2022 IEEE International Symposium on Information Theory (ISIT) | 978-1-6654-2159-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISIT50566.2022.9834643

Energy Efficiency Analysis of a Feedback-Aided IRSA Scheme

Javad Haghighat and Tolga M. Duman
EEE Department, Bilkent University, Ankara, Turkey
{javad,duman}@ee.bilkent.edu.tr

***Abstract*— Irregular Repetition Slotted ALOHA (IRSA) achieves load thresholds very close to 1 at the expense of reduced energy efficiency compared to its competitor, Coded Slotted ALOHA (CSA). The efficiency is related to the expected number of transmitted replicas, and is upper-bounded by 0.5 in the case of IRSA. In this paper, we present a feedback-aided IRSA scheme, analyze its efficiency, and show that utilizing a very limited feedback will offer considerable improvements. Remarkably, the feedback-aided scheme enables IRSA to achieve efficiencies greater than 0.5, and in some cases, perform very close to the more complex CSA schemes.**

I. Introduction

Irregular Repetition Slotted ALOHA (IRSA), proposed in [1], is a random access protocol that achieves impressive performance gains compared to its predecessors, including Contention Resolution Diversity Slotted ALOHA (CRDSA) [2]. In [1], IRSA schemes are optimized to achieve load thresholds approaching 1, that is, the maximum attainable ratio of the number of active users to the number of time slots in the MAC frame approaches unity, while guaranteeing that the Packet Loss Ratio (PLR) asymptotically converges to zero. However, the optimization approach of [1] is oblivious to the expected number of replicas transmitted by a generic user. The importance of this expected value is later addressed in [3] where the authors show the connection between the inverse of this expected quantity and the (energy) efficiency of the scheme. The inverse of the expected number of replicas is referred to as *rate* in [3] but is later renamed as *efficiency* in [9][1].

The authors in [3] propose a more advanced ALOHA scheme called Coded Slotted ALOHA (CSA) that includes IRSA as a special case. Applications of CSA and IRSA are studied in numerous works including [4]-[14]. CSA achieves better load-efficiency trade-offs compared to IRSA. Also, unlike IRSA, CSA is able to achieve efficiencies greater than 0.5 [3]. However, IRSA enjoys a simpler implementation and faster decoding. In addition, in the case of noisy packet recovery, IRSA has the potential for post-processing the decoded packets and improving the error rate. Specifically, a repetition code naturally exists across the replicas at the bit level that may be leveraged, e.g., via bit-wise voting, to improve the bit error rate. Therefore, it is of interest to find methods for improving the achievable load-efficiency trade-offs for IRSA schemes and making them competitive with those achieved by CSA.

In this paper, we propose a feedback-aided IRSA scheme that improves the achievable load-efficiency trade-off compared to the standard IRSA. In the proposed scheme, feedback signals are generated according to the outcomes of tentative Successive Interference Cancellation (SIC) decoding steps performed at the receiver. The idea of performing tentative decoding and sending feedback signals is previously investigated in [15], [16], both in the context of Frameless ALOHA (FA) [17]. The aim of employing feedback in [15], [16] is to efficiently update the so-called slot access probability, which is a key parameter in the FA schemes, in order to optimize the system throughput. However, in this paper, we focus on improving the energy efficiency of IRSA schemes with fixed frame lengths, and we propose a new feedback-aided solution for that purpose.

It is worth mentioning that the idea of employing simple feedback signals for increasing the energy efficiency is also examined in [18], where the energy efficiency gains offered by implementing feedback are quantified via simulations. Also, no tentative SIC process is applied, and the SIC is triggered only at the end of the MAC frame.

The paper is organized as follows. The standard IRSA scheme is reviewed in Section II. The proposed feedback-aided IRSA solution is introduced in Section III. Section IV is dedicated to the energy efficiency analysis. Several numerical examples are provided in Section V. Finally, the paper is concluded in Section VI.

II. System Model for Standard IRSA

In the standard IRSA setup, a large number of sporadically active users are present. Each of the N active users intends to transmit one data packet over a MAC frame. A generic active user independently selects a repetition degree n, according to a Probability Mass Function (PMF) $\Lambda(x) = \sum_{n=2}^{n_{max}} \Lambda_n x^n$. Here Λ_n is the probability of selecting a degree n, and n_{max} determines the maximum number of allowed replicas.

The MAC frame includes a total of M slots. The user selects n out of these M slots uniformly at random for transmitting the replicas of its packet. Each user replica contains pointers to the slots where the other $n-1$ replicas are transmitted. At the end of the Mth time slot, the receiver performs an iterative decoding process, known as SIC [1]. The SIC decoder sequentially scans slots 1 through M, and if it finds a singleton slot, i.e., a slot over which only one replica is transmitted. The corresponding user packet is decoded, and its value is

This research is funded by the Scientific and Technological Research Council of Turkey (TUBITAK) under the Grant 119E589.

[1]The original definition of rate (efficiency) presented in [3] is more general, however, in the case of IRSA, it reduces to the inverse of the number of transmitted replicas.

subtracted from all the other slots over which a replica of the packet is transmitted. Due to this cancellation process, further singleton slots may become available, via which additional user packets can be decoded. Decoding continues until either all the user packets are decoded, or in an entire sequence scan, no further singleton slots are identified.

Running the SIC is identical to running iterative erasure decoding over a bipartite graph $\mathcal{G}$, consisting of N user nodes and M slot nodes, in which an edge connects the jth user node to the kth slot node if and only if (iff) the jth user has selected the kth slot for transmitting one of its replicas [1]. The erasure decoding process over $\mathcal{G}$ is initialized by marking all the user nodes and all the edges of $\mathcal{G}$ as erased, whereas the value of each slot node is initialized as the value of the signal received over the corresponding slot in the MAC frame. The erasure decoding algorithm consists of consecutive slot-to-user and user-to-slot phases at each iteration. In the slot-to-user phase, the decoder identifies slot nodes that are connected to a single erased edge, namely, the singleton slot nodes. For each singleton slot node, the decoder copies the value of that slot node to the user node connected to the other end of the erased edge, marks that user node as decoded and the edge as revealed. In the user-to-slot phase, every erased edge that is connected to a decoded user node, is marked as revealed; and, the value of that user node is subtracted from the value of the slot node connected to the other end of that edge. Then, the decoder proceeds with the next slot-to-user iteration, and continues the iterations until either all the user nodes are decoded, or no further singleton slot nodes can be identified (or, equivalently, no further user nodes can be decoded). The SIC decoding algorithm for IRSA and its extension for CSA are explained in detail in [1] and [3], respectively.

After finalizing the decoding process, the PLR, denoted by P_e, is found by dividing the number of non-decoded packets by N. The (energy) efficiency of the scheme is defined as

$$\eta\left(\Lambda\left(x\right)\right) = \frac{1}{\overline{n}\left(\Lambda\left(x\right)\right)} \tag{1}$$

where $\overline{n}\left(\Lambda\left(x\right)\right)$ denotes the expected number of replicas transmitted by a generic user and is given by $\overline{n}\left(\Lambda\left(x\right)\right) = \sum_{n=2}^{n_{max}} n \times \Lambda_n$ [3]. The traffic load is defined as $G = \frac{N}{M}$. We define the load threshold $G^*\left(\Lambda\left(x\right)\right)$ as the maximum attainable load for which $P_e \to 0$ when $N, M \to \infty$ while their ratio is kept constant.

III. A New Approach: Feedback-Aided IRSA

In the proposed feedback-aided IRSA solution, similar to the standard IRSA, each active user randomly selects a *designated* number of transmitted replicas according to the degree distribution $\Lambda\left(x\right)$, and then, identifies its *designated* slots uniformly at random among the set of all M slots in the MAC frame. However, unlike the IRSA that runs SIC only once for each MAC frame, the proposed scheme consists of $L > 1$ stages of incremental transmission by the users, followed by intermediate tentative decoding steps at the receiver. Specifically, given a vector $\boldsymbol{\delta} = (\delta_0, \delta_1, \ldots, \delta_{L-1}, \delta_L)$, where $\delta_0 = 0$, $\delta_L = 1$, $0 < \delta_1 < \ldots < \delta_{L-1} < 1$, the proposed scheme divides the MAC frame into L sub-frames, where the lth sub-frame begins at slot number $\delta_{l-1}M + 1$ and ends at slot number $\delta_l M$ (assuming that $\delta_l M$ is an integer for all l). The user nodes will transmit for one sub-frame duration, then wait for the receiver to perform tentative decoding and generate a limited-length feedback signal via which each user separately decides whether to proceed with transmission for the next sub-frame or not.

Note that we employ the term *designated* to indicate that, although all the replicas are scheduled, some of them may not be transmitted in practice. Hence, the expected number of replicas transmitted by a generic user will reduce to[2] $\overline{n}_F\left(\Lambda\left(x\right); \boldsymbol{\delta}, G\right) \leq \overline{n}\left(\Lambda\left(x\right)\right)$, and the energy efficiency will increase to $\eta_F\left(\Lambda\left(x\right); \boldsymbol{\delta}, G\right)$ defined as

$$\eta_F\left(\Lambda\left(x\right); \boldsymbol{\delta}, G\right) = \frac{1}{\overline{n}_F\left(\Lambda\left(x\right); \boldsymbol{\delta}, G\right)}. \tag{2}$$

The proposed feedback-aided IRSA algorithm is presented as Algorithm 1 in the next page. Let us make the following notes regarding several steps of the algorithm.

- Step (i) defines one realization of the MAC frame for the standard IRSA. Let $\mathcal{G}$ denote the bipartite graph corresponding to this realization. Let $\mathcal{G}_l$ denote the sub-graph consisting of the first $\delta_l M$ slot nodes of $\mathcal{G}$, in addition to the edge nodes and the user nodes that are connected to those slot nodes, for $l = 1, \ldots, L$.
- In Step (iv), the slots numbered 1 through $\delta_{l-1}M$ are processed at the previous stages of the tentative decoding process, through which their values change. The SIC at the current decoding stage takes their most recent values into account.
- In Step (v), we define a resolved slot node as a slot node for which all the edges connected to it are revealed (i.e., a slot node that is not connected to any erased edge).
- From Step (vi), it is easy to observe that

$$\mathcal{J}_l = \bigcup_{h=1}^{l} \mathcal{J}\left(\mathcal{M}_h\right). \tag{3}$$

- Step (viii) indicates that if u_j has at least one replica transmitted over a slot node in $\mathcal{K}\left(\mathcal{M}_l\right)$, since all the replicas transmitted over that slot node are decoded, the replica belonging to u_j, i.e., the packet sent by u_j, is already decoded at the receiver, hence transmission of more replicas by u_j will be redundant.
- Step (ix) implies that since the end of the current MAC frame is not reached, the incremental transmission process by the users and the tentative decoding process by the receiver continue for the current MAC frame.
- Step (x) is visited only if $l = L$, i.e., at the end of the current MAC frame. Therefore, the receiver terminates the decoding process for this MAC frame, then a new MAC frame is randomly generated, and all the user flags are reset to 1 by returning to Step (i).

[2]Dependency of $\overline{n}_F$ to G originates from (13) where G_l, the load value for the lth sub-graph, is determined in terms of G, as will be explained later in the paper.

Algorithm 1: Feedback-aided IRSA algorithm.

(i) Each user u_j, $1 \leq j \leq N$, randomly generates its designated number of replicas according to the degree distribution $\Lambda(x)$, and then, selects its corresponding designated slots uniformly at random out of the M slots available over the entire MAC frame.

(ii) Let $l = 1$. Set $c_{j,0} = 1$ for all the users u_j, $1 \leq j \leq N$. Define the initial set of decoded users at the receiver as $\mathcal{J}_0 = \emptyset$.

(iii) Every user u_j with $c_{j,l-1} = 1$, transmits its scheduled replicas over the slots corresponding to the lth sub-frame, i.e., the slots numbered $\delta_{l-1}M + 1$ through $\delta_l M$.

(iv) The receiver performs SIC over all the available slots, i.e., the slots numbered 1 through $\delta_l M$. This SIC decoding process is equivalent to running iterative erasure decoding over a modified sub-graph, $\mathcal{M}_l$, defined in Remark 2.

(v) The receiver identifies the set of decoded user nodes and the set of resolved slot nodes over $\mathcal{M}_l$, denoted by $\mathcal{J}(\mathcal{M}_l)$ and $\mathcal{K}(\mathcal{M}_l)$, respectively.

(vi) The receiver updates the set of decoded user nodes as $\mathcal{J}_l = \mathcal{J}_{l-1} \cup \mathcal{J}(\mathcal{M}_l)$.

(vii) The receiver generates a binary sequence of length $\delta_l M$, denoted by $\mathbf{f}_l = (f_{l,1}, \ldots, f_{l,\delta_l M})$, where $f_{l,k} = 1$ iff the kth slot, denoted by s_k, is resolved, i.e., iff $s_k \in \mathcal{K}(\mathcal{M}_l)$; and $f_{l,k} = 0$ otherwise. The receiver broadcasts $\mathbf{f}_l$ back to the users.

(vii) Each user employs the one-to-one mapping between $\mathbf{f}_l$ and $\mathcal{K}(\mathcal{M}_l)$ to locally re-generate $\mathcal{K}(\mathcal{M}_l)$.

(viii) For all j, if $\mathcal{B}(u_j) \cap \mathcal{K}(\mathcal{M}_l) \neq \emptyset$, user u_j ceases transmission for the rest of the MAC frame duration, where $\mathcal{B}(u_j)$ denotes the set of designated slot nodes for u_j. Hence, the user sets its flag bit to $c_{j,l} = 0$, and keeps this value intact for the rest of the MAC frame.

(ix) If $l < L$, the receiver and the users update their counters by letting $l \leftarrow l + 1$, and the algorithm returns to Step (iii).

(x) The algorithm returns to Step (i).

Remark 1. Note that throughout our analysis, we neglect the excess number of bits required for transmitting the feedback signals, denoted by $\mathbf{f}_l$'s. If each user packet has a length of K_u bits, the length of the original MAC frame is $K_u M$ bits, whereas the total length of the feedback signals is $\left(\sum_{l=1}^{L} \delta_l\right) \times M$ bits. Also, since for every l, $\delta_l \leq 1$, then $\left(\sum_{l=1}^{L} \delta_l\right) \times M \leq L \times M$. Therefore, if the number of sub-frames (L) is fixed and the packet length (K_u) increases, the amount of feedback becomes negligible with respect to the total frame length. Also note that it is possible to further reduce the length of the feedback signal via compression.

Remark 2. For constructing the modified sub-graph $\mathcal{M}_l$ introduced in Step (iv) of Algorithm 1, all the previously decoded user nodes, $u_j \in \mathcal{J}_{l-1}$, and their neighboring edges are excluded from $\mathcal{G}_l$. Excluding these edges is justified by noticing that if $u_j \in \mathcal{J}_{l-1}$, for every $h \leq l$ where $c_{j,h} = 0$, u_j has not transmitted its designated replicas over the hth sub-frame (since u_j has been decoded at an earlier stage), hence the corresponding edges must be excluded. On the other hand, if $c_{j,h} = 1$, since u_j is eventually decoded at a later stage, replicas of u_j transmitted over the hth sub-frame are already subtracted from the corresponding slots, i.e., the contribution of u_j on those slots is no longer present. Therefore, the equivalent graph-based representation must be modified by excluding the edges connecting u_j to the slot nodes in the hth sub-frame.

We employ the following theorem for the energy efficiency analysis carried out in the next section.

Theorem 1. *Consider running Algorithm 1 over one realization of the MAC frame, i.e., from Step (iii) through Step (ix). Then, the union of the sets of the user nodes decoded over $\mathcal{M}_1$ through $\mathcal{M}_l$ is identical to the set of user nodes decoded over $\mathcal{G}_l$, i.e.,*

$$\mathcal{J}(\mathcal{G}_l) = \bigcup_{h=1}^{l} \mathcal{J}(\mathcal{M}_h). \tag{4}$$

Proof: A sketch of the proof is as follows. (i) We can show that every user that is decoded over $\mathcal{G}_{l-1}$ will be decoded over $\mathcal{G}_l$; i.e., $\mathcal{J}(\mathcal{G}_{l-1}) \subset \mathcal{J}(\mathcal{G}_l)$. (ii) Then, we note that running erasure decoding over $\mathcal{M}_l$ is identical to running erasure decoding over $\mathcal{G}_l$ if the values of all the previously decoded users, i.e., all the users in $\mathcal{J}_{l-1}$, are initially revealed to the decoder; i.e., if $\mathcal{J}_{l-1}$ is initially revealed, the additional set of users decoded over $\mathcal{G}_l$ will be identical to $\mathcal{J}(\mathcal{M}_l)$. (iii) Next, we can show that if erasure decoding over $\mathcal{G}_l$ is initialized by revealing an arbitrary subset of $\mathcal{J}(\mathcal{G}_l)$ to the decoder, the final set of users decoded over $\mathcal{G}_l$ will remain unchanged; i.e., the union of the initially revealed subset and the additional users decoded after revealing that subset, gives $\mathcal{J}(\mathcal{G}_l)$.

Using the above results, we can prove (4) by induction. From the definition of $\mathcal{M}_l$ in Remark 2, and since $\mathcal{J}_0 = \emptyset$, observe that $\mathcal{M}_1 = \mathcal{G}_1$, hence $\mathcal{J}(\mathcal{G}_1) = \mathcal{J}(\mathcal{M}_1)$. Now, assuming that (4) holds for $l-1$, from (3) we observe that $\mathcal{J}_{l-1} = \mathcal{J}(\mathcal{G}_{l-1})$; then from (i) we obtain $\mathcal{J}_{l-1} \subset \mathcal{J}(\mathcal{G}_l)$. Therefore, using (ii), (iii) we conclude that $\mathcal{J}(\mathcal{G}_l) = \mathcal{J}_{l-1} \cup \mathcal{J}(\mathcal{M}_l)$, and replacing $\mathcal{J}_{l-1}$ by $\bigcup_{h=1}^{l-1} \mathcal{J}(\mathcal{M}_h)$ according to (3), gives (4). ■

IV. Analysis of the Energy Efficiency

In this section we aim to derive the energy efficiency of the proposed feedback-aided IRSA scheme, denoted by $\eta_F(\Lambda(x); \boldsymbol{\delta}, G)$ in (2), asymptotically as $N, M \to \infty$ while $G = \frac{N}{M}$ is kept constant.

First, let us define a vector $\mathbf{D}_j = (D_{j,1}, \ldots, D_{j,L})$ where $D_{j,l}$ denotes the number of replicas scheduled for transmission over the first l sub-frames by a generic user u_j at Step (i) of Algorithm 1. Subsequently, the number of replicas scheduled for transmission over the lth sub-frame is found as $D_{j,l} - D_{j,l-1}$ (where we define $D_{j,0} = 0$). It is clear that $D_{j,l}$ is equal to the number of edges connected to u_j in $\mathcal{G}_l$, i.e., the degree of u_j in $\mathcal{G}_l$. Since at Step (i) of Algorithm 1, the MAC frame is randomly generated, i.e., the graph $\mathcal{G}$ is randomly generated, $\mathbf{D}_j$ is a random vector. Let $\mathbf{d} = (d_1, \ldots, d_L)$ be a realization of $\mathbf{D}_j$. Clearly, $0 \leq d_1 \leq \ldots \leq d_L \leq n_{max}$, where n_{max} is the maximum allowed number of replicas, determined by the degree distribution $\Lambda(x)$. As an example, $\mathbf{d} = (1, 5, 7)$ means that the user has scheduled a total of 7 replicas for transmission over a total of 3 sub-frames, out of which 1, 4, and 2 replicas are scheduled for transmission over the first, the second, and the third sub-frames, respectively.

Let $n_{F,j}$ be the number of replicas transmitted by u_j before it ceases transmission due to a feedback signal that indicates that it is decoded at the receiver (i.e., a feedback signal that marks at least one of the slot nodes containing a replica of u_j as a resolved slot node). The expected number of transmitted replicas per user, denoted by $\overline{n}_F(\Lambda(x); \boldsymbol{\delta}, G)$ in (2), can be obtained as

$$\overline{n}_F(\Lambda(x); \boldsymbol{\delta}, G) = \frac{1}{N} \sum_{j=1}^{N} E[n_{F,j}]. \tag{5}$$

Let us define the random variable ζ_j denoting the tentative decoding stage at which u_j is successfully decoded, i.e., $\zeta_j = l$ means that u_j is successfully decoded at stage l (i.e., over the modified graph $\mathcal{M}_l$, defined in Remark 2). Hence, u_j will cease transmission for sub-frames $l+1$ through L. Using the law of total expectation, we write

$$\begin{aligned} E[n_{F,j}] = \textstyle\sum_{\mathbf{d} \in \mathcal{D}} \sum_{l=1}^{L} E[n_{F,j} | \zeta_j = l, \mathbf{D}_j = \mathbf{d}] \\ \times P(\zeta_j = l | \mathbf{D}_j = \mathbf{d}) \times P(\mathbf{D}_j = \mathbf{d}) \end{aligned} \tag{6}$$

where $\mathcal{D}$ denotes the set of all possible realizations. Note that when it is given that $\mathbf{D}_j = \mathbf{d}$ and u_j is decoded at stage l, it is known that the number of replicas transmitted by u_j is equal to d_l, i.e.,

$$E[n_{F,j} | \zeta_j = l, \mathbf{D}_j = \mathbf{d}] = d_l. \tag{7}$$

Also, it can be shown that

$$P(\boldsymbol{D}_j = \boldsymbol{d}) = \Lambda_{d_L} \prod_{l=1}^{L} \binom{d_L - d_{l-1}}{d_l - d_{l-1}} (\delta_l - \delta_{l-1})^{d_l - d_{l-1}} \tag{8}$$

where we define $d_0 = 0$ and Λ_{d_L} is the coefficient corresponding to x^{d_L} in the degree distribution polynomial $\Lambda(x)$. To verify (8), note that given d_L, the number of remaining non-scheduled replicas at the beginning of the lth sub-frame is $d_L - d_{l-1}$, among which $d_l - d_{l-1}$ replicas should be selected for assignment to the slots in the lth sub-frame. Also, the probability that a replica is assigned to a slot in the lth sub-frame is equal to $\delta_l - \delta_{l-1}$ (since the length of the lth sub-frame is $(\delta_l - \delta_{l-1}) \times M$ and slots are selected uniformly at random). Therefore, the conditional probability of observing $d_1, \ldots, d_{L-1}$ given d_L is found as $\prod_{l=1}^{L} \binom{d_L - d_{l-1}}{d_l - d_{l-1}} (\delta_l - \delta_{l-1})^{d_l - d_{l-1}}$. Also, since d_L denotes the total number of replicas scheduled by u_j, i.e., the degree of u_j in $\mathcal{G}$, the probability of observing d_L is equal to Λ_{d_L}. Multiplying this probability by the conditional probability of observing $d_1, \ldots, d_{L-1}$ given d_L, gives (8).

In order to complete the derivation of $\overline{n}_F(\Lambda(x); \boldsymbol{\delta}, G)$, we need to derive $P(\zeta_j = l | \mathbf{D}_j = \mathbf{d})$. In the sequel, we show that $P(\zeta_j = l | \mathbf{D}_j = \mathbf{d})$ can be found via density evolution in the asymptotic case. For this, we employ the following lemma.

Lemma 1. *Define $A_{l,j}$ as the event that u_j is decoded over $\mathcal{G}_l$ assuming that the decoding over $\mathcal{G}_l$ continues up to a point where no further user nodes can be decoded. Then, the events $A_{l,j}$ and $\{\zeta_j \leq l\}$ are identical, i.e., $A_{l,j}$ occurs iff $\{\zeta_j \leq l\}$ occurs.*

Proof: A sketch of the proof is as follows. (i) we show that the event $\{\zeta_j \leq l\}$ is represented by the set of all realizations of the MAC frame, i.e., all the realizations of $\mathcal{G}$, for which $u_j \in \bigcup_{h=1}^{l} \mathcal{J}(\mathcal{M}_h)$. Let us refer to this set as $\Omega_{j,1}$. (ii) We note that $A_{l,j}$ is represented by $\Omega_{j,2}$, the set of all realizations of $\mathcal{G}$ for which $u_j \in \mathcal{J}(\mathcal{G}_l)$. (iii) Given any $\mathcal{G} \in \Omega_{j,1}$, we apply Theorem 1 to prove that $\mathcal{G} \in \Omega_{j,2}$, i.e., we argue that $\Omega_{j,1} \subset \Omega_{j,2}$. (iv) Using the same approach, we can show that $\Omega_{j,2} \subset \Omega_{j,1}$; hence $\Omega_{j,1} = \Omega_{j,2}$. ∎

By aid of Lemma 1, we observe that

$$P(\zeta_j \leq l | \mathbf{D}_j = \mathbf{d}) = P(A_{l,j} | \mathbf{D}_j = \mathbf{d}). \tag{9}$$

Therefore, given $P(A_{l,j} | \mathbf{D}_j = \mathbf{d})$, i.e., the probability that u_j is decoded over $\mathcal{G}_l$, we can obtain the conditional PMF of ζ_j, that is,

$$\begin{aligned} &P(\zeta_j = l | \mathbf{D}_j = \mathbf{d}) \\ &\quad = P(\zeta_j \leq l | \mathbf{D}_j = \mathbf{d}) - P(\zeta_j \leq l-1 | \mathbf{D}_j = \mathbf{d}) \end{aligned} \tag{10}$$

for all $l = 1, \ldots, L$.

In order to derive $P(A_{l,j} | \mathbf{D}_j = \mathbf{d})$ via density evolution, we find the traffic load and the user degree distribution corresponding to $\mathcal{G}_l$. Let us define the marginal distribution of the random variable $D_{j,l}$ as follows

$$\tilde{\Lambda}_l(x) = \sum_{h=0}^{n_{max}} \tilde{\Lambda}_{l,h} \times x^h \tag{11}$$

where $\tilde{\Lambda}_{l,h} = P(D_{j,l} = h)$ is found as

$$\tilde{\Lambda}_{l,h} = \sum_{\boldsymbol{d} \in \mathcal{D}} P(\boldsymbol{D}_j = \boldsymbol{d}) \times 1_{d_l = h}, \tag{12}$$

where $1_{d_l = h}$ is the indicator function, which is equal to 1 if its argument is true, and 0 otherwise, and $P(\boldsymbol{D}_j = \boldsymbol{d})$ is expressed according to (8). Note that since the right hand side of (8) and the set $\mathcal{D}$ do not depend on j, $\tilde{\Lambda}_l(x)$ does not depend on j, i.e., $D_{j,l}$'s are identically distributed for all user indexes. The probability that u_j is included in $\mathcal{G}_l$ is equal to the probability that $D_{j,l} > 0$, i.e., $\left(1 - \tilde{\Lambda}_{l,0}\right)$. Hence, when $N \to \infty$, $\left(1 - \tilde{\Lambda}_{l,0}\right) \times N$ user nodes are included in $\mathcal{G}_l$. Dividing this number by the number of slot nodes in $\mathcal{G}_l$, which is $\delta_l M$, gives the corresponding traffic load as

$$G_l = \frac{\left(1 - \tilde{\Lambda}_{l,0}\right)}{\delta_l} \times G. \tag{13}$$

Also, the probability that a generic user included in $\mathcal{G}_l$ has a degree h, for $h \geq 1$, is found by normalizing $\tilde{\Lambda}_{l,h}$ as $\Lambda_{l,h} = \frac{\tilde{\Lambda}_{l,h}}{1-\tilde{\Lambda}_{l,0}}$; therefore, we can define the degree distribution corresponding to $\mathcal{G}_l$ as

$$\Lambda_l(x) = \sum_{h=1}^{n_{max}} \Lambda_{l,h} \times x^h. \quad (14)$$

Using $\Lambda_l(x)$ and G_l, we can run the density evolution [1] and update the edge erasure probability, until reaching a value q_l, that cannot be reduced any further. We then calculate $P(A_{l,j}|\mathbf{D}_j = \mathbf{d})$ by noting that, given $\mathbf{D}_j = \mathbf{d}$, the number of edges connected to u_j in $\mathcal{G}_l$ is equal to d_l. Furthermore, u_j is *not decoded* over $\mathcal{G}_l$ iff all these d_l edges are erased, i.e.,

$$P(A_{l,j}|\mathbf{D}_j = \mathbf{d}) = 1 - q_l^{d_l}. \quad (15)$$

From (10), (9), (15), we find

$$P(\zeta_j = l|\mathbf{D}_j = \mathbf{d}) = q_{l-1}^{d_{l-1}} - q_l^{d_l} \quad (16)$$

for all l, $1 \leq l \leq L$, where we define $q_0 = 1$. Note that $\mathcal{G}_L$ is identical to $\mathcal{G}$; subsequently, $G_L = G$ and $\Lambda_L(x) = \Lambda(x)$. Furthermore, we assume that $G \leq G^*(\Lambda(x))$ where $G^*(\Lambda(x))$ is the load threshold corresponding to $\Lambda(x)$. This assumption guarantees that all the users will be decoded after the transmission of the Lth sub-frame, i.e., over $\mathcal{G}_L$; hence, $q_L = 0$.

By replacing (7), (8), (16) in (6) we find $E[n_{F,j}]$. We observe that $E[n_{F,j}]$ does not depend on j; therefore, from (5) we can write $\overline{n}_F(\Lambda(x); \boldsymbol{\delta}, G) = E[n_{F,1}]$ for an arbitrarily selected user. To summarize, we have

$$\begin{aligned}\overline{n}_F(\Lambda(x); \boldsymbol{\delta}, G) = \; & \textstyle\sum_{\mathbf{d}\in\mathcal{D}} \sum_{l=1}^{L} d_l \Lambda_{d_L} \left(q_{l-1}^{d_{l-1}} - q_l^{d_l}\right) \\ & \times \textstyle\prod_{r=1}^{L} \binom{d_L - d_{r-1}}{d_r - d_{r-1}} (\delta_r - \delta_{r-1})^{d_r - d_{r-1}}\end{aligned} \quad (17)$$

where $d_0 = 0$ and $q_0 = 1$. The complexity of (17) depends on the cardinality of the set $\mathcal{D}$, denoted by $|\mathcal{D}|$, which is a function of $\Lambda(x)$ and L. For example, for $\Lambda(x) = x^3$ and $L = 20$, $|\mathcal{D}| = 8.9 \times 10^3$.

Plugging (17) into (2) gives $\eta_F(\Lambda(x); \boldsymbol{\delta}, G)$, i.e., the efficiency of the proposed feedback-aided IRSA scheme.

V. Numerical Examples

Table I provides the efficiencies achieved by several feedback-aided IRSA schemes, when $L = 20$ and $\boldsymbol{\delta} = (0, 0.05, 0.10, \ldots, 1)$. The values given in the last column, labeled as η_{CSA}, are the efficiencies of the optimized CSA schemes taken from [3] with load thresholds identical to $G^*(\Lambda(x))$. Both the feedback-aided IRSA scheme and the CSA scheme operate at a traffic load $G = G^*(\Lambda(x))$. It is observed that the feedback-aided IRSA solution achieves efficiencies very close to those of the optimized CSA solution for the same traffic load.

Table I
Comparison between the efficiencies of feedback-aided IRSA scheme (η_F) and optimized CSA scheme (η_{CSA}). The CSA user degree distributions are $\Lambda_{10}(x), \Lambda_8(x), \Lambda_7(x)$, listed in ([3]-Table III), for the first, the second, and the third row, respectively.

$\Lambda(x)$	$G = G^*(\Lambda(x))$	$\eta_F(\Lambda(x); \boldsymbol{\delta}, G)$	η_{CSA}
$0.99x^2 + 0.01x^3$	0.505	0.672	0.667
$0.84x^2 + 0.16x^3$	0.594	0.613	0.599
$0.63x^2 + 0.37x^3$	0.746	0.516	0.502

As observed in Table I, the feedback-aided IRSA solution has the potential to slightly outperform the CSA (without feedback). However, if CSA is used with feedback, its energy efficiency will also improve. On the other hand, IRSA has advantages over CSA in some aspects. Since the number of *segments* [3] in CSA is greater than the number of packets in IRSA, more pointers are required for CSA compared to IRSA. In addition, since segments are shorter in CSA, shorter length error correcting codes will be employed at the physical layer, potentially increasing the error rate in practical schemes. All the above-mentioned points must be carefully studied in order to provide an accurate comparison between the energy efficiency of feedback-aided IRSA and CSA.

We also verify the results of Table I by simulating the proposed solution for a frame length of $M = 1000$ slots, which gives η_F as 0.671, 0.613, 0.515, for the first, the second, and the third setups, respectively. However, the PLR values are no longer negligible; instead, they are measured as 0.0098, 0.0166, 0.0913, for the first, the second, and the third setups, respectively.

As a final note, we observe that unlike the standard IRSA for which the achievable efficiency is upper-bounded by 0.5 [3], the feedback-aided solution is able to achieve efficiencies considerably higher.

VI. Summary and Conclusions

We propose and analyze a feedback-aided IRSA approach, where the feedback signals communicate the locations of resolved slot nodes after performing tentative SIC decoding at the end of each sub-frame. By processing the feedback signal, each user detects whether its packet is decoded at the receiver, in which case the user ceases transmission for the remainder of the frame. Hence, the expected number of replicas transmitted by a generic user decreases, and the energy efficiency improves. Through numerical examples, we show that the proposed solution achieves efficiencies very close to those offered by the more complex CSA schemes.

The proposed solution can be a promising choice for the scenarios with energy harvesting where the transmit energy budget is stochastic and limited by the energy arrival rate [10], [11]. For future work, the derived analytical results can be employed for optimization purposes, i.e., to design the most energy efficient feedback-aided scheme for a fixed load value. Furthermore, studying fundamental limits of energy efficiency for feedback-aided CSA is another interesting research direction.

REFERENCES

[1] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted ALOHA," IEEE Transactions on Communications, vol. 59, no. 2, pp. 477-487, Feb. 2011.

[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, Apr. 2007.

[3] E. Paolini, G. Liva and M. Chiani, "Coded slotted ALOHA: A graph-based method for uncoordinated multiple access," IEEE Transactions on Information Theory, vol. 61, no. 12, pp. 6815-6832, Dec. 2015.

[4] E. Sandgren, A. Graell i Amat and F. Brännström, "On frame asynchronous coded slotted ALOHA: Asymptotic, finite length, and delay analysis," IEEE Transactions on Communications, vol. 65, no. 2, pp. 691-704, Feb. 2017.

[5] M. Ivanov, F. Brännström, A. Graell i Amat and G. Liva, "Unequal error protection in coded slotted ALOHA," IEEE Wireless Communications Letters, vol. 5, no. 5, pp. 536-539, Oct. 2016.

[6] M. Ivanov, F. Brännström, A. Graell i Amat and P. Popovski, "Broadcast coded slotted ALOHA: A finite frame length analysis," IEEE Transactions on Communications, vol. 65, no. 2, pp. 651-662, Feb. 2017.

[7] D. Jia, Z. Fei, H. Lin, J. Yuan and J. Kuang, "Distributed decoding for coded slotted ALOHA," IEEE Communications Letters, vol. 21, no. 8, pp. 1715-1718, Aug. 2017.

[8] A. Graell i Amat and G. Liva, "Finite-length analysis of irregular repetition slotted ALOHA in the waterfall region," IEEE Communications Letters, vol. 22, no. 5, pp. 886-889, May 2018.

[9] G. Liva, E. Paolini, Č. Stefanović and A. Graell i Amat, "Coded slotted Aloha over the on-off fading channel: Performance bounds," 53rd Asilomar Conference on Signals, Systems, and Computers, Nov. 2019, pp. 31-35.

[10] U. Demirhan and T. M. Duman, "Irregular repetition slotted ALOHA with energy harvesting nodes," IEEE Transactions on Wireless Communications, vol. 18, no. 9, pp. 4505-4517, Sept. 2019.

[11] T. Akyıldız, U. Demirhan and T. M. Duman, "Energy harvesting irregular repetition ALOHA with replica concatenation," IEEE Transactions on Wireless Communications, vol. 20, no. 2, pp. 955-968, Feb. 2021.

[12] E. Nisioti and N. Thomos, "Fast Q-learning for improved finite length performance of irregular repetition slotted ALOHA," IEEE Transactions on Cognitive Communications and Networking, vol. 6, no. 2, pp. 844-857, June 2020.

[13] S. Saha, V. B. Sukumaran and C. R. Murthy, "On the minimum average age of information in IRSA for grant-free mMTC," IEEE Journal on Selected Areas in Communications, vol. 39, no. 5, pp. 1441-1455, May 2021.

[14] J. Su, G. Ren and B. Zhao, "NOMA-based coded slotted ALOHA for machine-type communications," IEEE Communications Letters, vol. 25, no. 7, pp. 2435-2439, July 2021.

[15] S. Ogata and K. Ishibashi, "Application of ZigZag decoding in frameless ALOHA," IEEE Access, vol. 7, pp. 39528-39538, Mar. 2019.

[16] F. Lázaro, Č. Stefanović and P. Popovski, "Reliability-latency performance of frameless ALOHA with and without feedback," IEEE Transactions on Communications, vol. 68, no. 10, pp. 6302-6316, Oct. 2020.

[17] C. Stefanovic and P. Popovski, "ALOHA random access that operates as a rateless code," IEEE Transactions on Communications, vol. 61, no. 11, pp. 4653-4662, Nov. 2013.

[18] L. Valentini, A. Faedi, M. Chiani and E. Paolini, "Coded random access for 6G: Intra-frame spatial coupling with ACKs," 2021 IEEE GLOBECOM Workshops (GC Wkshps), Dec. 2021, pp. 1-6.