

Sign-Compute-Resolve for Tree Splitting Random Access

Jasper Goseling, Čedomir Stefanović and Petar Popovski

Abstract—We present a framework for random access that is based on three elements: physical-layer network coding (PLNC), signature codes and tree splitting. In presence of a collision, physical-layer network coding enables the receiver to decode, *i.e.*, compute, the sum of the packets that were transmitted by the individual users. For each user, the packet consists of the user’s signature, as well as the data that the user wants to communicate. As long as no more than K users collide, their identities can be recovered from the sum of their signatures. This framework for creating and transmitting packets can be used as a fundamental building block in random access algorithms, since it helps to deal efficiently with the uncertainty of the set of contending terminals. In this paper we show how to apply the framework in conjunction with a tree-splitting algorithm, which is required to deal with the case that more than K users collide. We demonstrate that our approach achieves throughput that tends to 1 rapidly as K increases. We also present results on net data-rate of the system, showing the impact of the overheads of the constituent elements of the proposed protocol. We compare the performance of our scheme with an upper bound that is obtained under the assumption that the active users are a priori known. Also, we consider an upper bound on the net data-rate for any PLNC based strategy in which one linear equation per slot is decoded. We show that already at modest packet lengths, the net data-rate of our scheme becomes close to the second upper bound, *i.e.*, the overhead of the contention resolution algorithm and the signature codes vanishes.

I. INTRODUCTION

Uncertainty is the essential element of communication systems, caused by noise, errors, and random traffic (packet) arrivals at user(s). A canonical example of the latter is seen in random access protocols, used for handling transmissions of users to a common receiver, *e.g.*, a base station, over a shared wireless medium. Random access is necessary when the total number of users associated with the base station is large, but within a given short time interval, the number of active users that have packets to transmit is small and unknown *a priori*. Such is the case in, for instance, wide-area networks

The work of J. Goseling was supported in part by the Netherlands Organization for Scientific Research (NWO), grant 612.001.107. The work of Č. Stefanović was supported by the Danish Council for Independent Research (DFF), grant DFF-4005-00281. The work of P. Popovski has been in part supported by the European Research Council (ERC Consolidator Grant nr. 648382 WILLOW) within the Horizon 2020 Program.

Part of this work was presented at the IEEE International Conference on Communications, 2014 and at the 52nd Annual Allerton Conference on Communication, Control, and Computing, 2014.

J. Goseling is with the Department of Applied Mathematics, University of Twente, Drienerlolaan 5, 7522 NB Enschede, The Netherlands. (email: j.goseling@utwente.nl)

Č. Stefanović and P. Popovski are with the Department of Electronic Systems, Aalborg University, Denmark. (email: cs@es.aau.dk, petarp@es.aau.dk)

of sensors, where each sensor has a sporadic traffic pattern. The goal of random access protocols is to enable each of the active users to eventually send its packet successfully.

Traditionally, random access protocols have been designed under the *collision model*: when two or more users transmit at the same time, a collision occurs and all involved transmissions are lost. In other words, collisions are considered as destructive and the information contained in them as irrecoverable. Therefore, the objective of classical random access protocols, such as ALOHA [1] or tree splitting [2], is to ensure that each user gets the opportunity to send its packet without collision. On the other hand, the recent inclusion of elaborate physical layer techniques in random access protocols allows for the extension of the design space beyond the collision model [3]–[7], such that collisions are treated as sums of packets and, instead of being discarded, they are buffered and reused. We motivate the benefits of such an approach through a simple example, where the received signals in the first two slots are

$$\begin{aligned} Y_1 &= X_1 + X_2 + Z_1, \\ Y_2 &= X_2 + Z_2, \end{aligned} \quad (1)$$

where X_1 and X_2 are the user signals (packets) and Z_1 and Z_2 represent the noise. The received signal Y_1 is buffered, and, if X_2 is successfully decoded from slot Y_2 , it can be subtracted (*i.e.*, cancelled) from Y_1 . The receiver proceeds by attempting to decode X_1 from the noisy signal $Y_1 - X_2 = X_1 + Z_1$. Obviously, the exploitation of the information contained in the collision slot Y_1 boosts the protocol performance.

The motivating example also demonstrates that, in the general framework, the impact of the noise cannot be neglected. This is fundamentally changed by applying reliable physical layer network coding (PLNC) to the problem of random access. The key idea in PLNC is to decode a function of multiple received signals, rather than decoding the individual signals. Such operation is termed denoise-and-forward [8], [9], or compute-and-forward [10], the latter being the motivation for part of the name of the scheme proposed in the paper. Assume that W_1 and W_2 represent the data as a sequence of symbols from finite field \mathbb{F}_q that are mapped to the baseband signals X_1 and X_2 , respectively, in example (1). Upon receiving Y_1 from (1), the base station stores $W_1 + W_2$. If X_2 (*i.e.*, W_2) is decoded from Y_2 , then W_1 can be obtained from the stored signal $W_1 + W_2$, *i.e.*, the sum in \mathbb{F}_q of W_1 and W_2 . In brief, the use of PLNC removes the uncertainty of the noise, leaving the receiver only with the uncertainty about the contending set of users.

One of the main challenges in the application of PLNC in random access protocols is for the receiver to learn the set of

transmitting users [6]. In (1), for instance, the receiver does not know that X_1 is sent in slots 1 and 2, and X_2 in slot 2. This sets the motivation to introduce PLNC-based random access with signatures in [11], [12], where the users prepend to their messages a codeword of a K -out-of- N code [13]–[15]. More precisely, the ℓ -th user applies the following communication strategy: it prepends a *signature* W_ℓ^s , consisting of predefined number of symbols, to the message W_ℓ^d in order to obtain W_ℓ . The signature is based on a code that has the following property: if at most K users transmit in a given slot, then from the sum of the signatures the receiver knows exactly which transmitters have contributed to the data stored in the present slot. In other words, the sum

$$\sum_{\ell=1}^L W_\ell^s \quad (2)$$

is uniquely decodable if $L \leq K$, where L denotes the collision multiplicity.

The addition of signatures to packets and its combination with PLNC brings a conceptual novelty in the design of random access algorithms. The reason is that, when the number of colliding terminals is at most K , then the receiver can find out the identities of those terminals, but it cannot decode the data of all of them. This means that from a collision with K or less contenders, the receiver can extract protocol-related information, but not the data. This is a generalization of the common case when $K = 1$, where a single-user transmission is required to decode both protocol information and data. The efficiency of this approach is shown in the context of a tree-splitting algorithm for contention resolution. We emphasize that the mechanisms enabled by the signatures and PLNC are not limited to tree-splitting algorithms, but we have selected the tree-splitting framework as it is known to provide the highest possible throughput for the traditional collision model [16]. In fact, these mechanisms can be applied in any framework of multiple access protocols in which the collisions are not wasted, but used in decoding, as in *e.g.*, coded random access [7]. The use of PLNC transforms the multiple access channel into an \mathbb{F}_q adder channel, *i.e.*, it removes the uncertainty due to noise. The addition of signature coding facilitates the generalization of the concept of collision, which is a conceptual shift away from the standard tree splitting context. Specifically, it allows the receiver to obtain (i) the knowledge of the collision multiplicity L , *i.e.*, the number of the collided user signals in the slot, and (ii) the resolution of collided user identities, if $L \leq K$, where K is a design parameter. These features lead to a revision of the objective of contention resolution as compared to the standard schemes: to drive the contending users in a state in which the collision multiplicity becomes resolvable and the receiver is able to get a sufficient number of *equations* in the finite field in order to be able to decode the users' data. We analyze the proposed scheme and provide results related to the expected duration of the contention resolution interval, the expected number of resolved users per slot (*i.e.*, throughput) and the net rate of information transmission, which takes into account the overhead related to PLNC and signatures. We show that the use of signatures

is significantly reducing the average time required to extract useful information from the collisions, therefore improving the overall throughput performance. On the other hand, the assessment of the net rate provides insights in the basic tradeoffs and mechanisms that need to be considered for a contention resolution based on PLNC and signatures. To the best of our knowledge, this type of analysis is typically omitted in the literature on tree-splitting algorithms. The reason is that in random access literature a packet is the atomic unit of communication, such that the overhead related to the contention identifiers and channel coding are not modeled and throughput is the primary performance parameter. We compare the net rate of our scheme with upper bounds on the net rate. We also extend the scheme to include successive interference cancellation (SIC), enabling the use of collisions with multiplicities larger than K , and we provide the accompanying analysis and performance assessment. Note that SIC can be seamlessly incorporated into the proposed framework, as the PLNC reduces the interference cancellation to simple operations in \mathbb{F}_q . Finally, we note that the proposed scheme is simple to implement at the transmitter side, requiring use of a linear code and a unique signature. This is an important criterion for scenarios in which a massive number of sensors (or other small devices) transmit information to a common receiver. Our analysis of the achieved net rate, including the upper bounds, demonstrate that keeping the complexity low results only in a small penalty.

The paper is organized as follows. In Section II we discuss related work. In Section III we introduce the model. In Section IV we present results on PLNC, signature codes and tree splitting that will be used in the remainder of the paper. The proposed strategy is presented in Section V and its performance analyzed in Section VI. An extension of the strategy that involves successive interference cancellation is presented in Section VII. Finally, a discussion and concluding remarks are given in Section VIII.

II. RELATED WORK

The use of PLNC for random access was studied in [5], [17]–[21], where it was assumed that the receiver knows which users are active in each slot. Other ways to relax this assumption that do not involve signatures are to rely on the successful decoding of signals from singleton slots [4], [7], as suggested in example (1), or to rely on complex signal separation techniques [3].

The use of PLNC and signature codes was considered in [22] for broadcast in networks. The combination of PLNC and signature codes for random access was introduced in [11]. Both in [22] and [11] it is assumed that the number of contending users is bounded. In this work we set aside this assumption and design a contention resolution algorithm that deals with any number of contending users.

A comprehensive review of signature coding and its application to multiple access can be found in [23]. The reviewed results mainly involve existence proofs of certain types of signature codes, leaving the contention resolution and, in general, random access protocol operation out of focus. Two

exceptions can be found in [24] and [25], where the authors consider a tree-splitting and an ALOHA based random access solution, respectively, which exploit K -out-of- M multiple access codes proposed in [13]. The approach suggested in [24] resembles the one proposed in the paper, however, the authors neglect the impact of noise, conclude that a choice of $K = 3$ is optimal for the code construction from [13], and also show that the conventional tree splitting actually outperforms their solution in terms of user resolution rate in the case of blocked access, due to the fact that the code length is equal to M . On the contrary, our approach is based on much shorter codes, whose length scales roughly as $K \log M$, the impact of noise is explicitly taken into account, and the performance is thoroughly characterized in terms of K , including insights in the choice of optimal values.

The scheme employed in the paper provides the receiver with the knowledge of the collision multiplicity. Pippenger showed in a non-constructive way that this knowledge could be used to achieve throughputs that asymptotically tend to one under the collision channel model [26]. The construction of the protocol which leverages on these ideas was done in [27], however, the proposed solution involves exponential computational complexity. On the other hand, the scheme employed in this paper achieves comparable performance, albeit using much simpler operating principles.

The analysis of the tree-splitting algorithm presented in this paper is based on the approach pioneered by Massey [16]. The use of SIC in the contention resolution framework was first investigated in [4], where it was shown that throughput performance can be pushed to 0.693. Another approach was suggested in [28], where SIC is employed over a set of partially split trees, and optimization was performed over the splitting strategy that favors fast SIC evolution. The reported throughputs for the presented design example in [28] are close to 0.8. The part of this paper that deals with SIC can be perceived as a generalization of the ideas presented in [4], with the important difference that SIC in the proposed framework is performed in the digital domain, effectively removing the memory constraints [29] and potential imperfections of the interference cancellation in the analog domain [30].

III. MODEL AND PROBLEM STATEMENT

We consider a large set of potential transmitters (users) $1, \dots, M$, with $M \gg 1$. A small number of users are active and have a message of D bits that should be sent to a common receiver (base station). Messages are independent and drawn uniformly at random from $\{1, \dots, 2^D\}$. We model the user activity by a single batch arrival, where each user, independently of all other users becomes active (*i.e.*, arrives in the batch) with probability p before the protocol that is described below is initiated. After the protocol has been initiated, we assume that no user arrival occurs; this simplifying assumption allows us to focus on the basic analysis of the protocol operation, similar to other works on tree splitting, *c.f.* [16]. In Section VIII, we provide comments on more general models of the user activation and the protocol operation.

We denote by \mathcal{L} the set of active users, and by $L = |\mathcal{L}|$ the number of active users. Neither the receiver nor the users

know \mathcal{L} or L . Hence, L has a binomial distribution, where the probability that L users are contending is denoted by $q(L) = \binom{M}{L} p^L (1-p)^{M-L}$. For notational convenience, let $q_0 = q(0) = (1-p)^M$. We denote by $\hat{q}(L)$ the probability of having L contending users conditioned on the fact there is at least one, *i.e.*, $\hat{q}(L) = q(L)/(1-q_0)$.

The symbol transmitted by the m -th user in the τ -th channel use, $\tau \in \mathbb{N}$, is denoted by $X_m(\tau)$. We assume unit channel gains, *i.e.*, at the τ -th channel use the receiver observes

$$Y(\tau) = \sum_{m \in \mathcal{L}} X_m(\tau) + Z(\tau), \quad (3)$$

where $\{Z(\tau)\}_{\tau=1}^{\infty}$ is white Gaussian noise with unit variance.

The goal of this paper is to devise a protocol that allows the receiver to retrieve both the identities and the messages of all contending users. In particular, we consider protocol that uses multiple blocks of N channel uses. In line with other literature on random access, we refer to blocks of channel uses as *slots*. In each slot the receiver attempts to decode a linear combination of messages transmitted by the users. We restrict our attention to strategies in which the rate (in bits per channel use) is the same for all users and constant over slots. At the end of each slot, the receiver provides feedback to the users and, unless all messages are resolved at the receiver, a new slot is started. Feedback is instantaneous, error free and received by all users. We do not impose any constraints on the amount of feedback that can be provided and explicitly specify how feedback is used later in the paper.

Rephrasing the above, the constituent elements of the protocol are the use of a contention resolution mechanism across slots, dealing with randomness of the user activity pattern, and use of forward error correcting code within slots, dealing with noise. With respect to the latter, we ignore finite block length effects and assume that forward error correcting codes operate with vanishing error probability at any rate up to capacity. As a consequence, the task for the receiver is to recover all packets with arbitrarily small error probability.

The signal of each user needs to satisfy an average power constraint in each slot, *i.e.*,

$$\frac{1}{N} \sum_{\tau=1}^N |X_m(\tau)|^2 \leq P, \quad (4)$$

for all $m \in \{1, \dots, M\}$. We will assume that $P > 1$, such that a positive computation rate over the multiple access channel can be achieved, as seen in the next section.

We are interested in the following performance parameters. By $S(L)$ we denote the expected number of slots that the strategy uses to resolve L contending users, where the expectation in $S(L)$ is w.r.t. the randomness in the contention resolution mechanism. By $R_{\text{res}}(L) = L/S(L)$ we denote the expected number of users that is resolved per slot, commonly referred to as throughput in random access literature. We are also interested in \bar{R}_{res} , obtained by averaging $R_{\text{res}}(L)$ over L , *i.e.*,

$$\bar{R}_{\text{res}} = \mathbb{E}[R_{\text{res}}(L)|L > 0] = \sum_{L=1}^M \frac{L}{S(L)} \hat{q}(L). \quad (5)$$

In addition to \bar{R}_{res} , which is obtained under a binomially distributed number of active users as specified above, we also consider the worst-case scenario, *i.e.*, we analyze $R_{\text{res}}^* = \inf_L R_{\text{res}}(L)$ under any model with batch arrivals.

Further, we are interested in the effective number of bits that is transmitted across the channel per channel use (*i.e.*, net rate), denoted by $R_{\text{net}}(L)$. Taking into account that L users each transmit D bits in a total of $S(L)$ slots that each consist of N channel uses, we have

$$R_{\text{net}}(L) = \frac{LD}{S(L)N} = R_{\text{res}}(L) \frac{D}{N}. \quad (6)$$

Finally, we are interested in the average and worst-case net rate that we denote by $\bar{R}_{\text{net}} = \mathbb{E}[R_{\text{net}}(L)|L > 0]$ and $R_{\text{net}}^* = \inf_L R_{\text{net}}(L)$, respectively.

We will express some of our results in terms of $I_x(a, b)$, the regularized incomplete beta function, which is defined as

$$I_x(a, b) = B_x(a, b)/B_1(a, b), \quad (7)$$

where $B_x(a, b) = \int_0^x t^{a-1}(1-t)^{b-1} dt$. We will use the well-known result that

$$\sum_{L=0}^K q(L) = I_{1-p}(M-K, K+1) = 1 - I_p(K+1, M-K). \quad (8)$$

IV. PRELIMINARIES

In this section we introduce the three different techniques that constitute the random access mechanism. In Section IV-A, we present the physical-layer network coding strategy that is adopted in this paper. Next, we describe a signature coding scheme in Section IV-B. Finally, we discuss the basic idea of tree splitting in Section IV-C.

A. Reliable physical-layer network coding

The key ingredient of the random access strategy that is proposed in this paper is to employ physical-layer network coding (PLNC), *i.e.*, to organize the physical layer in such a way that the receiver can reliably decode sums of messages that are simultaneously transmitted by users. This requires a suitable choice of the forward error correcting codes as well as the decoding mechanism that is used by the receiver. In this section we provide a short introduction to physical-layer network coding and a result from [10] that will be needed later. There are various angles at which physical-layer network coding can be approached, *e.g.*, denoise-and-forward [8] or compute-and-forward [10]. A survey of these and other approaches is given in [31] and [32]. In this paper we adopt the compute-and-forward framework, as developed by Nazer and Gastpar in [10].

In order to formulate the result from [10] that we need in the remainder, we consider an arbitrary number of L transmitters. User ℓ has data W_ℓ to transmit, where

$$W_\ell = (W_\ell(1), W_\ell(2), \dots, W_\ell(\kappa)), \quad (9)$$

with $W_\ell(j) \in \mathbb{F}_q$, q prime and κ is the predefined message length. Each transmitter uses the same linear code F to encode

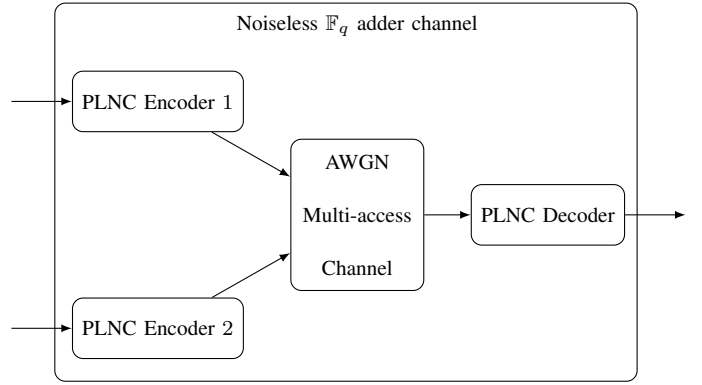


Fig. 1. Physical-layer network coding (PLNC) results in a noiseless \mathbb{F}_q adder channel. ($K = 2$ users)

the data into real-valued channel input of length N (*i.e.*, the length of a slot) that satisfies an average power constraint P . Let $X_\ell = F(W_\ell)$ denote the channel input of user ℓ . The decoder, upon observing $Y = \sum_{\ell=1}^L X_\ell + Z$ attempts to decode $\sum_{\ell} W_\ell = (\sum_{\ell} W_\ell(1), \dots, \sum_{\ell} W_\ell(\kappa))$. In this sense, the receiver recovers a function (namely, the sum) of the original messages, which is why this approach is referred to as computation coding. In a sense, as illustrated in Figure 1, we turn the multi-access AWGN channel in a noiseless \mathbb{F}_q adder channel.

We denote by R_{plnc} the rate of F , *i.e.*, $R_{\text{plnc}} = \kappa N^{-1} \log_2 q$ bits per channel use. We will refer to R_{plnc} as the computation rate and say that it is achievable if the probability of decoding erroneously can be made arbitrarily small by increasing N . The next result follows directly from the main result in [10], using the notation

$$\log_2^+(x) = \begin{cases} \log_2(x), & \text{if } x \geq 1, \\ 0, & \text{otherwise.} \end{cases}$$

Theorem 1 (*cf.* [10], Theorem 1). *For the standard AWGN multiple-access channel, the following computation rate is achievable*

$$R_{\text{plnc}} = \frac{1}{2} \log_2^+(P). \quad (10)$$

The above result does not exactly match the achievable rate as given in [10, Theorem 1], which is $\frac{1}{2} \log_2^+(\frac{1}{L} + P)$. Since we will be dealing with an unknown number of active users, we use a computation rate that corresponds to $L \rightarrow \infty$ and is thus a lower bound that is valid for arbitrary number of active users.

The proof of Theorem 1 in [10] is based on a random coding argument in which the code F is a lattice that is obtained through Construction A, *cf.* [33]. As a consequence, the value of q is a function of the block length N that satisfies $N/q \rightarrow 0$ as $N \rightarrow \infty$, see for instance [34]. We will assume that

$$q = N \log_2 N \quad (11)$$

in the remainder.

B. Signature codes

We are interested in signature codes for the \mathbb{F}_q adder channel, when up to K random users, out of total M users,

are active. So far, there has been a lot of work investigating the case when the signature symbols are binary, *i.e.*, $q = 2$; a summary of the known asymptotic results has been presented in [35]. However, the case of general q has been significantly less studied. We mention here the construction proposed in [13], which can be generalized to any q . However, this construction results in codewords of length M , and allows for simultaneous resolution of up to $K = (M - 1)/4$ signatures when the set of the contending users is a priori not known. In this paper we adopt a construction that does not require a fixed relation of K with respect to M and allows for significantly shorter signatures, as elaborated below.

1) *A result in additive number theory:* In this paper, we adopt the signature code construction presented by Lindström in [36], [35, pp. 42 - 43]. The construction appeared in [36] for the case $q = 2$, but can be readily generalized to arbitrary prime q . The construction is designed for the case that the number of users M is a prime; if M is not a prime, one could design signatures for the smallest prime larger than M and use only the first M signatures. The construction by Lindström is based on the following result in additive number theory by Bose and Chowla [37], which, for convenience of the reader, we present in a slightly less general form than that in [37].

Theorem 2 ([37]). *Let M be prime. There exist integers s_i , $i = 1, \dots, M$, $1 \leq s_i < M^K$, such that*

$$\sum_{i \in \mathcal{L}_1} s_i \neq \sum_{i \in \mathcal{L}_2} s_i, \quad (12)$$

for any $\mathcal{L}_1, \mathcal{L}_2 \subset \{1, \dots, M\}$, $|\mathcal{L}_1| \leq K$, $|\mathcal{L}_2| \leq K$ and $\mathcal{L}_1 \neq \mathcal{L}_2$.

Before giving the details, we describe the main idea in Lindström's construction. Each integer s_i is expressed through a r -ary representation. These r -ary representations are mapped to \mathbb{F}_q and used as the signatures of the users. The choice of r and the mapping to \mathbb{F}_q are such that, from the summation in \mathbb{F}_q of the signatures, the receiver can recover the sum of the integers s_i and thereby the identities of the users. In particular, we will ensure that $K(r - 1) \leq q - 1$. Next, we describe the details of our construction.

2) *Signature encoder:* We assume that a set of integers s_1, \dots, s_M , satisfying the conditions of Theorem 2, is given. The signature of user ℓ , denoted by W_ℓ^s , is a sequence of symbols from \mathbb{F}_q . The signature consists of two parts, each of them with a different functionality. The first part consists of a single symbol, whose value is fixed to 1 by each of the users. In this way, the sum of the first symbols of all active users \mathcal{L} will provide to the receiver $L = |\mathcal{L}|$. Note that this requires that $M \leq q - 1$ and hence a sufficiently long block length, since $q = N \log_2 N$. The second part of the signature of user i consists of the r -ary representation of s_i , where the symbol values are mapped from $\{0, \dots, r - 1\}$ to \mathbb{F}_q using the natural mapping.

3) *Signature decoder:* The signature decoder receives a sum $\sum_{\ell \in \mathcal{L}} W_\ell^s$, where the set \mathcal{L} is unknown to the receiver. The tasks are now the following: i) decide if $L \leq K$, and ii) if $L \leq K$ compute \mathcal{L} ; else, treat the received slot as a collision slot.

The first operation of the decoder is to map the sequence $\sum_{\ell \in \mathcal{L}} W_\ell^s$ from \mathbb{F}_q to a sequence of integers using a natural mapping. Through the sum of the first symbols, the receiver detects the number of active users L .

Next, consider the case when $L \leq K$. Since $K(r - 1) \leq q - 1$, the addition of at most K symbols $\{0, \dots, r - 1\}$ in \mathbb{F}_q will be equivalent to the addition over the integers. Therefore, the decoder can recover the sum of the r -ary representations of the s_ℓ , $\ell \in \mathcal{L}$. More precisely, each symbol in this summation is an element from $\{0, \dots, K(r - 1)\}$ that is obtained by adding integers from $\{0, \dots, r - 1\}$. The resulting sum immediately provides $\sum_{i \in \mathcal{L}} s_i$ and, by Theorem 2, also provides \mathcal{L} .

In case a collision is detected and $L > K$, the receiver can reliably compute the sum of the data of the active users, but cannot decode the identities of the active users, *i.e.*, who contributed to the sum. The resolution of the active user identities and their data packets in this case is addressed by the proposed scheme, as introduced in Section V.

4) *Analysis of signature length:* We now fix the value of r to

$$r = \left\lfloor \frac{q - 1}{K} \right\rfloor, \quad (13)$$

such that $K(r - 1) \leq q - 1$ is satisfied. This leads to the following upper bound on the lengths of the signatures. We express the length in the number of q -ary symbols that are used. The reason is that this number will determine the overhead of the signatures in the overall scheme as introduced in Section V.

Theorem 3. *There exist signatures of length*

$$N_w = \left\lceil \frac{K \log_2 M}{\log_2(q - 1) - \log_2 K} \right\rceil + 1, \quad (14)$$

r -ary symbols, where $r = \lfloor (q - 1)/K \rfloor$. These r -ary symbols can be represented by an equal number of q -ary symbols.

Proof: The length of the signatures follows from Theorem 2. In particular, the signatures are represented by $K \log_r M + 1$ long strings of r -ary symbols. The representation in q -ary symbols follows because $r \leq q$. ■

In the proposed random access strategy we will concatenate our signatures with a message. We will achieve reliable communication by considering asymptotically long messages, *i.e.*, we let $N \rightarrow \infty$. This implies that $q \rightarrow \infty$, since $q = N \log_2 N$. The influence of the signatures on the rate is negligible, *i.e.*, the ratio N_w/N is vanishing.

C. Tree splitting

We briefly outline the basic binary tree-splitting algorithm under a collision model [2]. Again, \mathcal{L} denotes the set of active users and $L = |\mathcal{L}|$, $1 \leq L \leq M$, denotes the number of active users. In the first slot all L users transmit a packet. If $L = 1$, the receiver successfully decodes the packet of the user and the contention period ends. If $L \geq 2$, a collision occurs and the receiver cannot decode the packets. The users then probabilistically split into two groups \mathcal{L}_1 and \mathcal{L}_2 . The splitting is uniform at random and independent over users, *i.e.*, each user flips a fair coin to decide on the group to join.

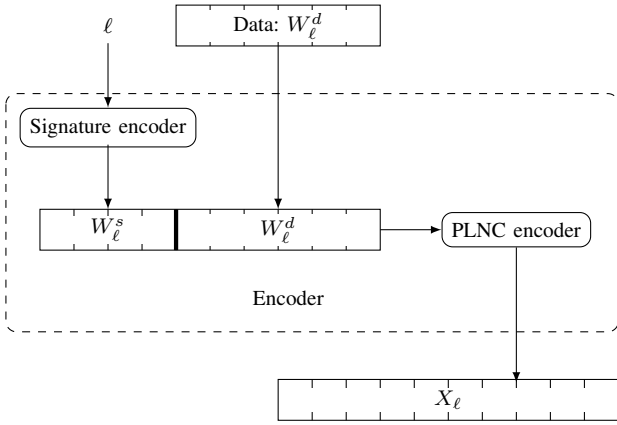


Fig. 2. Illustration of the encoder for user ℓ in a slot.

Both groups then contend for the medium in the same fashion: first the users from \mathcal{L}_1 , then the users from \mathcal{L}_2 . The splitting is done recursively, eventually leading to an instance in which only a single user is active and the corresponding transmission is successfully received. The algorithm continues until the transmissions of all active users from \mathcal{L} are successfully received. By means of feedback, after each slot the receiver informs the users whether there was a collision, a single transmission, or no transmission present, directing the future actions of the active users.

The above described algorithm and its variations were thoroughly analyzed in the literature, assessing the performance parameters such as throughput, delay and stability. The work closest to ours is presented in [16], the most important difference being that we investigate a generalized case in which collisions occur when $L > K$, where $K \geq 1$. The related analysis, which also covers the special case $K = 1$, is presented in Section V.

V. THE PROPOSED STRATEGY

We start with an overview of the proposed random access strategy. The strategy operates in rounds, where a round includes (i) a slot in which the active users transmit the PLNC encoded concatenation of their signatures and payloads, and (ii) the corresponding feedback from the common receiver. The use of PLNC enables the receiver to reliably obtain the q -ary sums of the user transmissions. As long as there are at most K active users, the receiver is able to uniquely decode their signatures, detect which users are active and direct them towards solving the linear combination of their payloads.

The receiver is also able to detect when more than K users are active via the sum of indicator symbols contained in the signatures. In this case, the receiver instructs the users to randomly split in two groups and the strategy is then executed in a recursive fashion for each of these groups. We proceed by presentation of the details.

A. Encoder

Let W_ℓ^s and W_ℓ^d denote the strings representing the signature and the data payload, respectively, of the active user ℓ . The

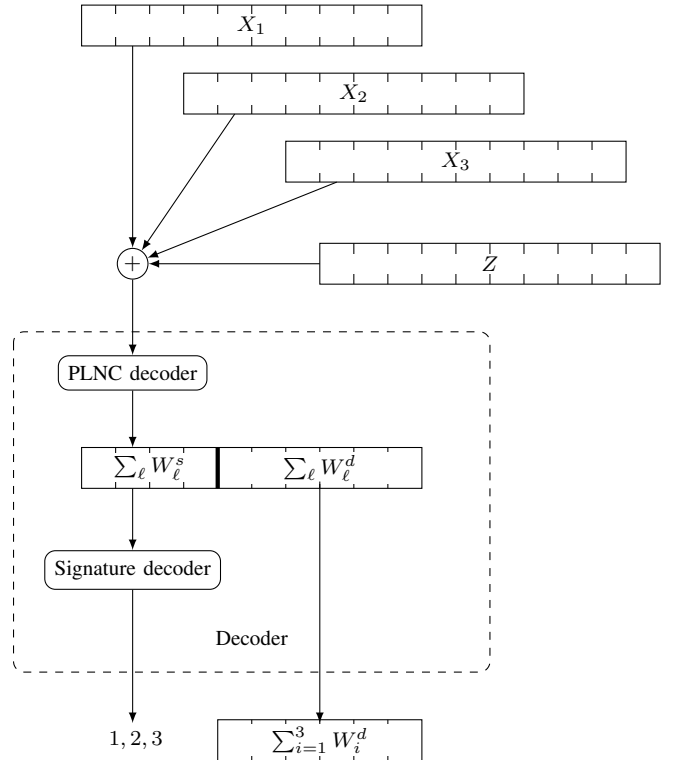


Fig. 3. Illustration of the decoder in a slot. ($L = 3$ users).

concatenation of signature and payload $W_\ell = W_\ell^s || W_\ell^d$ is used as the input of a PLNC encoder. Recall from Section IV-A that the PLNC encoder applies a linear forward error correcting code, the same code F for all users. The output of the PLNC encoder, denoted by $X_\ell = F(W_\ell) = F(W_\ell^s || W_\ell^d)$, is a channel input of user ℓ . The operation of the encoder of a single user in a slot is illustrated in Figure 2.

B. Decoder

The receiver observes Y , which is a real sum of X_ℓ , $\ell \in \mathcal{L}$, and additive noise Z ,

$$Y = \sum_{\ell \in \mathcal{L}} X_\ell + Z = \sum_{\ell \in \mathcal{L}} F(W_\ell) + Z. \quad (15)$$

It uses a PLNC decoder to decode Y and obtain

$$\sum_{\ell \in \mathcal{L}} W_\ell, \quad (16)$$

which decomposes into the sums of the signatures $\sum_{\ell \in \mathcal{L}} W_\ell^s$ and the sums of the codewords $\sum_{\ell \in \mathcal{L}} W_\ell^d$. This is illustrated in Figure 3. As explained in Section IV-A, since the first symbol in the signatures of all users is 1, the receiver directly obtains the number of active users $L = |\mathcal{L}|$,

C. User resolution for $L \leq K$

It follows from the properties of the signature code, that if $L \leq K$, the receiver obtains \mathcal{L} . Moreover, it also has the sum of the messages $\sum_{\ell \in \mathcal{L}} W_\ell^d$. By making use of the feedback mechanism to the users, the receiver ensures that in the next $L - 1$ rounds $L - 1$ of the users in \mathcal{L} are individually

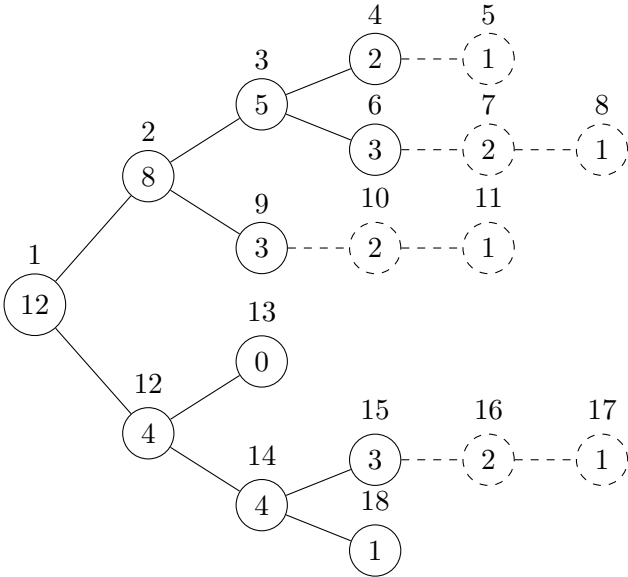


Fig. 4. Illustration of tree splitting, $K = 3$. Each node represents a transmission. The label above a node indicates the slot in which the transmission takes place and the number inside a node indicates the number of transmitting users. The dashed nodes (and corresponding edges) are obtained through polling/scheduling instead of splitting.

transmitting their messages. This can be achieved by, *e.g.*, polling the users through the feedback at the end of a round and requesting them to transmit in the next slot. In that case the feedback acts as an ACK as well as a scheduling mechanism. Note that such a polling mechanism is not possible in the case when $L > K$, since the receiver then does not know the identities of the users.

D. User resolution for $L > K$

When the receiver observes $L > K$, then this is a “collision” in our generalized setting and the receiver signals this fact via feedback. All users in \mathcal{L} now participate in a splitting protocol with uniform splits into two groups. Each user independently of the other users draws a uniformly distributed random number from $\{1, 2\}$. All users with value 1 enter a new contention resolution phase. The users with value 2 wait until this phase ends and start another contention resolution phase afterwards. If there are more than K users in one of these groups the splitting procedure is applied recursively. The splitting protocol is illustrated in Figure 4.

In the next section we analyze the proposed strategy, parameterized on the values of K . Note that case $K = 1$ reduces the scheme to the traditional tree splitting protocol that was discussed in Section IV-C.

VI. ANALYSIS

A. Expected length of the contention resolution phase and expected throughput

We provide an analysis in terms of a recursive expression for the expected number of slots in a contention resolution period given the number of active users L , denoted as $S(L)$.

The analysis is similar to the one by Massey [16] that deals with the case $K = 1$. Let

$$\alpha^* = 1 + \frac{1}{K}, \quad (17)$$

$$\beta^* = 1 + \frac{2}{(K+1)(1-2^{-K})}. \quad (18)$$

We will show that

$$\alpha^*L - 1 \leq S(L) \leq \beta^*L - 1. \quad (19)$$

Let $p_L(\ell)$ denote the probability that a group of L users split into two groups where one of the groups has size ℓ . We have

$$p_L(\ell) = \binom{L}{\ell} 2^{-L}. \quad (20)$$

Note that $p_L(0) > 0$, *i.e.*, it is possible that there are groups with no users, thus we also include the case that $L = 0$. We start by stating the recursion:

Lemma 1.

$$S(L) = \begin{cases} 1, & \text{if } L = 0, \\ L, & \text{if } 1 \leq L \leq K, \\ \frac{1 + 2 \sum_{i=0}^{L-1} p_L(i) S(i)}{1 - 2p_L(L)}, & \text{if } L > K. \end{cases} \quad (21)$$

Proof: Since we have a K out of M signature code, we have $S(0) = 1$, $S(1) = 1$, $S(2) = 2, \dots, S(K) = K$. For $L > K$ we have the following recursion

$$S(L) = 1 + \sum_{i=0}^L p_L(i) \{S(i) + S(L-i)\} \quad (22)$$

$$= 1 + \sum_{i=0}^L \{p_L(i)S(i) + p_L(L-i)S(L-i)\} \quad (23)$$

$$= 1 + 2 \sum_{i=0}^L p_L(i) S(i), \quad (24)$$

which can be rewritten as

$$S(L) = \frac{1 + 2 \sum_{i=0}^{L-1} p_L(i) S(i)}{1 - 2p_L(L)}, \quad (25)$$

by making use of $\binom{L}{L-i} = \binom{L}{i}$ and $\sum_{i=1}^L p_L(i) S(L-i) = \sum_{i=1}^L p_L(i) S(i)$. ■

For notational convenience, let $\gamma(L)$, $L > K$, be defined as

$$\gamma(L) = \frac{\sum_{i=0}^K (S(i) + 1) p_L(i)}{\sum_{i=0}^K p_L(i) i} = 1 + \frac{1 + \sum_{i=0}^K \binom{L}{i}}{\sum_{i=0}^K \binom{L}{i} i}. \quad (26)$$

The reason for introducing $\gamma(L)$ is that it can be used to express bounds on $S(L)$, as demonstrated next.

Lemma 2. *If α and β satisfy*

$$\alpha \leq \gamma(L) \leq \beta \quad (27)$$

for all $L > K$, then

$$\alpha L - 1 \leq S(L) \leq \beta L - 1 \quad (28)$$

for all $L > K$.

Proof: For all $L \leq K$ we have

$$S(L) \leq \beta L - 1 + \sum_{i=0}^K \delta_{iL} (S(L) - \beta L + 1), \quad (29)$$

K	α^*	β^*
1	2	3
2	1.5	1.889
4	1.25	1.427
8	1.125	1.223
16	1.063	1.118

where δ_{ij} is the Kronecker delta, defined to be 1 if $i = j$ and 0 otherwise. We now induction on L to show that (29) holds for all L , thereby providing the proof for the upper bound in (28).

As an induction hypothesis, assume that (29) holds for $S(K+1), S(K+2), \dots, S(L-1)$, $L > K+1$. Then, we have the following bound for $S(L)$

$$S(L) = \frac{1 + 2 \sum_{i=0}^{L-1} p_L(i) S(i)}{1 - 2p_L(L)} \quad (30)$$

$$\leq \frac{1 + 2 \sum_{i=0}^{L-1} p_L(i) (\beta i - 1)}{1 - 2p_L(L)} + \frac{2 \sum_{i=0}^K p_L(i) (S(i) - \beta i + 1)}{1 - 2p_L(L)} \quad (31)$$

$$= \frac{1 + 2 \sum_{i=0}^{L-1} p_L(i) (\beta i - 1)}{1 - 2p_L(L)} + \frac{2(\gamma(L) - \beta) \sum_{i=0}^K p_L(i) i}{1 - 2p_L(L)} \quad (32)$$

$$\leq \frac{1 + 2 \sum_{i=0}^{L-1} p_L(i) (\beta i - 1)}{1 - 2p_L(L)} \quad (33)$$

$$= \beta L - 1, \quad (34)$$

where (30) follows from Lemma 1, (31) from the induction hypothesis, (32) from the definition of $\gamma(L)$ in (26), (33) from condition (27), and finally, (34) from

$$\sum_{i=0}^{L-1} p_L(i) i = \sum_{i=0}^L p_L(i) i - p_L(L) L = \frac{L}{2} (1 - 2p_L(L)), \quad (35)$$

$$\sum_{i=0}^{L-1} p_L(i) = 1 - p_L(L). \quad (36)$$

The establishes the upper bound on $S(L)$. The proof of the lower bound follows in entirely analogous fashion. ■

Next, we provide an upper and a lower bound on $\gamma(L)$.

Lemma 3.

$$1 + \frac{1}{K} \leq \gamma(L) \leq 1 + \frac{2}{(K+1)(1-2^{-K})}. \quad (37)$$

Proof: For the lower bound we have

$$\gamma(L) = 1 + \frac{1 + \sum_{i=0}^K \binom{L}{i}}{\sum_{i=0}^K \binom{L}{i} i} \quad (38)$$

$$\geq 1 + \frac{\sum_{i=0}^K \binom{L}{i}}{K \sum_{i=0}^K \binom{L}{i}} \quad (39)$$

$$= 1 + \frac{1}{K} = \alpha^*. \quad (40)$$

TABLE I
VALUES FOR α^* AND β^* THAT SERVE IN THE BOUNDS ON $S(L)$.

In order to prove the upper bound, we first show that $\gamma(L)$ is decreasing in L , i.e., that $\gamma(L) - \gamma(L+1) \geq 0$. It immediately follows from (26) that $\gamma(L) - \gamma(L+1) \geq 0$ if

$$\sum_{i=0}^K \binom{L+1}{i} i + \sum_{i=0}^K \binom{L}{i} \sum_{j=0}^K \binom{L+1}{j} j - \sum_{i=0}^K \binom{L}{i} i - \sum_{i=0}^K \binom{L+1}{i} \sum_{j=0}^K \binom{L}{j} j \geq 0. \quad (41)$$

Since $\binom{L+1}{i} \geq \binom{L}{i}$ for all $0 \leq i \leq K < L$, it is sufficient to show that

$$\sum_{i=0}^K \binom{L}{i} \sum_{j=0}^K \binom{L+1}{j} j - \sum_{i=0}^K \binom{L+1}{i} \sum_{j=0}^K \binom{L}{j} j \quad (42)$$

is non-negative. Using $\binom{L+1}{i} = \frac{L+1}{L+1-i} \binom{L}{i}$ we rewrite (42) as

$$\sum_{i,j=0}^K \left(\frac{L+1}{L+1-j} - \frac{L+1}{L+1-i} \right) j \binom{L}{i} \binom{L}{j}. \quad (43)$$

Now, (41) follows from the observation that negative terms occur in (43) for $i > j$, in which case, by symmetry considerations, there are corresponding positive $j > i$ terms that are $j-i$ times larger by absolute values than their negative counterparts. The upper bound now follows directly from the value of $\gamma(L)$ at $L = K+1$, i.e.,

$$\gamma(K+1) = 1 + \frac{1 + \sum_{i=0}^K \binom{K+1}{i}}{\sum_{i=0}^K \binom{K+1}{i} i} \quad (44)$$

$$= 1 + \frac{1 + (2^{K+1} - 1)}{2^K(K+1) - (K+1)} \quad (45)$$

$$= 1 + \frac{2}{(K+1)(1-2^{-K})} = \beta^*. \quad (46)$$

■ We are now ready to state the main result of this section.

Theorem 4. $S(L) = L$ if $1 \leq L \leq K$, and, for $L > K$

$$\alpha^* L - 1 \leq S(L) \leq \beta^* L - 1. \quad (47)$$

Proof: Directly from Lemmas 2 and 3. ■

In Table I we provide a numerical evaluation of α^* and β^* . Also, in Figure 5 we illustrate $S(L)$, as well as its lower and upper bounds for various values of K . Finally, in Figure 6 we illustrate the expected number of users that is resolved per slot given that L users are active, $R_{\text{res}}(L) = L/S(L)$, including its upper and lower bounds derived from the bounds on $S(L)$.

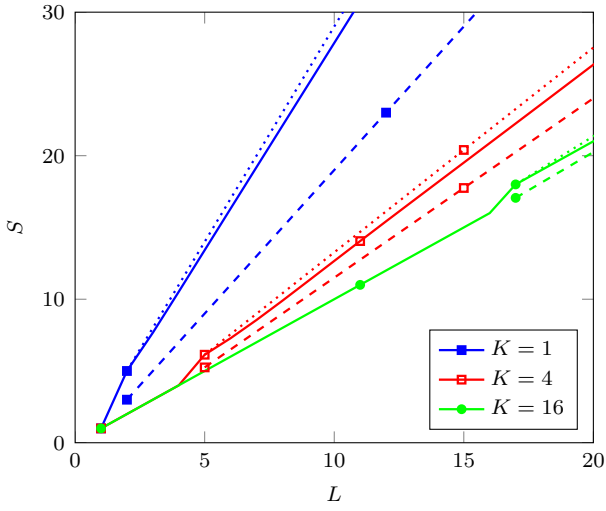


Fig. 5. $S(L)$ and its bounds for various values of K . Upper and lower bounds in dotted and dashed lines, respectively. Exact values of $S(L)$ in solid lines.

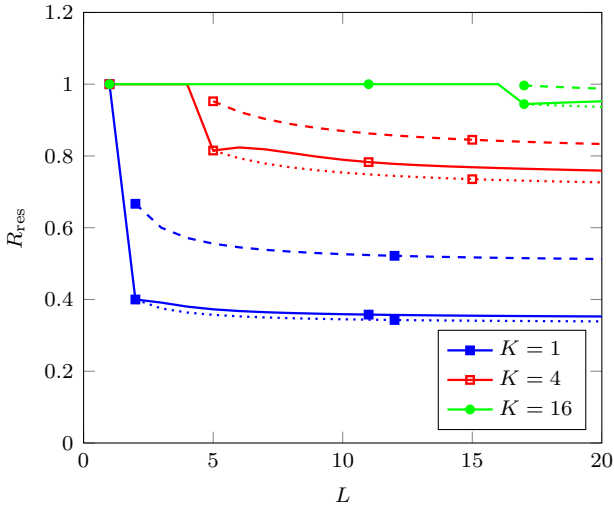


Fig. 6. $R_{\text{res}}(L)$ and its bounds for various values of K . Upper and lower bounds in dotted and dashed lines, respectively. Exact values of $R_{\text{res}}(L)$ in solid lines.

From Theorem 4, we derive results on the expected number of users that is resolved per slot \bar{R}_{res} , *i.e.*, the expected throughput.

Theorem 5. *The expected number of users that is resolved per slot is lower bounded as*

$$\bar{R}_{\text{res}} \geq 1 - \frac{\beta^* - 1}{\beta^*(1 - q_0)} I_p(K + 1, M - K). \quad (48)$$

Proof: We have

$$\bar{R}_{\text{res}} = \sum_{L=1}^M \frac{L}{S(L)} \hat{q}(L) \quad (49)$$

$$\geq \sum_{L=1}^K \hat{q}(L) + \sum_{L=K+1}^M \frac{L}{\beta^* L - 1} \hat{q}(L) \quad (50)$$

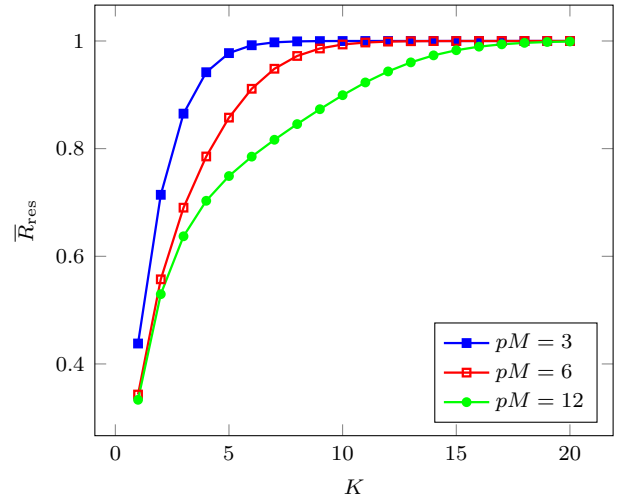


Fig. 7. Lower bounds on \bar{R}_{res} , the expected number of users that is resolved per slot. ($M = 1031$)

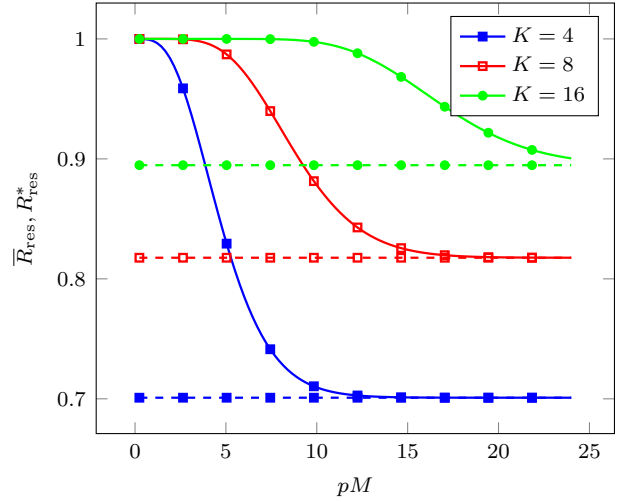


Fig. 8. Average and worst case number of users that is resolved per slot ($M = 1031$). R_{res}^* in dashed lines, \bar{R}_{res} in solid lines.

$$\geq (1 - q_0)^{-1} \left(\sum_{L=0}^K q(L) + \frac{1}{\beta^*} \sum_{L=K+1}^M q(L) - q_0 \right) \quad (51)$$

$$= 1 - \frac{\beta^* - 1}{\beta^*(1 - q_0)} I_p(K + 1, M - K), \quad (52)$$

where $\hat{q}(L)$ is the probability of having $L \geq 1$ active users and $I_p(K + 1, M - K)$ is the regularized incomplete beta function, see (7). ■

The result is illustrated in Figure 7 as a function of K for various values of p .

In addition to the expected throughput \bar{R}_{res} , we are also interested in the worst-case behavior, *i.e.*, we analyze $R_{\text{res}}^* = \inf_L R_{\text{res}}(L)$. The next result is an immediate corollary of Theorem 4.

Corollary 1.

$$R_{\text{res}}^* \geq \frac{1}{\beta^*}. \quad (53)$$

We illustrate Corollary 1 in Figure 8, depicting both R_{res}^* as well as \bar{R}_{res} for various values of K . The value of \bar{R}_{res} is given as a function of pM , which is the average number of active users.

B. Net rate in bits per channel use

Here we consider the net rate $R_{\text{net}}(L)$, *i.e.*, the overall throughput in bits per channel use that is effectively transmitted. This performance parameter takes into account the overhead that is generated by the physical-layer network coding, signatures and the tree-splitting. In this section we ignore finite block length effects and assume that PLNC can achieve rate $R_{\text{plnc}} = \frac{1}{2} \log_2^+(P)$ (as given by Theorem 1) with arbitrarily small error probability at a finite block length.

First, we analyze the number of message bits D that can be transmitted within a block of N channel uses. Let k denote the number of q -ary symbols that can be transmitted. We have

$$\frac{k}{N} \log_2 q = R_{\text{plnc}}. \quad (54)$$

Combining the message bits and the signature we have

$$k = \frac{D}{\log_2 q} + N_w, \quad (55)$$

where $q = N \log_2 N$ and N_w is given by Theorem 3. From this we can solve for D as a function of N and evaluate $R_{\text{net}}(L) = \frac{D}{N} R_{\text{res}}(L)$. We obtain

$$D = \frac{N}{2} \log_2^+(P) - N_w \log_2(N \log_2 N). \quad (56)$$

For completeness, we give the full result in the following corollary.

Corollary 2. *The expected number of bits per channel use \bar{R}_{net} is at least*

$$\bar{R}_{\text{net}} \geq \left(\frac{1}{2} \log_2^+(P) - \frac{N_w}{N} \log_2(N \log_2 N) \right) \cdot \left(1 - \frac{\beta^* - 1}{\beta^*(1 - q_0)} I_p(K + 1, M - K) \right), \quad (57)$$

where $N_w = \left\lceil \frac{K \log_2 M}{\log_2(N \log_2 N - 1) - \log_2 K} \right\rceil + 1$ and N is the block length.

In the next result we consider the case that $D, N \rightarrow \infty$ and consider the maximum of \bar{R}_{net} over K . The result states that the resulting net rate is $\frac{1}{2} \log_2^+(P)$.

Theorem 6. *As D increases, the value of \bar{R}_{net} optimized over K is $\frac{1}{2} \log_2^+(P)$, *i.e.*,*

$$\max_K \lim_{D \rightarrow \infty} \bar{R}_{\text{net}} = \frac{1}{2} \log_2^+(P). \quad (58)$$

Proof: First,

$$\lim_{D \rightarrow \infty} \bar{R}_{\text{net}} = \frac{1}{2} \log_2^+(P) \left(1 - \frac{\beta^* - 1}{\beta^*(1 - q_0)} I_p(K + 1, M - K) \right). \quad (59)$$

Now, for $K = M$, the regularized incomplete beta function is at its minimum value $I_p(M + 1, 0) = 0$ and $(\beta^* - 1)/\beta^*/(1 -$

$q_0)$ is finite. Therefore, the maximum on the right-hand side in the above expression is obtained for $K = M$ and equals $\frac{1}{2} \log_2^+(P)$. ■

C. Upper Bound

We consider an upper bound on R_{net} that must be satisfied by any multiple access protocol that serves a batch of arrived packets, each at a different user:

Theorem 7.

$$\bar{R}_{\text{net}} \leq \sum_{L=1}^M \binom{M}{L} p^L (1-p)^{M-L} \frac{1}{2} \log_2(1 + LP). \quad (60)$$

Proof: The bound is obtained by assuming that all users and the receiver have complete knowledge about which users are active. Under these assumptions, the problem reduces to a standard Gaussian multi-access channel. The sum rate that can be used by L active users is

$$R_{\text{net}}(L) \leq \frac{1}{2} \log_2(1 + LP). \quad (61)$$

This immediately leads to (60) by taking the expectation over L . ■

The following corollary follows directly from Theorem 7 by an application of Jensen's inequality.

Corollary 3.

$$\lim_{M \rightarrow \infty} \bar{R}_{\text{net}} \leq \frac{1}{2} \log_2(1 + pMP). \quad (62)$$

In the next subsection we will interpret the upper bound from this section and relate it to the value of \bar{R}_{net} achieved by the proposed scheme.

D. Evaluation

Figure 7 shows that as a function of K , \bar{R}_{res} quickly approaches the maximum value of 1. This performance parameter is a baseline measure of the efficiency of the random access protocols from the system perspective. Our results clearly demonstrate the potential of the proposed strategy. Figure 7 also shows that K should increase as the expected number of the active users pM increases, in order to achieve high throughput. Conversely, if K is fixed, the expected throughput drops with p to its lower bound, as shown in Figure 8.

In Figure 9 we have illustrated our lower bound on \bar{R}_{net} from Corollary 2 as a function of D , the size of a message transmitted by a user. Note that the lowest value for D in the figure is $D = 500$, for which at $K = 1$ and $P = 100$, $N \approx 156$ and we satisfy the required constraint $M \leq q - 1 = N \log_2 N - 1$ for $M = 1031$. The non-smooth behavior of \bar{R}_{net} is a result of the ceiling operation in the length of the signatures. In addition to \bar{R}_{net} , Figure 9 depicts in a dashed line the upper bound from Theorem 7, demonstrating what is the price to pay in information bits per channel use due to: (i) using physical-layer network coding to decode a linear combination of messages in each slot, (ii) the overhead related to the use of signatures, and (iii) the loss caused by the occurrence of collision and empty slots,

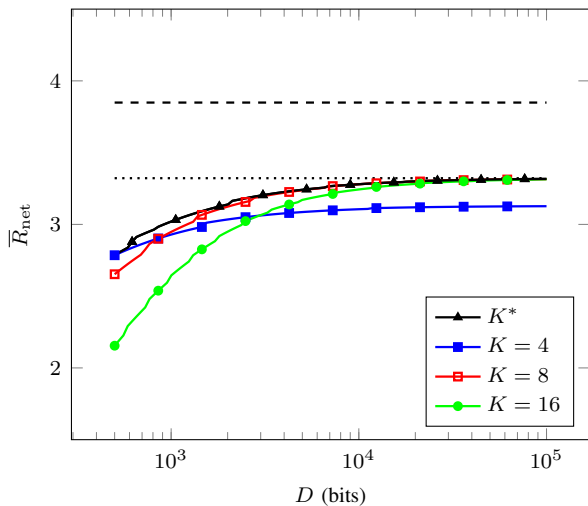


Fig. 9. The lower bound on \bar{R}_{net} from Corollary 2. In dotted line the value $1/2 \log_2^+(P)$ as given by Theorem 6. ($M = 1031$, $pM = 3$, $P = 10^2$)

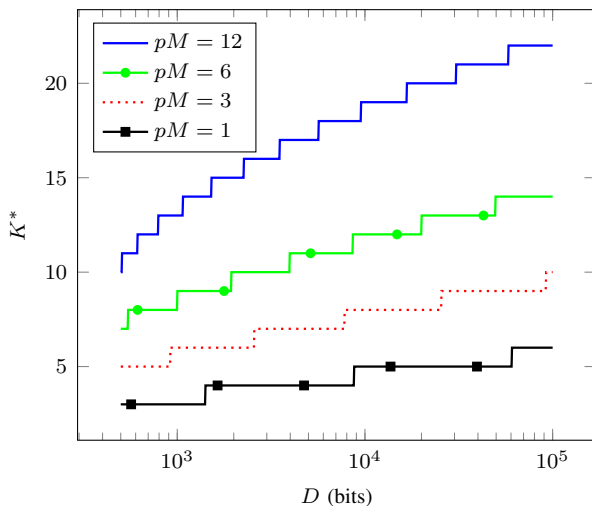


Fig. 10. Value of K that optimizes \bar{R}_{net} , denoted by K^* . ($M = 1031$, $P = 10^2$)

compared to the ideal scenario of beforehand knowing the set of active users and using the optimal multi-user code. Figure 9 also depicts in a dotted line the value $1/2 \log_2^+(P)$. From Theorem 6 it follows that $1/2 \log_2^+(P)$ is the limiting value for \bar{R}_{net} of our scheme for $D \rightarrow \infty$ and $M \rightarrow \infty$. It is clear that no scheme in which one linear combination of messages is decoded at the receiver per slot will be able to achieve a net rate larger than $1/2 \log_2^+(P)$. Therefore, Figure 9 illustrates that the performance degradation due to (ii) and (iii) diminishes as D increases, such that for already modest values of D in practice, the degradation is mainly due to the way in which physical-layer network coding is applied in our approach.

This is also illustrated by considering the upper bound from Corollary 3, which gives $\frac{1}{2} \log_2(1 + pMP)$, where pM is the expected number of active users, whereas our scheme achieves $1/2 \log_2^+(P)$. A similar behavior was observed in [6], in which the PLNC strategy was extended to allow for several

linear combinations to be decoded per slot. Even though this provided a slight improvement, the performance of the resulting strategy did not match the upper bound. Also, using Theorem 7 itself, instead of Corollary 3, will not close this gap. It is an open problem to determine whether the gap to optimality observed in this work and in [6] can be reduced by modifying our approach.

Figure 9 also shows that there is an optimal value of K that minimizes the combined overhead of (ii) and (iii), *i.e.*, maximizes \bar{R}_{net} with respect to D ; the maximum value of \bar{R}_{net} for the optimal K , denoted by K^* , is also depicted in Figure 9. Finally, Figure 10 shows K^* as function of payload length D and the expected number of active users. We note that finding analytical expressions for K^* involves dealing with partial derivatives of the regularized incomplete beta function and is out of the scope of the paper.

VII. INCORPORATING SUCCESSIVE INTERFERENCE CANCELLATION

In this section we consider an extension of the proposed scheme that includes successive interference cancellation. Specifically, in Section V, any computed sum of the received signals in the slot that involves more than K users is considered a collision and discarded. The rationale is that in this case the receiver is not able to determine the set of active users \mathcal{L} that contribute to the sum. However, as indicated in Section IV-A, due to the PLNC properties the receiver can successfully receive a linear combination of the packets of all active users. We show in this section how the receiver can efficiently make use of these collision slots by storing these sums of packets for later use. In the further text, we refer to this scheme as the SIC-enabled scheme.

A. Description of the SIC-enabled scheme

The encoder and decoder that are used in each slot are the same as in the proposed scheme of Section V, *i.e.*, encoding is performed as in Section V-A and decoding as in Section V-B. The difference is in the action taken by the receiver in case $L > K$, *i.e.*, the difference is in the splitting procedure that takes place when more than K users are transmitting. We proceed by describing the details.

Recall from Section V-B that the receiver obtains $\sum_{\ell \in \mathcal{L}} W_\ell$, which consists of the concatenation of the sums of the signatures $\sum_{\ell \in \mathcal{L}} W_\ell^s$ and the sums of the codewords $\sum_{\ell \in \mathcal{L}} W_\ell^d$. If $L > K$ then $\sum_{\ell \in \mathcal{L}} W_\ell^s$ cannot be uniquely decoded to learn \mathcal{L} . The key difference between the scheme of Section V-D and the SIC-enabled scheme is that the receiver stores $\sum_{\ell \in \mathcal{L}} W_\ell$ for future use, instead of discarding it. Again, all users in \mathcal{L} participate in a splitting protocol with uniform splits into two groups. Each user independently of the other users draws a uniformly distributed random number from $\{1, 2\}$. Let \mathcal{L}_i denote the users with value i , $i = 1, 2$. First, all users with value 1 enter a new contention resolution phase. At the end of this phase, which might include recursive splitting steps, the receiver knows \mathcal{L}_1 and $\sum_{\ell \in \mathcal{L}_1} W_\ell$. Using the stored information $\sum_{\ell \in \mathcal{L}} W_\ell$, the receiver now computes

$$\sum_{\ell \in \mathcal{L}_2} W_\ell = \sum_{\ell \in \mathcal{L}} W_\ell - \sum_{\ell \in \mathcal{L}_1} W_\ell. \quad (63)$$

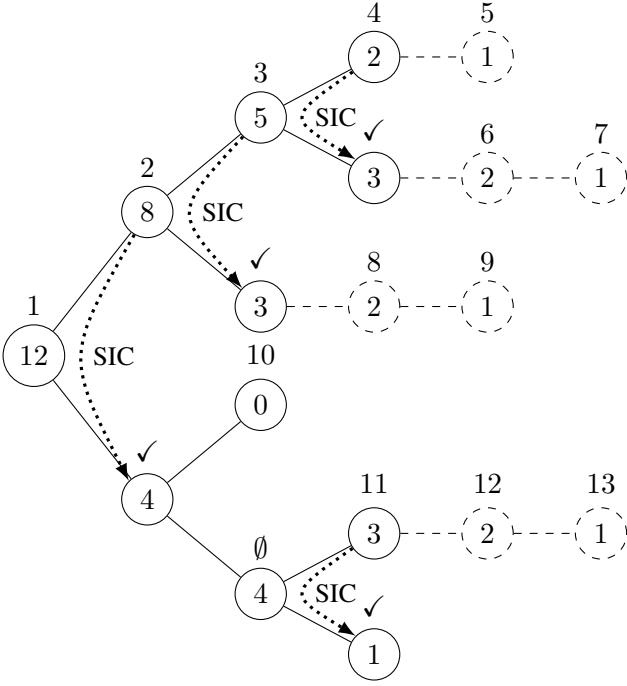


Fig. 11. Tree splitting with SIC, $K = 3$. A checkmark indicates that this signal is obtained by SIC and that no slot is used. The transmission labeled \emptyset is skipped, since it would not bring new information and users are instructed to split.

Note that $\sum_{\ell \in \mathcal{L}_2} W_\ell$ is exactly the signal that would occur in the first slot of the second subtree in the splitting procedure of Section V-D. Thus, instead of being obtained through an additional slot, $\sum_{\ell \in \mathcal{L}_2} W_\ell$ it is obtained through (63). The corresponding slot is omitted and the contention resolution of users in \mathcal{L}_2 proceeds in the similar fashion. An overview of the SIC approach is provided in Figure 11.

An important detail of the scheme that we have not yet discussed is also illustrated in Figure 11. Specifically, if $\mathcal{L}_1 = \emptyset$, then $\mathcal{L}_2 = \mathcal{L}$ and (63) does not provide new information. In this case, the first slot in the second subtree can also be omitted and the users are instructed to immediately split again.¹ Finally, in case $\mathcal{L}_1 = \mathcal{L}$, the second phase is (obviously) omitted completely.

An important difference between our SIC scheme and other SIC-based contention resolution mechanisms [4], [7], is that our approach is based on reliable PLNC, whereas other approaches work on noisy signals. This has the advantage that it leverages the receiver from the burden to store large quantities of physical-layer output with a high precision.

B. Analysis

We start with the equivalent of Lemma 1 for the SIC-enabled scheme. The notation that is used in this section is same the same as Section VI, with an additional subscript SIC whenever there is a difference with the scheme of Section V.

¹A similar modification to the original, $K = 1$ algorithm when $\mathcal{L}_1 = \emptyset$, was proposed in [16]. We also note that the immediate split of the second group, for any value of $|\mathcal{L}_1|$ and given that $|\mathcal{L}_2| > K$, naturally fits into the SIC framework and is the rule, rather than a modification.

Lemma 4.

$$S_{\text{SIC}}(L) = \begin{cases} 1, & \text{if } L = 0, \\ L, & \text{if } 1 \leq L \leq K, \\ \frac{2 \sum_{i=0}^{L-1} p_L(i) S_{\text{SIC}}(i)}{1 - 2p_L(L)}, & \text{if } L > K. \end{cases} \quad (64)$$

Proof: We focus on the recursive expression, when $L \geq K$ and when a split is performed. We distinguish three cases, depending on the number of users in the first group after the split. If there are $1 \leq i \leq L - 1$ users in the first group, then we require $S_{\text{SIC}}(i)$ slots to resolve this group. For the second group, the first linear combination of the $L - i$ users is obtained through SIC and $S_{\text{SIC}}(L - i) - 1$ additional slots are required. If there are no users in the first group, then $S_{\text{SIC}}(0) = 1$ slot is used for the first group. For the second group of L users, the first transmission can be omitted, since it is known in advance that it will not provide a new linear combination. Therefore, $S_{\text{SIC}}(L) - 1$ additional slots are required. Finally, if there are L users in the first group, $S_{\text{SIC}}(L)$ slots will be used for this group and no slots will be used for the second group. Combining all cases leads to

$$S_{\text{SIC}}(L) = 1 + \sum_{i=0}^{L-1} p_L(i) (S_{\text{SIC}}(i) + S_{\text{SIC}}(L - i) - 1) + p_L(L) S_{\text{SIC}}(L), \quad (65)$$

when $L > K$. The proof of the lemma readily follows from (65), using the facts that $S_{\text{SIC}}(0) = 1$ and $p_L(i) = p_L(L - i)$, $i = 0, \dots, L$. ■

The following lemma is the equivalent of Lemma 2. The proof is completely analogous to the proof of Lemma 2 and, therefore, omitted.

Lemma 5. If α_{SIC} and β_{SIC} satisfy

$$\alpha_{\text{SIC}} \leq \gamma_{\text{SIC}}(L) \leq \beta_{\text{SIC}}, \quad (66)$$

for all $L > K$, where

$$\gamma_{\text{SIC}}(L) = \frac{\sum_{i=0}^K S_{\text{SIC}}(i) p_L(i)}{\sum_{i=0}^K p_L(i) i}, \quad (67)$$

then

$$\alpha_{\text{SIC}} L \leq S_{\text{SIC}}(L) \leq \beta_{\text{SIC}} L, \quad (68)$$

for all $L > K$.

Next, we present bounds on $\gamma_{\text{SIC}}(L)$.

Lemma 6.

$$1 \leq \gamma_{\text{SIC}}(L) \leq 1 + \frac{1}{(K + 1)(2^K - 1)}. \quad (69)$$

Proof: Rewrite γ_{SIC} as

$$\gamma_{\text{SIC}}(L) = \frac{p_L(0) + \sum_{i=1}^K p_L(i) i}{\sum_{i=1}^K p_L(i) i} = 1 + \frac{1}{\sum_{i=1}^K \binom{L}{i} i}. \quad (70)$$

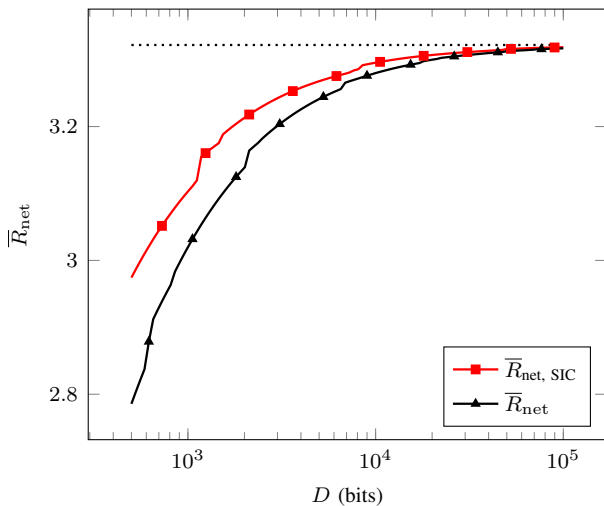


Fig. 12. Lower bounds on \bar{R}_{net} and $\bar{R}_{\text{net,SIC}}$ at the optimal value for K . In dashed line the upper bound on \bar{R}_{net} from Theorem 7. In dotted line the value $1/2 \log_2^+(P)$ as given by Theorem 6. ($M = 1031$, $pM = 3$, $P = 10^2$)

K	β_{SIC}^*
1	1.5
2	1.111
4	1.036
8	1.013

TABLE II

VALUES FOR β^* THAT SERVE AS THE UPPER BOUND ON $S_{\text{SIC}}(L)$.

The lower bound trivially holds. Regarding the upper bound, it should be observed that $\binom{L}{i}$, $i = 0, \dots, K$, increases with L . Therefore

$$\gamma_{\text{SIC}}(L) \leq \gamma_{\text{SIC}}(K+1) \quad (71)$$

$$= 1 + \frac{1}{\sum_{i=1}^K \binom{K+1}{i} i} \quad (72)$$

$$= 1 + \frac{1}{(K+1)(2^K - 1)}. \quad (73)$$

Based on the above lemma we define

$$\alpha_{\text{SIC}}^* = 1, \quad (74)$$

$$\beta_{\text{SIC}}^* = 1 + \frac{1}{(K+1)(2^K - 1)}, \quad (75)$$

to serve as bounds on $S_{\text{SIC}}(L)$ in Lemma 5. In Table II we list some values for β_{SIC}^* . Obviously, the upper bound quickly approaches the lower bound of 1 as K increases, implying that $S_{\text{SIC}}(L)$ also quickly approaches L , which is the minimal number of slots required to resolve L user transmissions.

We derive results on $\bar{R}_{\text{res,SIC}}$, the expected number of users that is resolved per slot in the SIC-enabled scheme. The next result is the equivalent of the non-SIC result in Theorem 5 with β^* replaced by β_{SIC}^* ; the proof is omitted, as it requires only a minor modification to the proof of Theorem 5.

Theorem 8. *The expected number of users that is resolved per slot is lower bounded as*

$$\bar{R}_{\text{res,SIC}} \geq 1 - \frac{\beta_{\text{SIC}}^* - 1}{\beta_{\text{SIC}}^*(1 - q_0)} I_p(K+1, M-K). \quad (76)$$

The result of Theorem 8 immediately leads to a result on $\bar{R}_{\text{net,SIC}}$, *i.e.*, the net rate in bits per channel use that is achievable with SIC. In Figure 12 we have compared maximum values for $\bar{R}_{\text{net,SIC}}$ and \bar{R}_{net} , both at their individual optimal values of K . In addition, similar to Figure 9, we have depicted the upper bound from Theorem 7 and the value $1/2 \log_2^+(P)$ as given by Theorem 6. We observe from Figure 12 that, even though SIC has a significant impact on the value of expected number of slots in a contention period (69) and on the expected number of users resolved per slot (76), the impact on the net rate is limited. In other words, reusing collision slots in the SIC-enabled scheme only modestly improves net-rate performance. We conclude this section by noting that it is straightforward to obtain the other results for the SIC-enabled scheme that correspond to the results in Section VI. Therefore, these are not presented here.

VIII. DISCUSSION

In the paper we have assumed a unit channel gain model in which users are synchronized to the start of a contention resolution period. Here we discuss how the assumption of the unit channel gains can be relaxed to take into account channels with fading, as well as how the synchronization can be achieved. The idea is that the start of a contention resolution period is marked by a beacon sent by the base station, synchronizing the users. Upon receiving the beacon, each user that has a message to send estimates its channel to the base station. This estimate is also obtained from the beacon, assuming channel reciprocity. If the channel is sufficiently strong (to be made precise in the following text), then the user becomes active for this contention period, *i.e.*, it joins the set of contending users. The active user *inverts* the channel and, during the contention process, *precodes* its transmission by sending the signal X_m/h_m , where h_m is the channel coefficient between the m -th user and the receiver. This channel, as perceived by the receiver, has unit channel gains. Note that in this case the uncertainty about the users that constitute the set \mathcal{L} comes both from the sporadic message arrival per user as well as the changes in the channel. We assume a quasi-static fading model, such that h_m stays constant during the contention period, which also implies that the set of active users \mathcal{L} remains invariant until the contention is resolved. Finally, due to the power constraint, if a user observes a channel with $|h_m|^2 < 1/P$, it does not become active and does not join the contention set \mathcal{L} . Such a user will wait for a next contention period in which it has a stronger channel.

It is interesting to note that in standard scenarios with PLNC, the channel coefficients are assumed to be known at the receiver [8]–[10], which is consistent with the fact that the receiver knows a priori the set of transmitting users; in other words, the only issue is to properly select the codebooks and the decoders. However, in our scenario there is uncertainty about the set of transmitting users and it is thus not reasonable

to assume that the receiver would know the channel coefficients $\{h_m\}$. Precoding results in a multiple access channel in which the coefficient of each user, as perceived by the receiver, is equal to one. Thus, although the receiver does not know the set of transmitting users, it knows a priori the channel coefficient by which each user is received. With these assumptions, the receiver is able to set up the correct decision regions in order to decode the superposition of the lattice-based codewords from the transmitters.

The random-access setup considered in the paper can be categorized as a batch contention resolution, where all users arrive before the start of the contention resolution procedure, and the contention resolution is performed by the binary fair-splitting of collisions with multiplicity larger than K . The initial collision multiplicity is binomially distributed, as the user arrivals are modeled by Bernoulli trials. The sporadic nature of message arrivals is reflected in the fact that p is small, such that with high probability a user that is already in \mathcal{L} does not get a new message during the contention period. In order to take into account additional arrivals at users already in \mathcal{L} , we could extend the protocol by allowing a user to indicate with an additional bit in their message that it has one more message. Such a user would be granted another collision-free transmission once it succeeds to send its first message successfully.

In the paper the focus of the analysis of the proposed scheme was on the derivation of the bounds on the protocol performance, characterizing the behavior as K grows, *i.e.*, as a larger number of users becomes resolvable simultaneously. The obtained bounds are given in closed form, depend only on K and provide a direct insight into the scheme's performance. We note that, in principle, one could reuse the results from [38] and [4], and obtain non-recursive expressions for the expected duration of the contention resolution period given that the initial collision multiplicity is L , for the original $S(L)$ and SIC-enabled variant $S_{\text{SIC}}(L)$, respectively. Specifically, the difference to the analysis carried out in [38] and [4] is in the generalization in the range of the initial conditions to $L = 0, 1, \dots, K$. However, taking into account that these expressions for $S(L)/S_{\text{SIC}}(L)$ will inevitably be in a convoluted form², and that the bounds that have been derived in the current paper become increasingly tight as K grows, we have not pursued obtaining non-recursive formulae for $S(L)/S_{\text{SIC}}(L)$.

The straightforward generalization of the proposed scheme is to assume Q -ary splitting and investigate the optimal values of Q and the optimal splitting probabilities, including the potential exploitation of the fact that the collision multiplicity L is always known. However, our preliminary analysis shows that the gain that could be achieved by such optimization is negligible compared to the gain that is achieved by optimizing K .

Further extensions could include more elaborate arrival models, where new users can become active after the start of the (current) contention resolution period, which can be handled in the blocked (gated), windowed and free-access manner [4], [16]. In this regard, we conjecture that, based

on the results from [16], the original variant of the proposed scheme will experience improvements w.r.t. the worst case performance, see Figure 8, if the windowed access is used. On the other hand, we conjecture that the blocked access for the SIC-enabled variant is optimal. Specifically, in this variant of the scheme, all non-empty slots that occur during the contention resolution period are useful, which eliminates the arguments for 'bounding' the initial collision multiplicity that fosters the windowed access. We also note that the analogous result is formally derived in [4] for the case $K = 1$.

We further remark that Figures 9 and 10 clearly illustrate the performance loss due to the access protocol elements, *i.e.*, physical layer network coding, signature coding and the contention resolution mechanism, w.r.t. the ideal scenario in which set of the active users and their identities are a-priori known and the optimal multi-user code is used. Such a comparison is omitted in related work, which consistently relies on idealized assumptions regarding the physical layer and does not take the overhead related to user identification explicitly into account.

As shown in this paper, the dominant loss as the length of the data portion of user packet grows is due to PLNC; it is currently an open problem if this loss is an inherent property of PLNC or an artifact of the computation coding construction that is developed in [10].

The current work has not addressed the complexity of decoding the signatures. It would be of interest to design low-complexity decoding algorithms by basing the construction of the signature codes on, for instance, BCH codes (instead of Lindström's construction) and by using existing decoding techniques as done in [39], [40].

As already mentioned in the introduction, the mechanisms introduced in this paper through signatures and physical-layer network coding are not limited to tree-splitting algorithms. In particular, PLNC and signature coding were also applied in the context of coded random access [41], where the advantage is that the feedback comes less often compared to the tree-splitting protocols. However, the proposed protocol offers several advantages over coded random access. One advantage is related to the sensitivity of the performance to the choice of parameters, which in both cases depend on the number of active users. In this respect, the proposed protocol can provide meaningful performance without a priori knowledge of the number of active users and the optimal selection of K , as demonstrated in Figure 8. On the other hand, the performance of coded random access is much more sensitive, with a chance of performing poorly if the parameters are not selected optimally, *c.f.* Figure 5 in [42] that shows that the throughput becomes very low if the frame has not been dimensioned properly. Further, the use of feedback, which drives the resolution process in the proposed scheme, enables the receiver to start resolving users after a number of slots that scales logarithmically with the initial collision multiplicity. On the other hand, in frame-based coded random access, the resolution process is initiated only at the end of the frame; similarly, frameless coded random access exhibits the well known threshold effect in terms of user resolution, when most of the users become resolved at the end of the contention

²Cf. (3.31) in [38] and (30) in [4].

period [7]. In this respect, it can be argued that the proposed protocol provides a more balanced profile of the user resolution delay. It will be of interest to compare the performance of our proposed strategy with work on models in which the user detection problem is related to compressed sensing, as done in, for instance, [43]. An important difference between [43] and the current work is that in [43] the number of users scales with the block length.

One relevant extension of this work is directed to the practical scenario in which physical-layer network coding is not carried out using lattices, but rather a practical symbol-by-symbol modulation combined with error-control coding. This fact, along with the finite block length, necessarily leads to a nonzero probability of errors at the receiver, which requires suitable modification of the protocol. Another problem of practical relevance is to develop efficient algorithms that decode sums of signatures. Finally, an interesting extension is related to the construction of signatures when the assumption of the equal powers at the point of reception does not hold.

REFERENCES

- [1] L. G. Roberts, "Aloha packet system with and without slots and capture," *SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, pp. 28–42, Apr. 1975.
- [2] J. Capetanakis, "Tree Algorithms for Packet Broadcast Channels," *IEEE Trans. Inf. Theory*, vol. 25, no. 5, pp. 505 – 515, sep 1979.
- [3] M. Tsatsanis, R. Zhang, and S. Banerjee, "Network-assisted Diversity for Random Access Wireless Networks," *IEEE Trans. Signal Processing*, vol. 48, no. 3, pp. 702–711, Mar 2000.
- [4] Y. Yu and G. B. Giannakis, "High Throughput Random Access Using Successive Interference Cancellation in a Tree Algorithm," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4628–4639, Dec. 2007.
- [5] J. Goseling, M. Gastpar, and J. H. Weber, "Physical-layer Network Coding on the Random-access Channel," in *Proc. of IEEE ISIT 2013*, Istanbul, Turkey, Jul. 2013.
- [6] J. Goseling, M. Gastpar, and J. Weber, "Random access with physical-layer network coding," *Information Theory, IEEE Transactions on*, vol. 61, no. 7, pp. 3670–3681, July 2015.
- [7] E. Paolini, C. Stefanovic, G. Liva, and P. Popovski, "Coded Random Access: Applying Codes on Graphs to Design Random Access Protocols," *IEEE Commun. Mag.*, vol. 53, no. 6, p. 144150, Jun. 2015.
- [8] P. Popovski and H. Yomo, "Physical Network Coding in Two-Way Wireless Relay Channels," in *Proc. of IEEE ICC 2007*, Glasgow, UK, Jun. 2007.
- [9] —, "The Anti-Packets Can Increase the Achievable Throughput of a Wireless Multi-Hop Network," in *Proc. of IEEE ICC 2006*, Istanbul, Turkey, Jun. 2006.
- [10] B. Nazer and M. Gastpar, "Compute-and-Forward: Harnessing Interference Through Structured Codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.
- [11] J. Goseling, "A Random Access Scheme with Physical-layer Network Coding and User Identification," in *Proc. of IEEE ICC 2014, Workshop on Massive Uncoordinated Access Protocols*, Sydney, Australia, Jun. 2014.
- [12] J. Goseling, C. Stefanovic, and P. Popovski, "Sign-Compute-Resolve for Random Access," in *52nd Annual Allerton Conference*, Monticello, IL, USA, Sep. 2014.
- [13] P. Mathys, "A Class of Codes for a T Active Users out of N Multiple-Access Communication System," *IEEE Trans. Inf. Theory*, vol. 36, no. 6, p. 12061219, Nov. 1990.
- [14] D. B. Jevtic, "On families of sets of integral vectors whose representatives form sum-distinct sets," *SIAM Journal on Discrete Mathematics*, vol. 8, no. 4, pp. 652–660, 1995.
- [15] L. Gyorfi and B. Laczay, "Signature coding and information transfer for the multiple access adder channel," in *Information Theory Workshop, 2004. IEEE*. IEEE, 2004, pp. 242–246.
- [16] J. L. Massey, "Collision-Resolution Algorithms and Random-Access Communications," in *Multi-User Communication Systems*, ser. International Centre for Mechanical Sciences, G. Longo, Ed. Springer Vienna, 1981, vol. 265, pp. 73–137.
- [17] A. ParandehGheibi, J. K. Sundararajan, and M. Médard, "Collision Helps Algebraic Collision Recovery for Wireless Erasure Networks," in *Proc. of IEEE WiNC 2010*, Boston, MA, USA, Jun. 2010.
- [18] —, "Acknowledgement Design for Collision-Recovery-Enabled Wireless Erasure Networks," in *Proc. of 48th Annual Allerton Conference*, Monticello, IL, USA, Sep. 2010.
- [19] G. Cocco, C. Ibars, D. Gunduz, and O. del Rio Herrero, "Collision Resolution in Slotted ALOHA with Multi-User Physical-Layer Network Coding," in *Proc. IEEE VTC 2011 (Spring)*, Yokohama, Japan, May 2011.
- [20] G. Cocco, N. Alagha, C. Ibars, and S. Cioni, "Network-coded diversity protocol for collision recovery in slotted ALOHA networks," *Int. J. Satell. Commun. Network*, pp. 225 – 241, 2014.
- [21] J. Goseling, M. Gastpar, and J. H. Weber, "Random Access with Physical-layer Network Coding," in *Proc. of Information Theory and Applications Workshop, 2013*, San Diego, CA, USA, Feb. 2013.
- [22] K. Censor-Hillel, B. Haeupler, N. Lynch, and M. Médard, "Bounded-Contention Coding for Wireless Networks in the High SNR Regime," in *Distributed Computing*. Springer, 2012, pp. 91–105.
- [23] E. Biglieri and L. Gyorfi, Eds., *Multiple Access Channels: Theory and Practice*. IOS press, 2007.
- [24] R. Al-Rumaih and P. Mathys, "Analysis of a Hybrid Random-Access System with Multi-User Coding (Throughput)," in *Proc. of IEEE ISIT 1993*, San Antonio, TX, USA, Jan. 1993.
- [25] —, "On the ALOHA Multiple Access System with Coding," in *Proc. of IEEE ISIT 1994*, Trondheim, Norway, Jun. 1994.
- [26] N. Pippenger, "Bounds on the Performance of Protocols for a Multiple-Access Broadcast Channel," *IEEE Trans. Inf. Theory*, vol. 27, no. 2, pp. 145–151, 1981.
- [27] M. Ruzinko and P. Vanroose, "How an Erdos-Renyi-Type Search Approach Gives an Explicit Code Construction of Rate 1 for Random Access with Multiplicity Feedback," *IEEE Trans. Inf. Theory*, vol. 43, no. 1, pp. 368–373, Jan. 1997.
- [28] J. H. Sørensen, C. Stefanovic, and P. Popovski, "Coded Splitting Tree Protocols," in *Proc. of IEEE ISIT 2013*, Istanbul, Turkey, Jul. 2013.
- [29] G. Peeters and B. Van Houdt, "Interference Cancellation Tree Algorithms with k-Signal Memory Locations," *IEEE Trans. Commun.*, vol. 58, no. 11, pp. 3056–3061, Nov. 2010.
- [30] A. Zanella and M. Zorzi, "Theoretical Analysis of the Capture Probability in Wireless Systems with Multiple Packet Reception Capabilities," *IEEE Trans. Commun.*, vol. 60, no. 4, pp. 1058–1071, Apr. 2012.
- [31] B. Nazer and M. Gastpar, "Reliable Physical Layer Network Coding," *Proceedings of the IEEE*, vol. 99, no. 3, pp. 438–460, Mar. 2011.
- [32] S. C. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: Tutorial, survey, and beyond," *Physical Communication*, vol. 6, p. 442, 2013.
- [33] J. Conway, N. Sloane, E. Bannai, R. Borcherds, J. Leech, S. Norton, A. Odlyzko, R. Parker, L. Queen, and B. Venkov, *Sphere Packings, Lattices and Groups*, ser. Grundlehren der mathematischen Wissenschaften. Springer New York, 2013.
- [34] U. Erez and R. Zamir, "Achieving $1/2 \log(1 + \text{SNR})$ on the AWGN Channel With Lattice Encoding and Decoding," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2293–2314, Oct. 2004.
- [35] D. Danyev, B. Laczay, and M. Ruzinko, "Multiple access adder channel," in *Multiple Access Channels: Theory and Practice*, E. Biglieri and L. Gyorfi, Eds. IOS press, 2007, pp. 26–53.
- [36] B. Lindström, "Determining subsets by unramified experiments," in *A Survey of Statistical Design and Linear Models*, J. N. Srivastava, Ed. North-Holland, 1975.
- [37] R. Bose and S. Chowla, "Theorems in the additive theory of numbers," *Commentarii Mathematici Helvetici*, vol. 37, no. 1, pp. 141–147, 1962. [Online]. Available: <http://dx.doi.org/10.1007/BF02566968>
- [38] P. Mathys and P. Flajolet, "Q-ary Collision Resolution Algorithms in Random-Access Systems with Free or Blocked Channel Access," *IEEE Trans. Inf. Theory*, vol. 31, no. 2, pp. 217–243, Mar. 1985.
- [39] I. Bar-David, E. Plotnik, and R. Rom, "Forward collision resolution-a technique for random multiple-access to the adder channel," *IEEE transactions on information theory*, vol. 39, no. 5, pp. 1671–1675, 1993.
- [40] O. Ordentlich and Y. Polyanskiy, "Low complexity schemes for the random access gaussian channel," in *Information Theory (ISIT), 2017 IEEE International Symposium on*. IEEE, 2017, pp. 2528–2532.
- [41] C. Stefanovic, D. Vukobratovic, J. Goseling, , and P. Popovski, "Identifying Randomly Activated Users via Sign-Compute-Resolve on Graphs," in *Proc. of IEEE ICC 2014, Workshop on Massive Uncoordinated Access Protocols*, Kuala Lumpur, Malaysia, May 2016.

- [42] G. Liva, "Graph-Based Analysis and Optimization of Contention Resolution Diversity Slotted ALOHA," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 477–487, 2011.
- [43] X. Chen and D. Guo, "Many-access channels: The gaussian case with random user activities," in *2014 IEEE International Symposium on Information Theory*. IEEE, 2014, pp. 3127–3131.