Frame Length Reduction for Massive-Machine Communications

Carole Al Bechlawi, Frederic Guilloud
Institut Mines-Telecom / Telecom Bretagne, Brest, France
Email: {carole.albechlawi, frederic.guilloud}@telecom-bretagne.eu

***Abstract*—Machine type communications (MTC) require short-length frames to improve the latency and to achieve high reliability when combined to advanced automatic request (ARQ) mechanisms. The use of short frames has a direct impact on the physical layer, especially on the forward-error correction code (FEC) performance. In this article, lattice based codes are used to achieve an efficient joint decoding of the code and the modulation. The linear group structure of lattices makes it possible to design a decoder based on a sphere decoder. To improve the performance for lower spectral efficiencies, the decoder design takes into account the inevitable lattice shaping processing of the transmitter. With a small number of dimensions, the frame lengths are very short and well suited to MTC. A comparison in frame error rate is performed between the proposed lattice code and an LTE inspired baseline, designed upon the LTE turbo code. Simulation results show that for equivalent spectral efficiencies, a gain in frame length is obtained. This frame length reduction can be employed for increasing the number of users in a machine type communication system.**

Key words: Massive Type Communication, Lattice decoding, lattice code decoding, shaping, shaping region, sphere decoder, Frame error rate, short block length.

I. Introduction

The forecast of increasing demand for connectivity in the future 5G has been identified to be partly based on the deployment of massive sets of machine type communication (MTC) devices [1]. MTC are characterized by very specific requirements such as low-latency and high reliability transmissions. The use of short packets is seen as a key enabler to fulfil these requirements [2]. Moreover, short packets offer the possibility to increase the number of users either in an orthogonal multiple access scheme or in a non-orthogonal multiple access scheme [3, 4].

The use of short packets has a direct impact on the physical layer, especially on the forward-error correction code (FEC) performance. Modern FEC schemes such as turbo codes or low-density parity check (LDPC) codes can achieve high coding gains close to the theoretical limits provided that the code length is long enough. Using long FECs with short packet transmissions requires spreading the FEC codeword on a possibly high number of packets. Such an approach preserves the coding gain but dramatically affects in turn the latency. The alternative is to match the code length to the packet size and the modulation order. It is then desirable to implement the best possible FEC scheme for a given small code length. If the design of capacity-approaching (or even achieving) FEC codes is now well understood in the asymptotic (long blocklength) regime, as demonstrated by the discovery of Polar codes [5] and spatially-coupled codes [6], recent theoretical work by Polyanskiy [7] has shown that there is a severe back-off from capacity at short blocklengths. To date and to the best of the authors' knwoledge, the design of optimal finite-length codes that match the bounds of [7] with an affordable decoding complexity remains an open issue.

LTE advanced channel coding scheme [8] is based on a turbo code for which the smallest interleaver length is set to 40 uncoded bits. If we neglect the tail bits and if we assume a spectral efficiency of 2 uncoded bits per complex symbol, a frame length of 20 complex symbols would be required.

In this paper, we propose to shorten by a factor of 5 the frame length, and thus to increase by a factor of 5 the number of users of the MTC scheme, the baseline being illustrated by the shortest frames in LTE. To this aim, we propose to employ a joint code and modulation scheme based on a lattice code. Simulation results will be illustrated by a lattice having 8 real dimensions, equivalent to 4 complex symbols, whatever the spectral efficiency.

The outline of the paper is the following: in Section II, we introduce some basic notations and definitions related to lattices and lattice codes, and we describe the transmission model. In Section III, we present a modified decoding strategy to improve the error rate in the low dimension and low spectral efficiency regime. Section IV is devoted to simulation results. The paper is concluded in Section V. Throughout the paper, matrices are set in boldface capital letters, columns vectors in boldface lowercase letters, and scalars in normal text lowercase letters. The superscript ^{T} stands for transpose.

II. Transmission scheme

A. Lattices and lattice codes

In the last decade, lattice theory has gained a renewed interest by offering a theoretical framework to overcome several modern digital communications issues. Erez and Zamir showed in [9] that lattice decoding (as opposed to the more complex lattice code decoding) can achieve the full additive white Gaussian noise (AWGN) channel capacity, that is $\frac{1}{2}\log_2(1+\text{SNR})$. While the latter demonstration requires two lattices for encoding, the capacity can be also achieved with only one lattice as proved in [10].

An $n-$dimensional lattice Λ in the real field $\mathbb{R}^m$, $n \leq m$, is an infinite discrete subset of $\mathbb{R}^m$, defined as the set of all the integer linear combinations of n linearly independent vectors in $\mathbb{R}^m$. These vectors form the columns of a lattice generator

matrix $\mathbf{G}$. A lattice point $\mathbf{x} = (x_0, x_1, \cdots x_{m-1})^T \in \Lambda$ is therefore a column vector, and the lattice is then:

$$\Lambda = \{\mathbf{x} \in \mathbb{R}^m | \mathbf{x} = \mathbf{G}\mathbf{b},\, \mathbf{b} \in \mathbb{Z}^n\} \tag{1}$$

In the remainder of the paper, we take $m = n$, and $\mathbf{G}$ becomes a full rank matrix.

In practice, the amount of information to be sent per channel use is finite, and so should be the number of lattice points considered for transmission. Therefore, the use of a lattice for digital transmission requires selecting a finite number of the lattice points to create what is known as a *lattice code* or a *lattice constellation*.

A lattice code $\Lambda_{\mathcal{B}}$ can be defined as the intersection between an $n-$dimensional lattice Λ and a compact bounding region of $\mathbb{R}^n$ called the shaping region and denoted by $\mathcal{B}$. The codewords are all the lattice points that belong to the shaping region $\mathcal{B}$ and translated by $-\mathbf{t}$:

$$\Lambda_{\mathcal{B}} = \{\mathbf{x} - \mathbf{t} \,|\, \mathbf{x} \in (\Lambda \cap \mathcal{B})\}, \tag{2}$$

where the translation vector $\mathbf{t}$ shall be equal to the codewords expectation: $\mathbf{t} = \mathbb{E}(\mathbf{x}), \mathbf{x} \in \Lambda_{\mathcal{B}}$. Hence, the expectation of the sent symbols is zero. Note that the higher the number of lattice points in $\mathcal{B}$, the higher the spectral efficiency η: $\eta = \log_2(|\Lambda_{\mathcal{B}}|)/n$ where $|\Lambda_{\mathcal{B}}|$ is the cardinal of $\Lambda_{\mathcal{B}}$.

Depending on the shape of $\mathcal{B}$, the error rate performance may vary. The gain related exclusively to the shape of $\mathcal{B}$ is called the shaping gain and is upper bounded by 1.53 dB [11], which is reached when $\mathcal{B}$ is a hypersphere. Unfortunately, hypersphere shaping is too complex to implement. If $\mathcal{B}$ is a hypercube, the shaping gain is 0 dB. Note that the shaping gain can even be negative, if the shape of $\mathcal{B}$ is worse than a hypercube. A survey of popular shaping techniques was proposed by Sommer, Feder and Shalvi in [12] and applied to low-density lattice codes to improve the shaping gain. Among them, the best shaping gain is achieved by the so-called nested shaping.

B. Transmission model and notations

The transmission scheme on the AWGN channel is described in Figure 1. The information to be sent is represented by a vector of integers $\mathbf{b}$. If the number of information bit per dimension is a power of 2, the mapping between these bits and the information integers is straightforward. Otherwise, it is possible to apply a non-uniform mapping like for example in [13, 14]. We assume hereafter that the information integers of vector $\mathbf{b}$ are uniformly drawn from the interval $(0, \cdots, L-1)$, L being the constellation size for each dimension. The spectral efficiency is thus defined as $\eta = \log_2(L)$ bits per real dimension.

The linear encoding of this information vector $\mathbf{x} = \mathbf{G}\mathbf{b}$ might not fall within the shaping region $\mathcal{B}$ associated to the lattice code $\Lambda_{\mathcal{B}}$. So the shaping operation consist in finding another vector of integers denoted $\mathbf{b}_s$ such that its linear encoding $\mathbf{x}_s = \mathbf{G}\mathbf{b}_s$ falls inside the shaping region. Popular shaping techniques described in [12] assume that

$$\mathbf{b}_s = \mathbf{b} - L\mathbf{k} \tag{3}$$

Thus, the shaping technique consists in searching for the vector $\mathbf{k} = (k_0, \cdots, k_{n-1})^T$ such that $\mathbf{x}_s$ lies inside the shaping region.

Since nested shaping achieves a quasi-optimal shaping with an affordable complexity, it will be used as an illustrative example throughout the remainder of the paper. Nested shaping requires two nested lattices Λ and Λ_S, which means that Λ_S is a sublattice of Λ ($\Lambda_S \subset \Lambda$):

$$\forall \mathbf{x} \in \Lambda_S, \mathbf{x} \in \Lambda$$

The Voronoi region of a lattice point $\mathbf{x}$ is the set of points in $\mathbb{R}^n$ that are closer to $\mathbf{x}$ than to any other lattice point in Λ. The shaping domain is the Voronoi region of the origin (nul vector) in Lattice Λ_S. A simple choice for Λ_S is to take a scaled version of the lattice Λ: $\Lambda_S = L\Lambda$, with generator matrix $L\mathbf{G}$.

Applying linear encoding to Equation (3) yields to:

$$\mathbf{x}_s = \mathbf{x} - L\mathbf{G}\mathbf{k} \tag{4}$$

Minimizing the amplitude of the transmitted symbol $\mathbf{x}_s$ is equivalent to minimizing the amplitude of $(\mathbf{x} - L\mathbf{G}\mathbf{k})$, which can be done by applying a sphere decoder on $\mathbf{x}$ in the lattice having the generator matrix $L\mathbf{G}$, i.e. Lattice Λ_S.

After linear encoding, the lattice point $\mathbf{x}_s$ is transmitted over the AWGN channel, and the receiver observes the vector $\mathbf{y} = \mathbf{x}_s + \mathbf{w}$, where $\mathbf{w}$ is a vector of n independent samples drawn from a centered Gaussian distribution with variance σ^2.

The signal-to-noise ratio (SNR) is defined as

$$\text{SNR} = \mathbb{E}\left(\frac{(\mathbf{x}_s - \mathbf{t})^T(\mathbf{x}_s - \mathbf{t})}{n\sigma^2}\right) \tag{5}$$

Note that the shifting operation is not mentionned in Figure 1 since it does not affect the error rate performance on a AWGN channel, providing that the SNR is calculated as in (5).

At the receiver side, the decoding operation consists in searching over the lattice Λ to find the closest point (in the sense of the Euclidean distance) to the received point $\mathbf{y}$. Then, based on this estimation, the initial vector of information integers is estimated by applying a modulo$-L$ operation on each coordinate of $\hat{\mathbf{b}}_s$:

$$\hat{\mathbf{b}} = \hat{\mathbf{b}}_s \mod L \tag{6}$$

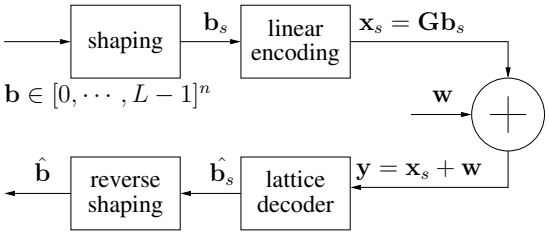

Figure 1: Transmission Model

III. Decoder design

At the receiver side, the signal can be decoded using several strategies. A maximum likelihood (ML) strategy on the Gaussian channel is the search for the closest lattice point inside the shaping region: this strategy is referred to as a lattice code decoder, and it is opposed to the lattice decoder. A lattice decoder is simply a decoder in the infinite lattice: there is no checking whether the estimated transmitted symbol lies inside the shaping region or not. When restricted to a lattice decoder, the capacity reached is then reduced to $\frac{1}{2}\log_2(\text{SNR})$ [15]. In [9], Erez and Zamir proposed a different lattice decoding method that uses a linear minimum mean-square error (MMSE) scaling along with dithering, lattice decoding and modulo operation between lattices; they showed that the SNR can be enhanced by "one" to achieve the full Gaussian capacity $\frac{1}{2}\log_2(1+\text{SNR})$.

In practice, the linear lattice structure makes it possible to search for the closest lattice point by efficiently implementing the popular sphere decoder [16, 17]. However, basic sphere decoding itself is not ML decoding since the closest lattice point found may lie outside $\mathcal{B}$. This is not an issue for high dimensions and high spectral efficiencies, but a loss in error rate is observed otherwise.

To get improved performance in the low dimension and low spectral efficiency case, sphere decoding must be performed inside the region $\mathcal{B}$. Without implementing a specific shaping technique (e.g. hypersphere shaping, nested shaping, hypercube shaping) the minimum and maximum values of the lattice point coordinates inside $\mathcal{B}$ are known: $[0, \cdots, L-1]$. They can be incorporated inside the sphere decoder processing to check for the boundaries of $\mathcal{B}$, as proposed in [18]. However, when a specific shaping technique is performed, the coordinates are modified in such a way that it becomes too complex to incorporate inside the sphere decoder processing the restriction of the solution to lie in the shaping region .

In this Section, we propose a decoding algorithm based on re-shaping, which does not depend on the shaping technique, provided it is known at the receiver side. However, as exposed in Section II, nested shaping is assumed at the transmitter side to illustrate the description of the decoder.

Decoding at the receiver side consists first in finding the closest lattice point to $\mathbf{y}$, then applying a modulo$-L$ operation to its integer coordinates to have an estimation of the initial integer information vector $\mathbf{b}$. As mentioned before, a lattice code decoder gives better performance than a naive lattice decoder but it is complex to implement within a sphere decoding algorithm, as the boundaries of the code are unknown to the receiver when a shaping processing has been implemented at the transmitter side. We suggest then to re-shape the result of the lattice sphere decoder in order to check whether it belongs to the lattice code or not. Moreover, replacing the sphere decoder by a list sphere decoder (LSD) [19] with a list of size l_s enables the re-shaping of a maximum of l_s points, increasing the chance to get a decoder output within the shaping region.

The proposed decoding algorithm is resumed in the following pseudo-code:

Algorithm 1 Search for the closest lattice point to $\mathbf{y}$ inside the Voronoi region of Λ_S

Input: $\mathbf{y}, \mathbf{G}, l_s, L$
Output: $\hat{b}$
1: $\left\{\hat{\mathbf{b}}_s^{(1)}, \cdots, \hat{\mathbf{b}}_s^{(l_s)}\right\} = \text{LSD}(\mathbf{y}, \mathbf{G}, l_s)$
2: **for** $i = 1 : l_s$ **do**
3: $\quad \hat{\mathbf{b}}^{(i)} = \hat{\mathbf{b}}_s^{(i)} \mod L$
4: $\quad$ **if** $\left(\text{LSD}(\hat{\mathbf{b}}^{(i)}, L\mathbf{G}, 1) == \hat{\mathbf{b}}_s^{(i)}\right)$ **then**
5: $\quad\quad$ **return** $\hat{\mathbf{b}} = \hat{\mathbf{b}}^{(i)}$
6: $\quad$ **end if**
7: **end for**

The inputs of the proposed algorithm are the received observation $\mathbf{y}$, the generator matrix $\mathbf{G}$ of the lattice Λ, the list size parameter l_s and the constellation size L. The notation $\text{LSD}(\mathbf{a}, \mathbf{B}, c)$ denotes the processing of the list sphere decoder on the observation $\mathbf{a}$, in the lattice described by the generator matrix $\mathbf{B}$ and which returns the c coordinate vectors of the lattice points which are the closest to $\mathbf{a}$, sorted in the ascending order. On line 1, the list sphere decoder $\text{LSD}(\mathbf{y}, \mathbf{G}, l_s)$ returns the list of coordinates of the l_s closest points to $\mathbf{y}$ in the lattice Λ. We begin with the first point in the list, and obtain its initial coordinate estimate through the modulo-L operation on line 3. Then, reshaping the result consists in applying a sphere decoding operation in the shaping lattice Λ_S (line 4). If the result is equal to the coordinates output by the LSD, it means that the candidate belongs to the shaping region. In this case, there is no need to proceed further. If the equality in line 4 is not satisfied, it means that the decoded point lies outside $\mathcal{B}$ and so the next point in the list has to be processed. Of course, the higher the list size l_s, the higher the chance to find the closest point to the observation inside the shaping region, and thus the closest the ML approximation.

IV. Simulation results

Simulations have been performed using the Gosset lattice, which is the densest lattice in $n = 8$ real dimensions and is denoted by $E8$ [20]. Using $E8$, the $n = 8$ real coordinates are equivalent to $n/2 = 4$ complex symbols. Thus the frame length will hereafter be measured as the number of complex symbols to be sent, and will be denoted F using an *ad hoc* subscript. The obtained frame length using Lattice $E8$ is then $F_{E8} = n/2 = 4$ complex symbols.

In order to evaluate the performance of Lattice $E8$ in the Gaussian channel, a baseline is designed using the LTE turbo code [8]. The minimum interleaver size has a length of $K = 40$ uncoded bits. The frame length depends on the interleaver length, the code rate and the constellation size. Note that the 12 tail bits will be neglected in the frame length, resulting in achieving close spectral efficiencies in comparison to $E8$, as well as upper bounding the baseline performance. Note also that a puncturing pattern will be employed as proposed in [21] instead of the rate matching process defined in [8]. Since $K = 40$ uncoded bits is fixed, the frame length decreases when the spectral efficiency increases. The parameters used for the LTE baseline have been chosen to achieve comparable spectral

efficiencies in bits per real dimension, and are described in Table I.

Table I: Code and modulation parameters used for lattice coding and the LTE baseline

$E8$			LTE baseline				
L	F_{E8}	η	K	R	M-QAM	F_{LTE}	η
2	4	1	40	$1/2$	$16-$QAM	20	1
3	4	1.585	40	$1/2$	$64-$QAM	13.3	1.5
4	4	2	40	$2/3$	$64-$QAM	10	2
5	4	2.32	40	$4/5$	$64-$QAM	8.3	2.4

Simulation results of the Frame Error Rate (FER) are plotted in Figure 2 as a function of Eb/N_0 expressed in dB. The lattice code based performances are illustrated by dotted curves, and the baseline performances are illustrated by solid lines.

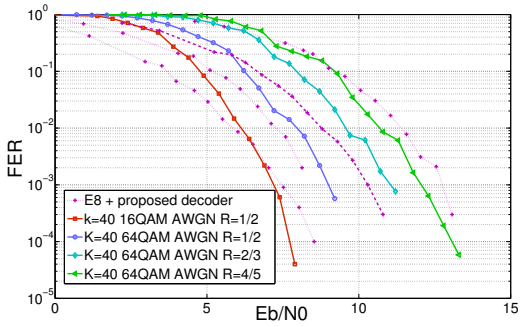

Figure 2: Frame error rate comparison between E8 and short frame LTE turbo code.

The performance curves on Figure 2 are observed at a moderate target FER, like e.g. 10^{-2}, since an automatic request scheme is often implemented when transmitting short frames. For such FER, the SNR between the performance curves of comparable spectral efficiencies are also comparable. Since the frame length are not the same, a possible gain in the number of users can be achieved, proportional to the frame length reduction, ranging from a factor of 5 at $\eta = 1$ bit/dimension downto a factor of 2 at $\eta = 2.4$ bit/dimension.

V. Conclusion

In this article, lattice based codes are used to achieve an efficient joint decoding of the code and the modulation. Indeed, the linear group structure of lattices makes it possible to implement a sphere decoder where the complexity is reasonable if the number of dimensions is small enough. An improvement of the decoder is proposed: the decoder is built upon a lattice sphere decoder which takes into account the inevitable shaping processing of the transmitter. With a small number of dimensions, the frame lengths are very short and well suited to machine type communications (MTC): short frames enable short latencies and high reliability when combined to automatic request (ARQ) mechanisms. Simulations to estimate the frame error rate have been performed. The results have been compared to the frame error rates of an LTE inspired baseline. The baseline is designed upon the LTE turbo code which is punctured with a classical puncturing pattern. For equivalent spectral efficiencies, a gain in frame length is obtained. This frame length reduction can be employed for e.g. increasing the number of users in a machine type communication system.

VI. Acknowledgements

Part of this work has been performed in the framework of the FP7 project ICT-317669 METIS, which is partly funded by the European Union. The authors would like to acknowledge the contributions of their colleagues in METIS, although the views expressed are those of the authors and do not necessarily represent the project.

References

[1] "Initial report on horizontal topics, first results and 5G system concept." Deliverable D6.2 of the METIS project, April 2014. available online at https://www.metis2020.com.

[2] "3GPP TS 22.368 service requirements for machine-type communications, V13.0.0," June 2014.

[3] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted aloha," *IEEE Trans. Commun*, vol. 59, pp. 477–487, Feb. 2011.

[4] E. Paolini, C. Stefanovic, G. Liva, and P. Popovski, "Coded random access: How coding theory helps to build random access protocols," *Computing Research Repository (CoRR)*, vol. abs/1405.4127, 2014.

[5] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, pp. 3051–3073, July 2009.

[6] D. J. Costello, L. Dolecek, T. Fuja, J. Kliewer, D. Mitchell, and R. Smarandache, "Spatially coupled sparse codes on graphs: theory and practice," *Commun. Magazine, IEEE*, vol. 52, pp. 168–176, July 2014.

[7] H. V. P. Y. Polyanskiy and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, pp. 2307–2359, May 2010.

[8] "3GPP TS 36.212 multiplexing and channel coding, V12.2.0," Sept. 2014.

[9] U. Erez and R. Zamir, "Achieving 1/2 Log (1+SNR) on the AWGN Channel with Lattice Encoding and Decoding," *IEEE Trans. Inf. Theor.*, vol. 50, no. 10, pp. 2293–2314, 2004.

[10] C. Ling and J.-C. Belfiore, "Achieving the AWGN channel capacity with lattice Gaussian coding," in *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, pp. 1416–1420, July 2013.

[11] J. Forney, G.D., "Trellis shaping.," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 281–300, 1992.

[12] N. Sommer, M. Feder, and O. Shalvi, "Shaping methods for low-density lattice codes," in *Information Theory Workshop, 2009. ITW 2009. IEEE*, pp. 238–242, 2009.

[13] F. Kschischang and S. Pasupathy, "Optimal nonuniform signaling for gaussian channels," *IEEE Trans. Inf. Theory*, vol. 39, pp. 913–929, May 1993.

[14] N. Palgy and R. Zamir, "Dithered probabilistic shaping," in *Electrical Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of*, pp. 1–5, Nov 2012.

[15] G. Poltyrev, "On Coding Without Restrictions for the AWGN Channel," *IEEE Trans. Inf. Theory*, vol. 40, no. 2, pp. 409–417, 1994.

[16] E. Viterbo and J. Boutros, "A Universal Lattice Code Decoder for Fading Channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1639–1642, 1999.

[17] B. Hassibi and H. Vikalo, "On the Sphere-decoding Algorithm I. Expected Complexity," *Trans. Sig. Proc.*, vol. 53, no. 8, pp. 2806–2818, 2005.

[18] C. Lamy, *Communication à grande efficacité spectrale sur le canal à évanouissements*. PhD thesis, Telecom Paris, 2000.

[19] B. Hochwald and S. Ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, pp. 389–399, March 2003.

[20] J. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*. Springer New York, 2010.

[21] S. Lembo, K. Ruttik, and O. Tirkkonen, "Modeling BLER performance of punctured turbo codes," in *12th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 7-10 Sept. 2009.