POLITECNICO DI TORINO
Repository ISTITUZIONALE

Design and Performance of Enhanced Spread Spectrum Aloha for Unsourced Multiple Access

*Original*
Design and Performance of Enhanced Spread Spectrum Aloha for Unsourced Multiple Access / Schiavone, R., Liva, G., Garello, R.. - In: IEEE COMMUNICATIONS LETTERS. - ISSN 1089-7798. - ELETTRONICO. - 28:8(2024), pp. 1790-1794. [10.1109/LCOMM.2024.3412409]

*Availability:*
This version is available at: 11583/2989443 since: 2024-07-06T07:46:01Z

*Publisher:*
IEEE

*Published*
DOI:10.1109/LCOMM.2024.3412409

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*
IEEE postprint/Author's Accepted Manuscript

©2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

28 November 2026

# Design and Performance of Enhanced Spread Spectrum Aloha for Unsourced Multiple Access

Riccardo Schiavone, *Graduate Student Member, IEEE,*, Gianluigi Liva, *Senior Member, IEEE,* and Roberto Garello, *Senior Member, IEEE*

***Abstract*—We analyze the performance of enhanced spread spectrum Aloha (E-SSA) in the framework of unsourced multiple access (UMAC). The asynchronous, unframed transmission of E-SSA is modified to enable a direct comparison with framed UMAC schemes and with Polyanskiy's achievability bound. The design of E-SSA is tailored to the UMAC setting, resorting to short polar codes and the use of a timing channel to improve the energy efficiency of the protocol. We assess the impact of the preamble length and of the spreading factor on the system efficiency. The resulting scheme exhibits simplicity at the transmitter and linear complexity with respect to the number of active users at the receiver, approaching the UMAC achievability bound in close competition with the best known UMAC schemes.**

***Index Terms*—Random access, enhanced spread spectrum Aloha, massive machine type communications, polar codes.**

## I. Introduction

ADDRESSING the connectivity of a massive number of intermittently transmitting devices—a setting often referred to as massive random access (MRA)—poses a significant challenge for the upcoming generations of wireless networks. This challenge has garnered increased attention in recent years due to emerging Internet-of-Things (IoT) applications. The interest in random access (RA) protocols is shared both in terrestrial [1] and non-terrestrial networks [2].

While the design of efficient (coordinated) multiple access channel (MAC) protocols is a well-studied problem in communication systems, the MRA setting reveals several crucial aspects that require new solutions [3]. These include finite blocklength effects [4], the fact that only a small fraction of the users is active at a given moment, and the absence of pre-agreed resource allocation with the base station. From an information theoretic viewpoint, the latter point can be captured by constraining the users to adopt the same codebook, leading to the definition of unsourced multiple access (UMAC) protocols. The development of an UMAC achievability bound [3] provided a fundamental benchmark, triggering the design of new schemes capable of approaching the bound [5]–[10].

R. Schiavone and R. Garello are with the Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Torino, Italy, e-mail: {riccardo.schiavone,roberto.garello}@polito.it

R. Schiavone and G. Liva are with the Institute of Communications and Navigation, German Aerospace Center (DLR), 82234 Wessling, Germany, e-mail: {riccardo.schiavone,gianluigi.liva}@dlr.de.

This work was supported in part by the project 6G-RIC of Germany (Program of "Souverän. Digital. Vernetz", Project No. 16KISK022) and in part by the partnership on "Telecommunications of the Future" of Italy (Program "RESTART", Grant No. PE00000001).

In this paper, we explore the design and the performance of enhanced spread spectrum Aloha (E-SSA) [11] in the UMAC framework. E-SSA is based on spread Aloha [12], and it is a purely asynchronous protocol that combines Aloha [13] with direct sequence spread spectrum to suppress multiuser interference. E-SSA improves the multipacket reception capability of spread Aloha by canceling the interference contribution of decoded packets. The design of E-SSA is based on the 3GPP W-CDMA air interface, and it is optimized to harvest the gains that successive interference cancellation (SIC) can provide. Due to its outstanding performance and lean transmitter/receiver design, E-SSA emerged as a high-efficiency MRA solution, currently adopted in satellite IoT networks [14].

The objective of this letter is two-fold. On one hand, we aim at characterizing the performance of E-SSA in the UMAC setting, showing how a well-established random access protocol that is already deployed in real systems can yield a performance that competes with state-of-the-art UMAC solutions. On the other hand, we address specific E-SSA design choices that stems from the adaptation to the UMAC setting, with emphasis on the short packet transmission regime.

To the best of our knowledge no effort has been put to analyze the performance of E-SSA in the UMAC setting. An obstacle to this task is represented by the asynchronous/unframed nature of E-SSA, which collides with the fixed frame size analysis of [3]. We circumvent this problem by casting a wrap-around version of E-SSA. By doing so, we can provide a fair basis to compare E-SSA with finite block length limits [3] and with advanced UMAC schemes. The focus is on the Gaussian multiple access channel (GMAC) channel, which is a realistic model for satellite communications. We show that a judicious design of E-SSA allows approaching the bounds up to moderate-size user populations, competing with some of the best known schemes [5]. Considering the transmission of small data units (in the order of a few tens of bits), the result is achieved by modifying the original E-SSA design according to the following observations. First, the adoption of codes that performs close to finite-length over single-user channels [4] allows keeping the gap from the UMAC bound [3] small, at low channel loads. Second, the use of protocol information [15] to carry part of the message content improves the energy efficiency.[1] To address the first point, we employ polar codes [17] concatenated with an outer cyclic redundancy

[1]This observation was exploited, for example, in [6], [8] by mapping a portion of the message onto the preambles / signature sequences envisaged by the scheme, in contrast with the random preamble choice adopted in the 5GNR random access protocol [16].

check (CRC), together with successive cancellation list (SCL) decoding [18], [19] at the receiver side. For the second point, we make use of the *timing channel* associated with the packet transmission time. The use of timing channels to convey information was pioneered in [15], [20] (see also [21] for an insightful review of the topic). In the context of random access protocols, timing channels were used in [22] to improve the efficiency of slotted Aloha. In our construction, the timing channel is used to improve the error detection capability of the channel decoder, thus reducing the false alarm rate (which can have a detrimental effect on the SIC algorithm performance). The resulting scheme retains the simple transmitter structure of the original E-SSA, with a decoding complexity that scales linearly in the number of active users.

## II. Preliminaries

We denote random variables with capital letters ($X$), and their realizations with lower-case letters ($x$). We define $[n] = \{0, 1, \dots, n-1\}$. The order-2 finite field is denoted by $\mathbb{F}_2$.

We consider the transmission of $K_a$ active users over the GMAC. The $K_a$ active users transmit simultaneously their message $\boldsymbol{X}_i$, with $i = 1, ..., K_a$, in the same frame of length $n$ real c.u., through a channel affected by additive white Gaussian noise (AWGN). The received vector $\boldsymbol{Y}$ is

$$\boldsymbol{Y} = \boldsymbol{X}_1 + \boldsymbol{X}_2 + ... + \boldsymbol{X}_{K_a} + \boldsymbol{Z},\ \boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}_n, \sigma^2 \boldsymbol{I}_n)$$

where $\boldsymbol{Z}$ is an $n$-dimensional Gaussian vector with $\boldsymbol{0}_n$ (the all-zero vector of length $n$) mean and $\sigma^2 \boldsymbol{I}_n$ covariance matrix, and where $\boldsymbol{I}_n$ is the identity matrix of size $n \times n$ and $\sigma^2$ is the noise variance. All $\boldsymbol{X}_i$ and $\boldsymbol{Y}$ have length $n$. We enforce the power constraint $||\boldsymbol{X}_i||_2^2 \leq nP$ for $i = 1, 2, \dots, K_a$. Following the UMAC setting, all messages are select from a common codebook $\mathscr{C}$ with cardinality $|\mathscr{C}| = 2^K$, i.e., each message carries $K$ bits of information. The per-user signal-to-noise ratio (SNR) is denoted as $E_b/N_0$, where $E_b$ is the energy per information bit, and $N_0$ is the single-sided noise power spectral energy. We have

$$\frac{E_b}{N_0} = \frac{nP}{2K\sigma^2}.$$

Upon observing $\boldsymbol{Y}$, the decoder outputs a list $\mathsf{D}(\boldsymbol{Y})$ containing $K_a$ messages from $\mathscr{C}$. The per-user probability of error (PUPE) is defined as

$$\epsilon := \frac{1}{K_a} \sum_{i=1}^{K_a} \mathbb{P}[\boldsymbol{X}_i \notin \mathsf{D}(\boldsymbol{Y})]\,.$$

### A. Polar Codes

An $(N_p, K_p)$ polar code [17] is specified by the set of indices of the information bits $\mathcal{A} \subseteq [N_p]$. We denote by $\mathcal{F} = [N_p] \backslash \mathcal{A}$ the set of *frozen* bits. From an information vector $\boldsymbol{u} \in \mathbb{F}_2^{K_p}$, the corresponding codeword $\boldsymbol{c}$ is obtained by mapping the $K_p$ bits in $\boldsymbol{u}$ to the $\mathcal{A}$ positions of a binary vector $\boldsymbol{v}$ and freeze to zero the bits of $\boldsymbol{v}$ with index in $\mathcal{F}$. The codeword is generated as $\boldsymbol{c} = \boldsymbol{v}\boldsymbol{F}^{\otimes m}$ where $\boldsymbol{F}^{\otimes m}$ denotes the $m$-fold Kronecker power of Arikan's $2 \times 2$ polarization kernel. The blocklength $N_p$ is equal to $2^m$. The performance of polar codes under SCL decoding can be improved by concatenating an inner polar code with an outer high-rate code. Following [18], we employ an outer $(K_p, K)$ CRC code, resulting in a concatenated CRC-polar code with blocklength $N_p$ and dimension $K$. In the rest of the paper, we adopt the polar code design of the 5GNR standard [18], with the CRC code polynomial of degree 11. We refer to this code construction as CRC-aided polar (CA-polar) code. Following the standard, we perform rate matching (puncturing) to obtain a $(N, K)$ CA-polar code of prescribed blocklength $N \leq N_p$. We use adaptive SCL [19] decoding with maximum list size 256.

## III. Protocol Design

In this section, we describe the design of the scheme. We first define the transmission protocol (Section III-A), including the modifications applied to E-SSA to comply with the fixed frame size setting of [3]. We then outline the algorithms performed at the receiver (Section III-B). Finally, we discuss the role played by the timing channel in Section III-C.

### A. Message Transmission

The transmission of a message resembles closely the one employed originally by E-SSA [11], with a few important differences. First, we adopt a wrap-around approach to turn the continuous, unframed transmission behavior of E-SSA into a framed one. This choice is dictated by the interest in comparing the protocol performance with available performance bounds, which rely on a fixed frame size [3]. Second, owing to their excellent performance in the short blocklength regime, we employ the CA-polar codes in place of the turbo codes used in [11]. Furthermore, we define the transmission time of a packet within a frame to be a function of the message. This information will be used, at the receiver end, to improve the error detection capability of the CA-polar code.

Transmission takes place by encoding the information message with the $(N, K)$ CA-polar code, by spreading the codeword through a long spreading sequence $\boldsymbol{b}$ with a period larger than the spreading factor $s$, and by appending to the spread packet a preamble $\boldsymbol{p}$. The transmission proceeds with a start time obtained by hashing the information message. As for the original E-SSA design, both the preamble and the spreading sequence are unique, i.e., all users employ the same preamble and the same spreading sequence. Let us denote by $\boldsymbol{p} = (p_0, p_1, \dots, p_{L_0-1})$ the length-$L_0$ preamble, with symbols $p_i \in \{-1, +1\}$ for $i = 0, \dots, L_0 - 1$. Furthermore, we denote by $\boldsymbol{b} = (b_0, b_1, \dots, b_{L-1})$ the spreading sequence with spreading factor $s = L/N$ (the length of the spreading sequence is a multiple of the CA-polar code blocklength $N$). As for the preamble, the spreading sequence is binary ($b_i \in \{-1, +1\}$ for $i = 0, \dots, L-1$). We partition the spreading sequence into $N$ disjoint blocks of $s$ symbols each as $\boldsymbol{b} = (\boldsymbol{b}_0, \boldsymbol{b}_1, \dots, \boldsymbol{b}_{N-1})$. Given the spreading sequence $\boldsymbol{b}$, for $\boldsymbol{c} \in \mathbb{F}_2^N$ we introduce the spreading function

$$f_{\mathsf{s}}(\boldsymbol{c}; \boldsymbol{b}) := ((-1)^{c_0}\boldsymbol{b}_0, (-1)^{c_1}\boldsymbol{b}_1, \dots, (-1)^{c_{N-1}}\boldsymbol{b}_{N-1})\,.$$

We finally denote by $f_{\mathsf{H}} : \mathbb{F}_2^K \mapsto [n]$ a uniform hash function. The transmission of a message proceeds as follows.

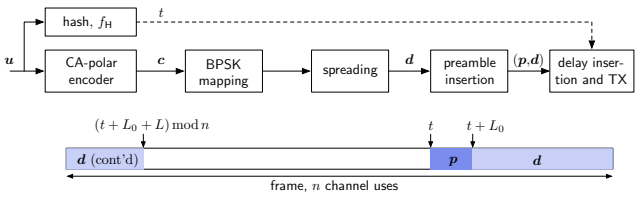



Fig. 1. Structure of the transmission chain.

1. *Polar Encoding:* The $K$-bits information vector $\boldsymbol{u}$ is encoded via the $(N, K)$ CA-polar encoder, resulting in the $N$-bits codeword $\boldsymbol{c}$.
2. *Spreading:* The $N$-bits codeword is spread via the sequence $\boldsymbol{b}$, resulting in the length-$L$ vector $\boldsymbol{d} = f_{\mathsf{S}}(\boldsymbol{c}; \boldsymbol{b})$.
3. *Preamble Insertion and Zero-Padding:* The preamble $\boldsymbol{p}$ is appended to the vector $\boldsymbol{d}$. The resulting length-$(L_0 + L)$ vector is padded with $(n-(L_0+L))$ zeros, resulting into length-$n$ vector $\boldsymbol{x}' = (\boldsymbol{p}, \boldsymbol{d}, 0, 0, \ldots, 0)$.
4. *Hashing and Time Selection:* The transmission time is obtained by hashing the information message as $t = f_{\mathsf{H}}(\boldsymbol{u})$.
5. *Delay Shift and Transmission*. The transmitted vector $\boldsymbol{x}$ is given by the circular-right shift of $\boldsymbol{x}'$ by $t$ positions.

A sketch of the transmitter is shown in Figure 1. Due to the choice of the binary alphabet, the average power is $P = (L_0+L)/n$. Note that, the system employs a pre-defined preamble and a single, pre-defined spreading sequence, i.e., each user employs the same preamble and the same spreading sequence, which are known at the receiver. Remarkably, due to the asynchronous nature of spread Aloha, the use of a single spreading sequence is sufficient to reduce multiuser interference: users that transmit at different times will experience a mutual interference that closely resembles the one of a synchronous spread spectrum system with random spreading sequences assigned to the different users [13].

### B. Detection and Decoding

The receiver runs iteratively. In each iteration, a list of candidate start-of-packet positions is produced through a preamble search. For each candidate position, a decoding attempt is performed via adaptive SCL decoding [19]. If the decoder list contains a codeword that satisfies the CRC code constraints, the associated information message is hashed with the function $f_{\mathsf{H}}$, and the output is checked with the start time of the detected preamble. If the output of the hash function matches the start time, the decision is considered correct and the decoded packet interference contribution is removed from the received signal. The behavior of the receiver is detailed next.

**Initialization:** The iteration count is set to $I = 0$.

**Iterative Decoding and SIC:** At the $\ell$th iteration, the following steps are performed:

1. *Preamble Detection:* The correlation between the preamble and the vector $\boldsymbol{y}$ is computed at each $t \in [n]$ as

$$\Lambda_t = \sum_{j=0}^{L_0-1} p_j y_{(j+t) \bmod n}.$$

The times associated to the $W$ largest values of $\Lambda_t$ are stored in the list $\mathcal{T}$.

2. *Despreading, Decoding, and SIC:* For each $t \in \mathcal{T}$
   a. The observation vector $\boldsymbol{r}$ is extracted as $\boldsymbol{r} = (y_{t'}, \ldots, y_{t''})$ where $t' = (t + L_0) \bmod n$ and $t'' = (t + L_0 + L - 1) \bmod n$.
   b. A soft-estimate of the $j$th codeword bit is obtained by matched filtering (despreading) as

$$\tilde{r}_j = \frac{1}{s} \sum_{k=js}^{(j+1)s-1} r_k b_k.$$

   c. The soft-estimate vector $\tilde{\boldsymbol{r}} = (\tilde{r}_0, \tilde{r}_1, \ldots, \tilde{r}_{N-1})$ is input to the polar code adaptive SCL decoder [19]. The decoder returns a valid codeword $\hat{\boldsymbol{c}}$ or an erasure flag (if no polar code word in the SCL decoder list satisfies the CRC code constraints).
   d. If the decoder outputs a decision $\hat{\boldsymbol{c}}$, the corresponding information message $\hat{\boldsymbol{u}}$ is hashed generating the time index $\hat{t} = f_{\mathsf{H}}(\hat{\boldsymbol{u}})$. If $\hat{t} = t$, then the message $\hat{\boldsymbol{u}}$ is deemed as correct and it is included in the output list $\mathsf{D}(\boldsymbol{y})$.
   e. If the check at point 2.d succeeds, the message is re-encoded according to the transmission procedure described in Section III-A, resulting in the vector $\boldsymbol{x}(\hat{\boldsymbol{u}})$, which is subtracted from the vector $\boldsymbol{y}$

$$\boldsymbol{y} \leftarrow \boldsymbol{y} - a\boldsymbol{x}(\hat{\boldsymbol{u}})$$

   where $a$ is the estimate of the channel amplitude[2] obtained by normalizing the soft-correlation between $\boldsymbol{y}$ and $\boldsymbol{x}(\hat{\boldsymbol{u}})$ by the vectors' length $(L_0 + L)$.

Steps 1 and 2 are iterated for a maximum number of iterations $I_{\max}$. As early stopping criterion, the process ends if—within an iteration—no decoding attempt succeeds according to the test described at step 2.d. It is important to remark that, if we fix $I_{\max}$ and we scale $W$ with the number of active users $K_a$, the detection/decoding algorithm outlined above entails a complexity that is linear in $K_a$.

### C. Error Detection via Timing Channel

The use of the timing channel for error detection, as described in the algorithm described in Section III-B (step 2.d) allows one to drastically reduce the *false alarm rate*, i.e., the rate at which the decoder outputs packets that were not transmitted. False alarms have a detrimental effect on the efficiency of the system, since they introduce artificial interference through the SIC process. It is hence of paramount importance to keep the false alarm rate some orders of magnitude lower than the target PUPE. In our simulations, for instance, with 125 active users, we counted exactly three false alarms in 800 simulated frames, whereas with $K_a = 100$ and $K_a = 75$ no false alarms were detected. As an alternative to the use of the timing channel, one may improve the error detection capability of the SCL decoder. The result can be achieved by limiting the SCL list size, or by introducing a stronger CRC. In either case, the price to be paid is an deterioration of the error correction capability of the decoder, resulting in a loss of coding gain,

[2]Note that the detection/decoding algorithm described in this section does not make use of any prior information on the unitary channel amplitude.

thus in a loss of energy efficiency. An obvious question relates to the practicality of using the timing information in a real system. Answering thoroughly this question goes beyond the scope of this paper. However, a possible direction to explore is the use of a beacon signal (transmitted periodically by the base station) to announce the start of frame, which is used by the terminals as reference to compute the transmission time. Uncertainties related to the delay incurred by the transmission may be taken into account by relaxing the criterion of step 2.d in the algorithm of Section III-B: a decoding attempt may be deemed successful if $\hat{t} \in [t-\delta, t+\delta]$, where $\delta$ is a suitably chosen maximum delay offset. For example, in our numerical test we did not experience any visible degradation in error detection capability with values of $\delta$ up to 150 channel uses. Closed-loop timing control mechanisms, as the ones implemented by the 5GNR [16] and by the DVB-RCS standards [23], may be also considered as means to keep the timing error within a pre-defined range.

## IV. Numerical Results

In this section, we provide numerical results obtained via Monte Carlo simulations. The results are obtained for a frame size $n = 30000$. A $(1000, 100)$ CA-polar code is used, and the number of iterations is set to $I_{\max} = 50$. The target PUPE is $\epsilon^\star = 5 \times 10^{-2}$. For the simulations, we used a single randomly-generated spreading sequence, with symbols that are independent and uniformly distributed in $\{\pm 1\}$.

A first set of results deals with the choice of the spreading factor. In Figure 2, we report the SNR required to achieve the target PUPE as a function of the spreading factor. The results are provided for various numbers of active users, and are obtained under genie-aided preamble detection. That is, the preamble is omitted. The chart shows how small spreading factors tend to penalize the performance. For larger spreading factors, and at low channel loads (small $K_a$), the required SNR closely approaches the one of the single-user case. The result holds up to $K_a = 75$. Above this value, there is a rapid deterioration of the performance. This behavior is analyzed in Figure 3. Here, the PUPE is depicted as a function of the SNR, for various numbers of active users. The spreading factor is set to 25. On the same chart, the normal approximation [4] for $(1000, 100)$ codes is provided, as well as the single-user $(1000, 100)$ CA-polar code performance. The PUPE approaches the single-user performance at low enough error probability. The SNR at which the convergence to the single user case varies with the number of active users. When the number of active users is sufficiently low, the PUPE tightly matches the single-user probability of error already at error probabilities larger than the target $\epsilon^\star$. As the number of users grows, the convergence happens at PUPE values that are below $\epsilon^\star$, giving rise to a visible increase of the required SNR. An analysis of this phenomenon, which is common in multiuser systems, was provided in [24].

A second set of results analyzes the impact on the system performance of the number $W$ of start-of-packet candidates used at step 1 of the algorithm described in Section III-B. The results, obtained for $s = 15$ and for various preamble lengths

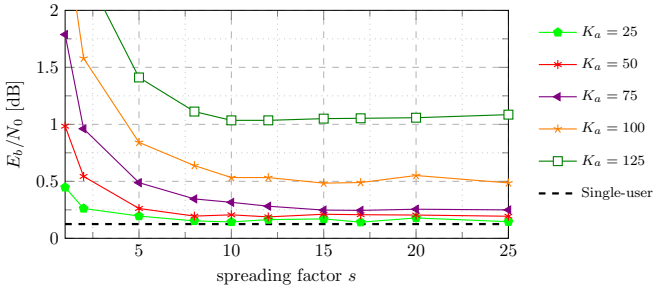



Fig. 2. Effect of the spreading factor $s$ on the minimum $E_b/N_0$ required for $\epsilon^\star = 5 \times 10^{-2}$. Genie-aided preamble detection.

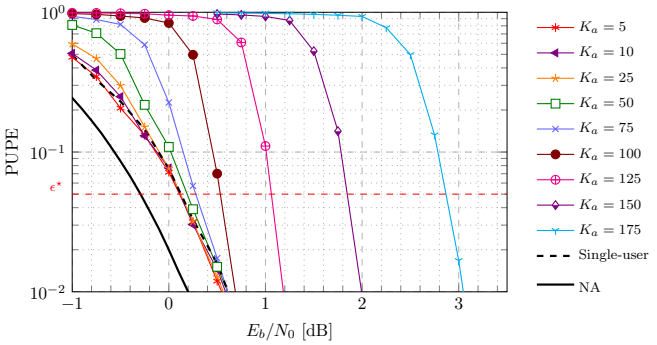



Fig. 3. Effect of the channel load $K_a$ on the PUPE in a system with no preamble. Spreading factor $s = 25$. Genie-aided preamble detection.

$L_0$, are depicted in Figure 4 in terms of SNR required to achieve the target PUPE as a function of the energy overhead introduced by the preamble, given by $\Delta E := 1 + L_0/L$. As one would expect, a larger preamble overhead (larger $L_0$) enables an accurate preamble detection already with small values of $W$, resulting also in a smaller average number of decoding attempts. However, the result comes at the expense of an increase in the energy cost entailed by the preamble. Operating the system with lower preamble lengths allows to limit the energy loss. Anyhow, the preamble miss-detection probability can be kept low only by increasing $W$, with an obvious implication in complexity due to the larger number of decoding attempts. The analysis highlights the trade-off that exists at fixed SNR/PUPE between a reduction of the preamble overhead and the corresponding increase of decoding complexity, suggesting that the preamble length shall be selected by finding an acceptable compromise between energy overhead and size of the list of start-of-packet candidates.

Figure 5 provides a comparison of E-SSA with the UMAC achievability bound and several UMAC schemes. The results are obtained by setting the E-SSA preamble length to $L_0 = 3050$ and a spreading factor $s = 25$, yielding an energy overhead $\Delta E \approx 0.5$ dB. The E-SSA performance is provided for $W = 100$ and for $W = 250$. The performance with genie-aided preamble detection is given as reference, together with the UMAC achievability bound from [3]. The comparison includes the 5GNR two-step random access procedure [16] under SIC with parameters that have been chosen to match the simulation setup [25], the sparse Kronecker-product coding of [26], the synchronous spread spectrum scheme of [5], the enhanced irregular repetition slotted Aloha scheme of [7], sparse interleave-division multiple access [6] (which exploits

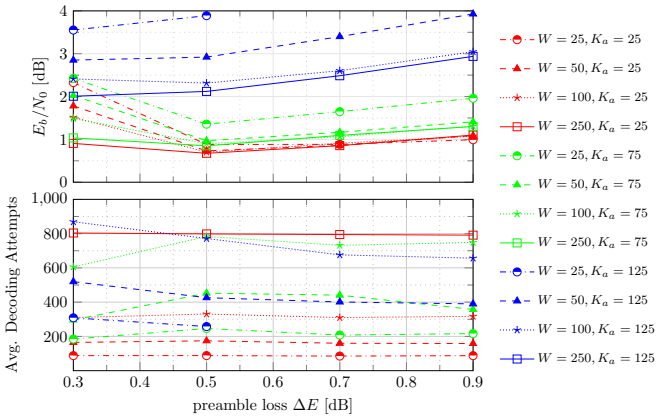



Fig. 4. Minimum $E_b/N_0$ (above) and average number of decoding attempts (below) for $\epsilon^\star = 5 \times 10^{-2}$ as a function of the preamble loss, $s = 15$.

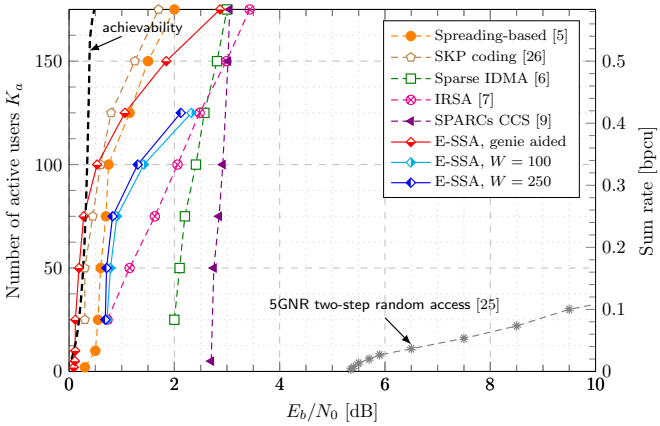



Fig. 5. $E_b/N_0$ required to achieve the target $\epsilon^\star = 5 \times 10^{-2}$.

joint multiuser decoding), and sparse regression codes / coded compressive sensing [9]. The performance of E-SSA shows to be competitive, especially in the moderate load regime (up to 100 active users), requiring only single-user detection and decoding blocks at the receiver side. Among the competitors, only the schemes [5], [26] outperform E-SSA over the entire channel load range, with a gain that is nevertheless limited to 0.2 dB for [5] and 0.5 dB for [26] up to 75 users.

## V. Conclusion

We analyzed the performance of enhanced spread spectrum Aloha (E-SSA) in the framework of unsourced multiple access (UMAC). The asynchronous, unframed transmission of E-SSA has been modified to enable a comparison with framed UMAC schemes. We have improved the energy efficiency of the scheme by introducing a short polar code and a timing channel. We have shown how the design of the different components of E-SSA affects the receiver complexity and the energy efficiency of the system. Results show that a careful design of E-SSA yields a performance that is competitive with state-of-the-art UMAC schemes, with a simple transmitter and a linear receiver complexity in the number of active users.

## Acknowledgments

The authors would like to thank Monica Visintin and they are also grateful to Yury Polyanskiy for providing the software used to compute the achievability bound of [3].

## References

[1] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive Machine-type Communications in 5G: Physical and MAC-layer solutions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 59–65, Sep. 2016.

[2] S. Cioni, R. De Gaudenzi, O. Del Rio Herrero, and N. Girault, "On the Satellite Role in the Era of 5G Massive Machine Type Communications," *IEEE Netw.*, vol. 32, no. 5, pp. 54–61, Sept./Oct. 2018.

[3] Y. Polyanskiy, "A perspective on massive random-access," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2017.

[4] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel Coding Rate in the Finite Blocklength Regime," *IEEE Trans. Inform. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[5] A. K. Pradhan, V. K. Amalladinne, K. R. Narayanan, and J.-F. Chamberland, "Polar Coding and Random Spreading for Unsourced Multiple Access," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2020.

[6] A. K. Pradhan, V. K. Amalladinne, A. Vem, K. R. Narayanan, and J.-F. Chamberland, "Sparse IDMA: A Joint Graph-Based Coding Scheme for Unsourced Random Access," *IEEE Trans. Commun.*, vol. 70, no. 11, pp. 7124–7133, Nov. 2022.

[7] E. Marshakov, G. Balitskiy, K. Andreev, and A. Frolov, "A Polar Code Based Unsourced Random Access for the Gaussian MAC," in *Proc. IEEE Veh. Technol. Conf.*, Sep. 2019.

[8] D. Truhachev, M. Bashir, A. Karami, and E. Nassaji, "Low-Complexity Coding and Spreading for Unsourced Random Access," *IEEE Commun. Lett.*, vol. 25, no. 3, pp. 774–778, Mar. 2021.

[9] A. Fengler, P. Jung, and G. Caire, "SPARCs for Unsourced Random Access," *IEEE Trans. Inf. Theory*, no. 10, pp. 6894–6915, Oct. 2021.

[10] M. Ozates, M. Kazemi, and T. M. Duman, "Unsourced Random Access Using ODMA and Polar Codes," *IEEE Wireless Commun. Lett.*, 2024.

[11] O. Del Rio Herrero and R. De Gaudenzi, "A High Efficiency Scheme for Quasi-Real-Time Satellite Mobile Messaging Systems," in *Proc. Int. Workshop Signal Processing for Space Commun.*, Oct. 2008.

[12] N. Abramson, "VSAT data networks," *Proc. IEEE*, vol. 78, no. 7, pp. 1267–1274, Jul. 1990.

[13] ——, "The ALOHA System - Another Alternative for Computer Communications," in *Proc. Fall Joint Comp. Conf.*, 1970, pp. 281–285.

[14] *TS 102 721-1 V1.1.1, Satellite Earth Stations and Systems; Air Interface for S-band Mobile Interactive Multimedia (S-MIM)*, ETSI Std., 2011.

[15] R. Gallager, "Basic Limits on Protocol Information in Data Communication Networks," *IEEE Trans. Inf. Theory*, pp. 385–398, Jul. 1976.

[16] *5GNR: MAC protocol specification*, 3rd Generation Partnership Project (3GPP) Std. TS 138.321, Rev. 16.1.0, Jul. 2020.

[17] E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[18] I. Tal and A. Vardy, "List Decoding of Polar Codes," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.

[19] B. Li, H. Shen, and D. Tse, "An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check," *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044–2047, 2012.

[20] V. Anantharam and S. Verdu, "Bits Through Queues," *IEEE Trans. Inf. Theory*, vol. 42, no. 1, pp. 4–18, Jan. 1996.

[21] A. Ephremides and B. Hajek, "Information Theory and Communication Networks: An Unconsummated Union," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2416–2434, Oct. 1998.

[22] L. Galluccio, G. Morabito, and S. Palazzo, "TC-Aloha: A Novel Access Scheme for Wireless Networks with Transmit-Only Nodes," *IEEE Trans. Wireless Commun.*, vol. 12, no. 8, pp. 3696–3709, Aug. 2013.

[23] *Interaction Channel for Satellite Distribution Systems (DVB-RCS)*, EN 301 790 v1.5.1, ETSI Std., May 2009.

[24] S. S. Kowshik and Y. Polyanskiy, "Fundamental Limits of Many-User MAC With Finite Payloads and Fading," *IEEE Trans. Inf. Theory*, vol. 67, no. 9, pp. 5853–5884, Sep. 2021.

[25] G. Liva and Y. Polyanskiy, "Unsourced Multiple Access: A Coding Paradigm for Massive Random Access," *in preparation, Proceedings of the IEEE*, 2024.

[26] Z. Han, X. Yuan, C. Xu, S. Jiang, and X. Wang, "Sparse Kronecker-product coding for unsourced multiple access," *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2274–2278, 2021.