

Low-Complexity Tree-Based Iterative Decoding for Coded SCMA

Ana Luiza Scharf, Bartolomeu F. Uchôa-Filho, Bruno Fontana da Silva and Didier Le Ruyet

Abstract—Sparse Code Multiple Access (SCMA) is a powerful multiple access technique for future generations of wireless communication where users are allowed to transmit through predefined channel resources with a controlled degree of collision. The base-station then recovers all the users' data through some iterative method. The well-known Message-Passing Algorithm (MPA) has excellent performance but has exponential decoding complexity. Alternative decoding algorithms, such as MPA in the log-domain (Log-MPA), have been proposed in the literature aiming to reduce the decoding complexity while not significantly decreasing performance. In recent work, the authors proposed a modification in the conventional Log-MPA by exploring a tree structure associated with the decoding equations. By properly avoiding symbols with low reliability, a pruned tree is obtained, yielding an arbitrary trade-off between performance and complexity in the joint detection. In the present work, we extend this contribution by showing that the advantages of the tree-based decoding algorithm are magnified when SCMA is coupled to an error-correcting code, in particular, a Low-Density-Parity-Check (LDPC) code. Through computer simulations, we show that an improved performance-decoding complexity trade-off is obtained.

Index Terms—SCMA, Log-MPA, LDPC codes, 5G system.

I. INTRODUCTION

Sparse Code Multiple Access is a non-orthogonal multiple access technique proposed for future-generation wireless networks, which provides high efficiency and good performance [1]. In SCMA, the users' data are mapped into codewords, that are allocated in a non-orthogonal way in resources such as sub-carriers of an Orthogonal Frequency Division Multiplexing (OFDM) [2].

SCMA is a generalization of Low-Density Spreading (LDS) [3], which, in turn, is a sparse version of Code Division Multiple Access (CDMA) with low density spreading sequences [4]. In the LDS technique, the information bits of each user is mapped into a complex symbol of a signal constellation, and this symbol is repeated in a small number of resources or slots (subcarriers, in the case of OFDM), while the other resources are not used (the power allocated to these resources is null). The choice of resources to be used by each user is made so that each resource is used by a small number

of users. The low number of collisions per resource allows for a representation of LDS by a sparse graph so that the Message-Passing Algorithm (MPA) [3] can be used to recover the symbols transmitted from all users

In the SCMA technique, each user's input bits are mapped into complex codewords belonging to a fixed codebook [4]. As in LDS, in SCMA the complex symbols (coordinates) of the codeword are spread in different resources, the only difference being that, in SCMA, the symbols are not mere repetitions. Thanks to this feature, SCMA provides better performance [1]. Also, unlike CDMA, LDS and SCMA meet the massive connectivity demand of future wireless communication systems, that is, they allow a massive number of users, due to inherent design for overloaded systems.

One of the challenges of SCMA is to establish a good relationship between performance and detection complexity [5]. In general, the better the performance, the more complex the decoder becomes. There are several decoders algorithms that try to balance this relationship. The Maximum Likelihood (ML) detector has excellent performance, however, its complexity is prohibitively high. A widely used decoder is the MPA [3], which offers lower detection complexity with little performance loss compared to ML. But MPA is still quite complex (exponential).

In the conventional MPA [3], the decoding equations involve exponential functions, resulting in large storage and complexity [6]. The problem is reduced with the use of the Log-MPA algorithm [7], whose calculations are realized in the logarithmic domain, in a way similar to turbo codes [8].

The purpose of this article is to extend the study of our previous work [9], whose objective was to present a possible solution to reduce the detection complexity of Log-MPA. The idea was to introduce a tree structure associated with the decoding equations of Log-MPA, and then pruning some terms (branches in the tree) considered of little relevance. This results in a detection complexity reduction, with an adjustable complexity-performance trade-off through a single parameter, r , which indicates the number of terms (branches) left unpruned.

In [9], it was claimed that a performance close to that of the original Log-MPA could be obtained with a lower r if error-correcting codes were used in the SCMA system. In this work, we consider coded SCMA where a Low-Density Parity-Check (LDPC) code is used in each user unit. The iteration process between the LDPC decoder and the tree-based Log-MPA is presented. Through computer simulations, we show that, indeed, as claimed in [9], the performance of coded SCMA with the tree-based Log-MPA approaches that of coded

A. L. Scharf and B. F. Uchôa-Filho are with GPqCom/LCS/EEL, Federal University of Santa Catarina, Florianópolis, SC, 88040-900, Brazil, e-mail: (ana.scharf@posgrad.ufsc.br, uchoua@eel.ufsc.br).

B. F. da Silva is with Federal Institute Sul-rio-grandense (IFSul), Saporanga-RS, Brazil, e-mail: brunosilva@ifsul.edu.br.

D. L. Ruyet is with CEDRIC/LAETITIA, Conservatoire National des Arts et Metiers, Paris, 75003, France, e-mail: (didier.le_ruyet@cnam.fr).

This paper is partially funded by CNPq (148251/2018-8, 157663/2019-1) and supported by CAPES through PrIntProgram, #698503P.

Digital Object Identifier: 10.14209/jcis.2020.19

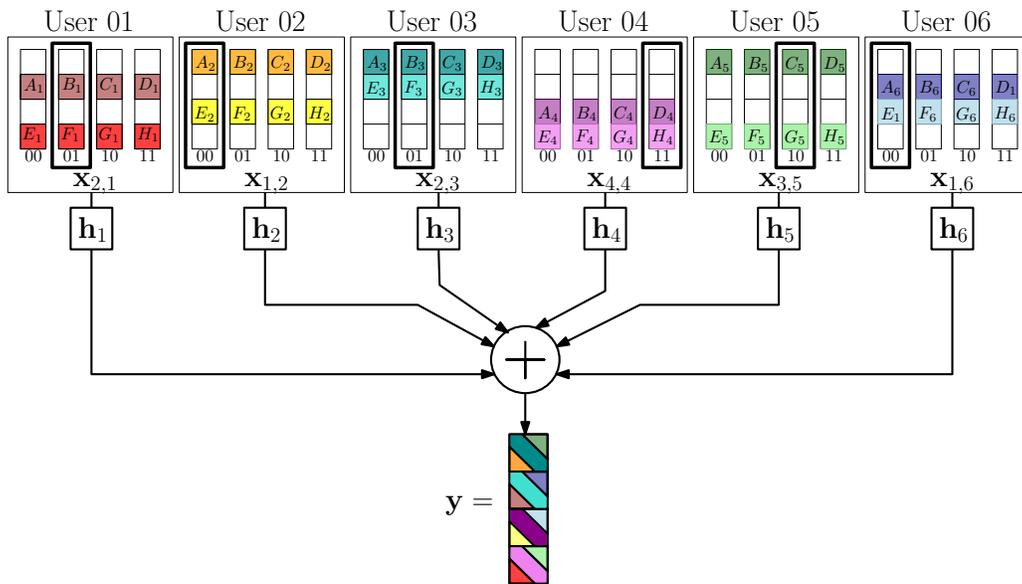


Fig. 1. Example of SCMA transmission for $J = 6$, $K = 4$, $\# = 2$ and $r = 4$.

SCMA with the conventional Log-MPA for a smaller r (higher degree of pruning), thus adding value to the decoder proposed in [9].

The contributions of this work are:

- We introduce error-correcting codes, in particular LDPC codes, to the SCMA system proposed in [9], and show that the performance of coded SCMA with the proposed decoding algorithm approaches the one with the conventional Log-MPA with an even larger number of pruned nodes, thus with an even lower decoding complexity;
- We revisit the Log-MPA detector in [9], in terms of a tree structure based on the conventional Log-MPA equations, to allow for the introduction of the LDPC decoder;
- We present simulation results for an additional scenario, besides the standard 6-user, 4-resource scenario considered in [9];
- We show that the complexity order of the proposed detector under a linear pruning grows exponentially but at a much lower rate than the conventional detector.

Finally, we would like to mention that most works with proposals for reducing the complexity of SCMA decoding do not provide enough information for reproducibility. For example, if the system adopts turbo codes, nothing but the code rate is specified and, in some case not even this information is provided. In some other cases, only the codebook size is reported. For this reason, in this paper, the proposed algorithm is directly compared with the original Log-MPA. We herein provide, in the end of the simulation results section, a GitHub link through which all SCMA allocation matrices and codebooks used in the simulations can be downloaded.

This article is organized as follows. In section II, the SCMA system is described. In Section III, the conventional Log-MPA detector for SCMA is briefly presented. The proposed coded SCMA and the associated decoder are presented in Section IV. In Section V, simulation results are presented. Finally, in

Section VI, we conclude the paper.

Notation: bold lowercase and uppercase indicate vectors and matrices, respectively. $\mathbf{X}_{i,j}$ represents the element in the i -th row and j -th column of the matrix \mathbf{X} . the superscript T denotes transposed matrix. The Euclidean norm of a matrix or a vector is represented by $\|\cdot\|$; uppercase and calligraphic variable represents a set. Finally, $\mathbf{z} \sim \mathcal{CN}(\eta, \mathbf{Z})$ is a complex random vector with Gaussian distribution with average η and covariance matrix \mathbf{Z} ; \mathbf{I} denotes the $K \times K$ identity matrix.

II. SCMA TRANSMISSION SYSTEM

The SCMA system is made up of J independent users, which are multiplexed into K orthogonal resources. In this system the number of users is greater than the number of resources, therefore, the load factor $J \cdot K$ will always be greater than 1 ($J > K$), configuring an overloaded system.

In the transmitter, the j -th user's information index i , with $i \in \{1, \dots, M\}$, is mapped into the codeword $\mathbf{c}_{i,j} \in \mathcal{C}_i$, where \mathcal{C}_i is a complex N -dimensional codebook of size M_i .

The N (non-null) complex values of the codeword are spread over K resources, $N < K$, according to a resource allocation matrix, \mathbf{F} , of dimension $K \times J$, that specifies which resources are used by each user. The $F_{k,j}$ element of the matrix is equal to 1 implies that user j makes use of the k -th resource and, therefore, one of the symbols of the codeword $\mathbf{c}_{i,j}$ is allocated to this resource. Otherwise, $F_{k,j}$ is 0. As an example, consider the matrix:

$$\mathbf{F}_{4 \times 6} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}, \quad (1)$$

with $K = 4$ (resources) and $J = 6$ (users). A characteristic of the resource allocation matrix is its sparsity, that is, it must have few non-null elements. Define the two sets:

$\varphi_g = k : F_{:,g} < 0$, which specifies the N resources used by the j -th user, and the set $\mathbb{f}\phi_j = j : F_{:,g} < 0$, which informs which users are allocated to the k -th resource. The cardinality of φ_g is N , and the cardinality of ϕ_j will be denoted by d_j . In the resource allocation matrix in (1), we have $d_j = 3$ and $N = 2$.

With the application of \mathbf{F} , the codebook \mathbf{C}_g is resized to a sparse version, $\mathbf{X}_g \in \mathbb{C}^{K \times N}$, according to each user's signature, \mathbf{f}_g . In the resized codeword, $\mathbf{x}_{g-g} \in \mathbb{C}^{K \times N}$ components will be the zero symbol, and the other components will receive the original values of the N symbols of \mathbf{C}_{g-g} .

Thus, the received signal is given by:

$$\mathbf{y} = \sum_{g=1}^G \text{diag}(\mathbf{h}_g) \mathbf{x}_{g-g} + \mathbf{z}, \quad (2)$$

where $\mathbf{h}_g = (h_{g-1}, h_{g-2}, \dots, h_{g-K})$ is the vector that represents the channel fading gains of user j in the different resources $k, k = 1, \dots, K$, $\mathbf{x}_{g-g} = (x_{g-g-1}, x_{g-g-2}, \dots, x_{g-g-K})$ represents the codeword associated with the information index i_g of the user j , $\mathbf{y} = (y_1, y_2, \dots, y_K)$ denotes the vector of the signals received in the K resources, and $\mathbf{z} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I})$ indicates the additive white Gaussian complex noise.

Figure 1 exemplifies a possible SCMA system transmission scenario, in which the users codebooks (\mathbf{X}_g) in the SCMA, respecting the resource allocation matrix \mathbf{F} . The colored squares are the N non-null complex values of the codeword for \mathbf{c}_g , which have been allocated to the corresponding resources. Each user selects a codeword, represented by a bold rectangle. The receiver will receive \mathbf{y} , which in turn is the sum of the codewords of the J users in the K resources, jointly with the Gaussian noise.

III. CONVENTIONAL LOG-MPA DETECTOR

The SCMA's sparsity characteristic allows the application of the Log-MPA [7] iterative decoding algorithm. For this, a bipartite graph is constructed whose nodes represent, on one side, the user nodes (UN), and on the other side, the resource nodes (RN) [10]. The allocation of the edges between the UNs and RNs is carried out based on the previously defined sets φ_g and ϕ_j .

Decoding in Log-MPA for the SCMA system is carried out by exchanging messages between UNs and RNs. The message from RN k to UN j is described as [7]:

$$M_{:,j}^{C-\text{ext}k} i_g = \beta_{:,j} M_{:,j}^{C-\text{ext}k} i_g, \quad (3)$$

where

$$\beta_{:,j} = \log \sum_{\phi_j} \exp \left(\sum_{k \in \phi_j} M_{:,j}^{C-\text{ext}k} i_g \right) \quad (4)$$

is a normalization factor of messages from RN to UN, and $M_{:,j}^{C-\text{ext}k} i_g$ is set to

$$M_{:,j}^{C-\text{ext}k} i_g = \max_{i: 2 \in \mathbb{f}1, \dots, M, g} \frac{1}{2\sigma^2} \sum_{l \in \phi_j} h_{g-l} x_{g-g-l} y_l$$

$$\sum_{\varphi_g} h_{g-k} x_{g-g-k} \sum_{\phi_j} M_{:,j}^{C-\text{ext}k} i_g \quad (5)$$

The function \max is the Jacobian logarithm, whose use in the Log-MPA algorithm was considered in [8]. It is defined as follows. Consider initially this maximization for two values, namely a and b . In this case, the function \max returns:

$$\max(a, b) = \log \left(1 + \exp \left(\frac{ka - bk}{k} \right) \right) \quad (6)$$

Note that the result will not be the maximum value, since the \log part adds an adjustment to the final value. The extension of \max to a list with more than two values, as it appears in (5), is possible by first considering the maximization in (6) between any two values in the list and then, by doing the maximization several times, each time between a new value in the list and the result of the maximization at the previous time.

Based on (3) and (5), and considering $M_g = M$ for all $j \in \mathbb{f}1, \dots, J, g$, the complexity of conventional Log-MPA has order $\mathcal{O}(M^3 g)$. It is important to note that, for the maximization in (5), the algorithm needs to consider all the codewords of all users that use the k -th resource except user j . For more details on the Log-MPA algorithm, see [7].

IV. PROPOSED CODED SCMA AND ASSOCIATED DECODING ALGORITHM

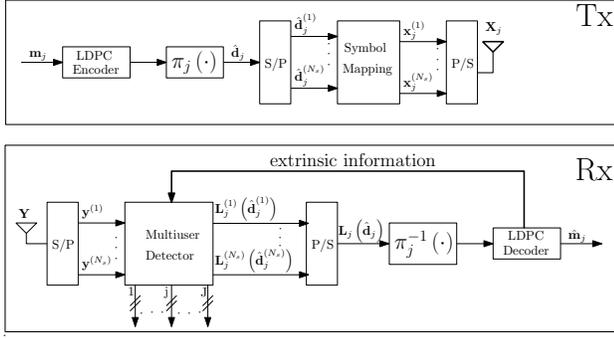
In this section, we will detail the fundamental aspects of implementing the SCMA system with the LDPC code using a Log-MPA detector based on a pruned-tree structure. First, we will show how the LDPC code was added to an SCMA system. Subsequently, we present the process of implementing the pruned tree structure over the Log-MPA detection algorithm, detailing each of its equations. Later, we will explain the calculation of the complexity order of the proposed algorithm and how it is directly linked to the number of pruned branches.

A. SCMA System With LDPC

For better understanding, Figure 2 presents an outline of how the LDPC code for the j -th user interacts with the SCMA system.

In the transmitter of the j -th user, a binary message (\mathbf{m}_j) is LDPC encoded and the codeword is bit-interleaved, to avoid error bursts. Subsequently, the bit-interleaved codeword ($\hat{\mathbf{d}}_j$) is divided into N_B groups ($\hat{\mathbf{d}}_j^{1^0}, \dots, \hat{\mathbf{d}}_j^{\#B^0}$) of b bits each, where $b = \log_2 M$. Each group will be processed by the SCMA system, generating a specific codeword of K complex symbols from the SCMA codebook. To perform transmission, the parallel-to-serial process must take place and finally the compound vector (\mathbf{X}_g) formed by the N_B codewords $\mathbf{x}_g^{1^0}, \dots, \mathbf{x}_g^{\#B^0}$. It is worth mentioning that the SCMA codeword uses K resources, $K - N$ of which receives the zero symbol. The received signal \mathbf{Y} is the superposition (over the fading channels) of all compound vectors from J users, as in eq. (2).

In the reception, the received vector \mathbf{Y} will be processed by a serial-to-parallel block, producing N_B vectors $\mathbf{y}^{1^0}, \dots, \mathbf{y}^{\#B^0}$, which are jointly processed by the SCMA's

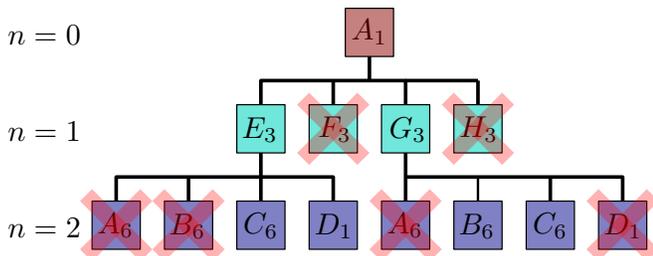

 Fig. 2. SCMA system with LDPC for the g -th user.

multiuser detector. Herein, the detector is the Log-MPA based on the pruned tree structure. The detector outputs soft bits in the form of Log-Likelihood Ratios (LLRs) which are then serialized into $\mathbf{L}_j^g \mathbf{d}_j^g$. Subsequently, $\mathbf{L}_j^g \mathbf{d}_j^g$ is deinterleaved and, finally, delivered to the LDPC decoder as *a priori* information. These steps are repeated iteratively, until the j -th LDPC decoder performs a hard-decision and outputs the decoded message ($\hat{\mathbf{m}}_j$). The multiuser detection takes all users into account, while an independent LDPC decoder is used for each user. Note that we are free to choose the J LDPC codes independently. Herein, we adopt the same LDPC code for all users. But a more general code design is possible.

B. Tree-Based Log-MPA

The central idea of the proposed algorithm is to consider a tree structure in which each level is associated with a certain user that makes use of the resource k , and each branch of this level is associated with a codeword of that user. The tree has $d - 1$ levels.

To get the $\mathcal{M}_{:1}^{C-ext-k} \gg i^k$ message in (5), we have that the level 0 of the tree is associated with the user j . The lower $d - 1$ levels are associated with users in the set $\phi: n f j g$. In the proposed algorithm, with the exception of level zero, only r branches ($r < M_{=}$) at level n , descendants of a node at level $n - 1$, are considered. Considering the resource allocation matrix 1 and Figure 1, Figure 3 demonstrates a tree for the resource $k = 2$, $j = 1$ and for the first codeword $i_1 = 1$. It is noticed that for each new branches, r descending nodes are pruned, represented by the symbols \times . In this way, pruned nodes do not generate new branches, that is, they do not have descending nodes.


 Fig. 3. Possible tree structure of the Log-MPA for the resource $k = 2$, $g = 1$ and $q_1 = 1$.

The calculation of $\mathcal{M}_{:1}^{C-ext-k} \gg i^k$ is broken into steps, performed as you go through the tree from top to bottom, but considering only the sub-tree that remains after pruning. At the $n = 0$ level, we have:

$$\mathcal{M}_{:1}^{C-ext-0} \gg i^k = \frac{1}{2\sigma_f^2} y: h_g \gg k^k x_{g-g} \gg k^k \quad (7)$$

Before we move to the next level, some definitions are needed. In the derivations that follow from this point, users using the k -th resource (except the user j) are sorted in decreasing order, according to the absolute value of the channel gain. The set of indexes of ordered users is denoted by: $\gamma_{n g} = \gamma_1, \gamma_2, \dots, \gamma_{3-1}$, as :

$$h_{w_1} > h_{w_2} > \dots > h_{w_{3-1}} \quad (8)$$

So, at the $n = 1$ level, we have:

$$\mathcal{M}_{:1}^{C-ext-1} \gg i^k = \max_{\delta_{w_1 2 f 1 \dots} \dots} r_{w_1 g} \frac{1}{2\sigma_f^2} k y: h_g \gg k^k x_{g-g} \gg k^k h_{w_1} \gg k^k x_{\delta_{w_1-w_1}} \gg k^k k^2, M_{w_1!}^{C-1-int^k} i_{w_1} \quad (9)$$

where $\max_{-r f g}$ generalizes $\max f g$ in order to select the largest r values that the expression can take. The elements of the set $\mathcal{M}_{:1}^{C-ext-1} \gg i^k$ in (9) are:

$$\mathcal{M}_{:1}^{C-ext-1 \rightarrow 1} \gg i^k, \dots, \mathcal{M}_{:1}^{C-ext-1 \rightarrow A^k} \gg i^k \quad (10)$$

The r indices that maximize (9) are:

$$B = \arg \max_{\delta_{w_1 2 f 1 \dots} \dots} r_{w_1 g} \mathcal{M}_{:1}^{C-ext-1} \gg i^k \quad (11)$$

Generalizing, at the n level, for $n = 1, \dots, d - 1$, and for $b: 2 B_{1 \dots 1} \dots$, with $l = 1, \dots, n - 1$, we have

$$\mathcal{M}_{:1}^{C-ext-1 \rightarrow 1 \dots 1} \gg i^k = \max_{\delta_{w_{=2 f 1 \dots} \dots} \dots} r_{w_{=g}} \frac{1}{2\sigma_f^2} k y: h_g \gg k^k x_{g-g} \gg k^k h_{w_{=}} \gg k^k x_{\delta_{w_{=-w_{=}}} \gg k^k} h_{w_{=}} \gg k^k x_{1 \dots w_{=}} \gg k^k k^2 M_{w_{=1}!}^{C-1-int^k} i_{w_{=}} \dots M_{w_{=1}!}^{C-1-int^k} \gg b_{=} \quad (12)$$

on what

$$B_{1 \dots 1} \dots = \arg \max_{\delta_{w_{=2 f 1 \dots} \dots} \dots} r_{w_{=g}} \mathcal{M}_{:1}^{C-ext-1 \rightarrow 1 \dots 1} \gg i^k \quad (13)$$

Finally, the operation \max is performed, now considering only the surviving nodes in the last level of the tree. Thus, in the proposed algorithm, the new message $\mathcal{M}_{:1}^{C-ext-k} \gg i^k$ is obtained by:

$$\mathcal{M}_{:1}^{C-ext-k} \gg i^k = \max_{1, 2 B_{1 \dots 1} \dots, 1} \mathcal{M}_{:1}^{C-ext-3 \dots 1 \rightarrow 1 \dots 1} \gg i^k \quad (14)$$

Then, the other steps of the conventional Log-MPA algorithm are continued. It is worth mentioning that the normaliza-

tion factor in (4) will be applied only to the surviving nodes at the last level since they are the ones that carry relevant information.

C. Complexity Order

As the objective of the proposed algorithm is to reduce the complexity order, this section presents the necessary calculations to determine the complexity order of the proposed Log-MPA. We will not consider the number of additions and multiplications involved in the calculation of the message $M_{j,l}^{c-ext}$. Instead, we will count the number of times the message is calculated. Then, the conventional Log-MPA has the following complexity order:

$$O = M^3 \tag{3}$$

For the detector proposed in this work, and considering $M_j = M$ for all $j \in \{1, \dots, J\}$, the complexity order is

$$O_{prop} = M \sum_{r=1}^{\infty} M^{2r} \tag{15}$$

or, in closed form,

$$O_{prop} = \begin{cases} M \sum_{r=1}^{\infty} \frac{M^1 r^{3-1}}{r-1}, & \text{se } r = 2, \dots, M-1 \\ M^1 \sum_{r=1}^{\infty} M^{1-d} r^{0}, & \text{se } r = 1. \end{cases} \tag{16}$$

A simple analysis of eq. (16) shows that the decoding complexity of the proposed algorithm grows from M^2 (if r is constant) up to M^3 (if $r = \alpha M$, for $0 < \alpha < 1$). Even so, in the latter case, there is a significant reduction in complexity in comparison to Log-MPA, as can be seen in Table I.

TABLE I
COMPARISON OF COMPLEXITY ORDERS FOR DIFFERENT M VALUES.

M	d	r	O prop.	O conv.	% (prop./conv.)
4	3	1	36	64	56,25
		2	52	64	81,25
		3	68	64	106,25
4	4	1	52	256	20,31
		2	116	256	45,31
		3	212	256	82,81
8	3	1	136	512	26,56
		2	200	512	39,06
		3	264	512	51,56
		4	328	512	64,06
		5	392	512	76,56
		6	456	512	89,06
		7	520	512	101,56
8	4	1	200	4096	4,88
		2	456	4096	11,13
		3	840	4096	20,50
		4	1352	4096	33,00
		5	1992	4096	48,63
		6	2760	4096	67,38
		7	3656	4096	89,25

From the table, it can be seen that, except for the case $d = 3$ with $r = M$, the complexity order of the proposed approach is considerably lower. Moreover, as d increases, the complexity reduction obtained with the proposed detector becomes more significant, since the more levels the tree has, that is, the larger d is, the higher the fraction of pruned nodes.

V. SIMULATION RESULTS

This section presents Monte Carlo simulation results for the bit error rate (BER), as a function of the average bit energy ratio per noise power spectral density (E_b/N_0).

In our coded SCMA system, the rate 1/2 LDPC code of length 128 designed in [11] is adopted in all scenarios simulated (except for the simulation results shown in Fig. 8, which considers uncoded SCMA). We consider 10 iterations in the LDPC decoding in our simulations. The bit-interleaving strategy was also applied, to avoid error bursts. The stopping criterion is 50 frames errors for all scenarios. The flat Rayleigh fading channel model is adopted for all users.

In all simulations, the conventional Log-MPA [7] detector and the proposed one perform 5 iterations. Although the multiuser detector and the LDPC decoder in Fig. 2 can exchange messages repeatedly before the LDPC decoder makes a final hard decision, we considered the open-loop (i.e., one outer iteration) in all of our simulations.

For the proposed decoder, we show results for all possible values of the variable r . The case $r = M$ corresponds to the conventional Log-MPA itself, adopted as a reference. This condition appears in all simulations, therefore the performance of the Log-MPA detector is used as a reference. To reproduce our simulation results, the resource allocation matrices and the codebooks used herein can be found in [12].

The simulation results for the uncoded SCMA system are presented in Figs. 4, 6, and 8, and for the (LDPC) coded SCMA system, in Figs. 5, 7, and 9, respectively, for the following scenarios:

SCMA codebook designed in [13] with parameters $M = 4, J = 6, K = 4$ and $d = 3$;

LDS codebook with parameters $M = 8, J = 6, K = 4$ and $d = 3$;

LDS codebook with parameters $M = 8, J = 12, K = 9$ and $d = 4$.

As can be seen from Figs. 4-9, in all three scenarios the gaps between the BER curves of the proposed tree-based Log-MPA and the corresponding BER curve of the conventional Log-MPA are relatively smaller in the coded SCMA scenarios than in the uncoded SCMA scenarios, for all values of r . This shows that the proposed tree-based Log-MPA not only works well in the coded SCMA system, but it also magnifies the advantages of the method with regard to the uncoded case, first presented in [9].

It is worth mentioning that, if the outer loop in the Rx part of Fig. 2 is closed, and some iterations are performed between the multiuser detector and the LDPC decoders, the aforementioned gaps can be further decreased.

As a final result, we observe that, for all coded SCMA scenarios, a value of r close to $0.5M$ yields a good performance

(near the conventional Log-MPA performance). Although the number of scenarios considered in our simulations is small for this to become a rule of thumb, it is worth investigating how the complexity order of the tree-based Log-MPA grows with M for $r = 0.5M$. The result is shown in Figure 10, for values of M up to 300. It can be seen that, although the complexity order of the proposed detector is still exponential under this condition, it grows much slower than the one for the conventional Log-MPA.

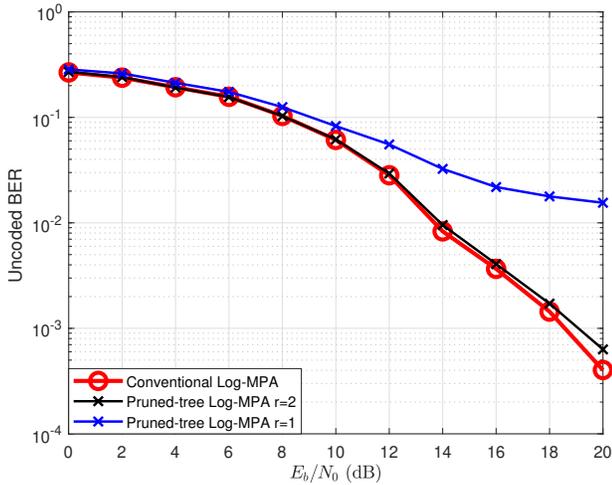


Fig. 4. BER of the uncoded SCMA system with the SCMA codebook designed in [13] with parameters $M = 4$, $N = 6$, $L = 4$ and $K = 3$.

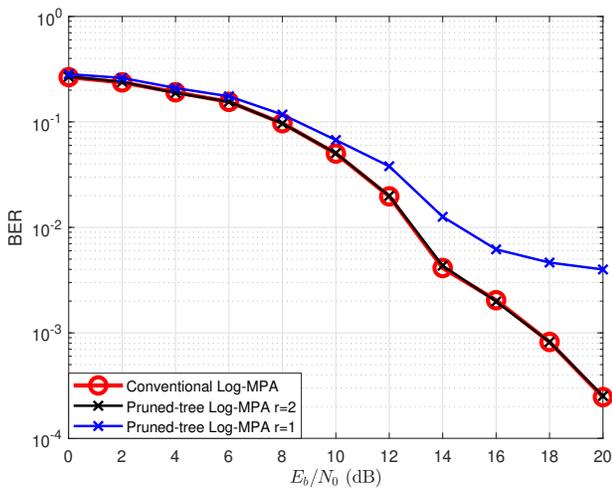


Fig. 5. BER of the coded SCMA system with rate 1/2 LDPC code of length 128 and the SCMA codebook designed in [13] with parameters $M = 4$, $N = 6$, $L = 4$ and $K = 3$.

VI. CONCLUSIONS

In this work, a modified Log-MPA detector for SCMA, which presents good performance with a significant reduction in detection complexity, was proposed. The detector is based on a tree structure that, when properly pruned, provides a reduction in the number of codewords processed by the

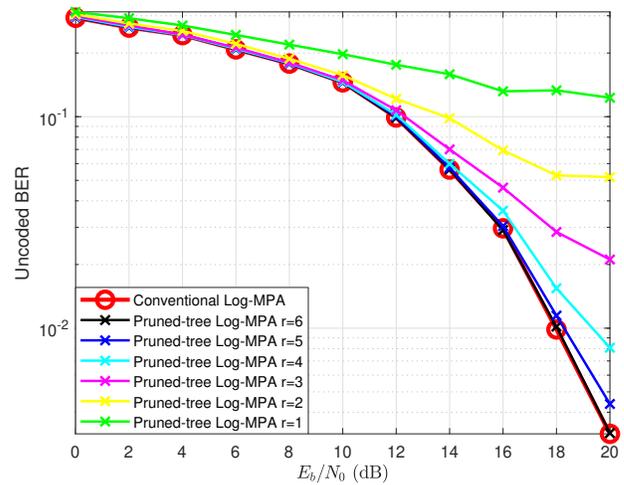


Fig. 6. BER of the uncoded SCMA system with the LDS codebook with parameters $M = 8$, $N = 6$, $L = 4$ and $K = 3$.

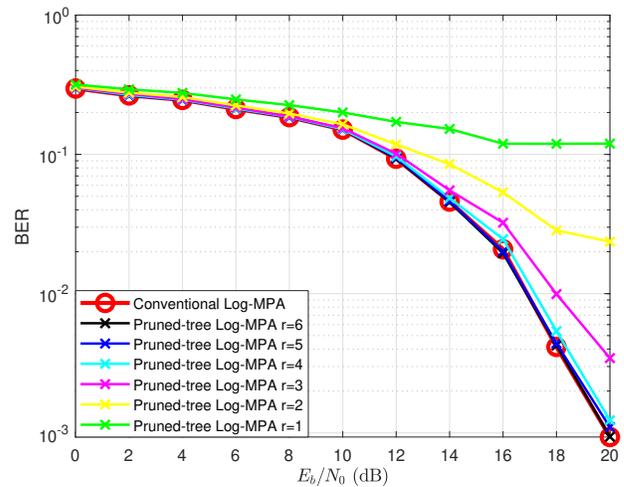


Fig. 7. BER of the coded SCMA system with rate 1/2 LDPC code of length 128 and the LDS codebook with parameters $M = 8$, $N = 6$, $L = 4$ and $K = 3$.

iterative detector. Simulation results for the bit error rate were presented for both uncoded and LDPC coded cases, for three different system scenarios. It was found that the proposed detector performs very well, close to the conventional Log-MPA, with reduced complexity. Moreover, it was verified that the advantages of the proposed detector is even more pronounced in the coded system. We also showed that the complexity order of the proposed detector under a linear pruning, although still exponential in the SCMA codebook size, grows at a much smaller rate when compared to the conventional Log-MPA. Therefore, the contribution of this work is of great interest in practical scenarios.

REFERENCES

[1] H. Nikopour and H. Baligh, "Sparse code multiple access," in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2013, pp. 332–336, doi:10.1109/PIMRC.2013.6666156.

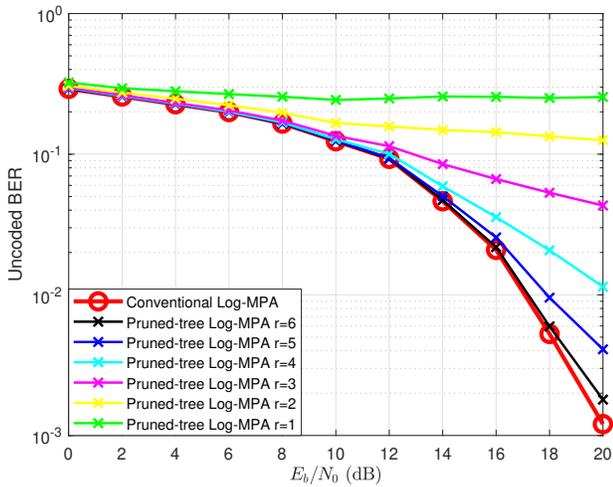


Fig. 8. BER of the uncoded SCMA system with the LDS codebook with parameters $M = 8$, $N = 12$, $L = 9$ and $K = 4$.

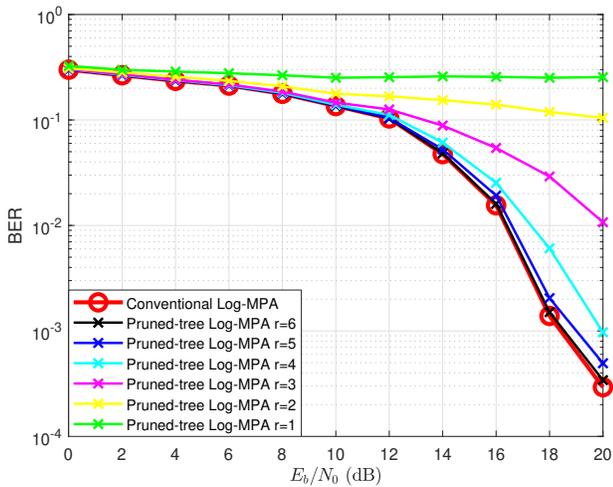


Fig. 9. BER of the coded SCMA system with rate 1/2 LDPC code of length 128 and the LDS codebook with parameters $M = 8$, $N = 12$, $L = 9$ and $K = 4$.

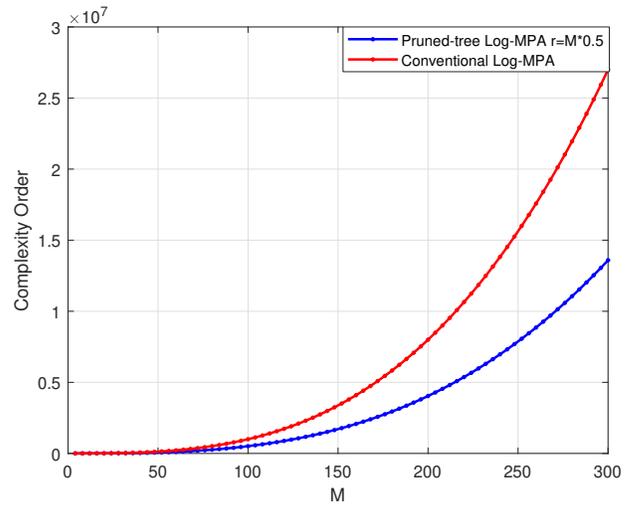


Fig. 10. Complexity order for values of M up to 300, for $A = 0.5$ and $K = 3$.

[2] R. Alizadeh, N. Bäcker, Y. Savaria, and F. Boyer, "Performance characterization of an scma decoder," in *2016 14th IEEE International New Circuits and Systems Conference (NEWCAS)*, June 2016, pp. 1–4, doi: 10.1109/NEWCAS.2016.7604820.

[3] R. Hoshyar, F. P. Wathan, and R. Tafazolli, "Novel low-density signature for synchronous cdma systems over awgn channel," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1616–1626, April 2008, doi: 10.1109/TSP.2007.909320.

[4] M. Taherzadeh, H. Nikopour, A. Bayesteh, and H. Baligh, "Scma codebook design," in *2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*, Sep. 2014, pp. 1–5, doi: 10.1109/VTC-Fall.2014.6966170.

[5] L. Li, J. Wen, X. Tang, and C. Tellambura, "Modified sphere decoding for sparse code multiple access," *IEEE Communications Letters*, vol. 22, no. 8, pp. 1544–1547, Aug 2018, doi: 10.1109/LCOMM.2018.2848273.

[6] M. Vaezi, Z. Ding, and H. V. Poor, *Multiple access techniques for 5G wireless networks and beyond*. Springer, 2019. [Online]. Available: <https://www.springer.com/gp/book/9783319920894>

[7] L. Lu, Y. Chen, W. Guo, H. Yang, Y. Wu, and S. Xing, "Prototype for 5g new air interface technology scma and performance evaluation," *China Communications*, vol. 12, no. Supplement, pp. 38–48, December 2015, doi: 10.1109/CC.2015.7386169.

[8] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of opti-

mal and sub-optimal map decoding algorithms operating in the log domain," in *Proceedings IEEE International Conference on Communications ICC '95*, vol. 2, June 1995, pp. 1009–1013 vol.2, doi: 10.1109/ICC.1995.524253.

[9] A. Scharf, B. Uchôa-Filho, B. Silva, and D. Ruyet, "Decodificação iterativa em árvore de baixa complexidade para SCMA," in *Anais de XXXVII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*. Sociedade Brasileira de Telecomunicações, 2019, doi:10.14209/sbrt.2019.1570558880.

[10] P. Gao, Y. Du, B. Dong, W. Zhu, Z. Chen, and X. Wang, "Low-complexity cs-aided mpa detector for scma systems," *IEEE Communications Letters*, vol. 22, no. 4, pp. 784–787, April 2018, doi: 10.1109/LCOMM.2017.2779859.

[11] *Short Block length LDPC codes for TC synchronization and channel coding*. Washington, D.C: Recommendation for Space Data System Standards (Orange Book), CCSDS 231.1-0-1, no. 1, April 2015. [Online]. Available: <https://public.ccsds.org/Pubs/231x1o1.pdf>

[12] A. L. Scharf. (2020, apr) Scma system parameters presented in the article: Low-complexity tree-based iterative decoding for coded scma. [Online]. Available: <https://github.com/analuscharf/Low-Complexity-Tree-Based-Iterative-Decoding-for-Coded-SCMA>

[13] J. Bao, Z. Ma, M. Xiao, T. A. Tsiftsis, and Z. Zhu, "Bit-interleaved coded scma with iterative multiuser detection: Multidimensional constellations design," *IEEE Transactions on Communications*, vol. 66, no. 11, pp. 5292–5304, Nov 2018, doi: 10.1109/TCOMM.2017.2782325.



Ana Luiza Scharf was born in Florianópolis-SC, Brazil. She received the B.Sc. degree in Telecommunications Engineering from the Federal Institute of Santa Catarina (IFSC), São José, SC, Brazil, in 2018, and her M.Sc. degree from the Federal University of Santa Catarina (UFSC), Florianópolis (SC), Brazil, in 2019. She is currently a doctoral student of Electrical Engineering at UFSC and since 2019 and she is member of Communications Research Group (GPqCom), Florianópolis, Brazil. Her research interests in channel coding, error-correcting

codes, low-complexity receivers and multiple access.

