Compute-and-Forward on a Multiaccess Relay Channel: Coding and Symmetric-Rate Optimization

Mohieddine El Soussi, Abdellatif Zaidi, and Luc Vandendorpe

***Abstract*—We consider a system in which two users communicate with a destination with the help of a half-duplex relay. Based on the compute-and-forward scheme, we develop and evaluate the performance of coding strategies that are of network coding spirit. In this framework, instead of decoding the users' information messages, the destination decodes two integer-valued linear combinations that relate the transmitted codewords. Two decoding schemes are considered. In the first one, the relay computes one of the linear combinations and then forwards it to the destination. The destination computes the other linear combination based on the direct transmissions. In the second one, accounting for the side information available at the destination through the direct links, the relay compresses what it gets using lattice-based Wyner-Ziv compression and conveys it to the destination. The destination then computes the two linear combinations, locally. For both coding schemes, we discuss the design criteria, and derive the allowed symmetric-rate. Next, we address the power allocation and the selection of the integer-valued coefficients to maximize the offered symmetric-rate; an iterative coordinate descent method is proposed. The analysis shows that the first scheme can outperform standard relaying techniques in certain regimes, and the second scheme, while relying on feasible structured lattice codes, can at best achieve the same performance as regular compress-and-forward for the multiaccess relay network model that we study. The results are illustrated through some numerical examples.**

***Index Terms*—Compute-and-forward, network coding, lattice codes, relay channel, geometric programming, mixed-integer quadratic programming.**

I. Introduction

NETWORK coding was introduced by Ahlswede *et al.* in [1] for wired networks. It refers to each intermediate node sending out a function of the packets that it receives, an operation which is more general than simple routing [2], [3]. In linear network coding, intermediate nodes compute and send out linear combinations over an appropriate finite field of the packets that they receive. In general, the function does not need to be linear. Although they are generally suboptimal for general wireline networks, linear network codes have been shown optimum for multicasting [4], [5]. Moreover they have appreciable features, in particular simplicity (e.g., see [6], [7] and references therein). For these reasons, most of the research on network coding has focused on linear codes.

The development of efficient network coding techniques for wireless networks is more involved than for wired network coding, essentially because of fading, interference and noise effects. For general wireless networks, the quantize-map-and-forward scheme of [8] and the more general noisy network coding scheme of [9] can be seen as interesting and efficient extensions for wireless settings of the original network coding principle. However, quantize-map-and-forward and noisy network coding are based on random coding arguments. For wireless networks, efficient linear network coding techniques make use of structured codes, and in particular lattices [10]. Lattices play an important role in network coding for diverse network topologies, such as the two-way relay channel [11], [12], the Gaussian network [13] and others.

Recently, Nazer and Gastpar propose and analyse a scheme in which receivers decode finite-field linear combinations of transmitters' messages, instead of the messages themselves. The scheme is called "Compute-and-forward" (CoF) [13], and can be implemented with or without the presence of relay nodes. In this setup, a receiver that is given a sufficient number of linear combinations recovers the transmitted messages by solving a system of independent linear equations that relate the transmitted symbols. Critical in this scheme, however, is that the coefficients of the equations to decode must be integer-valued. This is necessitated by the fact that a combination of codewords should itself be a codeword so that it is decodable. Lattice codes have exactly this property, and are thus good candidates for implementing compute-and-forward.

Compute-and-forward is a promising scheme for network coding in wireless networks. However, the problem of selecting the integer coefficients optimally, i.e., in a manner that allows to recover the sent codewords from the decoded equations and, at the same time, maximizes the transmission rate is not an easy task. As shown by Nazer and Gastpar [13], the compute-and-forward optimally requires a match between the channel gains and the desired integer coefficients. However, in real communication scenarios, it is unlikely that the channels would produce gains that correspond to integer values. This problem has been addressed in [14], where the authors develop a superposition strategy to mitigate the non-integer channel coefficients penalty. The selection of which integer combinations to decode is then a crucial task to be performed by the receivers. While it can be argued that linear combinations that are recovered at the same physical entity can always be chosen appropriately, i.e., in a way enabling system inversion to solve for the sent codewords, selecting these linear combinations in a distributed manner, i.e., at physically separated nodes, is less easy to achieve. By opposition to

Manuscript received March 20, 2013; revised August 25, 2013; accepted January 8, 2014. The associate editor coordinating the review of this paper and approving it for publication was A. Kwasinski.

This work has been supported in part by the IAP BESTCOM project funded by BELSPO, FP7 NEWCOM# and the SCOOP project. The material in this paper has been presented in part at the IEEE International Conference on Communication, Ottawa, Canada, June 2012 and the IEEE International Conference on Communication, Budapest, Hungary, June 2013.

M. El Soussi and L. Vandendorpe are with ICTEAM, Université catholique de Louvain, Place du Levant, 2, 1348 Louvain-la-Neuve, Belgium (e-mail: {mohieddine.elsoussi,luc.vandendorpe}@uclouvain.be).

A. Zaidi is with the Université Paris-Est Marne La Vallée, 77454 Marne la Vallée Cedex 2, France (e-mail: abdellatif.zaidi@univ-mlv.fr).

Digital Object Identifier 10.1109/TWC.2014.022614.130502

previous works, part of this paper focuses on this issue.

In this work, we consider communication over a two-user multiaccess relay channel. In this model, two independent users communicate with a destination with the help of a common relay node, as shown in Figure 1. The relay is assumed to operate in half-duplex mode.

A. Contributions

We establish two coding schemes for the multiaccess relay model that we study. The first coding scheme is based on compute-and-forward at the relay node. On this aspect, this strategy is conceptually similar to the compute-and-forward approach of Nazer and Gastpar [13]. The relay uses what it receives from the transmitters during the first transmission period to compute a linear combination with integer coefficients of the users' codewords. It then sends this combination to the destination during the second transmission period. In addition to the linear combination that it gets from the relay's transmission, the destination recovers the required second linear combination from what it gets directly from the transmitters, through the direct links. If the set of integer coefficients that are selected at the relay and destination are chosen appropriately, the destination can solve for the transmitted codewords.

In the second coding scheme both required linear integer combinations of the users' codewords are recovered locally at the destination. More specifically, the relay quantizes its output from the users' transmission during the first transmission period using lattice-based Wyner-Ziv compression [15]. In doing so, it accounts for the output at the destination during this transmission period as available side information at the decoder. Then, the relay sends the lossy version of its output to the destination during the second transmission period. On this aspect, the analysis is in part similar to the lattice-based Wyner-Ziv strategy of [15], [16]. The destination determines the two required linear combinations, as follows. It utilizes an appropriate combination of the output from the users' transmission during the first period and of the compressed version of the relay's output during the second period; from this combination, two independent linear combinations relating the users' codewords are recovered.

For the two coding schemes, we target the optimization of the transmitters and the relay powers, and of the integer coefficients of the linear combinations to maximize the achievable symmetric-rate. These optimization problems are NP hard. For the two coding schemes, we develop an iterative approach that finds the appropriate power and integer coefficients alternately. More specifically, we show that the problem of finding appropriate integer coefficients for a given set of powers has the same solution as a mixed integer quadratic programming (MIQP) problem with quadratic constraints. Also, we show that the problem of finding the appropriate power policy at the transmitters and relay for a given set of integer coefficients is a non-linear non-convex optimization problem. We formulate and solve this problem through geometric programming and a successive convex approximation approach [17].

Our analysis shows that, for certain channel conditions, the first scheme outperforms known strategies for this model that do not involve forms of network coding, such as those based on having the relay implement classic amplify-and-forward (AF), decode-and-forward (DF) or compress-and-forward (CF) relaying schemes. The second scheme offers rates that are at best as large as those offered by compress-and-forward for the multiaccess relay network that we study. However, this scheme relies on feasible structured lattice codes and utilizes linear receivers, and so, from a practical viewpoint it offers advantages over standard CF which is based on random binning arguments. We illustrate our results by means of some numerical examples. The analysis also shows the benefit obtained from allocating the powers and the integer coefficients appropriately.

B. Outline and Notation

An outline of the remainder of this paper is as follows. Section II describes in more details the communication model that we consider in this work. It also contains some preliminaries on lattices and known results from the literature for the setup under consideration where the relay uses standard techniques. In Section III, we describe our coding strategies and analyse the symmetric rates that are achievable using these strategies. Section IV is devoted to the optimization of the power values and the integer-valued coefficients for an objective function which is the symmetric-rate. Section V contains some numerical examples, and Section VI concludes the paper.

We use the following notations throughout the paper. Lowercase boldface letters are used to denote vectors, e.g., $\mathbf{x}$. Upper case boldface letters are used to denote matrices, e.g., $\mathbf{X}$. Calligraphic letters designate alphabets, i.e., $\mathcal{X}$. The cardinality of a set $\mathcal{X}$ is denoted by $|\mathcal{X}|$. For matrices, we use the notation $\mathbf{X} \in \mathbb{R}^{m\times n}$, $m, n \in \mathbb{N}$, to mean that $\mathbf{X}$ is an m-by-n matrix, i.e., with m rows and n columns, and its elements are real-valued. Also, we use $\mathbf{X}^T$ to designate the n-by-m matrix transpose of $\mathbf{X}$. We use $\mathbf{I}_n$ to denote the n-by-n identity matrix; and $\mathbf{0}$ to denote a matrix whose elements are all zeros (its size will be evident from the context). Similarly, for vectors, we write $\mathbf{x} \in \mathbb{A}^n$, e.g., $\mathbb{A} = \mathbb{R}$ or $\mathbb{A} = \mathbb{Z}$, to mean that $\mathbf{x}$ is a column vector of size n, and its elements are in $\mathbb{A}$. For a vector $\mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}\|$ designates the norm of $\mathbf{x}$ in terms of Euclidean distance; and for a scalar $x \in \mathbb{R}$, $|x|$ stands for the absolute value of x, i.e., $|x| = x$ if $x \geq 0$ and $|x| = -x$ if $x \leq 0$. For two vectors $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^n$, the vector $\mathbf{z} = \mathbf{x} \circ \mathbf{y} \in \mathbb{R}^n$ denotes the Hadamard product of $\mathbf{x}$ and $\mathbf{y}$, i.e., the vector whose ith element is the product of the ith elements of $\mathbf{x}$ and $\mathbf{y}$, i.e., $z_i = (\mathbf{x} \circ \mathbf{y})_i = x_i y_i$. Also, let $\det(\mathbf{x}, \mathbf{y})$ denote the determinant of the matrix formed by the vectors $\mathbf{x}$ and $\mathbf{y}$, i.e., $[\mathbf{x}\,\mathbf{y}]$. The Gaussian distribution with mean μ and variance σ^2 is denoted by $\mathcal{N}(\mu, \sigma^2)$. Finally, throughout the paper except where otherwise mentioned, logarithms are taken to base 2; and, for $x \in \mathbb{R}$, $\log^+(x) := \max\{\log(x), 0\}$.

II. Preliminaries and System Model

In this section, we first recall some basics on lattices, and then present the system model that we study and recall some known results from the literature, obtained through classic relaying, i.e., amplify-and-forward, decode-and-forward and compress-and-forward. The results given in Section II-C will be used later for comparison purposes in this paper.

A. Preliminaries on Lattices

Algebraically, an n-dimensional lattice Λ is a discrete additive subgroup of $\mathbb{R}^n$. Thus, if $\boldsymbol{\lambda}_1 \in \Lambda$ and $\boldsymbol{\lambda}_2 \in \Lambda$, then $(\boldsymbol{\lambda}_1 + \boldsymbol{\lambda}_2) \in \Lambda$ and $(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2) \in \Lambda$. A lattice can always be written in terms of a lattice generator matrix $\mathbf{G} \in \mathbb{R}^{n\times n}$

$$\Lambda = \{\boldsymbol{\lambda} = \mathbf{z}\mathbf{G} : \mathbf{z} \in \mathbb{Z}^{\mathrm{n}}\}. \tag{1}$$

A lattice quantizer $Q_\Lambda : \mathbb{R}^n \to \Lambda$ maps a point $\mathbf{x} \in \mathbb{R}^n$ to the nearest lattice point in Euclidean distance, i.e.,

$$Q_\Lambda(\mathbf{x}) = \arg\min_{\boldsymbol{\lambda}\in\Lambda} \|\mathbf{x} - \boldsymbol{\lambda}\|. \tag{2}$$

The Voronoi region $\mathcal{V}(\boldsymbol{\lambda})$ of $\boldsymbol{\lambda} \in \Lambda$ is the set of all points in $\mathbb{R}^n$ that are closer to $\boldsymbol{\lambda}$ than to any other lattice point, i.e.,

$$\mathcal{V}(\boldsymbol{\lambda}) = \{\mathbf{x} \in \mathbb{R}^n \,:\, Q_\Lambda(\mathbf{x}) = \boldsymbol{\lambda}\}. \tag{3}$$

The *fundamental Voronoi region* $\mathcal{V}$ of lattice Λ is the Voronoi region $\mathcal{V}(\mathbf{0})$, i.e., $\mathcal{V} = \mathcal{V}(\mathbf{0})$. The modulo reduction with respect to Λ returns the quantization error, i.e.,

$$[\mathbf{x}] \bmod \Lambda = \mathbf{x} - Q_\Lambda(\mathbf{x}) \in \mathcal{V}. \tag{4}$$

The *second moment* σ_Λ^2 quantifies per dimension the average power for a random variable that is uniformly distributed over $\mathcal{V}$, i.e.,

$$\sigma_\Lambda^2 = \frac{1}{n\mathrm{Vol}(\mathcal{V})}\int_{\mathcal{V}} \|\mathbf{x}\|^2 d\mathbf{x} \tag{5}$$

where $\mathrm{Vol}(\mathcal{V})$ is the volume of $\mathcal{V}$. The *normalized second moment* of Λ is defined as

$$G(\Lambda) = \frac{\sigma_\Lambda^2}{\mathrm{Vol}(\mathcal{V})^{2/n}}. \tag{6}$$

A lattice Λ is said to be nested into another lattice Λ_{FINE} if $\Lambda \subseteq \Lambda_{\mathrm{FINE}}$, i.e., every point of Λ is also a point of Λ_{FINE}. We refer to Λ as the coarse lattice and to Λ_{FINE} as the fine lattice. Also, given two nested lattices $\Lambda \subseteq \Lambda_{\mathrm{FINE}}$, the set of all the points of the fine lattice Λ_{FINE} that fall in the fundamental Voronoi region $\mathcal{V}$ of the coarse lattice Λ form a codebook

$$\mathcal{C} = \Lambda_{\mathrm{FINE}} \cap \mathcal{V} = \{\mathbf{x} = \boldsymbol{\lambda} \bmod \Lambda,\ \boldsymbol{\lambda} \in \Lambda_{\mathrm{FINE}}\}. \tag{7}$$

The rate of this codebook is

$$R = \frac{1}{n}\log_2(|\mathcal{C}|). \tag{8}$$

Finally, the mod operation satisfies the following properties:

$$\begin{aligned}
&(\mathrm{P1})\ [[\mathbf{x}] \bmod \Lambda + \mathbf{y}] \bmod \Lambda = [\mathbf{x} + \mathbf{y}] \bmod \Lambda,\ \forall\ \mathbf{x},\ \mathbf{y} \in \mathbb{R}^n\\
&(\mathrm{P2})\ [k([\mathbf{x}] \bmod \Lambda)] \bmod \Lambda = [k\mathbf{x}] \bmod \Lambda,\ \forall\ k \in \mathbb{Z},\ \mathbf{x} \in \mathbb{R}^n\\
&(\mathrm{P3})\ \gamma([\mathbf{x}] \bmod \Lambda) = [\gamma\mathbf{x}] \bmod \gamma\Lambda,\ \forall\ \gamma \in \mathbb{R},\ \mathbf{x} \in \mathbb{R}^n.
\end{aligned} \tag{9}$$

B. System Model

We consider the communication system shown in Figure 1. Two transmitters A and B communicate with the destination with the help of a common relay. Transmitter A and B want to transmit the messages $W_a \in \mathcal{W}_a$, and $W_b \in \mathcal{W}_b$ to the destination reliably, in $2n$ uses of the channel. At the end of the transmission, the destination guesses the pair of transmitted messages using its output. Let R_a be the transmission rate of message W_a and R_b be the transmission rate of message W_b. We concentrate on the *symmetric* rate case, i.e., $R_a = R_b = R$, or equivalently, $|\mathcal{W}_a| = |\mathcal{W}_b| = 2^{2nR}$. We measure the system performance in terms of the allowed achievable symmetric-rate $R_{\mathrm{sym}} = R_a = R_b = R$. Also, we divide the transmission time into two transmission periods with each of length n channel uses. The relay operates in a half-duplex mode.

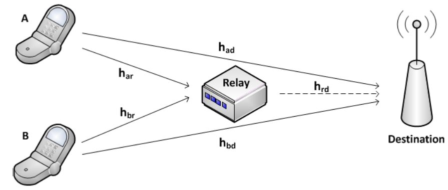

Fig. 1. Multiple-access channel with a half-duplex relay

During the first transmission period, Transmitter A encodes its message $W_a \in [1, 2^{2nR}]$ into a codeword $\mathbf{x}_a$ and sends it over the channel. Similarly, Transmitter B encodes its message $W_b \in [1, 2^{2nR}]$ into a codeword $\mathbf{x}_b$ and sends it over the channel. Let $\mathbf{y}_r$ and $\mathbf{y}_d$ be the signals received respectively at the relay and destination during this period. These signals are given by

$$\begin{aligned}
\mathbf{y}_r &= h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_b + \mathbf{z}_r\\
\mathbf{y}_d &= h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d,
\end{aligned} \tag{10}$$

where h_{ad} and h_{bd} are the channel gains on the links transmitters-to-destination, h_{ar} and h_{br} are the channel gains on the links transmitters-to-relay, and $\mathbf{z}_r$ and $\mathbf{z}_d$ are additive background noises at the relay and destination.

During the second transmission period, the relay sends a codeword $\tilde{\mathbf{x}}_r$ to help both transmitters. During this period, the destination receives

$$\tilde{\mathbf{y}}_d = h_{rd}\tilde{\mathbf{x}}_r + \tilde{\mathbf{z}}_d, \tag{11}$$

where h_{rd} is the channel gain on the link relay-to-destination, and $\tilde{\mathbf{z}}_d$ is additive background noise.

Throughout the paper, we assume that all channel gains are real-valued, fixed and known to all the nodes in the network; and the noises at the relay and destination are independent among each others, and independently and identically distributed (i.i.d) Gaussian, with zero mean and variance N. Furthermore, we consider the following individual constraints on the transmitted power (per codeword),

$$\begin{aligned}
&\mathbb{E}[\|\mathbf{x}_a\|^2] = n\beta_a^2 P \le nP_a, \qquad \mathbb{E}[\|\mathbf{x}_b\|^2] = n\beta_b^2 P \le nP_b,\\
&\mathbb{E}[\|\tilde{\mathbf{x}}_r\|^2] = n\beta_r^2 P \le nP_r,
\end{aligned} \tag{12}$$

where $P_a \ge 0$, $P_b \ge 0$ and $P_r \ge 0$ are some constraints imposed by the system; $P \ge 0$ is given, and β_a, β_b and β_r are some scalars that can be chosen to adjust the actual transmitted powers, and are such that $0 \le |\beta_a| \le \sqrt{P_a/P}$, $0 \le |\beta_b| \le \sqrt{P_b/P}$ and $0 \le |\beta_r| \le \sqrt{P_r/P}$. For convenience, we will sometimes use the shorthand vector notation $\mathbf{h}_d = [h_{ad}, h_{bd}]^T$, $\mathbf{h}_r = [h_{ar},\ h_{br}]^T \in \mathbb{R}^2$ and $\boldsymbol{\beta} = [\beta_a, \beta_b, \beta_r]^T \in \mathbb{R}^3$, and the shorthand matrix notation $\mathbf{H} = [\mathbf{h}_d,\ \mathbf{h}_r]^T \in \mathbb{R}^{2\times 2}$. Also, we will find it useful to sometimes use the notation $\boldsymbol{\beta_s}$ to denote the vector composed of the first two components of vector $\boldsymbol{\beta}$, i.e., $\boldsymbol{\beta_s} = [\beta_a, \beta_b]^T$ – the subscript "s" standing for "sources". Finally, the signal-to-noise ratio will be denoted as $\mathrm{snr} = P/N$ in the linear scale, and by $\mathrm{SNR} = 10\log_{10}(\mathrm{snr})$ in decibels in the logarithmic scale.

C. Symmetric Rates Achievable Through Classic Relaying

In this section, we review some known results from the literature for the model we study. These results will be used for comparisons in Section V.

1) Amplify-and-Forward: The relay receives $\mathbf{y}_r$ as given by (10) during the first transmission period. It simply scales $\mathbf{y}_r$ to the appropriate available power and sends it to the destination during the second transmission period. That is, the relay outputs $\tilde{\mathbf{x}}_r = \gamma\ \mathbf{y}_r$, with $\gamma = \sqrt{\beta_r^2 \text{snr}/(1 + \text{snr}\,\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2)}$.

The destination estimates the transmitted messages from its output vectors $(\mathbf{y}_d, \tilde{\mathbf{y}}_d)$. Using straightforward algebra, it can be shown [18] that this results in the following achievable sum rate

$$R_{\text{sum}}^{\text{AF}} = \max \frac{1}{4} \log\left(\det\left(\mathbf{I}_2 + \beta_a^2 \text{snr}(\mathbf{h}_a \mathbf{h}_a^T) + \beta_b^2 \text{snr}(\mathbf{h}_b \mathbf{h}_b^T)\right)\right), \tag{13}$$

where the vectors are given by $\mathbf{h}_i = [h_{id},\ h_{ir} h_{rd} \gamma / (\sqrt{1 + \gamma^2 |h_{rd}|^2})]^T$ for $i = a, b$, and the maximization is over $\boldsymbol{\beta}$.

The achievable sum rate (13) does not require the two users to transmit at the same rate. Recall that, for a symmetric rate point to be achievable, both transmitters must be able to communicate their messages with at least that rate. Under the constraint of *symmetric-rate*, it can be shown in a straightforward manner [13] that the following symmetric-rate is achievable with the relay operating on the amplify-and-forward mode,

$$\begin{aligned} R_{\text{sym}}^{\text{AF}} = \max \frac{1}{4} \min \Bigg\{ & \log\left(\det\left(\mathbf{I}_2 + \beta_a^2 \text{snr}(\mathbf{h}_a \mathbf{h}_a^T)\right)\right), \\ & \log\left(\det\left(\mathbf{I}_2 + \beta_b^2 \text{snr}(\mathbf{h}_b \mathbf{h}_b^T)\right)\right), \\ & \frac{1}{2} \log\left(\det\left(\mathbf{I}_2 + \beta_a^2 \text{snr}(\mathbf{h}_a \mathbf{h}_a^T) + \beta_b^2 \text{snr}(\mathbf{h}_b \mathbf{h}_b^T)\right)\right) \Bigg\}. \end{aligned} \tag{14}$$

2) Decode-and-Forward: At the end of the first transmission period, the relay decodes the message pair (W_a, W_b) and then, during the second transmission period, sends a codeword $\tilde{\mathbf{x}}_r$ that is independent of $\mathbf{x}_a$ and $\mathbf{x}_b$ and carries both messages. The relay employs superposition coding and splits its power among the two messages. It can be shown easily that the resulting achievable sum rate is given by [19]

$$\begin{aligned} R_{\text{sum}}^{\text{DF}} = \max \frac{1}{4} \min \Bigg\{ & \log\left(1 + \text{snr}\,\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right), \\ & \log\left(1 + \text{snr}\,\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right) + \log\left(1 + \text{snr}\,|h_{rd}|^2 \beta_r^2\right) \Bigg\}, \end{aligned} \tag{15}$$

where the maximization is over $\boldsymbol{\beta}$. Under the constraint of *symmetric-rate*, it can be shown that the following symmetric-rate is achievable with the relay operating on the decode-and-forward mode,

$$R_{\text{sym}}^{\text{DF}} = \max \frac{1}{4} \min \left\{ R(\mathbf{h}_r), R(\mathbf{h}_d) + \frac{1}{2} \log\left(1 + \text{snr}|h_{rd}|^2 \beta_r^2\right) \right\}, \tag{16}$$

where

$$\begin{aligned} R(\mathbf{h}_i) = \min \Bigg\{ & \log\left(1 + \text{snr}|h_{ai}|^2 \beta_a^2\right), \log\left(1 + \text{snr}|h_{bi}|^2 \beta_b^2\right), \\ & \frac{1}{2} \log\left(1 + \text{snr}\,\|\boldsymbol{\beta_s} \circ \mathbf{h}_i\|^2\right) \Bigg\}. \end{aligned} \tag{17}$$

3) Compress-and-Forward: At the end of the first transmission period, the relay quantizes the received $\mathbf{y}_r$ using Wyner-Ziv compression [20], accounting for the available side information $\mathbf{y}_d$ at the destination. It then sends an independent codeword $\tilde{\mathbf{x}}_r$ that carries the compressed version of $\mathbf{y}_r$. The destination guesses the transmitted messages using its output from the direct transmission along with the lossy version of the output of the relay that is recovered during the second transmission period. It can be shown that the resulting achievable sum rate is given by [19], [21],

$$R_{\text{sum}}^{\text{CF}} \quad = \quad \max \frac{1}{4} R^{\text{CF}}, \tag{18}$$

where

$$\begin{aligned} R^{\text{CF}} = \log\Bigg(& \frac{\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)\left(1 + D/N + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{(1 + D/N)} \\ & - \frac{\text{snr}^2\left((\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T (\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{(1 + D/N)} \Bigg), \end{aligned} \tag{19}$$

the maximization is over $\boldsymbol{\beta_s}$ and $D \geq 0$, where D is the distortion due to Wyner-Ziv compression, which is given by,

$$D = \frac{N^2\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2\left(\text{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T (\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)}. \tag{20}$$

Under the constraint of *symmetric-rate*, it can be shown that the following symmetric-rate is achievable with the relay operating on the compress-and-forward mode,

$$\begin{aligned} R_{\text{sym}}^{\text{CF}} = \max \frac{1}{4} \min \Bigg\{ & \log\left(1 + \text{snr}|h_{ad}|^2 \beta_a^2 + \frac{\text{snr}|h_{ar}|^2 \beta_a^2}{1 + D/N}\right), \\ & \log\left(1 + \text{snr}|h_{bd}|^2 \beta_b^2 + \frac{\text{snr}|h_{br}|^2 \beta_b^2}{1 + D/N}\right), \frac{1}{2} R^{\text{CF}} \Bigg\} \end{aligned} \tag{21}$$

III. Network Coding Strategies

In this section, we develop two coding strategies that are both based on the compute-and-forward strategy of [13]. The two strategies differ essentially through the operations implemented by the relay. In the first strategy, the relay computes an appropriate linear combination that relates the transmitters' codewords and forwards it to the destination. The destination computes the other required linear combination from what it gets through the direct links. In the second strategy, the relay sends a lossy version of its outputs to the destination, obtained through lattice-based Wyner-Ziv compression [15], [16]. The destination then obtains the desired two linear combinations locally, by using the recovered output from the relay and the output obtained directly from the transmitters.

A. Compute-and-Forward at the Relay

The following proposition provides an achievable symmetric-rate for the multiaccess relay model that we study.

Proposition 1: For any set of channel vector $\mathbf{h} = [h_{ar}, h_{br}, h_{ad}, h_{bd}, h_{rd}]^T \in \mathbb{R}^5$, the following symmetric-rate is achievable for the multiaccess relay model that we study:

$$R_{\text{sym}}^{\text{CoF}} = \max \frac{1}{4} \min \Bigg\{ \log^+ \left(\left(\|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta_s} \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2} \right)^{-1} \right), \\ \log^+ \left(\left(\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2} \right)^{-1} \right), \\ \log \left(1 + \frac{P|h_{rd}|^2 \beta_r^2}{N} \right) \Bigg\}, \tag{22}$$

where the maximization is over $\boldsymbol{\beta}$ and over the integer coefficients $\mathbf{k} \in \mathbb{Z}^2$ and $\mathbf{t} \in \mathbb{Z}^2$ such that $\det(\mathbf{k}, \mathbf{t}) \neq 0$.

In the coding scheme of Proposition 1, the relay first computes a linear combination with integer coefficients of the transmitters codewords and then forwards this combination to the destination during the second transmission period. The destination computes another linear combination that relates these codewords using its output from the direct transmissions. With an appropriate choice of the integer-valued coefficients of the combinations, the destination obtains two equations that can be solved for the transmitted codewords.

Remark 1: The scheme of Proposition 1 is conceptually similar to the compute-and-forward approach of Nazer and Gastpar [13]. This can be seen by noticing that the multiaccess relay network that we study in this paper can be thought as being a Gaussian network with two users, two relays and a central processor. The first relay in the equivalent network plays the role of the relay in our MARC model, and the second relay in the equivalent network plays the role of the destination in our MARC model. The second relay in the equivalent network is connected with the central processor, which is the destination itself, via a bit-pipe of infinite capacity. Furthermore, it can be seen that, in the equivalent model, the bit-pipe with infinite capacity can be replaced with one that has the same capacity as that of the relay-to-destination link. This follows since the two equations that are forwarded to the central processor have the same rate. Hence, in what follows, we outline the encoding procedures at the transmitters and relay, and the decoding procedures at the relay and destination. Then the rate of Proposition 1 can be readily obtained by viewing the MARC network that we study as described in this remark and applying the result of [13, Theorem 5].

Let Λ be an n-dimensional lattice that is good for quantization in the sense of [22] and whose second moment is equal to P, i.e., $\sigma_\Lambda^2 = P$. We denote by $G(\Lambda)$ and $\mathcal{V}$ respectively the normalized second moment and the fundamental Voronoi region of lattice Λ. Also, let $\Lambda_{\text{FINE}} \supseteq \Lambda$ be a lattice that is good for AWGN in the sense of [13, Definition 23], and chosen such that the codebook $\mathcal{C} = \Lambda_{\text{FINE}} \cap \mathcal{V}$ be of cardinality 2^{2nR} [15]. We designate by $\mathcal{V}_{\text{FINE}}$ the fundamental Voronoi region of lattice Λ_{FINE}. The coarse lattice Λ and the fine lattice Λ_{FINE} form a pair of nested lattices that we will utilize as a structured code.

Let (W_a, W_b) be the pair of messages to be transmitted. Let $\mathbf{u}_a$, $\mathbf{u}_b$ and $\mathbf{u}_r$ be some dither vectors that are drawn independently and uniformly over $\mathcal{V}$ and known by all nodes in the network. Since the codebook $\mathcal{C}$ is of size $2^{2nR} = |\mathcal{W}_a|$, there exists a one-to-one mapping function $\phi_a(\cdot)$ between the set of messages $\{W_a\}$ and the nested lattice code $\mathcal{C}$. Similarly, there exists a one-to-one mapping function $\phi_b(\cdot)$ between the set of messages $\{W_b\}$ and the nested lattice code $\mathcal{C}$. Let $\mathbf{v}_a = \phi_a(W_a)$ and $\mathbf{v}_b = \phi_b(W_b)$, where $\mathbf{v}_a \in \mathcal{C}$ and $\mathbf{v}_b \in \mathcal{C}$.

During the first transmission period, to transmit message W_a, Transmitter A sends

$$\mathbf{x}_a = \beta_a \left([\mathbf{v}_a - \mathbf{u}_a] \bmod \Lambda \right), \tag{23}$$

for some $\beta_a \in \mathbb{R}$ such that $0 \leq |\beta_a| \leq \sqrt{P_a/P}$; and to transmit message W_b, Transmitter B sends

$$\mathbf{x}_b = \beta_b \left([\mathbf{v}_b - \mathbf{u}_b] \bmod \Lambda \right), \tag{24}$$

where $0 \leq |\beta_b| \leq \sqrt{P_b/P}$. The scalars β_a and β_b are chosen so as to adjust the transmitters' powers during this period.
The relay decodes correctly an integer combination $\mathbf{e}_2 = k_a\mathbf{v}_a + k_b\mathbf{v}_b$, [13, Theorem 5], from what it receives during the first transmission period. It then sends

$$\tilde{\mathbf{x}}_r = \beta_r \left([k_a\mathbf{v}_a + k_b\mathbf{v}_b - \mathbf{u}_r] \bmod \Lambda \right) \tag{25}$$

during the second transmission period, where the scalar β_r is chosen so as to adjust its transmitted power during this period. Similar to the relay, the destination computes an integer combination $\mathbf{e}_1 = t_a\mathbf{v}_a + t_b\mathbf{v}_b$ from what it receives during the first transmission period. During the second transmission period, the destination can obtain a second integer combination $\mathbf{e}_2 = k_a\mathbf{v}_a + k_b\mathbf{v}_b$ of the users' codewords using its output component from the relay.

<u>Summary:</u> Over the entire transmission time, the destination collects two linear combinations with integer coefficients that relate the users' codewords, as

$$\begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} = \begin{bmatrix} t_a & t_b \\ k_a & k_b \end{bmatrix} \begin{bmatrix} \mathbf{v}_a \\ \mathbf{v}_b \end{bmatrix}. \tag{26}$$

Now, since the integer-valued matrix in (26) is invertible (recall that the integer-valued coefficients are chosen such that $\det(\mathbf{k}, \mathbf{t}) \neq 0$), the destination obtains the transmitted codewords by solving (26).
The destination is able to recover the messages $\hat{W}_a$ and $\hat{W}_b$ reliably if the message rate is less or equal to the computational rate $R_{\text{sym}}^{\text{CoF}}$ [13]. Hence, it can decode the transmitters' codewords correctly at the transmission symmetric-rate $R_{\text{sym}}^{\text{CoF}}$ given in (22). We should note that we can replace β_r^2 by its maximum value P_r/P without altering the symmetric-rate $R_{\text{sym}}^{\text{CoF}}$.

Although the achievable symmetric-rate in Proposition 1 requires the relay to only decode a linear combination of the codewords transmitted by the users, not the individual messages, this can be rather a severe constraint in certain cases. In the following section, the relay only compresses its output and sends it to the destination. The computation of the desired linear combinations of the users' codewords takes place at the destination, locally.

B. Compress-and-Forward at the Relay and Compute at the Destination

The following proposition provides an achievable symmetric-rate for the multiaccess relay model that we study.

***Proposition** 2:* For any set of channel vector $\mathbf{h} = [h_{ar}, h_{br}, h_{ad}, h_{bd}, h_{rd}]^T \in \mathbb{R}^5$, the following symmetric-rate is achievable:

$$R^{\text{CoD}}_{\text{sym}} = \max \frac{1}{4} \min\left\{ \log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T\boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T\mathbf{n}_d}\right), \log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T\boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T\mathbf{n}_d}\right)\right\}, \tag{27}$$

where $\boldsymbol{\alpha}_t = [\alpha_{1t}, \alpha_{2t}]^T$ and $\boldsymbol{\alpha}_k = [\alpha_{1k}, \alpha_{2k}]^T \in \mathbb{R}^2$ are some inflation factors with $\boldsymbol{\alpha}_t = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1}\mathbf{G}\mathbf{t}$, $\boldsymbol{\alpha}_k = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1}\mathbf{G}\mathbf{k}$, $\mathbf{G} = [(\boldsymbol{\beta}_s \circ \mathbf{h}_d), (\boldsymbol{\beta}_s \circ \mathbf{h}_r)]^T \in \mathbb{R}^{2\times 2}$, $\mathbf{N}_d = [1/\text{snr}, 0; 0, 1/\text{snr} + D/P] \in \mathbb{R}^{2\times 2}$, $\mathbf{n}_d = [1, 1 + D/N]^T \in \mathbb{R}^2$, and D is given by

$$D \geq \frac{N^2\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2\left(\text{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T(\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)}, \tag{28}$$

and the maximization is over $\boldsymbol{\alpha}_t$, $\boldsymbol{\alpha}_k$, $\boldsymbol{\beta_s}$, and over the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ such that $\det(\mathbf{k}, \mathbf{t}) \neq 0$.

In the coding scheme that we use for the proof of Proposition 2, the relay conveys a lossy version of its output to the destination during the second transmission period. In doing so, it accounts for the available side information at the destination, i.e., what the destination has received during the first transmission period. The destination computes two linearly independent combinations that relate the users' codewords using its outputs from both transmission periods, as follows. The destination combines appropriately the obtained lossy version of the relay's output (that it recovered from the relay's transmission during the second transmission period) and from what it received during the first transmission period. Then it computes two linearly independent combinations with integer coefficients that relate the users' codewords.

Proof: The transmission scheme and the encoding procedures at the transmitters are similar to those of Proposition 1. Therefore, for brevity, they will only be outlined. We will insist more on aspects of the coding scheme that are inherently different from those of the coding scheme of Proposition 1. We should note that the analysis, shown below, is in part similar to [15], [16].

In addition to the codebook described in Section III-A for the transmitters, we construct a channel codebook for the relay and a Quantization/Compression codebook as in [16]. Let Λ_{r} be an n-dimensional lattice that is good for quantization in the sense of [22] and whose second moment is equal to P_r, i.e., $\sigma^2_{\Lambda_{\text{r}}} = P_r$. We denote by $\mathcal{V}_{\text{r}}$ the fundamental Voronoi region of lattice Λ_{r}. Also, let $\Lambda_{\text{r}_{\text{FINE}}} \supseteq \Lambda_{\text{r}}$ be a lattice that is good for AWGN in the sense of [13, Definition 23], and chosen such that the codebook $\mathcal{C}_{\text{r}} = \Lambda_{\text{r}_{\text{FINE}}} \cap \mathcal{V}_{\text{r}}$ be of cardinality $2^{2nR_{\text{r}}}$ [15]. We designate by $\mathcal{V}_{\text{r}_{\text{FINE}}}$ the fundamental Voronoi region of lattice $\Lambda_{\text{r}_{\text{FINE}}}$. Also, we associate each compression index $\{q\} \in [1, 2^{2nR_r}]$ with a codeword $\mathbf{v}_r = \phi_r(q)$, where $\phi_r(\cdot)$ is a one-to-one mapping function between the compression index $\{q\}$ and the nested lattice code $\mathcal{C}_{\text{r}}$. Moreover, let Λ_{RC} be an n-dimensional lattice that is Poltyrev-good and Λ_{q} be an n-dimensional lattice that is Rogers-good such that $\Lambda_{\text{q}} \supseteq \Lambda_{\text{RC}}$. The existence of such a nested lattice pair good for quantization is guaranteed as in [15]. Also, we denote by $\mathcal{V}_{\text{RC}}$ and $\mathcal{V}_{\text{q}}$ the fundamental Voronoi regions of lattices Λ_{RC} and Λ_{q}, respectively. We define the Quantization/Compression codebook as $\mathcal{C}_{\text{q}} = \Lambda_{\text{q}} \cap \mathcal{V}_{\text{RC}}$. Also, let the second moment of Λ_{RC} be equal to

$$\sigma^2_{\Lambda_{\text{RC}}} = D + N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2 - \frac{[P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)]^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2}, \tag{29}$$

the second moment of Λ_{q} to be equal to $\sigma^2_{\Lambda_{\text{q}}} = D$ such that the source coding rate is

$$\begin{aligned} \hat{R} &= \frac{1}{2n}\log\left(\frac{\text{Vol}(\mathcal{V}_{\text{RC}})}{\text{Vol}(\mathcal{V}_{\text{q}})}\right) \\ &= \frac{1}{4}\log\left(1 + \frac{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2}{D} - \frac{[P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)]^2}{D(N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2)}\right). \end{aligned} \tag{30}$$

Encoding: During the first transmission period, the transmitters send the same inputs as in the coding scheme of Proposition 1, i.e., to transmit message W_a, Transmitter A sends the input $\mathbf{x}_a$ given by (23); and to transmit message W_b, Transmitter B sends the input $\mathbf{x}_b$ given by (24).
During this period, the relay receives $\mathbf{y}_r$ given by (10). Then, it quantizes the received signal $\mathbf{y}_r$ to

$$\begin{aligned} \mathbf{q} &= [Q_q(\mathbf{y}_r - \mathbf{u}_q)] \bmod \Lambda_{\text{RC}} \\ &= [\mathbf{y}_r - \mathbf{u}_q - \mathbf{d}] \bmod \Lambda_{\text{RC}} \\ &= [h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_b + \mathbf{z}_r - \mathbf{u}_q - \mathbf{d}] \bmod \Lambda_{\text{RC}} \end{aligned} \tag{31}$$

by using the quantization lattice code pair $(\Lambda_{\text{q}}, \Lambda_{\text{RC}})$ where $\mathbf{u}_q$ is a quantization dither that is uniformly distributed over $\mathcal{V}_{\text{q}}$ and $\mathbf{d}$ is given by $[h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_a + \mathbf{z}_r - \mathbf{u}_q] \bmod \Lambda_{\text{q}}$ and is independent of all other signals and uniformly distributed over $\mathcal{V}_{\text{q}}$ with second moment D.
During the second transmission period, the relay conveys the description $\mathbf{q}$ to the destination. To this end, the relay chooses the codeword $\mathbf{v}_r = \phi_r(q)$ associated with the index $\{q\}$ of $\mathbf{q}$ and sends

$$\tilde{\mathbf{x}}_r = [\mathbf{v}_r - \mathbf{u}_r] \bmod \Lambda_{\text{r}} \tag{32}$$

where $\mathbf{u}_r$ is a dither vector that is drawn independently and uniformly over $\mathcal{V}_{\text{r}}$.

Decoding: During the two transmission periods, the destination receives,

$$\begin{aligned} \mathbf{y}_d &= h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d \\ \tilde{\mathbf{y}}_d &= h_{rd}\tilde{\mathbf{x}}_r + \tilde{\mathbf{z}}_d. \end{aligned} \tag{33}$$

It first recovers the compressed version of the relay's output sent by the relay during the second transmission period, by utilizing its output $\tilde{\mathbf{y}}_d$ as well as the available side information $\mathbf{y}_d$. As it will be shown below, the destination recovers the compressed version $\hat{\mathbf{y}}_r = \mathbf{y}_r - \mathbf{d}$ of $\mathbf{y}_r$ if the constraint (45) below is satisfied (see the "Rate Analysis" section).
Next, the destination combines $\mathbf{y}_d$ and $\hat{\mathbf{y}}_r$ as follows

$$\mathbf{y}_i = \alpha_{1i}\mathbf{y}_d + \alpha_{2i}\hat{\mathbf{y}}_r, \qquad \text{for } i = t, k \tag{34}$$

and uses the obtained signals to compute two linear combinations with integer coefficients of the users' codewords [23],

$$\mathbf{y}'_t = [t_a\mathbf{v}_a + t_b\mathbf{v}_b + \mathbf{z}'_t] \bmod \Lambda \quad (35)$$
$$\mathbf{y}'_k = [k_a\mathbf{v}_a + k_b\mathbf{v}_b + \mathbf{z}'_k] \bmod \Lambda \quad (36)$$

where $\mathbf{z}'_t$ and $\mathbf{z}'_k$ are the effective noises given by

$$\mathbf{z}'_t \triangleq \Big[\alpha_{1t}\mathbf{z}_d + \alpha_{2t}\mathbf{z}_r + \alpha_{2t}\mathbf{d} + (\alpha_{1t}h_{ad} + \alpha_{2t}h_{ar} - \frac{t_a}{\beta_a})\mathbf{x}_a + (\alpha_{1t}h_{bd} + \alpha_{2t}h_{br} - \frac{t_b}{\beta_b})\mathbf{x}_b\Big] \bmod \Lambda, \quad (37)$$

$$\mathbf{z}'_k \triangleq \Big[\alpha_{1k}\mathbf{z}_d + \alpha_{2k}\mathbf{z}_r + \alpha_{2k}\mathbf{d} + (\alpha_{1k}h_{ad} + \alpha_{2k}h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_{1k}h_{bd} + \alpha_{2k}h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b\Big] \bmod \Lambda. \quad (38)$$

The effective noises $\mathbf{z}'_t$ and $\mathbf{z}'_k$ are the sum of signals uniformly distributed over fundamental Voronoi regions of Rogers-good lattices and Gaussian noises. Finally, by decoding the lattice points $\mathbf{e}_1 = [t_a\mathbf{v}_a + t_b\mathbf{v}_b] \in \Lambda$ and $\mathbf{e}_2 = [k_a\mathbf{v}_a + k_b\mathbf{v}_b] \in \Lambda$ using the modulo-lattice additive noise (MLAN) channels $\mathbf{y}'_t$ and $\mathbf{y}'_k$, respectively, the destination obtains the two linear combinations with integer coefficients of the users' codewords. As it will be shown below, this can be accomplished with probabilities of error $\Pr(\mathbf{z}'_t \notin \mathcal{V}_{\text{FINE}})$ and $\Pr(\mathbf{z}'_k \notin \mathcal{V}_{\text{FINE}})$ that are as small as desired.

Rate Analysis:

The relay compresses its output $\mathbf{y}_r$ and sends index $\{q\}$ of $\mathbf{q}$ at the per-channel use rate [15], [16]

$$\hat{R} = \frac{1}{2n}\log\Big(\frac{\text{Vol}(\mathcal{V}_{\text{RC}})}{\text{Vol}(\mathcal{V}_{\text{q}})}\Big) = \tfrac{1}{4}\log\left(1 + \frac{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2}{D} - \frac{[P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)]^2}{D(N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2)}\right). \quad (39)$$

The destination receives the index $\{q\}$ of $\mathbf{q}$ and recovers $\hat{\mathbf{y}}_r$ as

$$\begin{aligned}\hat{\mathbf{y}}_r &= [\mathbf{q} + \mathbf{u}_q - \alpha(h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d)] \bmod \Lambda_{\text{RC}} + \alpha(h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d)\\ &= [h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_b + \mathbf{z}_r - \mathbf{d} - \alpha(h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d)] \bmod \Lambda_{\text{RC}} + \alpha(h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d)\\ &\overset{(a)}{=} h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_b + \mathbf{z}_r - \mathbf{d}\\ &= \mathbf{y}_r - \mathbf{d}\end{aligned} \quad (40)$$

where (a) follows since the probability of decoding error P_e, given by

$$P_e = \Pr\Big\{\Big[(h_{ar} - \alpha h_{ad})\mathbf{x}_a + (h_{br} - \alpha h_{bd})\mathbf{x}_b + \mathbf{z}_r - \alpha\mathbf{z}_d - \mathbf{d}\Big] \bmod \Lambda_{\text{RC}} \neq (h_{ar} - \alpha h_{ad})\mathbf{x}_a + (h_{br} - \alpha h_{bd})\mathbf{x}_b + \mathbf{z}_r - \alpha\mathbf{z}_d - \mathbf{d}\Big\}, \quad (41)$$

vanishes asymptotically ($P_e \longrightarrow 0$) as $n \longrightarrow \infty$ [15, Proof of (4.19)], [16, Proof of Corollary 11] for a sequence of a good nested lattice codes since

$$\frac{1}{n}\mathbb{E}\|(h_{ar} - \alpha h_{ad})\mathbf{x}_a + (h_{br} - \alpha h_{bd})\mathbf{x}_b + \mathbf{z}_r - \alpha\mathbf{z}_d - \mathbf{d}\|^2 = \sigma^2_{\Lambda_{\text{RC}}}. \quad (42)$$

We should note that, from [13, Lemma 8] and [16, Lemma 5], $(h_{ar} - \alpha h_{ad})\mathbf{x}_a + (h_{br} - \alpha h_{bd})\mathbf{x}_b + \mathbf{z}_r - \alpha\mathbf{z}_d - \mathbf{d}$ can be upper bounded by the density of an i.i.d. zero-mean Gaussian vector whose variance approaches (42), since $\mathbf{x}_a$ and $\mathbf{x}_b$ are uniformly distributed over the Rogers-good $\mathcal{V}$, $\mathbf{d}$ is uniformly distributed over the Rogers-good $\mathcal{V}_{\text{q}}$ and $\mathbf{z}_r - \alpha\mathbf{z}_d$ is Gaussian. As Λ_{RC} is Poltyrev-good, (41) can be made arbitrary small as $n \longrightarrow \infty$. We also note that α is chosen so as to guarantee (42).

At the end of the second transmission period, the destination can decode the correct relay input $\tilde{\mathbf{x}}_r$ reliably [22] if

$$R_r < \frac{1}{4}\log\left(1 + \frac{P_r|h_{rd}|^2}{N}\right). \quad (43)$$

We should note that the source coding rate of $\mathbf{q}$, $\hat{R}$, must be less than the channel coding rate R_r,

$$\hat{R} \leq R_r < \frac{1}{4}\log\left(1 + \frac{P_r|h_{rd}|^2}{N}\right). \quad (44)$$

From (39) and (44), we get the following constraint on the distortion

$$D \geq \frac{N^2\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2\big(\text{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T(\boldsymbol{\beta_s} \circ \mathbf{h}_d)\big)^2}{|h_{rd}|^2 P_r\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)}. \quad (45)$$

The above implies that, under the constraint (45), the destination recovers the lossy version $\hat{\mathbf{y}}_r$ of what was sent by the relay during the second transmission period.

Using the MLAN channel $\mathbf{y}'_t$ given by (35) and proceeding in a way that is essentially similar to [13], the destination can decode the linear combination $\mathbf{e}_1 = t_a\mathbf{v}_a + t_b\mathbf{v}_b$ with a probability of error $\Pr(\mathbf{z}'_t \notin \mathcal{V}_{\text{FINE}})$ going to zero exponentially in n if

$$R_1 < \frac{1}{4}\log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T\boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T\mathbf{n}_d}\right), \quad (46)$$

where the distortion D satisfies the constraint (45) and $\boldsymbol{\alpha}_t$ should be chosen to minimize the effective noise $\mathbf{z}'_t$ in (37), i.e., such that

$$\boldsymbol{\alpha}^\star_t = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1}\mathbf{G}\mathbf{t}, \quad (47)$$

where $\mathbf{G} = [\boldsymbol{\beta}_s \circ \mathbf{h}_d,\ \boldsymbol{\beta}_s \circ \mathbf{h}_r]^T \in \mathbb{R}^{2\times 2}$ and $\mathbf{N}_d = [1/\text{snr},\ 0;\ 0,\ 1/\text{snr} + D/P] \in \mathbb{R}^{2\times 2}$. Similarly, in decoding the linear combination $\mathbf{e}_2 = k_a\mathbf{v}_a + k_b\mathbf{v}_b$, the probability of error at the destination $\Pr(\mathbf{z}'_k \notin \mathcal{V}_{\text{FINE}})$ goes to zero exponentially in n if

$$R_2 < \frac{1}{4}\log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T\boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T\mathbf{n}_d}\right), \quad (48)$$

where $\boldsymbol{\alpha}_k$ should be chosen to minimize the effective noise $\mathbf{z}'_k$ in (38), i.e., such that

$$\boldsymbol{\alpha}^\star_k = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1}\mathbf{G}\mathbf{k}. \quad (49)$$

The above means that using the lattice-based coding scheme that we described, the destination can decode the transmitters' codewords correctly at the transmission symmetric-rate

$R_{\text{sym}}^{\text{CoD}} = \min\{R_1, R_2\}$ provided that the condition (45) is satisfied. This completes the proof of Proposition 2. ∎

Remark 2: There are some high level similarities among the coding strategies of proposition 1 and proposition 2. Both strategies decode two linearly independent equations with integer coefficients. However, the required two equations are obtained differently in the two cases. More specifically, while the two equations are computed in a distributed manner using the coding strategy of proposition 1, they are both computed locally at the destination in a joint manner using the coding strategy of proposition 2. The advantage of decoding the equations locally at the destination is that the decoder utilizes *all* the output available. This means that the destination utilizes the outputs received during the first and second transmission periods in a joint manner. By opposition, the coding strategy of proposition 1 is such that the computation of one equation utilizes only the output received directly from the transmitters during the first transmission period. The computation of the other equation is limited by the weaker output among the output at the relay during the first transmission period and the output at the destination during the second transmission period (since the equation decoded at the relay has to be recovered at the destination). The reader may refer to Section V where this aspect will be illustrated through some numerical examples and discussed further.

Remark 3: For the multiaccess relay network that we study, the coding strategy of Proposition 2 can at best achieve the same performance as that allowed by regular compress-and-forward. This can be observed as follows. After conveying a quantized version of the relay's output to the destination, the decoding problem at the destination is equivalent to that over a regular two-user multiaccess channel with the output at the receiver given by $(\hat{\mathbf{y}}_r, \mathbf{y}_d)$. Optimal decoding of the messages can then be accomplished directly using joint decoding of the messages as in the CF-based approach of Section II-C. However, note that even though the coding strategy of Proposition 2 can not achieve larger rates, it has some advantages over standard CF. For instance, it is based on feasible structured codes instead of random codes which are infeasible in practice. Also, it utilizes linear receivers that have a computational complexity similar to that of the decorrelator and minimum-mean-squared error receiver. These two linear receivers are often used as low-complexity alternatives instead of the maximum likelihood receiver which has high computational complexity. From this angle, note that this work also connects with [16] in which the authors show that, for the standard Gaussian three-terminal relay channel, the rate achievable using standard CF can also be achieved alternately using lattice codes.

Remark 4: The system model that we study can be extended to the case in which the relay is full-duplex. This can be done in the same spirit as described in [16, Section IV-C].

Remark 5: The system model that we study can be extended to the case of multiple transmitters. Let us consider a system in which M transmitters would like to communicate with a common destination with the help of a single relay. The destination needs M independent linear combinations of the transmitters' codewords to be able to recover the transmitted messages. Hence, in scheme 1, the relay and destination compute M independent linear combinations that yield the highest symmetric-rate. In scheme 2, the destination, using the available outputs, computes the M independent linear combinations.

IV. Symmetric Rates Optimization

Section IV-A is devoted to finding optimal powers and integer-coefficients that maximize the symmetric-rate of Proposition 1. Section IV-B deals with the optimization problem of Proposition 2.

A. Compute-and-Forward at Relay

1) Problem Formulation: Consider the symmetric-rate $R_{\text{sym}}^{\text{CoF}}$ as given by (22) in Proposition 1. The optimization problem can be stated as:

$$\begin{aligned}\text{(A)} \quad :\max \frac{1}{4}\min\Bigg\{&\log^+\left(\left(\|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s\circ\mathbf{h}_d)^T\mathbf{t})^2}{N+P\|\boldsymbol{\beta}_s\circ\mathbf{h}_d\|^2}\right)^{-1}\right),\\ &\log^+\left(\left(\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s\circ\mathbf{h}_r)^T\mathbf{k})^2}{N+P\|\boldsymbol{\beta}_s\circ\mathbf{h}_r\|^2}\right)^{-1}\right),\\ &\log\left(1+\frac{P_r|h_{rd}|^2}{N}\right)\Bigg\},\end{aligned} \tag{50}$$

where the maximization is over $\boldsymbol{\beta}_s$ such that $0 \le |\beta_a| \le \sqrt{P_a/P}$ and $0 \le |\beta_b| \le \sqrt{P_b/P}$, and over the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ such that $\det(\mathbf{k},\mathbf{t}) \neq 0$.

The optimization problem (A) is non-linear and non-convex. Also, it is a MIQP optimization problem; and, so, it is not easy to solve it optimally. In what follows, we solve this optimization problem iteratively, by finding appropriate preprocessing vector $\boldsymbol{\beta}_s$ and integer coefficients $\mathbf{t}$ and $\mathbf{k}$, alternately. We note that the allocation of the vector $\boldsymbol{\beta}_s$ determines the power that each of the transmitters should use for the transmission. For this reason, we will sometimes refer loosely to the process of selecting the vector $\boldsymbol{\beta}_s$ as the power allocation process.
Let, with a slight abuse of notation, $R_{\text{sym}}^{\text{CoF}}[\iota]$ denote the value of the symmetric-rate at some iteration $\iota \ge 0$. To compute $R_{\text{sym}}^{\text{CoF}}$ as given by (50) iteratively, we develop the following algorithm, to which we refer to as "Algorithm A" in reference to the optimization problem (A).

Algorithm A Iterative algorithm for computing $R_{\text{sym}}^{\text{CoF}}$ as given by (50)

1: Initialization: set $\iota = 1$ and $\boldsymbol{\beta}_s = \boldsymbol{\beta}_s^{(0)}$
2: Set $\boldsymbol{\beta}_s = \boldsymbol{\beta}_s^{(\iota-1)}$ in (50), and solve the obtained problem using Algorithm A-1 given below. Denote by $\mathbf{k}^{(\iota)}$ the found $\mathbf{k}$, and by $\mathbf{t}^{(\iota)}$ the found $\mathbf{t}$
3: Set $\mathbf{k} = \mathbf{k}^{(\iota)}$ and $\mathbf{t} = \mathbf{t}^{(\iota)}$ in (50), and solve the obtained problem using Algorithm A-2 given below. Denote by $\boldsymbol{\beta}_s^{(\iota)}$ the found $\boldsymbol{\beta}_s$
4: Increment the iteration index as $\iota = \iota + 1$, and go back to Step 2
5: Terminate if $\|\boldsymbol{\beta}_s^{(\iota)} - \boldsymbol{\beta}_s^{(\iota-1)}\| \le \epsilon_1$, $|R_{\text{sym}}^{\text{CoF}}[\iota] - R_{\text{sym}}^{\text{CoF}}[\iota-1]| \le \epsilon_2$

As described in "Algorithm A", we compute the appropriate preprocessing vector $\boldsymbol{\beta}_s$ and integer coefficients $\mathbf{k}$ and $\mathbf{t}$, alternately. More specifically, at iteration $\iota \ge 1$, the algorithm computes appropriate integer coefficients $\mathbf{k}^{(\iota)} \in \mathbb{Z}^2$ and $\mathbf{t}^{(\iota)} \in \mathbb{Z}^2$ that correspond to a maximum of (50) computed with the choice of the preprocessing vector $\boldsymbol{\beta}_s$ set to its value obtained from the previous iteration, i.e., $\boldsymbol{\beta}_s = \boldsymbol{\beta}_s^{(\iota-1)}$ (for the initialization, set $\boldsymbol{\beta}_s^{(0)}$ to a default value). As we will show, this sub-problem is a MIQP problem with quadratic constraints; and we solve it using "Algorithm A-1". Next, for the found integer coefficients, the algorithm computes the adequate preprocessing vector $\boldsymbol{\beta}_s^{(\iota)}$ that corresponds to a maximum of (50) computed with the choice $\mathbf{k} = \mathbf{k}^{(\iota)}$ and $\mathbf{t} = \mathbf{t}^{(\iota)}$. As we will show, this sub-problem can be formulated as a complementary geometric programming problem. We solve it through a series of geometric programming and successive convex optimization approach (see "Algorithm A-2" below). The iterative process in "Algorithm A" terminates if the following two conditions hold: $\|\boldsymbol{\beta}_{\mathrm{s}}{}^{(\iota)} - \boldsymbol{\beta}_{\mathrm{s}}{}^{(\iota-1)}\|$ and $|R_{\mathrm{sym}}^{\mathrm{CoF}}[\iota] - R_{\mathrm{sym}}^{\mathrm{CoF}}[\iota-1]|$ are smaller than prescribed small strictly positive constants ϵ_1 and ϵ_2, respectively — in this case, the optimized value of the symmetric-rate is $R_{\mathrm{sym}}^{\mathrm{CoF}}[\iota]$, and is attained using the preprocessing power vector $\boldsymbol{\beta}_s^{\star} = \boldsymbol{\beta}_{\mathrm{s}}{}^{(\iota)}$ and integer vectors $\mathbf{k}^{\star} = \mathbf{k}^{(\iota)}$ and $\mathbf{t}^{\star} = \mathbf{t}^{(\iota)}$.

In the following two sections, we study the aforementioned two sub-problems of problem (A), and describe the algorithms that we propose to solve them.

2) Integer Coefficients Optimization: In this section, we focus on the problem of finding appropriate integer vectors $\mathbf{k} \in \mathbb{Z}^2$ and $\mathbf{t} \in \mathbb{Z}^2$ for a given choice of the preprocessing vector $\boldsymbol{\beta}_s$. Investigating the objective function in (50), it can easily be seen that this problem can be equivalently stated as

$$\min_{\mathbf{k},\,\mathbf{t},\,\Delta_1} \quad \Delta_1 \tag{51a}$$

$$\text{s. t.} \quad \Delta_1 \geq \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \tag{51b}$$

$$\Delta_1 \geq \|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \tag{51c}$$

$$\Delta_1 \geq \frac{N}{N + P_r|h_{rd}|^2}, \tag{51d}$$

$$\det(\mathbf{k}, \mathbf{t}) \neq 0 \tag{51e}$$

$$\mathbf{k} \in \mathbb{Z}^2,\ \mathbf{t} \in \mathbb{Z}^2,\ \ \Delta_1 \in \mathbb{R}. \tag{51f}$$

Note that Δ_1 is simultaneously an extra optimization variable and the objective function in (51). Also, it is easy to see that the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ that achieve the minimum value of Δ_1 also achieve a maximum value of the objective function in (50).

To find a convenient solution, we reformulate the problem (51) and introduce the following quantities. Let $\mathbf{a}_0 = [0,0,0,0,1]^T$; $\mathbf{a}_1 = \mathbf{a}_2 = \mathbf{a}_3 = [0,0,0,0,-1]^T$ and $\mathbf{a}_4 = [0,0,0,0,0]^T$. Also, let $\mathbf{b} = [t_a, t_b, k_a, k_b, \Delta_1]^T$; and the scalars $c_1 = c_2 = 0$, $c_3 = N/(N + P_r|h_{rd}|^2)$ and $c_4 = -1$. We also introduce the following five-by-five matrices $\mathbf{F}_1$, $\mathbf{F}_2$, $\mathbf{F}_3$ and $\mathbf{F}_4$, where

$$\mathbf{F}_1 = \begin{bmatrix} 2(\mathbf{I}_2 - \boldsymbol{\Omega}_1) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 2(\mathbf{I}_2 - \boldsymbol{\Omega}_2) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 0 \end{bmatrix}, \tag{52}$$

with

$$\boldsymbol{\Omega}_1 := \frac{P}{N + P\|\mathbf{h}_d\|^2}(\boldsymbol{\beta}_s \circ \mathbf{h}_d)(\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T,$$
$$\boldsymbol{\Omega}_2 := \frac{P}{N + P\|\mathbf{h}_r\|^2}(\boldsymbol{\beta}_s \circ \mathbf{h}_r)(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T, \tag{53}$$

$$\mathbf{F}_3 = \mathbf{0}, \text{ and } \mathbf{F}_4 = \begin{bmatrix} \mathbf{0} & 0 & -2 & 0 \\ \mathbf{0} & 2 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}. \tag{54}$$

The optimization problem (51) can now be reformulated equivalently as

$$\begin{aligned} \min_{\mathbf{b}} \quad & \mathbf{a}_0^T \mathbf{b} \\ \text{s. t.} \quad & \frac{1}{2}\mathbf{b}^T \mathbf{F}_i \mathbf{b} + \mathbf{a}_i^T \mathbf{b} \leq c_i \quad i = 1, \cdots, 4 \\ & \mathbf{k} \in \mathbb{Z}^2,\ \mathbf{t} \in \mathbb{Z}^2,\ \ \Delta_1 \in \mathbb{R} \end{aligned} \tag{55}$$

The equivalent optimization problem (55) is a MIQP problem with quadratic constraints [24]. If the involved matrices associated with the quadratic constraints (i.e., the matrices $\mathbf{F}_1$, $\mathbf{F}_2$ and $\mathbf{F}_4$ here) are all semi-definite, there are known approaches for solving MIQP optimization problems, such as cutting plane, decomposition, logic-based and branch and bound approaches [24]. In our case, it is easy to see that the matrices $\mathbf{F}_1$ and $\mathbf{F}_2$ are positive semi-definite. However, the matrix $\mathbf{F}_4$ is indefinite, irrespective to the values of $\mathbf{k}$ and $\mathbf{t}$. To solve the optimization problem (55), we transform it into one that is MIQP-compatible (i.e., in which all the quadratic constraints are associated with semi-definite matrices). First, we solve the problem without considering the quadratic constraint (51e) and we denote by $\boldsymbol{\kappa}_0$ and $\boldsymbol{\tau}_0$ the found solutions of $\mathbf{k}$ and $\mathbf{t}$, respectively. The optimization problem terminates only if $\boldsymbol{\kappa}_0$ and $\boldsymbol{\tau}_0$ are independent. However, if $\boldsymbol{\kappa}_0$ and $\boldsymbol{\tau}_0$ are not independent, we resolve the optimization problem (51) replacing the quadratic constraint (51e) with a linear constraint given by

$$\det(\boldsymbol{\kappa}_0, \mathbf{k}) + \det(\boldsymbol{\tau}_0, \mathbf{t}) \neq 0. \tag{56}$$

Then, we denote by $\boldsymbol{\kappa}_1$ and $\boldsymbol{\tau}_1$ the found solutions of $\mathbf{k}$ and $\mathbf{t}$, respectively. It can easily be seen that i) one of the integer vectors either $\boldsymbol{\kappa}_1$ or $\boldsymbol{\tau}_1$ will keep its original solution $\boldsymbol{\kappa}_0$ or $\boldsymbol{\tau}_0$; and ii) the other integer vector either $\boldsymbol{\kappa}_1$ or $\boldsymbol{\tau}_1$ will be independent from its original solution. Let us consider the case in which $\boldsymbol{\kappa}_1$ keeps its original solution $\boldsymbol{\kappa}_0$, then $\det(\boldsymbol{\kappa}_0, \boldsymbol{\kappa}_1)$ is equal to zero, and thus $\det(\boldsymbol{\tau}_0, \boldsymbol{\tau}_1)$ must be different than zero to satisfy the constraint (56). In this case, $\boldsymbol{\kappa}_1$ and $\boldsymbol{\tau}_1$ are independent since $\boldsymbol{\tau}_0$ and $\boldsymbol{\tau}_1$ are independent, and $\boldsymbol{\tau}_0$ and $\boldsymbol{\kappa}_1$ are not independent. Similarly, we can show that, for the case in which $\boldsymbol{\tau}_1$ keeps its original solution, $\det(\boldsymbol{\kappa}_0, \boldsymbol{\kappa}_1)$ is different than zero and $\boldsymbol{\kappa}_1$ and $\boldsymbol{\tau}_1$ are independent. Hence, the optimization problem with the constraint given in (56) yields two independent vectors that minimize (55). The optimization problem (55) can be solved using "Algorithm A-1" hereinafter.

Algorithm A-1 Integer coefficients selection for $R_{\mathrm{sym}}^{\mathrm{CoF}}$ as given by (50)

1: Use the branch-and-bound algorithm of [25], [26] to solve for Δ_1, $\mathbf{k}$ and $\mathbf{t}$ without considering the constraint (51e). Denote the found solutions of $\mathbf{k}$ and $\mathbf{t}$ as $\boldsymbol{\kappa}_0$ and $\boldsymbol{\tau}_0$, respectively
2: Terminate if $\det(\boldsymbol{\kappa}_0, \boldsymbol{\tau}_0) \neq 0$ otherwise GOTO Step 3
3: Use the branch-and-bound algorithm to solve for Δ_1, $\mathbf{k}$ and $\mathbf{t}$ with the constraint (51e) substituted with $\det(\boldsymbol{\kappa}_0, \mathbf{k}) + \det(\boldsymbol{\tau}_0, \mathbf{t}) > 0$. Denote the found solution of Δ_1 as $\Delta_1^{\min,1}$
4: Again, use the branch-and-bound algorithm to solve for Δ_1, $\mathbf{k}$ and $\mathbf{t}$ with the constraint (51e) substituted with $-\det(\boldsymbol{\kappa}_0, \mathbf{k}) - \det(\boldsymbol{\tau}_0, \mathbf{t}) > 0$. Denote the found solution of Δ_1 as $\Delta_1^{\min,2}$
5: Select the integer coefficients corresponding to the minimum among $\Delta_1^{\min,1}$ and $\Delta_1^{\min,2}$

Remark 6: We should note that, since the integer vectors $\mathbf{t}$ and $\mathbf{k}$ are not coupled through the objective function, they can be optimized separately. One solution is to find the best two linearly independent solutions for $\mathbf{t}$ and the best two linearly independent solutions for $\mathbf{k}$ using the Lenstra-Lenstra-Lovász (LLL) algorithm as in [27] or the branch and bound method. Then, we search for the two independent vectors $\mathbf{t}$ and $\mathbf{k}$ that yield the highest symmetric rate.

3) Power Allocation Policy: Let us now focus on the problem of finding an appropriate preprocessing vector $\boldsymbol{\beta}_s$ for given integer vectors $\mathbf{k}$ and $\mathbf{t}$. Again, investigating the objective function in (50), it can easily be seen that this problem can be equivalently stated as

$$\min_{\boldsymbol{\beta}_{\mathrm{s}},\,\Delta_2} \quad \Delta_2 \tag{57a}$$

$$\text{s. t.} \quad \Delta_2 \geq \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \tag{57b}$$

$$\Delta_2 \geq \|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \tag{57c}$$

$$\Delta_2 \geq \frac{N}{N + P_r|h_{rd}|^2}, \tag{57d}$$

$$-\sqrt{\frac{P_i}{P}} \leq \beta_i \leq \sqrt{\frac{P_i}{P}}, \quad i = a, b \tag{57e}$$

$$\boldsymbol{\beta}_s \in \mathbb{R}^2, \; \Delta_2 \in \mathbb{R}. \tag{57f}$$

Here, similar to the previous section, Δ_2 is simultaneously an extra optimization variable and the objective function in (57). Also, it is easy to see that the value of $\boldsymbol{\beta}_s$ that achieves the minimum value of Δ_2 also achieves a maximum value of the objective function in (50).

The optimization problem in (57) is non-linear and non-convex. We use geometric programming [17] to solve it. Geometric programming is a special form of convex optimization for which efficient algorithms have been developed and are known in the related literature [28]. There are two forms of GP: the standard form and the convex form. In its standard form, a GP optimization problem is generally written as [28]

$$\text{minimize} \quad f_0(\boldsymbol{\beta}_{\mathrm{s}}, \Delta_2) \tag{58a}$$

$$\text{subject to} \quad f_j(\boldsymbol{\beta}_{\mathrm{s}}, \Delta_2) \leq 1, \quad j = 1, \cdots, J, \tag{58b}$$

$$g_l(\boldsymbol{\beta}_{\mathrm{s}}, \Delta_2) = 1, \quad l = 1, \cdots, L, \tag{58c}$$

where the functions f_0 and f_j, $j = 1, \cdots, J$, are posynomials and the functions g_l, $l = 1, \cdots, L$, are monomials in $\boldsymbol{\beta}_{\mathrm{s}}$ and Δ_2. In its standard form, (58) is not a convex optimization problem. However, when possible, a careful application of an appropriate logarithmic transformation of the involved variables and constants generally turns the problem (58) into one that is equivalent and convex. That is, (58) is a GP nonlinear, nonconvex optimization problem that can be transformed into a nonlinear, convex optimization problem.

In the problem (57), the constraints (57b) and (57c) contain functions that are non posynomial. Also, the variables in (57) are not all positive, thus preventing a direct application of logarithmic transformation. In what follows, we first transform the problem (57) into an equivalent one in which the constraints involve functions that are all posynomial and the variables are all positive; and then we develop an algorithm to solve the equivalent problem.

Let $\mathbf{c} = [c_a, c_b]^T \in \mathbb{R}^2$ and $\boldsymbol{\delta}_s = [\delta_a, \delta_b]^T \in \mathbb{R}^2$, such that $c_i > \sqrt{P_i/P}$ and $\delta_i = \beta_i + c_i$ for $i = a, b$. Note that the elements of $\boldsymbol{\delta}_s$ are all strictly positive. Also, for convenience, we define the following functions, for $\mathbf{z} = [z_a, z_b] \in \mathbb{Z}^2$,

$$\begin{aligned}\psi_1^i(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2, \mathbf{z}) = {} & 2\Delta_2 P\left(|h_{ai}|^2\delta_a c_a + |h_{bi}|^2\delta_b c_b\right) \\ & + P\left(z_a^2 + z_b^2\right)\left(|h_{ai}|^2(\delta_a^2 + c_a^2) + |h_{bi}|^2(\delta_b^2 + c_b^2)\right) \\ & + 2P|h_{ai}|^2 z_a^2 \delta_a c_a + 2P|h_{bi}|^2 z_b^2 \delta_b c_b \\ & + 2P h_{ai} h_{bi} z_a z_b(\delta_a c_b + \delta_b c_a) + N\left(z_a^2 + z_b^2\right)\end{aligned}$$

$$\begin{aligned}\psi_2^i(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2, \mathbf{z}) = {} & \Delta_2\left(N + P|h_{ai}|^2(\delta_a^2 + c_a^2) + P|h_{bi}|^2(\delta_b^2 + c_b^2)\right) \\ & + 2P\left(z_a^2 + z_b^2\right)\left(|h_{ai}|^2\delta_a c_a + |h_{bi}|^2\delta_b c_b\right) \\ & + P|h_{ai}|^2 z_a^2(\delta_a^2 + c_a^2) + P|h_{bi}|^2 z_b^2(\delta_b^2 + c_b^2) \\ & + 2P h_{ai} h_{bi} z_a z_b(\delta_a\delta_b + c_a c_b).\end{aligned} \tag{59}$$

Let us now define the following functions, $f_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) = \psi_1^d(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2, \mathbf{t})$, $f_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) = \psi_1^r(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2, \mathbf{k})$, $g_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) = \psi_2^d(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2, \mathbf{t})$, $g_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) = \psi_2^r(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2, \mathbf{k})$, $f_3(\Delta_2) = \Delta_2^{-1} N(N + P_r|h_{rd}|^2)^{-1}$.

It is now easy to see that the optimization problem (57) can be stated in the following form.

$$\min_{\boldsymbol{\delta}_{\mathrm{s}},\,\Delta_2} \; \Delta_2 \tag{60a}$$

$$\text{s. t.} \; \frac{f_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)}{g_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)} \leq 1, \; \frac{f_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)}{g_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)} \leq 1, \; f_3(\Delta_2) \leq 1 \tag{60b}$$

$$-\sqrt{\frac{P_i}{P}} + c_i \leq \delta_i \leq \sqrt{\frac{P_i}{P}} + c_i \;, \; i = a, b \tag{60c}$$

$$\boldsymbol{\delta}_s \in \mathbb{R}^2, \; \mathbf{c} \in \mathbb{R}^2, \; \Delta_2 \in \mathbb{R}. \tag{60d}$$

The constraints (60b) involve functions that consist of ratios of posynomials, i.e., are not posynomial — recall that a ratio of posynomials is in general non posynomial. Minimizing or upper bounding a ratio of posynomials belongs to a class of non-convex problems known as complementary GP [28]. We can solve a complementary GP problem by transforming it into a series of GPs. For this, we use Lemma 1 [17] to approximate the functions $g_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ and $g_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ with monomials around some initial value.

Lemma 1: Let $g(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) = \sum_j u_j(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ be a posynomial. Then

$$g(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) \geq \tilde{g}(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2) = \prod_j \left(\frac{u_j(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)}{\gamma_j}\right)^{\gamma_j}. \tag{61}$$

Here, $\gamma_j = u_j({\boldsymbol{\delta}_{\mathrm{s}}}^\dagger, \Delta_2^\dagger)/g({\boldsymbol{\delta}_{\mathrm{s}}}^\dagger, \Delta_2^\dagger)$, $\forall j$, for any fixed positive ${\boldsymbol{\delta}_{\mathrm{s}}}^\dagger$ and $\Delta_2^\dagger$, then $\tilde{g}({\boldsymbol{\delta}_{\mathrm{s}}}^\dagger, \Delta_2^\dagger) = g({\delta_{\mathrm{s}}}^\dagger, \Delta_2^\dagger)$, and $\tilde{g}({\boldsymbol{\delta}_{\mathrm{s}}}^\dagger, \Delta_2^\dagger)$ is the best local monomial approximation to $g({\boldsymbol{\delta}_{\mathrm{s}}}^\dagger, \Delta_2^\dagger)$ near ${\boldsymbol{\delta}_{\mathrm{s}}}^\dagger$ and $\Delta_2^\dagger$.

Let $\tilde{g}_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ and $\tilde{g}_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ be the monomial approximations of the posynomial functions $g_1(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ and $g_2(\boldsymbol{\delta}_{\mathrm{s}}, \Delta_2)$ obtained using Lemma 1. Using these monomial approximations, the ratios of posynomials involved in the constraint (60b) can be upper bounded by posynomials. Hence, we have transformed the optimization problem (60) into a GP one that can be solved easily using an interior point approach. Then, the found solution is used as initial value to transform the complementary GP into a new GP problem. This process is repeated until convergence as described in "Algorithm A-2".

We should note that each GP in the iteration loop tries to improve the accuracy of the approximation to a particular minimum in the original feasible region.

The optimal solution of the problem obtained using the convex approximations is also optimal for the original problem (57), i.e., satisfies the Karush-Kuhn-Tucker (KKT) conditions of the original problem (57), since the applied approximations satisfy the following three properties [29], [17]:

1) $g_j(\boldsymbol{\delta}_{\rm s}, \Delta_2) \leq \tilde{g}_j(\boldsymbol{\delta}_{\rm s}, \Delta_2)$ for all $\boldsymbol{\delta}_{\rm s}$ and Δ_2 where $\tilde{g}_j(\boldsymbol{\delta}_{\rm s}, \Delta_2)$ is the approximation of $g_j(\boldsymbol{\delta}_{\rm s}, \Delta_2)$.
2) $g_j(\boldsymbol{\delta}_{\rm s}{}^{\ddagger}, \Delta_2^{\ddagger}) = \tilde{g}_j(\boldsymbol{\delta}_{\rm s}{}^{\ddagger}, \Delta_2^{\ddagger})$ where $\boldsymbol{\delta}_{\rm s}{}^{\ddagger}$ and $\Delta_2^{\ddagger}$ are the optimal solutions of the approximated problem in the previous iteration.
3) $\triangledown g_j(\boldsymbol{\delta}_{\rm s}{}^{\ddagger}, \Delta_2^{\ddagger}) = \triangledown \tilde{g}_j(\boldsymbol{\delta}_{\rm s}{}^{\ddagger}, \Delta_2^{\ddagger})$, where $\triangledown g_j(\cdot)$ stands for the gradient of function $g_j(\cdot)$.

The "Algorithm A-2" is provably convergent [17] since all the three conditions for convergence described above are satisfied.

Algorithm A-2 Power allocation policy for $R_{\rm sym}^{\rm CoF}$ as given by (50)

1: Compute $\Delta_2^{(0)}$ using $\boldsymbol{\delta}_{\rm s}{}^{(0)}$ (the initial value) and set $\iota_2 = 1$
2: Approximate $g(\boldsymbol{\delta}_{\rm s}{}^{(\iota_2)}, \Delta_2^{(\iota_2)})$ with $\tilde{g}(\boldsymbol{\delta}_{\rm s}{}^{(\iota_2)}, \Delta_2^{(\iota_2)})$ around $\boldsymbol{\delta}_{\rm s}{}^{(\iota_2-1)}$ and $\Delta_2^{(\iota_2-1)}$ using (61)
3: Solve the resulting approximated GP problem using an interior point approach. Denote the found solutions as $\boldsymbol{\delta}_{\rm s}{}^{(\iota_2)}$ and $\Delta_2^{(\iota_2)}$
4: Increment the iteration index as $\iota_2 = \iota_2 + 1$ and go back to Step 2 using $\boldsymbol{\delta}_{\rm s}$ and Δ_2 of step 3
5: Terminate if $\|\boldsymbol{\delta}_{\rm s}{}^{(\iota_2)} - \boldsymbol{\delta}_{\rm s}{}^{(\iota_2-1)}\| \leq \epsilon_1$

B. Compress-and-Forward at Relay and Compute at Destination

The algorithms that we develop in this section to solve the optimization problem of Proposition 2, are essentially similar to those that we developed in the previous section. For brevity, we omit the details in this section.

1) Problem Formulation: Recall the expression of $R_{\rm sym}^{\rm CoD}$ as given by (27) in Proposition 2. The optimization problem can be stated as:

$$\text{(B)}: \max \frac{1}{4} \min \left\{ \log^+ \left(\frac{\text{snr}}{\text{snr} \|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d} \right), \right.$$
$$\left. \log^+ \left(\frac{\text{snr}}{\text{snr} \|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d} \right) \right\}, \quad (62)$$

where the distortion D is given by

$$D \geq \frac{N^2\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2\left(\text{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T (\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)}, \quad (63)$$

and the maximization is over $\boldsymbol{\alpha}_t$, $\boldsymbol{\alpha}_k$, $\boldsymbol{\beta_s}$ such that $0 \leq |\beta_a| \leq \sqrt{P_a/P}$, $0 \leq |\beta_b| \leq \sqrt{P_b/P}$, and over the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ such that $\det(\mathbf{k}, \mathbf{t}) \neq 0$.

To compute $R_{\rm sym}^{\rm CoD}$ as given by (62), we develop the following iterative algorithm which optimizes the integer coefficients and the powers alternately, and to which we refer to as "Algorithm B" in reference to the optimization problem (B).

Algorithm B Iterative algorithm for computing $R_{\rm sym}^{\rm CoD}$ as given by (62)

1: Choose an initial feasible vector $\boldsymbol{\beta}_{\rm s}{}^{(0)}$ and set $\iota = 1$
2: Solve (62) with $\boldsymbol{\beta}_{\rm s} = \boldsymbol{\beta}_{\rm s}{}^{(\iota-1)}$ for the optimal $\mathbf{k}$ and $\mathbf{t}$ using "Algorithm B-1" and assign it to $\mathbf{k}^{(\iota)}$ and $\mathbf{t}^{(\iota)}$
3: Solve (62) with $\mathbf{k} = \mathbf{k}^{(\iota)}$ and $\mathbf{t} = \mathbf{t}^{(\iota)}$ for the optimal $\boldsymbol{\beta}_{\rm s}$ using "Algorithm B-2" and assign it to $\boldsymbol{\beta}_{\rm s}{}^{(\iota)}$
4: Increment the iteration index as $\iota = \iota + 1$ and go back to Step 2
5: Terminate if $\|\boldsymbol{\beta}_{\rm s}{}^{(\iota)} - \boldsymbol{\beta}_{\rm s}{}^{(\iota-1)}\| \leq \epsilon_1$, $|R_{\rm sym}^{\rm CoD}[\iota] - R_{\rm sym}^{\rm CoD}[\iota-1]| \leq \epsilon_2$

2) Integer Coefficients Optimization: Proceeding similarly as above, the problem of finding the integer vectors $\mathbf{k}$ and $\mathbf{t}$ for a fixed choice of the preprocessing vector $\boldsymbol{\beta}_{\rm s}$ can be written as

$$\min_{\mathbf{k},\,\mathbf{t},\,\Theta_1} \quad \Theta_1 \quad (64a)$$
$$\text{s. t.} \quad \Theta_1 \geq \mathbf{t}^T \boldsymbol{\Omega} \mathbf{t} \quad (64b)$$
$$\Theta_1 \geq \mathbf{k}^T \boldsymbol{\Omega} \mathbf{k} \quad (64c)$$
$$\det(\mathbf{k}, \mathbf{t}) \neq 0 \quad (64d)$$
$$\mathbf{k},\, \mathbf{t} \in \mathbb{Z}^2, \;\; \Theta_1 \in \mathbb{R}, \quad (64e)$$

where $\boldsymbol{\Omega} = (\mathbf{G}^T(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d)^{-1}\mathbf{G} - \mathbf{I}_2)^T(\mathbf{G}^T(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d)^{-1}\mathbf{G} - \mathbf{I}_2) + ((\mathbf{G}\mathbf{G}^T + \mathbf{N}_d)^{-1}\mathbf{G})^T\mathbf{N}_d((\mathbf{G}\mathbf{G}^T + \mathbf{N}_d)^{-1}\mathbf{G})$.

We reformulate the problem (64) into a MIQP problem with quadratic constraints [24] and introduce the following quantities. Let $\mathbf{a}_0 = [0, 0, 0, 0, 1]^T$, $\mathbf{a}_1 = \mathbf{a}_2 = [0, 0, 0, 0, -1]^T$ and $\mathbf{a}_3 = [0, 0, 0, 0, 0]^T$. Also, let $\mathbf{b} = [t_a, t_b, k_a, k_b, \Theta_1]^T$; and the scalars $c_1 = c_2 = 0$, and $c_3 = -1$. We also introduce the following five-by-five matrices $\mathbf{F}_1$, $\mathbf{F}_2$, and $\mathbf{F}_3$, where

$$\mathbf{F}_1 = \begin{bmatrix} 2\boldsymbol{\Omega} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{F}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 2\boldsymbol{\Omega} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 0 \end{bmatrix},$$

$$\text{and} \qquad \mathbf{F}_3 = \begin{bmatrix} \mathbf{0} & 0 & -2 & 0 \\ \mathbf{0} & 2 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}. \quad (65)$$

The optimization problem (64) can now be reformulated equivalently as

$$\min_{\mathbf{b}} \quad \mathbf{a}_0^T \mathbf{b}$$
$$\text{s. t.} \quad \frac{1}{2}\mathbf{b}^T \mathbf{F}_i \mathbf{b} + \mathbf{a}_i^T \mathbf{b} \leq c_i \quad i = 1, 2, 3$$
$$\mathbf{k} \in \mathbb{Z}^2,\; \mathbf{t} \in \mathbb{Z}^2,\;\; \Theta_1 \in \mathbb{R} \quad (66)$$

It is easy to see that the matrices $\mathbf{F}_1$ and $\mathbf{F}_2$ are positive semi-definite. However, the matrix $\mathbf{F}_3$ is indefinite, irrespective to the values of $\mathbf{k}$ and $\mathbf{t}$.

To solve the optimization problem (66), we transform it into one that is MIQP-compatible. We find the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ in a similar way as described in Section IV-A2. We should note that the optimization problem always gives identical value for the vectors $\mathbf{k}$ and $\mathbf{t}$ if we do not consider the constraint (64d), since (64b) and (64c) have identical $\boldsymbol{\Omega}$. Therefore, "Algorithm B-1", given below, is slightly different than "Algorithm A-1".

Algorithm B-1 Integer coefficients selection for $R_{\text{sym}}^{\text{CoD}}$ as given by (62)

1: Use the branch-and-bound algorithm to solve for Θ_1, $\mathbf{k}$ and $\mathbf{t}$ without considering the constraint (64d). Denote the found solution of $\mathbf{k}$ as $\boldsymbol{\kappa}_0$
2: Use the branch-and-bound algorithm to solve for Θ_1, $\mathbf{t}$ with the constraint (64d) substituted with $\det(\mathbf{t}, \boldsymbol{\kappa}_0) > 0$. Denote the found solution of Θ_1 as $\Theta_1^{\min,1}$
3: Again, use the branch-and-bound algorithm to solve Θ_1, $\mathbf{t}$ with the constraint (64d) substituted with $-\det(\mathbf{t}, \boldsymbol{\kappa}_0) > 0$. Denote the found solution of Θ_1 as $\Theta_1^{\min,2}$
4: Select the integer coefficients corresponding to the minimum among $\Theta_1^{\min,1}$ and $\Theta_1^{\min,2}$

3) Power Allocation Policy: The problem of optimizing the power value $\boldsymbol{\beta}_{\text{s}}$ for a fixed integer coefficients $\mathbf{k}$, and $\mathbf{t}$, can be written as,

$$\min_{\boldsymbol{\beta}_{\text{s}},\, \boldsymbol{\alpha}_t,\, \boldsymbol{\alpha}_k,\, \Theta_2} \Theta_2 \tag{67a}$$

$$\text{s. t.} \quad \Theta_2 \geq \frac{\text{snr}||\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}||^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d}{\text{snr}}, \tag{67b}$$

$$\Theta_2 \geq \frac{\text{snr}||\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}||^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d}{\text{snr}}, \tag{67c}$$

$$D \geq \frac{N^2 \left(1 + \text{snr}\|\boldsymbol{\beta}_{\boldsymbol{s}} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2 \left(\text{snr}(\boldsymbol{\beta}_{\boldsymbol{s}} \circ \mathbf{h}_r)^T (\boldsymbol{\beta}_{\boldsymbol{s}} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta}_{\boldsymbol{s}} \circ \mathbf{h}_d\|^2\right)} \tag{67d}$$

$$-\sqrt{\frac{P_i}{P}} \leq \beta_i \leq \sqrt{\frac{P_i}{P}}, \ \ i = a, b \tag{67e}$$

$$\boldsymbol{\beta}_s \in \mathbb{R}^2, \ \Theta_2 \in \mathbb{R}. \tag{67f}$$

As before, let $\mathbf{c} = [c_a, c_b]^T \in \mathbb{R}^2$ and $\boldsymbol{\delta}_s = [\delta_a, \delta_b]^T \in \mathbb{R}^2$, such that $c_i > \sqrt{P_i/P}$ and $\delta_i = \beta_i + c_i$ for $i = a, b$. We can reformulate the optimization problem as,

$$\min_{\boldsymbol{\delta}_{\boldsymbol{s}},\, \boldsymbol{\alpha}_t,\, \boldsymbol{\alpha}_k,\, \Theta_2} \Theta_2 \tag{68a}$$

$$\text{s. t.} \quad \frac{f_1(\boldsymbol{\delta}_{\boldsymbol{s}}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)}{g_1(\boldsymbol{\delta}_{\boldsymbol{s}}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)} \leq 1, \tag{68b}$$

$$\frac{f_2(\boldsymbol{\delta}_{\boldsymbol{s}}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)}{g_2(\boldsymbol{\delta}_{\boldsymbol{s}}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)} \leq 1, \tag{68c}$$

$$\frac{f_3(\boldsymbol{\delta}_{\boldsymbol{s}}, \Theta_2)}{g_3(\boldsymbol{\delta}_{\boldsymbol{s}}, \Theta_2)} \leq 1 \tag{68d}$$

$$-\sqrt{\frac{P_i}{P}} + c_i \leq \delta_i \leq \sqrt{\frac{P_i}{P}} + c_i \ \ , \, i = a, b \tag{68e}$$

$$\boldsymbol{\delta}_s \in \mathbb{R}^2, \ \mathbf{c} \in \mathbb{R}^2, \ \Theta_2 \in \mathbb{R}. \tag{68f}$$

The constraints (68b), (68c), and (68d) correspond to the constraints (67b), (67c), and (67d), respectively. These functions consist of ratios of posynomials, i.e., are not posynomial. As before, we transform the complementary GP problem into a series of GPs using convex approximations.

To get the solutions of $\boldsymbol{\delta}_{\boldsymbol{s}}$, $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$ that minimize (68), the optimization problem is carried out in two steps. First, for a fixed value of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$, the algorithm computes the vector $\boldsymbol{\delta}_{\boldsymbol{s}}$ that corresponds to a minimum of (68). We solve this subproblem through a geometric programming and successive convex approximation in a similar way as described in Section IV-A3. Next, for the found $\boldsymbol{\delta}_{\boldsymbol{s}}$, the algorithm computes $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$ that correspond to a minimum of (68) using (47) and (49). This process is repeated until convergence. We should note that the algorithm of finding $\boldsymbol{\delta}_{\boldsymbol{s}}$ for a fixed value of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$ is provably convergent [17] since all the three conditions for convergence explained in Section IV-A3 are satisfied. Moreover, since "Algorithm B-2" is based on coordinate descent method, and the objective function (68) decreases as the iterations continue, the convergence for "Algorithm B-2" is guaranteed. The problem of finding the appropriate preprocessing vector $\boldsymbol{\delta}_s$, $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$ for given integer vectors $\mathbf{k}$ and $\mathbf{t}$ can be solved using "Algorithm B-2".

Algorithm B-2 Power allocation policy for $R_{\text{sym}}^{\text{CoD}}$ as given by (62)

1: Set $\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(0,0)}$ to some initial value and set $\iota_2 = 1$ and $\iota_3 = 0$
2: Compute $\Theta_2^{(\iota_2-1,\iota_3)}$, $\boldsymbol{\alpha}_t^{(\iota_2-1,\iota_3)}$ and $\boldsymbol{\alpha}_k^{(\iota_2-1,\iota_3)}$ using $\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2-1,\iota_3)}$
3: Approximate $g(\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2,\iota_3)}, \Theta_2^{(\iota_2,\iota_3)})$ with $\tilde{g}(\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2,\iota_3)}, \Theta_2^{(\iota_2,\iota_3)})$ around $\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2-1,\iota_3)}$ and $\Theta_2^{(\iota_2-1,\iota_3)}$ using (61)
4: Solve the resulting approximated GP problem using an interior point approach. Denote the found solutions as $\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2,\iota_3)}$ and $\Theta_2^{(\iota_2,\iota_3)}$
5: Increment the iteration index as $\iota_2 = \iota_2 + 1$ and go back to Step 3 using $\boldsymbol{\delta}_{\boldsymbol{s}}$ and Θ_2 of step 4.
6: Terminate if $\|\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2,\iota_3)} - \boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2-1,\iota_3)}\| \leq \epsilon_1$ and denote by $\boldsymbol{\delta}$ the final value
7: Increment the iteration index as $\iota_3 = \iota_3 + 1$, set $\iota_2 = 1$, and $\boldsymbol{\delta}_{\boldsymbol{s}}{}^{(\iota_2-1,\iota_3)} = \boldsymbol{\delta}$ and then go back to Step 2
8: Terminate if $|R_{\text{sym}}^{\text{CoD}}[\iota_3] - R_{\text{sym}}^{\text{CoD}}[\iota_3 - 1]| \leq \epsilon_2$

V. Numerical Examples

In this section, we provide some numerical examples. We measure the performance of the coding strategies using symmetric-rate. We compare our coding strategies with those described in Section II-C.

Throughout this section, we assume that the channel coefficients are modeled with independent and randomly generated variables, each generated according to a zero-mean Gaussian distribution whose variance is chosen according to the strength of the corresponding link. More specifically, the channel coefficient associated with the link from Transmitter A to the relay is modeled with a zero-mean Gaussian distribution with variance σ_{ar}^2 and to the destination is modeled with a zero-mean Gaussian distribution with variance σ_{ad}^2. Similar assumptions and notations are used for Transmitter B and the relay. Furthermore, we assume that, at every time instant, all the nodes know, or can estimate with high accuracy, the values taken by the channel coefficients at that time, i.e., full channel state information (CSI). Also, we set $P_a = 20$ dBW, $P_b = 20$ dBW, $P_r = 20$ dBW and $P = 20$ dBW.

Figure 2 depicts the evolution of the symmetric-rate obtained using the so-called *compute-and-forward at the relay* approach, i.e., $R_{\text{sym}}^{\text{CoF}}$ of proposition 1 as given by (22); and the symmetric-rate obtained using the so-called *compress-and-forward at the relay and compute at the destination* approach, i.e., $R_{\text{sym}}^{\text{CoD}}$ of proposition 2 as given by (27), as functions of the signal-to-noise ratio SNR (in decibels). Note that the

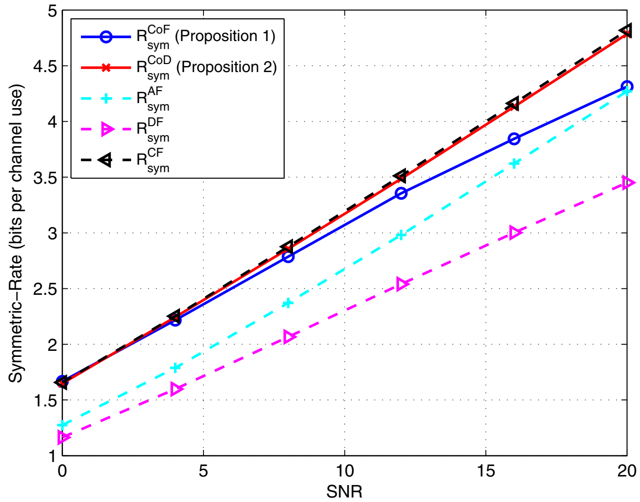

Fig. 2. Achievable symmetric rates. Numerical values are P = 20 dBW, $\sigma_{\rm ar}^2$ = 26 dB, $\sigma_{\rm br}^2$ = 26 dB, $\sigma_{\rm rd}^2$ = 18 dB, $\sigma_{\rm ad}^2$ = 14 dB and $\sigma_{\rm bd}^2$ = 0 dB.

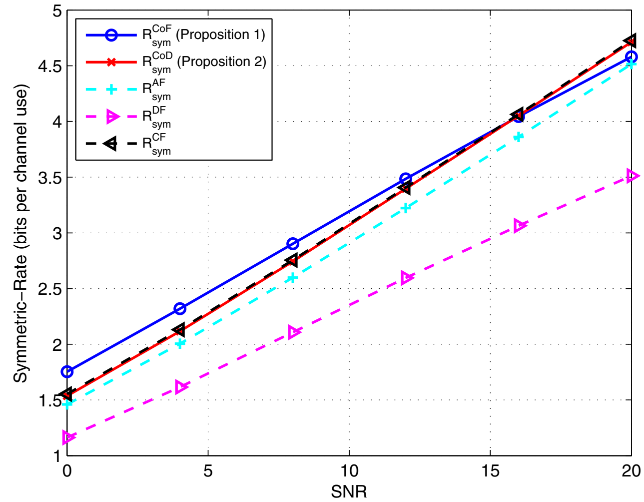

Fig. 4. Achievable symmetric rates. Numerical values are P = 20 dBW, $\sigma_{\rm ar}^2$ = 30 dB, $\sigma_{\rm br}^2$ = 18 dB, $\sigma_{\rm rd}^2$ = 15 dB, $\sigma_{\rm ad}^2$ = 26 dB and $\sigma_{\rm bd}^2$ = 0 dB.

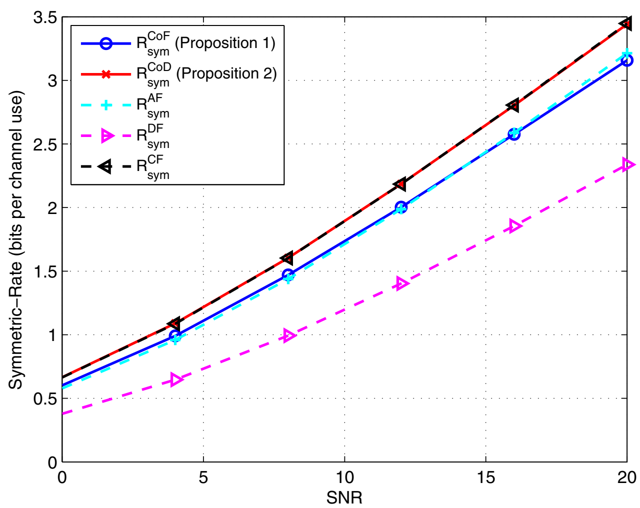

Fig. 3. Achievable symmetric rates. Numerical values are P = 20 dBW, $\sigma_{\rm ar}^2$ = 26 dB, $\sigma_{\rm br}^2$ = 0 dB, $\sigma_{\rm rd}^2$ = 26 dB, $\sigma_{\rm ad}^2$ = 26 dB and $\sigma_{\rm bd}^2$ = 0 dB.

curves correspond to numerical values of channel coefficients chosen such that $\sigma_{\rm ar}^2$ = 26 dB, $\sigma_{\rm br}^2$ = 26 dB, $\sigma_{\rm rd}^2$ = 18 dB, $\sigma_{\rm ad}^2$ = 14 dB and $\sigma_{\rm bd}^2$ = 0 dB. For comparison reasons, the figure also shows the symmetric rates obtained using the classical strategies of Section II-C, i.e., the symmetric-rate $R_{\rm sym}^{\rm AF}$ allowed by standard amplify-and-forward as given by (14), the symmetric-rate $R_{\rm sym}^{\rm DF}$ allowed by standard decode-and-forward as given by (16), and the symmetric-rate $R_{\rm sym}^{\rm CF}$ allowed by standard compress-and-forward as given by (21).

For the example shown in Figure 2, we observe that the strategy of proposition 2 achieves a symmetric-rate that is larger than what is obtained using standard DF and AF, and slightly less than what is obtained using standard CF (related to this aspect, recall the discussion in Remark 3). Also, we observe that the strategy of proposition 1 achieves a symmetric-rate that is larger than what is obtained using standard DF and AF.

Figure 3 depicts the same curves for other combinations of channel coefficients, chosen such that $\sigma_{\rm ar}^2$ = 26 dB, $\sigma_{\rm br}^2$ = 0 dB, $\sigma_{\rm rd}^2$ = 26 dB, $\sigma_{\rm ad}^2$ = 26 dB and $\sigma_{\rm bd}^2$ = 0 dB. These channel coefficients powers represent a situation where Transmitter A has better channel quality than Transmitter B. In this case, we observe that the strategy of proposition 2 achieves a symmetric-rate that is as good as what is obtained using standard CF. Also, note that, the strategy of proposition 1 provides a symmetric-rate that is slightly less than what is obtained using standard AF and is larger than what is obtained using standard DF.

Remark 7: Recall that the optimization "Algorithm B" associated with the strategy of proposition 2 is non-convex. In the figures shown in this paper, the symmetric rate provided by this strategy are obtained by selecting *only* certain initial points for "Algorithm B". For this reason, the symmetric-rate offered by the coding strategy of proposition 2, i.e., $R_{\rm sym}^{\rm CoD}$, can possibly be as good as the symmetric-rate offered by CF if one considers more initial points.

Remark 8: The comparison of the coding strategies of proposition 1 and proposition 2 is insightful. Generally, none of the two coding schemes outperforms the other for all ranges of SNR, and which of the two coding schemes performs better depends on both the operating SNR and the relative strength of the links. For example, observe that while the strategy of proposition 2 outperforms that of proposition 1 in the examples shown in Figures 2 and 3, the situation is reversed for the example shown in Figure 4 for some SNR ranges (related to this aspect, recall the discussion in Remark 2).

Figure 5 shows the symmetric-rate $R_{\rm sym}^{\rm CoF}$ of proposition 1 with optimum preprocessing allocation $\boldsymbol{\beta}_s^*$ and with no preprocessing allocation, i.e., $\boldsymbol{\beta}_s = \mathbf{1}$; the symmetric-rate $R_{\rm sym}^{\rm CoD}$ of proposition 2 with optimum preprocessing allocation $\boldsymbol{\beta}_s^*$ and with no preprocessing allocation, i.e., $\boldsymbol{\beta}_s = \mathbf{1}$. We observe that the strategy of proposition 1 with optimum preprocessing vector $\boldsymbol{\beta}_s^*$ offers significant improvement over the one with no preprocessing allocation, and this improvement increases with the SNR. We also observe that the strategy of proposition 2 with optimum preprocessing vector $\boldsymbol{\beta}_s^*$ offers small improve-

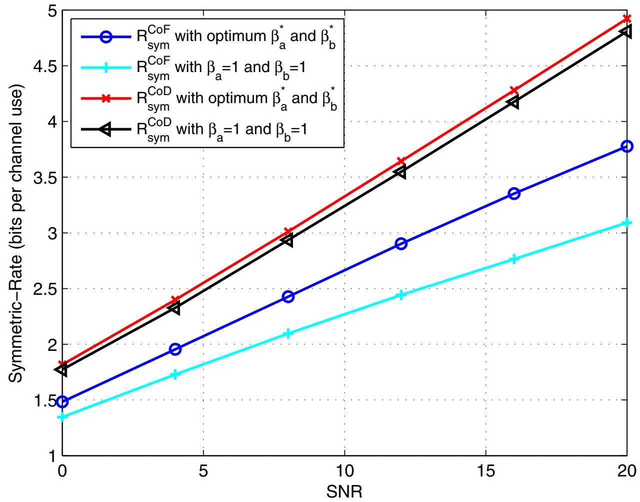

Fig. 5. Achievable symmetric rates. Numerical values are P = 20 dBW, $\sigma^2_{\rm ar}$ = 20 dB, $\sigma^2_{\rm br}$ = 20 dB, $\sigma^2_{\rm rd}$ = 20 dB, $\sigma^2_{\rm ad}$ = 14 dB and $\sigma^2_{\rm bd}$ = 14 dB.

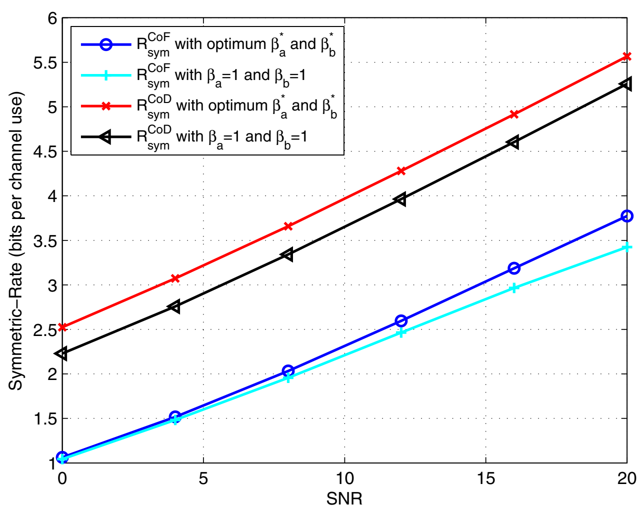

Fig. 6. Achievable symmetric rates. Numerical values are P = 20 dBW, $\sigma^2_{\rm ar}$ = 26 dB, $\sigma^2_{\rm br}$ = 26 dB, $\sigma^2_{\rm rd}$ = 0 dB, $\sigma^2_{\rm ad}$ = 26 dB and $\sigma^2_{\rm bd}$ = 26 dB.

ment over the one with no preprocessing allocation. However, with different numerical values of channel coefficients, we observe in Figure 6 that the strategy of proposition 2 with optimum preprocessing vector $\boldsymbol{\beta}_s^*$ offers significant improvement over the one with no preprocessing allocation.

We close this section with a brief discussion of the convergence speed of "Algorithm A" that we use to solve the optimization problem (A) given by (50), as described in section IV-A. Recall that the algorithm involves allocating the integer coefficients and the transmitters' powers alternately, in an iterative manner. For a given set of powers, we find the best integer coefficients by solving a MIQP problem with quadratic constraints using the optimization software MOSEK. For a given set of integer-valued coefficients, we find the best powers at the transmitters by solving a series of geometric programs by means of an interior point approach [28].

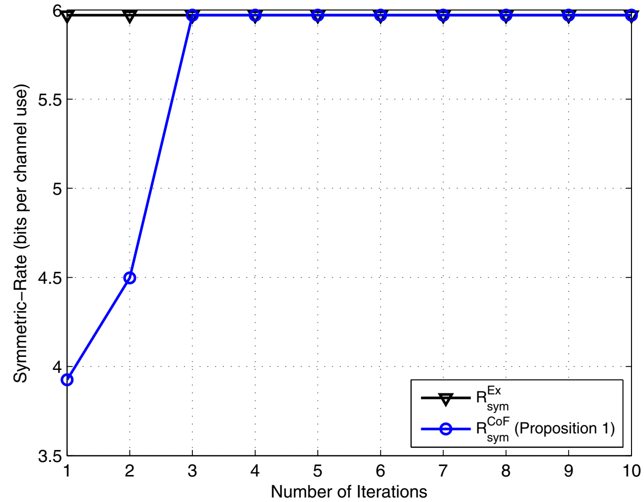

Fig. 7. Algorithm A v.s. Exhaustive search. Numerical values are P = 20 dBW, SNR= 15dB, $\sigma^2_{\rm ar}$ = $\sigma^2_{\rm br}$ = $\sigma^2_{\rm rd}$ = 20 dB, and $\sigma^2_{\rm ad}$ = $\sigma^2_{\rm bd}$ = 0 dB.

In order to investigate the convergence speed of the proposed algorithm, we compare it with one in which the integer coefficients search is performed in an exhaustive manner and the power allocation is kept as in Section IV-A3. Note that, using this exhaustive-search algorithm, for the integer valued equations coefficients to be chosen optimally, the search can be restricted to the set of integer values that satisfy $||\mathbf{k}||^2 \leq 1 + ||\mathbf{h}_r||^2\text{snr}$ and $||\mathbf{t}||^2 \leq 1 + ||\mathbf{h}_d||^2\text{snr}$, since otherwise the allowed symmetric rate is zero [30]. Let $R^{\rm Ex}_{\rm sym}$ denote the symmetric rate obtained by using the described exhaustive search-based algorithm. Figure 7 shows that the number of iterations required for "Algorithm A" to converge, i.e., yield the same symmetric-rate as the one obtained through exhaustive search, is no more than three. Also, we note that, in comparison, the exhaustive search-based algorithm is more largely time and computationally resources consuming, especially at large values of SNR. Similar observations, that we omit here for brevity, also hold for "Algorithm B".

Also, we discuss the complexity of "Algorithm A-2" and "Algorithm B-2". GP is typically solved by interior-point methods which have provably polynomial time complexity [31] and are very fast in practice. However, the complexity for complementary GP (series of GP) can only be calculated numerically. For a given channel realization and integer vectors $\mathbf{k}$ and $\mathbf{t}$, we compare the maximized symmetric-rate $R^{\rm CoF}_{\rm sym}$ achieved by "Algorithm A-2" over 360 different initial vectors $\boldsymbol{\beta}_s^{(0)}$. We notice that "Algorithm A-2" converges to the same optima over the entire set of initial vectors and achieves (or comes very close to) the optimal solution given by an exhaustive search over $\boldsymbol{\beta}_s$. The average number of GP iterations required is 15 if an extremely tight exit condition is picked for the algorithm $\epsilon_1 = 10^{-10}$. Similarly, the average number of iterations required for "Algorithm B-2" is 40 if an extremely tight exit condition is picked for the algorithm $\epsilon_1 = 10^{-10}$ and $\epsilon_2 = 10^{-10}$.

VI. Conclusion

In this paper, we study a two-user half-duplex multiaccess relay channel. Based on Nazer-Gastpar compute-and-forward scheme, we develop and evaluate the performance of coding strategies that are of network coding spirit. In this framework, the destination does not decode the information messages directly from its output, but uses the latter to first recover two linearly independent integer-valued combinations that relate the transmitted symbols. We establish two coding schemes. In the first coding scheme, the two required linear combinations are computed in a distributive manner: one equation is computed at the relay and then forwarded to the destination, and the other is computed directly at the destination using the direct transmissions from the users. In the second coding scheme, the two required linear combinations are both computed locally at the destination, in a joint manner. In this coding scheme, accounting for the side information available at the destination through the direct links, the relay compresses what it gets from the users using lattice-based Wyner-Ziv compression and conveys it to the destination. The destination then computes the desired two linear combinations, locally, using the recovered output at the relay, and what it gets from the direct transmission from the users. For both coding schemes, we discuss the design criteria and establish the associated computation rates and the allowed symmetric rate. Next, for each of the two coding schemes, we investigate the problem of allocating the powers and the integer-valued coefficients of the recovered equations in a way to maximize the offered symmetric rate. This problem is NP hard; and in this paper we propose an iterative solution to solve this problem, through a careful formulation and analysis. For a given set of powers, we transform the problem of finding the best integer coefficients into a mixed-integer quadratic programming problem with quadratic constraints. Also, for a given set of integer-valued coefficients, we transform the problem of finding the best powers at the transmitters into series of geometric programs. Comparing our coding schemes with classic relaying techniques, we show that for certain channel conditions the first scheme outperforms standard relaying techniques; and the second scheme, while relying on feasible structured lattice codes, can offer rates that are as large as those offered by regular compress-and-forward for the multiaccess relay network that we study.

Acknowledgment

The authors would like to thank Prof. Stephen Boyd and his team, from Information Systems Laboratory, Stanford University, USA, and MOSEK Aps for providing the software and programs used in the optimization part in this paper. Also, the authors would like to thank Dr. Andres Kwasinski and the anonymous reviewers for their valuable suggestions to improve the quality of this work.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, pp. 1204–1216, July 2000.

[2] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," *Can. J. Math.*, vol. 8, pp. 399–404, 1956.

[3] P. Elias, D. R. Feinstein, and C. E. Shannon, "A note on the maximal flow through a network," *IRE Trans. Inf. Theory*, vol. IT-72, pp. 117–119, Dec. 1956.

[4] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inf. Theory*, vol. IT-49, pp. 371–381, Feb. 2003.

[5] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Trans. Netw.*, vol. 11, pp. 782–795, Oct. 2003.

[6] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang, "Network coding theory," *Found. Trends Commun. Inf. Theory*, vol. 2, pp. 241–381, 2005.

[7] C. Fragouli and E. Soljanin, *Network Coding Fundamentals*. Found. Trends Commun. Info. Theory, 2007.

[8] A. Avestimeher, S. Diggavi, and D. Tse, "Wireless network information flow: a determenistic approach," *IEEE Trans. Inf. Theory*, vol. 57, pp. 1872–1905, Apr. 2011.

[9] S. H. Lim, Y.-H. Kim, A. E. Gamal, and S.-Y. Chung, "Noisy network coding," *IEEE Trans. Inf. Theory*, vol. 57, pp. 3132–3152, May 2011.

[10] J. H. Conway and N. J. A. Sloane, *Sphere Packing, Lattices and Groups*, 3rd ed. John Wiley & Sons Inc., 1988.

[11] K. Narayanan, M. P. Wilson, and A. Sprinston, "Joint physical layer coding and network coding for bidirectioal relaying," in *Proc. 2007 Allerton Conf. on Comm., Control, and Computing*.

[12] S. J. Kim, P. Mitran, and V. Tarokh, "Performance bounds for bidirectional coded cooperation protocols," *IEEE Trans. Inf. Theory*, vol. IT-54, pp. 5253–5241, Nov. 2008.

[13] B. Nazer and M. Gastpar, "Compute-and-forward: harnessing interference through structured codes," *IEEE Trans. Inf. Theory*, vol. 57, pp. 6463–6486, 2011.

[14] B. Nazer, A. Sanderovich, M. Gastpar, and S. Shamai, "Structured superposition for backhaul constrained cellular uplink," in *Proc. 2009 IEEE International Symposium on Information Theory*, pp. 1530–1534.

[15] R. Zamir, S. Shamai, , and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. Inf. Theory*, vol. 48, pp. 1250–1276, June 2002.

[16] Y. Song and N. Devroye, "Lattice codes for the Gaussian relay channel: decode-and-forward and compress-and-forward," *IEEE Trans. Inf. Theory*, 2011, submitted for publication. Available: http://arxiv.org/abs/1111.0084

[17] M. Chiang, C. W. Tan, D. P. Palomar, D. O'Neill, and D. Julian, "Power control by geometric programming," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 2640–2651, July 2007.

[18] L. Sankaranarayanan, G. Kramer, and N. B. Mandayam, "Hierarchical sensor networks: capacity theorems and cooperative strategies using the multiple-access relay channel model," in *Proc. 2004 IEEE Conference on Sensor and Ad Hoc Communications and Networks*, pp. 191–199.

[19] L. Sankar, Y. Liang, V. H. Poor, and N. B. Mandayam, "Opportunistic communications in an orthogonal multiaccess relay channel," in *Proc. 2007 IEEE International Symposium on Information Theory*, pp. 1261–1265.

[20] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, pp. 1–10, 1976.

[21] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Trans. Inf. Theory*, vol. 51, pp. 3037–3063, Sept. 2005.

[22] U. Erez and R. Zamir, "Achieving $\frac{1}{2}\log(1+\mathrm{SNR})$ on the AWGN channel with lattice encoding and decoding," *IEEE Trans. Inf. Theory*, vol. 50, pp. 2293–2314, Oct. 2004.

[23] J. Zhan, B. Nazer, U. Erez, and M. Gastpar, "Integer-forcing linear receivers," *IEEE Trans. Inf. Theory*, 2010, submitted for publication. Available: http://arxiv.org/abs/1003.5966

[24] C. A. Floudas, *Nonlinear and Mixed Integer Optimization*. Oxford University Press, 1995.

[25] E. L. Lawler and D. Wood, "Branch-and-bound methods: a survey," *Operations Research*, pp. 699–719, 1966.

[26] L. A. Wolsey, *Integer Programming*. John Wiley and Sons, 1998.

[27] C. Feng, D. Silva, and F. R. Kschischang, "Algebraic approach to physical-layer network coding," *IEEE Trans. Inf. Theory*, 2011, submitted for publication. Available: http://arxiv.org/abs/1108.1695

[28] M. Chiang, *Geometric Programming for Communication Systems*. Foundations and Trends of Communications and Information Theory, 2005.

[29] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Operations Research*, vol. 26, pp. 681–683, 1978.

[30] J. Zhan, B. Nazer, M. Gastpar, and U. Erez, "MIMO compute-and-forward," in *Proc. 2009 IEEE Int. Symposium on Inf. Theory*, pp. 2848–2852.

[31] Y. Nesterov and A. Nemirovsky, *Interior Point Polynomial Algorithms in Convex Programming*. SIAM Press, 1994.

Mohieddine El Soussi received the B.E. degree in Communication and Electronics Engineering from Beirut Arab University, Beirut, Lebanon, in 2004 and the M.Sc. degree in Communication Engineering from Technische Universität München, Munich, Germany, in 2007.

From 2008 to 2009, he was a Research Assistant with the ICTEAM institute, Université catholique de Louvain (UCL), Louvain-la-Neuve, Belgium. Since 2009, he has been working toward the Ph.D. degree with the ICTEAM institute, Université catholique de Louvain (UCL), Louvain-la-Neuve, Belgium. His research interests include relaying and cooperation, network coding, and optimization.

Abdellatif Zaidi received the B.S. degree in Electrical Engineering from École Nationale Supérieure de Techniques Avancés, ENSTA ParisTech, France in 2002 and the M. Sc. and Ph.D. degrees in Electrical Engineering from École Nationale Supérieure des Télécommunications, TELECOM ParisTech, Paris, France in 2002 and 2005, respectively.

From December 2002 to December 2005, he was with the Communications and Electronics Dept., TELECOM ParisTech, Paris, France and the Signals and Systems Lab., CNRS/Supélec, France pursuing his PhD degree. From May 2006 to September 2010, he was at École Polytechnique de Louvain, Université catholique de Louvain, Belgium, working as a reaserch assistant. Dr. Zaidi was "Research Visitor" at the University of Notre Dame, Indiana, USA, during fall 2007 and Spring 2008. He is now, an assistant professor at Universite Paris-Est Marne-La-Vallee, France. He is a member of Laboratoire d'Informatique Gaspard Monge (LIGM).

Dr. Zaidi's research interests cover a broad range of topics from network information theory and signal processing for communication. Of particular interest are the problems of multi-terminal information theory, Shannon theory, relaying and cooperation, network coding, physical layer security, source coding and interference mitigation in multi-user channels. Dr. Zaidi currently serves as an editor for the *EURASIP Journal on Wireless Communications and Networking* (JWCN).

Luc Vandendorpe (F'06) was born in Mouscron, Belgium in 1962. He received the Electrical Engineering degree (*summa cum laude*) and the Ph.D. degree from the Université catholique de Louvain (UCL) Louvain-la Neuve, Belgium in 1985 and 1991 respectively. Since 1985, he is with the Communications and Remote Sensing Laboratory of UCL where he first worked in the field of bit rate reduction techniques for video coding. In 1992, he was a Visiting Scientist and Research Fellow at the Telecommunications and Traffic Control Systems Group of the Delft Technical University, The Netherlands, where he worked on Spread Spectrum Techniques for Personal Communications Systems. From October 1992 to August 1997, L. Vandendorpe was Senior Research Associate of the Belgian NSF at UCL, and invited assistant professor. Presently he is Professor and head of the Institute for Information and Communication Technologies, Electronics and Applied Mathematics. His current interest is in digital communication systems and more precisely resource allocation for OFDM(A) based multicell systems, MIMO and distributed MIMO, sensor networks, turbo-based communications systems, physical layer security and UWB based positioning. In 1990, he was co-recipient of the Biennal Alcatel-Bell Award from the Belgian NSF for a contribution in the field of image coding. In 2000 he was co-recipient (with J. Louveaux and F. Deryck) of the Biennal Siemens Award from the Belgian NSF for a contribution about filter bank based multicarrier transmission. In 2004 he was co-winner (with J. Czyz) of the Face Authentication Competition, FAC 2004. L. Vandendorpe is or has been TPC member for numerous IEEE conferences (VTC Fall, Globecom Communications Theory Symposium, SPAWC, ICC) and for the Turbo Symposium. He was co-technical chair (with P. Duhamel) for IEEE ICASSP 2006. He was an editor of the IEEE TRANSACTIONS ON COMMUNICATIONS for Synchronisation and Equalization between 2000 and 2002, associate editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS between 2003 and 2005, and associate editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING between 2004 and 2006. He was chair of the IEEE Benelux joint chapter on Communications and Vehicular Technology between 1999 and 2003. He was an elected member of the Signal Processing for Communications committee between 2000 and 2005, and an elected member of the Sensor Array and Multichannel Signal Processing committee of the Signal Processing Society between 2006 and 2008. He was an elected member of the Signal Processing for Communications committee between 2000 and 2005, and between 2009 and 2011, and an elected member of the Sensor Array and Multichannel Signal Processing committee of the Signal Processing Society between 2006 and 2008. He is the Editor in Chief for the *EURASIP Journal on Wireless Communications and Networking* and a Fellow of the IEEE.