Tree-Algorithms With Multi-Packet Reception and Successive Interference Cancellation

Čedomir Stefanović, Senior Member, IEEE, Yash Deshpande, Graduate Student Member, IEEE, H. Murat Gürsu, Member, IEEE, and Wolfgang Kellerer, Senior Member, IEEE

Abstract—In this paper, we study binary tree-algorithms that exploit a combination of multi-packet reception (MPR) and successive interference cancellation (SIC), which so far has not been considered in the literature. Specifically, we assume that the receiver is capable of successfully decoding any collision of up to and including \( K \) concurrent packet transmissions and can perform SIC along the tree. We show a number of novel results for this type of tree algorithms. We first derive the basic performance parameters, which are the expected length of the collision resolution interval and the throughput normalized with \( K \), conditioned on the number of contending users. We then analyze their asymptotic behavior, identifying an oscillatory component that amplifies as \( K \) increases. In the next step, we derive the maximum stable throughput (MST) for the gated and windowed access assuming Poisson arrivals. We show that for windowed access, the bound on MST normalized with \( K \) increases with \( K \). Finally, we discuss practical issues related to implementation of such scheme, as well as compare it to slotted ALOHA-based schemes that exploit both \( K \)-MPR and SIC.

Index Terms—Random-access protocols, tree algorithms, multi-packet reception (MPR), successive interference cancellation (SIC), massive IoT communications.

I. INTRODUCTION

In the last decade, there have been significant theoretical advances in the area of random-access protocols, instigated by the novel use-cases pertaining to the Internet of Things (IoT). A typical IoT scenario involves a massive number of sporadically active users exchanging short messages. Such user activity mandates the use of random-access protocols, however, their application in massive IoT scenarios faces the challenge of an increased requirement for efficient performance. Particularly, as the amount of exchanged data is low, the overhead of the random-access scheme should be minimal in order not to create a bottleneck in the overall communication setup.

Manuscript received 9 August 2022; revised 26 December 2022 and 27 May 2023; accepted 7 July 2023. Date of publication 15 August 2023; date of current version 20 November 2023. The work of Čedomir Stefanović is supported by the European Horizon 2020 project SIxThSense (grant agreement number 883315). The associate editor coordinating the review of this article and approving it for publication was J. Yuan. (Corresponding author: Čedomir Stefanović.)

Čedomir Stefanović is with the Department of Electronic Systems, Aalborg University, 9220 Aalborg, Denmark (e-mail: cs@es.aau.dk).

Yash Deshpande and Wolfgang Kellerer are with the Chair of Communication Networks, Technical University of Munich, 80333 Munich, Germany (e-mail: yash.deshpande@tum.de; wolfgang.kellerer@tum.de).

H. Murat Gürsu is with Nokia Standards, 81541 Munich, Germany (e-mail: murat.gursu@nokia.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCOMM.2022.3305467.

Digital Object Identifier 10.1109/TCOMM.2022.3305467

A way to improve the performance of random-access protocols is to embrace interference from the contending users. Effectively, this is achieved by employing multi-packet reception (MPR), enabled by the advanced capabilities of the physical layer (i.e., via the use of advanced signaling processing). Tree-algorithms [1] and ALOHA [2], [3] are families of random-access protocols that by design suffer from collisions by contending users; as such, it is fitting to assume that protocols from these families will benefit from MPR. Indeed, it was shown that MPR improves the performance of slotted ALOHA, e.g., [4], [5], [6].

Another line of research in the context of slotted ALOHA is the use of successive interference cancellation (SIC) across slots.1 In this class of protocols [7], the users transmit multiple replicas of their packets on purpose. Decoding of a packet replica occurring in a singleton slot (i.e., a slot containing only a single transmission) enables the removal of all the other related replicas, potentially transforming some of the collision slots into singletons from which packets of other contending users can be decoded, and thus propelling new iterations of SIC. The use of SIC pushes the throughput performance significantly, asymptotically reaching the ultimate bound for the collision channel of 1 packet/slot [8], [9]. It was also shown that a combination of MPR and SIC pushes the performance further than any of the two techniques separately [10].

A tree-algorithm based scheme exploiting SIC, named SICTA, was proposed in [11]. It was shown that the maximum stable throughput (MST) of SICTA reaches \( \ln^2 2 \approx 0.693 \) packet/slot, which is significantly better than the best performing variant of the algorithm without SIC [12]. Finally, the use of MPR in tree-algorithms was analyzed in our recent work [13], which showed that MPR pushes the normalized2 MST in the version of the scheme with the windowed access.

Motivated by the insights in [10] and [13], in this paper we study the performance of tree-algorithms with \( K \)-MPR and SIC; that is, we assume that the receiver is capable of successfully decoding any collision of up to and including \( K \) concurrent packet transmissions and can perform SIC along the tree. We show an interesting fact that, for the gated access, the bound on the MST (normalized with \( K \)) decreases with

1Strictly speaking, SIC is just another form of MPR, and many MPR schemes rely on interference cancellation. In this paper, we use the term SIC to denote the application of interference cancellation across slots, while we assume that MPR operates on an individual slot basis.

2Normalized with respect to the assumed linear increase in physical resources to achieve MPR.
Tree Algorithms

Tree algorithms were introduced by Capetanakis [1]. Their key ingredient is the collision-resolution protocol (CRP) that is driven by the feedback sent by the receiver. The basic variant of the CRP, denoted as BTA, operates on a time-slotted multiple-access collision channel with feedback as follows. Assume that $n$ users transmitted their packets in a slot:
- If $n = 0$, the slot is idle, and the corresponding feedback is sent by the receiver.
- If $n = 1$, there is a single packet in the slot (the slot is a singleton), the packet is decoded and the user that transmitted it becomes resolved, which is acknowledged by the receiver.
- If $n > 1$, a collision occurs and the receiver sends the corresponding feedback, initiating the collision resolution. The collided users are split into two groups, e.g., group 0 and group 1; the decisions of which group to join are made uniformly at random (i.e., the users pick one of the two groups with a probability of 0.5) and independently of any other user. In the next slot, the users in group 0 transmit. If the slot is idle (i.e., no user selected group 0) the users from group 1 transmit in the next slot. If the slot is a singleton, the packet in it is decoded and the users from group 1 transmit in the next slot. Finally, if the slot is a collision slot, the users in group 0 split again into two groups, and the procedure is recursively repeated. In this case, the users in group 1 wait until all packets from users in group 0 become decoded (the information of which is obtained via monitoring the feedback).
- The collision resolution ends when all $n$ packets are decoded.

Fig. 1a) shows an example of the scheme. In practice, the CRP is combined with a channel-access protocol (CAP), which specifies when the arriving (i.e., active) users can access the channel. The basic variants of CAP are the gated (also known as blocked), windowed, and free access; the details about the former two are presented in Section V. It was shown that the MST throughput of BTA with the gated access is 0.346 packet/slot [1].

The seminal paper [1] inspired a number of works on tree-algorithms. Here we mention the modified tree-algorithm, which omits the slots that are certain to repeat the immediate previous collision (e.g., slot 10 in Fig. 1a) would be skipped), increasing the MST with the gated access to 0.375 packet/slot [14]. Another significant improvement can be obtained by using the BTA in the windowed access framework, which increases the MST to 0.429 packet/slot [14].

A further modification to the framework is made by considering $d$-ary splitting, in which the colliding users split into $d > 2$ groups. It was shown that the ternary tree-algorithms with biased splitting (biased meaning that the probabilities of choosing a group are not uniform over the groups) are the optimal choice [15]. In this respect, a variant of the MTA scheme with clipped access (a modification of the windowed access) introduced in [12] is the best performing scheme in conventional setups (i.e., without MPR and SIC), achieving the MST of 0.4878 packet/slot.

Supplementing tree-algorithms with $K$-MPR capacity has the potential to improve their performance, as demonstrated in Fig. 1b). Here we mention the most relevant works studying the impact of $K$-MPR capacity on tree-algorithms. Upper and lower bounds on the MST (therein referred to as the capacity) for $K$-MPR tree algorithm were derived in [16]. The work in [17] proposes a $K$-MPR tree algorithms with an adaptive form of windowed access, where a part of the subsequent arrival window is added to the one being currently resolved, depending on the outcomes of the collision resolution. The work in [18] analyzes MPR in a tree-algorithm with continuous arrivals with a small number of users in the system (of the order of 10), proposing a transmission strategy that guarantees stability. Finally, the paper [13] performs the analysis of BTA with $K$-MPR in the standard windowed access setup; notably, as $K$ grows, the MST of this scheme tends to the lower bound on capacity derived in [16].

A modification of the original scheme that employs SIC, denoted as SICTA, was introduced in [11]. In SICTA, the receiver stores collision slots; once a packet becomes decoded in a slot occurring after a split has been performed, the
receiver removes its replica from the previous collision slot(s) using SIC, potentially instigating decoding of new packets and replica removal along the tree. Fig. 1c) shows an example of SICTA; obviously, the use of SIC enables skipping of the slots laying on the lower branches of the tree. The MST of binary SICTA is 0.693, which is a huge improvement over modified tree-algorithm (MTA). However, we note that the results presented in [11] for $d > 2$ do not hold - related insights on the performance analysis of $d$-ary SICTA, where $d > 2$, are presented in a separate work of ours [19].

We also mention the work presented in [20], proposing a hybrid multiple-access scheme in which the user signatures are resolved via a $K$-MPR tree-algorithm (both with and without SIC) and the user data via a polling mechanism. The analysis of the tree-algorithm-based part of the scheme is basic, only providing bounds on the expected length of CRI given the number of colliding users, and the main performance parameter is the net-rate of the user data, which represents a dominant part of users’ transmissions. Another hybrid scheme was proposed in [21], consisting of a phase in which users access the channel for a certain number of consecutive slots with probability 0.5, and if this phase does not end by decoding of all the users, continuing with a SICTA-based phase. The authors also assume a more advanced SIC capability, in order to be able to attempt (a blind) decoding of the users’ transmissions in the first phase by treating them as linear combinations over slots. It was shown that this approach achieves a higher MST than SICTA, becoming arbitrarily close to 1 as the length of the first phase is extended. However, the complexity of the scheme increases with the extension of the first phase, ultimately becoming unbounded. For further details, we refer the reader to [21]. A query-driven tree-algorithm with SIC, tailored for a finite user population of users with an a priori defined addressing scheme and fixed number of user activated per CRI, was recently proposed in [22]. Depending on the population size and the number of active users, the throughput of this scheme can reach values close to 1. However, this scheme can neither be compared to SICTA nor the one examined in this work, as they deal with infinite user populations, Poisson arrivals and do not assume any specific addressing scheme. Other recent works were presented in [23] and [24], exploiting tree algorithms in the context of RFID tag identification and LTE RACH procedure, respectively. The former estimates the number of active tags, performs their optimized grouping, and applies MTA to split collisions, while the latter optimizes the splitting factor $d$ when resolving preamble collisions. In both cases, achievable throughputs are below 0.4878, which is the highest throughput in conventional communication-theoretic setups.

Finally, we mention the variant of tree-algorithms with the free access [15], in which the users are free to access the channel as soon as they experience a packet arrival. The MST performance of the ternary MTA with the free access is 0.392, which is in between the one of the gated and one of the windowed access [15]. The performance of tree-algorithms with free access, with SIC, or with $K$-MPR was investigated in [25]. The optimized variant with SIC achieves MST of 0.57. Regarding the variant with $K$-MPR, we perform a comparison to its MST performance in Section V-B. The approach to the analysis of this class of tree-algorithms is fundamentally different from the one presented in this paper [25].

B. Multi-Packet Reception

Research and design of multiple-access schemes that enable multi-packet reception have a long history; the canonical examples being CDMA, or Zadoff-Chu-preamble-based
random-access used in 3GPP standards from LTE onwards [26]. There are also coding techniques specifically designed for this purpose – we mention the $K$-out-of-$n$ coding for multiple-access channels, see [27, Chapters 2 and 3], [28].

Some general models of MPR capability from the perspective of random-access protocol can be found in, e.g., [29], [30]. In this paper, we adopt the following model: (i) 1) if there are up to and including $K$ packets colliding in a slot, all packets are successfully decoded, and 2) if the number of colliding packets in a slot is greater than $K$, no packet can be successfully decoded. This model can be understood as an extension of the collision channel model, which is the default channel model for the assessment of random-access algorithms. Specifically, it can be referred to as the $K$-collision channel.

The works studying random-access protocols with MPR typically abstain from modelling the investments required at the physical layer to enable the MPR. In this paper, we assume that the $K$-MPR capability requires $K$ times more (time-frequency) resources in comparison to single-packet reception case (i.e., the required number of resources is directly proportional to $K$). In effect, slots in $K$-collision channel are $K$ times larger compared to the standard (1-)collision channel, which is taken into account when assessing the performance, see Section III. This model is adequate for CDMA [31] or some $K$-out-of-$n$ coding schemes [20], [28], [32].

III. SYSTEM MODEL

Consider $n$ active users and a common access point (AP). The users are contending to access the AP over a multiple-access $K$-collision channel with feedback by transmitting fixed-length packets. The time-frequency resources of the channel are divided into slots dimensioned to accommodate a single packet transmission. The users are synchronized on a slot basis via means of the feedback sent by the AP. The feedback channel is broadcast, instantaneous, and perfect.

The contention starts with all $n$ users transmitting in the first slot that appears on the channel and lasts until all users’ packets are successfully decoded. Henceforth, we also denote the event of successful decoding of a user packet as the user resolution. In each slot, the contention outcome can be an idle (no user transmitted in the slot), collision (more than $K$ users transmitted in the slot), or success (up to and including $K$ users transmitted in the slot, and all their transmissions were successfully decoded). In the last case, the success may trigger subsequent decodings via SIC along the tree, as depicted in Fig. 1c). All successfully decoded users do not contend anymore, while the collided users split into two groups. We consider binary splitting, i.e., a user performing a split can join either a generic group 0 or group 1. The choice is performed independently by each user, where group 0 is joined with probability $p$ and group 1 with probability $1-p$. The users in group 0 contend in the next slot. The users in group 1 wait – they can either become resolved through the application of SIC after a successful slot, as depicted in Fig. 1c) in the siblings of slots 4, 3 and 7, or they will perform an immediate split, which takes place after slot 6 in Fig. 1c). The decisions whether to transmit, split, or wait are performed by monitoring the feedback channel and updating the state variables maintained by users. For the sake of brevity, we will not elaborate these aspects here, but note that they can be derived using the well-known principles of tree-algorithms, e.g., see [1], [11], [33].

The interval elapsed from the first slot up to and including the slot in which all $n$ users are resolved is denoted as CRI. The length of CRI in slots conditioned on $n$ is a random variable denoted by $l_n$. The basic performance parameter is the expected value of $l_n$, denoted by $L_n$, i.e., $L_n = E[l_n]$. Another performance parameter is the conditional throughput

$$T_n = \frac{1}{K} \frac{n}{n}.$$  

The throughput is the measure of the efficiency of resource use, where the normalization with $K$ reflects the assumed linear increase in resources required to achieve $K$-MPR.

The introduced CRP is just a building block of a complete random access protocol. Another block is a CAP which regulates how the activated users access the channel, determining the number of users $n$ that enter the CRP. The considered CRP can be combined either with the gated or windowed CAP, the details of which will be presented in Section V.

Finally, we remark that in this paper we assume perfect SIC, which is a common assumption in the related work, see [8], [11], [25]. This assumption is reasonable in some practical scenarios already for moderate signal-to-noise ratios [8].

IV. ANALYSIS

Here we analyze the performance of the introduced CRP, i.e., the expected conditional CRI length and the conditional throughput.

The conditional length of a CRI, given $n$ active users at its start, is

$$l_n = \begin{cases} 1 & 0 \leq n \leq K \\ l_i + l_{n-i} & n > K \end{cases}$$  

(2)

where $i$ and $n-i$ denote the number of users that chose group 0 and group 1, respectively.

The expected conditional length of CRI is thus

$$L_n = \sum_{i=0}^{n} \binom{n}{i} p^i (1-p)^{n-i} (l_i + L_{n-i})$$  

(3)

$^3$Recall that in the figure it is assumed that $K = 1$.

$^4$Treatment of the case with imperfect SIC and decoding errors would require a change of the channel model. In particular, imperfect SIC would leave residual interference, which can be modelled using the approaches like the one in [34]. Each round of SIC would increase the residual interference level, which would accumulate along the tree and progressively lower the probability of the successful decoding of the packets in the affected slots. Further analysis of this problem is out of the paper’s scope.

$^3$There is a benign misuse of notation in (2) for $n > K$, where both the random variables on the left-hand side (lhs) and right-hand side (rhs) are denoted by $l$, although the ones on the rhs correspond to the CRI’s that stem from splitting of the CRI that figures on the lhs.
where \( p \) is the probability of a user joining group 0. By developing (3), \( L_n \) can be calculated recursively through

\[
L_n = \begin{cases} 
1 & 0 \leq n \leq K \\
\frac{p^n + (1-p)^n + 2 \sum_{i=1}^{n-1} (\binom{n}{i}) p^{n-i}(1-p)^i L_i}{1 - p^n - (1-p)^n} & n > K.
\end{cases}
\]  

(4)

A. Direct Expression for \( L_n \)

For the derivation of the direct, i.e. non-recursive expression for \( L_n \), we rely on the method that exploits generating functions [15]. We start by introducing the conditional probability generating function (CPGF) of \( I_n \)

\[
Q_n(z) = E \left\{ z^{I_n} \right\}
\]

(5)

where, due to (2), the following holds

\[
Q_0(z) = Q_1(z) = \cdots = Q_K(z) = z.
\]

(6)

For \( n > K \), we have

\[
Q_n(z) = \sum_{i=0}^{n} \binom{n}{i} p^i (1-p)^{n-i} Q_i(z) Q_{n-i}(z).
\]

(7)

Note that the following holds

\[
L_n = \frac{dQ_n(z)}{dz} \bigg|_{z=1}.
\]

(8)

The (unconditional) probability generating function (PGF) of CRI, assuming that \( n \) obeys a Poisson distribution\(^6\) with a mean \( x \), is given by

\[
Q(x, z) = \sum_{n=0}^{\infty} Q_n(z) \frac{x^n}{n!} e^{-x} = \sum_{n=0}^{\infty} \frac{x^n}{n!} e^{-x} \sum_{i=0}^{n} \binom{n}{i} p^i (1-p)^{n-i} Q_i(z) Q_{n-i}(z) + (z-z^2) \sum_{k=0}^{K} \frac{x^k}{k!} e^{-x}
\]

(9)

(10)

where we exploited (6), (7), and the fact that for \( n \leq K \)

\[
\sum_{i=0}^{n} \binom{n}{i} p^{n-i} (1-p)^i Q_{n-i}(z) Q_i(z) = z^2.
\]

(11)

The first term on rhs in (10) can be transformed into

\[
\sum_{n=0}^{\infty} Q_n(z) \frac{(px)^n}{n!} e^{-px} \sum_{i=0}^{\infty} Q_i(z) \left( \frac{(1-p)x}{i} \right) e^{-i(1-p)x}
\]

(12)

so that (10) becomes

\[
Q(x, z) = Q(px, z) Q((1-p)x, z) + (z-z^2) \sum_{k=0}^{K} \frac{x^k}{k!} e^{-x}.
\]

(13)

\(^6\)This is an auxiliary assumption that will not limit the general nature of the derived results.

We introduce the transformed generating function (TGF) of \( L_n \) as

\[
L(x) = \frac{\partial Q(x, z)}{\partial z} \bigg|_{z=1} = \sum_{n=0}^{\infty} L_n \frac{x^n}{n!} e^{-x}.
\]

(14)

where we exploited (8) and (9) to obtain the last expression in (14).

Taking the partial derivative of (13) with respect to \( z \) at \( z = 1 \) yields

\[
L(x) = L(px) + L((1-p)x) - \sum_{k=0}^{K} \frac{x^k}{k!} e^{-x}
\]

(15)

where we used the fact that \( Q(x, 1) = 1, \forall x \).

In the next step, we assume the following power series representation of \( L(x) \)

\[
L(x) = \sum_{n=0}^{\infty} a_n x^n
\]

(16)

where, using (8), it can be shown that

\[
L_n = \sum_{k=0}^{n} \frac{n!}{(n-k)!} a_k.
\]

(17)

We now compute \( a_k \), \( k = 0, 1, \ldots, n \). From (4), it follows that

\[
a_k = \begin{cases} 
1 & k = 0 \\
0 & k = 1, \ldots, K.
\end{cases}
\]

(18)

Substituting (16) into (15) and using Maclaurin series expansion for \( e^{-x} \) yields

\[
\sum_{n=0}^{\infty} a_n (1-p^n - (1-p)^n) x^n = -\sum_{k=0}^{K} \frac{x^k}{k!} \sum_{n=0}^{\infty} (-1)^n \frac{x^n}{n!} = \sum_{n=0}^{\infty} \min(n, K) \left( \frac{(-1)^n k+1}{k!(n-k)!} \right) x^n.
\]

(19)

For \( n \leq K \), it can be shown that

\[
\sum_{k=0}^{n} \frac{(-1)^n k+1}{k!(n-k)!} = \begin{cases} 
-1 & n = 0 \\
0 & 0 < n \leq K
\end{cases}
\]

(20)

which, coupled with (18), transforms (19) into

\[
\sum_{n=K+1}^{\infty} a_n (1-p^n - (1-p)^n) x^n = \sum_{n=K+1}^{\infty} \sum_{k=0}^{K} \frac{(-1)^n k+1}{k!(n-k)!} x^n.
\]

(21)

Solving (21) for \( a_n \), \( n \geq K \), we get

\[
a_n = \frac{K}{k!(n-k)!} \cdot \frac{1}{1-p^n - (1-p)^n}.
\]

(22)

By substituting (18) and (22) in (17) for \( n > K \), and after some manipulation, we get

\[
L_n = 1 + \sum_{i=K+1}^{n} \binom{n}{i} \frac{(-1)^i+1}{1-p^i - (1-p) \sum_{k=0}^{K} \frac{1}{i} (i-k)^i}.
\]

(23)
Using the identity that holds for $K < j$

$$\sum_{k=0}^{K} (-1)^k \binom{i}{k} = (-1)^j \binom{i-1}{K}$$  \hspace{1cm} (24)$$

(23) simplifies to

$$L_n = 1 + \sum_{i=K+1}^{n} \binom{n}{i} \binom{i-1}{K} \frac{(-1)^{i-K+1}}{1 - p^i - (1-p)^i}$$

$$= 1 + \left( \frac{n}{K} \right) \sum_{i=K+1}^{n} \binom{n-K}{i-K} \binom{i-K}{i} \frac{(-1)^{i-K+1}}{i(1-p^i - (1-p)^i)}.$$  \hspace{1cm} (25)$$

Finally, we get

$$L_n = 1 - \left( \frac{n}{K} \right) \sum_{i=1}^{n-K} \frac{i(-1)^i(n-K)}{(i+K)(1-p^i+K-(1-p)^i)}.$$  \hspace{1cm} (26)$$

It is easy to show that (26) is minimized for $p = \frac{1}{2}$. In other words, fair splitting achieves minimal $L_n$, which is given by

$$L_n = 1 - \frac{n}{K} \sum_{i=1}^{n-K} \frac{i(-1)^i(n-K)}{(i+K)(1-2^{-i-K+1})}.$$  \hspace{1cm} (27)$$

In the rest of the paper, we assume fair splitting.

Fig. 2 shows $K \cdot L_n$, i.e., the expected conditional length of CRI weighted by $K$, to make the comparison fair among the curves obtained for different $K$ (recall that it is assumed that the slot size increases linearly with $K$). We first observe that all the curves have their values equal to $K$ for $n < K$, which is the consequence of the fact that just one slot is required to resolve a collision of no more than $K$ users. As $n$ increases, the curves for different $K$ tend to each other. In addition, a careful inspection reveals that the curves show an oscillatory behaviour around the straight line with the slope $1/\ln(2) \approx 1.443$, which is a fact shown in Section IV-B. In other words, as $n$ grows, the expected conditional length of a CRI can be approximated as

$$L_n \approx \frac{1.443}{K} \cdot n \text{[slot]}.$$  \hspace{1cm} (28)$$

directly corresponding to the average delay in slots of the resolution of an initial collision of $n$ users. Results on the bounds on $L_n$ as $n$ grows are provided in Section IV-C.

Fig. 3 shows the conditional throughput $T_n = 1/K \cdot n/L_n$ as function of $n$. Evidently, $T_n$ first increases as $n/K$ for $n \leq K$, due to the fact that a single slot is needed to resolve a collision of $n \leq K$ users (although not shown in the figure, all curves start at 0). $T_n$ then drops and the oscillatory behaviour becomes evident, with the oscillations periodicity depending on $\log(n)$ and its amplitude increasing with $K$. Further, the oscillations are non-vanishing, a fact identified in [15] for the binary tree-algorithms on the standard collision channel. We investigate this phenomenon in the next subsection.

More importantly, both Fig. 2 and Fig. 3 suggest that the use of MPR does not improve the performance of the tree-algorithm conditioned on $n$, when normalized with $K$. In particular, Fig. 3 shows that, as $n \to \infty$, $T_n$ oscillates around the value of $\ln(2) \approx 0.6931$, irrespective of the value of $K$. In Section V, we make further investigations of this issue.

### B. Asymptotical Behaviour of $L_n$

Here we turn to the analysis of asymptotic behavior of $L_n$, enhancing the approach presented in [15]. Rewriting (15) for the case of fair-splitting (i.e., $p = 1/2$), we get

$$L(x) = 2L_{\frac{x}{2}} - \sum_{k=0}^{\infty} \frac{x^k}{k!} e^{-x}.$$  \hspace{1cm} (29)$$

By differentiating (29) twice, we get

$$L''(x) = \frac{1}{2} L''_{\frac{x}{2}} - \frac{x^{K-1}}{(K-1)!} \frac{x^K}{K!} e^{-x} = g(x)$$  \hspace{1cm} (30)$$

which is a functional equation satisfying the contraction condition with the solution in the form [15]

$$L''(x) = \sum_{m=0}^{\infty} 2^m \frac{1}{2m} g\left(\frac{x}{2^m}\right)$$

$$= \sum_{m=0}^{\infty} 2^m \left[ \frac{(\frac{x}{2^m})^{K-1}}{(K-1)!} - \frac{(\frac{x}{2^m})^K}{K!} \right] e^{-\frac{x}{2^m}}.$$  \hspace{1cm} (31)$$
Integrating (31) twice, and taking into account the initial conditions \( L(0) = 1 \) and \( L'(0) = 0 \) that stem from (14), we obtain the following expression for the TGF

\[
L(x) = 1 + \sum_{m=0}^{\infty} 2^m - \sum_{m=0}^{\infty} 2^m e^{-\frac{\pi}{2m}} \sum_{k=0}^{K} \left( \frac{x}{2m} \right)^k k!
\]

(32)

Exploiting (14) further, the previous equation can be transformed to

\[
\sum_{n=0}^{\infty} L_n \frac{x^n}{n!} = e^x + e^x \sum_{m=0}^{\infty} 2^m \left[ 1 - e^{-\frac{\pi}{2m}} \sum_{k=0}^{K} \left( \frac{x}{2m} \right)^k k! \right].
\]

(33)

Using the Maclaurin series expansion for \( e^x \), and after some manipulation, (33) transforms into

\[
\sum_{n=0}^{\infty} L_n \frac{x^n}{n!} = \sum_{n=0}^{\infty} \frac{x^n}{n!} \left[ 1 + \sum_{m=0}^{\infty} 2^m \left( 1 - \sum_{k=0}^{m} \left( \frac{n}{k} \right) \frac{1}{2^{mk}} \right) \right.
\]

\[
\left. \left( 1 - \frac{1}{2^{m-1}} \right)^{n-k} e^{-\frac{\pi}{2m}} \Theta(n-1) \approx e^{-\frac{\pi}{2m}} \right] \frac{n^k}{k!} \left( 1 - \frac{k(k-1)}{2} \Theta(n-1) \right) \approx \frac{n^k}{k!}
\]

(34)

Equating coefficients for \( x^n, n > K \), we get

\[
L_n = 1 + \sum_{m=0}^{\infty} 2^m \left[ 1 - \sum_{k=0}^{m} \left( \frac{n}{k} \right) \frac{1}{2^{mk}} \right] \frac{n^{k-1}}{k!} \approx \frac{n^{k-1}}{k!}
\]

(35)

In principle, from (35) one can derive the same expression for \( L_n \) given by (27). However, we do not pursue this further. Instead, assuming that \( \gamma K \) is fixed, we exploit the following approximations for \( n \gg K \)

\[
\left( 1 - \frac{1}{2^{m-1}} \right)^{n-k} e^{-\frac{\pi}{2m}} \Theta(n-1) \approx e^{-\frac{\pi}{2m}} \frac{n^k}{k!} \left( 1 - \frac{k(k-1)}{2} \Theta(n-1) \right) \approx \frac{n^k}{k!}
\]

(36)

(37)

which, substituted into (35), yield

\[
L_n \approx 1 + \sum_{m=0}^{\infty} 2^m \left[ 1 - \sum_{k=0}^{m} \left( \frac{n}{k} \right) \frac{1}{2^{mk}} \right] \frac{n^k}{k!} \left( 1 - \frac{1}{2^{m-1}} \right)^n
\]

(38)

Now, the task at hand is to isolate \( n \) in (38), such that summation over \( m \) can be performed. For this purpose, we exploit the method for the asymptotic analysis of harmonic sums [15], [33]. We introduce the following function

\[
g(x) = 1 - \sum_{k=0}^{K} x^k e^{-x}.
\]

The Mellin transform of \( g(x) \) is

\[
G(s) = \int_0^{\infty} g(x)x^{s-1} dx = -\Gamma(s) \left[ 1 + \sum_{k=1}^{K} \prod_{i=1}^{k-1} (s+i) \frac{1}{k!} \right]
\]

(39)

\[
= -(s+1)\Gamma(s) \left[ 1 + \frac{s}{2!} + s \sum_{n=3}^{K} \prod_{i=2}^{K-1} (s+i) \frac{1}{k!} \right]
\]

(40)

where \( s \) is a complex variable laying in the fundamental strip (i.e., strip of convergence) given by \(-2 < \Re(s) < 0\) and \( \Gamma(s) \) is the meromorphic extension of the Gamma function. The inverse Mellin transform for \( x = n/2^m \) is given by

\[
g \left( \frac{n}{2^m} \right) = \frac{1}{2\pi i} \int_{\eta-j\infty}^{\eta+j\infty} G(s) \left( \frac{n}{2^m} \right)^{-s} ds
\]

(41)

where \( \eta \) belongs to the fundamental strip. Substituting (41) into (38), and interchanging the order of summation and integration, we obtain

\[
L_n \approx 1 + \frac{1}{2\pi i} \int_{\eta-j\infty}^{\eta+j\infty} G(s)n^{-s} \sum_{m=0}^{\infty} 2^{(s+1)m} ds
\]

(42)

The domain of absolute convergence of the series in (42) is \( \Re(s) < -1 \). Thus, the fundamental strip of the integrand

\[
H(s) = G(s)n^{-s} \frac{1}{1 - 2^{s+1}}
\]

(43)

is obtained as the intersection of the domain of absolute convergence of the series and the fundamental strip of \( G(s) \), and is given by \(-2 < \Re(s) < -1\). In this strip lies \( \eta \) in (42).

We compute the integral in (42) using the residue theorem. In order to evaluate \( L_n \) for \( n \to \infty \), we set the set up the contour in the complex plane, depicted in Fig. 4, where \( \gamma \) and \( D \) are some fixed values. We close the path of integration in the half of the complex plane that is right to the fundamental

Fig. 4. The contour of the integration in the complex plane.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
strip, allowing $D$ and $\gamma$ to tend to infinity. The gamma function decays exponentially fast as the absolute value of the imaginary component of the argument increases, thus, the integration on the horizontal parts of the contour tends to zero as $|D| \to \infty$. The integral on the vertical line $\Re(s) = \gamma$, $\gamma > 0$, is bounded by $O(n^{-\gamma})$, also tending to zero for large $n$ [35, Chapter 5.2.2], [33]. Thus, the integral in (42) is equal to the negative sum of the residues of the poles of $H(s)$ within the contour (negative due to the contour orientation).

The factor $n^{-s}$ trivially has no poles in the contour. Further, $g(s)$ has a simple pole in 0 which is due to the corresponding pole of $\Gamma(s)$.\footnote{For the sake of completeness, we note that the pole of $\Gamma(s)$ at $-1$ is cancelled out by the zero $(s+1)$ of $G(s)$, see (40).} We have

$$\text{Res}_{s=0} H(s) = -\Gamma(0) = -1. \quad (44)$$

The factor $1/(1 - 2^{s+1})$ has simple poles at $s_p \in \mathcal{P} = \{-1 + 2\pi jm/\ln 2, m \in \mathbb{Z}\}$, and it can be shown that the value of the corresponding residues is $-1/\ln 2$. We first compute the value of the residue at $s_p = -1$

$$\text{Res}_{s=-1} H(s) = \frac{-G(-1)n}{\ln 2} = \frac{n}{K \ln 2} \quad (45)$$

where we used the fact that

$$G(-1) = \left[ 1 - \sum_{k=2}^{K} \frac{(k-2)!}{k!} \right] = -\frac{1}{K}. \quad (46)$$

Further, for $s_p \in \{-1 + 2\pi jm, m \in \mathbb{N}\}$, we have

$$\text{Res}_{s=s_p} H(s) = \frac{-\frac{G(-1)n}{\ln 2}}{1 - 1 + 2\pi jm/\ln 2} e^{2\pi jm \log_2 n} A(K, m) \quad (47)$$

where

$$A(K, m) = 1 + \sum_{k=1}^{K} \prod_{j=0}^{k-1} \left( i - 1 + \frac{2\pi jm}{\ln 2} \right). \quad (48)$$

Similarly, for $s_p \in \{-1 - 2\pi jm, m \in \mathbb{N}\}$ we have

$$\text{Res}_{s=s_p} H(s) = \frac{-\frac{G(-1)n}{\ln 2}}{1 - \frac{2\pi jm}{\ln 2}} e^{-2\pi jm \log_2 n} A(K, -m). \quad (49)$$

Using the mirror-symmetry property that holds for the gamma function

$$\Gamma(s^*) = \Gamma^*(s) \quad (51)$$

where $^*$ denotes the complex conjugate, and the following identity (which can be trivially shown)

$$A(K, -m) = A^*(K, m) \quad (52)$$

we get

$$\sum_{s_p \in \mathcal{P} \setminus \{0\}} \text{Res}_{s=s_p} H(s)$$

$$= -\frac{2n}{\ln 2} \sum_{m=1}^{\infty} \Re \left( B(K, m) e^{2\pi jm \log_2 n} \right)$$

$$= -\frac{2n}{\ln 2} \sum_{m=1}^{\infty} |B(K, m)| \cos (2\pi m \log_2 n + \arg(B(K, m))) \quad (53)$$

where

$$B(K, m) = \Gamma \left( -1 + \frac{2\pi jm}{\ln 2} \right) A(K, m). \quad (54)$$

Again, since the gamma function decays exponentially fast as the imaginary component of the argument increases, (53) can be approximated as

$$\sum_{s_p \in \mathcal{P} \setminus \{0\}} \text{Res}_{s=s_p} H(s)$$

$$\approx -\frac{2n}{\ln 2} |B(K, 1)| \cos (2\pi \log_2 n + \arg(B(K, 1))). \quad (55)$$

Putting all the pieces together, we obtain for the expected conditional length of CRI, when $n \to \infty$, to be

$$L_n \approx \frac{n}{K \ln 2} \times \{1 - 2K|B(K, 1)| \cos(2\pi \log_2 n + \arg(B(K, 1)))\}. \quad (56)$$

The conditional throughput, when $n \to \infty$, is

$$T_n = \frac{n}{KL_n} \approx \frac{\ln 2}{1 - 2K|B(K, 1)| \cos(2\pi \log_2 n + \arg(B(K, 1)))}. \quad (57)$$

This oscillatory component in $\log_2 n$ was identified in, e.g., [15], [33], [36].\footnote{The oscillations are caused by the fact that a binary tree created during a CRP session has a discrete number of levels and that the depth of the tree grows as $\log_2 n$ [36].} In the case treated here, the difference is that its amplitude depends on $K$ and cannot be neglected, as it affects the stability bound (further discussed in Section V).

The expression $2K|B(K, 1)|$ can be easily computed for any $K$. The graph presented in Fig. 5 shows that its value increases with $K$, which is also confirmed in Fig. 2 and Fig. 3.\footnote{Although of a little practical relevance, an interesting problem in its own right is to determine the behaviour of $K|B(K, 1)|$ as $K \to \infty$. This problem is out of the paper scope; based on our preliminary investigation, we conjecture that there is an upper bound on the value of $K|B(K, 1)|$ as $K \to \infty$.} Finally, we validate the analysis by comparing its output with the results shown in Fig. 3. For instance, $2K|B(K, 1)|$ evaluates to 0.0607 for $K = 32$, implying that the asymptotic maximum and minimum values of $T_n$ are 0.7378 and 0.6536, respectively, see (57). Obviously, the curve for $T_n$ when $K = 32$ in Fig. 3 indeed tends to oscillate between these two values as $n$ increases.
C. Simple Bounds on $L_n$ and $T_n$

We conclude this section by developing simple, but useful bounds on $L_n$ and $T_n$, which do not require asymptotic evaluation presented in the previous subsection. In particular, these bounds can be computed for any finite $m$ and are valid for any $n \geq m$.

For $n > K$ and fair splitting, the expected conditional length of CRI reduces to

$$L_n = \frac{\sum_{i=0}^{n-1} \binom{n}{i} L_i}{2^n - 1}. \quad (58)$$

Building up on the method introduced by Massey [14], we want to find the constant $\alpha_m$ for which the following holds

$$L_n \leq \alpha_m n, \ n \geq m. \quad (59)$$

For $n < m$, we can write

$$L_n \leq \alpha_m n + \sum_{i=1}^{M-1} \delta_{i,n}(L_n - \alpha_m n) \quad (60)$$

where $\delta_{i,n}$ is the Kronecker delta, and where (60) holds by definition. In the induction step, we substitute (60) into (58), and after some manipulation, obtain

$$L_n \leq \alpha_m n + \frac{\sum_{i=0}^{m-1} \binom{n}{i}(L_i - \alpha_m i)}{2^n - 1} \quad (61)$$

and the condition (59) will be true for any

$$\alpha_m \geq \frac{\sum_{i=0}^{m-1} \binom{n}{i} L_i}{\sum_{i=0}^{m-1} \binom{n}{i} i} \quad (62)$$

as the summation in the second term on the right-hand side of (61) is non-positive in this case. The tightest upper bound is given by

$$\alpha_m = \sup_{n \geq m} \frac{\sum_{i=0}^{m-1} \binom{n}{i} L_i}{\sum_{i=0}^{m-1} \binom{n}{i} i}. \quad (63)$$

In an analogous fashion, the lower bound is found as

$$L_n \geq \beta_m n, \ n \geq m \quad (64)$$

where

$$\beta_m = \inf_{n \geq m} \frac{\sum_{i=0}^{m-1} \binom{n}{i} L_i}{\sum_{i=0}^{m-1} \binom{n}{i} i}. \quad (65)$$

Note that the bounds in (63) and (65) can be made arbitrarily tight by increasing $m$ and $n$.

The corresponding bounds on conditional throughput are

$$B_m = \frac{1}{K \beta_m} \geq T_n \geq \frac{1}{K \alpha_m} = A_m, \ n \geq m. \quad (66)$$

In Table I, we list $\alpha_m, \beta_m, A_m$ and $B_m$ (rounded up to four decimal places). Note the agreement between the bounds on $T_n$ shown in the table, i.e., $A_m$ and $B_m$, and the results plotted in Fig. 3.

V. PERFORMANCE UNDER POISSON ARRIVALS

In this section, we provide insights into the performance of a random access protocol that combines the CRP protocol introduced in Section III with the gated CAP and the windowed CAP. We adopt the standard evaluation approach by assuming Poisson arrivals in an infinite user population; the arrival intensity per slot is denoted by $\lambda$. We are interested to identify the bounds on $\lambda$ for which the random access protocol features a stable operation. In brief, the stability implies that the individual packets are successfully received with a finite delay almost surely [15].

A. Gated Access

The gated (also denoted as blocked) CAP is an obvious approach to deal with traffic arrivals. In particular, all users that arrive during a CRI have to wait until that CRI ends, i.e., they are blocked. Once the current CRI ends, all blocked users transmit in the next available slot, thus initiating the next CRI. Fig. 6 illustrates the principles of gated access.

The stability conditions of the gated access were investigated in a number of works, e.g., in [14] and [15]. The sufficient condition for stability is

$$\lambda < \lambda_S \quad (67)$$
and the sufficient condition for instability is
\[ \lambda > \lambda_U \]  
(68)
where the values of the bounds \( \lambda_S \) and \( \lambda_U \). Using (56), we get
\[ \limsup_{n \to \infty} \frac{L_n}{n} = \frac{1 + 2K[B(K, 1)]}{K \ln(2)} = L_S \]  
(69)
\[ \liminf_{n \to \infty} \frac{L_n}{n} = \frac{1 - 2K[B(K, 1)]}{K \ln(2)} = L_U \]  
(70)
from which it follows that [15]
\[ \lambda_S = L_S^{-1} = \frac{K \ln(2)}{1 + 2K[B(K, 1)]} \]  
(71)
\[ \lambda_U = L_U^{-1} = \frac{K \ln(2)}{1 - 2K[B(K, 1)]} \]  
(72)
Table II lists values of \( \lambda_S/K \) and \( \lambda_U/K \) for several values of \( K \).\(^{10}\) Obviously, as \( K \) increases, difference among \( \lambda_S/K \) and \( \lambda_U/K \) grows. This is expected, since the amplitude of the oscillations in (56) grows with \( K \).

**B. Windowed Access**

Another way to deal with the traffic arrivals is to use windowed CAP (also denoted as the epoch mechanism). In this approach, the time axis related to the traffic arrivals is divided into equal-length windows and every window is associated to a separate CRI. Specifically, the users arriving in \( i \)-th window transmit in the first slot after the CRI of the users arriving in \((i-1)\)-th window ends, thus starting their own CRI. Fig. 7 illustrates the windowed access.

Denoting the window length in slots by \( \Delta \) (which does not have to be an integer), the probability of \( n \) arrivals \( (n \in \mathbb{N}) \) in the window can be calculated as
\[ \Pr\{N = n\} = \frac{\lambda \Delta}{n!} e^{-\lambda \Delta} \]  
(73)
i.e., \( n \) is a Poisson random variable (r.v.) with mean \( \lambda \Delta \). The expected length of CRI is
\[ L(\lambda \Delta) = E\{L_n|\lambda \Delta\} = \sum_{n=0}^{\infty} L_n \frac{(\lambda \Delta)^n}{n!} e^{-\lambda \Delta}. \]  
(74)
The necessary condition for stability is the following
\[ L(\lambda \Delta) < \Delta \]  
(75)
\(^{11}\)Note that \( \lambda_S/K = \liminf_{n \to \infty} T_n \) and \( \lambda_U/K = \limsup_{n \to \infty} T_n \), where \( T_n \) is given by (57).

\[ \begin{array}{c|c|c}
K & \lambda_S/K & \lambda_U/K \\
\hline
1 & 0.6931 & 0.6931 \\
2 & 0.6931 & 0.6932 \\
4 & 0.6930 & 0.6932 \\
8 & 0.6916 & 0.6947 \\
16 & 0.6811 & 0.7056 \\
32 & 0.6536 & 0.7378 \\
64 & 0.6216 & 0.7833 \\
\end{array} \]  
**TABLE II**  
**STABILITY BOUNDS ON NORMALIZED TRAFFIC ARRIVAL INTENSITY FOR GATED ACCESS**

\[ \begin{array}{c|c|c|c|c|c}
K & \lambda_S/K & \lambda_U/K & \Delta_S & \lambda_S/K \text{ (too SIC) [13]} & \lambda_{PB}/K \text{ [25]} & G/K \text{ [37]} \\
\hline
1 & 0.6931 & 0.6931 & 17.4161 & 0.423 & 0.36 & / \\
2 & 0.6932 & 0.6932 & 10.2308 & 0.470 & 0.381 & 0.68 \\
4 & 0.6932 & 0.6932 & 6.8175 & 0.515 & 0.39 & 0.74 \\
8 & 0.6947 & 0.6947 & 2.6802 & 0.507 & 0.41 & / \\
16 & 0.7056 & 0.7056 & 1.2527 & 0.623 & 0.45 & / \\
32 & 0.737 & 0.737 & 1.1296 & 0.6862 & / & / \\
64 & 0.7816 & 0.7816 & 1.0753 & 0.747 & / & / \\
\end{array} \]  
**TABLE III**  
**STABILITY BOUNDS ON TRAFFIC ARRIVAL INTENSITY FOR WINDOWED ACCESS**  
Note that for stability to hold, the condition \( E\{L_n^2\} < \infty \) has also to be satisfied. This can be shown for \( L(\lambda \Delta) < \Delta \), however, we omit the proof.
as indicated by (75). The table lists the maximum normalized arrival intensity $\lambda^*/K$ for which the $K$-MPR BTA with windowed access (and without) has a stable operation [13]. The comparison between $\lambda_5/K$ and $\lambda_6/K$ reveals that, as $K$ increases, most of the gain comes from the MPR, while the contribution of SIC becomes limited. We also compare the performance of our scheme with the performance of other state-of-the-art schemes that use the same reception model. In particular, the column $\lambda_{FA}/K$ lists the available results on MST performance of algorithms with $K$-MPR and free access [25]. The comparison reveals that this scheme provides little gain in terms of the normalized MST as $K$ increases. Finally, the last column reproduces the available results on the performance of Irregular Repetition Slotted ALOHA (IRSA), a frame slotted ALOHA based algorithm that exploits SIC, with $K$-MPR capability. The results correspond to the ratio $G/K$, where $G$ is the load of the scheme (i.e., the ratio of the number of contending users and the slots in the frame) for which the probability of not decoding a user transmission is below $10^{-3}$ [12] for the frame length of 100 slots and some optimized degree distributions. In this case, $G/K$ can be used as a proxy for the throughput and the comparison reveals that the performance of IRSA with $K$-MPR is comparable with the one of the proposed algorithm. Further comparison between slotted ALOHA and algorithms with MPR and is performed in Section VI-B.

In Fig. 8, we plot $F(\lambda\Delta) = \frac{\lambda\Delta}{\ln(1 + \lambda\Delta)}$, see (79), as a function of $\lambda\Delta$, i.e., the sensitivity of the stability bound on the normalized arrival intensity per slot as a function of the arrival intensity within a window. The figure shows the characteristic oscillatory behaviour, which becomes more pronounced as $K$ increases. Nevertheless, the oscillations’ periodicity is rather large, implying that there is a certain tolerance on the potential estimation errors of $\lambda$ and/or dimensioning errors of window length $\Delta$. We also plot the sensitivity of the stability bound for the analogous protocol without, investigated in [13]. Obviously, as $K$ increases, the bound has a clearly pronounced maximum, after which the performance quickly deteriorates. It can be concluded that in this respect, the protocol with is an advantageous solution.

VI. DISCUSSION AND CONCLUSION

A. Asymptotic Performance of BTA With $K$-MPR

The method for the derivation of the asymptotic values of the expected conditional CRI length and the throughput presented in Section IV-B can be extended to the case of BTA with fair splitting and $K$-MPR (without) [13]. Here we give the final expressions without a formal proof. Specifically, for $n \geq K$, the expected conditional CRI length is

$$L^*_n = 1 + 2 \sum_{m=0}^{\infty} 2^m \left[ 1 - \sum_{k=0}^{K} \binom{n}{k} \left( 1 - \frac{1}{2^m} \right)^{n-k} \right]$$

which is asymptotically

$$L^*_n \approx 1 + 2 \sum_{m=0}^{\infty} 2^m \left[ 1 - \sum_{k=0}^{K} \binom{n}{2^m k} \frac{e^{-2^m}}{k!} \right].$$

Using the Mellin-transform based asymptotic analysis, we get

$$L^*_n \approx \frac{2n}{K \ln 2} \times (1 - 2K|B(K, 1)|) \cos(2\pi \log_2 n + \arg(B(K, 1))).$$

The conditional throughput is then simply

$$T^*_n \approx \frac{2}{2(1 - 2K|B(K, 1)|) \cos(2\pi \log_2 n + \arg(B(K, 1)))}.$$

The last expression confirms the result identified in [13], that the conditional throughput of BTA with $K$-MPR oscillates around the value of $\ln 2/2 \approx 0.347$ as $n \to \infty$.

B. Comparison With Slotted ALOHA-Based Schemes That Exploit $K$-MPR and

We now turn to the comparison between the considered scheme with analogous schemes from the slotted ALOHA family that exploit MPR and SIC. We mention IRSA [8], a frame slotted ALOHA protocol in which active users transmit several replicas of their packets in the frame. Asymptotically, IRSA supports load thresholds $G^*$ (defined as the ratio of the number of users and slots in the frame) close to 1 with the user resolution probability tending to 1 when the number of replicas transmitted per user is drawn according to a predefined, optimized distribution. Effectively, this performance parameter is equivalent to the throughput. In [38] it was shown that, when IRSA is coupled with MPR, the normalized load threshold $G^*/K$ for a fixed maximum number of replicas per user, decreases with $K$ when $n \to \infty$. A similar insight, in terms of the upper bound on $G^*/K$ was shown in [37]. On the other hand, the work in [10] showed that for the generalized variant of IRSA, denoted as Coded Slotted ALOHA (CSA), the converse bound on $G^*/K$ increases with $K$, quickly becoming very close to 1, and that, asymptotically,
spatially-coupled CSA operates close to the bound, which is out of the reach of the scheme considered in this paper. This evidence may lead to a conclusion that IRSA-based schemes represent a better choice, especially considering that they employ feedback infrequently, only after the end of the contention frame.

For a finite number of contending users, the normalized ratio of the number of contending users and the number of slots in the frame $G/K$ for which the probability of successful user resolution is close to 1 in IRSA is around 0.7 [37] (see Table III for some reference values), which is comparable to the MST of the scheme considered in this paper. The frame length of IRSA is then given by $n/(Kg) \approx 1.4 n/K$, which represents the contention resolution delay of $n$ users. For the proposed scheme with windowed access, the expected delay of resolving a user is upper bounded by the window length, see (75), which decreases with $K$, as shown in Table III.

It should be noted that slotted ALOHA-based protocols, in general, require some form of stabilization. In contrast, protocols are stable for loads up to the MST. Their drawback in comparison to IRSA-based schemes is that they require feedback after every uplink slot, so if one is willing to invest system resources in the feedback, algorithms with MPR and SIC may represent a suitable random-access solution.

C. Practical Considerations

Here we comment on the practical issues incurred by the considered system model, which are (i) the frequent, instantaneous and perfect feedback, (ii) challenges related to the implementation of K-MPR, and (iii) additional complexity that the use of entails.

Regarding feedback, note that mobile cellular systems in general feature an uplink random access channel accompanied by a downlink broadcast feedback channel, making them a candidate technology for the implementation of algorithms. These channels can be configured such that an uplink contention slot is followed by a downlink feedback slot, with the delay that is of the order frame-length, e.g., 10 ms the case of LTE. In cases that the feedback delay is intolerable in relation to the time-duration of slots, one could use the interleaving approach [14] in which independent instances of the scheme are run in consecutive slots appearing on the uplink channel before the corresponding feedback slots in the downlink start arriving. In case that the feedback channel can not be considered perfect, it is reasonable to assume that some form of error-detection code will be employed, enabling the users to detect that the feedback message is garbled and preventing decoding of a wrong feedback message. One of the ways to counteract such situations would be to modify the scheme such that (i) undecoded users receiving a garbled feedback transmit in the next slot, irrespective of whether they contended or not in the previous slot for which the feedback was received, and (ii) equip the transmitted packets with the information about the seed of the random number generator of the respective users (governing the choice of which groups to join during a session of the CRP) and about the slots in which the user transmitted immediately after the garbled feedback was received. This way, the common receiver would be able, once it decodes a packet, to deduce all the slots in which the packet replicas occurred and to remove them by. It can be shown that this approach will not jeopardize the CRP by creating deadlocks, and the only side effect would be that the contention process would become prolonged (hence, decreasing the throughput). Further investigations of this issue are out of the paper’s scope.

We now turn to practical challenges related to the implementation of K-MPR capability. There is a number of coding schemes that are a suitable choice to achieve K-MPR and to align with the use of, e.g., K-out-of-n coding for multiple-access adder channel [32], multiple-access Euclidean channel [39], Gaussian multiple-access channel [20], and multiple-access channel with fading [40]. We note that phyl layer considerations should also be carefully considered when $K$ becomes large, as collision detection, channel estimation, and interference cancellation may become particularly challenging. Further, in wireless random-access settings with time-varying channels, it is natural to assume that the contending users select in an uncoordinated manner their training sequences for channel estimation, which may lead to collisions among simultaneously transmitting users in the sequence domain. These may prevent the AP to decode such collided transmissions even if the total number of simultaneously transmitting users is not more than $K$. However, training sequences can be designed to feature (i) low cross-correlation and (ii) low auto-correlation when the timing offsets (caused by the different propagation delays among users’ transmissions) are larger than the duration of a single symbol of the sequence [41]. The latter property enables decoding of the transmissions of such colliding users, such that the number of sequences employed does not need to scale with the total number of users in order to avoid catastrophic consequences of collisions in the sequence domain. Moreover, the proposed scheme can be modified such that the users involved in undecodable collisions in sequence domain are instructed to choose new sequences and split again and/or the window size (in the windowed access) is optimized to control the collision probability. A consequence would be a decrease in the MST; further analysis is out of the paper scope.

SIC increases the memory requirements (spatial complexity) and processing complexity of the receiver in comparison to the schemes without, e.g., [13]. Firstly, it requires memory to buffer slots containing more than $K$ packets and the number of such slots scales with the CRI length. Assuming the scheme with the windowed-access, the CRI lengths are of the order of the window lengths, which are rather short (see $\Delta s$-column in Table III). In IRSA-based schemes the number of buffered slots is of the order of frame length, which scales with the total number of contending users $n$. Secondly, to perform the receiver needs to re-encode and re-modulate the baseband signal from the successfully decoded packets. To do so, the receiver needs to obtain information about the amplitude, delay of the transmissions from the start of the slot, frequency drift and phase offset of the user’s oscillator with respect to the nominal frequency. The first three parameters can be assumed to be constant for all the transmissions of the user and their estimates can be obtained from the decoded transmission [8].
However, the phase offset may change on transmission basis [8], [41]. In such cases, the receiver should estimate the phase offset of the replica directly from the slot in which the interference cancellation is to be performed. For this purpose, the receiver can use training sequences [41] and the estimate is obtained via correlation. This operation should be repeated for every packet replica. In BTA, the total number of replicas a user transmits scales logarithmically with the number of users starting the CR, which for the windowed access is given by the product $\lambda S \Delta_S$ in Table III. In IRSA-based schemes, the expected number of replicas $DK$ depends on the used distribution and on $K$ [37]. Finally, the receiver needs to rotate (for the estimated phase offsets) and subtract the replicas (i.e., the regenerated samples) from the affected slots. Table IV summarizes this discussion, where $S_{\text{packet}}$ and $S_{\text{seq}}$ are the number of samples contained in a packet and training sequence, respectively, and $C_{E}$ is the complexity of re-encoding and re-modulation.

### TABLE IV

| Scheme                  | memory requirements | added processing complexity of SIC |
|-------------------------|---------------------|------------------------------------|
| BTA with MPR            | $O(S_{\text{packet}})$ | $O((\Delta_S S_{\text{packet}})(\log(\Delta_S S_{\text{packet}}))/S_{\text{seq}})$ |
| BTA with MPR and SIC    | $O(S_{\text{packet}})$ | $O((\Delta_S S_{\text{packet}})(\log(\Delta_S S_{\text{packet}}))/S_{\text{seq}})$ |
| IRSA with MPR           | $O(n S_{\text{packet}})$ | $O(n(C_{E} + DK(S_{\text{seq}} + S_{\text{packet}})))$ |

D. Further Work

Finally, we comment on an approach through which the performance of the scheme could be pushed further. Specifically, as shown in [42], one of the factors limiting the performance of algorithms with SIC is a too high fraction of singleton slots in comparison to IRSA-like protocols, which are unavoidable due to the very nature of the collision resolution process. A way to address this drawback is to form a set of partially-split s pertaining to the same initial collision and perform SIC over the whole set. It remains to be seen how the addition of MPR would affect the performance of such scheme.

**REFERENCES**

[1] J. Capetanakis, “Tree algorithms for packet broadcast channels,” *IEEE Trans. Inf. Theory*, vol. IT-25, no. 5, pp. 505–515, Sep. 1979.

[2] N. Abramson, “The ALOHA system—Another alternative for computer communications,” in *Proc. Fall Joint Comput. Conf.*, vol. 37, Dallas, TX, USA: AFIPS Press, 1970, pp. 281–285.

[3] L. G. Roberts, “ALOHA packet system with and without slots and capture,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, pp. 28–42, Apr. 1975.

[4] S. Ghez, S. Verdu, and S. C. Schwartz, “Stability properties of slotted Aloha with multipacket reception capability,” *IEEE Trans. Automat. Control*, vol. AC-33, no. 7, pp. 640–649, Jul. 1988.

[5] 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.

[6] J. Goseling, C. Stefanovic, and P. Popovski, “A pseudo-Bayesian approach to sign-compute-resolve slotted Aloha,” in *Proc. IEEE Int. Conf. Comput. Workshop (ICCW)*, London, U.K., Jun. 2015, pp. 1–10.

[7] E. Paolini, C. Stefanovic, G. Liva, and P. Popovski, “Coded random access: How coding theory helps to build random access protocols,” *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 144–150, Jun. 2015.

[8] G. Liva, “Graph-based analysis and optimization of contention resolution diversity slotted ALOHA,” *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 477–487, Feb. 2011.

[9] E. Paolini, G. Liva, and M. Chiani, “Coded slotted ALOHA: A graph-based method for uncoordinated multiple access,” *IEEE Trans. Inf. Theory*, vol. 61, no. 12, pp. 6815–6832, Dec. 2015.

[10] C. Stefanovic, E. Paolini, and G. Liva, “Asymptotic performance of coded slotted Aloha with multipacket reception,” *IEEE Commun. Lett.*, vol. 22, no. 1, pp. 105–108, Jan. 2018.

[11] Y. Yu and B. G. 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.

[12] S. Verdu, “Computation of the efficiency of the Mosely-Humblet contention resolution algorithm: A simple method,” *Proc. IEEE*, vol. 74, no. 4, pp. 613–614, Apr. 1986.

[13] C. Stefanovic, H. M. Gürsu, Y. Deshpande, and W. Kellerer, “Analysis of tree-algorithms with multi-packet reception,” in *Proc. IEEE GLOBECOM*, Taipei, Taiwan, Dec. 2020, pp. 1–6.

[14] J. L. Massey, “Collision-resolution algorithms and random-access communications,” in *Multi-User Communications Systems*. Cham, Switzerland: Springer, 1981, pp. 73–137.

[15] P. Mathys and P. Fiajole, “Q-ary collision resolution algorithms in random-access systems with free or blocked channel access,” *IEEE Trans. Inf. Theory*, vol. IT-31, no. 2, pp. 217–243, Mar. 1985.

[16] B. S. Tsybakov, V. A. Mikhailov, and N. B. Likhanov, “Bounds for packet transmission rate in a random-multiple-access system,” *Problemy Peredachi Informatsii*, vol. 19, no. 1, pp. 61–81, 1983.

[17] N. B. Likhanov, I. Plotnik, Y. Shavitt, M. Sidi, and B. S. Tsybakov, “Random access algorithms with multiple reception capability and reply feedback channel,” *Problemy Peredachi Informatsii*, vol. 29, no. 1, pp. 82–91, 1993.

[18] R.-H. Gau, “Tree/stack splitting with remainder for distributed wireless medium access control with multipacket reception,” *IEEE Trans. Wireless Commun.*, vol. 10, no. 11, pp. 3909–3923, Nov. 2011.

[19] Y. Deshpande, C. Stefanovic, H. M. Gürsu, and W. Kellerer, “Corrections to ‘high-throughput random access using successive interference cancellation in a tree algorithm,’” *IEEE Trans. Inf. Theory*, vol. 69, no. 3, pp. 1658–1659, Mar. 2023.

[20] J. Goseling, C. Stefanovic, and P. Popovski, “Sign-compute-resolve for tree splitting random access,” *IEEE Trans. Inf. Theory*, vol. 64, no. 7, pp. 5261–5276, Jul. 2018.

[21] G. T. Peeters and B. Van Houdt, “On the capacity of a random access channel with successive interference cancellation,” in *Proc. IEEE Int. Conf. Comput. Workshop (ICCW)*, Jun. 2015, pp. 2051–2056.

[22] H. M. Gürsu, F. Guan, and W. Kellerer, “Hard latency-constraints for high-throughput random access: SICQTA,” in *Proc. IEEE Int. Conf. Comput. (ICC)*, Shanghai, China, May 2019, pp. 1–7.

[23] J. Su, Z. Sheng, A. X. Liu, Y. Han, and Y. Chen, “A group-based binary splitting algorithm for UHF RFID anti-collision systems,” *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 998–1012, Feb. 2020.

[24] H. Althumali, M. Othman, N. K. Noordin, and Z. M. Hanapi, “Dynamic tree-splitting random access for M2M communications in IoT networks,” *IEEE Syst. J.*, vol. 16, no. 2, pp. 3179–3190, Jun. 2022.

[25] G. T. Peeters and B. Van Houdt, “On the maximum stable throughput of tree algorithms with free access,” *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5087–5099, Nov. 2009.

[26] *Medium Access Control (MAC) Protocol Specification (Release 16)*, document TS36.321, v16.3.0, 3GPP, Dec. 2020.

[27] E. Biglieri and L. Györfi, *Multiple Access Channels*. Amsterdam, The Netherlands: IOS Press, 2007.

[28] O. Ordentlich and Y. Polyanskiy, “Low complexity schemes for the random-access Gaussian channel,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 2528–2532.

[29] S. Ghez, S. Verdu, and S. C. Schwartz, “Stability properties of slotted ALOHA with multipacket reception capability,” *IEEE Trans. Automat. Control*, vol. AC-33, no. 7, pp. 640–649, Jul. 1988.

[30] L. Tong, Q. Zhao, and G. Mergen, “Multipacket reception in random access wireless networks: From signal processing to optimal medium access control,” *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 108–112, Nov. 2001.

---

This can be shown using the approach from [36].
specializes in the IoT, 5G/beyond 5G, and mission-critical communications.

interests include communication theory and wireless communications and he has coauthored more than 130 peer-reviewed publications. His research interests include random access algorithms and industrial wireless communication.

---

A. Mengali, R. De Gaudenzi, and P.-D. Arapoglou, “Enhancing the physical layer of contention resolution diversity slotted ALOHA,” *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4295–4308, Oct. 2017.

D. Danev, B. Laczay, and M. Ruszinko, “Multiple access adder channel,” in *Multiple Access Channels*, E. Biglieri and L. Gyorfi, Eds. Amsterdam, The Netherlands: IOS Press, 2007, pp. 26–53.

P. Flajolet, X. Gourdon, and P. Dumas, “Mellin transforms and asymptotics: Harmonic sums,” *Theor. Comput. Sci.*, vol. 144, nos. 1–2, pp. 3–58, Jan. 1995.

Y. Gao, B. Xia, Y. Liu, Y. Yao, K. Xiao, and G. Lu, “Analysis of the dynamic ordered decoding for uplink NOMA systems with imperfect CSI,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6647–6651, Jul. 2018.

D. E. Knuth, *The Art of Computer Programming*, vol. 3, 2nd ed. Reading, MA, USA: Addison-Wesley, 1998.

A. J. E. M. Janssen and M. J. de Jong, “Analysis of contention tree algorithms,” *IEEE Trans. Inf. Theory*, vol. 46, no. 6, pp. 2163–2172, Sep. 2000.

I. Hnedouh, C. Adjih, P. Mühlethaler, and V. Kumar, “On the performance of irregular repetition slotted ALOHA with multiple packet reception,” in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, Jun. 2020, pp. 557–564.

M. Ghanbarinejad and C. Schlegel, “Irregular repetition slotted ALOHA with multiuser detection,” in *Proc. 10th Annu. Conf. Wireless Demand Netw. Syst. Services (WONS)*, Banff, AB, Canada, Mar. 2013, pp. 201–205.

D. Danev and M. R. Z. Füredi, “Multiple access Euclidean channel,” in *Multiple Access Channels*, E. Biglieri and L. Gyorfi, Eds. Amsterdam, The Netherlands: IOS Press, 2007, pp. 54–72.

G. Cocco, S. Pfetschinger, and M. Navarro, “Seek and decode: Random access with physical-layer network coding and multiuser detection,” *Trans. Emerg. Telecommun. Technol.*, vol. 28, no. 5, pp. e3129, May 2017.

E. Casini, R. De Gaudenzi, and O. R. Herrero, “Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access scheme for satellite access packet networks,” *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1408–1419, Apr. 2007.

J. H. Sørensen, C. Stefanovic, and P. Popovski, “Coded splitting tree protocols,” in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2013, pp. 2860–2864.