Enhanced Contention Resolution Aloha - ECRA

Federico Clazzer and Christian Kissling
German Aerospace Centre (DLR)
Oberpfaffenhofen, D-82234, Wessling, Germany
Email: {federico.clazzer, christian.kissling}@dlr.de

***Abstract*—Random Access (RA) Medium Access (MAC) protocols are simple and effective when the nature of the traffic is unpredictable and random. In the following paper, a novel RA protocol called Enhanced Contention Resolution ALOHA (ECRA) is presented. This evolution, based on the previous Contention Resolution ALOHA (CRA) protocol, exploits the nature of the interference in unslotted Aloha-like channels for trying to resolve most of the partial collision that can occur there. In the paper, the idea behind ECRA is presented together with numerical simulations and a mathematical analysis of its performance gain. It is shown that relevant performance increases in both throughput and Packet Error Rate (PER) can be reached by ECRA with respect to CRA. A comparison with Contention Resolution Diversity Slotted ALOHA (CRDSA) is also provided.**

I. Introduction

In the recent past RA MAC protocols are attracting increasing interest in many different fields, from car-to-car communication to underwater sensor networks, just to mention a few. RA protocols are especially suitable for all the scenarios where the traffic is unpredictable and completely random or in cases where only small data volumes and urgent data need to be transmitted and Demand Assigned Multiple Access (DAMA) would cause delay and signalling overhead.

The current RA protocols have evolved significantly from the original idea of Aloha proposed by Abramson in 1970 [1]. First the well known slotted evolution of Aloha has been presented and analyzed by Roberts [2] few years later.

In the last years CRDSA [3] and CRA [4] have been presented as very promising evolutions of respectively Slotted Aloha and Aloha. CRDSA, is an evolution of the Slotted Aloha scheme and in particular of Diversity Slotted ALOHA (DSA) proposed in [5]. DSA provides a lower delay and higher throughput than Slotted ALOHA (SA) under very moderate loading conditions by transmitting twice the same packet in a different Time Division Multiple Access (TDMA) slot, or a different frequency and time slot in case of Multi-Frequency TDMA. However, the throughput difference between Aloha and Slotted Aloha or DSA was limited and quite poor in absolute terms.

CRDSA takes from DSA the idea to send more than one packet instance per user for each frame. The original CRDSA protocol generates two replicas of the same packet at random times within the frame instead of only one as in SA. While the driver for DSA is to slightly enhance the SA performance by increasing the probability of successful reception of one of the replicas (at the expense of increased random access load), CRDSA in addition is designed in a way to resolve most of DSA packet contentions by using Successive Interference Cancellation (SIC). Packet collisions are cleared up through an effective SIC approach that uses frame composition information from the replica packets. The key idea of CRDSA is to provide in each replica the signaling information of where the other replicas of the corresponding user are placed in the frame. Every time a packet is recovered, this information can be used for removing the signal contribution of the other replica from the frame, thus possibly removing its interference contribution to other packets. The main CRDSA advantages compared to Slotted Aloha lie in an improved Packet Loss Rates (PLR) and a much higher operational throughput.

The CRA protocol exploits the same approach of using SIC as CRDSA, but in an Aloha-like MAC protocol. Unlike the slotted schemes, here no slots are present in the frame and thus the replicas of the users can be placed within the frame without constraints, except that replicas of a user may not interfere each other. The avoidance of slots results in significant advantages such as relaxation in synchronization requirements among users and possibility of varying packet length without padding overhead. Forward Error Correction (FEC) in CRA is, unlike in CRDSA, beneficial also when no power unbalance among users is present because partial interference is not only possible but also more probable than complete interference.

The Irregular Repetition slotted ALOHA (IRSA) [6] protocol evolution is a bipartite graph optimization of CRDSA, where the number of replicas for each user is not fixed but is taken from a probability distribution for maximizing the throughput. It was shown in [6] that the distribution can be optimized to either maximize the throughput or to minimize the PLR.

The performance evaluations within [7]-[8] have shown that the maximum throughput of CRDSA (normalized to slots) can be impressively extended from $T_{SA} = 0.36$ (where T_{SA} is the normalized throughput of Slotted Aloha), up to $T_{CRDSA} \cong 0.55$ and even up to $T_{CRDSA++} \cong 0.68$ when 4 replicas per user are sent. With the IRSA approach a maximum theoretical throughput of $T_{IRSA} = 0.97$ can be achieved with a distribution obtained via differential evolution [6] containing 16 replicas per user at maximum . In CRA, assuming FEC $= 1/2$ and QPSK modulation, the maximum theoretical throughput shown with two replicas per user is $T_{CRA} = 0.96$ using the Shannon's capacity limit. Moreover, the PLR drops down to very low values. While SA meets a PER of 10^{-3} for a normalized offered traffic load $G = 10^{-3}\,Erl$, the CRDSA scheme can meet the same PLR for offered traffic load of $G = 5.5{\cdot}10^{-2}\,Erl$ and CRDSA++ even for $G = 0.6\,Erl$. The IRSA protocol attains a PLR of 10^{-3} for $G = 0.3\,Erl$ while the CRA protocol obtains the same PLR for $G = 0.6\,Erl$ with FEC $= 1/2$ and QPSK modulation.

II. Partial Interference and Loops

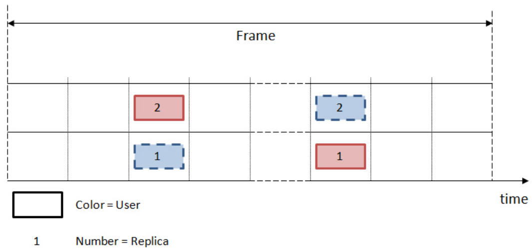
When slotted schemes like CRDSA are considered, for each packet sent in the frame only two cases are possible, either no interference between packets or entire overlapping. Moving to unslotted schemes like CRA different interference levels among packets are possible due to the elimination of the slots and random starting times. In this case, an entire overlapping between packets is only one possible interference scenario.

However, the SIC process can get stuck in situations where the replicas of two or several users interfere with each other in a way that none of the replicas can be recovered and thus all involved packets are lost. Such a case is denoted as a *loop*. While the probability of having such *loops* decreases with increasing frame length, practical frame lengths have a non negligible probability of *loops*.

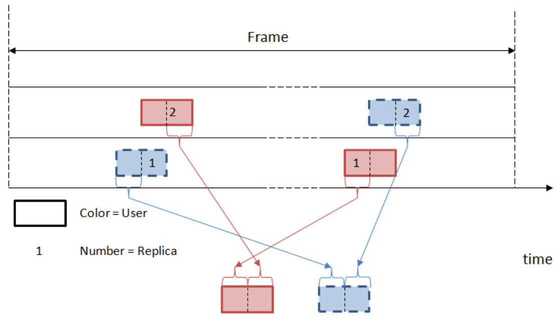
In Fig. 1(a) the simplest *loop* that can occur in case of CRDSA is presented. In the situation shown, if the degree d is equal to 2, both users cannot be decoded since both replicas of each user are fully interfered by the replica of the other user.

When CRA is considered, the interference might not be completely destructive for the users involved. In fact, since no slots exist here, partial interference among users can occur and is more probable than a complete interference. If the interference experienced by a replica is sufficiently small and the error correction code is strong enough, the packet can still be correctly decoded. However there are wide number of cases where this is not possible, in particular if the interference power is too high. E.g., if we consider the scenario presented in Fig. 1(b) and we suppose that the interference power is too high for being corrected by the FEC code, then in CRA both replicas of the user cannot be correctly decoded although different parts of the two replicas are affected by interference.

If it would be possible to combine the uninterfered symbols of the replicas of a users into a new packet, then the new combined packet might be decoded successfully and unlock the loop. In this case the two users which could not be decoded in CRA can now be correctly decoded and removed from the frame. In the case presented in Fig. 1(b), if we take the uninterfered first 50% of the blue dashed users first replica and the uninterfered second 50% of the second replica of the same user and we combine them creating a new packet, we could obtain a packet free of interference. The red user can then also be recovered using the same procedure and creating the combined packet. Creating a new combined packet from the replicas may however not work in all situations. If the same parts of all replicas of a user are interfered, the combined packet will not have a higher Signal to Noise and Interference ratio (SNIR) and the loop could not be resolved.



(a) CRDSA loop



(b) CRA loop

Fig. 1. CRDSA and CRA simplest loops

In [9] a similar scenario has been addressed, but the proposed solution exploits an iterative *chunk-by-chunk* decoding between the collided packets where decoding errors propagate, while in ECRA a combined packet is constructed and the decoding attempted on it in one step. Moreover, in ECRA the replicas generation is made regardless of the decoding success, while in [9] only the collided packets are replicated. [10] can be seen as the soft-decoding version of [9]. The decoding algorithm of [10] propagates the probabilities associated to the received symbols, instead of hard-decoding the symbols and use them for the back-substitution. Practical implementations issues arise in this second version, e.g. the enabling of bit permutations, how to access the soft information and finally the increase of complexity compared to [9].

III. ECRA decoding procedure

ECRA follows a two steps procedure for decoding the packets at the receiver side. At first the current frame is stored and the SIC is applied on the received packets. The SIC begins to scan the frame from the first received symbol and once it finds a packet it tries to decode it. If the decoding was successful, the content of the packet payload can be recovered. Since every packet contains information on the position of the current user replica(s), we can exploit this information for removing the other replica(s) from the frame.

In the following we assume ideal interference cancellation for this. If the decoding was not successful the packet remains in the frame and the interference contribution is not removed. Independently from the correct or incorrect decoding of the previous packet, the SIC pursues to scan the frame looking for the next packet. When the end of the frame is reached, either all the users packets have been correctly decoded or the replicas of at least two users are still not decoded and thus present in the frame. Hence, in the former the SIC procedure stops, while in the latter the SIC procedure tries to scan again the frame from the beginning. The SIC procedure is stopped if either all packets have been successfully decoded, if no further packets could be decoded in a round or if the maximum number of interference cancellation rounds has been reached.

The second step is the key novelty of the presented ECRA protocol. For each remaining user in the frame, the replicas sections without interference are taken and are used for creating a new combined packet for the considered user (see also Fig. 1(b)). If some portions of the user packets encounter interference in all the replicas, the replica symbols with the lowest interference are taken and exploited for creating the combined packet. This leads to create a user packet with the lowest possible interference. On the combined packet, the decoding is attempted and if the packet is correctly received, it is re-encoded modulated and removed in all the positions within the frame where the replicas were placed. Like in the first step, the mentioned procedure is iterated until either all the users can be correctly received or the maximum of possible iterations is reached.

It is possible to show that the ECRA approach can always generate a packet with higher, or at least equal, SNIR with respect to CRA. Given the packet of user u and replica r positioned within the frame and selected its symbol s, we can compute the interference contribution $I_{s,u,r}$ of other users replicas in symbol s to the replica r of user u:

$$I_{s,u,r} = \sum_{i=1, i\neq u}^{t_u} \sum_{d=1}^{deg} \delta_{id},\ with \begin{cases} \delta_{id} = 1 & \text{if replica } d \text{ of user } i \text{ has a symbol at position } s \text{ of user } u \text{, replica } r \\ \delta_{id} = 0 & \text{otherwise} \end{cases} \quad (1)$$

where t_u is the number of users in the frame and deg is the number of replicas per user. The interference ratio suffered by the given packet $x_{u,r}$ is then:

$$x_{u,r} = \frac{1}{t_s} \sum_{s=1}^{t_s} I_{s,u,r}$$

where t_s is the number of symbols in the packet of user u and replica r. Under the assumption of equal power conditions among users, the SNIR of user u and replica r gets:

$$SNIR_{u,r} = \frac{P}{x_{u,r} \cdot P + N} = \frac{SNR}{x_{u,r} \cdot SNR + 1},$$

with the transmission power P and noise power N.

For example, if the replica r of user u experiences an overall interference of 50%, then $x_{u,r} = 0.5$. Since the Signal-to-Noise ratio (SNR) of the packet is known, it is possible to evaluate the SNIR. Given $I_{s,u,r}$ from equation (1), for each replica $r \in \mathcal{D}_u$ with $\mathcal{D}_u$ the set of all the user u replicas, the ECRA protocol selects the symbol s with the lowest interference $I^*_{s,u,r}$ among all symbols at the same location in the current user replicas in $\mathcal{D}_u$

$$I^*_{s,u,r} = \min_{r \in \mathcal{D}} \{I_{s,u,r}\} \leq I_{s,u,r}, \quad r = 1, ..., deg.$$

The interference ratio suffered by the combined packet x^*_u created by ECRA is then:

$$x^*_u = \frac{1}{t_s} \sum_{s=1}^{t_s} I^*_{s,u,r} \leq \frac{1}{t_s} \sum_{s=1}^{t_s} I_{s,u,r} = x_{u,r}, \quad r = 1, ..., deg$$

Under the assumption of equal power conditions among users, the SNIR of the ECRA combined packet is:

$$SNIR^*_u = \frac{P}{x^*_u \cdot P + N} \geq \frac{P}{x_{u,r} \cdot P + N} = SNIR_{u,r}, \quad r = 1, ..., deg. \quad (2)$$

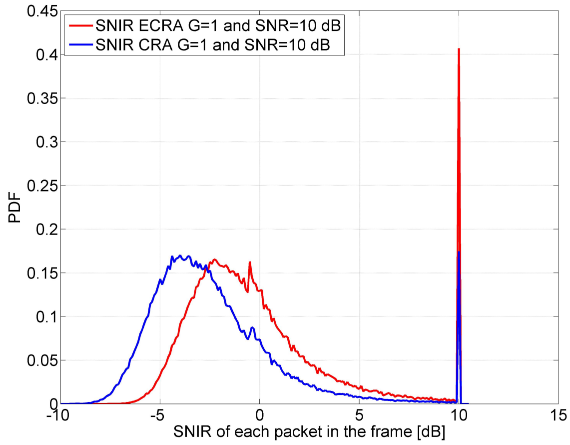


Fig. 2. SNIR Probability Density Function for ECRA and CRA

When each symbol with the lowest level of interference belongs to one single packet, the SNIR of ECRA coincides with the SNIR of CRA. In all the other cases we have $SNIR_{ECRA} > SNIR_{CRA}$. The result of equation (2) is confirmed by the simulations summarized in Fig. 2. The Probability Density Function (PDF) of CRA is shifted on the left of the graph with respect to the ECRA PDF. In other words, CRA shows a higher probability of low SNIR packets compared to ECRA. It can be noted that in both cases a peak in the PDF is found at $SNIR = 10$ dB, which corresponds to the packets free of interference. In fact, an SNR of 10 dB was selected for the simulations.

It is important to underline that the second step of ECRA, needs complete knowledge of the replicas position of the remaining users in the frame. Under this assumption, it is always possible to create the combined packet with the lowest level of interference, because the collided packets portions are known. For the knowledge of the frame composition in practical systems the replica location information stored in packet headers, which are protected with a very robust FEC, can be exploited. Very robust FEC applied to the headers can allow retrieving the information about replica locations although the packet itself is not decodable due to collisions. Investigations on the best position of the signaling info within the packet, on the advantage of replicating this info and/or the use of dedicated correction codes will be addressed in future work. It will be shown that compared to the case of perfect frame knowledge a smart positioning of this info within the packet results only in a minor performance degradation.

The first step of the ECRA protocol applies the SIC procedure exploited also in CRA while the second step increases the probability of correct decoding of the packets by generating new combined packets, but at the cost of increasing the complexity of the decoder. There are some situations where it may be more important to have less complex receivers even if they have the drawback of decreased performance, but in other scenarios it may be necessary to exploit the maximum performance. In the latter case, ECRA is a superior technique compared to CRA, when unslotted Aloha-like random access MAC protocols are considered.

IV. Numerical Results

Three different sets of simulations of ECRA are shown in the following section. The behavior of ECRA is analyzed in terms of SNR and in terms of the rate $R = R_c \cdot log_2(M)$, with code rate R_c and modulation index M. The comparison between the Shannon's capacity limit, called in the following Shannon Bound (SB), and Random Coding Bound (RCB) [11] is provided as well. In both cases, the co-user interference is assumed to be Gaussian distributed. It can be shown that this assumption is not far from reality due to phase-, time- and frequency offsets between the signals.

A. Shannon Bound

At first, the decoding threshold is approximated with the SB, assuming a Gaussian channel. The correct decoding of a given packet in this case, is only related with the quantity of interference plus noise that the packet experiences due to collision in the MAC frame and the noise given by the Additive White Gaussian Noise (AWGN) channel. Thanks to the Hartley-Shannon Theorem, it is commonly known that in an AWGN channel every rate R that accomplishes the relation $R \leq C/W = log_2(1+SNR)$, where C is the channel capacity and W the channel bandwidth, allows in theory an error free decoding. Thus the maximum allowable rate is $R = C/W = log_2(1+SNR)$. Moving from the SNR to the SNIR ratio, we find $R = C/W = log_2(1 + SNIR_{SHA})$, where $SNIR_{SHA}$ is the decoding threshold we are looking for. Elaborating the previous equation for extracting the $SNIR_{SHA}$ we get $SNIR_{SHA} = 2^R - 1$.

In ECRA, for each user $u \in \mathcal{U}$ where $\mathcal{U}$ is the set of users sending packets in a certain frame, the $SNIR_u^*$ for the combined packet of user u is given by equation (2). Each packet with $SNIR_u^* \geq SNIR_{SHA}$ is considered to be correctly decoded an its signal is removed from the frame as well as all the replicas of the corresponding user. Otherwise, the current packet remains in the frame.

B. Random Coding Bound

Moving from the theoretical limit given by the SB which is not reachable in practice, to a more realistic one, leads to consideration of the RCB. The RCB represents the upper bound on the average block error probability for codes of n symbols and for a given rate R. Since the RCB considers the average error probability of a set of codes, we are ensured that at least one code can reach this probability or less [11].

For the simulations, given the rate R, the RCB PER over the SNR curve is generated. The PER curve is then used as probability to correctly decode any given packet with its corresponding SNIR.

C. Simulations

The performed simulations show the average throughput $\overline{T}$ and the average packet error rate $\overline{PER}$ for a set of traffic offered load values G.

The first set of simulations provided are done for a rate $R = 2$ adopting the SB as decoding threshold. The considered scenario is characterized by a nominal $SNR = 10\,\text{dB}$ equal for each user generating traffic, the frame duration is selected to $T_f = 100\,\text{ms}$ and the symbol duration to $T_s = 1\,\mu\text{s}$. Moreover, the packet length $L_p = 1000$ bits is equal for every user, the number of replicas sent within the frame by any given user is $d = 2$ and the maximum number of SIC iterations for the three RA schemes is $I_{max} = 10$. The rate $R = 2$ leads to a packet length, in symbols, $L_s = L_p/R = 1000/2 = 500$ symbols. For any given value of G, $\overline{T}$ and $\overline{PER}$ are averaged over $N_f = 1000\, frames$.

We can suppose for example that FEC is adopted and the implemented encoder uses a code rate $R_c = 1/2$. In this case, the modulation index must be $M = 16$ to result in a rate $R = 2$, which corresponds to a 16-QAM modulation.

Therefore, the normalized traffic load G is given by:

$$G = \frac{N_u \cdot L_p \cdot T_s}{T_f \cdot R},$$

with N_u the number of users sending packets in the frame. The average throughput $\overline{T}$ is defined as the probability of successful reception of a packet, multiplied by the offered traffic load G. The average throughput here is related to the logical throughput, i.e. user packets, whereas the physical throughput would also consider the number of replicas generated per packet. The average packet error rate $\overline{PER}$, is evaluated as:

$$\overline{PER} = \frac{P_{err}}{N_u \cdot N_f}$$

where P_{err} is the number of lost packets at the receiver side, and N_f is the number of simulated frames for the corresponding G. Since the $\overline{PER}$ represents the average

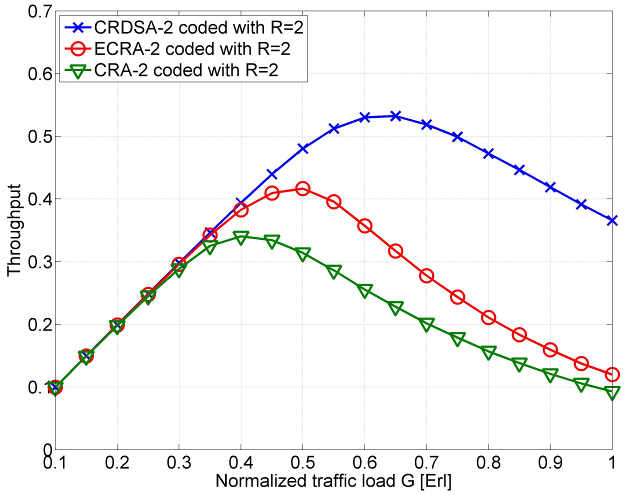


Fig. 3. CRDSA, CRA and ECRA throughput comparison for rate $R = 2$, $SNR = 10\,\text{dB}$ and SB

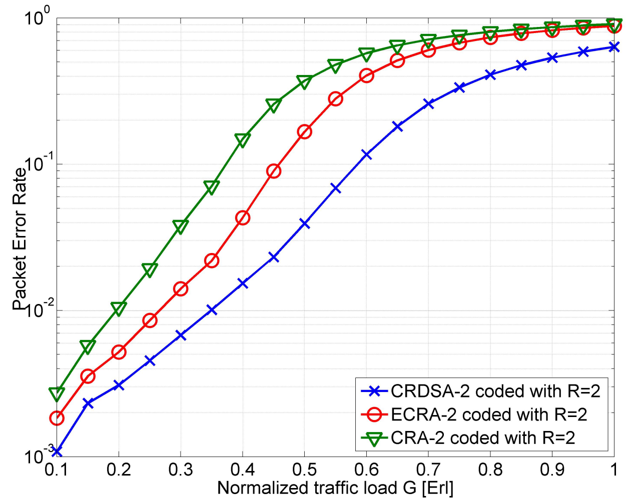


Fig. 4. CRDSA, CRA and ECRA packet error rate comparison for rate $R = 2$, $SNR = 10\,\text{dB}$ and SB

probability of a packet error, $\overline{T}$ is computed in the following way:

$$\overline{T} = (1 - \overline{PER}) \cdot G.$$

For simplicity of notation, the average $\overline{PER}$ and $\overline{T}$ are denoted as PER and T in the following.

In Fig. 3 the throughput comparison of ECRA, CRA and CRDSA-2 in the scenario discussed above is presented. The maximum throughput reached by CRA is $T_{max-CRA} \cong 0.34$ at $G = 0.4\,\text{Erl}$, while ECRA shows a maximum throughput of $T_{max-ECRA} \cong 0.42$ at $G = 0.5\,\text{Erl}$. The percentage of maximum throughput increase from CRA to ECRA is roughly 23%, which is a significant improvement. Finally, CRDSA[1] in the same conditions is able to reach a maximum throughput $T_{max-CRDSA} \cong 0.53$ at $G = 0.65\,\text{Erl}$. The ECRA RA scheme achieves a throughput in between the one of CRA and CRDSA in the region of positive slope while it shows a behavior more similar to CRA in the negative slope region. It is important to recall that the better performance of CRDSA w.r.t. to both CRA and ECRA comes at the expense of stronger synchronization constraints at the users.

In Fig. 4 the PER behavior of the three RA schemes over G is presented. The colors and symbols are the same as used in Fig. 3. For small to average values of G, ECRA PER shows a significant improvement compared to CRA but it is still worse than the CRDSA slotted scheme. Above the value of $G = 0.55\,\text{Erl}$, the ECRA and CRA PER curves tend to converge. The minimum PER of the three schemes is for CRA, $PER_{min-CRA} \cong 3 \cdot 10^{-3}$; for ECRA, $PER_{min-ECRA} \cong 2 \cdot 10^{-3}$; and for CRDSA, $PER_{min-CRDSA} \cong 1 \cdot 10^{-3}$, at $G = 0.1\,\text{Erl}$ for all the schemes.

[1]CRDSA-2 is shown here as the most basic representant of slotted SIC schemes. It should be noted that higher order CRDSA can achieve better performance than CRDSA-2

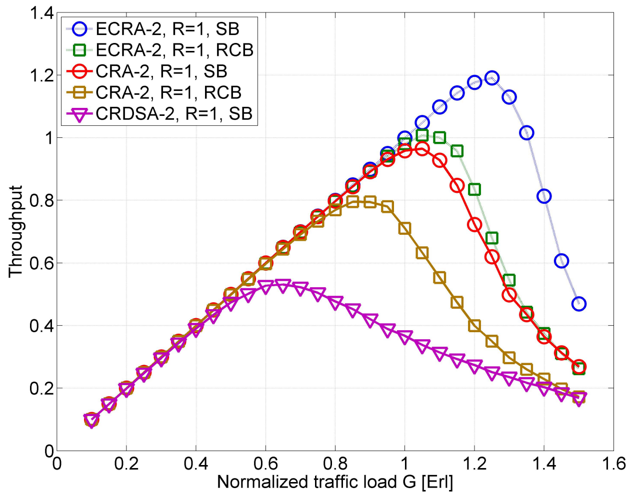


Fig. 5. CRDSA, CRA and ECRA throughput comparison for rate $R = 1$ and $SNR = 10\,\text{dB}$

The second set of simulations provided are done for a rate $R = 1$ comparing the SB as decoding threshold with the RCB. All the other simulations parameters are the same as explained before. In Fig. 5 the throughput of ECRA, CRA and CRDSA[2] are compared. ECRA with the SB as decoding threshold reaches the maximum throughput $T_{max-ECRA} = 1.19$ at $G = 1.25\,\text{Erl}$ outperforming both CRA and CRDSA-2, while ECRA with the RCB reaches $T_{max-ECRA} = 1.01$ at $G = 1.1\,\text{Erl}$. It is interesting to observe that the throughput increase of ECRA with respect to CRA with the SB is 23%, while it becomes 26% when the RCB is considered, confirming

[2]In CRDSA, $R = 1$ is not the best choice from a spectral efficiency point of view, if the $SNR = 10\,dB$. Since we are not interested in maximizing the spectral efficiency, the same rate is used for all the considered schemes to have equal conditions.

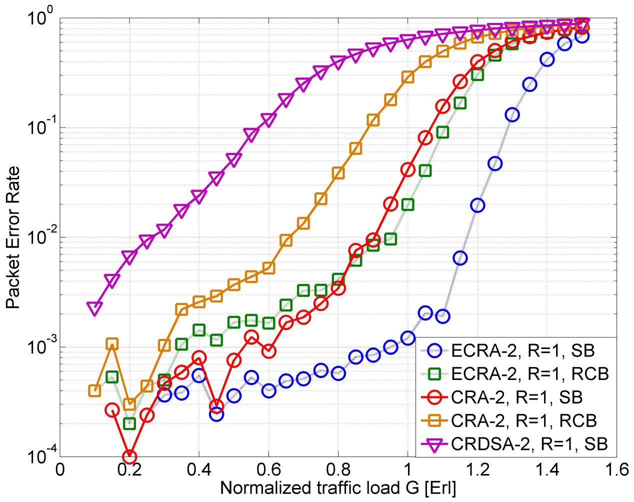


Fig. 6. CRDSA, CRA and ECRA packet error rate comparison for rate $R = 1$ and $SNR = 10\,\text{dB}$

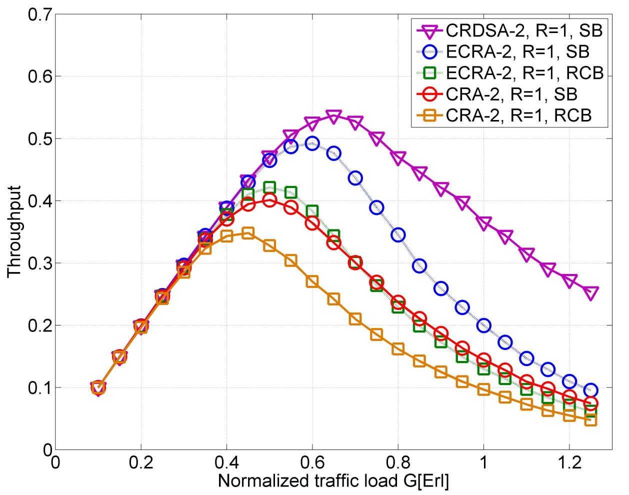


Fig. 7. CRDSA, CRA and ECRA throughput comparison for rate $R = 1$ and $SNR = 2\,\text{dB}$

the good performance of the proposed scheme also in more practical situations.

In Fig. 6 the PER behavior of the second set of simulations is shown. We can observe that the minimum PER for all the considered simulations is similar and close to 10^{-4}. This is due to the bound given by the number of simulated frames ($N_f = 1000$). When the SB is considered, ECRA can outperform CRA by more than one order of magnitude in the PER, for $G \geq 0.9\,\text{Erl}$. The same increase of performance can be found for the RCB simulations but for $G \geq 0.8\,\text{Erl}$.

The third set of simulations provided are done for a rate $R = 1$ and with a reduced $SNR = 2\,\text{dB}$, comparing the SB with the RCB. In Fig. 7 the throughput of ECRA, CRA and CRDSA are compared. It is CRDSA that reaches the best maximum throughput of $T_{max-CRDSA} \cong 0.53$ at $G = 0.65\,\text{Erl}$. ECRA with the SB reaches the maximum throughput $T_{max-ECRA} = 0.49$ at $G = 0.6\,\text{Erl}$ which is still close to CRDSA and highly outperforms CRA. The throughput increment of ECRA with respect to CRA with the SB is 23%, while it becomes 24% when the RCB is considered, in this second case.

V. Conclusions

In this paper a novel RA MAC protocol has been presented. Following the approach of CRA, the ECRA protocol exploits the presence of multiple packet replicas, together with the nature of occurring interference in Aloha-like channels combined with strong channel coding and the SIC process for resolving packet collisions. Moreover it was shown how ECRA attempts to resolve most of the partial collisions among packets, with the creation of a combined packet, generated from the lowest interfered parts of the replicas sent within the frame. It has been shown mathematically that this combined packet achieves always an equal or higher SNIR with respect to its corresponding replicas.

It has been also shown through numerical simulations that ECRA outperforms CRA in all the considered scenarios for both the throughput and the PER. The simulations have further shown that ECRA can achieve up to 26% of throughput gain compared to CRA when the RCB is considered. Under the same conditions, the PER of ECRA has a gain of one order of magnitude with respect to the CRA PER.

References

[1] N. Abramson, "The aloha system: Another alternative for computer communications," in *Proceedings of the 1970 Fall Joint Comput. Conf., AFIPS Conf.*, vol. 37, Montvale, N. J., 1970, pp. 281–285.

[2] L. G. Roberts, "Aloha packet system with and without slots and capture," *SIGCOMM Comput. Commun. Rev.*, vol. 5, pp. 28–42, April 1975.

[3] E. Casini, R. De Gaudenzi, and O. Herrero, "Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access scheme for satellite access packet networks," *Wireless Communications, IEEE Transactions on*, vol. 6, no. 4, pp. 1408 –1419, April 2007.

[4] C. Kissling, "Performance enhancements for asynchronous random access protocols over satellite," in *Communications (ICC), 2011 IEEE International Conference on*, june 2011, pp. 1 –6.

[5] G. Choudhury and S. Rappaport, "Diversity aloha–a random access scheme for satellite communications," *Communications, IEEE Transactions on*, vol. 31, no. 3, pp. 450 – 457, March 1983.

[6] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted aloha," *Communications, IEEE Transactions on*, vol. 59, no. 2, pp. 477 –487, february 2011.

[7] O. del Rio Herrero and R. D. Gaudenzi, "A high-performance MAC protocol for consumer broadband satellite systems," *IET Conference Publications*, vol. 2009, no. CP552, pp. 512–512, 2009.

[8] ——, "A high efficiency scheme for quasi-real-time satellite mobile messaging systems," in *Signal Processing for Space Communications, 2008. SPSC 2008. 10th International Workshop on*, 2008, pp. 1 –9.

[9] S. Gollakota and D. Katabi, "Zigzag decoding: Combating hidden terminals in wireless networks," *SIGCOMM'08*, pp. 159–170, 2008.

[10] A. G. D. A. S. Tehrani and M. J. Neely, "Sigsag: Iterative detection through soft message-passing," *Proceedings, IEEE INFOCOM*, 2011.

[11] R. G. Gallager, *Information Theory and Reliable Communication*. Wiley, 1968.