

Random Access Algorithms in Packet Networks— A Review of Three Research Decades

A. T. Burrell¹, P. Papantoni-Kazakos²

¹Computer Science Department, Oklahoma State University, Stillwater, USA

²Electrical Engineering Department, University of Colorado Denver, Denver, USA

Email: tburrell@cs.okstate.edu, Titsa.Papantoni@ucdenver.edu

Received July 3, 2012; revised August 15, 2012; accepted September 6, 2012

ABSTRACT

We present a coherent and systematic review of Random Access Algorithms for packet networks, as developed over three and a half decades. We consider the appropriate user models and we classify the algorithms according to the channel sensing constraints imposed. We also present a review of the analytical methodologies required for the performance analysis of these algorithms.

Keywords: Random Access; Packet Networks; Channel Sensing; Throughput and Delay Analysis; Sensitivity to Feedback Errors; Limit Poisson User Model

1. Introduction

In this paper, we are focusing on the “random-access” approach, for the accessing of a single, errorless, slotted channel, by independent, identical, packet transmitting, bursty users. The global properties of the user/channel model considered are as follows: All transmitted packets have identical length each requiring the length of a single slot for transmission; the transmission by all users is synchronous, where they are allowed to start transmission only at the beginning of some slot; and there are no propagation delays in the channel feedback information obtained by the users. Also, if at least two packets attempt transmission within the same slot, a collision occurs and such an event is initially the only cause for faulty transmissions; that is, a slot occupied with a single packet results in successful transmission. A collision results in complete loss of the information carried by the collided packets; thus, retransmission of such packets is then necessary. The outcome per slot possibly accessible by the users—named *feedback level*—is either binary, distinguishing between Collision (C) versus Non-Collision (NC), or ternary, distinguishing between collision (C), versus emptiness (E) versus success (S). We note that a NC event corresponds to a slot that is either empty or occupied with a single packet transmission, while an S event corresponds to a slot occupied with a single packet whose transmission is then successful. The accessibility of the feedback level outcomes by the users—named *channel sensing*—is a characteristic of each Random Access Algorithm (RAA) and specifies the time instants

(in slots) when each user is required to sense the feedback level outcomes (accessible by either channel sensing or broadcasting). Based on channel sensing requirements, the existing RAAs may be classified as members of one of the three distinct channel sensing classes below:

Minimal Sensing RAA Class: Each user is required to sense the feedback level outcome of only those slots within which it transmits.

Limited Sensing RAA Class: Each user is required to sense continuously the feedback level outcomes of all slots contained in time periods within which any of its packet is in the system; that is, from the slot within which the packet is generated to that within which this packet is successfully transmitted.

Full Sensing RAA Class: Each user is required to know the overall feedback history of the channel, from the beginning of time and even before the user became part of the system.

Regarding user population models, the following distinction will be necessary in our presentation:

Known User Population Model: The identities of all users are distinct and known to the system. This class implies finite membership.

Unknown User Population Model: The identities of the users are unknown to the system, usually due to time-varying user characteristics. The membership of this class may be either finite or infinite.

Limit Poisson User Population Model: Infinitely many identical Bernoulli users, comprising an aggregate Poisson packet generating process, where each packet is a sepa-

rate user. This is a special case of the unknown user population model.

Historically, the RAA evolution started with Abramson's ALOHA algorithm [1], which belongs in the Minimal Sensing RAA class, it assumes the availability of C versus NC feedback outcomes and is unstable, attaining throughput zero in the presence of the Limit Poisson User Population Model (see Tsybakov and Mikhailov [2]). The latter property of ALOHA initiated some deterministic probing techniques, applicable to bursty Known User Population environments. In particular, assuming the availability of C versus NC feedback outcomes, Hayes [3] proposed a probing technique for the resolution of collisions. In Hayes' scheme (which has source coding characteristics), each nonempty user (for finite number of users) is eventually probed individually by the collision resolution protocol; thus, when it transmits it does not have to include its identification. For the same model as in [3], Capetanakis [4,5] then proposed a modified collision resolution algorithm (CCRA), and he subsequently studied the induced throughput and the expected per-packet delay. As compared to the probing technique in [3], the CCRA (in the case of finite number of users) conserves some additional slots by not probing individually users who together have exactly one stored packet; it requires, however, the inclusion of user identification within each packet.

In the limit, when the number of users increases to infinity, Capetanakis' algorithm [5] is an RAA—named *Binary Split RAA*—that belongs in the Full Sensing RAA class and operates with C versus NC feedback outcomes, its collision resolution operations may be described by a stack with infinite cells, it is stable and there are two versions of it: a static version and a dynamic version. In the static version, after each collision, each one of the collided packets retransmits with probability 0.5. In the dynamic version, each one of an optimally chosen subset of the collided packets retransmits with probability 0.5. In the presence of the Limit Poisson Population model the throughput of the static algorithm is 0.346, while the throughput induced by the dynamic version of the algorithm is then 0.429. Observing that if a collision slot is followed by an empty slot, one slot can be saved by repeating the random retransmission (realized by the probability 0.5) *before* a predictably certain collision is allowed, Massey [6] improved Capetanakis' algorithms. In the presence of the Limit Poisson Population model, Massey's modified algorithms (MCCRA) induce a throughput equal to 0.375 in the static case, and a throughput equal to 0.462 in the dynamic case. However, in contrast to the CCRA, the MCCRA algorithms are very sensitive to channel errors manifested by erroneously received feedbacks. Both the CCRA and MCCRA algorithms were independently introduced by Tsybakov and Mikhailov

[7]. Considering the same model as in [5] and observing the equivalence between random retransmission of packets and subdivisions of the arrival time axis, Gallager [8] and independently Tsybakov and Mikhailov [9] introduced a different conflict resolution algorithm with guaranteed stability and first-come-first-served characteristics. The algorithm, which belongs in the Full Sensing RAA class and utilizes ternary feedback level (E vs. S vs. C), selects an initial arrival interval Δ . It performs subsequent subdivisions of Δ whenever collisions occur, and it reinitializes whenever a collision occurs within the first Δ subdivision. In the presence of the Limit Poisson Population model, the algorithm realizes its maximum throughput 0.4872, for $\lambda\Delta = 1.266$, at the expense of increased operational complexity, as well as increased sensitivity to feedback errors. Humblet [10] improved the algorithm in [8] and [9], increasing the latter throughput to 0.48775. Finally, Mosely [11] studied the possibility of improving the throughput further, by including external, unexamined arrival intervals at some steps. She found that no practical gain is obtained through this approach. Georgiadis *et al.* [12], as well as Tsybakov [13], independently considered an extended version of the algorithm in [5] and [7], assuming the availability of some additional information. In particular, it was assumed that after each collision, although the identity of the collided packets is completely lost, *the number* (up to a possible limit) of the packets involved in the collision is revealed and broadcasted to all users. As a result, a binary split protocol (CRAI) was developed that optimizes retransmission probabilities for throughput maximization, where it was found that its throughput 0.53237 is attainable in the presence of the Limit Poisson Population model via the deployment of only about eight energy detectors. A combination of the absorption concept in [8] and [9] with the availability of conflict multiplicity used in [12] and [13] was considered by Georgiopoulos *et al.* [14] in the development of an enhanced algorithm within the Full Sensing RAA class. Studies on algorithmic sensitivity to feedback errors were performed by Massey [6], Georgiadis *et al.* [12] and Vvedenskaya and Tsybakov [15]. In addition, a class of stack algorithms was developed by Tsybakov and Vvedenskaya [16].

The developed algorithms in the Full Sensing RAA class (summarized in the above paragraph) provided valuable insight. While the algorithms in this class are non-implementable in environments with unknown user models, due to the requirement that all users know the overall feedback history of the channel, the algorithmic studies in it led to implementable algorithms in the Limited Sensing RAA class. The first such algorithm was developed by Tsybakov and Mikhailov [17], where the feedback level is C versus NC , where each packet arrival accesses freely the infinite cell stack in [5] and [7] and

where the collision resolution process is as in [5] and [7]; in the presence of the Limit Poisson Population model, the throughput of the algorithm is 0.134. Later, a class of RAAs—named *Limited Sensing Stack RAAs*—were developed by Tsybakov and Likhanov [18], Vvedeskaya and Tsybakov [15], Georgiadis *et al.* [19], Paterakis *et al.* [20] and Burrell *et al.* [21]. The collision resolution process of these Limited Sensing stack RAAs can be depicted by a stack with finite cells, where for C versus NC feedback level and in the presence of the Limit Poisson Population model, the maximum attained throughput is 0.429. Error sensitivity studies of the latter stack algorithms were performed by Vvedeskaya and Tsybakov [15], by Georgiadis *et al.* [19], by Paterakis *et al.* [20] and by Burrell *et al.* [21]; in general this class is characterized by low sensitivity to feedback errors. Finally, Georgiadis *et al.* [22] modified the algorithm in [8] and [9] for Limited Sensing operation, at the expense of increased operational complexity; the modified Limited Sensing algorithm maintains the 0.4872 throughput in the presence of the Limit Poisson Population model, while, it also maintains its high sensitivity to feedback errors.

In parallel to the construction of specific collision resolution algorithms, several investigators studied upper bounds on the throughputs induced by the whole class of stable protocols. Pippenger [23] first found an upper bound equal to 0.744, for the stable protocols that utilize C versus NC feedback level. This bound was sharpened to 0.704 by Humblet (unpublished paper), and later to 0.6731 by Molle [24], to 0.6125 by Cruz and Hajek (unpublished paper), to 0.587 by Tsybakov and Mikhailov [25], and to 0.5254 by Berger, Mehravari, and Munson [26] (through some conjecture in the latter). If infinitely many energy detectors exist that reveal unlimited collision multiplicities, and if infinitely long initial delays are allowed, Pippenger [23] found that a throughput equal to one is achievable. The satisfaction of this limit one also implies infinite complexity.

RAAs have been modified to accommodate mixed priority data, by Delic [27], Delic *et al.* [28,29], Papantoni-Kazakos [30] and Papantoni-Kazakos *et al.* [31,32]. RAAs have also been adjusted to observe strict delay constraints, by Kurose *et al.* [33], Marcus *et al.* [34] and Paterakis *et al.* [35], while they have also been modified to be implemented in environments with capture by Lyons *et al.* [36]. RAAs deployed by interconnected networks have been studied by Bisdikian *et al.* [37], Hamilton *et al.* [38] and Sidi *et al.* [39]. Channel utilization issues have been considered by Eklundh [40], while an adaptive scheme for the broadcast channel has been proposed by Yemini *et al.* [41]. A methodology for the delay analysis of the 0.487 algorithm was considered by Huang *et al.* [42] and a generalized methodology for the delay analysis of a large class of RAAs has been presented by

Georgiadis *et al.* [43], while the justification of the Limit Poisson Population model as a worst case scenario is presented by Paterakis *et al.* [44]. A review paper was published by Gallager [45]. Additional results in the area are included in the papers by Bar-David *et al.* [46], Fayolle *et al.* [47], Merakos *et al.* [48] and Tsybakov *et al.* [49, 50]. Recommended books that include partial coverage of RAAs are those by Rom and Sidi [51] and Bertsekas *et al.* [52].

In this paper, we will discuss the logical flow leading from the Minimal Sensing RAA class, to the Full Sensing RAA class, to the Limited Sensing RAA class. This objective will be served by the presentation of the ALOHA, the Binary Split and the Limited Sensing Stack RAAs. In the process, we will present the outline of some of the analytical tools needed in the performance evaluation of these RAAs. The organization of the paper is as follows: in Section 2, fundamental concepts and definitions, as well as the analytical approaches used for throughput and delay computations of all considered RAAs are outlined. In Section 3, the ALOHA RAA is examined. In Section 4, the Binary Split RAA and its comparison to ALOHA are presented. In Section 5, the Limited Sensing Stack RAAs are presented and their relationship to the Binary Split RAA is discussed. In Section 6, some extensions of the RAAs discussed in Section 5 are summarized. In Section 7, conclusions are included.

2. Fundamental Concepts and Outline of Analytical Approaches

Random Access Algorithms (RAAs) are deployed when the user population is unknown. In the study of RAAs, the fundamental concepts arising, that also characterize their performance, are: system stability and induced delays. Given some RAA, given the user population, we define:

Throughput: The maximum aggregate packet traffic rate λ^* for which the user/RAA system is stable. Then, $(0, \lambda^*)$ is named the *stability region* of the system.

Per Packet Delay: The distance, in slot units, between the arrival instant of a packet arrival and the instant when its transmission has been completed.

At the same time, studies of error sensitivity correspond to identifying the effect of feedback errors on the throughput of the user/RAA system.

Regarding user/RAA system stability in the presence of independent users that generate memoryless packet streams, the existing developed RAAs induce a sequence $\{T_n\}_{n \geq 0}$ of time instants when consecutive Collision Resolution Intervals (CRIs) intervals occur, where the packet backlogs $\{S_n\}_{n \geq 0}$ at the instants $\{T_n\}_{n \geq 0}$ form an irreducible and aperiodic Markov Chain. Thus, system

stability corresponds then to the ergodicity of the Markov Chain $\{S_n\}_{n \geq 0}$. On the other hand, Paterakis *et al.* [44] proved that as the size of the above user population increases, the stability region of any of the existing RAAs is that corresponding to the Limit Poisson Population model. Therefore, the throughput of any one of the existing RAAs in the presence of the Limit Poisson Population model is a lower bound to the throughput attained by the RAA in the presence of independent users that generate memoryless packet streams. At the same time, the throughput of such an RAA in the presence of the Limit Poisson Population model is then the highest rate for which the Markov Chain $\{S_n\}_{n \geq 0}$ is ergodic, where this chain also induces then infinitely many states. The theories and results developed by Kaplan [53], Spankowskii [54], Spankowski *et al.* [55], Pakes [56] and Tweedie [57] apply in the latter case, to provide a general approach to the computation of the throughput of one of the existing RAAs in the presence of the Limit Poisson Population model. This approach is also applicable to finite independent user populations that generate memoryless traffic streams, and it is summarized by the following steps:

Throughput Computation of a Given RAA

- a) Given the user model, identify appropriate measure of system backlog.
- b) Consider the beginnings A and B of two consecutive CRIs, where A precedes B, and let S_A and S_B denote the backlogs at A and B, respectively.
- c) Require that the conditional expected backlog $E\{S_B/S_A = s\}$ be less than s , for all values of s that are larger than some finite number d determined by the RAA operation. Derive the throughput expression imposing this requirement, which represents negative expected backlog drift across consecutive CRIs.
- d) In the throughput expression in (c), the computation of the expected length of a CRI is required. Derive the tight bounds that may be needed in this computation.
- e) Use the result from step (d) to compute the value of the throughput.

We will follow the above steps when we derive the throughputs of the PAAs in Sections 3-5.

The study of packet delays within the stability region of any of the existing RAAs recruits results from the regenerative theory, as found in Cohen [58] and Kantovich *et al.* [59] and Stidman [60]. Indeed, in the presence of the user models considered above, each such RAA generates time instants when the algorithm restarts independently from its past. Then, the number of generated packets from the beginning of time to the point when such a time instant occurs constitutes a renewal process, where the delays determine a regenerative process with respect to the former renewal process and where

the time instants of algorithmic restarts constitute then renewal points. This allows for the computation of the expected per packet delays as a ratio of the expected aggregate packet delays between consecutive renewal points, over the expected number of packets transmitted between these points, as discussed in detail by Georgiadis *et al.* [43], where the computation of delay bounds is also necessary in the process. We will discuss the renewal points of the binary split and limited sensing stack algorithms in Sections 4 and 5, respectively.

3. The Aloha Random Access Algorithm

The ALOHA Random Access Algorithm operates with binary NC vs. C feedback level outcomes and belongs in the Minimal Sensing RAA class. Given a user population, its operations are described below.

3.1. Operations

- a) A common transmission probability p is deployed by all the users in the system.
- b) Each user has its generated packets queued, where these packets are transmitted on the first come-first serve basis, with the head packet being the oldest in the queue.
- c) Within each slot, each user whose queue is non-empty, transmits the head packet in its queue with probability p . A user that actually transmits within the slot, subsequently observes the resulting NC vs. C feedback outcome: if the outcome is NC , the user concludes that its packet is successfully transmitted; if the outcome is, instead, C , the user repeats the process within the next slot, by retransmitting its head packet with probability p .

We will now proceed with the throughput analysis of the ALOHA RAA, when deployed by M independent users, each generating memoryless packet streams.

3.2. Throughput Analysis for M Independent Users Generating Memoryless Packet Streams

Given M users, we will measure the backlog at the beginning of some slot t , as the total number k of packets queued by all the users. Then, due to the memoryless property of the traffics generated by the users, in conjunction with the memoryless characteristic of the ALOHA operations, the backlog at the beginning of slot $t + 1$ depends only on the value k and the common transmission probability p . In other words, if $\{T_n\}_{n \geq 0}$ denotes here the sequence of consecutive beginnings of slots and if $\{S_n\}_{n \geq 0}$ is then the sequence of the corresponding backlogs at these beginnings, then, $\{S_n\}_{n \geq 0}$ constitutes an aperiodic and irreducible Markov Chain. Let us define:

B : The set containing all aggregate backlogs, such that

at least one packet per user is included.

$E\{S_{n+1}/S_n = k\epsilon B\}$: The expected aggregate backlog at the time instant T_{n+1} , given that the backlog at the time instant T_n equals k and is such that at least one packet per user is included.

λ : The aggregate packet rate in expected number of packets per slot. Then,

$$E\{S_{n+1}/S_n = k\epsilon B\} = k + \lambda - Mp(1-p)^{M-1} \quad (1)$$

where, $Mp(1-p)^{M-1}$ equals the probability of a successful transmission per slot, as well as the expected number of successfully transmitted packets per slot, when all M users are active, as is the case when the backlog k is a member of the set B . The ergodicity theory for aperiodic and irreducible Markov Chains, induces here the following theorem.

3.3. Theorem 1

The ALOHA RAA is stable iff the Markov Chain $\{S_n\}_{n \geq 0}$ is ergodic. In turn, the Markov Chain is ergodic, iff:

$$E\{S_{n+1}/S_n = k\epsilon B\} - k < 0$$

which in view of (1) gives that the necessary and sufficient condition for ALOHA stability is:

$$\lambda < Mp(1-p)^{M-1} \equiv \lambda^*(p, M) \quad (2)$$

The quantity $\lambda^*(p, M)$ in Theorem 1 is the ALOHA throughput for M independent users generating memoryless packet streams, when the common transmission probability is p . The corresponding stability region is

then $(0, \lambda^*(p, M))$. The rates in the stability region are expected aggregate number of packets generated per slot.

Maintaining the number of users M fixed, the throughput in (2) can be maximized with respect to the value p . The latter maximum is easily found to equal $(1-1/M)^{M-1}$ and attained for $p = 1/M$. It is also straight forward to conclude that the quantity $(1-1/M)^{M-1}$ is monotonically decreasing with increasing M , converging asymptotically to the value $1/e$. The latter result led to the unfortunate early conclusion that the throughput of ALOHA for infinite users (or the Limit Poisson Population) is $1/e$, where the fact that this meaningless result can be only attained when the transmission probability is zero (limit of $1/M$ for $M \rightarrow \infty$) was ignored.

3.4. Throughput Analysis for Varying Number of Users

To study meaningfully the ALOHA throughput as the number of users increases, we should fix the transmission probability p to a strictly positive value and study the $\lambda^*(p, M)$ expression in (2), as M increases. **Figure 1** exhibits the behavior of $\lambda^*(p, M)$ as a function of M . From the figure, we observe that as M increases, the throughput $\lambda^*(p, M)$ converges to zero. As a consequence to this observation, in the presence of the Limit Poisson Population, the ALOHA throughput is zero.

3.5. Throughput of the Exponential Back off ALOHA Modification

Current applications in Cellular and Sensor technologies

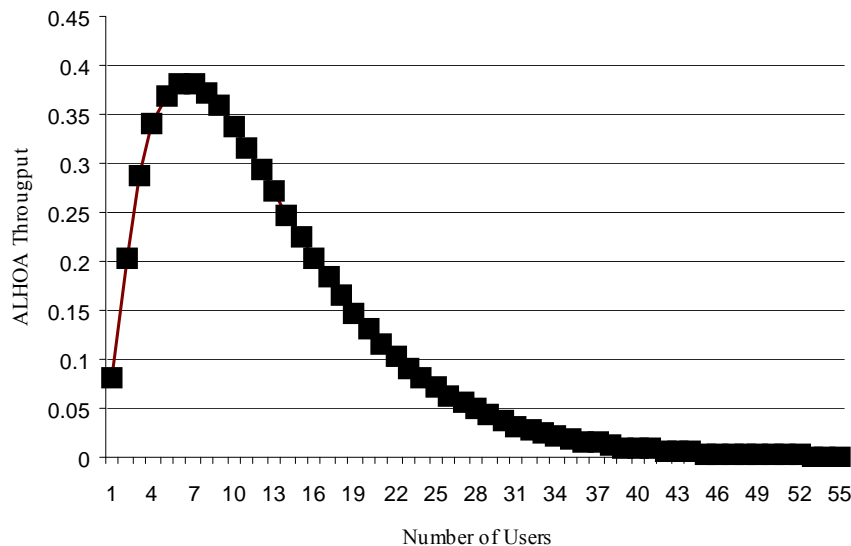


Figure 1. The ALOHA throughput as a function of the number M of users. The probability of transmission is $p > 0$.

$$M_1 = \left\lceil -\frac{1}{\ln(1-p)} \right\rceil, \quad M_2 = \left\lfloor -\frac{1}{\ln(1-p)} \right\rfloor \quad \# \text{ of Users attaining maximum throughput: } M^* = \arg \max(M_1 p(1-p)^{M_1-1}, M_2 p(1-p)^{M_2-1})$$

$$\text{Maximum Throughput: } \lambda^*(p) = \max_M M p(1-p)^{M-1} = M^* p(1-p)^{M^*-1}.$$

deploy an ALOHA modification, where after each unsuccessful transmission attempt, a user defers retransmission by a number of slots which grows exponentially with the number of its unsuccessful attempts. This operation has been named *exponential back off*. Since the modification also deploys packet abortion after the number of retransmissions exceeds a given limit, the number of back offs is bounded. In addition, in a straight forward fashion, the exponential back offs can be equivalently modeled by varying transmission probabilities per slot, whose possible values are then finite. Let q and r denote respectively the largest and smallest among these values. Then, for the model of M independent users whose packet traffics are memoryless, the throughput of the exponential back off ALOHA modified algorithm is bounded from above by the expression $Mp(1-p)^{M-1}$. Fixing the q and r values and allowing the latter bound to vary with the M value, we observe a behavior qualitatively identical to that in **Figure 1**. Thus, the throughput of the ALOHA exponential back off modification converges to zero as the number of users increases. As a consequence, in the presence of the Limit Poisson Population, the throughput of the exponential back off ALOHA modification is zero.

4. The Binary Split Random Access Algorithm

The binary split algorithm belongs in the full sensing RAA class, since user synchronization with the algorithmic operations requires knowledge of the overall system feedback history, as well as continuous feedback monitoring until successful transmission. The algorithm utilizes binary *NC* vs. *C* feedback outcomes per slot and its operations may be described by an initialization and a collision resolution processes, where initialization refers to the initially selected set of packet arrivals that are successfully transmitted during the collision resolution process; the time period (in slot units) that the latter process lasts is named *Collision Resolution Interval (CRI)*. The non-dynamic and the dynamic versions of the algorithm differ only in the initialization part, while their collision resolution processes are identical. In the non-dynamic version, all the waiting packet arrivals are selected for participation in the CRI, while, in the dynamic version, only a subset of these packets are selected, instead. We will proceed by first stating the algorithmic operations during a CRI.

4.1. Operations during a CRI

During a CRI, each involved packet/user is required to observe slot feedbacks continuously until successful transmission. The first slot of a CRI is occupied with simultaneous transmissions from the total set of packets

selected during the initialization part of the algorithm. If the latter set contains at most one packet, the CRI lasts one non-collision (*NC*) slot, during which the initially selected set of packets is successfully transmitted. If the set of arrivals selected during the initialization process contains at least two packets, instead, then the first CRI slot is a collision (*C*) slot and a collision resolution process begins. The algorithmic steps of this process may be described in two ways: 1) as viewed by an imaginary outside observer and 2) as implemented independently by each packet involved in the process. We provide both descriptions below, where time is measured in slot units, where slot t occupies the time interval $(t-1, t]$ and where x_t denotes the feedback outcome of slot t (*NC* vs. *C*).

4.2. Operations as Viewed by an Outside Observer

As viewed by an outside observer, the operations during a CRI that starts with a *C* slot may be described via a stack containing infinite cells indexed by i , where $i = 1, 2, \dots$. The lowest cell 1 in the stack is the transmission cell; that is, the packets transmitted in slot t are those placed in cell 1 at the same slot. The remaining cells in the stack are withholding cells, such that increased cell index implies lower withholding priority. The cell transitions during the CRI are described as follows:

a) At the first slot of the CRI, all packets in the initialization set are placed in cell 1.

b) If $x_t = NC$, then,

The packet (if any) that was in cell 1 at t is successfully transmitted within slot t .

The packets (if any) that were in cell i ; $i \geq 2$ at slot t , move to cell $i - 1$ at slot $t + 1$.

c) If $x_t = C$, then,

Each packet that was in cell 1 at slot t , stays in cell 1 at slot $t + 1$; with probability 0.5 or moves to cell 2 at slot $t + 1$; with probability 0.5.

The packets (if any) that were in cell i ; $i \geq 2$ at slot t , move to cell $i + 1$ at slot $t + 1$.

4.3. Independent Implementation per Packet/User

The CRI operations may be implemented independently by the packets/users, via the use of a counter whose values may be any one of the positive integers. Let then c_t denote the counter value of some packet at slot t , where $c_t = 1, 2, \dots$. The packet is transmitted in slot t if and only if $c_t = 1$. The algorithmic operations during a CRI for any involved packet are then described by transitions of its counter values, as follows:

a) If $x_t = NC$ and $c_t = 1$, then the packet is successfully transmitted within slot t .

- b) If $x_t = NC$ and $c_t \geq 2$, then $c_{t+1} = c_t - 1$.
- c) If $x_t = C$ and $c_t \geq 2$, then $c_{t+1} = c_t + 1$.
- d) If $x_t = C$ and $c_t = 1$, then, $c_{t+1} = 1$; with probability 0.5 or $c_{t+1} = 2$; with probability 0.5.

After careful observation, we may observe that, as induced by the operations of the algorithm, the stack cells or the counter values are unlimited, while, at the same time, there are no distinctive feedback patterns that identify CRI endings. As a consequence, for synchronization with the algorithmic operations via the identification of CRI endings, knowledge of the overall feedback history is required by the users.

Next, we will present the throughput analysis of the binary split RAA, in the presence of the Limit Poisson Population. This throughput will be a lower bound for the binary split RAA, within the class of independent users whose packet generating processes are memoryless, [12].

4.4. Throughput Analysis for the Limit Poisson Population

Since in the presence of the Limit Poisson Population each packet is a separate user, all packet arrivals within a given time interval may be candidates for participation within a CRI. Thus, the initialization part of the algorithm may be mapped onto the selection of the arrival interval to be resolved by each CRI, where the delays caused by the collision resolution process induce a lag between the beginning of a CRI and the time instant of the oldest arrival resolved within its duration. Given that the binary split RAA operates on the generalized first come-first serve basis and in view of the arrival interval mapping, the non-dynamic versus dynamic versions of the algorithm dictate here the following initializations: 1) In the non-dynamic version, the arrival interval selected at the beginning of a CRI is the corresponding lag, while, 2) in the dynamic case, the selected arrival interval is, instead, the minimum (in length) between the lag and an interval of given length Δ that is placed at the beginning of the lag. As will be seen in the process of this paper, the value of Δ is selected for throughput maximization, where the dynamic version of the algorithm is then named, *Window Binary Split RAA*, due to the window of size Δ used by it in its initialization.

As may be deduced from the algorithmic operations, including the algorithmic initialization, in the presence of the Limit Poisson Population, the Binary Split RAA (both dynamic and non-dynamic versions) induces a sequence of consecutive CRIs, where successive CRIs resolve adjacent arrival intervals (transmit successfully the arrivals within such intervals). Thus, in the present case, we will measure backlog by arrival intervals; specifically lags. In particular, let $\{T_n\}_{n \geq 0}$ be the sequence of con-

secutive time instants when the CRIs induced by the algorithm begin and let $\{S_n\}_{n \geq 0}$ be the sequence of the corresponding lags at these instants. Then, due to the memoryless of the Limit Poisson process, the sequence $\{S_n\}_{n \geq 0}$ is an irreducible and aperiodic Markov Chain. Let us first consider the Window Binary Split RAA with window size Δ and let us then define:

λ : The acting Limit Poisson rate, in expected number of packets per slot.

$E\{S_{n+1}/S_n = l \geq \Delta\}$: The expected lag at the beginning of the $(n+1)^{\text{th}}$ CRI, given that the lag at the beginning of the n^{th} CRI equals l and is not less than Δ .

$f(x)$: The expected length of a CRI that resolves the arrivals within a size Δ arrival interval, when the rate of the Limit Poisson process is λ .

$$x \equiv \lambda \Delta$$

As will be clear in the process of this section, in the expression of the expected CRI length, the parameters λ and Δ appear, indeed, only in the product form x . The ergodicity theory of irreducible and aperiodic Markov Chains induces then the following theorem.

4.5. Theorem 2

The Window Binary Split RAA is stable iff the Markov Chain $\{S_n\}_{n \geq 0}$ is ergodic. In turn, the latter Markov Chain is ergodic iff:

$$E\{S_{n+1}/S_n = l \geq \Delta\} - l < 0 \quad (3)$$

where,

$$\begin{aligned} E\{S_{n+1}/S_n = l \geq \Delta\} &= \lambda f(x) - \lambda \Delta + \lambda \\ &= \lambda f(x) - x + \lambda \end{aligned} \quad (4)$$

Thus, as concluded from (3) and (4), the Window Binary Split RAA is stable iff:

$$\lambda f(x) - x < 0 \text{ or } \lambda < x/f(x) \quad (5)$$

Inequality (5) in Theorem 2 provides the expression that leads to the simultaneous computation of both the throughput and the optimal window size Δ^* that attains it. The conceptual methodology towards that direction is as follows. Study the function $x/f(x)$ and search for its maximum. If this maximum exists and is attained at x^* , then $\lambda^* = x^*/f(x^*)$ is the throughput of the algorithm and is attained by a window size $\Delta^* = x^*/\lambda^*$.

From the above, it is clear that the analytical expression and the subsequent computation of the function $f(x)$ are essential. To satisfy this objective, let us first define:

L_k : The expected length of a CRI induced by the Binary Split RAA, given that it resolves a k -multiplicity collision (starts with the simultaneous transmission of k packets).

$P(k/\lambda, \Delta)$: The probability of k packets arriving in an interval of length Δ , when the Poisson rate is λ .

Then, clearly:

$$P(k/\lambda, \Delta) = e^{-\lambda\Delta} \frac{(\lambda\Delta)^k}{k!} = e^{-x} \frac{x^k}{k!}$$

$$f(x) = \sum_{k \geq 0} L_k e^{-x} \frac{x^k}{k!} \quad (6)$$

The quantities in the set $\{L_k\}_{k \geq 0}$ are induced by the collision resolution algorithmic steps and are the only fundamental algorithmic characteristics involved in the throughput computation. These quantities may be computed recursively, while, since the cardinality of the set $\{L_k\}_{k \geq 0}$ is infinite, the development of bounds is also necessary. We will first present the recursive expressions and will continue by presenting the form of the lower and upper bounds for k values above a certain limit. The details of the approach for the computation of the bounds may be found in [12]. Directly from the collision resolution process induced by the algorithm, we first state:

Initial conditions: $L_0 = L_1 = 1$.

For $k \geq 2$:

$$L_k = \sum_{0 \leq i \leq k} 2^{-k} \binom{k}{i} [L_i + L_{k-i}]$$

which after terms' rearrangement gives the recursive expression:

$$L_k = 2^{-k+1} [1 - 2^{-k+1}]^{-1} \sum_{0 \leq i \leq k-1} \binom{k}{i} L_i \quad (7)$$

In [12], the recursive expressions in (7) were used to compute the first thirteen L_k values, precisely. For k values above thirteen, tight upper and lower bounds on the L_k values were computed. These bounds provide linear approximations for the Binary Split RAA. That is,

For $k > 13$:

$$L_k \approx \alpha k + \beta \quad (8)$$

where the values of the α and β constants can be found in [12].

Substitution of the bounds in (8), in the expected CRI length expression (6) gives then:

$$f(x) = \alpha x + \beta + \sum_{0 \leq k \leq 13} [L_k - \alpha k - \beta] e^{-x} \frac{x^k}{k!} \quad (9)$$

The use of the expression in (9) led to the computation of the value $x^* = 1.149$ that attains the maximum of the ratio $x/f(x)$ in (5), where this latter maximum λ^* is the algorithmic throughput. The found throughput and optimal window size, in slot units, are:

For the *Window Binary Split* RAA:

$$\lambda^* = 0.4295 \approx 0.43; \Delta^* = 2.677 \quad (10)$$

Stability region: $\lambda \in (0, 0.43)$

We note that when no optimal window size is selected

and the non-dynamic Binary Split RAA is, instead, deployed, the throughput is the limit value of the ratio $x/f(x)$ and equals 0.34. Thus, the additional controlling parameter represented by the optimal selected window size is highly significant in performance optimization.

At this point, it is interesting to compare the Binary Split RAA with the ALOHA RAA. In the presence of the Limit Poisson Population, ALOHA's throughput is zero, while, for the same population, the throughput of the Window Binary Split RAA is 0.43. This is at the expense of implementability, however, since the latter requires knowledge of the overall system feedback history. As we will see in Section 5, the implementability problem is rectified by the Limited Sensing RAA class, at no loss in throughput.

4.6. Delay Analysis for the Limit Poisson Population

For the analysis of the delays induced by the algorithm, the renewal theory is utilized, where the algorithm induces regenerative points with respect to the delay process. In the presence of the Limit Poisson Population, the renewal points of the Binary Split RAA are the beginnings of CRIs whose initial lag is a single slot long. The algorithm regenerates itself independently at such points. For some given rate λ in the algorithm's stability region, the per packet delay is then computed as the ratio of the aggregate delays between consecutive renewal points, over the aggregate number of packet arrivals between the same points. This approach allows for the computation of per packet delay distributions rather than only expected delays, where the development of bounds on the induced delays is required in the process. The complete methodology is detailed in reference [43]. The expected delays induced by the Window Binary Split RAA are a convex function of the acting rate λ , monotonically increasing with increasing λ value and approaching asymptotically large values as λ approaches the throughput value 0.43; for λ value close to zero, the expected delay is close to 0.5 slots. In Section 5 below, we will compare expected delays induced by the Binary Split and the Limited Sensing Stack RAAs.

4.7. Sensitivity to Feedback Errors

In [12], the behavior of the Window Binary Split RAA has been studied, in the presence of feedback errors. In particular, the throughput of the algorithm was computed, when slots that are either empty or occupied with a single packet transmission may be perceived as collision slots with some probability (we note that if collision slots may be perceived as non-collision slots, packets are rejected and the concept of throughput is then obsolete). The re-

sults of these studies may be found in reference [12], where it is found that the throughput of the Window Binary Split RAA declines gracefully when such feedback errors occur. We will include some of these results in Section 5, when we compare the Window Binary Split RAA with the Limited Sensing Stack RAAs.

5. The Limited Sensing Stack Random Access Algorithm

The experience gained by the study of the Binary Split RAA in Section 4 leads to the following initial observation: if the collision resolution process can be modified to induce distinct CRI endings, then a user arriving in the system may be synchronized with the algorithmic operations without the need to know the overall system feedback history; thus, allowing for algorithmic implementability. The secondary observation arising then is that limitation of the stack size in the Binary Split RAA may attain this objective, where the remaining issue is the performance of the subsequently developed RAAs; that is, their throughput, induced delays and sensitivity to feedback errors. In this section, we present the outcome of the studies in [18,20,21] on the so arising Limited Sensing Stack RAAs. We will start by describing the operations during a CRI. Subsequently, we will explain the limited sensing initialization process, whose feasibility and specifics are direct consequence of the induced distinct CRI endings.

5.1. Operations during a CRI

As stated above, the Limited Sensing Stack RAAs were developed from a modification of the Window Binary Split RAA, via an initial imposed limitation on the size of the stack which represents the collision resolution process. Thus, the feedback outcomes utilized per slot are binary NC vs. C and each involved packet/user is required to observe slot feedbacks continuously until transmission, while the collision resolution process during a CRI, as viewed by an imaginary outside observer, may be described via the use of a stack containing K cells. The same process, as implemented independently by each involved packet/user, may be described via the use of a counter. We provide both descriptions below, where, as in Section 4, slot t represents the time interval $(t-1, t]$ and x_t denotes the NC vs. C feedback outcome of slot t .

1) Operations as Viewed by an Outside Observer

The stack contains $K \geq 2$ cells indexed from 1 to K . Cell 1 is the transmission cell; that is, packets contained in cell 1 at slot t are transmitted within the same slot, while the cells above it are withholding cells representing various withholding priorities. The cell transitions during a CRI are described as follows:

- a) At the first slot of the CRI, all packets contained in

the initialization set are placed in cell 1.

- b) If $x_t = NC$, then,

The packet (if any) that was in cell 1 at t is successfully transmitted within slot t .

The packets (if any) that were in cell i ; $i \geq 2$ at slot t , move to cell $i-1$ at slot $t+1$.

- c) If $x_t = C$, then,

Each packet that was in cell 1 at slot t , places itself in cell i ; $i = 1, \dots, K$, with probability $1/K$ at slot $t+1$.

The packets (if any) that were in cell i ; $i \geq 2$ at slot t , remain in cell i at slot $t+1$.

2) Independent Implementation per Packet/User

The CRI operations are carried independently by each involved packet/user via the use of a counter. Let c_t denote the counter value of a packet at slot t . This value may be one of the positive integer numbers 1 to K , where the packet is transmitted within slot t iff $c_t = 1$. The transition of the counter values during a CRI are as follows:

- a) If $x_t = NC$ and $c_t = 1$, then the packet is successfully transmitted within slot t .

- b) If $x_t = NC$ and $c_t \geq 2$, then $c_{t+1} = c_t - 1$.

- c) If $x_t = C$ and $c_t \geq 2$, then $c_{t+1} = c_t$.

- d) If $x_t = C$ and $c_t = 1$, then, $c_{t+1} = i$; $i = 1, \dots, K$, with probability $1/K$.

As concluded from the above collision resolution operations, after a collision within which it is involved, a packet places itself in either one of the transmission vs. withholding states with equal probability, while after a collision within which the packet is not involved, it maintains its existing state.

After careful observation of the collision resolution process described above, it is concluded that each CRI that starts with a collision ends with the unique pattern of K successive NC slots. In other words, for a CRI that starts with a collision, K successive slots can not occur somewhere in the middle its process, while, at the same time, they characterize uniquely its end. Taking now into consideration trivial CRIs that last a single slot (those resolving at most one packet), we finally conclude that the observation of K successive NC slots may lead to one of the following two assessments: either a CRI that started with a collision ended or K successive trivial (a single slot lasting) CRIs occurred. In either case, if a newly arrived user in the system waits until it observes K successive NC slots, it is assured that at the end of the K^{th} such slot a CRI has ended. This leads to the limited sensing initialization process described below for the Limit Poisson Population model.

5.2. The Limited Sensing Initialization Process for the Limit Poisson Population

The patterns of K successive NC slots explained above allow for the implementation of the stack algorithm

without the need of the overall feedback history knowledge. Such implementation is reflected by the initialization process, named **Limited Sensing**, which imposes the requirement that *each packet arrival observe continuously the slot feedbacks, from the slot within which it arrives to the slot within which it is successfully transmitted*. Specifically, in the presence of the Limit Poisson Population, where each packet is an independent user, a packet that arrives at time instant τ_1 located within slot t_1 , follows the process described below before entering a CRI, where a window of size Δ is utilized.

a) Starting with the slot t_1 of its arrival, the packet observes slot feedbacks passively, until the first occurrence of K successive NC slots ending with slot t_2 . At the end of slot t_2 , the packet is synchronized with the algorithmic collision resolution process, being in the position to recognize all CRI endings after t_2 (including those of the trivial CRIs).

b) k at the end of slot t_2 , the packet checks its arrival instant τ_1 against the arrival interval $(t_2 - K + 1 - \Delta, t_2 - K + 1)$:

i) If $\tau_1 \in (t_2 - K + 1 - \Delta, t_2 - K + 1)$, the packet participates in the CRI that begins with slot $t_2 + 1$ and is successfully transmitted during its process.

ii) If, instead, $\tau_1 < t_2 - K + 1 - \Delta$, the packet updates its arrival instant to $\tau_2 = \tau_1 + \Delta$ and continues observing slot feedbacks sequentially and passively, until the end slot t_3 of the next CRI, when it repeats step (b) for t_2 substituted by t_3 .

c) In general, if t_k is the ending slot of the $(k-1)^{th}$ CRI after the packet's arrival and the packet has not yet participated in a CRI, the packet's arrival instant update at t_k equals $\tau_{k-1} = \tau_1 + (k-1)\Delta$. The packet checks then τ_{k-1} against the arrival interval $(t_k - K + 1 - \Delta, t_k - K + 1)$ and:

i) If τ_{k-1} is contained in $(t_k - K + 1 - \Delta, t_k - K + 1)$, the packet participates in the CRI that begins with slot $t_k + 1$ and is successfully transmitted during its process.

ii) If, instead, $\tau_{k-1} < t_k - K + 1 - \Delta$, the packet updates its arrival instant update to $\tau_k = \tau_1 + k\Delta$ and continues observing slot feedbacks sequentially until the next CRI end.

Regarding the initialization process described above, we make the following observations: 1) The Limited Sensing initialization has last-come first-serve characteristics; 2) While at the initialization state, a packet generates a sequence of arrival instant updates, where each such update is generated at the beginning of each CRI which the packet does not participate in. The updates are responses to the fact that a CRI which does not include the packet resolves a length Δ interval which contains later (than the packet's arrival) packet arrivals. As a result, each update advances the arrival instant of the packet by Δ ; 3) The arrival instant update τ_{k-1} at the end t_k of the $(k-1)^{th}$ CRI is checked against a length Δ arrival

interval $(t_k - K + 1 - \Delta, t_k - K + 1)$, whose right edge lies $K-1$ slots to the left of t_k , where this is because packets that arrived within the latter $K-1$ slots have not yet observed the necessary for algorithmic synchronization and possible next-CRI participation initial K successive NC slots. This arrival interval $(t_k - K + 1 - \Delta, t_k - K + 1)$ is named *examined interval*, where a sequence of such intervals is generated during the initialization process; 4) A packet participates in the first CRI whose examined interval its arrival instant update falls in.

5.3. Throughput Analysis for the Limit Poisson Population

In the throughput analysis, the specific location of the examined intervals has no effect; only their length Δ does. Thus, the only algorithmic characteristics involved in this analysis are the statistical properties of the generated CRIs, as induced by the collision resolution process and the window size Δ . As with the Window Binary Split RAA, the Limited Sensing Stack RAA induces a sequence of consecutive CRIs which, in the presence of the Limit Poisson Population resolve arrival intervals. Thus, as in Section 4, the backlogs $\{S_n\}_{n \geq 0}$ at the beginnings $\{T_n\}_{n \geq 0}$ of consecutive CRIs are measured by arrival intervals. In addition, as in Section 4, in the presence of the Limit Poisson Population, the sequence $\{S_n\}_{n \geq 0}$ is an irreducible and aperiodic Markov Chain whose ergodicity conditions determine the algorithmic throughput. Theorem 2 in Section 4 applies directly here, where the only difference lies in the computation of the conditional expected CRI lengths $\{L_k\}_{k > 0}$. For completeness, we restate the theorem here, as applied to the Limited Sensing Stack RAAs, where lag in this case is defined as the composite total unexamined arrival interval. Considering window size Δ , let us define:

λ : The acting Limit Poisson rate, in expected number of packets per slot.

$E\{S_{n+1}/S_n = l \geq \Delta\}$: The expected lag at the beginning of the $(n+1)^{th}$ CRI, given that the lag at the beginning of the n^{th} CRI equals l and is not less than Δ .

$f(x)$: The expected length of a CRI that resolves the arrivals within a size Δ arrival interval, when the rate of the Limit Poisson process is λ .

$$x \equiv \lambda\Delta$$

The ergodicity theory of irreducible and aperiodic Markov Chains induces then the following theorem.

5.4. Theorem 3

The Limited Sensing Stack RAA is stable iff the Markov Chain $\{S_n\}_{n \geq 0}$ is ergodic. In turn, the latter Markov Chain is ergodic iff:

$$E\{S_{n+1}/S_n = l \geq \Delta\} - l = \lambda f(x) - x < 0 \quad (11)$$

Thus, the Limited Sensing Stack RAA is stable iff:

$$\lambda < x/f(x) \quad (12)$$

If x^* maximizes the ratio $x/f(x)$, then $\lambda^* < x^*/f(x^*)$ is the throughput of the algorithm and is attained by a window size $\Delta^* = x^*/\lambda^*$.

Defining by L_k the expected CRI length, given that the CRI resolves a k multiplicity collision, we obtain, as in Section IV, that the expected CRI length $f(x)$ is given by the following expression:

$$f(x) = \sum_{k \geq 0} L_k e^{-x} \frac{x^k}{k!} \quad (13)$$

We note that as the number of cells K increases from the value two to infinity, a class of RAAs arises which contains the Binary Split RAA. The members of this class which may be implementable in the limited sensing environment are only those corresponding to bounded K values. As with the Binary Split RAA, for any given finite K value, the computation of the expected CRI length $f(x)$ requires the development of recursive expressions and bounds on the conditional expected CRI lengths $\{L_k\}_{k \geq 0}$. We will present such recursions and bounds for the case of $K = 2$; that is, for the Limited Sensing Stack RAA whose stack contains two cells. The recursions for the Limited Sensing Stack RAA whose stack contains three cells may be found in [21]. The complexity of such recursions increases with increasing number K of cells in the stack.

5.5. Computation of the Expected CRI Lengths for the 2 Cell Stack RAA ($K = 2$)

Let us define:

L_k : The expected length of a CRI which starts with a k multiplicity collision.

$L(k, m)$: The expected remaining length of a CRI which is at the state of containing k packets in cell 1 and m packets in cell 2. Then, the following initial conditions and relationships are clear.

$$L_k = L(k, 0), L_0 = L(0, 0) = L_1 = L(1, 0) = 1$$

where the algorithmic operations during a CRI induce the following transitions of the $L(k, m)$ quantities:

For all $m \geq 1$, $L(0, m) = L(1, m) = 1 + L(m, 0)$;

For $k \geq 2$;

$$\begin{aligned} L(k, m) &= 1 + 2^{-k} \sum_{0 \leq i \leq k} \binom{k}{i} L(i, k+m-i) \\ &= 1 + 2^{-k} L(k, m) + 2^{-k} \sum_{0 \leq i \leq k-1} \binom{k}{i} L(i, k+m-i) \end{aligned}$$

Rearrangement of terms in the last equation leads to

the following expression which, for fixed sum $k + m$, is recursive with respect to k .

For $k \geq 2$;

$$\begin{aligned} L(k, m) &= 2^k [2^k - 1]^{-1} \\ &+ [2^k - 1]^{-1} \sum_{0 \leq i \leq k-1} \binom{k}{i} L(i, k+m-i) \end{aligned} \quad (14)$$

For fixed $k + m$ sum value, the recursive computation in (14) is repeated for increasing k values, while this repeated recursive computation is performed sequentially for increasing $k + m$ sum values. The end result of this process is the recursive computation of the quantities in the set $\{L_k\}_{k \geq 0}$. As with the Binary Split RAA, the cardinality of the latter set is infinite, requiring the development of tight upper and lower bounds for large k values. Such bounds were developed in [42], where, in contrast to the Binary Split RAA case, their form is quadratic here. That is, for specifically found α , β and γ constant values, we have:

For $k \geq 13$;

$$L_k \approx \alpha k^2 + \beta k + \gamma \quad (15)$$

In view of the expression in (15), the expected CRI length $f(x)$ in (13) may be written as,

$$\begin{aligned} f(x) &\approx \sum_{k \geq 0} (\alpha k^2 + \beta k + \gamma) e^{-x} \frac{x^k}{k!} \\ &+ \sum_{0 \leq k \leq 13} [L_k - \alpha k^2 - \beta k - \gamma] e^{-x} \frac{x^k}{k!} \\ &= \alpha x^2 + (\alpha + \beta) x + \gamma \\ &+ \sum_{0 \leq k \leq 13} [L_k - \alpha k^2 - \beta k - \gamma] e^{-x} \frac{x^k}{k!} \end{aligned} \quad (16)$$

where the $\{L_k\}_{0 \leq k \leq 13}$ values in (16) are computed precisely in a recursive fashion.

5.6. Results for Varying K Values

For given K value, the computation of the expected CRI length (as in (16) for $K = 2$), in conjunction with the statements in Theorem 3 lead to the computation of the throughput λ^* and the optimal examined interval length Δ^* that attains it. In **Table 1** below, we show these computed values for the Stack RAAs with $K = 2$ and $K = 3$ cells, from [20] and [21], respectively. For the sake of comparisons, we also show the same values for the Window Binary Split RAA in Section 4 (which corresponds to the limit K value).

Stack RAA with $K = 2$: $\lambda^* = 0.4295 \approx 0.43$ $\Delta^* = 2.33$;

Stack RAA with $K = 3$: $\lambda^* = 0.4295 \approx 0.43$ $\Delta^* = 2.5599$

Window Binary Split RAA: $\lambda^* = 0.4295 \approx 0.43$ $\Delta^* = 2.677$

Table 1. Stability regions and optimal windows for stack raas.

Stack RAA with $K = 2$:	$\lambda^* = 0.4295 \approx 0.43$	$\Delta^* = 2.33$
Stack RAA with $K = 3$:	$\lambda^* = 0.4295 \approx 0.43$	$\Delta^* = 2.5599$
Window Binary Split RAA:	$\lambda^* = 0.4295 \approx 0.43$	$\Delta^* = 2.677$
Common stability region: $\lambda \in (0, 0.43)$		

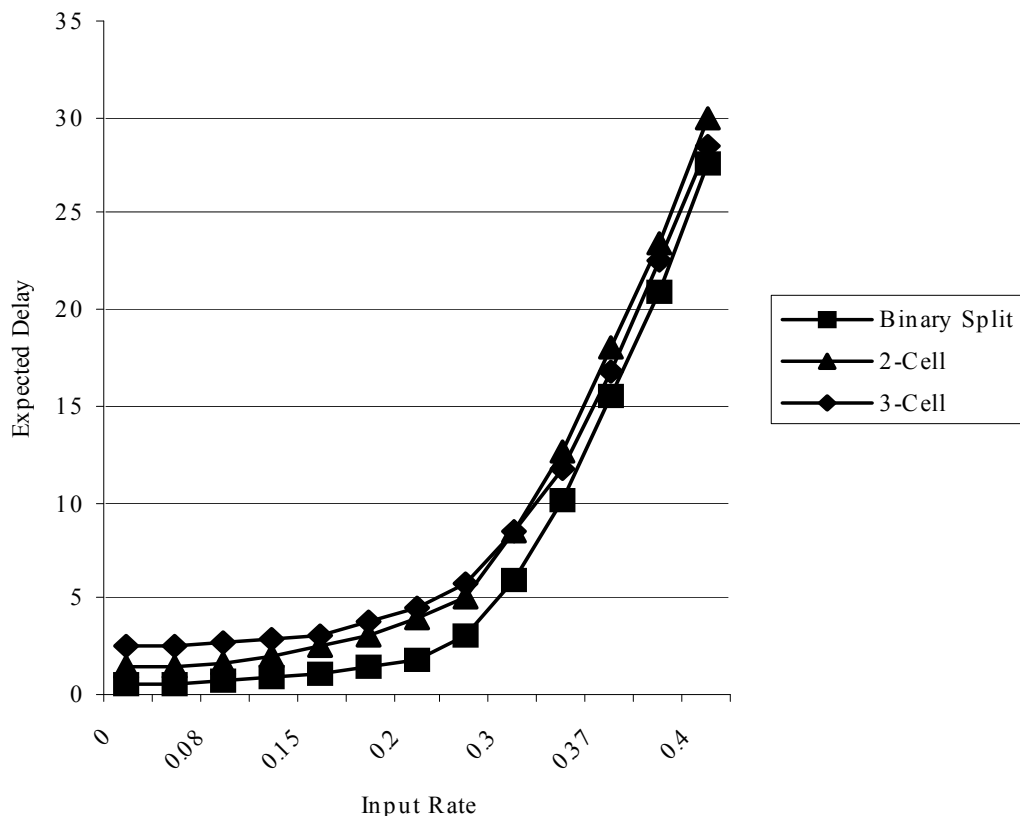
It may be conjectured that, as the size K of the stack increases, from 2 to infinity, the RAA throughput remains unchanged, while, as viewed from **Table 1**, the optimal length of the examined interval increases. Thus, the K value does not affect the throughput, but does affect implementability. As we will discuss below, the K value also affects delays and sensitivity to feedback errors.

As compared to the ALOHA RAA, in the presence of the Limit Poisson Population, the class of Limited Sensing Stack RAAs attain throughput 0.43, while ALOHA's throughput is then zero. This is at the expense of increased, but reasonable and implementable feedback sensing. Indeed, the Limited Sensing Stack RAAs require that each packet/user sense the per slot feedback outcomes continuously, from the time it is generated to the time that it is successfully transmitted. In contrast, ALO-

HA requires sensing of feedback outcomes only for slots within which the packet is transmitted, but its throughput rapidly converges to zero when the user population increases.

5.7. Delay Analysis for the Limit Poisson Population

As in Section 4, for the analysis of the delays induced by the Limited Sensing Stack RAAs, the renewal theory is utilized again, where the algorithm induces regenerative points with respect to the delay process. In the presence of the Limit Poisson Population and for given K , the renewal points are the beginnings of CRIs whose initial lag is K slots long. For given K value and some given rate λ in the stability region, the per packet delay is then computed as the ratio of the aggregate delays between consecutive renewal points, over the aggregate number of packet arrivals between the same points. The approach detailed in [18] allows for the computation of delay distributions and includes the development of tight bounds. In **Figure 2**, we plot computed expected delays as functions of the rate λ , for the 2-Cell Stack, the 3-Cell Stack and the Binary Split RAAs. We note that for very light traffic (λ near zero), a packet is delayed only due to the initial algorithmic synchronization requirement, since no

**Figure 2. Expected delays in slot units for the 2-Cell, 3-Cell and binary split RAAs.**

collisions occur then. The latter requirement is 2 slots; for the 2-Cell Stack RAA, 3 slots; for the 3-Cell Stack RAA and 0 slots for the Binary Split RAA. Adding to these numbers the 0.5 slot accounting for the delay due to the mid-point average arrival instant of the packet within its arrival slot, we finally obtain that for very light traffic, the expected delays induced by the 2-Cell Stack, the 3-Cell Stack and the Binary Split RAAs are approximately 1.5, 2.5 and 0.5 slots, respectively. As the rate λ approaches the throughput value 0.43, the expected delays of all three RAAs approach asymptotically high values, while the expected delay of the Binary Split RAA remains uniformly lower than that induced by the other two algorithms. Comparing the 2-Cell Stack RAA with the 3-Cell Stack RAA in terms of expected delays, we note that the 2-Cell Stack RAA outperforms the 3-Cell Stack RAA for low traffic rates, while the outperforming is reversed when traffic rates are relatively high. We finally note that the last-come first-serve characteristic of the limited sensing 2-Cell Stack and 3-Cell Stack RAAs induces higher variances of the delay distributions, as compared to that induced by the Binary Split RAA.

5.8. Sensitivity to Feedback Errors

Insensitivity to feedback errors, commonly caused by channel noise, is a significant attribute to consider in the performance evaluation of RAAs. The specific issue or interest is the throughput degradation when feedback errors cause misperception of NC feedbacks, where sure instability results from C feedbacks' misperceptions. We will draw from the results in [6,17,42], where the throughputs of the 2-Cell Stack, the 3-Cell Stack and the Binary Split RAAs were computed when empty or occupied with a single packet transmission slots may be perceived as collision slots. Let us define:

ε : The probability that an empty slot may be perceived as a collision slot;

δ : The probability that a slot occupied with a single packet transmission may be perceived as a collision slot.

Then, throughputs induced by the three RAAs, for different ε and δ values are exhibited in **Table 2**.

From **Table 2**, we observe that the 2-Cell Stack RAA is the most robust in the presence of feedback errors, where, among the three studied RAAs, its throughput degrades the least with increasing error probabilities. In addition, for all ε and δ values less than 1, the 2-Cell Stack and 3-Cell Stack RAAs induce positive throughputs, while the throughput of the Binary Split RAA is zero for ε values equal to or above 0.50.

6. Extensions of the Limited Sensing Stack Raas

The Limited Sensing Stack RAAs presented in Section 5

Table 2. Throughputs in the presence of feedback errors.

ε	δ	2-Cell Stack RAA	3-Cell Stack RAA	Binary Split RAA
0.00	0.01	0.424	-	0.410
0.00	0.20	-	0.312	-
0.00	0.40	0.255	-	0.262
0.00	0.50	0.363	0.156	-
0.10	0.00	0.408	-	0.401
0.10	0.10	0.363	0.332	-
0.20	0.00	0.377	-	0.355
0.20	0.10	0.328	0.254	-
0.20	0.20	0.279	0.175	0.266
0.30	0.10	0.261	-	0.192

have very attractive characteristics: Their feedback sensing requirements are implementable and reasonable, they remain stable when the user population increases to large values, their throughput is relatively high, they induce good delay characteristics and are robust in the presence of feedback errors. They are thus attractive candidates for deployment in environments which impose additional restrictions or requirements, such as imposed delay hard constraints or delay advantages granted to a subclass of priority users. In environments where hard constraints on delays are imposed, renting stability a void concept, the Limited Sensing Stack RAAs will outperform significantly any ALOHA-based RAA in terms of induced delays. In [31], the Limited Sensing 2-Cell Stack RAA was adapted to accommodate both regular and high priority data traffic. Summary of the generic approach is presented below.

Let us consider an environment containing both regular and high priority users. Let us then deploy M distinct transmission channels and adopt the following initial channel assignments for the users: 1) Each regular user is assigned one of the M channels for its transmissions, permanently; 2) Each high priority user is allowed the flexibility to select dynamically any one of the M channels for its transmissions. The algorithmic system is then described as follows:

- The K -Cell Stack RAA is deployed in all M channels;
- Each regular user follows the rules of the K -Cell Stack RAA within its pre-assigned channel;
- Upon arrival, each high priority packet/user starts sensing continuously the feedback outcomes from all M channels, generating separate per channel sequences of arrival instant updates. For its transmission, the user selects then the first CRI whose examined interval its corresponding arrival instant update falls into. If the latter

CRI corresponds to channel i , the user stops sensing the feedback outcomes from the remaining channels and transmits its packet during this CRI in channel i .

As deduced from the above description, as compared to the regular users, the advantage awarded to high priority users by the approach is reduced waiting time during the initialization process, at the expense of increased feedback sensing across the M channels. Since the waiting time comprises a significant part of the overall per packet delay, this advantage has a significant delay reduction effect for the high priority users, especially when their arrival rate is significantly lower than that of the regular users.

The algorithmic system induces coupling across the different channels. The stability analysis is thus involved and can be found in [31], for the case where a total of two channels and the Limited Sensing 2-Cell Stack RAA are deployed. In **Figure 3**, we exhibit the stability regions for the Limit Poisson Population in the latter case, parameterized by the rate of the high priority traffic. As can be seen from the figure, the stability region reduces as the rate of the high priority traffic increases. As shown

in [31] and [27,28], the stability region per channel also decreases with increasing number M of channels, at the gain then of increased reduction in the delays of the high priority traffic. In **Figure 4**, we exemplify then the effect of the approach on the expected delays of the high priority users when the rate of the regular traffic per channel is 0.3 and the rate of the high priority traffic is less than 0.1, where all traffics are Limit Poisson.

7. Conclusion

We reviewed the results of a more than two decades concentrated research effort on random access algorithms. We gave special attention to the level of feedback sensing required by each algorithm and its stability region. In view of the ALOHA instability in the presence of increasing user populations, the Limited Sensing Stack Random Access Algorithms (RAAs) are especially attractive for deployment in various applications exemplified by cellular and clustered sensor topologies. The latter algorithms are stable in the presence of increasing user populations, attaining a throughput lower bound

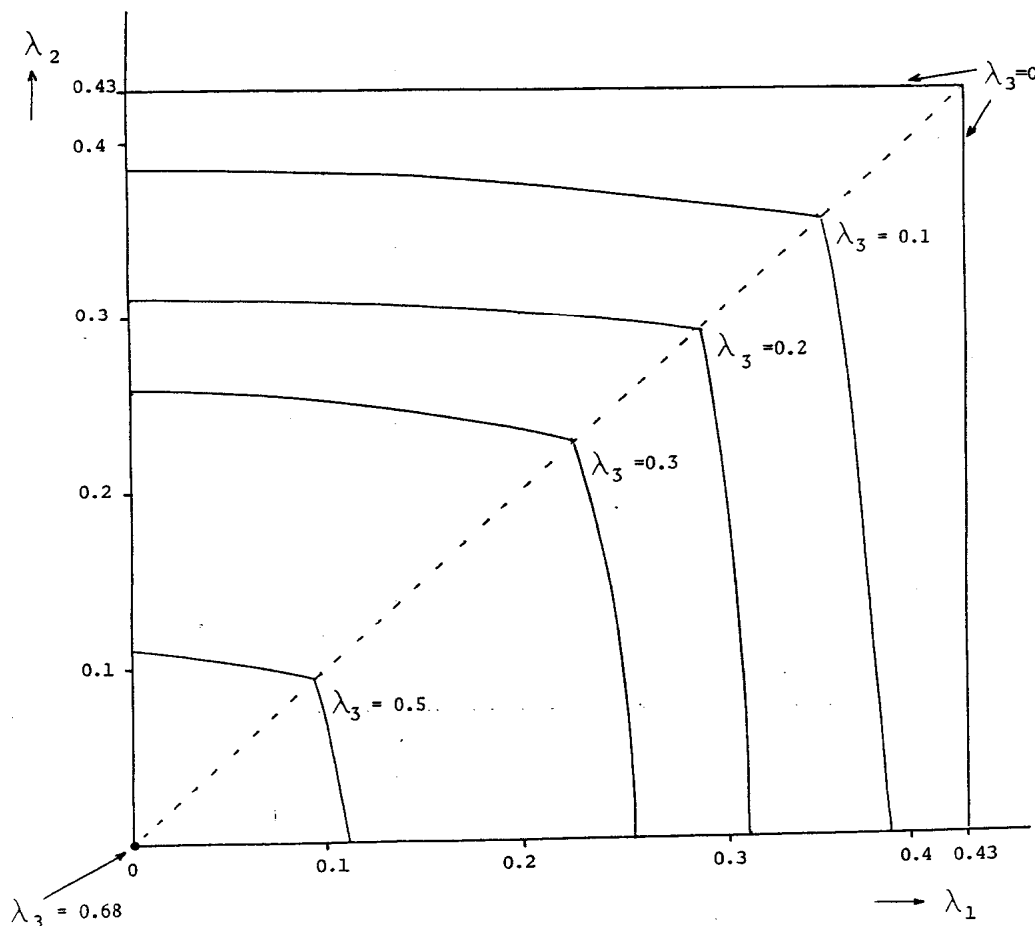


Figure 3. Stability Regions for a 2 Channel System with the 2-Cell Stack RAA Deployed. Limit Poisson Populations. The Rates of the Regular Traffic per Channel are λ_1 and λ_2 . The Rate of the High Priority Traffic is λ_3 .

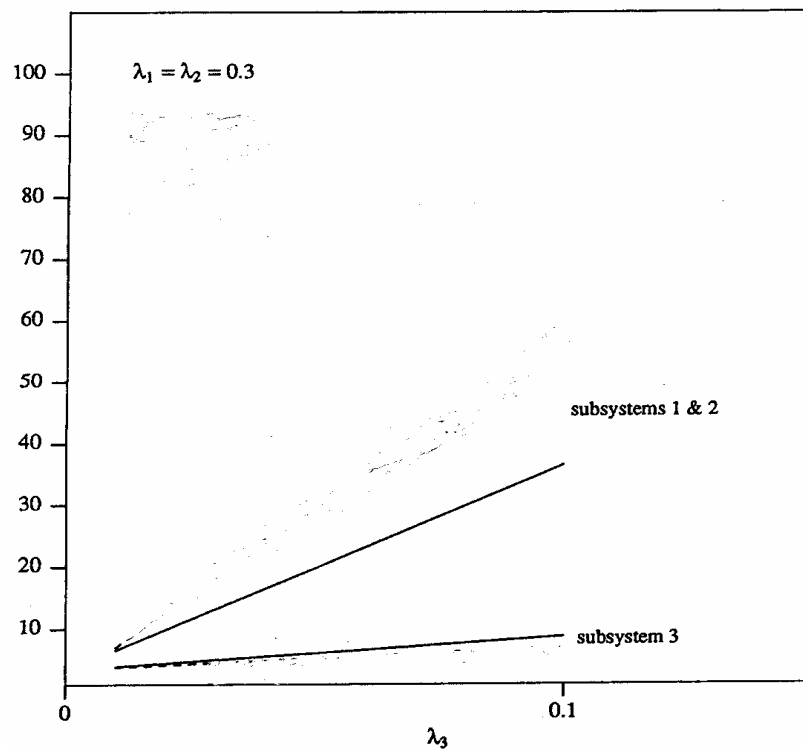


Figure 4. Expected delays for the 2 Channel System and the 2-Cell Stack RAA Limit Poisson Populations The Rates of the Regular Traffic per Channel are λ_1 and λ_2 . The Rate of the High Priority Traffic is λ_3 .

0.43, they induce good delay characteristics and are robust in the presence of channel errors. In environments where ALOHA-based algorithms are presently deployed, the Limited Sensing Stack RAAs may provide superior performance, well outbalancing the increased (as compared to ALOHA) feedback sensing they require. These RAAs may also be deployed to accommodate mixed high priority versus regular data traffics, with excellent results.

REFERENCES

- [1] N. Abramson, "The ALOHA System: Another Alternative for Computer Communications," *AFIPS'70 (Fall) Proceedings of the November 17-19, New York, 17-19 November 1970*, pp. 281-285.
- [2] B. S. Tsybakov and V. A. Mikhailov, "Ergodicity of a Slotted ALOHA System," *Problemy Peredachi Informatsii*, Vol. 15, No. 4, 1979, pp. 73-87.
- [3] J. F. Hayes, "An Adaptive Technique for Local Distribution," *IEEE Transactions on Communication*, Vol. 26, No. 8, 1978, pp. 1178-1186.
- [4] J. I. Capetanakis, "Tree Algorithms for Packet Broadcast Channels," *IEEE Transactions on Information Theory*, Vol. 25, No. 5, 1979, pp. 505-515.
[doi:10.1109/TIT.1979.1056093](https://doi.org/10.1109/TIT.1979.1056093)
- [5] J. Capetanakis, "Generalized TDMA: The Multi-Accessing Tree Protocol," *IEEE Transactions on Communication*, Vol. 27, No. 10, 1979, pp. 1476-1484.
- [6] J. L. Massey, "Collision Resolution Algorithms and Random-Access Communications," In: G. Longo, Ed., *Multi-User Communications (CISM Courses and Lectures Series)*, Springer-Verlag, New York, 1981, pp. 73-137.
- [7] B. S. Tsybakov and V. A. Mikhailov, "Slotted Multi-Access Packet Broadcasting Feedback Channel," *Problemy Peredachi Informatsii*, Vol. 14, No. 4, 1978, pp. 32-59.
- [8] R. G. Gallager, "Conflict Resolution in Random Access Broadcast Networks," *Proceedings of AFOSR Workshop in Communication Theory and Applications*, Provincetown, 17-20 September 1978, pp. 74-76.
- [9] B. S. Tsybakov and V. A. Mikhailov, "Packet Random Multiple Access; Part-and-Try Algorithm," *Problemy Peredachi Informatsii*, Vol. 16, No. 4, 1980, pp. 65-79.
- [10] P. Humblet, "On the Throughput of Channel Access Algorithms with Limited Sensing," *IEEE Transactions on Communication*, Vol. 34, No. 4, 1986, pp. 345-347.
[doi:10.1109/TCOM.1986.1096539](https://doi.org/10.1109/TCOM.1986.1096539)
- [11] J. Mosely, "An Efficient Contention Resolution Algorithm for Multiple Access Channels," Massachusetts Institute of Technology, Cambridge, 1979.
- [12] L. Georgiadis and P. Papantoni-Kazakos, "A Collision Resolution Protocol for Random Access Channels with Energy Detectors," *IEEE Transactions on Communication*, Vol. 30, No. 11, 1982, pp. 2413-2420.
[doi:10.1109/TCOM.1982.1095438](https://doi.org/10.1109/TCOM.1982.1095438)

- [13] B. S. Tsybakov, "Resolution of a Conflict of Known Multiplicity," *Problemy Peredachi Informatsii*, Vol. 16, No. 2, 1980, pp. 69-82.
- [14] M. Georgiopoulos and P. Papantoni-Kazakos, "Collision Resolution Protocols Utilizing Absorptions and Collision Multiplicities," *IEEE Transactions on Communication*, Vol. 33, No. 7, 1985, pp. 721-724.
[doi:10.1109/TCOM.1985.1096366](https://doi.org/10.1109/TCOM.1985.1096366)
- [15] N. D. Vvedenskaya and B. S. Tsybakov, "Random Multiple Access of Packets to a Channel with Errors," *Problemy Peredachi Informatsii*, Vol. 19, No. 2, 1982, pp. 52-68.
- [16] B. S. Tsybakov and N. D. Vvedenskaya, "Random Multiple-Access Stack Algorithm," *Problemy Peredachi Informatsii*, Vol. 16, No. 3, 1980, pp. 80-94.
- [17] B. S. Tsybakov and V. A. Mikhailov, "Free Synchronous Packet Access in a Broadcast Channel with Feedback," *Problemy Peredachi Informatsii*, Vol. 14, No. 4, 1978, pp. 259-280.
- [18] B. S. Tsybakov and N. B. Likhanov, "Some New Random Multiple Access Algorithms," *Problems of Information Transmission*, Vol. 21, No. 2, 1985, pp. 134-154.
- [19] L. Georgiadis and P. Papantoni-Kazakos, "Limited Feedback Sensing Algorithms for the Broadcast Channel," *IEEE Transactions on Information Theory*, Vol. 31, No. 2, 1985, pp. 280-294.
- [20] M. Paterakis and P. Papantoni-Kazakos, "A Simple Window Random Access Algorithm with Advantageous Properties," *IEEE Transactions on Information Theory*, Vol. 35, No. 5, 1989, pp. 1124-1130.
[doi:10.1109/18.42234](https://doi.org/10.1109/18.42234)
- [21] A. T. Burrell and P. Papantoni-Kazakos, "A Class of Limited Sensing Random Access Algorithms with Resistance to Feedback Errors and Effective Delay Control," *Journal of Communications and Networks*, Vol. 8, No. 1, 2006, pp. 21-27.
- [22] L. Georgiadis and P. Papantoni-Kazakos, "A 0.487 Throughput Limited Sensing Algorithm," *IEEE Transactions on Information Theory*, Vol. 33, No. 2, 1987, pp. 233-237.
[doi:10.1109/TIT.1987.1057278](https://doi.org/10.1109/TIT.1987.1057278)
- [23] N. Pippenger, "Bounds on the Performance of Protocols for a Multiple-Access Broadcast Channel," *IEEE Transactions on Information Theory*, Vol. 27, No. 2, 1981, pp. 145-151.
[doi:10.1109/TIT.1981.1056332](https://doi.org/10.1109/TIT.1981.1056332)
- [24] M. Molle, "On the Capacity of Infinite Population Multiple Access Protocols," *IEEE Transactions on Information Theory*, Vol. 28, No. 3, 1982, pp. 396-401.
[doi:10.1109/TIT.1982.1056509](https://doi.org/10.1109/TIT.1982.1056509)
- [25] B. S. Tsybakov and V. A. Mikhailov, "An Upper Bound for Maximum Throughput of Random Access System," *International Symposium on Information Theory*, Santa Monica, 1981.
- [26] T. Berger, N. Mehravari and G. Munson, "On Genie-Aided Upper Bounds to Multiple Access Contention Resolution Efficiency," Johns Hopkins University, Baltimore, 1981.
- [27] H. Delic, "Sharing Policies in Communication Networks," Master's Thesis, University of Virginia, Charlottesville, 1990.
- [28] H. Delic and P. Papantoni-Kazakos, "Performance Analysis for Multi-Channel Random Access Interconnection Analysis," *Proceedings of the 23rd Annual Conference on Information Sciences and Systems*, Baltimore, 18-20 March 1989, pp. 695-699.
- [29] H. Delic and P. Papantoni-Kazakos, "An Optimal Policy for Competitive Processing of High and Low Priority Arrivals," *International Journal of Digital and Analog Communication Systems*, Vol. 4, No. 3, 1991, pp. 209-224.
[doi:10.1002/dac.4510040305](https://doi.org/10.1002/dac.4510040305)
- [30] P. Papantoni-Kazakos, "Multiple Access Algorithms for a System with Mixed Traffic: High and Low Priority," *IEEE Transactions on Communication*, Vol. 40, No. 3, 1992, pp. 541-555.
[doi:10.1109/26.135724](https://doi.org/10.1109/26.135724)
- [31] P. Papantoni-Kazakos, H. Delic, M. Paterakis and M. Liu, "Transmission Algorithms for a Multi-Channel Packet Radio System with Priority Users," *International Journal of Digital and Analog Communication Systems (IJDACs)*, Vol. 6, No. 4, 1993, pp. 193-212.
- [32] P. Papantoni-Kazakos, N. Likhanov and B. S. Tsybakov, "A Protocol for Random Multiple Access of Packets with Mixed Priorities in Wireless Networks," *IEEE JSAC*, Vol. 13, No. 7, 1995, pp. 1324-1331.
- [33] J. Kurose, M. Schwartz and Y. Yemini, "Multiple Access Protocols and Time Constrained Communication," *ACM Computing Surveys*, Vol. 16, No. 1, 1984, pp. 43-70.
- [34] G. D. Marcus and P. Papantoni-Kazakos, "Dynamic Scheduling Protocols for a Multiple Access Channel," *IEEE Transactions on Communication*, Vol. 31, No. 9, 1983, pp. 1046-1054.
[doi:10.1109/TCOM.1983.1095934](https://doi.org/10.1109/TCOM.1983.1095934)
- [35] M. Paterakis, L. Georgiadis and P. Papantoni-Kazakos, "A Full Sensing Window Random-Access Algorithm for Messages with Strict Delay Constraints," *Algorithmica*, Vol. 4, No. 3, 1989, pp. 313-328.
[doi:10.1007/BF01553894](https://doi.org/10.1007/BF01553894)
- [36] D. F. Lyons and P. Papantoni-Kazakos, "A Window Random Access Algorithm for Environments with Capture," *IEEE Transactions on Communication*, Vol. 37, No. 7, 1989, pp. 766-770.
[doi:10.1109/26.31169](https://doi.org/10.1109/26.31169)
- [37] C. Bisdikian, L. Merakos and L. Georgiadis, "Stability Analysis of Interconnected Single-Hop Random-Access Networks," *IEEE Transactions on Communications*, Vol. 40, No. 3, 1992, pp. 556-567.
[doi:10.1109/26.135725](https://doi.org/10.1109/26.135725)
- [38] R. L. Hamilton and E. J. Coyle, "A Two-Hop Packet Radio Network with Product Form Solution," *Proceedings of the 20th Annual Conference on Information Sciences and Systems*, Princeton, 1986, pp. 871-876.
- [39] M. Sidi and I. Cidon, "A Multi-Station Packet Radio Network," *Performance Evaluation*, Vol. 8, No. 1, 1988, pp. 65-72.
[doi:10.1016/0166-5316\(88\)90013-2](https://doi.org/10.1016/0166-5316(88)90013-2)
- [40] B. Eklundh, "Channel Utilization and Blocking Probability in a Cellular Mobile Telephone System with Direct Retry," *IEEE Transactions on Communications*, Vol. 34, No. 4, 1986, pp. 329-337.
[doi:10.1109/TCOM.1986.1096544](https://doi.org/10.1109/TCOM.1986.1096544)
- [41] Y. Yemini and L. Kleinrock, "An Optimal Adaptive Scheme for Multiple Access Broadcast Communications,"

Proceedings of International Conference on Communications, 1978, pp. 1-5.

- [42] J.-C. Huang and T. Berger, "Delay Analysis of 0.487 Contention resolution Algorithm," *IEEE Transactions on Communications*, Vol. 34, No. 9, 1986, pp. 916-926. [doi:10.1109/TCOM.1986.1096639](https://doi.org/10.1109/TCOM.1986.1096639)
- [43] L. Georgiadis, L. Merakos and P. Papantoni-Kazakos, "A Method for the Delay Analysis of Random Multiple Access Algorithms Whose Delay Process Is Regenerative," *IEEE Journal on Selected Areas in Communications*, Vol. 5, No. 6, 1987, pp. 1051-1062. [doi:10.1109/JSAC.1987.1146613](https://doi.org/10.1109/JSAC.1987.1146613)
- [44] M. Paterakis, L. Georgiadis and P. Papantoni-Kazakos, "On the Relation between the Finite and the Infinite Population Models for a Class of RAA's," *IEEE Transactions on Communications*, Vol. 35, No. 11, 1987, pp. 1239-1240. [doi:10.1109/TCOM.1987.1096696](https://doi.org/10.1109/TCOM.1987.1096696)
- [45] R. Gallager, "A Perspective on Multiaccess Channels," *IEEE Transactions on Information Theory*, Vol. 31, No. 2, 1985, pp. 124-142. [doi:10.1109/TIT.1985.1057022](https://doi.org/10.1109/TIT.1985.1057022)
- [46] A. Bar-David and M. Sidi, "Collision Resolution Algorithms in Multistation Packet Radio Networks," *IEEE Transactions on Communications*, Vol. 37, No. 12, 1989, pp. 1387-1391. [doi:10.1109/26.44212](https://doi.org/10.1109/26.44212)
- [47] G. Fayolle, P. Flajolet, M. Hofri and P. Jacquet, "The Evaluation of Packet Transmission Characteristics in a Multi-Access Collision Detection Channel with Stack Collision Resolution Protocol," Israel Institute of Technology, Haifa, 1983.
- [48] L. Merakos and L. Georgiadis, "Stability of Interconnected Random Access Networks," *Proceedings of the 25th IEEE Conference on Decision and Control*, Athens, 10-12 December 1986, pp. 1330-1332.
- [49] B. S. Tsybakov and V. L. Bakirov, "Packet Transmission in Radio Networks," *Problemy Peredachi Informatsii*, Vol. 21, No. 1, 1985, pp. 80-101.
- [50] B. S. Tsybakov and V. B. Fayngold, "A Non-Homogeneous Frame RMA Algorithm," *Problems of Information Transmission*, Vol. 25, No. 2, 1989, pp. 126-136.
- [51] R. Rom and M. Sidi, "Multiple Access Protocols: Performance and Analysis," Springer-Verlag, New York, 1990. [doi:10.1007/978-1-4612-3402-9](https://doi.org/10.1007/978-1-4612-3402-9)
- [52] D. Bertsekas and R. G. Gallager, "Data Networks," Prentice-Hall, Englewood Cliffs, 1987.
- [53] M. Kaplan, "A Sufficient Condition of Non-Ergodicity of a Markov Chain," *IEEE Transactions on Information Theory*, Vol. 25, No. 4, 1979, pp. 470-471. [doi:10.1109/TIT.1979.1056059](https://doi.org/10.1109/TIT.1979.1056059)
- [54] W. Szpankowski, "Stability Conditions for Multidimensional Queueing Systems with Computer Applications," *Operations Research*, Vol. 36, No. 6, 1988, pp. 944-957. [doi:10.1287/opre.36.6.944](https://doi.org/10.1287/opre.36.6.944)
- [55] W. Szpankowski and V. Rego, "Some Theorems on Instability with Applications to Multi-Access Protocols," *Operations Research*, Vol. 36, No. 6, 1988, pp. 958-966. [doi:10.1287/opre.36.6.958](https://doi.org/10.1287/opre.36.6.958)
- [56] A. G. Pakes, "Some Conditions for Ergodicity and Recurrence of Markov Chains," *Operations Research*, Vol. 17, No. 6, 1969, pp. 1058-1061. [doi:10.1287/opre.17.6.1058](https://doi.org/10.1287/opre.17.6.1058)
- [57] R. L. Tweedie, "Criteria for Classifying General Markov Chains," *Advances in Applied Probability*, Vol. 8, No. 4, 1976, pp. 737-771. [doi:10.2307/1425932](https://doi.org/10.2307/1425932)
- [58] J. W. Cohen, "On Regenerative Processes in Queueing Theory," Springer-Verlag, New York, 1976. [doi:10.1007/978-3-642-95281-4](https://doi.org/10.1007/978-3-642-95281-4)
- [59] L. V. Kantorovich and V. I. Krylov, "Approximate Methods of Higher Analysis," Interscience, New York, 1958.
- [60] S. Stidham Jr., "Regenerative Processes in the Theory of Queues with Applications to the Alternating Priority Queue," *Advances in Applied Probability*, Vol. 4, No. 3, 1972, pp. 542-577. [doi:10.2307/1425993](https://doi.org/10.2307/1425993)