Asynchronous Packet Localization with Random SPOTiT in Satellite Communications

Selma Zamoum [*†], Jérôme Lacan [†‡], Marie-Laure Boucheret [†§]
Jean-Baptiste Dupé [¶], Mathieu Gineste [‖]
[*] *TéSA,* [†] *University of Toulouse,* [‡] *ISAE-SUPAERO,* [§] *ENSEEIHT,* [¶] *CNES,* [‖] *Thales Alenia Space*
Toulouse, France
selma.zamoum@tesa.prd.fr, jerome.lacan@isae-supaero.fr, marie-laure.boucheret@enseeiht.fr
jean-baptiste.dupe@cnes.fr, mathieu.gineste@thalesaleniaspace.com

***Abstract*—Recently, many different Random Access protocols have been developed and proposed for satellite return link communications. Synchronous and asynchronous solutions vary, mainly, in terms of signaling overhead regarding synchronization information. On the one hand, Contention Resolution Diversity Slotted Aloha (CRDSA) has emerged as a leader technique for synchronous transmissions with multiple replicas per packet and Successive Interference Cancellation at reception. On the other hand, Asynchronous Contention Resolution Diversity ALOHA (ACRDA) has been proposed as an equivalent asynchronous method to CRDSA. CRDSA and ACRDA incur a deadlock when no more packets can be retrieved due to high channel loads. Therefore, a complementary method to CRDSA: MultireplicA decoding using corRelation baSed locAlisation (MARSALA) proposed to combine replicas belonging to the same undecoded packet after localizing them through correlations. This allows to unlock some of the deadlock configurations which would relaunch CRDSA again. In asynchronous transmissions, Enhanced Contention Resolution Aloha (ECRA) uses different combining techniques for packets replicas to offer high system performance in terms of Packet Loss Ratio (PLR) and throughput. The former and latter techniques MARSALA and ECRA can be costly in localization complexity to the receiver. Therefore, Shared Position Technique for Interfered Random Transmissions (R-SPOTiT) defines a way to reduce the complexity of MARSALA's packets localization without degrading performance nor adding extra signaling information. Accordingly, this paper proposes AR-SPOTiT, an asynchronous design of R-SPOTiT, as a complementary method to ACRDA that introduces a way to locate replicas on their virtual frames with less complexity and significantly higher system performance compared to ACRDA.**

***Index Terms*—Satellite communications, Asynchronous transmissions, Multiuser channel, Random access, packet localization.**

I. INTRODUCTION

The effective coverage of satellites and the technology behind have motivated many actors to develop efficient communications for Internet access, television and telephony. As a matter of fact, reservation resources of Demand assigned multiple access techniques have been largely deployed in satellites. However, these resources cannot follow the technological growth with big users communities in applications like the Internet of Things and Machine to Machine communications. Thus, access protocols based on Aloha [1] took over a big part of the Random Access (RA) research area. Contention Resolution Diversity Slotted Aloha (CRDSA) [2] that is based on a multi-replica transmission and Successive Interference Cancellation (SIC) at reception emerged as a leading technique. Later, a complementary method to CRDSA was proposed; Multireplica Decoding using Correlation based Localisation (MARSALA) [3] which is based on a computation of correlations between a reference time slot and the remaining signal on the frame. This makes it possible to locate and combine burst replicas though summation, in order to have a higher probability of decoding when CRDSA is blocked. In parallel, asynchronous schemes have been developed; the idea was to eliminate the coordination overhead between transmitters. Enhanced Contention Resolution Aloha (ECRA) [4] proposed to combine, in its initial version, the non-interfered portions or the least interfered ones of packet replicas of the same user. In addition, Asynchronous Contention Resolution Diversity Aloha (ACRDA) [5] introduced a novel scheme. A specific Virtual Frame (VF) is assigned to each user where he can transmit his replicas within virtual time slots. This synchronization free system between users maintains replicas of a same packet synchronized together within a VF. At reception, SIC operations are kept with a decoding process based on a sliding window. This concept along with the VFs have also been adopted in the new version of ECRA [6] [7]. Also, other combining techniques have been used such as Maximum Ratio Combining (MRC) and Equal Gain combining (EGC). This, of course, requires perfect knowledge of replicas locations. The latter has been addressed in [6] where the fact that timing offset of a packet at reception is the same for all its replicas has been exploited. Therefore the delay between burst replicas was considered as a multiple of a virtual time slot duration.

ACRDA is considered as an alternative solution to CRDSA with reduced loops due to its asynchronous property. A loop occurs when two or more users transmit their packets in the same time slot positions. On that basis, we believe ACRDA can also take advantage of the Shared Position Technique for Interfered Random Transmissions (R-SPOTiT) [8] contribution for better performance. As a matter of fact, The proposed Asynchronous Random SPOTiT (AR-SPOTiT) offers a way to localize replicas with a shared information that does not require signaling overhead between the receiver and each of the transmitters. Identification information are used as an entry seed to a PseudoRandom Number Generator (PRNG) to generate replicas positions and the preamble to use among a set pseudo-orthogonal codes. The main feature of SPOTiT is that the receiver knows the potential locations of all users, and thus the distance between burst replicas, before any is decoded, which is one of the differences between AR-SPOTiT and ECRA. Indeed, since all identifiers are well-known (Logon) at reception, the receiver is able to generate using the same PRNG all potential packets positions. On the other hand, in ECRA one of the replicas should be decoded in order to retrieve the pointers that the receiver exploits to find the time slot positions of the other replicas.

The remainder of this paper is as follows. The next section gives an overview of the considered satellite system that focuses on the return link; then a brief summary about Synchronous R-SPOTiT is presented in Section III. Asynchronous decoding characteristics and AR-SPOTiT are detailed afterwards in Section IV and Section V respectively. Finally, simulation results are presented in Section VI.

II. SYSTEM OVERVIEW

Let us consider a random return link channel via satellite and focus on packet reception of multi-access transmissions. N_{U} users transmit, asynchronously according to ACRDA, N_{R} replicas on different positions. Each burst is located within a specific virtual frame of $N_{\mathrm{S}} = 100$ virtual time slots, over the same frequency. This means that no signaling information between users is necessary. In a general way, at the physical layer level, the set of $N_{\mathrm{b}} = 100$ information bits of a user are turned into a MODCOD using QPSK modulation $M = 4$ and 3GPP turbo code of rate $K = 1/3$. The supposed equipowered packets are then formed by adding, at the beginning and at the end of the resulted payloads of $N_{\mathrm{sym}} = N_{\mathrm{b}}/K\log_2(M)$ symbols a preamble and a postamble respectively, then at a known location in the payload a signaling field regarding other replicas positions on the VF. Each replica of a given packet is placed on a VF time slot, according to two ACRDA modes: all positions are randomly selected on the VF; or having the first replica imposed to the first virtual time slot, and the other replicas randomly placed on the VF. The latter allows to have a reduced transmission delay for non critical loads but appears to be less significant at Transport Control Protocol (TCP) layer level as reported in [5]. The results show that both modes are though equivalent in terms of Packet Loss Ratio (PLR) and throughput. We have chosen the first mode for AR-SPOTiT.

We assume the channel model is an Additive White Gaussian Noise (AWGN) with a 10 dB Energy per Symbol to Noise power spectral density ratio (E_{S}/N_0). At the receiver side, ACRDA analyzes the memory for preamble detection through a sliding window SW and then performs SIC operations to all decodable packets until it is blocked. The preamble detection phase will be adapted to a set of pseudo-orthogonal codes. We fix the maximum SIC iterations at 15, window size at 3 times the VF size which is the same for all users and a window step of $0,15$ the VF size. Before the SW is shifted by a window step, AR-SPOTiT can intervene. As a matter of fact, this complementary treatment is triggered to resolve ACRDA's deadlock when no more packets can be retrieved on the current SW. Perfect estimation of channel parameters such as the timing offset, the phase shift and the frequency offset is assumed in this paper. Once AR-SPOTiT has localized replicas positions, they are combined and attempted to be decoded again. When AR-SPOTiT is blocked (no more solvable packets), the SW steps forward and ACRDA starts decoding again.

III. SYNCHRONOUS RANDOM SPOTIT

R-SPOTiT [8] has been proposed as an alternative solution to MARSALA that aims to solve CRDSA's deadlock. In the perspective of reducing packets data localization complexity, R-SPOTiT introduced a possibility of having the information about all potential packets positions available at the receiver side, without extra signaling information. The goal was to reduce the number of data correlation operations that are performed to localize packets replicas. Therefore, a PseudoRandom number generator (PRNG) with shared seeds is exploited at both the transmitter and the receiver in order to generate the time slots positions of users. The seed can be static using the hardware identifier (HID) or dynamic through a combination between the latter and the frame identifier. If multiple preambles are used, the PRNG can also be used to select the preamble code for each user. In order to generate this information, seeds are taken individually, at each terminal, as an entry to the PRNG that gives as output the time slot positions for each one of them and the used preamble if applied. The receiver, knowing all identifiers of terminals attached to it (thanks to the logon phase) and the received frame identifier, constructs an information table that includes all possible positions on the frame and the potential preambles that could be detected there. This way, when a reference time slot is selected, fewer correlations are made, rather than the $N_{\mathrm{S}} - 1$ correlations of MARSALA. Indeed, when a preamble is detected, these would refer to all replicas positions of potentially collided packets on the reference time slot that use the same detected preamble. In the case of a single preamble, correlations are made over all replicas positions of potentially collided packets on the reference time slot.

IV. ASYNCHRONOUS DECODING CHARACTERISTICS WITH REPLICAS COMBINATION

It has been shown in [9] that the approximation of the interference term to an AWGN channel is accurate when the number of colliding packets is high enough in an unfaded environment but remains imprecise when this number is weak. Here, we assume the Signal to Noise Ratio (SNR) is approximated to the Signal to Noise plus Interference Ratio (SNIR). Thus, in the same way as in [8] Packet Error Rate (PER) with respect to the SNIR of the used MODCOD is exploited to determine the packet decoding probability in terms of the number of total and partial interference.

In a rough way, a mean Interference SNIR $(SNIR_{\mathrm{MIS}}(u,r))$ can be determined to approximate the simulation results. The $(SNIR_{\mathrm{MIS}}(u,r))$ of a replica r belonging to a user u calculates the mean interference value between symbols $C_{u,r}$. ECRA uses the same type of SNIR but with a combined less interfered packet from all replicas. The average mutual information (MI) over a replica has also been taken into account in ECRA for interference modeling. However in this paper, we consider only the equivalent SNIR mean value of a packet as a preliminary study. The mean interference value $C_{u,r}$ over a replica r of user u is expressed as follows:

$$C_{u,r} = \frac{1}{N_{\mathrm{sym}}} \sum_{s=1}^{N_{\mathrm{sym}}} C_{\mathrm{sym}}^{u,r}(s) \tag{1}$$

With $C_{\mathrm{sym}}^{u,r}(s)$ the number of interfering packets at symbol s level (of replica r of user u), and N_{sym} the number of symbols per packet.

As stated before, synchronous R-SPOTiT is able to retrieve more packets when CRDSA fails but with less complexity than MARSALA. We suggest that AR-SPOTiT can use the same mechanism as in synchronous systems given that signal combination takes place within the virtual frame between replicas of the same user independently from the others. Consequently, when AR-SPOTiT intervenes, signal combination takes place after replicas localization. Hence, the power of the packet of interest is expected to become significant. For two replicas r_1 and r_2, $C_{u,r}$ becomes $C_{\mathrm{ARS}}(u) = C_{u,r_1} + C_{u,r_2}$ that represents the equivalent interference rate. The equivalent SNIR value experienced over both replicas positions $SNIR_{\mathrm{MIS}}(u)$ can be expressed according to C_{ARS} as follows:

$$\begin{aligned} SNIR_{\mathrm{MIS}}(u) &= f\left(C_{\mathrm{ARS}}(u)\right) \\ &= \frac{{N_{\mathrm{R}}}^2 \times E_{\mathrm{S}}/N_0}{E_{\mathrm{S}}/N_0 \times C_{\mathrm{ARS}}(u) + N_{\mathrm{R}}} \end{aligned} \tag{2}$$

Furthermore, each value of $SNIR_{\mathrm{MIS}}(u)$ is associated to a certain PER value such as:

$$PER(u) = g\left(SNIR_{\mathrm{MIS}}(u)\right) = g \circ f(C_{\mathrm{ARS}}(u)) \tag{3}$$

The function g depends on the used MODCOD. In our case, it corresponds to the interpolation of the $1/3$ turbo coded PER with QPSK modulation in an AWGN channel environment with $E_{\mathrm{S}}/N_0 = 10$ dB.

Knowing that the decoding probability is equal to $1-PER$, Fig. 1 shows the latter with respect to the interference rate for ACRDA and AR-SPOTiT with two replicas (AR-SPOTiT-2) and three replicas (AR-SPOTiT-3). A given value of the interference rate over all positions corresponds to $C_{\mathrm{ARS}}(u)$ while a normalized one refers to $C_r(u) = C_{\mathrm{ARS}}/N_{\mathrm{R}}$. The interference rate belongs to $\mathbb{R}^+$; for example a value of 2.5 means all interfered portions of a packet constitute a collision as long as two packets and a half. We can clearly see on the figure the considerable intake of replicas combination in AR-SPOTiT compared to ACRDA; the average number of tolerated interference per slot (normalized) is doubled in AR-SPOTiT with two replicas, and tripled with three replicas.

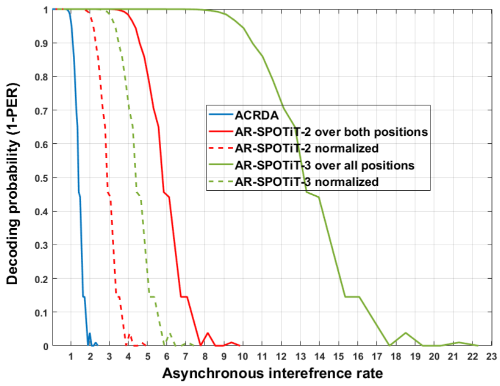


Fig. 1. Decoding probability with respect to the number of interference of 150 symbols, over an AWGN channel of $E_{\mathrm{S}}/N_0 = 10$ dB.

V. ASYNCHRONOUS RANDOM SPOTIT

The main contribution of AR-SPOTiT is to present a way to localize replicas at reception after ACRDA is blocked through a shared information between each transmitter and the receiver. The blocking situation of ACRDA means that the latter can no longer retrieve packets. Here is a description of transmission and reception operations in AR-SPOTiT.

A. *Transmission*

At transmission, the way to put each packet in its corresponding virtual frame is governed by AR-SPOTiT. Indeed, replicas positions are selected using a PRNG that has the terminal hardware identifier (HID) as an entry seed. The same seed is used to select one preamble among a set of pseudo-random codes. As a matter of fact, multiple preambles are considered in AR-SPOTiT. This reduces the complexity of the system as explained later in this section.

All seeds are static (replicas positions of each packet will always have the same positions at each new VF) because there is no virtual frame identification that could serve a dynamic seed, but this should not be an issue as the probability of having a repetitive loop remains very low due to the asynchronous nature of the system. Especially because the timing offset between the transmitted virtual frames remains random as there is no synchronization between users.

B. *Reception*

An information table is constructed by the AR-SPOTiT receiver in which the distances between replicas of a same users u are included, in terms of the number of virtual slots between them. These distances are computed by the receiver after the replicas positions are derived using the PRNG with the HID seeds available at reception.

There are $N_{\mathrm{d}} = N_{\mathrm{R}}(N_{\mathrm{R}}-1)/2$ distances for N_{R} replicas belonging the same packet (same VF). To compute these distances, first the virtual time slot positions $R_a \in [0; N_{\mathrm{S}}-1]$ of the N_{R} replicas are sorted in an ascending order $a \in [1 : N_{\mathrm{R}}]$, thus $R_1 < R_2 ... < R_{N_R}$. Then, subtraction operations are performed between all sorted positions as follows:

$$\{d_{u,j}\} = \begin{cases} d_{u,1} = R_2 - R_1 - 1, d_{u,2} = R_3 - R_2 - 1, \\ d_{u,3} = R_3 - R_1 - 1, \ldots\ldots\ldots\ldots\ldots\ldots, \\ d_{u,N_d - N_R + 2} = R_{N_\mathrm{R}} - R_{N_\mathrm{R}-1} - 1, \\ d_{u,N_d - N_R + 3} = R_{N_\mathrm{R}} - R_{N_\mathrm{R}-2} - 1, \ldots, \\ d_{u,N_d} = R_{N_\mathrm{R}} - R_1 - 1 \end{cases} \tag{4}$$

With $d_{u,j} \in [0; N_\mathrm{S} - 1]$ the j^{th} (with $j \in [1; N_\mathrm{d}]$) distance between two replicas of a packet belonging to user u. An AR-SPOTiT information table with three replicas per packet ($N_\mathrm{d} = 3$) and multiple preambles is presented in Table I, with P_u the preamble associated to user u. AR-SPOTiT decoding mechanism can take place according to two options: it can first exploit the resulting ACRDA information when a preamble is detected but the packet is not decodable due to the high interference level; or rely on power detection to reveal a whole packet presence when the preamble detection fails in determining a clear presence of a preamble. Let us consider in this paper that, even after ACRDA is blocked, preambles are correctly detected, and thus we know the beginning of the virtual time slot.

TABLE I
AR-SPOTiT INFORMATION TABLE

Users	**Distances between positions in slots**			**Preambles**
$\mathbf{U_1}$	$d_{1,1}$	$d_{1,2}$	$d_{1,3}$	P_{U_1}
$\mathbf{U_2}$	$d_{2,1}$	$d_{2,2}$	$d_{2,3}$	P_{U_2}
......	......	......	......	......
$\mathbf{U_{N_U}}$	$d_{N_\mathrm{U},1}$	$d_{N_\mathrm{U},2}$	$d_{N_\mathrm{U},3}$	$P_{U_{N_\mathrm{U}}}$

The localization process can now take place, and the correlations meant to locate replicas positions are hence performed. They are made at distances, in slots obtained from the information table, to replicas positions of packets with the detected preamble. In other words, the reference time slot used in synchronous MARSALA and R-SPOTiT is always, in AR-SPOTiT, set as the virtual time slot where a preamble is correctly detected. One can deduce that due to the fact that the virtual frames are independent one towards another. Having a single preamble will lead to make localization correlations, at the worst case, over the next $N_\mathrm{S} - 1$ slots from where the single preamble has been detected. Furthermore, we set the correlations to be made in both directions, at distances before and after the virtual reference time slot because before decoding the receiver has no knowledge on which replica's preamble is detected. Indeed, an a^{th} replica can be detected instead of first detecting an a^{th-1} replica. In this case a correlation is necessary for previous virtual time slot positions. This is valid as long as the $d_{u,j}$ is smaller than the distance between the virtual time slot of reference and the beginning of SW, respectively end of SW. Otherwise the correlations regarding a certain distance are made on a single direction. Therefore, using a single preamble that encounters $(2 \times N_\mathrm{S}) - 2$ correlations at its worst case should be too complex to be considered in AR-SPOTiT. During preamble detection, as and when a preamble is discovered, AR-SPOTiT will perform localization correlations on specific virtual time slots derived from the replicas distances on the information table. SIC operations will afterwards take place to eliminate interference. When no more packets can be retrieved the SW will move with one step at a time, and the mechanism of AR-SPOTiT with ACRDA is triggered again, until it reaches the end of the memory.

AR-SPOTiT mechanism with a single preamble can be approached to the 2 phases ECRA [7] but it benefits from additional information . Indeed, the first SIC phase of ECRA is performed by ACRDA (first phase in AR-SPOTiT), and its second step that is mainly based on Selection Combining, EGC or MRC is handled by AR-SPOTiT localization and combining according to the receiver's information table.

C. Complexity case study

In this section, an overview of the localization complexity regarding detection correlations is considered with a proposed refinement of AR-SPOTiT and ECRA. The main idea is to evaluate the impact, on replicas localization, when having a shared information between each terminal and the receiver along with the usage of multiple preambles. Two replicas per packet are used. Each detected preamble is assumed to belong to the first replica which is on the first virtual slot of the VF. Thus, the second replica search is only performed on one direction towards the right side for both algorithms AR-SPOTiT and ECRA.

Let us assume having a correct energy detection to search for a packet-like entity at AR-SPOTiT and ECRA combining phase after ACRDA and ECRA's SIC phase have failed in retrieving more packets. At a starting position of a detected entity, preamble search can be performed with a unique code word used in an ECRA-like algorithm and N_P preambles in AR-SPOTiT. In the latter case, the first preamble to be detected above a given threshold will be taken into account for packet decoding. In other words, one correlation is made in ECRA and $\frac{N_\mathrm{P}}{2}$ as a mean number of preamble correlation value is taken for AR-SPOTiT. At this point, replicas localization of the packet of interest is necessary. On the one hand, ECRA will proceed to correlation at the next virtual time slots starting from the detected preamble, considering that replicas of the same user have the same timing offset on a VF. An assumption can be made here to stop replica search at the first detected preamble spaced by an integer number of virtual time slots before going through the whole frame duration, thus a mean value of $\frac{N_\mathrm{S}}{2}$ correlations is considered. This assumption is particularly relevant in asynchronous transmissions because not only frames start at different random times, but each packet encounters a random timing offset. Therefore, the probability of having two packets with exactly the same starting position is unlikely to happen. On the other hand, AR-SPOTiT is able to define a number of specific virtual time slots N_s^p at distances derived from its information table where to perform preamble search of the detected code. This actually depends on the overall number of users attached to the gateway N_{U} and the number of preambles N_{P}. Furthermore, AR-SPOTiT can also benefit from the assumption of stopping the preamble search at the first position where the same preamble is found, especially because an integer number of time slots as a distance between both replicas is implicitly taken into account in the algorithm of AR-SPOTiT. Consequently, a mean value of the number of preamble detection correlations is $N_s^p = \frac{N_{\mathrm{U}}}{2N_{\mathrm{P}}}$ after having initially made $N_{\mathrm{P}}/2$ correlations. The total is then $N_s^p + N_{\mathrm{P}}/2 = (N_{\mathrm{P}}^2 + N_{\mathrm{U}})/2N_{\mathrm{P}}$.

Besides, since the number of users registered at a given gateway N_{U} is known, the latter can reduce the number of preambles to be used by transmitters to a single one. As a matter of fact, when the estimated $(N_{\mathrm{P}}^2 + N_{\mathrm{U}})/2N_{\mathrm{P}}$ for a given N_{P} and a given N_{U} exceeds $N_{\mathrm{S}}/2$, N_P can be set to 1. When a single preamble is used in AR-SPOTiT, all slots should be tested, but the first one to be compatible with the packet of interest will be taken into account. Thus, similarly, to ECRA, a mean value of $N_{\mathrm{S}}/2$ preamble correlations are performed.

To summarize, the total number of preamble correlations for one packet localization ν_{p} is expressed below in case of AR-SPOTiT and ECRA with two replicas per packet.

$$\nu_{\mathrm{p}} = \begin{cases} \frac{N_{\mathrm{S}}}{2} & \text{if ECRA} \\ \min\left(\frac{N_{\mathrm{P}}^2+N_{\mathrm{U}}}{2N_{\mathrm{P}}}, \frac{N_{\mathrm{S}}}{2}\right) & \text{if AR-SPOTiT} \end{cases} \tag{5}$$

Hereafter, we take a number of virtual slots per VF equal to 100 and 200 in order to compute the mean number of preamble detection correlations for ECRA as it only depends on N_{S}. In addition, different numbers of users $N_{\mathrm{U}} = \{1000, 2000, 4000, 8000\}$ and preambles $N_{\mathrm{P}} = \{31, 63, 127\}$ are taken to compute ν_p for AR-SPOTiT according to Eq. 5. The results are summarized in Fig. 2 when $N_{\mathrm{S}} = 100$ and in Fig. 3 when $N_{\mathrm{S}} = 200$. When the number of slots is equal to 100, AR-SPOTiT requires less preamble correlations to localize a packet's replica than ECRA. This is true when $N_{\mathrm{U}} = 1000$ and $N_{\mathrm{U}} = 2000$ and with a number of preambles equal to 31 and 63, otherwise The number of preambles will be reduced to a single one and then, ν_{p} for AR-SPOTiT will meet ECRA result. With 127 preambles, ECRA is less complex with any number of users, thus AR-SPOTiT meets its performance by using only one preamble. However, when the number of slots is equal to 200, AR-SPOTiT presents less preamble detection correlations meant to localize a packet's replica, compared to ECRA, for any number of users and preambles, except for the combination of 31 preambles with 8000 users. Therefore, a small number of slots is preferable to use when applying ECRA, and a number of preambles of 63 appears to be a good choice for AR-SPOTiT.

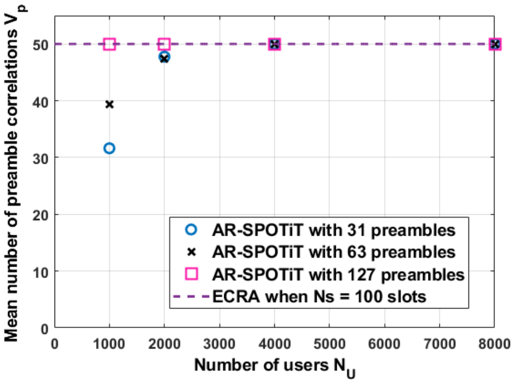


Fig. 2. Mean number of preamble detection correlations meant to localize a packet's second replica with 100 time slots per VF.

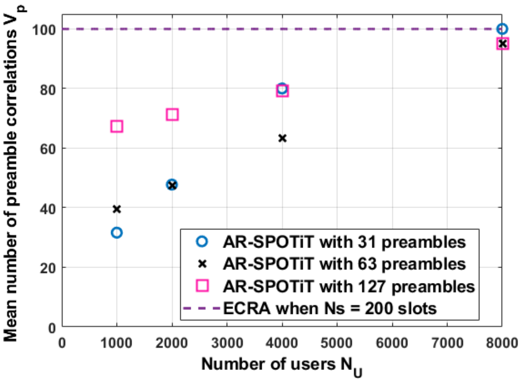
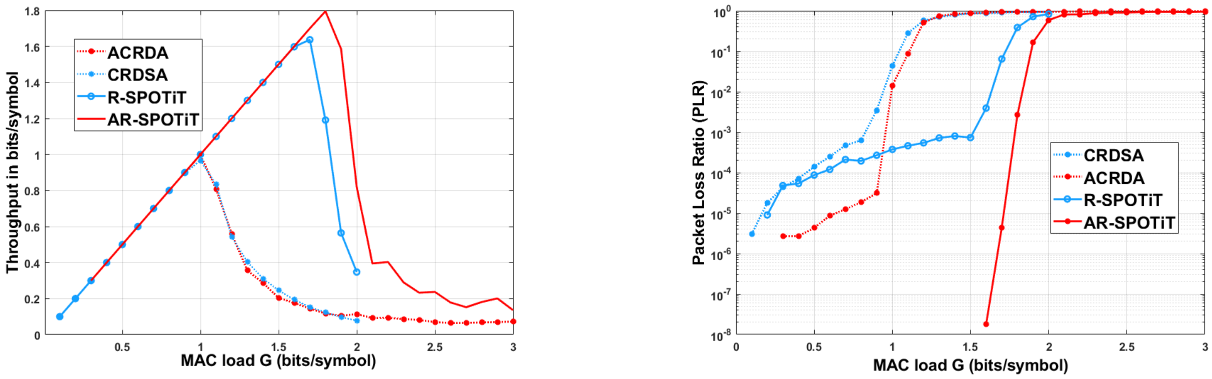


Fig. 3. Mean number of preamble detection correlations meant to localize a packet's second replica with 200 time slots per VF.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

The considered scenarios in this section use the parameters presented in Section II. We compare R-SPOTiT with AR-SPOTiT, in terms of PLR and throughput. In R-SPOTiT, preamble search operations are performed at each slot during CRDSA. The symbols over which these operations are made depend on the timing offset and the guard interval. As a matter of fact, in this synchronous slotted case, the beginning of each slot and the maximum timing offset are known. However in AR-SPOTiT, preamble research, using correlators, is made along the sliding window, over the whole memory until a preamble is found during ACRDA. This can make the receiver's complexity significant if the number of preambles is too big. For this reason, it is preferable to keep a small number of pseudo-orthogonal codes. We used here 63 Gold preambles according to the previous subsection. Also, we have chosen in our scenario $N_{\mathrm{R}} = 2$ replicas per packet because it is less complex in terms of association correlations after the first replica is localized. Moreover, as asynchronous transmissions mitigate data loops between packets, the PLR error floor that CRDSA experiences in low loads is lower in ACRDA when the number of replicas is equal to 2 (see Fig. 4.(b)). Indeed, with a channel load that is between 0.1 bits/symbol and 0.8 bits/symbol, CRDSA experiences an error floor that goes, approximately, from 3×10^{-6} to 6×10^{-4} respectively. For the same load, ACRDA presents an error floor that goes from 3×10^{-6} to 2×10^{-5}. On the one hand, we can notice (Fig. 4.(a)) that starting from a channel load of 1.6 bits/symbol,



(a) Throughput in bits per symbol

(b) Packet Loss Ratio

Fig. 4. AR-SPOTiT, R-SPOTiT, ACRDA and CRDSA performance, 100 information bits, QPSK modulation, Turbo code of rate $1/3$, AWGN channel and $E_S/N_0 = 10$ dB.

the throughput is higher with AR-SPOTiT compared to R-SPOTiT that reaches its maximum of 1.64 bits/symbol at a channel load of 1.7 bits/symbol. Therefore, AR-SPOTiT is preferable in high loads. Furthermore, we can observe on the same figure that AR-SPOTiT significantly enhances the throughput when coupled to ACRDA; 1.8 bits/symbol reached at a channel load of 1.8 bits/symbol approximately versus 1 bit/symbol at a load of 1 bits/symbol when ACRDA is used alone. On the other hand, AR-SPOTiT significantly reduces the PLR compared to R-SPOTiT (Fig. 4.(b)) in addition to the disappearance of the error floor. It attains approximately 4.4×10^{-6} at a channel load of 1.7 bits/symbol unlike R-SPOTiT that reaches 6×10^{-2} at the same load, which is above the usual PLR target for satellites return link set to 10^{-3}. An asynchronous scheme offering better results than a synchronous one is mainly due to the type of interference that is partial. Nevertheless, a compromise that takes into account the overall complexity and system performance has to be set according to applications needs.

VII. Conclusion and perspectives

Different RA methods and variants have been proposed for satellite communications. In this paper, we presented an asynchronous version of R-SPOTiT as a complementary process to ACRDA with better performance. Indeed, while ACRDA reaches a throughput of 1 bits/symbol, 1.77 bits/symbol is attained with AR-SPOTiT, which is higher than the throughput reached by R-SPOTiT in addition to the disappearance of the error floor of the PLR. The accomplishment of an asynchronous version of R-SPOTiT, which was originally designed to reduce MARSALA's complexity, allowed us to introduce a new way to localize replicas as a second option along with ECRA. According to Section V-C results and parameters, AR-SPOTiT significantly reduces the preamble detection correlations. An overall complexity analysis that includes the whole packet data correlations should be done in future work. Also, MRC combining technique assessed by ECRA in an asynchronous environment and MARSALA for synchronous transmissions showed a significant enhancement of the throughput. Furthermore, packet power unbalance using a half normal distribution of MARSALA [10] presented the best results when coupled to MRC. Therefore, we think applying MRC to AR-SPOTiT with packet power unbalance is expected to be considerably beneficial to the throughput.

References

[1] N. Abramson, "The ALOHA System - Another Alternative for Computer Communications," in *Fall Joint Computer Conference*, vol. 37. AFIPS Press, December 1970, pp. 281–285.

[2] E. Casini, R. De Gaudenzi , and O. Del Rio Herrero, "Contention Resolution Diversity Slotted ALOHA (CRDSA): An Enhanced Random Access Scheme for Satellite Access Packet Networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 4, pp. 1408–1419, April 2007.

[3] H.-C. Bui, K. Zidane, J. Lacan, and M.-L. Boucheret, "A Multi-Replica Decoding Technique for Contention Resolution Diversity Slotted Aloha," in *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*, September 2015.

[4] F. Clazzer and C. Kissling, "Enhanced Contention Resolution Aloha - Ecra," in *SCC 2013; 9th International ITG Conference on Systems, Communication and Coding*, January 2013.

[5] R. De Gaudenzi, O. del Río Herrero, G. Acar, and E. Garrido Barrabés, "Asynchronous Contention Resolution Diversity Aloha: Making CRDSA Truly Asynchronous," *IEEE Transactions on Wireless Communications*, vol. 13, no. 11, pp. 6193–6206, Nov 2014.

[6] F. Clazzer, F. Lazaro, G. Liva, and M. Marchese, "Detection and Combining Techniques for Asynchronous Random Access with Time Diversity," in *SCC 2017; 11th International ITG Conference on Systems, Communications and Coding*, Feb 2017, pp. 1–6.

[7] F. Clazzer, C. Kissling, and M. Marchese, "Enhancing Contention Resolution ALOHA Using Combining Techniques," *IEEE Transactions on Communications*, vol. 66, no. 6, pp. 2576–2587, June 2018.

[8] S. Zamoum, J. Lacan, M.-L. Boucheret, J.-B. Dupe, and M. Gineste, "Shared Position Technique for Interfered Random Transmissions in Satellite Communications," in *2018 9th Advanced Satellite Multimedia Systems Conference and the 15th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, Sepember 2018.

[9] A. Giorgetti and M. Chiani, "Influence of Fading on the Gaussian Approximation for BPSK and QPSK with Asynchronous Cochannel Interference," *IEEE Transactions on Wireless Communications*, vol. 4, no. 2, pp. 384–389, March 2005.

[10] K. Zidane, J. Lacan, M.-L. Gineste, C. Bes, and C. Bui, "Enhancement of MARSALA Random Access with Coding Schemes, Power Distributions and Maximum Ratio Combining," in *2016 8th Advanced Satellite Multimedia Systems Conference and the 14th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, Sep. 2016.