTS foundational technologies research project in 1995. Since late 1995, he has been serving as Editor of ITU-R SG 8 TG 8/1 FPLMTS.RSPC-4. He is currently a Principal Member of Research Staff, serving as Manager for the 'vocoder improvement & system research' and the '800 MHz CDMA smart antenna channel sounder development' projects at ETRI.