Delay and throughput analysis of tree algorithms for random access over noisy collision channels

Benny Van Houdt and Robbe Block
Performance Analysis of Telecommunication Systems Research Group
Department of Mathematics and Computer Science
University of Antwerp - IBBT
Middelheimlaan 1, B-2020 Antwerp, Belgium
Email:benny.vanhoudt@ua.ac.be

***Abstract*—The successive interference cancellation tree algorithm with first success (SICTA/FS) was specifically designed as a random access protocol over noisy collision channels. Given J users with an infinite buffer and subject to Poisson arrivals, SICTA/FS achieves throughputs as high as 0.6 if packet losses are allowed (up to 20%), while without packet losses its throughput quickly degrades as the number of users J increases.**

In this paper we indicate that SICTA/FS may remain stable for a considerable amount of time before becoming unstable when the arrival rate exceeds the maximum stable throughput. More importantly, we also study the ALOHA-SICTA/FS algorithm and show that it can achieve throughputs of 0.6 or above without packet loss. Additionally, we provide an accurate estimation of the mean packet delay under ALOHA-SICTA/FS using a simple queueing model with vacations. Finally, we indicate that ALOHA-SICTA/FS suffers from hardly any throughput reduction when the access point stores the last two collision signals only.

***Index Terms*—Random access, tree algorithms, interference cancellation, SICTA/FS, ALOHA-SICTA/FS, AWGN channel**

I. Introduction

Conventional tree algorithms (TAs) are known for their good stability properties both under finite and infinite populations [1], [2]. More recently, the SICTA tree algorithm was introduced that combines conventional TAs with a successive interference cancellation (SIC) mechanism [3]. Basically, SICTA may recover a packet from an otherwise lost slot (i.e., collision slot) by canceling out a number of earlier decoded packets. Assuming a perfect channel with errorless cancellation and an infinite number of users, SICTA achieves a maximum stable throughput (MST) of 0.693 under Poisson arrivals.

In order to apply SICTA in a wireless channel, the SICTA/FS algorithm was introduced in [4]. SICTA/FS is also a blocked access algorithm like SICTA, but instead of going through the entire conflict resolution tree, it terminates the conflict resolution interval (CRI) as soon as the first successful transmission occurs (and recovers as many other packets as possible by means of the SIC mechanism). As such SICTA/FS avoids the potential deadlock due to cancellation errors present in SICTA. Moreover, as the end of a CRI is now identified by a success, SICTA/FS is a limited sensing algorithm, allowing new users to join in easily. The downside of SICTA/FS is that occasionally some of the packets taking part in a CRI are not received successfully. Hence, these packets must either be retransmitted or are considered lost. The model introduced in [4] considers a finite population consisting of J users each having an infinite buffer fed by a Poisson process and an additive white Gaussian noise channel (AWGN). It provides a good approximation for the throughput and mean delay assuming that packets that are unsuccessful in a CRI are lost (the loss is about 20% for throughputs of 0.6).

SICTA/FS was also combined with the binary exponential backoff (BEB) algorithm in [5] in an 802.16 setting. The focus in this paper was mainly on the saturation throughput (that is, the throughput that can be achieved assuming all J users have packets ready for transmission at all times). Clearly, the saturation throughput remains identical irrespective of whether unsuccessful packets are retransmitted by means of the BEB algorithm. Furthermore, [5] also indicated that the saturation throughput of SICTA/FS degrades quickly as the number of users increases, while BEB-SICTA/FS still achieves high throughputs.

In this paper, we revisit the SICTA/FS protocol in the same setting as in [4], but demand that packets that are unsuccessful during a CRI are retransmitted. In contrast to the case where the packets are lost, we show that the high throughput of SICTA/FS quickly vanishes as the number of users J grows (as also indicated in [5]). We also show that there exists a value $\lambda^*_{max} > \lambda_{max}$, such that for arrival rates $\lambda \in (\lambda_{max}, \lambda^*_{max})$, the queue lengths remain small for a considerable amount of time (e.g., millions of CRIs) before growing to infinity. In other words, instability may occur after a considerable amount of time only (for $\lambda > \lambda^*_{max}$ the queues start growing without bound from time 0 onwards).

More importantly, we also analyze the ALOHA-SICTA/FS algorithm and demonstrate that high throughputs can be attained even when the unsuccessful packets in a CRI are retransmitted. These throughputs are comparable to the ones achieved by (C)BEB-SICTA/FS and reported in [5], which demonstrates that the less involved ALOHA algorithm suffices to get high throughputs. We also introduce a simple queueing model with vacations to get an accurate estimation of the mean packet delay when λ is below the MST. We should also stress that the CBEB-SICTA/FS algorithm of [5], where only the saturation throughput was studied analytically, behaves very similar to the ALOHA-SICTA/FS scheme as all the users use the same backoff window. Finally, we also indicate that the

MST of ALOHA-SICTA/FS hardly diminishes when the access point stores the last two collision signals only, indicating that ALOHA-SICTA/FS has a low memory complexity.

The paper is structured as follows. In Section II we present the main assumptions with regard to the random access channel under consideration. Afterward, in Section III we discuss the operation of SICTA/FS and ALOHA-SICTA/FS. The performance of these algorithms is analyzed in Section IV, while numerical results and comparisons can be found in Section V. Finally, some concluding remarks are presented in Section VI.

II. Model Assumptions

We consider a wireless channel with nearly identical properties as in [4]:

1) A finite number of J users is considered each having an infinite capacity buffer. Each user generates fixed length packets according to a Poisson process with the same rate λ (i.e., the system is homogeneous). The users transmit over a single slotted channel to a common access point (AP), the slot length of which is equal to one packet.
2) Three types of immediate feedback are provided (at the end of each slot): $0, k(k > 0)$ or e. A "0" indicates that the slot was idle, a "k" is used to indicate that k packets have been decoded successfully (that is, a successful transmission occurred and $k-1$ more packets were recovered via SIC) and "e" corresponds to an erroneous packet reception.
3) We consider a noisy collision channel which implies that any collision results in an erroneous reception and additionally even if only one user transmits, that packet might be corrupted due to channel noise. The noise model under consideration is the additive white Gaussian noise model (AWGN) discussed below.

Whenever a success occurs (meaning we get $k > 0$ feedback), we state that a new conflict resolution interval starts. In other words if the k-th success takes place in slot n_k, the k-th CRI consists of the slots $n_{k-1}+1$ up to and including slot n_k (with $n_0 = 0$). There is one exception to this rule: when a success is followed by a number of idle slots, each of these idle slots forms a separate CRI.

With regard to the assumptions above, only the feedback differs from [4], where the "k" feedback is replaced by a simple "1" feedback to identify a success. We however require the k feedback at the end of a CRI as each user taking part in the CRI needs to know whether retransmission is required. In [4] this is not necessary as the unsuccessful packets are not retransmitted, but dropped. So, all users that took part in the CRI remove the packet from their queue. The k feedback is also used by SICTA in [3].

Finally, let us discuss the noisy collision channel. Assume BPSK modulation and let E_b/N_0 be the signal to noise ratio (SNR) per bit (that is, $10\log_{10}(E_b/N_0)$ is the SNR in decibels). For the AWGN channel the bit error rate (BER) is given by $P_b^{(0)} = erfc(\sqrt{E_b/N_0})/2$, where $erfc(x) = 2\int_x^\infty e^{-t^2/2}dt/\sqrt{\pi}$ is the complementary error function. Therefore, a packet of length L has a packet error probability of $P_{error}^{(0)} = 1-(1-P_b^{(0)})^L$. As there is noise on the channel, there is also a probability that a cancellation operation fails. Let $P_{error}^{(i)}$ be the probability that cancellation fails given that the collision consists of $i+1$ packets and i of them have already been decoded correctly. Assuming identical E_b for each user and N_1 the induced noise density per cancellation (i.e., SIC is imperfect and induces Gaussian noise with variance N_1), we have

$$P_{error}^{(i)} = 1-(1-P_b^{(i)})^L,$$

with $P_b^{(i)} = erfc(\sqrt{E_b/(N_0+iN_1)})/2$. As soon as the first cancellation fails, no more packets can be decoded within the current CRI. Throughout the paper we set $N_1 = 0.1N_0$ (as in [4]).

III. Algorithms

A. SICTA/FS

Under SICTA/FS all users that have a packet ready for transmission at the start of a CRI take part in the CRI. When there are no users taking part in the CRI, the idle 0 feedback is provided and the CRI ends immediately. Otherwise, each time the error feedback e is provided (meaning at least one user transmitted), the users who just transmitted split into two groups, that is, each user joins the first group with probability p and the second group with probability $1-p$, independently of the other users (typically, $p = 1/2$). The users that selected the first group retransmit in the next time slot. If there is at least one user in the first group, meaning the next time slot has feedback k or e, the users in the second group refrain from retransmitting their packet in the current CRI. Notice, if the second group consists of one user only, his packet might still be decoded correctly via SIC. If the first group is empty on the other hand, the users in the second group immediately split again into two groups and continue applying the same procedure.

As soon as a single transmission is received without error the CRI ends and the AP recovers as many packets as possible using SIC. It does this by first canceling the success from the last collision. If this succeeds, a second packet is decoded and the AP cancels both packets from the last but one collision, in an effort to recover a third packet. This procedure is repeated until either a cancellation fails or all the collisions in the CRI have been used successfully. Notice, cancellation always fails if the second group corresponding to a collision held more than 2 users, e.g., 3 users split into 1 and 2. Even if there is only one user in the second group, cancellation is not guaranteed to succeed due to the noise. Recall, as opposed to [4], we demand that all users participating in the CRI without success retransmit their packet. Also note that it suffices for the AP to announce the number of decoded packets k, as a user knows whether he transmitted in the $k-1$-th collision from the last collision.

B. ALOHA-SICTA/FS

The ALOHA-SICTA/FS differs from SICTA/FS only in the manner in which a user decides to take part in a CRI. With SICTA/FS any user with a nonempty queue at the start of a CRI takes part, with ALOHA-SICTA/FS users with a nonempty queue only take part with probability p_A, where p_A is a system parameter. Notice, if a user takes part in a CRI, but fails to transmit his packet successfully, it will still only participate in the next CRI with probability p_A.

IV. Performance analysis

In order to assess the performance of both the SICTA/FS and ALOHA-SICTA/FS algorithm, we define a Markov chain that keeps track of the joint queue length at the start of each CRI. Stability of the algorithm thus corresponds to the positive recurrence of this multi-dimensional Markov chain.

Applying a similar generating function approach as in [6], [4], [7], we can establish the following relationship provided that the multi-dimensional Markov chain is positive recurrent:

$$\lambda = \frac{P_s p_A (1-P_e)}{P_e E[CRI_0] + (1-P_e)E[CRI_1]}, \quad (1)$$

where P_e is the stationary probability that a tagged user has an empty queue at the start of a CRI, P_s is the probability that a tagged user taking part in a CRI does not need to retransmit his packet, while $E[CRI_0]$ ($E[CRI_1]$) denotes the mean length of a CRI given that the tagged user does not (does) participate in the CRI. Notice, for SICTA/FS the parameter p_A equals one.

Next, as in [6], [4], [7], we rely on the following decoupling assumption to reduce the multi-dimensional Markov chain to a one-dimensional chain: at the start of any CRI we assume that the queue length distributions of the J users are independent. In other words, the number of users participating in a CRI is Binomially distributed with parameters $(J,(1-P_e)p_A)$.

This independence assumption may seem rather strong and implies that our model is no longer exact. However, we will show that the MST can still be determined exactly, while a simple M/G/1 queueing system with vacations will suffice to get a good approximation for the mean delay of ALOHA-SICTA/FS (when p_A is not poorly chosen), especially when the number of users J increases (e.g., for $J = 20$). This perhaps somewhat unexpected accuracy might be in part explained by the decoupling result proven in [8] for the simple ALOHA scheme. For the SICTA/FS algorithm, where $p_A = 1$, the probability of having an empty queue is still quite accurately captured using the decoupling assumption, but the distribution of the number of users participating in a CRI does not match well with a Binomial distribution, causing severe errors for the estimation of the mean packet delay when λ approaches the MST.

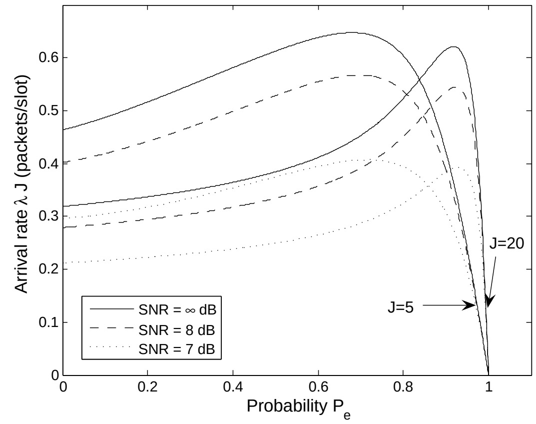

Figure 1. λJ as a function of P_e for $J = 5$ and 20 users for different SNRs for SICTA/FS.

Using the decoupling assumption, we have

$$P_s = \sum_{k=0}^{J-1} B^k_{J-1,(1-P_e)p_A} S(k+1)/(k+1),$$
$$E[CRI_0] = \sum_{k=0}^{J-1} B^k_{J-1,(1-P_e)p_A} EL(k),$$
$$E[CRI_1] = \sum_{k=0}^{J-1} B^k_{J-1,(1-P_e)p_A} EL(k+1)$$

where $B^i_{n,p} = \binom{n}{i} p^i (1-p)^{n-i}$, $EL(k)$ is the mean length of a CRI with k participants and $S(k)$ is the mean number of correctly decoded packets in a CRI with k participants. Both $S(k)$ and $EL(k)$ can be computed easily in a recursive manner as indicated in [4], for completeness we added expressions for both in the Appendix B. In other words, under the decoupling assumption Eqn. (1) provides us with a nonlinear equation for P_e, the stationary probability that a user has an empty queue at the start of a CRI.

A. SICTA/FS

In Figure 1 we have plotted λJ for the SICTA/FS algorithm as a function of P_e for $P_e \in [0,1]$ with $J = 5$ and 20 users and various SNRs (similar curves are found for other J values). This figure indicates that there exists a λ_1 and λ_2 such that P_e has a unique solution for Eqn. (1) if $\lambda < \lambda_1$, has two solutions for $\lambda \in [\lambda_1, \lambda_2)$ and no solutions for $\lambda > \lambda_2$, where λ_1 is found by setting $P_e = 0$ in (1), that is, λ_1 is the saturation throughput.

As we are dealing with a homogeneous system, meaning all the users have the same arrival rate λ and they all participate in a CRI when their buffer is nonempty (or with the same probability p_A in case of ALOHA-SICTA/FS), the MST of SICTA/FS λ_{max} can be proven to coincide with the saturation throughput λ_1, as shown in the Appendix A. Basically, this result holds because *all* the queues become saturated simultaneously at the stability limit, instead of just some of the queues. If we permitted a different arrival rate

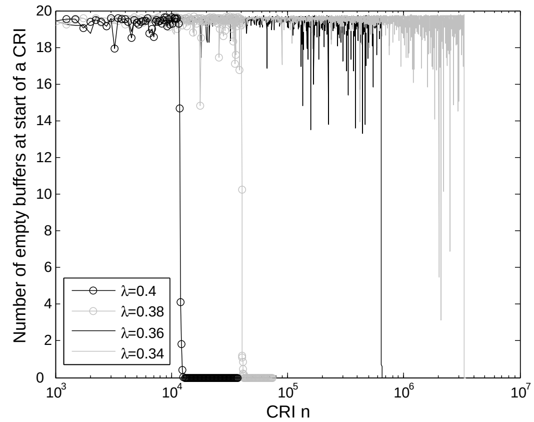

Figure 2. Simulation of the number of empty queues for $J = 20$ users, SNR = ∞ dB and various λ values (with $\lambda_{max} = 0.31868$) for SICTA/FS.

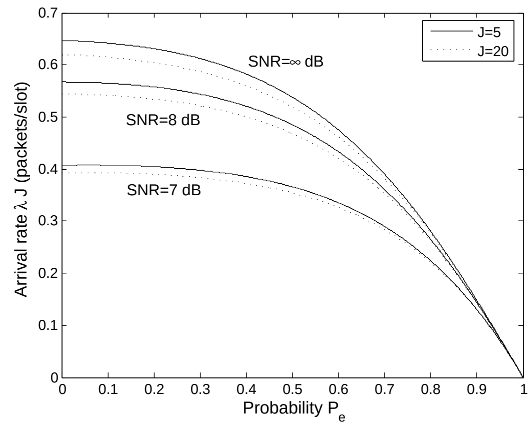

Figure 3. λJ as a function of P_e for $J = 5$ and 20 users for different SNRs for ALOHA-SICTA/FS with $p_A = 1.5/J$.

for each user, this result would no longer hold and dominant systems can be used to bound the stability region as in [9], [10]. Thus, looking at Figure 1, we numerically find that there exists a single solution for P_e via (1) whenever the system is stable. Remark, for SICTA/FS P_e does not decrease to zero as λ approaches λ_{max}, instead it jumps to zero when λ becomes λ_{max}.

Using Figure 1, let us give some intuition as to what implications the existence of the two solutions, denoted as $P_e^{(1)}$ and $P_e^{(2)}$ with $P_e^{(1)} < P_e^{(2)}$, for $\lambda \in [\lambda_{max}, \lambda_2]$ might have. If at some point in time t, the fraction of empty queues $P_e(t)$ is above $P_e^{(2)}$, we see that the λ-value that corresponds to $P_e(t)$ is below λ, therefore on average more queues will become nonempty and $P_e(t)$ is expected to decrease. Similarly, one finds that if $P_e(t)$ is between $P_e^{(1)}$ and $P_e^{(2)}$ it will tend to grow, while for $P_e(t)$ below $P_e^{(1)}$ it tends to decrease again. In other words, there is a drift toward $P_e^{(2)}$ on the interval $(P_e^{(1)}, 1]$ and a drift toward zero on $[0, P_e^{(1)})$.

Thus, if we start with an empty system ($P_e(0) = 1$), the fraction of empty queues might stay in the neighborhood of $P_e^{(2)}$ for quite a while, but given that it drops a sufficient number of times below $P_e^{(2)}$, which it is guaranteed to do because of the Poisson arrivals, it will get stuck in $P_e(t) = 0$ eventually. This is exactly what is happening in Figure 2 where we depict the number of empty queues during a typical simulation run for $J = 20$ users for various λ's in $[\lambda_{max}, \lambda_2]$ (where $P_e^{(2)} J$ varies between 19.52 and 19.62). Each point on this figure represents the mean number of empty queues over 250 consecutive CRIs. The simulation was stopped when all of the queues exceeded a length of a few thousand. Similar results were obtained for $J = 5$ users, though the length of the stable period is much shorter for $j = 5$ even when λ is only a fraction above λ_{max}, this is mainly because the drift toward $P_e^{(2)}$ is not as large as for $J = 20$ (see Figure 1).

B. ALOHA-SICTA/FS

In Figure 3 we depict the same curves as in Figure 1, but now for the ALOHA-SICTA/FS algorithm, where p_A was set to $1.5/J$, such that when all the queues are nonempty 1.5 users take part in a CRI on average. We observe that this choice of p_A results in a much higher saturation throughput compared to SICTA/FS.

Having obtained P_e numerically as the unique solution of Eqn. (1) when λ is below the MST, we will rely on the decoupling assumption to obtain an approximation for the mean delay of ALOHA-SICTA/FS. For this purpose, we will rely on an M/G/1 queueing system with server vacations as in [4]. However, as unsuccessful packets now require retransmission (that is, they are no longer dropped as in [4]) and users with a nonempty buffer only participate in a CRI with probability p_A, the analysis is somewhat more involved.

First, when a packet sees an empty queue upon arrival, it must wait until the current CRI ends, this time interval represents the residual lifetime of the vacation period. In other words, the distribution of the vacation period is equal to the duration of a CRI in which the tagged user does not participate. The number of participants in such a CRI is Binomial with parameters $(J-1, p_A(1-P_e))$ due to the decoupling assumption. As the first two moments of a CRI with k participants, denoted as $EL(k)$ and $EL^2(k)$ are easy to compute recursively (see [4] and Appendix B), the first two moments of the vacation length $E[V]$ and $E[V^2]$ are readily found.

When a packet arrives in a nonempty queue, it must wait until all previously arrived packets have been transmitted. This time interval represents the waiting time. When the packet becomes the head-of-line packet, we state that its service time starts. Its service will consist of several CRIs, in some of these CRIs the packet might not participate, in others it might, but its transmission fails, while in the last CRI part of its service time, it participates successfully. If we assume independence between the lengths of all these CRIs and neglect the correlation between the length of a CRI and whether a packet is successful, we can approximate the service

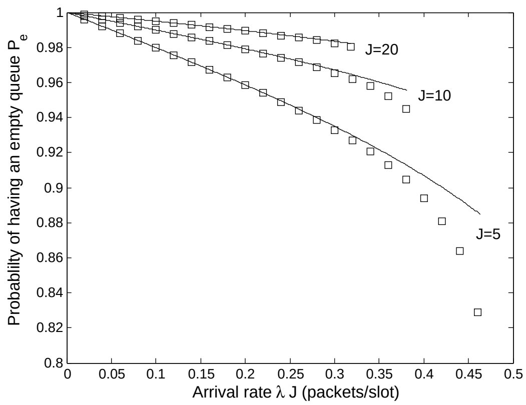

Figure 4. Analytical vs simulation results for P_e, the probability of having an empty queue, for SNR = ∞ dB for SICTA/FS

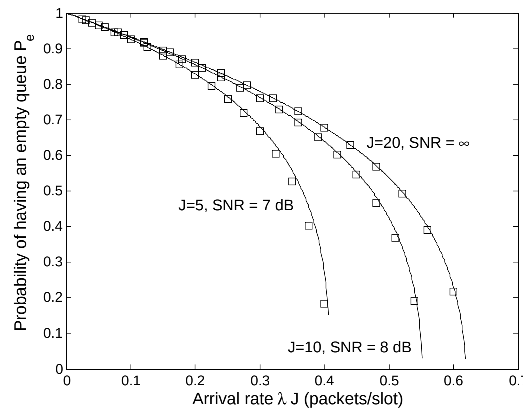

Figure 5. Analytical vs simulation results for P_e, the probability of having an empty queue, for SNR = $\infty, 8$ or 7 dB for ALOHA-SICTA/FS

time distribution, with generating function $S(z)$, as follows:

$$S(z) = \frac{P_s R(z)}{1 - (1 - P_s) R(z)},$$

where $R(z)$ can be expressed as

$$R(z) = \frac{p_A Q_1(z)}{1 - (1 - p_A) Q_0(z)},$$

where $Q_0(z)$ ($Q_1(z)$) denotes the generating function of the CRI length given that the tagged user does not (does) participate in the CRI. Using the decoupling assumption, we can compute the first two factorial moments of $Q_0(z)$ and $Q_1(z)$ easily (which we denote as $Q_0'(1)$, $Q_1'(1)$, $Q_0''(1)$ and $Q_1''(1)$). Using these we can express the first two moments of the service time as

$$E[S] = \frac{Q_1'(1) + Q_0'(1)(1 - p_A)/p_A}{P_s},$$

and

$$E[S^2] = E[S] + \frac{Q_0''(1)(1 - p_A) + Q_1''(1) p_A}{P_s p_A} + \frac{2 Q_0'(1) Q_1'(1)(1 - p_A)(2 - P_s)}{p_A P_s^2} + \frac{2}{P_s^2} \left(\frac{Q_0'(1)^2 (1 - p_A)^2}{p_A^2} + Q_1'(1)^2 (1 - P_s) \right).$$

We can now rely on the well-known expression for the mean delay in an M/G/1 queue with server vacations [11]

$$E[D] = E[S] + \frac{\lambda E[S^2]}{2(1 - \lambda E[S])} + \frac{E[V^2]}{2E[V]}.$$

Thus, we neglect the correlation between the length of consecutive service times as well as the correlation with the vacation length.

V. Numerical Results

Unless otherwise stated, p was chosen as $1/2$ and $N_1 = 0.1 N_0$. We should note that somewhat larger values for p typically slightly improve the performance. The packet length was set equal to 424 bits. When the SNR equals ∞, the

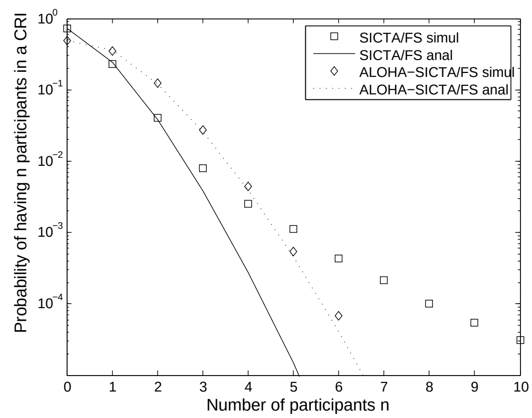

Figure 6. Analytical vs simulation results for the number of participants in a CRI with SNR = ∞ dB, $J = 20$. λ equals 0.3 for SICTA/FS and 0.5 for ALOHA-SICTA/FS (with $p_A = 1.5/J$)

packet length has no impact on the performance. For ALOHA-SICTA/FS the transmit probability p_A was chosen as $1.5/J$, again slightly modifying this value further optimizes the MST. Finally, in all the simulation results conducted we simulated the system for $1,000,000$ CRIs.

A. Decoupling assumption validation

We start by validating the decoupling assumption, more precisely we first compare the unique solution of Eqn. (1) under the decoupling assumption (full lines) with the probability of having an empty queue during simulation (squares). The results, shown in Figure 4, indicate that we get fairly accurate results for SICTA/FS except when λ is close to the saturation throughput and the number of users J is low. When we simulated the system with λ slightly above λ_{max} the system became unstable and P_e was equal to zero. Figure 5 presents a similar figure for ALOHA-SICTA/FS, showing a good agreement between the model and simulation. In this case we used a different SNR for $J = 5$ and 10, as the difference between these curves is small when using the same SNR.

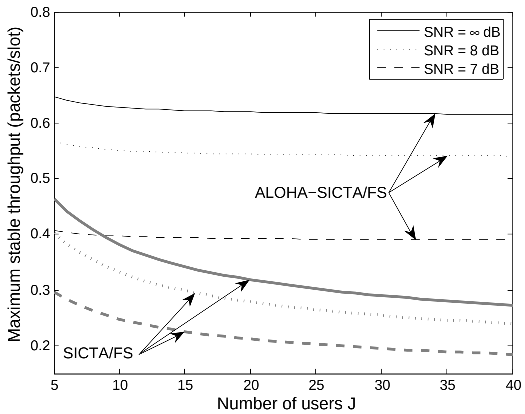

Figure 7. Throughput comparison of SICTA/FS and ALOHA-SICTA/FS (with $p_A = 1.5/J$) on an AWGN channel for various SNRs, fixed length packets of size 424 bits and BPSK modulation.

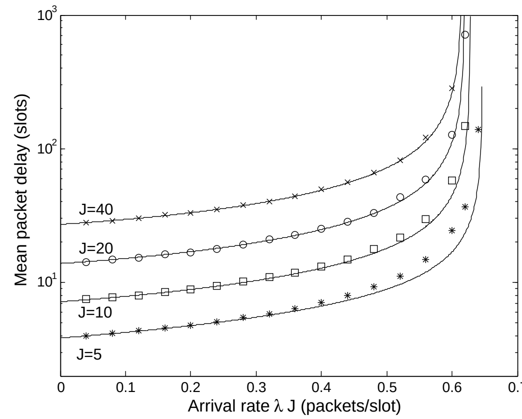

Figure 8. Comparison of the simulated and analytical mean packet delay for ALOHA-SICTA/FS (with $p_A = 1.5/J$) on an AWGN channel with SNR $= \infty$ and $J = 5, 10, 20$ and 40 users.

However, an accurate value for P_e does not guarantee that the decoupling assumption will provide accurate results. To get a better idea of the accuracy of the decoupling assumption, Figure 6 compares the simulated and analytical distribution for the number of users participating in an arbitrary CRI for both SICTA/FS and ALOHA/SICTA-FS for $J = 20$ users and SNR $= \infty$ dB. These results indicate that the decoupling assumption only matches the first few probabilities for SICTA/FS, while for ALOHA-SICTA/FS the decoupling assumption provides a good match for the entire distribution. This is probably because ALOHA-SICTA/FS tends to reduce the correlation between the number of users in consecutive CRIs, which results in the proper tail behavior. Similar results were obtained for other arrival rates λ. The accuracy of the decoupling assumption for ALOHA-SICTA/FS is also further validated by the comparison of the mean packet delay, though we must note that an additional approximation is introduced by the M/G/1 vacation queue.

B. MST and mean delay of ALOHA-SICTA/FS

We start by evaluating the maximum stable throughput of SICTA/FS and ALOHA-SICTA/FS as a function of the number user J for different SNR values in Figure 7. It confirms that the throughput quickly degrades with J for SICTA/FS, while the ALOHA-SICTA/FS provides similar throughputs over the entire range of J. We should note that the SICTA/FS throughputs for SNR = 7 and 8 dB are lower than the ones reported in [5], as we have packets of length 424 bits and use BPSK, while in [5] packets are request packets with a length of 80 bits only and QPSK is used, resulting in lower error rates.

In Figure 8 we compare the mean packet delay as computed by the M/G/1 vacation queue (solid lines) with simulation results (markings) for SNR = ∞. The estimation matches well with the simulation results and tends to improve as the number of users J increases. The agreement is also best for either low to medium arrival rates or rates close to the maximum stable throughput. Similar results can be obtained for other SNR values.

C. Memory requirements of ALOHA-SICTA/FS

In principle the access point (AP) needs to store the signals of the *last* $J-1$ collisions in a CRI as up to $J-1$ successful cancellations may occur at the end of a CRI. In this section we investigate the effect on the MST when reducing the number of stored collision signals to m_s, with $m_s \geq 1$. In order to compute the throughput when the AP stores the last m_s collision signals only, it suffices to set $P_{error}^{(i)} = 1$, for $i > m_s$.

From Figure 9 we may conclude that having only one memory location reduces the MST somewhat, but as soon as the last two collision signals are stored by the AP, throughputs close to those with $J-1$ memory locations are obtained, especially when there is some noise on the channel. This result is mostly due to the low average number of participants within a CRI. Further even if this average was larger, most of the throughput gained by the SIC mechanism is caught with a limited number of memory locations as shown in [12] for the SICTA algorithm.

D. MST and mean delay of ALOHA-MTA/FS

In this section we look at the effect of the SIC mechanism by determining the MST and mean packet delay when the SIC mechanism is not available at the physical layer. In this case the SICTA/FS algorithm reduces to the so-called MTA/FS algorithm, where MTA is the well-known modified tree algorithm [2]. The MTA/FS algorithm thus works in exactly the same manner as SICTA/FS, except that when the first success in a CRI occurs, no efforts are made to recover additional packets via SIC. We can combine MTA/FS with ALOHA in exactly the same manner to obtain the ALOHA-MTA/FS algorithm. To assess its performance we may rely on exactly the same model as for ALOHA-SICTA/FS, by setting $P_{error}^{(i)} = 1$ for $i > 0$ (as not trying to recover packets via SIC is equivalent to applying SIC when the probability of a

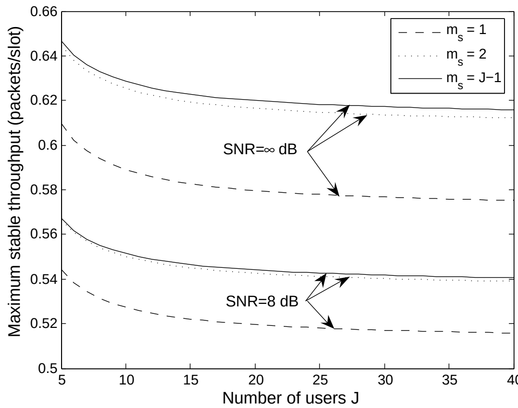

Figure 9. Throughput of ALOHA-SICTA/FS with m_s memory locations (with $p_A = 1.5/J$) on an AWGN channel for SNR $= \infty$ and 8 dB, fixed length packets of size 424 bits and BPSK modulation.

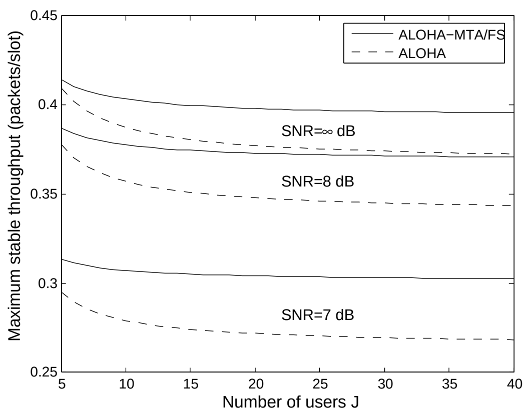

Figure 10. Throughput comparison of ALOHA-MTA/FS (with $p_A = 1.5/J$) and ALOHA (with $p_A = 1/J$) on an AWGN channel for various SNRs, fixed length packets of size 424 bits and BPSK modulation.

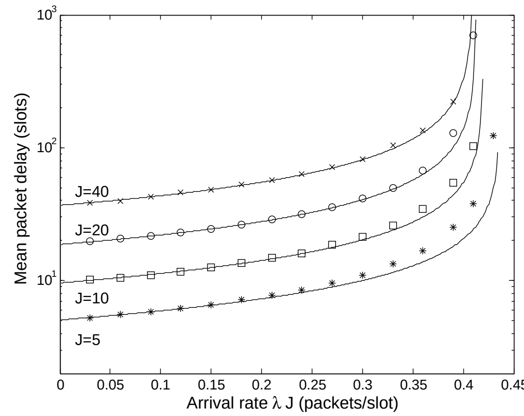

Figure 11. Comparison of the simulated and analytical mean packet delay for ALOHA-MTA/FS (with $p_A = 1.1/J$) on an AWGN channel with SNR $= \infty$ and $J = 5, 10, 20$ and 40 users.

successful cancellation is zero) or by setting $S(k) = 1$ for $k > 0$.

In Figure 10 we see that most of the throughput gains in comparison with a pure ALOHA scheme (with $p_A = 1/J$) disappear when the SIC mechanism is not available. We should note that the ALOHA-MTA/FS throughput can be improved somewhat (that is, by 0.01 to 0.015 depending on the number of users J) by reducing p_A to approximately $1.1/J$. In Figure 11 we compare the simulated mean packet delay of ALOHA-MTA/FS with $p_A = 1.1/J$ with the analytical results obtained by setting $S(k) = 1$, for $k > 0$. As with ALOHA-SICTA/FS, the accuracy improves as the number of users J increases, while the delays are overall somewhat larger for ALOHA-MTA/FS as expected.

VI. Conclusions

In this paper we studied a number of tree algorithms designed to operate on a noisy collision channel. We indicated that the SICTA/FS algorithm may remain stable for a considerable amount of time when the arrival rate is (well) above the maximum stable throughput. More importantly, we analyzed both the throughput and delay of ALOHA-SICTA/FS and demonstrated that throughputs of 0.6 can be achieved without losses, as opposed to SICTA/FS, the throughput of which degrades quickly with the number of users. The delay estimation made use of a simple M/G/1 queueing model with server vacations and was shown to be especially accurate as the number of users J increases.

We further demonstrated that this queueing model is also accurate for the ALOHA-MTA/FS scheme, which suggests that the same approach can also be used for assessing the mean delay of other algorithms like ALOHA-SICTA/F1 and ALOHA-MTA. The MTA algorithm can also be used on a noisy collision channel, but it is not a limited sensing algorithm like MTA/FS. SICTA/F1 operates in the same manner as SICTA/FS, but also terminates a CRI when an idle slot occurs. It was designed for channels with fading, where idle slots are sometimes recognized as collisions, causing SICTA/FS to deadlock.

Finally, we indicated that the ALOHA-SICTA/FS algorithm maintains its high throughput when the access point limits itself to storing the last two collision signals only.

Appendix A
Maximum Stable Throughput

Assume that user i, for $i = 1, \ldots, J$, is subject to a Poisson arrival rate λ_i and assume that the users are labeled such that $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_J$. Let $i < j$ and assume queue i is unstable. Then, using a coupling argument as in [10], this system is equivalent to the system with J users in which user i is saturated, i.e., where user i is assumed to have a nonempty buffer at all times. As user i now participates in all the CRIs, the success rate of any other user must is bounded by the success rate of user i (which is less than λ_i as user i is unstable), because all k users have an equal probability of being successful in a CRI with k participants under SICTA/FS. As such user j, with an arrival rate $\lambda_j \geq \lambda_i$, must be unstable as well.

In other words, if user i is unstable, then so are all the users $j > i$. Hence, if all the users are subject to the same Poisson arrival rate λ, they are all either stable or unstable and the maximum stable throughput therefore coincides with the saturation throughput. A similar argument can be used for the ALOHA-SICTA/FS algorithm given that all the users participate in a CRI with the same probability p_A whenever their buffer is nonempty.

APPENDIX B
CRI LENGTH AND NUMBER OF SUCCESSES

The mean duration $EL(k)$ of a CRI with k participants and the expected number of successes $S(k)$ in such a CRI can be computed as

$$\begin{aligned} EL(1) &= \frac{1 - P_{error}^{(0)} + P_{error}^{(0)}/p}{1 - P_{error}^{(0)}}, \\ EL(k) &= \frac{1 + \sum_{i=1}^{k-1} B_{k,p}^{i} EL(i)}{1 - p^k - (1-p)^k}, \end{aligned}$$

with $EL(0) = 1$ and

$$S(k) = \frac{\sum_{i=1}^{k-1} B_{k,p}^{i} S(i) + B_{k,p}^{k-1} C(k-1)}{1 - p^k - (1-p)^k},$$

with $S(0) = 0, S(1) = 1$ and

$$C(k) = (1 - P_{error}^{(k)}) \prod_{m=2}^{k} \frac{B_{m,p}^{m-1}(1 - P_{error}^{(m-1)})}{1 - p^m - (1-p)^m}.$$

The second moment $EL^2(k)$ of the length of a CRI with k participants obeys the following equations:

$$\begin{aligned} EL^2(1) &= \frac{2(1-p)P_{error}^{(0)}}{p^2(1 - P_{error}^{(0)})} + \left(1 + \frac{2P_{error}^{(0)}}{p(1 - P_{error}^{(0)})}\right) EL(1) \\ EL^2(k) &= \frac{(1 + p^k + (1-p)^k) EL(k)}{1 - p^k - (1-p)^k} + \\ &\quad \frac{\sum_{i=1}^{k-1} B_{k,p}^{i} (EL^2(k) + EL(k))}{1 - p^k - (1-p)^k}, \end{aligned}$$

with $EL^2(0) = 1$

REFERENCES

[1] M. L. Molle and G. Polyzos, "Conflict resolution algorithms and their performance analysis," University of Toronto, CS93-300, Tech. Rep., 1993.

[2] D. Bertsekas and R. Gallager, *Data Networks*. Prentice-Hall Int., Inc., 1992.

[3] Y. Yu and G. B. Giannakis, "SICTA: a 0.693 contention tree algorithm using successive interference cancellation." in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Miami (USA)*, March 2005, pp. 1908–1916.

[4] X. Wang, Y. Yu, and G. B. Giannakis, "A robust high-throughput tree algorithm using successive interference cancellation," *IEEE Transactions on Communications*, vol. 55, no. 12, pp. 2253–2256, 2007.

[5] ——, "Design and analysis of cross-layer tree algorithms for wireless random access," *IEEE Transactions on Wireless Communications*, vol. 7, no. 3, pp. 909–919, 2008.

[6] M. Tsatsanis, R. Zhang, and S. Banerjee, "Network-assisted diversity for random access wireless networks," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 702–711, Mar 2000.

[7] 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.

[8] C. Bordenave, D. McDonald, and A. Proutiére, "Performance of random medium access control, an asymptotic approach," in *SIGMETRICS '08: Proceedings of the 2008 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM, 2008, pp. 1–12.

[9] W. Luo and A. Ephremides, "Stability of N interacting queues in random-access systems," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 1579–1587, July 1999.

[10] W. Szpankowski, "Stability conditions for some multiqueue distributed systems: Buffered random access systems," *Advances in Applied Probability*, vol. 26, pp. 498–515, 1994.

[11] L. Kleinrock, *Queueing Systems Vol. I*. New York: Wiley, 1975.

[12] G. T. Peeters and B. Van Houdt, "Interference cancellation tree algorithms with k-signal memory locations," *To appear in IEEE Transactions on Communications*, 2010.