Analysis of Tree-Algorithms with Multi-Packet Reception

Čedomir Stefanović*, H. Murat Gürsu†, Yash Deshpande†, Wolfgang Kellerer†
*Department of Electronic Systems, Aalborg University, Denmark
†Chair of Communication Networks, Technical University of Munich, Germany
Email: cs@es.aau.dk, {murat.guersu,yash.deshpande,wolfgang.kellerer}@tum.de

***Abstract*—In this paper, we analyze binary-tree algorithms in a setup in which the receiver can perform multi-packet reception (MPR) of up to and including K packets simultaneously. The analysis addresses both traffic-independent performance as well as performance under Poisson arrivals. For the former case, we show that the throughput, when normalized with respect to the assumed linear increase in resources required to achieve K-MPR capability, tends to the same value that holds for the single-reception setup. However, when coupled with Poisson arrivals in the windowed access scheme, the normalized throughput *increases* with K, and we present evidence that it asymptotically tends to 1. We also provide performance results for the modified tree algorithm with K-MPR in the clipped access scheme. To the best of our knowledge, this is the first paper that provides an analytical treatment and a number of fundamental insights in the performance of tree-algorithms with MPR.**

I. Introduction

The promise of Internet of Things (IoT) has profoundly influenced research and development of cellular access systems in the past decade. The recent introduction of massive IoT (mIoT) and ultra-reliable service categories in 5G standardization is likely the best example of the foreseen significance of IoT communications. An outstanding feature of IoT communications is the need to efficiently deal with the randomness in users' activations in the access network. In mIoT terms, with potentially thousands of devices in a cell [1], the vital requirement from the system perspective is the throughput, i.e., the efficiency of the use of system resources. From users' perspective, the vital requirement is a low outage probability, which is directly related to a bounded access delay. A combination of these two requirements can be made in the form of stable throughput, which can be defined as the throughput for which the access delay is bounded.

The proverbial examples of random access (RA) algorithms are ALOHA [2] and tree algorithms [3]. The former serve as the basis for access networking in numerous cellular technologies, due to their simplicity. However, the latter achieve significantly higher stable throughputs [4]. Still, the throughput performance in both cases is modest: the maximum stable throughput of slotted ALOHA is $0.3679 \, \frac{\text{packet}}{\text{slot}}$ [2] and for tree-algorithms $0.4878 \, \frac{\text{packet}}{\text{slot}}$ [5], which motivated development of advanced RA schemes. A way to improve the performance is to employ multi-packet reception (MPR) that can be achieved through use of CDMA, multi-access coding techniques, MIMO, etc., as it more effectively deals with the mutual interference of concurrently active users. MPR pushes the performance of ALOHA-based schemes significantly, e.g., see [6]–[8]. However, to the best of our knowledge, use of MPR in tree-algorithms has not been genuinely investigated and the related work on the topic is scarce. The work in [9] analyzes MPR in a tree-algorithm with continuous arrivals with a small number of users in the system, proposing a transmission strategy that guarantees stability. These aspects limit reaching general conclusions. The work in [10] proposes a hybrid scheme combining user identity decoding via the K-MPR tree-algorithm and user data decoding via polling. The performance figures in [10] are derived for the scheme as a whole, while the analysis of the tree-algorithm based component is basic and not corresponding to the usual tree-algorithm setup.

In this paper we establish a communication-theoretic analysis of the use of MPR in tree-algorithms and provide some fundamental insights. For this purpose, we investigate binary tree-algorithms on a K-collision channel (as introduced in Section III), which is a simple, but useful abstraction of the K-MPR capability. The main contributions are:

- We investigate the traffic-independent performance of the binary tree-algorithm (BTA) [3] and derive both recursive and non-recursive expressions for the expected length of collision resolution interval (CRI), conditioned on the number of users n. We compute the throughput of the scheme conditioned on n, providing evidence that its value, when normalized with K, tends to stabilize around the value characteristic for the single-packet reception case.
- We analyse the scheme under Poisson arrivals in the windowed access setup and derive bounds on the stable throughput. We show that the performance improves with K, implying that MPR does pay-off in windowed access.
- Finally, we discuss extensions of the analysis, by showing results for the modified binary-tree algorithm (MTA) [4] and its combination with the clipped access [11] that is known for its superior performance. We also discuss applications to other types of tree algorithms [12].

The paper is organized as follows. Section II introduces the background and Section III the system model. Section IV analyzes the traffic-independent performance of BTA on K-collision channel. Section V investigates the performance under Poisson arrivals in windowed access. Section VI elaborates extensions of the analysis. Section VII concludes the paper.

II. Background

A. Tree Algorithms

We are interested in the classical random-access problem in which a set of of randomly arrived users, whose identities and the number n are a-priori unknown, contend for the access to a common access point (AP). We focus on tree algorithms, whose core concept is the collision resolution driven by the feedback from the AP. The basic form of the algorithm, BTA, on the standard collision channel (as defined in Section II-B), operates as follows. The channel resources are grouped in slots, and all n users transmit their packets in the first slot. If the slot is singleton (i.e., $n = 1$), the packet in it is decoded. In case of a collision (i.e., $n > 1$), the collision resolution starts. The users split into two groups, e.g., group 0 and group 1; every user decides which group to join randomly and independently of any other user. All users in group 0 transmit in the next slot. 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 singleton, the packet in it becomes decoded and the users from group 1 transmit in the next slot. Finally, if the slot contains a collision, the users in group 0 split in two new groups and the procedure is recursively repeated until all packets from users in group 0 become decoded, after which the users in group 1 transmit in the next slot. The procedure is then repeated for the users in group 1. The scheme ends when all n packets are decoded. Should a user transmit in a slot, perform a split, or wait, is decided by monitoring the feedback sent from the AP, see Section III for details. Fig. 1-a) shows an example of the scheme. When the scheme is used in a gated access setup, its stable throughput reaches $0.346 \frac{\text{packet}}{\text{slot}}$ [3].[1]

The work in [3] was followed by a number of improvements and generalizations; we outline only the ones relevant to this work. In the MTA, if the slot dedicated to group 0 in a generic split happens to be idle, the users from group 1 split immediately, thus potentially avoiding repetition of the same collision. In the example in Fig. 1-a), there would be an immediate split after slot 9 in the MTA. This simple modification increases the maximum stable throughput in gated access to $0.375 \frac{\text{packet}}{\text{slot}}$ [4]. Another improvement is to use tree-algorithms in the windowed access setup (see Section V) that pushes the maximum stable throughput, e.g., to $0.4294 \frac{\text{packet}}{\text{slot}}$ in case of BTA [4]. The performance can be further pushed by using MTA with biased splitting in the clipped access setup (which is a modification of the windowed access), achieving the stable throughput of $0.4877 \frac{\text{packet}}{\text{slot}}$ [11]. Finally, the best performing tree-algorithm so far is a variant of the previous scheme, elaborated in [5], achieving the stable throughput of $0.4878 \frac{\text{packet}}{\text{slot}}$.

B. K-Collision Channel

The performance figures stated in Section II-A were derived for the collision channel, the default channel model for the assessment of RA algorithms. Here, we introduce its generalization in the form of K-collision channel, as follows:

(i) If there are no packets in the slot, the slot is idle.
(ii) If there are up to and including K packets colliding in a slot, all packets are successfully decoded.
(iii) If the number of colliding packets in a slot is above K, the slot is a collision slot and no packet can be decoded.

This model reduces to the standard collision channel for $K = 1$. It is a simple model of MPR capability, adequate for K-out-of-n coding schemes [10], [13], [14], or spread-spectrum systems [15]. A feature common to all these works is that the amount of (time-frequency) resources needed to achieve K-MPR linearly increases with K. That is, slots in K-collision channel are K times larger compared to the standard collision channel. This assumption holds in this paper. Fig. 1-b) shows an example of BTA on 2-collision channel.

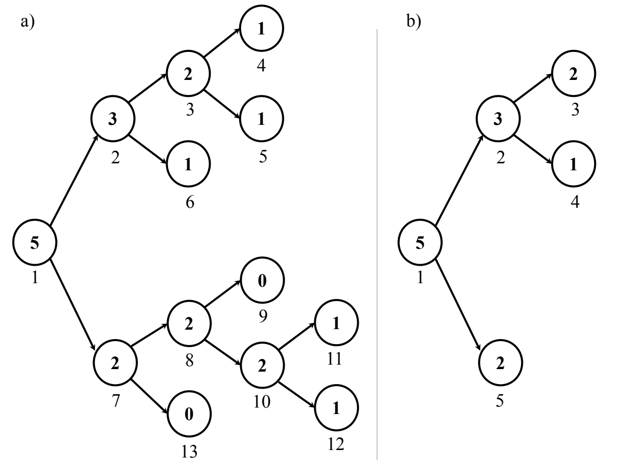

Fig. 1: Binary-tree algorithm (BTA): A node represents a slot, the number inside the node shows the number of users (i.e., packets) in the slot, and the number beneath shows the sequence number of the slot. There are $n = 5$ users. a) BTA on standard, 1-collision channel. b) BTA on 2-collision channel.

III. System Model

We assume a batch of n users, contending for the access to a common AP over a K-collision channel.[2] The channel resources are divided in slots, and users are slot-synchronized. The users contend by transmitting packets that perfectly fit in slots. There is also a perfect feedback channel from the AP.

All n users transmit in the first slot that appears on the channel. The interval elapsed from the first slot up to and including the slot when all n users become resolved is denoted as CRI. The length of a CRI conditioned on n is a random variable (r.v.) denoted by l_n.

We now proceed by elaboration of the. BTA. We denote the slots in a CRI by their numbers, starting with 1. The number of users transmitting in slot j is denoted by n_j. After every

[1] See Section V for details on gated access and stable throughput.

[2] In this section and in Section IV, our focus is on a single batch of n arrived users. Section V deals with the incorporation of an arrival process.

TABLE I: Values of feedback and states of the counters of the users contending in the example in Fig. 1-b).

slot no.	state of counter					feedback
	user 1	user 2	user 3	user 4	user 5	
1	0	0	0	0	0	e
2	0	1	0	0	1	e
3	0	2	1	0	2	1
4	-1	1	0	-1	1	1
5	/	0	-1	/	0	1
end	/	-1	/	/	-1	/

slot j, the AP broadcasts the following feedback

$$f_j = \begin{cases} 0, & \text{if } n_j = 0 \\ 1, & \text{if } n_j \leq K \\ \text{e}, & \text{if } n_j > K. \end{cases} \quad (1)$$

Each user i maintains a counter, whose state in slot j is denoted by $C_{i,j}$, where $C_{i,1} = 0$, $\forall i$, i.e., the initial value is 0. The state of the counter determines the user's transmission decisions: (i) if $C_{i,j} = 0$, user i transmits in slot j, (ii) if $C_{i,j} > 0$, user i abstains from transmitting in slot j, and (iii) if $C_{i,j} < 0$, user i has become resolved and does not contend further. The state of the counter is updated through feedback and splitting procedure, as follows:

$$C_{i,j+1} = \begin{cases} b_{i,j}, & \text{if } f_j = e \text{ and } C_{i,j} = 0 \\ C_{i,j} + 1, & \text{if } f_j = e \text{ and } C_{i,j} > 0 \\ C_{i,j} - 1, & \text{if } f_j \in \{0, 1\} \end{cases} \quad (2)$$

where $b_{i,j}$ is a Bernoulli r.v. with $\Pr\{b_{i,j} = 0\} = p$ and $\Pr\{b_{i,j} = 1\} = 1-p$ (if $p = \frac{1}{2}$, the splitting is fair; otherwise, the splitting is biased). The topmost case in (2) pertains to a split after the collision in slot j, where $b_{i,j}$ determines whether user i will join the generic group 0 or group 1. To facilitate easier understanding, Table I lists the values of the feedback and the states of the users' counters for the example in Fig. 1-b), assuming that: in slot 2, user 1, 3, and 4 chose group 0, while user 2 and 5 chose group 1; and in slot 3, user 1 and 4 chose group 0 and user 3 chose group 1.

Performance Parameters

The primary parameter of interest is the expected CRI length conditioned on n, denoted by $L_n = \mathrm{E}[l_n]$. We are also interested in the conditional throughput, defined as

$$\mathrm{T}_n = \frac{1}{K}\frac{n}{L_n}. \quad (3)$$

The throughput in (3) is the measure of the efficiency of resource use, where the normalization with K reflects the increase in the amount of resources contained in a slot required to achieve K-MPR.

IV. Analysis of Traffic-Independent Performance

The length of a CRI conditioned on n is

$$l_n = \begin{cases} 1, & n \leq K \\ 1 + l_i + l_{n-i}, & n > K \end{cases} \quad (4)$$

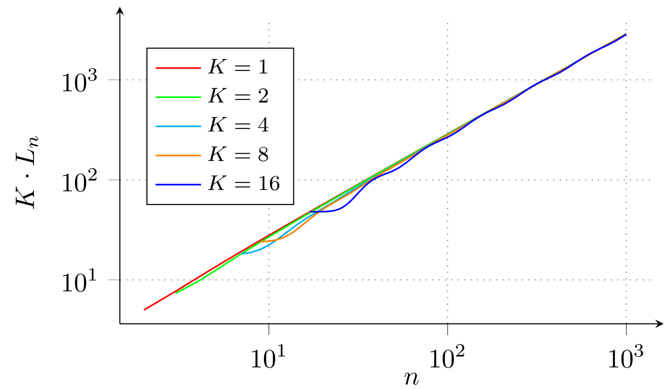

Fig. 2: Conditional length of CRI L_n of BTA on K-collision channel for varying number of users n and K.

where i is the number of users that joined group 0 and $n-i$ is the number of users that joined group 1.

The expected conditional length of CRI for $n > K$ is simply

$$L_n = 1 + \sum_{i=0}^{n} \binom{n}{i} p^i (1-p)^{n-i} (L_i + L_{n-i}). \quad (5)$$

By developing (5), L_n can be calculated recursively through

$$L_n = \begin{cases} 1, & n \leq K \\ \frac{1+\sum_{i=0}^{K} g(n,i,p)}{1-g(n,0,p)} & n = K+1 \\ \frac{1+\sum_{i=0}^{K} g(n,i,p)+\sum_{i=K+1}^{n-1} g(n,i,p)L_i}{1-g(n,0,p)} & n > K+1 \end{cases} \quad (6)$$

where $g(n,i,p) = \binom{n}{i}\left(p^i(1-p)^{n-i} + p^{n-i}(1-p)^i\right)$.

The next theorem gives the non-recursive expression for L_n.

Theorem 1.

$$L_n = 1 - \binom{n}{K} \sum_{j=1}^{n-K} \frac{2\,j\,(-1)^j \binom{n-K}{j}}{(j+K)(1-p^{j+K}-(1-p)^{j+K})}. \quad (7)$$

Proof: See appendix. ∎

The value of (7) is minimized for $p = \frac{1}{2}$, as the value of the denominator on right-hand side (rhs) becomes maximized. In other words, fair splitting achieves minimal L_n, given by

$$L_n = 1 - \binom{n}{K} \sum_{j=1}^{n-K} \frac{2\,j\,(-1)^j \binom{n-K}{j}}{(j+K)(1-2^{-j-K+1})}. \quad (8)$$

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

Fig. 2 shows $K \cdot L_n$ as function of n, for $K \in \{1, 2, 4, 8, 16\}$; the multiplication with K makes the comparison fair. Obviously, as n grows, the difference among curves vanish, and a linear trend of increase in $K \cdot L_n$ emerges (a fact exploited in Section V when developing bounds for L_n).

Fig. 3 plots the values of the conditional throughput T_n corresponding to L_n in Fig. 2. The maxima $\mathrm{T}_n = 1$ are reached for $n = K$, when the CRI lasts a single slot. As n increases, T_n shows a damped oscillatory behaviour, where the oscillations' amplitude increases with K. Also, T_n tends to stabilize around 0.346, irrespective of the value

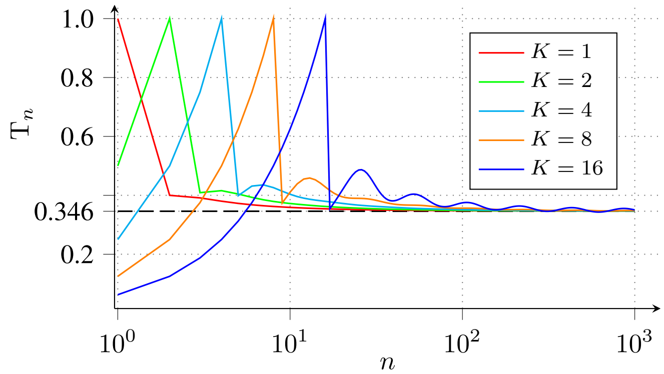

Fig. 3: Conditional throughput T_n of BTA on K-collision channel for varying number of arrived users n and K.

of K; this fact was already identified for $K = 1$ [4]. The conclusion is that, as n grows, increase in K-MPR capability does not increase the efficiency of the use of the system resources. A potential way to formally prove the conclusion is to extend the asymptotic analysis from [12]. However, the related investigation is beyond the scope of the paper and is part of our ongoing work.

V. Incorporation of Traffic Arrivals

So far, we analyzed properties of BTA on a K-collision channel assuming that the number of users n is given. An obvious way to incorporate traffic arrivals, where n is a r.v., is via gated access. In this case, all users that arrive during a CRI wait until that CRI ends, and then transmit in the next available slot, initiating the next CRI. An alternative is to use windowed access, where the time-axis of arrivals is divided into equal-length windows and every window is "served" in a separate CRI, as follows: all users arriving in the i-th window transmit in the first slot after the CRI of the users arriving in the $(i-1)$-th window ends, thus starting their own CRI.

BTA with windowed access significantly outperforms the gated access version in terms of maximum stable throughput for $K = 1$, as outlined in Section II-A. For Poisson arrivals, which are of our interest, the maximum stable throughput is defined as the maximum arrival rate per slot λ with a finite expected delay of user resolution. A fairly tight upper bound on maximum stable throughput for BTA with gated access and $K = 1$ is $\lambda < 0.346 \frac{\text{packet}}{\text{slot}}$ [4], which bears direct relation to the asymptotic behaviour of T_n, see Fig. 3. The figure also suggests that the maximum stable throughput will not increase with K, as the asymptotic behaviour of T_n does not change with K. Thus, hereafter we focus on performance evaluation of BTA on the K-collision channel with windowed access.

Assume a generic arrival window and denote its length by Δ slots (not necessarily an integer). The probability of n arrivals ($n \in \mathbb{N}$) in the window is $\Pr\{N = n\} = \frac{\lambda\Delta}{n!} e^{-\lambda\Delta}$, i.e., n is a Poisson r.v. with mean $\lambda\Delta$. The expected length of CRI is

$$L(\lambda\Delta) = \mathrm{E}\{L_n|\lambda\Delta\} = \sum_{n=0}^{\infty} L_n \frac{(\lambda\Delta)^n}{n!} e^{-\lambda\Delta} \quad (9)$$

TABLE II: BTA on K-collision channel with windowed access: Bounds on throughput and optimal window sizes.

K	α_m	β_m	λ_{U}/K	λ_{S}/K	$\lambda_{\mathrm{S}}\Delta_{\mathrm{S}}$	Δ_{S}
1	2.88538	2.8854	0.42951	0.42951	1.149	2.675
2	1.44267	1.44272	0.47068	0.47068	1.831	1.945
4	0.72158	0.72116	0.51751	0.51751	3.2	1.546
8	0.35907	0.36214	0.56779	0.56779	5.967	1.314
16	0.17355	0.1859	0.62388	0.62388	11.753	1.177

and the goal is to find $\lambda\Delta$ for which

$$L(\lambda\Delta) < \Delta. \quad (10)$$

This condition is necessary for stability, ensuring that arrivals in the window will be served on average in an interval shorter than the window.[3] One can compute (9) with arbitrary precision via (7) or (8) for any λ and Δ, and establish the bound in (10). Hereafter, we proceed with a convenient and insightful bounding method from [4]. It can be shown that

$$\alpha_m n - 1 \le L_n \le \beta_m n - 1 \quad (11)$$

which holds for any $m \ge 1$ and $n > m$, where $\alpha_m = \inf_{n>m} \frac{\sum_{i=0}^{m-1}\binom{n}{i}(L_i+1)}{\sum_{i=0}^{m-1}\binom{n}{i} i}$ and $\beta_m = \sup_{n>m} \frac{\sum_{i=0}^{m-1}\binom{n}{i}(L_i+1)}{\sum_{i=0}^{m-1}\binom{n}{i} i}$. By substituting (11) into (9), it can be shown that

$$f(\alpha_m, m, \lambda\Delta) \le L(\lambda\Delta) \le f(\beta_m, m, \lambda\Delta) \quad (12)$$

where

$$f(x, k, z) = x \cdot z - 1 + \sum_{i=0}^{k} (L_i - x \cdot i + 1) \frac{z^i}{i!} e^{-z}. \quad (13)$$

The scheme will be stable if

$$f(\beta_m, m, \lambda\Delta) < \Delta \quad (14)$$

which yields the bound

$$\lambda < \sup_{\lambda\Delta > 0} \frac{\lambda\Delta}{f(\beta_m, m, \lambda\Delta)} = \lambda_{\mathrm{S}}. \quad (15)$$

Similarly, the bound for which the scheme will be unstable is

$$\lambda > \sup_{\lambda\Delta > 0} \frac{\lambda\Delta}{f(\alpha_m, m, \lambda\Delta)} = \lambda_{\mathrm{U}}. \quad (16)$$

Table II lists values of α_m, β_m, $\frac{\lambda_{\mathrm{U}}}{K}$, $\frac{\lambda_{\mathrm{S}}}{K}$, $\lambda_{\mathrm{S}}\Delta_{\mathrm{s}}$ and Δ_{S} (the value of Δ that corresponds to λ_{S} in (15)), obtained for $m = 50$. Obviously, the bounds $\frac{\lambda_{\mathrm{S}}}{K}$ and $\frac{\lambda_{\mathrm{U}}}{K}$ very tight, their difference can be observed only for decimal places not shown in the table, where $\frac{\lambda_{\mathrm{S}}}{K} < \frac{\lambda_{\mathrm{U}}}{K}$. Notably, λ_{S}/K *increases* with K, implying that use of MPR can indeed pay off. Another interesting observation is that the values of $\lambda_{\mathrm{S}}\Delta_{\mathrm{s}}$ are less than K for $K > 1$. An intuitive explanation for this can be found in Fig. 3. The maximum value of T_n is 1 for any K, attained for $n = K$, and, for $K > 1$, the values of T_n are higher for n that approaches K from the left, than for $n > K$. Thus, the

[3]For stability to hold, the condition $\mathrm{E}\{L_n^2\} < \infty$ has also to be satisfied. This is indeed the case if $L(\lambda\Delta) < \Delta$, but we omit the proof.

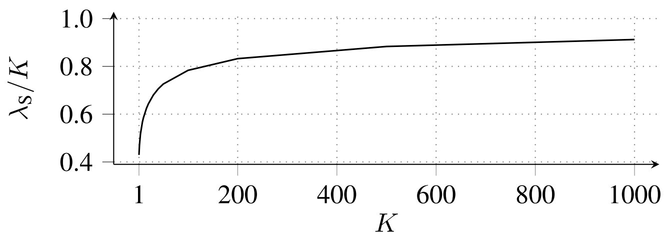

Fig. 4: Maximum stable throughput λ_S/K for increasing K.

optimal[4] mean of the Poisson distribution of the number of arrivals in a window is less than K. Also, it can be observed that the optimal window size in slots Δ_s decreases with K.

Fig. 4 plots λ_S/K as function of K. Obviously, the bound on maximum stable throughput for this scheme steadily increases with K. We conjecture that $\lim_{K\to\infty} \lambda_S/K = 1$, leaving the proof for our future work.

VI. Extensions to Other Variants of Tree-Algorithms

A straightforward extension of the analysis can be made for the case of MTA on a K-collision channel, omitted here due to space constraints. Instead, we depict T_n of the scheme with fair splitting in Fig. 5, where the y-axis is zoomed to emphasize the most important details. Obviously, as n grows, T_n drops as K increases, and tends to the value characteristic for BTA. This is a consequence of the fact that a collision slot involves at least K packets, and with increasing K, the probability of observing an idle slot immediately after a fair split of such slot decreases, counteracting the premise from which MTA draws its gain with respect to BTA.

We proceed by evaluating the performance of MTA with fair splitting and clipped access[5] on a K-collision channel. The impact of K-MPR can be taken into account by adjusting the expected number of resolved users per CRI in the analysis presented in [11]; we omit the details due to space constraints. The evolution of λ_S/K for this scheme with K is given in Table III. Obviously, when compared with λ_S/K in Table II, the performance gap between BTA with windowed access and MTA with clipped access diminishes as K grows, due to the reasons outlined in the previous paragraph.

The analysis can be also extended to d-ary tree algorithms, where we expect that use of K-MPR will bring some new insights. Specifically, for $K = 1$, the ternary MTA with biased splitting features the superior performance [12]. However, the analogous conclusion for $K > 1$ is not obvious, as increase in K increases the chances that packets contained in a slot become decoded, an effect also achieved with d-ary splitting. The corresponding investigation is part of our ongoing work.

VII. Conclusions

In this paper we analyzed binary tree algorithms with MPR, which is seen as a promising communication technique for massive access scenarios in 5G. Our results show that adoption of MPR in BTA with windowed access has a beneficial effect on the stable throughput performance. Interestingly, as MPR-capability increases, the performance gap between MTA with clipped access and BTA with windowed access closes, implying that the added complexity of the former scheme does not pay-off.

[4] Optimal in the sense that it maximizes the stable throughput.
[5] See [11] for the details of this scheme.

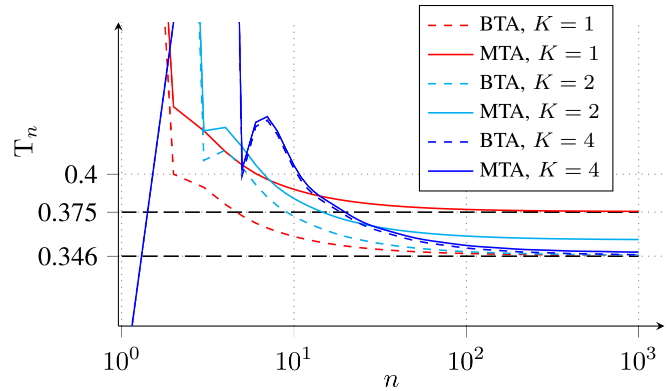

Fig. 5: Conditional throughput T_n of MTA on K-collision channel for varying number of arrived users n and K.

TABLE III: Maximum stable throughput for MTA with clipped access on K-collision channel.

K	1	2	4	8	16
λ_S/K	0.48703	0.4923	0.52257	0.56844	0.62388

Appendix

Proof of Theorem 1: We prove the theorem using the method elaborated in [12]. We first compute the conditional probability generating function (PGF) of l_n, given by

$$Q_n(z) = \mathrm{E}\left\{z^{l_n}\right\}, \text{ where } Q_n(z) = z \text{ for } 0 \le n \le K. \quad (17)$$

It is straightforward to verify that, for $n > K$,

$$Q_n(z) = z\sum_{i=0}^{n}\binom{n}{i}p^i(1-p)^{n-i}Q_i(z)Q_{n-i}(z). \quad (18)$$

We proceed by computing (unconditional) PGF of CRI, assuming that n is a Poisson r.v. with mean x. The PGF is

$$Q(x,z) = \sum_{n=0}^{\infty} Q_n(z)\frac{x^n}{n!}e^{-x} \quad (19)$$

$$= z\sum_{n=0}^{\infty}\frac{x^n}{n!}e^{-x}\sum_{i=0}^{n}\binom{n}{i}p^i(1-p)^{n-i}Q_i(z)Q_{n-i}(z) +$$

$$+ \left(z - z^3\right)\sum_{k=0}^{K}\frac{x^k}{k!}e^{-x} \quad (20)$$

where we exploited (17) and (18), and used the fact that

$$\sum_{i=0}^{n}\binom{n}{i}p^i(1-p)^{n-i}Q_i(z)Q_{n-i}(z) = z^2,\ n \le K. \quad (21)$$

With some manipulation, the first term on rhs in (20) becomes

$$z\sum_{i=0}^{\infty}Q_i(z)\frac{(px)^i}{i!}e^{-px}\sum_{n=0}^{\infty}Q_n(z)\frac{((1-p)x)^n}{n!}e^{-(1-p)x}. \quad (22)$$

Thus, (20) can be re-written as

$$Q(x,z)=zQ(px,z)Q\left((1-p)x,z\right)+\left(z-z^3\right)\sum_{k=0}^{K}\frac{x^k}{k!}e^{-x}. \quad (23)$$

To calculate the first moment of the PGF, we derive (23) with respect to (w.r.t.) z and substitute $z=1$. This yields the mean of the (unconditional) CRI length, denoted by $L(x)$

$$L(x)=1+L(px)+L((1-p)x)-2\sum_{k=0}^{K}\frac{x^k}{k!}e^{-x} \quad (24)$$

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

Similarly, deriving (19) w.r.t. z and substituting $z=1$ yields

$$L(x)=\sum_{n=0}^{\infty}L_n\frac{x^n}{n!}e^{-x}\overset{\text{(a)}}{=}\sum_{n=0}^{\infty}a_nx^n \quad (25)$$

where (a) stems from an assumed power series representation of $L(x)$. By multiplying (25) with e^x, and then applying the power series expansion of e^x, it can be shown that

$$L_n=\sum_{j=0}^{n}\frac{n!}{(n-j)!}a_j. \quad (26)$$

We now find a_j, $j=0,1,...,n$. From (6), it follows that

$$a_j=\begin{cases}1, & j=0\\ 0, & j=1,\ldots,K.\end{cases} \quad (27)$$

Further, substituting (25) into (24) and using the power series expansion of e^{-x} yields

$$\begin{aligned}\sum_{n=0}^{\infty}a_n(1-p^n-(1-p)^n)x^n &= 1-2\sum_{k=0}^{K}\frac{x^k}{k!}\sum_{n=0}^{\infty}(-1)^n\frac{x^n}{n!}\\ &= 1+2\sum_{n=0}^{\infty}\sum_{k=0}^{\min(n,K)}\frac{(-1)^{n-k+1}}{k!(n-k)!}x^n. \quad (28)\end{aligned}$$

For $n\leq K$, it can be shown that

$$\sum_{k=0}^{n}\frac{(-1)^{n-k+1}}{k!(n-k)!}=\begin{cases}-1, & n=0\\ 0, & 0<n\leq K\end{cases} \quad (29)$$

which, coupled with (27), transforms (28) into

$$\begin{aligned}\sum_{n=K+1}^{\infty}a_n(1-p^n-(1-p)^n)x^n &=\\ &= 2\sum_{n=K+1}^{\infty}\sum_{k=0}^{K}\frac{(-1)^{n-k+1}}{k!(n-k)!}x^n. \quad (30)\end{aligned}$$

Solving (30) for a_n, $n\geq K$, we get

$$a_n=\sum_{k=0}^{K}\frac{(-1)^{n-k+1}}{k!(n-k)!}\cdot\frac{2}{1-p^n-(1-p)^n}. \quad (31)$$

By substituting (27) and (31) in (26) for $n>K$, and after some manipulation, we get

$$L_n=1+\sum_{j=K+1}^{n}\binom{n}{j}\frac{2(-1)^{j+1}}{1-p^j-(1-p)^j}\sum_{k=0}^{K}\binom{j}{k}(-1)^{-k}. \quad (32)$$

Using the identity that holds for $K<j$

$$\sum_{k=0}^{K}(-1)^k\binom{j}{k}=(-1)^K\binom{j-1}{K} \quad (33)$$

we further simplify (32) to

$$L_n=1+\sum_{j=K+1}^{n}\binom{n}{j}\binom{j-1}{K}\frac{2\,(-1)^{j-K+1}}{1-p^j-(1-p)^j}. \quad (34)$$

Finally, we get

$$L_n=1-\binom{n}{K}\sum_{j=1}^{n-K}\frac{2\,j\,(-1)^j\binom{n-K}{j}}{(j+K)(1-p^{j+K}-(1-p)^{j+K})} \quad (35)$$

which concludes the proof. ■

References

[1] E2E-aware Optimizations and advancements for Network Edge of 5G New Radio (ONE5G), "Deliverable D2.1: Scenarios, KPIs, use cases and baseline system evaluation," Tech. Rep., nov 2017. [Online]. Available: https://one5g.eu/

[2] N. Abramson, "The ALOHA system – Another alternative for computer communications," in *Proc. of 1970 Fall Joint Computer Conf.*, vol. 37. AFIPS Press, 1970, pp. 281–285.

[3] J. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE transactions on information theory*, vol. 25, no. 5, pp. 505–515, 1979.

[4] J. L. Massey, "Collision-resolution algorithms and random-access communications," in *Multi-user communication systems*. Springer, 1981, pp. 73–137.

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

[6] V. Naware, G. Mergen, and L. Tong, "Stability and delay of finite-user slotted ALOHA with multipacket reception," *IEEE Trans. Info. Theory*, vol. 51, no. 7, pp. 2636–2656, Jul. 2005.

[7] J. Goseling, C. Stefanovic, and P. Popovski, "A pseudo-Bayesian approach to sign-compute-resolve slotted ALOHA," in *IEEE International Conference on Communications (ICC) 2015, MASSAP Workshop*, London, UK, Jun. 2015.

[8] A. Baiocchi and F. Ricciato, "Analysis of pure and slotted ALOHA with multi-packet reception and variable packet size," *IEEE Commun. Lett.*, vol. 22, no. 7, pp. 1482–1485, Jul. 2018.

[9] R.-H. Gau, "Tree/stack splitting with remainder for distributed wireless medium access control with multipacket reception," *IEEE transactions on wireless communications*, vol. 10, no. 11, pp. 3909–3923, 2011.

[10] 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.

[11] R. Rom and M. Sidi, *Multiple Access Protocols*. Springer, 1990, ch. 5.

[12] 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.

[13] D. Danyev, B. Laczay, and M. Ruszinko, "Multiple access adder channel," in *Multiple Access Channels*, E. Biglieri and L. Gyorfi, Eds. IOS press, 2007, pp. 26–53.

[14] O. Ordentlich and Y. Polyanskiy, "Low complexity schemes for the random access Gaussian channel," in *Proc. 2017 IEEE Int. Symp. Inf. Theory*, Aachen, Germany, Jun. 2017, pp. 2533–2537.

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