






# On LDPC Code Based Massive Random-Access Scheme for the Gaussian Multiple Access Channel

Anton Glebov<sup>1</sup>, Luiza Medova<sup>2</sup>, Pavel Rybin<sup>1,2</sup>, and Alexey Frolov<sup>1</sup>

<sup>1</sup> Skolkovo Institute of Science and Technology, Moscow, Russia  
anton.glebov@skolkovotech.ru, {p.rybin, al.frolov}@skoltech.ru

<sup>2</sup> Institute for Information Transmission Problems, Moscow, Russia  
luiza.medova@phystech.edu

**Abstract.** This paper deals with the problem of massive random access for Gaussian multiple access channel (MAC). We continue to investigate the coding scheme for Gaussian MAC proposed by A. Vem et al. in 2017. The proposed scheme consists of four parts: (i) the data transmission is partitioned into time slots; (ii) the data, transmitted in each slot, is split into two parts, the first one set an interleaver of the low-density parity-check (LDPC) type code and is encoded by spreading sequence or codewords that are designed to be decoded by compressed sensing type decoding; (iii) the another part of transmitted data is encoded by LDPC type code and decoded using a joint message passing decoding algorithm designed for the T-user binary input Gaussian MAC; (iv) users repeat their codeword in multiple slots. In this paper we are concentrated on the third part of considered scheme. We generalized the PEXIT charts to optimize the protograph of LDPC code for Gaussian MAC. The simulation results, obtained at the end of the paper, were analyzed and compared with obtained theoretical bounds and thresholds. Obtained simulation results shows that proposed LDPC code constructions have better performance under joint decoding algorithm over Gaussian MAC than LDPC codes considered by A. Vem et al. in 2017, that leads to the better performance of overall transmission system.

**Keywords:** NOMA · Gaussian MAC · Massive random-access LDPC code · PEXIT charts

## 1 Introduction

Current wireless networks are designed with the goal of servicing human users. Next generation of wireless networks is facing a new challenge in the form of machine-type communication: billions of new devices (dozens per person) with

---

The research was carried out at Skoltech and supported by the Russian Science Foundation (project no. 18-19-00673).

dramatically different traffic patterns are expected to go live in the next decade. The main challenges are associated with: (a) huge number of autonomous devices connected to one access point, (b) low energy consumption, (c) short data packets. This problem has attracted attention (3GPP and 5G-PPP) under the name of mMTC (massive machine-type communication).

There are  $K \gg 1$  users, of which only  $T$  have data to send in each time instant. A base station (BS) sends periodic beacons, announcing frame boundaries, so that the uplink (user-to-BS) communication proceeds in a frame-synchronized fashion. Length of each frame is  $n$ . Each active user has  $k$  bits that it intends to transmit during a frame, where a typical value is  $k \approx 100$  bit. The main goal is to minimize the energy-per-bit spent by each of the users. We are interested in grant-free access (5G terminology). That is, active users transmit their data, without any prior communication with the BS (without resource requests). We will focus on the Gaussian multiple-access channel (GMAC) with equal-power users, i.e.

$$y = \sum_{t=1}^T x^{(t)} + z,$$

where  $z \sim \mathcal{N}(0, N_0/2 = \sigma^2)$  and  $\mathbb{E}[|x^{(i)}|^2] \leq P$ .

This paper deals with construction of low-complexity random coding schemes for GMAC (indeed we restrict our consideration to the case of binary input GMAC). Let us emphasize the main difference from the classical setting. Classical information theory provided the exact solutions for the case of all-active users, i.e.  $T = K$ . Almost all well-known low-complexity coding solutions for the traditional MAC channel (e.g. [10]) implicitly assume some form of coordination between the users. Due to the gigantic number of users we assume them to be symmetric, i.e. the users use the same codes and equal powers. Here we continue the line of work started in [7, 8, 14]. In [8] the bounds on the performance of finite-length codes for GMAC are presented. In [7] Ordentlich and Polyanskiy describe the first low-complexity coding paradigm for GMAC. The improvement (in terms of required *energy-per-bit*  $E_b/N_0$ ) was given in [14]. Recall, that  $E_b/N_0$  is calculated as follows. Assume a user transmits  $k$  bits by means of  $n$  channel uses, then

$$E_b/N_0 = \frac{nP}{kN_0} = \frac{nP}{k2\sigma^2}.$$

In this paper we continue to investigate the coding scheme from [14]. The proposed scheme consists of four parts:

- the data transmission is partitioned into time slots;
- the data, transmitted in each slot, is split into two parts, the first one (preamble) allows to detect users that were active in the slot. It also set an interleaver of the low-density parity-check (LDPC) type code [2, 12] and is encoded by spreading sequence or codewords that are designed to be decoded by compressed sensing type decoding;

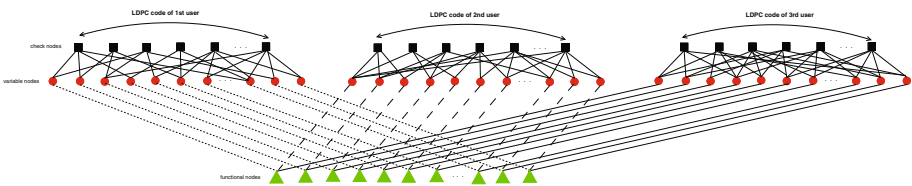
- the second part of transmitted data is encoded by LDPC type code and decoded using a joint message passing decoding algorithm designed for the  $T$ -user binary input GMAC;
- users repeat their codeword in multiple slots and use successive interference cancellation.

The overall scheme can be called  $T$ -fold irregular repetition slotted ALOHA (IRSA, [4, 6]) scheme for GMAC. The main difference of this scheme in comparison to IRSA is as follows: any collisions of order up to  $T$  can be resolved with some probability of error introduced by Gaussian noise.

In this paper we are concentrated on the third part of considered scheme. Our contribution is as follows. We generalized the protograph extrinsic information transfer charts (EXIT) to optimize the protograph of LDPC code for GMAC. The simulation results, obtained at the end of the paper, were analyzed and compared with obtained theoretical bounds and thresholds. Obtained simulation results shows that proposed LDPC code constructions have better performance under joint decoding algorithm over Gaussian MAC than LDPC codes considered in [14], that leads to the better performance of overall system.

## 2 Iterative Joint Decoding Algorithm

We consider  $T$  independent users, being sent to a single receiver. User  $t$ ,  $t \in \{1, \dots, T\}$ , is encoded by  $\mathcal{C}^{(t)}$ , where  $\mathcal{C}^{(t)}$  is an irregular LDPC code with codeword length  $n$  and rate  $r$ . The codewords  $\mathbf{c}^{(1)}, \mathbf{c}^{(2)}, \dots, \mathbf{c}^{(T)}$  are BPSK modulated, and therefore the sequences  $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(T)}, \mathbf{x}^{(i)} \in \{-1, +1\}^n$  are transmitted through a communication channel. The received signal  $\mathbf{y}$  is an element-wise sum of these sequences affected by Gaussian noise. The joint multi-user decoder is expected to recover all the codewords based on that signal.



**Fig. 1.** Joint decoder graph representation for  $T = 3$  (Color figure online)

The decoder employs a low-complexity iterative belief propagation (BP) decoder that deals with a received soft information presented in LLR (log likelihood ratio) form. The decoding system can be represented as a factor graph, which is shown in Fig. 1. The factor graph of the  $T$ -user LDPC-MAC is composed of the  $T$  LDPC graphs, which are connected through state nodes (marked with green color). These nodes correspond to the elements of the received sequence  $\mathbf{y}$ .

The belief propagation decoding algorithm proceeds as follows. The LLR values of variable nodes for each user are initialized with zero values assuming equal probability for 1 and  $-1$  values and the joint decoder perform  $\ell_O$  outer iterations, where each iteration includes the following steps:

- maximum likelihood decoding of state nodes;
- performing  $\ell_I$  inner iterations of BP decoding for users' LDPC codes and updating LLR values of variable nodes (it's done in parallel);

The message update rules in the graph of each user follow from usual LDPC BP decoding algorithm but it is necessary to describe the update rule through state nodes. In accordance with principles of message-passing algorithms, the outgoing message from the  $i^{\text{th}}$  variable node of user  $t$  to the connected state node is computed as

$$m_{vs,i}^t = \log \frac{p(x_i^t = 1)}{p(x_i^t = -1)}, \quad e^{m_{vs,i}^t} = \frac{p(x_i^t = 1)}{p(x_i^t = -1)},$$

where  $x_i^t$  denotes the  $i^{\text{th}}$  transmitted code bit and  $y_i$  denotes the channel output.

Considering standard function node message-passing rules [9], we compute the message sent to  $i^{\text{th}}$  variable node of user  $t$  from the state node:

$$m_{sv,i}^t = \log \frac{p(x_i^t = 1|y)}{p(x_i^t = -1|y)} = \log \left( \frac{\sum_{\sim x_i^{(t)} j \neq t} \prod p(x_i^j = 1) p(y_i | x_i^{(1)}, \dots, x_i^{(t)} = 1, \dots, x_i^{(n)})}{\sum_{\sim x_i^{(t)} j \neq t} \prod p(x_i^j = -1) p(y_i | x_i^{(1)}, \dots, x_i^{(t)} = -1, \dots, x_i^{(n)})} \right)$$

We can simplify it in the following way:

$$m_{sv,i}^t = \log \left( \frac{\sum_{\sim x_i^{(t)} j \neq t} \prod e^{1_{x_j} X_j} p(y_i | x_i^{(1)}, \dots, x_i^{(t)} = 1, \dots, x_i^{(n)})}{\sum_{\sim x_i^{(k)} j \neq t} \prod e^{1_{x_j} X_j} p(y_i | x_i^{(1)}, \dots, x_i^{(t)} = -1, \dots, x_i^{(n)})} \right), \quad (1)$$

$$\text{where } 1_{x_t} = \begin{cases} 1, & x_i^{(j)} = 1 \\ 0, & x_i^{(j)} = -1. \end{cases}$$

The number of computations necessary to obtain the outgoing messages from state nodes grows exponentially with the number of users, nevertheless, this number of users usually remains small, and we will therefore not be concerned with this fact.

### 3 PEXIT Charts

Extrinsic Information Transfer (EXIT) charts [1] can be used for the accurate analysis of the behavior of LDPC decoders. But since the usual PEXIT analysis

cannot be applied to the study of protograph-based [13] LDPC codes we will use a modified EXIT analysis for protograph-based LDPC codes (PEXIT) [5]. This method is similar to the standard EXIT analysis in that it tracks the mutual information between the message edge and the bit value corresponding to the variable node on which the edge is incident, while taking into account the structure of the protograph. In our work we use the notation from [5] to describe EXIT charts for protograph-based LDPC codes.

Let  $I_{Ev}$  denotes the extrinsic mutual information between a message at the output of a variable node and the codeword bit associated to the variable node:

$$I_{Ev} = I_{Ev}(I_{Av}, I_{Es}),$$

where  $I_{Av}$  is the mutual information between the codeword bits and the check-to-variable messages and  $I_{Es}$  is the mutual information between the codeword bits and the state-to-variable messages. Since the PEXIT tracks the mutual information on the edges of the protograph, we define  $I_{Ev}(i, j)$  as the mutual information between the message sent by the  $j^{th}$  variable node to the  $i^{th}$  check node and the associated codeword bit:

$$I_{Ev}(i, j) = J \left( \sqrt{\sum_{s \neq i} [J^{-1}(I_{Av}(s, j))]^2 + [J^{-1}(I_{Es}(j))]^2} \right)$$

where  $J(\sigma)$  is given by [1]:

$$J(\sigma) = 1 - \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma^2} \exp \left[ -\frac{1}{2} \left( \frac{y - \frac{\sigma^2}{2}x}{\sigma} \right)^2 \right] \log_2(1 + e^{-y}) dy.$$

Similarly, we define  $I_{Ec}$ , the extrinsic mutual information between a message at the output of a check node and the codeword bit associated to the variable node receiving the message:

$$I_{Ec} = I_{Ec}(I_{Ac}),$$

where  $I_{Ac}$  is the mutual information between one input message and the associated codeword bit and  $I_{Ac} = I_{Ev}$ . Accordingly, the mutual information between the message sent by  $i^{th}$  check node to  $j^{th}$  variable node and the associated codeword bit is described as:

$$I_{Ec}(i, j) = 1 - J \left( \sqrt{\sum_{s \neq j} [J^{-1}(1 - I_{ac}(i, s))]^2} \right).$$

The mutual information between the  $j^{th}$  variable node and the message passed to the state node is denoted as  $I_{Evs}(j)$  and is given by:

$$I_{Evs}(j) = J \left( \sqrt{\sum_s [J^{-1}(I_{av}(s, j))]^2} \right).$$

Next we need to compute the mutual information  $I_{Es}$ . In order to get an idea about the probability density function of (1) for user  $t$ , we generate samples of the outgoing LLRs through (1) based on the samples of the received LLRs from other users whose PDF is approximated with  $\mathcal{N}(\mu_{Evs}, 2\mu_{Evs})$ , where  $\mu_{Evs} = \frac{J^{-1}(I_{Evs})}{2}$ . To numerically estimate  $\mu_{Es}$  and obtain the required mutual information as  $I_{Es} = J(\mu_{Es})$ , we refer to [11], where the following three approaches are proposed:

- Mean-matched Gaussian approximation : the mean  $\mu$  is estimated from samples and we set  $\mu_{Evs} = \mu$  and  $\sigma_{Evs}^2 = 2\mu$ .
- Mode-matched Gaussian approximation : given a sufficiently large number of  $N$  samples generated through (1), the mode  $m$  is estimated from samples and we set  $\mu_{Evs} = m$  and  $\sigma_{Evs}^2 = 2m$ .
- Gaussian mixture approximation: mean values  $\mu_1, \dots, \mu_T$  and the weights  $a_1, \dots, a_T$  are estimated from samples and  $I_{Es} = a_1 J(\mu_1) + \dots + a_k J(\mu_T)$ .

The rationale for using these approximations was shown in [11]. Furthermore, the authors compared the performance of these approaches. The mode-matched method was found to give the maximum output mutual information and the joint codes designed by using this approximation also yield the lowest decoding bit error probability compared to the other two approaches.

Each user calculate  $I_{APP}(j)$ , the mutual information between the posteriori probability likelihood ratio evaluated by the  $j^{th}$  variable node and the associated codeword bit.

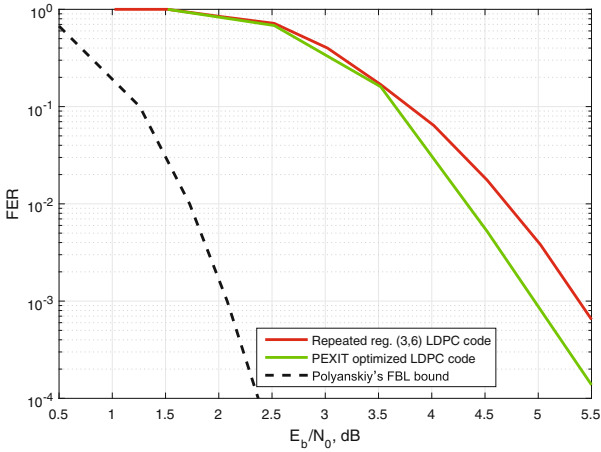
$$I_{APP}(j) = J \left( \sqrt{\sum_s [J^{-1}(I_{Av}(s, j))]^2 + [J^{-1}(I_{Es}(j))]^2} \right).$$

The convergence is declared if each of  $I_{APP}(j)$  reaches 1 as the iteration number tends to infinity.

## 4 Numerical Results

In this section the simulation results, obtained for the cases  $T = 2$  and  $T = 4$ , are represented. Let us first consider the simulation results for  $T = 2$  (Fig. 2). For this case we compare the Frame Error Rate (FER) performance of rate-1/4 LDPC code (364, 91) from [14] obtained by repetition of each code bit of regular (3,6) LDPC code twice, rate-1/4 LDPC code (364, 91) optimized by PEXIT charts method described above and Polyanskiy’s finite block length (FBL) bound for 2 user case.

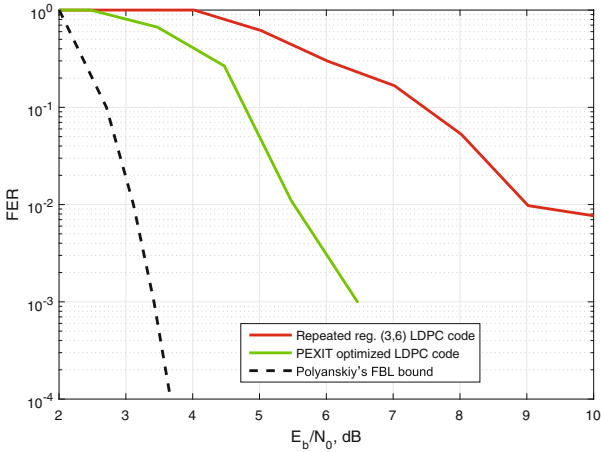
As we can see in Fig. 2 proposed PEXIT-optimized LDPC code construction outperforms LDPC code construction from [14] by about 0.5dB. In the same time the gap between Polyanskiy’s FBL bound and PEXIT-optimized LDPC code is about 3 dB. But we would like to point out that used here Polyanskiy’s FBL bound is for Gaussian signal and not for Binary Phase-Shift Keying (BPSK)



**Fig. 2.** Simulation results for  $T = 2$  and LDPC code (364, 91)

modulation, used for simulation. So, we believe that this gap will be reduced if FBL bound for BPSK modulation is used.

Now let us consider simulation results for  $T = 4$  (Fig. 3). For this case we obtain another PEXIT-optimized rate-1/4 LDPC code (364, 91) and compare FER performance of same LDPC code from [14] and Polyanskiy’s FBL bound for 4 users.



**Fig. 3.** Simulation results for  $T = 4$  and LDPC code (364, 91)

As we can see in Fig. 3 proposed PEXIT-optimized LDPC code construction outperforms LDPC code construction from [14] by more than 3 dB. And again

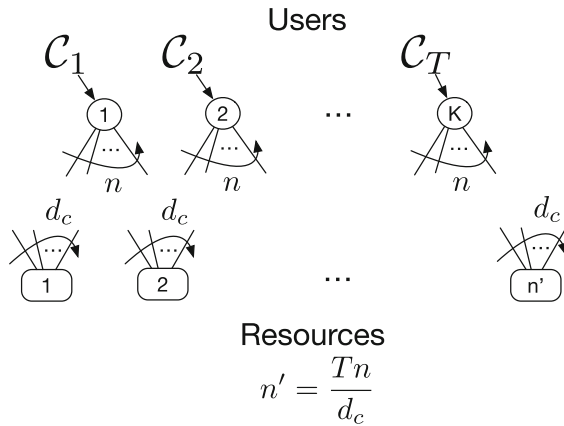
the gap between Polyanskiy’s FBL bound and PEXIT-optimized LDPC code is a little bit less than 3 dB.

### 5 Sparse Spreading of LDPC Codes

In this section we answer a very natural question: how to increase the order of collision, that can be decoded in a slot. E.g. consider the case from the previous section. Let the slot length  $n' = 364$ . We want to increase  $T$  up to 8. Here we face with two problems:

- The performance of LDPC joint decoder rapidly becomes bad with grows of  $T$ . We were not able to find (364, 91) LDPC codes, that work well for  $T = 8$ .
- The number of computations necessary to obtain the outgoing messages from the functional node grows exponentially with the number of users  $T$ .

We address both these problems in a scheme, which is proposed below (see Fig. 4). The idea is to use sparse spreading signatures [3] for LDPC codes, such that the degree of functional node is reduced from  $T$  to  $d_c$ . The slot length is now  $n'$ ,  $n' \neq n$ .



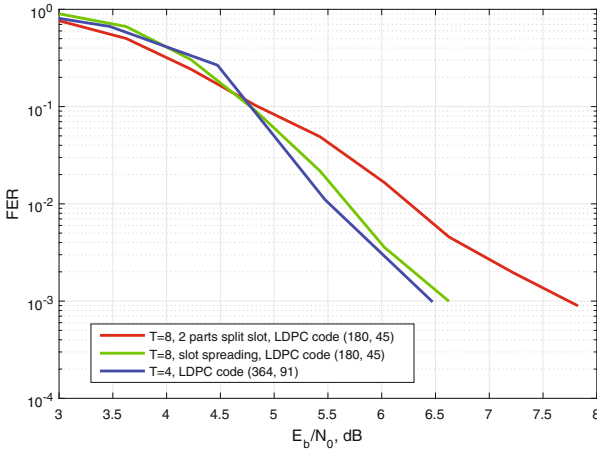
**Fig. 4.** Sparse spreading of LDPC codes

In Fig. 5 we present the simulation results. As we were not able to find (364, 91) LDPC codes, that work well for  $T = 8$  we consider 2 times shorter LDPC codes and compare 2 strategies:

- split the slot into 2 parts and send 4 users in each part;
- use sparse spreading;

We see, that our approach is much better and works practically the same in comparison to the case of 2 times longer LDPC codes and 2 times smaller number of users (see the previous section).





**Fig. 5.** Simulation results for spreading

## 6 Conclusion

We generalized the protograph extrinsic information transfer charts (EXIT) to optimize the protograph of LDPC code for GMAC. The simulation results, obtained at the end of the paper, were analyzed and compared with obtained theoretical bounds and thresholds. Obtained simulation results shows that proposed LDPC code constructions have better performance under joint decoding algorithm over Gaussian MAC than LDPC codes considered by A. Vem et al. in 2017, that leads to the better performance of overall system.

**Acknowledgment.** We want to thank Y. Polyanskiy for fruitful discussions.

## References

- ten Brink, S.: Convergence behavior of iteratively decoded parallel concatenated codes. *IEEE Trans. Commun.* **49**(10), 1727–1737 (2001). <https://doi.org/10.1109/26.957394>
- Gallager, R.G.: Low-density parity-check codes (1963)
- Hoshyar, R., Wathan, F.P., Tafazolli, R.: Novel low-density signature for synchronous CDMA systems over AWGN channel. *IEEE Trans. Signal Process.* **56**(4), 1616–1626 (2008). <https://doi.org/10.1109/TSP.2007.909320>
- Liva, G.: Graph-based analysis and optimization of contention resolution diversity slotted aloha. *IEEE Trans. Commun.* **59**(2), 477–487 (2011). <https://doi.org/10.1109/TCOMM.2010.120710.100054>
- Liva, G., Chiani, M.: Protograph LDPC codes design based on EXIT analysis. In: *IEEE GLOBECOM 2007 - IEEE Global Telecommunications Conference*, pp. 3250–3254, November 2007. <https://doi.org/10.1109/GLOCOM.2007.616>

6. Narayanan, K.R., Pfister, H.D.: Iterative collision resolution for slotted ALOHA: an optimal uncoordinated transmission policy. In: 2012 7th International Symposium on Turbo Codes and Iterative Information Processing (ISTC), pp. 136–139, August 2012. <https://doi.org/10.1109/ISTC.2012.6325214>
7. Ordentlich, O., Polyanskiy, Y.: Low complexity schemes for the random access Gaussian channel. In: 2017 IEEE International Symposium on Information Theory (ISIT), pp. 2528–2532, June 2017. <https://doi.org/10.1109/ISIT.2017.8006985>
8. Polyanskiy, Y.: A perspective on massive random-access. In: 2017 IEEE International Symposium on Information Theory (ISIT), pp. 2523–2527. IEEE (2017)
9. Richardson, T., Urbanke, R.: Modern Coding Theory. Cambridge University Press, New York (2008)
10. Rimoldi, B., Urbanke, R.: A rate-splitting approach to the Gaussian multiple-access channel. *IEEE Trans. Inf. Theory* **42**(2), 364–375 (1996)
11. Shahid, I., Yahampath, P.: Distributed joint source-channel code design for GMAC using irregular LDPC codes. *EURASIP J. Wirel. Commun. Netw.* **2014**(1), 3 (2014). <https://doi.org/10.1186/1687-1499-2014-3>
12. Tanner, R.: A recursive approach to low complexity codes. *IEEE Trans. Inf. Theory* **27**(5), 533–547 (1981)
13. Thorpe, J.: Low-density parity-check (LDPC) codes constructed from protographs. *Interplanet. Netw. Prog. Rep.* **154**, 1–7 (2003)
14. Vem, A., Narayanan, K.R., Cheng, J., Chamberland, J.F.: A user-independent serial interference cancellation based coding scheme for the unsourced random access Gaussian channel. In: Proceedings of the IEEE Information Theory Workshop (ITW), pp. 1–5 (2017)