Performance Analysis for A Stabilized Multi-channel Slotted ALOHA Algorithm

Dongxu Shen, Victor O.K. Li
Dept. of Elec. and Electronic Engineering
Univ. of Hong Kong, Pokfulam Road, Hong Kong,China
{dxshen, vli}@eee.hku.hk

Abstract—We study slotted ALOHA with multiple random access channels, the so called multi-channel ALOHA (MC-ALOHA). It is well known that single-channel ALOHA (SC-ALOHA) is unstable. Not surprisingly, MC-ALOHA is also unstable. A stabilization algorithm for MC-ALOHA has been proposed in [10], in which the pseudo-Bayesian algorithm in SC-ALOHA was extended to achieve stabilized MC-ALOHA. The idea is to estimate the number of attempting users so that user transmission probability can be adjusted accordingly. In this paper, we give a theoretical analysis on the algorithm performance for cases with limited and unlimited number of users by assuming perfect estimate. The theoretical results are validated by simulation, which shows the stabilization algorithm performs close to a system with perfect estimate. The simulation results also show that the performance of the stabilized algorithm is much better than the non-stabilized algorithm. With the stabilized algorithm, the system is always stable when the new packet arrival rate is less than system capacity. Even when the arrival rate is higher than capacity, system throughput can still be guaranteed.

Keywords— Multi-channel ALOHA, multiple access, stabilization, pseudo-Bayesian algorithm.

I. INTRODUCTION

ALOHA [1] is one of the most important algorithms for random access. It is well known that ALOHA is unstable with a single channel. Not surprisingly, it is also unstable with multiple channels [2], [3]. There has been much research on stabilizing the SC-ALOHA. The key is to determine an appropriate (re)transmission probability. Algorithms that estimate user population [5], [6] have been designed to stabilize the network.

In contrast, little has been done for the stability of MC-ALOHA. The stability issue, such as the region for stable operation of MC-ALOHA, was discussed in [2], [3], but unfortunately no stabilization algorithm was proposed. In [4], a backoff algorithm for MC-ALOHA was provided. However, the results in [4] showed the backoff approach was unable to sustain throughput under all network conditions. This is expected since it has been shown in SC-ALOHA that backoff algorithms are not stable [7].

In [10], a stabilized MC-ALOHA algorithm has been proposed by extending the pseudo-Bayesian algorithm [8] originally devised for SC-ALOHA. The idea is also based on the number of contending users, based on which a proper (re)transmission probability can be calculated. In this paper, we analyze the performance of the proposed algorithm. In the analysis, we assume the estimate of the number of users is perfect. Then the system can be modeled as a Markov chain, from which both the average throughput and user delay can be calculated. Such analytical results provide an upper bound on the actual performance of the algorithm. Through simulation, we demonstrate that the analysis matches very well with the simulation

results, which suggests that the analytical approach is accurate and the algorithm is indeed operating close to perfect estimate. The simulation results also show that the proposed algorithm is able to maintain good throughput under all network conditions.

This paper is organized as follows. In Section II, we describe the MC-ALOHA system and the stabilization algorithm. In Section III, we provide an analysis for the stabilization algorithm. In Section IV, we present simulation results. In Section V, we conclude the paper.

II. DESCRIPTION OF STABILIZATION ALGORITHM

A. System Description and Assumptions

We consider a time slotted MC-ALOHA system, where multiple users compete for uplink access to a base station. In such a system, each user always starts transmission at the beginning of a fixed-length time slot. The system has M non-interfering channels and a user should select one channel to transmit at a time. At the end of the slot, the base station delivers the feedback message for each channel. There are three possibilities: success, collision, or channel idleness.

In this paper, we call a user with a stored packet an *attempting user*, and a user with no packet to transmit an *inactive user*. An inactive user becomes an attempting user after the arrival of a new packet, and an attempting user returns to be inactive after a successful transmission.

We have the following assumptions for MC-ALOHA: (1) Each user has only one buffer to store a single packet; (2) When an inactive user has a new packet arrival, the user is treated the same as other existing attempting users; (3) Each attempting user decides whether to transmit in the next time slot with the same probability p_r; (4) The probability of selecting channel n is $1/M$ for $1 \leq n \leq M$; (5) Capture effect is not considered.

B. Pseudo-Bayesian Algorithm for MC-ALOHA

The motivation for stabilizing MC-ALOHA is as follows. Since users select a channel with equal probability, the attempt rate for each channel is 1 when the overall attempt rate is M. Thus it is obvious that the maximal throughput is achieved when the overall attempt rate is M. Therefore, if the total number of attempting users is known as U, system stabilization can be achieved if the transmission probability for every user is chosen adaptively as

$$p_r = \min(1, \frac{M}{U}). \tag{1}$$

Such a control scheme requires the knowledge of U, which is not directly available and can only be estimated. In [10], we extend the estimation approach in the pseudo-Bayesian algorithm that was originally designed for SC-ALOHA.

Let there be C collisions out of the M channels. Let $\widehat{U}_{k-1}$ be the estimate at slot $k - 1$. The estimate at slot k is updated as

$$\widehat{U}_k = \max\{\lambda_a, \widehat{U}_{k-1} + \lambda_a + C \cdot (e - 2)^{-1} - (M - C)\}. \quad (2)$$

In (2), λ_a accounts for the new arrival, while $C \cdot (e - 2)^{-1}$ represents the C collisions, and $M - C$ stands for the $M - C$ successes or idleness. Further, λ_a is the new packet arrival rate, which is usually unknown. Similar to the approach for SC-ALOHA, we fix $\lambda_a = Me^{-1}$. It has been shown in the simulation that such a selection of λ_a always maintains stability.

The transmission probability for each user is then chosen as

$$p_r(k) = \min(1, \frac{M}{\widehat{U}_k}). \quad (3)$$

III. PERFORMANCE ANALYSIS FOR STABILIZED MC-ALOHA

We first analyze the performance of the stabilized algorithm for a finite number of users. Then we study the case for infinite number of users.

A. Finite Users

We assume there are V users. We assume an inactive user has a constant probability p_g to generate a new packet. An inactive user which has generated a new packet will become an attempting user at the start of the next time slot. Let U_k denote the number of attempting users at the beginning of time slot k. Given retransmission probability $p_r(k)$, obviously U_{k+1} only depends on U_k. Therefore, $\{U_k, \ k = 1, 2, \ldots\}$ is a Markov chain given $p_r(k)$. However, $p_r(k)$ is determined by the estimate $\widehat{U}_k$ of U_k. This complicates system performance analysis. To simplify the analysis, we assume the estimate of U_k is perfect, i.e., $\widehat{U}_k = U_k$, $\forall k$. Under this assumption, $\{U_k, \ k = 1, 2, \ldots\}$ can be modeled as a Markov chain with state space $\{0, 1, 2, \ldots, V\}$. The analysis for this Markov chain can serve as a performance upper bound for practical systems with imperfect estimate of U_k.

A.1 State Transition Probability

It is obvious that the Markov chain $\{U_k, \ k = 1, 2, \ldots\}$ is homogeneous, aperiodic, and irreducible. Let $P_{i,j}$ be the transition probability from state i to state j, i.e., $P_{i,j} = Pr\{U_{k+1} = j | U_k = i\}$, $0 \leq i, j \leq V$. Let random variable D_k, $0 \leq D_k \leq \min(M, U_k)$ denote the number of successful transmissions at time slot k. Let A_{k+1}, $0 \leq A_{k+1} \leq V - U_k$ be the number of inactive users having new packet arrivals in time slot k, which means A_{k+1} users will become attempting users at slot $k + 1$.

The state transition of $\{U_k, \ k = 1, 2, \ldots\}$ satisfies

$$U_{k+1} = U_k - D_k + A_{k+1}. \quad (4)$$

Given $D_k = d, U_{k+1} = j, U_k = i$, we have

$$A_{k+1} = j - i + d. \quad (5)$$

It is obvious that when there is no attempting user, there is no departure, i.e., when $i = 0$, then $d = 0$, so that from (5), $U_{k+1} = A_{k+1}$. When there is one attempting user, i.e., $i = 1$, under the assumption of perfect user information, there is $p_r = 1/i = 1$. Thus the user will always transmit and be successful. Then $d = 1$, and accordingly $U_{k+1} = A_{k+1}$. It is easy to get transition probabilities of

$$P_{0,j} = P_{1,j} = \text{bin}(j, V, p_g), \quad (6)$$

where $\text{bin}(j, V, p_g)$ denotes the binomial probability $\text{bin}(j, V, p_g) = \binom{V}{j} p_g^j (1 - p_g)^{V-j}$.

We now consider the case when $i \geq 2$. The state transition probability $P_{i,j}$ can be expressed as the conditional probability on D_k,

$$P_{i,j} = \sum_{d=0}^{\min(M, i)} Pr\{U_{k+1} = j | U_k = i, D_k = d\} \quad (7)$$
$$\cdot Pr\{D_k = d | U_k = i\},$$

in which

$$Pr\{U_{k+1} = j | U_k = i, D_k = d\}$$
$$= \begin{cases} Pr\{A_{k+1} = j - i + d | U_k = i, D_k = d\}, \\ \qquad \text{when } i - d \leq j \leq V - d \\ \\ 0, \qquad \text{other values of } j. \end{cases} \quad (8)$$

Since inactive users have a fixed probability in generating new packets, A_{k+1} is binomial,

$$Pr\{A_{k+1} = a | U_k = i, D_k = d\}$$
$$= Pr\{A_{k+1} = a | U_k = i\} \quad (9)$$
$$= \text{bin}(a, V - i, p_g)$$

where p_g is the new packet arrival probability.

We then need to calculate $Pr\{D_k = d | U_k = i\}$ for (7). Let T_k be the number of transmitting users among U_k attempting users at time slot k. Then $Pr\{D_k = d | U_k = i\}$ can be expressed as the conditional probability of T_k,

$$Pr\{D_k = d | U_k = i\}$$
$$= \sum_{t=d}^{i} Pr\{D_k = d | U_k = i, T_k = t\} \cdot Pr\{T_k = t | U_k = i\}$$
$$= \sum_{t=d}^{i} Pr\{D_k = d | T_k = t\} \cdot Pr\{T_k = t | U_k = i, \}. \quad (10)$$

Obviously T_k is binomial given U_k, i.e.,

$$Pr\{T_k = t | U_k = i\} = \text{bin}(t, i, p_r(k)). \quad (11)$$

From our assumption of perfect estimation on U_k, $p_r(k) = \min(1, M/i)$.

The expression for $Pr\{D_k = d | T_k = t\}$ can be derived from the well-known combinatorial problem of assigning balls to boxes. In that problem, a number of balls are thrown into

boxes. Each box is selected with equal probability. Our corresponding problem is to assign T_k balls into M boxes, and to calculate the probability of having one ball in D_k boxes. This probability can be expressed as [2]

$$Pr\{D_k = d | T_k = t\} = \frac{(-1)^d M! t!}{M^t d!}$$
$$\times \sum_{l=d}^{\min(M, t)} \frac{(-1)^l (M-t)^{t-l}}{(l-d)!(M-l)!(t-l)!} . \quad (12)$$

With $Pr\{T_k = t\}$ and $Pr\{D_k = d | T_k = t\}$ given in (11) and (12), we can calculate $Pr\{D_k = d | U_k = i\}$ from (10).

The state transition probability can finally be obtained from (7) using (8), (9) and (12).

A.2 Performance Evaluation

The steady state probability $\pi = [\pi_0, \pi_1, \pi_2, \ldots, \pi_V]$ can be calculated from the transition matrix $\mathbf{P}$ by

$$\pi = \pi \mathbf{P},$$
$$\sum_{n=0}^{V} \pi_n = 1. \quad (13)$$

The elements of $\mathbf{P}$ are given in (7).

The average number of attempting users is given by

$$\overline{U} = \sum_{n=0}^{V} n \pi_n. \quad (14)$$

The average throughput is

$$\overline{D} = \sum_{n=0}^{M} n Pr\{D_k = n\}. \quad (15)$$

The probability $Pr\{D_k = n\}$ can be calculated as

$$Pr\{D_k = n\} = \sum_{i=n}^{V} Pr\{D_k = n | U_k = i\} \cdot Pr\{U_k = i\}$$
$$= \sum_{i=n}^{V} \pi_i Pr\{D_k = n | U_k = i\}, \quad (16)$$

where $Pr\{D_k = n | U_k = i\}$ is given in (10).

At the end of a time slot, D_k users are successful in transmission, and $B_k = U_k - D_k$ users continue to attempt in the next slot. We call B_k the backlog of slot k. The average backlog can be calculated as

$$\overline{B} = \overline{U} - \overline{D}. \quad (17)$$

B. Infinite Users

When there are infinite number of users, the system can still be modeled as a Markov chain. However, the state space of U_k is in $\{0, 1, 2, \ldots, +\infty\}$. Thus, it is possible for the number of attempting users to be infinite. We assume the distribution for new packet arrivals at inactive users is Poisson with rate λ_n.

To analyze the performance, we first consider the case for a stable system, i.e., the number of attempting users U_k is always limited. Later we show the condition for system stability

is $\lambda_n < Me^{-1}$. Under this condition, we can calculate the transition probability as in the finite user case. The dimension of the transition matrix $\mathbf{P}$ can be chosen as L, with requirement

$$L = \text{argmin}_{L'}\{P_{i,L'} < \epsilon, \ \forall i\}, \quad (18)$$

where ϵ is a very small value. However, the calculation of $Pr\{A_{k+1} = a | U_k = i\}$ is different from that of the finite user case in (9). Since we have Poisson arrival,

$$Pr\{A_{k+1} = a | U_k = i\} = \frac{e^{-\lambda_n} \lambda_n^a}{a!}. \quad (19)$$

Substituting (9) with (19), we can calculate state transition probability as in Section III-A. All performance parameters can be obtained as in Section III-A.2. Since the system is stable, the average throughput is

$$\overline{D} = \lambda_n, \quad when \ \lambda_n < Me^{-1}. \quad (20)$$

We then analyze system throughput when the system is unstable (i.e., U_k goes to infinity). Given $U_k = i$, the number of transmitting users T_k is binomial with

$$Pr\{T_k = t | U_k = i\} = \text{bin}(t, i, p_r). \quad (21)$$

Since we assume retransmission probability p_r is perfectly selected as $p_r = M/i$, equivalently there is $ip_r = M$. As $i \to \infty$, the probability in (21) is Poisson with mean M. The throughput distribution is given as

$$Pr\{D_k = d\} = \sum_{t=d}^{+\infty} Pr\{D_k = d | T_k = t\} \cdot Pr\{T_k = t\}, \quad (22)$$

where $Pr\{T_k = t\} = \frac{M^t e^{-M}}{t!}$. Average throughput can be calculated as

$$\overline{D}_{inf} = \sum_{d=0}^{M} d \cdot Pr\{D = d\}. \quad (23)$$

Substituting the expressions in (12) to (22), we can get

$$\overline{D}_{inf} = Me^{-1}. \quad (24)$$

Thus when there are infinite attempting users, the departure is stabilized at Me^{-1}, which is the maximal rate for M independent ALOHA channels. Thus the condition for system stability is

$$\lambda_n < Me^{-1}. \quad (25)$$

Combining the results in (20) and (24), we get the throughput with infinite users,

$$\overline{D} = \begin{cases} \lambda_n, & when \ \lambda_n < Me^{-1}, \\ Me^{-1}, & when \ \lambda_n \geq Me^{-1}. \end{cases} \quad (26)$$

IV. SIMULATION RESULTS

We consider an MC-ALOHA system that has M channels. Simulation results are obtained from $100,000$ time slots. In the simulation of the stabilization algorithm, we set $\lambda_a = Me^{-1}$ in (2). The throughput is normalized to the total number of channels M. The average backlog is calculated as the average number of attempting users at the end of a time slot.
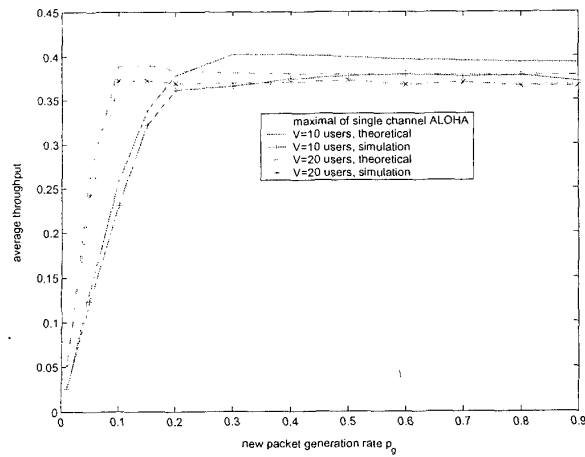
(a)
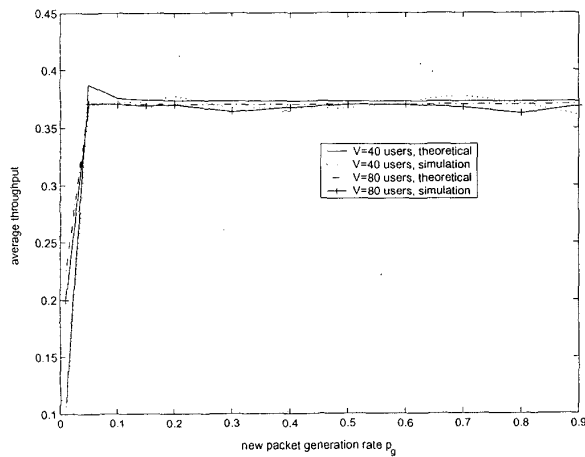
(b)

Fig. 1. Comparison of theoretical and simulation results on throughput. $M = 4$. (a) $V = 10$ and $V = 20$ users; (b) $V = 40$ and $V = 80$ users.

A. Limited Number of Users

In Fig. 1, we show the normalized throughput comparison between analysis and simulation. There are four channels. In Fig. 1 (a), we present results for $V = 10$ and $V = 20$ users. It is clear that theoretical calculation matches well with simulation results. However, the theory predicts higher throughput than simulation. This is because we assume perfect knowledge on the number of attempting users in the theoretical analysis. When there is one user, that user is always successful in transmission. This accounts for the higher throughput predicted by analysis. Further, at high packet generation rate p_g, theoretical throughput is even higher than e^{-1}. This is because the user attempts are binomial instead of Poisson so that the throughput could be a little higher than e^{-1}. In Fig. 1 (b), we present results for $V = 40$ and $V = 80$ users. It is clear that when there
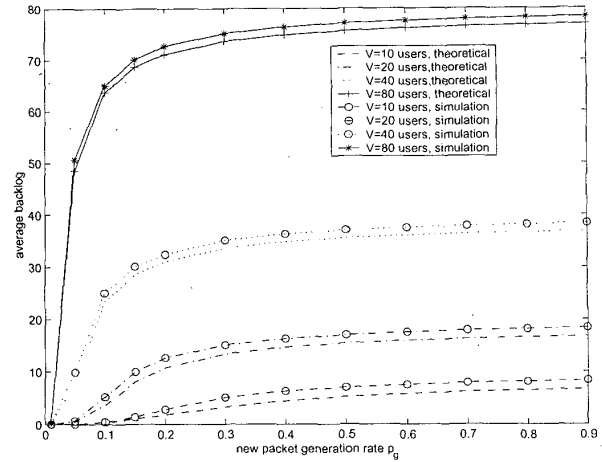

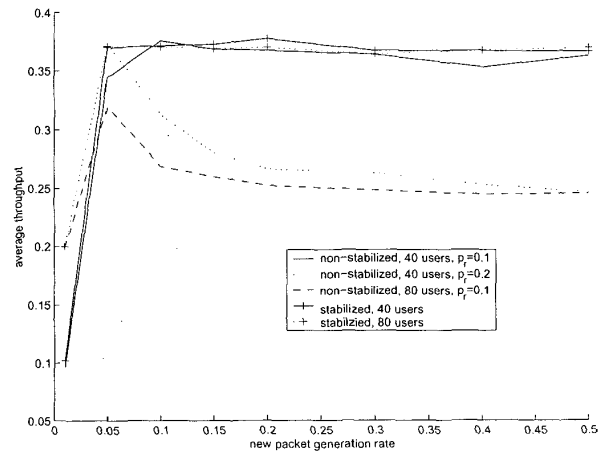Fig. 2. Comparison of theoretical and simulation results on backlog. $M = 4$.

Fig. 3. Comparison of stabilized and non-stabilized ALOHA on throughput with limited number of users. $M = 4$.

are more users, the analytical results become almost identical with the simulation results. With more users, the theory predicts better compared with results in (a). This is because the case of having only one attempting user is rare with more potential users. From both figures, we observe from simulation that the throughput is always stabilized around e^{-1} when p_g is high.

In Fig. 2, we show the results for backlogs. We notice there is little difference between the analytical and simulation results. The backlog from simulation is just a little higher than analysis. This is another indication of the good performance of the stabilization algorithm.

In Fig. 3, we demonstrate the throughput difference between the stabilized and non-stabilized algorithms, with $M = 4$. In the non-stabilized algorithm, a user transmits immediately after generating a new packet, while a backlogged user retransmits with probability p_r. Needlessly to say, the performance of the non-stabilized algorithm depends on the retransmission probability. As shown in Fig. 3, when $V = 40$ and $p_r = 0.1$, the
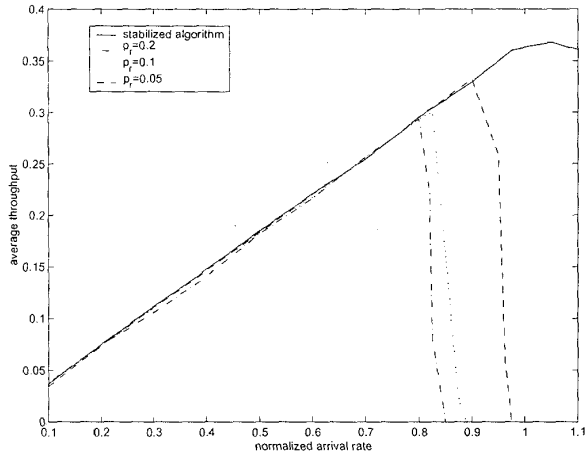
Fig. 4. Comparison of average throughput between stabilized and non-stabilized algorithms.

Fig. 5. Comparison of average backlog between stabilized and non-stabilized algorithms.

non-stabilized algorithm performs as good as the stabilized algorithm, especially when p_g is high. This is because the number of backlogged users approaches 40 when p_g is high. Thus $p_r = 0.1$ leads to approximately 4 transmitting users per slot, which is optimal with $M = 4$. However, the performance degrades when $p_r = 0.2$ with 40 users, or $p_r = 0.1$ with 80 users. Therefore, a preselected retransmission probability is unable to adapt to the variation of user parameters. On the other hand, the stabilized algorithm always maintains throughput at the maximal level regardless of user parameters.

B. Unlimited Number of Users

We assume the number of users is infinite, and the distribution for new packet arrival is Poisson with a fixed rate λ. The rate λ is normalized by Me^{-1}, which is the system capacity limit.

We compare the performance of the stabilized MC-ALOHA with that of the non-stabilized MC-ALOHA. In the non-stabilized MC-ALOHA, a new user always transmits immediately, while a backlogged user transmits with a fixed probability of p_r. In Fig. 4, we plot the average throughput for $M = 4$, $p_r = 0.2, p_r = 0.1, p_r = 0.05$, and the stabilized algorithm.

The non-stabilized algorithms exhibit similar throughput as the stabilized algorithm when the arrival rate is low. However, the throughput of the non-stabilized algorithm degrades dramatically when the arrival rate is higher than a threshold. This threshold depends on the retransmission probability p_r. From our experiment, the system is very likely to be unstable when $\lambda > 0.8$ for $p_r = 0.2$, $\lambda > 0.85$ for $p_r = 0.1$, and $\lambda > 0.9$ for $p_r = 0.05$. Thus, a lower p_r leads to a more stable system. However, even with a small λ, stability is not guaranteed by the non-stabilized algorithms.

In Fig. 5, we show the average backlog for the above four cases. It is clear that the stabilized algorithm is the best. When p_r is chosen to be very small, the non-stabilized algorithm can maintain stability at high arrival rate. For example, when $p_r = 0.05$, the maximal λ can be up to 0.9, as shown in Fig. 4. However, the average backlog is much higher compared to other
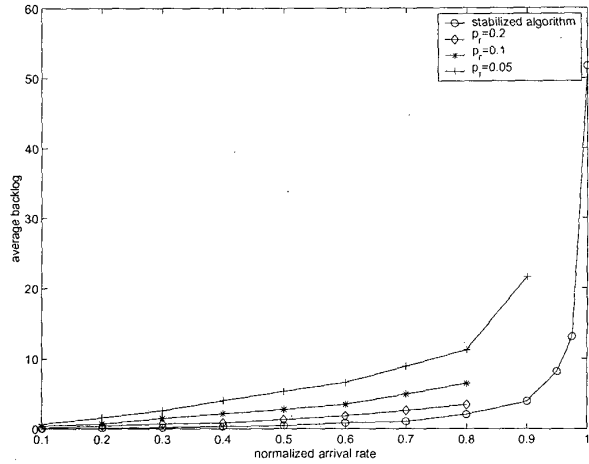
values of p_r. Therefore, stability and backlog optimization are two conflicting factors for the non-stabilized algorithm. In contrast, the two factors can be jointly optimized with the stabilized MC-ALOHA.

V. CONCLUSION

In this paper, we analyze the performance of a stabilization algorithm for MC-ALOHA, which is designed to adjust user transmission probability based on the estimated number of attempting users. The simulation results match the theoretical analysis very well, which proves that the stabilization algorithm operates close to a perfect estimation of the number of users. In the simulation, we also show that the performance of the stabilized algorithm is much better than the non-stabilized algorithm.

REFERENCES

[1] N. Abramson, "The ALOHA system–another alternative for computer communications," *Proc. Fall Joint Comput. Conf., AFIPS Conf.*, pp. 37.
[2] W. Szpankowski, "Packet switching in multiple radio channels: Analysis and stability of a random access system," *Comput. Netw.*, vol. 7, pp. 17-26, 1983.
[3] I. E. Pountourakis, E. D. Sykas, "Analysis, stability and optimization of ALOHA-type protocols for multi-channel networks," *Comput. Commun.*, vol. 15, no. 10, pp. 621-629, Dec. 1992.
[4] M. S. Alam, A. E. Hossain, "Throughput analysis of a multichannel slotted-ALOHA protocol in short-haul communication environment for an exponential backoff retransmission scheme," *Proc. International Conf. on Information, Commun. and Signal Processing (ICICS) '97*, Singapore, 1997, pp. 1034-1038.
[5] B. Hajek, T. Van Loon, "Decentralized dynamic control of a multiaccess broadcast channel," *IEEE Trans. Automat. Contr.*, vol. 27, pp. 559-569, June 1982.
[6] L. Merakos, D. Kazakos, "On retransmission control policies in multiple-access communications networks," *IEEE. Trans. Automat. Contr.*, vol. 30, pp. 109-117, Feb. 1985.
[7] D. Raychaudhuri, K. Joseph, "Performance evaluation of slotted ALOHA with generalized retransmission backoff," *IEEE Trans. Commun.*, vol. 38, pp. 117-122, Jan. 1990.
[8] R. L. Rivest, *Network Control by Bayesian Broadcast* (Report MIT/LCS/TM-285), Cambridge, MA: MIT, Lab. for Computer Science.
[9] J. N. Tsitsiklis, "Analysis of a multiaccess control scheme," *IEEE Trans. Automat. Control*, AC-12, pp. 1017-1020, 1987.
[10] D. Shen, V. O.K. Li, "Stabilized ALOHA for wireless OFDM networks," *IEEE Globecom 2002*.