Efficient Decoding Algorithms for the Compute-and-Forward Strategy

Asma Mejri and Ghaya Rekaya-Ben Othman, *Member, IEEE*

***Abstract*—We address in this paper decoding aspects of the Compute-and-Forward (CF) physical-layer network coding strategy. Under the CF framework, encoders use a special class of nested lattice codes and decoders are based on suboptimal minimum distance decoding of unknown performance gap with respect to optimal decoders. In this work, we develop and assess the performance of novel decoding algorithms for CF operating in the multiple access channel. Starting with the Gaussian channel, we investigate the *maximum a posteriori* (MAP) decoder. We derive a novel MAP decoding metric and develop practical decoding algorithms shown numerically to outperform the original one. For the fading channel, we analyze the ML decoder for integer-valued lattices and develop a novel Diophantine approximation-based near-ML decoding algorithm shown numerically to outperform the original CF decoder in the 1-D case using $\mathbb{Z}$ lattices.**

***Index Terms*—Physical-layer network coding, compute-and-forward, lattice decoding, maximum *a posteriori* decoding.**

I. Introduction

LAST few years have witnessed the emergence of a very promising linear physical-layer network coding protocol termed *Compute-and-Forward*. Introduced by Nazer and Gastpar in [1], this scheme takes advantage of the multiple access interference to achieve higher transmission rates. This new framework is applicable to any network configuration accomodating source nodes, relays and destinations that communicate through linear additive white Gaussian noise channels. The role of a relay node observing the output of a multiple access channel is to decode a *linear integer* combination of source codewords. Given enough linear equations, the end destination in the network can ideally recover the original source messages with high transmission rates thanks to the potential properties of nested lattice codes.

The original decoder for CF consists of a scaling operation followed by minimum distance decoding. Under this scheme, a union bound estimate of the error probability at the relays was derived in [2], and we have addressed in [3] and [4] the end-to-end error performance evaluation in the multi-source relay channel and the two-way relay channel respectively. Later on, the Maximum Likelihood (ML) decoder was investigated in [5], [6]. An algebraic extension of CF using lattice partitions related to finitely generated modules over principal ideal domains was proposed by Feng *et al.* in [2] assuming also minimum distance decoding. These works focus on the information theoretic performance of CF and study the rate achievability considering high dimensional lattices. What is missing up to now is to understand the error performance of this strategy in practical settings (finite lattice dimensions and low-complexity encoding schemes) as well as its gap to optimal decoders. We try in this work to study these issues by investigating optimal decoding criteria for CF in the basic multiple access channel studied in part in [7]. After reviewing the original CF encoding and decoding schemes in Section II, our contributions come into light as follows: in Section III we focus on the Gaussian channel case. We analyze the MAP decoder for CF in the Gaussian channel using real-valued lattices. Moreover, we derive a novel MAP decoding metric based on which we develop novel efficient decoding algorithms shown numerically to outperform the conventional CF decoder. In Section IV, we investigate efficient ML decoding for the fading channel considering integer-valued lattices. We first analyze the general multi-dimensional case. Then, by analyzing the one-dimensional case, we develop a novel near-ML decoder based on Diophantine approximation and show by numerical results its gain over the original CF decoder for $\mathbb{Z}$-lattices. Generalization of this decoder to the multidimensional case as well as its numerical analysis are left for future works due to its complexity. Finally, main results of this work are summarized in a concluding section.

II. Compute-and-Forward in Basic MAC: Original Work

A. Preliminaries on Lattice Coding

An n-dimensional ***lattice*** Λ is a discrete group of rank p, $p \leq n$ of the Euclidean space $\mathbb{R}^n$. It is the set spanned by the p linearly independent vectors $\mathbf{v}_1, \ldots, \mathbf{v}_p$ of $\mathbf{R}^n$. Explicitly, Λ is given by the set of integer linear combinations as $\Lambda = \{\mathbf{x} = \sum_{i=1}^{p} a_i \mathbf{v}_i,\ a_i \in \mathbb{Z}\}$. p is called the lattice dimension and the vectors $\mathbf{v}_1, \ldots, \mathbf{v}_p$ represent a non-unique basis of the lattice Λ. Any vector $\mathbf{x} \in \Lambda$ can be written in the form: $\mathbf{x} = \mathbf{Ms}, \mathbf{s} \in \mathbb{Z}^n$ where $\mathbf{M}$ is called a ***generator matrix*** of the lattice. The main characteristic of Λ is *linearity*, i.e. for any $a, b \in \mathbb{Z}$ and $\mathbf{x}, \mathbf{y} \in \Lambda$, $a\mathbf{x} + b\mathbf{y} \in \Lambda$.

A ***lattice quantizer*** Q_Λ satisfies for $\mathbf{x} \in \mathbb{R}^n$, $Q_\Lambda(\mathbf{x}) = \arg\min_{\lambda \in \Lambda} \|\mathbf{x} - \lambda\|$. The set of points that quantize to a given lattice

Manuscript received April 15, 2014; revised September 5, 2014, January 20, 2015, and May 9, 2015; accepted May 25, 2015. Date of publication June 2, 2015; date of current version July 13, 2015. The material in this paper was presented in part at the IEEE International Conference on Communications and Networking (ComNet) 2012 Hammamet, Tunisia, March 2012, and at the IEEE International Conference on Telecommunications (ICT) 2014, Lisbon, Portugal, April 2014. The associate editor coordinating the review of this paper and approving it for publication was A. Ramamoorthy.

The authors are with the Department of Communications and Electronics, Telecom-ParisTech, 75013 Paris, France (e-mail: amejri@telecom-paristech.fr; rekaya@telecom-paristech.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCOMM.2015.2440361

point is called the ***Voronoi Region*** corresponding to the given lattice point. The ***fundamental Voronoi Region*** $\mathcal{V}(\Lambda)$ of a lattice Λ corresponds to the voronoi region of the zero vector. The modulo operation returns the quantization error with respect to Λ such that for $\mathbf{x} \in \mathbb{R}^n$: $[\mathbf{x}] \bmod \Lambda = \mathbf{x} - Q_\Lambda(\mathbf{x})$.

A ***nested lattice code*** $\mathcal{C}$ is the set of all points of a lattice Λ_F (termed the *fine* lattice) that fall within the fundamental Voronoi region of a lattice Λ_C (termed the *coarse* lattice) as: $\mathcal{C} = \{\Lambda_\text{F} \cap \mathcal{V}(\Lambda_\text{C})\} = \{\lambda = [\lambda_\text{F}] \bmod \Lambda_\text{C}, \lambda_\text{F} \in \Lambda_\text{F}\}$.

B. System Model and Assumptions

We consider the real-valued fading Multiple Access Channel (MAC) composed of N sources S_i, $i = 1, \ldots, N$ and a common receiver. Extension of our results to the complex-valued channel follows by considering the real and imaginary parts of the channel outputs separately. Source S_i delivers a length-k finite field message $\mathbf{w}_i \in \mathbb{F}_p^k$ drawn independently and uniformly. Encoders $\mathcal{E}$ at the sources implement the same mapping ϕ to map the messages $\mathbf{w}_i$ onto codewords $\mathbf{x}_i$ from the same nested lattice code $\mathcal{C}$ designed using a fine lattice Λ_F and a coarse lattice Λ_C. Encoded vectors satisfy a symmetric power constraint given by:

$$\frac{1}{n}\mathbb{E}\left(\|\mathbf{x}_i\|\right) \leq P, \qquad P > 0. \tag{1}$$

Λ_F corresponds to the coding lattice and Λ_C acts to satisfy the power constraint P. The codewords are assumed to be independent and uniformly distributed over $\mathcal{C}$. The message rate is equal to $r = \frac{k}{n}\log p$ and is the same for all sources. After encoding their messages, the source nodes transmit their codewords simultaneously across the channel. The received vector is written as:

$$\mathbf{y} = \sum_{i=1}^{N} h_i \mathbf{x}_i + \mathbf{z} \tag{2}$$

where $h_i \in \mathbb{R}$ denotes the fading coefficient from source S_i to the receiver and $\mathbf{z} \in \mathbb{R}^n$ denotes the additive white Gaussian noise of zero-mean and variance $\sigma^2\mathbf{I}_n$. Let $\mathbf{h} = [h_1, \ldots, h_N]^t$ denote the channel coefficient vector. In this work we assume fixed channel vector. We assume also that channel state information (CSI) is available only at the receiver and denote by $\rho = \frac{P}{\sigma^2}$ the signal-to-noise ratio (SNR).

C. Decoding Scheme for Compute-and-Forward

The receiver attempts to decode a noiseless integer linear combination in the form:

$$\lambda = \left[\sum_{i=1}^{N} a_i \mathbf{x}_i\right] \bmod \Lambda_\text{C}\ , \qquad a_i \in \mathbb{Z}, i = 1, ..., N \tag{3}$$

where the network code vector $\mathbf{a} = [a_1, \ldots, a_N]^t \in \mathbb{Z}^N$ is chosen by the receiver. The latter is equipped with a decoder $\mathcal{D} : \mathbb{R}^n \to \mathcal{C}$, that recovers an estimate $\hat{\lambda}$ of λ. A decoding error occurs if $\hat{\lambda} \neq \lambda$ and the desired equation with a coefficient vector $\mathbf{a}$ is decoded with an average probability of error ϵ if $\hat{\lambda} \triangleq \mathcal{D}(\mathbf{y})$ and $\Pr\,(\hat{\lambda} \neq \lambda) < \epsilon$. A computation rate $\mathcal{R}(\mathbf{h}, \mathbf{a})$ is said to be achievable if for any $\epsilon > 0$ and n large enough, there exist an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$, such that for any channel vector $\mathbf{h} \in \mathbb{R}^N$ and network code vector $\mathbf{a} \in \mathbb{Z}^N$, the receiver can recover the desired equation with an average probability of error ϵ as long as the source message rate r satisfies: $r < \mathcal{R}(\mathbf{h}, \mathbf{a})$.

The receiver selects a scalar $\alpha \in \mathbb{R}$ and an integer vector $\mathbf{a}$ and performs the following steps:

1) Scale the channel output: $\tilde{\mathbf{y}} = \alpha\mathbf{y} = \sum_{i=1}^{N} a_i\mathbf{x}_i + \sum_{i=1}^{N}(\alpha h_i - a_i)\mathbf{x}_i + \alpha\mathbf{z}$. The resulting effective noise $\mathbf{z}_\text{eq} = \sum_{i=1}^{N}(\alpha h_i - a_i)\mathbf{x}_i + \alpha\mathbf{z}$ is not Gaussian since composed of a quantization error involving the original codewords. At this level, $\mathbf{t} = \sum_{i=1}^{N} a_i\mathbf{x}_i \in \Lambda_\text{F}$.
2) Decode to the nearest point in the fine lattice: $\hat{\mathbf{t}} = Q_{\Lambda_\text{F}}(\tilde{\mathbf{y}})$.
3) Take the modulo operation with respect to the coarse lattice: $\hat{\lambda} = [\hat{\mathbf{t}}] \bmod \Lambda_\text{C}$.

We summarize the results regarding the CF protocol in the following theorems [1].

Theorem II. 1 (Computation rate): For a real-valued MAC with channel vector $\mathbf{h}$, and network code vector $\mathbf{a} \in \mathbb{Z}^N$ the following computation rate R_comp, for $\alpha \in \mathbb{R}$, is achievable:

$$R_\text{comp}(\mathbf{h}, \mathbf{a}) = \frac{1}{2}\log^+\left(\frac{\rho}{\alpha^2 + \rho\|\alpha\mathbf{h} - \mathbf{a}\|^2}\right) \tag{4}$$

where $\log^+(x) = \max\left(\log(x), 0\right)$.

Theorem II.2 (Optimal scaling factor): The computation rate given in Theorem II.1 is only maximized for the MMSE scaling factor α_opt given by: $\alpha_\text{opt} = \frac{\rho\mathbf{h}^t\mathbf{a}}{1+\rho\|\mathbf{h}\|^2}$.

Theorem II. 3 (Optimal network code vector): The optimal network code vector satisfies:

$$\mathbf{a}_\text{opt} = \arg\min_{\substack{\mathbf{a}\in\mathbb{Z}^N \\ \mathbf{a}\neq\mathbf{0}}}\{\mathbf{a}^t\mathbf{G}\mathbf{a}\} \tag{5}$$

where $\mathbf{G} = \mathbf{I}_N - \frac{\rho}{1+\rho\|\mathbf{h}\|^2}\mathbf{h}\mathbf{h}^t$ is definite positive. $\mathbf{a}_\text{opt}$ corresponds to the shortest vector in the lattice $\Lambda_\mathbf{G}$ of Gram matrix $\mathbf{G}$.

The conventional decoding scheme for CF consists of an MMSE scaling operation and minimum distance decoding. The problem is that in presence of the non-Gaussian effective noise $\mathbf{z}_\text{eq}$, minimum distance decoding is not optimal and its performance gap to the optimal decoders is not known especially in practical settings using finite-dimensional lattices. We aim in the following to study optimal decoding criteria and develop practical efficient decoding algorithms. Although we will consider the real-valued channel, our results hold in the complex-valued channel case using the same techniques at the real and imaginary parts of the channel output separately. As a starting point, we consider in the following section the Gaussian channel case in which we consider equal unitary channel gains such that $h_i = 1, \forall\, i = 1, \ldots, N$. The generalization for fading channels follows in Section IV.

III. Efficient Decoders in Gaussian Channels

We are interested within this section in the real-valued Gaussian multiple access channel using real-valued nested lattice coding. By studying this channel model, our objective is twofold: first investigate optimal decoding for CF in this specific scenario, develop practical and efficients decoding algorithms and evaluate their error performance gap to the conventional decoder. Second, give some insights and tools we believe will be usefull to analyze the rate achievability under MAP decoding, an open problem in Information Theory rising particularly in the uplink transmission in the Gaussian two-way welay channel [8], [9].

A. Problem Statement

The channel output is given by: $\mathbf{y} = \sum_{i=1}^{N} \mathbf{x}_i + \mathbf{z}$. The receiver aims to decode the noiseless sum $\lambda = \left[\sum_{i=1}^{N} \mathbf{x}_i\right] \bmod \Lambda_{\mathrm{C}}$.

Let $\mathcal{C}_{\mathrm{s}}$ denote the *sum codebook* which is the set of all sum codewords $\lambda_{\mathrm{s}} = \sum_{i=1}^{N} \mathbf{x}_i$. Given the linear structure of the coding lattice, $\mathcal{C}_{\mathrm{s}}$ is a subset of the fine lattice Λ_{F} restricted to a *sum shaping region* $\mathcal{S}_{\mathrm{s}}$ such that all sum codewords $\mathcal{C}_{\mathrm{s}}$ fall within this region. In addition, given that $\mathcal{C}_{\mathrm{s}}$ is obtained through a superposition of the originally transmitted codewords, its distribution is **no longer uniform**.

Using the conventional CF decoder, the receiver decodes $\lambda_{\mathrm{s}} = \sum_{i=1}^{N} \mathbf{x}_i$ using an MMSE scaling followed by minimum distance decoding to the nearest point in the fine lattice. Using this method, there are three fundamental limitations: *i*) the effective noise $\mathbf{z}_{\mathrm{eq}} = \sum_{i=1}^{N} (1-\alpha)\mathbf{x}_i + \alpha \mathbf{z}$ is not Gaussian, *ii*) the shaping constraint is disregarded given that λ_{s} is decoded in Λ_{F} instead of $\mathcal{C}_{\mathrm{s}}$, and *iii*) the non uniform distribution of the sum codebook $\mathcal{C}_{\mathrm{s}}$ is not taken into account. A main contribution of this work is the analysis in the following of the optimal MAP decoding approach that takes into consideration the above mentioned drawbacks of the conventional CF decoder. To the best of our knowledge, this is the first investigation of the MAP decoder for the CF protocol. We will be interested in decoding $\lambda_{\mathrm{s}} = \sum_{i=1}^{N} \mathbf{x}_i$ given that modulo-lattice operation is done separately and does not impact the decoding error. We will evaluate therefore the error probability at the receiver as:

$$\mathrm{P_e} = \Pr(\hat{\lambda}_s \neq \lambda_s).$$

Under the non-uniform distribution of the sum codebook, the optimal decoder that minimizes the probability of decoding error at the receiver is the MAP decoder given according to the following:

$$\begin{aligned}\hat{\lambda}_{\mathrm{map}} &= \underset{\lambda_{\mathrm{s}} \in \mathcal{C}_{\mathrm{s}}}{\operatorname{argmax}}\, p(\boldsymbol{\lambda}_{\mathrm{s}}|\mathbf{y}) \\ &= \underset{\lambda_{\mathrm{s}} \in \mathcal{C}_{\mathrm{s}}}{\operatorname{argmax}} \left\{ p(\lambda_{\mathrm{s}}) \frac{1}{(\sigma\sqrt{2\pi})^n} \exp\left(-\frac{\|\mathbf{y}-\lambda_{\mathrm{s}}\|^2}{2\sigma^2}\right)\right\} \\ &= \underset{\lambda_{\mathrm{s}} \in \mathcal{C}_{\mathrm{s}}}{\operatorname{argmin}} \left\{ -\ln\left(p(\lambda_{\mathrm{s}})\right) + \frac{\|\mathbf{y}-\lambda_{\mathrm{s}}\|^2}{2\sigma^2}\right\}. \qquad (6)\end{aligned}$$

Notice that the MAP decoder does not involve a scaling step like the conventional decoder keeping the channel noise Gaussian. We aim in the following to develop practical decoding algorithms that allow to reliably find the optimal MAP estimate in this optimization problem. For this purpose, we study first the statistical distribution of the sum codewords. The original codewords are drawn uniformly and independently from the nested lattice code $\mathcal{C}$. They are modeled by uniform random vectors of zero-mean ($\mu_{\mathbf{x}} = 0$) and variance $\sigma_{\mathbf{x}}^2 = \frac{1}{n}\mathbb{E}(\|\mathbf{x}_i\|^2) \leq P$ for $i = 1, \ldots, N$. Consider now the sum codewords $\lambda_{\mathrm{s}} = \sum_{i=1}^{N} \mathbf{x}_i$ obtained through the superposition of the vectors sent by the sources. Given the independence between the original codewords, the sum vectors λ_{s} are random vectors of mean $\mu_{\mathrm{s}} = N\mu_{\mathbf{x}} = 0$ and variance $\sigma_{\mathrm{s}}^2 \mathbf{I}_n = N\sigma_{\mathbf{x}}^2 \mathbf{I}_n$. Since it is hard to exactly specify the true distribution of the sum codewords, we use in the following a heuristic model using lattice Gaussian distributions. This tool arises in several problems in coding theory [10], mathematics [11] and cryptography [12].

Let $f_{\sigma_{\mathrm{s}}}(\mathbf{x})$ denote the Gaussian distribution of variance $\sigma_{\mathrm{s}}^2 \mathbf{I}_n$ centered at the zero vector such that for $\sigma_{\mathrm{s}} > 0$ and all $\mathbf{x} \in \mathbb{R}^n$:

$$f_{\sigma_{\mathrm{s}}}(\mathbf{x}) = \frac{1}{(\sqrt{2\pi}\sigma_{\mathrm{s}})^n} e^{-\frac{\|\mathbf{x}\|^2}{2\sigma_{\mathrm{s}}^2}}$$

Consider also the Λ_{F}-periodic function defined by:

$$f_{\sigma_{\mathrm{s}}}(\Lambda_{\mathrm{F}}) = \sum_{\lambda_{\mathrm{s}} \in \Lambda_{\mathrm{F}}} f_{\sigma_{\mathrm{s}}}(\lambda_{\mathrm{s}}) = \frac{1}{(\sqrt{2\pi}\sigma_{\mathrm{s}})^n} \sum_{\lambda_{\mathrm{s}} \in \Lambda_{\mathrm{F}}} e^{-\frac{\|\lambda_{\mathrm{s}}\|^2}{2\sigma_{\mathrm{s}}^2}}.$$

Then we model the distribution of the sum codewords by discrete Gaussian distributions over Λ_{F} centered at the zero vector according to:

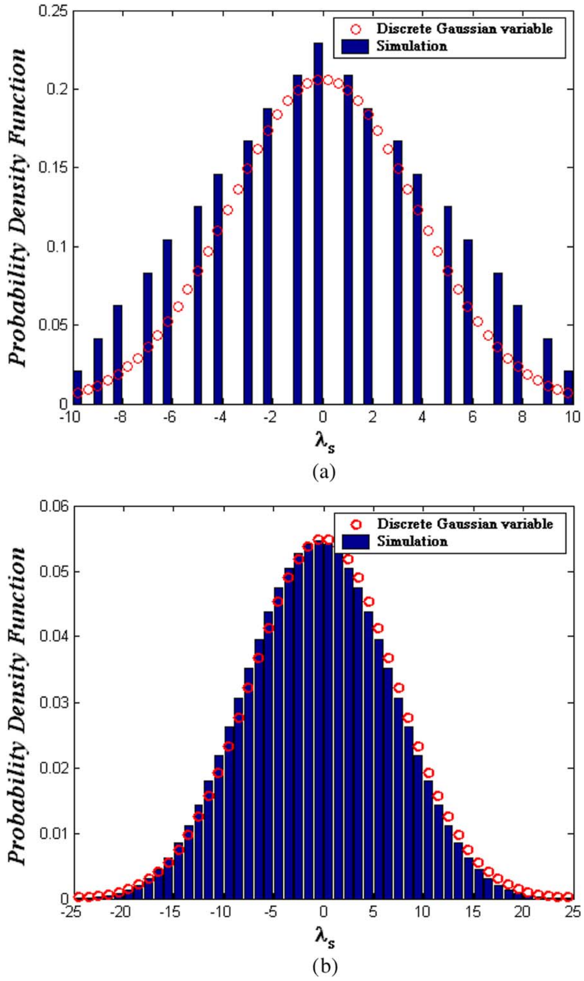
$$p(\lambda_{\mathrm{s}}) = \frac{f_{\sigma_{\mathrm{s}}}(\lambda_{\mathrm{s}})}{f_{\sigma_{\mathrm{s}}}(\Lambda_{\mathrm{F}})}.$$

It can be seen as a sampling of the Gaussian distribution over the points of the fine lattice Λ_{F}.

We illustrate in Fig. 1 two examples of the probability density function of the discrete Gaussian distribution we consider in our model and the sum codewords resulting from the superposition of 2-dimensional lattice codewords for the cases of $N = 2$ and $N = 5$ considering a fine lattice Λ_{F} of a generator matrix $\mathbf{M} = \begin{bmatrix} 2 & 3 \\ 3 & -1 \end{bmatrix}$ and the coarse lattice $\Lambda_{\mathrm{C}} = 11\mathbb{Z}^2$ (in this case, the messages delivered by the sources are drawn from $\mathbb{F}_{11}$). These examples show that the lattice Gaussian distribution fits our settings. In addition, although the approximation for low number of users ($N = 2$) seems to be questionable, we will show later, by numerical results, that this model is well justified in the context of lattice network coding even for low number of sources.

Now, once we have characterized the statistical distribution of the sum codewords, we go back to the MAP metric. Using the discrete Gaussian distribution, the decoding rule in (6) is equivalent to:

$$\hat{\lambda}_{\mathrm{map}} = \underset{\lambda_{\mathrm{s}} \in \mathcal{C}_{\mathrm{s}}}{\operatorname{argmin}} \left\{ \ln\left(f_{\sigma_{\mathrm{s}}}(\Lambda_{\mathrm{F}})\right) + n\ln(\sigma_{\mathrm{s}}\sqrt{2\pi}) + \frac{\|\lambda_{\mathrm{s}}\|^2}{2\sigma_{\mathrm{s}}^2} + \frac{\|\mathbf{y}-\lambda_{\mathrm{s}}\|^2}{2\sigma^2}\right\}.$$



(a)

(b)

Fig. 1. PDF of the codebook induced by the sum of codewords compared to the Gaussian model. (a) Probability density function for N=2. (b) Probability density function for N=5.

The first and second terms in this metric are independent of the variable λ_s, they can be disregarded in the optimization over λ_s. Then we obtain our novel decoding metric given by:

$$\hat{\lambda}_{\text{map}} = \underset{\lambda_s \in \mathcal{C}_s}{\operatorname{argmin}} \left\{ \|\mathbf{y} - \lambda_s\|^2 + \beta^2 \|\lambda_s\|^2 \right\} \tag{7}$$

where $\beta = \frac{\sigma}{\sigma_s}$. Using this new metric, we show in Proposition III.1 that MAP decoding reduces to solve for a closest vector problem.

Proposition III.1: The MAP decoding metric in (7) is equivalent to find the closest vector in the lattice Λ_{aug} of generator matrix $\mathbf{M}_{\text{aug}} = [\mathbf{M}\ \beta\mathbf{M}]^t \in \mathbb{R}^{2n \times n}$ to the vector $\mathbf{y}_{\text{aug}} = [\mathbf{y}\ \mathbf{0}_n]^t$ according to the following metric:

$$\hat{\lambda}_{\text{map}} = \underset{\substack{\lambda_s \in \mathcal{C}_s \\ \mathbf{x}_{\text{aug}} = \mathbf{M}_{\text{aug}} \lambda_s}}{\operatorname{argmin}} \|\mathbf{y}_{\text{aug}} - \mathbf{x}_{\text{aug}}\|^2. \tag{8}$$

Proof: The decoding metric in (7) can be written as:

$$\hat{\lambda}_{\text{map}} = \underset{\lambda_s \in \mathcal{C}_s}{\operatorname{argmin}} \left\{ \left\| \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_n \end{bmatrix} - \begin{bmatrix} \lambda_s \\ \beta\lambda_s \end{bmatrix} \right\|^2 \right\} = \underset{\lambda_s \in \mathcal{C}_s}{\operatorname{argmin}} \|\mathbf{y}_{\text{aug}} - \mathbf{I}_{\text{aug}} \lambda_s\|^2 \tag{9}$$

where $\mathbf{I}_{\text{aug}} = [\mathbf{I}_n\ \beta\mathbf{I}_n]^t \in \mathbb{R}^{2n \times n}$ is a full rank matrix. On the other hand, given that the sum codewords belong to the fine lattice according to the shaping region $\mathcal{S}_s$, any codeword λ_s can be written in the form $\lambda_s = \mathbf{M}\mathbf{u}$ where $\mathbf{u} \in \mathcal{A}_s \subset \mathbb{Z}^n$ and $\mathcal{A}_s$ translates the shaping constraint imposed by $\mathcal{S}_s$ and can be deduced from the shaping boundaries limited by the transmission power constraint P. Consequently the optimization problem in (9) is equivalent to solving

$$\hat{\lambda}_{\text{map}} = \underset{\substack{\lambda_s \in \mathcal{C}_s \\ \lambda_s = \mathbf{Mu}}}{\operatorname{argmin}} \|\mathbf{y}_{\text{aug}} - \mathbf{I}_{\text{aug}} \mathbf{Mu}\|^2 = \underset{\substack{\lambda_s \in \mathcal{C}_s \\ \lambda_s = \mathbf{Mu}}}{\operatorname{argmin}} \|\mathbf{y}_{\text{aug}} - \mathbf{M}_{\text{aug}} \mathbf{u}\|^2. \tag{10}$$

$\mathbf{M}_{\text{aug}}$ is a full rank matrix and $\mathbf{u}$ is an integer vector, then solving (10) consists in finding the closest vector $\mathbf{x}_{\text{aug}} = \mathbf{M}_{\text{aug}}\mathbf{u}$ to $\mathbf{y}_{\text{aug}}$ in the n-dimensional lattice Λ_{aug} of a generator matrix $\mathbf{M}_{\text{aug}}$. After finding the optimal integer vector $\mathbf{u}_{\text{opt}}$ that minimizes the metric in (10), the optimal MAP estimate is deduced by $\hat{\lambda}_{\text{map}} = \mathbf{M}\mathbf{u}_{\text{opt}}$. □

In our implementation, we use a modified version of the sphere decoder to solve this closest vector problem taking into account the shaping constraint.

Remark: The MAP decoding metric in (7) involves two terms each one of them is given by an Euclidean distance. When the first term is dominant, which is the case when $\beta^2 = \frac{\sigma^2}{\sigma_s^2} = \frac{\sigma^2}{N\sigma_{\mathbf{x}}^2} \ll 1$, the MAP decoding rule reduces to ML decoding (which is equivalent to minimum distance decoding in this case since we don't perform a scaling step). Given that $\sigma_{\mathbf{x}}^2$ depends on the power constraint P, we deduce that this case of figure is likely to happen either at high Signal-to-Noise Ratio or when $N\sigma_{\mathbf{x}}^2$ is sufficiently higher than the noise variance $\sigma^2\mathbf{I}_n$. We expect then that the MAP decoding and the conventional decoder achieve similar performance at high SNR range. Adversely, at the low and moderate SNR regime and when the product $N\sigma_{\mathbf{x}}^2$ is small, the second term in the decoding metric applies an incremental constraint that considers the non-uniform distribution of the sum codewords in $\mathcal{C}_s$ which is not taken into account under the conventional decoder. In this case, we expect that the MAP decoder outperforms the minimum distance decoding-based one.

We provide in the following proposition an equivalent formulation of the MAP decoding metric.

Proposition III.2: The MAP decoding metric in (7) is equivalent to MMSE-GDFE [13] preprocessed minimum Euclidean distance decoding according to the metric:

$$\hat{\lambda}_{\text{map}} = \underset{\lambda_s \in \mathcal{C}_s}{\operatorname{argmin}} \|\mathbf{Fy} - \mathbf{B}\lambda_s\|^2. \tag{11}$$

$\mathbf{F} \in \mathbb{R}^{n \times n}$ and $\mathbf{B} \in \mathbb{R}^{n \times n}$ denote respectively the forward and backward filters of the MMSE-GDFE preprocessing for the channel $\mathbf{y} = \lambda_s + \mathbf{z}$ such that $\mathbf{B}^t\mathbf{B} = (1 + \beta^2)\mathbf{I}_n$ and $\mathbf{F}^t\mathbf{B} = \mathbf{I}_n$.

Proof: Let $N(\lambda_s)$ denote the metric we aim to minimize in (7), we have the following:

$$\begin{aligned} N(\lambda_s) &= \|\mathbf{y} - \lambda_s\|^2 + \beta^2\|\lambda_s\|^2 \\ &= \mathbf{y}^t\mathbf{y} - 2\mathbf{y}^t\lambda_s + \lambda_s^t\lambda_s + \beta^2\lambda_s^t\lambda_s \\ &= (1 + \beta^2)\lambda_s^t\lambda_s + \mathbf{y}^t\mathbf{y} - 2\mathbf{y}^t\lambda_s \\ &= \lambda_s^t\mathbf{B}^t\mathbf{B}\lambda_s + \mathbf{y}^t\mathbf{y} - 2\mathbf{y}^t\mathbf{F}^t\mathbf{B}\lambda_s \\ &= \underbrace{\lambda_s^t\mathbf{B}^t\mathbf{B}\lambda_s + \mathbf{y}^t\mathbf{F}^t\mathbf{F}\mathbf{y} - 2\mathbf{y}^t\mathbf{F}^t\mathbf{B}\lambda_s}_{\|\mathbf{Fy} - \mathbf{B}\lambda_s\|^2} + \underbrace{\mathbf{y}^t(\mathbf{I}_n - \mathbf{F}^t\mathbf{F})\mathbf{y}}_{\Gamma(\mathbf{y})} \end{aligned} \tag{12}$$

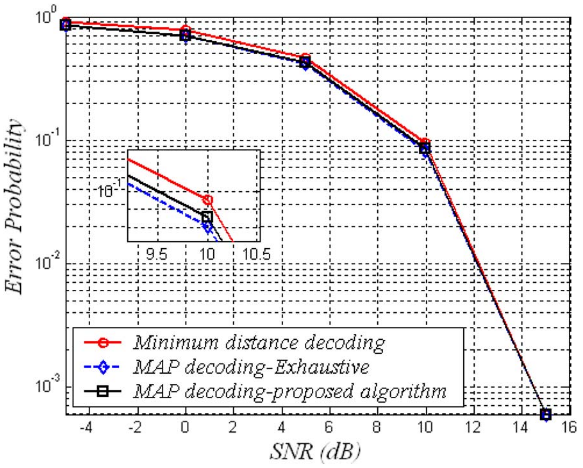


Fig. 2. Error performance for the case $n = 2, N = 2, P = 6.5$.

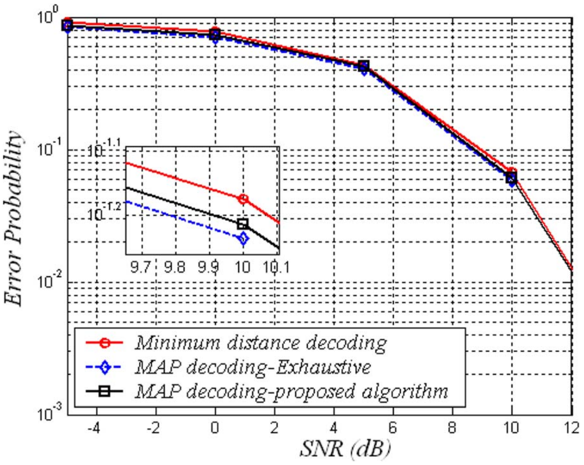


Fig. 3. Error performance for $n = 2, N = 5, P = 6.5$.

where $\mathbf{F} \in \mathbb{R}^{n\times n}$ and $\mathbf{B} \in \mathbb{R}^{n\times n}$ are chosen such that: $\mathbf{B}^t\mathbf{B} = (1+\beta^2)\mathbf{I}_n$ and $\mathbf{F}^t\mathbf{B} = \mathbf{I}_n$. Given that $\Gamma(\mathbf{y}) > 0$ and independent of λ_s, minimization of $N(\lambda_s)$ is equivalent to minimize $\|\mathbf{F}\mathbf{y} - \mathbf{B}\lambda_s\|^2$. The last piece to our proof is to show that the matrices $\mathbf{F}$ and $\mathbf{B}$ correspond to the filters of the MMSE-GDFE preprocessing [13] in the system $\mathbf{y} = \lambda_s + \mathbf{z}$ of input λ_s and AWGN $\mathbf{z}$. This proof is provided in Appendix A. □

In order to find the MAP estimate according to the decoding metric in (11), the receiver first performs MMSE-GDFE preprocessing, then performs minimum Euclidean distance decoding to find the nearest point to $\mathbf{F}\mathbf{y}$ in the lattice of generator matrix $\mathbf{BM}$ according to the shaping constraint imposed by the subset $\mathcal{C}_s$.

B. Numerical Results

We evaluate in this subsection the performance of the conventional decoder (based on MMSE scaling and minimum distance decoding) and the proposed MAP decoding algorithm implementing a modified sphere decoder. In addition, for validating the assumption of Gaussianity law assumption we considered to derive our MAP decoding metric, we also present a naive exhaustive search for solving (6). Using this approach, no assumptions on the distribution of the sum codewords are considered. The receiver, given the number of sources and the original codebook $\mathcal{C}$, derives the statistics of the sum codebook to compute the corresponding values of $p(\lambda_s)$ for all codewords $\lambda_s \in \mathcal{C}_s$, then, it exhaustively seeks the codeword which maximizes the decoding metric in (6). We study in our analysis two lattice examples as described below.

Example 1: 2-dimensional lattice ($n = 2$) for this example we consider the same nested lattice code used to get the statistical distributions plotted in Fig. 1 for $N = 2$ and $N = 5$. The shaping constraint in this case is given by $P = \sigma_{\mathbf{x}}^2 = 6.5$. Given the number of sources and the power constraint imposed by the coarse lattice, we calculate for each case the bounds requirements to be considered in the decoding process.

Numerical results concerning the case $N = 2$, depicted in Fig. 2, show that our proposed algorithm achieves almost identical performance as the exhaustive search, which confirms the effectiveness of our metric as well as the validity of the Gaussianity law assumption considered to model the sum-codewords even for the case of low number of sources $N = 2$. Moreover, plotted curves show that the MAP decoder outperforms the conventional minimum distance decoding. The gain for this case is limited to 0.5 dB for an error probability equal to 10^{-1}.

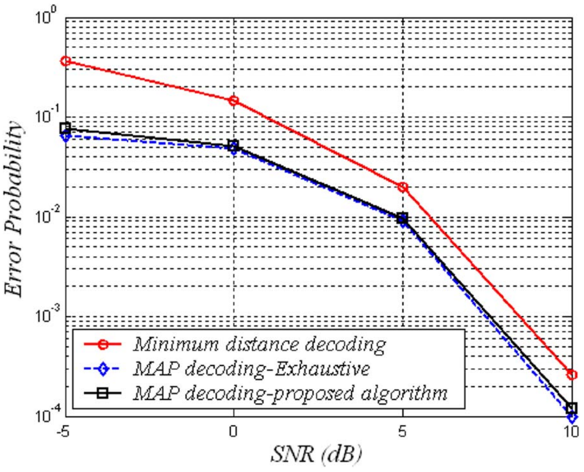


Fig. 4. Error performance for $n = 4, N = 2, P = 1$.

Results for the case of $N = 5$ plotted in Fig. 3 confirm the previous findings and show that the performance gap between the MAP and the Minimum distance decoder is also not high. Common to these two settings is the high value of $N\sigma_{\mathbf{x}}^2$, and as we expected in our previous remark, MAP and ML achieve indeed same performance for this case of figure.

Example 2: 4-dimensional lattice ($n = 4$) In this example we consider the integer fine lattice Λ_F of a generator matrix the identity $\mathbf{I}_4$ together with a cubic shaping region according to $P = 1$. The aim of considering this example is to analyze the performance of the MAP decoder when the lattice dimension increases. Simulation results depicted in Fig. 4 show that our proposed MAP algorithm achieves a gain of 1 dB at a codeword error rate of 10^{-3} over the minimum distance decoder and has a small gap to the decoder based on exhaustive search. This case shows the merit of applying the MAP decoding in settings where the product $N\sigma_{\mathbf{x}}^2$ is small.

IV. Efficient Decoders in Fading Channels

We move in this section to the general case of fading channels. The tools we will use in our analysis are valid only in the case of integer lattices, thus we will consider an n-dimensional nested lattice code $\mathcal{C} \subset \mathbb{Z}^n$ involving a fine lattice $\Lambda_{\mathrm{F}} \subset \mathbb{Z}^n$ of a generator matrix $\mathbf{M}$ and a coarse lattice $\Lambda_{\mathrm{C}} \subset \mathbb{Z}^n$. For this case, $\mathbf{M}$ is an integer full rank matrix. We will start with the multi-dimensional case which was independently studied in [6] then we provide more in depth analysis regarding the one-dimensional case.

A. Problem Statement

After selecting α and $\mathbf{a}$, the receiver scales the channel output to get:

$$\tilde{\mathbf{y}} = \sum_{i=1}^{N} a_i \mathbf{x}_i + \sum_{i=1}^{N} (\tilde{h}_i - a_i)\mathbf{x}_i + \tilde{\mathbf{z}} \tag{13}$$

where $\tilde{h}_i = \alpha h_i$, $i = 1, \ldots, N$ and $\tilde{\mathbf{z}} = \alpha\mathbf{z}$, and attempts to decode $\lambda = \left[\sum_{i=1}^{N} a_i \mathbf{x}_i\right] \bmod \Lambda_{\mathrm{C}}$. We are concerned in this part with decoding the integer combination $\mathbf{t} = \sum_{i=1}^{N} a_i \mathbf{x}_i$. The modulo-lattice operation is performed in a second stage separately and does not impact the error performance. Thus, we evaluate the decoding error probability defined as: $\mathrm{P_e} = \Pr(\hat{\mathbf{t}} \neq \mathbf{t})$. Given the vector $\mathbf{a}$ and the shaping boundaries for the original codewords, it is known that the desired vector $\mathbf{t}$ belongs to a subset $\mathcal{C}_f$ in the fine lattice Λ_{F} determined using the original shaping constraint of the source codewords and the knowledge of the network code vector $\mathbf{a}$. However, this shaping constraint is disregarded under the conventional CF decoder. In addition, in contrast to the Gaussian channel case, the statistical distribution of the combination codewords is unknown and difficult to model since it depends on the channel realizations. For this reason, we assume in the following a uniform distribution of the subset $\mathcal{C}_f$ and analyze the ML decoder that takes into consideration the shaping condition.

B. ML Decoding Metric

The ML criterion is based on maximizing the conditional probability $p(\tilde{\mathbf{y}}|\mathbf{t})$ according to:

$$\hat{\mathbf{t}} = \underset{\mathbf{t}\in\mathcal{C}_f}{\operatorname{argmax}}\, p(\tilde{\mathbf{y}}|\mathbf{t}). \tag{14}$$

Given that $\mathbf{t} = \sum_{i=1}^{N} a_i \mathbf{x}_i$, we can equivalently write (14) as:

$$\hat{\mathbf{t}} = \underset{\mathbf{t}\in\mathcal{C}_f}{\operatorname{argmax}} \sum_{\substack{(\mathbf{x}_1,\ldots,\mathbf{x}_N)\in\mathcal{C}^N \\ \sum_{i=1}^{N} a_i\mathbf{x}_i = \mathbf{t}}} p\left(\tilde{\mathbf{y}}\,|(\mathbf{x}_1,\ldots,\mathbf{x}_N)\right) p(\mathbf{x}_1,\ldots,\mathbf{x}_N). \tag{15}$$

The transmitted codewords are assumed to be uniformly distributed over the nested lattice code $\mathcal{C}$, i.e., $\mathbf{x}_1, \ldots, \mathbf{x}_N$ are equiprobable. On the other hand, we have,

$$p(\tilde{\mathbf{y}}|\mathbf{x}_1,\ldots,\mathbf{x}_N) \propto \exp\left(\frac{-1}{2\tilde{\sigma}^2}\left\|\tilde{\mathbf{y}} - \sum_{i=1}^{N}\tilde{h}_i\mathbf{x}_i\right\|^2\right) \tag{16}$$

where $\tilde{\sigma}^2 = \alpha^2\sigma^2$. Combining (16) and (15), we get:

$$\hat{\mathbf{t}} = \underset{\mathbf{t}\in\mathcal{C}_f}{\operatorname{argmax}} \sum_{\substack{(\mathbf{x}_1,\ldots,\mathbf{x}_N)\in\mathcal{C}^N \\ \sum_{i=1}^{N} a_i\mathbf{x}_i = \mathbf{t}}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\left\|\tilde{\mathbf{y}} - \sum_{i=1}^{N}\tilde{h}_i\mathbf{x}_i\right\|^2\right). \tag{17}$$

Let

$$\varphi(\mathbf{t}) = \sum_{\substack{(\mathbf{x}_1,\ldots,\mathbf{x}_N)\in\mathcal{C}^N \\ \sum_{i=1}^{N} a_i\mathbf{x}_i = \mathbf{t}}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\left\|\tilde{\mathbf{y}} - \sum_{i=1}^{N}\tilde{h}_i\mathbf{x}_i\right\|^2\right). \tag{18}$$

Our objective in the following is to express φ as a function of the desired equation $\mathbf{t}$. To this end, we need to express the codewords $\mathbf{x}_i$, $i = 1, \ldots, N$ as functions of $\mathbf{t}$. Given the integer nature of the vector $\mathbf{a}$ and the codewords $\mathbf{x}_i$, this task requires to solve the system of Diophantine equations $\mathbf{t} = \sum_{i=1}^{N} a_i \mathbf{x}_i$. For $n-$dimensional vectors this can be done using the Hermite Normal Form (HNF) of integral matrices [14],[15] as explained in the following.

Define the integer-valued matrix $\tilde{\mathbf{M}} \in \mathbb{Z}^{n\times nN}$ as $\tilde{\mathbf{M}} = [a_1\mathbf{M}a_2\mathbf{M}\ldots a_N\mathbf{M}]$. The Hermite Normal Form of $\tilde{\mathbf{M}}$ is such that: $\tilde{\mathbf{M}}\mathbf{U} = \left[\mathbf{0}^{n\times(N-1)n}|\mathbf{B}\right]$ where $\mathbf{U} \in \mathbb{Z}^{nN\times nN}$ is a unimodular matrix, and $\mathbf{B} \in \mathbb{Z}^{n\times n}$ is an invertible matrix. Then, we decompose the matrix $\mathbf{U}$ in the form:

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_1 & \mathbf{V}_1 \\ \mathbf{U}_2 & \mathbf{V}_2 \\ \vdots & \vdots \\ \mathbf{U}_N & \mathbf{V}_N \end{bmatrix}, \qquad \mathbf{V}_i \in \mathbb{Z}^{n\times n}, \mathbf{U}_i \in \mathbb{Z}^{n\times n(N-1)}. \tag{19}$$

Using the HNF of the matrix $\tilde{\mathbf{M}}$ and this decomposition of the matrix $\mathbf{U}$ we can write:

$$\begin{aligned} \sum_{i=1}^{N} a_i\mathbf{M}\mathbf{V}_i = \mathbf{B} &\Leftrightarrow \sum_{i=1}^{N} a_i\mathbf{M}\mathbf{V}_i\mathbf{B}^{-1} = \mathbf{I}_n \\ &\Leftrightarrow \sum_{i=1}^{N} a_i\mathbf{M}\mathbf{V}_i\mathbf{B}^{-1}\mathbf{t} = \mathbf{t}. \end{aligned} \tag{20}$$

By identifying equation (20) to the system $\sum_{i=1}^{N} a_i\mathbf{x}_i = \mathbf{t}$ we get a particular solution of the system of Diophantine equations as:

$$\mathbf{x}_i = \mathbf{M}\mathbf{V}_i\mathbf{B}^{-1}\mathbf{t}, \qquad \forall\, i = 1, \ldots, N.$$

The set of all solutions is then given by:

$$\mathbf{x}_i = \mathbf{M}\mathbf{V}_i\mathbf{B}^{-1}\mathbf{t} + \mathbf{d}_i$$

where $\mathbf{d}_i$ belong to the lattice of a generator matrix $\mathbf{M}\mathbf{U}_i$ for $i = 1, \ldots, N$.

C. Likelihood Function

We go back now to the ML decoding rule defined in (17) and replace the vectors $\mathbf{x}_i$ by the solution of the Diophantine equations we obtain

$$\hat{\mathbf{t}} = \arg\max_{\mathbf{t}\in\mathcal{C}_f} \sum_{\mathbf{q}\in\mathcal{A}_{\mathcal{L}}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\left\|\omega(\mathbf{t})-\mathbf{q}\right\|^2\right) \tag{21}$$

where $\mathbf{q} = \sum_{i=1}^{N} \tilde{h}_i \mathbf{d}_i$ belongs to $\mathcal{A}_{\mathcal{L}}$, a finite subset of the lattice $\mathcal{L}$ of a generator matrix $\sum_{i=1}^{N} \tilde{h}_i \mathbf{M}\mathbf{U}_i$ determined by the boundaries of the original codewords according to the transmission power constraint. Moreover, $\omega(\mathbf{t}) = \tilde{\mathbf{y}} - \sum_{i=1}^{N} h_i \mathbf{M}\mathbf{V}_i\mathbf{B}^{-1}\mathbf{t}$. To find the ML solution, we need to maximize the likelihood function:

$$\varphi(\mathbf{t}) = \sum_{\mathbf{q}\in\mathcal{A}_{\mathcal{L}}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\left\|\omega(\mathbf{t})-\mathbf{q}\right\|^2\right). \tag{22}$$

This function is a sum of Gaussian measures, it is periodic and depends on the Signal-to-Noise Ratio. Additionally, its most important characteristic is that it can be flat, which means that for some values of the channel coefficients, the network code vector and the Signal-to-Noise Ratio, the maximum of φ can be achieved by several values of $\mathbf{t}$, which makes the ML decision ambiguous and results in decoding errors. This flatness behavior is characterized by Belfiore and Ling in [6] by the so called the *Flatness Factor*. For the ML decoding rule, we should minimize the flatness factor of the lattice $\mathcal{L}$ over which is performed the sum of the Gaussian measures in order to be able to distinguish the maximum values of the likelihood function and perform a correct decoding decision. Solving the ML decoding metric requires more research on the sum of Gaussian measures. Alternatively, authors in [6] propose an approximation of ML decoding based on *Diophantine approximation* and consists in approximating the sum of the Gaussian functions by a single one according to the optimization problem given by:

$$\hat{\mathbf{t}} = \underset{\substack{\mathrm{t}\in\mathcal{C}f \\ \mathrm{q}\in\mathcal{A}\mathcal{L}}}{\operatorname{argmax}} \left\|\omega(\mathbf{t})-\mathbf{q}\right\|^2 \tag{23}$$

In order to solve the multi-dimensional case, we need to develop efficient algorithms that handle simultaneous Diophantine approximations which requires deeper investigation. In this work, we will study in the following the 1-D case in more details and leave the multi-dimensional case for future works for complexity reasons.

D. 1-D Lattices Case Study

Here, we focus on the case of 1-D lattices in $\mathbb{Z}$ and $N = 2$. Transmitted codewords x_1 and x_2 are just integer scalars drawn i.i.d. from the integer constellation over $\mathbb{Z}$ defined by $\mathcal{C} = [-S_{\mathrm{m}}, S_{\mathrm{m}}]$ for $S_{\mathrm{m}} \in \mathbb{Z}^+$. This integer codebook can be seen as a nested lattice code in $\mathbb{Z}$ involving the fine lattice $\Lambda_{\mathrm{F}} = \mathbb{Z}$ and the coarse lattice $\Lambda_{\mathrm{C}} = 2S_{\mathrm{m}}\mathbb{Z}$. The channel output in this case is given by: $y = h_1x_1 + h_2x_2 + z$, with $h_i \in \mathbb{R}$ and $z \sim \mathcal{N}(0, \sigma^2)$. The receiver selects the optimal scaling parameter and the optimal network code vector $\mathbf{a} = [a_1\ a_2]^t$ and attempts to decode the integer combination $t = a_1x_1 + a_2x_2$ from the integer set $\mathcal{C}_f$ determined by S_{m} and the values of the coefficients a_1 and a_2. The scaled channel output is given by:

$$\begin{aligned}\tilde{y} &= a_1x_1 + a_2x_2 + (\tilde{h}_1 - a_1)x_1 + (\tilde{h}_2 - a_2)\,x_2 + \tilde{z} \\ \tilde{h}_i &= \alpha h_i, \quad i = 1, 2; \quad \tilde{z} = \alpha z.\end{aligned} \tag{24}$$

Under these settings, the ML solution is given by:

$$\hat{t} = \underset{t\in\mathcal{C}_f}{\operatorname{argmax}} \sum_{\substack{(x_1,x_2)\in\mathcal{C}^2 \\ a_1x_1+a_2x_2=t}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\|\tilde{y} - \tilde{h}_1x_1 - \tilde{h}_2x_2\|^2\right). \tag{25}$$

And the likelihood function is given by:

$$\varphi(t) = \sum_{\substack{(x_1,x_2)\in\mathcal{C}^2 \\ a_1x_1+a_2x_2=t}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\|\tilde{y} - \tilde{h}_1x_1 - \tilde{h}_2x_2\|^2\right). \tag{26}$$

Our aim now is to express φ as a function of t only. Therefore, we need to solve the *Diophantine Equation* $t = a_1x_1 + a_2x_2$. Let $g = a_1 \wedge a_2$ denote the greatest common divisor (gcd) of a_1 and a_2. If the desired scalar t is a multiple of g, the Diophantine equation admits an infinite number of solutions in the form:

$$\begin{cases} x_1 = \frac{u_1}{g}t + \frac{a_2}{g}k \\ x_2 = \frac{u_2}{g}t - \frac{a_1}{g}k \end{cases} \tag{27}$$

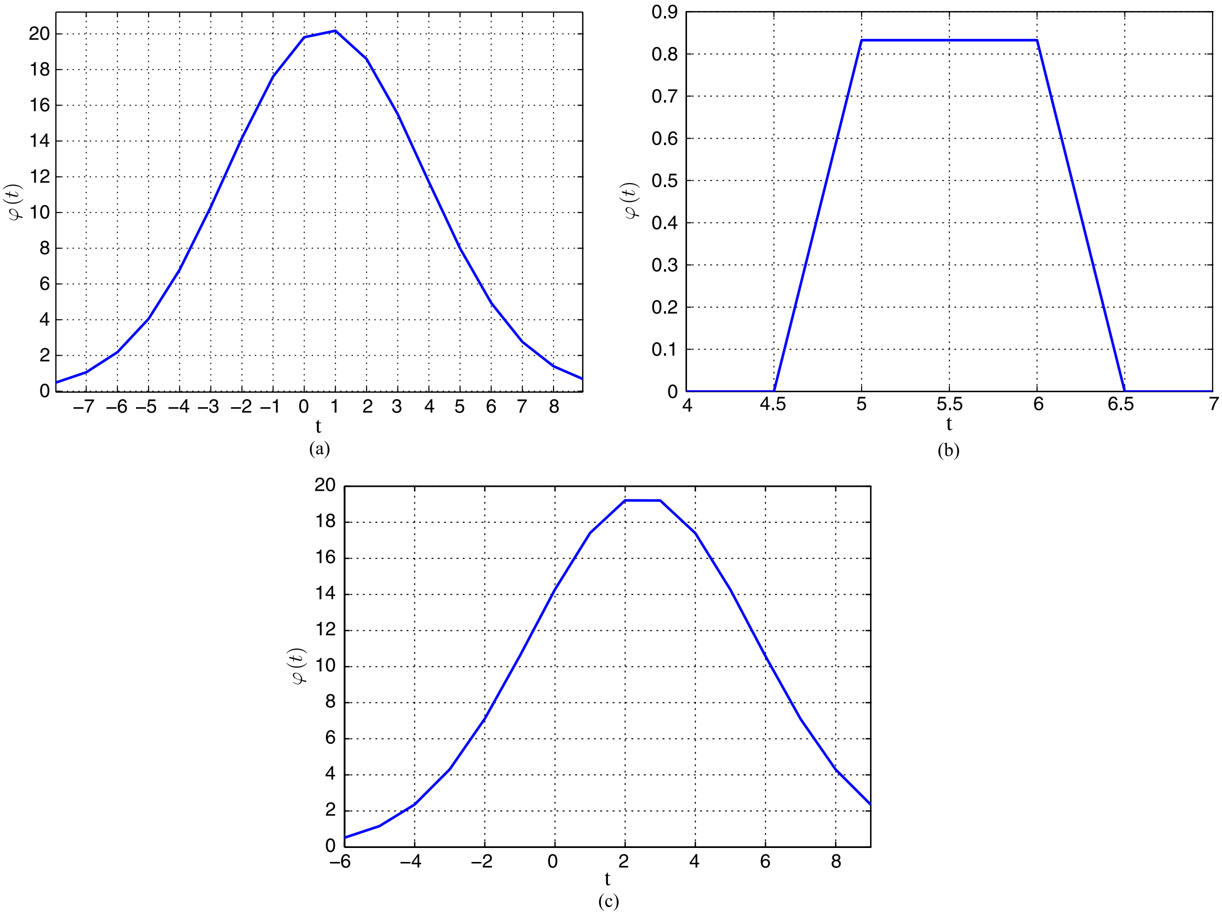
where $k \in \mathcal{A}_{\mathbb{Z}} \subset \mathbb{Z}$ such that the shaping constraint for the desired combination is satisfied. (u_1, u_2) is a particular solution of the equation $a_1x_1 + a_2x_2 = g$ that can be derived using the *Extended Euclid Algorithm* [16]. If t is not a multiple of g, then the Diophantine equation has no solutions. For what concerns our case, the network code vector $\mathbf{a}$ corresponds to the coordinates of a lattice shortest vector, then the coefficients a_1 and a_2 are coprime. Thus, the Diophantine equation under question has always infinite solutions given by the system in (27) with $g = 1$. Accordingly, we can write the ML solution in (25) as

$$\hat{t} = \underset{t\in\mathcal{C}_f}{\operatorname{argmax}} \underbrace{\sum_{k\in\mathcal{A}_{\mathbb{Z}}} \exp\left(\frac{-1}{2\tilde{\sigma}^2}\|\tilde{y} - \gamma t + \beta k\|^2\right)}_{\varphi(t)} \tag{28}$$

where $\gamma = \tilde{h}_1u_1 + \tilde{h}_2u_2$ and $\beta = a_1\tilde{h}_2 - a_2\tilde{h}_1$.

1) Properties of the Likelihood Function: φ is a sum of Gaussian functions, it is periodic with *mean* $m = \tilde{y}$, *period* $p = \frac{\beta}{2\tilde{\sigma}^2}$ and *width* $w = \frac{\gamma}{2\tilde{\sigma}^2}$. In addition, φ depends on the SNR, the channel coefficients, the coefficient vector $\mathbf{a}$ and obviously on the constellation bounds defined by S_m. We illustrate in Fig. 5(a) an example of the likelihood function obtained for $S_m = 5$, $x_1 = 3$, $x_2 = 4$ at SNR = 10 dB and $\mathbf{h} = [-1.191\ 1.189]^t$. The optimal network code vector for this case is equal to $\mathbf{a} = [-1\ 1]^t$. Accordingly, the desired combination should be equal to $t = 1$. The corresponding likelihood function depicted in Fig. 5(a) is well maximized at $\hat{t} = 1$. In this case, it is easy to decode the maximum of $\varphi(t)$ since we can distinguish a peak corresponding to the unique $\hat{t}$ for which this function is maximized.

- *Flatness behavior of the likelihood function* as we mentioned in the previous subsection, one of the properties of the likelihood function is that it can be flat. This



(a)

(b)

(c)

Fig. 5. Examples of the likelihood function. (a) $S_{\rm m} = 5$, SNR = 60 dB. (b) $S_{\rm m} = 5$, SNR = 60 dB. (c) $S_{\rm m} = 10$, SNR = 60 dB.

behavior is shown through Fig. 5(b) obtained at SNR = 60 dB $S_m = 5, x_1 = -5, x_2 = -4, \mathbf{h} = [1.3681 - 0.2359]^t$, $\mathbf{a} = [-1\ 0]^t$. The maximum of the likelihood function is obtained for two integer values $t_1 = 5$ and $t_2 = 6$ while the correct decodable value must be $\hat{t} = 5$ for the corresponding values of x_1 and x_2. In this case, the receiver can make a decoding error.

— *Impact of the constellation size*: the likelihood function depends on the constellation size and the values of S_m. When the size of the codebook increases, the set $\mathcal{C}_f$ over which the desired combination t should be searched becomes large. Consequently, the width of φ becomes large and the likelihood function is made flat. Thus, decoding the maximal value of t becomes ambiguous. An example of this scenario is illustrated in Fig. 5(c) obtained for $S_m = 10$, SNR = 10 dB, $x_1 = -2, x_2 = -4$, $\mathbf{h} = [1.4741 - 0.2839]^t$, $\mathbf{a} = [-1\ 0]^t$. We can see that the likelihood function attains its maximum for $t = 2$ and $t = 3$ while the correctly decoded value is $\hat{t} = 2$. This ambiguity leads to decoding errors.

2) Diophantine Approximation: The sum of Gaussian functions in the likelihood function makes the ML decoding hard to handle in practice. For easy implementation, we propose in the following a near-ML decoder by approximating the sum of the Gaussian measures by a single function. We use the result stating that for $t \in \mathbb{Z}$, φ is maximized for t which minimizes $|\tilde{y} - \gamma t + \beta k|$. Given this observation, we define a new optimization problem equivalent to (28) by:

$$\hat{t} = \underset{\substack{k \in \mathcal{A}\mathbb{Z} \\ t \in \mathcal{C}f}}{\arg\min} |\tilde{y} - \gamma t + \beta k|. \tag{29}$$

Let $\beta' = \frac{\beta}{\gamma}$ and $y' = -\frac{\tilde{y}}{\gamma}$, then this minimization problem is equivalent to:

$$\hat{t} = \underset{\substack{k \in \mathcal{A}\mathbb{Z} \\ t \in \mathcal{C}f}}{\arg\min} |\beta' k - t - y'| \tag{30}$$

This problem corresponds to solving the *Inhomogeneous Diophantine Approximation in the absolute sens* (IDA) [17], $F(t,k)$, defined as, $F(t,k) = |\beta' k - t - y'|$. It consists in finding the best rational approximation $\frac{t}{k}$, $k \in \mathbb{Z}$ of the real number β' assumed an additional real shift y'. In our setting, the set of the Diophantine approximations is determined by the limits imposed by the shaping boundaries of the subset $\mathcal{C}_f$. In literature, there exist simple and easy-to-implement algorithms to solve Diophantine approximations of reals. The best known one is the *Cassel's Algorithm* [18]. In this work we adopt a modified version of this algorithm to take into consideration the shaping constraint and ensure that the resulting solution (t, k) satisfies $t \in \mathcal{C}_f$. Details of this algorithm are provided in Appendix B.

3) Simulation Results: We address now the performance evaluation of the conventional decoder and the proposed

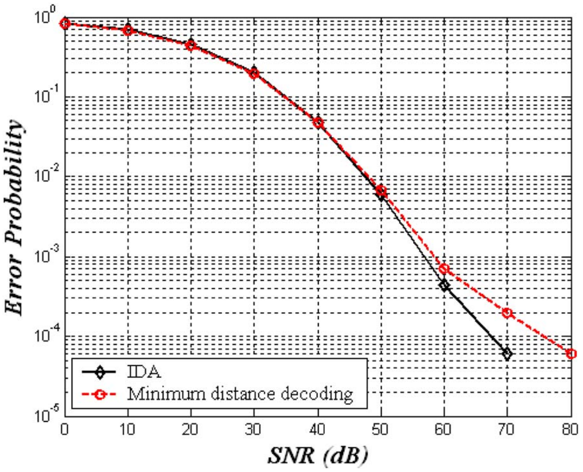


Fig. 6. Error probability for $S_m = 5$.

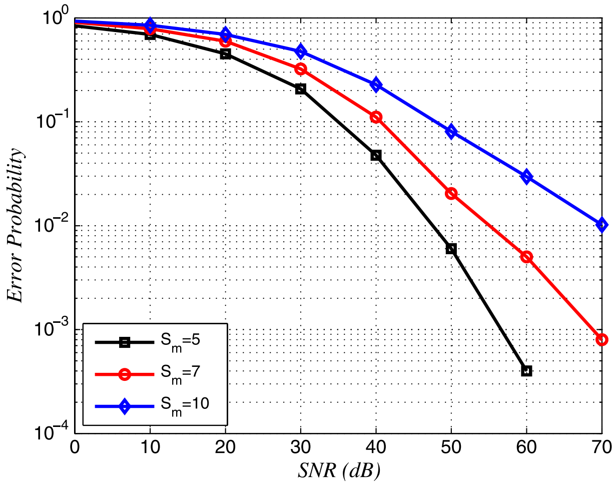


Fig. 7. Error probability using the IDA decoding.

Inhomogenous Diophantine Approximation (IDA) decoder. We are interested in our simulations only in the one-dimensional case due to the complexity of the multi-dimensional scenario. In addition, we consider the same settings analyzed previously involving two sources transmitting integer symbols x_1 and x_2 drawn from the constellation set $\mathcal{C} = [-S_m\ S_m]$. We analyze the error probability on decoding t. For what concerns the conventional decoder, the receiver solves for the best network code vector $\mathbf{a}$ solution of the shortest vector problem, scales the channel output, then decodes to the nearest integer value. For the IDA, given the vector $\mathbf{a}$, the receiver implements first the *Extended Euclid* algorithm to solve the Diophantine equation $a_1x_1 + a_2x_2 = g$, then uses the modified Cassel's algorithm to find the best inhomogeneous Diophantine approximation.

In Fig. 6, minimum distance decoding and IDA decoding are compared for $S_m = 5$. Our results show that both decoding methods achieve same performance for low and moderate SNR values.

The importance of the IDA method rises asymptotically. In Fig. 7, we analyze the performance of the proposed IDA decoding for three values of the constellation bound, defined by $S_m = 5,7,10$. This is to understand the impact of the constellation size on the diversity order. Fig. 7 illustrates that for $S_m = 5$ or less, the system has a diversity order equal to 1 for real symbols (which would correspond to a diversity order equal to 2 with complex-valued symbols). However, for higher constellation size, e.g., for $S_m = 7$ and $S_m = 10$, the diversity order is limited to 1/2. This is because when the constellation range increases, the likelihood function becomes flat, which makes the error function $F(t, k)$ subject to the Diophantine approximation flat. This result confirms our previous analysis on the impact of the constellation on the likelihood function.

V. Conclusion

This work was dedicated to decoding aspects for the Compute-and-Forward protocol in the basic multiple access real-valued channel. In the first part, we addressed the Gaussian channel case using real-valued lattices. After analyzing the MAP decoding rule, we derived a novel decoding metric and developed practical algorithms based on lattice spherical decoding showed to outperform the standard minimum distance decoder. In the second part, we studied the fading channel case assuming integer-valued lattices. We analyzed the n-dimensional case and proposed a novel near-ML decoder based on Diophantine approximation. Numerical results for the 1-D scenario show the gain of this method over the conventional CF decoder at high SNR range. Having developed MAP decoders in the Gaussian channel, we aim in the future to investigate the information theoretic performance of this decoder and evaluate the achievable rate in the two-way Gaussian relay channel. Additionally, we will explore ML decoders for the fading channel using multi-dimensional lattices and assess their complexity compared to the conventional CF decoder.

Appendix A
MMSE-GDFE Preprocessing Filters

We aim to show that the matrices $\mathbf{F}$ and $\mathbf{B}$ in the equivalent MAP decoding metric correspond respectively to the forward and backward filters of the MMSE-GDFE preprocessing in the channel $\mathbf{y} = \lambda_s + \mathbf{z}$ with input λ_s such that $\frac{1}{n}\mathbb{E}(\|\lambda_s\|^2) = \sigma_s^2$. Let $\mathbf{F}_m$ and $\mathbf{B}_m$ be the filters of the MMSE-GDFE preprocessing such that: $\mathbf{F}_m\mathbf{y} = \mathbf{F}_m\lambda_s + \mathbf{F}_m\mathbf{z} = \mathbf{B}_m\lambda_s + (\mathbf{F}_m - \mathbf{B}_m)\lambda_s + \mathbf{F}_m\mathbf{z}$. Let $\mathbf{w} = (\mathbf{F}_m - \mathbf{B}_m)\lambda_s + \mathbf{F}_m\mathbf{z}$ be the effective noise. First, with reference to [13], it is known that the MMSE-GDFE filters are connected, via the relation: $\mathbf{F}_m = \mathbf{B}_m^{-t}\mathbf{H}^t$ for a general multipath fading channel with channel matrix $\mathbf{H}$. In our case, the corresponding matrix in the studied Gaussian channel is the identity matrix, then we have the relation between the forward and backward matrices as $\mathbf{F}_m = \mathbf{B}_m^{-t}$. On the other hand, the MMSE-GDFE filters correspond to the minimization of the variance ε of the effective noise given by:

$$\begin{aligned}\varepsilon &= \frac{1}{n}\mathbb{E}[\mathbf{w}^t\mathbf{w}] = \frac{1}{n}\mathbb{E}\left[\mathrm{tr}(\mathbf{w}\mathbf{w}^t)\right]\\ &= \frac{1}{n}\mathrm{tr}\left(\mathbb{E}\left[(\mathbf{F}_m - \mathbf{B}_m)\lambda_s\lambda_s^t(\mathbf{F}_m - \mathbf{B}_m)^t\right] + \mathbb{E}\left[\mathbf{F}_m\mathbf{z}\mathbf{z}^t\mathbf{F}_m^t\right]\right)\\ &= \frac{1}{n}\mathrm{tr}\left((\mathbf{F}_m - \mathbf{B}_m)\underbrace{\mathbb{E}\left[\lambda_s\lambda_s^t\right]}_{\sigma_s^2\mathbf{I}_n}(\mathbf{F}_m - \mathbf{B}_m)^t + \mathbf{F}_m\underbrace{\mathbb{E}[\mathbf{z}\mathbf{z}^t]}_{\sigma^2\mathbf{I}_n}\mathbf{F}_m^t\right)\\ &= \frac{\sigma_s^2}{n}\mathrm{tr}\left(\mathbf{F}_m(\mathbf{I}_n+\beta^2\mathbf{I}_n)\mathbf{F}_m^t-\mathbf{F}_m\mathbf{B}_m^t-\mathbf{B}_m\mathbf{F}_m^t+\mathbf{B}_m\mathbf{B}_m^t\right) \quad (31)\end{aligned}$$

For notational reason, we introduce the matrix $\mathbf{T}$ such that $\mathbf{T}\mathbf{T}^t = (1+\beta^2)\mathbf{I}_n$ where $\mathbf{T}$ should be a unimodular matrix and let $\mathbf{G}$ such that $\mathbf{G} = \mathbf{F}_{\mathrm{m}}\mathbf{T}$, then ε equals to:

$$\begin{aligned}\varepsilon &= \frac{\sigma_{\mathrm{s}}^2}{n}\mathrm{tr}\Big((\mathbf{G}-\mathbf{B}_{\mathrm{m}}\mathbf{T}^{-t})\Big(\mathbf{G}^t-\mathbf{T}^{-1}\mathbf{B}_{\mathrm{m}}^t\Big)\\ &\qquad +\mathbf{B}_{\mathrm{m}}\Big(\mathbf{I}_n-(\mathbf{T}\mathbf{T}^t)^{-1}\Big)\mathbf{B}_{\mathrm{m}}^t\Big)\\ &= \frac{\sigma_{\mathrm{s}}^2}{n}\mathrm{tr}\left((\mathbf{G}-\mathbf{B}_{\mathrm{m}}\mathbf{T}^{-t})\Big(\mathbf{G}^t-\mathbf{T}^{-1}\mathbf{B}_{\mathrm{m}}^t\Big)+\frac{\beta^2}{1+\beta^2}\mathbf{B}_{\mathrm{m}}\mathbf{B}_{\mathrm{m}}^t\right)\end{aligned}$$

For fixed $\mathbf{B}_{\mathrm{m}}$ we seek first the optimal forward matrix $\mathbf{F}_{\mathrm{m}}$ which minimizes ε. This corresponds to have $\mathbf{G} = \mathbf{B}_{\mathrm{m}}\mathbf{T}^{-t}$ which results in: $\mathbf{F}_{\mathrm{m}} = \frac{1}{\sqrt{(1+\beta^2)}}\mathbf{B}_{\mathrm{m}}$. We get:

$$\varepsilon_{\min} = \frac{\sigma_{\mathrm{s}}^2}{n}\frac{\beta^2}{1+\beta^2}\mathrm{tr}\left(\mathbf{B}_{\mathrm{m}}\mathbf{B}_{\mathrm{m}}^t\right) = \frac{\sigma_{\mathrm{s}}^2}{n}\frac{\beta^2}{1+\beta^2}\mathrm{tr}\left(\mathbf{B}_{\mathrm{m}}^t\mathbf{B}_{\mathrm{m}}\right) \quad (32)$$

We have $\mathbf{B}_{\mathrm{m}}^t\mathbf{B}_{\mathrm{m}} = (1+\beta^2)\mathbf{I}_n$ which corresponds to: $\varepsilon_{\min} = \sigma_{\mathrm{s}}^2\beta^2$. Now, we will show that $\mathbf{F} = \mathbf{F}_{\mathrm{m}}$ and $\mathbf{B} = \mathbf{B}_{\mathrm{m}}$. First, $\mathbf{F}$ and $\mathbf{B}$ satisfy same constraints as the MMSE-GDFE filters. The last piece to prove the equivalence then is to prove that $\mathbf{F}$ and $\mathbf{B}$ allow to minimize the variance of the effective noise $\mathbf{w}$. Using equation (31) we compute the corresponding variance refered to ε_{eq}:

$$\begin{aligned}\varepsilon_{eq} &= \frac{\sigma_{\mathrm{s}}^2}{n}\mathrm{tr}\left((\mathbf{F}-\mathbf{B})(\mathbf{F}-\mathbf{B})^t+\beta^2\mathbf{F}\mathbf{F}^t\right)\\ &= \frac{\sigma_{\mathrm{s}}^2}{n}\mathrm{tr}\left((1+\beta^2)\mathbf{F}\mathbf{F}^t-\mathbf{F}\mathbf{B}^t-\mathbf{B}\mathbf{F}^t+\mathbf{B}\mathbf{B}^t\right)\\ &\overset{(a)}{=} \frac{\sigma_{\mathrm{s}}^2}{n}\left((1+\beta^2)\mathrm{tr}(\mathbf{F}\mathbf{F}^t)-\mathrm{tr}(\mathbf{F}\mathbf{B}^t)-\mathrm{tr}(\mathbf{B}\mathbf{F}^t)+\mathrm{tr}(\mathbf{B}\mathbf{B}^t)\right)\\ &\overset{(b)}{=} \frac{\sigma_{\mathrm{s}}^2}{n}\left((1+\beta^2)\mathrm{tr}(\mathbf{F}^t\mathbf{F})-\mathrm{tr}(\mathbf{B}^t\mathbf{F})-\mathrm{tr}(\mathbf{F}^t\mathbf{B})+\mathrm{tr}(\mathbf{B}^t\mathbf{B})\right)\\ &\overset{(c)}{=} \frac{\sigma_{\mathrm{s}}^2}{n}\left((1+\beta^2)\mathrm{tr}(\mathbf{F}^t\mathbf{F})-2\underbrace{\mathrm{tr}(\mathbf{F}^t\mathbf{B})}_{n}+\underbrace{\mathrm{tr}(\mathbf{B}^t\mathbf{B})}_{(1+\beta^2)n}\right)\\ &= \frac{\sigma_{\mathrm{s}}^2}{n}\left((1+\beta^2)\mathrm{tr}(\mathbf{F}^t\mathbf{F})+(\beta^2-1)n\right)\end{aligned}$$

where (a) follows from linearity of trace, (b) follows from commutativity of trace of matrices ($\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$), (c) follows using $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{A}^t)$. Finally, we use the relation $\mathbf{F}^t\mathbf{B} = \mathbf{I}_n$ to deduce that $\mathbf{F}^t\mathbf{F} = (\mathbf{B}^t\mathbf{B})^{-1}$ which gives $\mathrm{tr}(\mathbf{F}^t\mathbf{F}) = \frac{n}{1+\beta^2}$. We get then $\varepsilon_{eq} = \sigma_{\mathrm{s}}^2\beta^2 = \varepsilon_{\min}$.

Appendix B
Modified Cassel's Algorithm

In this appendix we provide a modified Cassels's algorithm to solve the Inhomogenoues Diophantine Approximation in (30). The algorithm requires as inputs: the real values $y' = -\frac{\tilde{y}}{\gamma}$, $\beta' = \frac{\beta}{\gamma}$ and the shaping limit $\mathcal{A}_t$ defined given the original codebook $\mathcal{C}$ and the network code vector $\mathbf{a}$. The algorithm outputs the pair $(\hat{t}, \hat{k}) \in (\mathcal{A}_t, \mathbb{N})$ as the best approximation of the real β' given the additive shift y'. The constraint in line (5) allows to restrict the search in the finite set $\mathcal{C}_f$.

```
1: $\eta_1 = -1$; $\eta_0 = \beta'$; $\zeta_1 = -y'$;
2: $t_0 = 0$; $t_1 = 1$; $T_1 = 0$;
3: $k_0 = 1$; $k_1 = 0$; $K_1 = 0$;
4: $n = 2$
5: while $\eta_{n-1} \neq 0 \wedge \zeta_{n-1} \neq 0 \wedge T_{n-1} \in \mathcal{A}_t$ do
6:     $a_n = \left\lfloor \frac{-\eta_{n-2}}{\eta_{n-1}} \right\rfloor$;
7:     $t_n = t_{n-2} + a_n t_{n-1}$; $k_n = k_{n-2} + a_n k_{n-1}$;
8:     $\eta_n = \eta_{n-2} + a_n \eta_{n-1}$;
9:     if $K_{n-1} \leq k_{n-1}$ then
10:        $b_n = \left\lfloor \frac{-\zeta_{n-1} - \eta_{n-2}}{\eta_{n-1}} \right\rfloor$;
11:        $T_n = T_{n-1} + t_{n-2} + b_n t_{n-1}$; $K_n = K_{n-1} + k_{n-2} + b_n k_{n-1}$;
12:        $\zeta_n = \zeta_{n-1} + \eta_{n-2} + b_n \eta_{n-1}$;
13:    else
14:        $T_n = T_{n-1} - t_{n-1}$; $K_n = K_{n-1} - k_{n-1}$;
15:        $\zeta_n = \zeta_{n-1} - \eta_{n-1}$;
16:    end if
17:    $n = n + 1$;
18: end while
19: $\hat{t} = T_n$;
20: $\hat{k} = K_n$;
```

References

[1] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.

[2] C. Feng, D. Silva, and F. R. Kschischang, "An algebraic approach to physical-layer network coding," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7576–7596, Nov. 2013.

[3] A. Mejri and G. R.-B. Othman, "Practical physical layer network coding in multi-sources relay channels via the compute-and-forward," in *Proc. Wireless Commun. Netw. Conf. Workshops*, 2013, pp. 166–171.

[4] A. Mejri and G. R.-B. Othman, "Bidirectional relaying via network coding: Design algorithm and performance evaluation," in *Proc. Int. Conf. Telecommun.*, 2013, pp. 1–5.

[5] A. Mejri, G. R.-B. Othman, and J. C Belfiore, "Lattice decoding for the compute-and-forward protocol," in *Proc. 3rd Int. Conf. Commun. Netw.*, 2012, pp. 1–8.

[6] J.-C. Belfiore and C. Ling, "The flatness factor in lattice network coding: Design criterion and decoding algorithm," in *Proc. Int. Zurich Semin. Commun.*, 2012.

[7] A. Mejri and G. R.-B. Othman, "Map decoder for physical-layer network coding using lattice sphere decoding," in *Proc. Int. Conf. Telecommun.*, 2014, pp. 67–71.

[8] M. P. Wilson, K. Narayanan, H. D. Pfister, and A. Sprintson, "Joint physical layer coding and network coding for bidirectional relaying," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5641–5654, Nov. 2010.

[9] B. Nazer and M. Gastpar, "Reliable physical layer network coding," *IEEE Trans. Inf. Theory*, vol. 99, no. 3, pp. 438–460, Mar. 2011.

[10] G. D. Forney, M. D. Trott, and S.-Y. Chung, "Sphere-bound-achieving coset codes and multilevel coset codes," *IEEE Trans. Inf. Theory*, vol. 46, no. 3, pp. 820–850, May 2000.

[11] W. Banaszczyk, "New bounds in some transference theorems in the geometry of numbers," *Mathematische Annalen*, vol. 296, no. 4, pp. 625–635, 1993.

[12] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on gaussian measure," in *Proc. Annu. Symp. Found. Comput. Sci.*, Italy, 2004, pp. 371–381.

[13] H. El Gamal, G. Caire, and M.-O. Damen, "Lattice coding and decoding achieve the optimal diversity-multiplexing tradeoff of mimo channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 968–985, Jun. 2004.

[14] H. Cohen, *A Course in Computational Algebraic Number Theory*. New York, NY, USA: Springer-Verlag, 1993.

[15] F. Lazebnik, "On systems of linear diophantine equations," *Math. Mag.*, vol. 69, no. 4, pp. 261–266, 1996.

[16] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. Cambridge, MA, USA: The MIT Press, 2009.

[17] I. V. L. Clarkson, "Approximation of Linear Forms by Lattice Points with Applications to Signal Processing," Ph.D. dissertation, Australian Nat. Univ., Canberra, ACT, Australia, 1997.

[18] J. W. S. Cassel, *An Introduction to Diophantine Approximation*. Cambridge, U.K.: Cambridge Univ. Press, 1957.

[19] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1639–1642, Jul. 1999.

[20] F. Behnamfar, F. Alajaji, and T. Linder, "Performance analysis of map decoded space-time orthogonal block codes for non-uniform sources," in *Proc. IEEE Inf. Theory Workshop*, 2003, pp. 46–49.

[21] U. Erez, S. Litsyn, and R. Zamir, "Lattices which are good for (almost) everything," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3401–3416, Oct. 2005.

Asma Mejri was born in Tunis, Tunisia, in 1986. She received the engineering and M.Sc. degrees in telecommunications from the Higher School of Communications of Tunis (Sup'Com) both in 2010. She received the Ph.D. degree in communications and electronics from Telecom-ParisTech, France, in 2013. She is currently a Post-Doctoral Researcher at Telecom-ParisTech. Her research interests are in wireless communications, decoding for MIMO systems and network coding. She is one of the recipient of the Entrepreneurship Project Award at the Entrepreneurship Challenge organized by Sup'Com in 2009 and the Co-recipient of the Best Paper Award in The Third International Conference on Communications and Networking, Tunisia, 2012.

Ghaya Rekaya-Ben Othman was born in Tunis, Tunisia, in 1977. She received the degree in electrical engineering from ENIT, Tunisia, in 2000, and the Ph.D. degree from the Ecole Nationale Supérieure des Télécommunications (ENST) Paris, France, in 2004. In 2005, she joined, the Department of Communications and Electronics, Telecom-ParisTech (ex-ENST) as an Assistant Professor. Since 2012, she has been a Full Professor at Telecom-ParisTech. She is the recipient of the City of Paris Award of the Best Woman Scientist in 2007 and Co-recipient of the Best Paper Award in the Third International Conference on Communications and Networking, Tunisia 2012. Her research interests are in spacetime coding and lattice reduction and decoding.